

Incremental Learning-based MIMO Relay Selection

Ankit Gupta¹, Mathini Sellathurai¹, Venkata V. Mani² and Tharmalingam Ratnarajah³

¹*School of Engineering and Physical Science (EPS), Heriot-Watt University, Edinburgh, UK.*

²*Department of Electronics & Communication Engineering, National Institute of Technology, Warangal, India.*

³*Institute for Digital Communications, University of Edinburgh, Edinburgh, UK.*

Abstract

The forthcoming 6G wireless networks are expected to be much more machine-intelligent in resource allocation, including relay selections to serve ever-increasing users and the internet of things with extended coverage. Selecting an optimal multiple-input multiple-output (MIMO) relay using conventional methods becomes challenging due to dependency on perfect channel information, which exponentially increases feedback overhead. In this paper, we propose a novel incremental learning-based online MIMO relay selection algorithm, with only imperfect channel gain information available at the relay nodes in the framework of MIMO two-way amplify-and-forward (TWAF) relay networks. In particular, we develop naive Bayes, logistic regression, and support vector-based incremental learning classifiers for the near-optimal online relay selection. Using simulated results, we show that the proposed online relay selection approaches outperform the best conventional Gram-Schmidt algorithm while reducing the feedback overhead up to a factor of eight.

Keywords

Incremental learning, MIMO, two-way, amplify-and-forward, relay networks, online relay selection.

1. Introduction

With the advent of the internet of everything, the multiple-input multiple-output (MIMO) network is considered a pivotal technology to meet the high data rate requirements in the upcoming sixth-generation (6G) networks. Further, relay networks will play a crucial role in the 6G networks by enhancing network reliability, data coverage, and spectral efficiency. MIMO relaying networks have been recognized to achieve significant diversity gain and significant spectrum efficiency, to expand ubiquitous coverage on land and air in the upcoming 6G networks. Further, relay selection can reduce the total network power dissipation while increasing spectral efficiency [1]. Primarily, in conventional methods, the relay selection uses the procured channel state information (CSI) knowledge. However, the channels' time-varying nature and noise make the procurement of the perfect CSI for a cooperative MIMO network increase the feedback overhead exceptionally [2]. Thus, with the wide deployment of MIMO relay networks in wireless sensor devices and the internet of things in the upcoming 6G networks, the amount of feedback overhead will increase exponentially. Furthermore, with the increased feedback overhead, the latency of the networks in selecting an even near-optimal relay increases [3]. Hence, we need

AI6G'22: First International Workshop on Artificial Intelligence in beyond 5G and 6G Wireless Networks, July 21, 2022, Padua, Italy

✉ ag104@hw.ac.uk (A. Gupta); m.sellathurai@hw.ac.uk (M. Sellathurai); vvm mani@nitw.ac.in (V. V. Mani); t.ratnarajah@ed.ac.uk (T. Ratnarajah)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

to devise intelligent relay selection algorithms that perform accurately with minimum CSI dependency and low feedback overhead. It becomes even more challenging in the two-way relaying network, which provides double spectral efficiency gains compared to a one-way relaying network, because a two-way relay node simultaneously serves the terminal node in each transmission phase.

Recently, the relay selection problem has been studied using machine learning (ML)-based algorithms in [4, 5, 6, 7]. The authors in [4] operated on the unlabelled dataset using the k-means clustering algorithm, wherein the decode-and-forward (DF) relay nodes were selected under perfect CSI. The authors in [5] extracted key features in social ties network using stacked autoencoder and utilized these features for DF relay selection. Furthermore, the authors in [6] studied two-way (TW) DF relay selection policy using an artificial neural network for a fixed and variable number of relay nodes by using distance as the feature vector. Recently the authors in [7] devised a decision tree-based one-way (OW) DF relay selection with quantized perfect CSI knowledge and show promising results if the Gini-index for the decision trees is set properly. However, all of these works have focussed only on the single-input single-output (SISO) networks, DF relaying and considered perfect estimation of the channels. Furthermore, it is well-known that amplify-and-forward (AF) relaying is widely adopted industrially over DF relaying, because of reduced implementation complexity, but at the expense of noise amplification [8].

The works in [4, 5, 6, 7] utilized offline learning-based ML algorithms that were trained once and deployed. Offline learning leads to two fundamental problems: (1) it needs an extensive amount of training data that covers all the possible testing scenarios, which is not possible with ever changing propagation environments and channel conditions, and (2) it does not continuously integrate new information to the designed ML models, rather a new ML model has to be trained from scratch. However, incremental learning (also referred to as online learning, evolving learning, constructive learning, etc.) has appeared as a paradigm shift to provide streaming ML training and data processing [9]. Once deployed, instead of training the ML models from scratch, incremental learning updates the previously trained model with the streaming new data. Thus, retraining takes place only on a small amount of data at a time (reducing training time), we do not need to store all the data (reducing data storage) and we do not need extensive initial training data covering all possible scenarios (making ML models adaptive to propagation environment). Therefore, in this work, *we design incremental learning-based MIMO AF relay selection algorithms with imperfect CSI knowledge.*

We consider MIMO TWAF relay network, where the terminal nodes intend to communicate by selecting a relay node, under correlated fading channels with imperfect CSI (Section II). We propose an incremental learning framework for the online MIMO TWAF relay selection, detail the process of retraining the ML models and prepare the datasets (Section III). We propose a naive-Bayes (NB)-based online relay selection policy by employing the pairwise algorithm (Section IV). Also, we model the stochastic gradient descent (SGD) classifiers to design and develop the support vector classifier (SVC) and logistic regression (LR)-based online relay selection algorithms (Section V). We perform extensive performance evaluation by varying signal-to-noise-ratio (SNR), channel estimation quality (CEQ), antenna correlation, number of antennas in the MIMO links and relay nodes, and to further reduce the feedback overhead we consider the *quantized* imperfect channel gain as feedback vectors (Section VI). Lastly, we conclude this work (Section VII).

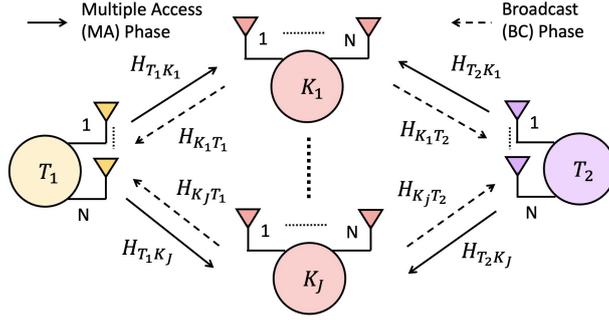


Figure 1: System model of the MIMO TWAF relay network with N antennas at each node.

2. System Model

We consider a MIMO TWAF relay network wherein the terminal nodes T_1 and T_2 intend to exchange signals by selecting a relay node $K_s \in \mathcal{K} \triangleq \{K_1, \dots, K_J\}$, as shown in Fig. 1. Each node is equipped with N antennas and the direct link is absent because of path-loss and shadow fading. For the sake of generality, let Γ and $\hat{\Gamma}$ indicate source and destination terminals, respectively, i.e., if $\Gamma = T_1$ then $\hat{\Gamma} = T_2$, and vice-versa.

2.1. Channel Model and its Estimation Quality

We employ the spatially correlated Rayleigh fading channels based on the Kronecker correlation by using the linear minimum mean squared error (LMMSE) technique for channel estimation [10]. Let the actual channel matrix be $\hat{\mathbf{H}}_{(\cdot)} \sim \mathcal{CN}(0, \sigma_h^2)$ and the transmission and reception correlation matrices [11] be $\mathbf{U}_{(\cdot)}$ and $\mathbf{R}_{(\cdot)}$, respectively, then the final channel matrix becomes $\mathbf{R}_{(\cdot)}^{1/2} \hat{\mathbf{H}}_{(\cdot)} \mathbf{U}_{(\cdot)}^{1/2}$. Let the errors in estimation be $\mathbf{E}_{(\cdot)} \sim \mathcal{CN}(0, \sigma_e^2)$ and LMMSE-based estimated channel be $\mathbf{H}_{(\cdot)} \sim \mathcal{CN}(0, \sigma_h^2)$, given by

$$\mathbf{H}_{(\cdot)} = \mathbf{R}_{(\cdot)}^{1/2} \left(\hat{\mathbf{H}}_{(\cdot)} + \mathbf{E}_{(\cdot)} \right) \mathbf{U}_{(\cdot)}^{1/2} \quad (1)$$

Without loss of generality, we assume that the error variance is dependent on SNR (γ) and indicate CEQ by δ , such that $\sigma_e^2 = \frac{\sigma_h^2}{1+\delta\gamma\sigma_h^2} \approx \frac{1}{1+\delta\gamma}$ and $\sigma_h^2 = \frac{\delta\gamma\sigma_h^2}{1+\delta\gamma\sigma_h^2} \approx \frac{\delta\gamma}{1+\delta\gamma}$ [10].

2.2. Signal Transmission Model

In multiple access phase $[T_1, T_2]$ transmit their signals to the \mathcal{K} relays, wherein the signal obtained at the j^{th} relay node becomes

$$\mathbf{y}_{K_j} = \mathbf{H}_{\Gamma K_j} \mathbf{s}_{\Gamma} + \mathbf{H}_{\hat{\Gamma} K_j} \mathbf{s}_{\hat{\Gamma}} + \mathbf{n}_{K_j} \quad (2)$$

where Γ node transmits \mathbf{s}_{Γ} with P_{Γ} transmission power, $\mathbb{E}\{\mathbf{s}_{\Gamma} \mathbf{s}_{\Gamma}^H\} = \frac{P_{\Gamma}}{N} \mathbf{I}_N$ and \mathbf{n}_{K_j} denotes noise at j^{th} relay node with $\mathbb{E}\{\mathbf{n}_{K_j} \mathbf{n}_{K_j}^H\} = \sigma_{K_j}^2 \mathbf{I}_N$. Now in broadcast phase, the j^{th} relay node

amplifies the received signal in (2) with P_R power, using amplification factor, given by

$$\alpha_j = \sqrt{P_R / \text{tr} \left\{ (P_\Gamma/N) \mathbf{H}_{\Gamma K_j} \mathbf{H}_{\Gamma K_j}^H + (P_{\hat{\Gamma}}/N) \mathbf{H}_{\hat{\Gamma} K_j} \mathbf{H}_{\hat{\Gamma} K_j}^H + \sigma_{K_j}^2 \mathbf{I}_N \right\}} \quad (3)$$

where $\text{tr}(\cdot)$ denotes trace of the matrix. Considering that the self-interference can be canceled at the terminal nodes, the received signal at terminal nodes [12] can be given by

$$\mathbf{y}_\Gamma = \sum_{j=1}^J \alpha_j \mathbf{H}_{K_j \Gamma} \left(\mathbf{H}_{\hat{\Gamma} K_j} \mathbf{s}_{\hat{\Gamma}} + \mathbf{n}_{K_j} \right) + \mathbf{n}_\Gamma \quad (4)$$

where \mathbf{n}_Γ denotes the noise with $\mathbb{E} \{ \mathbf{n}_\Gamma \mathbf{n}_\Gamma^H \} = \sigma_\Gamma^2 \mathbf{I}_N$ at Γ . The lower bound on the achievable capacity of the two-way channel links is the sum of lower bounds of the two one-way links, $C_{\Gamma \leftrightarrow \hat{\Gamma}, j} = C_{\Gamma \rightarrow \hat{\Gamma}, j} + C_{\hat{\Gamma} \rightarrow \Gamma, j}$, where the lower bound on the achievable capacity for the transmission from Γ to $\hat{\Gamma}$ terminal via j^{th} relay node is given by [12]

$$C_{\Gamma \rightarrow \hat{\Gamma}, j} = \frac{1}{2} \mathbb{E} \left\{ \log_2 \det \left(\mathbf{I}_N + \frac{\alpha_j^2 P_\Gamma}{N} \mathbf{H}_{K_j \Gamma} \mathbf{H}_{\hat{\Gamma} K_j} \mathbf{H}_{\hat{\Gamma} K_j}^H \mathbf{H}_{K_j \Gamma} \left(\alpha_j^2 \sigma_{K_j}^2 \mathbf{H}_{K_j \Gamma} \mathbf{H}_{K_j \Gamma}^H + \sigma_\Gamma^2 \mathbf{I}_N \right)^{-1} \right) \right\}$$

where $1/2$ is because of half-duplex transmission and similarly we can obtain $C_{\hat{\Gamma} \rightarrow \Gamma, j}$. We can pick the optimal relay node K_s^* that maximizes the achievable capacity by exhaustively searching (ES) over the \mathcal{K} relays, given by $K_s^* = \arg \max_{K_j \in \mathcal{K}} C_{\Gamma \leftrightarrow \hat{\Gamma}, j}, \forall K_j \in \mathcal{K}$. Nonetheless, with the increase in relays (J) and/or antennas (N) in MIMO links, ES becomes computationally very expensive, making it impractical for online relay selection purposes.

3. An Incremental Learning-Formulation for Online MIMO TWAf relay selection

In this section, we can model the TWAf relay selection problem as a multi-class classification problem, where the absolute values of the channel gains is provided as the input feature vector and J relay nodes resembles as J class labels. For an online setting, let us consider that there are $t = \{1, \dots, T\}$ discrete-time experiments conducted with p^{th} ML model L_p , before it is updated with new data online, where $p = \{1, \dots, P\}$. During any time instant t the absolute value of channel gains of MA and BC phases are provided as feature vector \mathbf{x}_t to the ML model L_p , which then predicts (select) the relay node K_{s_t} .

Training dataset creation policy – We create the dataset as follows. For the t^{th} time-instant, we formulate a $s = 4JN^2$ -dimensional feature vector (\mathbf{x}_t) containing the absolute values of the imperfect channel gains in the MA and BC phases, for all J relays, where 4 is because of CSI knowledge in dual-hop and two-phases, and N^2 is because of $N \times N$ channel between any two nodes. Correspondingly, we create the label u_t with the optimal relay node, using the ES method (detailed in Sec. 2). We repeat this process for T time intervals to create the dataset $\{\mathbf{X}_0, \mathbf{u}_0\}$. We repeat this process for P instants to create a large dataset $[\{\mathbf{X}_0, \mathbf{u}_0\}, \dots, \{\mathbf{X}_P, \mathbf{u}_P\}]$.

Traditionally, in offline learning, we train the model L_0 using dataset $\{\mathbf{X}_0, \mathbf{u}_0\}$ and deploy for future instances. However, these offline learning-based models assume that we have built the initial training set \mathbf{X}_0 covering all the possible scenarios (such as CSI, propagation environment

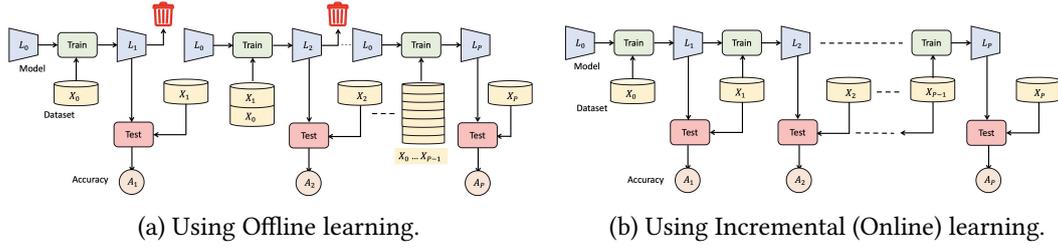


Figure 2: Retraining of the ML-based models using different learning-based approaches.

etc.) to be occurred in future, which is impractical with ever changing conditions. In practice, we need to retrain the designed ML models to keep them updated. In Fig. 2a, we depict conventional way of retraining the offline learning (OL)-based ML models and the process below:

- We train the ML model L_0 using \mathbf{X}_0 to obtain trained ML model L_1 .
- As new data \mathbf{X}_1 arrives, we test \mathbf{X}_1 on ML model L_1 and obtain the relay selection accuracy A_1 for \mathbf{X}_1 as our testing set. To update the current ML model L_1 (in an online setting) with newly obtained data \mathbf{X}_1 . We have to train the ML model L_0 from scratch using all the historical data $[\mathbf{X}_0, \mathbf{X}_1]$ stored till now to get updated ML model L_2 . And repeat this process of retraining from scratch to update the ML models in online setting.
- Consider we are at p^{th} instance with L_{p-1} ML model. And we wish to update the L_{p-1} ML model. Firstly, we need to store all the $[\mathbf{X}_0, \dots, \mathbf{X}_{p-1}]$ datasets for updating the ML model. Secondly, we have to train ML model L_0 from scratch using $[\mathbf{X}_0, \dots, \mathbf{X}_{p-1}]$ to get an updated model L_p at any p^{th} instance.

Thus, for retraining of offline learning-based ML models, we have to store the entire historical data and retrain the ML models from scratch every time. In Fig. 2b, we depict incremental learning (IL)-based ML models and the process below:

- We train the ML model L_0 using \mathbf{X}_0 to obtain trained ML model L_1 .
- As new data \mathbf{X}_1 arrives, we test \mathbf{X}_1 on ML model L_1 and obtain the relay selection accuracy A_1 for \mathbf{X}_1 as our testing set. To update the current ML model L_1 (in an online setting) with newly obtained data \mathbf{X}_1 . We retrain the model L_1 using *only* last instance testing data \mathbf{X}_1 to get updated ML model L_2 . And repeat this procedure of retraining to update the ML models in online setting.
- Consider we are at p^{th} instance with L_{p-1} ML model. And we wish to update the L_{p-1} ML model. Herein, we can retrain L_{p-1} ML model using *only* \mathbf{X}_{p-1} dataset to get an updated ML model L_p at any p^{th} instance.

Thus, in an online setting, designing ML models using incremental learning frameworks offers following advantages over retraining of offline learning based ML models:

- We do not need an exhaustive initial dataset \mathbf{X}_0 covering all the possible future testing scenarios because the incremental learning-based models are designed for retraining.
- We do not need to store-and-retrain on all the historical dataset, but just the previous instance testing dataset, reducing data storage cost and retraining time.

Now, we propose incremental learning-based ML models for online relay selection below.

4. Incremental Learning-based Naive Bayes Classifier

Naive Bayes classifier is a generative-model [13] that assumes (1) for given class labels, the attributes are conditionally independent and (2) no latent attribute impact on prediction. Herein, each feature vector \mathbf{x}_t belongs to only one class $K_j \in \mathcal{K}$, for any time instant t . Firstly, we learn the class priors (each relay's probability), given by $P(K_j) = L_{K_j}/L, \forall K_j$, where L_{K_j} and L denote the number of samples with label K_j and total samples. Secondly, for the given feature vector \mathbf{x}_t we generate a model for each label corresponding to each feature. This is done by calculating the mean $\mu_{\bar{d}, K_j}$ and standard deviation $\sigma_{\bar{d}, K_j}^2$ associated with all the classes $\in \mathcal{K}$ and for all the features $\bar{d} \in \{1, \dots, d\}$ given by a normal distribution $P(x_t^{\bar{d}}|K_j)$. Thirdly, we can calculate the conditional probability over query sample \mathbf{x}_t as $P(\mathbf{x}_t|K_j) = \prod_{\bar{d}=1}^d P(x_t^{\bar{d}}|K_j), \forall K_j \in \mathcal{K}$. Fourthly, by employing the bayes' theorem, the conditional probability of each label $K_j \in \mathcal{K}$ for the query sample $x_t^{\bar{d}}$ can be decomposed as

$$P(K_j|\mathbf{x}_t) = \frac{P(\mathbf{x}_t|K_j) P(K_j)}{\sum_{l=1, l \neq K_j}^J P(\mathbf{x}_t|l) P(l)} \quad (5)$$

Lastly, the naive Bayes classifier combines independent feature model obtained above with a decision rule like maximum a posteriori, that determines the selected relay $K_{s_t} = \arg \max_{K_j \in \mathcal{K}} P(K_j|\mathbf{x}_t)$. Please note we employ the pairwise algorithm [13] for updating the mean and standard deviation of the features incrementally.

5. Incremental Learning-based SGD (SVC and LR) Classifier

We now show the SGD classifier [14]-based formulation for linear SVC [15] and LR [16]-based online relay selection. We form a binary classifier where $\mathbf{y}_t \in \{-1, 1\}$ and our aim is to learn a linear scoring function $f(\mathbf{x}_t) = \mathbf{w}_t^\dagger \mathbf{x}_t + b_t$, wherein \mathbf{w}_t and b_t denotes the weight and bias at t^{th} time-instant. Further, the predictions for the binary classification is done by checking the sign of the scoring function $f(\mathbf{x}_t)$ and we aim to minimize:

$$E(\mathbf{w}_t, b_t) = \frac{\alpha}{2} \mathbf{w}_t^\dagger \mathbf{w}_t + \frac{1}{L} \sum_{i=1}^L \mathcal{L}(y_t^{(i)}, f(\mathbf{x}_t^{(i)})) \quad (6)$$

where $\alpha > 0$ controls the regularization strength. The loss function $\mathcal{L}(\cdot)$ can be defined as:

- *Hinge-loss for SVC classifier* [15]: $\mathcal{L}(y_t^{(i)}, f(\mathbf{x}_t^{(i)})) = \max(0, 1 - y_t^{(i)} f(\mathbf{x}_t^{(i)}))$.
- *Log-loss for LR classifier* [16]: $\mathcal{L}(y_t^{(i)}, f(\mathbf{x}_t^{(i)})) = \log(1 + \exp(-y_t^{(i)} f(\mathbf{x}_t^{(i)})))$

The TWAF MIMO relay selection is a multi-class classification (MCC) problem, thus we use one-vs-all classifier to implement the J binary classifiers for each relay (class) to find R_{s_t} . Moreover, the first order routine for SGD learning is applied for updating the weights as

$$\mathbf{w}_t \leftarrow \mathbf{w}_t - \eta_t \left[\alpha \frac{\partial \mathcal{V}(\mathbf{w}_t)}{\partial \mathbf{w}_t} + \frac{\partial \mathcal{L}(\mathbf{w}_t^\dagger \mathbf{x}_t + b_t, y_t^{(i)})}{\partial \mathbf{w}_t} \right] \quad (7)$$

Similarly, intercept term is updated and learning rate η_t is gradually decayed with time [14].

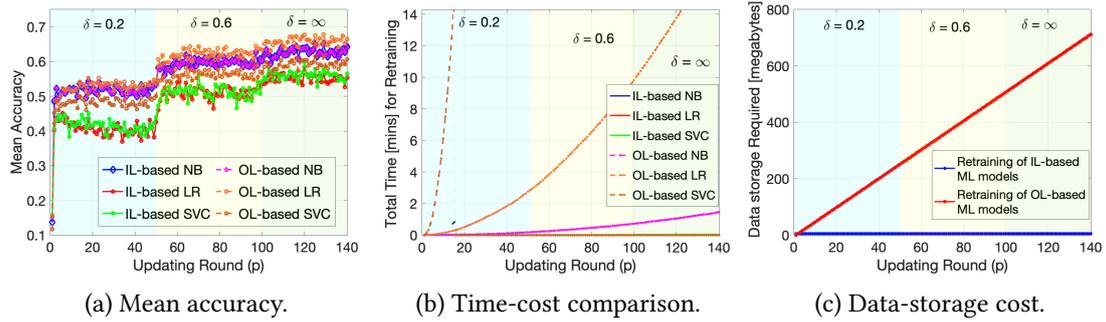


Figure 3: Comparison of retraining for OL and IL-based ML relay selection for SNR = 10 dB, $\rho = 0.8$, $J = 5$, and $N = 4$. Note that both OL and IL algorithms are re-trained (updated) for each p^{th} round, according to the policies detailed in Sec. 3.

6. Performance Evaluation

We keep the number of ML model updates (retraining) as $p = 50$ and utilize Rayleigh fading channels with the corresponding CEQ. We consider following benchmark algorithms – (1) *Exhaustive Search-based optimal Relay Selection (ES RS)* – Detailed in Sec. 2, (2) *Gram-Schmidt-based Relay Selection (GS RS)* [12] – Conventionally, GS-based relay selection policy performs the best, wherein we select the relay by $R_s = \arg \max_{K_j \in \mathcal{K}} \prod_{\Gamma=T_1}^{T_2} \det(\mathbf{H}_{K_j \Gamma}^H \mathbf{H}_{K_j \Gamma}) \det(\mathbf{H}_{\hat{\Gamma} K_j}^H \mathbf{H}_{\hat{\Gamma} K_j})$, and (3) *Random RS* – We choose relay $R_s \in \mathcal{K}$ at random.

6.1. Retraining of the ML Models Developed via Incremental-learning (IL) versus Offline Learning (OL)

We compare the retraining of the proposed relay selection ML models (SVC, NB, LR) designed via offline and incremental learning in Fig. 3. We vary the CEQ (δ) after every 50 rounds (i.e. for $p \leq 50$ the $\delta = 0.2$, for $50 < p \leq 100$ the $\delta = 0.6$ and for $p > 100$ the $\delta = \infty$), where $\delta = 0$ indicates fully erroneous channel estimation and $\delta = \infty$ denotes perfect channel estimation. In Fig. 3a, the accuracy of optimal relay selection via retraining of OL-based ML models is slightly better than the proposed IL approach. This is because retraining of OL-based ML models utilizes all the prior historical data, while retraining in IL takes place using only the previous instant of data. This also explains the reason behind exponential increase in time-cost and linear increase in data-storage cost for retraining OL algorithms, compared to the time-cost of few seconds and data-storage cost of few kilobytes for IL algorithms in Fig. 3b and Fig. 3c, respectively. Thus, IL solves the two fundamental problems (of data storage and re-training time) faced in deploying OL algorithms in practical settings. Hence, providing us a method to deploy the ML algorithms for real-world applications, where re-training the ML algorithms is inevitable.

6.2. Limited Feedback Scenarios and Time-Cost Analysis

Each real-value feedback requires 8 bits of information, thus, we focus on feedback overhead by considering the quantized feedback of the channel gains. We divide the channel gains into

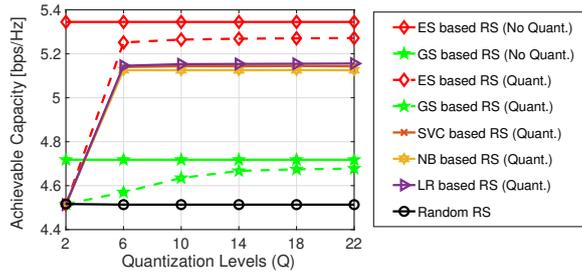


Figure 4: Impact of varying CSI quantization levels (SNR = 10 dB, $\rho = 0.8$, $\delta = 0.35$, $J = 10$, $N = 5$).

Table 1

Comparison of relay selection algorithms.

RS Algorithms	Time-cost [ms]	Complexity	Feedback Overhead
ES (Optimal upper bound)	0.779	$\mathcal{O}(16JN^3 + J)$	$4JN^2$ complex-values = $2 \times 4JN^2 \times 8$ bits
GS (Conventional best)	0.132	$\mathcal{O}(4JN^3 + J)$	
IL-based NB (Proposed)	0.023	$\mathcal{O}(4JN^2)$	No Quant.: $4JN^2$ real-values = $4JN^2 \times 8$ bits. Quant. levels ($Q = 4$): $4JN^2 \times 2$ bits
IL-based SVC (Proposed)	0.007	$\mathcal{O}(4JN^2)$	
IL-based LR (Proposed)	0.036	$\mathcal{O}(4JN^2 + 1)$	

Q levels that can be represented as $Q = 2^b$, where b denotes the number of bits required for feedback for corresponding Q levels. In Fig. 4, for benchmark, we consider the performance of ES and GS relay selection when no quantization is performed, i.e., complex-valued CSI feedback. The ES and GS with the quantized channel gains are not able to reach the achievable capacity of ES and GS with no quantization, even with 22 levels, because there is no phase information in the feedback. Also, incremental learning-based relay selection algorithms outperform the conventional best GS with complex-value feedback information (16 bits feedback) once $Q \geq 4$ (2 bits feedback) and achieves performance very close to ES for $Q = 6$ quantization levels (3 bits feedback), showing the merits of proposed incremental learning. We analyse the computational complexity, time-cost and feedback overhead for all the proposed relay selection algorithms in Table 1 for $J = 5$ and $N = 4$. Clearly, ES-based relay selection takes the most time, followed by GS approach. Also, SVC reduces time-cost by 95% compared to GS relay selection.

7. Conclusion

In this work, we proposed incremental learning-based (NB, SVC, LR) relay selection models for the MIMO TWAF networks with imperfect CSI. We showed that incremental learning-based retraining can help us in reducing the time-cost and data-storage-cost exponentially, while achieving similar performance gains, as the retraining of offline ML models. Further, the proposed incremental learning-based relay selection algorithms (using 2 bits feedback) achieve similar performance as conventional best GS algorithm (using 16 bits feedback) and can achieve performance close to optimal brute-force ES algorithm with 3 bits feedback. Also, SVC-based relay selection reduces the time-cost by 95% compared to GS-based relay selection.

Acknowledgments

We gratefully acknowledge the COG-MHEAR: Towards cognitively-inspired 5G IoT enabled, multi-modal Hearing Aids (<https://cogmhear.org>) under Grant EP/T021063/1, the Ministry of Human Resource Development Government of India for awarding the grant under SPARC, India (2019/249) and Grant SERB IMRC/AISTDF/CRD/2019/000178 for the support of this work.

References

- [1] Y. Zhao, R. Adve, T. J. Lim, Improving amplify-and-forward relay networks: Optimal power allocation versus selection, in: 2006 IEEE International Symposium on Information Theory, 2006, pp. 1234–1238.
- [2] A. Bletsas, et. al., A simple cooperative diversity method based on network path selection, *IEEE Journal on Selected Areas in Communications* 24 (2006) 659–672.
- [3] D. Love, R. Heath, W. Santipach, M. Honig, What is the value of limited feedback for mimo channels?, *IEEE Communications Magazine* 42 (2004) 54–59.
- [4] W. Song, F. Zeng, J. Hu, Z. Wang, X. Mao, An unsupervised-learning-based method for multi-hop wireless broadcast relay selection in urban vehicular networks, in: 2017 IEEE 85th Vehicular Technology Conference (VTC Spring), 2017, pp. 1–5.
- [5] P. Zhang, et. al., Overlapping community deep exploring-based relay selection method toward multi-hop d2d communication, *IEEE Wireless Commun. Letters* 8 (2019) 1357–1360.
- [6] Z. Zhang, et. al., Neural network-based relay selection in two-way swipt-enabled cognitive radio networks, *IEEE Transactions on Vehicular Technology* 69 (2020) 6264–6274.
- [7] X. Wang, F. Liu, Data-driven relay selection for physical-layer security: A decision tree approach, *IEEE Access* 8 (2020) 12105–12116.
- [8] A. Gupta, K. Singh, M. Sellathurai, Time-switching eh-based joint relay selection and resource allocation algorithms for multi-user multi-carrier af relay networks, *IEEE Transactions on Green Communications and Networking* 3 (2019) 505–522.
- [9] J. Wang, *Encyclopedia Of Data Warehousing And Mining*, IGI Global, USA, 2005.
- [10] A. R. Heidarpour, et. al., Network coded cooperation based on relay selection with imperfect csi, in: 2017 IEEE 86th Vehicular Technology Conference (VTC-Fall), 2017, pp. 1–5.
- [11] G. Liu, H. Ji, Y. Li, X. Zhang, Sum rate maximization antenna selection via discrete stochastic approximation in mimo two-way af relay with imperfect csi, in: 2012 IEEE Global Communications Conference (GLOBECOM), 2012, pp. 2487–2492.
- [12] C.-C. Hu, B.-H. Chen, Two-way mimo relaying systems employing layered relay-and-antenna selection strategies, *IEEE Systems Journal* 12 (2018) 854–861.
- [13] T. F. Chan, G. H. Golub, R. J. LeVeque, Updating Formulae and a Pairwise Algorithm for Computing Sample Variances, Technical Report, Stanford, CA, USA, 1979.
- [14] L. Bottou, Large-scale machine learning with stochastic gradient descent, in: *Proceedings of COMPSTAT'2010*, Physica-Verlag HD, Heidelberg, 2010, pp. 177–186.
- [15] C.-W. Hsu, C.-C. Chang, C.-J. Lin, *A Practical Guide to Support Vector Classification*, Technical Report, Department of Computer Science, National Taiwan University, 2003.
- [16] D. W. Hosmer, S. Lemeshow, *Applied logistic regression*, John Wiley and Sons, 2000.