Recent Results of Energy Disaggregation with Behind-the-Meter Solar Generation

Ming Yi, Meng Wang
Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute
Troy, USA

***Abstract*—The rapid deployment of renewable generations such as photovoltaic (PV) generations brings great challenges to the resiliency of existing power systems. Because PV generations are volatile and typically invisible to the power system operator, estimating the generation and characterizing the uncertainty are in urgent need for operators to make insightful decisions. This paper summarizes our recent results on energy disaggregation at the substation level with Behind-the-Meter solar generation.**

We formulate the so-called "partial label" problem for energy disaggregation at substations, where the aggregate measurements contain the total consumption of multiple loads, and the existence of some loads is unknown. We develop two model-free disaggregation approaches based on deterministic dictionary learning and Bayesian dictionary learning, respectively. Unlike conventional methods which require fully annotated training data of individual loads, our approaches can extract load patterns given partially labeled aggregate data. Therefore, our partial label formulation is more applicable in the real world. Compared with deterministic dictionary learning, the Bayesian dictionary learning-based approach provides the uncertainty measure for the disaggregation results, at the cost of increased computational complexity. All the methods are validated by numerical experiments.

***Index Terms*—energy disaggregation, Behind-the-Meter solar generation, partial labels, uncertainty modeling**

I. Introduction

The presence of renewable generations in power systems, especially solar generations, has increased rapidly in recent decades. Reference [1] reports that the global capacity of photovoltaic (PV) installment reached 634 GW in 2019. The solar capacity in 2019 has grown nearly 400 times since 2000. California Independent System Operator (ISO) estimates that the renewable energy generations will contribute 50% power supplies by 2030 in California [2].

The wide deployment of renewable generations decreases greenhouse gas emissions, however, but also brings great challenges to the reliability and resiliency of existing power systems. For example, at the substation level, the measurements of power consumptions are the net loads that contain different types of loads. The solar generation is invisible to the power system operator and thus is behind-the-meter (BTM). Because of the stochastic nature and high volatility of renewable generations, the accurate estimation of generated energy is challenging. Energy disaggregation at the substation level (EDS) aims to disaggregate each individual load[1] from aggregate measurement. The accurate information for load consumption are crucial for power system planning and operations, such as hosting capacity evaluation [3], [4], demand response and load dispatching [5], [6] and load forecasting [7], [8].

Energy disaggregation problem at the household level (EDH) has been extensively studied, see, e.g., [9]–[12], also under the terminology *non-intrusive load monitoring (NILM)* [9]–[15]. The electric appliances are typically single-state or multi-state devices and patterns of their power consumptions usually are repeatable. The general procedure for EDH methods is to first collect historical power consumption for each individual appliance and learn patterns from these well-annotated data. Then EDH methods disaggregate power consumptions for each appliance from the aggregate data based on these patterns. In comparison, obtaining historical power consumption for each individual load at the substation level is more difficult, as the measurements at the substation level are highly aggregated from different types of loads. Even though the operator has information about load types attached to a substation, whether a certain load is consuming/generating energy or not in a certain time interval is not already clear. One example is the BTM solar generation. Thus, the measurements at the substation usually contain multiple loads and are partially labeled. It is more challenging to learn distinctive load profiles under this situation than learning from measurements on individual loads. Moreover, the volatility of load and renewable generation often lead to significant estimation errors. However, to the best of our knowledge, there is no work that provides a confidence measure of the energy disaggregation results.

This paper summarizes our recent results for solving these two challenges. Given partially labeled training data, our work [16] proposes a deterministic dictionary learning-based method to learn load patterns and disaggregate the aggregate measurements into individual loads in real-time. Note that [16] is a deterministic approach and therefore is unable to provide the confidence measure of the estimation results. To estimate the reliability of the disaggregation results, in [17], we propose a probabilistic energy disaggregation approach based on Bayesian dictionary learning.

The contributions of this paper are three folds: 1. We summarize our works [16] and [17] for solving the "partial label" problem and modeling the uncertainty. 2. We compare these two methods and other two existing works in the experiment. (3) We provide more testing cases for these two methods in

[1]Generation is considered as a negative load in this paper.

this paper.

The remainder of this paper is organized as follows. Section II explains our partial label formulation. Section III discusses our proposed deterministic approach to solve the issue of partial labels and introduces our proposed Bayesian method for modeling the uncertainty of disaggregation results. Section IV summarizes this paper.

II. Problem Formulation

A substation is connected to C ($C > 1$) types of loads in total. Let $\boldsymbol{x} \in \mathbb{R}^P$ denote the aggregate measurement with window length P. Let a binary vector $\boldsymbol{y} = [y^1, y^2, ..., y^C] \subseteq \{0,1\}^C$ denote the load existence in $\boldsymbol{x}$. For example, when $C = 3$, $\boldsymbol{y} = [0,0,1]^T$ means that only load 3 exists in $\boldsymbol{x}$.

In our paper [16], we propose a "partial label formulation" where the operator only knows partial entries in $\boldsymbol{y}$. The partial labels can be obtained by designing a load detector for each load separately [18], [19] or from engineering experience. As described in [16], annotating partial labels has a lower cost for manpower or communication burdens than annotating all the labels. Moreover, if a detector fails to identify some loads [20], we can only obtain partial labels.

Let $\bar{\boldsymbol{X}} = [\bar{\boldsymbol{x}}_1, \bar{\boldsymbol{x}}_2, ..., \bar{\boldsymbol{x}}_N] \in \mathbb{R}^{P\times N}$ denote N measurements. $\bar{\boldsymbol{x}}_i$ denotes the data at the ith time window. $\boldsymbol{y}_i \in \{0,1\}^C$ denotes the labels in $\bar{\boldsymbol{x}}_i$. Let label matrix $\boldsymbol{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_N]$ denote all the labels in $\bar{\boldsymbol{X}}$. Let Ω denote the indices of known entries in $\boldsymbol{Y}$. $\boldsymbol{Y}_\Omega$ denotes all the known partial labels. In the above example, if one only knows $\bar{\boldsymbol{x}}$ contains load 3 and does not know whether the other two loads exist or not, then the corresponding $\boldsymbol{Y}_\Omega$ is $[?,?,1]^T$ where ? denotes one does not know the corresponding load exists or not.

Fig. 1 illustrates our partial label formulation. The aggregate data are aggregated from two industrial loads and one solar generation. Each subfigure shows patterns of aggregate data and the corresponding individual loads at the same time interval. In all these four cases, the label is $[?,?,1]^T$, indicating that load 3 always exists, while the existence of loads 1 and 2 is unknown.

Given training dataset $\bar{\boldsymbol{X}}$, the corresponding partial label matrix $\boldsymbol{Y}_\Omega$ and an aggregate measurement $\hat{\boldsymbol{x}} \in \mathbb{R}^P$, the objective of this paper is to: (1) learn distinctive patterns of individual loads from $\bar{\boldsymbol{X}}$ and disaggregate $\hat{\boldsymbol{x}}$, and (2) characterize the uncertainty of the disaggregation results.

III. Methodology

In this section, we present our two model-free approaches based on deterministic dictionary learning [16] and Bayesian dictionary learning [17], respectively.

A. Deterministic Energy Disaggregation

To learn patterns of each individual load from the given training data $\bar{X}$, we formulate a deterministic dictionary learning problem,

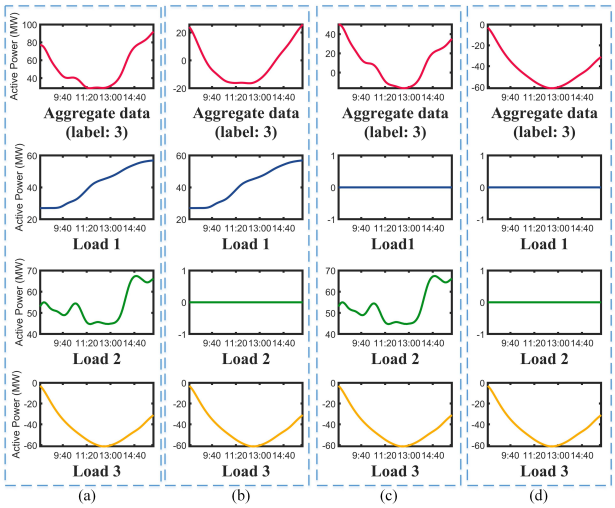

Fig. 1: An example of partial label formulation. There are three load types in the aggregate data. All four aggregate data have the same partial label $[?,?,1]^T$. The aggregate data contain (a) all three loads; (b) load 1 and 3; (c) load 2 and 3; (d) only load 3. [16]

$$\min_{A,D} f(A,D) = \|\bar{X} - \Sigma_{i=1}^{C} D_i A_i\|_F^2 + \Sigma_{i=1}^{C} \lambda_i \Sigma_{j:i\notin y_j} \|A_i^j\| + \lambda_D \mathrm{Tr}(D\Theta D^\intercal) \quad (1)$$

$$\text{s.t. } \|d^m\|_2 \le 1, d^m \ge 0, m = 1, \cdots, K \quad (2)$$

$$c_i A_i \ge 0, \forall i \quad (3)$$

where $D_i \in \mathbb{R}^{P\times K_i}$ denotes the dictionary for load i, and $A_i \in \mathbb{R}^{K_i\times N}$ denotes the corresponding coefficients of load i. A_i^j is the jth column in A_i. $A = [A_1; A_2; \cdots; A_C] \in \mathbb{R}^{K\times N}$ is the matrix that contains all coefficients. $D = [D_1, D_2, \cdots, D_C] \in \mathbb{R}^{P\times K}$ is the matrix that contains all dictionaries. d^m is the mth column in D. $K = \Sigma_{i=1}^{C} K_i$. $\mathrm{Tr}(\cdot)$ is the trace operator, and $D^\intercal$ represents the transpose matrix of D. λ_i and λ_D are pre-defined hyper-parameters.

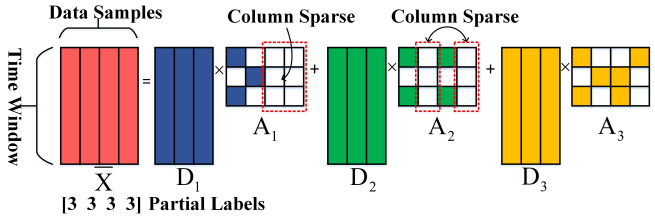

Fig. 2: The dictionary representation in Fig. 1. The coefficients A_1 and A_2 are column-sparse. [16]

The first term $\|\bar{X} - \Sigma_{i=1}^{C} D_i A_i\|_F^2$ is the standard reconstruction error in dictionary learning. It measures the reconstruction error between the original data and the learned dictionaries and coefficients. $\sum_{j:i\notin y_j} \|A_i^j\|$ is the column sparsity constraint. The motivation of using this regularization is illustrated in Fig. 2, which shows the dictionary representation of Fig. 1. Because the training data only have partial label 3, load 1 and load 2 may not exist in the training data. Therefore, we

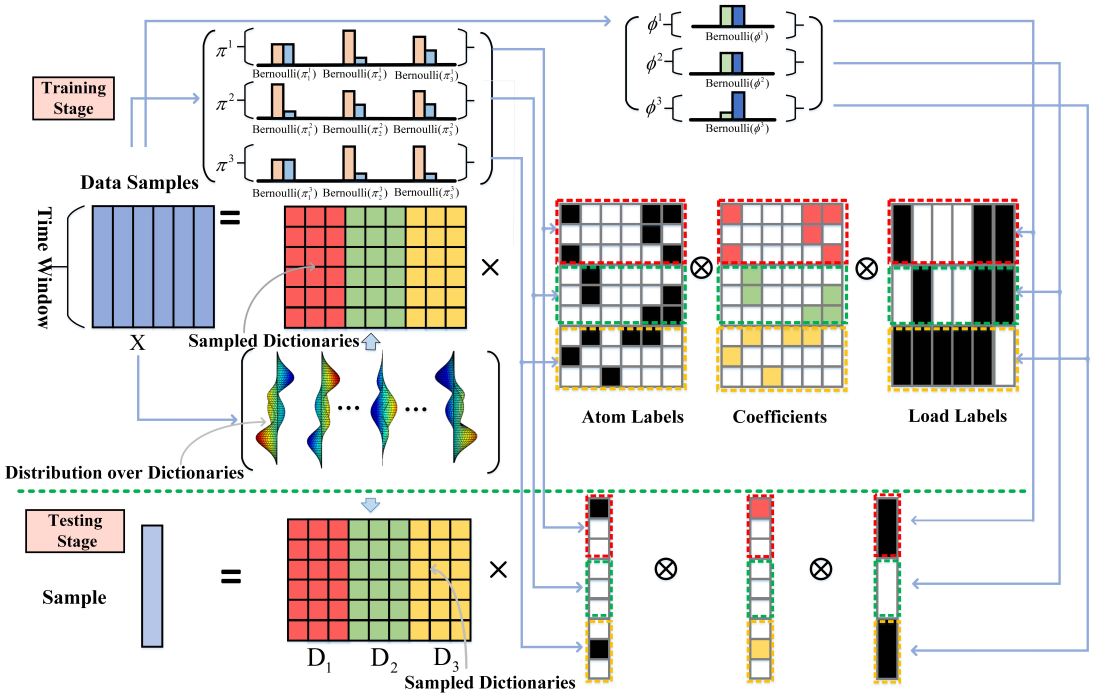

Fig. 3: An illustrative framework of the proposed Bayesian method. In the training stage, our method learns the posterior distribution of atom labels, coefficients and load labels from the training data. In the testing stage, the method samples learned distributions of the dictionary and learns the posterior distribution for atom labels, coefficients and load labels, respectively, for test data. [17]

impose the column sparsity on A_1 and A_2 to promote the group sparsity of A_1 and A_2.

The incoherence term $\text{Tr}(D\Theta D^\intercal)$ is defined as

$$\text{Tr}(D\Theta D^\intercal) = \Sigma_{m=1}^{K}\Sigma_{p=1}^{K}\theta_{mp}(d^m)^\intercal d^p. \tag{4}$$

The (m,p)th entry θ_{mp} in the weight matrix $\Theta \in R^{K\times K}$ is 0 if d^m and d^p are in the same dictionary and 1 otherwise. The incoherence term promotes a discriminative dictionary such that D_i and D_j are as different as possible. The discriminative dictionaries are able to enhance the disaggregation performance.

Given an aggregate test data $\hat{x}$, we aim to disaggregate the aggregate measurement into individual load c, denoted by $\hat{x}^c$. The objective function in the testing stage can be written as

$$\min_{w\in\mathbb{R}^q} \|\hat{x} - \hat{D}\tilde{A}w\|_2 + \mu\|w\|_1, \tag{5}$$

where we select a submatrix $\tilde{A} = [\tilde{A}_1; \cdots; \tilde{A}_n] \in \mathbb{R}^{K\times q}$ from $\hat{A}$. $\hat{D}$, $\hat{A}$ is the solution by solving the objective function (1). μ is a pre-defined hyper-parameter. The intuition is that some load combinations are repetitive in the training data. We can select some representative combinations and disaggregate the aggregate measurement with respect to these combinations to improve the disaggregation accuracy. Let $\hat{w}$ be the solution to (5), then the estimated load for load c is $\hat{x}^c = \hat{D}_i\tilde{A}_i\hat{w}$.

B. Bayesian Energy Disaggregation

In [17], we propose a Bayesian method to deal with partial label data and provide the confidence measure of our disaggregation results. An overall framework is shown in Fig. 3. Given the training data $\bar{X}$ and partial labels $\boldsymbol{Y}_\Omega$, the proposed Bayesian method learns the posterior distribution of dictionaries and coefficients in the training stage. At the testing stage, the method learns the distributions of coefficients based on the learned distributions of dictionaries. The distribution of $\hat{x}^c$ is then computed, where $\hat{x}^c$ is the estimated power consumption of load c. The mean of the distribution of $\hat{x}^c$ is used as the estimation of the load c and the covariance is computed to measure the uncertainty.

The proposed method is based on a hierarchical probabilistic model. The prior distribution of the aggregate data x_i can be written as

$$\bar{x}_i = \sum_{c=1}^{C} D_c\boldsymbol{\omega}_i^c + \boldsymbol{\epsilon}_i \tag{6}$$

$$\boldsymbol{\omega}_i^c = (\boldsymbol{z}_i^c \odot \boldsymbol{s}_i^c)y_i^c \tag{7}$$

for all $i = 1, 2, 3, ..., N$, $c = 1, 2, 3, ..., C$, where $\boldsymbol{\omega}_i^c \in \mathbb{R}^{K_c}$ is the coefficients for D_c, and $\boldsymbol{\epsilon}_i$ is the measurement noise. In (7), $\odot$ represents the element-wise product. Let $\boldsymbol{d}_k^c$ denote the kth column in the dictionary D_c. $\boldsymbol{d}_k^c$ is sampled from a multivariate Gaussian distribution $\mathcal{N}(\boldsymbol{0}, \frac{1}{\lambda_d}\boldsymbol{I}_P)$, where λ_d is a pre-defined scalar, and $\boldsymbol{I}_P$ is an identity matrix with size $P \times P$. The noise $\boldsymbol{\epsilon}_i$ is sampled from Gaussian $\mathcal{N}(\boldsymbol{0}, \frac{1}{\gamma_\epsilon}\boldsymbol{I}_P)$. One can see from (7) that $\boldsymbol{\omega}_i^c$ is the element-wise product of $\boldsymbol{z}_i^c$ and $\boldsymbol{s}_i^c$ and then multiplied by y_i^c. y_i^c is a binary variable sampled from a Bernoulli distribution and $y_i^c = 1$ indicates that load c exists in x_i. $\boldsymbol{z}_i^c$ is a binary vector. Let $\boldsymbol{z}_{ik}^c$ denote the kth

entry of $\boldsymbol{z}_i^c$. $\boldsymbol{z}_{ik}^c = 1$ indicates d_k^c is used to represent x_i and 0 otherwise. $\boldsymbol{z}_{ik}^c$ is sampled from the Bernoulli distribution. Note that the Bayesian method is able to infer the actual dictionary size K_c by gradually pruning the dictionary size based on $\boldsymbol{z}_i^c$ in the training stage. Therefore, the Bayesian method is not sensitive to the selection of initial dictionary size. $\boldsymbol{s}_i^c$ is sampled from $\mathcal{N}(\mathbf{0}, \frac{1}{\gamma_s^c}\boldsymbol{I}_{Kc})$. We put Gamma priors on γ_s^c and γ_ϵ, respectively. The Gamma priors are conjugate priors of the Gaussian distribution. If conjugate priors are selected, we can derive the analytical solution of the posterior distribution in the variational inference, which simplifies the updating process. Note that our model selects conjugate priors to simplify the updating process.

Let $\boldsymbol{\Theta}$ denote all the latent variables. Given $\bar{X}$ and partial labels $\boldsymbol{Y}_\Omega$, the objective is to obtain the posterior $P(\boldsymbol{\Theta}, \boldsymbol{Y}_{\bar{\Omega}}|\boldsymbol{X}, \boldsymbol{Y}_\Omega)$. From the Bayes theorem,

$$P(\boldsymbol{\Theta}, \boldsymbol{Y}_{\bar{\Omega}}|\bar{X}, \boldsymbol{Y}_\Omega) = \frac{P(\boldsymbol{\Theta}, \bar{X}, \boldsymbol{Y})}{P(\bar{X}, \boldsymbol{Y}_\Omega)} \tag{8}$$

Because computing (8) directly is intractable, we use Gibbs sampling [21] to compute the posterior distribution. Gibbs sampling sequentially samples from the conditional probability of one variable in $\boldsymbol{\Theta}$ and $\boldsymbol{Y}_{\bar{\Omega}}$ while keeping all other variables fixed. These conditional distributions have closed-form expressions because of the conjugate priors, which leads to an efficient updating process.

In the testing stage, given the aggregate test data $\hat{\boldsymbol{x}}$, the goal of our approach is to estimate $\hat{x}^c$. A similar probabilistic model for $\hat{\boldsymbol{x}}$ and $\hat{x}^c$ is described as:

$$\hat{x} = \sum_{c=1}^{C} D_c(\hat{\boldsymbol{z}}^{\boldsymbol{c}} \odot \hat{\boldsymbol{s}}^{\boldsymbol{c}})\hat{y}^c + \hat{\boldsymbol{\epsilon}} \tag{9}$$

$$\hat{x}^c = D_c(\hat{\boldsymbol{z}}^{\boldsymbol{c}} \odot \hat{\boldsymbol{s}}^{\boldsymbol{c}})\hat{y}^c + \frac{\hat{\boldsymbol{\epsilon}}}{C}. \tag{10}$$

for all $c = 1, ..., C$, $k = 1, ..., K_c$.

The dictionary atom d_k^c is sampled from learned distribution $p(d_k^c|\boldsymbol{X}, \boldsymbol{Y}_\Omega)$ in the training stage. We also assume that $\hat{y}^c$ and $\hat{\boldsymbol{z}}^c$ are sampled from Bernoulli distributions. $\hat{\boldsymbol{s}}^c$ is sampled from $\mathcal{N}(\mathbf{0}, \frac{1}{\gamma_s^c}\boldsymbol{I}_{K_c})$ and $\hat{\boldsymbol{\epsilon}}$ is sampled from $\mathcal{N}(\mathbf{0}, \frac{1}{\hat{\gamma}_\epsilon}\boldsymbol{I}_P)$. Gibbs sampling is also employed for computing probabilistic distributions of $\hat{y}^c$, $\hat{\boldsymbol{z}}^c$, $\hat{\boldsymbol{s}}^c$, and $\hat{\gamma}_\epsilon$.

The per-iteration computational complexity of the Bayesian offline training is $\mathcal{O}(CK_cPN)$. The per-iteration computational complexity of the online testing is $\mathcal{O}(CK_cP)$. Thus, the computational complexity scales linearly with respect to the number of loads.

C. Uncertainty Modeling

Equipped with all learned posterior distributions, we then estimate the distribution of $\hat{x}^c$. However, it is intractable to obtain the explicit expression for the distribution of $\hat{x}^c$. Monte-Carlo integration [22] is employed to approximately compute the predictive mean and predictive variance.

Define

$$f(\boldsymbol{\Psi}) = D_c(\hat{\boldsymbol{z}}^c \odot \hat{\boldsymbol{s}}^c)\hat{y}^c \tag{11}$$

where $\boldsymbol{\Psi} = \{D_c, \hat{\boldsymbol{z}}^c, \hat{\boldsymbol{s}}^c, \hat{y}^c, \hat{\gamma}_\epsilon\}$. The predictive mean of $\hat{x}^c$ is computed by

$$E[\hat{x}^c] \approx \frac{1}{L}\sum_{l=1}^{l=L} f(\boldsymbol{\Psi}^l) \tag{12}$$

where L is the number of Monte-Carlo samples. More Monte-Carlo samples increase the estimation accuracy, at the cost of higher computational burden. Our experiments show that 50 Monte-Carlo samples suffice to provide accurate estimations of the predictive mean and the predictive variance. $\boldsymbol{\Psi}^l$ is sampled from the learned distributions of variables in $\boldsymbol{\Psi}$. $E[\hat{x}^c]$ is then used as the estimation of the power consumption of load C.

The predictive covariance is approximated by

$$\begin{aligned}\text{Var}[\hat{x}^c] =& E[\hat{x}^c\hat{x}^{cT}] - E[\hat{x}^c]E[\hat{x}^c]^T \\ \approx& \frac{\boldsymbol{I}_P}{LC}\sum_{l=1}^{l=L}\frac{1}{\hat{\gamma}_\epsilon^l} + \frac{1}{L}\sum_{l=1}^{l=L} f(\boldsymbol{\Psi}_l)f(\boldsymbol{\Psi}^l)^T \\ &-(\frac{1}{L}\sum_{l=1}^{l=L} f(\boldsymbol{\Psi}^l))(\frac{1}{L}\sum_{l=1}^{l=L} f(\boldsymbol{\Psi}^l)^T)\end{aligned} \tag{13}$$

Let σ_i $(i = 1, ..., P)$ denote all the singular values of $\text{Var}[\hat{x}^c]$. The uncertainty index U_c for individual load c and the uncertainty index U_{all} for total estimated loads are computed as

$$U_c = \Sigma_{i=1}^P \sigma_i \tag{14}$$

$$U_{\text{all}} = \Sigma_{c=1}^C U_c \tag{15}$$

The intuition is that a large variance indicates higher uncertainty of the estimation. The uncertainty index is able to characterize the confidence level of disaggregation results.

IV. Numerical Experiment

The performance of the proposed methods is evaluated on a partially labeled dataset. The dataset contains two industry loads and one solar generation. $N = 360$ training samples and $M = 300$ testing samples are generated. Even though the generated training samples contain up to three loads, each sample is annotated with only one label. The testing samples also contain up to three loads and have no label. In the following experiments, γ represents the percentage of the training data that measure individual loads. For example, $\gamma = 50\%$ denotes that 50% training data labeled as load c contain pure load c and the remaining 50% data contain other loads.

1) Error Metrics: Several metrics are employed to compute the disaggregation error. The standard Root Mean Square Error (RMSE) [23], [24] is defined as,

$$\text{RMSE}_c = \sqrt{\frac{\Sigma_{i=1}^M \|\hat{x}_i^c - x_i^c\|_2^2}{P \times M}}. \tag{16}$$

where $\hat{x}_i^c, x_i^c \in \mathbb{R}^P$ are the estimated and the ground-truth load c in the ith testing sample, respectively.

A new Total Error Rate (TER) is proposed to compute the disaggregation error of all the loads as follows,

$$\text{TER} = \frac{\Sigma_{i=1}^{M}\Sigma_{c=1}^{C}\min(\|\hat{x}_i^c - x_i^c\|_1, \|x_i^c\|_1)}{\Sigma_{i=1}^{M}\Sigma_{c=1}^{C}\|x_i^c\|_1} \tag{17}$$

The Weighted Root Mean Square Error (WRMSE) is proposed to take the uncertainty index into account. The weighted average disaggregation error is computed as,

$$\text{WRMSE}_c = \sqrt{\frac{\Sigma_{i=1}^{M}\frac{\|\hat{x}_i^c - x_i^c\|_2^2}{U_c(\hat{x}_i^c)}}{P\Sigma_{i=1}^{M}\frac{1}{U_c(\hat{x}_i^c)}}} \tag{18}$$

where $U_c(\hat{x}_i^c)$ denotes the uncertainty index of $\hat{x}_i^c$. A larger $U_c(\hat{x}_i^c)$ represents a less reliable estimation. If the estimated loads with higher disaggregation errors are accompanied by larger uncertainty indices, the RMSE_c could be much larger than WRMSE_c. The scenario that RMSE_c is much larger than WRMSE_c indicates that the unreliable estimation results are correctly flagged by higher uncertainty indices.

2) Methods: Our deterministic EDS method in [16] is abbreviated as "D-EDS." Our Bayesian EDS method in [17] is abbreviated as "B-EDS." Two other existing methods are employed for comparison. The work in [9] that is based on discriminative sparse coding is abbreviated as "DDSC," and the work [23] based on sum-to-k matrix factorization is abbreviated as "sum-to-k". Because we set the Monte-Carlo samples $L = 50$ in our method B-EDS, then D-EDS, DDSC and sum-to-k are averaged over 50 runs for a fair comparison. The comparisons of disaggregation performance of B-EDS, D-EDS, DDSC and sum-to-k are shown in Table I. $\gamma = 70\%$. Note that all the existing works such as DDSC and sum-to-k methods require fully labeled data to obtain accurate estimation. Directly applying the existing methods to partially labeled data leads to a low disaggregation accuracy. The proposed two approaches B-EDS and D-EDS are designed for partially labeled data and can achieve state-of-the-art disaggregation performance. Between these two methods, the disaggregation accuracy of B-EDS is slightly better. Moreover, one can see from Table I that the WRMSE_c is much smaller than the corresponding RMSE_c. As we discussed above, this means that those estimations with larger disaggregation errors also have large uncertainty indices. This validates the effectiveness of applying the proposed uncertainty index to measure the reliability of the disaggregation results.

The major advantage of B-EDS over D-EDS is that B-EDS is able to measure the confidence level of disaggregation results from the uncertainty index. We provide five case studies to verify the performance of uncertainty modeling of B-EDS.

- Case 1: we select test data from the testing datasets in Table I and this test data contains three types of loads.
- Case 2: the test data is as same as the data in Case 1 with an additional Gaussian noise $\mathcal{N}(0, 4^2)$ in each entry.
- Case 3: the test data is as same as the data in Case 1 with an additional Gaussian noise $\mathcal{N}(0, 6^2)$ in each entry.

Table I: Comparison between B-EDS, D-EDS, sum-to-k and DDSC methods on disaggregation accuracy

	B-EDS	D-EDS	sum-to-k	DDSC
RMSE_1	6.20	6.62	13.17	22.77
RMSE_2	5.19	6.34	11.35	23.86
RMSE_3	5.82	4.65	10.70	13.49
WRMSE_1	0.16	-	-	-
WRMSE_2	0.13	-	-	-
WRMSE_3	0.13	-	-	-
TER	8.97%	9.95%	20.61%	37.12%

- Case 4: the test data only contains one solar generation, but the pattern of solar generation is different from the solar patterns in the training data.
- Case 5: the test data contains the same load 1 and 2 as those in Case 1, and as well as a solar generation with a pattern different from the solar patterns in the training data.

Figs. 4 and 5 show the disaggregation performance of D-EDS on Case1 and Case 4. The aggregate measurement is shown in (a), and the disaggregation results are shown in (b)-(d) in both figures. In Case 1, the disaggregation results by D-EDS follow the actual load pattern. In Case 4, the disaggregation result of the solar generation does not follow the actual solar pattern because it is different from the learned pattern from the training data. In both cases, the disaggregation results contain some errors. That motivates using the Bayesian approach to compute a probabilistic distribution of load consumption rather than computing one deterministic estimation.

Fig. 6 shows the disaggregation performance of B-EDS on these five cases, and Table II shows the corresponding uncertainty index. Each subfigure in Fig. 6 plots the ground-truth load, the estimated load and the 99.7% confidence interval of the estimated load. One can see that in Cases 1-3, although there are some errors in the disaggregation results, the ground-truth loads are within the confidence interval. Moreover, the estimation errors increase slightly when the noise levels increase. Correspondingly, Table II shows that the total uncertainty indices in Case 1-3 also increase as the noise level increases, which indicates the effectiveness of using the uncertainty index to characterize the uncertainty in the estimation. In Case 4 and Case 5, because the patterns of solar generation are far from the patterns in the training data, the ground-truth load consumption may not fall inside the confidence interval (especially Case 5). The uncertainty indices in Table II also increase significantly, indicating that the estimated results are less reliable in these cases. Therefore, the users can use the uncertainty index to evaluate the accuracy of the disaggregation results. Table II also compares the TER of B-EDS and D-EDS, and B-EDS has a smaller disaggregation error of D-EDS.

The Bayesian method B-EDS has slightly better disaggregation performance than the deterministic approach D-EDS. The major advantage of the Bayesian approach is to measure the confidence level of the disaggregation results. However, the deterministic approach is much more computationally efficient than the Bayesian method. In Table I, the B-EDS requires

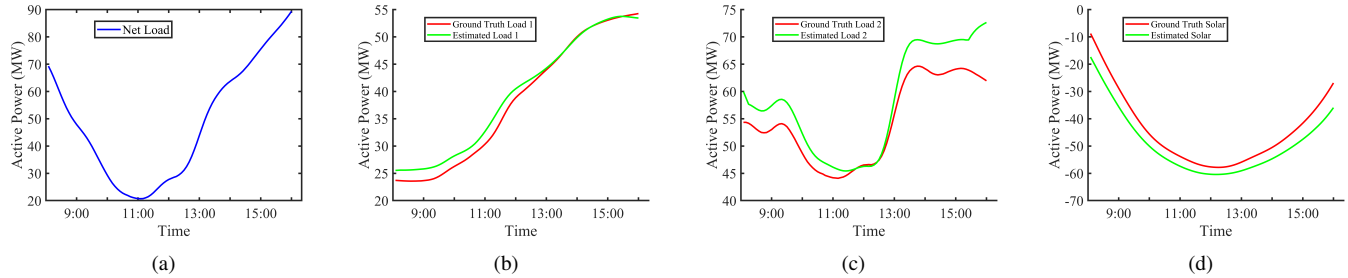

Fig. 4: Disaggregation performance of D-EDS on Case 1.(a) Net load. (b) The ground-truth and disaggregated load 1.(c) The ground-truth and disaggregated load 2. (d) The ground-truth and disaggregated solar.

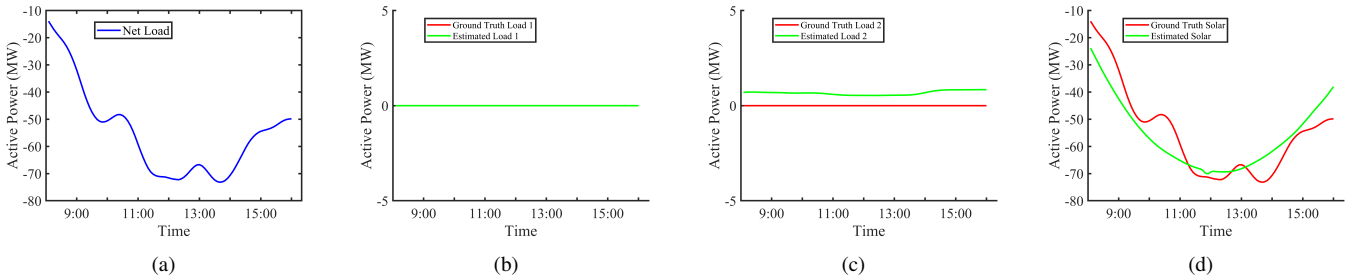

Fig. 5: Disaggregation performance of D-EDS on Case 4.(a) Net load. (b) The ground-truth and disaggregated load 1.(c) The ground-truth and disaggregated load 2. (d) The ground-truth and disaggregated solar.

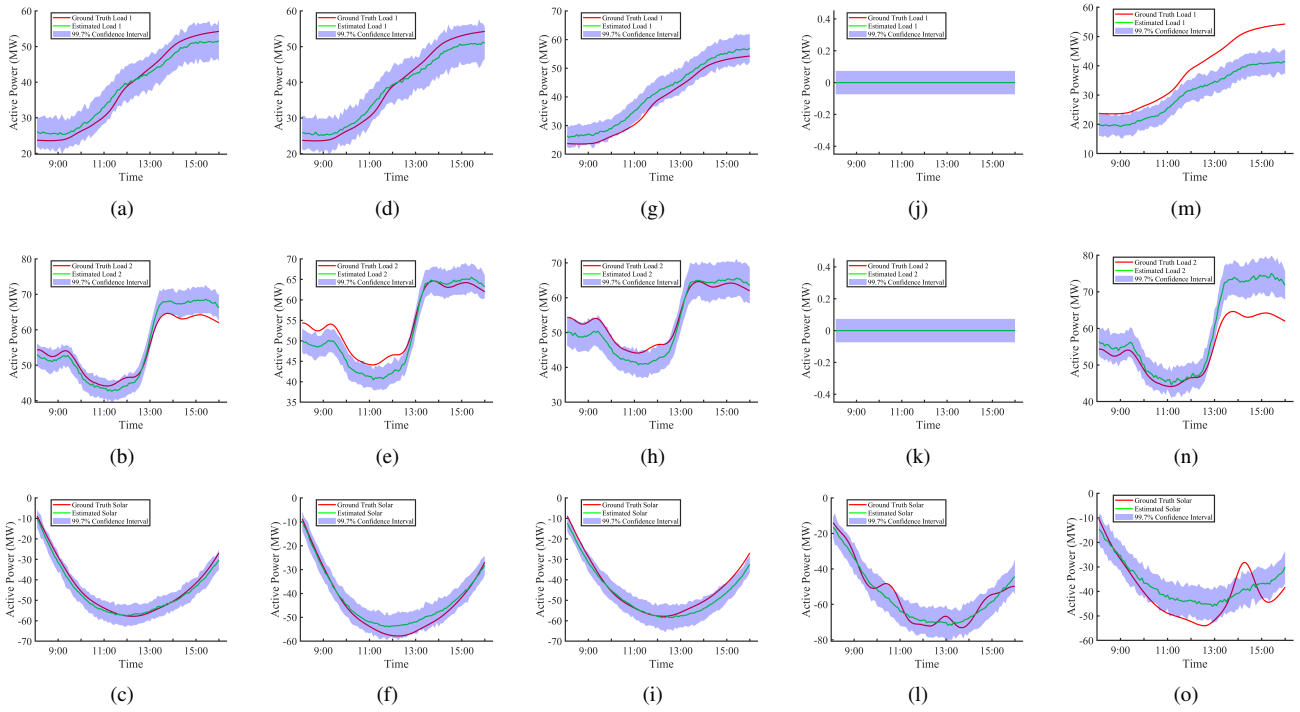

Fig. 6: The disaggregation performance of B-EDS on Cases 1-5. Each subfigure shows the ground-truth load, disaggregated load and the corresponding confidence interval. (a)-(c): the disaggregation results for Case 1. (d)-(f): the disaggregation results for Case 2 where the test data contains Gaussian noise $\mathcal{N}(0, 4^2)$. (g)-(i): the disaggregation results for Case 2 where the test data contains Gaussian noise $\mathcal{N}(0, 6^2)$. (j)-(l): the disaggregation results for Case 4 where the test data is a solar generation and its pattern is different from the training data. (m)-(o) the disaggregation results for Case 5 where the test data contains three loads, and the pattern of solar generation is different from the training data.

Table II: The uncertainty indices and disaggregation accuracy on five testing cases

	Case 1	Case 2	Case 3	Case 4	Case 5
U_1	243.72	280.35	201.52	0.058	160.89
U_2	116.07	101.24	215.23	0.060	394.41
U_3	249.89	257.78	287.23	703.48	440.26
U_{all}	609.69	639.37	703.98	703.60	788.44
B-EDS TER	4.77%	5.10%	7.00%	6.77%	12.97%
D-EDS TER	7.19%	8.86%	11.60%	11.01%	16.45%

around 50 seconds for the offline training and 4 seconds for each testing sample. In comparison, the D-EDS requires around 15 seconds for the offline training, and 0.9 seconds for each testing sample. If users want to know the reliability of the estimation results, the Bayesian method should be selected. In contrast, if users need to disaggregate the aggregate measurement with limited computational resources in real-time, the deterministic approach is a better option.

V. Conclusion

Energy disaggregation at substations with BTM solar generations has drawn increasing attention. Accurate energy disaggregation results are crucial for power system planning and operations. However, collecting training data with full labels at the substation level is challenging. Therefore, we propose the concept of partially labeled data which is applicable in practice and significantly reduces the burden of annotating data. This paper summarizes two new load disaggregation approaches. Both the deterministic approach and the Bayesian approach can achieve accurate disaggregation results on partially labeled data. Moreover, an uncertainty index is proposed to measure the reliability of the disaggregation results. To the best of our knowledge, this is the first work to provide the uncertainty measure for the energy disaggregation problem.

Acknowledgment

This work was supported in part by the NSF grant # 1932196, AFOSR FA9550-20-1-0122, and ARO W911NF-21-1-0255.

References

[1] Solar Power Europe, "Global market outlook for solar power 2020-2024," *[Online] Available: https://www.galileogreenenergy.com/wp-content/uploads/2020/10/SolarPowerEurope_Global_Market_Outlook_2020.pdf*, p. 20, 2020.

[2] ISO California, "What the duck curve tells us about managing a green grid," *[Online] Available: https://www.caiso.com/Documents/FlexibleResourcesHelpRenewables_FastFacts.pdf*, p. 1, 2016.

[3] M. S. S. Abad, J. Ma, D. Zhang, A. S. Ahmadyar, and H. Marzooghi, "Probabilistic assessment of hosting capacity in radial distribution systems," *IEEE Trans. Sustain. Energy*, vol. 9, no. 4, pp. 1935–1947, 2018.

[4] S. Wang, Y. Dong, L. Wu, and B. Yan, "Interval overvoltage risk based pv hosting capacity evaluation considering pv and load uncertainties," *IEEE Trans. Smart Grid*, vol. 11, no. 3, pp. 2709–2721, 2019.

[5] N. Mahdavi and J. H. Braslavsky, "Modelling and control of ensembles of variable-speed air conditioning loads for demand response," *IEEE Trans. Smart Grid*, vol. 11, no. 5, pp. 4249–4260, 2020.

[6] Z. Xuan, X. Gao, K. Li, F. Wang, X. Ge, and Y. Hou, "Pv-load decoupling based demand response baseline load estimation approach for residential customer with distributed pv system," *IEEE Trans. Ind. Appl.*, vol. 56, no. 6, pp. 6128–6137, 2020.

[7] Y. Wang, N. Zhang, Q. Chen, D. S. Kirschen, P. Li, and Q. Xia, "Data-driven probabilistic net load forecasting with high penetration of behind-the-meter pv," *IEEE Trans. Power Syst.*, vol. 33, no. 3, pp. 3255–3264, 2018.

[8] M. Sun, T. Zhang, Y. Wang, G. Strbac, and C. Kang, "Using bayesian deep learning to capture uncertainty for residential net load forecasting," *IEEE Trans. Power Syst.*, vol. 35, no. 1, pp. 188–201, 2019.

[9] J. Z. Kolter, S. Batra, and A. Y. Ng, "Energy disaggregation via discriminative sparse coding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 1153–1161.

[10] W. Kong, Z. Y. Dong, J. Ma, D. J. Hill, J. Zhao, and F. Luo, "An extensible approach for non-intrusive load disaggregation with smart meter data," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 3362–3372, 2016.

[11] K. Chen, Y. Zhang, Q. Wang, J. Hu, H. Fan, and J. He, "Scale-and context-aware convolutional non-intrusive load monitoring," *IEEE Trans. Power Syst.*, 2019.

[12] K. He, L. Stankovic, J. Liao, and V. Stankovic, "Non-intrusive load disaggregation using graph signal processing," *IEEE Trans. Smart Grid*, vol. 9, no. 3, pp. 1739–1747, 2016.

[13] G. W. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870–1891, 1992.

[14] J. M. Gillis, S. M. Alshareef, and W. G. Morsi, "Nonintrusive load monitoring using wavelet design and machine learning," *IEEE Trans. Smart Grid*, vol. 7, no. 1, pp. 320–328, 2015.

[15] S. M. Tabatabaei, S. Dick, and W. Xu, "Toward non-intrusive load monitoring via multi-label classification," *IEEE Trans. Smart Grid*, vol. 8, no. 1, pp. 26–40, 2016.

[16] W. Li, M. Yi, M. Wang, Y. Wang, D. Shi, and Z. Wang, "Real-time energy disaggregation at substations with behind-the-meter solar generation," *IEEE Trans. Power Syst.*, p. Early Access, 2020.

[17] M. Yi and M. Wang, "Bayesian energy disaggregation at substations with uncertainty modeling," *IEEE Transactions on Power Systems*, vol. 37, no. 1, pp. 764–775, 2021.

[18] X. He, L. Chu, R. C. Qiu, Q. Ai, Z. Ling, and J. Zhang, "Invisible units detection and estimation based on random matrix theory," *IEEE Trans. Power Syst.*, vol. 35, no. 3, pp. 1846–1855, 2019.

[19] X. Zhang and S. Grijalva, "A data-driven approach for detection and estimation of residential pv installations," *IEEE Trans. Smart Grid*, vol. 7, no. 5, pp. 2477–2485, 2016.

[20] S. Hosur and D. Duan, "Subspace-driven output-only based change-point detection in power systems," *IEEE Trans. Power Syst.*, vol. 34, no. 2, pp. 1068–1076, 2018.

[21] I. Yildirim, "Bayesian inference: Gibbs sampling," *Technical Note, University of Rochester*, 2012.

[22] J. Paisley, D. M. Blei, and M. I. Jordan, "Variational bayesian inference with stochastic search," in *Proc. 29th Int. Conf. Mach. Learn.*, 2012, pp. 1363–1370.

[23] A. Rahimpour, H. Qi, D. Fugate, and T. Kuruganti, "Non-intrusive energy disaggregation using non-negative matrix factorization with sum-to-k constraint," *IEEE Trans. Power Syst.*, vol. 32, no. 6, pp. 4430–4441, April 2017.

[24] O. Parson, S. Ghosh, M. Weal, and A. Rogers, "Non-intrusive load monitoring using prior models of general appliance types," in *26th AAAI Conf. Artificial Intelligence*, 2012.