

SFlexRCA: Lightweight, Scalable, and Flexible Root Cause Analysis for IIoT Edge Systems

Amr M. Zaki, Farhoud Jafari, Honggeun Ji, Komal Sarda, Marin Litoiu

Abstract—Industrial Internet of Things (IIoT) systems generate high-dimensional sensor telemetry from interconnected components, where faults can propagate across the system. Existing root cause analysis (RCA) approaches often depend on incomplete dependency or observability information, while non-causal methods may require abundant normal-operation data and causal approaches can incur substantial computational and parameter costs. To address these challenges, we propose SFlexRCA (Scalable and Flexible Root Cause Analysis), a topology-free RCA framework designed for resource-constrained IIoT environments. SFlexRCA transforms multivariate telemetry into compact orthogonal representations and applies shared lightweight linear modeling, avoiding explicit graph construction, message passing, and per-variable or lag-specific parameter growth. We evaluate SFlexRCA on three publicly available IIoT datasets, BATADAL, SWaT, and WADI, spanning different numbers of monitored variables, temporal characteristics, and training-data regimes. SFlexRCA is compared with 10 statistical, causal, and non-causal baselines in terms of RCA accuracy, training efficiency, inference latency, and memory consumption. In addition, inference efficiency and memory consumption are evaluated on Raspberry Pi 3 and Raspberry Pi 5, while energy consumption is additionally measured on Raspberry Pi 5. We further investigate temporal-context sensitivity, architectural and loss components, and alternative representations. SFlexRCA achieves the best AC@1 on BATADAL (0.476) and the best AC@5 on SWaT (0.367), while using only 279.4K and 108.1K parameters, respectively. Notably, SFlexRCA maintains strong localization performance on BATADAL despite its limited normal-operation training data, while its compact shared architecture avoids the parameter growth associated with causal and graph-based approaches. On Raspberry Pi 5, SFlexRCA achieves 20.3, 8.2, and 9.3 ms/sample on BATADAL, SWaT, and WADI, respectively. These results demonstrate that SFlexRCA provides a practical alternative to computationally intensive causal and graph-based RCA approaches, achieving strong localization with substantially smaller models while remaining effective under limited training data. Its lightweight shared architecture further enables efficient deployment on resource-constrained IIoT edge devices. The SFlexRCA code is available at <https://github.com/theamrzaki/RootCauseAnalysis-Correlation-Attentive-Modeling>.

Index Terms—Industrial Internet of Things, Root Cause Analysis, Multivariate Time Series, Linear Models, Edge Computing, Resource-Constrained Devices.

I. INTRODUCTION

As industrial processes increasingly integrate networked sensors, actuators, controllers, and computing platforms, Industrial Internet of Things (IIoT) systems have become critical

to modern water treatment, energy, manufacturing, and other cyber-physical infrastructures [1]. These systems continuously generate heterogeneous time-series measurements, making data-driven monitoring essential for operational reliability and safety [2], [3]. However, failures can propagate across physically and logically coupled components, producing correlated abnormal behavior across multiple sensors and subsystems [4]. Identifying the source of such abnormal behavior is therefore essential for reliable and resilient IIoT operation [5].

Anomaly detection can identify deviations from normal behavior [6], but does not determine which component or process variable initiated the failure [7]. Root cause analysis (RCA) addresses this limitation by identifying the most likely sources of abnormal behavior and supporting corrective actions [8]. This task is particularly challenging in industrial environments because sensors, actuators, control loops, and physical subsystems are strongly coupled. A fault originating from one component can propagate through the process and appear as correlated anomalies across many measurements, making it difficult to distinguish the initiating fault from downstream effects. Existing RCA approaches face several practical limitations in IIoT settings. Multimodal methods may rely on logs and distributed traces that are incomplete or unavailable in industrial environments [9], [10], [11], [12], [13]. Recent IoT and industrial approaches have increasingly explored causal structure discovery, causal interpretability, and dynamic causal graphs to capture dependencies among system variables [14], [15], [16]. Other approaches combine reinforcement learning and graph-based causal diagnosis for root cause localization [17]. Some methods further incorporate large language models (LLMs) to assist causal reasoning and root cause localization [16]. While these approaches can exploit rich dependency information, explicit causal structures, graph construction, reinforcement-learning-based discovery, and LLM-based components can introduce additional computational and memory requirements, making deployment challenging on resource-constrained edge devices. Moreover, graph message passing [8], independent models for individual variables [18], and lag- or context-dependent architectures [19], [20] can further increase parameter and computational costs. These constraints are particularly important for distributed IoT systems, where efficient computation, energy use, and resource allocation remain central challenges for edge intelligence [21], [22], [23]. At the same time, non-causal forecasting approaches can provide compact representations but may require substantial normal-operation data for reliable localization, which can be difficult to obtain in industrial settings. These limitations expose a gap between compact

Amr M. Zaki (corresponding author), Farhoud Jafari, Honggeun Ji, Komal Sarda, and Marin Litoiu are with the Lassonde School of Engineering, York University, Toronto, ON, Canada. (e-mail: amrzaki@yorku.ca; farhoud@yorku.ca; gihong96@yorku.ca; komal253@yorku.ca; mlitoiu@yorku.ca).

non-causal models and increasingly sophisticated but computationally demanding dependency-aware approaches. This motivates the question: Can root cause localization retain the benefits of dependency-aware modeling without requiring explicit dependency graphs, computationally intensive causal-discovery mechanisms, LLM-based components, or expensive per-variable and graph-based architectures, while remaining effective across different data regimes and practical for edge deployment?

We address this gap with **SFlexRCA (Scalable and Flexible Root Cause Analysis)**, a lightweight and topology-free framework for industrial root cause localization. SFlexRCA represents multivariate telemetry in a compact orthogonal space and uses shared linear modeling to capture temporal and cross-variable patterns. Industrial measurements often contain redundancy, correlation, and shared temporal structure, making compact representations a promising alternative to explicitly modeling every variable and dependency [24], [25], [26], [27]. By operating in a shared representation space, SFlexRCA avoids explicit dependency graphs, message passing, and independent per-variable models, thereby limiting parameter growth as telemetry dimensionality and temporal context increase. Its modular architecture also allows the model to adapt to different industrial systems.

We evaluate SFlexRCA on **SWaT**, **WADI**, and **BATADAL** [28], [29], [30], covering different telemetry dimensionalities and training-data regimes. We assess RCA accuracy, parameter complexity, training and inference efficiency, and memory consumption on both **Raspberry Pi 3** and **Raspberry Pi 5**, while evaluating energy consumption on Raspberry Pi 5. SFlexRCA achieves the best AC@1 on BATADAL and the best AC@5 on SWaT while using substantially fewer trainable parameters than competing approaches, demonstrating its practical scalability without the parameter growth associated with dependency-aware causal and graph-based models. It also remains effective under limited normal-operation training data, as observed on BATADAL. We further examine temporal-context sensitivity, architectural and loss components, alternative representation methods, and the learned representations underlying localization.

The main contributions of this work are:

- A lightweight, topology-free RCA framework based on orthogonal representations and shared linear modeling, avoiding explicit dependency graphs and per-variable or per-window architectures.
- An adaptive architecture that limits parameter growth with telemetry dimensionality and temporal context while supporting different industrial data regimes.
- A comprehensive evaluation across SWaT, WADI, and BATADAL, including accuracy, computational and memory efficiency, and edge energy consumption.
- Ablation and sensitivity analyses examining the proposed components, temporal context, representation choices, and localization behavior.

This work is guided by the following research questions:

RQ1: How does SFlexRCA balance root cause localization

accuracy and model complexity in terms of trainable parameters?

RQ2: How does SFlexRCA compare in training efficiency, memory consumption, and inference latency on Raspberry Pi 3 and 5?

RQ3: What are the contributions of SFlexRCA’s architectural and loss components to its localization performance?

RQ4: How does temporal context affect SFlexRCA’s root cause localization accuracy?

RQ5: How does SFlexRCA’s orthogonal representation compare with alternative representations in terms of localization accuracy and latent-space separability?

This paper is organized as follows. Section 2 reviews related work. Section 3 presents the problem formulation and SFlexRCA. Section 4 presents the experimental evaluation. Section 5 includes threat analysis and discussion. Section 6 concludes the paper.

II. RELATED WORK

RCA in multivariate systems lies at the intersection of performance engineering and system diagnosis, where the goal extends beyond anomaly detection toward scalable and operationally practical fault localization.

A. Root Cause Analysis

Existing RCA approaches can be broadly categorized into statistical, multimodal, non-causal forecasting, and neural dependency-based methods, each making different assumptions about the available observability, training data, and system structure.

Statistical RCA methods primarily identify abnormal dependencies or distributional shifts among monitoring signals. ϵ -Diagnosis [31] uses two-sample statistical testing, while RCD [32] employs conditional independence tests to infer causal relationships among system metrics. BARO [33] further improves robustness through median- and IQR-based scoring. These methods are computationally efficient and interpretable, but are primarily designed for microservice and software-system telemetry and can struggle to capture the complex temporal dependencies and nonlinear interactions characteristic of industrial processes.

A second line of work exploits multimodal observability by jointly analyzing metrics, logs, and traces. Approaches such as ART, Eadro, MSTGAD, and LLM4MST [9], [10], [11], [12], [13] leverage complementary information across modalities to improve fault localization. However, these approaches assume the availability of sufficiently complete and aligned multimodal telemetry, which may be impractical in industrial environments dominated by sensor and control measurements. Industrial systems may lack distributed traces or consistent logs across heterogeneous controllers and devices, making single-modality approaches more suitable for partially observable and resource-constrained deployments.

Non-causal forecasting approaches provide an alternative by learning temporal patterns directly from multivariate observations without explicitly modeling causal or dependency structures. Recent models such as TimeMixer++ [36] and iTransformer [35] can capture complex temporal and cross-variable

TABLE I: Comparison of Representative Root Cause Analysis and Efficient Time-Series Modeling Approaches.

Method	Telemetry	Architecture	Edge Evaluation	Target
Statistical RCA				
ϵ -Diagnosis [31]	M	Two-sample testing	–	Microservices
RCD [32]	M	Conditional independence	–	Microservices
BARO [33]	M	Robust statistical scoring	–	Microservices
Multimodal RCA				
ART [10]	L+M+T	Transformer + GRU	–	Distributed systems
Eadro [9]	L+M+T	GAT	–	Distributed systems
MSTGAD [11]	L+M+T	Spatial-temporal Transformer	–	Distributed systems
LLM4MST [12]	L+M+T	GPT-2	–	Distributed systems
Neural Dependency and Causal RCA				
cMLP/cLSTM [18]	M	Per-variable MLP/LSTM	–	Causal RCA
CUTS [34]	M	GNN + imputation	–	Causal RCA
GVAR [19]	M	Per-window MLP	–	Causal RCA
AERCA [20]	M	Per-window MLP	–	Causal RCA
CUTS+ [8]	M	Message passing	–	Causal RCA
Cao et al. [14]	M	Causal structure discovery	–	Industrial monitoring
Tang et al. [16]	M	Dynamic causal graph + LLM	–	IoT RCA
Non-Causal Forecasting				
iTransformer [35]	M	Inverted Transformer	–	Forecasting
TimeMixer++ [36]	M	Multiscale mixing	–	Forecasting
Efficient Linear and Edge Time-Series Methods				
FITS [26]	M	Linear / frequency	–	Forecasting
FilterNet [37]	M	Linear / frequency	–	Forecasting
OLinear [38]	M	Orthogonal linear	–	Forecasting
VLinear [25]	M	Linear projection	–	Forecasting
FADSD [2]	M	Lightweight statistical	RPi / STM32	Anomaly detection
LFTSAD [3]	M	Lightweight MLP	RPi / Jetson	Anomaly detection
SFlexRCA (Ours)	M	Orthogonal projection + shared linear modeling	RPi 3 / RPi 5	IIoT RCA

M: Metrics/sensor measurements, L: Logs, T: Traces.

patterns and have demonstrated strong forecasting capability. However, their effectiveness for root cause localization can depend on the availability of abundant normal-operation data. This limitation is particularly relevant to IIoT environments, where collecting representative normal-operation data can be difficult due to operational costs, system changes, or limited periods of stable operation. Moreover, forecasting accuracy does not necessarily translate directly into reliable root cause localization, since accurate prediction of system behavior does not explicitly identify the variable responsible for an observed failure. Thus, while non-causal models can provide attractive computational efficiency, their data requirements and indirect localization mechanism can limit their applicability to data-limited diagnostic scenarios and resource-constrained deployments.

Neural dependency-based methods instead attempt to learn temporal or causal relationships directly from multivariate observations. Methods such as cMLP and cLSTM [18] use neural networks associated with individual variables to recover dependency structures. CUTS [34] addresses irregular

and incomplete time-series data through a neural Granger-causality framework that jointly imputes missing observations and learns causal graphs using graph neural networks. Recent work has further investigated causal structure discovery for industrial processes, including nonlinear systems with complex noise [14], while IoT-oriented RCA approaches have employed dynamic causal graphs to model evolving dependencies and have incorporated LLMs to assist causal reasoning and root cause localization [16]. GVAR [19] and AERCA [20] use deep temporal architectures that process historical monitoring windows, requiring deep models to be repeatedly applied across temporal windows during training and inference. CUTS+ [8] further employs message passing over learned representations to model interactions among variables. These approaches provide more direct dependency-aware mechanisms for localization and can remain effective when training data are limited. However, learning explicit causal structures, using per-variable neural networks, repeatedly processing temporal windows, or performing graph-based message passing can introduce substantial computational and memory overhead. LLM-assisted causal approaches further increase these resource requirements, making deployment challenging on resource-constrained IIoT edge devices, particularly as the number of monitored variables and temporal context increase.

Overall, existing RCA approaches exhibit a fundamental trade-off between data requirements, dependency modeling, and computational cost. Non-causal forecasting methods can provide efficient temporal modeling but may require substantial normal-operation data and do not directly model root-cause dependencies. Causal and neural dependency-based methods provide more explicit dependency-aware localization but often rely on computationally intensive per-variable, temporal, graph-based, or LLM-assisted architectures. In contrast, SFlexRCA targets topology-free RCA from a single modality of industrial telemetry by combining orthogonal representations with shared lightweight linear components. This design captures temporal and cross-variable information without explicitly constructing dependency graphs, using message passing or LLMs, or relying on per-variable and lag-specific architectures, enabling efficient root cause localization across different training-data regimes and supporting deployment on resource-constrained IIoT edge devices.

B. Orthogonal and Linear Temporal Modeling

Recent advances in high-dimensional time-series modeling suggest that structured transformations can simplify temporal learning while reducing computational cost. Frequency-domain methods such as FITS [26], FEDformer [39], FilterNet [37], and FourierGNN [40] demonstrate the effectiveness of alternative representations for modeling long-range temporal dependencies. More recently, OLinear [38] and VLinear [25] show that dataset-specific linear projections can achieve competitive performance with substantially lower computational cost, suggesting that complex temporal structure can be captured in transformed spaces without deep nonlinear architectures. Recent efforts have also investigated efficient anomaly detection directly on resource-constrained

devices. For example, [2], [3] evaluate lightweight time-series anomaly detection models on edge platforms, including Raspberry Pi, STM32, and NVIDIA Jetson Xavier. While these studies demonstrate the feasibility of deploying anomaly detection at the edge, they focus on identifying anomalous behavior rather than determining its underlying root cause.

In contrast, SFlexRCA applies lightweight orthogonal projections and shared linear components to root cause localization. By avoiding explicit dependency graphs, message passing, and per-variable or per-window architectures, SFlexRCA enables topology-free RCA from single-modality industrial telemetry while targeting efficient deployment on resource-constrained IIoT edge devices.

III. SFLEXRCA

A. Problem Formulation

Let $X \in \mathbb{R}^{T \times P}$ denote a multivariate IIoT telemetry window containing P sensor variables observed over T time steps. Given normal operating data, SFlexRCA learns the temporal and cross-variable behavior of the system and uses deviations from this learned behavior to localize potential root causes.

Given an input window X , SFlexRCA produces a future-state prediction $\hat{Y}$ and an implicit variable-interaction representation C :

$$\hat{Y}, C = \text{SFlexRCA}(X), \quad (1)$$

where $\hat{Y}$ represents the predicted future system state and $C \in \mathbb{R}^{T \times P \times P}$ captures pairwise interactions inferred from the learned latent representations. Unlike topology-aware RCA approaches, SFlexRCA does not require a predefined dependency graph, process topology, or explicit communication structure among monitored variables.

B. SFlexRCA Architecture

SFlexRCA is designed for single-modality IIoT telemetry, where large numbers of heterogeneous sensor measurements must be modeled efficiently under the computational constraints of edge devices. As illustrated in Fig. 1, SFlexRCA consists of five main stages: (i) orthogonal representation, (ii) shared variable encoding, (iii) temporal context modeling, (iv) context refinement and cross-variable interaction modeling, and (v) temporal aggregation and prediction. The architecture uses lightweight shared components rather than per-variable neural networks or graph-based message-passing architectures.

(1) Orthogonal Representation.

IIoT monitoring data typically contain highly correlated measurements and substantial temporal redundancy across sensor windows. Directly learning from the raw temporal coordinates can therefore require the downstream RCA model to repeatedly capture dependencies that are already present in the normal operating data. To provide a compact temporal representation, SFlexRCA constructs a dataset-specific orthogonal basis from normal IIoT telemetry following [38].

For each monitored variable p , we compute its temporal correlation matrix over the normal training windows and obtain a shared correlation structure by averaging across variables:

$$\bar{\Sigma} = \frac{1}{P} \sum_{p=1}^P \text{Corr}(X^{(p)}), \quad (2)$$

where $X^{(p)}$ contains the temporal observations of variable p . The eigenvectors of $\bar{\Sigma}$ define the orthogonal basis Q :

$$\bar{\Sigma} = Q^\top \Lambda Q, \quad Q^\top Q = I. \quad (3)$$

The normalized telemetry is then projected as

$$X_{\text{orth}} = Q\tilde{X}. \quad (4)$$

Because Q is orthogonal, the transformation preserves the Euclidean geometry of the telemetry while expressing temporal variations along orthogonal directions. The basis is computed once using normal training data and remains fixed during model training, introducing no additional trainable parameters. This design is particularly suitable for IIoT settings because the representation separates recurring temporal structure from the subsequent learning of cross-variable interactions, allowing the latter to be modeled using a shared lightweight architecture rather than independent models for each sensor. The inverse transformation $Q^\top$ is used to recover predictions in the original measurement space.

(2) Shared Variable Encoding.

After transformation, each scalar observation is mapped into a shared latent space of dimension D using the same projection function across all variables. To distinguish heterogeneous sensor variables, SFlexRCA employs a learnable variable embedding $E^{(p)}$ for each monitored variable p [25]. In the default configuration, the latent representation is constructed as

$$H_t^{(p)} = \psi(X_{\text{orth},t}^{(p)}) \odot E^{(p)}. \quad (5)$$

This shared parameterization encodes variable-specific characteristics without requiring a separate neural network for each variable, keeping the projection parameters independent of the number of monitored variables [18], [34].

(3) Temporal Context Modeling.

To capture local temporal dependencies, SFlexRCA optionally applies a lightweight depthwise one-dimensional convolution:

$$H \leftarrow H + \text{Conv1D}(H), \quad (6)$$

where the convolution operates independently over latent channels, providing local temporal mixing with minimal additional parameters. Its contribution is evaluated through architectural ablation.

(4) Context Refinement and Cross-Variable Interaction Modeling.

The temporally encoded representation is refined using a lightweight context mechanism:

$$H' = \mathcal{A}(H), \quad (7)$$

where $\mathcal{A}(\cdot)$ denotes either gated modulation or linear attention [41]. The latter uses positive feature mappings to avoid

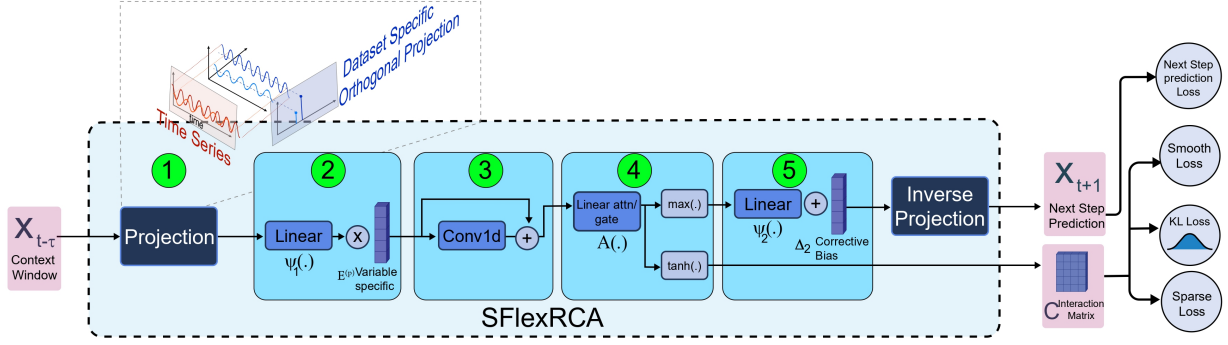


Fig. 1: SFlexRCA framework for unsupervised root-cause analysis in IIoT systems.

explicitly constructing the $T \times T$ attention matrix, reducing the memory cost of conventional self-attention.

Cross-variable interactions are then inferred directly from the refined representations without explicit dependency graphs or message passing:

$$C_{t,p,q} = \tanh \left(H_t^{(p)\top} H_t^{(q)} \right), \quad (8)$$

yielding $C \in \mathbb{R}^{T \times P \times P}$. Thus, variable interactions are inferred dynamically from the current telemetry window rather than imposed by a predefined system topology.

(5) Temporal Aggregation and Prediction.

The refined representation H' is aggregated across time using max pooling:

$$Z^{(p)} = \max_t H_t^{(p)}. \quad (9)$$

This retains the strongest temporal evidence for each latent feature while avoiding recurrent state propagation. The resulting representation is passed to a lightweight prediction head:

$$\hat{Y} = \psi_2(Z) + B, \quad (10)$$

where $\psi_2(\cdot)$ denotes either a linear or shallow nonlinear head and B is a learnable bias. The predicted representation is finally mapped back to the original measurement space using the inverse orthogonal transformation when applicable.

C. Training Objective

SFlexRCA jointly learns the expected system behavior and an explicit cross-variable interaction representation. The overall objective is

$$\mathcal{L} = \mathcal{L}_{\text{pred}} + \mathcal{L}_{\text{reg}}, \quad (11)$$

where

$$\mathcal{L}_{\text{pred}} = \left| \hat{Y}_{t+1} - Y_{t+1} \right|_2^2 \quad (12)$$

measures the prediction error, and

$$\mathcal{L}_{\text{reg}} = \lambda_1 \mathcal{L}_{\text{sparse}}(C) + \lambda_2 \mathcal{L}_{\text{smooth}}(C) + \lambda_3 \mathcal{L}_{\text{KL}}(C) \quad (13)$$

regularizes the learned interaction representation. The sparsity, temporal smoothness, and KL terms therefore directly

constrain the cross-variable interactions inferred from the telemetry. This encourages the model to represent normal IIoT behavior using a compact and temporally consistent interaction structure while jointly optimizing predictive accuracy.

At inference time, SFlexRCA ranks monitored variables using their prediction residuals. The interaction representation is not treated as a separate fault score; instead, it is learned jointly with the predictor and structurally constrained during training. This follows the common principle in neural RCA methods [18], [20] of learning a predictive model whose internal representations are constrained to capture meaningful system dependencies, while SFlexRCA differs in explicitly constructing these dependencies from shared latent representations rather than maintaining a separate predictor or parameterization for each variable.

D. Complexity Analysis

The computational complexity of SFlexRCA can be summarized according to its five architectural stages. Steps (1) and (2) perform the orthogonal representation and shared variable encoding, with the latter mapping each variable into the shared hidden space, giving a dominant complexity of $O(PH)$. Steps (3) and (4) perform temporal context modeling and context refinement using shared latent representations, resulting in a complexity of $O(H^2)$ without introducing separate neural networks for individual variables. Step (5) performs temporal aggregation and prediction in the hidden space, contributing $O(HK)$. Consequently, the overall complexity is summarized as

$$O(PH + H^2 + HK). \quad (14)$$

This formulation highlights that SFlexRCA performs its main neural computation in a shared hidden space rather than replicating the computation across monitored variables. In contrast, cMLP and cLSTM introduce complexity that grows with the number of variables, GVAR and AERCA introduce multiplicative dependence on the temporal order through explicit coefficient modeling, and CUTS+ incurs additional complexity from graph-based message passing and its associated layers. SFlexRCA therefore shifts the dominant computation toward the compact hidden dimension H , while requiring only a shared linear dependence on the number of monitored variables. Table II summarizes the complexity of SFlexRCA and representative causal RCA methods.

TABLE II: Complexity comparison of RCA approaches. P is the number of monitored variables, H the hidden dimension, K the temporal order, and L the number of hidden layers.

Method	Complexity	Description
cMLP	$O(P^2KH)$	Separate MLP predictor for each target variable.
cLSTM	$O(P^2KH + PH^2)$	Separate recurrent predictor for each target variable.
GVAR	$O(P^2K)$	Explicitly models pairwise variable relationships across lags.
AERCA	$O(3KHP^2)$	Three GVAR-based networks generate a full $P \times P$ matrix at each lag.
CUTS+	$O(LPH^2 + PH + P^2)$	Uses graph-based recurrent message passing with pairwise node interactions.
SFlexRCA	$O(PH + H^2 + HK)$	Shared lightweight model; avoids explicit pairwise coefficient matrices.

IV. EXPERIMENTS

We evaluate SFlexRCA on three IIoT datasets to assess its effectiveness for RCA under the computational constraints of edge deployment. We compare SFlexRCA with statistical and causal deep learning baselines and investigate: (1) RCA accuracy across multiple datasets, (2) training and inference efficiency, particularly on resource-constrained Raspberry Pi devices, (3) the contribution of architectural and loss components, and (4) sensitivity to the temporal window size.

A. Datasets

We evaluate SFlexRCA on three industrial control system datasets with different numbers of monitored variables and temporal characteristics, as summarized in Table III. BATADAL [30] is a water distribution benchmark containing 43 monitored variables and captures temporal dynamics in a water distribution system. SWaT [28] is a water treatment benchmark with 51 monitored variables and longer temporal dependencies arising from the interactions among multiple physical processes. WADI [29] is a water distribution benchmark containing 127 monitored variables and provides a higher-dimensional setting for evaluating the scalability of RCA methods. Together, these datasets provide progressively different dimensional and temporal settings for evaluating root cause localization.

B. Experimental Setup

Baselines and Comparison.

We compare SFlexRCA against statistical, non-causal, and causal deep learning RCA methods. Statistical baselines include RCD [32], which identifies influential variables using

TABLE III: Comparison of the BATADAL, SWaT, and WADI datasets.

Metric	BATADAL	SWaT	WADI
System Type	Simulated water distribution (C-Town)	Physical water treatment testbed	Physical water distribution testbed
Sampling Frequency	1 hour	1 second	1 second
Number of Channels	43 sensor/actuator features	51 sensor/actuator features	127 sensor/actuator features
Training Data (Normal)	8,760 samples (365 days)	410,400–480,800 samples (7 days)	~1,200,000 samples (14 days)
Amount of Attacks	14 attacks total (7 train, 7 test)	36–41 attack scenarios (4 days)	15 attack scenarios (2 days)

conditional independence testing, BARO [33], which performs robust statistical root cause scoring, and TORAI [43], adapted to metric-only inputs for a consistent comparison. Non-causal baselines include TimeMixer++ [36] and iTransformer [35], which model multivariate temporal dependencies without explicitly learning causal relationships. For causal deep learning, we consider per-variable architectures (cLSTM and cMLP [18]), lag-specific architectures (GVAR [19] and AERCA [20]), and shared message-passing architectures (CUTS+ [8]). To assess the contribution of the proposed representation, we compare orthogonal projection with seven alternatives: no projection, learned projection, and static Legendre, Laguerre, Chebyshev, Hermite, and Fourier projections, following the experimental methodology of [44], [38]. All methods operate on the same metric-only telemetry inputs, and RCA scores are ranked using the same evaluation procedure. In particular, no additional anomaly thresholding or EVT-based post-processing is applied to SFlexRCA, ensuring a consistent comparison with the statistical and causal baselines.

Evaluation Metrics.

We evaluate the methods from two complementary perspectives: RCA accuracy and computational efficiency. For RCA accuracy, we report Recall at Top- k (AC@ k) for $k \in \{1, 5, 10\}$ and AUC@10, following established RCA evaluation protocols [32], [42]. AC@ k measures whether the ground-truth root-cause variables are included among the top- k variables ranked by each method, while AUC@10 evaluates the ranking quality across the top-10 predictions. For training efficiency, we report the number of trainable parameters, peak memory consumption, and average training time per epoch. For inference efficiency, we report average inference latency per sample on resource-constrained edge devices. Specifically, experiments are conducted on a Raspberry Pi 5 equipped with an ARM Cortex-A76 processor and 16 GB RAM, and a Raspberry Pi 3 equipped with an ARM Cortex-A53 processor and 1 GB RAM. For edge energy evaluation, inference was performed on the Raspberry Pi 5 while monitoring PMIC power using the `vcgencmd pmic_read_adc` interface. Total energy was obtained by trapezoidal integration of the measured power trace over the inference workload, and energy per sample was calculated by dividing the measured energy by

the total number of evaluated samples.

Implementation Details.

All experiments are conducted using multiple random seeds on a Linux workstation equipped with an Intel(R) Core(TM) i9-10900K CPU @ 3.70GHz (20 cores), 32 GB RAM, and an NVIDIA GeForce RTX 3070 GPU with 8 GB memory, running Ubuntu 22.04.2 LTS. The implementation uses Python 3.10.12, PyTorch 2.7.1+cu126 with CUDA 12.6, and PyTorch Geometric 2.6.1. Hyperparameters are tuned independently for each dataset, and the best configuration is selected. The latent dimension D is set to 128 for SWaT and WADI and 256 for BATADAL. The number of training epochs is set to 200 for SWaT and BATADAL and 50 for WADI, with early stopping applied. The learning rate is set to 1×10^{-4} across all experiments. For SFlexRCA, the orthogonal transformation is implemented using a dataset-specific linear projection.

C. RQ1: Root Cause Analysis Accuracy

RCA Accuracy Across Statistical and Learned Methods.

Table IV compares SFlexRCA with statistical, non-causal forecasting, and causal RCA methods across BATADAL, SWAT, and WADI. Statistical methods require no trainable parameters but provide inconsistent localization accuracy across datasets, reflecting their limited ability to capture complex temporal and cross-variable dependencies.

Effect of Training Data and Causal Modeling Non-causal forecasting models, including TimeMixer++ and iTransformer, perform substantially better on SWAT and WADI than on BATADAL. This difference is particularly pronounced for BATADAL, where only approximately 8K normal training samples are available: iTransformer achieves an AC@1 of only 0.048, while TimeMixer++ achieves 0.095. With substantially larger training sets of approximately 400K and 1.2M samples on SWAT and WADI, respectively, both models become considerably more competitive, with iTransformer reaching AC@1 values of 0.167 and 0.378. These results suggest that forecasting-based approaches benefit substantially from abundant normal-operation data, but their localization performance can deteriorate under data-limited conditions. For IIoT environments where collecting large volumes of representative normal-operation data may be impractical, this dependence can limit their applicability. In contrast, causal methods explicitly model temporal and cross-variable dependencies associated with system behavior, providing a mechanism more directly aligned with root-cause localization than forecasting alone. This provides an advantage when training data is limited. For example, cLSTM achieves an AC@1 of 0.381 on BATADAL, substantially exceeding the 0.048 and 0.095 obtained by iTransformer and TimeMixer++, respectively, despite all methods being trained using the same limited dataset. However, this dependency modeling generally comes at a substantial architectural cost, with parameter requirements increasing considerably as the number of monitored variables and temporal dependencies grow. On WADI, for example, cMLP and cLSTM require 14.5M and 16.7M parameters, respectively, while GVAR and AERCA require 14.8M and 44.4M parameters.

RCA Accuracy with Compact Architecture SFlexRCA achieves competitive or superior RCA accuracy while maintaining a substantially smaller architecture than most causal deep learning baselines. On BATADAL, SFlexRCA achieves the highest AC@1 of 0.476 with only 279.4K parameters, compared with 13.3M for cLSTM and 7.1M for cMLP. The advantage is also seen on WADI, where SFlexRCA achieves AC@1, AC@5, and AC@10 of 0.333, 0.289, and 0.378, respectively, while requiring only 35.3K parameters. Notably, this lower parameter count on WADI does not result from having fewer monitored variables: WADI contains substantially more variables than both BATADAL and SWAT. Rather, as discussed in RQ3, the WADI configuration does not require the linear-attention component used in the other configurations, further reducing the model size. Thus, the increase in the number of monitored variables does not necessarily translate into increased model complexity in SFlexRCA. This scalability advantage stems from SFlexRCA’s shared variable encoding, which avoids the per-variable neural architectures used by cMLP and cLSTM and therefore prevents the substantial parameter growth observed as the number of monitored variables increases. Similarly, GVAR and AERCA incur substantially larger models due to their dependency- and lag-specific architectures. On SWAT, SFlexRCA remains competitive, achieving an AC@5 of 0.367, higher than the strongest baseline value of 0.350, while using only 108.1K parameters. Compared with the recent CUTS+ framework, SFlexRCA further achieves higher AC@1, AC@5, and AC@10 on SWAT, with values of 0.233, 0.367, and 0.417 compared with 0.217, 0.300, and 0.342, respectively. Although CUTS+ uses fewer parameters than SFlexRCA on BATADAL and SWAT, its message-passing architecture introduces additional computational overhead that is not reflected by parameter count alone, as examined in RQ2.

Summary. Overall, these results highlight the central trade-off addressed by SFlexRCA: non-causal forecasting methods can become competitive when large volumes of normal-operation data are available, while causal methods can provide stronger dependency-aware localization but typically require substantially larger architectures. SFlexRCA combines dependency-aware modeling with a compact shared representation, achieving competitive or superior RCA accuracy with substantially fewer parameters, without requiring the per-variable, lag-specific, or explicit message-passing mechanisms of competing approaches, and while remaining effective in data-limited settings. The resulting training and inference efficiency are examined in RQ2.

D. RQ-2: Efficiency Analysis

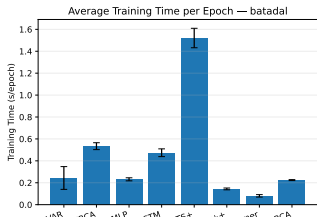
In this section, we evaluate the training and inference efficiency of SFlexRCA compared with causal deep learning baselines. These measurements characterize the training and deployment costs of the evaluated RCA methods under realistic edge-device constraints.

Training Efficiency

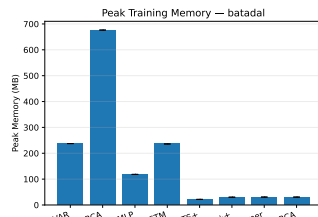
SFlexRCA exhibits low training overhead across all three datasets in terms of both training time and peak memory consumption. On BATADAL, SWAT, and WADI, SFlexRCA

TABLE IV: Root Cause Analysis Accuracy and Model Parameters Across BATADAL and SWAT and WADI (Mean $\pm$ SD).

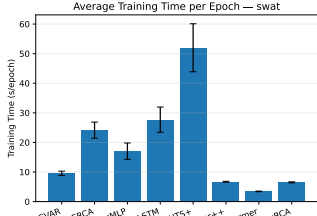
Method	BATADAL (K=16, P=43, training samples=8k)					SWAT (K=8, P=51, training samples=400k)					WADI (K=8, P=127, training samples=1.2M)				
	Params	AC@1	AC@5	AC@10	auc@10	Params	AC@1	AC@5	AC@10	auc@10	Params	AC@1	AC@5	AC@10	auc@10
Statistical Methods															
rcd	0	0.000 $\pm$ 0.00	0.086 $\pm$ 0.00	0.070 $\pm$ 0.00	0.091 $\pm$ 0.00	0	0.000 $\pm$ 0.00	0.000 $\pm$ 0.00	0.300 $\pm$ 0.00	0.100 $\pm$ 0.00	0	0.000 $\pm$ 0.00	0.000 $\pm$ 0.00	0.000 $\pm$ 0.00	0.000 $\pm$ 0.00
baro	0	0.286 $\pm$ 0.00	0.162 $\pm$ 0.00	0.221 $\pm$ 0.00	0.200 $\pm$ 0.00	0	0.000 $\pm$ 0.00	0.200 $\pm$ 0.00	0.250 $\pm$ 0.00	0.180 $\pm$ 0.00	0	0.267 $\pm$ 0.00	0.267 $\pm$ 0.00	0.356 $\pm$ 0.00	0.283 $\pm$ 0.00
torai	0	0.286 $\pm$ 0.00	0.178 $\pm$ 0.03	0.312 $\pm$ 0.04	0.241 $\pm$ 0.03	0	0.117 $\pm$ 0.03	0.200 $\pm$ 0.00	0.350 $\pm$ 0.00	0.227 $\pm$ 0.01	0	0.267 $\pm$ 0.00	0.267 $\pm$ 0.00	0.278 $\pm$ 0.00	0.269 $\pm$ 0.00
Non-Causal Methods															
TimeMixer++	576.6K	0.095 $\pm$ 0.08	0.121 $\pm$ 0.06	0.194 $\pm$ 0.06	0.125 $\pm$ 0.07	160.5K	0.167 $\pm$ 0.03	0.267 $\pm$ 0.03	0.367 $\pm$ 0.04	0.272 $\pm$ 0.04	199.9K	0.267 $\pm$ 0.00	0.304 $\pm$ 0.06	0.326 $\pm$ 0.04	0.302 $\pm$ 0.05
iTransformer	800.0K	0.048 $\pm$ 0.08	0.073 $\pm$ 0.04	0.180 $\pm$ 0.03	0.096 $\pm$ 0.04	201.4K	0.167 $\pm$ 0.03	0.250 $\pm$ 0.09	0.408 $\pm$ 0.10	0.281 $\pm$ 0.07	201.4K	0.378 $\pm$ 0.04	0.344 $\pm$ 0.04	0.370 $\pm$ 0.01	0.354 $\pm$ 0.03
Causal Methods															
cMLP	7.1M	0.286 $\pm$ 0.00	0.114 $\pm$ 0.00	0.142 $\pm$ 0.01	0.144 $\pm$ 0.01	2.3M	0.100 $\pm$ 0.00	0.300 $\pm$ 0.05	0.533 $\pm$ 0.03	0.328 $\pm$ 0.03	14.5M	0.267 $\pm$ 0.00	0.267 $\pm$ 0.00	0.289 $\pm$ 0.04	0.276 $\pm$ 0.02
cLSTM	13.3M	0.381 $\pm$ 0.08	0.181 $\pm$ 0.04	0.190 $\pm$ 0.05	0.216 $\pm$ 0.05	4.7M	0.300 $\pm$ 0.00	0.333 $\pm$ 0.03	0.383 $\pm$ 0.06	0.337 $\pm$ 0.03	16.7M	0.311 $\pm$ 0.04	0.271 $\pm$ 0.01	0.304 $\pm$ 0.03	0.279 $\pm$ 0.00
GVAR	14.2M	0.143 $\pm$ 0.14	0.181 $\pm$ 0.04	0.261 $\pm$ 0.03	0.197 $\pm$ 0.05	3.2M	0.283 $\pm$ 0.03	0.350 $\pm$ 0.00	0.375 $\pm$ 0.03	0.340 $\pm$ 0.01	14.8M	0.067 $\pm$ 0.00	0.122 $\pm$ 0.10	0.137 $\pm$ 0.12	0.107 $\pm$ 0.08
AERCA	42.6M	0.286 $\pm$ 0.00	0.143 $\pm$ 0.03	0.137 $\pm$ 0.04	0.146 $\pm$ 0.02	9.6M	0.150 $\pm$ 0.00	0.317 $\pm$ 0.03	0.450 $\pm$ 0.05	0.338 $\pm$ 0.03	44.4M	0.067 $\pm$ 0.00	0.060 $\pm$ 0.05	0.059 $\pm$ 0.04	0.061 $\pm$ 0.04
CUTS+	265.8K	0.143 $\pm$ 0.14	0.152 $\pm$ 0.07	0.262 $\pm$ 0.06	0.187 $\pm$ 0.05	91.6K	0.217 $\pm$ 0.03	0.300 $\pm$ 0.05	0.342 $\pm$ 0.12	0.287 $\pm$ 0.06	163.6K	0.133 $\pm$ 0.00	0.244 $\pm$ 0.02	0.278 $\pm$ 0.04	0.237 $\pm$ 0.01
Proposed Method															
SFlexRCA	279.4K	0.476 $\pm$ 0.08	0.162 $\pm$ 0.02	0.213 $\pm$ 0.01	0.234 $\pm$ 0.01	108.1K	0.233 $\pm$ 0.03	0.367 $\pm$ 0.08	0.417 $\pm$ 0.06	0.347 $\pm$ 0.04	35.3K	0.333 $\pm$ 0.00	0.289 $\pm$ 0.02	0.378 $\pm$ 0.02	0.310 $\pm$ 0.01



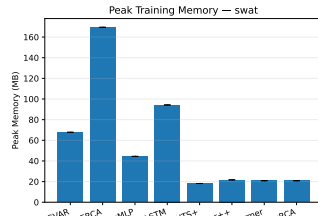
(a) Avg training time per epoch (BATADAL).



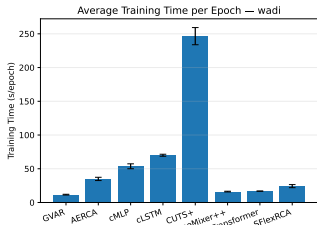
(b) Peak training memory (BATADAL).



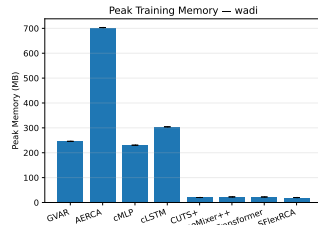
(c) Avg training time per epoch (SWaT).



(d) Peak training memory (SWaT).



(e) Avg training time per epoch (WADI).



(f) Peak training memory (WADI).

Fig. 2: Training efficiency metrics (average training time per epoch and peak training memory) across BATADAL, SWaT, and WADI datasets.

requires 0.224, 6.51, and 24.24 s/epoch, respectively, while peak memory remains below 31 MB. Compared with the causal RCA methods, SFlexRCA generally provides lower training overhead. This is particularly evident on SWAT, where SFlexRCA requires 6.51 s/epoch, compared with 9.56 s for GVAR, 24.15 s for AERCA, 17.05 s for cMLP, 27.70 s

for cLSTM, and 51.99 s for CUTS+. Its peak memory consumption is only 20.86 MB, substantially lower than GVAR (67.75 MB), AERCA (169.46 MB), cMLP (44.41 MB), and cLSTM (94.25 MB). Although CUTS+ uses only 18.15 MB, its message-passing architecture incurs considerably higher training time. The non-causal models, TimeMixer++ and iTransformer, achieve the lowest training overhead overall. On SWAT, however, SFlexRCA remains highly competitive with these models, requiring 6.51 s/epoch compared with 6.57 s for TimeMixer++, while using slightly less memory (20.86 MB versus 21.71 MB). iTransformer achieves lower training time at 3.42 s/epoch, with a comparable memory footprint of 20.88 MB. Nevertheless, the forecasting-based formulation of these non-causal models typically benefits from sufficient normal-operation data, which may be limited in IIoT deployments. This limitation is particularly relevant to BATADAL, where observations are sampled at hourly intervals. As shown in RQ1, SFlexRCA can provide more effective RCA under such data-limited settings while retaining a compact training footprint. Overall, SFlexRCA provides a favorable balance between training efficiency and RCA effectiveness, combining the low resource requirements of non-causal models with substantially lower training overhead than most causal RCA architectures.

Edge Inference Efficiency Inference experiments further demonstrate the suitability of SFlexRCA for resource-constrained IIoT edge devices. As expected, the Raspberry Pi 3 consistently incurs higher latency than the Raspberry Pi 5. On the Pi 3, SFlexRCA requires 149.8, 62.3, and 64.3 ms/sample on BATADAL, SWAT, and WADI, respectively. It achieves the lowest latency on SWAT and WADI, with the advantage becoming particularly pronounced on WADI, where it is substantially faster than GVAR (178.4 ms), cMLP (310.2 ms), CUTS+ (620.7 ms), AERCA (642.2 ms), and cLSTM (986.0 ms). On the Pi 5, SFlexRCA requires only 20.3, 8.2, and 9.3 ms/sample on BATADAL, SWAT, and WADI, respectively. In particular, SFlexRCA achieves the lowest latency among the causal RCA methods on all three datasets and is the overall fastest method on WADI. The non-causal transformer models provide a useful comparison. On the Raspberry Pi 3, SFlexRCA achieves lower inference

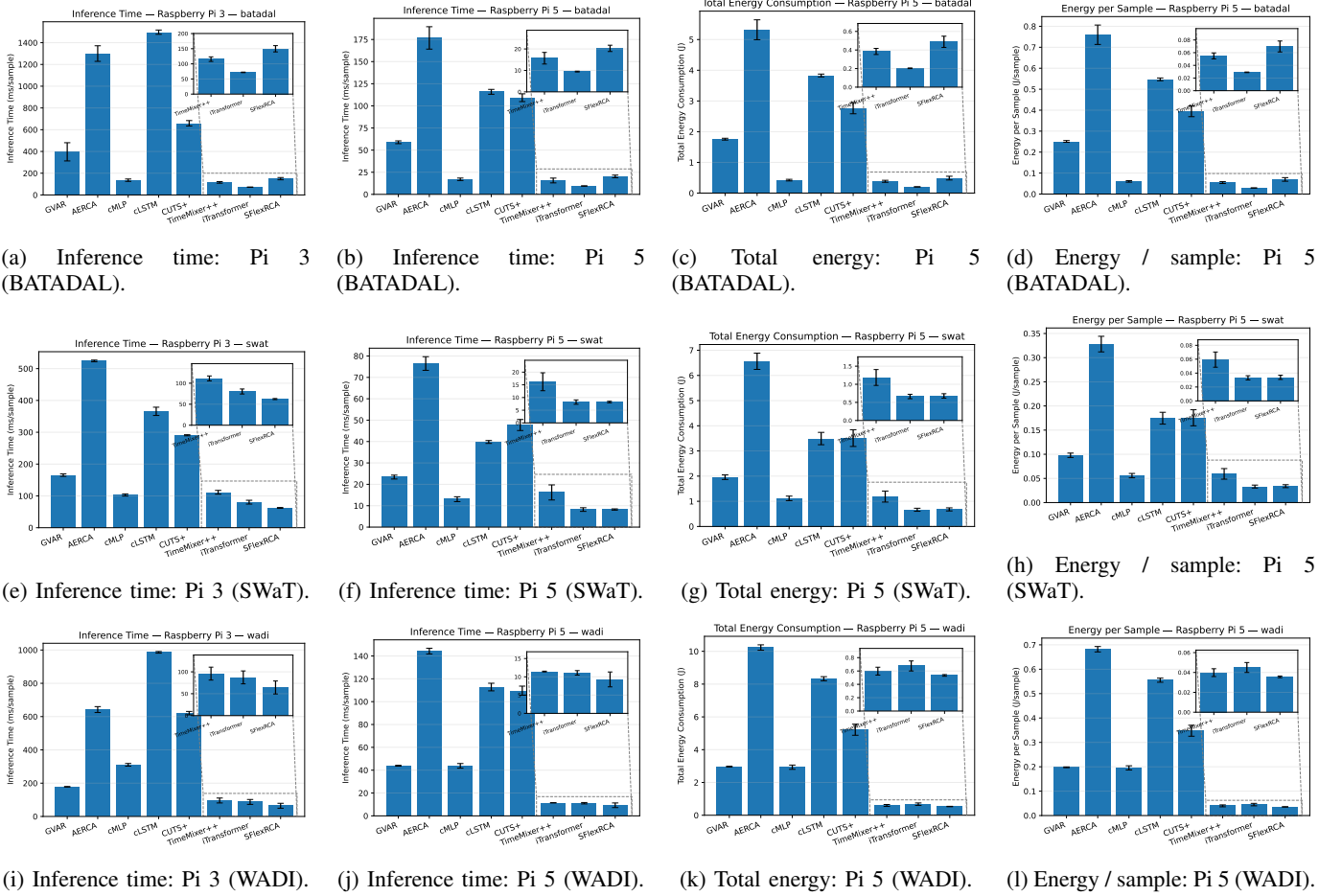


Fig. 3: Edge deployment efficiency metrics (inference latency on Pi 3/5 and energy consumption on Pi 5) across BATADAL, SWaT, and WADI datasets.

latency than both transformer models on SWAT and WADI, requiring 62.3 and 64.3 ms/sample compared with 80.3 and 87.3 ms/sample for iTransformer, respectively. On BATADAL, iTransformer is faster, requiring 71.8 ms/sample compared with 149.8 ms/sample for SFlexRCA. Similar behavior is observed on the Pi 5: iTransformer achieves the lowest latency on BATADAL (9.5 ms/sample) and is marginally faster than SFlexRCA on SWAT, whereas SFlexRCA becomes the fastest method on WADI, requiring only 9.3 ms/sample compared with 11.1 ms for iTransformer and 11.4 ms for TimeMixer++. These results indicate that SFlexRCA remains highly competitive with lightweight transformer architectures while providing more accurate RCA for systems with limited normal-operation data, as shown in RQ1.

Edge Energy Consumption Energy consumption is evaluated on the Raspberry Pi 5, where reliable power measurements are available. SFlexRCA consumes 0.488, 0.678, and 0.534 J per evaluation on BATADAL, SWAT, and WADI, respectively, corresponding to 0.070, 0.034, and 0.036 J/sample. SFlexRCA achieves the lowest energy consumption on WADI (0.036 J/sample), while on SWAT it is essentially tied with iTransformer (0.034 versus 0.033 J/sample). On BATADAL, the non-causal transformer models achieve lower energy consumption, with iTransformer requiring only

0.029 J/sample and TimeMixer++ 0.055 J/sample, compared with 0.070 J/sample for SFlexRCA. This advantage is partly attributable to BATADAL having the smallest number of monitored sensors ($P = 43$), which reduces the computational cost of the transformer-based models. However, as shown in RQ1, these lower energy costs do not translate into better RCA performance. Thus, SFlexRCA provides a favorable accuracy–efficiency trade-off across both causal and non-causal methods, placing it on the Pareto frontier rather than optimizing efficiency at the expense of RCA effectiveness. This trade-off is particularly relevant for continuous IIoT deployment, where both diagnostic accuracy and energy consumption accumulate over repeated inference.

Summary. Overall, the results show that SFlexRCA provides a favorable accuracy–efficiency trade-off for edge-based IIoT RCA. Rather than optimizing only model size or individual inference latency, SFlexRCA maintains low training and inference overhead while preserving effective RCA across systems with substantially different numbers of monitored variables (BATADAL: 43, SWaT: 51, WADI: 127) and temporal characteristics (Table III). Although non-causal transformer models can achieve lower computational and energy costs on smaller systems such as BATADAL, their lower RCA accuracy limits this efficiency advantage in practice. In contrast,

SFlexRCA remains competitive with these efficient models while substantially reducing the overhead of most causal architectures. This balance places SFlexRCA on the accuracy–efficiency Pareto frontier across the evaluated methods and supports its use for continuous, on-device RCA in resource-constrained IIoT environments.

E. RQ-3: Ablation Study

We evaluate the contribution of the major loss and architectural components of SFlexRCA through ablation studies on BATADAL, SWaT, and WADI. To isolate the effect of the loss formulation, all loss ablations use a common reference architecture with the linear prediction head and linear attention enabled. Starting from this fixed architecture, we remove one loss component at a time while keeping all other architectural and training settings unchanged. Architectural ablations are conducted separately to examine the contribution of the orthogonal representation, temporal mixer, linear attention, and prediction head and to determine the dataset-specific configurations reported in Table IV. Thus, the parameter counts in the loss-ablation results correspond to the fixed reference architecture rather than necessarily to the final selected configuration for each dataset.

Contribution of Loss Components. The loss ablations show that the sparse and smoothness losses are particularly important for the more complex SWaT and WADI datasets. Removing the sparse loss reduces AC@1 from 0.233 to 0.217 on SWaT and from 0.333 to 0.133 on WADI, indicating that sparsity helps isolate the variables most strongly associated with failures. Similarly, removing the smoothness loss reduces AC@1 to 0.167 on SWaT and 0.133 on WADI, suggesting that temporally stable interaction estimates become more important as temporal dependencies become more pronounced. The KL loss has its strongest effect on WADI, where removing it reduces AC@10 from 0.378 to 0.293. Its effect on BATADAL is limited: removing the KL term preserves AC@1 at 0.476 and produces comparable overall performance, consistent with the relatively simpler dependency structure of BATADAL, therefore in our experiments we maintain the KL loss for WADI and SWaT but remove it for BATADAL. Overall, the results indicate that the loss components are complementary, with their importance depending on the temporal and structural complexity of the dataset.

Contribution of Architectural Components. The architectural ablations highlight the complementary roles of orthogonal representations, temporal mixing, linear attention, and the prediction head. Removing the orthogonal coefficients consistently degrades performance across the three datasets, particularly on BATADAL and WADI, supporting their importance for constructing informative variable interactions. The temporal mixer primarily benefits top-ranked localization: removing it reduces AC@1 from 0.233 to 0.133 on SWaT and from 0.333 to 0.133 on WADI, indicating that temporal mixing provides useful information for distinguishing variables associated with failures. The contribution of linear attention is dataset-dependent. While BATADAL and SWAT retain linear attention in their selected configurations,

WADI achieves its best overall RCA performance without it, demonstrating that SFlexRCA can achieve effective localization using its core orthogonal representation and lightweight prediction components alone. This architectural flexibility also supports scalability across systems with different monitoring dimensionalities: although WADI has the largest number of monitored sensors ($P = 127$), its selected configuration requires only 35.3K parameters. Rather than imposing increasing architectural complexity as the number of sensors grows, SFlexRCA can adapt its components to the characteristics of each system, highlighting its flexibility and scalability across heterogeneous IIoT environments. The prediction head also exhibits dataset-dependent behavior. The linear prediction head is used as the default configuration because it provides a more compact model, while the MLP prediction head is considered as an alternative in the ablation study. Accordingly, Pred (Linear) denotes the default linear prediction head with linear attention retained, whereas Pred (MLP) replaces the prediction head with an MLP while keeping linear attention enabled. The w/o L-Attn configuration, in contrast, removes the linear attention module while retaining the default linear prediction head. Based on the results, the linear prediction head is used for BATADAL and WADI, whereas the MLP prediction head provides the selected configuration for SWAT. Overall, these results highlight the flexibility of SFlexRCA to adapt its architecture to dataset-specific characteristics while balancing RCA accuracy and model complexity.

Summary. Overall, the ablation results demonstrate that SFlexRCA provides both **architectural flexibility** and **scalability**. Its components can be selectively enabled or simplified according to dataset characteristics, allowing RCA accuracy to be balanced against model complexity. This adaptability enables SFlexRCA to remain compact across systems with different monitoring dimensionalities, making it well suited to resource-constrained edge environments.

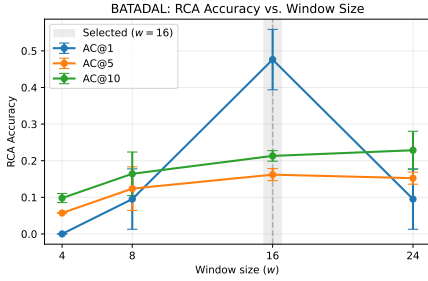
F. RQ-4: Sensitivity to Temporal Context

We further examine the sensitivity of SFlexRCA to temporal window size to determine how much historical context is required for effective RCA. The results show that the optimal temporal context is dataset-dependent. We evaluate windows of 4, 8, 12, and 16 for SWaT and WADI (4–16 seconds), and 4, 8, 16, and 24 for BATADAL (4–24 hours), reflecting their different sampling frequencies as summarized in Table III.

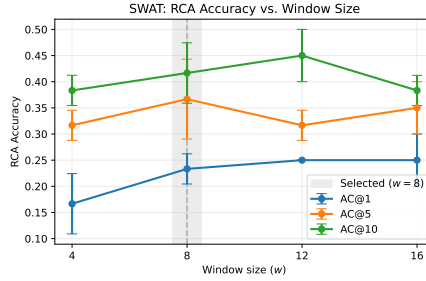
Effect of Temporal Context. On BATADAL, increasing the window size from 4 to 16 improves AC@1 from 0.0 to 0.4762, the highest value observed. Although AC@1 decreases at larger windows, AC@5 remains stable from window 8 onward and AC@10 continues improving through window 24. We therefore select a window size of 16 based on its superior top-ranked localization performance. SWaT exhibits a similar benefit from additional context, with AC@1 increasing from 0.167 at a window of 4 to 0.233 at 8 and reaching 0.250 at 12 and 16. However, the window of 8 already provides a strong performance level while requiring less historical context; we therefore select the earliest window achieving competitive performance to reduce the amount of context required for RCA.

TABLE V: Case-Based Ablation Study of Architectural Components (Mean $\pm$ SD). Boxed cells ($\boxed{\dots}$) indicate selected configurations.

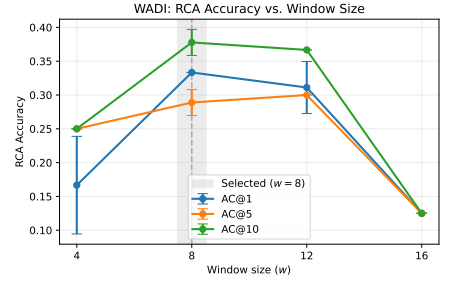
Method	BATADAL					SWAT					WADI				
	Params	AC@1	AC@5	AC@10	auc@10	Params	AC@1	AC@5	AC@10	auc@10	Params	AC@1	AC@5	AC@10	auc@10
<i>Loss Ablations</i>															
w/o Sparse	279.4K	0.381 $\pm$ 0.08	0.181 $\pm$ 0.02	0.247 $\pm$ 0.04	0.238 $\pm$ 0.03	74.2K	0.217 $\pm$ 0.03	0.333 $\pm$ 0.03	0.400 $\pm$ 0.00	0.338 $\pm$ 0.02	84.5K	0.133 $\pm$ 0.00	0.278 $\pm$ 0.02	0.319 $\pm$ 0.01	0.265 $\pm$ 0.01
w/o Smooth	279.4K	0.381 $\pm$ 0.08	0.181 $\pm$ 0.02	0.247 $\pm$ 0.04	0.238 $\pm$ 0.03	74.2K	0.167 $\pm$ 0.03	0.383 $\pm$ 0.03	0.433 $\pm$ 0.06	0.353 $\pm$ 0.01	84.5K	0.133 $\pm$ 0.00	0.289 $\pm$ 0.02	0.404 $\pm$ 0.03	0.294 $\pm$ 0.02
w/o KL	279.4K	0.476 $\pm$ 0.08	0.162 $\pm$ 0.02	0.213 $\pm$ 0.01	0.234 $\pm$ 0.01	74.2K	0.217 $\pm$ 0.03	0.400 $\pm$ 0.05	0.433 $\pm$ 0.03	0.367 $\pm$ 0.04	84.5K	0.133 $\pm$ 0.00	0.271 $\pm$ 0.01	0.293 $\pm$ 0.04	0.266 $\pm$ 0.02
<i>Architecture Ablations</i>															
w/o Orth	279.4K	0.429 $\pm$ 0.00	0.143 $\pm$ 0.03	0.209 $\pm$ 0.02	0.210 $\pm$ 0.02	74.2K	0.217 $\pm$ 0.03	0.333 $\pm$ 0.06	0.383 $\pm$ 0.03	0.322 $\pm$ 0.03	84.5K	0.133 $\pm$ 0.00	0.278 $\pm$ 0.02	0.304 $\pm$ 0.01	0.264 $\pm$ 0.01
w/o Mixer	278.4K	0.429 $\pm$ 0.00	0.162 $\pm$ 0.02	0.213 $\pm$ 0.01	0.227 $\pm$ 0.01	73.7K	0.133 $\pm$ 0.03	0.400 $\pm$ 0.05	0.483 $\pm$ 0.06	0.365 $\pm$ 0.03	84.0K	0.133 $\pm$ 0.00	0.300 $\pm$ 0.03	0.389 $\pm$ 0.05	0.294 $\pm$ 0.02
w/o L-Attn	82.8K	0.429 $\pm$ 0.00	0.181 $\pm$ 0.02	0.223 $\pm$ 0.02	0.236 $\pm$ 0.02	25.1K	0.100 $\pm$ 0.00	0.500 $\pm$ 0.00	0.550 $\pm$ 0.00	0.430 $\pm$ 0.01	35.3K	0.333 $\pm$ 0.00	0.289 $\pm$ 0.02	0.378 $\pm$ 0.02	0.310 $\pm$ 0.01
Pred (MLP)	414.9K	0.429 $\pm$ 0.00	0.181 $\pm$ 0.02	0.188 $\pm$ 0.04	0.225 $\pm$ 0.02	108.1K	0.233 $\pm$ 0.03	0.367 $\pm$ 0.08	0.417 $\pm$ 0.06	0.347 $\pm$ 0.04	118.4K	0.133 $\pm$ 0.00	0.267 $\pm$ 0.03	0.315 $\pm$ 0.05	0.261 $\pm$ 0.02
Pred (Linear)	279.4K	0.476 $\pm$ 0.08	0.162 $\pm$ 0.02	0.213 $\pm$ 0.01	0.234 $\pm$ 0.01	74.2K	0.217 $\pm$ 0.06	0.333 $\pm$ 0.03	0.417 $\pm$ 0.03	0.322 $\pm$ 0.01	84.5K	0.133 $\pm$ 0.00	0.260 $\pm$ 0.01	0.300 $\pm$ 0.02	0.254 $\pm$ 0.01



(a) BATADAL: RCA vs. window size.



(b) SWaT: RCA vs. window size.



(c) WADI: RCA vs. window size.

Fig. 4: Sensitivity analysis of RCA accuracy with respect to window size for BATADAL, SWaT and WADI datasets.

WADI exhibits a different behavior: performance improves from a window of 4 to 8, reaching AC@1 of 0.333 and AC@10 of 0.378, but decreases substantially at a window of 16, where all three metrics fall to 0.125. This suggests that excessive temporal context can introduce less relevant historical information and make it more difficult to isolate the measurements associated with the current failure. Therefore, we select a window of 8 for WADI to balance the benefits of temporal context with the risk of including less relevant historical information.

Summary. Overall, the sensitivity analysis demonstrates that SFlexRCA benefits from dataset-specific temporal context rather than simply favoring larger windows. BATADAL benefits from a longer intermediate context, while SWaT achieves strong performance with an earlier window size and WADI favors a shorter context. This supports the flexibility of SFlexRCA to adapt its temporal receptive field to the dynamics of the monitored system while avoiding unnecessary historical context.

G. RQ-5: Effect of Dataset-Specific Orthogonal Projections

We evaluate the effect of the proposed dataset-specific orthogonal projection by replacing it with alternative representations while keeping the remaining SFlexRCA architecture unchanged. Specifically, we compare the proposed projection against no projection (i.e., none), a learned projection, and several fixed basis transformations, including Legendre, Laguerre, Chebyshev, Hermite, and Fourier representations following the experimental methodology of [44], [38].

Comparison of Projection Methods: Table VI shows that the dataset-specific orthogonal projection provides the strongest overall RCA performance. On BATADAL, the proposed projection achieves the highest AC@1, AC@10, and AUC@10, reaching 0.476, 0.213, and 0.234, respectively. On SWaT, it ties for the highest AC@1 and AC@5. The gains are more modest on WADI, where the different projections produce similar results, but the proposed projection remains competitive across all metrics. The observed differences are consistent with how the projection is constructed. Unlike fixed bases, the proposed transformation is derived from the temporal covariance structure of normal operating data for each dataset. Consequently, the resulting orthogonal coordinates are adapted to the characteristic dynamics of the monitored IIoT system rather than imposing a universal temporal basis. This distinction is particularly relevant because the datasets differ substantially in sampling frequency and temporal characteristics (Table III). BATADAL is sampled hourly, whereas SWaT and WADI are sampled at one-second intervals; therefore, the temporal correlations captured by the learned basis can differ substantially across systems. The results also provide evidence for the importance of orthogonality in addition to dataset adaptation. The learned projection is dataset-adaptive but does not impose an orthogonal basis, whereas the proposed projection combines dataset adaptation with orthogonality. The proposed method generally outperforms the learned projection on BATADAL and SWaT, while achieving competitive or superior performance relative to the fixed orthogonal bases. This suggests that the observed benefit is not solely due to

TABLE VI: Root Cause Analysis Accuracy Across Different Projections For BATADAL and SWAT and WADI (Mean $\pm$ SD).

Method	BATADAL				SWAT				WADI			
	AC@1	AC@5	AC@10	auc@10	AC@1	AC@5	AC@10	auc@10	AC@1	AC@5	AC@10	auc@10
Baselines												
none	0.429 $\pm$ 0.00	0.143 $\pm$ 0.03	0.209 $\pm$ 0.02	0.210 $\pm$ 0.02	0.217 $\pm$ 0.06	0.317 $\pm$ 0.03	0.400 $\pm$ 0.00	0.308 $\pm$ 0.01	0.333 $\pm$ 0.00	0.289 $\pm$ 0.02	0.367 $\pm$ 0.00	0.309 $\pm$ 0.00
learned	0.381 $\pm$ 0.08	0.149 $\pm$ 0.03	0.177 $\pm$ 0.05	0.193 $\pm$ 0.05	0.217 $\pm$ 0.03	0.367 $\pm$ 0.03	0.417 $\pm$ 0.03	0.360 $\pm$ 0.02	0.333 $\pm$ 0.00	0.289 $\pm$ 0.02	0.367 $\pm$ 0.00	0.308 $\pm$ 0.00
Static Projections												
legendre	0.381 $\pm$ 0.08	0.152 $\pm$ 0.04	0.178 $\pm$ 0.03	0.204 $\pm$ 0.03	0.200 $\pm$ 0.00	0.333 $\pm$ 0.03	0.400 $\pm$ 0.05	0.318 $\pm$ 0.01	0.333 $\pm$ 0.00	0.289 $\pm$ 0.02	0.367 $\pm$ 0.00	0.307 $\pm$ 0.01
laguerre	0.381 $\pm$ 0.08	0.162 $\pm$ 0.03	0.178 $\pm$ 0.03	0.208 $\pm$ 0.03	0.200 $\pm$ 0.00	0.300 $\pm$ 0.05	0.400 $\pm$ 0.00	0.303 $\pm$ 0.02	0.333 $\pm$ 0.00	0.289 $\pm$ 0.02	0.367 $\pm$ 0.00	0.304 $\pm$ 0.01
chebyshev	0.333 $\pm$ 0.08	0.143 $\pm$ 0.03	0.190 $\pm$ 0.05	0.192 $\pm$ 0.06	0.233 $\pm$ 0.03	0.317 $\pm$ 0.03	0.433 $\pm$ 0.03	0.337 $\pm$ 0.01	0.333 $\pm$ 0.00	0.300 $\pm$ 0.00	0.367 $\pm$ 0.00	0.310 $\pm$ 0.01
hermite	0.381 $\pm$ 0.08	0.152 $\pm$ 0.04	0.178 $\pm$ 0.03	0.204 $\pm$ 0.03	0.167 $\pm$ 0.03	0.333 $\pm$ 0.08	0.400 $\pm$ 0.05	0.317 $\pm$ 0.03	0.333 $\pm$ 0.00	0.300 $\pm$ 0.00	0.367 $\pm$ 0.00	0.317 $\pm$ 0.00
fourier	0.429 $\pm$ 0.00	0.181 $\pm$ 0.04	0.200 $\pm$ 0.01	0.223 $\pm$ 0.02	0.217 $\pm$ 0.03	0.333 $\pm$ 0.03	0.450 $\pm$ 0.05	0.342 $\pm$ 0.02	0.333 $\pm$ 0.00	0.289 $\pm$ 0.02	0.389 $\pm$ 0.04	0.311 $\pm$ 0.01
Dataset Specific Projection												
orthogonal (proposed)	0.476 $\pm$ 0.08	0.162 $\pm$ 0.02	0.213 $\pm$ 0.01	0.234 $\pm$ 0.01	0.233 $\pm$ 0.03	0.367 $\pm$ 0.08	0.417 $\pm$ 0.06	0.347 $\pm$ 0.04	0.333 $\pm$ 0.00	0.289 $\pm$ 0.02	0.378 $\pm$ 0.02	0.310 $\pm$ 0.01

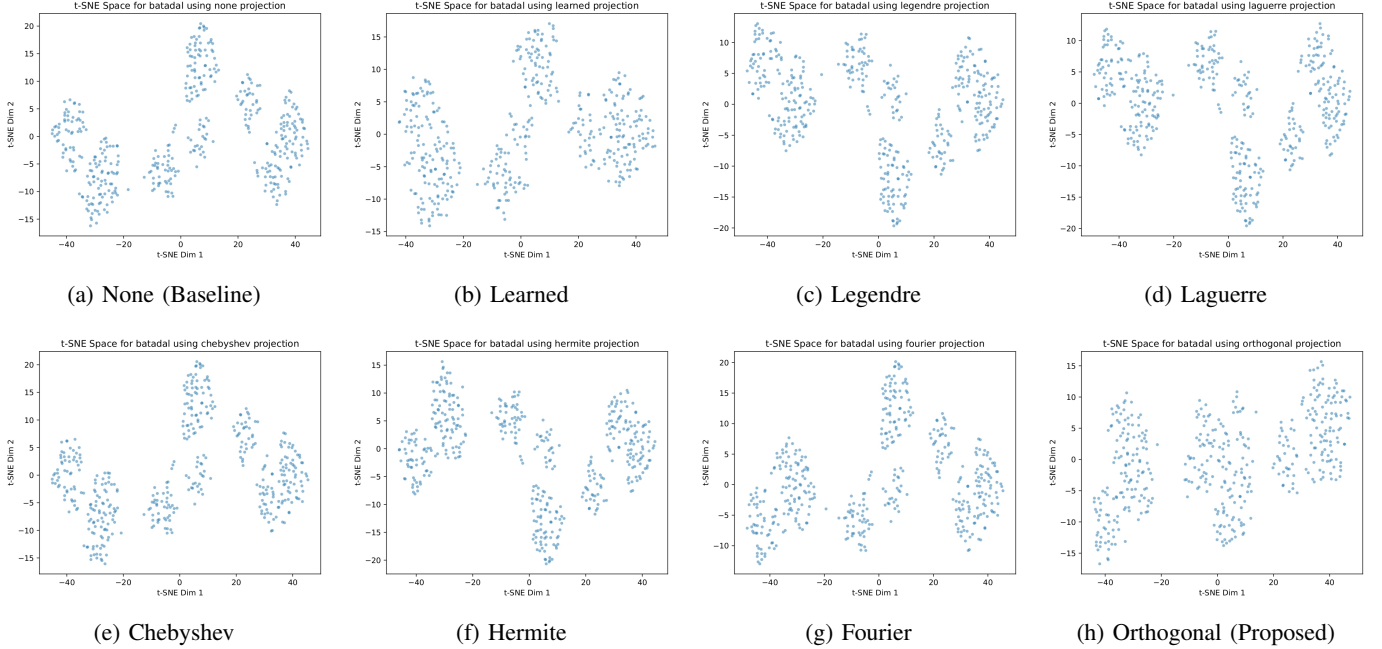


Fig. 5: t-SNE latent feature projections on BATADAL corresponding to baseline methods, static polynomial/frequency projections, and our proposed orthogonal projection (matching Table VI).

adapting the projection to the dataset, but also to organizing the adapted representation into orthogonal components. This property is particularly relevant to IIoT telemetry, where sensor measurements can exhibit substantial redundancy. The orthogonal basis reduces such redundancy while preserving the information captured by the normal-operation covariance structure, providing a more separated representation for subsequent latent modeling and residual-based localization.

TSNE Visualization of Projections: To visually inspect how different transformations structure the feature space, Figure 5 presents the t-SNE manifold projections of the latent representations on BATADAL across baseline, static polynomial/frequency, and proposed orthogonal projections following the experimental methodology of [45]. As shown in Figures 5a and 5b, both the unprojected baseline (*none*) and the *learned* non-orthogonal projection produce fragmented and dispersed feature clusters. This structural dispersion mirrors their lower localization accuracy in Table VI, where non-orthogonal learned features fail to maintain distinct boundaries between normal operational modes and anomalous deviations.

In contrast, static orthogonal polynomial bases (Legendre, Laguerre, Chebyshev, Hermite) and frequency transformations (Fourier) shown in Figures 5c–5g enforce a more structured latent topology. However, because these bases impose fixed, system-agnostic assumptions over time, the resulting clusters remain partially overlapping or stretched across non-linearly correlated dimensions. Our proposed dataset-specific orthogonal projection (Figure 5h) achieves the highest degree of cluster compactness and inter-cluster separation. By constructing an orthogonal basis directly from the system’s empirical covariance structure, it effectively decorrelates redundant sensor channels while preserving global system dynamics. This visual compactness directly corroborates the superior top- k accuracy (AC@1 = 0.476, AUC@10 = 0.234) observed on BATADAL, confirming that joint dataset adaptation and strict orthogonality yield cleaner decision spaces for residual localization.

Summary: In summary, the empirical metrics and qualitative t-SNE visualizations jointly validate the core design of our proposed representation. While dataset-specific adaptation tailors the transformation to the unique sampling frequencies and

temporal dynamics of individual IIoT deployments, orthogonality enforces strict feature decorrelation across redundant sensor channels. Geometrically, this dual constraint eliminates cluster dispersion and produces tight, well-separated latent representations—preventing failure modes from bleeding into normal operating regions. For IIoT systems characterized by high-dimensional telemetry and inter-variable coupling, this structural preservation is essential: it ensures that subtle anomaly residuals remain localized and distinct, thereby delivering superior top- k diagnostic precision over both static, system-agnostic bases and unconstrained learned projections.

V. THREATS TO VALIDITY

Internal Validity: Potential threats arise from implementation choices, hyperparameter tuning, and differences in the optimization requirements of competing RCA methods. We mitigate these effects through a consistent evaluation pipeline, multiple random seeds, and ablation and sensitivity analyses.

External Validity: We evaluate SFlexRCA on BATADAL, SWAT, and WADI, which cover different telemetry dimensionalities, temporal characteristics, and training-data regimes. However, these benchmarks may not capture the full diversity of real IIoT deployments, including noisier telemetry, incomplete fault labels, and evolving system behavior. Results on Raspberry Pi hardware may also not generalize to other edge platforms or deployment configurations.

Construct Validity: RCA effectiveness is measured primarily using ranking-based metrics (AC@1, AC@5, AC@10, and AUC@10), which do not directly capture diagnostic interpretability or remediation effectiveness. Training time, memory, latency, and energy consumption provide practical proxies for edge efficiency but do not capture all deployment costs, such as communication, storage, and orchestration overhead. Comparisons between causal and non-causal approaches may also depend on dataset characteristics and training-data availability.

VI. CONCLUSION

We presented SFlexRCA, a lightweight, scalable, and flexible root cause analysis framework for IIoT telemetry that avoids explicit dependency graph construction and message passing. Across BATADAL, SWAT, and WADI, SFlexRCA achieved competitive or superior RCA accuracy while maintaining a compact architecture. Its parameter requirements decreased on WADI despite having the largest number of monitored variables, demonstrating that SFlexRCA can adapt to different telemetry dimensionalities without parameter growth proportional to the number of variables. The efficiency evaluation further demonstrated low training memory and computational overhead and efficient inference on Raspberry Pi 3 and Raspberry Pi 5. The results highlight an important trade-off in IIoT RCA. Non-causal forecasting-based approaches can provide efficient modeling but may require substantial normal-operation data, limiting their effectiveness under data-constrained regimes. Causal and dependency-aware

approaches can capture relationships underlying fault propagation but may introduce substantial parameter and computational overhead through explicit dependency modeling, per-variable networks, or message passing. SFlexRCA addresses this trade-off by using orthogonal representations and shared lightweight linear modeling to capture cross-variable and temporal structure without explicitly constructing a dependency graph. This enables effective RCA under limited training data while avoiding parameter growth with increasing telemetry dimensionality. The ablation and sensitivity analyses further demonstrate that the architecture, loss formulation, and temporal context can be adapted to different dataset characteristics. Representation comparisons and t-SNE visualization provide additional evidence that the learned orthogonal representations capture informative and separable structure for root cause localization. Overall, SFlexRCA provides a practical balance among RCA accuracy, training-data requirements, model complexity, and edge computational cost, enabling topology-free and resource-efficient root cause localization for IIoT systems.

REFERENCES

- [1] Z. Zeng, X. Wang, L. Xu, X. Ye, J. Hu, F. H. Kumbhar, and K. Dev, “Cadis: Causality-driven transformer for anomaly detection and root cause diagnosis in industrial internet of things,” *IEEE Internet of Things Journal*, vol. 13, no. 10, pp. 20 047–20 060, 2026.
- [2] L. Chen, X. Liu, Y. Zou, J. Tang, C. Liu, B. Hu, and M. Lv, “Frequency-domain spectrum discrepancy-based fast anomaly detection for iiot sensor time-series signals,” *IEEE Transactions on Instrumentation and Measurement*, vol. 74, pp. 1–16, 2025.
- [3] L. Chen, J. Tang, Y. Zou, X. Liu, X. Xie, and G. Deng, “Lightweight and fast time-series anomaly detection via point-level and sequence-level reconstruction discrepancy,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 36, no. 9, pp. 17 295–17 309, 2025.
- [4] M. Li, S. Liu, H. Qian, and A. Zhou, “A simple yet scalable granger causal structural learning approach for topological event sequences,” in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. [Online]. Available: <https://openreview.net/forum?id=mP084aMFsd>
- [5] M. Rabbani, J. Gui, F. Nejati, Z. Zhou, A. Kaniyamattam, M. Mirani, G. Piya, I. Opushnyev, R. Lu, and A. A. Ghorbani, “Device identification and anomaly detection in iot environments,” *IEEE Internet of Things Journal*, vol. 12, no. 10, pp. 13 625–13 643, 2025.
- [6] H. Huang, P. Wang, J. Pei, J. Wang, S. Alexanian, and D. Niyato, “Deep learning advancements in anomaly detection: A comprehensive survey,” *IEEE Internet of Things Journal*, vol. 12, no. 21, pp. 44 318–44 342, 2025.
- [7] L. Pham, H. Ha, and H. Zhang, “Root cause analysis for microservice system based on causal inference: How far are we?” in *Proceedings of the 39th IEEE/ACM International Conference on Automated Software Engineering*, ser. ASE ’24. New York, NY, USA: Association for Computing Machinery, 2024, p. 706–715. [Online]. Available: <https://doi.org/10.1145/3691620.3695065>
- [8] Y. Cheng, L. Li, T. Xiao, Z. Li, J. Suo, K. He, and Q. Dai, “Cuts+: high-dimensional causal discovery from irregular time-series,” in *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, ser. AAAI’24/IAAI’24/EAAI’24. AAAI Press, 2024. [Online]. Available: <https://doi.org/10.1609/aaai.v38i10.29034>
- [9] C. Lee, T. Yang, Z. Chen, Y. Su, and M. R. Lyu, “Eadro: An end-to-end troubleshooting framework for microservices on multi-source data,” *arXiv preprint arXiv:2302.05092*, 2023.
- [10] Y. Sun, B. Shi, M. Mao, M. Ma, S. Xia, S. Zhang, and D. Pei, “Art: A unified unsupervised framework for incident management in microservice systems,” in *2024 39th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 2024, pp. 1183–1194.

- [11] J. Huang, Y. Yang, H. Yu, J. Li, and X. Zheng, "Twin Graph-Based Anomaly Detection via Attentive Multi-Modal Learning for Microservice System," in *2023 38th IEEE/ACM International Conference on Automated Software Engineering (ASE)*. Los Alamitos, CA, USA: IEEE Computer Society, Sep. 2023, pp. 66–78. [Online]. Available: <https://doi.ieeecomputersociety.org/10.1109/ASE56229.2023.00138>
- [12] S. Ding, Y. Xu, Z. Lu, F. Tang, T. Li, and J. Ge, "Power microservices troubleshooting by pretrained language model with multi-source data," in *2024 IEEE International Symposium on Parallel and Distributed Processing with Applications (ISPA)*, 2024, pp. 1768–1775.
- [13] P. Wang, X. Zhang, Y. Chen, and Z. Cao, "Unsupervised microservice system anomaly detection via contrastive multi-modal representation clustering," *Information Processing and Management*, vol. 62, no. 3, p. 104013, 2025. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0306457324003728>
- [14] L. Cao, J. Su, and Y. Luo, "Causal structure discovery for industrial process monitoring: A novel approach for nonlinear systems with complex noise," *IEEE Internet of Things Journal*, vol. 13, no. 7, pp. 12 826–12 836, 2026.
- [15] Z. Zeng, B. Zhao, X. Liu, and X. Deng, "Causal interpretability methods for iot anomaly traffic detection," *IEEE Internet of Things Journal*, vol. 12, no. 15, pp. 30 765–30 777, 2025.
- [16] L. Tang, E. Kou, W. Wang, and Q. Chen, "A root cause analysis framework for iot based on dynamic causal graphs assisted by llms," *IEEE Internet of Things Journal*, vol. 12, no. 16, pp. 34 563–34 581, 2025.
- [17] Z. Wang, Y. Zhou, Y. Zhang, F. Cong, D. Zhou, and Z. An, "Pa-rank: A gan and reinforcement learning powered framework for multimetric anomaly detection and causal diagnosis," *IEEE Internet of Things Journal*, vol. 12, no. 14, pp. 28 889–28 898, 2025.
- [18] A. Tank, I. Covert, N. Foti, A. Shojaie, and E. B. Fox, "Neural granger causality," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, p. 1–1, 2021. [Online]. Available: <http://dx.doi.org/10.1109/TPAMI.2021.3065601>
- [19] R. Marcinkevičs and J. E. Vogt, "Interpretable models for granger causality using self-explaining neural networks," in *International Conference on Learning Representations*, 2021.
- [20] X. Han, S. Absar, L. Zhang, and S. Yuan, "Root cause analysis of anomalies in multivariate time series through granger causal discovery," in *The Thirteenth International Conference on Learning Representations*, 2025. [Online]. Available: <https://openreview.net/forum?id=k38Th3x4d9>
- [21] A. M. Zaki, S. A. Elsayed, K. Elgazzar, and H. S. Hassanein, "Quality and budget-oriented task offloading for vehicular cooperative perception using reinforcement learning," *IEEE Internet of Things Journal*, vol. 12, no. 14, pp. 26 169–26 185, 2025.
- [22] H. Ma, Z. Huang, G. Lu, D. Fu, and C. Shi, "Greening edge ai: Optimizing inference accuracy and reducing carbon emissions with renewable energy," *IEEE Internet of Things Journal*, vol. 12, no. 13, pp. 24 300–24 312, 2025.
- [23] A. M. Alqahtani, K. A. Awan, A. Albakri, A. Jabbari, O. Z. Aletri, H. T. Alrakah, and H. H. Song, "Federated learning-based adaptive resource management for sustainable 6g-iot ecosystems," *IEEE Internet of Things Journal*, vol. 13, no. 1, pp. 1283–1298, 2026.
- [24] W. Yue, Y. Liu, H. Wang, H. Li, X. Ying, R. Guo, B. Xing, and J. Shi, "OLinear: A linear model for time series forecasting in orthogonally transformed domain," in *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. [Online]. Available: <https://openreview.net/forum?id=DyKPltvw1>
- [25] W. Yue, R. Guo, J. Shi, Z. Hao, S. Hu, and X. Ying, "vlinear: A powerful linear model for multivariate time series forecasting," 2026. [Online]. Available: <https://arxiv.org/abs/2601.13768>
- [26] Z. Xu, A. Zeng, and Q. Xu, "Fits: Modeling time series with 10k parameters," in *International Conference on Learning Representations (ICLR)*, 2024.
- [27] K. Yi, Q. Zhang, W. Fan, S. Wang, P. Wang, H. He, N. An, D. Lian, L. Cao, and Z. Niu, "Frequency-domain mlps are more effective learners in time series forecasting," in *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2023.
- [28] A. P. Mathur and N. O. Tippenhauer, "Swat: A water treatment testbed for research and training on ics security," in *2016 International Workshop on Cyber-Physical Systems for Smart Water Networks (CySWater)*. IEEE, 2016, pp. 31–36.
- [29] C. M. Ahmed, V. R. Palleti, and A. P. Mathur, "Wadi: a water distribution testbed for research in the design of secure cyber physical systems," in *Proceedings of the 3rd International Workshop on Cyber-Physical Systems for Smart Water Networks*, ser. CySWATER '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 25–28. [Online]. Available: <https://doi.org/10.1145/3055366.3055375>
- [30] R. Taormina, S. Galelli, N. O. Tippenhauer, E. Salomons, A. Ostfeld, D. G. Eliades, M. Aghashahi, R. Sundararajan, M. Pourahmadi, M. K. Banks, B. M. Brentan, E. Campbell, G. Lima, D. Manzi, D. Ayala-Cabrera, M. Herrera, I. Montalvo, J. Izquierdo, E. Luvizotto, S. E. Chandy, A. Rasekh, Z. A. Barker, B. Campbell, M. E. Shafiee, M. Giacomoni, N. Gatsis, A. Taha, A. A. Abokifa, K. Haddad, C. S. Lo, P. Biswas, M. F. K. Pasha, B. Kc, S. L. Somasundaram, M. Housh, and Z. Ohar, "Battle of the attack detection algorithms: Disclosing cyber attacks on water distribution networks," *Journal of Water Resources Planning and Management*, vol. 144, no. 8, p. 04018048, 2018. [Online]. Available: <https://ascelibrary.org/doi/abs/10.1061/%28ASCE%29WR.1943-5452.0000969>
- [31] H. Shan, Y. Chen, H. Liu, Y. Zhang, X. Xiao, X. He, M. Li, and W. Ding, "e-diagnosis: Unsupervised and real-time diagnosis of small-window long-tail latency in large-scale microservice platforms," in *The World Wide Web Conference (WWW)*, 2019, pp. 3215–3222.
- [32] A. Ikram, S. Chakraborty, S. Mitra, S. Saini, S. Bagchi, and M. Ko-caoglu, "Root cause analysis of failures in microservices through causal discovery," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 35, 2022, pp. 31 158–31 170.
- [33] L. Pham, H. Ha, and H. Zhang, "Baro: Robust root cause analysis for microservices via multivariate bayesian online change point detection," *Proc. ACM Softw. Eng.*, vol. 1, no. FSE, Jul. 2024. [Online]. Available: <https://doi.org/10.1145/3660805>
- [34] Y. Cheng, R. Yang, T. Xiao, Z. Li, J. Suo, K. He, and Q. Dai, "CUTS: Neural causal discovery from irregular time-series data," in *The Eleventh International Conference on Learning Representations*, 2023. [Online]. Available: <https://openreview.net/forum?id=UG8bQcD3Emv>
- [35] Y. Liu, T. Hu, H. Zhang, H. Wu, S. Wang, L. Ma, and M. Long, "itransformer: Inverted transformers are effective for time series forecasting," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.
- [36] S. Wang, J. Li, X. Shi, Z. Ye, B. Mo, W. Lin, S. Ju, Z. Chu, and M. Jin, "Timemixer++: A general time series pattern machine for universal predictive analysis," *arXiv preprint arXiv:2410.16032*, 2024.
- [37] K. Yi, J. Fei, Q. Zhang, H. He, S. Hao, D. Lian, and W. Fan, "Filternet: harnessing frequency filters for time series forecasting," in *Proceedings of the 38th International Conference on Neural Information Processing Systems*, ser. NIPS '24. Red Hook, NY, USA: Curran Associates Inc., 2025.
- [38] W. Yue, Y. Liu, H. Li, H. Wang, X. Ying, R. Guo, B. Xing, and J. Shi, "OLinear: A linear model for time series forecasting in orthogonally transformed domain," 2025. [Online]. Available: <https://arxiv.org/abs/2505.08550>
- [39] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, and R. Jin, "Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2022.
- [40] K. Yi, Q. Zhang, W. Fan, H. He, L. Hu, P. Wang, N. An, L. Cao, and Z. Niu, "FourierGNN: Rethinking multivariate time series forecasting from a pure graph perspective," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.
- [41] A. Katharopoulos, A. Vyas, N. Pappas, and F. Fleuret, "Transformers are rnns: fast autoregressive transformers with linear attention," in *Proceedings of the 37th International Conference on Machine Learning*, ser. ICML'20. JMLR.org, 2020.
- [42] M. Li, Z. Li, K. Yin, X. Nie, W. Zhang, K. Sui, and D. Pei, "Causal inference-based root cause analysis for online service systems with intervention recognition," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM, 2022, pp. 3230–3240.
- [43] L. Pham, H. Ha, X. Zhang, and H. Zhang, "Torai: Multi-source root cause analysis for blind spots in microservice service call graph," 2026. [Online]. Available: <https://arxiv.org/abs/2604.13522>
- [44] H. Wang, L. Pan, Z. Chen, D. Yang, S. Zhang, Y. Yang, X. Liu, H. Li, and D. Tao, "Fredf: Learning to forecast in the frequency domain," in *ICLR*, 2025.
- [45] R. Miles, I. Elezi, and J. Deng, " V_kD : Improving Knowledge Distillation Using Orthogonal Projections," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, Jun. 2024, pp. 15 720–15 730. [Online]. Available: <https://doi.ieeecomputersociety.org/10.1109/CVPR52733.2024.01488>