MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

ConvEDNet: A Convolutional Energy Disaggregation Network Using Continuous Point-On-Wave Measurements

Shirsat, Ashwin; Sun, Hongbo; Kim, Kyeong Jin; Guo, Jianlin; Nikovski, Daniel N.

TR2022-101 September 17, 2022

Abstract

Energy disaggregation means separating the dis- tribution system-level net-load measurements into native load and photovoltaic (PV) generation. This paper proposes a causal, context-aware, and fully-convolutional deep-learning network for simultaneous PV-load energy disaggregation using continuous point-on-wave measurements. The proposed network, called Conv-EDNet, uses an encoder-decoder framework combined with a separator network for performing disaggregation in the time domain. The separator network harnesses the power of stacked dilated temporal convolutions for learning two weighting functions that perform the disaggregation using the non-negative encoder output. Finally, the decoder converts the weighted encoder output into time-domain native-load and PV genera- tion measurements. Further, Conv-EDNet+, which combines the Conv-EDNet with a gated recurrent unit-based discriminator for adversarial learning, is proposed. Numerical evaluations of the proposed approaches outperform the present state-of-the-art methods widely used for disaggregation tasks.

IEEE PES General Meeting 2022

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

ConvEDNet: A Convolutional Energy Disaggregation Network Using Continuous Point-On-Wave Measurements

Ashwin Shirsat[1], Hongbo Sun[2], Kyeong Jin Kim[2], Jianlin Guo[2], and Daniel Nikovski[2]
[1]Department of Electrical and Computer Engineering
North Carolina State University, Raleigh, NC 27695, USA
ashirsa@ncsu.edu
[2]Mitsubishi Electric Research Laboratories, Cambridge, MA 02139, USA
{hongbosun, kkim, guo, nikovski}@merl.com

***Abstract*—Energy disaggregation means separating the distribution system-level net-load measurements into native load and photovoltaic (PV) generation. This paper proposes a causal, context-aware, and fully-convolutional deep-learning network for simultaneous PV-load energy disaggregation using continuous point-on-wave measurements. The proposed network, called Conv-EDNet, uses an encoder-decoder framework combined with a separator network for performing disaggregation in the time domain. The separator network harnesses the power of stacked dilated temporal convolutions for learning two weighting functions that perform the disaggregation using the non-negative encoder output. Finally, the decoder converts the weighted encoder output into time-domain native-load and PV generation measurements. Further, Conv-EDNet+, which combines the Conv-EDNet with a gated recurrent unit-based discriminator for adversarial learning, is proposed. Numerical evaluations of the proposed approaches outperform the present state-of-the-art methods widely used for disaggregation tasks.**

***Index Terms*—Energy disaggregation, continuous-point-on-wave measurements, supervised learning, adversarial learning.**

I. Introduction

Distribution networks and their role in power delivery are undergoing a paradigm shift with the increased penetration of distributed energy resources (DER). To smoothly operate such a complex system with bidirectional power flow, knowing the instantaneous generation and demand is critical. Utilities measure the net system load and are unaware of the native system load and the photovoltaic (PV) generation. Thus, energy disaggregation (ED) plays a pivotal role in increasing grid observability to support network planning and operational tasks. The low-resolution measurements obtained from advanced metering infrastructure and the substation supervisory control and data acquisition systems do not provide the much-needed observability into the DER-rich network. Hence, the utilities are now installing continuous-point-on-wave (CPOW) measurement units to obtain measurements sampled at 1 kHz or higher [1]. [2] has studied the relationship between data sampling rates and the quality of ED results and have concluded that the disaggregation accuracy increases with an increase in the sampling rate, with the best accuracy obtained for data sampling rates ranging between 1 kHz and 12 kHz. Thus, in order to support ED using CPOW data, novel computationally efficient and accurate algorithms are required.

The work of A. Shirsat was done while he was working at MERL.

The methodologies proposed in the existing literature on net-load disaggregation can be classified into two broad categories: model-based and model-free (data-driven). The idea of model-based approaches is to develop a parametric model of PV generation and native demand and statistically estimate the parameters using the available net-load and meteorological data [3]. The disaggregation performance of these approaches is highly dependent on the quality of the parametric model. On the contrary, model-free approaches do not require the parametric model but are affected by the data quality. Some of the model-free techniques for ED exploit correlation between load and generation [4], use a dictionary learning approach [5], or use supervised machine learning techniques [6].

Of all the approaches listed above, the use of CPOW data has not been considered. A source of inspiration for disaggregation using CPOW data can be found in single-channel blind audio source separation. The audio source separation task is similar to ED using CPOW data since the audio signals with frequencies greater than 1 kHz are used. The authors in [7], [8] have proposed approaches for time-domain audio source separation using deep learning methods.

Summarizing the literature review, we observe that the use of CPOW data has not been considered within all the existing ED approaches. Further, the ED algorithms have not considered the operational status of the PV generators, which could result in disaggregation errors. The methods for audio source separation tasks are designed to handle high granular data. However, they cannot be applied directly to ED due to the difference in the physical properties of sound waves and electric signals. To overcome these limitations and fill the gap in the existing literature, we propose the following:

1) A deep learning-based model-free time domain net-load disaggregation framework conditioned on time-frequency domain features using CPOW measurements.
2) A new framework that supports scale and context-aware disaggregation, wherein the time domain, time frequency-domain, and spatio-temporal information are used to quantify the scale and context of disaggregation.
3) A scheme for simultaneous disaggregation of the native load and PV components that eliminates the need for the building and training of two separate networks.
4) The addition of adversarial training component to Conv-EDNet (Conv-EDNet+) for further enhancing the model prediction accuracy and robustness.

II. Convolutional Energy Disaggregation Network (Conv-EDNet)

This section outlines the details of the four sub-networks that build the Conv-EDNet framework, which is shown in Fig. 1. Let $\boldsymbol{p}^{\mathrm{NL}} \in \mathbb{R}^{1\times T}$ be the measured net-load time series obtained from the voltage and current measurements of the CPOW measurement unit. Similarly, let $\boldsymbol{p}^{\mathrm{L}} \in \mathbb{R}_{+}^{1\times T}$ and $\boldsymbol{p}^{\mathrm{PV}} \in \mathbb{R}_{+}^{1\times T}$ be the unobservable native load and PV generation measurements. The relationship between the three is as follows: $\boldsymbol{p}^{\mathrm{NL}} = \boldsymbol{p}^{\mathrm{L}} - \boldsymbol{p}^{\mathrm{PV}}$. Upper-case boldfaced letters denote matrices, while lower-case boldfaced letters denote vectors.

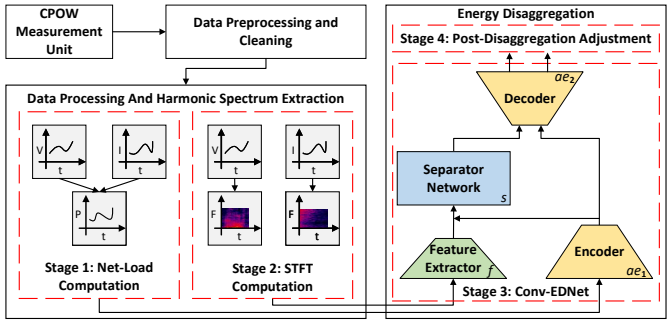

Fig. 1. Illustration of the proposed Conv-EDNet model.

A. Encoder network

The encoder network generates a compressed latent space representation of $\boldsymbol{p}^{\mathrm{NL}}$, which is then passed on to the separator network and the decoder network. First, the net-load time series $\boldsymbol{p}^{\mathrm{NL}}$ is segmented into I overlapping time frames of length $\hat{T}$, $\boldsymbol{p}_i^{\mathrm{NL}} \in \mathbb{R}^{1\times\hat{T}}$, where $i = 1,\ldots,I$, with an overlap of 50%. The encoder uses 1D convolution operation to transform $\boldsymbol{p}_i^{\mathrm{NL}}$ into a $N \times L$ dimensional representation, $\boldsymbol{X}_{ae_1,i} \in \mathbb{R}^{N\times L}$. The 1D convolution operation followed by a non-linear activation function using a kernel size of K, N output channels, and stride S can be described as follows:

$$\boldsymbol{X}_{ae_1,i} = \mathcal{H}(\boldsymbol{p}_i^{\mathrm{NL}} \odot \boldsymbol{\Theta}_{ae_1}), \tag{1}$$

where $\boldsymbol{\Theta}_{ae_1} \in \mathbb{R}^{N\times K}$ is the encoder weight matrix, $\odot$ represents the 1D convolution operation, and $\mathcal{H}(\cdot)$ represents the non-linear ReLU activation function.

B. Feature extractor network

The goal of the feature extractor network is to utilize the net-load data and the supplemental contextual data to assist with the disaggregation task. The CPOW unit measures the individual phase current and voltage, from which the aggregated net-load value is computed. This data also helps supplement the disaggregation task by extracting information on harmonic content, which helps quantify PV generation by assisting in detecting the generator connectivity status and generator faults [9]. Hence, a separate dedicated network is developed with the sole responsibility of extracting vital features from the instantaneous voltage and current measurement data along with the irradiance and temperature measurements.

The feature extraction process follows a two-step approach. First, the instantaneous time domain CPOW voltage and current measurements ($\boldsymbol{f}^{\mathrm{V}}, \boldsymbol{f}^{\mathrm{I}} \in \mathbb{R}^{1\times\hat{T}}$) are converted into time-frequency domain using short-term Fourier transform (STFT). Let the per-phase voltage and current STFT spectrograms be denoted by $\boldsymbol{F}^{\mathrm{STFT-V}} \in \mathbb{R}^{F\times\tilde{T}}$ and $\boldsymbol{F}^{\mathrm{STFT-I}} \in \mathbb{R}^{F\times\tilde{T}}$, where F denotes the spectrogram height and $\tilde{T}$ is the total window count (spectrogram width) for a signal of length $\hat{T}$. Second, the time domain contextual information such as temperature ($\boldsymbol{f}^{\mathrm{Temp}} \in \mathbb{R}^{1\times\hat{T}}$) and irradiance ($\boldsymbol{f}^{\mathrm{Irr.}} \in \mathbb{R}^{1\times\hat{T}}$) along with the previously obtained time-frequency domain representations are passed to a feature extraction network as shown in Fig. 2.

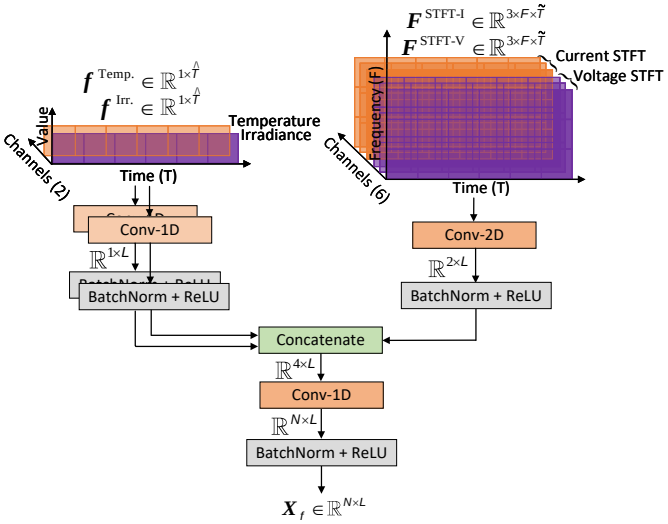

Fig. 2. Feature extractor network.

C. Separator network

The role of the separator network is to estimate the two weight matrices, $\boldsymbol{X}_{s,i}^{\mathrm{PV}}$, $\boldsymbol{X}_{s,i}^{\mathrm{L}} \in \mathbb{R}_{[0,1]}^{N\times L}$, to be multiplied element-wise with the encoder output $\boldsymbol{X}_{ae_1,i}$. Each separator matrix learns and assigns a unique weight to the encoder output. By assigning these weights, the key values in the encoder output, which are most significant for load or PV generation can be determined. To ensure that the weights are scale-invariant, the following constraint is added using a sigmoid activation function: $\boldsymbol{X}_{s,i}^{\mathrm{PV}} + \boldsymbol{X}_{s,i}^{\mathrm{L}} = \mathbf{1}$, where $\mathbf{1}$ denotes an $N \times L$ matrix of ones.

The separator network must learn the temporal features of the load and PV generation. Hence, to add a temporal learning mechanism, we propose using temporal convolutional network (TCN). This network consists of sequential layers of stacked dilated 1D TCN blocks, with each block consisting of a gated residual network. Within each sequential layer, the TCN blocks are stacked such that the dilation factor goes on exponentially increasing as the stack keeps getting taller. The exponential increase in dilation factor is designed to ensure a sufficiently large temporal receptive field size to facilitate the modeling of long-range temporal features observed within the load and PV generation time series. The stacked sequence is then repeated R times sequentially. The detailed representation of this design is shown in Fig. 3(a).

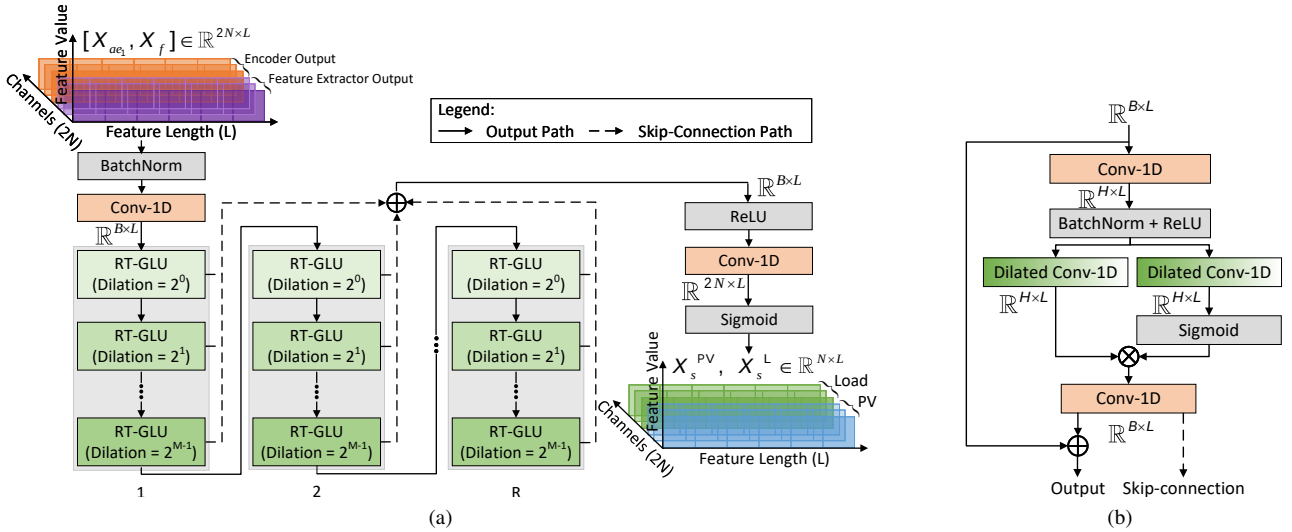

Fig. 3. Separator network: (a) Illustration of the separator network, (b) Illustration of the RT-GLU block.

Each 1D TCN block is carefully designed using gated linear units (GLUs) [10] and residual learning architecture [11]. Gating mechanisms are introduced to alleviate the vanishing gradient problems during training via backpropagation through time [12]. We modify the GLU by incorporating TCNs and residual learning into a new residual block shown in Fig. 3(b) [13]. The overall combination helps add temporal learning component along with addressing problems of vanishing gradients and degradation. Hereafter, this block is addressed as residual temporal GLU (RT-GLU).

D. Decoder

The decoder transforms the weighted latent representations corresponding to the load and PV generation back into their time domain representations. For doing so, 1D transposed convolutions are used, which can be represented as follows:

$$\boldsymbol{p}_i^{\mathrm{PV}} = (\boldsymbol{X}_{ae_1,i} \otimes \boldsymbol{X}_{s,i}^{\mathrm{PV}}) \circledast \boldsymbol{\Theta}_{ae_2}, \tag{2}$$

$$\boldsymbol{p}_i^{\mathrm{L}} = (\boldsymbol{X}_{ae_1,i} \otimes \boldsymbol{X}_{s,i}^{\mathrm{L}}) \circledast \boldsymbol{\Theta}_{ae_2}, \tag{3}$$

where $\boldsymbol{\Theta}_{ae_2} \in \mathbb{R}^{N \times K}$ is the decoder weight matrix, $\otimes$ represents element-wise multiplication, and $\circledast$ represents the 1D transposed convolution.

III. Adversarial Learning

To further enhance the disaggregation quality and robustness of Conv-EDNet, we extend the proposed model with an adversarial learning component. Following the generative adversarial networks (GAN) terminology, the basic Conv-EDNet model will be referred to as a generator ($\mathcal{G}$). A new component called the discriminator ($\mathcal{D}$) is added to facilitate adversarial learning. The adversarially modified Conv-EDNet, referred to as Conv-EDNet+ hereafter, is obtained by adding a discriminator network to the existing Conv-EDNet model.

Adversarial training means that the $\mathcal{G}$ is trained to trick the $\mathcal{D}$ into believing that the sequences predicted by $\mathcal{G}$ are the actual measurements. This relationship is achieved by solving a two-player min-max game as shown in (4). The $\mathcal{D}$ is trained to maximize the probability of correctly classifying the real samples (i.e., the measurements) and the generated samples (produced by $\mathcal{G}$). In contrast, $\mathcal{G}$ is trained to produce output samples that are hard to correctly distinguish by $\mathcal{D}$. Thus, we have the following min-max game for ED:

$$\min_{\mathcal{G}} \max_{\mathcal{D}} \big(E_{x \in \boldsymbol{p}^{\mathrm{NL}}}[\log \mathcal{D}(x)] + E_{x \in \boldsymbol{p}^{\mathrm{NL}}}[\log(1 - \mathcal{D}(\mathcal{G}(x)))] \big). \tag{4}$$

The input to $\mathcal{G}$ is the net load measurement, $\boldsymbol{p}_i^{\mathrm{NL}}$, and the input to $\mathcal{D}$ is shuffled between $\boldsymbol{p}_i^{\mathrm{NL}}$ and $\boldsymbol{p}_i^{\mathrm{L}} - \boldsymbol{p}_i^{\mathrm{PV}}$. To ensure $\mathcal{D}$ learns the temporal features of net-load measurements and does not get easily tricked by $\mathcal{G}$, we propose a deeper network for $\mathcal{D}$ with embedded regressive temporal characteristics by using gated recurrent units (GRUs). The convolutional layers perform the task of feature extraction, while the GRUs help model the temporal properties of net-load. Lastly, the output of the GRU layer is passed to sequentially stacked linear layers with decreasing dimensions, which output the classification probability value. The probability value indicates whether the input data is the actual CPOW measurement data or the generated data. The structure of $\mathcal{D}$ is shown in Fig. 4.

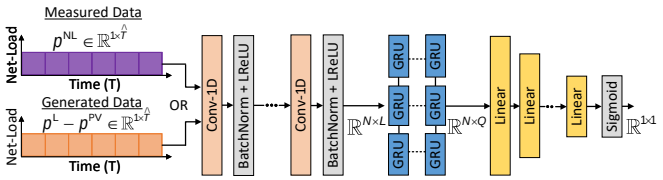

Fig. 4. Illustration of the proposed Conv-EDNet+ model.

IV. Training Conv-EDNet and Conv-EDNet+

An end-to-end disaggregation approach in the time domain allows using element-wise mean squared error (MSE) as the loss function for training the different blocks of the Conv-EDNet model. Let $\overline{\boldsymbol{p}}_i^{\mathrm{L}}$ and $\overline{\boldsymbol{p}}_i^{\mathrm{PV}}$ be the ground truth values of the native load and PV generation, respectively. The disaggregation loss function is modeled as follows:

$$\mathcal{L}^{\mathrm{MSE}} = \sum\nolimits_{j \in \{\mathrm{L,PV}\}} (\boldsymbol{p}_i^{\mathrm{j}} - \overline{\boldsymbol{p}}_i^{\mathrm{j}})^2 / \hat{T}. \tag{5}$$

For Conv-EDNet+ model, along with (5), the binary cross-entropy loss function used for training the $\mathcal{G}$ and $\mathcal{D}$ is shown below:

$$\mathcal{L}_{\gamma}^{\mathrm{BCE}} = -[\gamma \log \mathcal{D}(\boldsymbol{p}_i^{\mathrm{NL}}) + (1-\gamma) \log(1 - \mathcal{D}(\mathcal{G}(\boldsymbol{p}_i^{\mathrm{NL}})))], \tag{6}$$

where γ is the ground truth value. If the input to $\mathcal{D}$ is the actual measurement data $\boldsymbol{p}_i^{\text{NL}}$, then $\gamma = 1$, and if the input is the generated data $\boldsymbol{p}_i^{\text{L}} - \boldsymbol{p}_i^{\text{PV}}$, then $\gamma = 0$.

All the Conv-EDNet model blocks are jointly optimized. The model is trained for a pre-determined number of epochs (E), and the weights of all the different components are updated simultaneously to minimize the loss function (5). The training procedure for Conv-EDNet+ is shown in Algorithm 1. For notation simplicity, the subscript i is dropped. For the first $\hat{E}$ epochs, only the $\mathcal{G}$ is trained using the loss function (5). Beyond $\hat{E}$ epochs, $\mathcal{G}$ and $\mathcal{D}$ are jointly trained. While training $\mathcal{G}$, the loss of $\mathcal{D}$ is computed by using the true label 1 instead of using the fake label 0 (step 9). This is done since the goal of $\mathcal{G}$ is to trick the $\mathcal{D}$. The remainder of the training procedure is similar to that of the conventional GANs.

Algorithm 1 Conv-EDNet+ training algorithm

Input: Training data $(\boldsymbol{p}^{\text{NL}}, \boldsymbol{f}^{\text{Irr./Temp.}}, \boldsymbol{F}^{\text{STFT-V/I}}, \overline{\boldsymbol{p}}^{\text{L}}, \overline{\boldsymbol{p}}^{\text{PV}})$, total epoch count E, threshold epoch $\hat{E}$, total number of samples B, mini-batch size b, $\lambda = 0$.

Output: $\boldsymbol{\Theta}^*_{ae_1}, \boldsymbol{\Theta}^*_f, \boldsymbol{\Theta}^*_s, \boldsymbol{\Theta}^*_{ae_2}, \boldsymbol{\Theta}^*_d$

1: **for** epoch $e \in [1, E]$ **do**
2: **for** mini-batch $k \in [1, B/b]$ **do**
3: $\boldsymbol{X}_{\boldsymbol{ae_1}} \leftarrow f_{ae_1}(\boldsymbol{p}^{\text{NL}}, \boldsymbol{\Theta}_{ae_1})$
4: $\boldsymbol{X}_{\boldsymbol{f}} \leftarrow f_f(\boldsymbol{f}^{\text{Irr./Temp.}}, \boldsymbol{F}^{\text{STFT-V/I}}, \boldsymbol{\Theta}_f)$
5: $\boldsymbol{X}_{\boldsymbol{s}}^{\text{PV}}, \boldsymbol{X}_{\boldsymbol{s}}^{\text{L}} \leftarrow f_s(\boldsymbol{X}_{\boldsymbol{ae_1}}, \boldsymbol{X}_{\boldsymbol{f}}, \boldsymbol{\Theta}_s)$
6: $\boldsymbol{p}^{\text{L}}, \boldsymbol{p}^{\text{PV}} \leftarrow f_{ae_2}(\boldsymbol{X}_{ae_1} \otimes \boldsymbol{X}_s^{\text{PV}}, \boldsymbol{X}_{ae_1} \otimes \boldsymbol{X}_s^{\text{L}}, \boldsymbol{\Theta}_{ae_2})$
7: $\zeta^{\text{Gen.}} \leftarrow f_d(\boldsymbol{p}^{\text{L}} - \boldsymbol{p}^{\text{PV}}, \boldsymbol{\Theta}_d)$
8: Compute generator $\mathcal{G}$ loss $\mathcal{L}^{\text{MSE}}$ using (5)
9: Compute discriminator $\mathcal{D}$ loss $\mathcal{L}^{\text{BCE}}_{\gamma=1}$ using $\zeta^{\text{Gen.}}$ and (6)
10: $\mathcal{L}^{\mathcal{G}} = \mathcal{L}^{\text{MSE}} + \lambda \mathcal{L}^{\text{BCE}}_{\gamma=1}$
11: $\boldsymbol{\Theta}_{ae_1}, \boldsymbol{\Theta}_f, \boldsymbol{\Theta}_s, \boldsymbol{\Theta}_{ae_2} = \text{Adam}(\nabla_\Theta \mathcal{L}^{\mathcal{G}}, \boldsymbol{\Theta}_{ae_1}, \boldsymbol{\Theta}_f, \boldsymbol{\Theta}_s, \boldsymbol{\Theta}_{ae_2})$
12: **if** $e \geq \hat{E}$ **then**
13: $\lambda = 1$
14: $\zeta^{\text{Real}} \leftarrow f_d(\boldsymbol{p}^{\text{NL}}, \boldsymbol{\Theta}_d)$
15: $\zeta^{\text{Gen.}} \leftarrow f_d(\boldsymbol{p}^{\text{L}} - \boldsymbol{p}^{\text{PV}}, \boldsymbol{\Theta}_d)$
16: Compute discriminator $\mathcal{D}$ loss $\mathcal{L}^{\text{BCE}}_{\gamma=1}$ using ζ^{Real} and (6)
17: Compute discriminator $\mathcal{D}$ loss $\mathcal{L}^{\text{BCE}}_{\gamma=0}$ using $\zeta^{\text{Gen.}}$ and (6)
18: $\mathcal{L}^{\mathcal{D}} = (\mathcal{L}^{\text{BCE}}_{\gamma=1} + \mathcal{L}^{\text{BCE}}_{\gamma=0})/2$
19: $\boldsymbol{\Theta}_d = \text{Adam}(\nabla_\Theta \mathcal{L}^{\mathcal{D}}, \boldsymbol{\Theta}_d)$
20: **end if**
21: **end for**
22: **end for**
23: **return** $(\boldsymbol{\Theta}^*_{ae_1}, \boldsymbol{\Theta}^*_f, \boldsymbol{\Theta}^*_s, \boldsymbol{\Theta}^*_{ae_2}, \boldsymbol{\Theta}^*_d)$

V. Case Study

In this section, numerical simulations are performed using Conv-EDNet and Conv-EDNet+. Due to limited deployment of CPOW measurement units in distribution grids and privacy/security concerns, this type of data is not publicly available. Hence, we rely on synthetic CPOW data obtained using Matlab-Simulink. Care has been taken to insert random noise into the synthetic data to reflect measurement errors and prevent the proposed disaggregation model from learning the synthetic network model as is. The network model built is shown in Fig. 5. Currently, only one CPOW unit located at the feeder head is considered. The PV generator rating is 1 MW, and the peak aggregated demand is 15 MW. The data sampling frequency of 3 kHz is selected to meet the criteria of CPOW measurements. A holistic synthetic dataset is then obtained by considering different days of the week, different seasons, different cloud-cover scenarios, and different network fault scenarios. The total training data spans over 15 days and the testing data over five days. Owing to the limited computation resource availability, we downsample the encoder input from 3 kHz to 0.2 kHz. However, the feature extractor network is provided with 3 kHz data without downsampling to ensure that the harmonic features are accurately extracted. The hyperparameters and their values are shown in Table I. All experiments are implemented using an NVIDIA Titan X (Pascal) GPU with 12GB RAM using Python and Pytorch.

TABLE I
Hyperparameters

Param.	Description	Value
b	Mini-batch size in ae_1, ae_2, s, f, d.	32
B	Number of bottleneck channels in M.	64
E	Total training epochs.	200
$\hat{E}$	Epochs for which $\mathcal{G}$ is independently trained.	75
F	STFT frequency components (spectrogram height).	75
H	Number of channels in RT-GLU block.	256
K	Kernel size* in ae_1, ae_2, f, s, d.	2
L	Length* of each channel in ae_1, ae_2, f, s, d.	199
M	Number of RT-GLU blocks in sequence.	4
N	Number of channels in ae_1, ae_2, f, s, d.	512
Q	Number of features in GRU hidden state in d.	64
R	Number of sequential repeats of RT-GLU blocks.	6
S	Convolution stride* in ae_1, ae_2, f, s, d.	1
$\hat{T}$	Data segment length* in ae_1, ae_2, s, d.	200
$\hat{T}$	Data segment length* in f.	3000
$\tilde{T}$	STFT spectrogram width*.	200

*Size in samples.

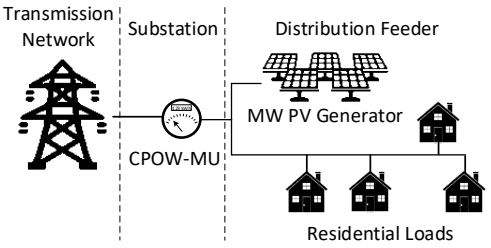

Fig. 5. Distribution network structure for synthetic data generation.

A. Performance analysis

Herein, we compare the performance of the Conv-EDNet and Conv-EDNet+ approach with the existing approaches in the literature. From the existing audio source separation methods, we have used Conv-TasNet [7], and from ED literature, we have used denoising autoencoder (DAE-Net) approach [14]. The models for the different methods are accordingly modified for CPOW data, trained, and evaluated independently. Finally, to ensure the predictions satisfy the relationship shown in Section II, the predictions are passed through a post-disaggregation adjustment stage shown in [3]. Fig. 6 and Fig. 7 show the results under normal operation and PV generator fault condition, respectively. Table II shows performance metrics quantifying the predictions made by the four methods: mean absolute error (MAE), root mean squared error (RMSE), and signal aggregate error (SAE) [15].

From Table II, we observe that the Conv-EDNet+ model outperforms all the other compared methods while the Conv-EDNet is the second-best performing model. These results can be corroborated using Fig. 6, which shows the predictions

for a partially cloudy day, wherein the day transitions from overcast to sunny. For load predictions, similar and accurate predictions are observed for all the compared methods, with Conv-EDNet+ predictions being closest to the ground truth. For PV predictions, we observe that the Conv-EDNet and Conv-EDNet+ models are accurately able to detect the onset of PV generation. This can be attributed to the novel feature extraction module, which has enhanced PV generation detection capability owing to its dependence on irradiance and STFT-based harmonic features. Due to its absence in the remaining two methods, they underperform, with DAE-Net producing the least accurate results. For the remainder of the PV generation profile, we observe that Conv-EDNet+ significantly outperforms amongst the methods compared.

Fig. 7 shows the predictions on a partly cloudy winter day for a fault event wherein the PV generator was disconnected from the grid at noon. We observe that Conv-EDNet+, followed by Conv-EDNet with a small time-lag, is accurately able to detect the fault and immediately drop the PV generation value. The Conv-TasNet and the DAE-Net are slow in detecting the fault. Thus, Conv-EDNet and Conv-EDNet+ outperform the traditional approaches even when CPOW data is used for training all the approaches. This demonstrates that the use of CPOW data combined with a dedicated network for time-frequency domain feature characterization results in a sharp increase in prediction accuracy.

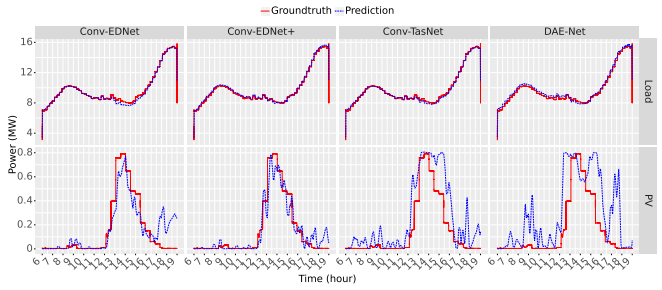

Fig. 6. Comparison of Conv-EDNet, Conv-EDNet+, Conv-TasNet, and DAE-Net under normal operating condition.

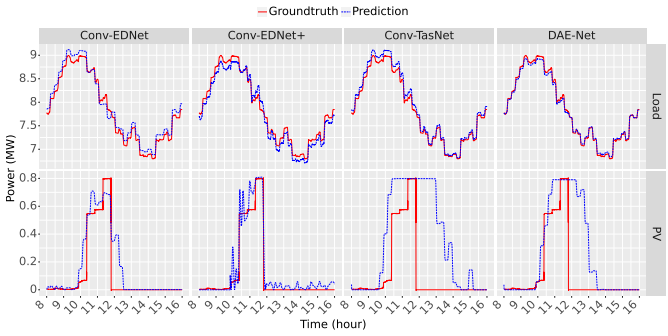

Fig. 7. Comparison of Conv-EDNet, Conv-EDNet+, Conv-TasNet, and DAE-Net under PV generator fault condition.

VI. Conclusion

In this paper, we have presented a fully convolutional time-frequency domain-based ED network for performing electric distribution network feeder level ED using CPOW measurements. We have also proposed an enhanced version of Conv-EDNet, called Conv-EDNet+, which uses adversarial learning to further enhance robustness and accuracy. For the first time, the use of CPOW measurements along with novel deep learning-based models for distribution system-level ED has been proposed. The numerical results have validated the superiority of the approach compared to the current state-of-the-art approaches. Future work will focus on a detailed analysis of Conv-EDNet/Conv-EDNet+, more complex distribution networks having PV generators of different sizes, and utilizing domain adaptation to synthetically trained models on real-world data.

TABLE II
Comparison of performance metrics.

Method	Load (MW)			PV (MW)		
	MAE	RMSE	SAE	MAE	RMSE	SAE
Conv-EDNet	0.115	0.121	0.127	0.083	0.086	4.604
Conv-EDNet+	**0.108**	**0.112**	**0.124**	**0.055**	**0.059**	**1.505**
Conv-TasNet	0.339	0.352	0.391	0.272	0.274	10.86
DAE-Net	0.205	0.373	0.195	0.532	0.544	16.45

References

[1] Alexandra von Meier, et al., "Synchrophasor Monitoring for Distribution Systems: Technical Foundations and Applications," North American SynchroPhasor Initiative, Tech. Rep., 01 2018.

[2] J. M. Alcalá, J. Urena, A. Hernánde, and D. Gualda, "Sustainable homecare monitoring system by sensing electricity data," *IEEE Sensors Journal*, vol. 17, no. 23, pp. 7741–7749, 2017.

[3] F. Kabir *et al.*, "Estimation of behind-the-meter solar generation by integrating physical with statistical models," in *2019 IEEE Int. Conf. on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, 2019, pp. 1–6.

[4] F. Bu *et al.*, "Disaggregating customer-level behind-the-meter PV generation using smart meter data and solar exemplars," *IEEE Trans. Power Syst.*, pp. 1–1, 2021.

[5] W. Li *et al.*, "Real-time energy disaggregation at substations with behind-the-meter solar generation," *IEEE Trans. Power Syst.*, vol. 36, no. 3, pp. 2023–2034, 2021.

[6] R. Saeedi *et al.*, "An adaptive machine learning framework for behind-the-meter load/PV disaggregation," *IEEE Trans. Ind. Informat.*, vol. 17, no. 10, pp. 7060–7069, 2021.

[7] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing Ideal Time–Frequency Magnitude Masking for Speech Separation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 27, no. 8, pp. 1256–1266, 2019.

[8] D. Samuel, A. Ganeshan, and J. Naradowsky, "Meta-learning extractors for music source separation," in *2020 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 816–820.

[9] X. Zhou *et al.*, "Harmonic impacts of inverter-based distributed generations in low voltage distribution network," in *2012 3rd IEEE Int. Symp. on Power Electronics for Distributed Generation Systems (PEDG)*, 2012, pp. 615–620.

[10] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," in *Int. Conf. on machine learning*, 2017, pp. 933–941.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conf. on computer vision and pattern recognition*, 2016, pp. 770–778.

[12] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Int. conf. on machine learning*, 2013, pp. 1310–1318.

[13] K. Tan, J. Chen, and D. Wang, "Gated residual networks with dilated convolutions for monaural speech enhancement," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 27, no. 1, pp. 189–198, 2019.

[14] J. Kelly and W. Knottenbelt, "Neural NILM: Deep neural networks applied to energy disaggregation," in *Proceedings of the 2nd ACM Int. Conf. on embedded systems for energy-efficient built environments*, 2015, pp. 55–64.

[15] M. Kaselimi *et al.*, "EnerGAN++: A generative adversarial gated recurrent network for robust energy disaggregation," *IEEE Open Journal of Signal Processing*, vol. 2, pp. 1–16, 2021.