



Article

# A Low-Complexity Algorithm for a Reinforcement Learning-Based Channel Estimator for MIMO Systems

Tae-Kyoung Kim <sup>1</sup> and Moonsik Min <sup>2,3,\*</sup><sup>1</sup> Department of Electronic Engineering, Gachon University, Seongnam 13120, Korea; tk415kim@gmail.com<sup>2</sup> School of Electronics Engineering, Kyungpook National University, Daegu 41566, Korea<sup>3</sup> School of Electronic and Electrical Engineering, Kyungpook National University, Daegu 41566, Korea

\* Correspondence: mmin@knu.ac.kr

**Abstract:** This paper proposes a low-complexity algorithm for a reinforcement learning-based channel estimator for multiple-input multiple-output systems. The proposed channel estimator utilizes detected symbols to reduce the channel estimation error. However, the detected data symbols may include errors at the receiver owing to the characteristics of the wireless channels. Thus, the detected data symbols are selectively used as additional pilot symbols. To this end, a Markov decision process (MDP) problem is defined to optimize the selection of the detected data symbols. Subsequently, a reinforcement learning algorithm is developed to solve the MDP problem with computational efficiency. The developed algorithm derives the optimal policy in a closed form by introducing backup samples and data subblocks, to reduce latency and complexity. Simulations are conducted, and the results show that the proposed channel estimator significantly reduces the minimum-mean square error of the channel estimates, thus improving the block error rate compared to the conventional channel estimation.

**Keywords:** multiple-input multiple-output; channel estimation; Markov decision process; reinforcement learning



**Citation:** Kim, T.-K.; Min, M. A Low-Complexity Algorithm for a Reinforcement Learning-Based Channel Estimator for MIMO Systems. *Sensors* **2022**, *22*, 4379. <https://doi.org/10.3390/s22124379>

Academic Editor: Davy P. Gaillot

Received: 20 May 2022

Accepted: 7 June 2022

Published: 9 June 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Currently, multiple-input multiple-output (MIMO) is an essential technology in wireless communications [1–6]. Multiple antennas are easy to implement in wireless systems, and their use significantly increases system reliability and capacity. However, to utilize the advantages of multiple antennas, perfect channel information is required at both the transmitter and receiver. Meeting this necessity is generally impossible because of the characteristics of wireless channels.

Although perfect channel information is unavailable, many studies have been conducted to improve the accuracy of channel estimation [7–21]. These investigations were mostly based on the use of pilots whose information is shared by both the transmitter and receiver and employed least-squares and linear minimum-mean square-error (LMMSE) estimations [10–12]. This is because the two estimation methods reasonably perform with affordable complexities for wireless systems. However, their performance strongly depends on the number of pilots, which is generally limited in wireless systems because employing several pilots as resources degrades the spectral efficiency.

This limitation can be overcome using data in channel estimation, i.e., conducting data-aided channel estimation [13–21]. Its concept is to exploit a detected data symbol as an additional pilot. Because a detected data symbol may have an error, the accuracy of the channel estimation may be degraded by it. An iterative turbo approach is a good method to address this degradation because the improved detection performance achieved using an iterative turbo equalizer also increases the estimation accuracy of a channel [19–25]. However, the use of this iterative turbo approach is limited in wireless systems because of its inherent high complexity and latency.

Recently, a reinforcement learning (RL) approach was introduced in [26] for data-aided channel estimation. In this approach, a Markov decision process (MDP) problem is described to minimize the estimation error, and an RL algorithm is used to solve the MDP problem. Without an iterative approach, the RL solution resulted in a significant improvement compared to conventional channel estimations. However, this solution is difficult to implement in practical systems because of its considerable complexity and latency in computing the optimal policy. For example, using the approach in [26] to calculate the optimal policy requires all a posteriori probabilities (APPs) in a data block. In addition, its limitation is that the optimal policy is characterized by a specific discounting factor.

In this paper, a low-complexity channel estimator using an RL approach is proposed for MIMO systems. The key concept of this estimator is the selection of the detected data symbols obtained during data detection as additional pilot symbols. To achieve this, an MDP problem is first defined to minimize the channel estimation error where the Q-value function is generalized by a discounting factor. Subsequently, an RL solution is proposed that can be easily implement in wireless systems. To this end, concepts of backup samples and data subblocks are introduced, which significantly reduce the complexity and latency. The main contributions of this study are summarized as follows:

- A data-aided channel estimator is developed to optimize the selection of detected symbols for MIMO systems. An MDP problem is defined for this selection to minimize the mean-square-error (MSE) of the channel estimates. Compared with [26], a discounting factor is introduced in the Q-value function. The discounting factor adjusts the effects of rewards after the current state.
- A low-complexity RL algorithm is proposed. To achieve this efficiently, a data block is separated into multiple data subblocks and the optimal policy for the data subblocks is characterized. In the characterization, only partial soft information obtained from data detection is utilized to reduce the calculation latency. Unlike in [26], the optimal policy is calculated using only this partially obtained information; the remaining rewards are approximated under the assumption of perfect detection. Finally, the optimal policy is obtained using a closed-form expression. Note that the conventional RL algorithm in [26] can be employed after obtaining all soft information in a data block.
- The performance enhancement achieved for MIMO systems using the developed RL algorithm is evaluated. Simulations are conducted, and the results demonstrate that the proposed algorithm significantly reduces the performance degradation of conventional channel estimation. Based on the simulations, the proposed channel estimator using an approximate MDP presents a similar performance to that of the original MDP. In addition, the proposed channel estimator provides robustness in time-varying channels.

The remainder of this paper is organized as follows. Section 2 introduces a signal model including the channel estimation and data detection considered in this study. In Section 3, an MDP problem to select detected data symbols optimally to minimize the channel estimation error is defined. A low-complexity RL algorithm is proposed in Section 4. In Section 5, simulation results are discussed, to demonstrate the effectiveness of the developed algorithm. Finally, conclusions are presented in Section 6.

### Notation

Matrices  $\mathbf{0}_m$  and  $\mathbf{I}_m$  represent  $m \times m$  all-zero and the  $m \times m$  identity matrices, respectively. The superscripts  $(\cdot)^T$  and  $(\cdot)^H$  denote the transpose and the conjugate transpose, respectively. Operators  $\mathbb{E}(\cdot)$  and  $\mathbb{P}(\cdot)$  denote the expectation of a random variable and the probability of an event, respectively. Operators  $|\cdot|$  and  $\|\cdot\|^2$  denote the cardinality of a set and the norm, respectively. Operators  $(\cdot)^{-1}$ ,  $\text{Tr}(\cdot)$ , and  $\mathcal{CN}$  denote the inverse, trace, and complex normal distribution, respectively. Set  $\mathbb{C}$  represents a set of complex numbers.

## 2. Signal Model

This section describes the signal model for a MIMO system. Based on the signal model, the channel estimator and data detector considered in this study are introduced.

### 2.1. Signal Model

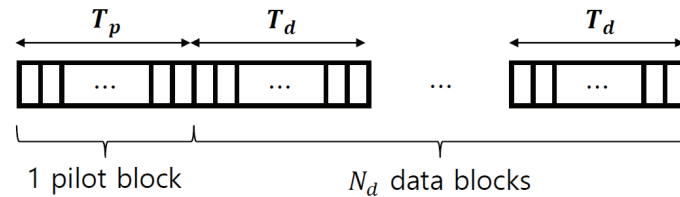
A MIMO system is considered; in it, a transmitter with  $N_t$  antennas communicates with a receiver with  $N_r$  antennas through a wireless channel. A wireless channel is denoted as  $\mathbf{H} \in \mathbb{C}^{N_t \times N_r}$ , where each channel element  $h_{t,r} \in \mathbb{C}$  between the  $t$ -th transmitter and  $r$ -th receiver is modeled by Rayleigh fading  $h_{t,r} \sim \mathcal{CN}(0, 1)$ . The transmitter sends a frame consisting of one pilot block and  $N_d$  data blocks, as shown in Figure 1. During the pilot transmission, the transmitter sends a pilot symbol  $\mathbf{x}^p[n] \in \mathbb{C}^{N_t \times 1}$  for  $n \in \mathcal{N}_p = \{1, \dots, T_p\}$ , where  $T_p$  is the pilot length. When the pilot symbol  $\mathbf{x}^p[n]$  is transmitted to the receiver, the received symbol  $\mathbf{y}^p[n] \in \mathbb{C}^{N_r \times 1}$  at time slot  $n$  is given as

$$\mathbf{y}^p[n] = \mathbf{H}^H \mathbf{x}^p[n] + \mathbf{z}^p[n], \quad (1)$$

where  $\mathbf{z}^p[n]$  is an additive white Gaussian noise (AWGN) at time slot  $n$  whose distribution follows  $\mathcal{CN}(\mathbf{0}_{N_r}, N_0 \mathbf{I}_{N_r})$ . After the pilot transmission is completed, the transmitter sends a data symbol  $\mathbf{x}^d[n] \in \mathbb{C}^{N_t \times 1}$  for  $n \in \mathcal{N}_d = \{(d-1)T_d + 1, \dots, dT_d\}$ , where  $T_d$  is the data length. Supposing  $\mathcal{X}$  is a constellation set, the data symbol  $\mathbf{x}^d[n] \in \mathcal{X}^{N_t}$ . After the data transmission, the received symbol  $\mathbf{y}^d[n] \in \mathbb{C}^{N_r \times 1}$  is expressed as

$$\mathbf{y}^d[n] = \mathbf{H}^H \mathbf{x}^d[n] + \mathbf{z}^d[n], \quad (2)$$

where  $\mathbf{z}^d[n]$  is also an AWGN at time slot  $n$ .



**Figure 1.** Frame consisting of one pilot block with  $T_p$  symbols and  $N_d$  data blocks with  $T_d$  symbols.

### 2.2. Channel Estimator and Data Detector

The LMMSE channel estimator is considered in this study because of its satisfactory performance with low complexity. Using the received symbol in (1), the LMMSE channel estimator,  $\mathbf{W} \in \mathbb{C}^{N_t \times T_p}$ , is expressed as follows:

$$\hat{\mathbf{W}} = \underset{\mathbf{W}}{\operatorname{argmin}} \mathbb{E} \left[ \|\mathbf{W}(\mathbf{y}_r^p)^H - \mathbf{h}_r\|^2 \right] = \left( \mathbf{X}^p (\mathbf{X}^p)^H + N_0 \mathbf{I}_{N_t} \right)^{-1} \mathbf{X}^p, \quad (3)$$

where  $\mathbf{y}_r^p$  and  $\mathbf{X}^p$  are sets of the received and pilot symbols and are defined as  $\mathbf{y}_r^p = [y_r^p[1], \dots, y_r^p[T_p]]$  and  $\mathbf{X}^p = [\mathbf{x}^p[1], \dots, \mathbf{x}^p[T_p]]$ , respectively. Using the channel estimator in (3), a channel estimate is expressed as

$$\hat{\mathbf{h}}_r = \hat{\mathbf{W}} (\mathbf{y}_r^p)^H = \left( \mathbf{X}^p (\mathbf{X}^p)^H + N_0 \mathbf{I}_{N_t} \right)^{-1} \mathbf{X}^p (\mathbf{y}_r^p)^H, \quad (4)$$

where  $\hat{\mathbf{h}}_r$  is the  $r$ -th row of the channel estimate matrix  $\hat{\mathbf{H}}$ .

A maximum a posteriori probability (MAP) data detector is considered in this study to ensure the optimal detection performance. The APP from the MAP data detector is computed as

$$\theta_k[n] = \mathbb{P}[\mathbf{x}^d[n] = \mathbf{x}_k | \mathbf{y}^d[n]] = \frac{\mathbb{P}[\mathbf{y}^d[n] | \mathbf{x}^d[n] = \mathbf{x}_k] \mathbb{P}[\mathbf{x}^d[n] = \mathbf{x}_k]}{\sum_{j \in \mathcal{K}} \mathbb{P}[\mathbf{y}^d[n] | \mathbf{x}^d[n] = \mathbf{x}_j] \mathbb{P}[\mathbf{x}^d[n] = \mathbf{x}_j]}, \quad (5)$$

where  $\mathbf{x}_k \in \mathcal{X}^{N_t}$  is the  $k$ -th possible symbol for  $k \in \mathcal{K} = \{1, \dots, |\mathcal{X}^{N_t}|\}$ . In (5), the a priori probability,  $\mathbb{P}[\mathbf{x}^d[n] = \mathbf{x}_k]$ , is assumed to be equal for all possible symbols  $\mathbf{x}_k$  for  $k \in \mathcal{K}$ , i.e.,  $\mathbb{P}[\mathbf{x}^d[n] = \mathbf{x}_k] = \frac{1}{|\mathcal{X}^{N_t}|}$ . Concurrently, under the AWGN assumption, the likelihood probability  $\mathbb{P}[\mathbf{y}^d[n] | \mathbf{x}^d[n] = \mathbf{x}_k]$  in (5) can be expressed as

$$\mathbb{P}[\mathbf{y}^d[n] | \mathbf{x}^d[n] = \mathbf{x}_k] = \frac{1}{(\pi N_0)^{N_r}} e^{-\frac{\|\mathbf{y}^d[n] - \hat{\mathbf{H}} \mathbf{x}_k\|^2}{N_0}}. \quad (6)$$

The MAP data detector detects the data symbol  $\hat{\mathbf{x}}[n]$  that has the best APP value at time slot  $n$ , and it is given by

$$\hat{\mathbf{x}}[n] = \underset{\mathbf{x}_k \in \mathcal{X}^{N_t}}{\operatorname{argmax}} \theta_k[n] = \underset{\mathbf{x}_k \in \mathcal{X}^{N_t}}{\operatorname{argmax}} \mathbb{P}[\mathbf{y}^d[n] | \mathbf{x}^d[n] = \mathbf{x}_k]. \quad (7)$$

Note that the accuracy of the detected symbol  $\hat{\mathbf{x}}[n]$  depends on the accuracy of the channel estimator,  $\hat{\mathbf{H}}$ . However, the accuracy of the channel estimator cannot be ensured in practical systems where the pilot length,  $T_p$ , is limited. To address this limitation, this study focused on improving the accuracy of the channel estimator.

### 3. Optimization Problem

This section defines the optimization problem for the channel estimator proposed subsequently, which uses detected symbols to improve the MSE of the channel estimates. Subsequently, to solve the optimization problem, the MDP problem and the optimal policy are presented.

#### 3.1. Optimization Problem

This study considers a channel estimator that uses the detected symbols in (7) as additional pilot symbols. However, the data detector may generate detection errors at the receiver. Consequently, the use of detected symbols with errors degrades the accuracy of the channel estimator. To overcome this problem, the detected symbols should be selectively exploited by the channel estimator.

Let  $\mathbf{a} \in \{0, 1\}^{T_d}$  be the set of actions whose  $n$ -th component is the selection of a detected symbol of the  $d$ -th data block for  $n \in \mathcal{N}_d$ . Specifically, when  $a = 1$ , a detected symbol is used as an additional pilot symbol; otherwise, it is not used. By exploiting  $\mathbf{a}$ , the LMMSE channel estimate in (4) can be updated as

$$\hat{\mathbf{h}}_r(\mathbf{a}) = \left( \mathbf{X}(\mathbf{a}) \mathbf{X}(\mathbf{a})^H + N_0 \mathbf{I}_{N_t} \right)^{-1} \mathbf{X}(\mathbf{a}) \bar{\mathbf{y}}_r(\mathbf{a})^H, \quad (8)$$

where  $\bar{\mathbf{y}}_r(\mathbf{a}) = [\mathbf{y}_r^p, y_r^d[u_1(\mathbf{a})], \dots, y_r^d[u_{\|\mathbf{a}\|_0}(\mathbf{a})]]$  and  $\mathbf{X}(\mathbf{a}) = [\mathbf{X}^p, \hat{\mathbf{x}}[u_1(\mathbf{a})], \dots, \hat{\mathbf{x}}[u_{\|\mathbf{a}\|_0}(\mathbf{a})]]$ .

Here,  $u_i(\mathbf{a})$  is the time slot index of the  $i$ -th nonzero element in  $\mathbf{a}$ . Thus, the optimization problem that maximizes the accuracy of the proposed channel estimator can be expressed as

$$\mathbf{a}^* = \underset{\mathbf{a} \in \{0, 1\}^{T_d}}{\operatorname{argmax}} \mathbb{E}\{\|\hat{\mathbf{H}}(\mathbf{a}) - \mathbf{H}\|^2\}. \quad (9)$$

Solving the optimization problem in (9) is difficult. First, the distribution of  $\hat{\mathbf{H}}(\mathbf{a})$  requires information regarding the transmitted symbols. However, this information is generally unknown to a receiver. In addition, the number of candidates for actions  $\mathbf{a}$

exponentially increases with data length  $T_d$ . Accordingly, an exhaustive search for these actions is impractical because of the unsatisfactory complexity and latency for the receiver.

### 3.2. Markov Decision Process

To efficiently solve the problem in (9), an MDP was formulated in [26] that sequentially selected detected symbols. In this formulation, a detected symbol is selected if the updated channel estimator reduces the estimation error.

Similar to [26], for this study, the state set of the MDP at time slot  $n$  is expressed as

$$\begin{aligned} \mathcal{S}_n = \left\{ (\mathbf{X}_n, \hat{\mathbf{X}}_n, \mathcal{M}_n) \mid \mathbf{X}_n = [\mathbf{X}^p, \mathbf{x}_{k_{\mathcal{M}_n(1)}}, \dots, \mathbf{x}_{k_{\mathcal{M}_n(|\mathcal{M}_n|)}}], k_i \in \mathcal{K}, \right. \\ \hat{\mathbf{X}}_n = [\mathbf{X}^p, \hat{\mathbf{x}}[\mathcal{M}_n(1)], \dots, \hat{\mathbf{x}}[\mathcal{M}_n(|\mathcal{M}_n|)]], \\ \left. \mathcal{M}_n \subset \{T_p + 1, \dots, n - 1\} \right\}, \end{aligned} \quad (10)$$

where  $k_n$  denotes the transmitted symbol index at time slot  $n$ . Set  $\mathcal{M}_n$  represents the set of time slot indices of the data symbols to be utilized as additional pilot symbols.  $\mathcal{M}_n(i)$  is the  $i$ -th smallest element of  $\mathcal{M}_n$ . Based on the above notations, the proposed channel estimate at state  $\mathcal{S}_n = (\mathbf{X}_n, \hat{\mathbf{X}}_n, \mathcal{M}_n) \in \mathcal{S}_n$  is expressed as

$$\hat{\mathbf{h}}_r(\mathcal{S}_n) = \left( \hat{\mathbf{X}}_n \hat{\mathbf{X}}_n^H + N_0 \mathbf{I}_{N_t} \right)^{-1} \hat{\mathbf{X}}_n \bar{\mathbf{y}}_r^H(\mathcal{S}_n), \quad (11)$$

where  $\bar{\mathbf{y}}_r(\mathcal{S}_n) = [y_r^p, y_r^d[\mathcal{M}_n(1)], \dots, y_r^d[\mathcal{M}_n(|\mathcal{M}_n|)]]$ .

The action set of the MDP is expressed as  $\mathcal{A} = \{0, 1\}$ . An action is defined as whether to utilize a current detected symbol as an additional pilot symbol. Specifically, when  $a = 1 \in \mathcal{A}$ , the current detected symbol is used as an additional pilot symbol.

Based on the state and action sets, the state transition function of the MDP for  $a \in \mathcal{A}$  and  $\mathcal{S}_n \in \mathcal{S}_n$  is expressed as follows:

$$\mathbb{T}_{n+1}^{(a,j)}(\mathcal{S}_n) = \mathbb{P} \left[ \mathbb{U}_{n+1}^{(a,j)}(\mathcal{S}_n) \mid \mathcal{S}_n, a \right] = \begin{cases} \mathbb{I}[\mathbf{x}^d[n] = \mathbf{x}_j], & j \in \mathcal{J}_a, a = 1, \\ 1, & j \in \mathcal{J}_a, a = 0. \end{cases} \quad (12)$$

where  $\mathcal{J}_0 = \{0\}$  and  $\mathcal{J}_1 = \{1, \dots, K\}$ . State  $\mathbb{U}_{n+1}^{(a,j)}(\mathcal{S}_n) \in \mathcal{S}_{n+1}$  is the valid state from the current state  $\mathcal{S}_n = (\mathbf{X}_n, \hat{\mathbf{X}}_n, \mathcal{M}_n) \in \mathcal{S}_n$ , and is expressed as

$$\mathbb{U}_{n+1}^{(a,j)}(\mathcal{S}_n) = \begin{cases} ([\mathbf{X}_n, \mathbf{x}_j], [\hat{\mathbf{X}}_n, \hat{\mathbf{x}}[n]], [\mathcal{M}_n \cup n]), & j \in \mathcal{J}_a, a = 1, \\ (\mathbf{X}_n, \hat{\mathbf{X}}_n, \mathcal{M}_n), & j \in \mathcal{J}_a, a = 0. \end{cases} \quad (13)$$

The reward function of the MDP is obtained by the MSE improvement between the channel estimates at the current state  $\mathcal{S}_n$  and the next state  $\mathcal{S}_{n+1}$ . Thus, the reward function from  $\mathcal{S}_n \in \mathcal{S}_n$  to  $\mathcal{S}_{n+1} \in \mathcal{S}_{n+1}$  is defined as

$$R(\mathcal{S}_n, \mathcal{S}_{n+1}) = \mathcal{E}_r(\mathcal{S}_n) - \mathcal{E}_r(\mathcal{S}_{n+1}), \quad (14)$$

where  $\mathcal{E}_r(\mathcal{S}_n)$  is the MSE of the channel estimate for the  $r$ -th receive antenna at state  $\mathcal{S}_n \in \mathcal{S}_n$ , which can be computed as

$$\mathcal{E}_r(\mathcal{S}_n) = \mathbb{E} \left[ \|\hat{\mathbf{h}}_r(\mathcal{S}_n) - \mathbf{h}_r\|^2 \right] = \text{Tr}[\mathbf{C}_e(\mathcal{S}_n)], \quad (15)$$

where the error covariance matrix  $\mathbf{C}_e(\mathcal{S}_n)$  is defined as  $\mathbb{E}\{(\hat{\mathbf{h}}_r(\mathcal{S}_n) - \mathbf{h}_r)(\hat{\mathbf{h}}_r(\mathcal{S}_n) - \mathbf{h}_r)^H\}$ .

Here,  $\mathbf{C}_e(\mathcal{S}_n)$  is independent of the receiver antenna index,  $r$ , because the channel and noise distributions are the same for different receive antenna indices. Thus, the reward function in (14) can be simplified as

$$R(S_n, S_{n+1}) = \text{Tr}[\mathbf{C}_e(S_n) - \mathbf{C}_e(S_{n+1})]. \quad (16)$$

The optimal policy of the MDP at time slot  $n$  is defined as

$$\pi^*(S_n) = \underset{a \in \mathcal{A}}{\text{argmax}} Q(S_n, a). \quad (17)$$

where the Q-value function  $Q(S_n, a)$  is the optimal sum of the rewards. Based on the state transition function in (12), the Q-value function can be expressed as

$$Q(S_n, a) = \sum_{j \in \mathcal{J}_a} T_{n+1}^{(a,j)}(S_n) \left[ R(S_n, U_{n+1}^{(a,j)}(S_n)) + \gamma V^*(U_{n+1}^{(a,j)}(S_n)) \right], \quad (18)$$

where  $0 \leq \gamma \leq 1$  is a discounting factor whose value depends on the target of the optimization problem. For example, a small value is desirable when the accuracy of the channel estimator obtained at the current state is significant. In contrast, a larger value is preferred when the accuracy of the channel estimator obtained at the ending state is significant.

$V^*(U_{n+1}^{(a,j)}(S_n))$  is the optimal sum of the future rewards. The future value function  $V^*(S_m)$  at state  $S_m \in \mathcal{S}_m$  for  $n+1 \leq m$  can be recursively computed, as follows:

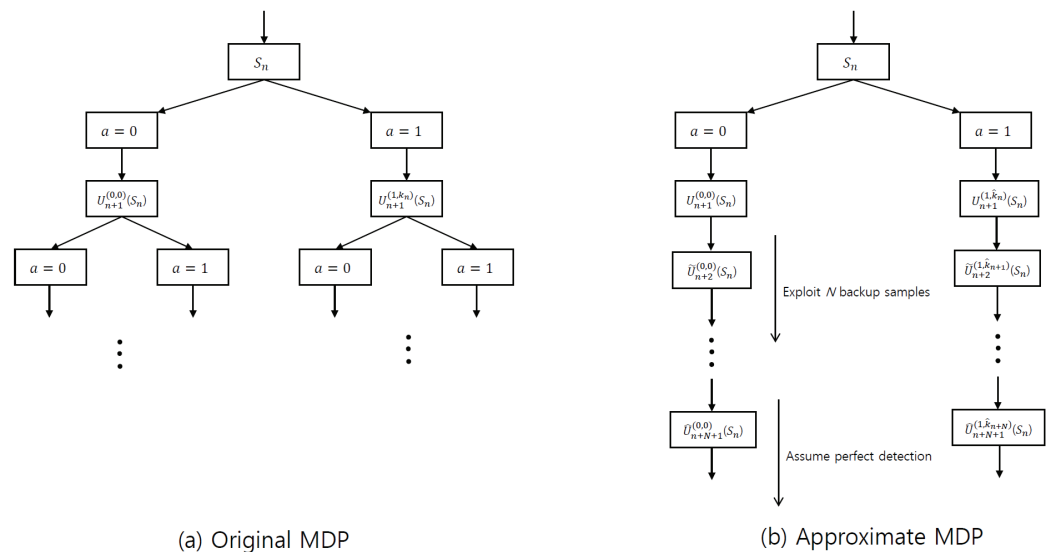
$$V^*(S_m) = \sum_{a \in \mathcal{A}} \pi(S_m, a) \sum_{j \in \mathcal{J}_a} T_{m+1}^{(a,j)}(S_m) \left[ R(S_m, U_{m+1}^{(a,j)}(S_m)) + \gamma V^*(U_{m+1}^{(a,j)}(S_m)) \right], \quad (19)$$

where  $\pi(S_m, a)$  is a state–action transition function, expressed as

$$\pi(S_m, a) = \mathbb{I}\{a = \underset{a' \in \mathcal{A}}{\text{argmax}} Q(S_m, a')\}, \quad (20)$$

where  $Q(S_m, a)$  is the Q-value function that can be calculated as the sum of the rewards obtained after taking action  $a \in \mathcal{A}$  at state  $S_m \in \mathcal{S}_m$ .

Using the MDP in (10), (12), and (13), the state–action diagram of the original MDP is depicted in Figure 2a. In this figure, state  $S_n$  is transited to the next valid state,  $U_{n+1}^{(a,j)}(S_n)$ , based on action  $a$ . Particularly, when  $a = 1$ , state  $S_n$  is transited to state  $U_{n+1}^{(1,k_n)}(S_n)$  by utilizing the transmitted symbol index,  $k_n$ . Based on the state and state–action transition functions in (12) and (20), the state is transited to the next valid state until the end of a data block. As previously mentioned, the original MDP, which is shown in Figure 2a, cannot be solved by dynamic programming.



**Figure 2.** State–action diagrams of the original MDP (a) where  $k_n$  is the transmitted symbol index, and the approximate MDP (b) where  $\hat{k}_n$  is the detected symbol index for  $a \in \mathcal{A}$  and  $S_n \in \mathcal{S}_n$ .

First, the state and state–action functions are unavailable to the receiver because the information of the transmitted symbols,  $\mathbf{x}_{k_n}$ , and the true channel information,  $\mathbf{H}$ , are unknown. In addition, the computational complexity and latency required to solve the original MDP are extremely high because the number of states exponentially increases with data length  $T_d$ .

#### 4. Proposed RL-Based Channel Estimator

In this section, an RL-based channel estimator is proposed. To address the unknown state and state–action functions, an RL algorithm is adopted because it provides a solution for the partially observable MDP [27,28]. Based on this algorithm, a computationally efficient RL solution is also proposed. The key concept of the proposed solution is to approximate the state–action transition functions to determine the optimal policy by separating the cases using the APPs.

The overall procedure of the proposed RL-based channel estimator is illustrated in Figure 3. The proposed channel estimator exploits the information of  $(\hat{x}[m], \theta_j[m])$  obtained from the MIMO detector. In the proposed channel estimator, the optimal policy is calculated by using only  $N$  APPs  $(\theta_j[n], \dots, \theta_j[n + N])$  for a computationally efficient algorithm. The channel estimate is then updated according to the optimal policy. Details of the proposed channel estimator, i.e., how to approximate the MDP and how to derive the optimal policy in a closed form, are explained in this section.

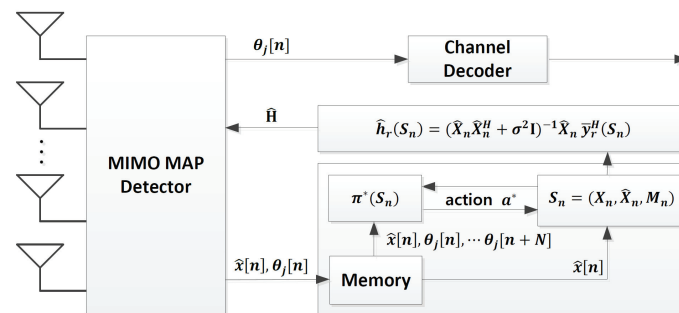


Figure 3. System structure of the proposed data-aided channel estimator.

##### 4.1. Statistical State Transition

In this section, the state transition function in (12) at time slot  $n$  is approximated using the APP  $\theta_j[n]$ . The basic concept was introduced in [26] by assuming the APP  $\theta_j[n]$  as the probability of the event,  $\{\mathbf{x}[n] = \mathbf{x}_j\}$ . Thus, the state transition function in (12) at time slot  $n$  is approximated as follows:

$$\hat{\mathbf{T}}_{n+1}^{(a,j)}(S_n) = \begin{cases} \theta_j[n], & j \in \mathcal{J}_a, a = 1, \\ 1, & j \in \mathcal{J}_a, a = 0. \end{cases} \quad (21)$$

where the detected symbol index at time slot  $n$  is denoted as  $\hat{k}_n$ . Note that APP  $\theta_j[n]$  can be interpreted as the probability of the event  $\{\mathbf{x}[n] = \mathbf{x}_j\}$ ; thus, it is called a statistical transition. In addition, when the data detection performance is improved, i.e.,  $\theta_{k_n}[n] \rightarrow 1$ , the approximate state transition function in (21) approaches the true state transition function in (12).

##### 4.2. State–Action Transition Using Backup Samples

After time slot  $n + 1 \leq m$ , the state in (20) is assumed to be transitioned to a virtual state that mimics the possible next states by exploiting the expected transmitted symbol,  $\tilde{\mathbf{x}}[m]$ . The expected transmitted symbol,  $\tilde{\mathbf{x}}[m]$ , is defined as

$$\tilde{\mathbf{x}}[m] = \sum_{j=1}^K \theta_j[m] \mathbf{x}_j. \tag{22}$$

In this study, the use of the expected transmitted symbol is the same as in [26], except its use is limited to  $N$  backup samples to reduce the complexity. A backup sample is defined as APP  $\theta_j[m]$  for  $n + 1 \leq m \leq n + N$  because the expected transmitted symbol can be computed by  $\theta_j[m]$ . Thus, the Q-value function can be calculated after all  $\theta_j[m]$  for  $n + 1 \leq m \leq n + N$  values are obtained. Using a backup sample of an APP, the state–action transition is expressed as

$$\hat{\pi}(S_m, a) = 1. \tag{23}$$

Thus, the virtual state,  $\tilde{U}_m^{(a,j)}(S_n) \in \mathcal{S}_m$ , that can be transited from  $S_n \in \mathcal{S}_n$  is expressed as

$$\tilde{U}_m^{(a,j)}(S_n) = (\mathbf{X}_m^{(a,j)}, \hat{\mathbf{X}}_m^{(a)}, \mathcal{M}_m^{(a)}), \tag{24}$$

where their components are

$$\begin{aligned} \mathbf{X}_m^{(a,j)} &= \begin{cases} [\mathbf{X}_n, \mathbf{x}_j, \tilde{\mathbf{x}}[n + 1], \dots, \tilde{\mathbf{x}}[n + N]], & a = 1, \\ [\mathbf{X}_n, \tilde{\mathbf{x}}[n + 1], \dots, \tilde{\mathbf{x}}[n + N]], & a = 0. \end{cases} \\ \hat{\mathbf{X}}_m^{(a)} &= \begin{cases} [\hat{\mathbf{X}}_n, \hat{\mathbf{x}}[n], \tilde{\mathbf{x}}[n + 1], \dots, \tilde{\mathbf{x}}[n + N]], & a = 1, \\ [\hat{\mathbf{X}}_n, \tilde{\mathbf{x}}[n + 1], \dots, \tilde{\mathbf{x}}[n + N]], & a = 0. \end{cases} \\ \mathcal{M}_m^{(a)} &= \begin{cases} [\mathcal{M}_n \cup \{n, \dots, n + N\}], & a = 1, \\ [\mathcal{M}_n \cup \{n + 1, \dots, n + N\}], & a = 0. \end{cases} \end{aligned}$$

Because a virtual state mimics the transitions to the candidate symbols, state  $\tilde{U}_m^{(a,j)}(S_n) \in \mathcal{S}_m$  is always transited to a virtual state  $\tilde{U}_{m+1}^{(a,j)}(S_n) \in \mathcal{S}_{m+1}$ . Therefore, the corresponding state transition function is written as

$$\hat{\tau}_{m+1}^{(a,j)}(\tilde{U}_m^{(a,j)}(S_n)) = 1, \tag{25}$$

where  $n + 1 \leq m \leq n + N$ .

### 4.3. State–Action Transition after Backup Samples

In this subsection, the virtual states after  $n + N$  that can be transited without the information of the backup samples,  $\theta_j[m]$ , are described for  $n + N + 1 \leq m$ . To achieve this, the states,  $\hat{U}_{m+1}^{(a,j)}(S_n)$ , for  $n + N + 1 \leq m$  are assumed to optimally act when all symbols are correctly detected. By using the property of  $\mathbf{x}[m] = \hat{\mathbf{x}}[m]$  after time slot  $n + N + 1$ , an approximate virtual state is expressed as

$$\hat{U}_m^{(a,j)}(S_n) = (\mathbf{X}_m^{(a,j)}, \hat{\mathbf{X}}_m^{(a)}, \mathcal{M}_m^{(a)}), \tag{26}$$

where its components are defined as

$$\begin{aligned} \mathbf{X}_m^{(a,j)} &= [\mathbf{X}_{n+N+1}^{(a,j)}, \hat{\mathbf{x}}[n + N + 1], \dots, \hat{\mathbf{x}}[m - 1]], \\ \hat{\mathbf{X}}_m^{(a)} &= [\hat{\mathbf{X}}_{n+N+1}^{(a)}, \hat{\mathbf{x}}[n + N + 1], \dots, \hat{\mathbf{x}}[m - 1]], \\ \mathcal{M}_m^{(a)} &= [\mathcal{M}_{n+N+1}^{(a)} \cup \{n + N + 1, \dots, m - 1\}], \end{aligned}$$

where  $(\mathbf{X}_{n+N+1}^{(a,j)}, \hat{\mathbf{X}}_{n+N+1}^{(a)}, \mathcal{M}_{n+N+1}^{(a)})$  are the components of  $\tilde{U}_{n+N+1}^{(a,j)}(S_n)$ .



In Figure 2b, a state–action diagram of the approximate MDP is depicted. The original MDP requires information regarding the transmitted symbols for the state transition, as shown in Figure 2a. In contrast, the approximate MDP utilizes virtual states  $\tilde{U}_m^{(a,j)}(S_n)$  and  $\hat{U}_m^{(a,j)}(S_n)$ , which mimic the transitions to the candidate symbols for an unknown transmitted symbol and action. Specifically, virtual state  $\tilde{U}_m^{(a,j)}(S_n)$  is used at time slot  $n + 1 \leq m \leq n + N$  and after time slot  $n + N$ , respectively. These two approximations decrease the number of transitions to the next state transition, so the calculation to solve the MDP is considerably reduced.

#### 4.4. Proposed Optimal Policy

Using the approximations in (21), (23), and (24), the optimal policy can be determined. However, the calculation latency is still considerable, because the optimal policy can be computed at the end of a data block. To prevent this computational burden, the proposed solution separates a data block into  $N_b$  data subblocks and subsequently characterizes the optimal policy for each data subblock, as shown in Figure 4. Based on this characterization, the state in (10) and the corresponding channel estimate using (11) are updated for a data subblock. To realize this data subblock separation, the data subblock length is defined as  $T_b$ , which satisfies  $N_b = T_d/T_b$ . Thus, a set of time slot indices of the  $b$ -th data subblock in the  $d$ -th data block,  $\mathcal{N}_{b,d}$ , is defined as  $\{T_p + (b - 1)T_b + (d - 1)T_d + 1, \dots, T_p + bT_b + (d - 1)T_d\}$ , for  $b \in \{1, \dots, N_b\}$  and  $d \in \{1, \dots, N_d\}$  (see Figure 4).

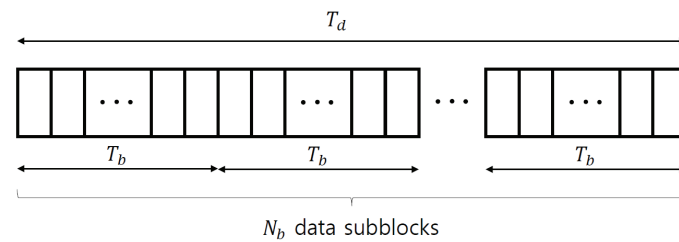


Figure 4.  $d$ -th data block consists of  $N_b$  data subblocks with  $T_b$  symbols.

Using the virtual states in (24) and (26), the  $Q$ -value function is written as

$$Q(S_n, a) = \sum_{j \in \mathcal{J}_a} T_{n+1}^{(a,j)}(S_n) \left[ R(S_n, \tilde{U}_{n+1}^{(a,j)}(S_n)) + \sum_{m=n+1}^{n+N} \gamma^{m-n} R(\tilde{U}_m^{(a,j)}(S_n), \tilde{U}_{m+1}^{(a,j)}(S_n)) \right. \\ \left. + \gamma^{N+1} V^*(\hat{U}_{n+N+1}^{(a,j)}(S_n)) \right], \quad (27)$$

where the future value function,  $V^*(\hat{U}_{n+N+1}^{(a,j)}(S_n))$ , is obtained based on the approximation of  $\hat{U}_m^{(a,j)}(S_n)$  as follows:

$$V^*(\hat{U}_{n+N+1}^{(a,j)}(S_n)) \approx R(\tilde{U}_{n+N+1}^{(a,j)}(S_n), \hat{U}_{n+N+2}^{(a,j)}(S_n)) + \sum_{m=n+N+2}^{\mathcal{N}_{b,d}(T_b)} R(\hat{U}_m^{(a,j)}(S_n), \hat{U}_{m+1}^{(a,j)}(S_n)). \quad (28)$$

In the future reward in (28), the discounting factor is assumed to be 1 to reduce the complexity by a simple calculation.

Based on (27) and (28), the optimal policy for each state is obtained as a closed-form expression, as described in the following theorem:

**Theorem 1.** Under the virtual states and the use of backup samples, the optimal policy for the state  $S_n = (\mathbf{X}_n, \hat{\mathbf{X}}_n, \mathcal{M}_n) \in S_n$  is

$$\pi^*(S_n) = \mathbb{I} \left[ \frac{\sum_{m=n}^{n+N} \gamma^{m-n} (1-\gamma) U_m(S_n) + \gamma^{N+1} U_{\mathcal{N}_{b,d}(T_b)+1}(S_n)}{\sum_{m=n}^{n+N} \gamma^{m-n} (1-\gamma) L_m(S_n) + \gamma^{N+1} L_{\mathcal{N}_{b,d}(T_b)+1}(S_n)} \geq 1 \right], \quad (29)$$

where functions  $U_m(S_n)$  and  $L_m(S_n)$  are respectively defined as

$$U_m(S_n) = \|\mathbf{t}_m\|^2 (N_0 + N_0^2 \|\mathbf{t}_m\|^2 + \|\mathbf{v}_m\|^2)$$

$$L_m(S_n) = \|\mathbf{t}_m\|^2 (2N_0^2 \beta_m + \delta_m + \|\mathbf{e}_m - \mathbf{u}_m + \mathbf{v}_m\|^2)$$

All components are defined as

$$\mathbf{Q}_m = \left( \hat{\mathbf{x}}_n \hat{\mathbf{x}}_n^H + \sum_{l=n+1}^m \tilde{\mathbf{x}}[l] \tilde{\mathbf{x}}^H[l] + N_0 \mathbf{I}_{N_t} \right)^{-1}, \quad \mathbf{D}_m = \hat{\mathbf{x}}_n (\hat{\mathbf{x}}_n - \mathbf{x}_n)^H + \sum_{l=n+1}^m \tilde{\mathbf{x}}[l] (\tilde{\mathbf{x}}[l] - \tilde{\mathbf{x}}[l])^H + N_0 \mathbf{I}_{N_t},$$

$$\mathbf{t}_m = \frac{1}{\sqrt{1 + \alpha_m}} \mathbf{Q}_m \hat{\mathbf{x}}[n], \quad \mathbf{e}_m = \frac{1}{\sqrt{1 + \alpha_m}} (\hat{\mathbf{x}}[n] - \tilde{\mathbf{x}}[n]), \quad \mathbf{u}_m = \mathbf{D}_m^H \mathbf{t}_m, \quad \mathbf{v}_m = \frac{\mathbf{D}_m^H \mathbf{Q}_m \mathbf{t}_m}{\|\mathbf{t}_m\|^2},$$

$$\alpha_m = \hat{\mathbf{x}}^H[n] \mathbf{Q}_m \hat{\mathbf{x}}[n], \quad \beta_m = \frac{\mathbf{t}_m^H \mathbf{Q}_m \mathbf{t}_m}{\|\mathbf{t}_m\|^2}, \quad \delta_m = \frac{1}{1 + \alpha_m} \left( \sum_{j=1}^K \theta_j[n] \|\hat{\mathbf{x}}[n] - \mathbf{x}_j\|^2 - \|\hat{\mathbf{x}}[n] - \tilde{\mathbf{x}}[n]\|^2 \right)$$

$$\mathbf{Q}_{\mathcal{N}_{b,d}(T_b)+1} = \left( \mathbf{Q}_{n+N}^{-1} + (\mathcal{N}_{b,d}(T_b) - (n + N - 1)) \mathbf{I}_{N_t} \right)^{-1}, \quad \mathbf{D}_{\mathcal{N}_{b,d}(T_b)+1} = \mathbf{D}_{n+N}. \quad (30)$$

**Proof.** See Appendix A.  $\square$

#### 4.5. Summary: The Proposed Algorithm

The proposed channel estimator is summarized in Algorithm 1. First, the receiver initializes the state during pilot transmission. In this algorithm, the current state is updated and transitioned to the next state according to the optimal action obtained using (29). For example, the most probable state transition is used when  $\alpha^* = 1$  for the unknown transmitted symbol index. This transition ensures a true state transition as  $\theta_j[n]$  approaches 1 in reliable communication. At the end of a data subblock, the proposed channel estimator updates the channel estimate using the current state,  $S_n$ .

---

#### Algorithm 1: The proposed channel estimator.

---

- 1 Set  $\mathbf{H} \leftarrow \hat{\mathbf{H}} = [\hat{\mathbf{h}}_1, \dots, \hat{\mathbf{h}}_{N_r}]$  from (4)
  - 2 Initialize  $S_1 = (\mathbf{X}^P, \mathbf{X}^P, \phi)$ .
  - 3 **for**  $d = 1$  to  $N_d$  **do**
  - 4     **for**  $b = 1$  to  $N_b$  **do**
  - 5         **for**  $n \in \mathcal{N}_{b,d}$  **do**
  - 6             Obtain  $\hat{\mathbf{x}}[n]$  from (8) and  $\{\theta_j[n], \dots, \theta_j[n+N]\}$  from (5) for  $j \in \mathcal{K}$
  - 7             Compute  $a^* = \pi^*(S_n)$  from (29).
  - 8             Set  $j^* = 0$  for  $a^* = 0$  and  $\mathbf{x}_{j^*} = \hat{\mathbf{x}}[n]$  for  $a^* = 1$ .
  - 9             Update  $S_{n+1} \leftarrow U_{n+1}^{(a^*, j^*)}(S_n)$  from (13).
  - 10         **end**
  - 11         Set  $\mathbf{H} \leftarrow \hat{\mathbf{H}} = [\hat{\mathbf{h}}_1(S_n), \dots, \hat{\mathbf{h}}_{N_r}(S_n)]$  from (11).
  - 12     **end**
  - 13 **end**
- 

#### 4.6. Complexity Analysis

In this subsection, the complexity of both the proposed channel estimator and that in [26] is discussed based on the number of states visited in the calculation of the optimal

policy. This is because the rewards in the optimal policy are computed based on the states, and the calculation in (29) is similar to that in [26]. First, when the current state is  $\mathbf{S}_n \in \mathcal{S}_n$  in the  $d$ -th data block, the number of visiting states in [26] is exactly  $dT_d - n$ . By contrast, the number of visiting states using the proposed channel estimator in the  $b$ -th data subblock is exactly  $(b - 1)T_b + 1 + (d - 1)T_d - n$ . Thus, the number of states  $(T_d - (b - 1)T_b - 1)$  is not used in the policy calculation on introducing the data subblocks. In addition to the complexity, the proposed optimal policy can be calculated after obtaining  $N$  backup samples, whereas in the approach in [26], this is possible at the end of a data block. Thus, the latency of the optimal policy by the approach in [26] is much longer than that of the proposed optimal policy.

## 5. Simulation Results

This section discusses the performance of the proposed channel estimator. The number of antennas in MIMO systems is  $(N_t, N_r) = (4, 4)$ . A rate 1/2 turbo code is adopted for channel coding, and 4-quadrature amplitude modulation (QAM) is adopted for symbol mapping. The frame consists of  $(T_p, T_d, N_d) = (8, 64, 20)$ , and the proposed channel estimator utilizes a data subblock as  $(T_b, N_b) = (16, 4)$ . In addition, the parameters of the proposed channel estimator are  $(N, \gamma) = (1, 0.5)$ , unless specified otherwise. The per-bit signal-to-noise ratio (SNR) is defined as  $E_b/N_0 = \frac{1}{\log_2 |\mathcal{X}| N_0}$ .

In all figures, the performance with perfect and imperfect channel estimates using the LMMSE method are denoted as PCSI and CE, respectively. For performance benchmarking, the optimal cases of the proposed channel estimator and the expected-symbol-based channel estimator utilizing perfect knowledge of the transmitted symbol and the expected symbol in (22) as an additional pilot symbol, respectively, are compared. The performance is measured in terms of the block error rate (BLER) and the normalized MSE (NMSE). In Figure 5, the proposed channel estimator is compared with other channel estimators, and the conventional RL method used in [26] is also depicted. It shows that the BLER of the proposed estimator is better than those of the conventional and expected-symbol-based estimators regardless of the per-bit SNR. Moreover, the proposed channel estimator outperforms the conventional estimator of [26]. This is because the proposed channel estimator updates a channel estimate by  $N_b$  in a data block, whereas the method in [26] updates it once at the end of a data block.

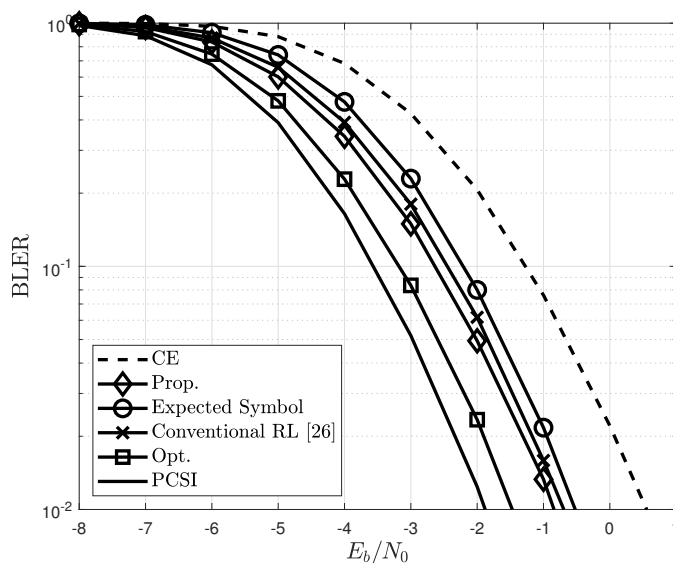


Figure 5. BLERs of conventional and proposed channel estimators for the different estimations.

Figure 6 compares the BLERs of the conventional and proposed channel estimators for different modulations. For 16-QAM, a MIMO system with  $(N_t, N_r) = (2, 4)$  is considered because of the SNR range. The proposed channel estimator achieves an improved BLER

compared to the conventional LMMSE channel estimators. This result demonstrates the effectiveness of the proposed channel estimator, which optimizes the selection of detected symbols. The improvements to achieve a BLER of  $10^{-1}$  are approximately 1.2 dB and 0.7 dB for the 4- and 16-QAM, respectively. The BLER for the 16-QAM is more improved than that of the PCSI, which is better than that of the 4-QAM. This is because in 16-QAM, the number of reliable detected symbols that can be used as additional pilot symbols is larger than in 4-QAM.

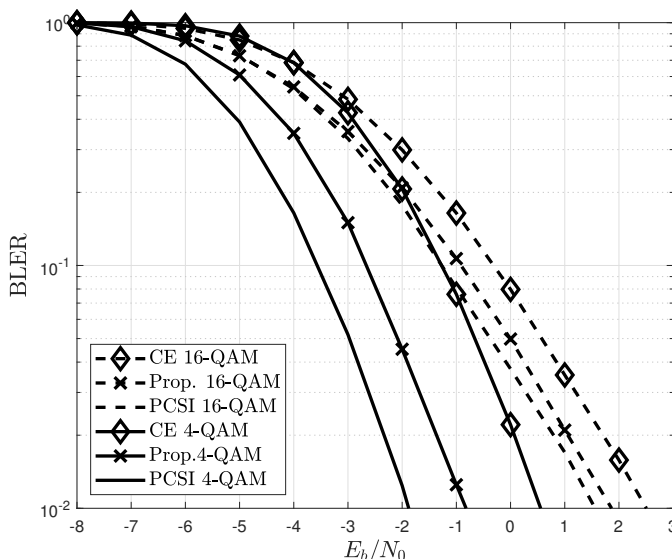


Figure 6. BLERs of conventional and proposed channel estimators for different modulations.

The NMSEs of the proposed channel estimator for different data subblock lengths are shown in Figure 7. The NMSE improves as  $N_b$  decreases. This is because the approximate MDP using data subblocks approaches the original MDP as  $N_b$  decreases. However, as shown in Figure 7, the NMSE improvement is insignificant, whereas the complexity exponentially increases with  $T_b$ . Thus,  $(T_b, N_b) = (16, 4)$  is considered in this study for the simulations.

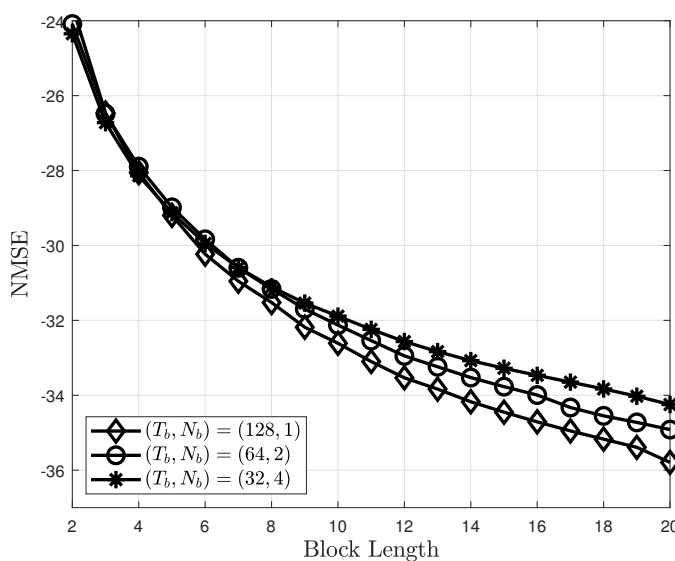


Figure 7. NMSEs of the proposed channel estimator for different  $T_b$  and  $N_b$ .

The NMSE of the proposed channel estimator based on the number of backup samples is shown in Figure 8. Noticeably, the NMSE is improved as the number of backup samples

increases. This is because the accuracy of the state–action diagram model improves as the number of backup samples increases. In addition, with a small value of  $N$ , the proposed channel estimator achieves a sufficient NMSE performance. It should be noted that the complexity and latency required to determine the optimal policy increase with the number of backup samples.

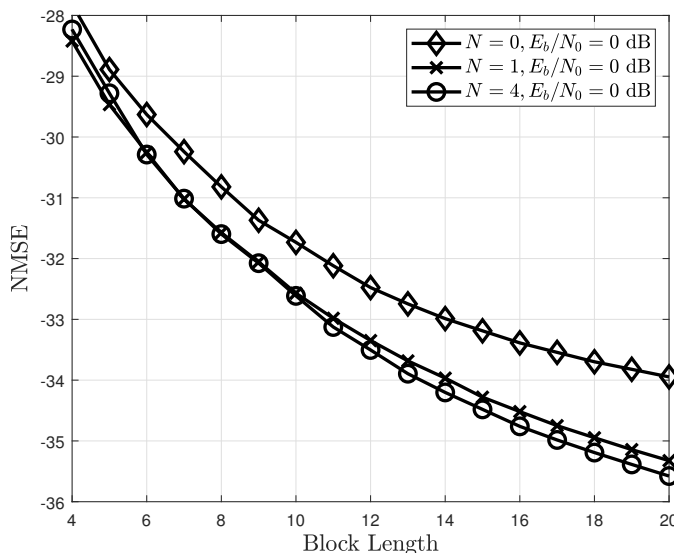


Figure 8. NMSE of the proposed channel estimator based on the number of backup samples  $N$ .

Figures 9 and 10 are the results obtained using the proposed channel estimator in time-varying channels. Specifically, a first-order Gaussian–Markov process used in [29,30] was adopted. In this process, the channel matrix at time slot  $n$  is defined as

$$\mathbf{H}^{(n)} = \sqrt{1 - \epsilon^2} \mathbf{H}^{(n-1)} + \epsilon \mathbf{e}^{(n)}, \tag{31}$$

where  $n \in \mathcal{N}_{b,d}$  for  $b \in \{1, 2, \dots, N_b\}$  and  $d \in \{1, 2, \dots, N_d\}$ .  $\epsilon \in [0, 1]$  is a temporal correlation coefficient depending on the velocity, and  $\mathbf{H}^{(0)}$  is an initial channel estimate. Each element in  $\mathbf{e}^{(n)} \in \mathbb{C}^{N_r \times N_t}$  is assumed to follow  $\mathcal{CN}(0, 1)$ . Temporal correlation coefficients  $\epsilon = 5 \times 10^{-3}$  and  $\epsilon = 10^{-2}$  are used for the simulations.

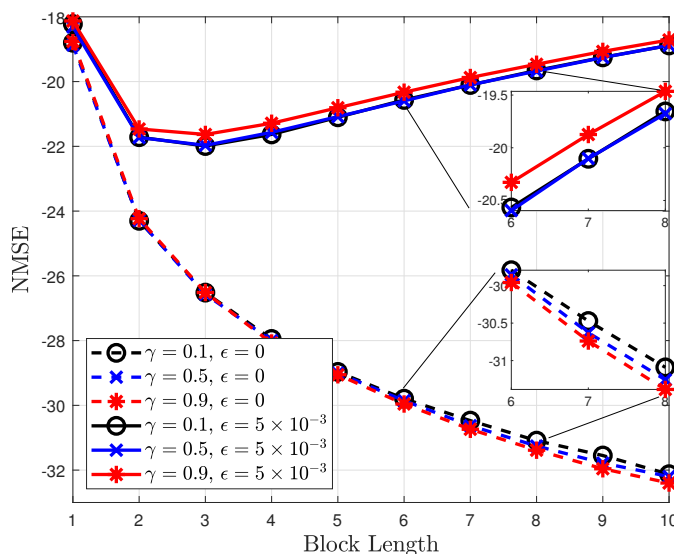


Figure 9. NMSEs of the proposed channel estimator for different discounting factors.

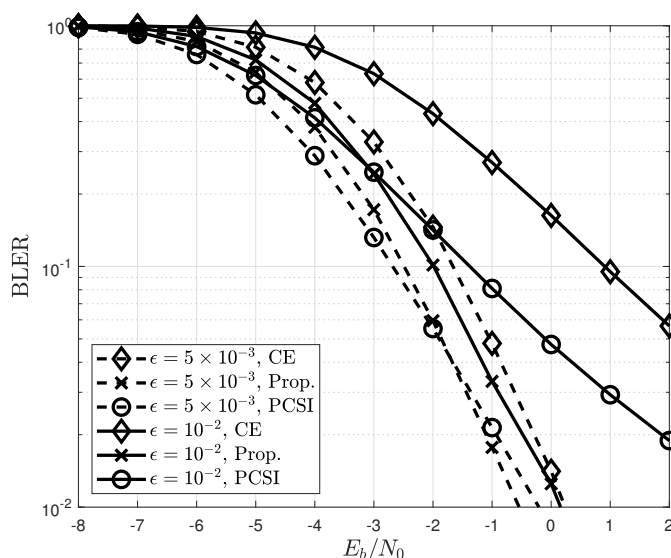


Figure 10. BLERs of the proposed channel estimators in time-varying channels.

Figure 9 shows the variation in the NMSE of the proposed channel estimator with the discounting factor. When a channel varies over time as  $\epsilon = 5 \times 10^{-3}$ , an NMSE with  $\gamma = 0.1$  is better than it is with  $\gamma = 0.9$ . This is because the rewards at the future states in the time-varying channels are insignificant; therefore, a small value of the discounting factor is preferable. By contrast, when the channels are time-invariant, the rewards at the future states as well as those at the current state are important. Thus, the large value of  $\gamma = 0.9$  improves the NMSE compared to  $\gamma = 0.1$ . Figure 10 compares the BLERs of the proposed and conventional channel estimators. When  $\epsilon = 10^{-2}$ , the BLERs of the CE are severely degraded because the CE method cannot capture the channel variation. However, the proposed channel estimator shows robustness in time-varying channels because the channel variation can be tracked efficiently by selecting the detected symbols.

## 6. Conclusions

In this paper, a low-complexity algorithm for an RL-based channel estimator for MIMO systems was proposed. The proposed channel estimator adaptively selects detected symbols as additional pilot symbols to minimize the channel estimation error. In this study, an MDP problem was introduced, and a practical algorithm to solve it was developed using backup samples and data subblocks. Simulation results showed that the proposed channel estimator significantly improves the BLER and the NMSE compared to the conventional channel estimator.

A future direction of this study is to develop the RL approach for a realistic channel. The proposed method was derived based on the Rayleigh fading channel, but the realistic channel may have a line of sight. Thus, the MDP under the Rician fading channel should be investigated. Another important direction is to develop the RL approach for frequency-selective channels. In frequency-selective channels, the use of multiple sub-carriers can increase computational complexity considerably. Thus, a low-complexity algorithm in frequency-selective channels is necessary. Lastly, the RL approach can also be extended to other advanced channel estimators, such as the iterative method. In this method, the MDP should be reformulated according to the channel estimator.

**Author Contributions:** Conceptualization, M.M. and T.-K.K.; methodology, T.-K.K.; software, M.M. and T.-K.K.; validation, M.M. and T.-K.K.; formal analysis, M.M. and T.-K.K.; investigation, M.M. and T.-K.K.; resources, T.-K.K.; data curation, T.-K.K.; writing—original draft preparation, T.-K.K.; writing—review and editing, M.M. and T.-K.K.; visualization, T.-K.K.; supervision, M.M.; project administration, M.M. and T.-K.K.; funding acquisition, M.M. and T.-K.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** The work of M.M. was supported in part by a National Research Foundation of Korea (NRF) grant, funded by the Korea Government (MSIT) (No. 2020R1F1A1071649), and in part by the BK21 FOUR Project, funded by the Ministry of Education, Korea (4199990113966). The work of T.-K.K. was supported by a National Research Foundation of Korea (NRF) grant, funded by the Korea Government (MIST) (No. 2021R1F1A1063273).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare that there is no conflict of interest.

## Appendix A. Proof of Theorem 1

Although the basic derivation of the optimal policy is based on [26], two additional factors are considered, which are presented in this appendix. The first is that the proposed derivation considers a discounting factor in the Q-value; thus, the intermediate rewards do not disappear, unlike in [26]. Second, a finite number of backup samples are used in the derivation; thus, the rewards that do not exploit the APPs are approximated differently compared to [26].

Under the assumption that the discounting factor is 1, the future value function at state  $\tilde{U}_{n+N+1}^{(a,j)}(S_n) \in \mathcal{S}_{n+N+1}$  is expressed by substituting (14) in (28), as follows:

$$\begin{aligned} V^*(\tilde{U}_{n+N+1}^{(a,j)}(S_n)) &= \text{Tr} \left[ \mathbf{C}_e(\tilde{U}_{n+N+1}^{(a,j)}(S_n)) - \mathbf{C}_e(\hat{U}_{n+N+2}^{(a,j)}(S_n)) \right. \\ &\quad \left. + \sum_{m=n+N+2}^{\mathcal{N}_{b,d}(T_b)} \mathbf{C}_e(\hat{U}_m^{(a,j)}(S_n)) - \mathbf{C}_e(\hat{U}_{m+1}^{(a,j)}(S_n)) \right] \\ &= \text{Tr} \left[ \mathbf{C}_e(\tilde{U}_{n+N+1}^{(a,j)}(S_n)) - \mathbf{C}_e(\hat{U}_{\mathcal{N}_{b,d}(T_b)+1}^{(a,j)}(S_n)) \right]. \end{aligned} \quad (\text{A1})$$

By substituting (14) and (A1) into (27), the Q-value function can be obtained as follows:

$$Q(S_n, a) = \sum_{j \in \mathcal{J}_a} \mathbb{T}_{n+1}^{(a,j)}(S_n) \text{Tr} \left[ \mathbf{C}_e(S_n) + \sum_{m=n}^{n+N} \gamma^{m-n} (\gamma - 1) \mathbf{C}_e(\tilde{U}_{m+1}^{(a,j)}(S_n)) - \gamma^{N+1} \mathbf{C}_e(\hat{U}_{\mathcal{N}_{b,d}(T_b)+1}^{(a,j)}(S_n)) \right]. \quad (\text{A2})$$

Thus, the optimal policy in (17) is expressed as

$$\begin{aligned} \pi^*(S_n) &= \underset{a \in \{0,1\}}{\text{argmax}} Q(S_n, a) \\ &= \mathbb{I}[(Q(S_n, 1) - Q(S_n, 0)) \geq 0] \\ &= \mathbb{I} \left[ \text{Tr} \left[ \sum_{m=n}^{n+N} \gamma^{m-n} (\gamma - 1) \left( \sum_{j=1}^K \theta_j[n] \mathbf{C}_e(\tilde{U}_{m+1}^{(1,j)}(S_n)) - \mathbf{C}_e(\tilde{U}_{m+1}^{(0,0)}(S_n)) \right) \right. \right. \\ &\quad \left. \left. - \gamma^{N+1} \left( \sum_{j=1}^K \theta_j[n] \mathbf{C}_e(\hat{U}_{\mathcal{N}_{b,d}(T_b)+1}^{(1,j)}(S_n)) - \mathbf{C}_e(\hat{U}_{\mathcal{N}_{b,d}(T_b)+1}^{(0,0)}(S_n)) \right) \right] \geq 0 \right]. \end{aligned} \quad (\text{A3})$$

In (17), the optimal policy is determined by the difference between the error covariance matrices with  $a = 0$  and  $a = 1$ . The error covariance matrices for virtual states  $\tilde{U}_m^{(a,j)}(S_n)$  and  $\hat{U}_m^{(a,j)}(S_n)$  are derived as described below.

### Appendix A.1. Error Covariance Calculation for $\tilde{U}_m^{(a,j)}(S_n)$

To obtain the error covariance matrix, the distribution of the received symbols,  $\bar{\mathbf{y}}_r^H(\tilde{U}_m^{(a,j)}(S_n))$ , in (2) is required, which is given by

$$\tilde{\mathbf{y}}_r^H \left( \tilde{\mathbf{U}}_m^{(a,j)}(S_n) \right) \sim \mathcal{CN} \left( \mathbf{0}_{|\mathcal{M}_m^{(a)}|}, \left( \mathbf{X}_m^{(a,j)} \right)^H \mathbf{X}_m^{(a,j)} + N_0 \mathbf{I}_{|\mathcal{M}_m^{(a)}|} \right), \quad (\text{A4})$$

for  $j \in \mathcal{J}_a$  and  $a \in \mathcal{A}$ . Thus, the error covariance matrix in (A3) is computed using the result in [26], as follows:

$$\mathbf{C}_e \left( \tilde{\mathbf{U}}_m^{(a,j)}(S_n) \right) = N_0 \mathbf{Q}_m^{(a)} - N_0^2 \left( \mathbf{Q}_m^{(a)} \right)^2 + \mathbf{Q}_m^{(a)} \mathbf{D}_m^{(a,j)} \left( \mathbf{D}_m^{(a,j)} \right)^H \mathbf{Q}_m^{(a)}, \quad (\text{A5})$$

where

$$\mathbf{Q}_m^{(a)} = \left( \hat{\mathbf{X}}_m^{(a)} \left( \hat{\mathbf{X}}_m^{(a)} \right)^H + N_0 \mathbf{I}_{N_t} \right)^{-1} \stackrel{(a)}{=} \begin{cases} \left( \hat{\mathbf{X}}_n \hat{\mathbf{X}}_n^H + \sum_{l=n+1}^{m-1} \tilde{\mathbf{x}}[l] \tilde{\mathbf{x}}^H[l] + N_0 \mathbf{I}_{N_t} \right)^{-1}, & a = 0, \\ \left( \left( \mathbf{Q}_m^{(0)} \right)^{-1} + \hat{\mathbf{x}}[n] \hat{\mathbf{x}}^H[n] \right)^{-1}, & a = 1. \end{cases}$$

$$\mathbf{D}_m^{(a,j)} = \hat{\mathbf{X}}_m^{(a)} \left( \hat{\mathbf{X}}_m^{(a)} - \mathbf{X}_m^{(a,j)} \right)^H + N_0 \mathbf{I}_{N_t}$$

$$\stackrel{(b)}{=} \begin{cases} \hat{\mathbf{X}}_n \left( \hat{\mathbf{X}}_n - \mathbf{X}_n \right)^H + \sum_{l=n+1}^{m-1} \tilde{\mathbf{x}}[l] \left( \tilde{\mathbf{x}}[l] - \tilde{\mathbf{x}}[l] \right)^H + N_0 \mathbf{I}_{N_t}, & j \in \mathcal{J}_a, a = 0, \\ \mathbf{D}_m^{(0,0)} + \hat{\mathbf{x}}[n] \left( \hat{\mathbf{x}}[n] - \mathbf{x}_j \right)^H, & j \in \mathcal{J}_a, a = 1. \end{cases}$$

Thus, the matrix  $\mathbf{Q}_m^{(1)}$  is re-expressed as

$$\mathbf{Q}_m^{(1)} = \mathbf{Q}_m^{(0)} - \frac{\mathbf{Q}_m^{(0)} \hat{\mathbf{x}}[n] \hat{\mathbf{x}}^H[n] \mathbf{Q}_m^{(0)}}{1 + \hat{\mathbf{x}}^H[n] \mathbf{Q}_m^{(0)} \hat{\mathbf{x}}[n]}. \quad (\text{A6})$$

In addition,  $\mathbf{D}_m^{(1,j)} \left( \mathbf{D}_m^{(1,j)} \right)^H$  can be computed as

$$\mathbf{D}_m^{(1,j)} \left( \mathbf{D}_m^{(1,j)} \right)^H = \left( \mathbf{D}_m^{(0,0)} + \hat{\mathbf{d}}_n \right) \left( \mathbf{D}_m^{(0,0)} + \hat{\mathbf{d}}_n \right)^H + \hat{\delta}_n \hat{\mathbf{x}}[n] \hat{\mathbf{x}}^H[n], \quad (\text{A7})$$

where  $\hat{\mathbf{d}}_n = \hat{\mathbf{x}}[n] \left( \hat{\mathbf{x}}[n] - \tilde{\mathbf{x}}[n] \right)^H$ , and

$$\hat{\delta}_n = \sum_{j=1}^K \theta_j [n] \left\| \hat{\mathbf{x}}[n] - \mathbf{x}_j \right\|^2 - \left\| \hat{\mathbf{x}}[n] - \tilde{\mathbf{x}}[n] \right\|^2. \quad (\text{A8})$$

#### Appendix A.2. Error Covariance Calculation for $\hat{\mathbf{U}}_m^{(a,j)}(S_n)$

Similar to the description in Appendix A.1, the error covariance matrix for  $\hat{\mathbf{U}}_m^{(a,j)}(S_n)$  can be obtained as

$$\mathbf{C}_e \left( \hat{\mathbf{U}}_{\mathcal{N}_{b,d}(T_b)+1}^{(a,j)}(S_n) \right) = N_0 \mathbf{Q}_{\mathcal{N}_{b,d}(T_b)+1}^{(a)} - N_0^2 \left( \mathbf{Q}_{\mathcal{N}_{b,d}(T_b)+1}^{(a)} \right)^2 + \mathbf{Q}_{\mathcal{N}_{b,d}(T_b)+1}^{(a)} \mathbf{D}_{\mathcal{N}_{b,d}(T_b)+1}^{(a,j)} \left( \mathbf{D}_{\mathcal{N}_{b,d}(T_b)+1}^{(a,j)} \right)^H \mathbf{Q}_{\mathcal{N}_{b,d}(T_b)+1}^{(a)}, \quad (\text{A9})$$

where  $\mathbf{Q}_{\mathcal{N}_{b,d}(T_b)+1}^{(0)}$  and  $\mathbf{D}_{\mathcal{N}_{b,d}(T_b)+1}^{(0,0)}$  can be obtained from (26) as

$$\mathbf{Q}_{\mathcal{N}_{b,d}(T_b)+1}^{(0)} = \left( \hat{\mathbf{X}}_n \hat{\mathbf{X}}_n^H + \sum_{l=n+1}^{n+N} \tilde{\mathbf{x}}[l] \tilde{\mathbf{x}}^H[l] + \sum_{l=n+N+1}^{\mathcal{N}_{b,d}(T_b)} \tilde{\mathbf{x}}[l] \tilde{\mathbf{x}}^H[l] + N_0 \mathbf{I}_{N_t} \right)^{-1}$$

$$\mathbf{D}_{\mathcal{N}_{b,d}(T_b)+1}^{(0,0)} = \mathbf{D}_{n+N+1}^{(0,0)}.$$



To resolve the detected symbols after  $n + N + 1$  in (A9),  $\mathbf{Q}_{\mathcal{N}_{b,d}(T_b)+1}^{(0)}$  is further approximated. To this end, the expectation value of  $\mathbf{Q}_{\mathcal{N}_{b,d}(T_b)+1}^{(0)}$  is used with Jensen's inequality in (A9), yielding

$$\begin{aligned} \mathbf{Q}_{\mathcal{N}_{b,d}(T_b)+1}^{(0)} &\approx \mathbb{E} \left\{ \left( \hat{\mathbf{x}}_n \hat{\mathbf{x}}_n^H + \sum_{l=n+1}^{n+N} \tilde{\mathbf{x}}[l] \tilde{\mathbf{x}}^H[l] + \sum_{l=n+N+1}^{\mathcal{N}_{b,d}(T_b)} \hat{\mathbf{x}}[l] \hat{\mathbf{x}}^H[l] + N_0 \mathbf{I}_{N_t} \right)^{-1} \right\} \\ &\geq \left( \hat{\mathbf{x}}_n \hat{\mathbf{x}}_n^H + \sum_{l=n+1}^{n+N} \tilde{\mathbf{x}}[l] \tilde{\mathbf{x}}^H[l] + (\mathcal{N}_{b,d}(T_b) - (n + N - 1) + N_0) \mathbf{I}_{N_t} \right)^{-1}, \quad (\text{A10}) \end{aligned}$$

where  $\mathbb{E}\{\hat{\mathbf{x}}[n] \hat{\mathbf{x}}^H[n]\} \approx \mathbb{E}\{\mathbf{x}[n] \mathbf{x}^H[n]\} = \mathbf{I}_{N_t}$ . Thus, by substituting (A5) and (A9) into (A3), a result in (29) is obtained where  $\mathbf{Q}_m = \mathbf{Q}_{m+1}^{(0)}$  and  $\mathbf{D}_m = \mathbf{D}_{m+1}^{(0,0)}$ .

## References

1. Foschini, G.J. Layered Space-Time Architecture for Wireless Communication in a Fading Environment When Using Multi-Element Antennas. *Bell Labs Tech. J.* **1996**, *1*, 41–59. [\[CrossRef\]](#)
2. Telatar, I.E. Capacity of Multi-Antenna Gaussian Channels. *Eur. Trans. Telecommun.* **1999**, *10*, 585–595. [\[CrossRef\]](#)
3. Zheng, L.; Tse, D.N.C. Diversity and Multiplexing: A Fundamental Tradeoff in Multiple-Antenna Channels. *IEEE Trans. Inf. Theory* **2003**, *49*, 1073–1096. [\[CrossRef\]](#)
4. Björnson, E.; Hoydis, J.; Sanguinetti, L. Massive MIMO Has Unlimited Capacity. *IEEE Trans. Wirel. Commun.* **2018**, *17*, 574–590. [\[CrossRef\]](#)
5. Larsson, E.G.; Edfors, O.; Tufvesson, F.; Marzetta, T.L. Massive MIMO for Next Generation Wireless Systems. *IEEE Commun. Mag.* **2014**, *52*, 186–1954. [\[CrossRef\]](#)
6. Lu, L.; Li, G.; Swindlehurst, A.; Ashikhmin, A.; Zhang, R. An Overview of Massive MIMO: Benefits and Challenges. *IEEE J. Sel. Top. Signal Process.* **2014**, *8*, 742–758. [\[CrossRef\]](#)
7. Simeone, O.; Bar-Ness, Y.; Spagnolini, U. Pilot-based Channel Estimation for OFDM Systems by Tracking the Delay-Subspace. *IEEE Trans. Wirel. Commun.* **2004**, *3*, 315–325. [\[CrossRef\]](#)
8. Morelli, M.; Mengali, U. A Comparison of Pilot-Aided Channel Estimation Methods for OFDM System. *IEEE Trans. Signal Process.* **2001**, *49*, 3065–3073. [\[CrossRef\]](#)
9. Kim, H.M.; Kim, D.; Kim, T.K.; Im, G.H. Frequency Domain Channel Estimation for MIMO SC-FDMA Systems with CDM Pilots. *J. Commun. Netw.* **2014**, *16*, 447–457. [\[CrossRef\]](#)
10. Biguesh, M.; Gershman, A.B. Training-based MIMO Channel Estimation: A Study of Estimator Tradeoffs and Optimal Training Signals. *IEEE Trans. Signal Process.* **2006**, *54*, 884–893. [\[CrossRef\]](#)
11. Ozdemir, M.K.; Arslan, H. Channel Estimation for Wireless OFDM Systems. *IEEE Commun. Surv. Tutor.* **2007**, *9*, 18–48. [\[CrossRef\]](#)
12. Neumann, D.; Wiese, T.; Utschick, W. Learning the MMSE Channel Estimator. *IEEE Trans. Signal Process.* **2018**, *66*, 2905–2917. [\[CrossRef\]](#)
13. Dowler, A.; Nix, A.; McGeehan, J. Data-derived Iterative Channel Estimation with Channel Tracking for a Mobile Fourth Generation Wide Area OFDM System. In Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM), San Francisco, CA, USA, 1–5 December 2003.
14. Le, H.A.; Van Chien, T.; Nguyen, T.H.; Choo, H.; Nguyen, V.D. Machine Learning-Based 5G-and-Beyond Channel Estimation for MIMO-OFDM Communication Systems. *Sensors* **2021**, *21*, 4861. [\[CrossRef\]](#) [\[PubMed\]](#)
15. Naem, M.; De Pietro, G.; Coronato, A. Application of Reinforcement Learning and Deep Learning in Multiple-Input and Multiple-Output (MIMO) Systems. *Sensors* **2022**, *22*, 309. [\[CrossRef\]](#)
16. Li, X.; Wang, Q.; Yang, H.; Ma, X. Data-Aided MIMO Channel Estimation by Clustering and Reinforcement-Learning. In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), Austin, TX, USA, 10–13 April 2022.
17. Üçüncü, A.B.; Güvensen, G.M.; Yılmaz, A.Ö. A Reduced Complexity Ungerboeck Receiver for Quantized Wideband Massive SC-MIMO. *IEEE Trans. Commun.* **2021**, *69*, 4921–4936. [\[CrossRef\]](#)
18. Yuan, J.; Ngo, H.Q.; Matthaiou, M. Machine Learning-Based Channel Prediction in Massive MIMO with Channel Aging. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 2960–2973. [\[CrossRef\]](#)
19. Zhao, M.; Shi, Z.; Reed, M.C. Iterative Turbo Channel Estimation for OFDM System over Rapid Dispersive Fading Channel. *IEEE Trans. Wirel. Commun.* **2008**, *7*, 3174–3184. [\[CrossRef\]](#)
20. Ma, J.; Ping, L. Data-Aided Channel Estimation in Large Antenna Systems. *IEEE Trans. Signal Process.* **2014**, *62*, 3111–3124.
21. Park, S.; Shim, B.; Choi, J.W. Iterative Channel Estimation Using Virtual Pilot Signals for MIMO-OFDM Systems. *IEEE Trans. Signal Process.* **2015**, *63*, 3032–3045. [\[CrossRef\]](#)
22. Huang, C.; Liu, L.; Yuen, C.; Sun, S. Iterative Channel Estimation Using LSE and Sparse Message Passing for mmWave MIMO Systems. *IEEE Trans. Signal Process.* **2018**, *67*, 245–259. [\[CrossRef\]](#)
23. Park, S.; Choi, J.W.; Seol, J.Y.; Shim, B. Expectation-Maximization-based Channel Estimation for Multiuser MIMO Systems. *IEEE Trans. Commun.* **2017**, *65*, 2397–2410. [\[CrossRef\]](#)

24. Valenti, M.C.; Woerner, B.D. Iterative Channel Estimation and Decoding of Pilot Symbol Assisted Turbo Codes Over Flat-Fading Channels. *IEEE J. Sel. Areas Commun.* **2001**, *19*, 1697–1705. [[CrossRef](#)]
25. Song, S.; Singer, A.C.; Sung, K.M. Soft Input Channel Estimation for Turbo Equalization. *IEEE Trans. Signal Process.* **2004**, *52*, 2885–2894. [[CrossRef](#)]
26. Jeon, Y.S.; Li, J.; Tavangaran, N.; Poor, H.V. Data-Aided Channel Estimator for MIMO Systems via Reinforcement Learning. In Proceedings of the IEEE International Conference on Communications (ICC), Dublin, Ireland, 7–11 June 2020.
27. Jeon, Y.S.; Lee, N.; Poor, H.V. Robust Data Detection for MIMO Systems with One-Bit ADCs: A Reinforcement Learning Approach. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 1663–1676. [[CrossRef](#)]
28. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*; The MIT Press: Cambridge, MA, USA, 2018.
29. Dong, M.; Tong, L.; Sadler, B.M. Optimal Insertion of Pilot Symbols for Transmissions over Time-Varying Flat Fading Channels. *IEEE Trans. Signal Process.* **2004**, *52*, 1403–1418. [[CrossRef](#)]
30. Kim, T.K.; Jeon, Y.S.; Min, M. Training Length Adaptation for Reinforcement Learning-Based Detection in Time-Varying Massive MIMO Systems With One-Bit ADCs. *IEEE Trans. Veh. Technol.* **2021**, *70*, 6999–7011. [[CrossRef](#)]