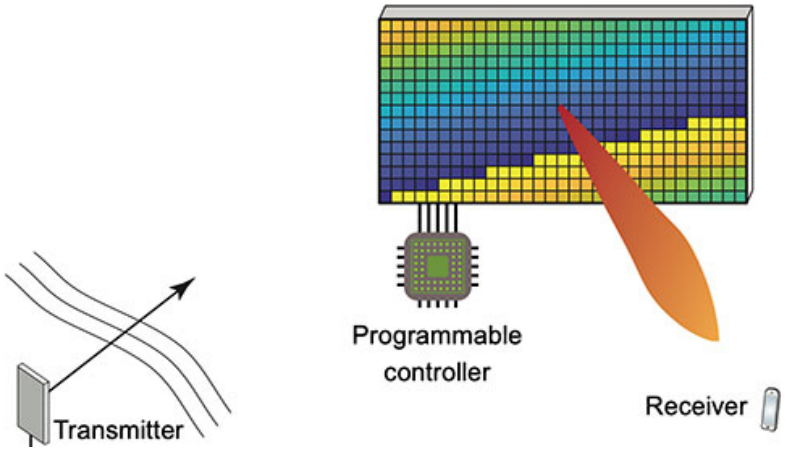
IEEE Signal Processing Cup 2021

Team SIPL Report

Submitted By:	*email:*
Tomer Fireaizen	tomerf@campus.technion.ac.il
Dan Ben-David	bdan-ilan@campus.technion.ac.il
Shaked Hadad	hadad.shaked@campus.technion.ac.il
Nir Kurland	nirkurland@campus.technion.ac.il
Sima Etkind	sima.etkind@campus.technion.ac.il
Tutor:	
Yair Moshe	myair@technion.ac.il
Mentor:	
Israel Cohen	icohen@ee.technion.ac.il

May, 2021



Signal and Image Processing Laboratory (SIPL)
Andrew and Erna Viterbi Faculty of Electrical and Computer Engineering
Technion – Israel Institute of Technology, Haifa, Israel
https://sipl.eelabs.technion.ac.il

ABSTRACT

This document describes Team SIPL's solution for the IEEE Signal Processing Cup 2021 [1], in which competing teams were required to configure an Intelligent Reflecting Surface (IRS) for efficient wireless communications. To that end, we first estimated the wireless channel and then designed an algorithm that finds an IRS configuration that maximizes the rate of that channel. We began with the provided far-from-optimal IRS configurations and applied an iterative optimization technique based on gradient descent and adaptive quantization. Further optimization is obtained by training a deep generative neural network to find a configuration that maximizes the rate function. Each of these techniques allowed us to discover better configurations and increase data rates from the base station to the provided users. The best configurations we have discovered provide a significant improvement of the weighted average rate from 104.07 Mbit/s to 120.70 Mbit/s, compared to the best provided configurations over all users. Non IRS based solution provides average rate of 4.38 Mbit/s, which is two orders of magnitude below the IRS baseline. The code of our solution will be available at `https://github.com/BIueMan/IEEE_SP_CUP_2021-SIPL_TEAM`.

1. INTRODUCTION

An Intelligent Reflecting Surface (IRS) is a two-dimensional array of metamaterial whose interaction with electromagnetic waves can be controlled. It consists of an array of discrete passive elements that can be controlled to alter the amplitude and/or phase of the reflected signal, thereby smartly reconfigure the wireless propagation environment. The goal of the IEEE Signal Processing Cup 2021 challenge is to characterize the behaviour of an IRS based on received signals from an over-the-air signalling phase and develop a control algorithm to configure the surface to increase communication performance [1].

The relation in the frequency domain between a transmitted signal $\boldsymbol{x}$ and the received signal $\boldsymbol{z}$ can be modeled as

$$\boldsymbol{z} = \boldsymbol{h}_\theta \odot \boldsymbol{x} + \boldsymbol{w}, \tag{1}$$

where $\boldsymbol{h}_\theta$ is the frequency response of the system with a specific IRS configuration $\boldsymbol{\theta}$, $\boldsymbol{w}$ is the receiver noise, and $\odot$ is the Hadamard (element-wise) multiplication. The transmission over the communication channel is carried out using orthogonal frequency-division multiplexing (OFDM) with finite impulse response (FIR) filter of order M and $K > M$ subcarriers. A more detailed overview of fundamental properties of IRS technology from a signal processing perspective is given in [2].

Evaluation is based on the users that are considered in the provided "dataset2". The transmitting base station and the IRS are at fixed locations and have a line-of-sight (LOS) channel between them. All users have non-line-of-sight (NLOS) channels to the base station. Some users have LOS channels from the IRS, other do not. The weighted average data rate over all user with the submitted configurations is computed according to [1] with double weight for NLOS users. The noise power spectral density is required to compute the rate, thus we should estimate the wireless channel for each user. "Dataset1" contains more data and is focused on a single user. Therefore, it can be utilized to learn channel-related and IRS-related channel properties, and to develop a technique for channel estimation. In Section 2 we describe our technique for channel estimation. This technique uses the geometrical shape of the IRS as described in Section 4.

Finding the global optimal configuration is mathematically intractable. With the growing interest in this technology, many heuristic solutions have recently been proposed. One notable solution is based on strongest tap maximization (STM) [3][4]. However, most of these solutions, including the STM, are inapplicable in this challenge as they provide a transmission protocol, whereas we are given in this challenge a pre-determined pilot signal and a pre-determined small set of IRS configurations. Furthermore, most of these solutions assume a continuous phase or a set of discrete phases, whereas we are constrained to only two phases with a phase-shift difference between the elements of approximately π. Our solution for finding a good IRS configuration under these constraints is described in Section 3. In Section 5, we present the technique we have developed to discover the spatial arrangement of the users.

2. CHANNEL ESTIMATION

The channel we estimate is used to calculate the data rate. Accurate channel estimation is critical for finding a good configuration because estimating the channel inaccurately may result in optimizing the wrong objective function. We model the system by

$$\boldsymbol{z} = (\boldsymbol{h}_d + \boldsymbol{V}\boldsymbol{\theta}) \odot \boldsymbol{x} + \boldsymbol{w}, \tag{2}$$

where $\boldsymbol{x}$, $\boldsymbol{z}$, $\boldsymbol{h}_d$ and $\boldsymbol{w}$ are vectors of size $K \times 1$ representing the transmitted signal, the received signal, the direct channel and the noise, respectively. $\boldsymbol{V}$ is a $K \times N$ matrix representing the cascade of the channel from the transmitter to the IRS and the channel from the IRS to the receiver, and $\boldsymbol{\theta}$ is a vector of size $N \times 1$ representing the configuration of the IRS.

During the pilot transmission, all elements of $\boldsymbol{x}$ are equal to a scalar value $x_i \equiv \alpha$. Dataset2 consists of a set of N configurations $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_N$ that form the columns of an $N \times N$ Hadamard matrix. Dataset1 consists of a set of $4N$ configurations such that each configuration from dataset2 and its negation appear twice in the $N \times 4N$ configuration matrix of this dataset. Thus, summation over all received signals in dataset1 yields $4N\boldsymbol{h}_d \odot \boldsymbol{x} + \sum_{i=1}^{4N} \boldsymbol{w}_i$, and we can find the direct channel by dividing this expression by $4N\alpha$. Summing over all N received signals of a single user in dataset2 we get $(N\boldsymbol{h}_d + \boldsymbol{V}\sum_{i=1}^{N}\boldsymbol{\theta}_i) \odot x + \sum_{i=1}^{N}\boldsymbol{w}_i$, where $\sum_{i=1}^{N}\boldsymbol{\theta}_i = [N, 0, .., 0]^T$ is a vector of size N. This expression gives us

the sum of the direct channel and the response of the first element of the IRS to the +1 state. Since the first element is negligible (our IRS has $N = 4096$ elements) we estimate the direct channel of a single user in dataset2 by

$$\hat{\boldsymbol{h}}_d = \frac{\sum_{i=1}^{N} \boldsymbol{z}_i}{N\alpha} = \boldsymbol{h}_d + \boldsymbol{V} \cdot [1, 0, .., 0]^T + \frac{\sum_{i=1}^{N} \boldsymbol{w}_i}{N\alpha}. \quad (3)$$

After estimating the direct channel $\boldsymbol{h}_d$ we can estimate the non-direct channel $\boldsymbol{V}$. Since the provided pilot matrix on dataset2 is invertible, we can use the least-square (LS) estimator that is given by

$$\boldsymbol{V} = (\boldsymbol{Z} - \boldsymbol{X}\,\boldsymbol{H}_d)\,\boldsymbol{\Omega}^{-1}/\alpha, \quad (4)$$

where $\boldsymbol{X} = \alpha I_{K\times K}$ is a $K \times K$ matrix of the transmitted signal (the scalar α), $\boldsymbol{Z}$ is a $K \times N$ matrix whose columns are the received signals, $\boldsymbol{H}_d$ is a $K \times N$ matrix whose columns are the direct channels $\boldsymbol{h}_d$, and $\boldsymbol{\Omega}$ is the pilot matrix.

We used our discovery of the array's geometrical shape (described in Section 4) to compensate for the addition of the response of the first element of the IRS to the +1 state to the estimated channel. We assume, and verified this on dataset1, that the MSE between the frequency response of each element i and the frequency response of elements $i+64n$ (elements in the same column) is small. So, we take the first column of $\boldsymbol{V}$ to be $\boldsymbol{V}(:,1) = \frac{1}{63}\sum_{n=1}^{63} \boldsymbol{V}(:, 1 + 64 \cdot n)$. We also perform a similar compensation for the direct channel $\boldsymbol{h}_d$.

LS-based channel estimation is simple to implement and does not require any a-priori knowledge of channel statistics. However, its performance is not as good as more advanced channel estimation techniques. Therefore, we applied a denoising strategy inspired by [5]. We have the channel in the frequency domain. This channel in the time domain is a vector of size K with small non-zero values at its $[M + 1, .., K]$ elements. Since the channel in the time domain is modeled as an FIR filter of order M, we assume that the elements $[M + 1, .., K]$ in the channel vector are noise, and set them to zero before returning to the frequency domain. The estimator we get from this process yields zero MSE on the N configurations provided in dataset2. When verified on dataset1, this estimator resulted in the smallest MSE of any other channel estimation methods we tried. Figure 1 shows the verification of our channel estimation process on dataset1. Both channel estimators from N configurations follow closely the amplitude and phase of the channel computed from $4N$ configurations. The channel estimated after denoising is smoother.

Note that we also tried to perform channel estimation using the Khatri-Rao factorization (KRF) algorithm to obtain separate estimates of the base-station-to-IRS channel and IRS-to-user channel via rank-1 approximation steps [6]. The estimation of the IRS-to-user channel resulted in a matrix with constant columns since in our case of a fixed transmitted pilot signal, we can represent the cascaded transfer function as a single matrix multiplying from the left a vector of configuration $\boldsymbol{\theta}$, just like we have showed earlier. As a result, we did not use this technique for our final submission.

3. CONFIGURATION SEARCH

To find a configuration that maximizes the data rate, we used two techniques. The first is based on gradient descent and adaptive quantization, while the second uses a deep generative neural network. Each technique resulted in one configuration, and for each user we chose the one with the highest rate.

In the first technique, we try to maximize the rate $\mathcal{R}$ using the gradient descent algorithm by defining $\nabla_{\vec{\theta}}\mathcal{R} = \frac{\partial \mathcal{R}}{\partial \vec{\theta}}$ and an update rule

$$\vec{\theta}^{t+1} = \exp\left[\angle\theta^t + \frac{\angle\nabla_{\vec{\theta}}\mathcal{R}}{\left\|\nabla_{\vec{\theta}}\mathcal{R}\right\|}\right] \quad (5)$$

where $\angle\theta$ is the phase of the complex vector $\theta \in \mathcal{C}^{4096}$. After several iterations, the optimization converges to a local maximum that serves as the initial value for an iterative optimization and quantization algorithm. We quantize the phase into two values by finding the best line for separating the phases on the unit circle for each user. An example of such separating line for phase quantization is depicted in Figure 2. Let the best separating line be $[\tilde{\phi}, \tilde{\phi}+\pi]$. Then, for all element phases calculated using the STM estimator, we calculate their cyclic distance from the separating line: $\mathcal{D} \equiv \sin^2(\angle\theta - \tilde{\phi})$. This metric assigns a lower distance to elements that have more ambiguous clustering. The iterative process is as follows:

Algorithm 1: Iterative Optimization and Quantization

Result: IRS configuration
repeat
  Calculate $\mathcal{D}(\angle\theta)$
  $e = \text{argmin}(\mathcal{D}(\angle\theta))$
  Create two assumptions - e is either $+1$ or -1.
  For each assumption, run gradient descent optimization for all unfixed elements until convergence.
  Fix e to the value that yields higher data rate.
until *All IRS elements are fixed*;

The selection of more ambiguous elements first is based on the assumption that these elements are more likely to be assigned an incorrect phase. Therefore, as the algorithm proceed, it is more likely to converge to the optimal IRS configuration.

To prevent the above algorithm from entering a local maximum, at each 4^n step of the algorithm, where $n \in \{1, 2, ... \log_4(4096)\}$, we check each element to see if its phase change gives higher rate. Another means for preventing entry to a local maximum is to use a probabilistic method at the end of the algorithm. Each element is assigned a probability that is inversely proportional to the distance from its phase value assigned by the continuous phase STM estimator. We then select a random configuration based on these probabilities. If the new configuration has higher rate than the original,

the new configuration is set as the base configuration for an additional random iteration (with the original probabilities). This process is repeated a fixed number of iterations.

The second technique we use for configuration search generates IRS configurations using a deep generative neural network. The design of this network is based on two insights. First, it was recently shown [7] that a deep neural network can be used as a regulator. Therefore, we define our loss function to be minus the rate, and train the network to overfit for one specific user. Second, we noticed that LOS users achieve high data rates with very smooth, columns-like, configurations. We conclude that the steering components of the configurations are the most important, especially for LOS users. Therefore, we divide the generative architecture into two sub-networks, as shown in Figures 3 and 4. The first sub-network receives a random scalar as input and generates a steering configuration (according the the geometrical shape of the array, see Section 4), while the second sub-network receives this steering configuration as input and generates another, not necessarily steering, final configuration. The purpose of the second sub-network is to fine tune the steering configuration by adding elevation components. The first sub-network consists of five fully-connected layers and one hyperbolic tangent activation function layer. This is not equivalent to having a single larger fully-connected layer, as shown in [8]. The resulting vector of size 64×1 represents the columns of a steering configuration and is replicated to create a 64×64 matrix that is the input to the second sub-network. The second sub-network is fully convolutional neural network with intermediate layers of hyperbolic tangent activation function that generates the final configuration as vector of size 4096×1. For training the sub-networks, we used a cosine annealing learning rate. Compared to our first configuration search technique using gradient descent and adaptive quantization, the generative network converges faster, is computationally more efficient and typically discovers configurations with higher data rates for LOS users. However, for NLOS users, the first technique is usually better.

It should be noted that we also tried to combine our two configuration search techniques by decomposing the configuration into its steering and elevation components, as depicted in Figure 5. We extracted the elevation component from the best configuration discovered by the first technique, and re-estimated the steering component using the first generative sub-network from our second technique. Then, we summed the two components to obtain the final configuration. This technique was effective for several NLOS users.

Figure 6 shows the data rate for each user with the best IRS configuration we found compared to the data rate obtained by the best configuration provided in dataset2 and to the data rate obtained by STM (with continuous phase). Users at the top of the graph have LOS with the IRS, while users at the bottom do not. Figure 7 shows examples of the best configurations discovered by our proposed solution. Users in LOS with the IRS have best configurations that are almost pure steering, while users with no LOS with the IRS have more intricate patterns in their best configurations.

4. GEOMETRICAL SHAPE OF THE IRS

Knowing the geometrical shape of the IRS allows to utilize the spatial correlation between elements as described in Section 3. The IRS is rectangular and has $N = 4096$ elements. Hence, it can have one of the sizes $1 \times 4096, 2 \times 2048, ..., 4096 \times 1$. We assume that for users in LOS with the IRS, a pure steering configuration will provide a high data rate. We noticed that these steering configurations for LOS users have many columns of the same value along the y-axis. Thus, we have implemented a version of STM that allows only steering configurations. We then used this version to find the best steering configuration for all LOS users for each possible geometrical shape of the array, and compared the rates obtained by the different geometrical shapes. The results of this experiment are depicted in Figure 8. From this figure we can conclude that the physical shape of the IRS is 64×64 elements.

5. SPATIAL RELATION BETWEEN USERS

There may be common scattering clusters that affect multiple users [1]. Discovering these clusters may allow us to estimate the channel between the base station and multiple users together and thus improve estimation accuracy. To this end, we calculate the data rate obtained by each user for each of the N configurations provided in dataset2. We then calculate for each two users the cross correlation between their rate vectors and use it as a measure of similarity between the users. We use the k-means algorithm to cluster the users based on this similarity measure. The number of clusters k was determined using the 'elbow method' to be 13. Figures 9 and 10 visualize the spatial arrangement of all 50 users according to this measure of similarity using the multidimensional scaling (MDS) [9] and t-SNE [10] algorithms, respectively. Note that the visualization is performed in two dimensions, but clustering is performed in the original high dimension.

We utilized the scattering clusters by calculating a new configuration for each user based on all of the other users in the same cluster and using this configuration as the initial configuration for Algorithm 1. The value of each element in the new configuration is the majority vote of that element from the best configurations of all users in the same cluster. If the cluster has an even cardinality, the considered user is double weighted. This technique allowed us to discover several new configurations with slightly increased data rates.

6. REFERENCES

[1] E. Björnson, "IEEE Signal Processing Cup 2021 - Configuring an Intelligent Reflecting Surface for Wireless Communications," 2021. [Online]. Available: https://github.com/emilbjornson/SP_Cup_2021

[2] E. Björnson, H. Wymeersch, B. Matthiesen, P. Popovski, L. Sanguinetti, and E. de Carvalho, "Reconfigurable intelligent surfaces: A signal processing perspective with wireless applications," *arXiv preprint arXiv:2102.00742*, 2021.

[3] B. Zheng and R. Zhang, "Intelligent reflecting surface-enhanced OFDM: Channel estimation and reflection optimization," *IEEE Wireless Communications Letters*, vol. 9, no. 4, pp. 518–522, 2019.

[4] S. Lin, B. Zheng, G. C. Alexandropoulos, M. Wen, F. Chen *et al.*, "Adaptive transmission for reconfigurable intelligent surface-assisted OFDM wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 11, pp. 2653–2665, 2020.

[5] P. Sure and C. M. Bhuma, "A survey on OFDM channel estimation techniques based on denoising strategies," *Engineering Science and Technology, an International Journal*, vol. 20, no. 2, pp. 629–636, 2017.

[6] L. C. d. P. P. Gilderlan T. de Araújo and A. L. F. de Almeida, "Channel Estimation for MIMO System Assisted by Intelligent Reflective Surface," 2020.

[7] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9446–9454.

[8] S. Bell-Kligler, A. Shocher, and M. Irani, "Blind super-resolution kernel estimation using an internal-gan," *arXiv preprint arXiv:1909.06581*, 2019.

[9] A. Buja, D. F. Swayne, M. L. Littman, N. Dean, H. Hofmann, and L. Chen, "Data visualization with multidimensional scaling," *Journal of Computational and Graphical Statistics*, vol. 17, no. 2, pp. 444–472, 2008.

[10] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of machine learning research*, vol. 9, no. 11, 2008.

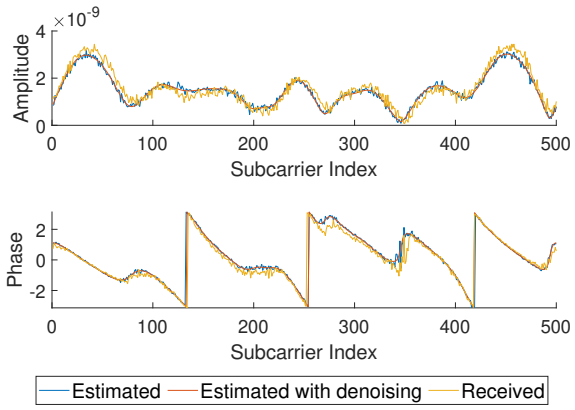


Fig. 1. Verification of our channel estimation process from N configurations, with and without denoising, with the $4N$ configurations of dataset1.

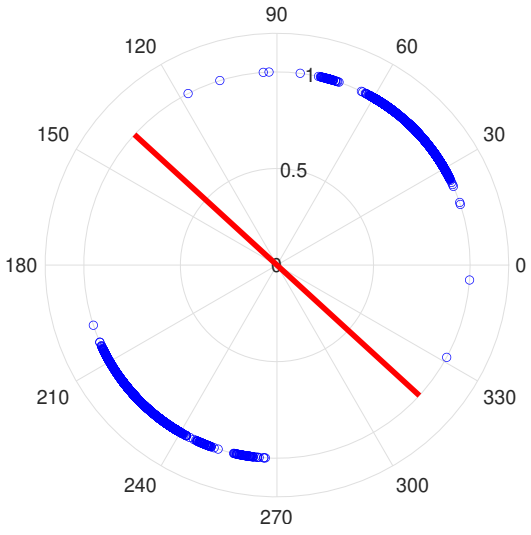


Fig. 2. An example of phase quantization for the resulting STM configuration for user 20. The best separating line is drawn in red.

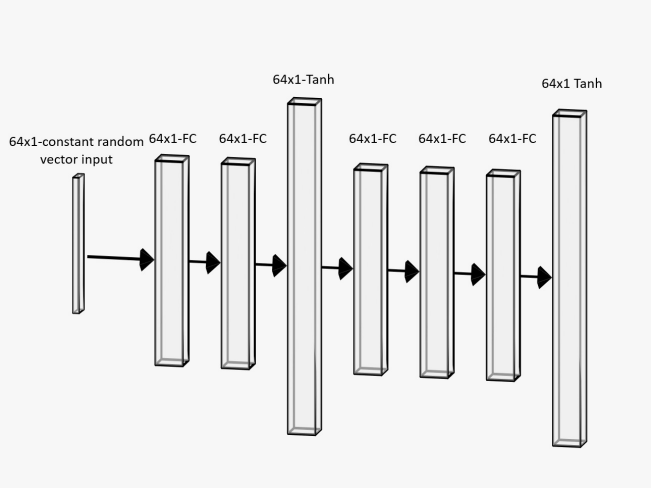


Fig. 3. First sub-network of the proposed generative neural network architecture. It receives a random scalar (in vector form) as input and generates a steering configuration (in vector form).

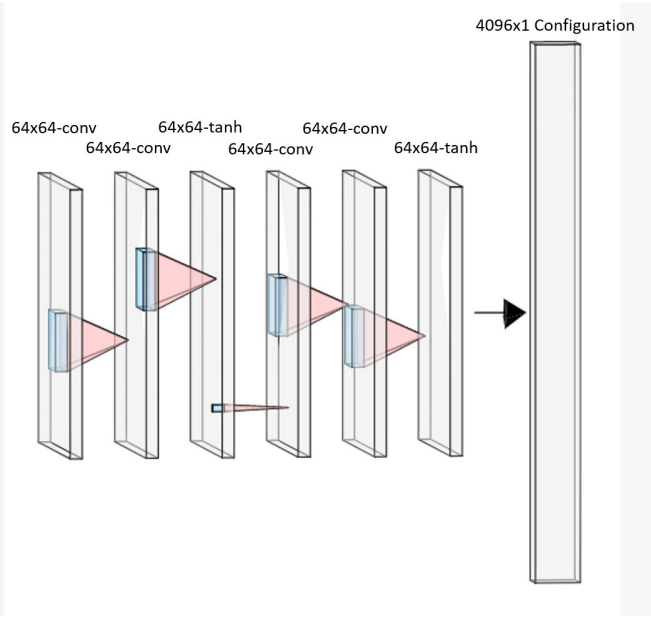


Fig. 4. Second sub-network of the proposed generative neural network architecture. It receives a steering configuration as input and generates another, not necessarily steering, configuration (in vector form).

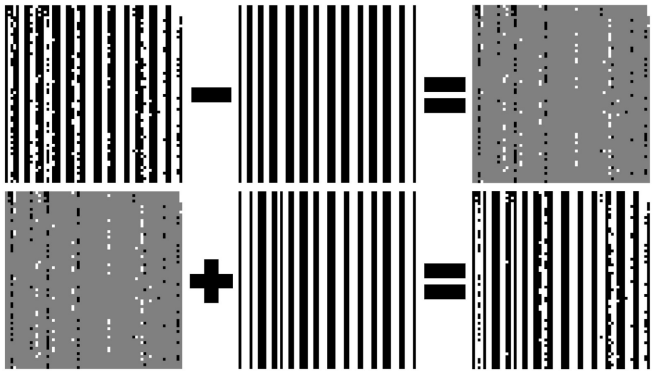


Fig. 5. Combination of our two configuration search techniques - we extract the elevation component from the best configuration discovered by the first technique, and re-estimated the steering component using the first generative sub-network from our second technique. Then, we sum the two components to obtain the final configuration.

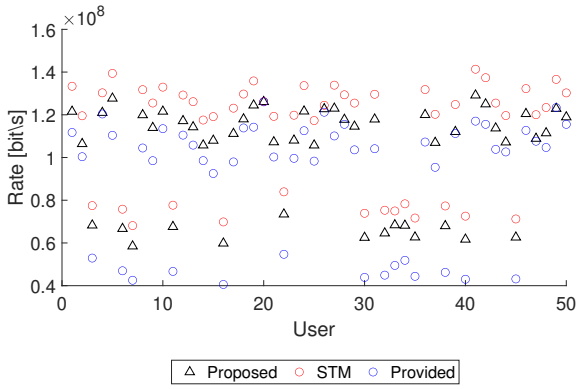


Fig. 6. Data rates obtained for each user with the best IRS configuration in our proposed solution compared to the data rates obtained by the best configuration provided in dataset2 and to the data rate obtained by STM (with continuous phase). The 36 users at the top of the graph have LOS with the IRS, while other 14 users at the bottom do not.

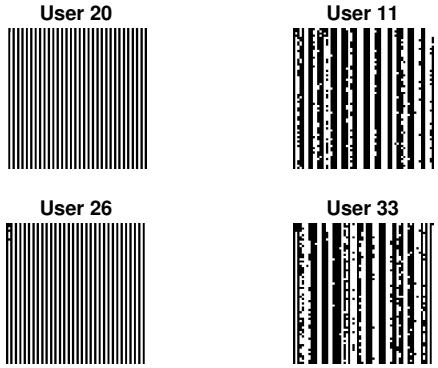


Fig. 7. An example of the best configurations discovered by our proposed solution. Both users on the left have LOS with the IRS so their best configuration is almost pure steering. Both users on the right do not have LOS with the IRS so the patterns in their best configuration are more intricate.

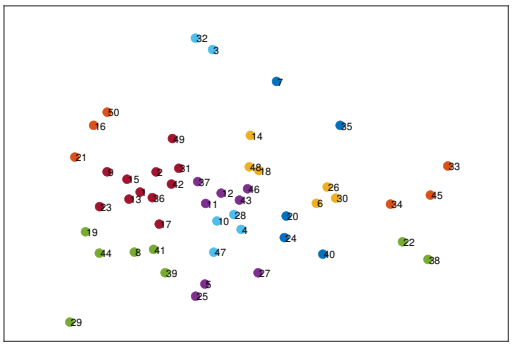


Fig. 9. Spatial arrangement of all users using the multidimensional scaling (MDS). Different colors represent different user clusters.

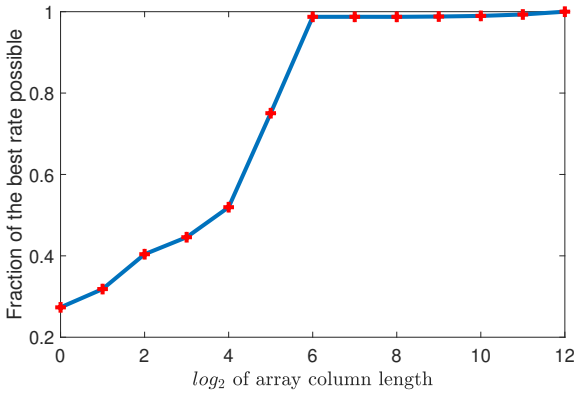


Fig. 8. Average data rate for all LOS users as a function of the IRS array column length. The best possible rate begins with $log_2 n = 6$, so we can conclude that the physical shape of the array is 64×64 elements.

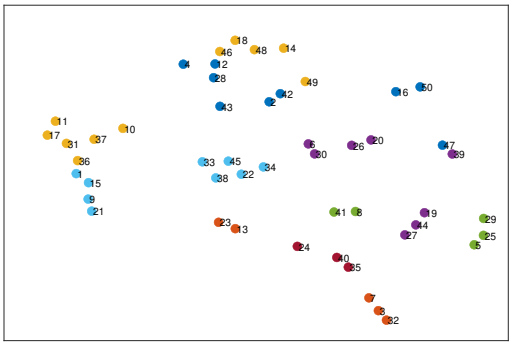


Fig. 10. Spatial arrangement of all users using the t-SNE. Different colors represent different user clusters.