To Supervise or Not To Supervise: How to Effectively Learn Wireless Interference Management Models?

Bingqing Song, Haoran Sun, Wenqiang Pu, Sijia Liu, and Mingyi Hong

***Abstract*—Machine learning has become successful in solving wireless interference management problems. Different kinds of deep neural networks (DNNs) have been trained to accomplish key tasks such as power control, beamforming and admission control. There are two state-of-the-art approaches to train such DNNs based interference management models: supervised learning (i.e., fits labels generated by an optimization algorithm) and unsupervised learning (i.e., directly optimizes some system performance measure). However, it is by no means clear which approach is more effective in practice. In this paper, we conduct some theory- and experiment- guided study about these two training approaches. First, we show a somewhat surprising result, that for some special power control problem, the unsupervised learning can perform much worse than its counterpart, because it is more likely to stuck at some low-quality local solutions. We then provide a series of theoretical results to further understand the properties of the two approaches. To our knowledge, these are the first set of theoretical results trying to understand different training approaches in learning-based wireless communication system design.**

***Index Terms*—Deep learning, wireless communication, semi-supervised learning, power control**

I. Introduction

Motivation. Recently, machine learning techniques have become very successful in solving wireless interference management problems. Different kinds of deep neural network (DNN), such as fully connected network (FCN) [2], recurrent neural network (RNN) [3], graph neural network (GNN) [4] have been designed to accomplish key tasks such as power control [5], beamforming [2], , MIMO detection [6], among others. These DNN based models are capable of achieving competitive and sometimes even superior performance compared to the state-of-the-art optimization based algorithms [5].

However, despite its success, there is still a fundamental lack of understanding about *why* DNN based approaches work so well for this class of wireless communication problems – after all, the majority of interference management problems (e.g., beamforming) are arguably more complex than a typical machine learning problem such as image classification. It is widely believed that, exploiting task-specific properties in designing network architectures, as well as training objectives can help reduce the network complexity and input feature dimension [5], boost the training efficiency [5], and improve the expressiveness [2].

The overarching goal of this research is to understand how problem-specific properties can be effectively utilized in the DNN design. More concretely, we attempt to provide an in-depth understanding about how to utilize problem structures in designing efficient training procedures. Throughout the paper, we will utilize the classical weighted sum rate (WSR) maximization problem in single-input single output (SISO) interference channel as a working example, but we believe that our approaches and the phenomenon we observed can be extended to many other related problems.

Problem Statement and Contributions. Consider training DNNs for power control, or more generally for beamforming. There are two state-of-the-art approaches for training:
1) *supervised learning (SL)*, in which "labels" of optimal power allocations are generated by an optimization algorithm, then the training step minimizes the mean square error (MSE) between the DNN outputs and the labels [2];
2) *unsupervised learning (UL)*, which optimizes some system performance measure such as WSR [5].

It is clear that the above unsupervised approach is unique to the interference management problem, because the specific task of WSR maximization offers a natural training objective to work with. Further, it does not require any existing algorithms to help generate high-quality labels (which could be fairly expensive). On the other hand, such an objective is difficult to optimize since the WSR is a highly non-linear function with respect to (w.r.t.) the transmit power, which is again a highly non-linear function of the DNN parameters.

Which training method shall we use in practice? Can we rigorously characterize the behavior of these methods? Is it possible to properly integrate these two approaches to yield a more efficient training procedure? Towards addressing these questions, this work makes the following key contributions:
❶ We focus on the SISO power control problem in interference channel (IC), and identify a simple 2-user setting, in which UL approach has *non-zero probability* of getting stuck at low-quality solutions (i.e., the local minima), while the SL approach always finds the global optimal solution;
❷ We provide rigorously analysis to understand properties of UL and SL for DNN-based SISO-IC problem. Roughly speaking, we show that when high-quality labels are provided, SL should outperform UL in terms of solution quality. Further, the SL approach converges faster when the labels have better solution quality;

B. Song, H. Sun and M. Hong are with Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455, USA. W. Pu is with the Shenzhen Research Institute of Big Data, Shenzhen, China. S. Liu is with the CSE Department, Michigan State University, East Lancing, MI.
A short version of this paper [1] has been submitted to SPAWC 2021.

❸ In an effort to leverage the advantage of both approaches, we develop a *semi-supervised* training objective, which regularizes the unsupervised objective by using a few labeled data points. Surprisingly, by only using a small fraction ($\approx 1\%$) of samples of the supervised approach, the proposed method is able to avoid bad local solutions and attain similar performance as supervised learning.

To the best of our knowledge, this work provides the first in-depth understanding about the two popular approaches for training DNNs for wireless communication.

II. Preliminaries

Consider a wireless network consisting of K pairs of transmitters and receivers. Suppose each pair equips with a single antenna, denote $h_{kj} \in \mathbb{C}$ as the channel between the kth transmitter and the jth receiver, p_k as the power allocated to the kth transmitter, $P_{\max}$ as the budget of transmitted power, and σ^2 as the variance of zero-mean Gaussian noise in the background. Further, we use w_k to represent the prior importance of the kth receiver, then the classical WSR maximization problem can be formulated as

$$\begin{aligned} \max_{p_1,\ldots,p_K} \; & \sum_{k=1}^{K} w_k \log\left(1+\frac{|h_{kk}|^2 p_k}{\sum_{j\neq k}|h_{kj}|^2 p_j+\sigma_k^2}\right) := R(\mathbf{p};|\mathbf{h}|) \\ \text{s.t.} \; & 0 \le p_k \le P_{\max}, \forall k = 1,2,\ldots,K \end{aligned} \tag{1}$$

where $\mathbf{h} := \{h_{kj}\}$ collects all the channels; $|\cdot|$ is the componentwise absolute value operation; and $\mathbf{p} := (p_1, p_2, \ldots, p_K)$ denotes the transmitted power of K transmitters. The above problem is well-known in wireless communication, and it is known to be NP-hard [7] in general. For problem (1) and its generalizations such as the beamforming problems in MIMO channels, many iterative optimization based algorithms have been proposed, such as waterfilling algorithm [8], interference pricing [9] , WMMSE [10], and SCALE [11].

Recently, there has been a surge of works that apply DNN based approach to identify good solutions for problem (1) and its extensions [2], [5]. Although these works differ from their problem settings and/or DNN architectures, they all use either the SL, UL, or some combination of the two to train the respective networks. Below let us take problem (1) as an example and briefly compare the SL and UL approaches.

• **Data Samples:** Both approaches require a collection of the channel information over N different snapshots, denoted as $\mathbf{h}^{(n)}$, $n = 1, 2, \ldots, N$. SL requires an additional N labels $\bar{\mathbf{p}} := \{\bar{\mathbf{p}}^{(n)}\}_{n\in[N]}$ (where $[N] := \{1, \cdots, N\}$), which are usually obtained by solving N independent problems (1) using some optimization algorithm, such as the WMMSE [10]. Notice that the quality of such labels may depend on the accuracy of the optimization algorithm being selected.

• **DNN Structure:** We will assume that the power allocation $\mathbf{p}$ is parameterized by some DNN. More precisely, the inputs of the DNN are absolute values of channel samples $\mathbf{h}^{(n)}$, and let $\mathbf{\Theta}$ be the parameters of the DNN (of appropriate size), then the output of DNN can be expressed as $\mathbf{p}(\mathbf{\Theta}; |\mathbf{h}^{(n)}|) \in \mathbb{R}^K$.

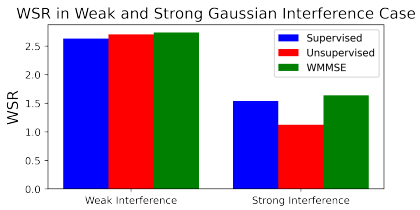


Figure 1. Comparison between SL, UL and WMMSE in testing time, when SL, UL are trained using data where the interference channel power is equal to direct channel power (weak interference), or 10 times of the direct channel power (strong interference) when there are 10 users. In strong interference case, SL can achieve 92% of the WMMSE sum-rate, while UL achieves relatively lower sum-rate.

To simplify notation, we write the output of the DNN and its kth component as:

$$\mathbf{p}^{(n)} = \mathbf{p}\left(\mathbf{\Theta}; \left|\mathbf{h}^{(n)}\right|\right), \quad p_k^{(n)} := p_k\left(\mathbf{\Theta}; \left|\mathbf{h}^{(n)}\right|\right). \tag{2}$$

Unless otherwise noted, we will assume that different training approaches will use the same DNN architecture, so we can better focus on the training approaches itself.

For the SL approach, it is common to minimize the MSE loss, and the resulting training problem is given by:

$$\begin{aligned} \min_{\mathbf{\Theta}} \; & \sum_{n=1}^{N} \|\mathbf{p}(\mathbf{\Theta}; |\mathbf{h}^{(n)}|) - \bar{\mathbf{p}}^{(n)}\|^2 := f_{\sup}(\mathbf{\Theta}) \\ \text{s.t.} \; & \mathbf{0} \le \mathbf{p}(\mathbf{\Theta}; |\mathbf{h}^{(n)}|) \le \mathbf{P}_{\max}, \forall\, n. \end{aligned} \tag{3}$$

On the other hand, UL does not need the labels $\bar{\mathbf{p}}^{(n)}$, and it directly optimizes the sum of the samples' WSR as follows:

$$\begin{aligned} \min_{\mathbf{\Theta}} \; & \sum_{n=1}^{N} -R\left(\mathbf{p}(\mathbf{\Theta}; |\mathbf{h}^{(n)}|), |\mathbf{h}^{(n)}|\right) := f_{\text{unsup}}(\mathbf{\Theta}) \\ \text{s.t.} \; & \mathbf{0} \le \mathbf{p}(\mathbf{\Theta}; |\mathbf{h}^{(n)}|) \le \mathbf{P}_{\max}, \forall\, n. \end{aligned} \tag{4}$$

Remark 1. *Problem* (4) *provides a reasonable formulation as it directly stems from the WSR maximization* (1)*. However, this problem can be much harder to optimize compared with* (1) *because of the following: i) Each* $R\left(\mathbf{p}(\mathbf{\Theta}; |\mathbf{h}^{(n)}|), |\mathbf{h}^{(n)}|\right)$ *is a composition of two non-trivial nonlinear functions,* $R(\cdot; |\mathbf{h}|)$ *and* $\mathbf{p}(\cdot; |\mathbf{h}|)$*; ii) It finds a single parameter* $\mathbf{\Theta}$ *that maximizes the sum of the WSR across all snapshots, so it couples* N *difficult problems.* ■

III. A Study of SL and UL approaches

Are there any fundamental differences between these two popular training approaches? This section provides a number of different ways to address this question. Please note that due to space limitation, all proofs in this section will be relegated to the online version [12].

Comparing SL and UL Approaches. Before we start, we use a simple example to illustrate the potential performance difference of the two training approaches. Specifically, Fig. 1 shows that for a 2-user network with different interference situation, the DNN generated by SL and UL can have significantly different test-time performance.

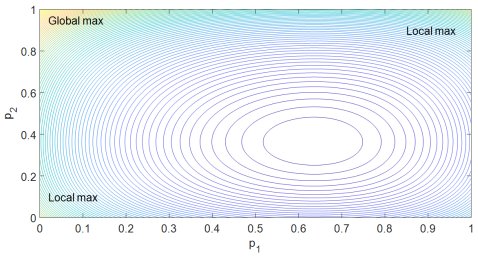


Figure 2. For two-user IC with 2 snapshots, the true labels $\bar{\mathbf{p}}^{(1)} = (0,1)$, $\bar{\mathbf{p}}^{(2)} = (1,0)$. For both users, keep the sum of label of each snapshot to be 1 (since we know that the global optimal solution has this structure), that is $\mathbf{p}^{(1)} = (p_1, 1-p_1)$, $\mathbf{p}^{(2)} = (p_2, 1-p_2)$. We plot the sum-rate of the two snapshots. The upper right and lower left corners are local maxima while the upper left is the global maximum.

To understand such a phenomenon, let us examine the two optimization problems (3) and (4). From Remark 1, we know that problem (4) can be challenging because the complicated relationship between R and $\mathbf{\Theta}$, and because there are multiple components in the objective. For now, let us focus on cases where one factor is dominating. Suppose $K = 2$ (two user), $w_k = 1, \forall\ k$ (equal weights), and use a linear network to parameterize $\mathbf{p}$: $\mathbf{p} = \mathbf{\Theta}|\mathbf{h}|$, where $\mathbf{\Theta} \in \mathbb{R}^{K \times K^2}$, and $\mathbf{\Theta} := [\mathbf{\Theta}_1; \cdots; \mathbf{\Theta}_K]$, with $\mathbf{\Theta}_k := \{\mathbf{\Theta}_{k,(uv)}\}_{(uv)\in W} \in \mathbb{R}^{1 \times K^2}$, where $W := \{(i,j) : i,j \in \{1, \cdots, K\}$ is a set of index tuples. In this case, from the classical results for 2-user IC [13], [14], we know that for each sample n, the sum rate maximization problem (3) is easy to solve, and the solution will be binary. Further, the linear network significantly simplifies the relation between $\mathbf{p}$ and $\mathbf{\Theta}$. Under this setting, we have the following observation.

Claim 1. *Consider the simple SISO-IC case with two users and two samples (i.e., $K = 2$, $N = 2$); let $P_{\max} = 1$, $\sigma = 1$, and suppose a linear network is used: $\mathbf{p}(\mathbf{\Theta}; |\mathbf{h}|) = \mathbf{\Theta}|\mathbf{h}|$. If we use the UL loss* (4)*, then there exist some channel realizations $\mathbf{h}^{(1)} \in \mathcal{C}^{2\times 2}$ and $\mathbf{h}^{(2)} \in \mathcal{C}^{2\times 2}$ whose true labels are $\bar{\mathbf{p}}^{(1)} = (0,1)$, $\bar{\mathbf{p}}^{(2)} = (1,0)$, for which problem* (4) *has at least two stationary solutions $\mathbf{\Theta}_{\text{global}}$ and $\mathbf{\Theta}_{\text{local}}$. However, these two solutions generate different predictions:*

$$\mathbf{p}(\mathbf{\Theta}_{\text{global}}, |\mathbf{h}^{(1)}|) = (0,1),\ \mathbf{p}(\mathbf{\Theta}_{\text{global}}, |\mathbf{h}^{(2)}|) = (1,0), \quad (5)$$

$$\mathbf{p}(\mathbf{\Theta}_{\text{local}}, |\mathbf{h}^{(1)}|) = \mathbf{p}(\mathbf{\Theta}_{\text{local}}, |\mathbf{h}^{(2)}|) = (1,0). \quad (6)$$

On the other hand, if the SL loss (3) *is used, then $f_{\sup}(\mathbf{\Theta})$ is a convex function w.r.t. $\mathbf{\Theta}$, and the problem has an optimal solution satisfying* (5)*.*

This result illustrates that when multiple channel realizations are directly and jointly optimized using UL, it is more likely to possess bad local minima; see Fig 2.

Next, we analyze more general cases. Towards this end, we first investigate the relationship between stationary solutions of the SL problem (3) and the UL problem (4).

Claim 2. *Consider an SISO-IC training problem with K users and N training samples. Suppose the following hold:*
i). For each data sample $n \in \{1, \cdots, N\}$, we can generate a stationary solution $\bar{\mathbf{p}}^{(n)}$ of (1) *as the training label.*
ii). Let $\mathbf{\Theta}^*(\bar{\mathbf{p}})$ denote the optimal solution for the SL problem (3) *with label $\bar{\mathbf{p}}$, and it achieves zero loss: $f_{\sup}(\mathbf{\Theta}^*(\bar{\mathbf{p}})) = 0$.*
iii) The solution $\mathbf{\Theta}^*(\bar{\mathbf{p}})$ can be computed for all $\bar{\mathbf{p}}$.
Let $\mathcal{B}$ denote the set of stationary points of (4)*. Then the following holds:*

$$\{\mathbf{\Theta}^*(\bar{\mathbf{p}}) \mid \bar{\mathbf{p}}^{(n)} \text{ is a stationary solution of (1)}, \forall\ n\} \subseteq \mathcal{B}. \quad (7)$$

Intuitively, this result shows that if we impose some additional assumptions to the SL approach (i.e., good labels, zero training loss, and good training algorithm), then it is less likely for SL to be trapped by local minima. Additionally, if each label $\bar{\mathbf{p}}^{(n)}$ exactly maximizes (1), then SL can find a neural network that simultaneously optimizes all training instances. On the other hand, it is difficult to impose favorable assumptions for the UL approach to induce better solution quality. This result is a generalization of Claim 1.

It certainly appears that assumptions *ii)* and *iii)* are stringent. However, recent advances in deep learning suggest that they can be both achieved for certain special neural networks. In particular, the assumption that $f_{\sup}(\mathbf{\Theta}^*) = 0$ has been verified when the neural network is "overparameterized"; see. e.g., [15]. Further, it has been shown in [16], [17] that, gradient descent (GD) can indeed find such a global optimal solution. However, these works cannot be applied to analyze our training problem because they require that the inputs are normalized, and that the outputs are scalars instead of vectors.

In the following, we show that it is possible to construct a special neural network and a training algorithm, such that condition *ii)* and *iii)* in Claim 2 can be satisfied, so that (29) holds true. Our result extends the recent work [18].

To proceed, consider an L-layer fully connected network with activation function denoted by $f : \mathbb{R} \to \mathbb{R}$. The weights of each layer are $(W_l)_{l=1}^L$. Let $\|\cdot\|_F$ denote the Frobenius norm and $\|\cdot\|_2$ denote the L_2 norm. The input and output of the network (across all samples) are $\mathbf{h} \in \mathbb{R}^{N \times K^2}$ and $\mathbf{p} \in \mathbb{R}^{N \times K}$, respectively. Let $\otimes$ denote the Kronecker product. Let the output of the l-th layer (across all samples) be $F_l \in \mathbb{R}^{N \times n_l}$, which can be expressed as:

$$F_l = \begin{cases} \mathbf{h} & l = 0 \\ \sigma\left(F_{l-1} W_l\right) & l \in [1 : L-1] \\ F_{L-1} W_L & l = L \end{cases} \quad (8)$$

where σ is some activation function. In our problem setting, the output of the neural network is the power allocation vector, therefore $n_L = K$. Let us vectorize the output of each layer by concatenating each of its column, and denote it as $f_l = \text{vec}\left(F_l\right) \in \mathbb{R}^{N n_l}$. Similarly, denote the vectorized label as $y = \text{vec}(\mathbf{p}) \in \mathbb{R}^{NK}$. At m-th iteration of training, we use $\mathbf{\Theta}^m = (W_l^m)_{l=1}^L$ to denote all the parameters. Also, denote $\Sigma_l = \text{diag}\left[\text{vec}\left(\sigma'\left(F_{l-1} W_l\right)\right)\right] \in \mathbb{R}^{N n_l \times N n_l}$ as the derivative of activation function at each layer.

Let us define the following quantities, which are related to the singular values of weight matrices at initialization:

$$\bar{\lambda}_l = \begin{cases} \frac{2}{3}\left(1 + \left\|W_l^0\right\|_2\right), & \text{for } l \in \{1,2\}, \\ \left\|W_l^0\right\|_2, & \text{for } l \in \{3, \ldots, L\}, \end{cases} \quad (9)$$

and $\lambda_l = \sigma_{\min}\left(W_l^0\right), \lambda_{i\to j} = \prod_{l=i}^j \lambda_l, \bar{\lambda}_{i\to j} = \prod_{l=i}^j \bar{\lambda}_l$ and $\lambda_F = \sigma_{\min}\left(\sigma\left(X W_1^0\right)\right)$, where $\sigma_{\min}(A)$ and $\|A\|_2$ are the smallest and largest singular value of matrix A.

Let us make the following assumptions about the neural network structure as well as the activation function.

Assumption 1. *(Pyramidal Network Structure) Let* $n_1 \geq N$ *and* $n_2 \geq n_3 \geq \ldots \geq n_L$.

Assumption 2. *There exist constants* $\gamma \in (0,1)$ *and* $\beta > 0$*, such that the activation function* $\sigma(\cdot)$ *satisfies:*

$$\sigma'(x) \in [\gamma, 1], \ \ |\sigma(x)| \leq |x|, \ \forall \ x \in \mathbb{R}, \ \ \sigma' \ \text{ is } \beta\text{-Lipschitz.}$$

The first assumption defines the so-called Pyramidal Network structure [18], which consists of at least one wide layer (i.e., the number of neurons is at least the sample size). The second assumption is shown to hold true for certain activation functions [18] .

Next we discuss how to train such a network using the SL and UL approaches. Towards this end, we need to fix a training algorithm. Different than the conventional neural network training, problems (3) – (4) have n constraints (one for each sample), and it is difficult for conventional gradient-based algorithms to enforce them. To overcome such a difficulty, we adopt the following approaches.

For the SL training, we will directly consider the unconstrained version of (3) (by removing all power constraints). This is acceptable because, *if* zero training loss can be achieved, and if all the labels are feasible, then the output for each sample will also be feasible. However, for the UL training, we cannot simply drop the constraints, so a sigmoid function should be added to the last layer of the output to enforce feasibility. Specifically, the modified network has the following output:

$$F_L = \text{sig}(F_{L-1}\mathbf{\Theta}_L) = \frac{\mathbf{1} \times P_{\max}}{1 + e^{-F_{L-1}\mathbf{\Theta}_L}}. \tag{10}$$

And the output F_L is the allocated power $\mathbf{p}$. So the UL loss (4) can be converted into unconstrained version:

$$\min_{\mathbf{\Theta}} \quad \sum_{n=1}^{N} -R\left(\mathbf{p}(\mathbf{\Theta}; |\mathbf{h}^{(n)}|), |\mathbf{h}^{(n)}|\right) := f_{\text{UL}}(\mathbf{\Theta})$$

Note that in the above expression there is some abuse of notation, since we still use $p(\mathbf{\Theta}; |\mathbf{h}^{(n)}|)$ to denote the output of the neural network, despite the fact that the neural network structure is slightly different than before.

Now that both training problems become unconstrained, we can use the conventional gradient-based algorithms. We have the following convergence results.

Claim 3. *Consider an SISO-IC training problem with* K *users and* N *training samples. Let* $P_{max} = 1$*. Construct a fully connected neural network satisfying Assumption 1 - 2. Initialize* $\mathbf{\Theta}^0$ *so that it satisfies [18, Assumption 3.1]. Then the following holds:*
(a) When the initialization condition satisfies Assumption 3, consider optimizing the unconstrained version of (3) *using the gradient descent algorithm*

$$\mathbf{\Theta}^{m+1} = \mathbf{\Theta}^m - \eta \nabla f_{\text{sup}}\left(\mathbf{\Theta}^m\right).$$

There exists constant stepsize η *such that the training loss converges to zero at a geometric rate, that is:*

$$f_{\text{sup}}\left(\mathbf{\Theta}^m\right) \leq \left(1 - \eta\alpha_0\right)^m f_{\text{sup}}\left(\mathbf{\Theta}^0\right) \tag{11}$$

where α_0 *is a constant.*
(b) Consider minimizing the unconstrained version of (4) *using the last layer as* (10) *and use the gradient descent algorithm (with step size* η*). Suppose all the weights are bounded during training, then* $\mathbf{\Theta}$ *will converge to a stationary point of the training objective.*

Claim 3-(a) indicates that when the neural network satisfied Assumptions 1 – 2, and with some special initialization, then conditions (ii) – (iii) in Claim 2 can be satisfied, so the conclusion in Claim 2 holds. On the other hand, for UL problem, even under very strong condition such as bounded training weights, the best one can prove is that a stationary solution for the training problem is obtained. No global optimality can be claimed, nor any convergence rate analysis can be done. Intuitively, this result again says one can identify sufficient conditions that SL can perform well, while the UL approach is much more challenging to analyze. We note that the analysis of Claim 3-(a) follows similar approaches as [18, Theorem 3.2]. However, Claim 3-(b) is different since we need to analyze the special network with the sigmoid activation function and sum-rate objective function.

Impact of Label Quality. The above results show different objective functions can have different performance in maximizing the sum-rate. Next, we show an additional property about the SL approach – that the *quality of labels* can affect training efficiency. Intuitively, it is reasonable to believe that neural networks trained using high-quality labeled data can achieve higher sum-rate compared with those trained with with low-quality labels. To see this, we conduct two simple experiments. We generate two training sets, one with low-quality labels and the other with high-quality labels. The low-quality labels are the power allocations that achieve an average of 1.65 bits/sec (resp. 1.88 bits/sec) for 10 users (resp. for 20 users) case. The high-quality labels are the power allocations that achieve an average of 1.87 bits/sec (resp. 2.00 bits/sec) for 10 users (resp. for 20 users) case. We use different number of samples to train the network, derive the sum-rate using test samples and compare the result to the corresponding sum rate achieved by the given labels; the results are shown in Table I. We see that for a particular setting, using high-quality labels not only achieves higher absolute sum-rate, but also higher *relative* sum-rate comparing with what can be achieved by the labels.

Below, we argue the benefit of high-quality label from a slightly different perspective – the label quality can influence the convergence speed of training algorithm.

Claim 4. *Suppose* $(\mathbf{h}, \mathbf{p})$ *and* $(\mathbf{h}^{'}, \mathbf{p}^{'})$ *are two sets of data, and they are constructed below:*

- *Each dataset consists of* N *samples;*
- *The features of two data samples are identical:* $\mathbf{h}^{'} = \mathbf{h}$*;*
- *In the first dataset, for any* $n \in [N]$*, the labels* $\mathbf{p}^{(n)}$ *is the unique globally optimal power allocation for problem* (1)*, given channel realization* $\mathbf{h}^{(n)}$*; Further, two samples in* $\mathbf{h}$ *are identical, say,* $\mathbf{h}^{(1)} = \mathbf{h}^{(2)}$*, and all the other samples are linearly independent.*
- *For the second dataset, the labels are constructed as*

Quality \ # samples	30,000	40,000	50,000
Low	1.38 (83.6%)	1.38 (83.6%)	1.39 (84.2%)
High	1.72 (92.0%)	1.76 (94.1%)	1.78 (95.2%)
Quality \ # samples	**50,000**	**100,000**	**200,000**
Low	1.11 (59.0%)	1.32 (70.2%)	1.39 (73.9%)
High	1.31 (65.6%)	1.55 (77.5%)	1.74 (87.0%)

Table I

Comparison between using high-quality labels and low-quality labels in SL. The top (resp. bottom) table shows the $K = 10$ (resp. $K = 20$) case. The number in each entry shows the testing performance (in bits/sec), where the model is trained using a fixed number of training sample (shown at the first row), with either low or high quality labels. The percentages mean the relative sum-rate achieved at testing time v.s. what is achieved by the given labels.

follows:

$$\mathbf{p}^{',(2)} \neq \mathbf{p}^{(2)}, \quad \mathbf{p}^{',(n)} = \mathbf{p}^{(n)}, \forall\, n, \neq 2. \tag{12}$$

Further, since $\mathbf{h}^{(1)} = \mathbf{h}^{(2)}$ *and* $\mathbf{h} = \mathbf{h}'$*, we also have* $\mathbf{h}^{'(1)} = \mathbf{h}^{'(2)}$.

Suppose that Assumption 1 and Assumption 2 hold true, and use the same training algorithm as Claim 3-(a) to optimize the unconstrained version of (3) *using* $(\mathbf{h}, \mathbf{p})$ *and* $(\mathbf{h}^{'}, \mathbf{p}^{'})$ *respectively. Let* $\mathbf{\Theta}^m$ *and* $\mathbf{\Theta}^{'m}$ *denote the sequences of weights generated by the algorithm for the two data sets respectively. Suppose that the initial solutions of the two algorithms are the same:* $\mathbf{\Theta}^{'0} = \mathbf{\Theta}^0$*. Define*

$$\begin{aligned} A(\mathbf{\Theta}) &:= \left(\mathbb{I}_{n_2} \otimes F_1^T\right) \prod_{q=3}^{L} \Sigma_{q-1} \left(W_q \otimes \mathbb{I}_N\right), \\ A_0 &:= A(\mathbf{\Theta}^0). \end{aligned} \tag{13}$$

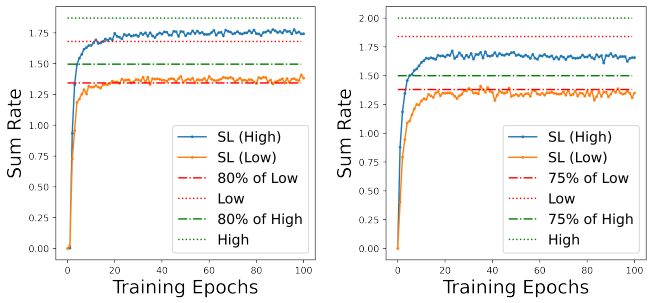
Suppose all the eigenvalues of $A_0^T A_0$ *are within the interval* $[0, 1]$ *. Then if we choose the stepsize* η *small enough, there exist* $\beta > 0$ *and* $\beta^{'} > 0$ *such that the following holds true*

$$f_{\text{sup}}\left(\mathbf{\Theta}^1\right) \leq \beta f_{\text{sup}}\left(\mathbf{\Theta}^0\right), \quad f_{\text{sup}}\left(\mathbf{\Theta}^{'1}\right) \leq \beta^{'} f_{\text{sup}}\left(\mathbf{\Theta}^{'0}\right).$$

Further, we have $\beta < \beta^{'}$*, that is, the objective function with the correct label decreases faster.*

In our analysis, we combined the pyramidal network analysis with the decomposition technique from [19]. This result uses a simple construction to reveal the importance of *consistency* of labels among "similar" samples. Intuitively, it somewhat explains why in Table I, the models trained by high-quality labels can achieve higher percentage of the rates. The reason may be that when the quality of the label is better, the training speed is also faster.

To empirically understand how the quality of labels affect convergence speed, we conduct the following experiments. Consider 10- and 20-user case under the strong interference setting as illustrated in Fig. 1. We generate two sets of labels for each case, the low-quality one directly obtained by WMMSE while the high-quality one first passes a given sample through a pretrained GNN model in [4] and then is fine-tuned by WMMSE. We use a fully connected network with 3 hidden layers, with the number of neurons being 200, 80, 80 for 10-user case and 600, 200, 200 for 20-user case. From Fig. 4, we see that SL with higher-quality labels achieves 80% of the baseline sum rate faster than with lower-quality labels for 10-user case. Similar result can be derived in matching 75% of the baseline for 20-user case.



(a) Strong interference with K=10 (b) Strong Interference with K=20

Figure 3. Comparison between SL using different labels. 'Low' and 'High' in the legend means the quality of labels are low or high. We also draw the sum rate of the generated data and labels as baseline, as well as the 80% of the sum rate in 10-user case and 75% of the sum rate in 20-user case.

IV. A SEMI-SUPERVISED LEARNING REMEDY FOR POWER ALLOCATION

From the previous section, we know that under a few assumptions, especially when high-quality labels are available, SL could perform better than the UL. However, one drawback of the SL approach is that finding high-quality labels can be costly. Is there a way to design a proper learning strategy that only requires a few labels, while still achieving the state-of-the-art training and testing performance? In this section, we address this by proposing a *semi-supervised* learning (SSL) strategy which combines both the SL and UL approaches in (3) – (4).

As indicated by Claim 1, UL may get stuck at some local solutions once parameters enter some "bad" regions. To alleviate such an issue, we propose to add some (label-dependent) regularization in the training objective to change the landscape of loss function. Specifically, suppose we collect the unlabeled samples $\{|\mathbf{h}^{(n)}|\}$ in a set $\mathcal{N}$, and the labelled samples $\{|\mathbf{h}^{(m)}|, \bar{\mathbf{p}}^{(m)}\}$ in a set $\mathcal{M}$. Then, we propose to combine the formulations (3) – (4) and construct the following training objective:

$$\begin{aligned} \max_{\mathbf{\Theta}} \sum_{n \in \mathcal{N}} R\left(\mathbf{p}(\mathbf{\Theta}; |\mathbf{h}^{(n)}|), |\mathbf{h}^{(n)}|\right) - \\ \lambda \sum_{m \in \mathcal{M}} \left\| \mathbf{p}(\mathbf{\Theta}; |\mathbf{h}^{(m)}|) - \bar{\mathbf{p}}^{(m)} \right\|^2, \end{aligned} \tag{14}$$

where $\lambda > 0$ is a constant which controls the trade-off between two different loss functions. Intuitively, the regularizer enforces the classical *cluster assumption* [20], which says samples with the same label should belong to the same class. In the numerical results (to be shown shortly), we will observe that the above SSL approach can outperform the UL approach by only using a few samples.

V. Simulation Results

Data Generation. The Rayleigh fading channel model [21] is considered in the simulation and the number of users is 5, 10 or 20. Direct channels h_{kk} and interfering channels $h_{kj}, k \neq j$ are generated from zero-mean complex Gaussian distribution $\mathcal{CN}(0, \sigma^2)$, where σ denotes the standard deviation. To evaluate the stability of different learning approaches, two representative cases are considered. In the first case (referred as "weak interference"), both direct and interfering channels are generated from the same complex Gaussian distribution with $\sigma = 1$. For the second case (referred as "strong interference"), direct channels are generated using the same method as in the first case, while the interfering channel has larger standard deviation (i.e., $\sigma = 10$).

Neural Network Structure. A fully connected neural network with 3 hidden layers is used. The number of neurons in each hidden layers are $200, 80, 80$ for 5- and 10-user case and $600, 200, 200$ for 20-user case, respectively. The activation function of the hidden layers is *ReLu* function, and the *Sigmoid* function is used at output layer. To stabilize the training process, the *Batch Normalization* [22] is used after each hidden layer.

Benchmarks and Label Generation. In our results, we compare the following algorithms: *1)* The UL approach (4); *2)* A standard SSL approach, where problem (4) is trained based on an initialization generated by the training over labeled data (subsequently referred to as *pre-trained SSL*). *4)* The proposed approached based on optimizing (14), with $\lambda = 1$ (subsequently referred to as *regularized SSL*) *4)* The WMMSE [10] algorithm.

We adopt the following approach to generate high-quality labels. Instead of directly using WMMSE, we first pass a given sample $\mathbf{h}^{(n)}$ through a pretrained GNN model (trained using the method proposed in [4]), and then fine tune the result using WMMSE. To generate the low-quality labels, we simply perform WMMSE to obtain the labels.

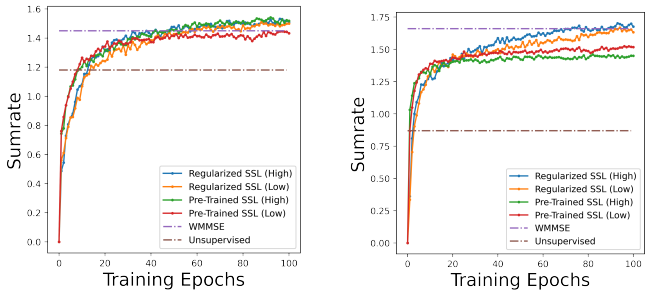
Training Procedure. In the strong interference case, the total number of unlabeled and labeled samples are $50,000$ and 400 for the 10-user case, while $10,000$ and 100 for the 5-user case, respectively. In the weak interference case, the total number of unlabeled and labeled training samples are $20,000$ and 100, respectively. The number of data used here is smaller than the strong interference case because in this setting, the UL approach can already work well with fewer samples. In both cases, the RMSprop [23] algorithm is used as the optimizer, where each mini-batch consists of 200 (randomly sampled) unlabeled data, and all the available labeled data.

To evaluate the performance, $1,000$ additional unlabeled samples are generated and their averaged sum rate is used as the performance metric.

Results and Analysis. The performance of the UL and the two SSL approaches in the strong interference case are shown in Fig. 4. Compared with the UL, the proposed regularized SSL significantly improves the sum rate in the 10-user case. However, the pre-trained SSL does not bring significant improvement. One possible reason is that only a few labeled samples are not enough to pre-train a *good* initialization.

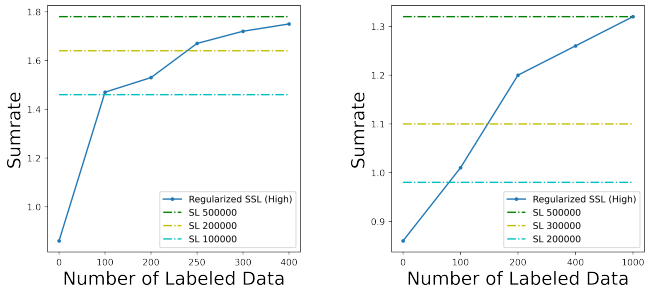
Using our proposed regularized SSL approach, we gradually increase the number of labeled data to train the network, and the result is shown in Fig. 5. From the result, the increasing labeled data can improve the performance of our proposed regularized SSL in high interference scenario. Furthermore, higher-quality labels can produce better performance than lower-quality labels.

In the weak interference scenario, we also perform our proposed regularized SSL approach in 5- and 10-user case. The result is shown in Table II, which indiates that in this setting the performance of UL and our proposed regularized SSL are similar. So in this case, regularization seems not required. A future work is to address that, in the scenario where UL can already work, whether and how the labeled data can still improve the performance?



(a) Strong interference with K=5 (b) Strong Interference with K=10

Figure 4. Comparison between proposed semi-supervised learning, pre-training, unsupervised learning and WMMSE uner strong interference case in sum-rate maximization. 'Low' and 'High' in the legend means the quality of labels are low or high.



(a) Strong interference case $K = 10$. (b) Strong interference case $K = 20$.

Figure 5. Comparison between using different number of (high-quality) labeled data in proposed semi-supervised learning. Pre-Trained SSL represents is based on an initialization generated by the training over labeled data and Regularized SSL means our proposed approach.

Method \ User Number	K=5	K=10
Semi-supervised	2.09 (bits/sec)	2.60 (bits/sec)
Unsupervised	2.09 (bits/sec)	2.64 (bits/sec)
WMMSE	2.06 (bits/sec)	2.74 (bits/sec)

Table II

For weak interference scenario, compare the performance of unsupervised learning and proposed semi-supervised learning both using $20,000$ samples, and semi-supervised learning with 100 additional labeled data.

VI. Conclusion

This work analyzes the SL and UL approaches for learning communication systems. It is shown that under certain conditions (such as having access to high-quality labels), SL can exhibit better convergence properties than UL. To our knowledge, this is the first work that rigorously analyzes the relation between these two approaches. Of course, finding high-quality labels is challenging. Is there a way to design a proper learning strategy that only requires a few high-quality labels, while still achieving the state-of-the-art performance? In our full paper [12], we developed some semi-supervised learning approach to address this question. Due to space limitation, we do not include them here.

References

[1] B. Song, H. Sun, W. Pu, S. Liu, and M. Hong. To supervise or not to supervise: How to effectively learn wireless interference management models?

[2] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos, "Learning to optimize: Training deep neural networks for interference management," *IEEE Transactions on Signal Processing*, vol. 66, no. 20, pp. 5438–5453, 2018.

[3] E. Nachmani, E. Marciano, L. Lugosch, W. J. Gross, D. Burshtein, and Y. Be'ery, "Deep learning methods for improved decoding of linear codes," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 119–131, 2018.

[4] Y. Shen, Y. Shi, J. Zhang, and K. B. Letaief, "A graph neural network approach for scalable wireless power control," in *proceedings of the 2019 IEEE Globecom Workshops (GC Wkshps)*.

[5] F. Liang, C. Shen, W. Yu, and F. Wu, "Towards optimal power control via ensembling deep neural networks," *IEEE Transactions on Communications*, vol. 68, no. 3, pp. 1760–1776, 2019.

[6] N. Samuel, T. Diskin, and A. Wiesel, "Deep MIMO detection," in *2017 IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2017, pp. 1–5.

[7] Z.-Q. Luo and S. Zhang, "Dynamic spectrum management: Complexity and duality," *IEEE Journal of Selected Topics in Signal Processing*, vol. 2, no. 1, pp. 57–73, 2008.

[8] G. Scutari, D. P. Palomar, and S. Barbarossa, "The MIMO iterative waterfilling algorithm," *IEEE Transactions on Signal Processing*, vol. 57, no. 5, pp. 1917–1935, 2009.

[9] D. A. Schmidt, C. Shi, R. A. Berry, M. L. Honig, and W. Utschick, "Distributed resource allocation schemes," *IEEE Signal Processing Magazine*, vol. 26, no. 5, pp. 53–63, 2009.

[10] Q. Shi, M. Razaviyayn, Z.-Q. Luo, and C. He, "An iteratively weighted mmse approach to distributed sum-utility maximization for a mimo interfering broadcast channel," *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4331–4340, 2011.

[11] J. Papandriopoulos and J. S. Evans, "Scale: A low-complexity distributed protocol for spectrum balancing in multiuser dsl networks," *IEEE Transactions on Information Theory*, vol. 55, no. 8, pp. 3711–3724, 2009.

[12] B. Song, H. Sun, W. Pu, S. Liu, and M. Hong. To supervise or not to supervise: How to effectively learn wireless interference management models? [Online]. Available: http://people.ece.umn.edu/ mhong/mingyi.html

[13] G. O. A. Gjendemsjo, D. Gesbert and S. Kiani, "Optimal power allocation and scheduling for two-cell capacity maximization," in *IEEE WiOpt*, 2006, pp. 1–5.

[14] M. Charafeddine and A. Paulraj, "Maximum sum rates via analysis of 2-user interference channel achievable rates region," in *43rd Annual Conference on Information Sciences and Systems*, march 2009, pp. 170 –174.

[15] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires rethinking generalization," *arXiv preprint arXiv:1611.03530*, 2016.

[16] Z. Allen-Zhu, Y. Li, and Z. Song, "A convergence theory for deep learning via over-parameterization," in *International Conference on Machine Learning*, 2019, pp. 242–252.

[17] S. Du, J. Lee, H. Li, L. Wang, and X. Zhai, "Gradient descent finds global minima of deep neural networks," in *International Conference on Machine Learning*, 2019, pp. 1675–1685.

[18] Q. Nguyen and M. Mondelli, "Global convergence of deep networks with one wide layer followed by pyramidal topology," *arXiv preprint arXiv:2002.07867*, 2020.

[19] S. Arora, S. Du, W. Hu, Z. Li, and R. Wang, "Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks," in *International Conference on Machine Learning*, 2019, pp. 322–332.

[20] O. Chapelle, B. Scholkopf, and A. Zien, "Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]," *IEEE Transactions on Neural Networks*, vol. 20, no. 3, pp. 542–542, 2009.

[21] B. Sklar, "Rayleigh fading channels in mobile digital communication systems. i. characterization," *IEEE Communications magazine*, vol. 35, no. 7, pp. 90–100, 1997.

[22] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[23] T. Tieleman and G. Hinton, "Rmsprop: Divide the gradient by a running average of its recent magnitude. coursera: Neural networks for machine learning," *COURSERA Neural Networks Mach. Learn*, 2012.

VII. Appendix

A. Proof for Claim 1

Claim 1. *Consider the simple SISO-IC case with two users and two samples (i.e., $K=2$, $N=2$); let $P_{\max}=1$, $\sigma=1$, and suppose a linear network is used: $\mathbf{p}(\mathbf{\Theta};|\mathbf{h}|)=\mathbf{\Theta}|\mathbf{h}|$. If we use the UL loss* (4)*, then there exist some channel realizations $\mathbf{h}^{(1)}\in\mathcal{C}^{2\times 2}$ and $\mathbf{h}^{(2)}\in\mathcal{C}^{2\times 2}$ whose true labels are $\bar{\mathbf{p}}^{(1)}=(0,1)$, $\bar{\mathbf{p}}^{(2)}=(1,0)$, for which problem* (4) *has at least two stationary solutions $\mathbf{\Theta}_{\rm global}$ and $\mathbf{\Theta}_{\rm local}$. However, these two solutions generate different predictions:*

$$\mathbf{p}(\mathbf{\Theta}_{\rm global},|\mathbf{h}^{(1)}|)=(0,1),\ \mathbf{p}(\mathbf{\Theta}_{\rm global},|\mathbf{h}^{(2)}|)=(1,0), \tag{15}$$

$$\mathbf{p}(\mathbf{\Theta}_{\rm local},|\mathbf{h}^{(1)}|)=\mathbf{p}(\mathbf{\Theta}_{\rm local},|\mathbf{h}^{(2)}|)=(1,0). \tag{16}$$

On the other hand, if the SL loss (3) *is used, then $f_{\rm sup}(\mathbf{\Theta})$ is a convex function w.r.t. $\mathbf{\Theta}$, and the problem has an optimal solution satisfying* (15)*.*

Proof. **Unsupervised learning problem** (4). Our plan is to construct the channels of two snapshots, in such a way that the true labels are $\bar{\mathbf{p}}^{(1)}=(0,1),\bar{\mathbf{p}}^{(2)}=(1,0)$. We will verify that for this problem, there is a global optimal solution $\mathbf{\Theta}_{\rm global}$ which produces the true labels as (5); we will also show that there exists a local solution $\mathbf{\Theta}_{\rm local}$ which produces $\mathbf{p}(\mathbf{\Theta}_{\rm local},|\mathbf{h}^{(1)}|)=\mathbf{p}(\mathbf{\Theta}_{\rm local},|\mathbf{h}^{(2)}|)=(1,0)$, and such a solution achieves the smallest training objective around a neighborhood. However, at this local solution, the allocated power is different from the optimal solution, and it is easy to check that the UL loss (4) is larger than the value achieved by the global optimal solution.

For notation simplicity, let us define the following short-handed notations:

$$\mathbf{p}\left(\mathbf{\Theta}_{\text{local}},|\mathbf{h}|\right):=\left[\mathbf{p}\left(\mathbf{\Theta}_{\text{local}},\left|\mathbf{h}^{(1)}\right|\right);\mathbf{p}\left(\mathbf{\Theta}_{\text{local}},\left|\mathbf{h}^{(2)}\right|\right)\right], \tag{17}$$

$$\mathbf{p}^*:=\mathbf{p}\left(\mathbf{\Theta}_{\text{local}},|\mathbf{h}|\right)=\left[p_1^{(1),*};p_2^{(1),*};p_1^{(2),*};p_2^{(2),*}\right]=[1;0;1;0], \tag{18}$$

$$\mathbf{p}:=\mathbf{p}\left(\mathbf{\Theta},|\mathbf{h}|\right)=[p_1^{(1)};p_2^{(1)};p_1^{(2)};p_2^{(2)}]. \tag{19}$$

More specifically, we will show that there exists a neighborhood $N_\delta(\mathbf{\Theta}_{\rm local}):=\{\mathbf{\Theta}:\|\mathbf{\Theta}_{\rm local}-\mathbf{\Theta}\|\le\delta\}$, such that the following holds true:

$$f_{\rm unsup}(\mathbf{\Theta})-f_{\rm unsup}(\mathbf{\Theta}_{\rm local})\ge 0,\ \text{for all } \mathbf{\Theta}\in N_\delta(\mathbf{\Theta}_{\rm local}) \text{ and } \mathbf{p}(\mathbf{\Theta},|\mathbf{h}^{(1)}|) \text{ and } \mathbf{p}(\mathbf{\Theta},|\mathbf{h}^{(2)}|) \text{ feasible.} \tag{20}$$

To show that the above holds, we will follow two steps:

Step 1. Show that there exists a region $N_\epsilon(\mathbf{p}^*):=\{\mathbf{p}:\|\mathbf{p}-\mathbf{p}^*\|\le\epsilon\}$ around $\mathbf{p}^*$ such that the following holds:

$$f_{\rm unsup}(\mathbf{p}^*)-f_{\rm unsup}(\mathbf{p})\le 0,\quad \text{for all } \mathbf{p}\in N_\epsilon(\mathbf{p}^*) \text{ and } \mathbf{p} \text{ feasible.} \tag{21}$$

Step 2. Show that for every $\widetilde{\mathbf{\Theta}}$ such that $\mathbf{p}\left(\widetilde{\mathbf{\Theta}},|\mathbf{h}|\right)=\mathbf{p}^*$, by letting $\mathbf{\Theta}_{\rm local}=\widetilde{\mathbf{\Theta}}$, there exists a region $N_\delta(\mathbf{\Theta}_{\rm local})$ such that (20) holds true.

To begin our proof, let us construct $|\mathbf{h}^{(1)}|$ and $|\mathbf{h}^{(2)}|$ in such a way that the following holds:

$$|h_{12}^{(1)}|=|h_{21}^{(1)}|\gg|h_{22}^{(1)}|>|h_{11}^{(1)}|,\ |h_{12}^{(2)}|=|h_{21}^{(2)}|\gg|h_{11}^{(2)}|>|h_{22}^{(2)}|. \tag{22}$$

It is easy to show that the true label for snapshot $\mathbf{h}^{(1)}$ is $\bar{\mathbf{p}}^{(1)}=(0,1)$ and $\mathbf{h}^{(2)}$ is $\bar{\mathbf{p}}^{(2)}=(1,0)$. Further, we assume that the cross channels are strong enough such that the following inequality holds

$$\frac{2(2+h_{11}^{(n)})|h_{22}^{(n)}|^2}{|h_{11}^{(n)}|^2|h_{12}^{(n)}|^2}<1. \tag{23}$$

Proof of Step 1. Let us first show that (21) holds true by using contradiction. Suppose $\mathbf{p}^*=[p_1^{(1),*};p_2^{(1),*};p_1^{(2),*};p_2^{(2),*}]=[1;0;1;0]$ is not a local minimum, then for all neighborhood around $\mathbf{p}^*$, there exists a different feasible $\mathbf{p}$ which satisfies

$$f_{\rm unsup}(\mathbf{p})<f_{\rm unsup}(\mathbf{p}^*).$$

Then by the Mean Value Theorem, there exists a feasible $\hat{\mathbf{p}}$ between $\mathbf{p}$ and $\mathbf{p}^*$ which satisfies

$$f_{\text{unsup}}\left(\mathbf{p}^*\right)-f_{\text{unsup}}\left(\mathbf{p}\right)=\langle\nabla_{\mathbf{p}}f_{\text{unsup}}\left(\hat{\mathbf{p}}\right),\mathbf{p}^*-\mathbf{p}\rangle>0. \tag{24}$$

Since we have assumed $P_{\max}=1$, then $p_1^{(1),*}$ and $p_1^{(2),*}$ both reach the maximal power. It follows that any feasible $p_1^{(1)},p_1^{(2)}$ must satisfy $p_1^{(1)}<p_1^{(1),*},p_1^{(2)}<p_1^{(2),*}$. Similarly, $p_2^{(1),*}$ and $p_2^{(2),*}$ reach the minimal power, so any feasible $p_2^{(1),*}$ and $p_2^{(2),*}$ must satisfy $p_2^{(1)}>p_2^{(1),*},p_2^{(2)}>p_2^{(2),*}$. Further, the gradient of the unsupervised objective w.r.t. $\mathbf{p}$ is given by:

$$\nabla_{\mathbf{p}}f_{\text{unsup}}\left(\mathbf{p}\right)=\left(\frac{\partial f_{\text{unsup}}}{\partial p_1^{(1)}},\frac{\partial f_{\text{unsup}}}{\partial p_2^{(1)}};\frac{\partial f_{\text{unsup}}}{\partial p_1^{(2)}},\frac{\partial f_{\text{unsup}}}{\partial p_2^{(2)}}\right).$$

If we can show that there exists a neighborhood around $\mathbf{p}^*$ such that $\frac{\partial f_{\text{unsup}}}{\partial p_1^{(n)}} < 0$ and $\frac{\partial f_{\text{unsup}}}{\partial p_2^{(n)}} > 0$ for $n = 1, 2$, then within this region, there is always $\langle \nabla_{\mathbf{p}} f_{\text{unsup}}(\hat{\mathbf{p}}), \mathbf{p}^* - \mathbf{p} \rangle < 0$, which contradicts to (24). Next, we show the existence of such a region. Given our channel construction, the corresponding objective function (4) becomes:

$$\begin{aligned} f_{\text{unsup}}(\mathbf{p}) = &- \log\left(1 + \frac{|h_{11}^{(1)}|^2 p_1^{(1)}}{|h_{12}^{(1)}|^2 p_2^{(1)} + 1}\right) - \log\left(1 + \frac{|h_{22}^{(1)}|^2 p_2^{(1)}}{|h_{21}^{(1)}|^2 p_1^{(1)} + 1}\right) \\ &- \log\left(1 + \frac{|h_{11}^{(2)}|^2 p_1^{(2)}}{|h_{12}^{(2)}|^2 p_2^{(2)} + 1}\right) - \log\left(1 + \frac{|h_{22}^{(2)}|^2 p_2^{(2)}}{|h_{21}^{(2)}|^2 p_1^{(2)} + 1}\right). \end{aligned} \tag{25}$$

Based on the objective function expression (25), we can obtain that

$$\begin{cases} \frac{\partial f_{\text{unsup}}}{\partial p_1^{(n)}} = -\frac{|h_{11}^{(n)}|^2}{|h_{11}^{(n)}|^2 p_1^{(n)} + |h_{12}^{(n)}|^2 p_2^{(n)} + 1} + \frac{|h_{21}^{(n)}|^2 |h_{22}^{(n)}|^2 p_2^{(n)}}{\left(|h_{21}^{(n)}|^2 p_1^{(n)} + |h_{22}^{(n)}|^2 p_2^{(n)} + 1\right)\left(|h_{21}^{(n)}|^2 p_1^{(n)} + 1\right)} \\ \frac{\partial f_{\text{unsup}}}{\partial p_2^{(n)}} = -\frac{|h_{22}^{(n)}|^2}{|h_{21}^{(n)}|^2 p_1^{(n)} + |h_{22}^{(n)}|^2 p_2^{(n)} + 1} + \frac{|h_{11}^{(n)}|^2 |h_{12}^{(n)}|^2 p_1^{(n)}}{\left(|h_{12}^{(n)}|^2 p_2^{(n)} + |h_{11}^{(n)}|^2 p_1^{(n)} + 1\right)\left(|h_{12}^{(n)}|^2 p_2^{(n)} + 1\right)} \end{cases}. \tag{26}$$

Note that it is always possible to find a feasible $p_1^{(n)}$, $p_2^{(n)}$ such that the following holds

$$\frac{2(2 + h_{11}^{(n)})|h_{22}^{(n)}|^2}{|h_{11}^{(n)}|^2 |h_{12}^{(n)}|^2} < p_1^{(n)} < 1, \quad 0 < p_2^{(n)} < \min\left\{\frac{|h_{11}^{(n)}|^2}{(|h_{11}^{(n)}|^2 + |h_{12}^{(n)}|^2 + 1)|h_{21}^{(n)}|^2 |h_{22}^{(n)}|^2}, \frac{1}{|h_{12}^{(n)}|^2}\right\}, \quad \forall\, n \in \{1, 2\}, \tag{27}$$

where the first relation holds because of (23), and the second relations holds trivially. Using these two relations, we can show that the gradient expression (26) satisfies the following:

$$\begin{aligned} \frac{\partial f_{\text{unsup}}}{\partial p_1^{(n)}} &\leq -\frac{|h_{11}^{(n)}|^2}{|h_{11}^{(n)}|^2 + |h_{12}^{(n)}|^2 + 1} + |h_{21}^{(n)}|^2 |h_{22}^{(n)}|^2 p_2^{(n)} \\ &< -\frac{|h_{11}^{(n)}|^2}{|h_{11}^{(n)}|^2 + |h_{12}^{(n)}|^2 + 1} + |h_{21}^{(n)}|^2 |h_{22}^{(n)}|^2 \cdot \frac{|h_{11}^{(n)}|^2}{(|h_{11}^{(n)}|^2 + |h_{12}^{(n)}|^2 + 1)|h_{21}^{(n)}|^2 |h_{22}^{(n)}|^2} \\ &< 0 \\ \frac{\partial f_{\text{unsup}}}{\partial p_2^{(n)}} &\geq -\frac{|h_{22}^{(n)}|^2}{|h_{21}^{(n)}|^2 p_1^{(n)} + 1} + \frac{|h_{11}^{(n)}|^2 |h_{12}^{(n)}|^2 p_1^{(n)}}{\left(|h_{12}^{(n)}|^2 p_2^{(n)} + |h_{11}^{(n)}|^2 + 1\right)\left(|h_{12}^{(n)}|^2 p_2^{(n)} + 1\right)} \\ &> -|h_{22}^{(n)}|^2 + \frac{|h_{11}^{(n)}|^2 |h_{12}^{(n)}|^2 p_1^{(n)}}{2\left(|h_{11}^{(n)}|^2 + 2\right)} \\ &> 0. \end{aligned}$$

That is, there exists a region $N_{\epsilon^*}(\mathbf{p}^*)$ of power allocation around $\mathbf{p}^* = [1; 0; 1; 0]$, where $\frac{\partial f_{\text{unsup}}}{\partial p_1^{(n)}} < 0$ and $\frac{\partial f_{\text{unsup}}}{\partial p_2^{(n)}} > 0$ hold true for $n = 1, 2$. Then as discussed before, we have a contradiction to (24), and the proof of step 1 is completed.

Proof of Step 2. Next we show that for every $\widetilde{\mathbf{\Theta}}$ such that $\mathbf{p}\left(\widetilde{\mathbf{\Theta}}, |\mathbf{h}|\right) = \mathbf{p}^*$, there exists a region $N_\delta(\widetilde{\mathbf{\Theta}})$, such that for all $\mathbf{\Theta} \in N_\delta(\widetilde{\mathbf{\Theta}})$ and $\mathbf{\Theta}$ is feasible, $\mathbf{p}(\mathbf{\Theta}, |\mathbf{h}|)$ is feasible and falls in $N_{\epsilon^*}(\mathbf{p}^*)$ identified in the previous step. That is, (20) holds true.

Notice that the output of the linear neural network is $\mathbf{p} = \mathbf{\Theta}|\mathbf{h}|$, which is a continuous function of $\mathbf{\Theta}$. Let us fix a $\widetilde{\mathbf{\Theta}}$ satisfying $\mathbf{p}^* = \widetilde{\mathbf{\Theta}}|\mathbf{h}|$. Then by using the property of a continuous function, for the constant $\epsilon^* > 0$ identified in the previous step, there always exists δ, such that when $\|\mathbf{\Theta} - \widetilde{\mathbf{\Theta}}\| \leq \delta$, and when $\mathbf{0} \leq \mathbf{\Theta}|\mathbf{h}| \leq \mathbf{1}$, the following holds

$$\|\mathbf{p}(\mathbf{\Theta}, |\mathbf{h}|) - \mathbf{p}(\widetilde{\mathbf{\Theta}}, |\mathbf{h}|)\| \leq \epsilon^*.$$

In Step 1, we have shown that when a feasible $\mathbf{p} = \mathbf{p}(\mathbf{\Theta}, |\mathbf{h}|)$ falls in $N_{\epsilon^*}(\mathbf{p}^*)$, then $f_{\text{unsup}}(\mathbf{p}^*) - f_{\text{unsup}}(\mathbf{p}) \leq 0$.

In conclusion, we showed that there exist channel realizations, for which $\mathbf{\Theta}_{\text{local}}$ satisfying $\mathbf{\Theta}_{\text{local}}|\mathbf{h}| = [1; 0; 1; 0]$, for which the following holds true:

$$f_{\text{unsup}}(\mathbf{\Theta}) - f(\mathbf{\Theta}_{\text{local}}) \geq 0, \text{ for all } \mathbf{\Theta} \in N_\delta(\mathbf{\Theta}_{,\text{local}}) \text{ and } \mathbf{p}(\mathbf{\Theta}, |\mathbf{h}^{(1)}|) \text{ and } \mathbf{p}(\mathbf{\Theta}, |\mathbf{h}^{(2)}|) \text{ are feasible.} \tag{28}$$

By definition, such a $\mathbf{\Theta}_{\text{local}}$ is a local optimal solution of (4).

Supervised learning problem (3) We will check that $f_{\text{sup}}(\mathbf{\Theta})$ is a convex function w.r.t. $\mathbf{\Theta}$. For convenience, let us explicitly write down the expression for the output of the neural network as follows

$$f_{\text{sup}}(\mathbf{\Theta}) = \sum_{n=1}^{2} \sum_{k=1}^{2} \left(\mathbf{\Theta}_k \cdot |\widetilde{\mathbf{h}}^{(n)}| - \bar{p}_k^{(n)}\right)^2.$$

It is clear that the objective is a convex quadratic function of $\mathbf{\Theta}$.

Meanwhile, we know that $\mathbf{\Theta}$ contains 8 scalar parameters and there are four linear equations to be solved, which are given below:

$$\mathbf{\Theta}|\mathbf{h}^{(1)}| = [0; 1], \quad \mathbf{\Theta}|\mathbf{h}^{(2)}| = [1; 0].$$

It follows that as long as the channel realizations are randomly generated so that they are linearly independent, there always exists $\mathbf{\Theta}$ which can predict the true labels. That is, at the global optimal solution of the unsupervised learning, the objective value will be zero. □

B. Proof of Claim 2

Claim 2. *Consider an SISO-IC training problem with K users and N training samples. Suppose the following hold:*
i). For each data sample $n \in \{1, \cdots, N\}$, we can generate a stationary solution $\bar{\mathbf{p}}^{(n)}$ of (1) *as the training label.*
ii). Let $\mathbf{\Theta}^*(\bar{\mathbf{p}})$ denote the optimal solution for the SL problem (3) *with label $\bar{\mathbf{p}}$, and it achieves zero loss: $f_{\sup}(\mathbf{\Theta}^*(\bar{\mathbf{p}})) = 0$.*
iii) The solution $\mathbf{\Theta}^*(\bar{\mathbf{p}})$ can be computed for all $\bar{\mathbf{p}}$.
Let $\mathcal{B}$ denote the set of stationary points of (4) *which satisfy the KKT condition. Then the following holds:*

$$\{\mathbf{\Theta}^*(\bar{\mathbf{p}}) \mid \bar{\mathbf{p}}^{(n)} \text{ is a stationary solution of (1), } \forall\ n\} \subseteq \mathcal{B}. \tag{29}$$

Proof. The main idea of the proof is as follows. First, we characterize the set of optimal solutions of SL problem (3), under the zero training loss conditions. Second, we show that each one of such optimal solution is also a stationary point of UL loss (4), which is a point that satisfies the KKT condition. Finally, we show that there exists a solution in $\mathcal{B}$ which does not optimize the supervised problem (3).

To begin with, let us denote $[K] := \{1, 2, \cdots, K\}$ and $[N] := \{1, 2, \cdots, N\}$. With our assumption that zero loss can be achieved at $\mathbf{\Theta}^*(\bar{\mathbf{p}})$, the following holds true:

$$p_k(\mathbf{\Theta}^*(\bar{\mathbf{p}}); |\mathbf{h}^{(n)}|) = \bar{p}_k^{(n)}, \quad \forall\ n \in [N],\ \forall\ k \in [K]. \tag{30}$$

Next, we verify that $\mathbf{\Theta}^*(\bar{\mathbf{p}})$ is a subset of stationary points of UL (4). Recall that the stationary solutions are those satisfy the KKT condition, so we aim to check this by showing that $\mathbf{\Theta}^*(\bar{\mathbf{p}})$ satisfies the KKT condition for the UL problem (4). Towards this end, let us write down the Lagrangian for the UL problem (4) as:

$$\begin{aligned} L_{\mathrm{UL}}(\mathbf{p}(\mathbf{\Theta}; |\mathbf{h}^{(n)}|), \boldsymbol{\lambda}, \boldsymbol{\mu}) = &\sum_{n=1}^{N} -R(\mathbf{p}(\mathbf{\Theta}; |\mathbf{h}^{(n)}|), |\mathbf{h}^{(n)}|) - \sum_{n=1}^{N}\sum_{k=1}^{K} \lambda_k^{(n)} p_k(\mathbf{\Theta}; |\mathbf{h}^{(n)}|) \\ &+ \sum_{n=1}^{N}\sum_{k=1}^{K} \mu_k^{(n)} (p_k(\mathbf{\Theta}; |\mathbf{h}^{(n)}|) - P_{\max}). \end{aligned} \tag{31}$$

The KKT condition for problem (4) is that, there is a tuple $(\tilde{\mathbf{\Theta}}, \tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\mu}})$ such that the following set of relations holds (for all $k \in [K], n \in [N], (u, v) \in W$:

$$\begin{cases} \frac{\partial L_{\mathrm{UL}}\left(\mathbf{p}\left(\tilde{\mathbf{\Theta}}; |\mathbf{h}^{(n)}|\right), \tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\mu}}\right)}{\partial \mathbf{\Theta}_{k,(u,v)}} = -\frac{\sum_{n=1}^{N} R\left(\mathbf{p}\left(\tilde{\mathbf{\Theta}}; |\mathbf{h}^{(n)}|\right); |\mathbf{h}^{(n)}|\right)}{\partial p_k^{(n)}} \cdot \frac{\partial p_k(\tilde{\mathbf{\Theta}}; |\mathbf{h}^{(n)}|)}{\mathbf{\Theta}_{k,(u,v)}} \\ \qquad\qquad - \sum_{n=1}^{N}\sum_{k=1}^{K} \tilde{\lambda}_k^{(n)} \cdot \frac{\partial p_k(\tilde{\mathbf{\Theta}}; |\mathbf{h}^{(n)}|)}{\partial \mathbf{\Theta}_{k,(u,v)}} + \sum_{n=1}^{N}\sum_{k=1}^{K} \tilde{\mu}_k^{(n)} \cdot \frac{\partial p_k(\tilde{\mathbf{\Theta}}; |\mathbf{h}^{(n)}|)}{\partial \mathbf{\Theta}_{k,(u,v)}} = 0 \\ \mathbf{0} \le \mathbf{p}(\tilde{\mathbf{\Theta}}; \mathbf{h}) \le \mathbf{P}_{\max} \\ \tilde{\lambda}_k^{(n)} \ge 0 \\ \tilde{\mu}_k^{(n)} \ge 0 \\ \tilde{\lambda}_k^{(n)} \cdot p_k(\tilde{\mathbf{\Theta}}; |\mathbf{h}^{(n)}|) = 0 \\ \tilde{\mu}_k^{(n)} \cdot \left(p_k(\tilde{\mathbf{\Theta}}; |\mathbf{h}^{(n)}|) - P_{\max}\right) = 0 \end{cases}. \tag{32}$$

To show that $\mathbf{\Theta}^*(\bar{\mathbf{p}})$ (together with some multipliers) will satisfy (32), we will utilize the zero-loss property (30) and the fact that $\bar{\mathbf{p}}^{(n)}$ is a stationary solution for problem (1) evaluated at each data points $\mathbf{h}^{(n)}$.

Define the Lagrangian function for problem (1) as:

$$L_{\mathrm{WSR}}^{(n)}(\mathbf{p}^{(n)}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = -R(\mathbf{p}^{(n)}, |\mathbf{h}^{(n)}|) - \sum_{k=1}^{K} \lambda_k^{(n)} p_k^{(n)} + \sum_{k=1}^{K} \mu_k^{(n)} (p_k^{(n)} - P_{\max}). \tag{33}$$

Since by assumption $\bar{\mathbf{p}}^{(n)}$ is a stationary solution for $n \in [N]$, so there exists a tuple $(\bar{\mathbf{p}}^{(n)}, \bar{\boldsymbol{\lambda}}^{(n)}, \bar{\boldsymbol{\mu}}^{(n)})$ such that the following holds true for all $k \in [K]$

$$\begin{cases} \frac{\partial L_{\mathrm{WSR}}^{(n)}\left(\bar{\mathbf{p}}^{(n)}, \bar{\boldsymbol{\lambda}}, \bar{\boldsymbol{\mu}}\right)}{\partial p_k^{(n)}} = -\frac{\partial R(\bar{\mathbf{p}}^{(n)}, |\mathbf{h}^{(n)}|)}{\partial p_k^{(n)}} - \sum_{k=1}^K \bar{\lambda}_k^{(n)} + \sum_{k=1}^K \bar{\mu}_k^{(n)} = 0 \\ 0 \le \bar{p}_k^{(n)} \le P_{\max} \\ \bar{\lambda}_k^{(n)} \ge 0 \\ \bar{\mu}_k^{(n)} \ge 0 \\ \bar{\lambda}_k^{(n)} \bar{p}_k^{(n)} = 0 \\ \bar{\mu}_k^{(n)} (\bar{p}_k^{(n)} - P_{\max}) = 0 \end{cases} \tag{34}$$

Now we argue that tuple $(\boldsymbol{\Theta}^*(\bar{\mathbf{p}}), \bar{\boldsymbol{\lambda}}, \bar{\boldsymbol{\mu}})$ satisfies the KKT condition in (32). Since we have $\bar{p}_k^{(n)} = p_k(\boldsymbol{\Theta}^*(\bar{\mathbf{p}}); |\mathbf{h}^{(n)}|)$ for $k \in [K],\ n \in [N]$, it is obvious that the second to the last relation holds in (32). To verify that the first relation in (32) holds, we have:

$$\begin{aligned} \frac{\partial L_{\mathrm{UL}}\left(\mathbf{p}\left(\boldsymbol{\Theta}^*; |\mathbf{h}|\right), \bar{\boldsymbol{\lambda}}, \bar{\boldsymbol{\mu}}\right)}{\partial \boldsymbol{\Theta}_{k,(u,v)}} &= \sum_{n=1}^N \left(-\frac{\partial R\left(\mathbf{p}(\boldsymbol{\Theta}^*(\bar{\mathbf{p}}); |\mathbf{h}^{(n)}|), |\mathbf{h}^{(n)}|\right)}{\partial p_k^{(n)}} - \sum_{k=1}^K \bar{\lambda}_k^{(n)} + \sum_{k=1}^K \bar{\mu}_k^{(n)} \right) \frac{\partial p_k\left(\boldsymbol{\Theta}^*(\bar{\mathbf{p}}); |\mathbf{h}^{(n)}|\right)}{\partial \boldsymbol{\Theta}_{k,(u,v)}} \\ &= \sum_{n=1}^N \frac{\partial L_{\mathrm{WSR}}^{(n)}\left(\bar{\mathbf{p}}^{(n)}, \bar{\boldsymbol{\lambda}}, \bar{\boldsymbol{\mu}}\right)}{\partial p_k^{(n)}} \frac{\partial p_k\left(\boldsymbol{\Theta}^*(\bar{\mathbf{p}}); |\mathbf{h}^{(n)}|\right)}{\partial \boldsymbol{\Theta}_{k,(u,v)}} \\ &= 0 \end{aligned}$$

where the second equality comes from the zero-loss property; the last equation comes from the stationary condition in (34). Thus, we have found a feasible tuple $(\boldsymbol{\Theta}^*(\bar{\mathbf{p}}), \bar{\boldsymbol{\lambda}}, \bar{\boldsymbol{\mu}})$ that satisfies the KKT condition in (32). Hence, $\boldsymbol{\Theta}^*(\bar{\mathbf{p}})$ is a stationary solution of UL (4) and $\boldsymbol{\Theta}^*(\bar{\mathbf{p}}) \subseteq \mathcal{B}$.

Finally, it is easy to show that there exists a solution in $\mathcal{B}$ that is not an optimal solution for (3). Consider the example we construct in Claim 1, which has a global optimal solution at $[0; 1; 1; 0]$, and a local solution $\mathbf{p}^* = \mathbf{p}\left(\boldsymbol{\Theta}_{\text{local}}, |\mathbf{h}|\right) = [1; 0; 1; 0]$. In this example, $\boldsymbol{\Theta}_{\text{local}}$ is local minimum, which is also the stationary point. However, it does not produce the optimal label, so $f_{\text{unsup}}(\boldsymbol{\Theta})$ does not achieve minimum at $\boldsymbol{\Theta}_{\text{local}}$. And it is easy to check that $\boldsymbol{\Theta}_{\text{local}}$ does not optimize the supervised problem (3) because the zero loss condition is not satisfied. So $\boldsymbol{\Theta}_{\text{local}} \in \mathcal{B}$ but not a sationary solution of (3). □

C. Proof of Claim 3

Before we show the proof for Claim 3, let us re-state our objective function and introduce some notations. Recall that in Section III we have defined $f_l = \mathrm{vec}(F_l), l \in [L]$ and $y = \mathrm{vec}(\bar{\mathbf{p}})$, which are the vectorized output of the l-th layer and the vectorized label, respectively. Notice that

$$\begin{aligned} f_{\text{sup}}(\boldsymbol{\Theta}) &= \sum_{n=1}^N \left\| \mathbf{p}\left(\boldsymbol{\Theta}; \left|\mathbf{h}^{(n)}\right|\right) - \bar{\mathbf{p}}^{(n)} \right\|^2 \\ &= (F_L(\boldsymbol{\Theta}) - y)^T (F_L(\boldsymbol{\Theta}) - y) \end{aligned} \tag{35}$$

Later we will use (35) as the expression of the unconstrained SL objective function.

In the UL training problem, recall that we still use the fully connected network with the structure define in (8). However, in order to cast the training problem into an *unconstrained* problem, a sigmoid function should be added to the last layer of the output to enforce feasibility. The modified network has the following output:

$$F_L = \mathrm{sig}(F_{L-1}\boldsymbol{\Theta}_L) = \frac{\mathbf{1} \times P_{\max}}{1 + e^{-F_{L-1}\boldsymbol{\Theta}_L}}, \tag{36}$$

where F_L and $\boldsymbol{\Theta}_L$ is defined in Section III. The output F_L is the allocated power. Now our objective function is converted to the unconstrained version of (4):

$$f_{\mathrm{UL}}(\boldsymbol{\Theta}) := \sum_{n=1}^N -R\left(\tilde{\mathbf{p}}(\boldsymbol{\Theta}; |\mathbf{h}^{(n)}|), |\mathbf{h}^{(n)}|\right) \tag{37}$$

where $\tilde{\mathbf{p}}$ is the our output of the neural network of unconstrained UL problem, which is also the allocated power.

Next, let us further define some notations. Let $\otimes$ denote the Kronecker product. Recall that $\boldsymbol{\Theta} = (W_l)_{l=1}$ denotes all the parameters in (37), and $\boldsymbol{\Theta}^m = (W_l^m)_{l=1}^L\, L, W_l, l = 1, 2, \cdots, L$ denote all the parameters in the m-th iteration and parameters in the l-th layer in the m-th iteration respectively; given N samples, let us define

$$\Sigma_l := \mathrm{diag}\left[\mathrm{vec}\left(\sigma'\left(F_{l-1}W_l\right)\right)\right] \in \mathbb{R}^{Nn_l \times Nn_l}, \quad \Sigma_l^m := \Sigma_l\left(\boldsymbol{\Theta}^m\right),$$

where Σ_l represents the derivative of activation at each layer l; Σ_l^m means the derivative of activation of each layer at iteration m. Further, define $F_l^m := F_l\left(\mathbf{\Theta}^m\right) \in \mathbb{R}^{N\times n_L}$ as the output of the l-th layer at iteration m for all samples. Note that this notation vectorizes the output of all the samples.Denote JF_L as the Jacobian of the network, that is

$$Jf_L = \left[\frac{\partial f_L}{\partial \operatorname{vec}(W_1)}, \ldots, \frac{\partial f_L}{\partial \operatorname{vec}(W_L)}\right], \text{ where } \frac{\partial f_L}{\partial \operatorname{vec}(W_l)} \in \mathbb{R}^{(Nn_L)\times(n_{l-1}n_l)} \text{ for } l \in [L]. \tag{38}$$

We first write down the assumptions and lemmas needed in the proof.

Assumption 1 (*Pyramidal network topology, [18, Assumption 2.1]) Let $n_1 \geq N$ and $n_2 \geq n_3 \geq \ldots \geq n_L$.*

Assumption 2 *(Activation function, [18, Assumption 2.2]) Fix $\gamma \in (0,1)$ and $\beta > 0$. Let σ satisfy that: (i) $\sigma'(x) \in [\gamma, 1], (ii)|\sigma(x)| \leq |x|$ for every $x \in \mathbb{R}$, and $(iii)\sigma'$ is β - Lipschitz.*

Assumption 3 *(Initial conditions, [18, Assumption 3.1])* Assume that the following holds:

$$\lambda_F^2 \geq \frac{\gamma^4}{3}\left(\frac{6}{\gamma^2}\right)^L \|\mathbf{h}\|_F \sqrt{f_{\sup}\left(\mathbf{\Theta}^0\right)} \frac{\bar{\lambda}_{3\to L}}{\lambda_{3\to L}^2} \max\left(\frac{2\bar{\lambda}_1\bar{\lambda}_2}{\min_{l\in\{3,\ldots,L\}}\lambda_l\bar{\lambda}_l}, \bar{\lambda}_1, \bar{\lambda}_2\right),$$
$$\lambda_F^3 \geq \frac{2\gamma^4}{3}\left(\frac{6}{\gamma^2}\right)^L \|\mathbf{h}\|_2\|\mathbf{h}\|_F \sqrt{f_{\sup}\left(\mathbf{\Theta}^0\right)} \frac{\bar{\lambda}_{3\to L}}{\lambda_{3\to L}^2}\bar{\lambda}_2.$$

where $\bar{\lambda}_l, \lambda_l, l = 1, 2, \cdots, L$ and λ_F are defined in (9). This assumption provides an initialization condition that the parameters are not far away from the global optimum since $f_{\sup}(\mathbf{\Theta}^0)$ could not be very large. Furthermore, the assumption regularizes the feature matrix $\mathbf{h}$, which means that channel samples should not be highly linearly dependent, because otherwise λ_F will be small.

Lemma 1. *([18, Lemma 4.1]) Let Assumption 1 hold. Then, for the unsupervised loss function f_{UL} the following results hold:*

$$\operatorname{vec}\left(\nabla_{W_l} f_{\mathrm{UL}}\right) = \left(\mathbb{I}_{n_l} \otimes F_{l-1}^T\right) \prod_{q=l+1}^{L} \Sigma_{q-1}\left(W_q \otimes \mathbb{I}_N\right) \Sigma_L \frac{\partial f_{\mathrm{UL}}}{\partial \tilde{\mathbf{p}}},$$
$$\frac{\partial f_L}{\partial \operatorname{vec}(W_l)} = \Sigma_L \prod_{q=0}^{L-l-1} \left(W_{L-q}^T \otimes \mathbb{I}_N\right) \Sigma_{L-q-1}\left(\mathbb{I}_{n_l} \otimes F_{l-1}\right).$$

The above lemma provides expressions of the gradient of objective function f_{UL}.

Lemma 2. *([18, Lemma 4.2]) Let Assumption 2 hold. For every $\mathbf{\Theta} = (W_q)_{l=1}^L$ in f_{UL} the following holds*

$$\|F_l\|_F \leq \|\mathbf{h}\|_F \prod_{q=1}^{l} \|W_q\|_2, \quad \forall\, l \in [L-1], \quad \text{and} \quad \|F_L\|_F \leq \sqrt{Nn_L} \tag{39}$$

$$\|\nabla_{W_l} f_{\mathrm{UL}}\|_F \leq \|\mathbf{h}\|_F \prod_{\substack{q=1\\q\neq l}}^{L} \|W_q\|_2 \left\|\frac{f_{\mathrm{UL}}}{\partial \tilde{\mathbf{p}}}\right\|_2, \quad \forall\, l \in [L] \tag{40}$$

Furthermore, let $\mathbf{\Theta}^a := (W_l^a)_{l=1}^L, \mathbf{\Theta}^b := \left(W_l^b\right)_{l=1}^L$, and $\bar{\lambda}_l \geq \max\left(\|W_l^a\|_2, \left\|W_l^b\right\|_2\right)$ for some scalars $\bar{\lambda}_l$. Let $R := \prod_{q=1}^L \max\left(1, \bar{\lambda}_q\right)$, β' be the Lipschitz constant of the gradient of sigmoid function, i.e, $\forall x, z, \|\mathrm{sigmoid}'(x) - \mathrm{sigmoid}'(z)\|_2 \leq \beta'\|x - z\|_2$. Then, for $l \in [L]$,

$$\left\|F_L^a - F_L^b\right\|_F \leq \frac{1}{4}\sqrt{LNn_L}\|\mathbf{h}\|_F \frac{\prod_{l=1}^L \bar{\lambda}_l}{\min_{l\in[L]}\bar{\lambda}_l} \left\|\mathbf{\Theta}^a - \mathbf{\Theta}^b\right\|_2 \tag{41}$$

$$\left\|\frac{\partial f_L\left(\mathbf{\Theta}^a\right)}{\partial \operatorname{vec}\left(W_l^a\right)} - \frac{\partial f_L(\mathbf{\Theta}^b)}{\partial \operatorname{vec}\left(W_l^b\right)}\right\|_2 \leq \sqrt{L}\|\mathbf{h}\|_F R\left(1 + L\beta\|\mathbf{h}\|_F R + \beta'\|\mathbf{h}\|_F R\right)\left\|\mathbf{\Theta}^a - \mathbf{\Theta}^b\right\|_2. \tag{42}$$

First, the above lemma provides an upper bound for the output of each layer and the gradient for weight in each layer. Second, it shows how 'smooth' is the network and the gradient. This lemma is slightly different from [18, Lemma 4.2] since we need to adapt to the last layer with additional sigmoid activation.

Lemma 3. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a C^2 function. For any $x, y \in \mathbb{R}^n$ be given, and assume that $\|\nabla f(y) - \nabla f(x)\|_2 \leq C\|y - x\|_2$. Then,*

$$f(y) \leq f(x) + \langle \nabla f(x), y - x\rangle + \frac{C}{2}\|x - y\|^2. \tag{43}$$

We are now ready to prove Claim 3.

Claim 3 *Consider an SISO-IC training problem with K users and N training samples. Let $P_{max} = 1, \sigma = 1$. Construct a fully connected neural network satisfying Assumption 1 - 2. Then the following holds:*
(a) When the initialization condition satisfies Assumption 3, consider optimizing the unconstrained version of (3) *using the gradient descent algorithm*

$$\mathbf{\Theta}^{m+1} = \mathbf{\Theta}^m - \eta \nabla f_{\text{sup}}\left(\mathbf{\Theta}^m\right).$$

There exists a stepsize $\eta > 0$ such that the training loss converges to zero at a geometric rate, that is:

$$f_{\text{sup}}\left(\mathbf{\Theta}^m\right) \leq \left(1 - \eta\alpha_0\right)^m f_{\text{sup}}\left(\mathbf{\Theta}^0\right), \tag{44}$$

where $\alpha_0 = \frac{4}{\gamma^4}\left(\frac{\gamma^2}{4}\right)^L \lambda_F^2 \lambda_{3\to L}^2$.
(b) Consider minimizing the unconstrained version of (4)*, which is* (37) *using the last layer as* (10) *and use the gradient descent algorithm:*

$$\mathbf{\Theta}^{m+1} = \mathbf{\Theta}^m - \eta \nabla f_{\text{UL}}(\mathbf{\Theta}^m).$$

Suppose all the weights are bounded during training, then $\mathbf{\Theta}$ will converge to a stationary point of the training objective.

Proof. Claim 3-(a) is a direct application of [18, Theorem 3.2], so we do not include the proof here.

For Claim 3-(b), we first state our sketch of the proof. The idea is similar to [18, Theorem 3.2]. However, the objective function is not the squared loss but the sum-rate (37). This function has a more complex structure and is no longer strictly convex over the output of the neural network $\mathbf{p}$. Therefore it has more complicated optimization landscape. Moreover, the linear convergence of the form (44) is not possible, since with the sum-rate as the objective, even the global min can be found, the training objective will not shrink to zero. Finally, with the last layer using sigmoid function (10), vanishing gradient may occur and showing fast decrease is more difficult. Overall, it is intuitive that this is a much harder problem than the first one.

Our proof step is as follows. First we will verify that the Lipschitz condition (43) holds at every iteration, and this will consist the main part of the proof. Second, we will show that the objective decreases until converting to a stationary solution.
Step 1: At each iteration m, we will show that, there exists a constant C such that the following holds:

$$\left\|\nabla f_{\text{UL}}\left(\mathbf{\Theta}^{m+1}\right) - \nabla f_{\text{UL}}\left(\mathbf{\Theta}^m\right)\right\|_2 \leq C \cdot \left\|\mathbf{\Theta}^{m+1} - \mathbf{\Theta}^m\right\|_2. \tag{45}$$

Denote $g(\mathbf{\Theta}^m) := \text{vec}\left(\frac{\partial f_{\text{UL}}(\mathbf{\Theta}^m)}{\partial \tilde{\mathbf{p}}}\right)$. Rewrite (45) and by triangle inequality,

$$\begin{aligned}\left\|\nabla f_{\text{UL}}\left(\mathbf{\Theta}^{m+1}\right) - \nabla f_{\text{UL}}\left(\mathbf{\Theta}^m\right)\right\|_2 &= \left\|Jf_L\left(\mathbf{\Theta}^{m+1}\right)^T g\left(\mathbf{\Theta}^{m+1}\right) - Jf_L\left(\mathbf{\Theta}^m\right)^T g\left(\mathbf{\Theta}^m\right)\right\|_2 \\ &\leq \left|\left|g\left(\mathbf{\Theta}^{m+1}\right) - g\left(\mathbf{\Theta}^m\right)\right|\right|_2 \left\|Jf_L\left(\mathbf{\Theta}^{m+1}\right)\right\|_2 + \left\|Jf_L\left(\mathbf{\Theta}^{m+1}\right) - Jf_L\left(\mathbf{\Theta}^m\right)\right\|_2 \left\|g\left(\mathbf{\Theta}^m\right)\right\|_2.\end{aligned} \tag{46}$$

In the rest of the proof, we aim to bound each term in (46).
(Step 1.1) First, show that the following holds $||Jf_L(\mathbf{\Theta}^{m+1})||_2 \leq C_1$, for some constant $C_1 > 0$.

$$\begin{aligned}\left\|Jf_L\left(\mathbf{\Theta}^{m+1}\right)\right\|_2 &\overset{(i)}{\leq} \sum_{l=1}^{L}\left\|\frac{\partial f_L\left(\mathbf{\Theta}^{m+1}\right)}{\partial \,\text{vec}\left(W_l\right)}\right\|_2 \overset{(ii)}{=} \sum_{l=1}^{L}\prod_{q=0}^{L-l-1}\left|\left|\Sigma_L\left(W_{L-q}^T(\mathbf{\Theta}^{m+1}) \otimes \mathbb{I}_N\right)\Sigma_{L-q-1}\left(\mathbb{I}_{n_l} \otimes F_{l-1}(\mathbf{\Theta}^{m+1})\right)\right|\right|_2 \\ &\overset{(iii)}{\leq} \sum_{l=1}^{L}\prod_{q=l+1}^{L}\left\|W_q\left(\mathbf{\Theta}^{m+1}\right)\right\|_2 \left\|F_{l-1}\left(\mathbf{\Theta}^{m+1}\right)\right\|_2 \\ &\overset{(iv)}{\leq} \sum_{l=1}^{L}\prod_{q=l+1}^{L}\left\|W_q^{m+1}\right\|_2 \|F_{l-1}^{m+1}\|_F \\ &\overset{(v)}{\leq} \|\mathbf{h}\|_F \sum_{l=1}^{L}\prod_{q=l+1}^{L}\left\|W_q^{m+1}\right\|_2 \prod_{q=1}^{l}\|W_q\|_2 \\ &= \|\mathbf{h}\|_F \sum_{l=1}^{L}\prod_{\substack{q=1 \\ q\neq l}}^{L}\left\|W_q^{m+1}\right\|_2\end{aligned}$$

where (i) is because of Cauchy-Schwards inequality; (ii) comes from Lemma 1; (iii) follows Assumption 2 that activation function at each layer satisfies $0 < \sigma' < 1$ (including the last layer with sigmoid activation); (iv) is because Frobenius norm is always no less than l_2 norm; (v) comes from Lemma 2.

With the assumption in Claim 3 that all the weights are bounded during training, it is obvious that for any $q \in [L]$, $\|W_q^{m+1}\|_2$ is bounded. With this assumption, it is easy to see that the Jacobian $Jf_L\left(\mathbf{\Theta}^{m+1}\right)$ is bounded given fixed N samples. Thus

we can find C_1 such that $||Jf_L(\mathbf{\Theta}_t^m)|| \leq C_1$.
(Step 1.2) Next, show that $\left\|Jf_L\left(\mathbf{\Theta}^{m+1}\right) - Jf_L\left(\mathbf{\Theta}^m\right)\right\|_2 \leq C_2 \left\|\mathbf{\Theta}^{m+1} - \mathbf{\Theta}^m\right\|_2$. By Lemma 2, we have

$$\begin{aligned}\left\|Jf_L\left(\mathbf{\Theta}^{m+1}\right) - Jf_L\left(\mathbf{\Theta}^m\right)\right\|_2 &\leq \sum_{l=1}^{L}\left\|\frac{\partial f_L\left(\mathbf{\Theta}^{m+1}\right)}{\partial \operatorname{vec}\left(W_l\right)} - \frac{\partial f_L\left(\mathbf{\Theta}^m\right)}{\partial \operatorname{vec}(W_l)}\right\|_2 \\ &\leq \sqrt{L}\|\mathbf{h}\|_F R\left(1 + L\beta\|\mathbf{h}\|_F R + \beta^{'}\|\mathbf{h}\|_F R\right)\left\|\mathbf{\Theta}^{m+1} - \mathbf{\Theta}^m\right\|_2\end{aligned} \tag{47}$$

where $R = \prod_{p=1}^{L}\max\left(1, \bar{\lambda}_p\right)$. The first in equality is because of Cauchy-Schwards inequality; the second inequality comes from Lemma 2. Notice that when all the weights are bounded, R is bounded. So we can find C_2 such that

$$\left\|Jf_L\left(\mathbf{\Theta}_t^m\right) - Jf_L\left(\mathbf{\Theta}^m\right)\right\|_2 \leq C_2\left\|\mathbf{\Theta}_t^m - \mathbf{\Theta}^m\right\|_2 \tag{48}$$

(Step 1.3) Then we show $\|g\left(\mathbf{\Theta}^m\right)\| \leq C_3$. Denote the sum rate of n-th sample as $R_{\mathbf{\Theta}}^{(n)} := R\left(\mathbf{p}\left(\mathbf{\Theta}; \left|\mathbf{h}^{(1)}\right|\right), \left|\mathbf{h}^{(n)}\right|\right)$. The vectorized gradient $g(\mathbf{\Theta}^m)$ can be written as

$$g(\mathbf{\Theta}^m) = \left(-\frac{\partial R_{\mathbf{\Theta}^m}^{(1)}}{\partial \tilde{\mathbf{p}}_1^{(1)}}, \cdots, -\frac{\partial R_{\mathbf{\Theta}^m}^{(N)}}{\partial \mathbf{p}_1^{(N)}}, \cdots, -\frac{\partial R_{\mathbf{\Theta}^m}^{(1)}}{\partial \tilde{\mathbf{p}}_{nN_L}^{(1)}}, \cdots, -\frac{\partial R_{\mathbf{\Theta}^m}^{(N)}}{\partial \tilde{\mathbf{p}}_{nN_L}^{(N)}}\right)$$

Note that for $i = 1, 2, \cdots, nN_L, n = 1, 2, \cdots, N$, it is easy to show $\frac{\partial R_{\mathbf{\Theta}^m}^{(n)}}{\partial \tilde{\mathbf{p}}_i^{(n)}}$ is bounded (by using simple calculation and the assumption that all weights are bounded), so there exists constant C_3 such that

$$\left\|g\left(\mathbf{\Theta}^m\right)\right\|_2 \leq C_3$$

(Step 1.4) Finally, show $\left\|g(\mathbf{\Theta}^{m+1}) - g(\mathbf{\Theta}^m)\right\|_2 \leq C_3\|\mathbf{\Theta}^{m+1} - \mathbf{\Theta}^m\|$. Denote $g'(\mathbf{\Theta}) = \operatorname{vec}(\frac{\partial g(\mathbf{\Theta})}{\partial \tilde{\mathbf{p}}})$, which can be written as

$$g'(\mathbf{\Theta}) = \left(-\frac{\partial^2 R_{\mathbf{\Theta}}^{(1)}}{(\partial \tilde{\mathbf{p}}_1^{(1)})^2}, \cdots, -\frac{\partial^2 R_{\mathbf{\Theta}}^{(N)}}{(\partial \tilde{\mathbf{p}}_1^{(N)})^2}, \cdots, -\frac{\partial^2 R_{\mathbf{\Theta}}^{(1)}}{(\partial \tilde{\mathbf{p}}_{nN_L}^{(1)})^2}, \cdots, -\frac{\partial^2 R_{\mathbf{\Theta}}^{(N)}}{(\partial \tilde{\mathbf{p}}_{nN_L}^{(N)})^2}\right) \tag{49}$$

Notice that $g'(\mathbf{\Theta})$ denotes the continuous derivative of $g(\mathbf{\Theta})$ over $\tilde{\mathbf{p}}$. For every feasible $\mathbf{\Theta}$, it is easy to show that for $i = 1, 2, \cdots, nN_L, n = 1, 2, \cdots, N$, $\frac{\partial^2 R_{\mathbf{\Theta}}^{(n)}}{(\partial \tilde{\mathbf{p}}_i^{(n)})^2}$ is bounded. Then we aim to argue that there exists $C_4^{'}$ such that $\|g\left(\mathbf{\Theta}^{m+1}\right) - g\left(\mathbf{\Theta}^m\right)\|_2 \leq C_4^{'}\|\tilde{\mathbf{p}}(\mathbf{\Theta}^{m+1}); |\mathbf{h}| - \tilde{\mathbf{p}}(\mathbf{\Theta}^m; |\mathbf{h}|)\|_2$. By Mean Value Theorem, we can show that there exists $\tilde{\mathbf{p}}(\hat{\mathbf{\Theta}}; |\mathbf{h}|)$ between $\tilde{\mathbf{p}}(\mathbf{\Theta}^{m+1}; |\mathbf{h}|)$ and $\tilde{\mathbf{p}}(\mathbf{\Theta}^m; |\mathbf{h}|)$, such that

$$\begin{aligned}\left\|g\left(\mathbf{\Theta}^{m+1}\right) - g\left(\mathbf{\Theta}^m\right)\right\|_2 &= \left\|\left\langle g'(\hat{\mathbf{\Theta}}), \operatorname{vec}\left(\tilde{\mathbf{p}}(\mathbf{\Theta}^{m+1}; |\mathbf{h}|)\right) - \operatorname{vec}\left(\tilde{\mathbf{p}}(\mathbf{\Theta}^m; |\mathbf{h}|)\right)\right\rangle\right\|_2 &(50)\\ &\leq \left\|g'(\hat{\mathbf{\Theta}})\right\|_2 \left\|\operatorname{vec}\left(\tilde{\mathbf{p}}(\mathbf{\Theta}^{m+1}; |\mathbf{h}|)\right) - \operatorname{vec}\left(\tilde{\mathbf{p}}(\mathbf{\Theta}^m; |\mathbf{h}|)\right)\right\|_2 &(51)\\ &\leq C_4^{'}\left\|\operatorname{vec}\left(\tilde{\mathbf{p}}(\mathbf{\Theta}^{m+1}; |\mathbf{h}|)\right) - \operatorname{vec}\left(\tilde{\mathbf{p}}(\mathbf{\Theta}^m; |\mathbf{h}|)\right)\right\|_2, &(52)\end{aligned}$$

where the first inequality comes from Cauchy-Schwards inequality; the second inequality is because each component of $\|g'(\hat{\mathbf{\Theta}})\|_2$ is bounded. By Lemma 2, we know that the following holds:

$$\begin{aligned}\left\|\operatorname{vec}\left(\tilde{\mathbf{p}}(\mathbf{\Theta}^{m+1}; |\mathbf{h}|)\right) - \operatorname{vec}\left(\tilde{\mathbf{p}}(\mathbf{\Theta}^m; |\mathbf{h}|)\right)\right\|_2 &\leq \left\|\tilde{\mathbf{p}}\left(\mathbf{\Theta}^{m+1}; |\mathbf{h}|\right) - \tilde{\mathbf{p}}\left(\mathbf{\Theta}^m; |\mathbf{h}|\right)\right\|_F \\ &\leq \frac{1}{4}\sqrt{LNK}\|\mathbf{h}\|_F \frac{\prod_{l=1}^{L}\bar{\lambda}_l}{\min_{l\in[L]}\bar{\lambda}_l}\left\|\mathbf{\Theta}^{m+1} - \mathbf{\Theta}^m\right\|_2.\end{aligned}$$

From the assumption in Claim 3 that all the weights are bounded, for $l \in [L]$, $\bar{\lambda}_l$ is bounded. So there exists constant $C_4^{'}$ and C_4 such that

$$\left\|g\left(\mathbf{\Theta}_t^m\right) - g\left(\mathbf{\Theta}^m\right)\right\|_2 \leq \frac{C_4^{'}\sqrt{LNK}}{4}\|\mathbf{h}\|_F \frac{\prod_{l=1}^{L}\bar{\lambda}_l}{\min_{l\in[L]}\bar{\lambda}_l}\left\|\mathbf{\Theta}_t^m - \mathbf{\Theta}^m\right\|_2 = C_4\left\|\mathbf{\Theta}_t^m - \mathbf{\Theta}^m\right\|_2. \tag{53}$$

Now we have shown that

$$\left\|\nabla f_{\text{UL}}\left(\mathbf{\Theta}_t^m\right) - \nabla f_{\text{UL}}\left(\mathbf{\Theta}^m\right)\right\|_2 \leq C_1C_4\left\|\mathbf{\Theta}_t^m - \mathbf{\Theta}_m\right\|_2 + C_2C_3\left\|\mathbf{\Theta}_t^m - \mathbf{\Theta}^m\right\|_2 = \left(C_1C_4 + C_2C_3\right)\left\|\mathbf{\Theta}_t^m - \mathbf{\Theta}^m\right\|_2$$

Step 2: The above already shows the Lipschitz gradient of f_{UL} under certain assumptions. Now it applies the condition in Lemma 3. By Lemma 3, let $\eta < \frac{1}{C_1C_4 + C_2C_3}$. There is

$$f_{\mathrm{UL}}\left(\mathbf{\Theta}^{m+1}\right) \leq f_{\mathrm{UL}}\left(\mathbf{\Theta}^{m}\right)+\left\langle \nabla f_{\mathrm{UL}}\left(\mathbf{\Theta}^{m}\right), \mathbf{\Theta}^{m+1}-\mathbf{\Theta}^{m}\right\rangle+\frac{1}{2}\left(C_1C_4+C_2C_3\right)\left\|\mathbf{\Theta}^{m+1}-\mathbf{\Theta}^{m}\right\|_2^2$$
$$\leq f_{\mathrm{UL}}\left(\mathbf{\Theta}^{m}\right)-\frac{1}{2}\eta\left\|\nabla f_{\mathrm{UL}}\left(\mathbf{\Theta}^{m}\right)\right\|_2^2.$$

Sum up from $m=1,2,\cdots,M$ and divide it by M so we can get

$$f_{\mathrm{UL}}\left(\mathbf{\Theta}^{M}\right)-f_{\mathrm{UL}}\left(\mathbf{\Theta}^{0}\right) \leq -\frac{\eta}{2M}\sum_{m=1}^{M}\left\|\nabla f_{\mathrm{UL}}\left(\mathbf{\Theta}^{m}\right)\right\|_2^2.$$

Thus, for UL loss (4), $\mathbf{\Theta}$ will converge to a stationary point. □

D. Proof of Claim 4

Lemma 4. *Let Assumption 1 hold. Then, for unconstrained version of SL problem* (3)*, the gradient of loss f_{sup} over parameters in layer 2 satisfies:*

$$\mathrm{vec}\left(\nabla_{W_2} f_{\mathrm{sup}}\right)=\left(\mathbb{I}_{n_2}\otimes F_1^T\right)\prod_{q=3}^{L}\Sigma_{q-1}\left(W_q\otimes\mathbb{I}_N\right)\frac{\partial f_{\mathrm{sup}}}{\partial \mathrm{vec}(\mathbf{p})} \tag{54}$$
$$:= A(\mathbf{\Theta})\cdot\frac{\partial f_{\mathrm{sup}}}{\partial\,\mathrm{vec}(\mathbf{p})} \tag{55}$$

where we have defined the matrix $A(\mathbf{\Theta}):=\mathrm{vec}(\nabla_{W_2} f_{\mathrm{sup}})=\left(\mathbb{I}_{n_2}\otimes F_1^T\right)\prod_{q=3}^{L}\Sigma_{q-1}\left(W_q\otimes\mathbb{I}_N\right)$, which is a function of the network weights $\mathbf{\Theta}$. Denote each column of A as $a_i, i\in[Nn_L]$. Given a dataset $(\mathbf{h},\mathbf{p})$, suppose features of the first two samples are the same, i.e $\mathbf{h}^{(1)}=\mathbf{h}^{(2)}$, then we have $a_{(k-1)N+1}=a_{(k-1)N+2}, k\in[n_L]$.

Claim 4. *Suppose $(\mathbf{h},\mathbf{p})$ and $(\mathbf{h}^{'},\mathbf{p}^{'})$ are two sets of data, and they are constructed below:*

- *Each dataset consists of N samples;*
- *The features of two data samples are identical: $\mathbf{h}^{'}=\mathbf{h}$;*
- *In the first dataset, for any $n\in[N]$, the labels $\mathbf{p}^{(n)}$ is the unique globally optimal power allocation for problem* (1)*, given channel realization $\mathbf{h}^{(n)}$; Further, two samples in $\mathbf{h}$ are identical, say, $\mathbf{h}^{(1)}=\mathbf{h}^{(2)}$, and all the other samples are linearly independent.*
- *For the second dataset, the labels are constructed as follows:*

$$\mathbf{p}^{',(2)}\neq\mathbf{p}^{(2)},\quad \mathbf{p}^{',(n)}=\mathbf{p}^{(n)},\forall\ n,\neq 2. \tag{56}$$

Further, since $\mathbf{h}^{(1)}=\mathbf{h}^{(2)}$ and $\mathbf{h}=\mathbf{h}'$, we also have $\mathbf{h}^{'(1)}=\mathbf{h}^{'(2)}$.

Suppose that Assumption 1 and Assumption 2 hold true, and use the same training algorithm as Claim 3-(a) to optimize the unconstrained version of (3) *using $(\mathbf{h},\mathbf{p})$ and $(\mathbf{h}^{'},\mathbf{p}^{'})$ respectively. Let $\mathbf{\Theta}^m$ and $\mathbf{\Theta}^{'m}$ denote the sequences of weights generated by the algorithm for the two data sets respectively. Suppose that the initial solutions of the two algorithms are the same: $\mathbf{\Theta}^{'0}=\mathbf{\Theta}^0$. Define*

$$A(\mathbf{\Theta}):=\left(\mathbb{I}_{n_2}\otimes F_1^T\right)\prod_{q=3}^{L}\Sigma_{q-1}\left(W_q\otimes\mathbb{I}_N\right),\quad A_0:=A(\mathbf{\Theta}^0). \tag{57}$$

Suppose all the eigenvalues of $A_0^TA_0$ are within the interval $[0,1]$. Then if we choose the stepsize η small enough, there exist $\beta>0$ and $\beta^{'}>0$ such that the following holds true

$$f_{\mathrm{sup}}\left(\mathbf{\Theta}^1\right)\leq\beta f_{\mathrm{sup}}\left(\mathbf{\Theta}^0\right),\ \ f_{\mathrm{sup}}\left(\mathbf{\Theta}^{'1}\right)\leq\beta^{'}f_{\mathrm{sup}}\left(\mathbf{\Theta}^{'0}\right).$$

Further, we have $\beta<\beta^{'}$, that is, the objective function with the correct label decreases faster.

Proof. The idea of the proof is following: 1) We first argue that $f_{\mathrm{sup}}(\mathbf{\Theta})$ satisfies the Lipschitz condition in Lemma 3; 2) use Lemma 3 and the spectral decomposition to derive an upper bound of $f_{\mathrm{sup}}(\mathbf{\Theta}^1)$; 3) analyze the difference in the upper bound and training speed using two different training samples.

Step 1: First, we claim that $f_{\mathrm{sup}}(\mathbf{\Theta})$ satisfies the condition in Lemma 3, which means f_{sup} has Lipschitz gradient. The proof is almost the same as Step 1 in the proof of Claim 3, so we do not include the proof here. It can be concluded that, there exists constant Q_0, such that

$$\|\nabla f_{\mathrm{sup}}(\mathbf{\Theta}^1)-\nabla f_{\mathrm{sup}}(\mathbf{\Theta}^0)\|_2\leq Q_0\cdot\|\mathbf{\Theta}^1-\mathbf{\Theta}^0\|_2. \tag{58}$$

Step 2: With the condition for Lemma 3 being satisfied, then we can apply Lemma 3 and derive the upper bound of $f_{\sup}(\mathbf{\Theta}^1)$. Recall that $f_l^m = \mathrm{vec}(F_l^m)$ and $y = \mathrm{vec}(\mathbf{p})$. There exists stepsize $\eta < \frac{1}{Q_0}$ such that

$$\begin{aligned} f_{\sup}\left(\mathbf{\Theta}^1\right) &\overset{(i)}{\leq} f_{\sup}\left(\mathbf{\Theta}^0\right) + \left\langle \nabla f_{\sup}\left(\mathbf{\Theta}^0\right), \mathbf{\Theta}^1 - \mathbf{\Theta}^0 \right\rangle + \frac{Q_0}{2}\left\|\mathbf{\Theta}^1 - \mathbf{\Theta}^0\right\|_2^2 \\ &= f_{\sup}\left(\mathbf{\Theta}^0\right) - \eta\left\|\nabla f_{\sup}\left(\mathbf{\Theta}^0\right)\right\|_2^2 + \frac{Q_0}{2}\eta^2\left\|\nabla f_{\sup}\left(\mathbf{\Theta}^0\right)\right\|_2^2 \\ &\overset{(ii)}{\leq} f_{\sup}\left(\mathbf{\Theta}^0\right) - \frac{1}{2}\eta\left\|\nabla f_{\sup}\left(\mathbf{\Theta}^0\right)\right\|_2^2 \\ &\overset{(iii)}{\leq} f_{\sup}\left(\mathbf{\Theta}^0\right) - \frac{1}{2}\eta\left\|\mathrm{vec}\left(\nabla_{W_2} f_{\sup}\left(\mathbf{\Theta}^0\right)\right)\right\|_2^2 \\ &\overset{(iv)}{=} f_{\sup}\left(\mathbf{\Theta}^0\right) - \frac{1}{2}\eta\left(f_L^0 - y\right)^\top A_0^\top A_0\left(f_L^0 - y\right) \\ &\overset{(v)}{=} \left(f_L^0 - y\right)^\top\left(I - \frac{1}{2}\eta A_0^\top A_0\right)\left(f_L^0 - y\right) \\ &\overset{(vi)}{:=} \left(f_L^0 - y\right)^\top S_0^T S_0\left(f_L^0 - y\right) \end{aligned} \tag{59}$$

where in (vi) we have defined:

$$S_0 := \left(I - \frac{1}{2}\eta A_0^\top A_0\right)^{\frac{1}{2}}$$

(i) applies Lemma 3; (ii) is because stepsize $\eta < \frac{1}{Q_0}$; (iii) comes from the property of l_2 norm; (iv) uses the definition of A_0 in (57); (v) comes from the expression of $f_{\sup}$ using vectorized variables.

Now we utilize the spectral decomposition method to decompose the upper bound. The idea is to express the vector f_L^0 as a linear combination of the eigenvectors of S_0. The eigendecomposition of $A_0^T A_0$ is given by

$$A_0^T A_0 = P\Lambda P^T \tag{60}$$

where $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \cdots, \lambda_N n_L)$ and $\lambda_i, i \in [Nn_L]$ are the eigenvalues, and $P = [v_1, \cdots, v_{Nn_L}]$ is the eigenvector matrix. Then it is easy to see that the following holds:

$$S_0 = \left(I - \frac{1}{2}\eta A_0^\top A_0\right)^{\frac{1}{2}} = \left(PP^T - \frac{1}{2}\eta P\Lambda P^T\right)^{\frac{1}{2}} = \sum_{i=1}^{Nn_L}(1 - \frac{1}{2}\eta\lambda_i)^{\frac{1}{2}} v_i v_i^T. \tag{61}$$

Now we can express the vector $f_L^0 - y$ as a linear combination of the $v_i, i \in [Nn_L]$. We have

$$f_L^0 - y = \sum_{i=1}^{Nn_L}\left(v_i^T\left(f_L^0 - y\right)\right) v_i.$$

So now we can rewrite the upper bound derived in (59) as following:

$$\begin{aligned} f_{\sup}\left(\mathbf{\Theta}^1\right) &\leq \left(f_L^0 - y\right)^\top\left(I - \frac{1}{2}\eta A_0^\top A_0\right)\left(f_L^0 - y\right) \\ &= \frac{1}{2}\left[S_0(f_L^0 - y)\right]^T\left[S_0(f_L^0 - y)\right] \\ &= \frac{1}{2}\left[\sum_{i=1}^{Nn_L}\left(1 - \frac{1}{2}\eta\lambda_i\right)^{\frac{1}{2}} v_i v_i^T \sum_{i=1}^{Nn_L}\left(v_i^\top\left(f_L^0 - y\right)\right) v_i\right]^T\left[\sum_{i=1}^{Nn_L}\left(1 - \frac{1}{2}\eta\lambda_i\right)^{\frac{1}{2}} v_i v_i^\top \sum_{i=1}^{Nn_L}\left(v_i^\top\left(f_L^0 - y\right)\right) v_i\right] \\ &= \frac{1}{2}\left[\sum_{i=1}^{Nn_L}\left(1 - \frac{1}{2}\eta\lambda_i\right)^{\frac{1}{2}}\left(v_i^\top\left(f_L^0 - y\right)\right) v_i\right]^T\left[\sum_{i=1}^{Nn_L}\left(1 - \frac{1}{2}\eta\lambda_i\right)^{\frac{1}{2}}\left(v_i^\top\left(f_L^0 - y\right)\right) v_i\right] \\ &= \frac{1}{2}\sum_{i=1}^{Nn_L}\left(v_i^\top\left(f_L^0 - y\right)\right)^2(1 - \frac{1}{2}\eta\lambda_i). \end{aligned} \tag{62}$$

Step 3: With the decomposed upper bound of $f_{\sup}(\mathbf{\Theta})$, we will next find its special structures under our constructed data.

Recall that we have $\mathbf{h} = \mathbf{h}^{'}$ and within each dataset $\mathbf{h}^{(1)} = \mathbf{h}^{(2)}, \mathbf{h}^{',(1)} = \mathbf{h}^{',(2)}$. Recall that $\{a_i\}_{i\in[n_L]}$ represents all the columns in A_0. From Lemma 4, we know that if two input data points are identical, then we have $a_{(k-1)N+1} = a_{(k-1)N+2}, k \in [n_L]$. Thus, we have $\mathrm{rank}(A_0) \leq (N-1)n_L$. Since other samples are linear independent and parameters are generated randomly, it follows that $\mathrm{rank}(A_0) = Nn_L - n_L$. Notice that $\mathrm{rank}(A_0^T A_0) = \mathrm{rank}(A_0)$, so there are n_L eigenvalues equal to 0. Without loss of generality, we assume $\lambda_i = 0, i \in [n_L]$. And for $i = n_L + 1, n_L + 2, \cdots, Nn_L$, we have $\lambda_i > 0$. Now let us find the

eigenvectors corresponding to these zero eigenvalues. Denote $e_i \in \mathbb{R}^{Nn_L}$ as unit vector such that the i-th component is 1, and the others are 0 Assume $v_i = \frac{\sqrt{2}}{2} e_{(i-1)N+1} - \frac{\sqrt{2}}{2} e_{(i-1)N+2}$ for $i \in [n_L]$, so v_i is the eigenvector corresponds to λ_i.

Next, we use the above construction to show that with dataset $(\mathbf{h}, \mathbf{p})$, the convergence at first iteration can be faster than using dataset $(\mathbf{h}^{'}, \mathbf{p}^{'})$. Notice in the two datasets, the labels for the second sample are not the same, $\mathbf{p}^{',(2)} \neq \mathbf{p}^{(2)}$. That is, there exists at least one index i such that $y_{(i-1)N+1} = y_{(i-1)N+2}$ but $y^{'}_{(i-1)N+1} = y^{'}_{(i-1)N+2}$, where $y = \mathrm{vec}(\mathbf{p})$ and $y' = \mathrm{vec}(\mathbf{p}')$. This is because we concatenate each column of $\mathbf{p}$ and $\mathbf{p}^{'}$ to get y and $y^{'}$. Without loss of generality, assuming that $i = 1$ only. Since we have two identical samples $\mathbf{h}^{(1)} = \mathbf{h}^{(2)}$, the outputs of these two samples are identical. In f_L^0, we must have $f^0_{L,(i-1)N+1} = f^0_{L,(i-1)N+2}$ for $i \in [n_L]$. Recall that we have $v_i = \frac{\sqrt{2}}{2} e_{(i-1)N+1} - \frac{\sqrt{2}}{2} e_{(i-1)N+2}$. Then for the first dataset, the following holds:

$$\left(v_i^\top \left(f_L^0 - y\right)\right)^2 = \left(\frac{\sqrt{2}}{2}(f^0_{L,(i-1)N+1} - y_1) - \frac{\sqrt{2}}{2}(f^0_{L,(i-1)N+2} - y_2)\right)^2 = 0, \quad i \in [n_L]. \tag{63}$$

However, for the second dataset, we have:

$$\begin{aligned} \epsilon := \left(v_1^\top \left(f_L^0 - y^{'}\right)\right)^2 &= \left(\frac{\sqrt{2}}{2}(f^0_{L,1} - y^{'}_1) - \frac{\sqrt{2}}{2}(f^0_{L,2} - y^{'}_2)\right)^2 > 0, \\ \left(v_i^\top \left(f_L^0 - y^{'}\right)\right)^2 &= \left(\frac{\sqrt{2}}{2}(f^0_{L,(i-1)N+1} - y^{'}_1) - \frac{\sqrt{2}}{2}(f^0_{L,(i-1)N+2} - y^{'}_2)\right)^2 = 0, \quad i = 2, 3, \cdots, Nn_L. \end{aligned} \tag{64}$$

Now let us denote the SL loss using labels $\mathbf{p}$ as $f_{\sup}(\mathbf{\Theta}; y)$ and using labels $\mathbf{p}^{'}$ as $f_{\sup}(\mathbf{\Theta}; y')$. At initialization, let us define $\epsilon_1 := f_{\sup}\left(\mathbf{\Theta}^0; y\right)$, $\epsilon_2 := f_{\sup}\left(\mathbf{\Theta}^0; y^{'}\right)$, then after the first iteration, we have the following series of relations:

$$\begin{aligned} f_{\sup}\left(\mathbf{\Theta}^1; y\right) &\overset{(i)}{\leq} \sum_{i=1}^{Nn_L} \left(v_i^\top \left(f_L^0 - y\right)\right)^2 (1 - \frac{1}{2}\eta\lambda_i) \\ &\overset{(ii)}{=} \sum_{i=n_L+1}^{Nn_L} \left(v_i^\top \left(f_L^0 - y\right)\right)^2 (1 - \frac{1}{2}\eta\lambda_i) \\ &\overset{(iii)}{\leq} (1 - \frac{1}{2}\eta \min_{i \geq Nn_L+1} \lambda_i) \cdot \epsilon_1 \\ &\overset{(iv)}{:=} \beta f_{\sup}(\mathbf{\Theta}^0; y) \end{aligned} \tag{65}$$

where (i) is from (62); (ii) is because of (63); (iii) uses the property of orthogonal matrix; in (iv) we have defined $\beta := 1 - \frac{1}{2}\eta \min_{i \geq Nn_L} \lambda_i < 1$.

Similarly, the following series of relations hold for the second dataset:

$$\begin{aligned} f_{\sup}\left(\mathbf{\Theta}^{',1}; y'\right) &\overset{(i)}{\leq} \sum_{i=1}^{Nn_L} \left(v_i^\top \left(f_L^0 - y'\right)\right)^2 (1 - \frac{1}{2}\eta\lambda_i) \\ &\overset{(ii)}{=} \epsilon + \sum_{i=Nn_L+1}^{Nn_L} \left(v_i^\top \left(f_L^0 - y'\right)\right)^2 (1 - \frac{1}{2}\eta\lambda_i) \\ &\overset{(iii)}{\leq} \epsilon + (1 - \frac{1}{2}\eta \min_{i \geq Nn_L+1} \lambda_i)(\epsilon_2 - \epsilon) \\ &\overset{(iv)}{=} (1 - \frac{1}{2}\eta \min_{i \geq Nn_L+1} \lambda_i) \left(1 - \frac{\epsilon}{\epsilon_2} + \frac{\epsilon}{\epsilon_2(1 - \frac{1}{2}\eta \min_{i \geq Nn_L+1} \lambda_i)}\right) \epsilon_2 \\ &\overset{(v)}{:=} \beta^{'} f_{\sup}(\mathbf{\Theta}^0) \end{aligned} \tag{66}$$

where (i) is from (62); (ii) uses (64); (iii) uses the property of orthogonal matrix; (iv) is a simple algebraic transformation; in (v) we have defined $\beta^{'} := (1 - \min_{i \geq Nn_L+1} \lambda_i) \left(1 - \frac{\epsilon}{\epsilon_2} + \frac{\epsilon}{\epsilon_2(1 - \frac{1}{2}\eta \min_{i \geq Nn_L+1} \lambda_i)}\right)$. Since $1 - \frac{\epsilon}{\epsilon_2} + \frac{\epsilon}{\epsilon_2(1 - \frac{1}{2}\eta \min_{i \geq Nn_L+1} \lambda_i)} > 1$, there is $\beta < \beta^{'}$. The claim is proved. □