# DUAL-PID-DRIVEN ADAPTIVE LATENT FACTORIZATION OF TENSORS FOR NILM DATA REPRESENTATION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Non-intrusive load monitoring (NILM) plays an important role in intelligent energy management systems. However, due to device failures, communication interruptions, and sensor instability, NILM data are often highly incomplete, making it difficult to obtain reliable load representations for downstream tasks. Latent factorization of tensors (LFT) provides an effective solution for NILM data representation by preserving intrinsic multidimensional structures and capturing temporal correlations. However, existing stochastic gradient descent (SGD)-based LFT models often suffer from limited optimization efficiency and representation accuracy. To address these challenges, this paper proposes a dual-proportional-integral-derivative (PID)-driven adaptive latent factorization of tensors (DPAL) model. Its main innovations include: (i) a nonlinear PID controller-driven training error regulation mechanism that exploits historical error information to improve optimization efficiency and representation accuracy; (ii) an incremental PID controller-integrated fine-grained learning rate adjustment strategy that enhances convergence speed and optimization stability; and (iii) a fuzzy logic-based adaptive gain tuning mechanism for PID controllers that improves model practicality and adaptability. Moreover, we provide a conditional coordinate-wise convergence and stability analysis under explicit smoothness and boundedness assumptions. Experimental results on three real-world NILM datasets demonstrate that the proposed DPAL model consistently outperforms state-of-the-art methods in terms of representation accuracy and computational efficiency.

**Keywords.** Non-Intrusive Load Monitoring, Latent Factorization of Tensors, Adaptive Optimization, PID Controller, Representation Learning.

## 1 INTRODUCTION

Non-intrusive load monitoring (NILM) has become an important technology for smart grid energy management and power-system digitalization (Schirmer & Mporas, 2023; Lu et al., 2026). By leveraging aggregate electrical measurements, such as voltage, current, and power, NILM can estimate appliance-level energy consumption without intrusive sensing infrastructures (Li et al., 2025; Wu & Jia, 2026). Continuous monitoring over long time horizons produces high-dimensional data with complex temporal correlations, making effective representation learning essential for downstream applications such as load identification, anomaly detection, and device monitoring. In practice, sensor failures, communication interruptions, device malfunctions, and environmental disturbances frequently introduce missing observations, resulting in high-dimensional and incomplete (HDI) NILM data. Such incompleteness degrades data integrity and obscures intrinsic correlations among variables, making robust representation learning from HDI NILM data particularly challenging.

As illustrated in Fig. 1, NILM data can be naturally organized as a third-order "measurement-variable $\times$ day $\times$ sampling-time" HDI tensor. This representation preserves multidimensional structure and temporal dependencies that are difficult to capture with conventional vector- or matrix-based models. Recent work on incomplete multidimensional and time-series data has explored generalized nonconvex low-rank tensor learning (Xia et al., 2026), fast low-rank tensor completion (Ghadiri et al., 2025), diffusion-based time-series imputation (Xu et al., 2026), uncertainty-aware imputation (Lim et al., 2026), and explicit temporal modeling under missing observations (Wang et al.,



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

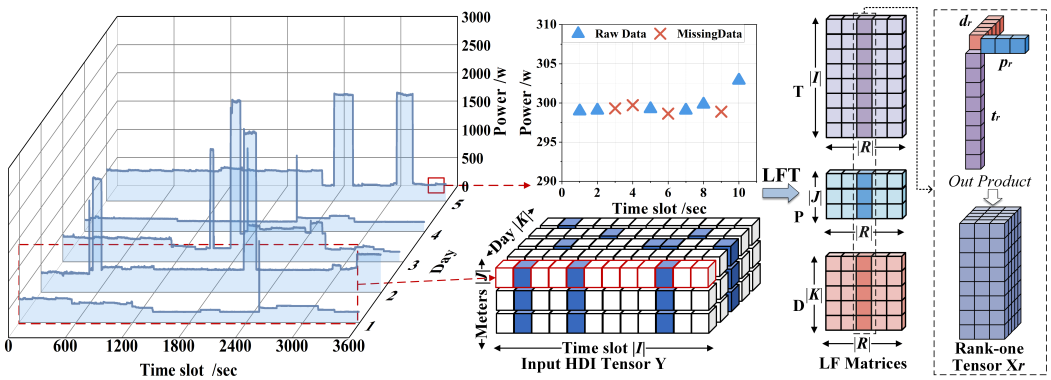



Figure 1: Illustration of HDI NILM tensor construction and its latent factorization process.

2025; Peng et al., 2025). Among these approaches, latent factorization of tensors (LFT) is particularly attractive because of its scalability and ability to learn compact latent representations directly from incomplete tensors.

Despite their effectiveness, many LFT models rely on stochastic gradient descent (SGD) or its variants for optimization. Conventional SGD primarily exploits the current stochastic gradient and does not explicitly account for the history or variation trend of sample-wise reconstruction errors. This limitation can produce less informative update signals and reduce the representation accuracy of SGD-based LFT models. SGD performance is also sensitive to the learning rate. An overly small value slows convergence, whereas an overly large value may cause unstable or oscillatory updates. Recent studies have therefore investigated adaptive optimization and learning-rate regulation to improve convergence behavior (Liang & Xu, 2026; Tang et al., 2026; Oikonomou & Loizou, 2025). Feedback control provides a complementary perspective for regulating iterative optimization (Zhang et al., 2026). In particular, proportional-integral-derivative (PID) control can combine the current error, accumulated historical error, and short-term error variation into a unified feedback signal. This motivates introducing PID-based feedback into SGD-based LFT to regulate both the optimization signal and learning rate according to the evolving training state.

Motivated by the above observations, this paper proposes a dual-PID-driven adaptive latent factorization of tensors (DPAL) model for HDI NILM data representation. DPAL introduces two complementary feedback loops into the LFT optimization process. First, a nonlinear PID controller refines the instance-level reconstruction error by exploiting current, historical, and trend information. Second, an incremental PID controller adaptively adjusts the instance-specific learning rate according to the evolving error signal. To reduce the dependence on manually selected controller parameters, a fuzzy logic-based mechanism further tunes the controller gains online. The main contributions are summarized as follows:

1) We propose DPAL for HDI NILM data representation, which establishes a dual-feedback optimization framework combining nonlinear PID-based error regulation with incremental PID-based fine-grained learning-rate adjustment.
2) We develop a fuzzy logic-based adaptive gain-tuning mechanism to dynamically adjust the PID controller parameters, improving the adaptability and practicality of the optimization process.
3) We provide a conditional coordinate-wise convergence analysis for the underlying SGD-based LFT updates and establish boundedness of the NPID-regulated coordinate gradients under explicit assumptions.

Extensive experiments on three real-world NILM datasets demonstrate that DPAL consistently achieves superior representation accuracy while maintaining competitive convergence and computational efficiency.

# 2 PRELIMINARIES

## 2.1 MODELING NILM DATA INTO AN HDI TENSOR

NILM systems continuously collect electrical measurements, such as current, voltage, and power, over long monitoring periods. The observations within each day are organized into a matrix of size $|I| \times |J|$, where $|I|$ and $|J|$ denote the numbers of sampling points and measurement variables, respectively. By stacking these daily matrices along the monitoring-day dimension, a third-order HDI tensor $\mathbf{Y} \in \mathbb{R}^{|I|\times|J|\times|K|}$ is obtained, where $|K|$ denotes the number of monitoring days. Each entry $y_{ijk}$ represents the $j$-th electrical variable measured at the $i$-th sampling instant on the $k$-th day, while missing observations lead to an incomplete tensor.

As illustrated in Figure 1, this tensor representation preserves both temporal dependencies and cross-variable correlations in NILM data. For example, three electrical variables collected over 28 consecutive days with 86,400 samples per day form a tensor of size $86400 \times 3 \times 28$.

## 2.2 LATENT FACTORIZATION OF TENSORS

Under the third-order tensor canonical polyadic decomposition framework, an LFT model aims to learn the low-dimensional representations, i.e., latent feature (LF) matrices $\mathbf{T}$, $\mathbf{P}$, and $\mathbf{D}$ of HDI tensor $\mathbf{Y} \in \mathbb{R}^{|I|\times|J|\times|K|}$. Accordingly, $R$ rank-one tensors $\mathbf{X}_r$ $(r = 1, \ldots, R)$ can be constructed from $R$ groups of latent feature (LF) vectors, namely $\mathbf{t}_r$, $\mathbf{p}_r$ and $\mathbf{d}_r$. Each rank-one tensor is represented as the outer product of the three LF vectors: $\mathbf{X}_r = \mathbf{t}_r \circ \mathbf{p}_r \circ \mathbf{d}_r$. By summing these $R$ rank-one tensors, a rank-$R$ approximation of the original tensor $\mathbf{Y}$ is obtained, i.e.,

$$\hat{\mathbf{Y}} = \sum_{r=1}^{R} \mathbf{X}_r. \tag{1}$$

Therefore, LFT aims to learn the LF matrices $\mathbf{T}$, $\mathbf{P}$, and $\mathbf{D}$ by minimizing the Euclidean distance between $\mathbf{Y}$ and $\hat{\mathbf{Y}}$. As in standard tensor-completion formulations (Ghadiri et al., 2025; Xia et al., 2026), the loss is defined only on the observed-entry set $\Lambda$. Thus, the objective function is formulated as

$$\begin{aligned}\varepsilon &= \frac{1}{2}\|\mathcal{P}_\Lambda(\mathbf{Y} - \hat{\mathbf{Y}})\|_F^2 = \frac{1}{2}\sum_{y_{ijk}\in\Lambda}(y_{ijk} - \hat{y}_{ijk})^2 \\ &= \frac{1}{2}\sum_{y_{ijk}\in\Lambda}\left(y_{ijk} - \sum_{r=1}^{R} x_{ijk}^r\right)^2 = \frac{1}{2}\sum_{y_{ijk}\in\Lambda}\left(y_{ijk} - \sum_{r=1}^{R} t_{ir}p_{jr}d_{kr}\right)^2.\end{aligned} \tag{2}$$

Here $\mathcal{P}_\Lambda$ keeps the observed entries indexed by $\Lambda$ and sets the others to zero. The LF matrices can then be obtained by optimizing 2 using SGD or its variants, enabling downstream tasks such as missing-data estimation.

## 2.3 PID CONTROLLER

The PID controller is one of the most widely used feedback control algorithms in industrial automation. It generates a control action through a combination of proportional (P), integral (I), and derivative (D) terms based on the system error. In the discrete domain, the control output is given by

$$\tilde{e}^l = K_p e^l + K_i \sum_{h=0}^{l} e^h + K_d(e^l - e^{l-1}), \tag{3}$$

where $e^l$ denotes the system error at iteration $l$, and $K_p$, $K_i$, and $K_d$ represent the proportional, integral, and derivative gains, respectively. $\tilde{e}^l$ denotes the PID-regulated error at iteration $l$.

# 3 A DPAL MODEL

Figure 2 presents the overall architecture of the proposed DPAL model. Developed upon the LFT framework, DPAL introduces a dual PID-driven optimization mechanism consisting of a nonlinear

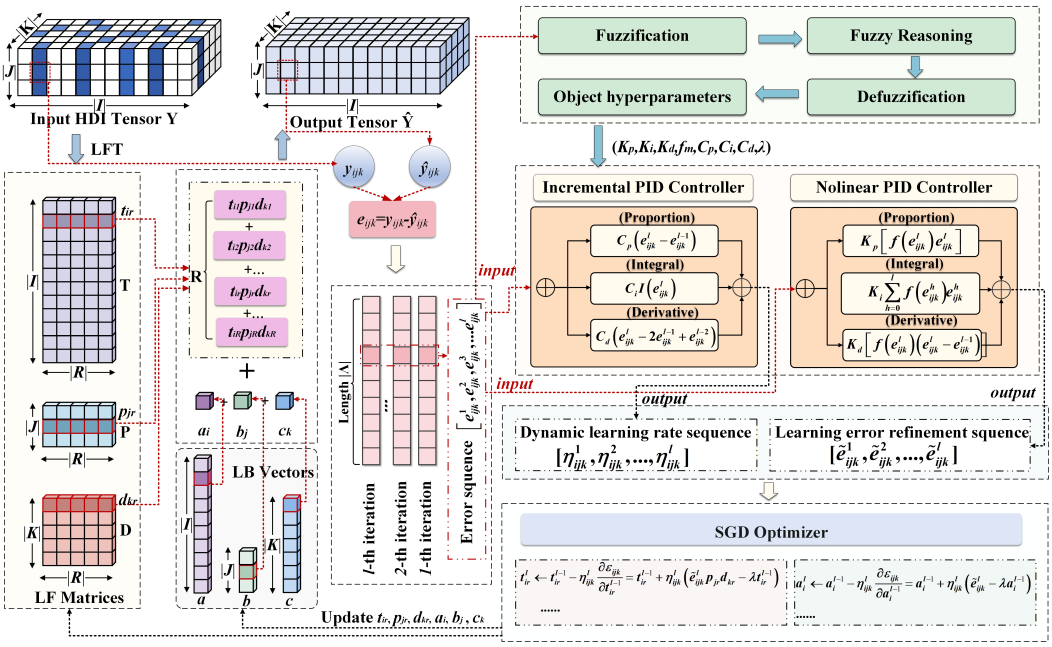



Figure 2: Framework of DPAL.

PID-driven training error regulation loop and an incremental PID-driven fine-grained learning-rate adjustment loop. To further enhance adaptability, a fuzzy logic-based gain tuning mechanism is employed to automatically adjust the controller parameters during training. As a result, DPAL enables efficient, stable, and accurate representation learning for HDI NILM data. The key components of DPAL are described in detail in the following subsections.

### 3.1 Objective Function

DPAL is developed on the LFT framework for HDI NILM data representation. To capture systematic variations across different tensor modes, three linear bias (LB) vectors $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$ with lengths $|I|$, $|J|$, and $|K|$ are incorporated into the model (Dong et al., 2024). Meanwhile, an $L_2$-norm regularization term is introduced to prevent overfitting. For a third-order HDI tensor $\mathbf{Y} \in \mathbb{R}^{|I|\times|J|\times|K|}$, the objective function is formulated as

$$\varepsilon = \frac{1}{2} \sum_{y_{ijk}\in\Lambda} \left[ \left( y_{ijk} - \left( \sum_{r=1}^{R} t_{ir} p_{jr} d_{kr} + a_i + b_j + c_k \right) \right)^2 + \lambda \left( \sum_{r=1}^{R} (t_{ir}^2 + p_{jr}^2 + d_{kr}^2) + a_i^2 + b_j^2 + c_k^2 \right) \right], \tag{4}$$

where $\lambda > 0$ is the regularization coefficient. Following common practice in LFT models, we first use SGD to derive the basic parameter updates, which are then enhanced by the proposed dual-PID mechanisms. For brevity, only the updates of $\mathbf{T}$ and $\mathbf{a}$ are presented, while those of $\mathbf{P}$, $\mathbf{D}$, $\mathbf{b}$, and $\mathbf{c}$ are obtained analogously. At iteration $l$, let $e_{ijk}^l = y_{ijk} - \hat{y}_{ijk}^l$. Then,

$$t_{ir} \leftarrow t_{ir} - \eta \frac{\partial \varepsilon_{ijk}}{\partial t_{ir}} = t_{ir} + \eta(e_{ijk}^l p_{jr} d_{kr} - \lambda t_{ir}), \quad a_i \leftarrow a_i - \eta \frac{\partial \varepsilon_{ijk}}{\partial a_i} = a_i + \eta(e_{ijk}^l - \lambda a_i), \tag{5}$$

where $\eta$ denotes the learning rate.

### 3.2 Nonlinear PID-Driven Error Regulation

As shown in equation 5, the basic SGD updates use only the instantaneous learning error $e_{ijk}^l$ and do not explicitly exploit its history or variation trend. This may produce less informative update

signals, oscillatory optimization, and reduced representation accuracy. To incorporate richer error information, we introduce PID-based feedback to regulate the learning error. Because the error dynamics are generally nonlinear during training, a nonlinear PID (NPID) controller is adopted to adjust the regulation strength according to the current error state.

According to Su et al. (2005), the output of the NPID controller is defined as

$$\tilde{e}^l = K_p F(e^l)e^l + K_i \sum_{h=0}^{l} F(e^h)e^h + K_d F(e^l)(e^l - e^{l-1}), \tag{6}$$

where $F(e^l)$ denotes an error-dependent nonlinear gain function, and $K_p$, $K_i$, and $K_d$ represent the proportional, integral, and derivative gains, respectively. Unlike conventional PID controllers, the nonlinear gain function enables adaptive regulation under different error conditions. A commonly used form of $F(e^l)$ is given by

$$F(e^l) = \cosh(f_m e^l) = \frac{\exp(f_m e^l) + \exp(-f_m e^l)}{2}, \tag{7}$$

where $f_m$ controls the variation rate of the nonlinear gain and $\exp(\cdot)$ denotes the exponential function. Based on the above NPID controller, the learning error $e^l_{ijk}$ associated with the observed entry $y_{ijk}$ is refined as

$$\tilde{e}^l_{ijk} = K_p F(e^l_{ijk})e^l_{ijk} + K_i \sum_{h=0}^{l} F(e^h_{ijk})e^h_{ijk} + K_d F(e^l_{ijk})(e^l_{ijk} - e^{l-1}_{ijk}), \tag{8}$$

where $e^l_{ijk}$ and $\tilde{e}^l_{ijk}$ denote the original and NPID-regulated learning errors at the $l$-th iteration, respectively. From an optimization perspective, the proportional term characterizes the current learning error, the integral term accumulates historical learning errors to suppress oscillatory updates, and the derivative term captures the error variation trend to improve optimization responsiveness. Consequently, $\tilde{e}^l_{ijk}$ provides a more informative optimization signal than the original learning error $e^l_{ijk}$. By replacing $e^l_{ijk}$ in equation 5 with $\tilde{e}^l_{ijk}$, the nonlinear PID-driven error-regulation mechanism is obtained. Accordingly, the parameter updates become

$$t_{ir} \leftarrow t_{ir} + \eta \left(\tilde{e}^l_{ijk} p_{jr} d_{kr} - \lambda t_{ir}\right), \quad a_i \leftarrow a_i + \eta \left(\tilde{e}^l_{ijk} - \lambda a_i\right). \tag{9}$$

### 3.3 Incremental PID-Driven Fine-Grained Learning Rate Adjustment

After replacing $e^l_{ijk}$ with the NPID-regulated error $\tilde{e}^l_{ijk}$, the update direction incorporates richer historical and trend information. However, the learning rate $\eta$ in 9 remains fixed, which may still limit convergence efficiency and cause unstable updates across different optimization states. To further improve adaptivity, we dynamically regulate the learning rate using the evolving training error. Specifically, an incremental PID (IPID) controller adjusts the learning-rate increment rather than the absolute learning rate, allowing the update magnitude to change progressively with the error dynamics while requiring only limited historical information. Following the incremental PID formulation in Wang & Zhang (2024), the control increment $\Delta u^l$ is defined as

$$\Delta u^l = C_p(e^l - e^{l-1}) + C_i e^l + C_d(e^l - 2e^{l-1} + e^{l-2}), \tag{10}$$

where $C_p$, $C_i$, and $C_d$ denote the proportional, integral, and derivative gains, respectively. To avoid integral windup when large training errors occur, an integral separation strategy is incorporated into the controller. Specifically, the integral term is activated only when the absolute error is smaller than a predefined threshold $\sigma$. Consequently, the resulting controller is formulated as

$$\Delta u^l = C_p(e^l - e^{l-1}) + C_i I(e^l) + C_d(e^l - 2e^{l-1} + e^{l-2}), \quad I(e^l) = \begin{cases} e^l, & |e^l| \le \sigma, \\ 0, & |e^l| > \sigma. \end{cases} \tag{11}$$

To realize adaptive learning-rate adjustment, the learning error $e^l_{ijk}$ of each observed entry is used as the feedback input to the controller. Based on the generated control increment, the learning rate is updated as

$$\eta^l_{ijk} = \eta^{l-1}_{ijk} + \Delta u^l_{ijk}, \Delta u^l_{ijk} = C_p(e^l_{ijk} - e^{l-1}_{ijk}) + C_i I(e^l_{ijk}) + C_d(e^l_{ijk} - 2e^{l-1}_{ijk} + e^{l-2}_{ijk}), \tag{12}$$

where $\Delta u_{ijk}^l$ is determined by the incremental PID controller. The proposed strategy dynamically adjusts the learning rate according to the optimization state of each training instance, thereby enabling fine-grained optimization. The two PID loops jointly determine the regulated error $\tilde{e}_{ijk}^l$ and the instance-specific learning rate $\eta_{ijk}^l$. Substituting them into the SGD updates gives

$$t_{ir}^l = t_{ir}^{l-1} + \eta_{ijk}^l \left(\tilde{e}_{ijk}^l p_{jr} d_{kr} - \lambda t_{ir}^{l-1}\right), \quad a_i^l = a_i^{l-1} + \eta_{ijk}^l \left(\tilde{e}_{ijk}^l - \lambda a_i^{l-1}\right). \tag{13}$$

### 3.4 Fuzzy Logic-Based Adaptive Gain Tuning

Although equation 13 combines NPID-regulated errors with IPID-adjusted learning rates, the PID gains and regularization coefficient are still fixed. Since the reconstruction error evolves throughout training, a fixed parameter configuration may limit model adaptability. DPAL therefore introduces a fuzzy logic-based mechanism to dynamically update $\Theta = \{K_p, K_i, K_d, f_m, C_p, C_i, C_d, \lambda\}$ according to the current reconstruction-error magnitude,

$$\Delta_{ijk}^l = |e_{ijk}^l| = |y_{ijk} - \hat{y}_{ijk}^l|. \tag{14}$$

The error domain is represented by five ordered fuzzy reference points $\Delta_1 < \cdots < \Delta_5$. When $\Delta_i \le \Delta_{ijk}^l < \Delta_{i+1}$, only the two adjacent triangular memberships are active:

$$\mu_i = (\Delta_{i+1} - \Delta_{ijk}^l)/(\Delta_{i+1} - \Delta_i), \quad \mu_{i+1} = (\Delta_{ijk}^l - \Delta_i)/(\Delta_{i+1} - \Delta_i), \quad \mu_i + \mu_{i+1} = 1. \tag{15}$$

For any tuned parameter $\theta \in \Theta$, centroid defuzzification with these two active sets reduces to the weighted interpolation

$$\theta^l = \mu_i \theta_i + \mu_{i+1} \theta_{i+1}. \tag{16}$$

The rule design follows the optimization role of the controller components and recent adaptive fuzzy/control formulations (Huang & Zeng, 2025; Zhang et al., 2026). Larger reconstruction-error magnitudes receive stronger proportional action through $K_p$ and $C_p$ to accelerate correction, while integral and derivative actions are kept weaker to reduce accumulated-error amplification and oscillation. As the error magnitude decreases, the integral and derivative gains are increased to improve steady-state accuracy and stability. The regularization coefficient is also increased toward the low-error regime to place relatively more emphasis on generalization after the dominant reconstruction error has been reduced. Thus, fuzzy adaptation changes the controller behavior continuously rather than switching among a few manually tuned settings.

All quantities in $\Theta$ are updated online. Their iteration superscripts are suppressed when this does not cause ambiguity. The complete fuzzy reasoning table, monotonic ordering of the parameter levels, boundary handling, and explicit parameter-wise defuzzification equations are given in Appendix A.1. The full training procedure and complexity analysis are provided in Appendix A.2.

### 3.5 Convergence Analysis for DPAL

We provide a conditional coordinate-wise analysis for a fixed $\lambda > 0$. The full derivation is given in Appendix C. With all other variables fixed, each factor or bias coordinate has positive curvature. For example,

$$\nabla^2 \varepsilon_{ijk}(t_{ir}) = p_{jr}^2 d_{kr}^2 + \lambda \ge \lambda > 0, \quad \nabla^2 \varepsilon_{ijk}(a_i) = 1 + \lambda > 0, \tag{17}$$

which establishes coordinate-wise smoothness and strong convexity, but not joint convexity of the tensor-factorization objective. Let $g_l$ be an unbiased stochastic coordinate gradient with $\mathbb{E}\|g_l\|_2^2 \le \rho^2$. For $\eta_l = c/[\delta(l + \ell_0)]$, $c > 1$, the standard recursion gives

$$\mathbb{E}\|t_{ir}^l - t_{ir}^*\|_2^2 \le (1 - \eta_l \delta)\mathbb{E}\|t_{ir}^{l-1} - t_{ir}^*\|_2^2 + \eta_l^2 \rho^2, \mathbb{E}[\mathcal{F}_{ir}(t_{ir}^l) - \mathcal{F}_{ir}(t_{ir}^*)] \le \frac{LQ(c)}{2(l + \ell_0)} = \mathcal{O}(1/l). \tag{18}$$

For the NPID branch, bounded factor states, controller gains, nonlinear gain, and integral state imply

$$\|\tilde{e}_{ijk}^l\|_2 \le B_{\tilde{e}}, \quad \|\nabla \tilde{\varepsilon}_{ijk}(t_{ir}^{l-1})\|_2 \le \lambda B_t + B_{pd} B_{\tilde{e}} \equiv G_t < \infty. \tag{19}$$

Thus NPID regulation preserves bounded coordinate gradients. The $\mathcal{O}(1/l)$ rate applies to fixed-$\lambda$ coordinate SGD with a positive diminishing effective step size. It does not establish global convergence of the full jointly nonconvex objective or of an unrestricted fuzzy/IPID trajectory.

Table 1: RMSE and MAE on D1–D3. Results are reported as mean ± standard deviation over 20 runs. Lower is better.

| | D1.1 | | D1.2 | | D1.3 | |
|---|---|---|---|---|---|---|
| **Model** | **RMSE** | **MAE** | **RMSE** | **MAE** | **RMSE** | **MAE** |
| M1 | **0.0256 ± 8.0E−05** | **0.0181 ± 4.0E−05** | **0.0258 ± 3.0E−05** | **0.0183 ± 5.0E−05** | **0.0263 ± 5.0E−05** | **0.0187 ± 6.0E−05** |
| M2 | 0.0375 ± 6.0E−04 | 0.0258 ± 7.0E−04 | 0.0591 ± 1.9E−03 | 0.0302 ± 2.4E−03 | 0.0954 ± 7.0E−04 | 0.0407 ± 1.4E−03 |
| M3 | 0.0268 ± 1.0E−04 | 0.0200 ± 6.0E−05 | 0.0275 ± 3.0E−04 | 0.0199 ± 5.0E−05 | 0.0312 ± 3.0E−04 | 0.0209 ± 5.0E−05 |
| M4 | 0.0342 ± 5.0E−04 | 0.0227 ± 5.0E−04 | 0.0533 ± 2.0E−04 | 0.0250 ± 4.0E−04 | 0.0869 ± 6.0E−04 | 0.0313 ± 6.0E−04 |
| M5 | 0.0276 ± 3.0E−04 | 0.0195 ± 3.6E−03 | 0.0356 ± 1.0E−03 | 0.0204 ± 3.0E−04 | 0.0504 ± 3.4E−03 | 0.0238 ± 7.0E−04 |
| M6 | 0.0375 ± 6.0E−04 | 0.0305 ± 8.0E−04 | 0.0434 ± 5.0E−04 | 0.0351 ± 4.0E−04 | 0.0601 ± 5.0E−04 | 0.0483 ± 4.0E−04 |
| M7 | 0.0273 ± 2.0E−05 | 0.0200 ± 3.0E−05 | 0.0283 ± 4.0E−05 | 0.0208 ± 5.0E−05 | 0.0301 ± 6.0E−05 | 0.0223 ± 9.0E−05 |
| M8 | 0.0275 ± 1.0E−04 | 0.0199 ± 6.0E−05 | 0.0328 ± 5.0E−04 | 0.0213 ± 8.0E−05 | 0.0460 ± 9.0E−04 | 0.0242 ± 2.0E−04 |
| | **D2.1** | | **D2.2** | | **D2.3** | |
| **Model** | **RMSE** | **MAE** | **RMSE** | **MAE** | **RMSE** | **MAE** |
| M1 | **0.0244 ± 7.0E−06** | **0.0173 ± 3.0E−06** | **0.0246 ± 1.0E−05** | **0.0177 ± 1.0E−05** | **0.0247 ± 6.0E−05** | **0.0180 ± 7.0E−06** |
| M2 | 0.0373 ± 9.0E−04 | 0.0255 ± 8.0E−04 | 0.0545 ± 1.2E−03 | 0.0284 ± 1.3E−03 | 0.1006 ± 3.4E−03 | 0.0478 ± 4.3E−03 |
| M3 | 0.0252 ± 8.0E−05 | 0.0184 ± 4.0E−05 | 0.0264 ± 1.0E−04 | 0.0190 ± 4.0E−05 | 0.0303 ± 3.0E−04 | 0.0205 ± 8.0E−05 |
| M4 | 0.0333 ± 5.0E−04 | 0.0211 ± 5.0E−04 | 0.0488 ± 5.0E−04 | 0.0234 ± 5.0E−04 | 0.0871 ± 5.0E−04 | 0.0306 ± 6.0E−04 |
| M5 | 0.0266 ± 3.0E−04 | 0.0183 ± 2.0E−04 | 0.0336 ± 1.3E−03 | 0.0194 ± 2.0E−04 | 0.0529 ± 2.2E−03 | 0.0233 ± 5.0E−04 |
| M6 | 0.0368 ± 3.0E−04 | 0.0288 ± 5.0E−04 | 0.0438 ± 7.0E−04 | 0.0347 ± 5.0E−04 | 0.0618 ± 9.0E−04 | 0.0498 ± 5.0E−04 |
| M7 | 0.0263 ± 3.0E−05 | 0.0189 ± 5.0E−05 | 0.0273 ± 4.0E−05 | 0.0197 ± 3.0E−05 | 0.0287 ± 5.0E−05 | 0.0215 ± 6.0E−05 |
| M8 | 0.0279 ± 7.0E−04 | 0.0195 ± 2.0E−04 | 0.0332 ± 7.0E−04 | 0.0211 ± 4.0E−04 | 0.0483 ± 4.0E−04 | 0.0253 ± 6.0E−04 |
| | **D3.1** | | **D3.2** | | **D3.3** | |
| **Model** | **RMSE** | **MAE** | **RMSE** | **MAE** | **RMSE** | **MAE** |
| M1 | **0.0316 ± 7.0E−05** | **0.0247 ± 5.0E−05** | **0.0318 ± 8.0E−05** | **0.0248 ± 1.0E−04** | **0.0323 ± 1.0E−04** | **0.0252 ± 2.0E−04** |
| M2 | 0.0961 ± 1.1E−03 | 0.0498 ± 1.4E−03 | 0.1463 ± 1.3E−03 | 0.0669 ± 1.7E−03 | 0.2541 ± 7.8E−03 | 0.1342 ± 8.1E−03 |
| M3 | 0.0394 ± 6.0E−04 | 0.0286 ± 5.0E−05 | 0.0530 ± 1.2E−03 | 0.0338 ± 2.0E−04 | 0.0888 ± 1.5E−03 | 0.0507 ± 4.0E−04 |
| M4 | 0.0822 ± 8.0E−04 | 0.0364 ± 1.2E−03 | 0.1327 ± 1.3E−03 | 0.0554 ± 1.2E−03 | 0.2276 ± 1.1E−03 | 0.0967 ± 2.6E−03 |
| M5 | 0.0512 ± 2.5E−03 | 0.0313 ± 1.4E−03 | 0.0759 ± 3.9E−03 | 0.0388 ± 9.0E−04 | 0.1243 ± 5.9E−03 | 0.0636 ± 2.9E−03 |
| M6 | 0.0425 ± 2.1E−03 | 0.0331 ± 1.9E−03 | 0.0419 ± 2.0E−03 | 0.0334 ± 1.2E−03 | 0.0464 ± 8.0E−04 | 0.0382 ± 6.0E−04 |
| M7 | 0.0336 ± 1.0E−04 | 0.0267 ± 1.0E−04 | 0.0337 ± 2.0E−04 | 0.0267 ± 2.0E−04 | 0.0338 ± 2.0E−04 | 0.0268 ± 2.0E−04 |
| M8 | 0.0429 ± 1.2E−03 | 0.0300 ± 2.0E−04 | 0.0575 ± 1.2E−03 | 0.0354 ± 3.0E−04 | 0.0942 ± 2.5E−03 | 0.0531 ± 8.0E−04 |

*Note:* Best results are shown in bold, and second-best results are underlined.

# 4 Experimental Results and Analysis

## 4.1 Experimental Setup

**Datasets and protocol.** We evaluate DPAL on iAWE (D1) (Batra et al., 2013), UK-DALE (D2) (Kelly & Knottenbelt, 2015), and MORED (D3) (Ahajjam et al., 2020). D1 and D2 form tensors of size $86400 \times 3 \times 28$, while D3 forms a tensor of size $17280 \times 2 \times 22$. Each dataset is evaluated at known-entry densities of 15%, 10%, and 5%, corresponding to setting indices 1, 2, and 3, respectively. The construction and dataset statistics are provided in Appendix B.1.

The observed-entry set $\Lambda$ is randomly divided into training, validation, and test subsets, denoted by $\Lambda_{\mathrm{tr}}$, $\Lambda_{\mathrm{val}}$, and $\Lambda_{\mathrm{te}}$, respectively, at an 8:1:1 ratio. Each experiment is repeated 20 times, and the reported results are averaged over these runs. We use rank $R = 20$ for all factorization methods to maintain a common latent dimensionality. Training stops after at most 300 iterations or when the changes in both RMSE and MAE between successive iterations fall below $10^{-4}$. All models are implemented in Python and run on a 2.1 GHz Intel Core i7-13700 CPU with 32 GB RAM.

**Baselines.** DPAL (M1) is compared with seven representative tensor completion or factorization methods: graph-regularized tensor decomposition (M2) (Che et al., 2024), Cauchy-loss-based tensor factorization (M3) (Ye et al., 2021), gradient-based CP factorization (M4) (Su et al., 2021), transfer-learning-based biased tensor factorization (M5) (Dong et al., 2024), prior-guided tensor completion (M6) (Yuan & Huang, 2024), Tensor-Ring-based biased factorization (M7) (Wang et al., 2023), and Nesterov-accelerated tensor completion (M8) (Kang et al., 2025).

**Metrics.** Recovery accuracy is measured by RMSE and MAE,

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{y_{ijk}\in\Lambda_{\mathrm{te}}}(y_{ijk}-\hat{y}_{ijk})^2}{|\Lambda_{\mathrm{te}}|}}, \qquad \mathrm{MAE} = \frac{\sum_{y_{ijk}\in\Lambda_{\mathrm{te}}}|y_{ijk}-\hat{y}_{ijk}|}{|\Lambda_{\mathrm{te}}|}, \tag{20}$$

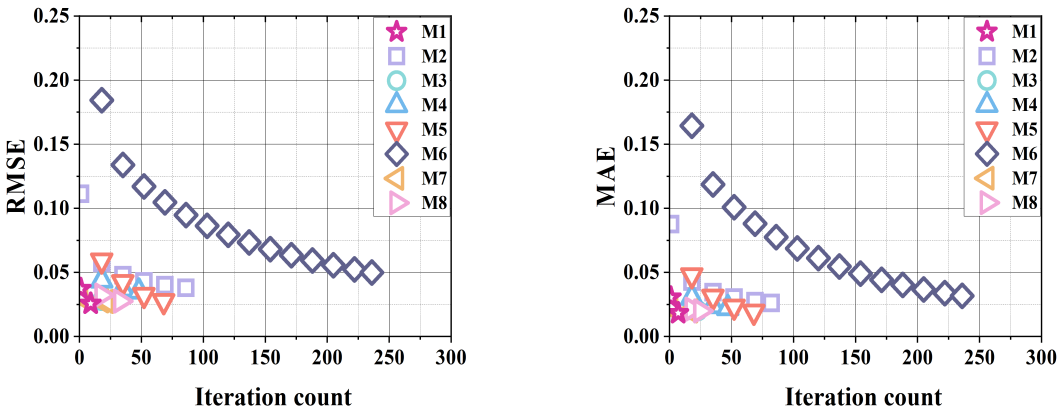



Figure 3: Convergence curves of M1–M8 on D1.1.

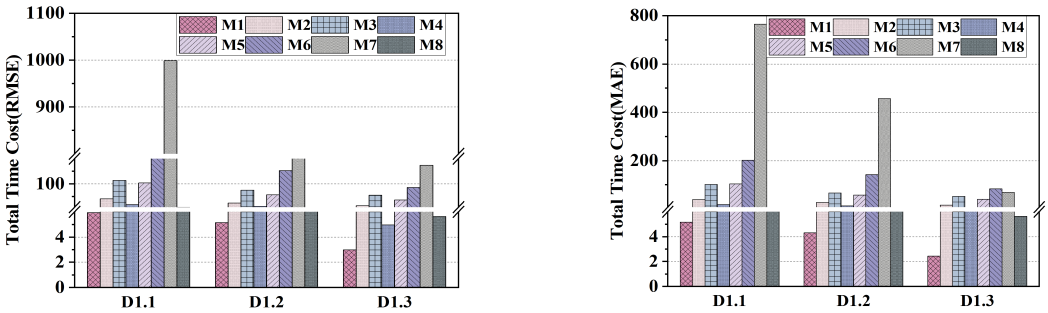



Figure 4: Training time of M1–M8 on D1.

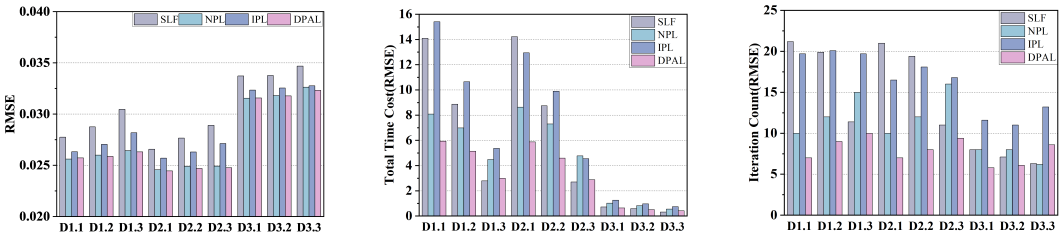
where lower values indicate better reconstruction. Training time and iteration counts are used to assess computational efficiency.

### 4.2 Comparison with State-of-the-Art Models

**Comparison with competing methods.** Table 1 reports the recovery errors across all nine settings. DPAL achieves the lowest reported mean RMSE and MAE in every setting. At 15% known-entry density, DPAL reduces RMSE relative to the strongest competing result by 4.48% on D1.1, 3.17% on D2.1, and 5.95% on D3.1. The corresponding MAE reductions are 7.18%, 5.46%, and 7.49%. These consistent improvements across datasets with different sampling rates and tensor sizes demonstrate the effectiveness of DPAL for NILM data representation.

**Performance under different data densities.** The advantage of DPAL remains evident as the observations become sparser. At 5% known-entry density, M1 still achieves the lowest RMSE and MAE on all three datasets. Compared with the strongest competing results, M1 reduces RMSE and MAE by 12.62% and 10.53% on D1.3, 13.94% and 12.20% on D2.3, and 4.44% and 5.97% on D3.3. These results indicate that DPAL maintains a clear representation-accuracy advantage even under severe data incompleteness.

**Convergence and computational efficiency.** Figures 3 and 4 provide representative convergence and runtime comparisons. On D1.1, DPAL reaches the low-error region in fewer iterations than most competing methods, reflecting the complementary roles of the two feedback loops. The NPID branch refines the instance-level error signal, while the IPID branch adaptively regulates the learning rate according to the current optimization state. Although the controller and fuzzy-inference operations introduce additional per-iteration overhead, DPAL remains computationally competitive while achieving substantially lower reconstruction errors. Additional convergence curves and runtime results are provided in Appendix B.2.




(a) RMSE (b) Training time (c) Iteration count

Figure 5: RMSE-based ablation of the dual-PID framework on D1–D3. Panels (a)–(c) report recovery error, training time, and iteration count, respectively. Lower values are better.

### 4.3 Ablation Studies

**Fuzzy adaptation.** We compare DPAL with DPAL-M, which replaces online fuzzy adaptation with manual grid search for the controller parameters. The complete results in Table 4 (Appendix B.3) show that the two variants have similar reconstruction accuracy, while the reported total cost of DPAL is much smaller because it avoids repeated manual parameter search. For example, on D1.1 both variants obtain an RMSE of 0.0256, whereas the reported RMSE-oriented total cost is 5.94 s for DPAL and 459.56 s for DPAL-M. On D3.3, DPAL obtains an RMSE of 0.0323 compared with 0.0330 for DPAL-M, with reported total costs of 0.36 s and 19.76 s, respectively. Thus, the fuzzy mechanism primarily improves practicality by adapting the controller parameters online without sacrificing recovery quality.

**Dual PID framework.** The controller ablation compares SLF (neither controller), NPL (NPID only), IPL (IPID only), and DPAL (both controllers) under the same initialization and training settings. Figure 5 summarizes the RMSE-based recovery accuracy, training time, and iteration count. NPL generally provides the larger direct reduction in reconstruction error because it refines the instance-level error signal using current, accumulated, and changing error information. IPL regulates a different part of the optimization process by adapting the learning rate, which mainly affects convergence behavior. Combining the two mechanisms yields competitive or lower RMSE in most settings and generally reduces the required iterations, although the additional controller operations can increase the cost of individual updates. The three RMSE views jointly show the accuracy–convergence tradeoff of the dual-PID design. DPAL generally converges in fewer iterations than the single-controller variants, although NPL requires fewer iterations on D3.3. Fewer iterations do not always translate into the lowest runtime because the controller updates add per-iteration overhead. The corresponding MAE-based ablation is reported in Appendix B.4.

## 5 Conclusion

This paper presents DPAL, a dual-PID-driven adaptive latent factorization model for representing HDI NILM data. DPAL combines nonlinear PID-based error regulation, incremental PID-based learning-rate adjustment, and fuzzy logic-based online gain tuning. For the fixed-$\lambda$ coordinate SGD updates with a positive diminishing step size, the analysis in Appendix C establishes a diminishing expected suboptimality bound under smoothness, coordinate-wise strong convexity, and bounded-gradient assumptions. It also shows that the NPID-regulated coordinate gradients remain bounded under the stated boundedness conditions. Across three real-world NILM datasets and nine missing-data settings, DPAL obtains the lowest reported mean RMSE and MAE in every setting, including the 5% known-entry-density cases. Runtime experiments further show that DPAL is computationally competitive, although it is not uniformly the fastest method. The fuzzy and controller ablations indicate that online gain adaptation reduces reliance on manual tuning and that the two PID loops provide complementary optimization effects. Future work will investigate more general adaptive-control mechanisms and less restrictive convergence conditions for incomplete NILM data representation.

## REFERENCES


Mohamed Aymane Ahajjam, Daniel Bonilla Licea, Chaimaa Essayeh, Mounir Ghogho, and Abdellatif Kobbane. MORED: A moroccan buildings' electricity consumption dataset. *Energies*, 13 (24):6737, 2020. doi: 10.3390/en13246737.

Nipun Batra, Manoj Gulati, Amarjeet Singh, and Mani B. Srivastava. It's different: Insights into home energy consumption in india. In *Proceedings of the 5th ACM Workshop on Embedded Systems for Energy-Efficient Buildings*, pp. 1–8, 2013. doi: 10.1145/2528282.2528293.

Hangjun Che, Baicheng Pan, Man-Fai Leung, Yuting Cao, and Zheng Yan. Tensor factorization with sparse and graph regularization for fake news detection on social networks. *IEEE Transactions on Computational Social Systems*, 11(4):4888–4898, 2024. doi: 10.1109/TCSS.2023.3296479.

Jiaying Dong, Yan Song, Ming Li, and Hong Rao. Biased collective latent factorization of tensors with transfer learning for dynamic QoS data predicting. *Digital Signal Processing*, 146:104360, 2024. doi: 10.1016/j.dsp.2023.104360.

Mehrdad Ghadiri, Matthew Fahrbach, Yunbum Kook, and Ali Jadbabaie. Fast tensor completion via approximate richardson iteration. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 19248–19265. PMLR, 2025.

Haoen Huang and Zhigang Zeng. A fuzzy adaptive network with learnable parameters for mixed-integer optimization. *IEEE Transactions on Fuzzy Systems*, 33(10):3823–3834, 2025. doi: 10.1109/TFUZZ.2025.3604526.

Yifan Kang, Mengyuan Zhang, and Kai Liu. Acceleration in low-rank tensor completion. In *Proceedings of the 2025 SIAM International Conference on Data Mining*, pp. 31–40, 2025. doi: 10.1137/1.9781611978520.4.

Jack Kelly and William Knottenbelt. The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes. *Scientific Data*, 2:150007, 2015. doi: 10.1038/sdata.2015.7.

Shijie Li, Zijun Su, Guanlin Chen, Lulu Chen, Haoqin Li, Huaiguang Jiang, Jun Zhang, and David Wenzhong Gao. Transferable nonintrusive load monitoring in smart grids via frequency-division fusion scattering time-series transformer. *IEEE Transactions on Industrial Informatics*, 21(10):7531–7541, 2025. doi: 10.1109/TII.2025.3576846.

Yuqing Liang and Dongpo Xu. A unified theoretical framework for the last-iterate convergence of stochastic adaptive optimization. *Artificial Intelligence*, 353:104504, 2026. doi: 10.1016/j.artint.2026.104504.

Jaesung Lim, Seunghwan An, and Jong-June Jeon. Impute missing entries with uncertainty. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 23514–23522, 2026. doi: 10.1609/aaai.v40i28.39523.

Taoyu Lu, Yehui Li, Ruiyang Yao, and Yi Wang. Unsupervised lightweight transfer learning for edge non-intrusive load monitoring. *IEEE Transactions on Smart Grid*, 2026. doi: 10.1109/TSG.2026.3699466. Early Access.

Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009. doi: 10.1137/070704277.

Dimitris Oikonomou and Nicolas Loizou. Stochastic polyak step-sizes and momentum: Convergence guarantees and practical performance. In *International Conference on Learning Representations*, 2025.

Jing Peng, Meiqi Yang, Qiong Zhang, and Xiaoxiao Li. S4M: S4 for multivariate time series forecasting with missing values. In *International Conference on Learning Representations*, 2025.

Pascal A. Schirmer and Iosif Mporas. Non-intrusive load monitoring: A review. *IEEE Transactions on Smart Grid*, 14(1):769–784, 2023. doi: 10.1109/TSG.2022.3189598.

Xing Su, Minjie Zhang, Yi Liang, Zhi Cai, Limin Guo, and Zhiming Ding. A tensor-based approach for the QoS evaluation in service-oriented environments. *IEEE Transactions on Network and Service Management*, 18(3):3843–3857, 2021. doi: 10.1109/TNSM.2021.3074547.

Yuxin Su, Dong Sun, and Baoyan Duan. Design of an enhanced nonlinear PID controller. *Mechatronics*, 15(8):1005–1024, 2005. doi: 10.1016/j.mechatronics.2005.03.003.

Xuan Tang, Jichu Li, and Difan Zou. A convergence analysis of adaptive optimizers under floating-point quantization. In *International Conference on Learning Representations*, 2026.

Hao Wang, Zhengnan Li, Haoxuan Li, Xu Chen, Mingming Gong, Bin Chen, and Zhichao Chen. Optimal transport for time series imputation. In *International Conference on Learning Representations*, 2025.

Wei Wang, Guoqiang Sun, Siwen Zhao, Yujun Li, and Jianli Zhao. Tensor ring decomposition for context-aware recommendation. *Expert Systems with Applications*, 217:119533, 2023. doi: 10.1016/j.eswa.2023.119533.

Zenghui Wang and Jun Zhang. Incremental PID controller-based learning rate scheduler for stochastic gradient descent. *IEEE Transactions on Neural Networks and Learning Systems*, 35(5):7060–7071, 2024. doi: 10.1109/TNNLS.2022.3213677.

Haotian Wu and Qingquan Jia. DU-NILM: Sequence-to-segmental subsequence learning with improved DoubleU-Net for nonintrusive load monitoring. *IEEE Transactions on Industrial Informatics*, 22(4):3169–3179, 2026. doi: 10.1109/TII.2025.3650026.

Sijia Xia, Michael K. Ng, and Xiongjun Zhang. Low-rank tensor learning by generalized nonconvex regularization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(5):5604–5619, 2026. doi: 10.1109/TPAMI.2025.3650712.

Changjian Xu, Yong Wang, Ruizheng Huang, Zhicheng Zhang, Wen Yin, and Kexin Li. CaT-Diff: Cascaded text-enhanced diffusion model for time-series imputation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 27215–27223, 2026. doi: 10.1609/aaai.v40i32.39937.

Fanghua Ye, Zhiwei Lin, Chuan Chen, Zibin Zheng, and Hong Huang. Outlier-resilient web service QoS prediction. In *Proceedings of The Web Conference 2021*, pp. 3099–3110, 2021. doi: 10.1145/3442381.3449938.

Shiran Yuan and Kaizhu Huang. A generalizable framework for low-rank tensor completion with numerical priors. *Pattern Recognition*, 155:110678, 2024. doi: 10.1016/j.patcog.2024.110678.

Pengfei Zhang, Xinde Li, Le Yu, Zhentong Zhang, Fir Dunkin, Huaping Liu, and Zhijun Li. Boosting learning efficiency in few-shot tasks with layer-adaptive PID control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(6):7079–7096, 2026. doi: 10.1109/TPAMI.2026.3663608.

# A Implementation Details

## A.1 Additional Fuzzy-Inference Details

This section specifies the fuzzy inference mechanism introduced in Section 3. The input is the current reconstruction-error magnitude defined in equation 14.

The main paper gives the two active triangular memberships and their interpolation form. For completeness, when $\Delta_i \le \Delta_{ijk}^l < \Delta_{i+1}$, the membership degrees associated with the adjacent reference points are

$$\mu_i = \frac{\Delta_{i+1} - \Delta_{ijk}^l}{\Delta_{i+1} - \Delta_i}, \qquad \mu_{i+1} = \frac{\Delta_{ijk}^l - \Delta_i}{\Delta_{i+1} - \Delta_i}. \tag{A1}$$

At the endpoints, the corresponding endpoint membership is set to one and the other memberships to zero. When the training-error magnitude is large, the rule base assigns stronger proportional action and weaker integral and derivative action. As the error magnitude decreases, the integral and derivative gains are increased. The regularization coefficient $\lambda$ follows the opposite trend to the proportional gains so that regularization receives relatively greater weight in the low-error regime. The complete parameter-level rule base is summarized in Table 2.

For example, if $\Delta_1 \le \Delta_{ijk}^l < \Delta_2$, centroid defuzzification reduces to linear interpolation between the two active parameter levels:

$$\begin{cases} \mu_1 = (\Delta_2 - \Delta_{ijk}^l)/(\Delta_2 - \Delta_1), \quad \mu_2 = (\Delta_{ijk}^l - \Delta_1)/(\Delta_2 - \Delta_1), \\ K_p = \mu_1 K_{p1} + \mu_2 K_{p2}, \quad K_i = \mu_1 K_{i1} + \mu_2 K_{i2}, \\ K_d = \mu_1 K_{d1} + \mu_2 K_{d2}, \quad f_m = \mu_1 f_{m1} + \mu_2 f_{m2}, \\ C_p = \mu_1 C_{p1} + \mu_2 C_{p2}, \quad C_i = \mu_1 C_{i1} + \mu_2 C_{i2}, \\ C_d = \mu_1 C_{d1} + \mu_2 C_{d2}, \quad \lambda = \mu_1 \lambda_1 + \mu_2 \lambda_2. \end{cases} \tag{A2}$$

Table 2: Fuzzy reasoning rule base for the controller and regularization parameters.

| Parameter | $\Delta_1$ | $\Delta_2$ | $\Delta_3$ | $\Delta_4$ | $\Delta_5$ |
|---|---|---|---|---|---|
| $K_p$ | $K_{p1}$ | $K_{p2}$ | $K_{p3}$ | $K_{p4}$ | $K_{p5}$ |
| $K_i$ | $K_{i1}$ | $K_{i2}$ | $K_{i3}$ | $K_{i4}$ | $K_{i5}$ |
| $K_d$ | $K_{d1}$ | $K_{d2}$ | $K_{d3}$ | $K_{d4}$ | $K_{d5}$ |
| $f_m$ | $f_{m1}$ | $f_{m2}$ | $f_{m3}$ | $f_{m4}$ | $f_{m5}$ |
| $C_p$ | $C_{p1}$ | $C_{p2}$ | $C_{p3}$ | $C_{p4}$ | $C_{p5}$ |
| $C_i$ | $C_{i1}$ | $C_{i2}$ | $C_{i3}$ | $C_{i4}$ | $C_{i5}$ |
| $C_d$ | $C_{d1}$ | $C_{d2}$ | $C_{d3}$ | $C_{d4}$ | $C_{d5}$ |
| $\lambda$ | $\lambda_1$ | $\lambda_2$ | $\lambda_3$ | $\lambda_4$ | $\lambda_5$ |

The following inequalities hold: $\Delta_1 < \Delta_2 < \Delta_3 < \Delta_4 < \Delta_5$, $K_{p1} < K_{p2} < K_{p3} < K_{p4} < K_{p5}$, $K_{i1} > K_{i2} > K_{i3} > K_{i4} > K_{i5}$, $K_{d1} > K_{d2} > K_{d3} > K_{d4} > K_{d5}$, $f_{m1} > f_{m2} > f_{m3} > f_{m4} > f_{m5}$, $C_{p1} < C_{p2} < C_{p3} < C_{p4} < C_{p5}$, $C_{i1} > C_{i2} > C_{i3} > C_{i4} > C_{i5}$, $C_{d1} > C_{d2} > C_{d3} > C_{d4} > C_{d5}$, $\lambda_1 > \lambda_2 > \lambda_3 > \lambda_4 > \lambda_5$.

Through the above procedure, the gains of both PID controllers are automatically adjusted online according to the training state, enabling adaptive error regulation and learning rate adjustment without manual parameter tuning.

### A.2 Algorithm Design and Complexity

The algorithm of the DPAL model is shown in Algorithm 1.

The computational complexity of DPAL is primarily determined by the iterative optimization of observed tensor entries. Under the common assumption that $|\Lambda| \gg \max\{|I|, |J|, |K|\}$, the time complexity of DPAL is $\Theta(N \times |\Lambda| \times R)$, where $N$ and $R$ denote the number of training iterations and the latent feature dimensionality, respectively. Therefore, when $N$ and $R$ are fixed, the computational cost grows approximately linearly with the number of observed entries. The storage cost mainly consists of the LF matrices, linear bias vectors, observed entries, and auxiliary variables required by the dual PID-driven optimization mechanism, resulting in a space complexity of $\Theta((|I| + |J| + |K|) \times R + |\Lambda|)$. Consequently, DPAL maintains the linear scalability characteristic of SGD-based LFT models and remains suitable for large-scale HDI tensor representation learning.

**Algorithm 1** DPAL

**Require:** Rank $R$, training set $\Lambda_{\mathrm{tr}}$, maximum iterations $N$

Initialize $\mathbf{T}$, $\mathbf{P}$, $\mathbf{D}$, $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$ with non-negative random values
Initialize $\eta_{ijk}^{0}$ and PID state variables, and set $e_{ijk}^{-1} = e_{ijk}^{0} = 0$
**for** $l = 1$ to $N$ **do**
  **for** each $y_{ijk} \in \Lambda_{\mathrm{tr}}$ **do**
    Compute $e_{ijk}^{l} = y_{ijk} - \hat{y}_{ijk}^{l}$
    Update $K_p, K_i, K_d, f_m, C_p, C_i, C_d, \lambda$ using the fuzzy mechanism
    Compute $F(e_{ijk}^{l})$ with equation 7
    Compute $\tilde{e}_{ijk}^{l}$ with equation 8
    **if** $|e_{ijk}^{l}| \leq \sigma$ **then**
      $\Delta u_{ijk}^{l} = C_p(e_{ijk}^{l} - e_{ijk}^{l-1}) + C_i e_{ijk}^{l} + C_d(e_{ijk}^{l} - 2e_{ijk}^{l-1} + e_{ijk}^{l-2})$
    **else**
      $\Delta u_{ijk}^{l} = C_p(e_{ijk}^{l} - e_{ijk}^{l-1}) + C_d(e_{ijk}^{l} - 2e_{ijk}^{l-1} + e_{ijk}^{l-2})$
    **end if**
    $\eta_{ijk}^{l} = \eta_{ijk}^{l-1} + \Delta u_{ijk}^{l}$
    **for** $r = 1$ to $R$ **do**
      Update $t_{ir}^{l}, p_{jr}^{l}, d_{kr}^{l}$ according to equation 13
    **end for**
    Update $a_i^{l}, b_j^{l}, c_k^{l}$ analogously to equation 13
  **end for**
  **if** the stopping criterion is satisfied **then**
    **break**
  **end if**
**end for**

**Ensure:** $\mathbf{T}, \mathbf{P}, \mathbf{D}, \mathbf{a}, \mathbf{b}, \mathbf{c}$

# B Additional Experimental Results

## B.1 Dataset Construction and Statistics

**Datasets.** Experiments are conducted on three real-world NILM datasets: iAWE (D1) (Batra et al., 2013), UK-DALE (D2) (Kelly & Knottenbelt, 2015), and MORED (D3) (Ahajjam et al., 2020). Following the tensor construction procedure in Section 2, the data are organized as third-order HDI tensors with the form of "sampling time×measurement variable×monitoring day." For D1 and D2, 28 consecutive days of current, voltage, and power measurements sampled at 1 Hz are used, resulting in tensors of size $86400 \times 3 \times 28$. For D3, 22 consecutive days of voltage and power measurements sampled every 5 s are used, resulting in a tensor of size $17280 \times 2 \times 22$.

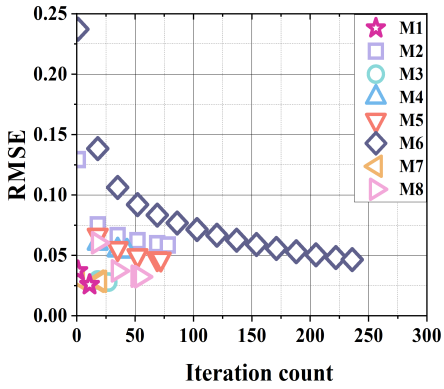
To evaluate robustness under severe data incompleteness, tube-wise missing patterns are introduced at known-entry densities of 15%, 10%, and 5% on each dataset, corresponding to missing rates of 85%, 90%, and 95%, respectively. This setting generates both random missing entries and continuous missing intervals, closely reflecting practical NILM data-loss scenarios. Detailed configurations are summarized in Table 3.

Table 3: Dataset configurations used in the experiments.

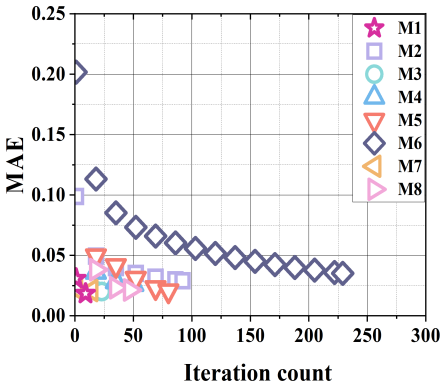
| Dataset | Setting | Tensor size | Known entries | Density |
|---|---|---|---|---|
| D1 | D1.1 | $86400 \times 3 \times 28$ | 1,088,640 | 15% |
| | D1.2 | $86400 \times 3 \times 28$ | 725,760 | 10% |
| | D1.3 | $86400 \times 3 \times 28$ | 362,880 | 5% |
| D2 | D2.1 | $86400 \times 3 \times 28$ | 1,088,640 | 15% |
| | D2.2 | $86400 \times 3 \times 28$ | 725,760 | 10% |
| | D2.3 | $86400 \times 3 \times 28$ | 362,880 | 5% |
| D3 | D3.1 | $17280 \times 2 \times 22$ | 114,048 | 15% |
| | D3.2 | $17280 \times 2 \times 22$ | 76,032 | 10% |
| | D3.3 | $17280 \times 2 \times 22$ | 38,016 | 5% |

## B.2 Training Curves and Runtime Comparison

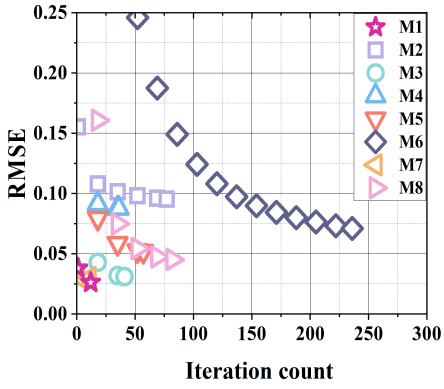
Figures 6–8 provide the remaining RMSE and MAE convergence curves not shown in the main paper. Figure 9 reports the D2 and D3 runtime comparisons, while the representative D1.1 curves and D1 runtime results are shown in Figures 3 and 4. Together, these results supplement the main accuracy comparison in Table 1.



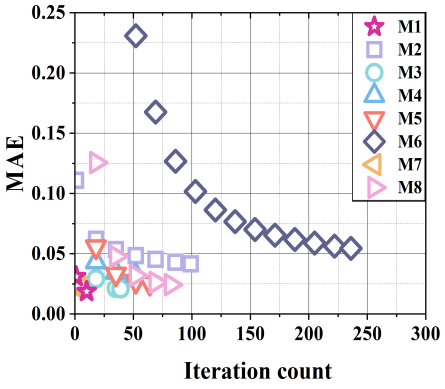

(a) D1.2: RMSE (10%)




(b) D1.2: MAE (10%)

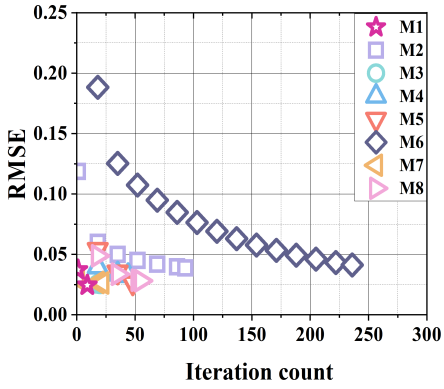



(c) D1.3: RMSE (5%)

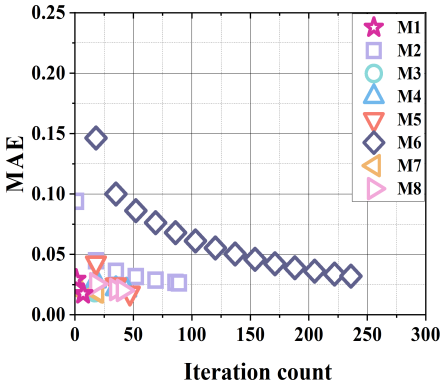



(d) D1.3: MAE (5%)

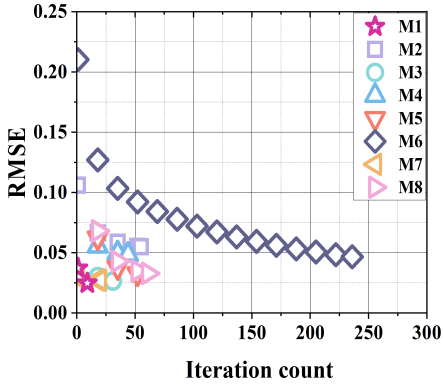
Figure 6: Additional training curves of M1–M8 on D1. The D1.1 curves are shown in the main paper.



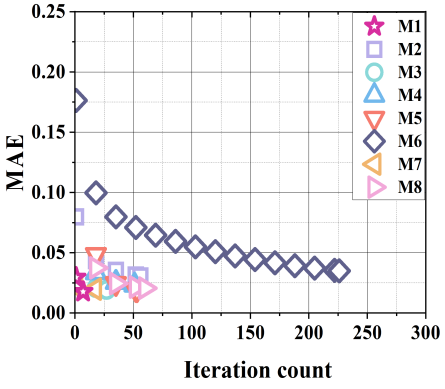

(a) D2.1: RMSE (15%)



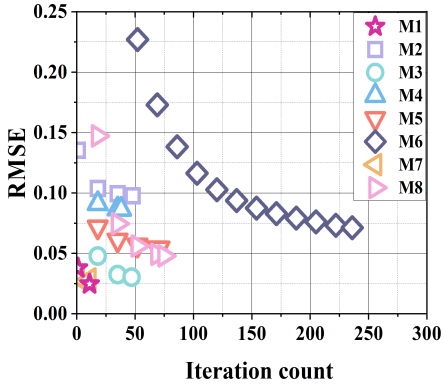

(b) D2.1: MAE (15%)



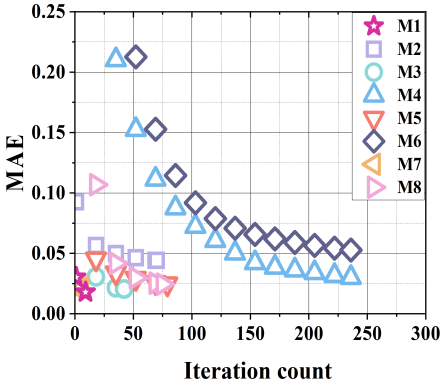

(c) D2.2: RMSE (10%)




(d) D2.2: MAE (10%)

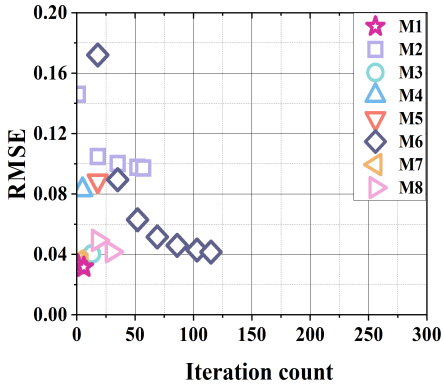



(e) D2.3: RMSE (5%)

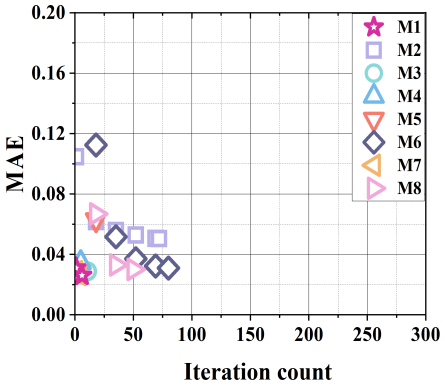



(f) D2.3: MAE (5%)

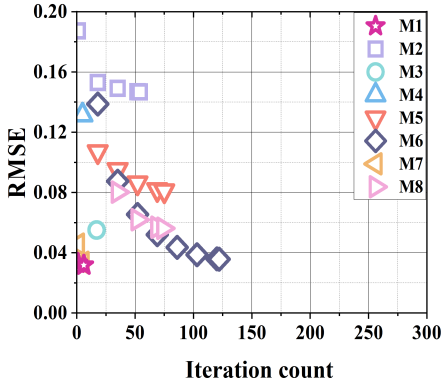
Figure 7: Training curves of M1–M8 on D2.



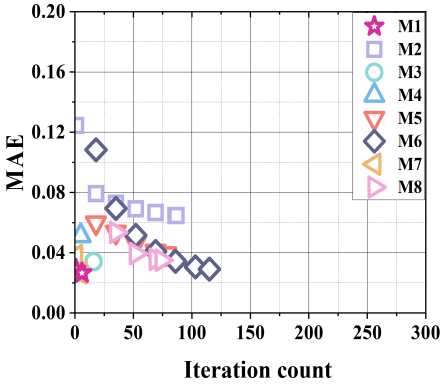

(a) D3.1: RMSE (15%)



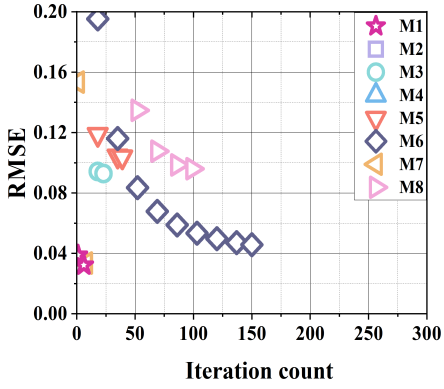

(b) D3.1: MAE (15%)



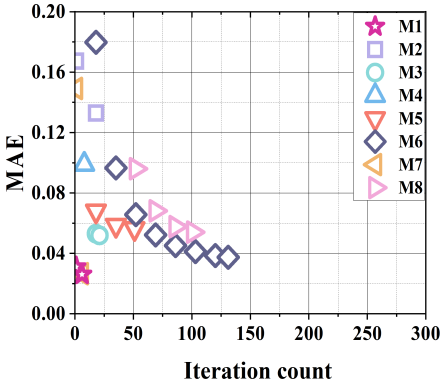

(c) D3.2: RMSE (10%)




(d) D3.2: MAE (10%)

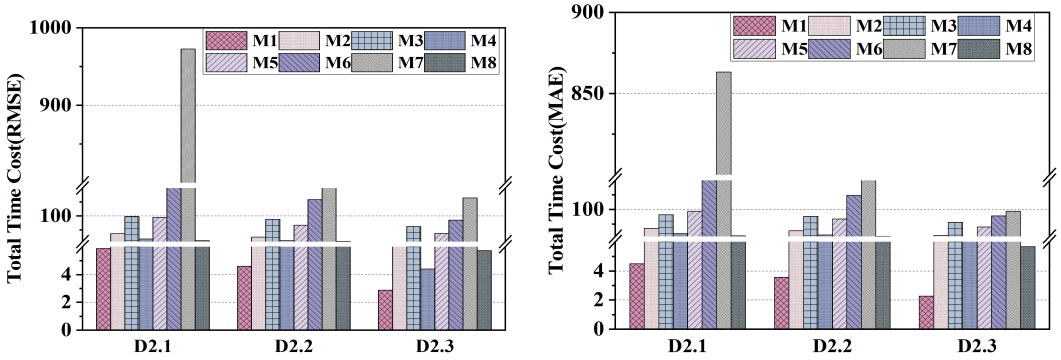



(e) D3.3: RMSE (5%)




(f) D3.3: MAE (5%)

Figure 8: Training curves of M1–M8 on D3.




(a) D2: RMSE (b) D2: MAE

(c) D3: RMSE (d) D3: MAE

Figure 9: Training time (seconds) of M1–M8 on D2 and D3. The D1 runtime results are shown in the main paper.

### B.3 Fuzzy Adaptation Ablation

Table 4 compares online fuzzy adaptation (DPAL) with manual parameter search (DPAL-M) across the nine settings. It includes the recovery errors and reported total times for both metrics.

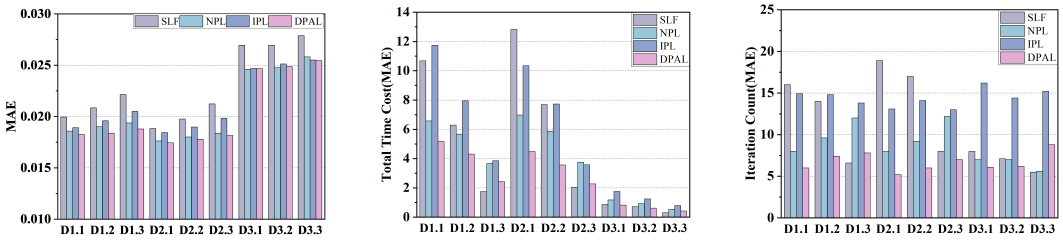
Table 4: Performance of M1 with and without the hyperparameter adaptation mechanism.

| Data | Metric | DPAL | DPAL-M |
|---|---|---|---|
| D1.1 | RMSE | $0.0256 \pm 8\text{E}{-05}$ | $0.0256 \pm 2\text{E}{-05}$ |
| | RMSE Time | $5.94 \pm 0.75$ | $459.56 \pm 39.88$ |
| | MAE | $0.0181 \pm 4\text{E}{-05}$ | $0.0184 \pm 1\text{E}{-05}$ |
| | MAE Time | $5.17 \pm 0.74$ | $371.49 \pm 38.84$ |
| D1.2 | RMSE | $0.0258 \pm 3\text{E}{-05}$ | $0.0259 \pm 2\text{E}{-05}$ |
| | RMSE Time | $5.14 \pm 0.72$ | $413.10 \pm 40.10$ |
| | MAE | $0.0183 \pm 5\text{E}{-05}$ | $0.0186 \pm 5\text{E}{-05}$ |
| | MAE Time | $4.31 \pm 0.65$ | $318.81 \pm 49.04$ |
| D1.3 | RMSE | $0.0263 \pm 5\text{E}{-05}$ | $0.0265 \pm 7\text{E}{-05}$ |
| | RMSE Time | $2.99 \pm 0.73$ | $232.87 \pm 37.14$ |
| | MAE | $0.0187 \pm 6\text{E}{-05}$ | $0.0190 \pm 7\text{E}{-05}$ |
| | MAE Time | $2.43 \pm 0.75$ | $193.21 \pm 37.19$ |
| D2.1 | RMSE | $0.0244 \pm 7\text{E}{-06}$ | $0.0248 \pm 3\text{E}{-05}$ |
| | RMSE Time | $5.88 \pm 0.78$ | $540.99 \pm 52.00$ |
| | MAE | $0.0173 \pm 3\text{E}{-06}$ | $0.0175 \pm 3\text{E}{-05}$ |
| | MAE Time | $4.49 \pm 1.09$ | $394.74 \pm 42.13$ |
| D2.2 | RMSE | $0.0246 \pm 1\text{E}{-05}$ | $0.0250 \pm 4\text{E}{-05}$ |
| | RMSE Time | $4.60 \pm 0.80$ | $412.98 \pm 43.04$ |
| | MAE | $0.0177 \pm 1\text{E}{-05}$ | $0.0177 \pm 3\text{E}{-05}$ |
| | MAE Time | $3.57 \pm 0.80$ | $298.14 \pm 39.99$ |
| D2.3 | RMSE | $0.0247 \pm 6\text{E}{-05}$ | $0.0251 \pm 4\text{E}{-05}$ |
| | RMSE Time | $2.88 \pm 0.74$ | $251.83 \pm 46.94$ |
| | MAE | $0.0180 \pm 7\text{E}{-06}$ | $0.0181 \pm 5\text{E}{-05}$ |
| | MAE Time | $2.27 \pm 0.78$ | $199.15 \pm 43.78$ |
| D3.1 | RMSE | $0.0316 \pm 0.0001$ | $0.0318 \pm 8\text{E}{-05}$ |
| | RMSE Time | $0.52 \pm 0.63$ | $37.99 \pm 2.78$ |
| | MAE | $0.0247 \pm 5\text{E}{-05}$ | $0.0253 \pm 0.0001$ |
| | MAE Time | $0.53 \pm 0.63$ | $32.91 \pm 2.85$ |
| D3.2 | RMSE | $0.0318 \pm 8\text{E}{-05}$ | $0.0323 \pm 0.0001$ |
| | RMSE Time | $0.41 \pm 0.60$ | $28.70 \pm 2.04$ |
| | MAE | $0.0248 \pm 0.0001$ | $0.0256 \pm 0.0001$ |
| | MAE Time | $0.42 \pm 0.60$ | $23.33 \pm 1.19$ |
| D3.3 | RMSE | $0.0323 \pm 0.0001$ | $0.0330 \pm 0.0002$ |
| | RMSE Time | $0.36 \pm 0.60$ | $19.76 \pm 0.65$ |
| | MAE | $0.0252 \pm 0.0002$ | $0.0265 \pm 0.0001$ |
| | MAE Time | $0.36 \pm 0.60$ | $16.66 \pm 1.22$ |

*Note:* Time is measured in seconds. Bold values indicate lower reconstruction error or lower total computational time.

### B.4 MAE Results of the Dual-PID Variants

Figure 10 complements the RMSE-based results in Figure 5 by reporting the corresponding MAE recovery error, training time, and iteration count for SLF, NPL, IPL, and DPAL.




(a) MAE (b) Training time (c) Iteration count

Figure 10: MAE-based ablation of the dual-PID framework on D1–D3. Panels (a)–(c) report recovery error, training time, and iteration count, respectively. Lower values are better.

## C Convergence Analysis of DPAL

This appendix gives a conditional, coordinate-wise convergence analysis for a fixed positive regularization coefficient $\lambda$. Following standard stochastic-approximation arguments (Nemirovski et al., 2009) and recent analyses of stochastic adaptive optimization (Liang & Xu, 2026; Oikonomou & Loizou, 2025), we separate the argument into two parts. First, with all other factor coordinates fixed, we establish smoothness and strong convexity of each scalar coordinate objective and derive a diminishing expected suboptimality bound for its SGD update. Second, we show that nonlinear PID regulation preserves bounded coordinate stochastic gradients when the factor states, controller gains, nonlinear-gain output, and accumulated PID state are bounded. The $\mathcal{O}(1/l)$ rate requires a positive diminishing effective step size. Therefore, an arbitrary IPID-generated learning-rate trajectory is not covered unless it satisfies the same step-size condition. Likewise, the fuzzy update of $\lambda$ is excluded from the fixed-objective rate analysis. The results should be interpreted as conditional coordinate-wise convergence and boundedness guarantees, not as joint global convergence of the full nonconvex DPAL algorithm.

### C.1 Analysis of the Underlying SGD Updates

For a single observed entry $y_{ijk}$, define the regularized instantaneous objective

$$\begin{aligned}\varepsilon_{ijk} = \frac{1}{2}\left(y_{ijk} - \left(\sum_{s=1}^{R} t_{is}p_{js}d_{ks} + a_i + b_j + c_k\right)\right)^2 \\ + \frac{\lambda}{2}\left(\sum_{s=1}^{R}(t_{is}^2 + p_{js}^2 + d_{ks}^2) + a_i^2 + b_j^2 + c_k^2\right).\end{aligned} \tag{S1}$$

**Definition 1 (smoothness).** A differentiable function $\varepsilon : \mathbb{R}^d \to \mathbb{R}$ is $L$-smooth if

$$\|\nabla\varepsilon(\theta_1) - \nabla\varepsilon(\theta_2)\|_2 \le L\|\theta_1 - \theta_2\|_2, \qquad \forall\, \theta_1, \theta_2 \in \mathbb{R}^d. \tag{S2}$$

Fix all variables except $t_{ir}$. Terms independent of $t_{ir}$ can be absorbed into a constant, giving

$$\varepsilon_{ijk}(t_{ir}) = \frac{1}{2}\left(y_{ijk} - \left(t_{ir}p_{jr}d_{kr} + \sum_{s\neq r} t_{is}p_{js}d_{ks} + a_i + b_j + c_k\right)\right)^2 + \frac{\lambda}{2}t_{ir}^2 + \text{const}. \tag{S3}$$

The derivative with respect to $t_{ir}$ is

$$\nabla\varepsilon_{ijk}(t_{ir}) = \left(\sum_{s=1}^{R} t_{is}p_{js}d_{ks} + a_i + b_j + c_k - y_{ijk}\right)p_{jr}d_{kr} + \lambda t_{ir}. \tag{S4}$$

For two values $t_{ir}^{(1)}$ and $t_{ir}^{(2)}$, with every other coordinate held fixed,

$$\begin{aligned}&\nabla\varepsilon_{ijk}(t_{ir}^{(1)}) - \nabla\varepsilon_{ijk}(t_{ir}^{(2)})\\&= \left[\left(t_{ir}^{(1)} - t_{ir}^{(2)}\right) p_{jr}d_{kr}\right] p_{jr}d_{kr} + \lambda\left(t_{ir}^{(1)} - t_{ir}^{(2)}\right).\end{aligned} \tag{S5}$$

Hence,

$$\nabla\varepsilon_{ijk}(t_{ir}^{(1)}) - \nabla\varepsilon_{ijk}(t_{ir}^{(2)}) = \left(p_{jr}^2 d_{kr}^2 + \lambda\right)\left(t_{ir}^{(1)} - t_{ir}^{(2)}\right), \tag{S6}$$

and therefore

$$\left\|\nabla\varepsilon_{ijk}(t_{ir}^{(1)}) - \nabla\varepsilon_{ijk}(t_{ir}^{(2)})\right\|_2 \le \left(p_{jr}^2 d_{kr}^2 + \lambda\right)\left\|t_{ir}^{(1)} - t_{ir}^{(2)}\right\|_2. \tag{S7}$$

Thus $L_t = p_{jr}^2 d_{kr}^2 + \lambda$. By the same calculation,

$$L_p = t_{ir}^2 d_{kr}^2 + \lambda, \qquad L_d = t_{ir}^2 p_{jr}^2 + \lambda. \tag{S8}$$

For the bias coordinate $a_i$,

$$\nabla\varepsilon_{ijk}(a_i) = \left(\sum_{s=1}^{R} t_{is}p_{js}d_{ks} + a_i + b_j + c_k - y_{ijk}\right) + \lambda a_i. \tag{S9}$$

For arbitrary $a_i^{(1)}$ and $a_i^{(2)}$,

$$\nabla\varepsilon_{ijk}(a_i^{(1)}) - \nabla\varepsilon_{ijk}(a_i^{(2)}) = (1+\lambda)\left(a_i^{(1)} - a_i^{(2)}\right), \tag{S10}$$

so

$$\left\|\nabla\varepsilon_{ijk}(a_i^{(1)}) - \nabla\varepsilon_{ijk}(a_i^{(2)})\right\|_2 \le (1+\lambda)\left\|a_i^{(1)} - a_i^{(2)}\right\|_2. \tag{S11}$$

The same result holds for $b_j$ and $c_k$:

$$L_a = L_b = L_c = 1 + \lambda. \tag{S12}$$

**Definition 2 (coordinate-wise strong convexity).** A scalar coordinate objective $\phi$ is $\delta$-strongly convex if, for all $\theta_1, \theta_2 \in \mathbb{R}$,

$$\phi(\theta_1) \ge \phi(\theta_2) + \nabla\phi(\theta_2)(\theta_1 - \theta_2) + \frac{\delta}{2}\|\theta_1 - \theta_2\|_2^2. \tag{S13}$$

For the $t_{ir}$ coordinate,

$$\nabla^2\varepsilon_{ijk}(t_{ir}) = p_{jr}^2 d_{kr}^2 + \lambda, \tag{S14}$$

and for the bias coordinate,

$$\nabla^2\varepsilon_{ijk}(a_i) = 1 + \lambda. \tag{S15}$$

Because $\lambda > 0$, these curvatures admit uniform positive lower bounds while the other coordinates are fixed:

$$\nabla^2\varepsilon_{ijk}(t_{ir}) \ge \lambda \equiv \delta_t > 0, \qquad \nabla^2\varepsilon_{ijk}(a_i) \ge 1 + \lambda \equiv \delta_a > 0. \tag{S16}$$

The same coordinate-wise argument applies to $p_{jr}$, $d_{kr}$, $b_j$, and $c_k$. This does not imply that the jointly factorized objective is globally convex.

Let $\mathcal{F}_{ir}(t)$ denote the expected coordinate objective obtained by sampling an observed entry involving $t_{ir}$ while all other coordinates are fixed. Let $g_l = \nabla\varepsilon_{ijk}(t_{ir}^{l-1})$ be an unbiased stochastic gradient of $\mathcal{F}_{ir}$, and let $\eta_l > 0$ be the step size. The update is

$$t_{ir}^{l} = t_{ir}^{l-1} - \eta_l g_l. \tag{S17}$$

If $t_{ir}^*$ minimizes $\mathcal{F}_{ir}$, expanding the squared distance gives

$$\|t_{ir}^{l} - t_{ir}^*\|_2^2 = \|t_{ir}^{l-1} - t_{ir}^*\|_2^2 - 2\eta_l g_l(t_{ir}^{l-1} - t_{ir}^*) + \eta_l^2\|g_l\|_2^2. \tag{S18}$$

Strong convexity of $\mathcal{F}_{ir}$ yields

$$\mathcal{F}_{ir}(t_{ir}^*) \ge \mathcal{F}_{ir}(t_{ir}^{l-1}) + \nabla\mathcal{F}_{ir}(t_{ir}^{l-1})(t_{ir}^* - t_{ir}^{l-1}) + \frac{\delta}{2}\|t_{ir}^{l-1} - t_{ir}^*\|_2^2. \tag{S19}$$

At the coordinate optimum,

$$\mathcal{F}_{ir}(t_{ir}^*) \le \mathcal{F}_{ir}(t), \quad \forall t, \qquad \nabla \mathcal{F}_{ir}(t_{ir}^*) = 0. \tag{S20}$$

Rearranging the strong-convexity inequality and discarding the nonnegative function-value gap gives

$$\nabla \mathcal{F}_{ir}(t_{ir}^{l-1})(t_{ir}^{l-1} - t_{ir}^*) \ge \frac{\delta}{2} \|t_{ir}^{l-1} - t_{ir}^*\|_2^2. \tag{S21}$$

Taking conditional expectation in the distance recursion and using the unbiasedness of $g_l$ therefore gives

$$\mathbb{E}_l\left[\|t_{ir}^l - t_{ir}^*\|_2^2\right] \le (1 - \eta_l \delta)\|t_{ir}^{l-1} - t_{ir}^*\|_2^2 + \eta_l^2 \mathbb{E}_l\left[\|g_l\|_2^2\right]. \tag{S22}$$

Taking total expectation,

$$\mathbb{E}\left[\|t_{ir}^l - t_{ir}^*\|_2^2\right] \le (1 - \eta_l \delta)\mathbb{E}\left[\|t_{ir}^{l-1} - t_{ir}^*\|_2^2\right] + \eta_l^2 \mathbb{E}\left[\|g_l\|_2^2\right]. \tag{S23}$$

Assume, as in standard stochastic-approximation analyses (Nemirovski et al., 2009), that the stochastic-gradient second moment is uniformly bounded:

$$\mathbb{E}\left[\|g_l\|_2^2\right] \le \rho^2 < \infty. \tag{S24}$$

Then

$$\mathbb{E}\left[\|t_{ir}^l - t_{ir}^*\|_2^2\right] \le (1 - \eta_l \delta)\mathbb{E}\left[\|t_{ir}^{l-1} - t_{ir}^*\|_2^2\right] + \eta_l^2 \rho^2. \tag{S25}$$

Choose $\eta_l = c/[\delta(l + \ell_0)]$ with $c > 1$ and $\ell_0 \ge c$, and define $m_l = l + \ell_0$. A standard induction on the preceding recursion gives

$$\mathbb{E}\left[\|t_{ir}^l - t_{ir}^*\|_2^2\right] \le \left(1 - \frac{c}{m_l}\right)\mathbb{E}\left[\|t_{ir}^{l-1} - t_{ir}^*\|_2^2\right] + \frac{c^2\rho^2}{\delta^2 m_l^2} \le \frac{Q(c)}{m_l}. \tag{S26}$$

For an $L$-smooth coordinate objective with $\nabla \mathcal{F}_{ir}(t_{ir}^*) = 0$,

$$\mathcal{F}_{ir}(t_{ir}^l) - \mathcal{F}_{ir}(t_{ir}^*) \le \frac{L}{2}\|t_{ir}^l - t_{ir}^*\|_2^2. \tag{S27}$$

Consequently,

$$\mathbb{E}\left[\mathcal{F}_{ir}(t_{ir}^l) - \mathcal{F}_{ir}(t_{ir}^*)\right] \le \frac{LQ(c)}{2m_l}. \tag{S28}$$

A sufficient induction constant is

$$Q(c) = \max\left\{m_1\|t_{ir}^1 - t_{ir}^*\|_2^2,\ \frac{c^2\rho^2}{\delta^2(c-1)}\right\}. \tag{S29}$$

Thus, under the stated assumptions, the expected suboptimality of the sampled coordinate objective decays as $\mathcal{O}(1/l)$. The same argument applies to the other scalar factor and bias coordinates when the remaining variables are held fixed.

### C.2 Analysis of PID-Regulated Updates

We next verify that the NPID-regulated error does not make a coordinate stochastic gradient unbounded under explicit boundedness assumptions. Before applying NPID regulation, the $t_{ir}$ gradient is

$$\nabla \varepsilon_{ijk}(t_{ir}^{l-1}) = \lambda t_{ir}^{l-1} - e_{ijk}^l p_{jr} d_{kr}. \tag{S30}$$

Therefore,

$$\|\nabla \varepsilon_{ijk}(t_{ir}^{l-1})\|_2 \le \lambda\|t_{ir}^{l-1}\|_2 + \|e_{ijk}^l p_{jr} d_{kr}\|_2, \tag{S31}$$

or equivalently,

$$\|\nabla \varepsilon_{ijk}(t_{ir}^{l-1})\|_2 \le \lambda\|t_{ir}^{l-1}\|_2 + |p_{jr} d_{kr}|\,\|e_{ijk}^l\|_2. \tag{S32}$$

Assume that the relevant factor and bias states and the instantaneous error remain bounded on the training trajectory:

$$\|t_{ir}^{l-1}\|_2 \le B_t, \qquad |p_{jr} d_{kr}| \le B_{pd}, \qquad \|e_{ijk}^l\|_2 \le B_e, \qquad |a_i^{l-1}| \le B_a. \tag{S33}$$

The fuzzy rule table restricts each controller gain to a finite interval. Let $\bar{K}_p, \bar{K}_i, \bar{K}_d$ be upper bounds on the absolute NPID gains. Because $f_m$ is also bounded and $F(e) = \cosh(f_m e)$ is continuous, bounded $e$ implies $|F(e)| \le M_F < \infty$. We additionally assume that the accumulated integral state is bounded, $\| \sum_{h=0}^{l} F(e_{ijk}^h) e_{ijk}^h \|_2 \le B_I$. This quantity is finite over a finite training horizon and can also be enforced by standard integral clipping or anti-windup. Applying the triangle inequality to equation 8 gives

$$\begin{aligned} \|\tilde{e}_{ijk}^l\|_2 &\le \bar{K}_p M_F B_e + \bar{K}_i B_I + \bar{K}_d M_F \|e_{ijk}^l - e_{ijk}^{l-1}\|_2 \\ &\le \bar{K}_p M_F B_e + \bar{K}_i B_I + 2\bar{K}_d M_F B_e \equiv B_{\tilde{e}} < \infty. \end{aligned} \tag{S34}$$

After NPID regulation, the corresponding coordinate gradient becomes

$$\nabla \tilde{\varepsilon}_{ijk}(t_{ir}^{l-1}) = \lambda t_{ir}^{l-1} - \tilde{e}_{ijk}^l p_{jr} d_{kr}. \tag{S35}$$

Its norm satisfies

$$\|\nabla \tilde{\varepsilon}_{ijk}(t_{ir}^{l-1})\|_2 \le \lambda \|t_{ir}^{l-1}\|_2 + |p_{jr} d_{kr}| \, \|\tilde{e}_{ijk}^l\|_2. \tag{S36}$$

In particular,

$$\|\tilde{e}_{ijk}^l\|_2 \le B_{\tilde{e}}. \tag{S37}$$

Combining the state bounds with the regulated-error bound yields

$$\|\nabla \tilde{\varepsilon}_{ijk}(t_{ir}^{l-1})\|_2 \le \lambda B_t + B_{pd} B_{\tilde{e}} \equiv G_t < \infty. \tag{S38}$$

Hence the regulated factor-coordinate gradient is uniformly bounded along the assumed training trajectory:

$$\|\nabla \tilde{\varepsilon}_{ijk}(t_{ir}^{l-1})\|_2 \le G_t. \tag{S39}$$

For the bias coordinate, the same reasoning gives

$$\|\nabla \tilde{\varepsilon}_{ijk}(a_i^{l-1})\|_2 \le \lambda B_a + B_{\tilde{e}} \equiv G_a < \infty. \tag{S40}$$

The preceding bounds show that, under bounded factor states, bounded fuzzy-controller gains, bounded nonlinear-gain output, and a bounded integral state, NPID regulation preserves bounded coordinate stochastic gradients. This verifies the bounded-gradient requirement used by the stochastic analysis, but boundedness alone does not make the PID-regulated direction an unbiased gradient of the same fixed objective. Accordingly, the $\mathcal{O}(1/l)$ rate above is established for the underlying fixed-$\lambda$ coordinate SGD with the stated diminishing step size, while the NPID result is a stability/boundedness guarantee. No claim of joint global convergence is made for the full nonconvex DPAL objective, the fuzzy-varying $\lambda$, or an unrestricted IPID learning-rate trajectory.