Towards Secure Meter Data Analysis via Distributed Differential Privacy

Xiaojing Liao[†], David Formby[†], Carson Day[‡], Raheem A. Beyah[†]
[†]Communications Assurance and Performance (CAP) group
[‡]National Electric Energy Testing Research and Applications Center (NEETRAC)
School of Electrical and Computer Engineering, Georgia Institute of Technology

***Abstract*—The future electrical grid, i.e., smart grid, will utilize appliance-level control to provide sustainable power usage and flexible energy utilization. However, load trace monitoring for appliance-level control poses privacy concerns with inferring private information. In this paper, we introduce a privacy-preserving and fine-grained power load data analysis mechanism for appliance-level peak-time load balance control in the smart grid. The proposed technique provides rigorous provable privacy and an accuracy guarantee based on distributed differential privacy. We simulate the scheme as privacy modules in the smart meter and the concentrator, and evaluate its performance under a real-world power usage dataset, which validates the efficiency and accuracy of the proposed scheme.**

I. Introduction

The future electrical grid, i.e., smart grid, introduces information and communication technology (e.g., Advanced Metering Infrastructure (AMI)) to the traditional electrical grid to improve the efficiency and reliability of the system. Its development has been actively driven by governments in the United States and Europe. Given the mandatory transition to the new generation smart grid, which has been signed into law, 80% of consumers should be equipped with smart meters by year 2020 in the United States [1]. Neighborhood area network (NAN) based smart grids, as an initial smart grid instance, plays an important role in this transition because of the quick and lightweight communication deployment of the NAN. The NAN, as a bi-directional online communication infrastructure between smart meters and the concentrator, consists of data management systems and monitoring systems to collect metering data and to distribute control information. It realizes the neighborhood-level meter reading collection and control information distribution flow between the utility company and the residents.

Towards sustainable power usage and flexible energy utilization, many different load control policies have been introduced for the smart grid. Among them, *appliance-level load control policy* in *NAN based smart grids* has begun to receive attention. Appliance-level control policy is supported by Non-intrusive Load Monitoring (NILM) technology in smart meters [2], where the load profile is analyzed to deduce the individual energy consumption of appliances. It allows fine-grained power consumption to be profiled in real time, and enables remote diagnostics and controls to increase the performance of the grid infrastructure. When executing the appliance-level control policy, to respond to a rapid power consumption increase in peak time among neighborhoods, the purpose of *peak-time load balancing control* for the smart grid is to temporarily (to allow time to start up a larger generator) or continuously (in the case of limited resources) shut down the appliances which are not in use but connected to the circuit.

However, the usage of fine-grained energy data for appliance-level load control poses privacy concerns. As recent research indicated [3], personal information can be derived from energy consumption data, such as the individuals' behaviors and locations in their houses. For instance, burglars, who eavesdrop on the communication on the wireless link in the NAN, could identify the time to break into the house based on the fact that power consumption drops when the house is vacant. Also, one who colludes with the controller could infer the location of the residents by monitoring the appliances being used. This kind of behavioral inference is known as an NILM attack [2]. In addition to privacy, quality-of-service (QoS) is also a key issue of the smart grid. Performance degradation like response delay or output inaccuracy, which can be introduced by the data manipulation to achieve privacy, must be rigidly quantified for the system reliability and stability.

In this paper, we propose a privacy-preserving fine-grained power usage data analysis mechanism for the appliance-level peak-time load balance control in the smart grid based on distributed differential privacy. The main contributions of the paper are summarized as follows:

- We explore the distributed top-k differential privacy problem to propose a privacy-preserving load analysis mechanism for appliance-level peak-time load balance control.
- We prove that the proposed scheme achieves a rigorous privacy guarantee called 3ε-differential privacy. Also, we show the provable upper bound of the error rate for our scheme.
- We demonstrate an evaluation for our scheme using a real-world dataset. Our results indicate the efficiency and validity of our scheme.

This paper proceeds as follows. We discuss the related work in Section II. In Section III, we describe the necessary background concepts of this work. In Section IV, we show the problem formulation. In Section V, our approach is described in detail. In Section VI, the security and accuracy analyses are discussed theoretically. In Section VII, our scheme's performance is evaluated using a real-world dataset. In Section VIII, the conclusion and the future work are presented.

II. Related Work

Data disclosure in the smart grid is attracting increasing attention from researchers. In particular, secure and privacy-preserving communication and data management in the AMI have been extensively studied. In [3], security and privacy analyses of the Automatic Meter Reading (AMR) technology were presented. As the authors indicated, AMR is susceptible to a neighborhood-level NILM attack because of its lack of basic security mechanisms, such as insecure wireless transmissions and the continuous broadcast of energy traces.

In order to protect the data privacy while guaranteeing the ability to manage the data, two types of complementary privacy-preserving approaches have been proposed: *non-cryptographic approaches* and *cryptographic approaches*. One promising non-cryptographic approach is Battery-based Load hiding (BLH), which utilizes a battery to partially supply the demand load so as to alter the meter reading. Rajagopalan et al. [4] proposed a best effort privacy protection algorithm, which quantified the loss of benefit resulting from the privacy-preserving approach. McLaughlin et al. [5] proposed a non-intrusive load leveling method for BLH and performed a rigorous physical simulation under substantial real-world data. However, these schemes face the vulnerabilities of load peak leakage as revealed in [6]. Accordingly, a stepping-based framework for BLH was proposed by Yang et al. [6], which maximized error between the demand load and external load in load peaks. But, BLH approaches limit the ability of the smart grid to provide appliance-level load control. In the cryptographic approaches category, Deng et al. [7] proposed a secure communication scheme for AMI. Li et al. [8] proposed a secure information aggregation scheme for the smart grid. Rottondi et al. [9] proposed a privacy-preserving load scheduling scheme to prevent the NILM attack. However, these schemes [7]–[9] require the use of homomorphic encryption (HE) [10]–[12] such as Paillier's cryptosystem [12], which is computationally expensive and far from practical. Yan et al. [13] proposed a symmetric encryption (SE) based secure communication scheme for AMI to protect from eavesdropping. However, the approach described in [13] is not provable security and does not support privacy-preserving data management. As illustrated by Table I, our scheme based on differential privacy (DP) is the most full-featured.

III. Background

A. Differential privacy

Differential privacy has become a popular privacy method due to its lightweight implementation and rigorous provable security. Differential privacy was proposed by Dwork et al. [14] in 2006, which makes no trust assumption about the adversary.

Definition 1: (ε-differential privacy [14]) A randomized algorithm A gives ε-differential privacy if for all datasets D_1 and D_2 differing on at most one row, and for all $S \subseteq Range(A)$,

$$Pr\{A(D_1) \subseteq S\} \leq e^{\varepsilon} \times Pr\{A(D_2) \subseteq S\},$$

where ε is the privacy budget of the randomized algorithm A.

In the definition of differential privacy, the data sets D_1 and D_2, which the randomized algorithm targets, differ on at most one row. In other words, if the removal or addition of a single user's data does not substantially affect the result, there is no risk for users to join and answer the query. The privacy budget ε is the parameter to measure the privacy level of the randomized algorithm. The choice of ε is a tradeoff between the privacy and the accuracy of the output.

Definition 2: (Sensitivity [14]) For a function $f : D \to R^k$, the sensitivity of f is

$$\Delta f = \max_{D_1, D_2} \|f(D_1) - f(D_2)\|_1,$$

where D_1 and D_2 differ on at most one row, and $D_1, D_2 \in D$.

Sensitivity measures the outputs' change in the function $f()$, when the targeted data set changes.

A significant ε-differential privacy mechanism, which was introduced by Dwork et al., is Laplace noise on counting query [14], i.e., $A(X) = f(X) + Lap(\frac{\Delta f}{\varepsilon})$. In Laplace noise on counting query, $f()$ is a counting query on the data set X, and $Lap()$ is the Laplace distribution with standard deviation $\frac{\sqrt{2}\Delta f}{\varepsilon}$ to scale the counting query result. Laplace noise on counting query is claimed to be ε-differential privacy, because for all D_1 and D_2 differing on at most one row, $\frac{Pr\{A(D_1) \subseteq R\}}{Pr\{A(D_2) \subseteq R\}} = e^{\frac{-|f(D_1) - f(D_2)|\varepsilon}{\Delta f}}$. As the sensitivity of counting query $|f(D_1) - f(D_2)| \leq \Delta f$, $\frac{Pr\{A(D_1) \subseteq R\}}{Pr\{A(D_2) \subseteq R\}} \leq e^{-\varepsilon}$. Hence, Laplace noise on counting query $f(X) + Lap(\frac{\Delta f}{\varepsilon})$ is ε-differential privacy.

B. Non-intrusive Load Monitoring (NILM)

NILM, which was initially described by Hart et al. [2], is a process to determine each appliance's individual energy

TABLE I
A comparison of Related Works with our scheme.

	Cryptographic Approach					Non-cryptographic Approach		
	Our work	DY [7]	LLL [8]	RV [9]	YQS [13]	RSM [4]	MMA [5]	YLQ [6]
Category	DP	HE	HE	HE	SE	BLH	BLH	BLH
High-Efficiency	✓	✗	✗	✗	✓	✓	✓	✓
Provable Security	✓	✗	✗	✓	✗	✗	✗	✗
Fine-grained control	✓	✓	✓	✓	✓	✗	✗	✗

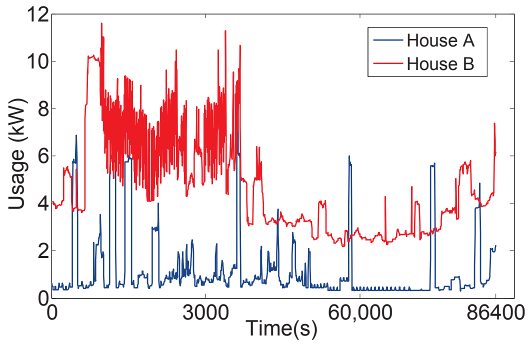

Fig. 1. A one-day load profile for two houses from UMASS Smart* dataset [15].

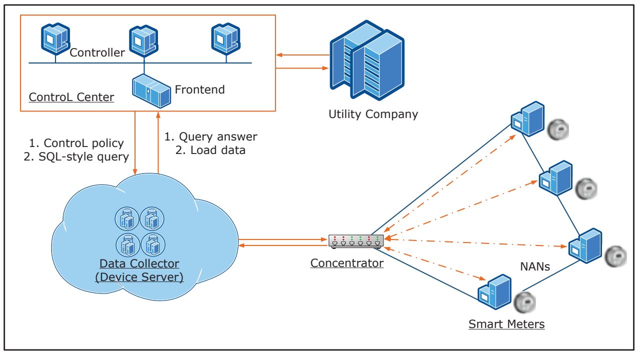

Fig. 2. Overview of the NAN based smart grid system.

consumption by analyzing changes of the load profile from the interface of the smart meter. NILM is considered a lightweight alternative to attaching individual monitors on each appliance. It detects an appliance's activities such as ON/OFF events for appliance-level load control and appliance management. However, NILM technology can also be used by the adversary to deduce the residents' behavior in the house.

A one-day load profile for two houses is shown in Figure 1. Even though two houses have distinguishable diurnal patterns, it is easy for an adversary to deduce whether the house is vacant by observing the load profiles. Moreover, consumption trace analysis can be mapped directly to ON/OFF events of identifiable appliances so as to determine the location of the resident in the house [2]. Thus, a privacy-preserving load analysis scheme is needed for the smart grid.

IV. Problem Formulation

In this section, the system model and assumptions are given. Then, we outline the adversary model and design goals.

A. System Model and Assumptions

We consider an instance of a NAN based smart grid system as shown in Figure 2, which is composed of four components including smart meters, a concentrator, a data collector and a control center. In the NAN based smart grid system, the *smart meter* of each house in the neighborhood transmits its data to the concentrator through the NAN, which has a star topology. Then, the *concentrator* forwards the data to the data collector. The purpose of the concentrator is to efficiently forward the data from the smart meters to the data collector. The *data collector* collects and stores the data from the smart meter, and also distributes the query from the control center to the smart meters of each house. The *control center* generates control policy based on the query answers returned from the data collector. Our scheme introduces privacy modules in both the smart meters and the concentrator as a blackbox to provide privacy without modifying existing programs. We further assume that the public key pairs are shared between the smart meters and the control center. The smart meters utilize the control center's public key K_p for secure communication.

The control policy that we utilize in this work is *appliance-level peak-time load balance control*, i.e., the policy is to generate an appliance-level control to respond to a rapid power consumption increase among neighborhoods during peak time. In a NAN based smart grid system, the control center balances the load in peak time by shutting down the appliances temporarily (to allow time to start up a larger generator) or continuously (in the case of limited resources), where the appliances are not in use but connected to the circuit. Hence, the corresponding meter reading analysis is a real-time query $Q_t =< ID, t, q >$, where ID is the query ID, t is the query timestamp and q is the SQL-style query request, such as 'SELECT k appliances WHERE they are not in use but connected to the circuit AND $t = t_p$ IN ORDER of power consumption', where t_p is the peak-time timestamp.

For the above smart grid architecture and control policy, without loss of generality, we assume that the NAN (with a star topology) has N smart meters distributed in the neighborhood. Each smart meter in a house samples the power consumption of appliances with the sample rate r_s. Also, each smart meter has similar storage capacity and computation power. Each concentrator and data collector are powerful and resourceful enough to store data and process query requests. Also, the controller has full access to the data collector.

B. Adversary Model and Design Goals

As recent works in [4]–[6], [9], [13], we consider a similar adversary model as follows: (1) *Honest-but-curious controller* where the controller follows the designated protocol specification honestly while it is curious to analyze data in the data collector or the concentrator so as to learn additional information besides those obtained for control policy generation. (2) *Honest-but-curious user* where the user acts in an 'honest' fashion to answer the query for load scheduling, but in a 'curious' fashion to obtain other users' load profiles by eavesdropping the communications or colluding with the untrusted controller. (3) *Malicious eavesdropper* where the eavesdropper tries to obtain the load profile on the smart grid.

To address the adversary models above, a privacy-preserving and fine-grained data analysis scheme for the smart grid is proposed. Our scheme achieves security and performance guarantees as follows:

- *Provable Privacy*: The untrusted controller does not learn additional information of the residents' load profiles from data collector except for those for control policy. Moreover, other components of the smart grid or the channel eavesdropper are unable to learn the residents' load.
- *Accuracy Guarantee*: The accuracy of the query results is quantified and bounded. In other words, the performance degradation, which is introduced by the data manipulation to achieve privacy, is limited.
- *Performance*: The above goals for privacy and accuracy guarantees should be power efficient with low response time on the smart grid system.

V. System for Peak-Time Load Balance

Our proposed scheme has three steps. First, when the concentrator obtains the query request Q_t from the control center, the concentrator fuzzes the parameters of the query request, and then distributes the new query request Q'_t to the smart meters of each house. Second, the smart meter of each house i answers the query Q'_t, and then encrypts the query answer with the controller's public key K_p. With the encrypted query answers from each smart meter through a secure channel such as TLS, the concentrator adds noise into the set of the encrypted query answers, then returns k answers among them by uniformly sampling. Finally, the controller decrypts the query answers with its private key K_s. The challenge of the scheme is to add the noise blindly in the concentrator while guaranteeing the accuracy of the query answer. The details of the scheme are as follows:

Query Initialization. A controller formulates a SQL-style query $Q_t =< ID, t, q >$ for peak-time load balancing, where ID is the query ID, t is the query timestamp and q is the SQL-style query request, i.e., $q =$ 'SELECT k appliances IN ORDER FROM history log WHERE $p > P_K$ and $t \in [t_{pl}, t_{ph}]$'. In query request q, p is the power consumption of an appliance in idle mode, P_K is the estimated threshold power consumption to select the top-k appliances with the largest consumption, and t_{pl} and t_{ph} indicate the timestamp range of peak times. Then the controller transmits the query to the data collector, which forwards the query to the concentrator.

Query Transformation. Once the concentrator receives the query request Q_t from the controller. Its privacy module transforms the parameters of the query request to add noise, i.e., the transformed query $Q'_t =< ID, t, q' >$, where $q' =$'SELECT appliances FROM history log WHERE $p > P_K$ and $t \in [t_{pl} - m, t_{ph}]$', where m is the noise added by the concentrator, and $m > \frac{r_s \ln(e+\varepsilon^2 k OPT_e)}{\varepsilon}$, where r_s is the sample rate of the data collector, k is the number of the returned query answers, ε is the privacy budget and OPT_e is the estimated sum of top-k appliances power usage based on historical information. Then, the transformed query request Q'_t is forwarded to each of the smart meters in the NAN.

Query Response. After receiving the query request, each smart meter of a house searches the history log based on the query request. When the smart meter of house i obtains the set of the query answers $QA = \{< a_i, t_i, p_{a_i} >\}$, where a_i, p_{a_i}, t_i are the UID of the appliance, the power consumption and the timestamp respectively, it adds noise then encrypts the set of answers before sending them back to the concentrator. The format of the data message sent from the smart meter, S, to the concentrator, P, is as follows:

$$S \rightarrow P : H_{K_p}(i||a_i)||E_{K_p}(p_{a_i} + n_i)||E_{K_p}(t_i) \qquad (1)$$

where E_{k_p} is an asymmetric encryption scheme (such as RSA), H_{k_p} is a hash function (such as SHA-2), and n_i is the power consumption noise, $n_i = LAP(\frac{\Delta f_s}{\varepsilon})$, $\Delta f_s < 1$.

```
Algorithm $Keygen(l)$
1. return $(K_p, K_s) \leftarrow RSA.Keygen(l)$

Algorithm $InitQuery(q)$
1. Relax the original query $q$'s time range from $[t_{pl}, t_{ph}]$ to $[t_{pl} - m, t_{ph}]$.
2. return $q'$

Algorithm $ProctEnc(q')$
1. search the history log to obtain the answers $QA$ of the query $q'$, $QA = \{< a_i, t_i, p_{a_i} >\}$
2. add the noise $n_i$ in the power consumption to generate a fuzzy query answer $QA'$, i.e., $p_{a_i} + n_i$.
3. return $RSA.Enc(K_p, QA')$

Algorithm $InitAnswer(C_{QA})$
1. add $c_i$ noise query answers based on the frequency $f_i$ of each appliance appearing (i.e., $H_{K_p}(i||a_i)$)
2. uniformly sample $k$ distinct items $R$ from the set of the query responses including the noise query answers.
3. return $R$

Algorithm $Dec(R)$
1. return $RSA.Dec(K_s, R)$
```

Fig. 3. The sequential scheme of algorithms.

Response Process. To guarantee the differential privacy, the concentrator processes the query responses from the smart meters of each house under the following two policies:

- adding c_i noise query answers based on the frequency f_i of each appliance a_i appearing, and $c_i = ne^{\varepsilon f_i} - f_i$, where n is the size of query answers set, c_i is the number of noise query answers, ε is the privacy budget for differential privacy and f_i is the frequency of the appliance a_i appearing in the set of the query answers, i.e., $f_i = \frac{\#\ of\ the\ pattern\ `H_{K_p}(i||a_i)'}{n}$.
- uniformly sampling k distinct items from the set of the query responses including the noise query answers.

Answer Response. The concentrator returns the k distinct encrypted items to the controller through the data collector. Then, the controller decrypts the message using its private key K_s, and obtains the appliances under idle mode which have the top-k largest power consumptions in peak time. Then, the controller generates the peak-time load balance control policy, which shuts down the appliances that are in idle mode.

VI. Security analysis and performance

In this section, we present the syntax of our scheme, and describe the provable privacy and the upper bound of the error rate theoretically.

A. Syntax

Definition 3: A privacy-preserving and fine-grained power load data analysis mechanism for the appliance-level peak-time load balance control consists of a tuple ($Keygen$, $InitQuery$, $ProctEnc$, $InitAnswer$, Dec) as follows:

- ***Key generation:*** $(K_p, K_s) \leftarrow Keygen(l)$. Keygen runs at the controller side, which generates the public-secret key pair (K_p, K_s) for encryption.
- ***Initialize query:*** $q' \leftarrow InitQuery(q)$. InitQuery runs on the concentrator side, which transforms the query request q from the controller to a new query request q'.
- ***Response:*** $C_{QA} \leftarrow ProctEnc(q')$. ProctEnc runs on the smart meter of each house, which answers the query q' then encrypts the query answers.
- ***Initialize answer:*** $R \leftarrow InitAnswer(C_{QA})$. InitAnswer runs at the concentrator side to output k encrypted query answers R.
- ***Decryption:*** $M \leftarrow Dec(R)$. Dec runs at the controller side to decrypt then obtain the query answer M.

The algorithms are shown in Figure 3.

B. Privacy Analysis

Theorem 1: (compositionality [14]) The sequential scheme of randomized algorithms $\{A_i\}$, each giving $\{\varepsilon_i\}$-differential privacy respectively, gives $(\sum_i \varepsilon_i)$-differential privacy.

Theorem 2: The scheme we proposed gives 3ε-differential privacy.

Proof: In *ProctEnc()*, as the power consumption noise n_i is added as the Laplace noise, i.e., $n_i = LAP(\frac{\Delta f_s}{\varepsilon})$, the algorithm *ProctEnc()* is ε-differential privacy.

In *InitAnswer()* (a.k.a., $IA()$), considering $f_c(a_i) < 1$, where $f_c(a_i)$ is the chosen frequency of the appliance a_i , hence the sensitivity of the chosen frequency $\Delta f_c(a_i) < 1$. With the noises $c(a_i) = ne^{\varepsilon f_c(a_i)} - f_c(a_i)$ added for the appliance a_i and uniformly sample, the sampled probability of the appliance a_i is $e^{\varepsilon f_c(a_i)}$. For two data sets D_1 and D_2 differing on at most one row,

$$\begin{aligned} \frac{Pr(IA(D_1))}{Pr(IA(D_2))} &= \frac{e^{\varepsilon(f_c(D_1,a_i)-f_c(D_2,a_i))}}{e^{-\varepsilon(f_c(D_1,a_i)-f_c(D_2,a_i))}} && (2) \\ &= \frac{e^{\varepsilon \Delta f_c(a_i)}}{e^{-\varepsilon \Delta f_c(a_i)}} && (3) \\ &= e^{2\varepsilon \Delta f_c(a_i)} \leq e^{2\varepsilon} && (4) \\ \therefore Pr(IA(D_1)) &\leq e^{2\varepsilon} Pr(IA(D_2)) && (5) \end{aligned}$$

Hence, the algorithm *InitAnswer()* is 2ε-differential privacy.

By the use of Theorem 1, the scheme we proposed gives 3ε-differential privacy. ■

C. Accuracy Analysis

Definition 4: The error rate of the query results in our appliance-level control scheme is defined as follows:

$$d = \frac{OPT - \sum_{i=1}^{k} p(a_i)}{OPT} \qquad (6)$$

where $p(a_i)$ is the power consumption of the appliance a_i in the top-k query results, and OPT is the real sum of top-k appliances' power usage.

Theorem 3: The scheme we proposed has the upper bound of the error rate as $\frac{3\ln(e+\varepsilon^2 kOPT)}{\varepsilon OPT}$, where OPT is the real sum of top-k appliances' power usage.

Proof: Assume $S_{2t} : \{a_i : A(a_i) > OPT - 2t\}$, where $A()$ is the sequential scheme we proposed.

$$\begin{aligned} \therefore\ E[A(a_i)] &= (OPT - 2t)(1 - A(S_{2t})) && (7) \\ \because\ A(S_{2t}) &\leq \frac{A(S_{2t})}{A(S_t)} \leq \frac{e^{-\varepsilon t}}{\mu(S_t)} && (8) \\ \because m &> \frac{r_s \ln(e + \varepsilon^2 kOPT_e)}{\varepsilon} && (9) \\ &> \frac{r_s \ln \frac{OPT}{t\mu(S_t)}}{\varepsilon} && (10) \\ \therefore\ 1 - A(S_{2t}) &\geqslant 1 - \frac{e^{-\varepsilon t}}{\mu(S_t)} > 1 - \frac{m}{r_s OPT} && (11) \\ \therefore\ E[A(a_i)] &\geq OPT - \frac{3m}{r_s} && (12) \\ &\geq OPT - \frac{3\ln(e + \varepsilon^2 kOPT)}{\varepsilon} && (13) \\ \therefore\ d &\leq \frac{3\ln(e + \varepsilon^2 kOPT)}{\varepsilon OPT} && (14) \end{aligned}$$

Hence, the upper bound of the error rate in our scheme is $\frac{3\ln(e+\varepsilon^2 kOPT)}{\varepsilon OPT}$. ■

VII. Evaluation

In this section, we present an evaluation of the accuracy and efficiency of our scheme, based on a real-world dataset: UMASS SMART* dataset [15], which included the power usage of about 30 appliances, and the average sample rate of appliance usage is 30 seconds/appliance per house. The simulation is implemented in Python on a PC which had two 3.10 GHz Intel Core i5-2400 processors running the Linux 3.5 kernel. We used pycrypto (a.k.a., Python Cryptography Toolkit) to implement the RSA-OAEP and SHA-2 as instances of the public-key encryption and hash function, respectively. The performance of the scheme is evaluated regarding the tradeoff between the privacy and the accuracy as well as the response delay of the scheme. In particular, we answer the following questions:

- *(Accuracy)* What is the accuracy of the query results of the scheme under different privacy budgets? And how do the error rates under different privacy budgets compare with the theoretical upper bound?
- *(Delay)* How does the response time increase when our scheme is used? That is, how much overhead is incurred by the use of the privacy modules on the concentrator and the smart meter in the smart grid?

A. Accuracy Analysis

To evaluate the accuracy of the scheme, we measure the error rate of our scheme compared with the upper bound we proved theoretically. Figure 4 shows the error rate of the scheme with different privacy budgets ε. Both the theoretical upper bound of the error rate and the error rate in the experiment are presented. Overall, the error rate of the scheme decreases as the number of query results k increases. With the larger privacy budget ε, i.e., $\varepsilon = 0.1$, both the upper bound of the error rate and the experimental error rate are smaller than those with smaller privacy budget, i.e., $\varepsilon = 0.01$. Compared to the upper bound of the error rate under the same privacy budget, the experimental error rate is much lower than the theoretical one. Moreover, when the privacy budget is small, the difference between the upper bound of the error rate and the experimental error rate becomes larger. Overall, the theoretical upper bound of the error rate ranges from 15% - 40% given the stated privacy budgets. However, the observed error rates based on the experiments are less than 14% when $\varepsilon = 0.01$ and less than 7% when $\varepsilon = 0.1$.

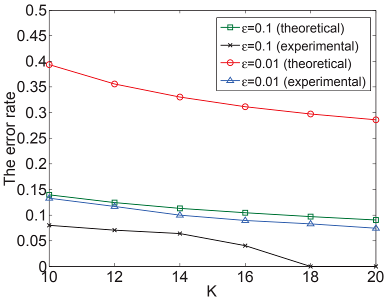

Fig. 4. The error rate of our scheme with different privacy budgets.

B. Delay Analysis

To evaluate the Delay of our scheme, we measured the response time of our scheme under the real-world dataset compared with the original scheme without any security and privacy protection. Figure 5 presents the response time of our scheme with different privacy budgets (i.e., $\varepsilon = 0.1$ and 0.01). To indicate the performance degradation, the response time of our scheme is compared with that without any security mechanism. As the number of query results k increases, the response time of the scheme increases. Also, the smaller privacy budget introduces a larger performance degradation, i.e., when the privacy budget $\varepsilon = 0.01$, the response time becomes larger than that with a smaller privacy budget. In our privacy-preserving scheme with privacy budget $\varepsilon = 0.1$, the increase in the response time is below $0.4s$, which is about 105% of that without any security mechanism.

VIII. Conclusion and Future Work

In this paper, we present a privacy-preserving fine-grained power usage data analysis mechanism for appliance-level peak-time load balance control in a NAN based smart grid. Our scheme provides provable privacy and accuracy guarantees. The scheme we proposed is based on distributed differential privacy to protect residents from the NILM attack. Through the evaluation based on a real-world dataset, we showed that our scheme provided privacy and accuracy guarantees while achieving good performance. In our future work, we will consider additional metrics for peak-time load balance control, such as the fairness of the appliances being chosen to shut off and the total power usage of the household. Also, we will implement a prototype of the scheme in a testbed to evaluate its performance.

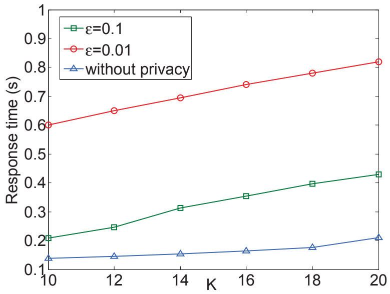

Fig. 5. The response time with different privacy budgets.

References

[1] M. Jawurek, F. Kerschbaum, G. Danezis. SoK: Privacy Technologies for Smart Grids - A Survey of Options. In 2012 Microsoft Technical Report.

[2] G. W. Hart. Nonintrusive appliance load monitoring. In Proceedings of the IEEE, 1992, 80(12): 1870-1891.

[3] I. Rouf, H. Mustafa, M. Xu, et al. Neighborhood watch: security and privacy analysis of automatic meter reading systems. In Proceedings of the 2012 ACM CCS: 462-473.

[4] R. Rajagopalan, L. Sankar, S. Mohajer, et al. Smart meter privacy: A utility-privacy framework. In Proceedings of the 2011 IEEE SmartGridComm: 190-195.

[5] S. McLaughlin, P. McDaniel, W. Aiello. Protecting consumer privacy from electric load monitoring. In Proceedings of the 2011 ACM CCS: 87-98.

[6] W. Yang, N. Li, Y. Qi, et al. Minimizing private data disclosures in the smart grid. In Proceedings of the 2012 ACM CCS: 415-427.

[7] P. Deng, L. Yang. A secure and privacy-preserving communication scheme for Advanced Metering Infrastructure. In Proceedings of the 2012 IEEE ISGT: 1-5.

[8] F. Li, B. Luo, P. Liu. Secure information aggregation for smart grids using homomorphic encryption. In Proceedings of the 2010 IEEE SmartGridComm: 327-332.

[9] C. Rottondi, G. Verticale. Privacy-friendly appliance load scheduling in smart grids. In Proceedings of the 2013 IEEE SmartGridComm: 420-425.

[10] M. Dijk, C. Gentry, S. Halevi, et al. Fully Homomorphic Encryption over the Integers. In Advances in Cryptology-Eurocrypt 2010: 24-43.

[11] J. Coron, A. Mandal, D. Naccache, et al. Fully Homomorphic Encryption over the Integers with Shorter Public Keys. In Advances in Cryptology-Crypto 2011: 487-504.

[12] P. Paillier. Public-key Cryptosystems based on Composite Degree Residuosity Classes. In Advances in cryptology-EUROCRYPT 1999: 223-238.

[13] Y. Yan, Y. Qian, H. Sharif. A secure and reliable in-network collaborative communication scheme for advanced metering infrastructure in smart grid. In Proceedings of the 2011 IEEE WCNC: 909-914.

[14] C. Dwork. Differential privacy. In Automata, languages and programming. Springer Berlin Heidelberg, 2006: 1-12.

[15] S. Barker, A. Mishra, D. Irwin, et al. Smart*: An open data set and tools for enabling research in sustainable homes. In Proceedings of the 2012 SustKDD.