energies

MDPI

Article

Impedance-Informed and Self-Adaptive Graph Neural Networks for Short-Term Load Forecasting in Industrial Distribution Networks

Seunghyeon Park and Yongho Kim *

Energy AI Research Center, Korea Electronics Technology Institute, Seongnam 13509, Republic of Korea; park.seunghyeon@keti.re.kr
* Correspondence: yhkim@keti.re.kr

Abstract

Graph neural networks are often applied to load forecasting on the assumption that electrically related meter points should be modeled jointly, but this is rarely tested against a graph-free control, and building the graph usually requires a network model that is not always available. We ask whether an impedance-informed graph—estimated directly from PT/CT voltage and current, with no topology model—improves short-term load forecasting beyond what the forecasting architecture achieves alone. We compare an impedance-informed ST-GCN against binary and correlation-based graphs, a self-adaptive learned graph, a graph-free (identity-adjacency) ST-GCN control, an LSTM baseline, and a persistence baseline, at two industrial facilities (six and nine buses), using chronological splits, five seeds, and Bonferroni-corrected significance testing. The result is conditional and architecture-dependent. At one site, LSTM significantly beat every ST-GCN variant, including the graph-free control, so the shortfall is architectural, not graph-specific. At the other, every ST-GCN variant beat LSTM at the conventional level (not after Bonferroni correction), and the impedance graph gave among the lowest RMSE (20.2% below an untuned LSTM, 13.1% below a tuned LSTM); however, the graph-free control alone captured most of this gain, and impedance's further improvement over it was not significant. Lagged cross-correlation, Granger-causality, and input-perturbation analyses corroborate this: the trained ST-GCN's sensitivity tracks the impedance graph more closely than that of LSTM does, and a self-adaptive graph converges toward the impedance graph's structure, with seed-to-seed variability, only where graph structure helps. Graph-based load forecasting is therefore not universally beneficial; most of its benefit traces to the sequence-modeling architecture rather than to any specific graph, and both claims should be checked against a graph-free control and corrected significance tests before deployment.

Keywords: short-term load forecasting; smart distribution networks; spatio-temporal graph convolutional network; impedance-informed machine learning; impedance estimation

Check for updates

Academic Editor: Stéphane Grieu

Received: 28 August 2026
Revised: 29 September 2026
Accepted: 8 October 2026
Published: 3 November 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Short-term load forecasting at the feeder and facility level is a core information layer for industrial energy-management systems. Reliable forecasts support demand-side management, peak-load reduction, energy-storage scheduling, distributed-generation coordination, and other supervisory decisions that keep electrical energy systems flexible. Industrial facilities increasingly deploy potential transformers (PTs), current transformers (CTs), and

networked power meters at internal buses and subpanels. As a result, forecasting is increasingly a multivariate problem, in which several electrically related measurement points jointly describe facility demand. Temporal load patterns are therefore no longer the only useful signal; the electrical relationships among metered buses can be exploited as well. Thus, recent smart-grid forecasting studies have moved from single-series models toward architectures that represent both temporal dynamics and network structure [1,2]. Industrial facilities and microgrids with on-site generation have been a particular focus of this shift, given their combination of internal multi-point metering and variable local supply [3–5].

Long short-term memory (LSTM) networks and related recurrent models remain widely used for short-term electrical load forecasting [6–8]. When applied to multi-bus measurements, these models typically concatenate node features into a single input vector. Cross-bus dependence is then left to be learned implicitly. Spatio-temporal graph neural networks take a different approach. Measurements are represented as nodes, and information is exchanged according to an adjacency matrix that encodes electrical, spatial, or statistical relationships. ST-GCN and related architectures were first developed for networked spatio-temporal prediction [9,10] and have since been applied to power-system forecasting and monitoring at building, feeder, and substation scale [11–19]. For industrial energy applications, however, the practical question is not simply whether a GNN can forecast load accurately, but whether the graph it is given actually represents the electrical relationships in the monitored distribution system.

When a validated network model is available, line admittance, impedance, or known feeder connectivity provides a physically meaningful graph [20–22]. In many operating industrial facilities, however, the maintained topology may be incomplete at the level of retrofitted submetering. A common alternative is to construct the adjacency matrix from pairwise load correlation [23–25], but statistical correlation can reflect shared production schedules or common operating conditions as well as electrical coupling. This matters for energy-management applications because a graph used for forecasting or supervisory analytics needs to stay interpretable as production schedules and distributed-generation conditions change. This raises a question that has not, to our knowledge, been tested directly on real industrial metering data: when no topology model is available, can a graph estimated from electrical first principles be built, validated, and shown to behave physically, and does it actually forecast better than a graph-free model or a graph learned by the model itself? We answer this by evaluating an impedance-informed graph, estimated directly from available PT/CT measurements, against binary and correlation-based graphs, an LSTM baseline, and a self-adaptive graph, under the same forecasting protocol.

To answer this question, the contributions of this study are to:

1. Formulate an impedance-informed graph construction for industrial distribution-network measurements when a complete topology model is not available, validate it for physical plausibility, and benchmark it against topology-only, correlation-based, and self-adaptive graphs alongside a non-graph LSTM baseline under a repeated-seed evaluation protocol.
2. Use a cross-site comparison to identify the conditions under which graph structure helps industrial load forecasting and document a metering configuration that can undermine impedance-based graph construction.
3. Supplement aggregate forecasting metrics with lagged cross-correlation, Granger-causality, and input-perturbation analyses to check whether the observed forecasting behavior is actually consistent with the assumed electrical coupling.

The aim is a smart-grid-oriented methodology for graph selection and validation, not a claim that any single GNN architecture is universally superior.

The remainder of this paper is organized as follows. Section 2 reviews related work on load forecasting, spatio-temporal GNNs, and graph construction methods. Section 3 describes the two sites, the data processing pipeline, the three graph constructions, the model architectures, and the training, evaluation, and mechanistic-validation protocols. Section 4 answers the question posed above: whether, why, and under what conditions graph structure helps. Section 5 discusses practical implications, limitations, and future work, and Section 6 concludes the paper.

2. Related Work

Load forecasting with recurrent neural networks: LSTM networks [6] and their variants have been a standard deep-learning approach to short-term load forecasting for over a decade, applied to building, feeder, and system-level demand [7,8]. These models capture temporal dependence within a single series (or a flattened multivariate one) but have no explicit notion of spatial or electrical structure among multiple input channels; when applied to multi-bus data, every bus's features are concatenated into one input vector, and any cross-bus dependency must be learned implicitly, without a prior on which pairs of buses are more likely to be related.

Spatio-temporal graph neural networks: Graph convolutional networks [26] extend convolution to graph-structured data by aggregating each node's features with its neighbors' according to a (typically fixed) adjacency matrix. Spatio-temporal extensions interleave this graph convolution with a temporal model—1D convolution in ST-GCN [9] or a recurrent unit in DCRNN [10]—and were originally developed and validated on traffic-speed forecasting benchmarks, where the graph is unambiguous (physical road connectivity, often weighted by distance). Surveys of GNN-based spatio-temporal forecasting [27] note that most benchmark applications share this property of a known, externally supplied graph, which is a key point of departure for the present study.

GNNs for power systems and load forecasting: GNN-based forecasting has been applied to electrical load at building and feeder scale, for example combining graph convolution with a recurrent temporal model for residential load forecasting over a consumer-similarity graph [11], with recent extensions to multi-bus and substation-scale networks [13,16] and to sites with on-site generation [28]; a recent review catalogues a broader and growing range of GNN applications across power-system tasks, including forecasting, state estimation, and fault diagnosis [12]. Where a network model is available, the admittance or line-impedance matrix is a natural, physically grounded adjacency choice for power-system applications, as in physics-informed GNNs that reconstruct or directly exploit line admittance for state and parameter estimation [20,21], echoing physics-informed neural network approaches elsewhere in power systems that embed known physical relationships (e.g., power-flow equations) directly into model structure or loss [22,29]. Unlike physics-informed neural networks that embed governing equations directly into the loss function or model architecture, the present study uses electrical measurements to construct an impedance-informed adjacency matrix while retaining a conventional data-driven forecasting objective. In this setting, no validated network model is available, and the effective impedance information must itself be estimated from metering data rather than assumed known.

Data-driven graph construction: When topology is unavailable, prior work has most commonly built the GNN's adjacency matrix from statistical association between node series—typically Pearson or similar correlation, sometimes combined with a distance or threshold criterion, or learned end-to-end as an adaptive graph alongside the forecasting model [23,24,30–32]. This approach cannot say why two series are associated—the limitation that this paper's impedance-based construction is designed to address.

Topology and impedance identification from meter data: Separately from GNN-based forecasting, a distinct line of work has shown that unknown low-voltage distribution network topology, and in some cases line impedance itself, can be recovered directly from smart-meter voltage and current measurements without a maintained network model [33–38]. These results establish that the per-edge impedance regression used in this paper (Section 3.3) is a physically grounded and independently validated way to recover electrical structure from metering data alone, but its use as a basis for constructing GNN adjacency matrices for short-term load forecasting has received little attention—the gap this paper addresses.

Heterogeneous-condition and fusion-based forecasting: A related line of recent work addresses forecasting under heterogeneous data conditions through fusion rather than graph structure: combining multiple sensor modalities or forecasting sub-models to handle inter-sensor dependencies and variable operating conditions [39], or fusing multi-source meteorological inputs to improve short-term photovoltaic power prediction under variable weather [40]. These approaches target a related but distinct problem—combining heterogeneous inputs or model outputs—rather than determining which cross-node relationships should inform a graph-based spatial model, which is the question addressed here; the present study's finding that architecture, not graph choice, explains most of the cross-site accuracy difference (Section 5.1) suggests that fusion-based approaches robust to heterogeneous conditions may be a productive alternative or complement to graph-based spatial modeling at sites like SHIC, where the measured network offers little discriminatory structure for a graph to exploit.

Forecast evaluation methodology: Comparative forecasting studies are frequently evaluated on a single training run per model, which does not distinguish a genuine accuracy difference from noise attributable to random initialization; we follow the more statistically conservative practice of multi-seed repetition with paired significance testing in the spirit of the Diebold–Mariano framework for comparing predictive accuracy [41] and additionally use Granger causality [42] as a descriptive (not merely predictive) tool to characterize the temporal structure of inter-node coupling in the raw data, a use distinct from its more common role in identifying candidate causal drivers for a forecasting model's inputs.

Positioning: Relative to this literature, the present study contributes: (i) an impedance-informed graph construction that requires only PT/CT time series, not a network model, and an explicit validation protocol for the resulting impedance estimates in the absence of ground truth; (ii) a same-protocol comparison across two real sites with different correlation structure, revealing that graph-based forecasting benefit is site-dependent rather than universal—a finding obscured in single-site studies; and (iii) mechanistic validation (temporal-coupling and model-sensitivity analysis) of why a graph-based model helps or does not, going beyond the aggregate accuracy comparisons that dominate prior work.

3. Materials and Methods

Algorithm 1 summarizes the overall analysis pipeline described in this section, from data preprocessing (Section 3.2) through graph construction (Section 3.3), forecasting model architectures (Section 3.4), and the training and evaluation protocol (Section 3.5), to the mechanistic validation methods (Section 3.6).

3.1. Site Description and Data Acquisition

Data were collected from two industrial facilities, denoted SHIC and YC2, each instrumented with multiple potential transformers (PTs; one per metered bus, reporting per-phase and average line-to-line voltage) and current transformers (CTs; one or more per bus, reporting per-phase current and active/reactive power; where a bus has multiple CT

channels—e.g., a main feed plus several sub-loads—their active power, reactive power, and average current are summed to a single bus-level value). Table 1 summarizes the two sites.

Algorithm 1. End-to-End Analysis Pipeline for Impedance-Informed Load Forecasting

Input: per-bus PT/CT time series $\{V_b(t), I_b(t), P_b(t), Q_b(t)\}$ for buses $b = 1$ ...N; hub bus h

Output: trained forecasting models; adjacency matrices {A_bin, A_corr, A_imp, A_adp}; evaluation metrics; mechanistic-validation diagnostics

$X \leftarrow$ Preprocess(raw per-minute PT/CT exports)//device mapping, alignment, missing-value handling, resampling (Section 3.2)
(train, val, test) $\leftarrow$ ChronologicalSplit(X, 0.70, 0.15, 0.15);
Normalize X using mean/SD computed on X[train] only;
A_bin $\leftarrow$ BinaryHubLeaf(bus hierarchy); //data-independent
A_corr $\leftarrow$ |PearsonCorr(P_i, P_j)| over X[train], for all bus pairs i, j;
for each leaf bus $\ell \neq h$ do
    Fit $\Delta V_\{h,\ell\}(t) = Z_\ell \cdot I_\ell(t) + \varepsilon(t)$ by OLS on X[train] (bottom decile of I_ℓ excluded);
    Record Z_ℓ, R^2_ℓ;
end for
A_imp $\leftarrow$ Normalize($1/|Z_\ell|$) for all ℓ;
for each model $M \in$ {Persistence, LSTM, ST-GCN(identity, binary, corr, imp, adaptive)} do
    for seed $s = 1$ to 5 do
        Train M on (X[train], X[val]) with early stopping (patience = 15);
        if M = ST-GCN(adaptive) then
            A_adp $\leftarrow$ softmax(ReLU($E_1 E_2^{\mathrm{T}}$)), learned jointly with M;
        end if
        Evaluate M on X[test] $\rightarrow$ RMSE, MAE, NRMSE, sMAPE (Section 4.2);
    end for
end for
Test all model pairs: paired t-test on RMSE (5 seeds); apply Bonferroni correction, $\alpha/22$ (Section 4.10);
Validate A_imp: check sign/magnitude plausibility of each Z_ℓ (Section 4.8);
Compute lagged cross-correlation and Granger causality on ΔP for representative bus pairs (Section 4.5);
Compute input-perturbation sensitivity $S_\{i \rightarrow j\}$ for LSTM and ST-GCN(imp); correlate S with {A_bin, A_corr, A_imp} (Section 4.6);
return trained models, {A_bin, A_corr, A_imp, A_adp}, evaluation metrics, validation diagnostics.

Table 1. Site and dataset summary.

Property	SHIC	YC2
Metered buses (nodes)	6 (A01–A06)	9 (A02, A03, A04, A06–A10, A12)
Hub bus (main feed)	A01	A02
On-site generation	Rooftop PV; current aggregated at hub meter	Rooftop PV at A03
Sampling interval	1 min	1 min
Study period	9 January 2026 to 1 April 2026 (82 days)	19 January 2026 to 10 March 2026 (50 days)
Raw samples per bus	≈118,000	≈72,000
Missing-data fraction (worst bus)	5.6%	17.6% (A05); 97.2% zero-voltage (A11, excluded)
Forecast target nodes	6	8 (A03, the PV bus, excluded as a target but retained as a graph node)

Two further buses at YC2 (A01, A05/A11 region) were excluded from the final dataset for excessive missingness or a data-quality fault (a near-total zero-voltage fraction, Table 1); a third candidate site (denoted DSPM in project records) was excluded entirely because its available PT data period did not overlap with the other instrumentation and is earmarked as a future third validation site (Section 5.4). At YC2, bus A03 carries the facility's rooftop photovoltaic interconnect; because it represents generation rather than load, it is retained as a graph node (its behavior affects neighboring buses' voltage) but excluded from the set of forecast targets and flagged with an is_generation indicator feature (Section 3.2).

3.2. Data Preprocessing Pipeline

Raw per-minute CSV exports from each PT/CT channel were processed through a six-stage, site-agnostic pipeline: (1) device/bus mapping extraction from a facility device list, associating each PT and its co-located CT channels with a bus identifier; (2) per-bus time series loading and alignment; (3) consolidation onto a common minute-resolution index spanning each site's full study period, with a missingness report per bus (Table 1); (4) cleaning—zero-valued voltage readings (indicating a communication or sensor fault rather than a true zero, since these are line-to-line voltages on an energized LV bus) are treated as missing, and short gaps (≤5 min) are linearly interpolated; (5) manual, site-specific finalization of which bus/period combinations to retain, given the missingness patterns in Table 1; and (6) feature engineering, producing a final per-site tensor $\mathbf{X} \in \mathbb{R}\text{^}(T \times N \times F)$ with $F = 9$ features per bus-timestep: voltage (V), active power (P), reactive power (Q), average current (Iavg), a binary is-generation flag, and cyclical hour-of-day and day-of-week encodings (sine/cosine pairs). Active power P is the forecast target feature throughout.

3.3. Graph Construction Methods

Three $N \times N$ adjacency matrices were constructed per site, all derived solely from the metering data (no topology diagram was available at either site): The hub–leaf representation follows the available device mapping and metering hierarchy, in which one upstream main-feed meter is paired with downstream branch meters. Because a validated single-line diagram was unavailable, this representation is treated as a metering-informed structural assumption rather than a fully verified feeder topology.

Binary: A_bin connects the hub bus to every leaf bus with weight 1 (0 elsewhere), encoding only the assumption that the facility's internal distribution is a star radiating from the main feed, with no leaf–leaf edges and no weighting of hub–leaf edges by strength.

Correlation: A_corr is the absolute Pearson correlation of active power, $|r(P_i, P_j)|$, between every bus pair i, j, computed over each site's training split only (the same 70% used to fit the forecasting models; Section 3.5), to avoid leaking validation/test-period information into the graph construction. Unlike the binary graph, this is dense (every pair gets a nonzero weight) and requires no assumption about which buses are electrically related, but, as shown in Section 4.5, it cannot distinguish genuine short-term electrical/operational coupling from two buses simply sharing a production schedule.

Impedance: A_imp estimates an effective voltage-drop impedance proxy $|Z_{\text{eff}}|$ between the hub and each leaf bus, likewise from the training split only, from the physical relationship

$$\Delta V_{\text{hub,leaf}}(t) = V_{\text{hub}}(t) - V_{\text{leaf}}(t) \approx Z_{\text{eff}} \cdot I_{\text{leaf}}(t)$$

fit as an ordinary least-squares regression of $\Delta V_{\text{hub,leaf}}$ on I_{leaf} for each hub–leaf pair, with the bottom decile of the current range excluded (voltage-drop signal is unstable at near-zero current). Only hub–leaf edges are estimated (a star topology, matching the binary graph's connectivity but not its uniform weight); the resulting $|Z_{\text{eff}}|$ values are inverted ($1/|Z_{\text{eff}}|$) and

max-normalized to produce an affinity-style edge weight consistent in scale with the other two graphs. A negative regression slope is retained as a signed diagnostic (interpreted in Section 4.8 as reverse power flow from local generation), but the graph convolution itself, like the other two variants, uses the non-negative magnitude. The construction code is site-agnostic, parameterized only by the hub and leaf-node lists, and the same procedure was applied at both facilities.

The fitted coefficient is interpreted as an effective voltage-drop impedance proxy rather than an exact complex line impedance. Because the available measurements use voltage and current magnitudes, the regression does not separately identify resistance and reactance or explicitly account for the instantaneous power-factor angle. The proxy is therefore used only to derive relative electrical affinity for graph construction, and its sign, magnitude, and regression fit are evaluated separately for physical plausibility in Section 4.8.

Electrical-distance graph (Section 4.11 only): As an alternative comparison point for the leaf–leaf structure, which the hub-and-spoke A_imp cannot represent, we additionally define an electrical-distance graph A_ed; hub–leaf edges are identical to A_imp, and each leaf–leaf edge (i, j) is weighted by the inverse of the series impedance along the only path between them through the hub, $1/(|Z_i| + |Z_j|)$, using the same training-split $|Z|$ estimates as A_imp. Like A_imp, the result is inverted and max-normalized. A_ed is used only for the structural comparison in Section 4.11 and was not used to train a forecasting model (see that section for the reasoning).

3.4. Model Architectures

LSTM baseline: Input $X \in \mathbb{R}\text{^}(B \times T \times N \times F)$ (batch × 60-min lookback × nodes × features) is flattened to $\mathbb{R}\text{^}(B \times T \times NF)$ and passed through a 2-layer LSTM (hidden size 64), and the final timestep's hidden state is linearly projected to the n_{target} forecast outputs. This architecture has no explicit node structure: every node's features at every timestep are concatenated into a single per-timestep vector.

ST-GCN: Each of nblocks = 2 residual blocks applies (i) a temporal 1D convolution (kernel size 3, shared across nodes) mapping Fin → $h = 32$ hidden channels, (ii) a graph convolution $H' = \hat{A}HW$ using the symmetric-normalized adjacency $\hat{A} = D - 1/2(A + \mathrm{I})D - 1/2$ (self-loops added, degree-normalized) built once per site/graph-variant and held fixed during training, followed by ReLU, and (iii) a second temporal convolution $h \to h$, with a linear residual connection when input and output channel counts differ. After the final block, the last timestep's per-node hidden state is linearly projected to a scalar, and the target nodes' scalars are gathered as the model's ntarget outputs. The adjacency matrix is the only difference among the four ST-GCN variants built on this shared backbone—identity (self-loops only), binary, correlation, and impedance—evaluated in Section 4; all other architecture and training settings are identical, isolating the effect of graph choice (including the graph-free identity control) from confounding architectural differences.

Self-adaptive ST-GCN: A further model tests whether a fixed, impedance-informed graph still adds value once the network is free to learn its own cross-bus relationships. Following the self-adaptive graph construction used in Graph WaveNet [24] and related data-driven graph-learning approaches [23,30–32], the adjacency matrix is replaced by one learned end-to-end from the forecasting loss: two node-embedding matrices $E_1, E_2 \in \mathbb{R}\text{^}(N \times 10)$ are initialized randomly and optimized jointly with the rest of the network, and the adjacency used at every forward pass is $A_\text{adp} = \text{softmax}(\text{ReLU}(E_1 {E_2}^{\mathrm{T}}))$, row-normalized so each node's outgoing weights sum to one. A_adp is recomputed from the current embeddings at every training step, rather than fixed after construction as for the other three graphs, and is otherwise substituted directly for $\hat{A}$ inside the same block structure described above. Because the embeddings start from random initialization with

no reference to voltage, current, or correlation, any resemblance A_adp develops to the impedance or correlation graphs by the end of training reflects what the forecasting objective favors, not something imposed on it.

3.5. Training and Evaluation Protocol

Each site's tensor is normalized per feature (mean/SD computed on the training split only) and split chronologically 70%/15%/15% into training/validation/test segments (no shuffling across time, so the test period strictly follows training—appropriate for a deployment scenario where the model is trained on past data and evaluated on the future). A sliding window of 60 min of input predicts active power 15 min ahead (single-step) at the target nodes. Models are trained with Adam (learning rate 1×10^{-3}) and mean-squared-error loss for up to 100 epochs with early stopping (patience 15 epochs on validation loss, best-validation checkpoint retained). Each of the seven model configurations (persistence; LSTM; ST-GCN × {identity, binary, correlation, impedance, adaptive}) is trained at both sites with five random seeds (1–5; persistence requires no training) to characterize initialization variance; we report mean ± SD across seeds and test pairwise differences with paired t-tests on per-seed test RMSE ($n = 5$; the complete set of comparisons is given in Section 4.10).

Four metrics are reported on the (denormalized) test set: root-mean-square error (RMSE) and mean absolute error (MAE) in the same units as active power; normalized RMSE (NRMSE; RMSE relative to mean absolute test-period load, as a percentage); and symmetric mean absolute percentage error (sMAPE). NRMSE and sMAPE are reported alongside RMSE/MAE because plain MAPE is distorted during low-load periods (e.g., night shifts, when the denominator approaches zero).

A follow-up hyperparameter screen (Section 4.9) varies ST-GCN hidden width {32, 64} × depth {2, 3} and LSTM hidden size {32, 64, 128} × depth {1, 2} (learning rate fixed) with one seed per configuration, re-confirming with a full 5-seed run any configuration whose single-seed RMSE improves on the default by more than 1%.

3.6. Mechanistic Validation Methods

Beyond aggregate accuracy, three additional analyses test whether the graph-based model's premise and behavior are physically grounded, with results reported in Section 4.

Impedance plausibility (Section 4.8): Each hub–leaf $|Z_{\text{eff}}|$ estimate from Section 3.3 is checked for (i) sign consistency—positive for conventional loads, negative only where reverse power flow from local generation is physically expected—and (ii) magnitude plausibility for low-voltage in-plant wiring, in the absence of a single-line diagram to validate against directly.

Temporal coupling (Section 4.5): For eight representative bus pairs in total—SHIC A01–A02, A01–A04, and A01–A06 (hub–leaf edges spanning the well-fit and anomalous cases identified in Section 4.8) and A02–A05 (a non-hub pair included for cross-site balance; SHIC's correlation structure is too uniform, see Section 4.4, for any non-hub pair to stand out as specifically "flagged"), and YC2 A02–A12 and A02–A03 (hub–leaf edges) and A06–A07 and A10–A12 (non-hub pairs flagged by the correlation graph (Section 4.4) as strongly associated: $r = 0.89$ and $r = 0.98$, respectively)—we compute the lagged cross-correlation function (±30 min) of first-differenced active power (isolating short-term fluctuation from shared daily/production schedule) and bidirectional Granger causality (lags 1–15 min, matching the forecast horizon), testing whether inter-node influence in the raw data shows a detectable propagation delay.

Input-perturbation sensitivity (Section 4.6): For the trained LSTM and ST-GCN(impedance) models (5 independently trained seeds at each site), each source node's input active power is perturbed by +1 SD (holding all else fixed) and the resulting change in every target

node's forecast is measured, over a sample of test-set windows, producing a source-node × target-node "learned sensitivity" matrix per model per seed, averaged across seeds. This matrix is compared (Pearson correlation, self-pairs excluded) against each of the three adjacency matrices from Section 3.3, testing whether the trained model's behavior—not just its accuracy—reflects the impedance-graph structure it was either given (ST-GCN) or not (LSTM).

The analysis workflow, trained-model diagnostic outputs, and scripts underlying Sections 3.3–3.6 may be shared through a project repository or by the corresponding author where permitted (Data Availability Statement).

4. Results

The results below answer this question in stages. Section 4.2 reports the headline result: whether graph structure improves forecasting accuracy, and at which site. Sections 4.3 and 4.4 ask why the answer differs by site, using per-node error and the underlying correlation structure. Sections 4.5 and 4.6 test whether the trained models actually behave in a way consistent with electrical coupling, using temporal-coupling and input-perturbation analyses. Section 4.7 checks the stability of training across seeds, Section 4.8 evaluates whether the impedance graph itself is physically plausible, and Section 4.9 repeats the main comparison under alternative hyperparameters. Section 4.10 summarizes the statistical testing behind these results, and Section 4.11 asks whether a fixed impedance-informed graph is still necessary once the model is allowed to learn on its own.

4.1. Experimental Setup

For both sites, minute-resolution active power (P), reactive power (Q), voltage (V), and average current (I_avg) at each bus were assembled into a single feature tensor and split chronologically into training (70%), validation (15%), and test (15%) segments—no shuffling across time, so the test period strictly follows the training period. Inputs use a 60 min lookback window to predict active power 15 min ahead (single-step). All models were trained with the Adam optimizer (learning rate 1×10^{-3}) and mean-squared-error loss for up to 100 epochs with early stopping (patience = 15 epochs on validation loss). To account for initialization variance, each site/model configuration was trained with five random seeds (1–5); we report the mean ± SD across seeds and assess pairwise differences against the LSTM baseline with a paired t-test on per-seed test RMSE ($n = 5$).

Seven models are compared per site: a naive persistence baseline (the most recently observed value, carried forward to the forecast horizon), an LSTM baseline (flattened node × feature input, no explicit spatial structure), and five ST-GCN variants sharing the same temporal-convolution backbone that differ only in the adjacency matrix used for graph convolution—identity (self-loops only, i.e., no cross-node information reaches the model, isolating the architecture's own contribution), binary (hub–leaf connectivity only), correlation (|Pearson r| of active power between bus pairs, computed on the training split only), impedance (weights derived from a per-edge regression of the hub–leaf voltage difference on leaf current, also computed on the training split only; see Sections 3.3 and 4.8), and self-adaptive (a graph learned end-to-end from the forecasting objective; Section 3.4). Figure 1 shows all three graphs for both sites as node–edge networks; the impedance graph is visibly sparser and more differentiated than the correlation graph, particularly at SHIC (Section 4.4). These are drawn in the style of a single-line diagram: the square busbar with an incoming arrow marks the hub bus (point of common coupling with the grid), the open triangle marks on-site generation (YC2, node A03), and filled circles mark other metered buses; line width scales with graph weight.

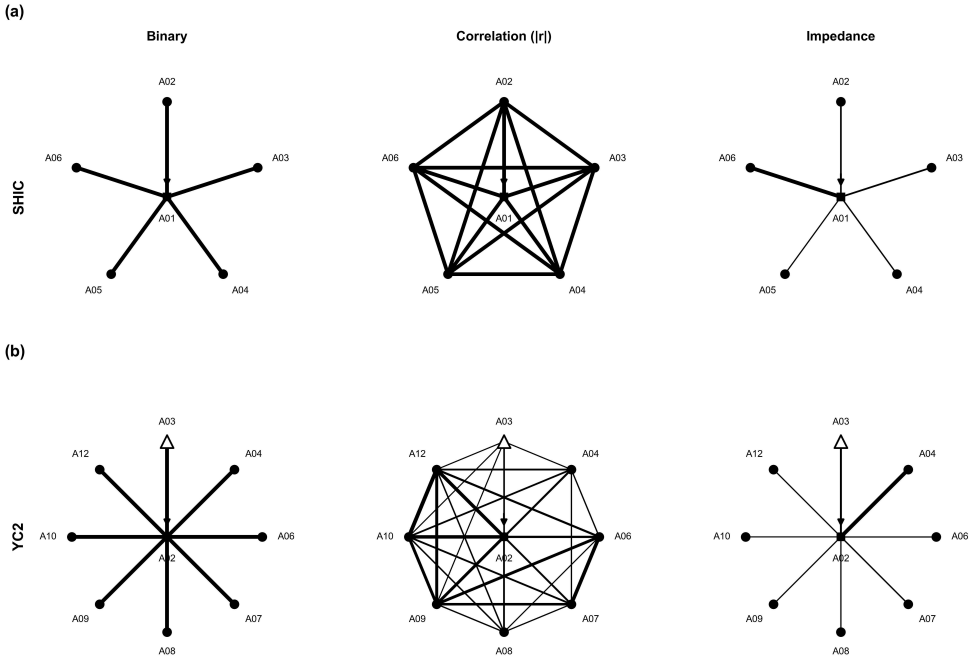


Figure 1. Adjacency graphs used for graph convolution: binary, correlation, and impedance networks. (**a**) SHIC; (**b**) YC2.

4.2. Load Forecasting Performance

Table 2 summarizes test-set RMSE, MAE, normalized RMSE (NRMSE, relative to mean absolute load), and symmetric MAPE for both sites. NRMSE/sMAPE are reported alongside RMSE/MAE because plain MAPE is distorted during low-load periods (e.g., night shifts) when the denominator approaches zero.

Table 2. Test-set forecasting error by model and site.

Site	Model	RMSE	MAE	NRMSE (%)	sMAPE (%)
SHIC	Persistence	11.11 ± 0.00	6.33 ± 0.00	5.33 ± 0.00	**3.31 ± 0.00**
SHIC	LSTM	**9.46 ± 0.10**	**5.76 ± 0.17**	**4.54 ± 0.05**	3.35 ± 0.18
SHIC	ST-GCN (identity)	10.18 ± 0.05	6.24 ± 0.10	4.89 ± 0.02	3.67 ± 0.21
SHIC	ST-GCN (binary)	10.25 ± 0.13 †	6.29 ± 0.13	4.92 ± 0.06	3.76 ± 0.19
SHIC	ST-GCN (correlation)	10.30 ± 0.10 †	6.38 ± 0.11	4.95 ± 0.05	3.81 ± 0.16
SHIC	ST-GCN (impedance)	10.18 ± 0.09 †	6.23 ± 0.04	4.89 ± 0.04	3.75 ± 0.25
SHIC	ST-GCN (adaptive)	10.15 ± 0.05 †	6.29 ± 0.11	4.87 ± 0.02	3.77 ± 0.28
YC2	Persistence	12.68 ± 0.00	**3.40 ± 0.00**	44.87 ± 0.00	**9.75 ± 0.00**
YC2	LSTM	14.50 ± 1.53	6.65 ± 0.60	51.30 ± 5.41	50.81 ± 1.93
YC2	ST-GCN (identity)	11.77 ± 0.13	4.70 ± 0.50	41.67 ± 0.46	45.04 ± 5.26
YC2	ST-GCN (binary)	11.99 ± 0.21	5.01 ± 0.21	42.42 ± 0.76	48.32 ± 1.38
YC2	ST-GCN (correlation)	12.00 ± 0.14 †	5.14 ± 0.25	42.48 ± 0.49	46.92 ± 1.71
YC2	ST-GCN (impedance)	11.57 ± 0.06 †	4.74 ± 0.21	40.95 ± 0.20	44.34 ± 1.24
YC2	ST-GCN (adaptive)	**11.56 ± 0.15**	4.74 ± 0.30	**40.89 ± 0.52**	46.27 ± 4.20

Values marked with † reflect a value that changed from the previous submission: SHIC binary/correlation/impedance/adaptive and YC2 correlation/impedance. Correlation and impedance changed because both graphs are now reconstructed from the training split only (Sections 3.3 and 4.8); LSTM, identity, binary, adaptive, and persistence use the same methodology as before, but the entire experiment matrix (all seven models, both sites, and five seeds) was retrained from scratch for this revision, and GPU training is not bit-

for-bit-reproducible across runs even with the same random seed, so binary and adaptive at SHIC also show small changes that reflect ordinary retraining variability rather than a methodology change. At YC2, this same retraining variability happened to round to the values already reported for LSTM, identity, binary, and adaptive; correlation and impedance are true methodology changes (training-split graphs) whose resulting accuracy happens to be numerically close to the previous values at the reported precision (Section 4.8 gives the full-precision comparison and the per-edge graph changes that produced it).

Values in bold mark the best-performing model per site per metric.

At SHIC, the LSTM baseline achieves the lowest RMSE, and every ST-GCN variant is significantly worse by a paired t-test on RMSE, surviving Bonferroni correction for the $m = 22$ comparisons reported in Section 4.10 (binary: $t = -7.43$, $p = 0.0017$; correlation: $t = -11.64$, $p = 0.0003$; impedance: $t = -9.75$, $p = 0.0006$; self-adaptive: $t = -10.70$, $p = 0.0004$; identity: $t = -10.34$, $p = 0.0005$; $n = 5$ seeds). Critically, the identity variant—ST-GCN with no cross-node information at all—performs statistically indistinguishably from the impedance-informed graph (RMSE 10.18 ± 0.05 vs. 10.18 ± 0.09; $t = -0.03$, $p = 0.98$), so none of the ST-GCN family's shortfall at SHIC is attributable to a specific choice of graph: the architecture itself, independent of graph structure, is simply less accurate than the LSTM at this site. At YC2, the pattern reverses at the conventional $\alpha = 0.05$ level: every ST-GCN variant outperforms the LSTM baseline (binary: $t = 3.32$, $p = 0.029$; correlation: $t = 3.14$, $p = 0.035$; impedance: $t = 3.86$, $p = 0.018$; self-adaptive: $t = 3.66$, $p = 0.022$; identity: $t = 3.74$, $p = 0.020$), though none of these five comparisons survives the stricter Bonferroni-corrected threshold (Section 4.10). The impedance-weighted graph gives among the largest and most consistent improvements—a 20.2% RMSE reduction and a 28.7% MAE reduction over the LSTM baseline (13.1% RMSE reduction relative to a tuned LSTM, Section 4.9), with the lowest seed-to-seed variance of any model at either site (RMSE SD = 0.06, versus 1.53 for LSTM at the same site)—and it significantly outperforms the binary and correlation graphs at the uncorrected level ($t = 4.32$, $p = 0.012$ and $t = 4.49$, $p = 0.011$, respectively; neither survives Bonferroni correction). It is statistically indistinguishable from the self-adaptive graph (Section 4.11). Unlike at SHIC, however, the identity control shows that much of this advantage is architectural rather than graph-specific: ST-GCN(identity) alone already reaches RMSE 11.77 ± 0.13 at YC2, an 18.8% reduction over LSTM by itself, and the further improvement from adding the impedance graph ($11.77 \rightarrow 11.57$, 1.7%) is not statistically significant ($t = 2.58$, $p = 0.061$). We return to this decomposition—how much of the YC2 result is "any ST-GCN architecture" versus "specifically the impedance graph"—in Section 5.1. Table 2 also reports a naive persistence baseline: the untuned LSTM is in fact slightly worse than persistence at YC2 (14.50 vs. 12.68 RMSE), consistent with the overfitting visible in the training-loss curves (Section 4.7), whereas every ST-GCN variant, including the graph-free identity control, outperforms persistence there (11.56–11.99 vs. 12.68); at SHIC, LSTM has the largest margin over persistence of any model (9.46 vs. 11.11). Figure 2 shows the mean $\pm$ SD comparison for both sites; Figure 3 shows the full per-seed distribution, which makes the LSTM instability at YC2 (one seed reaching RMSE = 17.5, Figure 3) visually apparent alongside the tight clustering of the impedance-graph runs.

The comparatively high NRMSE and sMAPE values at YC2 can be amplified by frequent low-load intervals and substantial differences in load magnitude across target nodes. Accordingly, the absolute normalized errors should be interpreted together with RMSE and MAE, while the model-to-model comparison remains informative because all methods are evaluated on an identical chronological test set. A related caveat applies to Table 2's bold markings: naive persistence attains the lowest MAE and sMAPE at YC2, and the lowest sMAPE at SHIC, despite having the worst or near-worst RMSE at both sites. A forecast that simply repeats the last observed value neither over- nor under-shoots sys-

tematically and rarely produces a large single-step miss in relative terms, which keeps its typical (MAE and sMAPE) error small, but it also cannot anticipate the load changes that occasionally produce the large absolute errors RMSE penalizes most heavily. We therefore treat RMSE, used throughout the significance testing in Sections 4.2 and 4.10, as the primary comparison metric, and report MAE/NRMSE/sMAPE for completeness rather than as independent rankings.

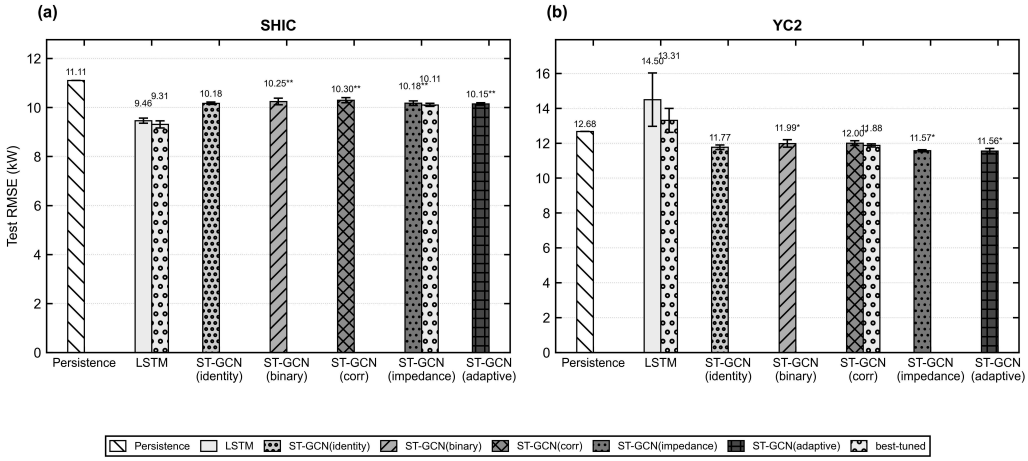


Figure 2. Test RMSE by model at each site (mean ± SD, $n = 5$ seeds). (**a**) SHIC; (**b**) YC2. * $p < 0.05$ (uncorrected); ** significant after Bonferroni correction (paired t-test vs. LSTM, RMSE across 5 seeds; Section 4.10).

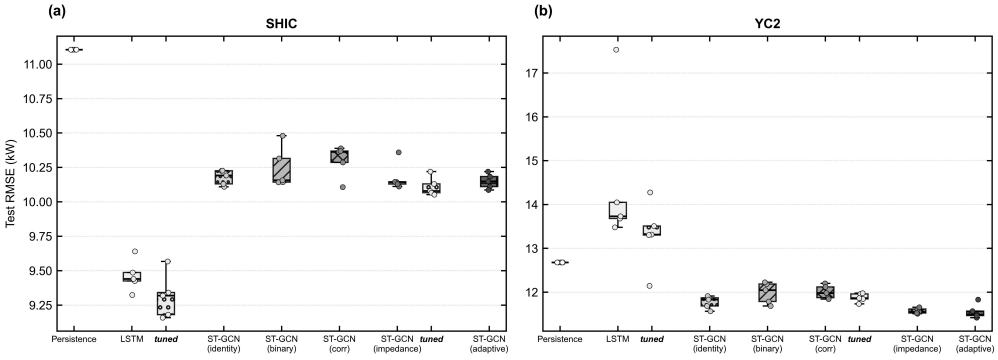


Figure 3. Per-seed test RMSE distributions by model at each site. (**a**) SHIC; (**b**) YC2. Italic "tuned" boxes use the best screened hyperparameters (Section 4.9) and sit immediately right of their default counterpart.

Table 2 also reports the persistence and identity controls and the self-adaptive model, ST-GCN (adaptive), all already discussed above; Section 4.11 examines the self-adaptive model further, including whether the graph it learns resembles the impedance graph.

4.3. Node-Level Forecasting Error and Example Forecasts

The site-level result in Section 4.2 could come from a uniform shift at every node or from a few strongly affected ones; distinguishing these matters for understanding why the graph helps where it does. Table 2 aggregates error over all target nodes; Figure 4 breaks test RMSE down per node for the LSTM and ST-GCN (impedance) models, averaged over all five seeds. The per-node pattern is uneven rather than a uniform shift. At SHIC, ST-GCN is slightly worse than LSTM at every node, matching the site-level result. At YC2, the aggregate RMSE advantage is concentrated at a subset of nodes rather than uniform and is not even in ST-GCN's favor everywhere at the per-node level: the largest

reductions appear at A08 (RX gas-reformer node; LSTM 12.63 → ST-GCN 4.34, a 65.6% reduction) and A06 (3.62 → 1.33, 63.3%), with moderate reductions at A10, A12, and A09 (36–48%) and a modest reduction at the hub node A02 itself (36.57 → 30.63, 16.2%); at A04 and A07, however, ST-GCN is worse per-node than LSTM (8.74 vs. 8.84, and 1.98 vs. 2.67, respectively). A07's hub edge is among the least reliable in the training-split impedance regression ($R^2 = 0.02$, Section 4.8), consistent with an unreliable edge weight actively hurting rather than merely failing to help there, but A04's edge fit is unremarkable ($R^2 = 0.62$), so its per-node shortfall is not fully explained by edge quality alone. The aggregate RMSE advantage at YC2 therefore comes from large gains at a subset of high-magnitude nodes, not a uniform improvement everywhere.

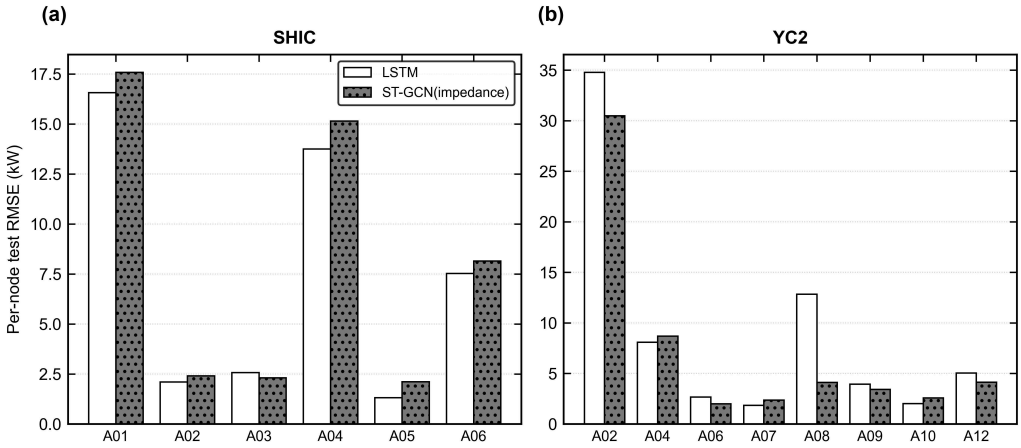


Figure 4. Per-node test RMSE for LSTM and ST-GCN(impedance). (**a**) SHIC; (**b**) YC2. Bars show the mean over 5 seeds.

Figure 5 shows example 15 min ahead forecasts against an observed load for a representative node at each site over a multi-day window drawn from the held-out test period. Both models track the dominant diurnal/operational pattern closely; the visually clearest divergence is at YC2 in the later part of the window, where the LSTM forecast drifts upward relative to the observed load, while ST-GCN(impedance) continues to track it—the same instability quantified in Section 4.7.

4.4. Site-Dependent Effectiveness of Graph Structure

The opposite outcomes at SHIC and YC2 trace back to how informative the graph topology is at each site. Figure 6 shows why: the pairwise correlation of active power across buses is nearly uniform at SHIC ($|r| = 0.94$–0.99 across all bus pairs) but highly variable at YC2 ($|r| = 0.02$–0.98). A near-uniform correlation structure gives graph convolution little to work with—every node looks like every other node's neighbor—so at SHIC, imposing any of the three graph structures adds inductive bias without adding useful signal, while a model with no spatial constraint (LSTM) is free to fit whatever cross-node dependencies exist in the flattened input. At YC2, inter-bus relationships are heterogeneous, so a graph that correctly encodes which buses are electrically related lets the model share information efficiently across neighbors. Both lower error and markedly lower seed-to-seed variance follow from that.

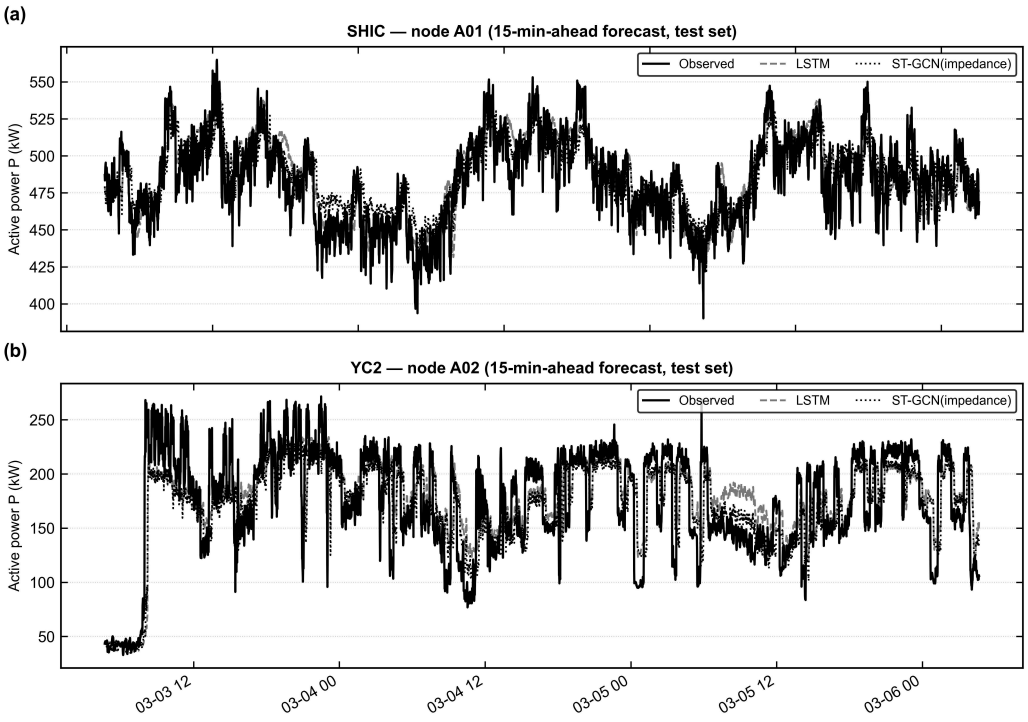


Figure 5. Observed vs. forecast active power for a representative node. (**a**) SHIC; (**b**) YC2. Shown for a single representative seed (seed 1); quantitative comparisons elsewhere use all 5 seeds.

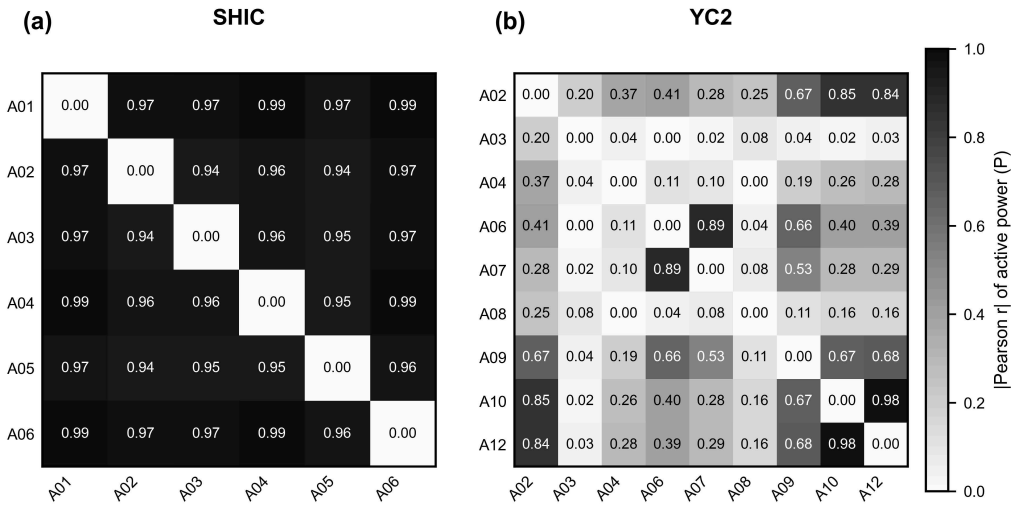


Figure 6. Pairwise correlation of active power between buses. (**a**) SHIC; (**b**) YC2.

4.5. Temporal Coupling Between Metered Buses

Section 4.4 shows that graph structure helps where inter-bus correlation is heterogeneous; this leaves open a mechanistic question, whether the coupling that makes the graph useful is contemporaneous or lagged, and thus whether a spatial graph is even the right kind of model to use. The motivation for a graph-structured model is the premise that a change at one bus propagates, electrically or operationally, to others—the same reasoning behind treating road segments as a graph for traffic forecasting. That premise has a testable implication: if node A's load change drives node B's, the effect should be detectable at some positive lag in B relative to A. We tested this directly on the first-differenced active power series (ΔP, which isolates short-term fluctuations from the shared daily/production schedule) via lagged cross-correlation (±30 min) and Granger causality (lags 1–15 min, matching the

forecast horizon) for the eight representative bus pairs listed in Section 3.6 (five hub–leaf edges and three non-hub pairs, chosen to span well-fit and anomalous impedance edges and, where available, strongly correlated non-hub pairs).

Contrary to the traffic-propagation framing, the cross-correlation peaks at lag 0 for seven of the eight pairs (Figure 7); the one exception (SHIC A02–A05) peaks at 1 min with a correlation too small ($r = 0.07$) to distinguish from noise. At minute resolution, coupling between buses is essentially contemporaneous rather than lagged. This behavior differs fundamentally from transportation networks, in which spatial interactions can be associated with observable propagation delays. This is not a negative result for the graph-based approach: it is precisely why a spatial graph convolution (which mixes information across nodes within the same timestep) is the appropriate architecture, rather than a model that searches for a fixed inter-node lag. Granger causality tests were significant in both directions for every well-correlated pair ($p < 0.05$, often $\ll 10^{-10}$). Combined with the zero-lag peak, this fits genuine mutual, simultaneous coupling through a shared electrical bus rather than a sequential, one-directional cause-then-effect chain. Given the sample sizes involved ($n > 60{,}000$ min per site), these p-values are best read as confirming detectable association, not as evidence for one direction over the other.

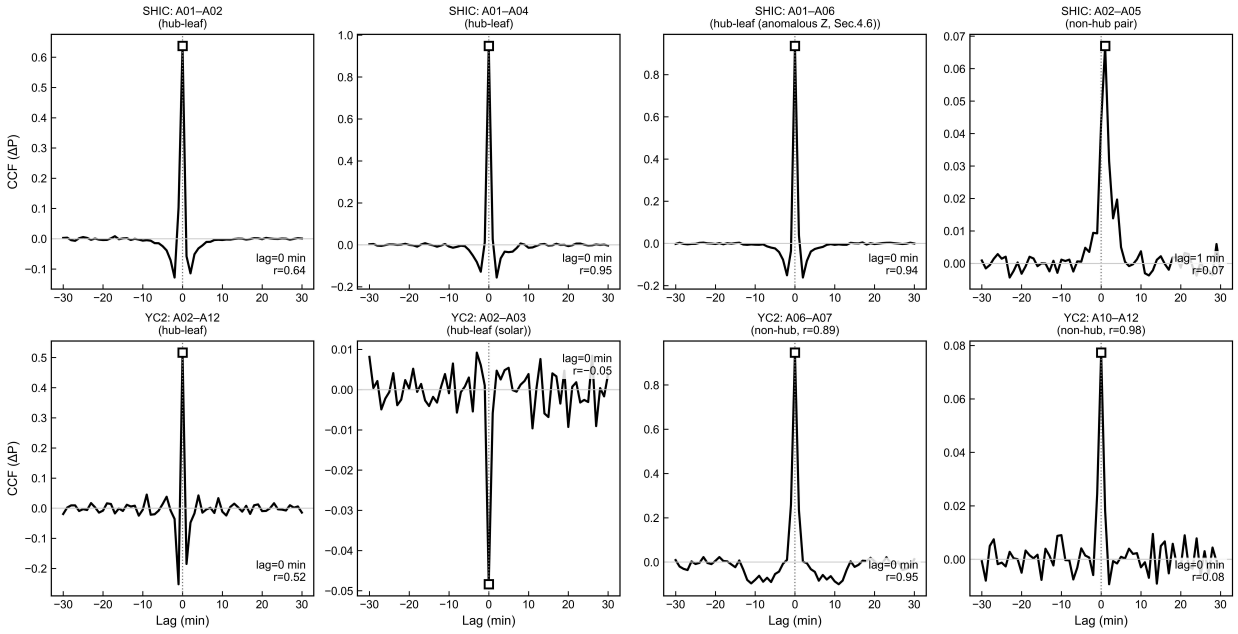


Figure 7. Cross-correlation of active power fluctuations for representative bus pairs. (**Top**) SHIC; (**bottom**) YC2.

A second, unplanned finding refines the site-dependent story from Section 4.4. YC2's A10–A12 pair has a very high raw-level correlation ($r = 0.98$, Figure 6), but its short-term fluctuation correlation collapses to $r = 0.08$ at lag 0 (Figure 7)—the two buses track the same production schedule but do not move together minute-to-minute. A06–A07, by contrast, is strongly correlated in both the raw level ($r = 0.89$) and the fluctuation ($r = 0.95$): a genuinely tight short-term coupling, not just a shared schedule. Because the correlation-graph construction (Section 3.3) uses raw P levels, it cannot distinguish these two cases—it would treat A10–A12 as at least as strongly connected as A06–A07, when only the latter reflects real-time electrical/operational coupling. This is an additional, independent argument for preferring the impedance graph over the correlation graph: impedance is estimated from the physical voltage–current relationship and is not affected by two buses happening to follow the same production schedule.

4.6. Consistency Between Learned Sensitivity and Graph Structure

Section 4.5 establishes that inter-node coupling in the raw data is contemporaneous. This section asks the complementary question: given that, does the trained ST-GCN actually route information along the physical (impedance) graph, or could an unstructured model learn the same behavior on its own? For each trained model (LSTM and ST-GCN(impedance), five independently trained seeds per site), we perturbed each source node's input P by +1 SD (holding all else fixed) and measured the resulting change in every target node's 15 min ahead forecast, producing a source-node × target-node "learned sensitivity" matrix per seed; the seed-averaged matrix (Figures 8 and 9 also report the seed-to-seed SD) was then correlated (Pearson *r*, self-pairs excluded) against each of the three adjacency matrices.

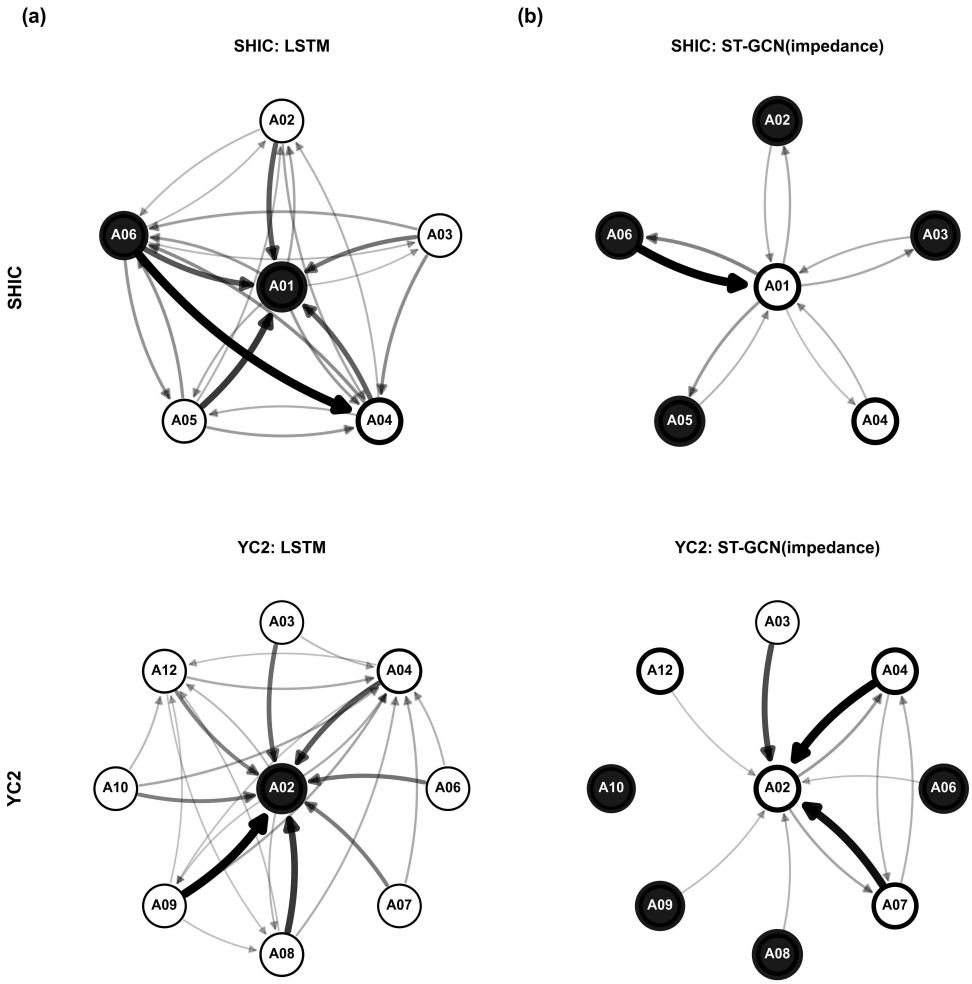


Figure 8. Input-perturbation sensitivity network for LSTM and ST-GCN(impedance). (**a**) SHIC; (**b**) YC2. Edge weights are the mean sensitivity over 5 independently trained seeds. Arrow thickness scales with sensitivity magnitude.

The contrast is stark (Figure 8). LSTM's sensitivity network is dense and tangled: perturbing almost any node sends a visible arrow to almost every other node, with no consistent direction, most strongly through whichever bus happens to carry the largest share of total load (A01 at SHIC, A02 at YC2)—a plausible but graph-agnostic "the biggest node matters most" heuristic learned from the flattened input. ST-GCN(impedance)'s sensitivity network, by contrast, is sparse and almost entirely self-referential: perturbing node i mostly changes only node i's own forecast (shown as a filled ring), with the one

consistent set of arrows running to and from the hub—exactly the star topology of the impedance graph, which by construction only assigns nonzero weight to hub–leaf edges (Section 3.3), never leaf–leaf. Figure 9 quantifies this: ST-GCN(impedance)'s sensitivity correlates with the impedance matrix at $r = 0.93$ (SHIC) and $r = 0.77$ (YC2), in both cases its highest correlation with any of the three graphs and well above LSTM's correlation with the same impedance matrix ($r = 0.22$ at SHIC, $r = 0.48$ at YC2).

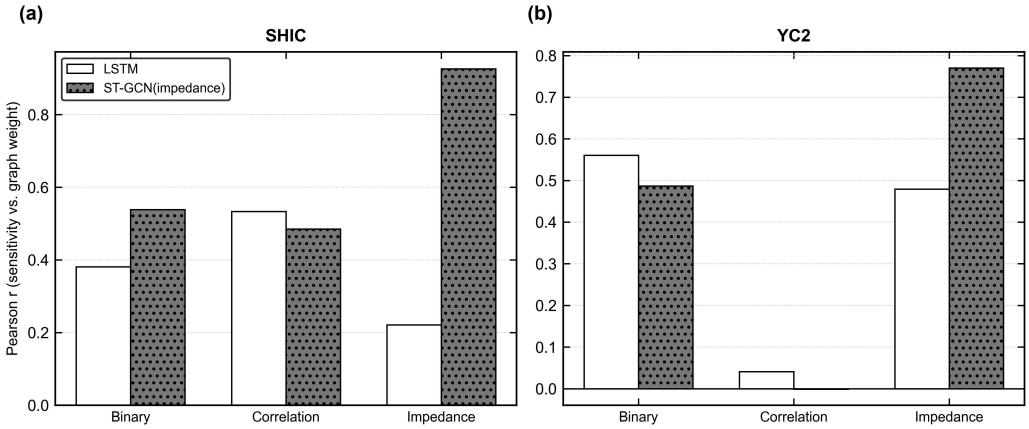


Figure 9. Correlation between learned sensitivity and graph structure. (**a**) SHIC; (**b**) YC2. Computed from the 5-seed-averaged sensitivity matrices in Figure 8.

One caveat is worth flagging before reading too much into this: ST-GCN's forward pass computes its output directly as a function of the impedance matrix, so some alignment is guaranteed by construction, not discovered—the model cannot propagate information along an edge the graph does not have. The informative part is what is not guaranteed: the graph convolution's learned weights and the temporal convolution could in principle have suppressed or distorted that structure (e.g., converging to near-uniform attention that ignores edge weights), and they did not—the star pattern from Figure 1 survives intact through training. The LSTM side of the comparison is just as telling: with no graph supplied, its implicit sensitivity pattern is weaker, more diffuse, and inconsistent across sites—best aligned with the correlation graph at SHIC but the binary graph at YC2. An unstructured model, in other words, does not spontaneously recover the same physically grounded, hub-centric propagation structure that ST-GCN is given directly. Together with the contemporaneous-coupling finding of Section 4.5, this supports the paper's central claim in the form the data actually allow: nodes on the same electrical bus move together within the same timestep. Supplying that structure explicitly, as the impedance graph does, yields a model whose internal behavior—not just its aggregate accuracy—reflects it; an architecture without that prior does not reliably discover it on its own.

4.7. Training Dynamics and Model Stability

The high seed-to-seed variance of LSTM at YC2 (Section 4.2, RMSE SD = 1.53 vs. 0.06 for ST-GCN(impedance)) is visible not only across seeds but within individual training runs. Figure 10 plots training and validation loss by epoch for a representative run of each model at each site. At SHIC, both models show validation loss stabilizing within roughly 10 epochs with no divergence from the training curve. At YC2, the LSTM's validation loss rises steadily from its very first epoch even as training loss keeps falling—a classic overfitting signature—triggering early stopping at epoch 16; the ST-GCN(impedance) run at YC2 shows a milder version of the same pattern (validation loss drifts upward after its epoch-5 minimum) but degrades far less before stopping. This pattern points to a possible regularizing effect of the graph structure at YC2. Constraining the model to route information only through

physically meaningful edges seems to limit how much it can overfit the idiosyncrasies of any single training run—a plausible partial explanation for both the lower RMSE and the much tighter seed-to-seed spread ST-GCN(impedance) shows in Figure 3b.

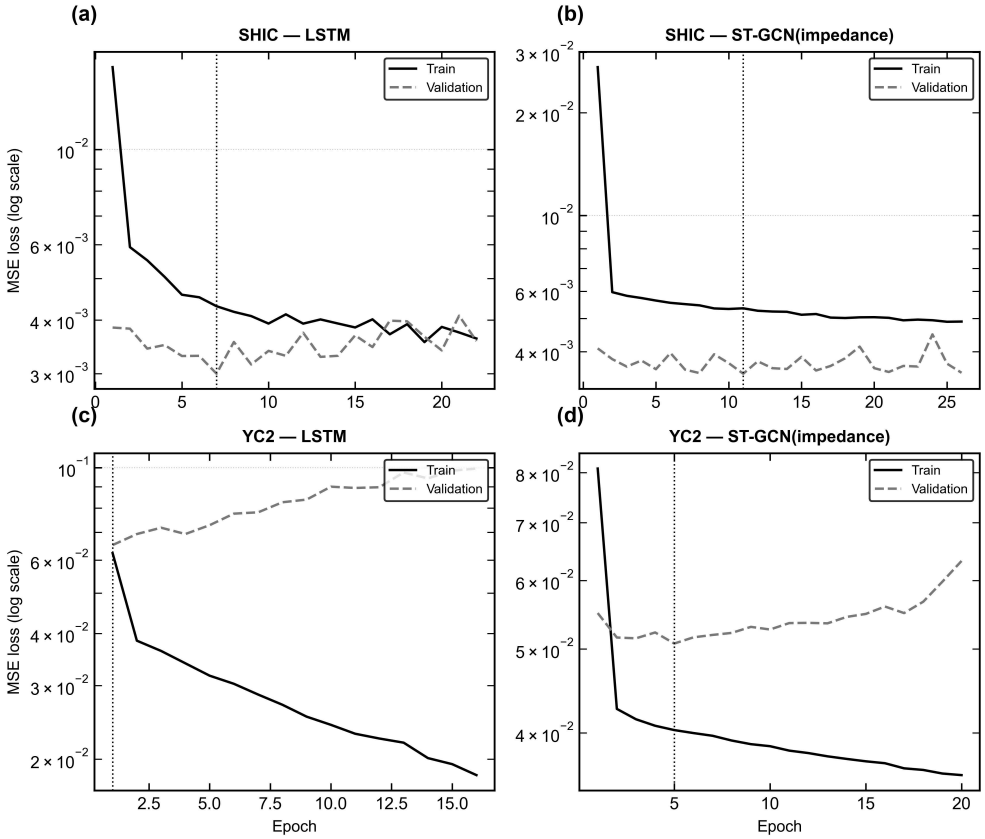


Figure 10. Training and validation loss by epoch. (**a**) SHIC—LSTM; (**b**) SHIC—ST-GCN(impedance); (**c**) YC2—LSTM; (**d**) YC2—ST-GCN(impedance). The dotted vertical line marks the early-stopping checkpoint (best validation-loss epoch). Shown for a single representative seed (seed 1) per site/model.

4.8. Physical Plausibility of the Impedance-Informed Graph

Because no single-line diagram was available for either site, we validate the impedance estimates indirectly: for each hub–leaf edge, $|Z_{\text{eff}}|$ is estimated as the slope of a linear regression of the hub–leaf voltage difference on leaf current (bottom 10% of the current range excluded as unstable), and we check (i) sign consistency—positive for conventional loads, negative only where reverse power flow from local generation is physically expected—and (ii) whether the magnitude falls in a plausible range for low-voltage in-plant wiring (mΩ–hundreds of mΩ, given the measured hub voltages of ~377–390 V line-to-line, consistent with a 380 V LV distribution system).

At YC2, the A02–A03 edge (A03 is the site's rooftop solar interconnect) is the only negative estimate over the full study period ($Z = -0.0097\ \Omega$, $R^2 = 0.79$) among eight edges—consistent with the presence of reverse power flow from local generation; all others are positive over the full period, with $R^2 = 0.55$–1.00 and $|Z|$ in the 3–96 mΩ range. Restricted to the training split only (the graph actually used throughout this paper; Sections 3.3 and 4.8), a second edge, A02–A06, also becomes negative (training-split $Z = -0.062\ \Omega$ vs. full-period $+0.051\ \Omega$), with its fit collapsing at the same time ($R^2 = 0.14$ vs. 0.71); unlike A02–A03, this sign flip is not attributable to known local generation, and Table 3 below reports both periods for every edge at both sites so the distinction between the two is unambiguous. Because the regression uses voltage and current magnitudes, however, the negative slope

should not be interpreted as a direct measurement of the sign of a complex line impedance; it is used here as a diagnostic of the effective voltage-drop relationship.

Table 3. Hub–leaf impedance regression by period: full study period vs. training split only.

Site	Edge	Z_Full (Ω)	R^2_Full	Z_Train (Ω)	R^2_Train
SHIC	A01–A02	+0.0060	0.35	+0.0064	0.63
SHIC	A01–A03	+0.0051	0.29	+0.0062	0.61
SHIC	A01–A04	+0.0100	0.88	+0.0083	0.73
SHIC	A01–A05	+0.0177	0.57	+0.0046	0.18
SHIC	A01–A06	−0.0013	0.18	−0.0009	0.05
YC2	A02–A03	−0.0097	0.79	−0.0097	0.79
YC2	A02–A04	+0.0032	0.56	+0.0034	0.62
YC2	**A02–A06**	**+0.0507**	**0.70**	**−0.0623**	**0.14**
YC2	A02–A07	+0.0254	0.55	+0.0043	0.02
YC2	A02–A08	+0.0427	0.98	+0.0429	0.94
YC2	A02–A09	+0.0512	0.97	+0.0550	0.97
YC2	A02–A10	+0.0955	0.97	+0.0950	0.97
YC2	A02–A12	+0.0426	0.99	+0.0428	1.00

Two further checks address how much these estimates depend on specific analysis choices. First, because the graphs above were originally estimated from the full study period, we re-estimated every edge using only the training split (the same 70% used to fit the forecasting models; Section 3.5) to rule out validation/test-period information leaking into the graph construction; Table 2 and all forecasting results in this paper now use these training-split-only graphs throughout. Most edges are stable under this restriction (SHIC: maximum change in normalized edge weight ≤ 0.12; YC2: ≤0.66), but two are not: YC2's A02–A06 edge changes sign (training-only $Z = -0.062$ Ω vs. full-period +0.051 Ω) and its fit quality collapses ($R^2 = 0.14$ vs. 0.71), and SHIC's A01–A05 edge similarly degrades ($R^2 = 0.18$ vs. 0.57). Both were already among the lower-R^2 edges in the full-period estimate, so we read this as confirming, rather than contradicting, that low-R^2 edges are unreliable and should not be treated as precise electrical estimates—a limitation of the regression itself, not of restricting it to the training split. Second, because the analysis above uses only the current magnitude, we also tested separating resistive and reactive components by regressing the voltage difference on both active- and reactive-power-derived current terms (a linearized DistFlow-style approximation, $\Delta V \approx (R{\cdot}P + X{\cdot}Q)/V$). The resulting $|Z|$ estimates agree closely with the magnitude-only estimates for well-fit edges (e.g., YC2 A02–A12: 0.043 Ω vs. 0.043 Ω; A02–A09: 0.057 Ω vs. 0.055 Ω) and, for one edge, fit distinctly better (SHIC A01–A04: R^2 improves from 0.73 to 0.95, suggesting a non-negligible reactive contribution there); for the already-poor edges (A01–A05/A06 and A02–A06/A07), separating R and X does not rescue the fit (R^2 remains ≤0.14), consistent with these reflecting a genuine measurement or topology confound rather than an artifact of the magnitude-only simplification. We therefore retain the simpler magnitude-only $|Z_{\text{eff}}|$ for the main graph construction, since it is adequate for the well-behaved majority of edges and the R/X-separated model does not change which edges are unreliable. As a further robustness check, we retrained ST-GCN on a version of the impedance graph that excludes any edge with training-split $R^2 < 0.3$ (SHIC: excluding A01–A05 and A01–A06, leaving three of five hub–leaf edges; YC2: excluding A02–A06 and A02–A07, leaving six of eight) and re-normalizing the remainder. This "robust" variant gives RMSE 10.15 ± 0.11 at SHIC, statistically indistinguishable from the standard impedance graph (10.18 ± 0.09; $t = 0.42$, $p = 0.70$), and 11.74 ± 0.18 at YC2, numerically worse than the standard impedance graph (11.57 ± 0.06; $t = -2.10$, $p = 0.10$, not significant). Excluding low-R^2 edges therefore does

not reliably improve, and at YC2 numerically worsens, forecasting accuracy; we read this as showing that a low-quality regression fit does not straightforwardly translate into a harmful graph edge for message passing, so we report this comparison transparently rather than adopting R^2-based exclusion as the default graph construction.

At SHIC, four of five edges are similarly well-behaved (positive, $R^2 = 0.36$–0.88, $|Z| = 6$–18 mΩ), but the A01–A06 edge is anomalous, being negative ($Z = -0.0013\ \Omega$) with a poor fit ($R^2 = 0.18$), and the A01–A03 edge also has a low R^2 (0.29) despite a plausible-looking magnitude. Cross-referencing the facility device map indicates that unlike YC2, where solar generation is metered as its own bus (A03), SHIC's hub bus (A01) itself aggregates a "main" CT channel and a rooftop solar CT channel into a single node. The hub voltage/current used as the regression's reference is therefore not a clean grid-side measurement—it is contaminated by solar-generation variability that is uncorrelated with the leaf-side load current, which degrades the impedance regression for edges away from the hub and is a plausible contributor to the impedance graph's lack of benefit at SHIC (Section 4.2). We flag this as a data-construction limitation specific to SHIC's metering topology rather than a general weakness of the impedance-estimation method and recommend it as a caveat in the Section 5.5 (Limitations). Figure 11 shows representative regression fits for both well-fit edges (YC2 A02–A03, A02–A12; SHIC A01–A04) and the anomalous SHIC A01–A06 edge. Detailed per-edge diagnostic results are available from the corresponding author where permitted.

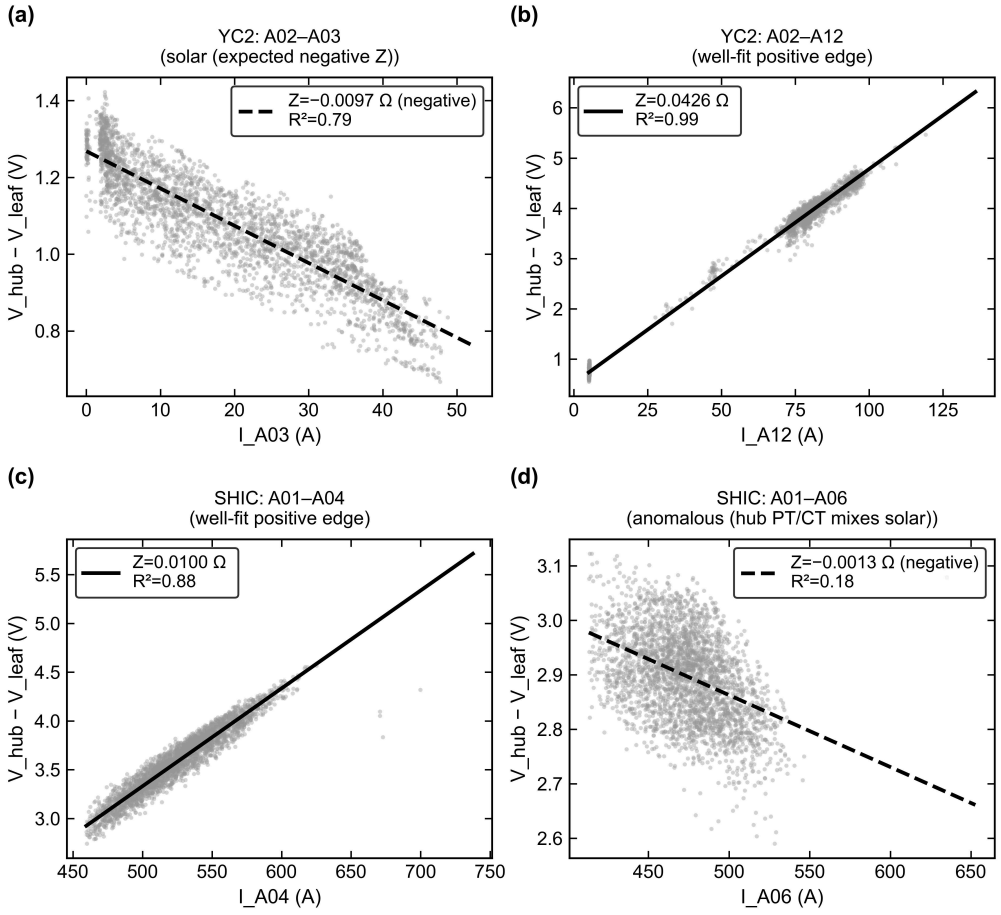


Figure 11. Representative hub–leaf impedance regressions. (**a**) YC2 A02–A03; (**b**) YC2 A02–A12; (**c**) SHIC A01–A04; (**d**) SHIC A01–A06.

4.9. Hyperparameter Sensitivity

The site-dependent result of Section 4.2 could in principle be an artifact of a single, arbitrarily chosen network size rather than a genuine property of the data. We checked this by screening ST-GCN hidden width {32, 64} × depth {2, 3} and LSTM hidden size {32, 64, 128} × depth {1, 2} (learning rate fixed at 1×10^{-3}) with a single seed per configuration and re-confirmed with a full five-seed run any configuration whose single-seed RMSE improved on the default (hidden width/size 32/64, depth 2) by more than 1%. Table 4 reports the four configurations that met this bar.

Table 4. Best screened hyperparameters vs. default.

Site	Model	Default RMSE	Tuned RMSE	Best Config	*P* (Tuned vs. Default)
SHIC	LSTM	9.46 ± 0.10	9.31 ± 0.15	Hidden = 128, layers = 1	0.065
SHIC	ST-GCN (impedance)	10.18 ± 0.09	10.11 ± 0.06	Width = 32, depth = 3	0.275
YC2	LSTM	14.50 ± 1.53	13.31 ± 0.68	Hidden = 32, layers = 1	0.234
YC2	ST-GCN (correlation)	12.00 ± 0.14	11.88 ± 0.09	Width = 32, depth = 3	0.268

In every case, the tuned configuration's mean RMSE is lower than the default (and, for YC2 LSTM, the seed-to-seed SD roughly halves—1.53 → 0.68, which is itself consistent with the overfitting behavior discussed in Section 4.7), but none of the four differences reach significance at $\alpha = 0.05$ with only five seeds ($p = 0.065$–0.275). We read this as a directionally consistent but statistically inconclusive effect: a smaller, shallower LSTM and a deeper (three-block) ST-GCN each modestly help, but larger capacity is not the primary driver of the SHIC/YC2 divergence reported in Section 4.2—no tuned configuration closes the gap between LSTM and ST-GCN at either site, or reverses which model wins. Detailed screening and confirmation outputs are available from the corresponding author where permitted. The remaining eight site/model combinations showed <1% single-seed improvement across the grid and were not re-confirmed; notably, ST-GCN(impedance) at YC2—the paper's headline result—was among them (screened improvement = −0.3%), so Table 2's reported impedance-graph result already reflects the better of the two settings. This hyperparameter screen was performed before the training-split-only graph correction described in Section 4.8; Table 4's Default RMSE column has been updated to match Table 2's corrected values, but the screen itself was not rerun on the corrected graphs. Because that correction changed forecasting accuracy only marginally overall (Section 4.8), we expect the qualitative conclusion here—tuning helps modestly but does not reverse which model wins at either site—to still hold, though this was not re-verified directly.

4.10. Statistical Significance Summary

Reported *p*-values (Section 4.2) are from a paired t-test on RMSE across the five shared random seeds. Table 5 reports all 22 comparisons performed in this study (11 per site): every ST-GCN variant (binary, correlation, impedance, self-adaptive, and identity) against LSTM; impedance against binary, correlation, and identity; self-adaptive against identity; and both impedance and LSTM against the naive persistence baseline. Because 22 tests are performed on the same seed set, we apply a Bonferroni correction ($\alpha = 0.05/22 = 0.0023$) in addition to the uncorrected $\alpha = 0.05$ threshold. At SHIC, all five "ST-GCN vs. LSTM" comparisons, including identity, survive Bonferroni correction ($p \leq 0.0017$), as do LSTM vs. persistence and impedance vs. persistence (both $p < 0.0001$); impedance is not significantly different from binary, correlation, or identity at SHIC even before correction, nor is adaptive significantly different from identity ($p = 0.48$). At YC2, the five "ST-GCN vs. LSTM" comparisons (including identity, $p = 0.020$) and the impedance-vs.-binary/correlation comparisons are significant at the uncorrected $\alpha = 0.05$ level ($p = 0.011$–0.035), but none

survives Bonferroni correction; impedance vs. persistence survives at YC2 ($p < 0.0001$), but LSTM vs. persistence does not reach even the uncorrected threshold ($p = 0.076$) and is numerically in LSTM's disfavor (mean RMSE 14.50 vs. 12.68). The correct statement is therefore that every ST-GCN variant, but not LSTM itself, reliably beats the naive persistence forecast at YC2; an earlier version of this section incorrectly extended that claim to LSTM, which we have corrected. Neither adaptive-vs.-identity comparison is significant at either site (SHIC $p = 0.48$, YC2 $p = 0.15$), reinforcing Section 5.1's conclusion that a graph learned from data does not measurably outperform no graph at all once the ST-GCN architecture is in place. We report both thresholds throughout rather than treating only one as definitive: given the small sample ($n = 5$ seeds) and the number of comparisons, the uncorrected p-values are best read as evidence of a consistent direction of effect within each site, while the Bonferroni-surviving comparisons are the more conservative, higher-confidence claims. A Diebold–Mariano test on per-timestep forecast residuals, or a block-bootstrap confidence interval on the test-period forecast errors, would provide a complementary and likely more powerful test of predictive accuracy; because the present manuscript reports seed-level aggregate errors, this analysis is identified as future statistical validation rather than claimed here.

Table 5. Complete pairwise significance testing (paired t-test on RMSE, 5 seeds; all 22 comparisons, both sites).

Site	A	B	t	p	Sig. (Bonferroni)	Better
SHIC	Binary	LSTM	−7.434	0.0017	Yes	LSTM
SHIC	Correlation	LSTM	−11.644	0.0003	Yes	LSTM
SHIC	Impedance	LSTM	−9.749	0.0006	Yes	LSTM
SHIC	Adaptive	LSTM	−10.698	0.0004	Yes	LSTM
SHIC	Identity	LSTM	−10.343	0.0005	Yes	LSTM
SHIC	Impedance	Binary	+0.717	0.5128	No	Impedance
SHIC	Impedance	Correlation	+2.003	0.1157	No	Impedance
SHIC	Impedance	Identity	−0.028	0.9793	No	Identity
SHIC	Adaptive	Identity	+0.784	0.4767	No	Adaptive
SHIC	Impedance	Persistence	+20.305	0.0000	Yes	Impedance
SHIC	LSTM	Persistence	+31.878	0.0000	Yes	LSTM
YC2	Binary	LSTM	+3.322	0.0293	No	Binary
YC2	Correlation	LSTM	+3.135	0.0350	No	Correlation
YC2	Impedance	LSTM	+3.864	0.0181	No	Impedance
YC2	Adaptive	LSTM	+3.662	0.0215	No	Adaptive
YC2	Identity	LSTM	+3.742	0.0201	No	Identity
YC2	Impedance	Binary	+4.324	0.0124	No	Impedance
YC2	Impedance	Correlation	+4.485	0.0109	No	Impedance
YC2	Impedance	Identity	+2.580	0.0613	No	Impedance
YC2	Adaptive	Identity	+1.755	0.1542	No	Adaptive
YC2	Impedance	Persistence	+39.303	0.0000	Yes	Impedance
YC2	LSTM	Persistence	−2.375	0.0764	No	Persistence

4.11. Comparison with a Self-Adaptive Graph Baseline

A natural question is whether a fixed, impedance-informed graph is still useful once the network is allowed to learn its own adjacency matrix directly from the forecasting objective (Section 3.4), or indeed whether any explicit cross-node structure is needed at all once the ST-GCN's temporal-convolution backbone is in place (the identity control introduced in Section 4.2). Table 2 includes both models. At SHIC, self-adaptive's accuracy (RMSE 10.15 ± 0.05) sits among the other graph variants and the identity control (10.15–10.30),

all significantly worse than LSTM; at YC2, it reaches 11.56 ± 0.15, closely matching the impedance-informed graph (11.57 ± 0.06) and significantly better than LSTM at the uncorrected level ($p = 0.022$, not Bonferroni-significant). ST-GCN(adaptive)'s seed-to-seed SD at YC2, however, is roughly 2.5 times that of ST-GCN(impedance) (0.15 vs. 0.06), so the two models reach comparable accuracy with different reliability, and—as with the impedance graph itself (Section 4.2)—self-adaptive's advantage over the graph-free identity control (11.77 ± 0.13) is not statistically significant.

Whether the learned graph itself resembles the impedance graph is a separate question from accuracy, and the answer differs by site—though less cleanly than a single training run suggests. Because A_adp is re-initialized randomly and re-learned independently for each of five seeds, we compare the seed-averaged learned adjacency (and, for transparency, each individual seed) against the impedance, correlation, and binary graphs using only unique off-diagonal edge pairs, together with a permutation (QAP-style) test that reshuffles node labels rather than assuming independent samples (Section 3.4); this corrects an earlier analysis that inadvertently counted each edge twice (once in each triangle of the symmetrized matrix) when estimating significance. At YC2, the seed-averaged learned graph is positively and significantly correlated with the impedance graph ($r = 0.39$, permutation $p = 0.008$) but not with the binary or correlation graphs ($|r| < 0.02$, $p > 0.9$); individual seeds range from $r = -0.09$ to $r = 0.54$, so the correspondence, while directionally consistent (four of five seeds positive) and significant on average, is not a deterministic outcome of training. At SHIC, no fixed graph shows a significant relationship with the learned graph, on average or in any individual seed (impedance: $r = 0.14$, $p = 0.64$); Figure 12a shows the SHIC adaptive graph is instead dominated by leaf–leaf edges that the hub-and-spoke impedance graph cannot represent at all. As an alternative comparison for this leaf–leaf structure, we also constructed an electrical-distance graph in which leaf–leaf pairs are weighted by the inverse of their series impedance through the hub ($1/(|Z_i| + |Z_j|)$; Section 3.3); we compared A_ed directly against the same seed-1 learned adaptive graph shown in Figure 12, using the identical unique-edge, permutation-tested procedure used above. The result is a null one: A_ed does not show a significant resemblance to the learned graph at either site (SHIC: $r = -0.19$, permutation $p = 0.43$; YC2: $r = 0.20$, $p = 0.16$), so the leaf–leaf structure the adaptive model discovers at SHIC is not explained by simple series-impedance distance either. Given this null result, and that the identity-vs.-impedance comparison in Section 4.2 already shows the specific choice of fixed graph contributes less to accuracy than the ST-GCN architecture itself at both sites, we did not retrain a further forecasting model on A_ed. Taken together, this is a noisier, more qualified version of the site-dependent pattern reported in Section 4.4: a graph learned from data with no electrical information shows a statistically detectable, if imperfect and seed-variable, resemblance to the impedance graph only at the site where fixed graph structure is actually informative, and no such resemblance where it is not.

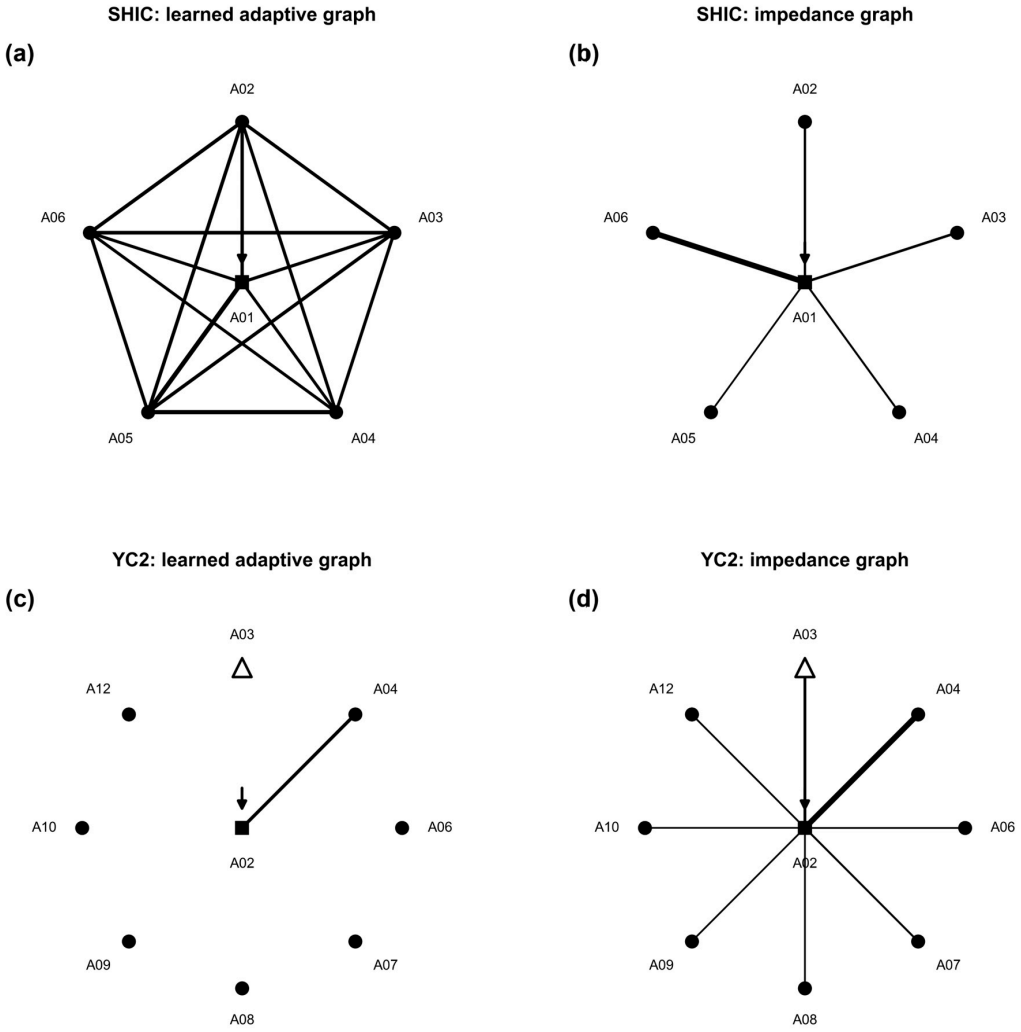


Figure 12. Learned adaptive graph compared with the impedance graph. (**a**,**c**) Adaptive graph; (**b**,**d**) impedance graph. Top: SHIC; bottom: YC2. Drawn in the same style as Figure 1.

5. Discussion

5.1. Conditions for Effective Graph-Based Load Forecasting

The main cross-site finding is that graph structure does not help uniformly, and disentangling why requires separating two effects that a comparison against LSTM alone conflates: the ST-GCN architecture itself (temporal convolution with message passing, regardless of what the adjacency matrix contains) and the specific graph supplied to it. The identity control (Section 4.2, Table 2) isolates the first effect by fixing the adjacency matrix to self-loops only, i.e., no cross-node information reaches the model at all. At SHIC, the identity control performs statistically indistinguishably from every other ST-GCN variant, including impedance ($t = -0.03$, $p = 0.98$)—the architecture, not any particular graph, is what underperforms the LSTM there. At YC2, the identity control alone already recovers most of the ST-GCN family's advantage over LSTM (RMSE 11.77 vs. 14.50, an 18.8% reduction), and adding the impedance graph on top of it yields a further, non-significant 1.7% improvement ($t = 2.58$, $p = 0.061$). The choice among the three fixed graph alternatives does matter at YC2—impedance significantly outperforms both binary and correlation at the uncorrected level (Section 4.2)—but the larger and more robust effect in this dataset is which temporal architecture is used, not whether a graph, or which one, is attached to it. This reframes rather than overturns the original question: SHIC's near-uniform correlation structure ($|r| = 0.94$–0.99) leaves little discriminatory information for neighborhood aggregation to

exploit, so no graph, including a well-motivated impedance graph, can outperform a model that ignores cross-node structure altogether; YC2's wide correlation range (0.02–0.98) is where graph choice starts to matter, but even there the effect is smaller, and less statistically robust once corrected for multiple comparisons (Section 4.10), than the effect of the underlying sequence model. For industrial energy-management applications, this suggests two things rather than one: first, that the choice of temporal forecasting architecture should be validated locally—against a graph-free control—before assuming a graph-based model is warranted; and second, more tentatively, and only where an architecture with message-passing capacity is already justified, that checking inter-bus correlation heterogeneity before investing effort in a specific graph construction is a hypothesis motivated by these two sites, not a validated general screening rule (Section 5.5). Section 4.11 checks the graph-recovery side of this independently: a graph learned from scratch, with no electrical information at all, shows a statistically detectable resemblance to the impedance graph only at YC2, and not at SHIC.

5.2. Implications of Contemporaneous Electrical Coupling

The temporal-coupling analysis gives the forecasting results an electrical interpretation. For most representative bus pairs, the maximum cross-correlation of first-differenced active power falls at zero lag at the one-minute sampling resolution. This fits with how fast voltage and current relationships evolve within a low-voltage distribution network relative to the measurement interval. Thus, the graph's role in this application is better understood as contemporaneous information exchange among electrically related nodes than as a representation of any minute-scale propagation delay. The analysis also separates electrical coupling from shared production scheduling: some bus pairs show high raw load correlation but weak correlation once differenced. This distinction is itself an argument for physically informed adjacency, wherever reliable electrical information can be recovered from the metering system.

5.3. Physical Interpretability of the Learned Forecasting Response

The input-perturbation analysis shows that the ST-GCN built on the impedance-informed graph produces a sensitivity pattern that tracks the supplied impedance graph far more closely than the LSTM's response does. This should not, however, be read as independent recovery of network topology, since the ST-GCN architecture constrains information exchange through the adjacency matrix by design. Its practical value lies elsewhere, in interpretability: the dependence of a target forecast on neighboring measurements can be checked against an impedance-informed graph. For industrial energy-management and smart-distribution monitoring, this gives an additional validation layer beyond aggregate error metrics and can help identify whether the model uses measurement relationships that are plausible from an electrical-system perspective. A further comparison (Section 4.11) shows that a fully data-driven adaptive graph reaches comparable accuracy at YC2, but with higher run-to-run variance and without the same impedance-based interpretability, indicating that the impedance-informed graph's advantage lies less in raw accuracy than in stability and physical accountability.

5.4. Implications for Industrial Energy Monitoring and Metering

The SHIC metering artifact identified in Section 4.8 has a direct implication for energy-monitoring system design. The hub measurement aggregates grid-side and rooftop photovoltaic current on a shared channel, so the current used for branch-impedance regression does not represent a clean upstream reference. The affected edges also show the lowest regression fit quality at that site. For future industrial metering retrofits, distributed gen-

eration, storage, and major branch loads should therefore be measured separately when the same data are expected to support impedance-informed network identification. More generally, impedance-derived graph weights should be used only after checking the sign, magnitude, fit quality, and consistency with the available electrical configuration.

For an operator considering graph-based short-term load forecasting, these results suggest a practical sequence. Where a validated feeder model exists, use the known electrical topology or impedance directly. Where it does not, electrical relationships can be estimated from synchronized meter data, but the resulting parameters should pass plausibility and data-quality checks before use and should be re-estimated from the training split alone to rule out validation/test leakage into the graph. Next, check the heterogeneity of inter-bus relationships to gauge whether graph-based aggregation is likely to add anything. Finally, compare models with repeated training runs and statistical testing against both an LSTM baseline and a graph-free (identity-adjacency) ST-GCN control, since initialization variability alone can be large enough to change a single-run conclusion, and much of a graph model's apparent advantage may come from the sequence-modeling architecture rather than the graph itself. This sequence ties forecasting-model selection to both the quality of the energy-monitoring infrastructure and the physical structure of the distribution network.

5.5. Limitations

This study has several limitations. First, only two sites were available for a controlled comparison; while their contrasting correlation structure was informative precisely because they differed, two sites cannot establish how the correlation-heterogeneity hypothesis of Section 5.1 generalizes across industrial sectors, facility sizes, or climates. A third site (DSPM, Section 3.1) with sufficient overlapping data would be a natural next validation point. The two sites also differ in several respects beyond correlation structure—data span (82 vs. 50 days), node count (six vs. nine), missingness patterns, and the presence of on-site photovoltaic metering at YC2 but not SHIC—any of which could in principle contribute to the observed divergence instead of, or alongside, correlation heterogeneity; with only two sites, we cannot fully separate these factors, and we flag correlation heterogeneity as the most parsimonious explanation we identified rather than the only possible one. Second, forecasting was evaluated at a single horizon (15 min ahead, single-step); whether the site-dependent reversal, or the graph's relative advantage, holds at longer horizons is untested. Third, the hyperparameter screen (Section 4.9) found directionally consistent but not statistically significant improvements with only five confirmation seeds; a larger seed budget could resolve whether these effects are genuine. Fourth, the statistical significance testing in Sections 4.2 and 4.10 is a paired t-test on seed-level RMSE ($n = 5$); we now additionally apply a Bonferroni correction across the 22 comparisons performed (Table 5), which most of the YC2 "graph beats LSTM" claims do not survive (Section 4.10). A Diebold–Mariano test on per-timestep residuals, or a block-bootstrap confidence interval on the test-period forecast errors, would have higher statistical power than either the corrected or uncorrected t-test and remains as future work. Fifth, the Granger-causality results in Section 4.5 are computed on very large samples (tens of thousands of minutes), which inflates statistical significance for even small effect sizes; we rely on the effect sizes (peak lag and peak correlation) rather than the p-values alone for interpretation. Sixth, the chronological test windows are short in absolute terms (12 days at SHIC, 7.5 days at YC2, roughly 15% of each site's data), so the reported errors reflect a single realized period rather than an average over multiple periods; a rolling-origin (walk-forward) evaluation with several folds would test whether the site-dependent result is specific to this window and is a natural extension once additional compute budget is available. Finally, the graph construction assumes a hub–leaf topology inferred from the metering hierarchy rather than

verified from a complete single-line diagram, and the voltage–current regression yields an effective impedance proxy rather than a full complex line impedance. Facilities with meshed, multi-hub, or strongly phase-unbalanced internal distribution would require an extended formulation and additional electrical measurements.

5.6. Future Research Directions

Future work should extend the present evaluation in three directions. First, the relationship between inter-bus heterogeneity and graph-model benefit should be assessed across a larger set of industrial facilities, voltage levels, operating schedules, and distributed-energy-resource configurations. Second, dynamic or adaptive graph formulations could be evaluated to determine whether time-varying electrical and operational relationships improve forecasting at sites where a fixed adjacency matrix is insufficient. Third, multi-horizon forecasting should be studied for time scales directly relevant to energy-management actions such as demand-response scheduling, peak-load management, and energy-storage dispatch. These extensions would clarify how impedance-informed graph forecasting can be integrated into practical industrial energy-management systems.

6. Conclusions

This paper proposed and evaluated an impedance-informed graph construction for short-term electrical load forecasting at two industrial distribution networks, using minute-resolution multi-bus PT/CT measurements, and benchmarked it against binary, correlation-based, self-adaptive, and graph-free (identity) alternatives, an LSTM baseline, and a naive persistence baseline, under a common chronological evaluation protocol with five random seeds and adjacency matrices built exclusively from the training split. Graph structure helped only where the measured inter-bus relationships offered useful spatial discrimination, and even there, most of the benefit traced to the ST-GCN architecture rather than to the specific graph supplied to it. At YC2, every ST-GCN variant outperformed the LSTM baseline at the conventional significance level, and the impedance-informed model gave among the lowest RMSE, a 20.2% reduction relative to the untuned LSTM (13.1% relative to a tuned LSTM, Section 4.9); however, a graph-free identity control already captured an 18.8% reduction by itself, and none of these comparisons survived Bonferroni correction for the number of tests performed (Section 4.10)—a qualification we regard as central to the paper's contribution, not a weakening of it. At SHIC, the LSTM stayed more accurate than every ST-GCN variant, including the identity control, indicating an architecture-level rather than graph-level effect at that site. This split tracked whether the inter-bus correlation structure was heterogeneous or nearly uniform, though we treat this as a hypothesis from two sites rather than a general rule (Section 5.5). The mechanistic analyses added further, more qualified support. Coupling at the one-minute resolution was predominantly contemporaneous, and ST-GCN's forecasting sensitivity corresponded more closely to the impedance-informed graph than that of LSTM did. A self-adaptive graph learned without any electrical information showed a statistically detectable, but seed-variable and imperfect, resemblance to the impedance-informed graph at YC2 only, reached comparable forecasting accuracy there, but did so with higher run-to-run variance. A practical limitation also emerged: meter configurations that aggregate grid and photovoltaic currents can undermine the reliability of impedance identification, and edge-level regression quality is itself unstable across time for some bus pairs when graphs are built from the training split alone (Section 4.8). Taken together, these findings support a reliability-aware approach in which the temporal architecture, the graph construction (if any), meter-data quality, and repeated, corrected-for-multiple-comparisons forecasting evaluation are considered together rather than in isolation. An impedance-informed graph is worth constructing only where the

metered network offers heterogeneous, exploitable structure and where the specific graph, not just a message-passing architecture in general, demonstrably adds value beyond that architecture alone. Such an approach is directly relevant to industrial energy management, smart metering, and data-driven monitoring of distribution networks, where accurate load forecasts need to be paired with a realistic accounting of which modeling choice is actually responsible for the accuracy gain.

Author Contributions: Conceptualization, S.P.; methodology, S.P.; software, S.P.; validation, S.P. and Y.K.; formal analysis, S.P.; investigation, S.P.; resources, S.P. and Y.K.; data curation, S.P. and Y.K.; writing—original draft preparation, S.P.; writing—review and editing, Y.K.; visualization, S.P.; supervision, Y.K.; project administration, Y.K.; funding acquisition, Y.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Korea Institute of Energy Technology Evaluation and Planning (KETEP) and the Ministry of Climate, Energy & Environment (MCEE) of the Republic of Korea (No. RS-2022-KP002749).

Data Availability Statement: The industrial measurement data are not publicly available because they contain facility-specific operational information. They may be made available by the corresponding author upon reasonable request and subject to the applicable data-sharing restrictions. The analysis workflow and non-confidential diagnostic outputs may be shared through a project repository or by the corresponding author where permitted.

Acknowledgments: During manuscript preparation, the authors used Claude Sonnet 5 (Anthropic) for English-language editing and formatting assistance. The authors reviewed and revised all generated text and take full responsibility for the manuscript's content.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

Abbreviation	Definition
CT	Current transformer
DCRNN	Diffusion convolutional recurrent neural network
GNN	Graph neural network
LSTM	Long short-term memory
LV	Low voltage
MAE	Mean absolute error
MAPE	Mean absolute percentage error
NRMSE	Normalized root-mean-square error
PT	Potential (voltage) transformer
PV	Photovoltaic
RMSE	Root-mean-square error
SD	Standard deviation
sMAPE	Symmetric mean absolute percentage error
ST-GCN	Spatio-temporal graph convolutional network

References

1. Habbak, H.; Mahmoud, M.; Metwally, K.; Fouda, M.M.; Ibrahem, M.I. Load Forecasting Techniques and Their Applications in Smart Grids. *Energies* **2023**, *16*, 1480. [CrossRef]
2. Hachache, R.; Labrahmi, M.; Grilo, A.; Chaoub, A.; Bennani, R.; Tamtaoui, A.; Lakssir, B. Energy Load Forecasting Techniques in Smart Grids: A Cross-Country Comparative Analysis. *Energies* **2024**, *17*, 2251. [CrossRef]
3. Ungureanu, S.; Topa, V.; Cziker, A.C. Deep Learning for Short-Term Load Forecasting—Industrial Consumer Case Study. *Appl. Sci.* **2021**, *11*, 10126. [CrossRef]
4. Wen, L.; Zhou, K.; Yang, S.; Lu, X. Optimal load dispatch of community microgrid with deep learning based solar power and load forecasting. *Energy* **2019**, *171*, 1053–1065. [CrossRef]

5. Vontzos, G.; Laitsos, V.; Bargiotas, D.; Fevgas, A.; Daskalopulu, A.; Tsoukalas, L.H. Microgrid Multivariate Load Forecasting Based on Weighted Visibility Graph: A Regional Airport Case Study. *Electricity* **2025**, *6*, 17. [CrossRef]
6. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
7. Kong, W.; Dong, Z.Y.; Jia, Y.; Hill, D.J.; Xu, Y.; Zhang, Y. Short-Term Residential Load Forecasting Based on LSTM Recurrent Neural Network. *IEEE Trans. Smart Grid* **2019**, *10*, 841–851. [CrossRef]
8. Wu, W.; Liao, W.; Miao, J.; Du, G. Using Gated Recurrent Unit Network to Forecast Short-Term Load Considering Impact of Electricity Price. *Energy Procedia* **2019**, *158*, 3369–3374. [CrossRef]
9. Yu, B.; Yin, H.; Zhu, Z. Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. In Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI-18), Stockholm, Sweden, 13–19 July 2018; pp. 3634–3640. [CrossRef]
10. Li, Y.; Yu, R.; Shahabi, C.; Liu, Y. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. In Proceedings of the International Conference on Learning Representations (ICLR 2018), Vancouver, BC, Canada, 30 April–3 May 2018.
11. Arastehfar, S.; Matinkia, M.; Jabbarpour, M.R. Short-Term Residential Load Forecasting Using Graph Convolutional Recurrent Neural Networks. *Eng. Appl. Artif. Intell.* **2022**, *116*, 105358. [CrossRef]
12. Liao, W.; Bak-Jensen, B.; Pillai, J.R.; Wang, Y.; Wang, Y. A Review of Graph Neural Networks and Their Applications in Power Systems. *J. Mod. Power Syst. Clean Energy* **2022**, *10*, 345–360. [CrossRef]
13. Huang, N.; Wang, S.; Wang, R.; Cai, G.; Liu, Y.; Dai, Q. Gated spatial-temporal graph neural network based short-term load forecasting for wide-area multiple buses. *Int. J. Electr. Power Energy Syst.* **2023**, *145*, 108651. [CrossRef]
14. Wei, C.; Pi, D.; Ping, M.; Zhang, H. Short-term load forecasting using spatial-temporal embedding graph neural network. *Electr. Power Syst. Res.* **2023**, *225*, 109873. [CrossRef]
15. Mansoor, H.; Gull, M.S.; Rauf, H.; Shaikh, I.U.H.; Khalid, M.; Arshad, N. Graph Convolutional Networks based short-term load forecasting: Leveraging spatial information for improved accuracy. *Electr. Power Syst. Res.* **2024**, *230*, 110263. [CrossRef]
16. Giamarelos, N.; Zois, E.N. Boosting short term electric load forecasting of high & medium voltage substations with visibility graphs and graph neural networks. *Sustain. Energy Grids Netw.* **2024**, *38*, 101304. [CrossRef]
17. Jiang, Y.; Liu, M.; Li, Y.; Liu, Y.; Zhang, J.; Liu, Y.; Liu, C. Enhanced neighborhood node graph neural networks for load forecasting in smart grid. *Int. J. Mach. Learn. Cybern.* **2024**, *15*, 129–148. [CrossRef]
18. Lv, Y.; Wang, L.; Long, D.; Hu, Q.; Hu, Z. Multi-area short-term load forecasting based on spatiotemporal graph neural network. *Eng. Appl. Artif. Intell.* **2024**, *138*, 109398. [CrossRef]
19. Zhao, P.; Hu, W.; Cao, D.; Zhang, Z.; Liao, W.; Chen, Z.; Huang, Q. Enhancing multivariate, multi-step residential load forecasting with spatiotemporal graph attention-enabled transformer. *Int. J. Electr. Power Energy Syst.* **2024**, *160*, 110074. [CrossRef]
20. Pagnier, L.; Chertkov, M. Physics-Informed Graphical Neural Network for Parameter & State Estimations in Power Systems. *arXiv* **2021**, arXiv:2102.06349.
21. de Jongh, S.; Gielnik, F.; Mueller, F.; Schmit, L.; Suriyah, M.; Leibfried, T. Physics-informed geometric deep learning for inference tasks in power systems. *Electr. Power Syst. Res.* **2022**, *211*, 108362. [CrossRef]
22. Huang, B.; Wang, J. Applications of Physics-Informed Neural Networks in Power Systems—A Review. *IEEE Trans. Power Syst.* **2023**, *38*, 572–588. [CrossRef]
23. Wu, Z.; Pan, S.; Long, G.; Jiang, J.; Chang, X.; Zhang, C. Connecting the Dots: Multivariate Time Series Forecasting with Graph Neural Networks. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '20), Virtual, 23–27 August 2020; pp. 753–763. [CrossRef]
24. Wu, Z.; Pan, S.; Long, G.; Jiang, J.; Zhang, C. Graph WaveNet for Deep Spatial-Temporal Graph Modeling. In Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI-19), Macao, China, 10–16 August 2019; pp. 1907–1913. [CrossRef]
25. Wang, Y.; Rui, L.; Ma, J.; Jin, Q. A short-term residential load forecasting scheme based on the multiple correlation-temporal graph neural networks. *Appl. Soft Comput.* **2023**, *146*, 110629. [CrossRef]
26. Kipf, T.N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. In Proceedings of the International Conference on Learning Representations (ICLR 2017), Toulon, France, 24–26 April 2017.
27. Jin, G.; Liang, Y.; Fang, Y.; Shao, Z.; Huang, J.; Zhang, J.; Zheng, Y. Spatio-Temporal Graph Neural Networks for Predictive Learning in Urban Computing: A Survey. *IEEE Trans. Knowl. Data Eng.* **2024**, *36*, 5388–5408. [CrossRef]
28. Simeunović, J.; Schubnel, B.; Alet, P.-J.; Carrillo, R.E.; Frossard, P. Interpretable temporal-spatial graph attention network for multi-site PV power forecasting. *Appl. Energy* **2022**, *327*, 120127. [CrossRef]
29. Ngo, Q.-H.; Nguyen, B.L.H.; Vu, T.V.; Zhang, J.; Ngo, T. Physics-Informed Graphical Neural Network for Power System State Estimation. *Appl. Energy* **2024**, *358*, 122602. [CrossRef]
30. Zhang, W.; Zhu, F.; Lv, Y.; Tan, C.; Liu, W.; Zhang, X.; Wang, F.-Y. AdapGL: An adaptive graph learning algorithm for traffic prediction based on spatiotemporal neural networks. *Transp. Res. Part C Emerg. Technol.* **2022**, *139*, 103659. [CrossRef]

31. Jiang, R.; Wang, Z.; Yong, J.; Jeph, P.; Chen, Q.; Kobayashi, Y.; Song, X.; Fukushima, S.; Suzumura, T. Spatio-Temporal Meta-Graph Learning for Traffic Forecasting. *Proc. AAAI Conf. Artif. Intell.* **2023**, *37*, 8078–8086. [CrossRef]
32. Feng, A.; Tassiulas, L. Adaptive Graph Spatial-Temporal Transformer Network for Traffic Forecasting. In Proceedings of the 31st ACM International Conference on Information & Knowledge Management (CIKM '22), Atlanta, GA, USA, 17–21 October 2022; pp. 3933–3937. [CrossRef]
33. Watson, J.D.; Welch, J.; Watson, N.R. Use of Smart-Meter Data to Determine Distribution System Topology. *J. Eng.* **2016**, *2016*, 94–101. [CrossRef]
34. Iakovlev, S.; Evans, R.J.; Mareels, I. Low-Voltage Distribution Network Impedances Identification Based on Smart Meter Data. *arXiv* **2018**, arXiv:1809.06657. [CrossRef]
35. Tong, L.; Chai, W.; Wu, D. Topology and Impedance Identification Method of Low-Voltage Distribution Network Based on Smart Meter Measurements. *Front. Energy Res.* **2022**, *10*, 895397. [CrossRef]
36. Wang, C.; Lou, Z.; Li, M.; Zhu, Z.; Jing, D. Identification of Distribution Network Topology and Line Parameter Based on Smart Meter Measurements. *Energies* **2024**, *17*, 830. [CrossRef]
37. Xu, Z.; Jiang, W.; Xu, J.; Wu, L.; Wang, M.; Chen, J.; Chen, Y.; Zhao, J. Distribution Network Topology Identification Using Smart Meter Data and Considering the Same-Bus-Different-Feeder Condition. *IEEE Trans. Ind. Inform.* **2023**, *19*, 8600–8613. [CrossRef]
38. O'Malley, A.; Palacios-Garcia, E.J.; Hayes, B.P. Model-Free Voltage Estimation of Low Voltage Electrical Power Distribution Systems Using Smart Meter Data. In Proceedings of the 2024 IEEE PES Innovative Smart Grid Technologies Europe (ISGT EUROPE), Vienna, Austria, 3–6 June 2024; pp. 1–5. [CrossRef]
39. Hasan, R.; Mansouri, M.; Arkhangelski, J.; Tankari, M.A. Interpretable Multi-Sensor Fusion for Short-Term Energy Consumption Forecasting. *Energies* **2026**, *19*, 2230. [CrossRef]
40. Yu, H.; Ding, J.; Li, Y.; Wang, L.; Yuan, W.; Wang, X.; Zhang, F. Research on Short-Term Photovoltaic Power Prediction Method Using Adaptive Fusion of Multi-Source Heterogeneous Meteorological Data. *Energies* **2026**, *19*, 425. [CrossRef]
41. Diebold, F.X.; Mariano, R.S. Comparing Predictive Accuracy. *J. Bus. Econ. Stat.* **1995**, *13*, 253–263. [CrossRef]
42. Granger, C.W.J. Investigating Causal Relations by Econometric Models and Cross-Spectral Methods. *Econometrica* **1969**, *37*, 424–438. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.