An Unsupervised Deep Unfolding Framework for Robust Symbol Level Precoding

Abdullahi Mohammad, *Student Member, IEEE,* Christos Masouros, *Senior Member, IEEE,* and Yiannis Andreopoulos, *Senior Member, IEEE*

***Abstract*—Symbol Level Precoding (SLP) has attracted significant research interest due to its ability to exploit interference for energy-efficient transmission. This paper proposes an unsupervised deep-neural network (DNN) based SLP framework. Instead of naively training a DNN architecture for SLP without considering the specifics of the optimization objective of the SLP domain, our proposal unfolds a power minimization SLP formulation based on the interior point method (IPM) proximal *'log'* barrier function. Furthermore, we extend our proposal to a robust precoding design under channel state information (CSI) uncertainty. The results show that our proposed learning framework provides near-optimal performance while reducing the computational cost from $\mathcal{O}(n^{7.5})$ to $\mathcal{O}(n^3)$ for the symmetrical system case where n = number of transmit antennas = number of users. This significant complexity reduction is also reflected in a proportional decrease in the proposed approach's execution time compared to the SLP optimization-based solution.**

***Index Terms*—Symbol level precoding, Constructive Interference, downlink beamforming, power minimization, Deep Neural Networks.**

I. Introduction

INTERFERENCE has been known to yield a decrease in the throughput and communication reliability of a downlink multi-user multiple-inputs single-output (MU-MISO) wireless system. Traditionally, interference is regarded as the limiting factor against the ever-increasing needs for transmission rates and quality of service (QoS) in fifth-generation (5G) wireless communication systems and beyond [1]–[3]. However, recent studies on interference exploitation have transformed the traditional paradigm in which known inferences are effectively managed [1]–[5]. Consequently, transmit beamforming techniques for the downlink channels for power minimization problems under specific QoS become imperative for high-throughput systems under interference.

All authors are with the Department of Electronic and Electrical Engineering, University College London, London WC1E 7JEK, UK. (e-mail: abdullahi.mohammad.16@ucl.ac.uk; c.masouros@ucl.ac.uk; i.andreopoulos@ucl.ac.uk)

This work was supported by EPSRC under grants EP/S028455/1 and EP/R035342/1, and in part by the Petroleum Technology Development Fund (PTDF) Overseas Scholarship Scheme, Nigeria under the award PTDF/ED/PHD/MA/1003/16. This article was presented in part at the IEEE Global Communications Conference (GLOBECOM) 2021, Madrid, Spain.

The idea of exploiting interference was first introduced by Masouros and Alsusa [6], where instantaneous interference was classified into constructive and destructive. Initial suboptimal approaches to exploit constructive interference (CI) were first introduced by Masouros *et al.* [7] [8]. The first form of optimization-based CI precoding was introduced in the context of vector perturbation precoding through a quadratic optimization approach [9]. A convex optimization-based CI scheme termed symbol-level-precoding technique was proposed first with strict phase constraints on the received constellation point [10], and with a robust relaxed-angle formulation [2]. We refer to recent work [9]–[13] for more details on the optimization-based CI precoding techniques.

As a result of the performance gains over conventional block-level-precoding (BLP) schemes, the idea of CI has been applied in many domains, such as vector perturbation [14], wireless information and power transfer [15], mutual coupling exploitation [16], multiuser MISO downlink channel [17], directional modulation [18], relay and cognitive radio [1], [19]. Despite the superior performance offered by CI-based precoding methods, their increased computational complexity can hinder their practical application when performed on a symbol-by-symbol basis. To address this, Li and Masouros [20] proposed an iterative closed-form precoding design with optimal performance for CI exploitation in the MISO downlink by driving the optimal precoder's mathematical Lagrangian expression and Karush–Kuhn–Tucker conditions for optimization with both strict and relaxed phase rotations.

Lately, there has been growing interests in using deep neural networks (DNN) for wireless physical layer design [21]–[23]. More relevant to this work are the machine learning for classical beamforming optimization schemes for MU-MISO downlink transmission [22], [24]–[28]. The benefit of using DNNs is that the computational burden of the learning algorithm can be controlled via online training, and a variety of loss functions can be used for each optimization objective. One of the earliest attempts at using DNNs models for beamforming design was the work of Alkhateeb *et al.* [22], where a learning-based coordinated beamforming technique was proposed for link reliability and frequent poor hand-off between base stations (BSs) in millimetre-wave (mmWave) communications. Kerret and Gesbert [26] introduced DNNs precoding scheme to address the *"Team Decision problems"* for decentralized decision making in multiple-input-multiple-

output (MIMO) settings. Huang *et al.* [24] proposed a fast beamforming design based on unsupervised learning that yielded performance close to that of the weighted minimum mean-squared error (WMMSE) algorithm. A DNN-based precoding strategy that utilized a heuristic solution structure of the downlink beamforming was proposed by Huang *et al.* [27]. Furthermore, Xia *et al.* [28] developed deep convolutional neural networks (CNNs) framework for downlink beamforming optimization. The framework exploits expert knowledge based on the known structure of optimal iterative solutions for sum-rate maximization, power minimization, and Signal-to-Interference-plus-Noise (SINR) balancing problems.

A typical approach for solving constrained optimization with DNN for wireless physical layer design is via function approximation [28], where the authors assume perfect CSI condition. It involves solving the problem, first using iterative algorithms or convex optimization techniques and finally approximating the optimal solution with a DNN architecture [24], [25], [27]–[29]. Accordingly, the major drawback of these proposals is that the efficacy of supervised learning is bounded by the assumptions and accuracy of the optimal solutions obtained from the structural optimization algorithm.

Recently, learning-based methods have been extended to beamforming designs via CI [29]–[33]. In [29], a CI-NN model-driven precoding design with a customized loss function based on the CI optimization problem that needed no accurate CI solutions as training labels was developed. Nonetheless, the loss function layer design added unnecessary complexity to the learning framework. A deep auto-encoder for a robust SLP scheme and symbol detection with low computational complexity was proposed in [30]. However, the complex structure of the symbol detection decision rule at the receiver made its practical implementation challenging. An unsupervised learning approach for SLP that maximized the minimum QoS of every user in an MU-MISO using a straightforward decision rule was introduced in [32]. While the learning architecture used simple decision conventions to formulate the training loss function based on the simple convex power constraints, its applicability to other SLP problems, such as power minimization and maximum secrecy problems involving much more complex constraints, was not investigated. Contrary to [32], where the SINR balancing problem was solved, we focus on the power minimization problem in this correspondence. Correspondingly, [31] presented a deep unfolding method that used NN with a full-precision floating point representation based on the power minimization problem under perfect CSI and strict phase angle rotation. On the contrary, [33] applied weight quantization to the same problem under an imperfect SCI condition and relaxed phase angle rotation.

Moreover, the approach to deep unfolding for constrained problems depends on the constraints' nature. Again, the traditional deep unfolding approach may not apply to a problem with several simultaneous, non-differentiable constraints, especially those in wireless communications. While the deep unfolding that uses a proximal operator with a *'log'* barrier has been applied in image processing for image reconstruction [34], to the best of our knowledge, its applicability is rare in wireless communications. Inspired by these findings, in this work, we propose a more robust unsupervised learning approach that unrolls SLP power minimization problems involving non-smooth SINR constraints into a deep learning framework via interior point method (IPM) proximal log-barrier formulation.

This work focuses on an unsupervised learning-based approach for precoding design by exploiting known interference in MU-MISO systems for the power minimization problem under SINR constraints. The learning framework is designed by unfolding an IPM iterative algorithm via *'log'* barrier function. The proposed learning-based precoding scheme does not require generating the training dataset from conventional optimization solutions, thereby saving considerable computational effort and time. Our contributions are summarized below:

- We introduce an unsupervised DNN-based power minimization SLP scheme for MU-MISO downlink transmission. The proposed framework is designed by unfolding an IPM algorithm via a *'log'* barrier function that exploits the convexity associated with the SLP inequality constraints. The learning framework utilizes the domain knowledge to derive the Lagrange function of the original SLP optimization as a loss function. This is used to train the network in an unsupervised mode to learn a set of Lagrangian multipliers that directly minimize the objective function to satisfy the constraints. A regularization parameter is added to the Lagrange function to aid the training convergence, and we provide detailed formulations leading to the unfolded unsupervised learning architecture for constrained optimization problems.
- We extend the formulation to design a robust learning-based precoder where the uncertainty in channel estimation is considered.
- We derive analytic expressions for the computational complexity of various SLP and the proposed unsupervised learning precoding schemes. Our analysis demonstrates that the proposed deep unfolding (DU) framework offers a theoretical, computational complexity reduction from $\mathcal{O}(n^{7.5})$ to $\mathcal{O}(n^3)$ for the symmetrical system case where n = number of transmit antennas = number of users. This is reflected in a commensurate decrease in the execution time as compared to the SLP optimization-based method.

The remainder of the paper is organized as follows: The system model and the methods for traditional precoding and SLP optimization-based for downlink MU-MISO system are presented in Section II. The proposed unsupervised DU-based precoding designs under perfect channel conditions for power minimization are introduced in Section III, and extension to a robust precoding design under uncertainty channel condition is described in

Section IV. Section V presents a detailed analytic computational complexity evaluation of the proposed precoding schemes. Simulations and results are presented in Section VI. Finally, Section VII summarizes and concludes the paper.

Notations: We use bold uppercase symbols for matrices, bold lowercase symbols for vectors and lowercase symbols for scalars. The ℓ_2-norm and ℓ_1-norm are denoted by $\|\cdot\|_2$ and $\|\cdot\|_1$, respectively. The $|\cdot|$ represents the absolute value and $\boldsymbol{\theta}_i$ is the i-th trainable parameter associated with DNN layers. The Hermitian of the vector and the transpose of the real matrix are represented by $\mathbf{x}^H$ and $\mathbf{X}^T$, respectively. Operators $\mathrm{Re}(\cdot)$ and $\mathrm{Im}(\cdot)$ represent real and imaginary parts of a complex vector, respectively. Finally, notations $\boldsymbol{\mathcal{L}}(\cdot)$ and $\boldsymbol{\mathcal{H}}(\cdot)$ are reserved for the loss and parameter update functions, respectively.

II. System Model and Problem Description

A. Conventional Block Level Precoding for Power Minimization

Consider a single-cell downlink channel with N_t transmit antennas at the BS transmitting to K single-antenna users. Assume a quasi-static flat-fading channel between the BS and the users, denoted by $\mathbf{h}_i \in \mathbb{C}^{N_t \times 1}$. The received signal at user i is given by

$$\begin{aligned} y_i &= \mathbf{h}_i^H \sum_{k=1}^{K} \mathbf{w}_k s_k + v_i \\ &= \mathbf{h}_i^H \sum_{k=1}^{K} \mathbf{w}_k e^{j(\varphi_k - \varphi_i)} s_i + v_i \end{aligned} \tag{1}$$

where $\sum_{k=1}^{K} \mathbf{w}_k s_k = \sum_{k=1}^{K} \mathbf{w}_k e^{j(\varphi_k-\varphi_i)} s_i$ is the transmit signal and $s_i = s e^{j\varphi_i}$ is assumed to be a referenced phase-shift keying (PSK) modulated symbol with constant amplitude s. Also, $\mathbf{h}_i$, $\mathbf{w}_i$, s_i, v_i and φ_i represent the channel vector, precoding vector, data symbol, received noise and phase rotation for the i-th user. Conventionally, the power minimization problem seeks to minimize the average transmit power by treating all interference as detrimental subject to QoS constraints as defined below [35]

$$\begin{aligned} \min_{\{\mathbf{w_i}\}} \quad & \sum_{i=1}^{K} \|\mathbf{w}_i\|_2^2 \\ \text{s.t.} \quad & \frac{|\mathbf{h}_i^H \mathbf{w}_i|^2}{\sum_{k=1, k\neq i} |\mathbf{h}_i^H \mathbf{w}_k|^2 + v_0} \geq \Gamma_i \ , \ \forall i. \end{aligned} \tag{2}$$

where Γ_i is the SINR threshold of the i-th user. It has been proven that problem (2) is suboptimal from an instantaneous point of view, as it does not take into account the fact that interference can constructively enhance the received signal power [8].

B. Power Minimization via Symbol-Level Precoding

With the aim of utilizing the instantaneous interference in a multi-user downlink channel scenario, the interference can be categorized into constructive and destructive based on the known standards described [36]. Based on this, CI is defined as the interference that nudges the received symbols off the modulated-symbol constellation's decision thresholds [2]. In this situation, the instantaneous interference can contribute constructively to detecting the desired signal. With the knowledge of both the data symbols and the CSI at the BS, SINR constraints in (2) can be modified to include CI for generic multilevel or M-arry phase shift keying (M-PSK) modulated signals. The maximum phase shift in the CI region is given by $\phi = \pm\frac{\pi}{\mathbb{M}}$, where $\mathbb{M}$ is the modulation order.

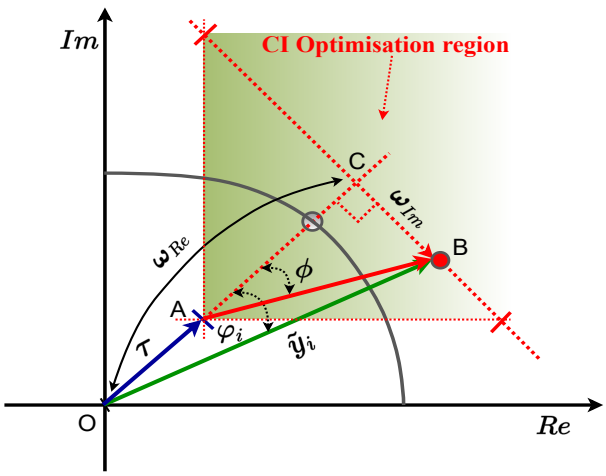


Fig. 1: Generic geometrical optimization regions for interference exploitation for Precoding design in $\mathbb{M}$-PSK [2].

Fig. 1 shows the generic geometrical representation of the CI, where the phase-rotated received signal is expressed as $\tilde{y}_i \triangleq \mathbf{h}_i^H \sum_{k=1}^{K} \mathbf{w}_k e^{j(\varphi_k-\varphi_i)}$. From this expression, the real and imaginary parts are respectively given by: $\boldsymbol{\omega}_{Re} = \mathrm{Re}(\tilde{y}_i)$ and $\boldsymbol{\omega}_{Im} = \mathrm{Im}(\tilde{y}_i)$. In Fig. 1, we show an indicative example corresponding to the constellation point $1+j$ in the quadrature phase shift keying (QPSK) constellation, where the green shaded area represents the constructive region of the constellation. It is also worth noting that for the instantaneous interference to be harnessed constructively, the received signal must be within the green area depending on the minimum distance (τ) from the decision boundaries. This enables the interfering signals to align constructively with the symbol of interest to improve the received signal's strength. Therefore, with the aid of geometry, the following vectors can be expressed as

$$\vec{AC} = [\omega_{Re} - \vec{OA}], \ \vec{BC} = j \cdot \omega_{Im}, \tag{3}$$

$\vec{OA}$ is the detection threshold (τ) and is obtained from the relation: $\vec{OA} = \sqrt{\Gamma_i v_0^2}$. For a point B to be located within the constructive region, the following condition must hold

$$\tan\varphi_i = \frac{\vec{BC}}{\vec{AC}} \leq \tan\phi \Rightarrow \frac{|j \cdot \mathrm{x}_{Im}|}{\left|[\mathrm{x}_{Re} - \vec{OA}]\right|} \leq \tan\phi, \tag{4}$$

which finally gives

$$\omega_{Im} \leq (\omega_{Re} - \tau)\tan\phi. \tag{5}$$

The CI for each user will be guaranteed by adopting (5) as the SINR constraint.

Therefore, for an $\mathbb{M}$-PSK, the power minimization problem in (2) can be reformulated based on the constructive/destructive interference classification criteria [37] and transformed into its equivalent CI-based SLP optimization form below [2]

$$\begin{aligned} \min_{\{\mathbf{w_i}\}} \quad & \left\| \sum_{k=1}^{K} \mathbf{w}_k e^{j(\varphi_k-\varphi_1)} \right\|_2^2 \\ \text{s.t.} \quad & \left| \text{Im}\left(\mathbf{h}_i^H \sum_{k=1}^{K} \mathbf{w}_k e^{j(\varphi_k-\varphi_i)} \right) \right| \leq \\ & \left(\text{Re}\left(\mathbf{h}_i^H \sum_{k=1}^{K} \mathbf{w}_k e^{j(\varphi_k-\varphi_i)} \right) - \sqrt{\Gamma_i v_0} \right) \tan\phi \ , \ \forall i. \end{aligned} \tag{6}$$

where $\omega_{Im} = Im\Big(\mathbf{h}_i^H \sum_{k=1}^{K} \mathbf{w}_k e^{j(\varphi_k-\varphi_i)}\Big)$ and $\omega_{Re} = Re\Big(\mathbf{h}_i^H \sum_{k=1}^{K} \mathbf{w}_k e^{j(\varphi_k-\varphi_i)}\Big)$. It can be oserve that (6) is data-dependent; therefore, the optimisation is done on a symbol-by-symbol basis.

It is important to note that problem (6) is a standard second-order cone program (SOCP), and thus can be optimally solved using numerical software optimization packages, such as CVX and CVXPY. Similarly, problem (2) is optimal from a stochastic perspective; hence the same optimization software can be used to solve it. It is essential to note that problem (2) is suboptimal from an instantaneous stance because the precoder disregards the fact that, instantaneously, interference can add constructively to the received signal power.

III. Learning-Based Power minimization for SLP

This section presents the formulation of a learning-based CI power minimization problem for SLP. Throughout this section, we assume a perfect CSI known at the BS.

Motivated by the recent adoption of an IPM for image restoration [38], we propose an unsupervised learning framework that unfolds a constrained optimization problem into a sequence of learning layers/iterations for a multi-user MISO beamforming.We first convert (6) into a standard IPM formulation containing a slack variable, where necessary. The measure of the fidelity of the solution to (6) is determined by learning a set of penalty parameters in the form of Lagrange multipliers associated with the constraints. From (6), we define the following

$$\hat{\mathbf{h}}_i = \mathbf{h}_i e^{j(\varphi_1-\varphi_i)} \tag{7}$$

$$\mathbf{w} = \sum_{k=1}^{K} \mathbf{w}_k e^{j(\varphi_k-\varphi_1)}. \tag{8}$$

Accordingly, to ease the analysis, we partition the complex rotations into the real and imaginary parts as follows

$$\hat{\mathbf{h}}_i = \hat{\mathbf{h}}_{Ri} + j\hat{\mathbf{h}}_{Ii} \tag{9a}$$

$$\mathbf{w} = \mathbf{w}_R + j\mathbf{w}_I \tag{9b}$$

where $\hat{\mathbf{h}}_{Ri} = \text{Re}(\hat{\mathbf{h}}_i)$, $\hat{\mathbf{h}}_{Ii} = \text{Im}(\hat{\mathbf{h}}_i)$, $\mathbf{w}_R = \text{Re}(\mathbf{w})$ and $\mathbf{w}_I = \text{Im}(\mathbf{w})$. The product of complex rotations of (9a) and (9b) can be written as

$$\hat{\mathbf{h}}_i\mathbf{w} = (\hat{\mathbf{h}}_{Ri} + j\hat{\mathbf{h}}_{Ii})(\mathbf{w}_R + j\mathbf{w}_I). \tag{10}$$

The real and imaginary parts of (10) can be written in vector forms as follows

$$\text{Re}(\hat{\mathbf{h}}_i\mathbf{w}) = \begin{bmatrix} \hat{\mathbf{h}}_{Ri} & \hat{\mathbf{h}}_{Ii} \end{bmatrix} \begin{bmatrix} \mathbf{w}_R \\ -\mathbf{w}_I \end{bmatrix} \tag{11a}$$

$$\text{Im}(\hat{\mathbf{h}}_i\mathbf{w}) = \begin{bmatrix} \hat{\mathbf{h}}_{Ri} & \hat{\mathbf{h}}_{Ii} \end{bmatrix} \begin{bmatrix} \mathbf{w}_I \\ \mathbf{w}_R \end{bmatrix} \tag{11b}$$

Let $\mathbf{\Lambda} = [\hat{\mathbf{h}}_{Ri} \quad \hat{\mathbf{h}}_{Ii}]^T$, $\mathbf{w}_1 = [\mathbf{w}_R \quad -\mathbf{w}_I]^T$ and $\mathbf{w}_2 = [\mathbf{w}_I \quad \mathbf{w}_R]^T$

$$\text{Re}(\hat{\mathbf{h}}_i^T\mathbf{w}) = \mathbf{\Lambda}_i^T\mathbf{w}_1 \text{ and } \text{Im}(\hat{\mathbf{h}}_i^T\mathbf{w}) = \mathbf{\Lambda}_i^T\mathbf{\Pi}\mathbf{w}_1 \tag{12}$$

where

$$\mathbf{w}_2 = \mathbf{\Pi}\mathbf{w}_1 \text{ and } \mathbf{\Pi} = \begin{bmatrix} \mathbf{O}_{N_t} & -\mathbf{I}_{N_t} \\ \mathbf{I}_{N_t} & \mathbf{O}_{N_t} \end{bmatrix}; \in \mathbb{R}^{2N_t \times 2N_t}. \tag{13}$$

Note that $\mathbf{I}_{N_t}$ is the identity matrix and $\mathbf{O}_{N_t}$ the matrix of zeros, respectively. Using the above definitions, problem (6) can be recast into its mutlicast formulation [2]

$$\begin{aligned} \min_{\{\mathbf{w}_1\}} \quad & \|\mathbf{w}_1\|_2^2 \\ \text{s.t.} \quad & \left|\mathbf{\Lambda}_i^T\mathbf{\Pi}\mathbf{w}_1\right| \leq \left(\mathbf{\Lambda}_i^T\mathbf{w}_1 - \sqrt{\Gamma_i v_0}\right) \tan\phi \ , \ \forall i \end{aligned} \tag{14}$$

A. Interior Point Method

Consider a general form of a nonlinear constrained optimization of the form [39]

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^{\mathbf{N}}} \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & g(\mathbf{x}) \geq 0 \\ & C(\mathbf{x}) = 0 \end{aligned} \tag{15}$$

The rationale of adopting IPM is to substitute the initial constrained optimization problem by a chain of unconstrained sub-problems of the form

$$\min_{\mathbf{x}\in\mathbb{R}^{\mathbf{N}}} f(\mathbf{x}) + \lambda C(\mathbf{x}) + \mu B(\mathbf{x}). \tag{16}$$

where $B(\cdot) \triangleq -\sum \ln(\cdot)$ is the logarithmic barrier function associated with inequality constraint with unbounded derivative at the boundary of the feasible domain, $C(\cdot)$ is a function associated with equality constraint, μ and λ are the Lagrangian multipliers for inequality and equality constraints, respectively. For K users, we define a vector $\boldsymbol{\mu} \triangleq [\mu_1, \cdots, \mu_K]$.

Following the above line of argument, the unconstrained sequence of (14) per user can be written as

$$\min_{\mathbf{w}_1\in\mathbb{R}^{2\mathbf{N_t}\times 1}} f(\mathbf{w}_1) + \mu B(\mathbf{w}_1), \tag{17}$$

It can be observed that the constraint in (14) cannot be efficiently handled because it is non-differentiable at some points ($\mathbf{w}_1 \to 0$). Intuitively, the difficulty of solving (14) stems from the constraint not being differentiable at 0; otherwise, we could follow the gradient of (17). In a

situation like this, the proximal operators (alternatively called "proximity operators") come to the rescue [34], [40]. Instead of directly minimizing the objective in (17), we can search its equivalent proximal operator function. The proximal operator is frequently used in proximal gradient methods in optimization algorithms associated with non-differentiable optimization problems [41]. In order to minimize the objective and address the non-smooth term while ensuring the feasibility of the iterates, the concept of a logarithmic barrier is combined with the proximity operator, which allows us to incorporate the proximal gradient approach into unrolled learning framework. For every inequality constraint, $\gamma \in \{0, +\infty\}$ and $\mathbf{w}_1 \in \mathbb{R}^{2N_t \times 1}$, we define the proximity function as in [39] with respect to (17), which we shall later use to compute the projected gradient descent as

$$\text{prox}_{\gamma\mu B}(\mathbf{w}_1) = \underset{\mathbf{w}_1 \in \mathbb{R}^{2\mathbf{N_t}\times 1}}{\text{argmin}} \quad \frac{1}{2}\left\|\mathbf{w}_0 - \mathbf{w}_1\right\|_2^2 + \gamma\mu B(\mathbf{w}_1), \tag{18}$$

where γ is the step-size for computing the gradients and $\mathbf{w}_0$ is the initial value of the precoding vector. To convert (11) into its equivalent barrier function problem, we integrate the inequality constraint into the objective by translating it into a barrier term as follows [42]

$$\begin{aligned} \min_{\mathbf{w}_1} \quad & f(\mathbf{w}_1) - \mu \sum_{i=1}^{p} \ln\left(g_i(\mathbf{w}_1)\right) \\ \text{s.t.} \quad & C(\mathbf{w}_1) = 0 \end{aligned} \tag{19}$$

where $g(\mathbf{w}_1) = \left(\mathbf{\Lambda}_i^T \mathbf{w}_1 - \sqrt{\Gamma_i v_0}\right) \tan\phi - \left|\mathbf{\Lambda}_i^T \mathbf{\Pi} \mathbf{w}_1\right|$ and p is the number of inequality constraints.

Going back to our initial SPL optimization to apply this framework, first we rewrite the constraint of (14) as

$$a \leq \mathbf{\Lambda}_i^T \mathbf{\Pi} \mathbf{w}_1 \leq b, \tag{20}$$

where

$$a = -\left(\mathbf{\Lambda}_i^T \mathbf{\Pi} \mathbf{w}_1 - \sqrt{\Gamma_i v_0}\right) \tan\phi, \tag{21a}$$

$$b = \left(\mathbf{\Lambda}_i^T \mathbf{\Pi} \mathbf{w}_1 - \sqrt{\Gamma_i v_0}\right) \tan\phi. \tag{21b}$$

Therefore, the original problem (14) becomes

$$\begin{aligned} \min_{\{\mathbf{w}_1\}} \quad & \|\mathbf{w}_1\|_2^2 \\ \text{s.t.} \quad & a \leq \mathbf{\Lambda}_i^T \mathbf{\Pi} \mathbf{w}_1 \leq b \ , \ \forall i. \end{aligned} \tag{22}$$

It is apparent that the constraint of (22) is contained within a hyperslab [43].

1) Hyperslab Constraints: Given the constraint in (22), the precoding vector $\mathbf{w}_1$ is contained within a set of hyperslab $\mathcal{C}$ and also bounded by $\{a,\ b\}$. Therefore, $\mathcal{C}$ is defined as follows

$$\mathcal{C} = \{\mathbf{w}_1 \in \mathbb{R}^{2N_t \times 1}\}|_{a \leq \mathbf{\Lambda}^T \mathbf{\Pi} \mathbf{w}_1 \leq b}. \tag{23}$$

For all $\gamma > 0$ and $\mu > 0$, a proximity barrier function related to (23) is given by

$$B(\mathbf{w}_1) = \begin{cases} -\ln(b - \hat{w}_1) - \ln(a + \hat{w}_1), & \text{if } a \leq \hat{w}_1 \leq b \\ +\infty, & \text{otherwise} \end{cases} \tag{24}$$

where for convenience, we let $\hat{w}_1 = \mathbf{\Lambda}_i^T \mathbf{\Pi} \mathbf{w}_1$.

B. Proximity Operator for the SLP Formulation

To unfold (22) into learning framework using IPM, we use its equivalent proximity *'log'* barrier function (24) and the proximal operator of $\gamma\mu B(\mathbf{w}_1)$ for every $\mathbf{w}_1$ defined as

$$\Phi(\mathbf{w}_1, \gamma, \mu) = \text{prox}_{\gamma\boldsymbol{\mu} B}(\mathbf{w}_1) = \mathbf{w}_1 + \frac{X(\mathbf{w}_1, \gamma, \mu) - \hat{\mathbf{\Lambda}}_i \mathbf{w}_1}{\|\hat{\mathbf{\Lambda}}_i\|_2^2} \hat{\mathbf{\Lambda}}_i \tag{25}$$

where $\hat{\mathbf{\Lambda}}_i = \mathbf{\Lambda}_i^T \mathbf{\Pi}$ and X is a typical solution of the following cubic equation of the form

$$\begin{aligned} x^3 - \left(b + a + \hat{\mathbf{\Lambda}}\mathbf{w}_1\right) x^2 + \\ \left(ba + \hat{\mathbf{\Lambda}}\mathbf{w}_1 (b + a) - 2\gamma\mu \|\hat{\mathbf{\Lambda}}\|_2^2\right) x \\ + \left(-ba\hat{\mathbf{\Lambda}}\mathbf{w}_1 + \gamma\mu\,(b + a)\,\|\hat{\mathbf{\Lambda}}\|_2^2\right) = 0. \end{aligned} \tag{26}$$

It is important to note that the solution to (26) is obtained using the analytic solution of the cubic equation. To build the structure of the learning framework, we need to obtain the Jacobian of $\Phi(\mathbf{w}_1, \gamma, \mu)$ with respect to $\mathbf{w}_1$ and the its derivatives with respect to γ and μ as follows

$$\begin{aligned} \mathcal{J}_\Phi\,|_{(\mathbf{w}_1)} = \mathbf{I}_{2N_t} + \frac{1}{\left\|\hat{\mathbf{\Lambda}}_i\right\|_2^2} \times \\ \left(\frac{(b - X(\mathbf{w}_1, \gamma, \mu))\,(a - X(\mathbf{w}_1, \gamma, \mu))}{\mathbf{\Upsilon}(\mathbf{w}_1, \gamma, \mu)} - 1\right) \hat{\mathbf{\Lambda}}_{\boldsymbol{i}} \hat{\mathbf{\Lambda}}_{\boldsymbol{i}}^{\boldsymbol{T}} \end{aligned} \tag{27}$$

$$\Delta_\Phi\,|_{(\mu)} = \frac{-\gamma\,(b + a - 2X(\mathbf{w}_1, \gamma, \mu))}{\mathbf{\Upsilon}(\mathbf{w}_1, \gamma, \mu)} \hat{\mathbf{\Lambda}}_{\boldsymbol{i}} \tag{28}$$

$$\Delta_\Phi\,|_{(\gamma)} = \frac{-\mu\,(b + a - 2X(\mathbf{w}_1, \gamma, \mu))}{\mathbf{\Upsilon}(\mathbf{w}_1, \gamma, \mu)} \hat{\mathbf{\Lambda}}_{\boldsymbol{i}}, \tag{29}$$

where $\mathbf{I}_{2N_t} \in \mathbb{R}^{2N_t \times 2N_t}$. For hyperslab constraints, $\mathbf{\Upsilon}(\cdot)$ is the derivative of (26) with respect to x. Finally, using similar abstraction as in subsection III-A, the SLP formulation can be expressed as a succession of sub-problems with respect to the inequality constraint

$$\min_{\mathbf{w}_1 \in \mathbb{R}^{2\mathbf{N_t}\times 1}} \quad \|\mathbf{w}_1\|_2^2 + \lambda \mathbf{w}_1 + \mu B(\mathbf{w}_1). \tag{30}$$

It is important to note that the original problem (14) does not have an equality constraint. However, the term $\lambda\mathbf{w}_1$ introduced in (30) is to provide additional stability to the network. Using the proximity operator of the barrier, the update rule for every iteration is given by

$$\mathbf{w}_1^{[r+1]} = \text{prox}_{\gamma^{[r]}\boldsymbol{\mu}^{[r]} B}\left(\mathbf{w}_1^{[r]} - \gamma^{[r]} \Delta\left(D(\mathbf{w}_1^{[r]}, \lambda^{[r]})\right)\right) \tag{31}$$

where

$$D(\mathbf{w}_1^{[r]}, \lambda^{[r]}) = \|\mathbf{w}_1\|_2^2 + \lambda \mathbf{w}_1, \tag{32}$$

and $\Delta\left(D(\mathbf{w}_1^{[r]}, \lambda^{[r]})\right) = \frac{\partial D(\mathbf{w}_1^{[r]}, \lambda^{[r]})}{\partial \mathbf{w}_1^{[r]}}$.

C. Deep SLP Network (SLP-DNet)

To build the proposed learning-based SLP architecture, we combine an IPM with a proximal forward-backward procedure [44] and transform it into an NN structure represented by the proximity barrier term (see Fig. 2). The learning architecture strictly follows the formulation (31). We show a striking similarity between our proposal and the feed-forward. Intuitively, we form cascade layers of NN from (31) as follows

$$\mathbf{w}_1^{[r+1]} = \text{prox}_{\gamma^{[r]}\mu^{[r]}B}\left[\left(\mathbf{I}_{2N_t} - 2\gamma^{[r]}\right)\mathbf{w}_1^{[r]} + \gamma^{[r]}\lambda^{[r]}\mathbf{1}^T\right]. \tag{33}$$

where $\mathbf{1} \in \mathbb{R}^{1\times 2N_t}$ is a vector of ones. By letting $\mathbf{W}_r = \mathbf{I}_{2N_t} - 2\gamma^{[r]}$, $\mathbf{b}_r = \gamma^{[r]}\lambda^{[r]}\mathbf{1}^T$ and $\mathbf{\Theta}_r = \text{prox}_{\gamma^{[r]}\mu^{[r]}B}$, the r-layer network $\mathscr{L}^{[r-1]}\cdots\mathscr{L}^{[0]}$ will correspond to the following

$$\mathbf{\Theta}_0\left(\mathbf{W}_0 + \mathbf{b}_0\right), \cdots, \mathbf{\Theta}_r\left(\mathbf{W}_r + \mathbf{b}_r\right) \tag{34}$$

where $\mathbf{W}_r$ and $\mathbf{b}_r$ are described as weight and bias parameters respectively. The nonlinear activation functions are defined by $\mathbf{\Theta}_r$.

In the SLP-DNet design, the Lagrange multiplier associated with the equality constraint is wired across the network to provide additional flexibility. It is important to note that the architectures are the same for both nonrobust and robust power minimization problems described in Sections III and IV but differ in proximity barrier functions (PBFs). Therefore, to simplify our exposition, we build the structure of the learning framework based on (31) and the feed-forward-backward proximal IPM (see Algorithm 1). As shown in Fig. 2, SLP-DNet has two main units; the parameter update module (PUM) and the auxiliary processing block (APB). The PUM has three core components associated with Lagrangian multipliers (equality and inequality constraints) and the training step-size, which are updated according to the following

$$\mathscr{H}(\mathbf{w}_1, \mu, \gamma, \lambda) = \text{prox}_{\gamma^{[r]}\mu^{[r]}B}\left(\mathbf{w}_1^{[r]} - \gamma^{[r]}\Delta D(\mathbf{w}_1^{[r]}, \lambda^{[r]})\right). \tag{35}$$

Furthermore, the component that forms the barrier term is constructed with one convolutional layer, an average pooling layer, a fully connected layer, and a Softplus layer to curb the output to a positive real value to satisfy the inequality constraint. The APB unit is connected to the last r-th block of the PUM in the form of a deep CNN to convert the output of the last parameter update block into a target transmit precoding vector. The APB architecture comprises 3 convolution layers and 2 activation layers. In addition, a Batch Normalization layer is added between each convolutional layer and the activation layer to stabilize the mismatch in the distribution of the inputs caused by the internal covariant shift [45].

D. Parameter Update Module

The unfolded learning architecture comprises three main parameters forming each learning block, where μ, λ and γ are learned as shown in Fig. 2. We first initialize these parameters randomly such that $\mu > 0$, $\lambda > 0$ and $\gamma > 0$. The gradient of the proximal barrier function operator is calculated with respect to each parameter during training, and its value is updated iteratively in each block simultaneously using stochastic gradient descent (SGD) based on the following update rules

$$\mu^{[r+1]} = \mu^{[r]} - \eta\frac{\partial\Phi(\mathbf{w}_1^{[r]},\ \gamma^{[r]},\ \mu^{[r]})}{\partial\mu^{[r]}}, \tag{36}$$

$$\gamma^{[r+1]} = \gamma^{[r]} - \eta\frac{\partial\Phi(\mathbf{w}_1^{[r]},\ \gamma^{[r]},\ \mu^{[r]})}{\partial\gamma^{[r]}}, \tag{37}$$

$$\lambda^{[r+1]} = \lambda^{[r]} - \eta\frac{\partial D(\mathbf{w}_1^{[r]},\ \lambda^{[r]})}{\partial\lambda^{[r]}}. \tag{38}$$

Where η is the learning rate,Φ proximal operator, and it is defined in Appendix A.

For every r block (r-th layer), there are three core components; $\mathscr{L}_\mu^{[r]}$, $\mathscr{L}_\gamma^{[r]}$ and $\mathscr{L}_\lambda^{[r]}$ associated with the learnable parameters (μ, γ and λ), respectively. These components form a learning block for computing the barrier parameter (μ) associated with the inequality constraint, the step-size (γ) and the equality constraint (λ), if exists. To ensure that the constraints remain positive, a Softplus-sign function, $\text{Softplus}(z) = \ln(1+\exp(z))$ is used. The nonrobust SLP-DNet formulation and its training steps are summarized in Algorithm 1. The training variables are

Algorithm 1 Proximity Barrier Operator of a Nonrobust SLP-DNet

Input: $\mathbf{h}_{Ri}$, $\mathbf{h}_{Ii}$, Γ_i and n_0 (noise)
Output: $\mathbf{w}_1$, γ, μ and λ
Initialisation :
1: Randomly initialise $\mu^{[0]} > 0$, $\lambda^{[0]} > 0$, $\gamma^{[0]} > 0\ \forall\ i = 1, \cdots, K$ and $\mathbf{w}_0 \in \mathbb{R}^{2N_t\times 1}$ using (41)
2: Find the solution to (26) using Cardano formula.
3: For every solution in step 2, compute its corresponding Barrier function using (24).
4: Compute the Proximity Operator of the Barrier at $\mathbf{w}_0$ using (18), where $\Phi(\mathbf{w}_1, \gamma, \mu) = \text{prox}_{\gamma\mu B}(\mathbf{w}_1)$.
5: Compute the derivatives of the Proximity Operator w.r.t $\mathbf{w}_1$, μ and γ using (27), (28) and (29).
6: **for** $r = 0$ to L **do**
7: Update the training variables as follows:
(a) $\mu^{[r+1]} = \mu^{[r]} - \eta\frac{\partial\Phi(\mathbf{w}_1^{[r]},\ \gamma^{[r]},\ \mu^{[r]})}{\partial\mu^{[r]}}$
(b) $\gamma^{[r+1]} = \gamma^{[r]} - \eta\frac{\partial\Phi(\mathbf{w}_1^{[r]},\ \gamma^{[r]},\ \mu^{[r]})}{\partial\gamma^{[r]}}$
(c) $\lambda^{[r+1]} = \lambda^{[r]} - \eta\frac{\partial D(\mathbf{w}_1^{[r]},\ \lambda^{[r]})}{\partial\lambda^{[r]}}$ using (32)
where η is the learning rate.
Feed-forward-Backward Proximal IPM
8: $\mathbf{w}_1^{[r+1]} = \text{prox}_{\gamma^{[r]}\mu^{[r]}B}\left(\mathbf{w}_1^{[r]} - \gamma^{[r]}\Delta D(\mathbf{w}_1^{[r]},\ \lambda^{[r]})\right)$ or using (33).
9: **end for**
10: **return** $\mathbf{w}_1^*$ (Optimal precoding tensor).
11: To obtain the original optimal complex precoding vector $\mathbf{w}^*$, we use the relation $\mathbf{w}_1^* = [\mathbf{w}_R^* \ -\mathbf{w}_I^*]$ to separate it into real and imaginary parts.

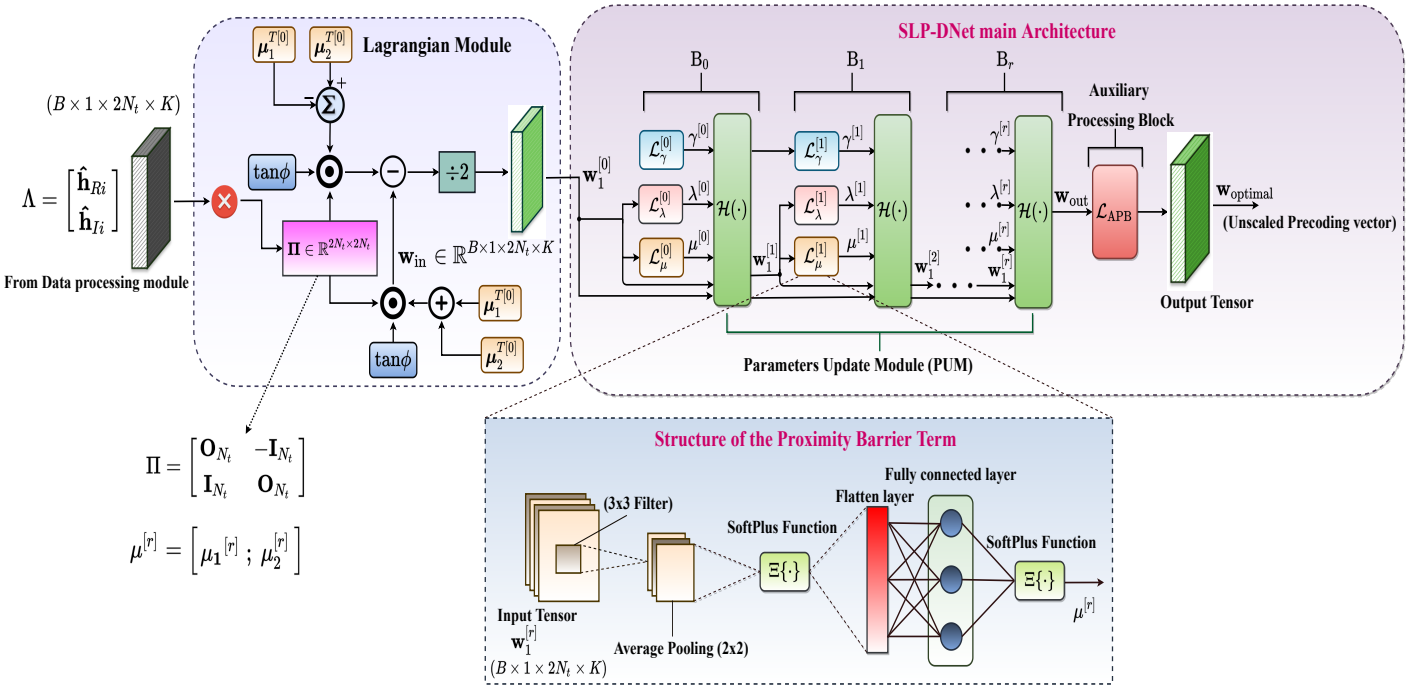


Fig. 2: Complete SLP-DNet Architecture, showing the parameter update module, the auxiliary processing block.

updated iteratively in each unfolding layer simultaneously using GDS. For every training step, a corresponding value of the precoding vector is updated based on (33). The output precoding vector from the last PUM is then fed into the APB to obtain the final optimal precoding vector. The same algorithm is also adopted for a robust SLP-DNet but with a different PBF based on a robust power minimization problem. Finally, the output from the APB is the precoding vector in the real domain. The relation: $\mathbf{w}_1 = [\mathbf{w}_R \ -\mathbf{w}_I]^T$ is used to convert it to its equivalent complex domain for every SINR value of the i-th user.

1) Duality and Loss Function of the SLP Formulation: In order to ease the formulation of the dual-problem of the original problem (14), the left-hand-side of the inequality constraint is split into its equivalent positive and negative parts as follows

$$\begin{aligned} \min_{\{\mathbf{w}_1\}} \quad & \|\mathbf{w}_1\|_2^2 \\ \text{s.t.} \quad & \boldsymbol{\Lambda}_i^T \boldsymbol{\Pi} \mathbf{w}_1 \leq \left(\boldsymbol{\Lambda}_i^T \boldsymbol{\Pi} \mathbf{w}_1 - \sqrt{\Gamma_i v_0}\right) \tan\phi, \ \forall i \\ & -\boldsymbol{\Lambda}_i^T \boldsymbol{\Pi} \mathbf{w}_1 \leq \left(\boldsymbol{\Lambda}_i^T \boldsymbol{\Pi} \mathbf{w}_1 - \sqrt{\Gamma_i v_0}\right) \tan\phi, \ \forall i. \end{aligned} \tag{39}$$

The Lagrangian of (39) is defined as

$$\begin{aligned} \mathcal{L}_{\text{rl}}(\mathbf{w}_1, \ \boldsymbol{\mu}_1, \ \boldsymbol{\mu}_2) &= \|\mathbf{w}_1\|_2^2 \\ &+ \boldsymbol{\mu}_1 \left(\boldsymbol{\Lambda}_i^T \boldsymbol{\Pi} \mathbf{w}_1 - \boldsymbol{\Lambda}_i^T \mathbf{w}_1 \tan\phi + \sqrt{\Gamma_i v_0}\right) \\ &- \boldsymbol{\mu}_2 \left(\boldsymbol{\Lambda}_i^T \boldsymbol{\Pi} \mathbf{w}_1 + \boldsymbol{\Lambda}_i^T \mathbf{w}_1 \tan\phi - \sqrt{\Gamma_i v_0}\right), \end{aligned} \tag{40}$$

where $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ are the Lagrangian multipliers associated with the constraints and are related to the proximity barrier. The subscript *'rl'* stands for relaxed phase rotation. It can be easily proven that the lower bound (**LB**) of (40) is $\boldsymbol{\mathcal{L}}_{\text{rl}}(\mathbf{w}_1, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2) \geq \boldsymbol{\mu}_1 \Lambda_i \left(\Pi - \tan\phi\right) - \boldsymbol{\mu}_2 \Lambda_i \left(\Pi + \tan\phi\right)$.

From (40), the optimal precoder is obtained by differentiating $\mathcal{L}_{\text{rl}}(\cdot)$ w.r.t $\mathbf{w}_1$ and equating to zero. By doing so, the optimal precoder can be found as

$$\mathbf{w}_1 = \frac{\left(\boldsymbol{\mu}_1^T + \boldsymbol{\mu}_2^T\right) \cdot \boldsymbol{\Lambda}_i \tan\phi - \left(\boldsymbol{\mu}_1^T - \boldsymbol{\mu}_2^T\right) \cdot \boldsymbol{\Pi}^T \boldsymbol{\Lambda}_i \tan\phi}{2}. \tag{41}$$

In the sequel, we show that (41) is used to generate the training input (precoding vector) by randomly initializing the Lagrangian multipliers ($\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$) and then train the network to learn their values that minimize the loss function (Lagrangian function). The loss function is modified by adding $\boldsymbol{\ell}_2$-norm regularization over the weights to calibrate the learning coefficients in order to adjust the learning process. It should be noted that the regularization here is not aimed at addressing the problem of overfitting as in the case of supervised learning. However, regularization in an unsupervised learning is used to normalize and moderate weights attached to a neuron to help stabilize the learning algorithm [46]. The loss function (40) over N training samples is thus expressed as

$$\begin{aligned} \mathcal{L}_{\text{rl}}(\mathbf{w}_1, \ \boldsymbol{\mu}_1, \ \boldsymbol{\mu}_2) &= \frac{1}{N} \sum_{i=1}^{N} \|\mathbf{w}_1\|_2^2 \\ &+ \frac{1}{N} \sum_{i=1}^{N} \left(\boldsymbol{\mu}_1 \left(\boldsymbol{\Lambda}_i^T \boldsymbol{\Pi} \mathbf{w}_1 - \boldsymbol{\Lambda}_i^T \mathbf{w}_1 \tan\phi + \sqrt{\Gamma_i v_0}\right)\right) \\ &- \frac{1}{N} \sum_{i=1}^{N} \left(\boldsymbol{\mu}_2 \left(\boldsymbol{\Lambda}_i^T \boldsymbol{\Pi} \mathbf{w}_1 + \boldsymbol{\Lambda}_i^T \mathbf{w}_1 \tan\phi - \sqrt{\Gamma_i v_0}\right)\right) \\ &+ \frac{\vartheta}{NL} \sum_{i=1}^{N} \sum_{i=1}^{L} \|\boldsymbol{\theta}_i\|_2^2, \end{aligned} \tag{42}$$

where $\boldsymbol{\theta}_i$ are the trainable parameters of the i-th layers associated with the weights and biases, and $\vartheta > 0$ is the penalty parameter that controls the bias and variance of the trainable coefficients, N, L is the number of training samples (batch size or the number of channel realization) and the number of layers, respectively.

E. Learning-Based SLP for Strict Angle Rotation

In the previous subsection, we presented SLP-DNet based on relaxed angle formulation. In this subsection, we provide a formulation for strict phase angle rotation where all the interfering signals align exactly to the phase the signal of interest (i.e. $\phi = 0$ in Fig. 1), the optimization problem is [2]

$$\begin{aligned} \min_{\{\mathbf{w}_1\}} \quad & \|\mathbf{w}_1\|_2^2 \\ \text{s.t.} \quad & \boldsymbol{\Lambda}_i^T \boldsymbol{\Pi} \mathbf{w}_1 = 0 \ , \ \forall i \\ & \boldsymbol{\Lambda}_i^T \mathbf{w}_1 \geq \sqrt{\Gamma_i v_0} \ , \ \forall i. \end{aligned} \tag{43}$$

We observe that the inequality constraint in (43) is affine. Based on this, the proximal barrier function for the strict phase rotation is

$$B_{\text{st}}(\mathbf{w}_1) = \begin{cases} -\ln\left(\boldsymbol{\Lambda}_i^T \mathbf{w}_1 - \sqrt{\Gamma_i v_0}\right), & \text{if } \boldsymbol{\Lambda}_i^T \mathbf{w}_1 \geq \sqrt{\Gamma_i v_0} \\ +\infty, & \text{otherwise.} \end{cases} \tag{44}$$

The subscript *'st'* represents strict phase rotation. Therefore, for every precoding vector $\mathbf{w}_1 \in \mathbb{R}^{2N_t \times 1}$, the proximity operator of $\mu\gamma B_{\text{st}}$ at $\mathbf{w}_1$ is given by

$$\Phi_{\text{st}}(\mathbf{w}_1, \mu, \gamma) = \mathbf{w}_1 + \frac{\boldsymbol{\Lambda}_i^T \mathbf{w}_1 - \sqrt{\Gamma_i v_0} - \sqrt{(\boldsymbol{\Lambda}_i^T \mathbf{w}_1 - \sqrt{\Gamma_i v_0})^2 + 4\gamma\mu \left\|\boldsymbol{\Lambda}_i^T\right\|_2^2}}{2\left\|\boldsymbol{\Lambda}_i\right\|_2^2} \boldsymbol{\Lambda}_i. \tag{45}$$

Similar to the steps in subsection III-B, the learning-based framework for SLP strict phase rotation is designed by finding the Jacobian matrix of $\Phi(\mathbf{w}_1, \mu, \gamma)$ with respect to $\mathbf{w}_1$, and the derivatives of $\Phi(\mathbf{w}_1, \mu, \gamma)$ with respect to γ and μ can be easily obtained from (45). The loss function over N training batches is given by

$$\mathcal{L}_{\text{st}}(\mathbf{w}_1, \boldsymbol{\lambda}, \ \boldsymbol{\mu}) = \frac{1}{N} \sum_{i=1}^{N} \left(\|\mathbf{w}_1\|_2^2 + \boldsymbol{\lambda} \boldsymbol{\Lambda}_i^T \boldsymbol{\Pi} \mathbf{v}_1 \right) + \frac{1}{N} \sum_{i=1}^{N} \left(\boldsymbol{\mu} \left(\sqrt{\Gamma_i v_0} - \boldsymbol{\Lambda}_i^T \mathbf{w}_1 \right) \right) + \frac{\vartheta}{NL} \sum_{i=1}^{N} \sum_{i=1}^{L} \|\boldsymbol{\theta}_i\|_2^2, \tag{46}$$

where $\boldsymbol{\mu}$ and $\boldsymbol{\lambda}$ are the Lagrangian multipliers for inequality and equality constraints, respectively. Finally, minimizing (46) with respect to $\mathbf{w}_1$ (differentiating $\mathcal{L}_{\text{st}}(\cdot)$ w.r.t $\mathbf{w}_1$), gives the optimal precoder as

$$\mathbf{w}_1 = \frac{\boldsymbol{\mu}^T \cdot \boldsymbol{\Lambda}_i - \boldsymbol{\lambda}^T \cdot \boldsymbol{\Pi} \boldsymbol{\Lambda}_i}{2}. \tag{47}$$

IV. Learning-Based Robust Power Minimization SLP with Channel Uncertainty

A. Channel Uncertainty and Channel error Model

So far, we have derived the unsupervised learning scheme in which the uncertainty in estimating the channel coefficients is not considered. The exact CSI is often unobtainable in practice. To model the user's actual channel in the uncertainty region, we consider an ellipsoid ξ such that

$$\hat{\mathbf{h}}_i = \bar{\mathbf{h}}_i + \bar{\boldsymbol{e}}_i \ \forall k, \tag{48}$$

where $\bar{\mathbf{h}}_i$ is the known CSI estimates at the BS, and $\bar{\boldsymbol{e}}_i$ denotes the channel error within the uncertainty region of the ellipsoid (i.e. $\hat{\mathbf{h}}_i \in \xi$). The model of the uncertainty ellipsoid with the center $\bar{\mathbf{h}}_i$ is expressed as

$$\xi = \left\{ \bar{\mathbf{h}}_i + \bar{\mathbf{e}}_i |_{\|\bar{\mathbf{e}}_i \leq 1\|} \right\}. \tag{49}$$

As shown in [2], the channel error is given by $\left\{ \bar{\mathbf{e}}_i : \|\bar{\mathbf{e}}_i\|_2^2 \leq \varsigma_i^2 \right\}$. It is important to note that the BS is assumed to have the knowledge about the channel error, excluding its corresponding error bound ς_i^2. For details and formulation of the conventional robust BLP, we refer the reader to [47].

B. Robust Optimization-Based SLP Formulation

The multi-cast constructive interference formulation of the power minimization problem for the worst-case CSI error is given by [47]

$$\begin{aligned} \min_{\{\mathbf{w}\}} \quad & \|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & \left| \text{Im}\left(\hat{\mathbf{h}}_i^T \mathbf{w}\right) \right| - \left(\text{Re}\left(\hat{\mathbf{h}}_i^T \mathbf{w}\right) - \sqrt{\Gamma_i v_0} \right) \tan\phi \leq 0, \\ & \forall \left\|\bar{\mathbf{e}}_i\right\|^2 \leq \varsigma_i^2, \ \forall i. \end{aligned} \tag{50}$$

The intractability of the constraint in (50) can be effectively handled using convex optimization methods. Therefore, to guarantee that the robust constraint in (50) is satisfied, it is modified as follows [2]

$$\max_{\|\bar{e}_i\|^2 \leq \varsigma_i^2} \left(\left| \text{Im}\left(\hat{\mathbf{h}}^T \mathbf{w}\right) \right| - \left(\text{Re}\left(\hat{\mathbf{h}}^T \mathbf{w}\right) - \sqrt{\Gamma v_0} \right) \tan\phi \right) \leq 0. \tag{51}$$

It is worth noting that the subscripts in (51) are ignored in order to simplify the problem formulation. By defining the equivalent real-valued channel and channel error vectors, the real and imaginary parts in the constraint can be decomposed into two separate constraints as explained in Section III (see (11a) and (11b)). Thus the robust formulation of the constraint is equivalent to two separate real-valued constraints as follows

$$\boldsymbol{\Lambda}^T \mathbf{w}_1 - \boldsymbol{\Lambda}^T \mathbf{w}_2 \tan\phi + \varsigma \left\|\mathbf{w}_1 - \mathbf{w}_2 \tan\phi\right\|_2 + \sqrt{\Gamma v_0} \tan\phi \leq 0, \tag{52}$$

$$-\boldsymbol{\Lambda}^T \mathbf{w}_1 - \boldsymbol{\Lambda}^T \mathbf{w}_2 \tan\phi + \varsigma \left\|\mathbf{w}_1 + \mathbf{w}_2 \tan\phi\right\|_2 + \sqrt{\Gamma v_0} \tan\phi \leq 0, \tag{53}$$

where $\boldsymbol{\Lambda} = \begin{bmatrix} \bar{\mathbf{h}}_R & \bar{\mathbf{h}}_I \end{bmatrix}^T$, $\mathbf{e} \triangleq \begin{bmatrix} \bar{\mathbf{e}}_R & \bar{\mathbf{e}}_I \end{bmatrix}^T$ and $\hat{\mathbf{h}} = \bar{\mathbf{h}}_R + j\bar{\mathbf{h}}_I + \bar{\mathbf{e}}_R + j\bar{\mathbf{e}}_I$. Finally, the robust CI formulation for the power minimization problem becomes

$$\begin{aligned} \min_{\{\mathbf{w}_1,\mathbf{w}_2\}} \quad & \|\mathbf{w}_1\|_2^2 \\ \text{s.t.} \quad & \text{Constraints (52) and (53), } \forall i \\ & \text{where } \ \mathbf{w}_1 = \mathbf{\Pi}\mathbf{w}_2. \end{aligned} \tag{54}$$

C. Proposed Unsupervised Learning-Based Robust SLP

As an extension of the previous formulations in subsection III-B, the focus here is to derive a PBF for the robust learning-based precoding scheme. Substituting for $\mathbf{w}_1$ in (54), we have

$$\left(\mathbf{\Lambda}^T\mathbf{\Pi} - \mathbf{\Lambda}^T\tan\phi\right)\mathbf{w}_2 + \varsigma\left\|\left(\mathbf{\Pi} - \tan\phi\right)\mathbf{w}_2\right\|_2 + \sqrt{\Gamma v_0}\tan\phi \leq 0, \tag{55}$$

$$-\left(\mathbf{\Lambda}^T\mathbf{\Pi} + \mathbf{\Lambda}^T\tan\phi\right)\mathbf{w}_2 + \varsigma\left\|\left(\mathbf{\Pi} + \tan\phi\right)\mathbf{w}_2\right\|_2 + \sqrt{\Gamma v_0}\tan\phi \leq 0. \tag{56}$$

Apparently, the constraints (55) and (56) are bounded by the ℓ_2-norm. Therefore, problem (54) is rewritten as

$$\begin{aligned} \min_{\{\mathbf{w}_2\}} \quad & \|\mathbf{w}_2\|_2^2 \\ \text{s.t.} \quad & \text{Constraints (55) and (56), } \forall i. \end{aligned} \tag{57}$$

The resulting barrier function of the corresponding constraints of (57) is the sum of the individual barrier functions associated with the two inequality constraints. We begin by introducing the feasible set of solutions bounded by the Euclidean ball.

1) Bounded Euclidean ball Constraint: Suppose a problem whose set of feasible solutions is bounded by the Euclidean ball [40]

$$\mathcal{C} = \{\mathbf{z} \in \mathbb{R}^n \big| \ \|\mathbf{z} - \mathbf{x}\|_2 \leq \beta\}, \tag{58}$$

where $\beta > 0$ and $\mathbf{x} \in \mathbb{R}^n$. Let $\gamma > 0$ and $\mu > 0$ be the step-size and the Lagrange multiplier associated with the inequality constraint, respectively. Then the barrier function is expressed as [40]

$$B(\mathbf{z}) = \begin{cases} -\ln\left(\beta - \|\mathbf{z} - \mathbf{x}\|_2\right), & \text{if } \ \|\mathbf{z} - \mathbf{x}\|_2 < \beta, \\ +\infty, & \text{otherwise} \end{cases} \tag{59}$$

For simplicity, we let $\mathbf{Q}_1 = (\mathbf{\Pi} - \mathbf{I}_{2N_t}\tan\phi)$ and $\mathbf{Q}_2 = (\mathbf{\Pi} + \mathbf{I}_{2N_t}\tan\phi)$. Based on (59), the barrier function corresponding to the constraint (55) is written at the bottom of the page. In the case of constraint (56), similar expression is also written for $B_2(\mathbf{w}_2)$ using $\mathbf{Q}_2$. The resulting barrier function is thus

$$B_{\text{robust}}(\mathbf{w}_2) = B_1(\mathbf{w}_2) + B_2(\mathbf{w}_2) \tag{61}$$

Without loss of generality, the constraints (55) and (56) can be further written as

$$\mathbf{\Lambda}^T\mathbf{Q}_1\mathbf{w}_2 + \varsigma\left\|\mathbf{Q}_1\mathbf{w}_2\right\|_2 + \sqrt{\Gamma v_0}\tan\phi \leq 0, \tag{62}$$

$$\mathbf{\Lambda}^T\mathbf{Q}_2\mathbf{w}_2 + \varsigma\left\|\mathbf{Q}_2\mathbf{w}_2\right\|_2 + \sqrt{\Gamma v_0}\tan\phi \leq 0. \tag{63}$$

It can be seen that the upper bound of the two constraints (62) and (63) is zero, Therefore, the effective proximity operator of (61) is obtained the by squaring (62) and (63) and adding the results. Following similar steps presented in subsection III-B, we obtain the proximity operator of the barrier for the robust SLP-DNet (see Appendix A for details).

2) Loss Function of the Robust Power Minimization Problem: The training loss function is the Lagrangian of (57), and can be written as

$$\begin{aligned} \min_{\{\mathbf{w}_2\}} \quad & \|\mathbf{w}_2\|_2^2 \\ \text{s.t.} \quad & \mathbf{\Lambda}^T\mathbf{Q}_1\mathbf{w}_2 + \varsigma\left\|\mathbf{Q}_1\mathbf{w}_2\right\|_2 + \sqrt{\Gamma v_0}\tan\phi \leq 0 \ \forall i \\ & \mathbf{\Lambda}^T\mathbf{Q}_2\mathbf{w}_2 + \varsigma\left\|\mathbf{Q}_2\mathbf{w}_2\right\|_2 + \sqrt{\Gamma v_0}\tan\phi \leq 0 \ \forall i. \end{aligned} \tag{64}$$

Therefore, the loss function of (64) is the regularized Lagrangian parameterized by $\boldsymbol{\theta}_i$ over the entire layers

$$\begin{aligned} \mathcal{L}_{\text{robust}}(\mathbf{w}_2,\ \boldsymbol{\mu}_1,\ \boldsymbol{\mu}_2) &= \frac{1}{N}\sum_{i=1}^{N}\|\mathbf{w}_2\|_2^2 \\ &+ \frac{\boldsymbol{\mu}_1}{N}\sum_{i=1}^{N}\left(\varsigma^2\left\|\mathbf{Q}_1\mathbf{w}_2\right\|_2^2 - \left(\sqrt{\Gamma v_0}\tan\phi - \mathbf{\Lambda}^T\mathbf{Q}_1\mathbf{w}_2\right)^2\right) \\ &+ \frac{\boldsymbol{\mu}_2}{N}\sum_{i=1}^{N}\left(\varsigma^2\left\|\mathbf{Q}_2\mathbf{w}_2\right\|_2^2 - \left(\sqrt{\Gamma v_0}\tan\phi - \mathbf{\Lambda}^T\mathbf{Q}_2\mathbf{w}_2\right)^2\right) \\ &+ \frac{\vartheta}{NL}\sum_{i=1}^{N}\sum_{i=1}^{L}\|\boldsymbol{\theta}_i\|_2^2. \end{aligned} \tag{65}$$

The minimum of (65) with respect to $\mathbf{w}_2$ by equating its derivative to zero

$$\left(1 + \left(\boldsymbol{\mu}_1\left\|\mathbf{Q}_1\right\|_2^2 + \boldsymbol{\mu}_2\left\|\mathbf{Q}_2\right\|_2^2\right)\left(\varsigma^2 - \mathbf{\Lambda}^T\mathbf{\Lambda}\right)\right)\mathbf{w}_2 = -\left(\boldsymbol{\mu}_1\mathbf{Q}_1 + \boldsymbol{\mu}_2\mathbf{Q}_2\right)\mathbf{\Lambda}\sqrt{\Gamma v_0\tan\phi}. \tag{66}$$

For convenience, we redefine the real matrices and vectors as $\begin{bmatrix}\|\mathbf{Q}_1\|_2^2 & \|\mathbf{Q}_2\|_2^2\end{bmatrix} = \bar{\mathbf{q}}_{\text{norm}}$; $\begin{bmatrix}\mathbf{Q}_1 & \mathbf{Q}_2\end{bmatrix} = \bar{\mathbf{Q}}$ and $\begin{bmatrix}\boldsymbol{\mu}_1 & \boldsymbol{\mu}_2\end{bmatrix} = \bar{\boldsymbol{\mu}}$. With these new notations, (66) is simplified to

$$\left(\mathbf{I}_{2N_t} + \bar{\mathbf{q}}_{\text{norm}}\bar{\boldsymbol{\mu}}^T\left(\varsigma^2 - \mathbf{\Lambda}^T\mathbf{\Lambda}\right)\right)\mathbf{w}_2 = -\mathbf{\Lambda}\bar{\mathbf{Q}}\bar{\boldsymbol{\mu}}^T\sqrt{\Gamma w_0}\tan\phi \tag{67}$$

From (67), the optimal transmit precoder is thus

$$\mathbf{w}_2 = -\mathbf{\Lambda}\bar{\mathbf{Q}}\bar{\boldsymbol{\mu}}^T\mathbf{A}^{-1}\sqrt{\Gamma v_0}\tan\phi, \tag{68}$$

where $\mathbf{A} = \left(\mathbf{I}_{2N_t} + \bar{\mathbf{q}}_{\text{norm}}\bar{\boldsymbol{\mu}}^T\left(\varsigma^2 - \mathbf{\Lambda}^T\mathbf{\Lambda}\right)\right)$. Note that the Lagrange multipliers $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ are associated with the barrier term and are randomly initialized from a uniform distribution.

$$B_1(\mathbf{w}_2) = \begin{cases} -\ln\left(-\sqrt{\Gamma v_0}\tan\phi - \left(\mathbf{\Lambda}^T\mathbf{Q}_1\mathbf{w}_2 + \varsigma\left\|\mathbf{Q}_1\mathbf{w}_2\right\|_2\right)\right), & \text{if } \mathbf{\Lambda}^T\mathbf{Q}_1\mathbf{w}_2 + \varsigma\left\|\mathbf{Q}_1\mathbf{w}_2\right\|_2 < -\sqrt{\Gamma v_0}\tan\phi \\ +\infty & \text{otherwise} \end{cases} \tag{60}$$

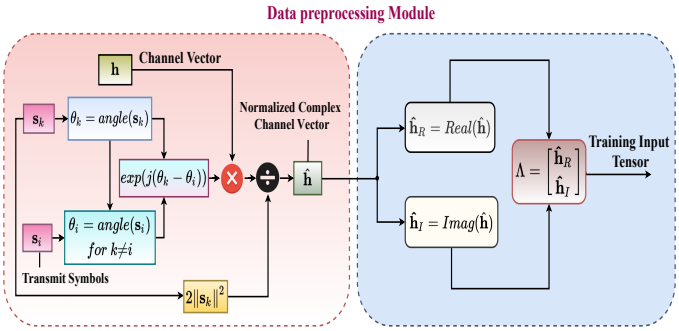


Fig. 3: Dataset Generating Block.

V. Data Generation, Training and Computational Complexity

A. Dataset Generation

The channel coefficients are used to form a dataset and are generated randomly from a normal distribution with zero mean and unit variance. The data input tensor is obtained using (7). We summarize the entire dataset preprocessing procedure in Fig. 3. It can be observed that the input dataset is normalized by the transmit data symbol so that data entries are within the nominal range, and this could potentially aid the training.

B. SLP-DNet Training and Testing

The training of DNNs generally involves three steps: forward propagation, backward propagation, and parameter update. Except where necessary, the training SINR is drawn from a random uniform distribution to enable learning over a wide range of SINR values. The PUM contains r blocks, which form a learning layer. Therefore, each block contains three core components and is trained block-wise for l number of iterations.

Similarly, the APB is trained for k iterations. It is important to note that the number of training iterations of the parameter update module may not necessarily be equal to that of the APB. We train the PUM for 15 iterations and the APB for 10 iterations. To improve the training efficiency, we modify the learning rate by a factor $\alpha \in \mathbb{R}^+$ for every training step. All the training is done with a stochastic gradient descent algorithm using an Adam optimizer. Since the learning is done in an unsupervised fashion, the loss function is the Lagrangian function's statistical mean over the entire training batch samples. During the inference, a feed-forward pass is performed over the entire architecture using the learned Lagrangian multipliers to calculate the precoding vector using (41) and (68) for both SLP and robust SLP formulations, respectively. Finally, at inference, the trained model is run with different SINR values to obtain the required optimal precoding matrix.

C. Computational Complexity Evaluation

The complexities are evaluated in terms of the number of real arithmetic operations involved. For ease of analysis, we convert the SOCP (14) into a standard linear programming (LP)

$$\begin{aligned} \min_{\{\mathbf{z}\}} \quad & \mathbf{c}^T\mathbf{z} \\ \text{s.t.} \quad & \mathbf{c}_k^T\mathbf{z} \leq -\tan\phi\sqrt{\Gamma_i v_0}\ , \ \forall i \end{aligned} \tag{69}$$

where $\mathbf{c} = \begin{bmatrix} 0 & \mathbf{w}_1^T \end{bmatrix}^T \in \mathbb{R}^{(2N_t+1)\times 1}$, $\mathbf{z} = \begin{bmatrix} 1 & \mathbf{w}_1 \end{bmatrix}^T \in \mathbb{R}^{(2N_t+1)\times 1}$, $\mathbf{c}_k = \begin{bmatrix} \left|\mathbf{\Lambda}_i^T\mathbf{\Pi}\mathbf{w}_1\right| & \mathbf{\Lambda}_i^T\tan\phi \end{bmatrix}^T \in \mathbb{R}^{(2N_t+1)\times 1}$ and $\mathbf{W} = [\mathbf{w}_{11}, \cdots, \mathbf{w}_{1K}]$; $\forall i = 1, \cdots, K$. The complexity per iteration for solving convex optimization via IPM is dominated by the formation ($\mathtt{C}_{\text{form}}$) and factorization ($\mathtt{C}_{\text{fact}}$) of the matrix coefficients of m linear equations in m unknowns [48]. For generic IPMs, the complexity is expressed as [48]

$$\mathtt{C}_{\text{total}} = \mathtt{C}_{\text{iter}} \cdot (\mathtt{C}_{\text{form}} + \mathtt{C}_{\text{fact}}) \tag{70}$$

where $\mathtt{C}_{\text{iter}}$ is the iteration complexity required to attain an optimal solution. For a given optimal target accuracy, $\epsilon > 0$, $\mathtt{C}_{\text{iter}}$ is given by

$$\mathtt{C}_{\text{iter}} = \sqrt{\sum_{j=1}^{N_{\text{lc}}} d_j + 2N_{\text{sc}}} \times \ln\left(\frac{1}{\epsilon}\right) \tag{71}$$

where d is the dimension of the constraints, N_{lc} and N_{sc} are the numbers of linear inequality matrix and second order cone (SOC) constraints, respectively. The costs of formation and factorization of the matrix are respectively given by [48]

$$\mathtt{C}_{\text{form}} = \underbrace{m\sum_{j=1}^{N_{\text{lc}}} d_j^3 + m^2\sum_{j=1}^{N_{\text{lc}}} d_j^2}_{\text{due to } N_{\text{lc}}} + \underbrace{m\sum_{j=1}^{N_{\text{sc}}} d_{j=1}^2}_{\text{due to } N_{\text{sc}}};\ \mathtt{C}_{\text{fact}} = m^3. \tag{72}$$

Specifically, we observe that problem (69) has K constraints with dimension $2N_t+1$. Therefore, using (71) and (72), the total computational complexity is thus

$$\mathtt{C}_{\text{total}} = \sqrt{2N_t+1}\left[m(2N_t+1) + m(2N_t+1)^2 + m^3\right]\ln\left(\tfrac{1}{\epsilon}\right). \tag{73}$$

The complexity of BLP can be derived in a similar way and is shown directly in Table I. Conversely, the complexity of the proposed SLP-DNet schemes is the sum of PUM and the APB complexities. Moreover, the complexity of the PUM is dominated by the costs of computing the *'log barrier'* and the feed-forward pass of the shallow CNN (see Table III) that makes up the barrier term associated with the inequality constraint. Similarly, the complexity of the APB is also obtained by computing the arithmetic operations involved during the forward pass of the deep CNN (see Table IV). To derive the analytical complexity of SLP-DNet, we assume a sliding window is used to perform the dominant computation of the convolution operation in the CNN and ignore the nonlinear computational overhead

TABLE I: Complexity analysis of proposed SLP-DNet and benchmark SLP schemes.

Problem	Arithmetic Operations (term; $m = \mathcal{O}(2N_tK)$)	Complexity Order ($n = N_t = K$)
Conventional BLP	$\sqrt{(4N_t+K+2)}\left[m(2N_t+1)+m(2N_t+1)^2+m(K+1)^2+m^3\right]\ln\left(\frac{1}{\epsilon}\right)$	$\mathcal{O}(n^{6.5})$
SLP Optimization-based	$\sqrt{2N_t+1}\left[m(2N_t+1)+m(2N_t+1)^2+m^3\right]\ln\left(\frac{1}{\epsilon}\right)$	$\mathcal{O}(n^{6.5})$
SLP-DNet	$4K^2N_t+42K^2+48KN_t+512K+2$	$\mathcal{O}(n^3)$
SLP-DNet Strict	$4K^2N_t+39K^2+46KN_t+512K+2$	$\mathcal{O}(n^3)$
Robust Conventional BLP	$\sqrt{2K(2N_t+1)}\left[mK(2N_t+1)^3+m^2K(2N_t+1)^2+m^3\right]\ln\left(\frac{1}{\epsilon}\right)$	$\mathcal{O}(n^{7.5})$
Robust SLP Optimization-based	$\sqrt{2(2N_t+1)}\left[2mK(2N_t+1)^2+m^3\right]\ln\left(\frac{1}{\epsilon}\right)$	$\mathcal{O}(n^{6.5})$
Robust SLP-DNet	$16KN_t^2+42K^2+48KN_t+512K$	$\mathcal{O}(n^3)$

due to activations. Therefore, the total computational complexity is expressed as

$$C_{SLP-DNet} = C_{\text{log-br}} + 2\sum_{l=1}^{L_{\text{conv}}} n_{\text{h}}^{[l-1]} n_{\text{w}}^{[l-1]} \left[C_{\text{in}}^{[l-1]} f^{[l]2} + 1\right] C_{\text{out}}^{[l]} + \sum_{j=1}^{L_{\text{fc}}} \left(2M_{\text{in}}^{[j-1]} + 1\right) M_{\text{out}}^{[i]} \quad (74)$$

where n_{h}, n_{w}, f, C_{in} and C_{out} are the height, width of the input tensor, kernel size, number of input and output channels, respectively. Similarly, L_{conv}, L_{fc}, M_{in} and M_{out} are the number of convolution and fully connected (FC) layers, number of input and output neurons in the FC layer, respectively. $C_{\text{log-br}}$ denotes the complexity of the *'log-barrier'* function. Table I shows the summary of the computational complexities of our proposals and the benchmark precoding schemes. As an illustration, we consider the case of a symmetrical system ($N_t = K = n$), and show that the proposed approach has substantially reduced computational complexity of $\mathcal{O}(n^3)$, while the optimization-based SLP approach of $\mathcal{O}(n^{6.5})$ and the conventional BLP is $\mathcal{O}(n^{7.5})$.

VI. Simulation, Results and Discussion

A. Simulation Set-up

We consider a single-cell MISO downlink in which the BS is equipped with four antennas ($N_t = 4$) that serve $K = 4$ single users. We generate 50,000 training and 2000 test samples of Rayleigh fading channel coefficients drawn from the same statistical distribution. The transmit data symbols are modulated using QPSK and 8PSK modulation schemes. The training SINR is randomly drawn from uniform distribution $\Gamma_{\text{train}} \sim \mathcal{U}(\Gamma_{\text{low}}, \Gamma_{\text{high}})$. Adam optimizer is used for stochastic gradient descent algorithm with Lagrangian function as a loss metric. A parametric rectified linear unit (**PReLu**) activation function is used for both convolutional and fully connected layers instead of the traditional **ReLu** function. The reason for this is to address the problem of the dying gradient. After every iteration, the learning rate is reduced by a factor $\alpha = 0.65$ to aid the learning algorithm in converging faster. The proposed learning model is implemented in Pytorch 1.7.1 and Python 3.7.8 on a computer with the following specifications: Intel(R) Core (TM) i7-6700 CPU Core, 32.0GB of RAM. Table II summarizes the simulation parameters

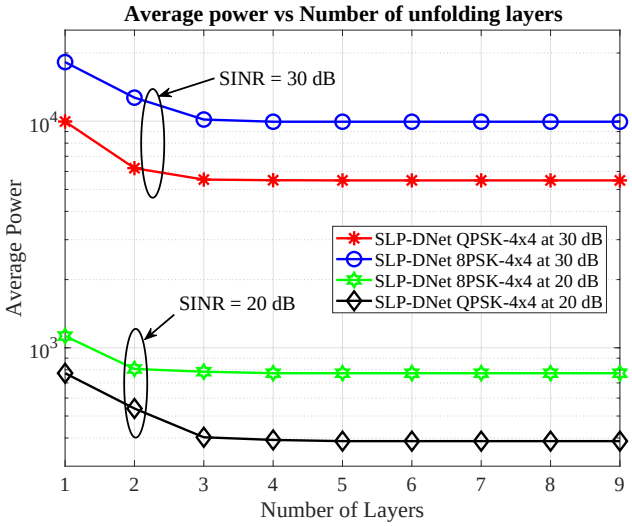


Fig. 4: Transmit Power vs SINR averaged over 2000 test samples vs number of unfolding layers.

used in the experiments. For clarity, the internal neural network structures forming the barrier term component of the PUM and the APB of the SLP-DNet architecture are outlined in Tables III and IV.

Intuitively, to choose the appropriate number of unfolded block layers, we ran the experiments with the different numbers of unfolded blocks (layers) and plotted the average transmit power against the number of layers. Specifically, in our case, we find that the transmit power decreases with the number of layers until the power gains saturate beyond a certain number of layers, as shown in Fig. 4.

TABLE II: Simulation settings

Parameters	Values
Training Samples	50000
Batch Size (B)	200
Test Samples	2000
Training SINR range	0.0dB - 45.0dB
Test SINR range (i-th user SINR)	0.0dB - 35.0dB
Optimizer	SGD with Adam
Initial Learning Rate η	0.001
Learning Rate decay factor α	0.65
Weight Initializer	Xavier Initializer
Number of blocks in the parameter update unit	$B_r = 3$
Training Iterations for each block of the parameter update unit	15
Training iterations for the auxiliary unit	10

TABLE III: Proximity Barrier Function NN Layout

Layer	Parameter, kernel size = 3×3
Input Layer	Input size (B, 1, $2N_t$, K)
Layer 1: Convolutional	Size (B, 1, K, 20); zero padding
Layer 2: Average Pooling	Size ((1, 1), stride = (1, 1))
Layer 3: Activation	Soft-Plus
Layer 4: Flat	Size (B $\times$ 20 $\times$ K^2)
Layer : Fully-connected	Size(B $\times$ 20 $\times$ K^2, 1)
Layer 5: Activation	Soft-Plus function

TABLE IV: Auxiliary Processing Block (APB) NN Structure

Layer	Parameter, kernel size = 3×3
Input Layer	Input size (B, 1, $2N_t$, K)
Layer 1: Convolutional	Size (B, 1, K, 64), dilation = 1 and unit padding
Layer 2: Batch Normalization	eps = 10^{-6}, momentum = 0.1
Layer 3: Activation	PReLu
Layer 4: Convolutional	Size (B, 1, 64, $2N_tK$), dilation = 1 and unit padding
Layer 5: Batch Normalization	eps = 10^{-6}, momentum = 0.1
Layer 6: Activation	PReLu
Layer 7: Convolutional	Size (B, 1, $2N_tK$, 1), dilation = 1 and unit padding

B. Performance Evaluation of Non-Robust SLP-DNet

Firstly, we compare the average transmit power of the classical BLP (2), the SLP optimization-based problems (14), (43) and the proposed SLP-DNet schemes based on (31) and Algorithm 1 for both strict and relaxed angle rotations. The performances of the proposed SLP-DNet and the benchmark schemes (conventional BLP and SLP optimization-based) for strict angle rotation are shown in Fig. 5. The blue line is the classical BLP obtained by solving (2), and the black line is the optimization-based SLP by solving (14) and (43), which are multicast versions of (6) for strict angle and relaxed angle cases, respectively (see Fig. 5 and Fig. 6). It can be observed that the transmit power of the proposed SLP-DNet closely matches the optimisation based SLP, both with significant gains against BLP.

Similarly, we discern the same trend in Fig. 6 for the relaxed angle scenario as observed in Fig. 5. Accordingly, we find from Fig. 6 that the relaxed angle formulation offers significant power savings over the strict angle formulation and is therefore adopted in the subsequent experiments. Furthermore, at 30dB, the performance of SLP-DNet is within 5% of the SLP optimization-based solution. Thus, while the SLP optimization-based offers a slightly lower transmit power at SINR above 30dB, the proposed learning-based model's performance is within $96\% - 98\%$ of the optimization-based solution.

C. Performance Evaluation of Robust SLP-DNet

Figs. 7 and 8 compare the performance of the proposed robust SLP-DNet with the traditional robust block-level precoder [47] and robust SLP precoder [2] for the 4×4 MISO system evaluated at $\varsigma^2 = 10^{-4}$. For simplicity, we use QPSK modulation scheme. Fig. 7 depicts how the average transmit power increases with the SINR thresholds,

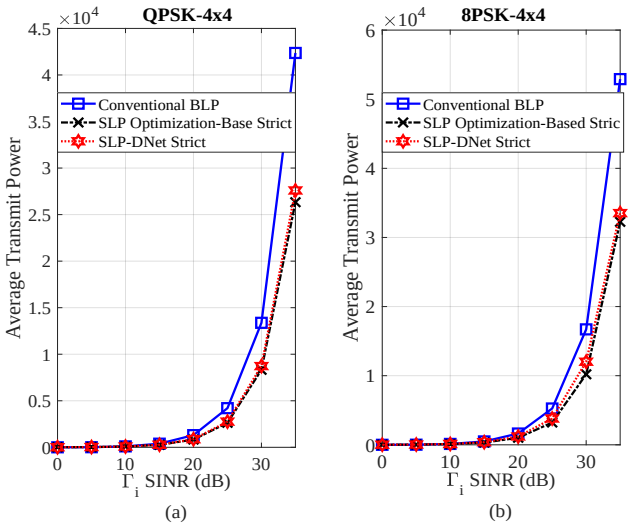


Fig. 5: Transmit Power vs SINR averaged over 2000 test samples for conventional BLP, SLP optimization-based and nonrobust SLP-DNet schemes for $\mathbb{M}$-PSK modulation with $N_t = 4$, $K = 4$ under strict angle rotation.

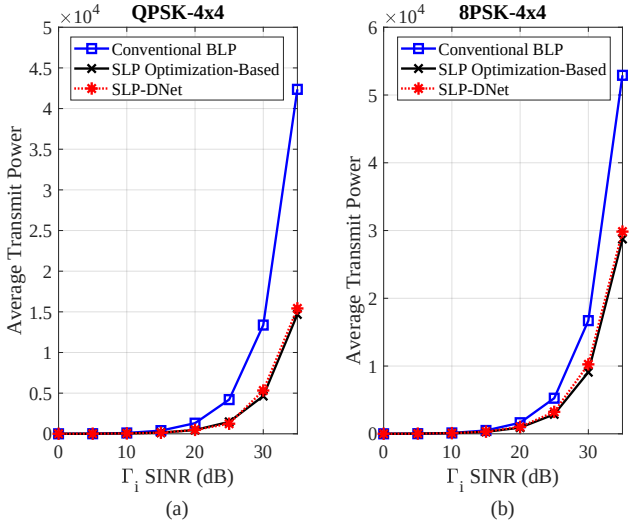


Fig. 6: Transmit Power vs SINR averaged over 2000 test samples for conventional BLP, SLP optimization-based and nonrobust SLP-DNet schemes for $\mathbb{M}$-PSK modulation with $N_t = 4$, $K = 4$ under relaxed angle rotation.

for CSI error bounds $\varsigma^2 = 10^{-4}$. The SLP optimization-based precoding scheme is observed to show significant power savings of more than 60% compared to the conventional optimization solution. Similarly, the proposed unsupervised learning-based precoder portrays a similar transmit power reduction trend. They show considerable power savings of $40\% - 58\%$ against the conventional BLP.

Furthermore, we investigate the effect of the CSI error bounds on the transmit power at 30 dB. Fig. 8 depicts the transmit power variation with increasing CSI error bounds. Moreover, a significant increase in transmit power can be observed where the channel uncertainty lies within the region of CSI error bounds of $\varsigma^2 = 10^{-3}$. Interestingly, like the SLP optimization-based algorithm, the proposed SLP-DNet also shows a descent or moderate increase in

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2023.3270455

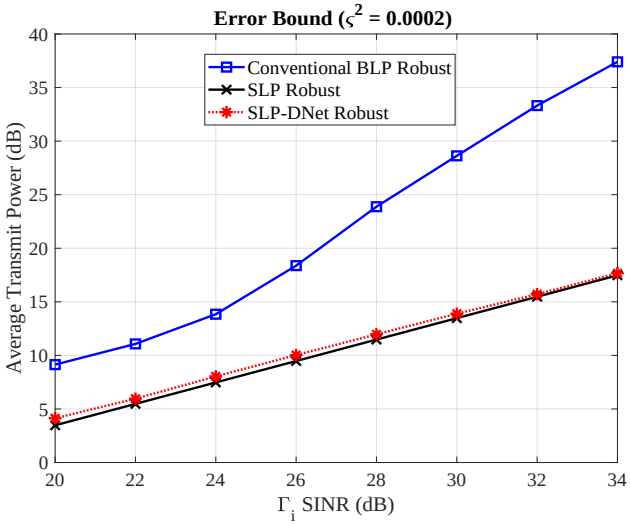


Fig. 7: Transmit Power vs SINR averaged over 2000 test samples for robust conventional, SLP optimization-based and SLP-DNet solutions with $N_t = 4$, $K = 4$ and $\varsigma^2 = 0.0002$.

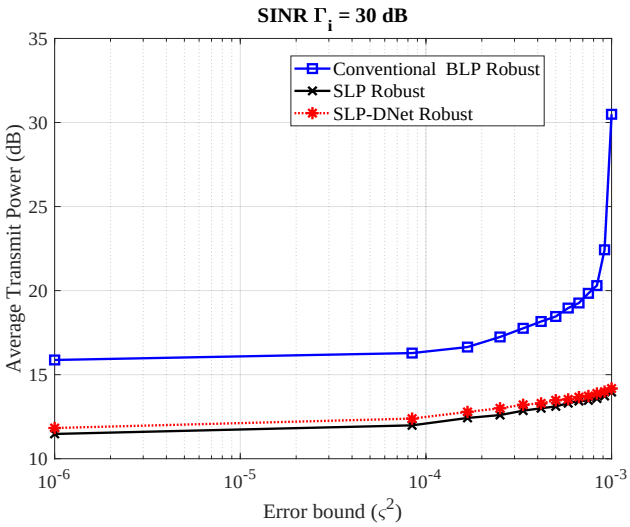
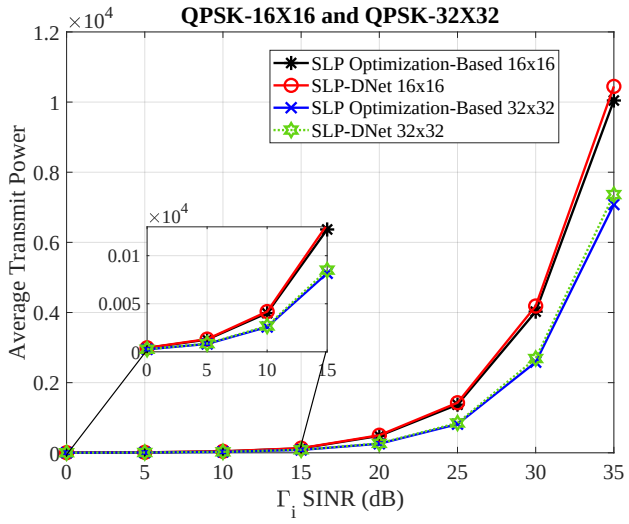
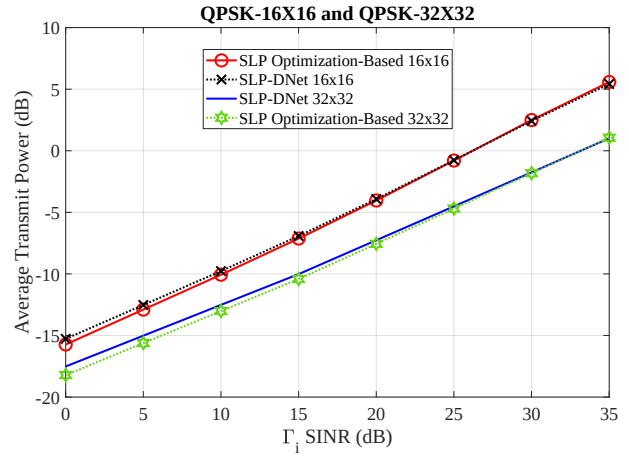


Fig. 8: Transmit Power vs Error-bound for robust conventional BLP, SLP optimization-based and SLP-DNet solutions with $N_t = 4$, $K = 4$.



(a) Transmit Power vs SINR for nonrobust conventional SLP optimization-based and SLP-DNet schemes



(b) Transmit Power vs SINR for robust conventional SLP optimization-based and SLP-DNet schemes

Fig. 9: Transmit Power vs SINR averaged over 2000 test samples for nonrobust and robust conventional SLP optimization-based and SLP-DNet schemes for M-PSK modulation with $N_t = 16, K = 16$ and $N_t = 32, K = 32$ under relaxed angle rotation.

transmit power by exploiting the constructive interference.

Moreover, we perform additional simulations to demonstrate how the proposed SLP learning architecture can scale to more antennas and users for both nonrobust and robust CSI scenarios to investigate the adaptability to different multiantenna layouts. It should be noted that by scaling up the number of users and the BS antennas, the math does not change whatever the parameters. Therefore, the learning architecture will remain unchanged. It can be observed that in Figs. 9(a) and 9(b), the results follow similar trends as those obtained with a moderate number of users (4×4 systems). We also note the performance gap between the proposed learning SLP scheme and the conventional optimization-based approach is rapidly closing beyond 5 dB SINR and matching up with the optimization-based solution at higher SINRs. As expected, scaling up the number of users and BS antennas would naturally result in extra computational complexity. Still, a significant reduction in the transmit power is achieved compared to the 4×4 scenarios.

Figs. 10(a) and 10(b) depict the execution times for nonrobust and robust formulations. It can be seen that both SLP optimization-based algorithm and the proposed learning schemes are feasible for all sets of N_t BS antenna and K mobile users. However, for conventional BLP, the solution is only feasible for $N_t \geq K$. Fig. 10(a) shows the average execution time of the proposed unsupervised learning solutions per symbol averaged over 2000 test samples for nonrobust formulations. The SLP-DNet is observed to be significantly faster than the SLP optimization-based. For example, the theoretical complexity is polynomial order-3 and polynomial order-6.5 or order-7.5 for SLP-DNet and conventional methods, respectively. This is shown in the execution times, where there is a significantly steeper

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2023.3270455

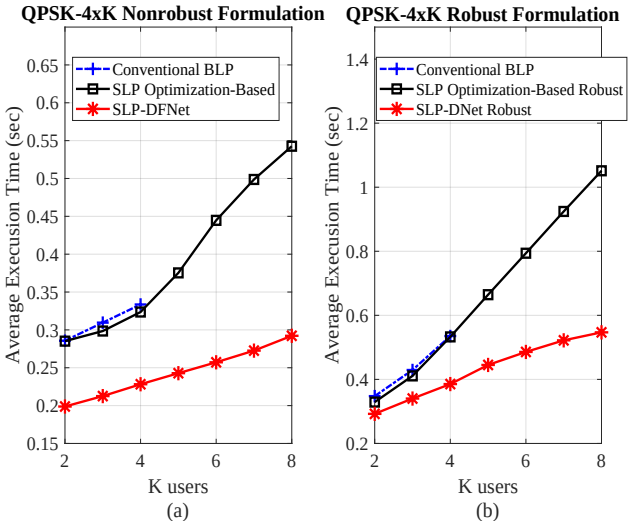


Fig. 10: Comparison of average execution time per sample averaged over 200 test samples for conventional BLP, optimization-based and SLP-DNet solutions with $N_t = 4$ and K users $(2, \cdots, 8)$.

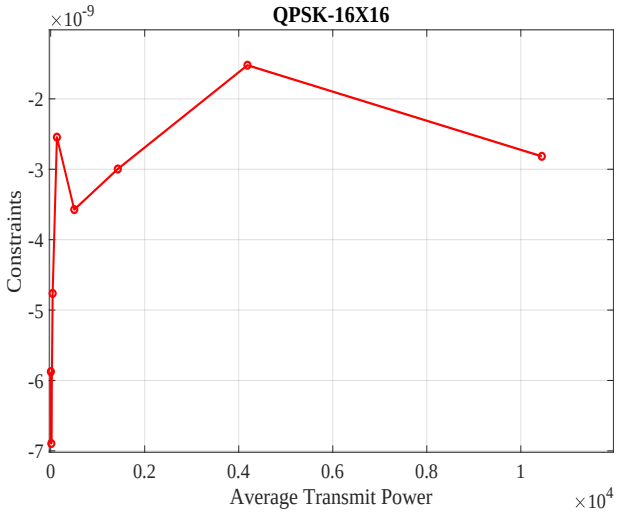


Fig. 11: Average Transmit Power vs. Constraint for SLP-DNet solutions with $N_t = 16$ and $K = 16$ users.

increase in run-time as the number of users increases. The decrease in computational cost is because the dominant operations involved in SLP-DNet at the inference are simple matrix-matrix or vector-matrix convolution. The same trend is also observed in the case of a robust channel scenario, as shown in Fig. 10(b). Therefore, the results in Figs. 10(a) and 10(b) demonstrate that the proposed unsupervised learning-based precoding solutions offer a good trade-off between the performance and computational complexity. Moreover, as per the results obtained, SLP-DNet's performance is within the range of $89\% - 99\%$ of the optimal SLP optimization-based precoding solution. Thus, our proposals demonstrate a favorable tradeoff between the performance and the computational complexity involved.

Finally, to check whether the optimal precoding vector satisfies the constraint, we use the learned optimal precoding vector obtained from the SLP-DNet and plot the constraint against the average transmit power using the constraints in (14). To ensure the constraint is satisfied, the residual constraints (difference between the right-hand and left-hand sides of the constraint) in (14) must always be negative according to the following expression $\left|\mathbf{\Lambda}_i^T \mathbf{\Pi} \mathbf{w}_1\right| \leq \left(\mathbf{\Lambda}_i^T \mathbf{w}_1 - \sqrt{\Gamma_i v_0}\right) \tan\phi$. Therefore, we can further write this expression as $\left|\mathbf{\Lambda}_i^T \mathbf{\Pi} \mathbf{w}_1\right| - \left(\mathbf{\Lambda}_i^T \mathbf{w}_1 - \sqrt{\Gamma_i v_0}\right) \tan\phi \leq 0$, which must be satisfied for every optimal precoding vector. As an illustration, in Fig. 11, we plot constraints against the average transmit power for a BS with 16 antennas serving 16 single antenna users (QPSK-16X16) scenario. It can be observed that the constraints are bounded below the origin on the vertical axis, thus satisfying the constraint requirement for each transmitted power. This means that the points (optimal precoding vectors) are feasible with respect to these constraints (constraints are within the optimization feasible regions).

VII. Conclusion

This paper proposes an unsupervised learning-based precoding framework for a multi-user downlink MISO system. The proposed learning technique exploits the constructive interference for the power minimization problem so that for given QoS constraints, the transmit power available for transmission is minimized. We use domain knowledge to design unsupervised learning architectures by unfolding the proximal interior point method barrier *'log'* function. The proposed learning scheme is then extended to robust precoding designs with imperfect CSI bounded by CSI errors. We demonstrate that our proposal is computationally efficient and allows for feasible solutions to be obtained for problems where traditional numerical optimization like IPM and brute-force maximum likelihood solvers would not converge or would be prohibitively costly.

Appendix A
Proximity operator barrier for Robust SLP

For every transmit precoding vector $\mathbf{w}_2 \in \mathbb{R}^{2N_t \times 1}$, the proximity operator of the barrier $\gamma\mu B_{\text{robust}}(\mathbf{w}_2)$ is given by

$$\Phi_{\text{rb}}(\mathbf{w}_2, \gamma, \mu) = \frac{2\Gamma v_0 \tan^2\phi - X(\mathbf{w}_2, \gamma, \mu)^2}{2\Gamma v_0 \tan^2\phi - X(\mathbf{w}_2, \gamma, \mu)^2 + 2\gamma\mu} \mathbf{w}_2 \tag{75}$$

where $X(\mathbf{w}_2, \gamma, \mu)$ is the unique solution of the cubic equation expressed as [38]

$$\begin{aligned} &x^3 - \left(\left(\varsigma^2 - \mathbf{\Lambda}^T\mathbf{\Lambda}\right)\|\mathbf{w}_2\|_2 + 4\mathbf{\Lambda}^T\mathbf{w}_2 \tan\phi\sqrt{\Gamma v_0}\right) x^2 \\ &+ \left(2\Gamma v_0 \tan^2\phi + 2\gamma\mu\right) x + \\ &2\Gamma v_0 \tan^2\phi \left(\left(\varsigma^2 - \mathbf{\Lambda}^T\mathbf{\Lambda}\right)\|\mathbf{w}_2\|_2 + 4\mathbf{\Lambda}^T\mathbf{w}_2 \tan\phi\sqrt{\Gamma v_0}\right) = 0. \end{aligned} \tag{76}$$

It can be observed that (76) is a cubic equation and can be solved analytically. In the final analysis, following similar steps as in (25)-(29), the robust deep-unfolded model is obtained by finding the Jacobean matrix of (75) with respect to the optimization variable $\mathbf{w}_2$, and the derivatives with respect to the step-size $\gamma > 0$ and the Lagrange multiplier

associated with the inequality constraint $\mu > 0$. We use similar concepts presented in subsection III-B to formulate the learning algorithm of the robust SLP as a series of subproblems with respect to the combined effect of the two inequality constraints as follows

$$\min_{\mathbf{w}_2 \in \mathbb{R}^{2\mathbf{N_t}\times 1}} \quad \|\mathbf{w}_2\|_2^2 + \lambda \mathbf{w}_2 + \mu B_{\text{robust}}(\mathbf{w}_2). \tag{77}$$

Similar to a nonrobust SLP-DNet, the update rule for every iteration is expressed as

$$\mathbf{w}_2^{[r+1]} = \text{prox}_{\gamma^{[r]}\mu^{[r]}B_{\text{robust}}}\left(\mathbf{w}_2^{[r]} - \gamma^{[r]}\Delta D_{\text{robust}}(\mathbf{w}_2^{[r]}, \lambda^{[r]})\right) \tag{78}$$

where

$$D_{\text{robust}}(\mathbf{w}_2^{[r]}, \lambda^{[r]}) = \|\mathbf{w}_2\|_2^2 + \lambda \mathbf{w}_2. \tag{79}$$

References

[1] C. Masouros and T. Ratnarajah, "Interference as a source of green signal power in cognitive relay assisted co-existing mimo wireless transmissions," *IEEE Transactions on Communications*, vol. 60, no. 2, pp. 525–536, 2011.

[2] C. Masouros and G. Zheng, "Exploiting known interference as green signal power for downlink beamforming optimization," *IEEE Transactions on Signal processing*, vol. 63, no. 14, pp. 3628–3640, 2015.

[3] C. Masouros, "Harvesting signal power from constructive interference in multiuser downlinks," in *Wireless Information and Power Transfer: A New Paradigm for Green Communications*. Springer, 2018, pp. 87–122.

[4] Y. Ni, S. Jin, W. Xu, Y. Wang, M. Matthaiou, and H. Zhu, "Beamforming and interference cancellation for d2d communication underlaying cellular networks," *IEEE Transactions on Communications*, vol. 64, no. 2, pp. 832–846, 2015.

[5] S. Hong, J. Brand, J. I. Choi, M. Jain, J. Mehlman, S. Katti, and P. Levis, "Applications of self-interference cancellation in 5g and beyond," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 114–121, 2014.

[6] C. Masouros and E. Alsusa, "A novel transmitter-based selective-precoding technique for ds/cdma systems," in *2007 IEEE International Conference on Communications*. IEEE, 2007, pp. 2829–2834.

[7] C. Masouros and E. Alsusa, "Dynamic linear precoding for the exploitation of known interference in mimo broadcast systems," *IEEE Transactions on Wireless Communications*, vol. 8, no. 3, pp. 1396–1404, 2009.

[8] C. Masouros, "Correlation rotation linear precoding for mimo broadcast communications," *IEEE Transactions on Signal Processing*, vol. 59, no. 1, pp. 252–262, 2010.

[9] C. Masouros, M. Sellathurai, and T. Ratnarajah, "Vector perturbation based on symbol scaling for limited feedback miso downlinks," *IEEE Transactions on Signal Processing*, vol. 62, no. 3, pp. 562–571, 2014.

[10] M. Alodeh, S. Chatzinotas, and B. Ottersten, "Constructive multiuser interference in symbol level precoding for the miso downlink channel," *IEEE Transactions on Signal processing*, vol. 63, no. 9, pp. 2239–2252, 2015.

[11] P. V. Amadori and C. Masouros, "Constant envelope precoding by interference exploitation in phase shift keying-modulated multiuser transmission," *IEEE Transactions on Wireless Communications*, vol. 16, no. 1, pp. 538–550, 2016.

[12] M. Alodeh, S. Chatzinotas, and B. Ottersten, "Symbol-level multiuser miso precoding for multi-level adaptive modulation," *IEEE Transactions on Wireless Communications*, vol. 16, no. 8, pp. 5511–5524, 2017.

[13] A. Li, C. Masouros, F. Liu, and A. L. Swindlehurst, "Massive mimo 1-bit dac transmission: A low-complexity symbol scaling approach," *IEEE Transactions on Wireless Communications*, vol. 17, no. 11, pp. 7559–7575, 2018.

[14] A. Li and C. Masouros, "A two-stage vector perturbation scheme for adaptive modulation in downlink mu-mimo," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 7785–7791, 2015.

[15] S. Timotheou, G. Zheng, C. Masouros, and I. Krikidis, "Exploiting constructive interference for simultaneous wireless information and power transfer in multiuser downlink systems," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 5, pp. 1772–1784, 2016.

[16] A. Li and C. Masouros, "Exploiting constructive mutual coupling in p2p mimo by analog-digital phase alignment," *IEEE Transactions on Wireless Communications*, vol. 16, no. 3, pp. 1948–1962, 2017.

[17] D. Spano, M. Alodeh, S. Chatzinotas, and B. Ottersten, "Symbol-level precoding for the nonlinear multiuser miso downlink channel," *IEEE Transactions on Signal Processing*, vol. 66, no. 5, pp. 1331–1345, 2017.

[18] A. Kalantari, C. Tsinos, M. Soltanalian, S. Chatzinotas, W.-K. Ma, and B. Ottersten, "Spatial peak power minimization for relaxed phase m-psk mimo directional modulation transmitter," in *2017 25th European Signal Processing Conference (EUSIPCO)*. IEEE, 2017, pp. 2011–2015.

[19] K. L. Law, C. Masouros, and M. Pesavento, "Transmit precoding for interference exploitation in the underlay cognitive radio z-channel," *IEEE Transactions on Signal Processing*, vol. 65, no. 14, pp. 3617–3631, 2017.

[20] A. Li and C. Masouros, "Interference exploitation precoding made practical: Optimal closed-form solutions for psk modulations," *IEEE Transactions on Wireless Communications*, vol. 17, no. 11, pp. 7661–7676, 2018.

[21] A. Mohammad, C. Masouros, and Y. Andreopoulos, "Complexity-scalable neural-network-based mimo detection with learnable weight scaling," *IEEE Transactions on Communications*, vol. 68, no. 10, pp. 6101–6113, 2020.

[22] A. Alkhateeb, S. Alex, P. Varkey, Y. Li, Q. Qu, and D. Tujkovic, "Deep learning coordinated beamforming for highly-mobile millimeter wave systems," *IEEE Access*, vol. 6, pp. 37 328–37 348, 2018.

[23] A. Mohammad, C. Masouros, and Y. Andreopoulos, "Accelerated learning-based mimo detection through weighted neural network design," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, 2020, pp. 1–6.

[24] H. Huang, W. Xia, J. Xiong, J. Yang, G. Zheng, and X. Zhu, "Unsupervised learning-based fast beamforming design for downlink mimo," *IEEE Access*, vol. 7, pp. 7599–7605, 2018.

[25] 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.

[26] P. de Kerret and D. Gesbert, "Robust decentralized joint precoding using team deep neural network," in *2018 15th International Symposium on Wireless Communication Systems (ISWCS)*. IEEE, 2018, pp. 1–5.

[27] H. Huang, Y. Peng, J. Yang, W. Xia, and G. Gui, "Fast beamforming design via deep learning," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 1065–1069, 2019.

[28] W. Xia, G. Zheng, Y. Zhu, J. Zhang, J. Wang, and A. P. Petropulu, "A deep learning framework for optimization of MISO downlink beamforming," *IEEE Transactions on Communications*, vol. 68, no. 3, pp. 1866–1880, 2019.

[29] Z. Lei, X. Liao, Z. Gao, and A. Li, "Ci-nn: A model-driven deep learning based constructive interference precoding scheme," *IEEE Communications Letters*, 2021.

[30] F. Sohrabi, H. V. Cheng, and W. Yu, "Robust symbol-level precoding via autoencoder-based deep learning," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 8951–8955.

[31] A. Mohammad, C. Masouros, and Y. Andreopoulos, "An unsupervised learning-based approach for symbol-level-precoding," in *2021 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2021, pp. 1–6.

[32] Z. Bo, R. Liu, M. Li, and Q. Liu, "Deep learning based efficient symbol-level precoding design for mu-miso systems," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 8, pp. 8309–8313, 2021.

[33] A. Mohammad, C. Masouros, and Y. Andreopoulos, "Learning-based symbol level precoding: A memory-efficient unsupervised learning approach," in *2022 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2022, pp. 429–434.

[34] N. Pustelnik, C. Chaux, and J.-C. Pesquet, "Parallel proximal algorithm for image restoration using hybrid regularization," *IEEE transactions on Image Processing*, vol. 20, no. 9, pp. 2450–2462, 2011.
[35] E. Björnson, M. Bengtsson, and B. Ottersten, "Optimal multiuser transmit beamforming: A difficult problem with a simple solution structure [lecture notes]," *IEEE Signal Processing Magazine*, vol. 31, no. 4, pp. 142–148, 2014.
[36] C. Masouros, T. Ratnarajah, M. Sellathurai, C. Papadias, and A. Shukla, "Known interference in wireless communications: a limiting factor or a potential source of green signal power?" *IEEE Comms. Mag*, vol. 51, no. 10, pp. 162–171, 2013.
[37] A. Li, D. Spano, J. Krivochiza, S. Domouchtsidis, C. G. Tsinos, C. Masouros, S. Chatzinotas, Y. Li, B. Vucetic, and B. Ottersten, "A tutorial on interference exploitation via symbol-level precoding: Overview, state-of-the-art and future directions," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 2, pp. 796–839, 2020.
[38] C. Bertocchi, E. Chouzenoux, M.-C. Corbineau, J.-C. Pesquet, and M. Prato, "Deep unfolding of a proximal interior point method for image restoration," *Inverse Problems*, vol. 36, no. 3, p. 034005, 2020.
[39] R. Hauser, "Interior-point methods for inequality constrained optimization," 2007.
[40] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and Trends in optimization*, vol. 1, no. 3, pp. 127–239, 2014.
[41] R. M. Freund, "Penalty and barrier methods for constrained optimization," *Lecture Notes, Massachusetts Institute of Technology*, 2004.
[42] N. Pustelnik and L. Condat, "Proximity operator of a sum of functions; application to depth map estimation," *IEEE Signal Processing Letters*, vol. 24, no. 12, pp. 1827–1831, 2017.
[43] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.
[44] A. Kaplan and R. Tichatschke, "Proximal methods in view of interior-point strategies," *Journal of optimization theory and applications*, vol. 98, no. 2, pp. 399–429, 1998.
[45] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*. PMLR, 2015, pp. 448–456.
[46] B. Wang and D. Klabjan, "Regularization for unsupervised deep neural nets," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 2017, pp. 2681–2681.
[47] G. Zheng, K.-K. Wong, and T.-S. Ng, "Robust linear mimo in the downlink: A worst-case optimization with ellipsoidal uncertainty regions," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, pp. 1–15, 2008.
[48] K.-Y. Wang, A. M.-C. So, T.-H. Chang, W.-K. Ma, and C.-Y. Chi, "Outage constrained robust transmit optimization for multiuser miso downlinks: Tractable approximations by conic optimization," *IEEE Transactions on Signal Processing*, vol. 62, no. 21, pp. 5690–5705, 2014.