Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Utility-Gated Personalized Federated Forecasting for Peak-Constrained Building Energy Management

Rui Wang

rekingbc@gmail.com

Microsoft (Canada)

Research Article

Keywords: federated learning, load forecasting, building energy management, battery storage, peer-to-peer learning, demand response

Posted Date: November 2nd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11065747/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Utility-Gated Personalized Federated Forecasting for Peak-Constrained Building Energy Management

Rui Wang

Microsoft Canada, Vancouver, British Columbia, Canada

Corresponding author: rekingbc@gmail.com

Abstract

Distributed building energy management depends on forecasts, yet federated forecasting is commonly evaluated without testing the decisions that those forecasts support. This study examines whether selective, communication-efficient collaboration improves both load prediction and battery operation. The proposed UtilityGossipmethod maintains separate private and collaborative low-rank adapters at each node. A peer adapter is incorporated only when randomized cross-fitting on local held-out data indicates a meaningful reduction in forecast error. Evaluation on public electricity and solar benchmarks covers longer horizons, alternative network topologies, node failures, and transmit-time noise. At the 96-step horizon, the method reduces Solar-Energy mean squared error by 1.59% relative to local training and avoids the greater than 3% degradation caused by unconditional federation on Electricity, using half the communication of federated averaging. Operational value is assessed by replaying every forecast source through the same frozen battery policy for a held-out week. An external projection prevents coordinated charging from increasing the running no-storage district peak. The proposed forecast supports a 19.34% cost reduction relative to no storage, and diagnostic replays confirm that forecast timing affects dispatch. Its additional control gain over a strong local forecast is not statistically resolved. The results support selective federation for distributed forecasting but show that lower prediction error does not necessarily yield proportional energy savings.

Keywords: federated learning; load forecasting; building energy management; battery storage; peer-to-peer learning; demand response

1 Introduction

Forecasts are operational inputs in modern building energy systems. They inform battery scheduling, demand response, and the coordination of distributed resources, where the timing of future demand can matter as much as its total amount [1–3]. The data needed to produce these forecasts are, however, distributed across buildings, owners, and control domains. Centralizing meter records can conflict with data-governance requirements and may be impractical when edge connectivity is limited.

Federated learning addresses this constraint by exchanging model information rather than raw measurements [4, 5]. Its use in load forecasting has shown that distributed smart-meter data can support collaborative models [6, 7]. The main difficulty is that buildings are not statistically interchangeable. Occupancy, weather exposure, equipment, scale, and rare events differ by site. A model update that improves average performance can therefore degrade an individual client's forecast. Central aggregation also introduces a coordination point that is undesirable in intermittently connected building networks.

Personalized and decentralized methods reduce dependence on a single global model, but a practical question remains unresolved: what evidence should a building require before using a model received from another site? Similarity or graph proximity can guide peer selection, yet neither establishes that the received information improves the receiver's forecast. This distinction is important in heterogeneous energy data, where superficially similar temporal patterns can arise from different operating conditions.

Prediction quality is only the first part of the engineering problem. Energy cost and battery dispatch depend on when errors occur and on how the controller responds. A small error near a tariff or demand peak may be more consequential than a larger error during an unconstrained period. Moreover, decentralized controllers can create coincident charging and increase aggregate demand even when each building acts rationally. Studies that connect forecasts with microgrid dispatch or photovoltaic-battery scheduling demonstrate the value of evaluating prediction within an operating problem [2, 3]. For distributed forecasting, however, the operational contribution of federation must be separated from the broader benefit of using any competent learned forecast.

This study therefore examines two questions. First, can receiver-side validation improve the accuracy–communication trade-off of serverless federated forecasting? Second, when assets and control policy are fixed, do the resulting forecasts improve distributed battery operation? The first question concerns selective information sharing; the second concerns the economic and physical value of the forecast that reaches the controller.

The proposed method, UtilityGossip, equips each node with a frozen PatchTST backbone and two low-rank adapters. The private adapter never leaves the node. The collaborative adapter can be exchanged with a neighbor, but the receiver uses randomized four-fold cross-fitting on local held-out data before accepting it. Accepted updates must show both a favorable gain-to-uncertainty ratio and a non-negligible effect size. This local decision replaces unconditional model mixing without transmitting raw data or private model state.

Operational value is measured with a parameter-shared battery policy that is trained once and then frozen. Every forecast source is replayed with the same assets, photovoltaic generation, tariff, carbon signal, and policy parameters. Persistence and temporally shuffled forecasts test whether the policy uses forecast timing. Local forecasting isolates the incremental contribution of federation, while a perfect forecast defines the remaining control gap. An external projection limits grid-supplied charging when coordinated actions would exceed the running no-storage district peak.

The contribution is consequently twofold. Methodologically, the study combines private and collaborative low-rank adaptation with receiver-side evidence for peer acceptance in a serverless network. From an energy-engineering perspective, it evaluates that forecasting design through controlled battery replay rather than inferring operational value from prediction error alone. The evaluation covers forecast accuracy, communication, longer horizons, topology, node failure, transmit-time noise, operating cost, grid imports, carbon emissions, and aggregate peak behavior.

Figure 1 shows the intended setting. The model network and the electrical network serve different purposes. Buildings use the first to exchange compact forecasting adapters and the second to supply power and aggregate the few values required by the peak guard. They are coupled only through the forecasts that inform battery actions.

The results distinguish forecasting benefit from control benefit. Selective sharing improves the observed federated accuracy–communication trade-off and avoids the negative transfer produced by unconditional federation on the Electricity benchmark. Learned forecast timing also improves battery operation. The incremental control gain over a strong local forecast remains unresolved, showing why prediction and operation should be reported as related but separate outcomes.

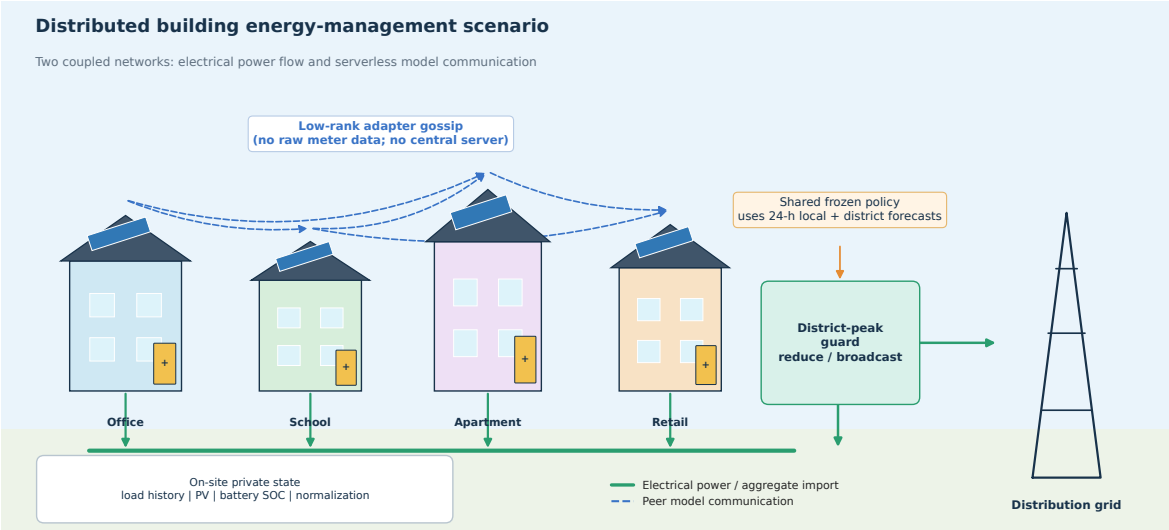


Figure 1: Application scenario. Heterogeneous buildings keep operational time series and asset states on site. Dashed links exchange low-rank forecasting adapters without a parameter server, while solid links represent electrical power and the aggregate district-import exchange used by the peak guard.

2 Related work

2.1 Federated forecasting under heterogeneity

FedAvgestablished the standard arrangement in which clients optimize locally and a server averages their updates [4]. Although this reduces the need to centralize raw data, one global model can perform poorly when client distributions differ. FedProxlimits local drift during optimization [8]; personalized methods instead retain client-specific behavior. FedFomo weights other clients according to local usefulness [9], FedAMP uses attentive inter-client messages [10], and Ditto learns a private model alongside a shared reference [11].

Heterogeneity is intrinsic to energy forecasting. Metered sites may share daily and seasonal structure while differing in scale, occupancy, equipment, and weather response. Data-driven energy studies increasingly model spatial and temporal dependence across multiple consumers or assets [12]. Federated recurrent models have likewise been applied to distributed smart-meter forecasting [6], with residential studies reporting both prediction accuracy and communication [7]. Personalized transfer, masking, and ensemble methods further address heterogeneous building sensing [13]. These approaches establish the value of cross-site information, but they do not imply that every received model is useful to every site.

2.2 Decentralized and parameter-efficient collaboration

Peer-to-peer optimization removes the aggregation server and changes the communication pattern. Decentralized stochastic-gradient methods show that network-wide learning is possible through local mixing [14], while DisPFL combines decentralized exchange with sparse personalized models [15]. Routine mixing still leaves a receiver exposed to updates that may be poorly matched to its own series. The present work focuses on this receiver-side decision rather than on peer exchange alone.

Parameter-efficient adaptation provides a complementary way to reduce network traffic. Low-rank adaptation represents a dense weight change with two thin matrices while keeping the pretrained

model fixed [16]. PatchTST supplies the shared temporal representation [17]; only small prediction-head corrections are adapted locally. Separating those corrections into private and collaborative components permits useful information to be exchanged without making the node's entire adapted model public.

2.3 Forecasts in building energy management

Building controllers use forecasts to shift storage and flexible demand across time. Model-predictive control has long incorporated future weather and load into building operation [1, 18]. Forecasting has also been coupled directly with photovoltaic-battery scheduling and urban microgrid dispatch [2, 3]. These studies motivate evaluating forecasts by their operational consequences rather than by statistical error alone.

Reinforcement learning can represent nonlinear control policies, but energy applications require explicit attention to rewards, physical constraints, and reproducibility [19, 20]. CityLearn and related benchmarks have helped standardize multi-building coordination studies [21, 22], and proximal policy optimization is widely used for continuous control [23]. The present study does not introduce a new control algorithm. The controller is held fixed so that changes in operation can be attributed to the forecast source rather than to policy retraining. Persistence, temporal shuffling, and an external peak constraint provide additional tests of timing value and safe aggregate behavior.

3 Selective serverless forecasting

3.1 Forecasting model and local state

Let $G = (V, E)$ denote the communication graph, with one forecasting node for each distributed client group. Node i observes a private chronological series and predicts the next H values from an input window $\mathbf{x}_{i,t}$:

$$\hat{\mathbf{y}}_{i,t} = f_{\theta_0,\phi_i}(\mathbf{x}_{i,t}) \in \mathbb{R}^H, \tag{1}$$

where θ_0 is a shared backbone trained before federation and then frozen. The node-specific parameters ϕ_i are small adapters attached to the prediction head. Input windows, targets, and normalization statistics remain local.

Each node maintains two adapters. The private adapter ϕ_i^p is trained only on local data and is never transmitted. The collaborative adapter ϕ_i^c also receives local updates and may incorporate a neighbor's adapter after a receiver-side utility test. At inference, locally fitted horizon-specific coefficients combine the two forecasts. The network is thus not driven toward a single consensus model. A node uses shared information only when that information improves prediction on data excluded from fitting the candidate mixture.

3.2 Low-rank adaptation

Let W_0 be the frozen prediction-head matrix and d the dimension of the backbone representation. The adapted weight is

$$W_i = W_0 + B_i A_i, \qquad A_i \in \mathbb{R}^{r\times d}, \quad B_i \in \mathbb{R}^{H\times r}, \tag{2}$$

where $r \ll \min(d, H)$. The LoRA scaling is absorbed into the factors and equals one in the rank-8 configuration. Separate factor pairs represent the private and collaborative adapters, and only the collaborative pair is transmitted.

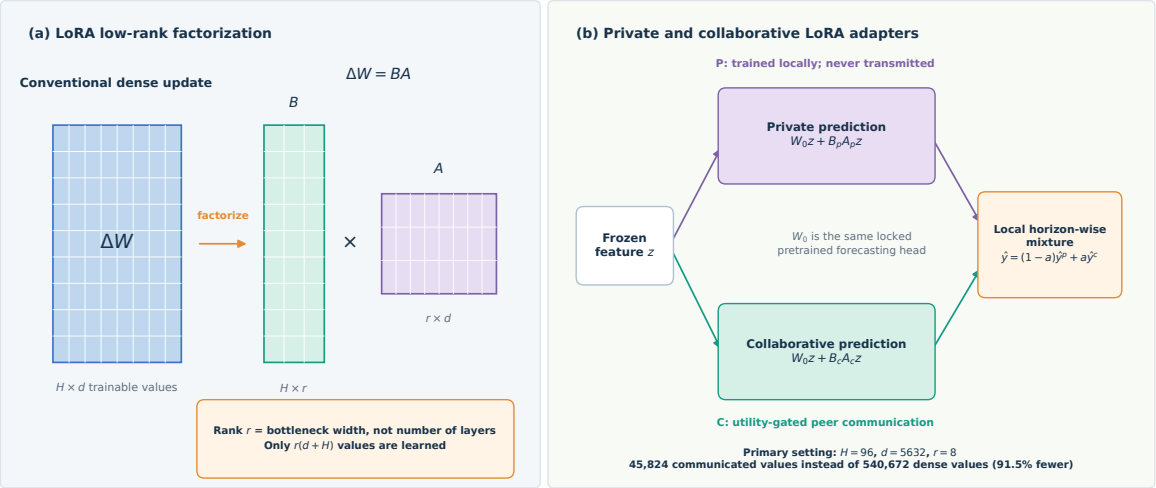


Figure 2: Low-rank adaptation (LoRA) framework. (a) A dense $H \times d$ update is represented by two thin matrices with bottleneck width r, reducing the trainable and communicated values to $r(d+H)$. Matrix cells are schematic. (b) The proposed method maintains separate private and collaborative rank-8 corrections around the same frozen forecasting head and combines their predictions locally.

The rank is the adapter bottleneck width, not the depth of the neural network. Matrix A_i maps the d backbone features to r correction coordinates, and B_i maps those coordinates to the H outputs (Figure 2). For $d = 5632$, $H = 96$, and $r = 8$, the adapter contains 45,824 values rather than the 540,672 values required by a dense head update, a reduction of 91.5%. The approximately 13-million-parameter PatchTST backbone is never exchanged.

3.3 Receiver-side utility test

Suppose that node i receives a collaborative adapter from node j. On node i's validation sample, the current and peer adapters produce forecasts $\mathbf{f}_i$ and $\mathbf{f}_j$. For horizon h, the current residual r_h and peer direction d_h are

$$r_h = y_h - f_{i,h}, \qquad d_h = f_{j,h} - f_{i,h}. \tag{3}$$

Coefficient a_h moves the candidate forecast from $f_{i,h}$ toward $f_{j,h}$. Ridge shrinkage toward a pooled coefficient stabilizes the horizon-specific estimate:

$$a_h = \mathrm{clip}\left(\frac{\langle r_h, d_h\rangle + \lambda a_{\mathrm{pool}}}{\|d_h\|_2^2 + \lambda}, 0, a_{\max}\right), \qquad \tilde{f}_{i,h} = f_{i,h} + a_h d_h. \tag{4}$$

The validation window follows adapter fitting and precedes the final test segment. For each utility decision, the receiver samples 128 rows without replacement and divides them randomly into four folds. Mixing coefficients are fitted on three folds, and the row-level reduction in squared error, g_n, is measured on the fourth. Each fold is held out once. The candidate passes only if

$$\frac{\bar{g}}{\mathrm{SE}(g)} > \tau_t, \qquad \frac{\bar{g}}{s_g} > \tau_d, \tag{5}$$

and the configured relative-gain floor is met. The first ratio compares mean gain with its uncertainty; the second excludes negligible standardized effects. The reported configuration uses $\tau_t = 2.0$,

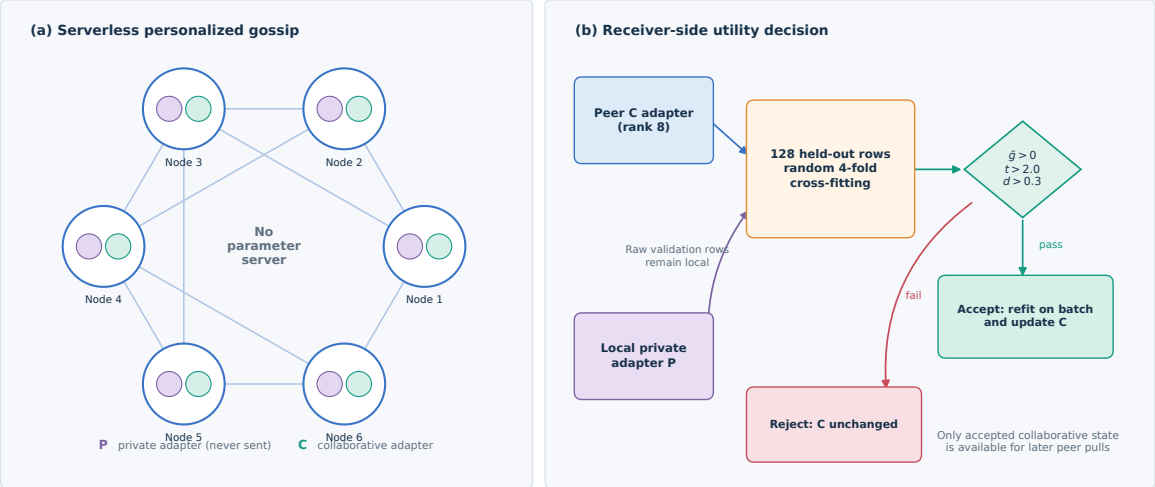


Figure 3: Serverless federated architecture and receiver-side utility decision. Every node maintains a non-transmitted private adapter (P) and a peer-visible collaborative adapter (C). A received C adapter is evaluated using randomized cross-fitting on locally held validation rows. Passing candidates update C, while rejected candidates leave it unchanged.

$\tau_d = 0.3$, $a_{\max} = 0.5$, and a zero relative-gain floor. The ridge parameter λ equals the mean energy of the peer direction. Once a candidate passes, the coefficients are refitted on all 128 sampled rows.

The gate is a model-selection rule rather than a sequential hypothesis test. It is applied adaptively over 100 rounds without multiplicity correction, so acceptance does not guarantee lower error on unseen data. Generalization is measured on a final test segment that is inaccessible to the gate.

3.4 Serverless collaboration

Each active node takes two local adapter steps and then pulls at most one neighbor adapter over the exponential graph (Figure 3). A passing correction is mixed with the receiver's collaborative correction and refactored to rank 8. A rejected correction leaves the receiver unchanged.

No parameter server coordinates the exchange. Unlike FedAvg, the protocol has no server broadcast; unlike FedGossip, receipt does not imply acceptance. A missing or straggling peer removes one collaboration opportunity from the round, while the private adapter remains available. Network cost is calculated from the number and precision of transmitted adapter values rather than from an assumed link rate.

3.5 Transmit-time differential privacy

The main experiments rely on data locality and make no differential-privacy claim. A separate sensitivity study clips each collaborative delta before transmission and adds Gaussian noise. Privacy loss is composed with a Rényi differential-privacy accountant [24]. The same utility gate evaluates the perturbed adapter and can reject an update made unhelpful by its direction or added noise.

Each release is clipped to unit ℓ_2 norm. The accounting composes 100 releases per node at $\delta = 10^{-5}$ over Rényi orders $\{1.1, 1.2, \ldots, 10.9, 11, \ldots, 63, 128, 256, 512\}$. The resulting guarantee is local to the release stream of one node. It does not include privacy amplification or additional exposure from subsequent peer re-sharing. The privacy results are interpreted only within this

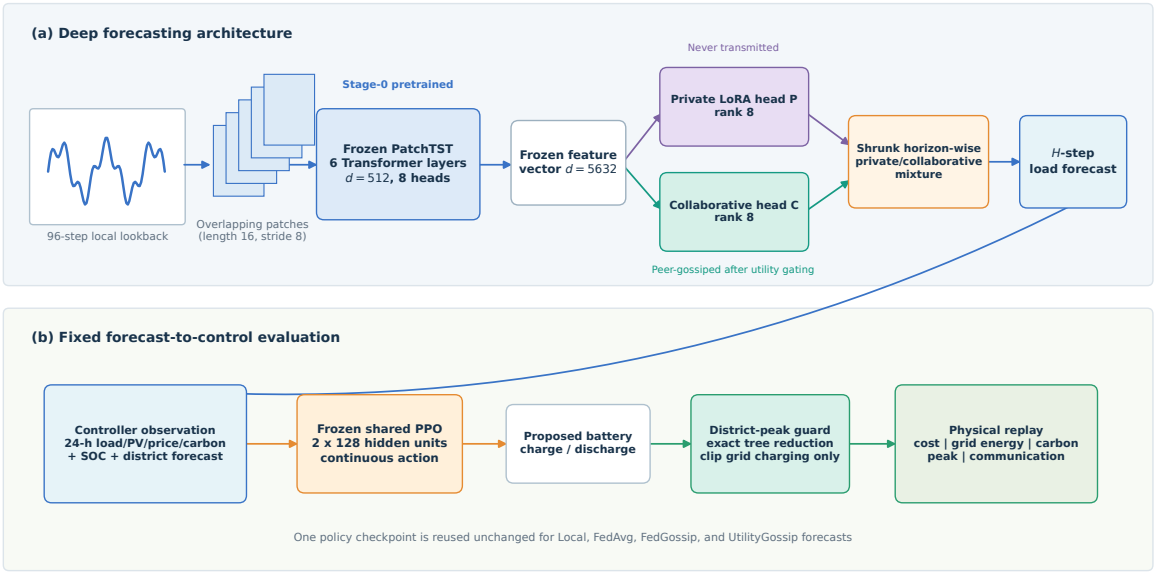


Figure 4: Deep forecasting and downstream control architecture. A frozen PatchTST encoder transforms the local lookback into a 5632-dimensional feature vector. Separate rank-8 private and collaborative LoRA heads are combined by a local horizon-wise mixture. The resulting forecast enters one frozen shared PPO policy before the district-peak projection and physical replay.

accounting scope.

4 Forecast-conditioned battery management

4.1 Battery and grid model

Each selected forecasting node is represented as a building portfolio with non-shiftable demand $l_{i,t}$, photovoltaic output $s_{i,t}$, and battery energy $e_{i,t}$. Over one hourly interval,

$$e_{i,t+1} = e_{i,t} + \eta_c p^c_{i,t} \Delta t - \frac{p^d_{i,t}}{\eta_d} \Delta t, \tag{6}$$

$$0 \le e_{i,t} \le E_i, \tag{7}$$

$$0 \le p^c_{i,t}, p^d_{i,t} \le P_i. \tag{8}$$

where $p^c_{i,t}$ and $p^d_{i,t}$ are charging and discharging power, η_c and η_d are the corresponding efficiencies, and E_i and P_i are the energy and power limits. Net grid import is

$$q_{i,t} = \max\left(0, l_{i,t} - s_{i,t} + p^c_{i,t} - p^d_{i,t}\right). \tag{9}$$

Export receives no credit, so negative net demand is clipped at zero. The operating objective includes imported energy, a building demand charge, carbon emissions, and a small throughput charge to discourage unnecessary cycling.

Asset sizes are derived from the calibration period, not the evaluation week. For portfolio i, the reference load is the 95th percentile of demand during the 672-hour calibration segment. Photovoltaic peak capacity and maximum battery power are each 50% of this reference; battery

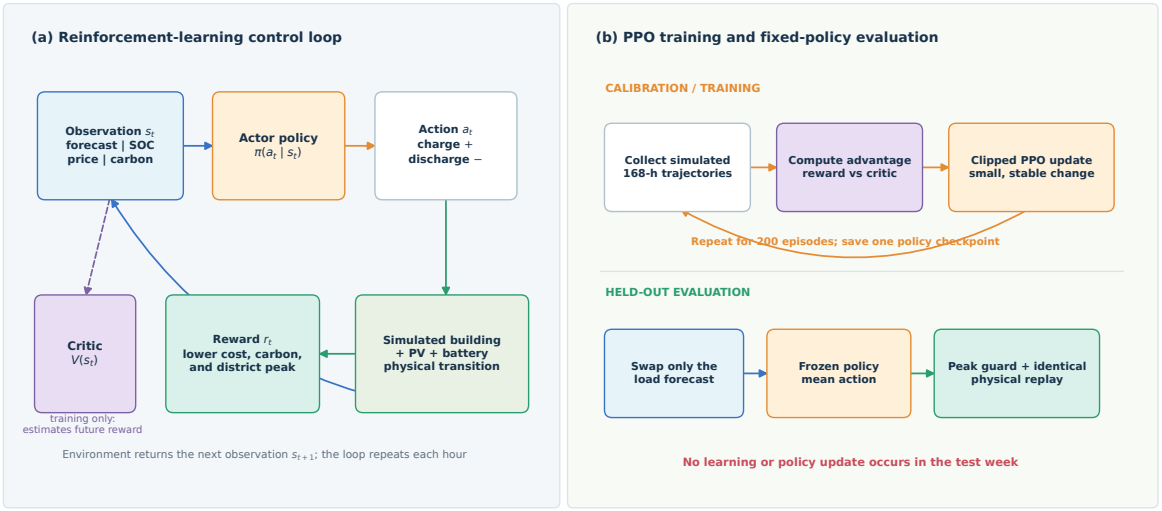


Figure 5: Reinforcement-learning control framework. (a) The hourly actor–environment loop converts forecasts and physical state into battery actions and receives outcome-based feedback. The critic is a training aid rather than a second controller. (b) PPO is trained only in the calibration segment. Evaluation freezes the resulting checkpoint, changes only the load-forecast source, and replays every method through the same policy, district-peak guard, and physical model.

energy capacity is two reference-load hours. Charging and discharging efficiencies are 0.95, and the battery starts empty.

Exogenous signals are fixed across forecast methods. The time-of-use tariff is \$0.11/kWh overnight, \$0.20/kWh from 07:00 to 16:00, and \$0.36/kWh from 16:00 to 21:00. Carbon intensity varies from 0.28 to 0.48 $\mathrm{kgCO_2}$/kWh, with a carbon price of \$0.05/$\mathrm{kgCO_2}$. The demand-charge coefficient is \$0.15/kW. These synthetic signals provide a matched forecast-to-control experiment rather than a reproduction of a specific utility tariff.

4.2 Shared forecast-conditioned policy

The controller is a parameter-shared actor–critic policy trained once with proximal policy optimization (PPO). It converts forecasts and the current physical state into a battery action; it does not produce the forecasts. The hourly observation contains current demand, battery state of charge, 24-hour forecasts of load, photovoltaic output, price, and carbon intensity, a quantized district-load forecast received through gossip, and a fixed phase encoding. The actor produces a continuous value in $[-1, 1]$, scaled by the portfolio's battery power limit.

During training, the simulator returns the next observation and a reward after each action (Figure 5). The reward credits reductions in energy cost, carbon cost, building demand charge, and district peak, while penalizing infeasible commands and battery throughput. The critic estimates future reward for learning purposes and does not issue a second action. PPO updates the actor and critic through a clipped objective that limits abrupt policy changes.

The policy is warm-started by imitating a forecast-conditioned battery heuristic. This avoids two uninformative behaviors observed in early training: inaction and repeated discharge requests from an empty battery. The actor and critic share two hidden layers of 128 units. PPO runs for 200 episodes of 168 hourly steps, with learning rate 3×10^{-4}, discount factor 0.99, generalized-advantage parameter 0.95, and four update epochs.

Training forecasts are independent of the methods evaluated later. One quarter of the trajectories use persistence; the remainder use perfect forecasts perturbed by zero-mean, unbounded Gaussian noise with a standard deviation sampled between 0 and 10% of the node's reference load. The policy therefore experiences weak and accurate forecasts without seeing outputs from LOCAL, FEDAVG, FEDGOSSIP, or UTILITYGOSSIP. During evaluation, only the load-forecast input changes (Figure 4); policy weights, other forecasts, asset states, the peak projection, and the physical model remain fixed.

4.3 Chronological isolation

The control record is divided into 672 calibration hours and a disjoint 168-hour evaluation week. Asset sizing, normalization, policy training, and checkpoint selection use only the calibration segment. Training origins are retained only when their complete 96-step forecast horizon ends before the evaluation boundary, preventing future targets from entering training through an overlapping window.

Checkpoint `9a20e4a3` is then frozen. The controller uses the first 24 hours of each 96-step forecast and is replayed with persistence, local, federated, shuffled, and perfect forecasts. No policy update occurs during evaluation.

4.4 District-peak rebound guard

Let $Q_t^0 = \sum_i \max(0, l_{i,t} - s_{i,t})$ denote no-storage district import and $\bar{Q}_t^0 = \max_{\tau \le t} Q_\tau^0$ its running peak. Before executing an action, the nodes perform an exact tree reduction and broadcast of baseline import, proposed import, and grid-supplied charging. If the proposed district import exceeds $\bar{Q}_t^0$, the guard computes

$$\alpha_t = \min\left(1, \max\left(0, \frac{\bar{Q}_t^0 - Q_t^{\text{prop}} + \sum_i p_{i,t}^{g,c}}{\sum_i p_{i,t}^{g,c} + \epsilon}\right)\right), \tag{10}$$

and scales only the grid-supplied charging components by α_t. Photovoltaic-supplied charging and battery discharge are unchanged. Within the aggregate model, this projection prevents coordinated charging from raising district import above the running no-storage peak. Keeping this protection outside the learned policy makes its action explicit and auditable, consistent with the need for physical constraint handling in reinforcement-learning energy management [20].

The guard communicates three float32 values per node and broadcasts one scale factor per step, adding 0.04032 MB during the evaluation week. This traffic is reported separately from forecast gossip. The constraint applies only to aggregate active power. It does not represent voltage, phase imbalance, reverse flow, feeder thermal limits, or protection operation.

5 Experimental setup

5.1 Datasets and chronology

Electricity and Solar-Energy form the energy-domain evaluation. Both are public multivariate benchmarks used in long-horizon time-series research [17, 25]. Electricity contains 321 hourly customer series; Solar-Energy contains 137 photovoltaic production series sampled every 10 minutes.

Traffic is an auxiliary cross-domain stress test, not evidence for building-energy performance. Its 862 hourly road-sensor series provide a larger heterogeneous population with recurring temporal structure but no energy-specific semantics. The dataset tests whether receiver-side selection remains

Table 1: Meaning of the five forecasting methods. All methods update rank-8 adapters around the same frozen backbone.

Method	Coordinator	Operation after local training	Purpose in the comparison
Local	None	The node keeps its locally trained adapter. No model state is sent or received.	Establishes the accuracy attainable without collaboration.
FedAvg	Central server	Every node uploads its adapter. The server computes an equal-weight average and broadcasts the same global adapter back to all nodes.	Represents standard centralized, unconditional federation.
FedGossip	None	Each active node pulls one uniformly selected neighbor adapter and merges it unconditionally with its current adapter. There is no separate private adapter.	Isolates plain serverless exchange without personalization or receiver-side acceptance.
Personalized GossipTime	None	Each node keeps private and collaborative adapters. It favors neighbors with similar spectral fingerprints, weights the received adapter by graph and staleness information, and blends private and collaborative forecasts locally.	Tests similarity-based personalized gossip without a held-out performance gate.
UtilityGossip	None	Each node keeps private and collaborative adapters, pulls one graph neighbor, and forms a candidate collaborative update. The update is used only if cross-fitted local validation shows a sufficiently large forecast-error reduction.	Tests whether receiver-side evidence can prevent harmful peer transfer while retaining useful sharing.

stable in a structurally different forecasting problem. It is excluded from battery control and from all energy-saving claims.

The original channels are grouped into 32 nodes through balanced clustering of fitting-period spectral fingerprints. Each node is therefore a client portfolio rather than an individual meter, sensor, or solar plant. Every method uses the same grouping and chronological partitions:

- 0–40%: pooled Stage-0 backbone and base-head training;
- 40–76%: federated/local adapter fitting;
- 76–80%: collaboration validation used by the utility gate;
- 80–100%: untouched forecasting test data.

Grouping features and normalization statistics are computed before the test period. The utility gate accesses only the 76–80% validation window and never the final 20%.

5.2 Forecasting comparisons

The five methods progress from no collaboration to centralized averaging, unconditional peer exchange, personalized heuristic exchange, and validation-gated exchange. Table 1 summarizes the operation that follows each local adapter update. The backbone, rank-8 parameterization, node groups, data partitions, local-step budget, and random seeds are held constant. Only the shared state, peer-selection rule, and acceptance rule differ.

Personalized GossipTimedenotes the personalized version of an earlier serverless design in this project rather than a generic class of gossip algorithms. It favors neighbors with similar spectral fingerprints, downweights stale information, and blends private and collaborative forecasts. It does not test whether a received adapter improves held-out error before merging it. Because active nodes advance synchronously in the primary runs, the staleness multiplier is neutral; spectral peer

selection, graph-based mixing, and the private–collaborative blend are its effective distinctions from FedGossip.

UtilityGossipreplaces heuristic trust with local predictive evidence. After pulling one collaborative adapter, the receiver evaluates a candidate mixture on its validation window and accepts it only when the cross-fitted error reduction passes the thresholds in Section 3. In this name, *utility* means lower validation loss for the receiving node. Electricity cost, energy savings, and utility-company revenue are not part of the gate; operational outcomes are measured separately by the frozen controller.

The primary horizon is $H = 96$; Solar-Energy is also evaluated at $H = 192$ and $H = 336$. Each configuration runs for 100 communication rounds with seeds 42, 43, and 44. FedAvgcommunication includes client uploads and server downloads, while peer methods count one adapter per completed pull. Megabytes are calculated from serialized float32 values and represent protocol traffic, not wall-clock latency.

5.3 Robustness, topology, and privacy

Robustness is examined under node unavailability, alternative peer graphs, and transmit-time noise. The resilience study makes 30% of nodes unavailable transiently or persistently while retaining stragglers. The topology study compares exponential, ring, random-regular, and Erdős–Renyi graphs. The privacy sensitivity study clips collaborative deltas and adds Gaussian noise at multipliers 1, 2, and 4, reporting the local composed privacy budget, forecast error, and gate acceptance rate.

5.4 Energy-management protocol

The operational study uses Electricity because it represents customer demand. All 32 nodes produce forecasts, and 16 portfolios participate in battery control. The policy observes the first 24 hours of each 96-hour forecast. It is calibrated for 672 hours and evaluated on the next 168 hours. One SharedPPOcheckpoint is reused for every method and seed.

The control references isolate different sources of value. No storage measures the total contribution of battery flexibility, and a forecast-free rule controller provides a simple benchmark. Persistence tests learned prediction against repetition of the latest observation. Temporal shuffling preserves the forecast values but destroys their timing. Localseparates cross-node sharing from learned forecasting, and the perfect forecast estimates the remaining cost gap.

5.5 Metrics and statistical analysis

Forecasting is evaluated by mean and worst-node mean squared error (MSE), mean absolute error (MAE), and transmitted megabytes. The control analysis reports electricity cost, grid imports, carbon emissions, aggregate district peak, building-level gains, sensitivity to forecast shuffling, and communication. Mean results are accompanied by sample standard deviations across seeds.

The primary analysis uses seeds 42, 43, and 44. After the experiment matrix was complete, six paired contrasts were selected for concise reporting: UtilityGossipversus Localand versus FedGossipon both energy datasets and the auxiliary Traffic dataset. Two-sided paired t tests use matched seeds, with Holm adjustment across these six post-hoc comparisons. The exploratory export also contains Welch tests, bootstrap intervals, and Cliff's delta for 138 metric–method pairs. With only three matched pairs, normality cannot be assessed credibly. The resulting p values are descriptive, and the interpretation places greater weight on effect magnitude and consistency across settings.

Table 2: Claim-bearing experiment matrix.

Study	Datasets	Methods/configurations	Seeds
Main forecasting	Electricity, Solar; Traffic (aux.)	5 methods, $H = 96$	3
Horizon robustness	Solar	5 methods, $H = 192, 336$	3
Resilience	Solar	2 methods, 2 failure modes	3
Topology	Solar	2 methods, 4 graphs	3
Privacy	Solar	Local, no-DP, 3 noise levels	3
Energy control	Electricity	4 forecast methods, shared policy	3

Table 3: Forecasting results over three seeds. Electricity and Solar-Energy form the energy-domain evaluation; Traffic is an auxiliary cross-domain stress test. Lower error and communication are better. Bold values are the best forecast errors within each dataset.

Dataset	Method	MSE	MAE	Worst-node MSE	Communication (MB)
Electricity	Local	0.2540 ± 0.0010	0.3453 ± 0.0010	0.5905 ± 0.0045	0.00
	FedAvg	0.2628 ± 0.0002	0.3542 ± 0.0002	0.6070 ± 0.0002	163.84
	FedGossip	0.2622 ± 0.0002	0.3537 ± 0.0003	0.6069 ± 0.0003	81.92
	Personalized GossipTime	0.2547 ± 0.0009	0.3459 ± 0.0009	0.5930 ± 0.0037	81.92
	UtilityGossip	**0.2535 ± 0.0012**	**0.3445 ± 0.0014**	**0.5892 ± 0.0017**	81.92
Solar	Local	0.4033 ± 0.0009	0.4283 ± 0.0016	0.4510 ± 0.0060	0.00
	FedAvg	0.4041 ± 0.0007	0.4297 ± 0.0008	0.4505 ± 0.0006	1173.09
	FedGossip	0.4046 ± 0.0007	0.4303 ± 0.0008	0.4510 ± 0.0014	586.55
	Personalized GossipTime	0.4035 ± 0.0009	0.4286 ± 0.0015	0.4510 ± 0.0053	586.55
	UtilityGossip	**0.3969 ± 0.0005**	**0.4213 ± 0.0009**	**0.4461 ± 0.0027**	586.55
Traffic (aux.)	Local	0.8009 ± 0.0006	0.5726 ± 0.0003	1.5178 ± 0.0009	0.00
	FedAvg	0.8013 ± 0.0002	0.5731 ± 0.0001	1.5182 ± 0.0003	1173.09
	FedGossip	0.8016 ± 0.0002	0.5732 ± 0.0003	1.5183 ± 0.0007	586.55
	Personalized GossipTime	0.8009 ± 0.0005	0.5727 ± 0.0002	1.5178 ± 0.0008	586.55
	UtilityGossip	**0.8004 ± 0.0005**	0.5727 ± 0.0002	**1.5169 ± 0.0009**	586.55

Configuration markers allow interrupted jobs to resume without rerunning completed cases. Duplicate forced reruns are removed before aggregation and testing.

The final suite contains 126 forecasting runs and 12 control replays. Table 2 gives the complete allocation.

6 Results

6.1 Forecast accuracy and communication

Electricity provides the clearest evidence of negative transfer (Table 3). FedAvgand FedGossipincrease MSE by 3.46% and 3.23%, respectively, relative to Local. UtilityGossipavoids this degradation, producing the lowest federated MSE and the lowest worst-node MSE. Its mean error is 3.33% below FedGossip($p_{\text{Holm}} = 0.0234$), whereas the 0.21% difference from Localis unresolved ($p_{\text{Holm}} = 0.3969$). The primary result on Electricity is therefore protection against harmful sharing rather than a large gain over local learning.

Solar-Energy shows a clearer benefit from collaboration. UtilityGossiplowers MSE by 1.59% relative to Localand by 1.90% relative to FedGossip; the corresponding Holm-adjusted paired values are 0.0096 and 0.0042. MAE declines by 1.65–2.10% across the reported comparisons.

The auxiliary Traffic test retains the same ordering, but the largest MSE reduction is only 0.16% and MAE is effectively unchanged from local training. This result indicates that the gate

Table 4: Post-hoc designated paired MSE contrasts over matched seeds. Traffic is an auxiliary cross-domain stress test. Positive differences favor UtilityGossip; p_{Holm} adjusts the family of six descriptive contrasts.

Dataset	Baseline	MSE difference	Reduction (%)	p_{Holm}
Electricity	Local	0.000523 ± 0.000848	0.21	0.3969
Electricity	FedGossip	0.008725 ± 0.001162	3.33	0.0234
Solar	Local	0.006410 ± 0.000488	1.59	0.0096
Solar	FedGossip	0.007677 ± 0.000354	1.90	0.0042
Traffic (aux.)	Local	0.000516 ± 0.000138	0.06	0.0463
Traffic (aux.)	FedGossip	0.001247 ± 0.000217	0.16	0.0297

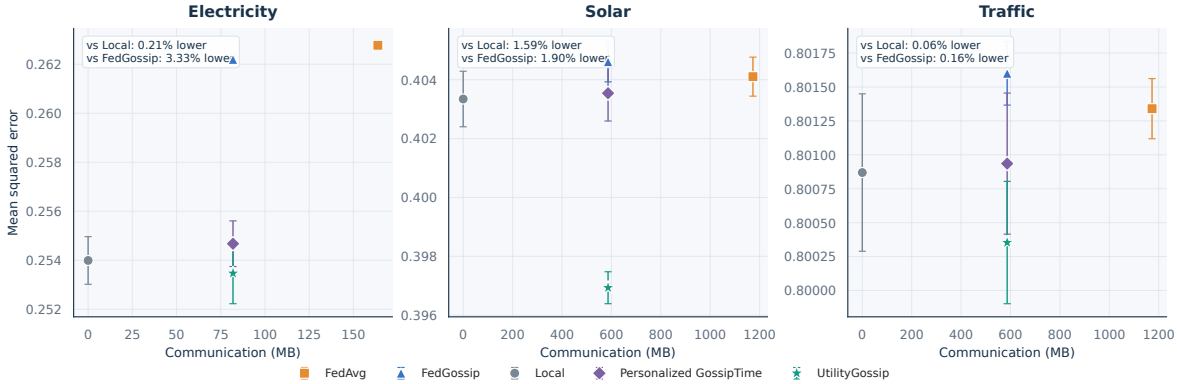


Figure 6: Forecast accuracy versus cumulative communication for all methods. Electricity and Solar-Energy are the energy-domain evaluations; Traffic is an auxiliary cross-domain stress test. Error bars show variation across seeds. Annotations report the proposed method's relative MSE reduction. Lower values are preferred on both axes.

remains well-behaved in a different sensor domain; it does not support an energy-performance claim. With three seeds, the paired tests describe consistency within the experiment rather than strong inferential evidence.

The accuracy gain does not require additional model traffic. UtilityGossipuses the same communication as FedGossipand exactly half that of FedAvg, which includes both upload and broadcast paths. It is consequently the best observed federated accuracy–communication point for each dataset in Figure 6.

6.2 Horizon robustness

The Solar-Energy ordering persists at longer horizons (Table 5). Relative MSE reductions range from 2.52% to 2.75% against Localand from 2.66% to 2.88% against FedGossipat $H = 192$ and $H = 336$. Communication remains equal to FedGossipand half that of FedAvg. The gain therefore survives the increase in forecast length and adapter size, although it is not monotonic in horizon.

6.3 Resilience and topology

Under 30% transient and persistent node unavailability, UtilityGossipremains 1.92% and 1.26% below FedGossip, respectively (Table 6). Communication is nearly identical between the peer

Table 5: Long-horizon Solar-Energy forecasting over three seeds.

Horizon	Method	MSE	Communication (MB)
192	Local	0.3864 ± 0.0005	0.00
	FedAvg	0.3867 ± 0.0001	1192.76
	FedGossip	0.3869 ± 0.0002	596.38
	Personalized GossipTime	0.3865 ± 0.0003	596.38
	UtilityGossip	**0.3758 ± 0.0003**	596.38
336	Local	0.4024 ± 0.0004	0.00
	FedAvg	0.4028 ± 0.0002	1222.25
	FedGossip	0.4030 ± 0.0004	611.12
	Personalized GossipTime	0.4027 ± 0.0002	611.12
	UtilityGossip	**0.3923 ± 0.0001**	611.12

Table 6: Resilience and topology sensitivity on Solar-Energy. The advantage is the relative MSE reduction of UtilityGossip over FedGossip under the same setting.

Study	Setting	FedGossip MSE	UtilityGossip MSE	Advantage (%)
Failures	Transient 30% failures	0.4015 ± 0.0009	**0.3938 ± 0.0006**	1.92
Failures	Persistent 30% failures	0.4013 ± 0.0009	**0.3963 ± 0.0004**	1.26
Topology	Exponential	0.4047 ± 0.0009	**0.3969 ± 0.0005**	1.92
Topology	Ring	0.4041 ± 0.0006	**0.3974 ± 0.0005**	1.67
Topology	Random regular	0.4046 ± 0.0007	**0.3971 ± 0.0004**	1.85
Topology	Erdős–Renyi	0.4038 ± 0.0005	**0.3971 ± 0.0001**	1.65

methods. Removing collaboration opportunities therefore weakens but does not reverse the observed advantage.

Graph structure has little effect within the tested set. UtilityGossipMSE varies by 0.107% across exponential, ring, random-regular, and Erdős–Renyi topologies and remains 1.65–1.92% below matched FedGossip. No single graph is required for the result.

6.4 Privacy–utility trade-off

Gaussian perturbation reduces the acceptance rate from 32.9% to approximately 20%, yet MSE remains stable across the three noise multipliers. The noisy settings also outperform the non-private run in this experiment. This pattern may reflect regularization combined with rejection of harmful candidates, but the study is not designed to establish that mechanism. The strongest tested setting still has $\epsilon = 14.13$; the result demonstrates tolerance to the tested perturbations, not strong privacy at negligible utility cost.

6.5 Operational value of forecasting

All learned forecasts support useful battery operation relative to no storage (Table 8). With UtilityGossip, the frozen policy reduces cost by 19.34%, grid imports by 3.24%, and carbon emissions by 9.89%. Replacing this forecast with persistence increases cost by 1.28%; temporally shuffling the same forecast values increases it by 8.74%. The associated action-sensitivity measures confirm that the policy changes battery dispatch when forecast timing changes.

UtilityGossipleaves the smallest cost gap to the perfect-forecast replay (0.129%) and improves cost savings over both unconditional federated baselines. Operational communication is 49.4% lower than FedAvgand equal to FedGossip.

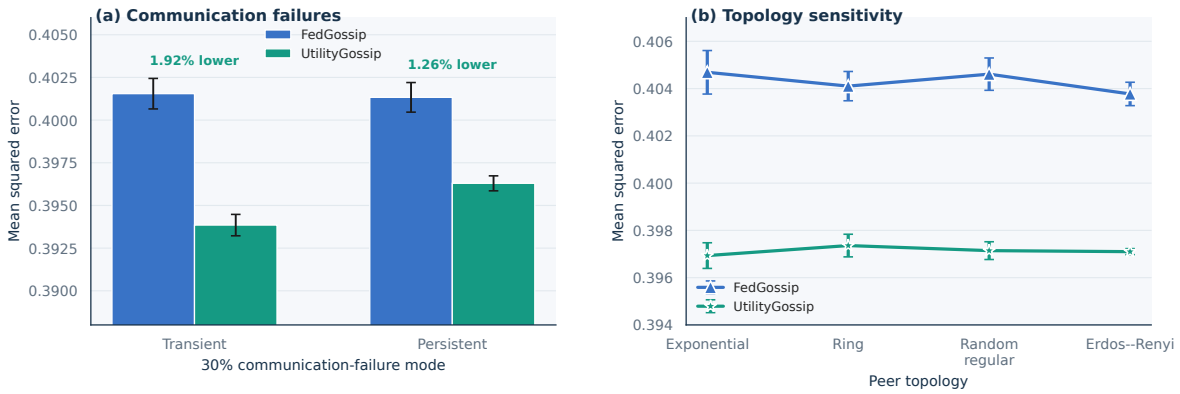


Figure 7: Resilience of selective collaboration: (a) forecast MSE under transient and persistent 30% communication failures and (b) sensitivity to peer topology. Error bars show variation across seeds.

Table 7: Transmit-time differential-privacy sensitivity on Solar-Energy ($\delta = 10^{-5}$; 100 unit-clipped releases per node).

Configuration	ϵ	MSE	Acceptance (%)
Local (no release)	–	0.4033 ± 0.0009	0.0
UtilityGossip, no DP	–	0.3969 ± 0.0005	32.9
UtilityGossip, $\sigma = 1$	96.12	0.3917 ± 0.0004	20.4
UtilityGossip, $\sigma = 2$	35.08	0.3916 ± 0.0003	20.1
UtilityGossip, $\sigma = 4$	14.13	0.3917 ± 0.0005	20.0

The comparison with LOCALis much closer. Across matched seeds, UTILITYGOSSIPchanges no-storage cost savings by 0.037 ± 0.043 percentage points, with one of the three differences favoring the local forecast. The paired value is $p = 0.275$. The experiment therefore does not resolve an additional control benefit from federation over local learned forecasting, despite the lower mean forecast error.

The peak guard activates during approximately 16.0% of control steps and adds 0.04032 MB of communication. District-peak differences remain at numerical tolerance around the running no-storage peak. This result applies only to the shared-policy replays protected by the projection; the separate forecast-free rule controller is not guarded.

7 Discussion

7.1 Selective sharing under client heterogeneity

The forecasting results support selective rather than unconditional collaboration. Electricity is the clearest case: conventional aggregation and plain gossip both degrade the local model, whereas receiver-side validation restores performance to approximately the local level. Solar-Energy provides a different pattern, in which shared information improves on local training. Together, the two datasets show why one collaboration rule need not have the same effect across client populations.

This behavior is consistent with personalized federated learning, where a population-wide model is not assumed to suit every client [9–11]. The distinctive element here is the decision point. Peer selection determines which update arrives; the utility gate determines whether the receiver uses it. The latter remains a fallible model-selection step because it relies on sampled validation data and repeated thresholds. Its value is empirical: the advantage over plain gossip persists across energy

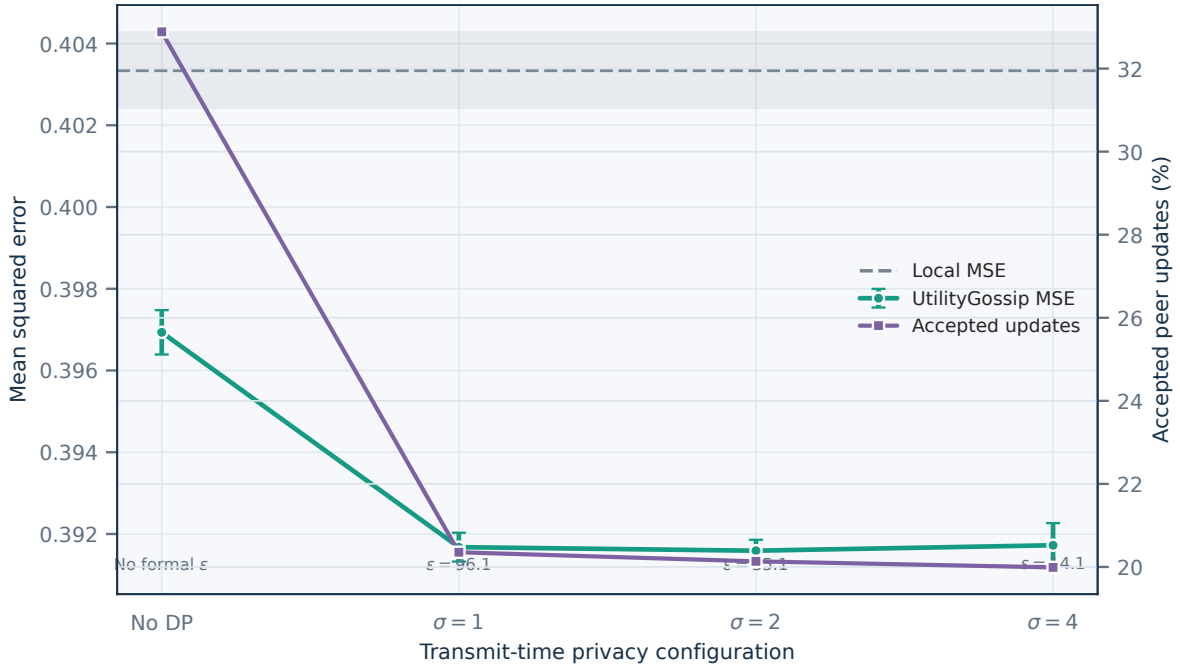


Figure 8: Forecast error and collaboration acceptance under transmit-time Gaussian noise. The dashed line and band show the local-only mean and seed variation. Privacy annotations report local per-node accounting at $\delta = 10^{-5}$.

Table 8: Forecast-to-control performance under one frozen shared PPO policy over three seeds. Savings are relative to no storage; peak values within numerical tolerance of zero indicate that the district-peak guard preserved the no-storage peak ceiling.

Forecast	Cost saving (%)	Grid saving (%)	Carbon saving (%)	Gain vs persistence (%)	Oracle gap (%)	Peak reduction (%)	Operational MB
Local	19.30 ± 0.02	3.21 ± 0.02	9.86 ± 0.02	1.23 ± 0.03	0.174 ± 0.026	$-1.45 \times 10^{-8} \pm 1.68 \times 10^{-8}$	2.00
FedAvg	18.99 ± 0.03	3.02 ± 0.01	9.62 ± 0.02	0.85 ± 0.03	0.562 ± 0.031	$-2.46 \times 10^{-8} \pm 4.02 \times 10^{-8}$	165.84
FedGossip	19.01 ± 0.02	3.04 ± 0.01	9.64 ± 0.01	0.87 ± 0.02	0.537 ± 0.019	$-7.35 \times 10^{-9} \pm 6.75 \times 10^{-9}$	83.92
UtilityGossip	**19.34 ± 0.03**	**3.24 ± 0.01**	**9.89 ± 0.02**	**1.28 ± 0.04**	**0.129 ± 0.036**	$-1.05 \times 10^{-8} \pm 1.02 \times 10^{-8}$	83.92

datasets, longer horizons, node failures, and the tested graphs. Improvement over local training is less uniform, so a private local model remains an essential fallback.

7.2 From forecast error to control value

The battery replays confirm that forecast timing has operational value. Persistence and temporal shuffling both increase cost, with the larger penalty from shuffling showing that the policy is not responding only to the average forecast level. It uses information about when demand will occur.

The additional value of federation is much less certain. The matched LOCALand UTILITYGOSSIPreplays produce nearly identical savings, and the difference is not statistically resolved. This finding is not inconsistent with the forecasting result. Near a strong local model, a reduction in aggregate MSE can occur at hours that do not change battery decisions. A timing error near an expensive or high-demand period may be far more consequential than a larger error elsewhere.

Forecast error and operating value are therefore connected through the controller, tariff, asset state, and constraint set. Forecast-aware building control has long depended on future loads and exogenous signals [1, 18]. Holding the policy fixed makes this mapping observable and prevents

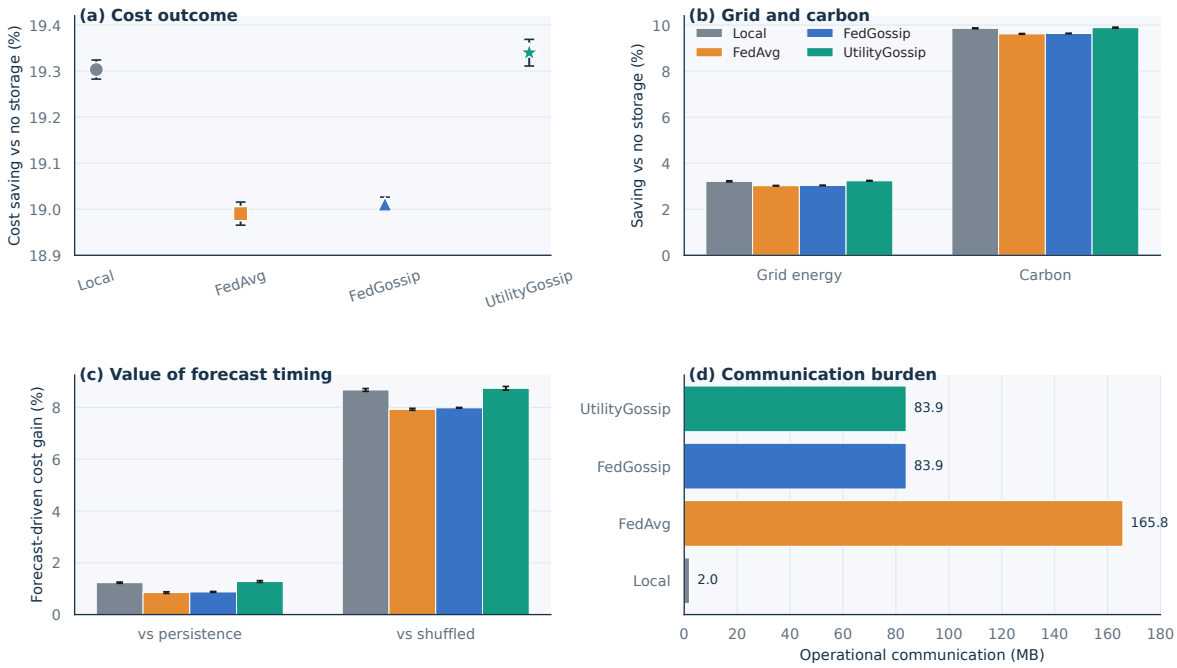


Figure 9: Operational outcomes under one frozen shared policy: (a) cost savings, (b) grid-energy and carbon savings, (c) value of forecast timing relative to persistence and shuffled forecasts, and (d) operational communication.

savings available under any competent forecast from being attributed to federation.

7.3 Communication and deployment considerations

Communication is separated by function because model exchange, district forecast exchange, peak protection, and one-time policy training have different lifecycle roles. Method-dependent traffic is dominated by adapter exchange. The peer protocols use half the bytes of FedAvgbecause they omit the return broadcast of a global model. Receiver-side validation changes local computation but not the payload: UtilityGossiptransmits the same rank-8 adapter as FedGossip.

This trade-off is attractive when communication is intermittent or metered and local validation is comparatively inexpensive. It is not a complete network cost model. Link latency, packet overhead, retransmission, and device energy were not measured. The privacy study is similarly scoped. The gate tolerates the tested Gaussian perturbations, but the reported privacy loss remains high and covers one node's transmitted stream rather than later peer re-sharing. The evidence supports noise tolerance, not deployment-ready privacy.

7.4 Operational safeguards and deployment path

Mean performance does not ensure that every participant benefits. UtilityGossiphas the lowest observed worst-node forecast MSE, but one controlled building records a small cost loss. An operational program would therefore require participant-level safeguards, such as opt-out rules, compensation, or explicit fairness constraints, rather than relying only on an aggregate reward.

Peak protection is kept outside the learned policy so that the constraint is auditable and remains active when a forecast or action is poor. The projection prevents synchronized grid charging from raising aggregate demand above the running no-storage peak in the active-power model. It does

not address voltage, phase balance, reverse flow, thermal loading, or protection coordination; those requirements need a distribution power-flow model and network-specific limits.

The resulting architecture separates forecast collaboration, policy execution, and network protection. This separation provides a practical path for staged validation: longer seasonal replays and independent building portfolios should precede tariff and asset-sizing sensitivity studies, followed by distribution-network co-simulation. Throughout that progression, forecast source should remain the controlled variable when the purpose is to identify its contribution to energy management.

8 Conclusion

This study evaluates personalized federated forecasting as an input to distributed battery management rather than as an isolated prediction task. UtilityGossipseparates private and collaborative low-rank adaptation and requires local held-out evidence before a peer update is used. Across the energy benchmarks, this design improves the observed federated accuracy–communication trade-off, avoids the negative transfer produced by unconditional sharing on Electricity, and uses half the communication of FedAvg.

Replaying each forecast through the same frozen policy shows that learned forecast timing contributes to battery value. The UtilityGossipforecast supports a 19.34% cost reduction relative to no storage while the external projection maintains the running no-storage district peak. The incremental control gain over a strong local forecast is not resolved, however. Improved forecast error alone is therefore insufficient evidence of additional energy savings from federation.

The main engineering implication is that collaboration, control, and network protection should be evaluated separately and then connected through a controlled forecast-to-operation experiment. This structure preserves the benefits of selective information sharing while making clear which outcomes arise from battery flexibility, learned forecasting, and federation.

Data availability

The energy-domain Electricity and Solar-Energy benchmarks and the auxiliary Traffic benchmark are publicly available through the Long-Term Time-Series Forecasting benchmark collection. The curated seed-level results, aggregate tables, statistical tests, and a checksum manifest used in this manuscript are included in the accompanying repository.

Conflict of interest

The author declares no conflict of interest.

Ethics statement

This computational study used public, non-identifiable benchmark time series and did not involve human participants or animals. Ethics approval was not required.

Acknowledgements

Funding and acknowledgement information will be added before submission.

References

[1] Frauke Oldewurtel, Alessandra Parisio, Colin N. Jones, Dimitrios Gyalistras, Markus Gwerder, Vanessa Stauch, Beat Lehmann, and Manfred Morari. Use of model predictive control and weather forecasts for energy efficient building climate control. *Energy and Buildings*, 45:15–27, 2012. doi: 10.1016/j.enbuild.2011.09.022.

[2] Mahtab Kaffash, Kenneth Bruninx, and Geert Deconinck. Data-driven forecasting of local pv generation for stochastic pv-battery system management. *International Journal of Energy Research*, 45(11):15962–15979, 2021. doi: 10.1002/er.6826.

[3] Navid Shirzadi, Fuzhan Nasiri, Claude El-Bayeh, and Ursula Eicker. Optimal dispatching of renewable energy-based urban microgrids using a deep learning approach for electrical load and wind power forecasting. *International Journal of Energy Research*, 46(3):3173–3188, 2022. doi: 10.1002/er.7374.

[4] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pages 1273–1282, 2017.

[5] Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, et al. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1–2):1–210, 2021. doi: 10.1561/2200000083.

[6] Mohammad Navid Fekri, Katarina Grolinger, and Syed Mir. Distributed load forecasting using smart meter data: Federated learning with recurrent neural networks. *International Journal of Electrical Power & Energy Systems*, 137:107669, 2022. doi: 10.1016/j.ijepes.2021.107669.

[7] Christopher Briggs, Zhong Fan, and Peter Andras. Federated learning for short-term residential load forecasting. *IEEE Open Access Journal of Power and Energy*, 9:573–583, 2022. doi: 10.1109/OAJPE.2022.3206220.

[8] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems*, volume 2, pages 429–450, 2020.

[9] Michael Zhang, Karan Sapra, Sanja Fidler, Serena Yeung, and Jose M. Alvarez. Personalized federated learning with first order model optimization. In *International Conference on Learning Representations*, 2021.

[10] Yutao Huang, Li Chu, Zirui Zhou, Lanjun Wang, Jiangchuan Liu, Jian Pei, and Yong Zhang. Personalized cross-silo federated learning on non-iid data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 7865–7873, 2021.

[11] Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *Proceedings of the 38th International Conference on Machine Learning*, pages 6357–6368, 2021.

[12] Yaohui Huang, Yuan Zhao, Zhijin Wang, Xiufeng Liu, and Yonggang Fu. Sparse dynamic graph learning for district heat load forecasting. *Applied Energy*, 371:123685, 2024. doi: 10.1016/j.apenergy.2024.123685.

[13] Hakjae Kim, Sarangerel Dorjgochoo, Hansaem Park, and Sungju Lee. Personalized federated transfer learning for building energy forecasting via model ensemble with multi-level masking in heterogeneous sensing environment. *Electronics*, 14(9):1790, 2025. doi: 10.3390/electronics14091790.

[14] Xiangru Lian, Ce Zhang, Huan Zhang, Cho-Jui Hsieh, Wei Zhang, and Ji Liu. Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[15] Rong Dai, Li Shen, Fengxiang He, Xinmei Tian, and Dacheng Tao. Dispfl: Towards communication-efficient personalized federated learning via decentralized sparse training. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

[16] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[17] Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023.

[18] Michaela Killian and Martin Kozek. Ten questions concerning model predictive control for energy efficient buildings. *Building and Environment*, 105:403–412, 2016. doi: 10.1016/j.buildenv.2016.05.034.

[19] José R. Vázquez-Canteli and Zoltan Nagy. Reinforcement learning for demand response: A review of algorithms and modeling techniques. *Applied Energy*, 235:1072–1089, 2019. doi: 10.1016/j.apenergy.2018.11.002.

[20] Yi Wang, Dawei Qiu, Mingyang Sun, Goran Strbac, and Zhiwei Gao. Secure energy management of multi-energy microgrid: A physical-informed safe reinforcement learning approach. *Applied Energy*, 335:120759, 2023. doi: 10.1016/j.apenergy.2023.120759.

[21] José R. Vázquez-Canteli, Jérôme Kämpf, Gregor Henze, and Zoltan Nagy. Citylearn v1.0: An openai gym environment for demand response with deep reinforcement learning. In *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, pages 356–357, 2019. doi: 10.1145/3360322.3360998.

[22] Gauraang Dhamankar, José R. Vázquez-Canteli, and Zoltan Nagy. Benchmarking multi-agent deep reinforcement learning algorithms on a building energy demand coordination task. In *Proceedings of the 1st International Workshop on Reinforcement Learning for Energy Management in Buildings and Cities*, pages 15–19, 2020. doi: 10.1145/3427773.3427870.

[23] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[24] Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium*, pages 263–275, 2017. doi: 10.1109/CSF.2017.11.

[25] Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long- and short-term temporal patterns with deep neural networks. In *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 95–104, 2018. doi: 10.1145/3209978.3210006.