Supplementary Information for "Introducing Edge Intelligence to Smart Meters via Federated Split Learning"

Yehui Li[1†], Dalin Qin[1†], H.Vincent Poor[2*], Yi Wang[1*]

[1]Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong SAR, China.
[2]Department of Electrical and Computer Engineering, Princeton University, Princeton, USA.

*Corresponding author(s). E-mail(s): poor@princeton.edu; yiwang@eee.hku.hk;
Contributing authors: yhli@eee.hku.hk; dlqin@eee.hku.hk;
†These authors contributed equally to this work.

Supplementary Notes

Evaluation metrics

Expressions of RMSE, MAPE, sMAPE, and MAE are as follows:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{1}$$

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\% \tag{2}$$

$$\text{sMAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{y_i + \hat{y}_i} \times 100\% \tag{3}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{4}$$

where $\hat{y}_i$ and y_i are the i-th forecasted and real load values, respectively, and n is the number of samples.

Method description

The implementation details of the full method are as follows and also given in Algorithm 1 and Algorithm 2.

1. **Model splitting**: the cloud server first finds the optimal split ratio to split the initial model $\mathbf{w}^0$ into three parts $\mathbf{w}_e^0$, $\mathbf{w}_p^0$, and $\mathbf{w}_r^0$ and then distributes them to edge servers and smart meters, respectively. After that, each smart meter generates an auxiliary regressor $\mathbf{w}_a^0$.
2. **Model training**: In the round t, smart meters perform forward propagation for each sample on the feature extractor $\mathbf{w}_e^t$ and uploads the activation h_e to edge servers. Then h_e is sequentially forwarded as input through the feature processor $\mathbf{w}_p^t$ and the regressor $\mathbf{w}_r^t$, and the loss ℓ_s of the main model is calculated. Simultaneously, the auxiliary regressor $\mathbf{w}_a^t$ also propagates forward with h_e to obtain the prediction y_c, which is combined with the main model's prediction y_s to calculate the auxiliary model's distillation loss ℓ_c. Afterwards, smart meters calculate the gradient for $\mathbf{w}_r^t$ based on the error $\ell_{s,k}$. After receiving the gradient from smart meters, edge servers next calculate the gradient for $\mathbf{w}_p^t$. In parallel with the above process, smart meters calculate the gradients for $\mathbf{w}_a^t$ and $\mathbf{w}_e^t$ locally based on the error $\ell_{c,k}$. Eventually, the model parameters are updated based on the calculated gradients to complete a round of training.
3. **Model aggregation**: In the initialization phase, the cloud server designates smart meters to the edge servers to collaborate with by clustering their feature vectors V_k. After each smart meter finishes the model training in the t-th round, edge servers aggregate their parameters $\mathbf{w}_k^{t+1}$ to form the complete model $\mathbf{w}_{(i)}^{t+1}$. Finally, edge

servers upload their aggregated model to update the global model $\mathbf{w}^{t+1}$. Cloud server distributes $\mathbf{w}^{t+1}$ to turn on the next round of model training.

Algorithm 1 Collaborative Split Learning

Function Model Splitting$(\mathbf{w}^0)$**:**

Choose optimal ratio α^* to split model: $\mathbf{w}^0 \rightarrow \mathbf{w}_e^0, \mathbf{w}_p^0, \mathbf{w}_r^0$

Cloud server allocates models to edge servers and smart meters

Smart meters generate auxiliary regressor $\mathbf{w}_a^0$

Function Model Training$(\mathbf{w}^t)$**:**

for *each* $x \in D$ **do**

`// Forward Propagation:`

$h_e \leftarrow f(\mathbf{w}_e^t, x)$

Smart meters send h_e to edge servers

$h_p \leftarrow f(\mathbf{w}_p^t, h_e)$

Edge servers return h_p to smart meters

$y_s \leftarrow f(\mathbf{w}_r^t, h_p)$ and $y_c \leftarrow f(\mathbf{w}_a^t, h_e)$

$\ell_s \leftarrow \ell(y, y_s)$ and $\ell_c \leftarrow \gamma\ell(y, y_c) + (1-\gamma)\ell(y_s, y_c)$

`// Backward Propagation:`

Smart meters calculate $\nabla_r \ell_s(\mathbf{w}_s^t, x)$ and send it to edge servers

do in parallel

Edge servers calculate $\nabla_p \ell_s(\mathbf{w}_s^t, x)$

do in parallel

Smart meters calculates $\nabla_a \ell_c(\mathbf{w}_c^t, x)$ and $\nabla_e \ell_c(\mathbf{w}_c^t, x)$

`// Parameter Update:`

$\mathbf{w}_r^{t+1} \leftarrow \mathbf{w}_r^t - \eta_t \nabla_r \mathcal{L}_s(\mathbf{w}_s^t)$

$\mathbf{w}_p^{t+1} \leftarrow \mathbf{w}_p^t - \eta_t \nabla_p \mathcal{L}_s(\mathbf{w}_s^t)$

$\mathbf{w}_a^{t+1} \leftarrow \mathbf{w}_a^t - \eta_t \nabla_a \mathcal{L}_c(\mathbf{w}_c^t)$

$\mathbf{w}_e^{t+1} \leftarrow \mathbf{w}_e^t - \eta_t \nabla_e \mathcal{L}_c(\mathbf{w}_c^t)$

$\mathbf{w}^{t+1} \leftarrow [\mathbf{w}_e^{t+1}, \mathbf{w}_p^{t+1}, \mathbf{w}_r^{t+1}, \mathbf{w}_a^{t+1}]$

Return $\mathbf{w}^{t+1}$

Algorithm 2 Semi-Asynchronous Federated Learning

`// Initialization:`

for *each* $k \in A$ **do in parallel**

Smart meters upload feature vector $V_k \leftarrow [\frac{1}{P_{ED}}, \frac{1}{R}]$ to cloud server

Cloud server designates smart meters to the i-th edge server: $A_i \leftarrow Cluster(V_k)$

$\mathbf{w}_k^0 \leftarrow$ Model Splitting $(\mathbf{w}^0)$

Function Model aggregation`():`

for *each round* $t = 1, 2, ..., T$ **do**

for *each* $i = 1, 2, ..., M$ **do in parallel**

for *each* $k = 1, 2, ..., K_i$ **do in parallel**

Downloads global model $\mathbf{w}^t$ from cloud server

$\mathbf{w}_k^t \leftarrow \mathbf{w}^t$

$\mathbf{w}_k^{t+1} \leftarrow$ Model Training $(\mathbf{w}_k^t)$

`// Synchronous aggregation at edge servers:`

Smart meters upload model $\mathbf{w}_k^{t+1}$ to edge servers

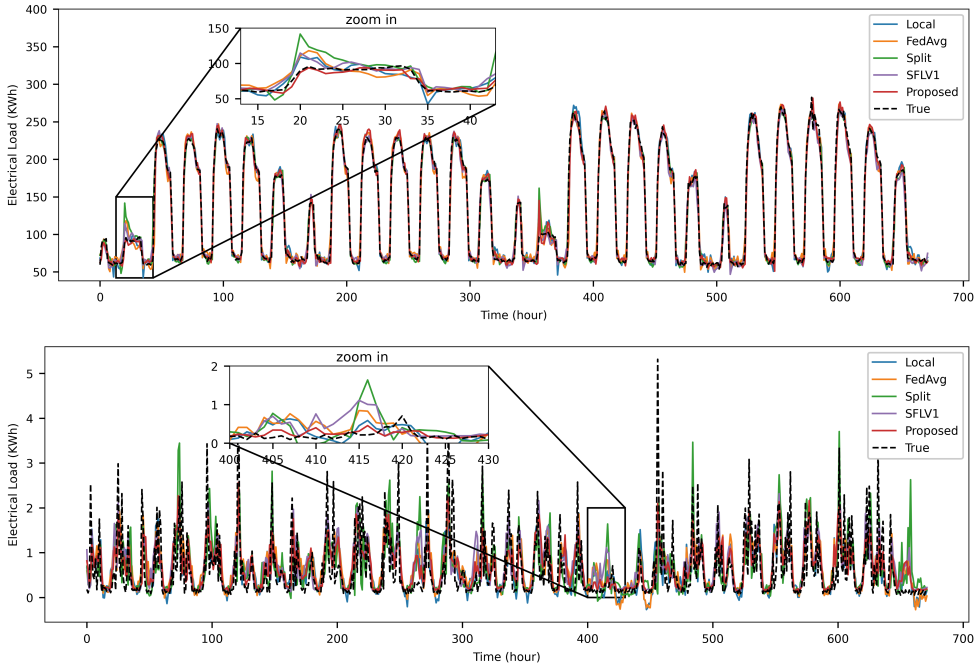
$\mathbf{w}_{(i)}^{t+1} \leftarrow \frac{1}{K_i} \sum_{k=1}^{K_i} \mathbf{w}_k^{t+1}$

`// Asynchronous aggregation at cloud server:`

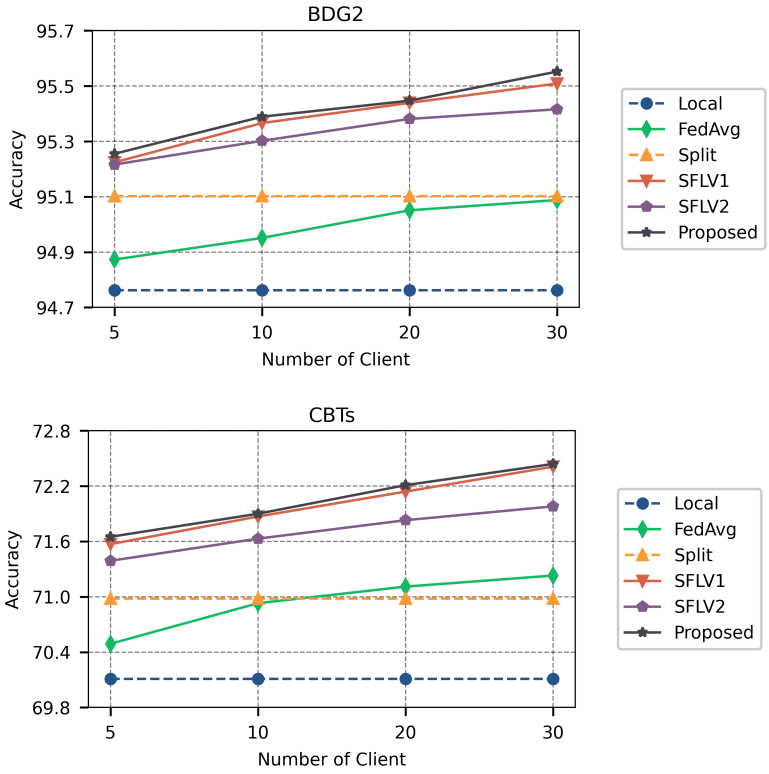
Edge servers upload model $\mathbf{w}_{(i)}^{t+1}$ to cloud server

$\mathbf{w}^{t+1} = (1 - \tau_i)\mathbf{w}^t + \tau_i \mathbf{w}_{(i)}^{t+1}$

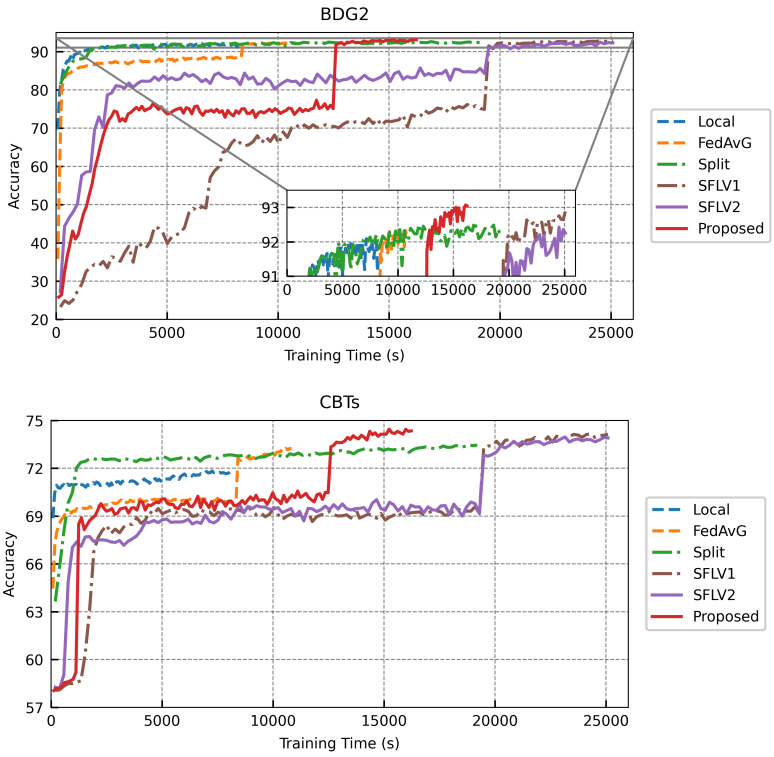
Supplementary Figures



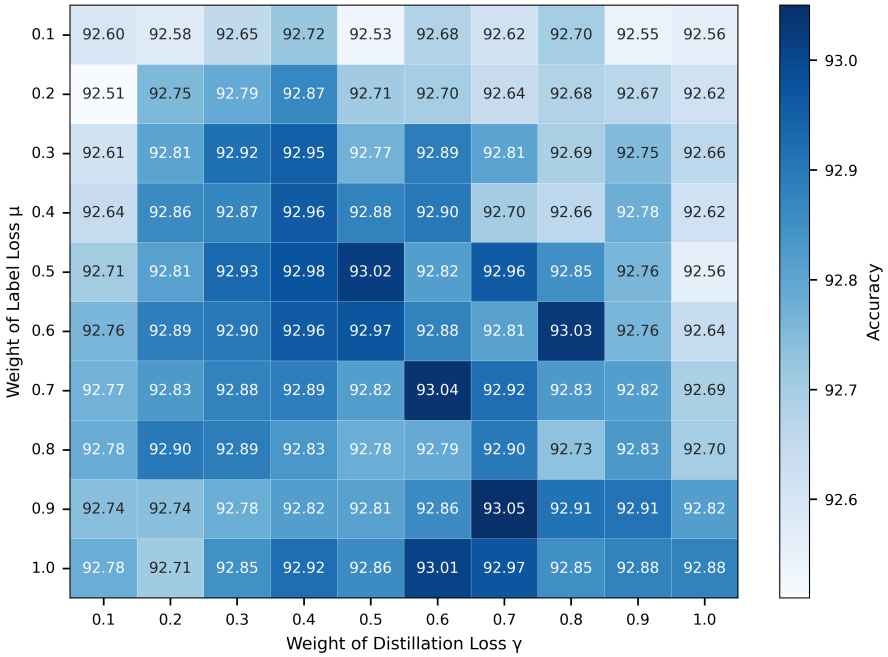
Supplementary Fig. 1: Comparison of forecasting results for six on-device feasible methods. We report the time series values of real and forecasting electrical load on datasets BDG2 and CBTs. The forecasting values of the proposed method are closest to the true values, especially in the case of high load volatility.



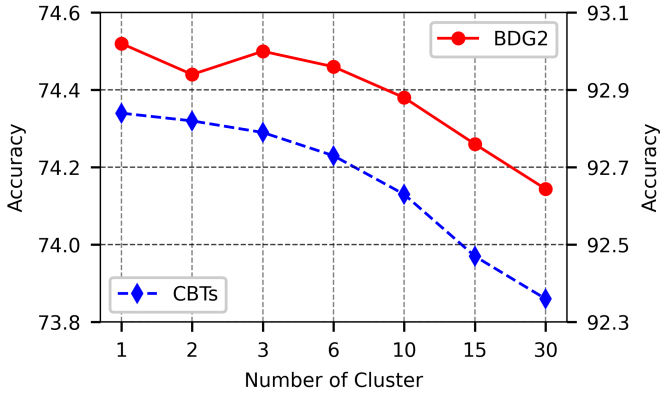
Supplementary Fig. 2: Performance evaluation of six on-device feasible methods on different client numbers. We compare the model accuracy of our method with benchmark methods for the first 5 clients. In federated learning-based approaches, more client participation can improve the model performance. We can observe that the proposed method always achieves the highest accuracy for different client numbers.



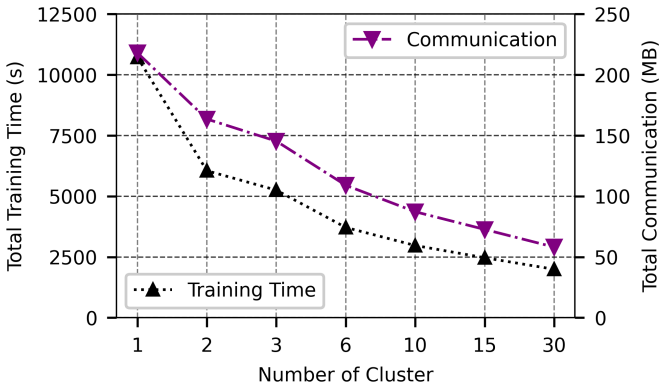
Supplementary Fig. 3: Comparison of forecasting accuracy versus training time for six on-device feasible methods. The federated learning-based method performs 30 rounds of local fine-tuning. We can see that the proposed method achieves the best performance while significantly reducing the training time compared to vanilla federated split learning methods.



Supplementary Fig. 4: Comparison of forecasting accuracy for choosing different loss weights on dataset BDG2. Columns: weight of distillation loss. Rows: weight of label loss. We find that models trained with too small label loss weights perform poorly. The model generally performs better when the weights are closer together.



(a)



(b)

Supplementary Fig. 5: Impact of the cluster number on the model performance. (a) Comparison of forecasting accuracy versus cluster number on datasets BDG2 and CBTs. Overall, the precision decreases with the number of clusters. (b) Comparison of training time and communication versus cluster number. Similarly, the training time and communication overhead decrease at a faster rate as the number of clusters increases. It shows that our method can dramatically enhance training efficiency without sacrificing precision.

Supplementary Tables

Supplementary Table 1: Number of network parameters for various layers

Layer	$\lvert\mathbf{w}\rvert$	$\lvert\mathbf{b}\rvert$
Fully connected layer	$M{\cdot}N$	N
Convolutional layer	$F{\cdot}C{\cdot}K$	F
Recurrent layer	$(M{+}N){\cdot}N$	N
LSTM layer	$4{\cdot}(M{+}N){\cdot}N$	$4{\cdot}N$

M: number of neurons in the previous layer.
N: number of neurons in the current layer.
F: number of filters in the previous layer.
N: number of filters in the current layer.
K: kernel size.

Supplementary Table 2: Number of intermediate parameters for various layers

Layer	$\lvert\mathbf{a}\rvert$	$\lvert\frac{\partial l}{\partial \mathbf{w}}\rvert$	$\lvert\frac{\partial l}{\partial \mathbf{b}}\rvert$	$\lvert\frac{\partial l}{\partial \mathbf{a}}\rvert$
Fully connected layer	N	$M{\cdot}N$	N	N
Convolutional layer	$F{\cdot}(M\text{-}K{+}1)$	$F{\cdot}C{\cdot}K$	F	$C{\cdot}N$
Recurrent layer	$S{\cdot}N$	$(M{+}N){\cdot}N$	N	$S{\cdot}N$
LSTM layer	$S{\cdot}N$	$4{\cdot}(M{+}N){\cdot}N$	$4{\cdot}N$	$S{\cdot}N$
Activation layer	M	-	-	M
Pooling layer	F	-	-	$F{\cdot}M$
Flatten layer	$F{\cdot}M$	-	-	$F{\cdot}M$

S: length of input series.

Supplementary Proofs

Supplementary Proof 1: Efficiency-optimal split ratio

Considering the first case, we have $\alpha \geq (\frac{P_{es}}{KP_{sm}} + 1)^{-1}$ which is equivalent to

$$\frac{\alpha\left(1-\beta\right)n\left|D\right|\left|\mathbf{w}\right|}{P_{sm}} \geq \frac{\left(1-\alpha\right)\left(1-\beta\right)n\left|D\right|\left|\mathbf{w}\right|K}{P_{es}} \tag{5}$$

Hence, the training time can be rewritten as

$$T = \frac{3s\left|D\right| + 2\alpha\left|\mathbf{w}\right|}{R} + \frac{\alpha n\left|D\right|\left|\mathbf{w}\right|}{P_{sm}} + \frac{(1-\alpha)n\beta\left|D\right|\left|\mathbf{w}\right|K}{P_{es}} \tag{6}$$

Here when $P_{es} \geq \beta K(\frac{2}{nR|D|} + \frac{1}{P_{sm}})^{-1}$, (6) is an increasing function of α and the optimal ratio minimizing the training time is $\alpha^* = (\frac{P_{es}}{KP_{sm}} + 1)^{-1}$ provided that α^* is not larger than the upper bound α_{upper}. Otherwise (6) is a decreasing function, i.e., the optimal ratio is the upper bound α_{upper}.

Then, considering the second case, we have $\alpha \leq (\frac{P_{es}}{KP_{sm}} + 1)^{-1}$, which is equivalent to

$$\frac{\alpha\left(1-\beta\right)n\left|D\right|\left|\mathbf{w}\right|}{P_{es}} \leq \frac{\left(1-\alpha\right)\left(1-\beta\right)n\left|D\right|\left|\mathbf{w}\right|K}{P_{sm}} \tag{7}$$

Hence, the training time can be rewritten as

$$T = \frac{3s\left|D\right| + 2\alpha\left|\mathbf{w}\right|}{R} + \frac{\alpha\beta n\left|D\right|\left|\mathbf{w}\right|}{P_{sm}} + \frac{\left(1-\alpha\right)n\left|D\right|\left|\mathbf{w}\right|K}{P_{es}} \tag{8}$$

Here when $P_{es} \leq K(\frac{1}{nR|D|} + \frac{\beta}{P_{sm}})^{-1}$, (8) is a decreasing function of α and the optimal ratio minimizing the training time is $\alpha^* = (\frac{P_{es}}{KP_{sm}} + 1)^{-1}$ provided that α^* is not smaller than the lower bound α_{lower}. Otherwise (8) is an increasing function, i.e., the optimal ratio is the lower bound α_{lower}. Note that β is between 0 and 1, so the upper bound is clearly larger than the lower bound. Hence, we complete the proof of the optimal ratio.

Supplementary Proof 2: Convergence of auxiliary model

Under Assumption 1, we can write

$$\mathcal{L}_c(\mathbf{w}_c^{t+1}) - \mathcal{L}_c(\mathbf{w}_c^t) \leq \nabla \mathcal{L}_c(\mathbf{w}_c^t)^T(\mathbf{w}_c^{t+1} - \mathbf{w}_c^t) + \frac{L}{2}\left\|\mathbf{w}_c^{t+1} - \mathbf{w}_c^t\right\|^2. \tag{9}$$

Note that we have

$$\mathbf{w}_c^{t+1} = \mathbf{w}_c^t - \tau_i \eta_t \frac{1}{K_i} \sum_{k \in A_i} \nabla \mathcal{L}_{c,k}(\mathbf{w}_{c,k}^t) \tag{10}$$

Substituting $\mathbf{w}_c^{t+1} - \mathbf{w}_c^t$, we have

$$\begin{aligned} \mathcal{L}_c(\mathbf{w}_c^{t+1}) - \mathcal{L}_c(\mathbf{w}_c^t) \leq & -\tau_i \eta_t \nabla \mathcal{L}_c(\mathbf{w}_c^t)^T \left(\frac{1}{K_i} \sum_{k \in A_i} \nabla \mathcal{L}_{c,k}(\mathbf{w}_{c,k}^t) \right) \\ & + \frac{L}{2} \tau_i^2 \eta_t^2 \left\| \frac{1}{K_i} \sum_{k \in A_i} \nabla \mathcal{L}_{c,k}(\mathbf{w}_{c,k}^t) \right\|^2. \end{aligned} \tag{11}$$

Taking expectation over (11) as follows

$$\begin{aligned} \mathbb{E}\left[\mathcal{L}_c(\mathbf{w}_c^{t+1})\right] - \mathbb{E}\left[\mathcal{L}_c(\mathbf{w}_c^t)\right] \leq & -\tau_i \eta_t \underbrace{\mathbb{E}\left[\nabla \mathcal{L}_c(\mathbf{w}_c^t)^T \left(\frac{1}{K_i} \sum_{k \in A_i} \nabla \mathcal{L}_{c,k}(\mathbf{w}_{c,k}^t) \right) \right]}_{A_1} \\ & + \frac{L}{2} \tau_i^2 \eta_t^2 \underbrace{\mathbb{E}\left[\left\| \frac{1}{K_i} \sum_{k \in A_i} \nabla \mathcal{L}_{c,k}(\mathbf{w}_{c,k}^t) \right\|^2 \right]}_{A_2}. \end{aligned} \tag{12}$$

In the following, we will bound A_1 and A_2, respectively. The equivalent form of A_1 is

$$\begin{aligned} A_1 &= \mathbb{E}\left[\nabla \mathcal{L}_c(\mathbf{w}_c^t)^T \left(\frac{1}{K_i} \sum_{k \in A_i} \nabla \mathcal{L}_{c,k}(\mathbf{w}_c^t) \right) \right] \\ &= \frac{1}{2}\mathbb{E}\left[\left\| \nabla \mathcal{L}_c(\mathbf{w}_c^t) \right\|^2 \right] + \frac{1}{2}\mathbb{E}\left[\left\| \frac{1}{K_i} \sum_{k \in A_i} \nabla \mathcal{L}_{c,k}(\mathbf{w}_{c,k}^t) \right\|^2 \right] \\ &\quad - \frac{1}{2} \underbrace{\mathbb{E}\left[\left\| \nabla \mathcal{L}_c(\mathbf{w}_c^t) - \frac{1}{K_i} \sum_{k \in A_i} \nabla \mathcal{L}_{c,k}(\mathbf{w}_{c,k}^t) \right\|^2 \right]}_{B}. \end{aligned} \tag{13}$$

Since the sampled gradient of each cluster is an unbiased estimator of the full gradient, i.e., $\nabla \mathcal{L}_c(\mathbf{w}_c^t) = \mathbb{E}\left[\frac{1}{K_i}\sum_{k\in A_i} \nabla \mathcal{L}_{c,k}(\mathbf{w}_{c,k}^t)\right]$, so we have

$$\begin{aligned} B &= \mathbb{E}\left[\left\|\nabla \mathcal{L}_c(\mathbf{w}_c^t) - \frac{1}{K_i}\sum_{k\in A_i} \nabla \mathcal{L}_{c,k}(\mathbf{w}_{c,k}^t)\right\|^2\right] \\ &= \mathbb{E}\left[\left\|\mathbb{E}\left[\frac{1}{K_i}\sum_{k\in A_i} \nabla \mathcal{L}_{c,k}(\mathbf{w}_{c,k}^t)\right] - \frac{1}{K_i}\sum_{k\in A_i} \nabla \mathcal{L}_{c,k}(\mathbf{w}_{c,k}^t)\right\|^2\right] \\ &\leq \mathbb{E}\left[\left\|\frac{1}{K_i}\sum_{k\in A_i} \nabla \mathcal{L}_{c,k}(\mathbf{w}_{c,k}^t)\right\|^2\right] - \mathbb{E}\left[\left\|\nabla \mathcal{L}_c(\mathbf{w}_c^t)\right\|^2\right] \end{aligned} \tag{14}$$

Substituting (14) into (13), we have

$$A_1 = \mathbb{E}\left[\left\|\nabla \mathcal{L}_c\left(\mathbf{w}_c^t\right)\right\|^2\right]. \tag{15}$$

Then we apply Cauchy-Schwarz inequality and Assumption 2 to find the lower bound of A_2:

$$\begin{aligned} A_2 &= \mathbb{E}\left[\left\|\frac{1}{K}\sum_{k\in A_t} \nabla \mathcal{L}_{c,k}\left(\mathbf{w}_{c,k}^t\right)\right\|^2\right] \\ &\leq \frac{1}{K_i}\sum_{k\in A_i} \mathbb{E}\left[\left\|\nabla \mathcal{L}_{c,k}\left(\mathbf{w}_{c,k}^t\right)\right\|^2\right] \\ &\leq \frac{1}{K_i}\sum_{k\in A_i} \frac{1}{|D_k|}\sum_{x\in D_k} \mathbb{E}\left[\left\|\nabla \ell_{c,k}(\mathbf{w}_{c,k}, x_k)\right\|^2\right] \\ &\leq G_1. \end{aligned} \tag{16}$$

Substituting (15) and (16) into (12), it follows that

$$\mathbb{E}\left[\mathcal{L}_c(\mathbf{w}_c^{t+1})\right] \leq \mathbb{E}\left[\mathcal{L}_c(\mathbf{w}_c^t)\right] - \tau_i\eta_t\mathbb{E}\left[\left\|\nabla \mathcal{L}_c(\mathbf{w}_c^t)\right\|^2\right] + G_1\frac{L}{2}\tau_i^2\eta_t^2. \tag{17}$$

Now by summing up for all global rounds $t = 1, ...T$, we have

$$\mathbb{E}\left[\mathcal{L}_c(\mathbf{w}_c^T)\right] \leq \mathbb{E}\left[\mathcal{L}_c(\mathbf{w}_c^0)\right] - \tau_i\sum_{t=1}^{T}\eta_t\mathbb{E}\left[\left\|\nabla \mathcal{L}_c(\mathbf{w}_c^t)\right\|^2\right] + G_1\frac{L}{2}\tau_i^2\sum_{t=1}^{T}\eta_t^2. \tag{18}$$

Finally due to $\mathcal{L}_c(\mathbf{w}_c^*) \leq \mathbb{E}\left[\mathcal{L}_c(\mathbf{w}_c^T)\right]$, we have

$$\sum_{t=1}^{T}\eta_t\mathbb{E}\left[\left\|\nabla \mathcal{L}_c(\mathbf{w}_c^t)\right\|^2\right] \leq \frac{1}{\tau_i}\left[\mathcal{L}_c(\mathbf{w}_c^0) - \mathcal{L}_c(\mathbf{w}_c^*)\right] + G_1\frac{L}{2}\tau_i\sum_{t=1}^{T}\eta_t^2, \tag{19}$$

where τ_i can be regarded as a constant due to the equal number of smart meters in each cluster. Hence, we complete the convergence proof for the auxiliary model.

Supplementary Proof 3: Convergence of Main Model

Under Assumption 1, We obtain the same form as the auxiliary model

$$\begin{aligned}\mathbb{E}\left[\mathcal{L}_s(\mathbf{w}_s^{t+1})\right] - \mathbb{E}\left[\mathcal{L}_s(\mathbf{w}_s^{t})\right] \leq & \tau_i \eta_t \underbrace{\mathbb{E}\left[-\nabla \mathcal{L}_s(\mathbf{w}_s^t)^T \left(\frac{1}{K_i}\sum_{k\in A_i} \nabla \mathcal{L}_{s,k}(\mathbf{w}_{s,k}^t)\right)\right]}_{C_1} \\ & + \frac{L}{2}\tau_i^2 \eta_t^2 \underbrace{\mathbb{E}\left[\left\|\frac{1}{K_i}\sum_{k\in A_i} \nabla \mathcal{L}_{s,k}(\mathbf{w}_{s,k}^t)\right\|^2\right]}_{C_2}.\end{aligned} \tag{20}$$

In the following, we will bound C_1 and C_2, respectively. Applying the Cauchy-Schwarz inequality, we have

$$\begin{aligned} C_1 &\leq \mathbb{E}\left[\nabla \left\|\mathcal{L}_s(\mathbf{w}_s^t)\right\|^2 - \nabla \mathcal{L}_s(\mathbf{w}_s^t)^T \left(\frac{1}{K_i}\sum_{k\in A_i} \nabla \mathcal{L}_{s,k}(\mathbf{w}_{s,k}^t)\right)\right] \\ &= \mathbb{E}\left[\nabla \mathcal{L}_s(\mathbf{w}_s^t)^T \left(\nabla \mathcal{L}_s(\mathbf{w}_s^t) - \frac{1}{K_i}\sum_{k\in A_i} \nabla \mathcal{L}_{s,k}(\mathbf{w}_{s,k}^t)\right)\right] \\ &\leq \underbrace{\mathbb{E}\Big[\left\|\nabla \mathcal{L}_s(\mathbf{w}_s^t)\right\|\Big]}_{D_1} \cdot \underbrace{\mathbb{E}\left[\left\|\nabla \mathcal{L}_s(\mathbf{w}_s^t) - \frac{1}{K_i}\sum_{k\in A_i} \nabla \mathcal{L}_{s,k}(\mathbf{w}_{s,k}^t)\right\|\right]}_{D_2} \end{aligned} \tag{21}$$

We first find a lower bound of D_1 as

$$\begin{aligned} D_1 &= \mathbb{E}\Big[\left\|\nabla \mathcal{L}_s(\mathbf{w}_s^t)\right\|\Big] \\ &\leq \sqrt{G_2} \end{aligned} \tag{22}$$

We apply Fubini's theorem to find a lower bound of D_2 as

$$\begin{aligned} D_2 &= \mathbb{E}\left[\left\|\nabla \mathcal{L}_s(\mathbf{w}_s^t) - \frac{1}{K_i}\sum_{k\in A_i} \nabla \mathcal{L}_{s,k}(\mathbf{w}_s^t)\right\|\right] \\ &\leq \frac{1}{K_i}\sum_{k\in A_i} \mathbb{E}\left[\left\|\frac{1}{|D_k|}\sum_{x\in D_k} \nabla \ell_{s,k}(\mathbf{w}_{s,k}^t, h_{e,k}^*(x)) - \nabla \ell_{s,k}\left(\mathbf{w}_{s,k}^t, h_{e,k}^t(x)\right)\right\|\right] \\ &\leq \frac{1}{K_i}\sum_{k\in A_i} \mathbb{E}\left[\int \left\|\nabla \ell_{s,k}(\mathbf{w}_{s,k}^t, h)\right\| \left\|p_{c,k}^t(h) - p_{c,k}^*(h)\right\| dh\right] \\ &\leq \sqrt{G_2}\frac{1}{K_i}\sum_{k\in A_i} \sqrt{d_{c,k}^t}. \end{aligned} \tag{23}$$

By combining (21), (22) and (23), we have

$$C_1 \leq G_2 \tau_i \eta_t \frac{1}{K_i} \sum_{k \in A_i} \sqrt{d_{c,k}^t} - \tau_i \eta_t \mathbb{E}\left[\nabla \left\| \mathcal{L}_s(\mathbf{w}_s^t) \right\|^2\right]. \tag{24}$$

With the same derivation process for the auxiliary model, we find a lower bound of C_2 as

$$\begin{aligned} C_2 &= \mathbb{E}\left[\left\| \frac{1}{K_i} \sum_{k \in A_i} \nabla \mathcal{L}_{s,k}(\mathbf{w}_{s,k}^t) \right\|^2\right] \\ &\leq G_2. \end{aligned} \tag{25}$$

Substituting (24) and (25) into (20), it follows that

$$\mathbb{E}\left[\mathcal{L}_s(\mathbf{w}_s^{t+1})\right] \leq \mathbb{E}\left[\mathcal{L}_s(\mathbf{w}_s^t)\right] - \tau_i \eta_t \mathbb{E}\left\| \nabla \mathcal{L}_s(\mathbf{w}_s^t) \right\|^2 + G_2 \left(\tau_i \eta_t \frac{1}{K_i} \sum_{k \in A_i} \sqrt{d_{c,k}^t} + \frac{L}{2} \tau_i^2 \eta_t^2 \right). \tag{26}$$

Now by summing up for all global rounds t = 1, ...T, we have

$$\begin{aligned} \mathbb{E}\left[\mathcal{L}_s(\mathbf{w}_s^T)\right] \leq \mathbb{E}\left[\mathcal{L}_s(\mathbf{w}_s^0)\right] &- \tau_i \sum_{t=1}^{T} \eta_t \mathbb{E}\left[\left\| \nabla \mathcal{L}_s(\mathbf{w}_s^t) \right\|^2\right] \\ &+ G_2 \left(\tau_i \sum_{t=1}^{T} \eta_t \frac{1}{K_i} \sum_{k \in A_i} \sqrt{d_{c,k}^t} + \frac{L}{2} \tau_i^2 \sum_{t=1}^{T} \eta_t^2 \right). \end{aligned} \tag{27}$$

Finally due to $\mathcal{L}_s(\mathbf{w}_s^*) \leq \mathbb{E}\left[\mathcal{L}_s(\mathbf{w}_s^T)\right]$, we have

$$\sum_{t=1}^{T} \eta_t \mathbb{E}\left[\left\| \nabla \mathcal{L}_s(\mathbf{w}_s^t) \right\|^2\right] \leq \frac{1}{\tau_i} \left[\mathcal{L}_s(\mathbf{w}_s^0) - \mathcal{L}_s(\mathbf{w}_s^*)\right] + G_2 \left(\sum_{t=1}^{T} \eta_t \frac{1}{K_i} \sum_{k \in A_i} \sqrt{d_{c,k}^t} + \frac{L}{2} \tau_i \sum_{t=1}^{T} \eta_t^2 \right), \tag{28}$$

Since Supplementary Proof. 2 has proved that the auxiliary model is convergent, we have $\sum_{t=1}^{T} d_{c,k}^t < \infty$. Hence, we complete the convergence proof for the main model.