Robust $\mathcal{H}_2/\mathcal{H}_\infty$ Tuning of PID-based Optimization and Frequency-Domain Comparison with Adam

Vishesh Jain [*] **Mayank Baranwal** [*,**]

[*] *Indian Institute of Technology, Bombay, 400076 India*
[**] *Tata Consultancy Services Research, Mumbai, 400607 India*

Abstract: PID-based optimization algorithms (PIDAO) have recently demonstrated empirical robustness against gradient noise in machine learning. However, a theoretical framework for tuning these algorithms to guarantee stability and noise rejection is lacking. In this work, we formulate PIDAO as a discrete-time Lur'e system and utilize Integral Quadratic Constraints (IQCs) to analyze its robustness. We propose an $\mathcal{H}_2/\mathcal{H}_\infty$ synthesis framework to optimally tune PIDAO gains, balancing convergence speed with disturbance attenuation. Furthermore, we introduce a fixed-point linearization of the Adam optimizer, enabling a comparative control-theoretic analysis. Frequency-domain results and neural network training experiments demonstrate that PIDAO, when tuned via our robust control framework, achieves superior noise attenuation and stability margins compared to Adam.

Keywords: Accelerated optimization, Continuous-time optimization, Optimization algorithms, Robust control, Integral quadratic constraints

1. INTRODUCTION

The training of deep neural networks is fundamentally a large-scale, non-convex optimization problem. As machine learning models grow in complexity, the efficiency and reliability of the optimization algorithms used to train them become critical. While Stochastic Gradient Descent (SGD) remains the bedrock of this field, modern deep learning relies heavily on accelerated variants that incorporate momentum and adaptive learning rates, such as Nesterov's Accelerated Gradient (NAG) (Nesterov, 1983) and Adam (Kingma and Ba, 2015). These algorithms are designed to navigate complex loss landscapes, escaping shallow local minima and traversing high-curvature regions. However, despite their widespread adoption, the tuning of their hyperparameters (effectively the controller gains) remains largely heuristic, often requiring expensive grid searches and trial-and-error.

In recent years, the intersection of control theory and optimization has provided a rigorous framework for analyzing these algorithms. By viewing the optimization process as a feedback control system, where the gradient computation acts as the plant and the parameter update rule acts as the controller, researchers have utilized tools such as Integral Quadratic Constraints (IQCs) and Lyapunov stability theory to certify convergence rates (Lessard et al., 2016; Hu and Lessard, 2017). This perspective treats the unknown curvature of the objective function as bounded uncertainty and the stochastic gradient noise as an exogenous disturbance (Cyrus et al., 2018).

While this literature has successfully analyzed existing algorithms, there has been less focus on synthesizing optimal controllers for machine learning tasks using robust control techniques. Specifically, the classic Proportional-Integral-Derivative (PID) controller, despite its ubiquity in industrial automation, has only recently been adapted for deep learning optimization (An et al., 2018; Chen et al., 2024). These PID accelerated optimization (PIDAO) algorithms introduce a differential term to dampen oscillations, theoretically offering superior stability compared to standard momentum methods. However, PIDAO introduces three coupled hyperparameters, making manual tuning difficult and rendering standard convergence bounds loose.

Furthermore, a significant gap exists in comparing linear control-based optimizers (like PIDAO) with nonlinear adaptive methods (like Adam). Adam is often viewed as the "gold standard" due to its adaptive scaling of gradients, yet it suffers from well-documented instability issues and poor generalization in certain regimes (Sashank et al., 2018). Because Adam involves nonlinear state updates, it does not fit naturally into standard linear robust control frameworks, making a direct frequency-domain comparison with PID methods elusive.

In this work, we bridge this gap by applying $\mathcal{H}_2/\mathcal{H}_\infty$ robust control synthesis to the design of optimization algorithms. We move beyond simple analysis to actively design the gains of a PIDAO optimizer, balancing convergence speed (bandwidth) against noise rejection (roll-off). The primary contributions of this work are fourfold:

1. Robust Control Formulation of PIDAO: We cast the PID-based Accelerated Optimization (PIDAO) algorithm as a discrete-time Lur'e system subject to sector-bounded uncertainty (representing gradient curvature) and additive stochastic disturbances (representing gradient noise). This formulation allows us to apply IQCs to rigorously guarantee stability and convergence rates.

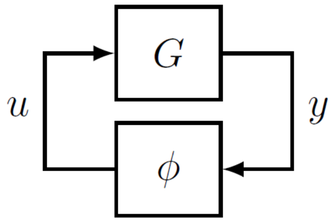

Fig. 1. Lur'e System setup as shown in Cyrus et al. (2018)

2. $\mathcal{H}_2/\mathcal{H}_\infty$ Synthesis for Hyperparameter Tuning: Unlike prior works that primarily analyze existing optimization parameters, we propose a synthesis framework to design them. We formulate a multi-objective optimization problem involving Bilinear Matrix Inequalities (BMIs) to solve for PIDAO gains that minimize the $\mathcal{H}_2$ norm (variance due to noise) while satisfying $\mathcal{H}_\infty$ constraints (robustness to curvature uncertainty).
3. Fixed-Point Linearization of Adam: We introduce a novel local linearization of the Adam optimizer around its equilibrium. By approximating the adaptive second-moment estimator as a fixed preconditioning term, we derive a linear transfer function for Adam. This enables, for the first time, a direct frequency-domain comparison between the adaptive Adam algorithm and linear momentum methods using classical control tools.
4. Frequency-Domain Insights & Empirical Validation: We perform a comparative Bode analysis revealing that the tuned PIDAO exhibits superior high-frequency roll-off compared to the equivalent linearized Adam dynamics. This theoretical finding, that PIDAO naturally suppresses gradient noise better than Adam's intrinsic dynamics, is validated through experiments.

2. BACKGROUND

Establishing a link between iterative optimization algorithms and robust control dynamics requires interpreting the former as feedback systems. In this section, we present the foundational mathematical concepts that will be used throughout the rest of the paper.

2.1 Lur'e Setup

We utilize this setup, shown in Fig. 1, to visualize a linear dynamical system as a set of recursive linear equations. The discrete-time Lur'e setup consists of a linear part, representing the optimizer's dynamics and connected to it, the algorithm's nonlinear gradient feedback ϕ.

$$\begin{aligned} \xi_{k+1} &= A\xi_k + Bu_k \\ y_k &= C\xi_k + Du_k \\ u_k &= \phi(y_k). \end{aligned} \tag{1}$$

As illustrated in Chen et al. (2024) and Lessard et al. (2016), this modeling approach effectively captures many optimization algorithms and facilitates IQC analysis. We demonstrate that PIDAO and a linearized ADAM can be expressed within this framework as well.

2.2 PIDAO(Single Integrator) and Adam Optimizers

The discrete-time PIDAO(SI), originally introduced in Chen et al. (2024) reads:

$$\begin{aligned} x_{k+1} - x_k &= h\, y_{k+1} - hk_d \nabla f(x_k), \\ y_{k+1} - y_k &= -hay_{k+1} - h(k_p - ak_d)\nabla f(x_k) - hk_i z_{k+1}, \\ z_{k+1} - z_k &= h\nabla f(x_k), \end{aligned} \tag{2}$$

where h is the step size (analogous to learning rate) and (k_p, k_i, k_d, a) denote the proportional, integral, derivative, and filter parameters, respectively. Here k represents the iteration index. Similarly, Adam's updates are described by the following set of iterative equations:

$$\begin{aligned} m_k &= \beta_1 m_{k-1} + (1-\beta_1)\,\nabla f(x_k), \\ v_k &= \beta_2 v_{k-1} + (1-\beta_2)\,(\nabla f(x_k))^2, \\ x_{k+1} &= x_k - \eta\,\frac{m_k}{\sqrt{v_k}+\varepsilon}, \end{aligned} \tag{3}$$

where η is the learning rate, $\beta_1, \beta_2 \in (0,1)$ are exponential decay rates, and $\varepsilon > 0$ ensures numerical stability.

3. PIDAO AS A FEEDBACK SYSTEM

Our primary objective is to model and analyze optimization algorithms as feedback systems, and employ tools from robust control theory for efficient parameter tuning. Below we provide detailed analysis of PIDAO(SI) as a feedback system.

3.1 Plant Transfer Function

From (2), we define $v_k = \nabla f(x_k)$, and denote the Z-transforms as, $X(z), Y(z), Z(z), V(z)$. This gives us the following transformations:

$$\begin{aligned} (z-1)X(z) &= hzY(z) - hk_d V(z), \\ Y(z)[z(1+ha)-1] &= -h(k_p - ak_d)V(z) - hk_i z Z(z), \\ Z(z) &= \frac{h}{z-1}V(z). \end{aligned}$$

One can refer to Anagnostides and Panageas (2022) for similar treatment on alternate gradient algorithms. For the closed-loop plant transfer function $G(z) = \frac{X(z)}{V(z)}$, we obtain

$$G(z) = \frac{-h^2 z(k_p - ak_d)(z-1) - h^3 z^2 k_i - hk_d(z-1)^2}{(z-1)^2(z(1+ha)-1)} \tag{4}$$

Invoking the forward–Euler approximation $z \approx 1+sT$, and upon substituting this expression into (4) while neglecting higher–order contributions in s and T in the denominator, we obtain

$$G(s) \approx \frac{1}{ha}\left[\frac{-h^3 k_i}{s^2T^2} - \frac{h^2A + 2h^3k_i}{sT} - (h^2A + h^3k_i + hk_d)\right] \tag{5}$$

where $A := k_p - ak_d$. Here, T denotes the sampling period. As discussed in Chen et al. (2024), the parameter h emerges from the forward Euler discretization used to derive the iterative updates. However, for the purposes of our analysis, we primarily interpret h as the learning rate and explicitly distinguish between the roles of h and T. Equation (5) then provides approximate expressions for the proportional, integral, and derivative gains in terms of the PIDAO(SI) parameters.

3.2 Margins and Crossover Frequency for a Quadratic Objective

Having derived the parametric form of the controller, we now focus on the special case of a quadratic objective function, $f(x) = \frac{q}{2}x^2$, to enable a more transparent loop-gain analysis. This facilitates the study of potential fundamental trade-offs between robustness and convergence. In particular, the robustness margins of the closed-loop system directly relate to the convergence behavior of the optimizer in the presence of stochastic gradient noise.

For the quadratic objective, the controller reduces to the constant Hessian $C(s) = q$. The open-loop transfer function is therefore given by the product of the controller and the plant transfer function $G(s)$ defined in (5):

$$\mathbf{L}(s) = q\, G(s).$$

Introduce the shorthand parameters

$$\theta = h^3 k_i, \qquad \kappa = h^2 A + 2h^3 k_i, \qquad \lambda = h^2 A + h^3 k_i + h k_d,$$

under which the open-loop transfer function becomes

$$L(j\omega) = q\, \frac{-\theta - j\omega T\, \kappa + \omega^2 T^2\, \lambda}{-ha\, \omega^2 T^2}.$$

The gain crossover frequency ω_{gc} is defined as the frequency at which the loop-gain magnitude equals unity, i.e., $|L(j\omega_{gc})| = 1$. Evaluating the magnitude condition yields

$$\frac{q^2}{(ha\, \omega^2 T^2)^2} \Big[\underbrace{(-\theta + \omega^2 T^2 \lambda)^2}_{\text{Real part}^2} + \underbrace{(-\omega T \kappa)^2}_{\text{Imaginary part}^2} \Big] = 1.$$

Letting $X = \omega^2$, the above expression can be rearranged into the following **cubic polynomial in** X:

$$X^3 A_{\text{poly}} + X^2 B_{\text{poly}} + X\, C_{\text{poly}} + D_{\text{poly}} = 0.$$

In our setting $D_{\text{poly}} = 0$, reducing the problem to a quadratic equation in X. The gain crossover frequency is then obtained from the positive real root(s):

$$\omega_{gc}^2 = \frac{-B_{\text{poly}} \pm \sqrt{B_{\text{poly}}^2 - 4A_{\text{poly}} C_{\text{poly}}}}{2A_{\text{poly}}}.$$

The polynomial coefficients are given by

$$\begin{aligned} A_{\text{poly}} &= q^2 T^4 \lambda^2 - (haT^2)^2, \\ B_{\text{poly}} &= q^2 T^2 (\kappa^2 - 2\theta\lambda), \\ C_{\text{poly}} &= q^2 \theta^2. \end{aligned} \tag{6}$$

The phase margin is then evaluated at ω_{gc}:

$$\text{PM} = 180^\circ + \angle \mathbf{L}(j\omega_{gc}),$$

where $\angle L(j\omega) = \tan^{-1}\left(\frac{\text{Im}(L(j\omega))}{\text{Re}(L(j\omega))}\right)$. Substituting the expressions for the real and imaginary parts gives $\text{PM} = 180^\circ + \tan^{-1}\left(\frac{-\omega_{gc} T\, \kappa}{-\theta + \omega_{gc}^2 T^2 \lambda}\right)$.

3.3 State-Space Derivation

We refer to the Lur'e setup and equations (1) to obtain matrices A, B, C, D for the PIDAO(SI) update. We define the state as $s_k = [x_k,\ y_k,\ z_k]^\top$ and its associated closed form $s_{k+1} = As_k + Bv_k$ and $x_k = Cs_k + Dv_k$ where $v_k = \nabla f(x_k)$. After some algebraic manuipulations, we obtain

$$A = \begin{bmatrix} 1 & h\alpha & h\gamma \\ 0 & \alpha & \gamma \\ 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} h\beta - hk_i d \\ \beta \\ h \end{bmatrix}, \quad C = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^\top$$

where $D = 0$ and we have α, β, γ defined as:

$$\alpha := \frac{1}{1+ha},\ \beta := \frac{-h\big(k_p - ak_d + hk_i\big)}{1+ha},\ \gamma := \frac{-hk_i}{1+ha}.$$

4. EQUILIBRIUM POINT LINEARIZED ADAM

We now turn our attention to a similar analysis for the Adam optimizer. Deriving a closed-form characterization for Adam is inherently challenging due to the nonlinear structure of its update rules. Consequently, we restrict our analysis to a linearized variant of Adam, obtained by linearizing the dynamics about the equilibrium point.

4.1 State-Space Setup

By linearizing ADAM about an equilibrium point, it can be translated into a linear time-invariant system. Hence, it allows for the quantification of its local stability, sensitivity, and robustness properties.

About the equilibrium point $(x^\star, m^\star, v^\star)$ we obtain, $x_{k+1} = x_k = x^\star$, $m_k = m_{k-1} = m^\star$, and $v_k = v_{k-1} = v^\star$. Substituting into (3) yields,

$$\begin{aligned} m^\star &= \beta_1 m^\star + (1-\beta_1)\nabla f(x^\star), \\ v^\star &= \beta_2 v^\star + (1-\beta_2)(\nabla f(x^\star))^2. \end{aligned}$$

About the equilibrium point, we define $e_k = x_k - x^\star$ and $g_k = \nabla f(x_k) \approx He_k$ where $H = \nabla^2 f(x^\star)$ denotes the local Hessian. We also assume that the variance update converges to some positive constant $v_* > 0$ leading to a constant preconditioning factor given by, $D = \text{diag}\Big(\frac{1}{\sqrt{v_\star} + \varepsilon}\Big)$.

Neglecting bias corrections and higher-order nonlinearities, the Adam updates reduce to:

$$\begin{aligned} m_k &= \beta_1 m_{k-1} + (1-\beta_1)He_k, \\ e_{k+1} &= e_k - \eta D m_k. \end{aligned}$$

Substituting $\mathbf{u}_k = He_k$ into the update equations, expressing all terms in terms of $\mathbf{s}_k = [e_k \quad m_{k-1}]^\top$ and $\mathbf{u}_k$, we obtain the state-space update equations,

$$\begin{bmatrix} e_{k+1} \\ m_k \end{bmatrix} = \underbrace{\begin{bmatrix} I & -\eta D\beta_1 \\ 0 & \beta_1 I \end{bmatrix}}_{\mathbf{A}} \begin{bmatrix} e_k \\ m_{k-1} \end{bmatrix} + \underbrace{\begin{bmatrix} -\eta D(1-\beta_1) \\ (1-\beta_1)I \end{bmatrix}}_{\mathbf{B}} u_k$$

$$y_k = \underbrace{[I\ 0]}_{\mathbf{C}} \begin{bmatrix} e_k \\ m_{k-1} \end{bmatrix} + \underbrace{0}_{\mathbf{D}}\, u_k$$

4.2 Margins and Crossover Frequency for a Quadratic Objective

Using the same derivation procedure employed to obtain (4), we now derive the corresponding linearized model for the Adam optimizer about the equilibrium point. The resulting discrete-time plant transfer function is given by

$$G(z) = \frac{-\eta D(1-\beta_1)z}{(z-1)(z-\beta_1)}.$$

To facilitate frequency–domain analysis, we invoke the forward Euler discretization $z = 1 + sT$ and introduce the decay-rate proxy $\alpha_1 := 1-\beta_1$. Under this substitution, the continuous-time approximation of the linearized Adam plant becomes

$$G(s) = -\eta D\alpha_1 \frac{1+sT}{sT(\alpha_1 + sT)}. \quad (7)$$

The corresponding open-loop transfer function is defined as $L(s) = q \cdot G(s)$, yielding

$$\mathbf{L}(s) = q \cdot \frac{-\eta D\alpha_1(1+sT)}{sT(\alpha_1+sT)}.$$

Letting $K = q\eta D\alpha_1$, the gain crossover frequency ω_{gc} is determined from the unit-magnitude condition $|L(j\omega_{gc})| = 1$, which leads to

$$\frac{-K}{\omega^2T^2(\omega^2T^2+\alpha_1^2)}\left[\underbrace{(-\omega^2T^2+\omega^2T^2\alpha_1)}_{\text{Real Part}^2}+\underbrace{(-\omega T\alpha_1-\omega^3T^3)}_{\text{Imaginary Part}^2}\right]=1.$$

Solving the above expression yields the closed-form gain crossover frequency

$$\omega_{gc} = \sqrt{\frac{(K^2-\alpha_1^2)+\sqrt{(K^2-\alpha_1^2)^2+4K^2}}{2T^2}}. \quad (8)$$

Next, evaluating the phase of the open-loop transfer function at ω_{gc} gives

$$\angle L(j\omega_{gc}) = \frac{\pi}{2} + \arctan(\omega_{gc}T) - \arctan\left(\frac{\omega_{gc}T}{\alpha_1}\right).$$

Finally, the corresponding phase margin is expressed as

$$\mathrm{PM} = \frac{3\pi}{2} + \arctan(\omega_{gc}T) - \arctan\left(\frac{\omega_{gc}T}{\alpha_1}\right). \quad (9)$$

5. IQC AND ROBUSTNESS ANALYSIS

To rigorously certify the stability of the closed-loop system defined in Section 3 and 4, we move beyond local linearizations and employ the framework of IQCs. This allows us to treat the gradient nonlinearity $\nabla f(y)$ as a bounded uncertainty within a feedback loop.

We adopt the formulation presented in Padmanabhan and Seiler (2024), viewing the optimization algorithm as a linear dynamical system connected to a nonlinear feedback block $u = \nabla f(y)$. We assume the objective function is L-smooth and m-strongly convex. Under these conditions, the gradient nonlinearity satisfies the pointwise sector IQC, which bounds the input-output relationship of the nonlinear block.

Specifically, the gradient function satisfies the inequality defined by the matrix M:

$$\begin{bmatrix} y_k \\ u_k \end{bmatrix}^\top M \begin{bmatrix} y_k \\ u_k \end{bmatrix} \geq 0, \quad M = \begin{bmatrix} -2mL & L+m \\ L+m & -2 \end{bmatrix}$$

We define the mapping matrix $\Psi := \begin{bmatrix} C & 0 \\ 0 & I \end{bmatrix}$ such that $\Psi[s_k^\top \; v_k]^\top = [x_k \; v_k]^\top$. The stability of the interconnection is guaranteed via a Dissipativity theory argument. If there exists a positive definite Lyapunov matrix $P = P^\top \succ 0$ and a scalar multiplier $\lambda \geq 0$ satisfying the Linear Matrix Inequality (LMI):

$$\begin{bmatrix} A^\top PA - P + C^\top C & A^\top PB \\ B^\top PA & B^\top PB \end{bmatrix} + \lambda\, \Psi^\top M \Psi \leq 0, \quad (10)$$

then the optimizer converges exponentially to the equilibrium. Furthermore, this framework allows us to quantify the rejection of stochastic gradient noise. If additive white noise with variance σ^2 enters the gradient channel, the steady-state variance of the estimation error (noise amplification γ) is bounded by:

$$\gamma \leq \sup_{\alpha\in\mathcal{A}} \operatorname{tr}\big(B^\top PB\big)^{1/2}.$$

If the additive white noise entering the gradient channel has noise σ^2, then the steady-state variance bound becomes (see Padmanabhan and Seiler (2024)):

$$\gamma^2 \leq \sigma^2 \sup_{\alpha\in\mathcal{A}} \operatorname{tr}(B^\top PB).$$

6. $\mathcal{H}_2/\mathcal{H}_\infty$ TUNING

Having established the analysis framework, we now turn to the synthesis problem: finding the optimal hyperparameters for PIDAO. Standard manual tuning is often heuristic; here, we propose a systematic design procedure.

6.1 Setup

We formulate the hyperparameter selection as a multi-objective robust control problem. The goal is two-fold: minimize the noise variance (an $\mathcal{H}_2$ performance objective) while guaranteeing stability against worst-case curvature uncertainty (an $\mathcal{H}_\infty$ robustness constraint).

For the $\mathcal{H}_2$ performance, we aim to minimize the steady-state error variance γ_2^2 subject to the Lyapunov stability constraint derived in (10). This is expressed as:

$$\min \gamma_2^2 \quad s.t. \quad \gamma_2^2 \geq \operatorname{tr}(B^\top PB)\sigma^2 \quad (11)$$

Simultaneously, we impose an $\mathcal{H}_\infty$ constraint to ensure robustness against worst-case disturbances and unmodeled curvature. This is enforced via the Bounded Real Lemma, requiring the $\mathcal{H}_\infty$ norm γ_∞ to satisfy the following LMI:

$$\begin{pmatrix} A^\top PA - P + C^\top C & A^\top PB & C^\top D \\ B^\top PA & B^\top PB - \gamma^2 I & D^\top D \\ D^\top C & D^\top B & D^\top D - \gamma^2 I \end{pmatrix} \leq 0 \quad (12)$$

6.2 Implementation and Tuning

To navigate the trade-off between convergence speed (related to bandwidth) and noise rejection (related to $\mathcal{H}_2$ norm), we construct a weighted objective function:

$$\min_{P,\lambda,\gamma_2^2,\gamma_\infty^2} \left\{\mu \cdot \gamma_\infty^2 + (1-\mu)\cdot\gamma_2^2\right\}, \quad (13)$$

where the scalar $\mu \in [0, 1]$ allows the user to prioritize robustness or noise rejection. Solving for the optimal gains (k_p, k_i, k_d, a) presents a computational challenge: the system matrices A and B contain the decision variables, and (12) involves products of the Lyapunov matrix P and these gains. This results in a non-convex Bilinear Matrix

Inequality (BMI). To solve this, we employ a coordinate descent approach combined with a grid search over the PIDAO parameter space. At every grid point, the problem reduces to a convex Semi-Definite Program (SDP) in P and λ, which can be solved efficiently.

6.3 Robustness vs Convergence Tradeoff

The proposed tuning reveals fundamental behaviors of the algorithms.

PIDAO: Analyzing (6) for small learning rates h, we find that the crossover frequency roughly scales as $\omega_{gc} \propto h$.

Assuming we treat $\omega_{gc} = Ch$, then the phase margin expression will always have a positive phase angle when the argument within the arctan, say x is positive. For small h, we can approximate x as

$$x \approx \frac{-CATh^3}{-h^3 k_i + C^2 T^2 k_d h^3} = \frac{-CAT}{-k_i + C^2 T^2 k_d}.$$

We see for specific gain conditions, the phase margin remains positive and stable regardless of h. This suggests that PIDAO does not suffer from a severe trade-off; it maintains robustness even as convergence speed increases.

Adam: In contrast, the linearized Adam dynamics show a distinct penalty. From derivatives of (8) and (9), we observe that increasing the learning rate η increases the crossover frequency (speed):

$$\omega_{gc} = \frac{K}{T\,\alpha_1} + \mathcal{O}(K^2).$$

If we take $\omega_{gc} T = \dfrac{K}{\alpha_1} + \mathcal{O}(K^2)$ and plug-in to the phase margin expression, we get:

$$\mathrm{PM} = \frac{3\pi}{2} + \left(\frac{1}{\alpha_1} - \frac{1}{\alpha_1^2}\right) K + \mathcal{O}(K^2),$$

exhibiting linear decrease in the phase margin (stability) with the increasing learning rate.

7. EXPERIMENTAL VALIDATION

We validate our robust control framework by applying the synthesized gains to three distinct optimization landscapes: quadratic objective, logistic regression, and a Convolutional Neural Network (CNN) on the MNIST dataset. Our goal is to demonstrate that parameters derived from the $\mathcal{H}_2/\mathcal{H}_\infty$ synthesis on a simple quadratic model transfer effectively to complex, non-convex deep learning tasks.

7.1 Optimized Parameters and Frequency Analysis

Using the optimization setup in (13), we solved the BMIs via a grid search in MATLAB to find optimal parameters that minimize steady-state variance ($\mathcal{H}_2$) subject to worst-case robustness constraints ($\mathcal{H}_\infty$). We assumed a standard quadratic objective with Hessian $H = 1$ and additive Gaussian noise ($\mu = 0, \sigma = 3$).

PIDAO Synthesis: The solver yielded the following gains: $a = 50.0$, $k_i = 0.5$, $k_p = 10.0$, and $k_d = 2.5$. The corresponding learning rate was found to be $h = 0.001$.

Adam Linearization: For the linearized Adam model, the optimal parameters were $\eta = 0.0005$, $\beta_1 = 0.75$ and pre-conditioner $D = 0.01$.

7.2 Performance on Quadratic and Logistic Objective

We first applied the tuned algorithms to a scalar quadratic loss $f(x) = \frac{1}{2}x^2$. The tuned PIDAO exhibited faster convergence compared to Adam in the noise-free regime. When significant gradient noise was introduced, Adam suppressed the noise effectively but at the cost of extremely slow convergence, attributed to its heavy low-pass filtering dynamics, see Fig. 2.

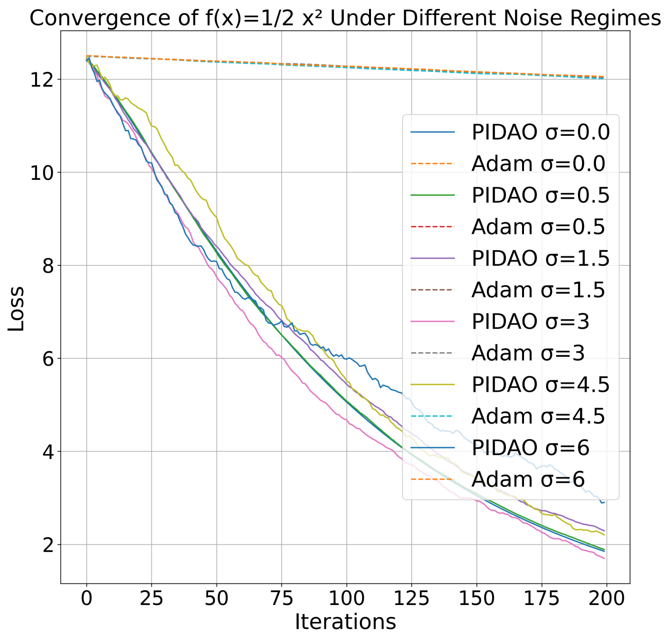

Fig. 2. Training on optimised parameters for noise values $\sigma = \{0.0, 0.5, 1.5, 3.0, 4.6, 6.0\}$ ran for 200 iterations

We trained a single-layer binary logistic regression model. Consistent with the frequency domain analysis, Adam outperformed PIDAO only in the zero-added-noise regime. However, in the presence of stochastic gradient noise, the robustly tuned PIDAO achieved lower loss values over 30 epochs, demonstrating better noise attenuation capabilities than Adam. Shown in Fig. 3.

7.3 Deep Learning Validation (CNN on MNIST)

To test the generalization of our linear control theory to non-convex landscapes, we trained a CNN on the MNIST dataset using the exact gains derived in Section 8.1. We compared the robustly tuned PIDAO against Adam (initialized with PyTorch defaults, $\eta = 0.001$), see Fig. 4.

The results mirrored the logistic regression experiments. While Adam showed a slight advantage in the initial low-noise epochs, PIDAO demonstrated greater stability and lower final loss variance as training progressed. Notably, PIDAO suppressed stochastic gradient noise more effectively than Adam, validating our theoretical claim that optimizing the $\mathcal{H}_2$ norm in the design phase leads to tangible improvements in generalization and stability during neural network training. This confirms that gains

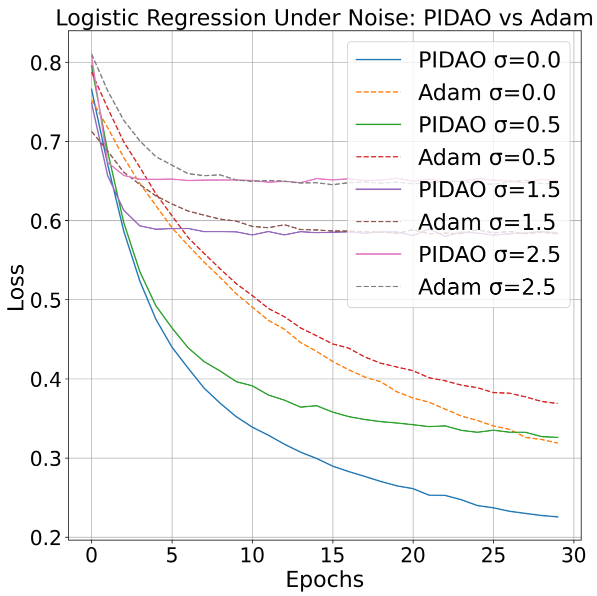

Fig. 3. Logistic Loss - PIDAO vs ADAM

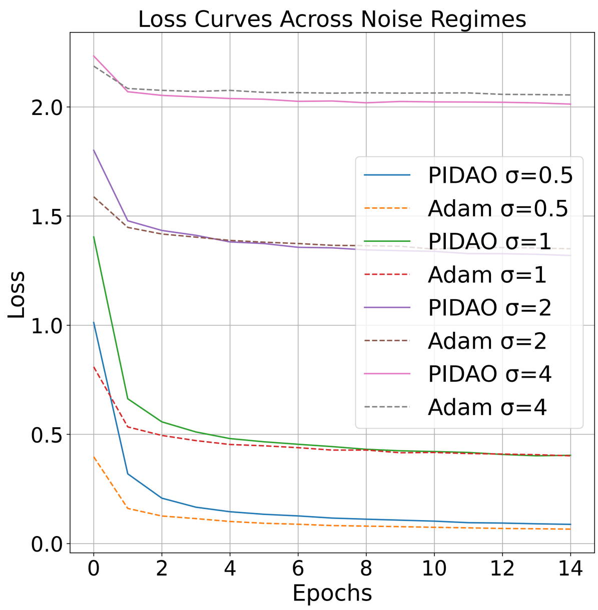

Fig. 4. MNIST Training Loss - PIDAO vs ADAM

synthesized via a local quadratic approximation are robust enough to navigate the global non-convex landscape of deep neural networks.

8. CONCLUSION

This paper proposed a robust $\mathcal{H}_2/\mathcal{H}_\infty$ synthesis framework for tuning PID-based optimization algorithms, guaranteeing stability under curvature uncertainty and stochastic noise. We introduced a novel linearization of the Adam optimizer, enabling a direct frequency-domain comparison of the two methods. Our theoretical analysis reveals that Adam exhibits an inherent trade-off between bandwidth and stability margins, a limitation that the proposed PIDAO design successfully overcomes. Empirical validation on the MNIST dataset confirms that the control-theoretic tuning yields an optimizer that is both faster and more robust to gradient noise than standard adaptive methods. These results highlight the potential of robust control synthesis as a powerful tool for automating the design of deep learning optimizers.

REFERENCES

An, W., Wang, H., Sun, Q., Xu, J., Dai, Q., and Zhang, L. (2018). A pid controller approach for stochastic optimization of deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 8522–8531.

Anagnostides, I. and Panageas, I. (2022). Frequency-domain representation of first-order methods: A simple and robust framework of analysis. In *Symposium on Simplicity in Algorithms (SOSA)*, 131–160. SIAM.

Chen, S., Liu, J., Wang, P., Xu, C., Cai, S., and Chu, J. (2024). Accelerated optimization in deep learning with a proportional-integral-derivative controller. *Nature Communications*, 15(1), 10263.

Cyrus, S., Hu, B., Van Scoy, B., and Lessard, L. (2018). A robust accelerated optimization algorithm for strongly convex functions. In *2018 Annual American Control Conference (ACC)*, 1376–1381. IEEE.

Hu, B. and Lessard, L. (2017). Dissipativity theory for nesterov's accelerated method. In *International Conference on Machine Learning*, 1549–1557. PMLR.

Kingma, D.P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

Lessard, L., Recht, B., and Packard, A. (2016). Analysis and design of optimization algorithms via integral quadratic constraints. *SIAM Journal on Optimization*, 26(1), 57–95.

Nesterov, Y. (1983). A method for solving the convex programming problem with convergence rate o (1/k2). In *Dokl akad nauk Sssr*, volume 269, 543.

Padmanabhan, R. and Seiler, P. (2024). Analysis of gradient descent with varying step sizes using integral quadratic constraints. *IEEE Transactions on Automatic Control*.

Sashank, J.R., Satyen, K., and Sanjiv, K. (2018). On the convergence of adam and beyond. In *International conference on learning representations*, volume 5, 7.