Privacy Preserving Smart Meter Streaming against Information Leakage of Appliance Status

Yuan Hong, *Member, IEEE,* Wen Ming Liu, and Lingyu Wang, *Member, IEEE*

The smart grid frequently collects consumers' fine-grained power usage data through smart meters to facilitate various applications such as billing, load monitoring, regional statistics, and demand response. However, the smart meter reading streams may also pose severe privacy threats to the consumers by leaking their appliances' ON/OFF status. In this paper, we first quantitatively measure the information leakage w.r.t. specific appliances' status from any reading stream, and define a novel privacy notion to bound such information leakage. In addition, we propose a privacy preserving streaming algorithm with different options to effectively convert readings and promptly stream safe readings in different fashions. The output time series readings satisfy our privacy notion while guaranteeing excellent utility, such as extremely low aggregation errors and billing errors. Finally, we experimentally validate the effectiveness and efficiency of our approach using real datasets.

***Index Terms*—Smart Metering, Privacy, Anonymity, Utility**

I. Introduction

The smart grid integrates sensors and communication networks into the existing power grid to ubiquitously collect data from the grid for operational intelligence [14]. As a critical component in such an infrastructure, smart meters frequently transmit fine-grained readings to the electric utility, e.g., a reading every 15 minutes [27]. Such reading streams greatly benefit the utilities (e.g., load balancing) as well as the energy consumers (e.g., optimizing electricity usage) [13]. However, some recent studies show that such features may also lead to serious breaches of consumers' privacy [4], [7]. The fine-grained meter readings could potentially reveal the consumers' personal daily behavior or habits, e.g., cooking time (by the stove or microwave), and frequency of going to the bathroom at night (by the light switched on).

To prevent adversaries from compromising energy consumers' personal privacy, three major categories of privacy preserving techniques were proposed. First, some existing approaches (e.g., [4]) inject tolerable noise into the original or aggregated meter readings. However, they trade off some output utility for desired privacy and may not be able to ensure high aggregation and billing accuracy due to the random noise. Second, some approaches (e.g., [30]) encrypt the meter readings with cryptographic primitives and only report the temporally or geographically aggregated data for specific applications (e.g., billing [12], regional statistics [7]). However, without reporting the fine-grained readings, the output cannot support many real world smart grid applications (e.g., load monitoring [15]). Finally, some approaches (e.g., [37]) attach batteries for households to mask the meter readings. However, they may require expensive devices or facilities to support the scheme and thus result in high cost for both implementation and maintenance.

More importantly, the privacy models in most of the existing solutions (e.g., [30], [7], [37]) only consider all the *fine-grained meter readings* (viz. a series of numbers) as sensitive data and simply aim to anonymize such "numbers". To the best of our knowledge, the privacy risks in terms of "appliances' ON/OFF status at different times" (which directly reflects the privacy concerns of energy consumers) has not been formally defined and quantified in literature. Specifically, the following are unclear in most of the prior privacy models: (1) which reading is sensitive and vulnerable? (2) how much information related to the appliance status can be leaked from the readings? and (3) what kind of background knowledge can be utilized to identify the appliance status from the reading streams?

To tackle such issues, in this paper, we investigate the *privacy risks* by linking the meter readings to appliances' ON/OFF status at different times, and formally define a privacy notion (denoted as (ϵ, δ^m)-Uncertainty) to quantify and bound such threats of information leakage in any reading stream. Different from most of the prior work, we propose an efficient privacy preserving algorithm to stream output readings *without any aggregation* while guaranteeing rigorous privacy and excellent utility. Therefore, the outputs can support most smart metering services, e.g., billing [12], regional statistics [7], and load monitoring [15], and such outputs can also be fed into the aggregation-based solutions when necessary.

Motivating Example. Table I presents a set of sample time series readings, and Table II shows the electric appliances and the labeled consumption rates in watts for a household [1].

In real world, the adversaries can obtain the readings and possess the background knowledge of common appliances' consumption rates. From the first reading 0.08kWh (in 0.1 hour), he/she can learn the overall consumption rate as 800watts. Then, with the background knowledge in Table II, the adversary can learn that exactly one of the following possibilities may occur: (1) microwave (800watts) is ON, (2) PC, light, vacuum cleaner, TV and stereo system (800watts in total) are ON, or (3) other combinations of appliances with overall consumption rate 800watts. Moreover, looking at the reading time 6:30pm, he/she can infer that microwave is highly

Y. Hong is with the Department of Computer Science, Illinois Institute of Technology, 10 W 31st Street, Chicago, IL 60614. E-Mail: yuan.hong@iit.edu.

W. M. Liu and L. Wang are with the Concordia Institute for Information Systems Engineering, Concordia University, Montreal, QC H3G 1M8 Canada. E-Mail: {l_wenmin, wang}@ciise.concordia.ca.

Manuscript received August 23, 2016; revised March 14, 2017; accepted April 25, 2017

TABLE I
READING STREAM (FREQUENCY: 1 READING PER 6 MINUTES)

Time (PM)	6:30	...	7:30	...	8:00	8:06	8:12	...	8:42
Reading (kWh)	0.08	...	0.13	...	0.05	0.15	0.17	...	0.2
Consumption Rate (watts)	800	...	1300	...	500	1500	1700	...	2000

TABLE II
APPLIANCES' CONSUMPTION RATES

Light 1 (60watts)	Light 2 (100watts)
Vacuum Cleaner (100watts)	Waterpik (100watts)
Stereo System (100watts)	PC (200watts)
TV (300watts)	Microwave (800watts)
Washer (1000watts)	Dishwasher (1200watts)
Dryer (1500watts)	...

likely to be ON due to the cooking time.

Second, at 7:30pm, consumption rate 1300watts can be learned. Thus, dishwasher is likely to be ON at 7:30pm (due to 1300watts). In reality, a sequential usage pattern of two appliances "microwave → dishwasher" (washing the dishes after dinner) could help the adversary confirm that dishwasher is ON at 7:30pm. Similarly, TV and stereo system might be ON at 8:00pm due to the TV's temporal usage pattern, as well as the correlation between TV and stereo system to be ON simultaneously (can be known as background knowledge).

Third, besides the consumption rate/time, some appliances also have their unique signatures on the length of usage. Then, adversaries can also utilize it to learn the status of different appliances. For instance, washer is likely to be ON at 8:06pm (due to 1500watts) and it is also likely to be ON at 8:12pm (due to 1700watts). Then, the adversary can confirm that the washer is extremely likely to ON at both times due to a common background knowledge that washer runs continuously for at least 30 minutes in general. ⊡

In this paper, we will investigate a set of possible information leakage to breach the consumers' privacy from smart meter reading streams, and define a novel privacy notion to quantify and bound such risks. Then, the primary contributions of this paper are summarized as below:

- We define a novel privacy notion to quantify and bound the privacy leakage w.r.t. the readings' actual implications on specific appliances' ON/OFF status.
- We propose an efficient privacy preserving streaming algorithm with different options to effectively convert readings and promptly stream safe readings with excellent output utility, e.g., negligible aggregation/billing errors.
- We conduct experiments to evaluate the performance of our streaming algorithm on real datasets and provide case studies for real-life households.

The rest of the paper is organized as follows. Section II first reviews the related work. Section III formally defines some models. In Section IV, we present our privacy preserving streaming algorithm. Then, we give analysis on privacy, complexity and implementation in Section V. Section VI presents the experimental results, and Section VII summarizes some limitations and challenges. Finally, we draw the concluding remarks and discuss the future work in Section VIII.

II. RELATED WORK

In the past decade, various privacy models were proposed to bound the privacy risks of identifying any individual or associating any individual with the sensitive values in many different datasets, such as k-anonymity [36] for anonymizing tabular data, and ρ-uncertainty [6] for preventing inferences in transaction data. Furthermore, differential privacy [11] has been extended to tackle the privacy concerns in many different contexts based on randomizations, such as recommender [23], search queries [21], [16], [18] and smart metering [4].

Recently, privacy-preserving techniques have been developed for mitigating privacy risks in fine-grained meter readings [4], [29], [31]. For instance, Rottondi et al. [29] presented a secure communication protocol which allows utilities to securely aggregate smart meter readings. Ács and Castelluccia [4] proposed a differentially private scheme that enables smart meters to periodically report data to power suppliers and compute aggregated statistics with rigorous privacy guarantee. In addition, Shi et al. [33] proposed a differentially private randomization based aggregation of distributed time series data (e.g., readings collected from multiple smart meters) with differential privacy guarantee. Different from the noise based data perturbation (e.g., state-dependent perturbation [38]), our privacy preserving streaming algorithm does not report probabilistic results, which can reduce errors and variance in general. In the context of smart metering aggregation and perturbation, more recently, Savi et al. [32] quantitatively analyzed a tradeoff between the aggregation set size, the precision on the aggregated measurements, and the privacy. Finally, renewable energy sources (e.g., battery) can be utilized to mask the original meter readings of households as well [37].

Non-Intrusive Load Monitoring (NILM). In some NILM algorithms [8], [9], [26], [25], privacy concerns have been identified since the NILM algorithms estimate the specific appliances' energy consumption at different times in households, e.g., [8], [9], [26]. However, such NILM algorithms cannot provide an upper bound for the probabilities of disaggregation (this was also indicated in [10]). Then, the privacy enhancing techniques extended from NILM algorithms (e.g., [26]) cannot quantify and bound the risks with theoretical guarantee of privacy. Instead, in this paper, the privacy leaking risks can be quantified, and can be bounded with our defined privacy notion to provide theoretical guarantee of privacy. Notice that, among the NILM studies, Dong et al. [10] have learned the upper bounds on the probabilities of distinguishing between scenarios of appliance usage based on the energy consumption distribution (which was missing in most of the NILM algorithms). However, in such work, privacy notions are not defined to quantify the privacy risks, and there does not exist a privacy preserving algorithm to output safe readings based on the derived upper bounds either.

III. Models

We now illustrate the information leakage, privacy notions, and three utility measures. Table III lists some frequently used notations.

TABLE III
Frequently Used Notations

A, E	appliance set, a subset of A
$a_x, \|a_x\|$	an appliance, a_x's labeled consumption rate
$\|A\|$	number of appliances in A
$h(\cdot)$	consumption rate function
ϵ, δ, m	privacy parameters
r, ω, ϕ	reading, consumption rate, reading frequency
$c(\cdot)$	candidate appliance set function
$\|c(\omega)\|$	size of the candidate appliance set $c(\omega)$
$\vec{R_{in}}, \vec{R_{out}}$	input and output reading streams
K	number of readings in a stream

A. Preliminary Models

We denote a smart meter's associated *appliance set* as $A = \{a_1, \dots, a_{|A|}\}$, where $|A|$ is the number of appliances (how smart meter populates and maintains its appliance set A is discussed in Section V-C). We use $|a_1|, \dots, |a_{|A|}|$ to represent their labeled consumption rates. In addition, we define *reading frequency* as ϕ: the time interval between two consecutive readings (e.g., 15 minutes). The readings can be converted into consumption rates, and vice-versa.

Given an appliance set A and the consumption rate of each appliance in A, we first define a function to calculate the overall consumption rate of any subset of A (which is a combination of appliances).

Definition 1 (Consumption Rate Function $h(\cdot)$): Given any subset of an appliance set A: $\forall E \subseteq A$, function $h(\cdot)$ is defined to calculate the overall consumption rate of all the appliances in E: $h(E) = \sum_{\forall a_x \in E} |a_x|$, where $|a_x|$ denotes a_x's consumption rate.

Then, $h(\cdot)$ can be used to calculate the unique consumption rates of all the subsets of A, which are denoted as:

Definition 2 (Candidate Rate Set G): Given the power set 2^A of an appliance set A, the set of unique consumption rates:

$$G = \{h(E) : \forall E \subseteq 2^A\} \tag{1}$$

where E is a subset of A.

As a result, for any consumption rate $\omega \in G$, we can find all the subsets of A whose consumption rate equals ω by traversing 2^A. We consider such process as a function:

Definition 3 (Candidate Appliance Set Function $c(\cdot)$): Given any consumption rate $\omega \in G$, function $c(\cdot)$ is defined as

$$c(\omega) = \{E : E \subseteq 2^A, h(E) = \omega\} \tag{2}$$

B. Privacy Leakage

In this paper, we look at the case that each appliance is either completely "ON" or completely "OFF" between two adjacent readings (which occurs very often in the reading stream due to short time intervals). Indeed, this is the worst case of leaking consumers' privacy since the overall consumption rate in that time interval (e.g., 15 minutes) accurately reflects all the appliances which are "ON".

1) Leakage in a Single Reading

The "ON/OFF" status of any appliance can be possibly leaked from a single reading which includes the consumption amount/rate and consumption time.

Consumption Rate. Denoting the size of ω's candidate appliance set $c(\omega)$ as $|c(\omega)|$, we can represent $c(\omega)$ as $\{c(\omega)_1, c(\omega)_2, \dots, c(\omega)_{|c(\omega)|}\}$. Since there are $|c(\omega)|$ combinations of appliances that would lead to the consumption rate ω, adversaries can enumerate all the entries in $c(\omega)$ and infer a view for all the possible combinations of appliances $\{c(\omega)_1, c(\omega)_2, \dots, c(\omega)_{|c(\omega)|}\}$. Indeed, in such view, each combination of appliances $\forall y \in [1, |c(\omega)|]$ can have a probability P_y such that $\sum_{y=1}^{|c(\omega)|} P_y = 1$. We can quantify the information leakage in the adversary's view using their Entropy [34]:

$$H(c(\omega)) = -\sum_{y=1}^{|c(\omega)|} (P_y \log P_y) \tag{3}$$

Therefore, the maximum information leakage occurs in case that $P_1 = P_2 = \cdots = P_{|c(\omega)|}$ (maximum entropy). In other words, among all the possible inferences in the adversary's view, $P_1 = P_2 = \cdots = P_{|c(\omega)|}$ would result in (households') maximum privacy leakage (viz. adversary's maximum information gain) from the consumption rate ω. Then, given a reading $r = \omega\phi$, adversaries can have the maximum privacy leaking view which is $P_1 = P_2 = \cdots = P_{|c(\omega)|} = \frac{1}{|c(\omega)|}$ for each of the possible combinations of appliances with overall consumption rate ω.

As a result, the information leakage w.r.t. "any appliance a_x is ON" can be quantified from all the possible combinations (entries in the candidate appliance set $c(\omega)$): $\forall y \in [1, |c(\omega)|]$, if appliance a_x is in the appliance set $c(\omega)_y$, then $\frac{1}{|c(\omega)|}$ is added into the overall information leakage. Thus, given the consumption rate ω, the information leakage w.r.t. "appliance a_x is ON" can be represented as:

$$\mathcal{I}[\omega \to a_x] = \sum_{y=1}^{|c(\omega)|} \frac{I_{xy}}{|c(\omega)|} \in [0, 1] \tag{4}$$

where $\forall y \in [1, |c(\omega)|], I_{xy} \in \{0, 1\}$ and if $a_x \in c(\omega)_y$ then $I_{xy} = 1$; otherwise $I_{xy} = 0$.

Consumption Time. Besides the consumption rate, since many appliances may have temporal usage patterns, the timestamp of a reading can also be exploited by adversaries to further identify appliances' "ON/OFF" status at that time. For instance, microwave might be "ON" with a very high probability at 6pm, and TV is very likely to be "ON" between 7-9pm. Note that the temporal usage patterns can be readily estimated by the adversaries via exterior knowledge, e.g., the power usage of most households, weather conditions, and other public resources. Then, we also use the [0,1] range to measure such information leakage where 0 represents "impossible to be ON" whereas 1 means "impossible to be OFF" (note that it refers to the likelihood of using a certain appliance at a specific time by most households, which can be simply estimated by everyone). Then, the adversary can envision a view of the information leakage of all the appliances' status (based on how likely each appliance is ON at different time). For instance, $\mathcal{I}[3am \rightarrow Microwave] = 0.02$, $\mathcal{I}[8pm \rightarrow TV] = 0.3$. Thus, given a consumption time t, the information leakage w.r.t. "appliance a_x is ON" can be represented as:

$$\mathcal{I}[t \rightarrow a_x] \in [0, 1] \tag{5}$$

Information Leakage Quantification. We then measure the information leakage w.r.t. "an appliance is ON" from a reading, which discloses to the adversaries the overall consumption rate ω and time t.

Definition 4 (Information Leakage of Appliance Status): Given a reading r (consumption rate ω) at time t, we merge the information leakage w.r.t. "appliance a_x is ON" from the consumption rate ω and time t using their union:

$$\begin{aligned}\mathcal{I}[(\omega, t) \rightarrow a_x] =& \mathcal{I}[(\omega \rightarrow a_x) \cup (t \rightarrow a_x)] \\ =& \mathcal{I}[\omega \rightarrow a_x] + \mathcal{I}[t \rightarrow a_x] \\ & - \mathcal{I}[(\omega \rightarrow a_x) \cap (t \rightarrow a_x)]\end{aligned} \tag{6}$$

Notice that both the consumption rate ω and time t leak private information regarding a_x's status. Nevertheless, in our privacy model, the joint information leakage $\mathcal{I}[(\omega, t) \rightarrow a_x]$ should be bounded in any case, and $\mathcal{I}[(\omega, t) \rightarrow a_x]$ achieves its maximum value when the two correlated information leakage from the consumption rate ω and time t individually leak information – two fixed amounts of leakage from ω and t have the least overlap, and thus make the joint leakage (the union) achieve the maximum value. Then, we only need to bound $\max\{\mathcal{I}[(\omega, t) \rightarrow a_x]\}$ in our privacy notion:

$$\begin{aligned}\max\{\mathcal{I}[(\omega, t) \rightarrow a_x]\} =& \mathcal{I}[\omega \rightarrow a_x] + \mathcal{I}[t \rightarrow a_x] \\ & - \mathcal{I}[\omega \rightarrow a_x] \cdot \mathcal{I}[t \rightarrow a_x]\end{aligned} \tag{7}$$

where ω and t individually leak privacy w.r.t. "a_x is ON".

In summary, the information leakage $\omega \rightarrow a_x$ to the adversaries based on the observations of the consumption rate ω is similar to the information leakage in the datasets applied with k-anonymity [36]. Each possible combination of the appliances in the candidate appliance set has an equal risk to be linked to the overall consumption rate ω, and thus the information leakage $\mathcal{I}[\omega \rightarrow a_x]$ can be obtained. Furthermore, the information leakage from the consumption time in the reading $\mathcal{I}[t \rightarrow a_x]$ can increase the joint information leakage of each appliance's ON status via the union of two leakages.

Note that the appliances are not necessarily unique in A (e.g., multiple lights) and an appliance may have more than one consumption rate for different running modes (e.g., Microwave). For the former case, we consider such appliances as different appliances in A to calculate the candidate rate set and the candidate appliance set. For the latter case, we consider such appliance as a single appliance (with multiple possible consumption rates) in A to calculate the candidate rate set and candidate appliance set as well as measure the information leakage.

2) Leakage in a Reading Stream

First, given a reading stream $\vec{R}$, the sequential patterns [35] of appliances in multiple readings can also help adversaries identify the usage of appliances. A typical sequential pattern can be stated as "if an appliance a_x is ON at time t, it is likely to be ON at time $t{+}1, \dots, t{+}N$". For instance, the information leakage w.r.t. "dishwasher is ON" is 0.5 at 7pm, and also 0.5 at 7:05pm and 7:10pm, respectively. Since a dishwasher typically runs for an hour (its sequential pattern) without interruption, its information leakage may increase from 0.5 to 0.8 by correlating the information leakage in multiple readings.

Second, another type of sequential patterns result from the correlation between the usage of multiple appliances. For instance, if a washer runs at time t, a dryer will frequently run at a later time; if a microwave runs at time t, a dishwasher will be very likely to run at a later time.

Third, many appliances not only have usage patterns within a sequence of readings (as described above), but also frequently run at the same time, e.g., TV and stereo system.

In sum, the above usage patterns (for one or multiple appliances) could correlate information leakage from multiple readings and appliances to pose additional privacy risks.

3) Summary of Information Leakages

As described above, adversaries may easily obtain any of the following common background knowledge:

- The reading frequency ϕ.
- A list of common appliances, their consumption rates and temporal usage patterns (e.g., TV frequently runs at 8pm, microwave rarely runs at 3am).
- Single appliances' sequential usage patterns (e.g., dishwasher continuously runs for one hour).
- The usage patterns of multiple appliances (in sequence), e.g., washer runs first and then dryer runs.
- The usage patterns of multiple appliances (at the same time), e.g., TV and stereo system.

Then, we formally illustrate three kinds of information leakages based on the above background knowledge:

Information Leakage (1). For any reading r in a reading stream $\langle \vec{R_{in}}[1], \dots, \vec{R_{in}}[K] \rangle$, its consumption rate ω and time t could leak the information of the appliances' status (per Equation 6, the worst case of information leakage occurs as both ω and t individually leak information with the least overlap). Then, adversaries can learn the status of many appliances as ON at different times with a high $\mathcal{I}[(\omega, t) \rightarrow a_x]$.

Information Leakage (2). Appliances may have sequential usage patterns (e.g., dishwasher, and oven), which occur in some consecutive readings in the stream. Without loss of generality, assuming that appliance a_x has a sequential pattern to run in N consecutive readings, the information leakage w.r.t. "a_x is ON" in consecutive readings can be obtained:

- $\mathcal{I}[(\omega_1, t+1) \rightarrow a_x]$
- $\mathcal{I}[(\omega_2, t+2) \rightarrow a_x]$
- $\dots$
- $\mathcal{I}[(\omega_N, t+N) \rightarrow a_x]$

Then, the information leakage can be higher than any of the above due to their correlation (correlating multiple leakage in sequential readings).

Information Leakage (3). Multiple appliances may frequently run in sequence or simultaneously (e.g., washer and dryer). Specifically, assuming that two appliances a_x and a_y frequently run in sequence, if the following are relatively high:

- $\mathcal{I}[(\omega_1, t+1) \to a_x]$
- $\mathcal{I}[(\omega_N, t+N) \to a_y]$

Then, the information leakage w.r.t. each of a_x and a_y's status (at $t+1$ and $t+N$, respectively) can be higher than their original information leakage due to their correlation (correlating two leakages in sequential readings). Similarly, if a_x and a_y frequently run simultaneously and if the following are relatively high:

- $\mathcal{I}[(\omega, t) \to a_x]$
- $\mathcal{I}[(\omega, t) \to a_y]$

Therefore, the information leakage w.r.t. each of a_x and a_y's status at time t can be higher than their original information leakage due to their correlation (correlating two leakages in the same reading). ⊡

C. Privacy Notions

To prevent the information leakage illustrated in Section III-B3, we first define a privacy notion for quantifying and bounding such risks in any single reading as below:

Definition 5 (ϵ-Uncertainty): Given an appliance set A, we say a meter reading r satisfies ϵ-Uncertainty if $\forall a_x \in A$, $\mathcal{I}[(\omega, t) \to a_x] \leq \epsilon$ holds, where $\omega = \frac{r}{\phi}$ and t represent the reading r's consumption rate and consumption time respectively, and $0 \leq \epsilon \leq 1$.

Thus, if any given reading r satisfies ϵ-Uncertainty (or say r is ϵ-Uncertain), the information leakage of all the appliances' ON status is no greater than ϵ. Note that ϵ-Uncertainty can only bound the *Information Leakage (1)* in any single reading. To bound the *Information Leakage (2) and (3)* in a reading stream $\vec{R} = \langle r_1, \dots, r_K \rangle$ (denoting the number of readings in the stream as K), we define the following privacy model:

Definition 6 ((ϵ, δ^m)-Uncertainty): A reading stream $\vec{R}$ satisfies (ϵ, δ^m)-Uncertainty if the following conditions hold:

1) All the readings in $\vec{R}$ are ϵ-Uncertain;
2) The information leakage of any appliance's ON status in any m consecutive readings in $\vec{R}$ is bounded by δ;
3) The information leakage of any combination of appliances' ON status in any m consecutive readings in $\vec{R}$ is bounded by δ.

Note that meeting the three conditions would mitigate the risks of three categories of information leakages. Two additional privacy parameters δ and m are defined: δ limits the information leakage from any usage pattern (of single or multiple appliances) in any m consecutive readings in $\vec{R}$. Smaller ϵ or δ and larger m provides stronger privacy protection.

D. Utility Measures

We define three different utility measures for our approach. We first consider the billing accuracy. In real world, besides the standard energy plan (constant tariff), two other popular plans are widely used (1) time-of-use (TOU) plan, and (2) tiered base (TB) plan [2], [27], [20]. In these two plans, the electricity tariff may vary at different times (e.g., in TOU plan, peak vs. off-peak) or for different tiered consumption amounts (e.g., in TB plan, < 1000 kWh/month vs. ≥ 1000 kWh/month). Thus, the billing error rate is defined as below:

Definition 7 (Billing Error Rate): Given an input reading stream $\vec{R_{in}}$, an equal-length output reading stream $\vec{R_{out}}$ and a billing function $f(\cdot)$ of an energy plan, if $\vec{R_{out}}$ is utilized to calculate the billed amount, the billing error rate is defined as

$$err_b = \frac{|f(\vec{R_{out}}) - f(\vec{R_{in}})|}{f(\vec{R_{in}})} \tag{8}$$

Note that $f(\cdot)$ can be a constant tariff, or a function given in the TOU or TB plan.

In addition, for some aggregation based smart grid applications [12] (e.g., regional statistics [7]), we define another measure to quantify the utility of our output reading streams:

Definition 8 (Aggregation Error Rate): Given an input reading stream $\vec{R_{in}}$ with K readings and an equal-length output reading stream $\vec{R_{out}}$, the aggregation error rate is defined as

$$err_a = \frac{\left|\sum_{i=1}^{K} \vec{R_{out}}[i] - \sum_{i=1}^{K} \vec{R_{in}}[i]\right|}{\sum_{i=1}^{K} \vec{R_{in}}[i]} \tag{9}$$

where $\vec{R_{in}}[i]$ and $\vec{R_{out}}[i]$ are the i^{th} reading in $\vec{R_{in}}$ and $\vec{R_{out}}$, respectively.

Furthermore, since the output reading stream $\vec{R_{out}}$ might be used to function some real-time services (e.g., load monitoring [15]), the difference between two reading streams $\vec{R_{in}}$ and $\vec{R_{out}}$ should also be measured. Then, we define the reading error rate to quantify such difference:

Definition 9 (Reading Error Rate): Given an input reading stream $\vec{R_{in}}$ with K readings and an equal-length output reading stream $\vec{R_{out}}$, the reading error rate is defined as

$$err_r = \frac{\sum_{i=1}^{K} |\vec{R_{out}}[i] - \vec{R_{in}}[i]|}{\sum_{i=1}^{K} \vec{R_{in}}[i]} \tag{10}$$

where $\vec{R_{in}}[i]$ and $\vec{R_{out}}[i]$ are the i^{th} reading in $\vec{R_{in}}$ and $\vec{R_{out}}$, respectively.

IV. Privacy Preserving Algorithm

In this section, we first derive the conditions for deciding whether a reading is safe to stream or not in Section IV-A, and then present our algorithms in Section IV-B and IV-C.

A. Safe Readings

Given any appliance set A, the candidate rate set can be derived per Definition 2 as G and then we can derive:

Definition 10 (Candidate Reading Set $\mathcal{R}$): A set of all the possible readings $\mathcal{R} = \{r : \forall \omega \in G, r = \omega\phi\}$

Among all the possible readings in $\mathcal{R}$, we define a safe reading as below:

Definition 11 (Safe Reading): Given an (ϵ, δ^m)-Uncertain reading stream $\vec{R}$, a reading r is a safe reading, if adding r (with a specific time t) into $\vec{R}$ also results in an (ϵ, δ^m)-Uncertain reading stream.

Information leakage w.r.t. appliances' ON/OFF status results from the reading r's consumption rate ω, candidate rate set $c(\omega)$ and also the consumption time t, thus r might be a safe reading at time t but not a safe reading at time t' (e.g., some appliances in $c(\omega)$ may have high information leakage at t' caused by the temporal usage patterns). As a result, safe readings cannot be determined/precomputed before loading the input reading stream with timestamps. Therefore, we develop a privacy preserving algorithm to stream safe readings in sequence. The basic idea is – while incrementally generating every safe reading, the algorithm checks the new reading and previous $m-1$ readings whether (ϵ, δ^m)-Uncertainty is still satisfied: if yes, then outputs it in the reading stream; otherwise, iteratively checks the next reading in $\mathcal{R}$.

Conditions for Safe Readings. We now explore the conditions for generating a new safe reading in addition to an existing reading stream. W.l.o.g., we denote m consecutive readings as $r_1, \dots, r_m$ at time $t+1, \dots, t+m$ respectively and consider r_m as the new reading, and the consumption rates $\forall i \in [1, m], \omega_i = \frac{r_i}{\phi}$. Denoting any arbitrary appliance as a_x, the new reading r_m should satisfy ϵ-Uncertainty, thus we have:

$$\forall a_x \in c(\omega_m),\ \max\{\mathcal{I}[(\omega_m, t+m) \to a_x]\} \leq \epsilon \tag{11}$$

Second, we denote the information leakage of a_x's ON status from multiple readings in the m consecutive readings as $\mathcal{I}[a_x]$, which is bounded by δ. For simplicity of notations, we denote $\forall i \in [1, m],\ \mathcal{I}[(\omega_i, t+i) \to a_x]$ as $\mathcal{I}_1, \dots, \mathcal{I}_m$. Then, we consider the worst case that information leakages of a_x's ON status from the m consecutive readings have the least overlap (which leads to the highest union of the information leakages from multiple readings), discussed as below:

- Correlating the information leakages of a_x's ON status from multiple readings clearly increases the joint information leakage (which is the union of multiple individual leakages). As all the information leakages (e.g., $\mathcal{I}_i$ and $\mathcal{I}_{i+1}$) are fixed, the union of them achieves the maximum value when the individual leakages (e.g., $\mathcal{I}_i$ and $\mathcal{I}_{i+1}$) are independent to have the least overlap.

Then, we need to bound the information leakage w.r.t. "a_x is ON in multiple readings". Specifically, since $\forall i \in [1, m], \mathcal{I}_i \in [0, 1]$ (normalized), the information leakage of a_x's OFF status in all the m readings can be represented as $\prod_{i=1}^{m}(1-\mathcal{I}_i)$, and the information leakage w.r.t. "a_x is ON in exactly one out of the m consecutive readings" is $\sum_{i=1}^{m}[\mathcal{I}_i \prod_{j=1; j\neq i}^{m}(1-\mathcal{I}_j)]$. Thus, the maximum $\mathcal{I}[a_x]$ can be derived and bounded as:

$$\begin{aligned}\max\{\mathcal{I}[a_x]\} =& 1 - \prod_{i=1}^{m}(1-\mathcal{I}_i) - \sum_{i=1}^{m}[\mathcal{I}_i \prod_{j=1;j\neq i}^{m}(1-\mathcal{I}_j)] \\ =& 1 - \prod_{i=1}^{m}[(1-\mathcal{I}_i) + \mathcal{I}_i \prod_{j=1;j\neq i}^{m}(1-\mathcal{I}_j)] \leq \delta \end{aligned} \tag{12}$$

Third, similarly, letting a_y be another appliance other than a_x, we denote the information leakage w.r.t. a_x and a_y's ON status from one or multiple readings out of the m consecutive readings as $\mathcal{I}[a_x, a_y]$, which is also bounded by δ. Again, we denote $\forall i \in [1, m],\ \mathcal{I}[(\omega_i, t+i) \to a_y]$ as $\mathcal{I}'_1, \dots, \mathcal{I}'_m$. Then, we also consider the worst case that information leakages of a_x or a_y's ON status from all the m consecutive readings have the least overlap (which also leads to the highest union of the information leakages of a_x or a_y's ON status from multiple readings), discussed as below:

- Correlating the information leakages of a_x and a_y's ON status from one or multiple readings clearly increases the joint information leakage (which is the union of multiple individual leakages); as all the information leakages (e.g., $\mathcal{I}_i$ and $\mathcal{I}'_i$) are fixed, the union of them achieves the maximum value when the individual leakages (e.g., $\mathcal{I}_i$ and $\mathcal{I}'_i$) are independent to have the least overlap.

Again, we need to bound the information leakage w.r.t. "a_x and a_y are ON in one or multiple reading". Specifically, the information leakage w.r.t. "both a_x and a_y are OFF in all the m readings" can be represented as $\prod_{i=1}^{m}(1-\mathcal{I}_i)(1-\mathcal{I}'_i)$, the information leakage w.r.t. "a_x is ON and a_y is OFF in the m consecutive readings" is $\sum_{i=1}^{m}[\mathcal{I}_i \prod_{j=1}^{m}(1-\mathcal{I}'_j)]$, and the information leakage w.r.t. "a_y is ON and a_x is OFF in the m consecutive readings" is $\sum_{i=1}^{m}[\mathcal{I}'_i \prod_{j=1}^{m}(1-\mathcal{I}_j)]$. Thus, the maximum $\mathcal{I}[a_x, a_y]$ can be derived and bounded as:

$$\begin{aligned}\max\{\mathcal{I}[a_x, a_y]\} =& 1 - \prod_{i=1}^{m}(1-\mathcal{I}_i)(1-\mathcal{I}'_i) \\ - \sum_{i=1}^{m}[\mathcal{I}_i \prod_{j=1}^{m}(1-\mathcal{I}'_j)] &- \sum_{i=1}^{m}[\mathcal{I}'_i \prod_{j=1}^{m}(1-\mathcal{I}_j)] \leq \delta \end{aligned} \tag{13}$$

Notice that the information leakage w.r.t. "any combination of appliances including a_x and a_y (w.l.o.g. $a_x, a_y, a_z, \dots$) can be ON in m consecutive readings" is no greater than $\mathcal{I}[a_x, a_y]$ (simply because leakage w.r.t. "a_x and a_y are ON" and leakage w.r.t. "$a_z, \dots$ are ON" should concur to leak information of $a_x, a_y, a_z, \dots$'s ON status). Hence, such information leakage is also bounded by δ if Equation 13 holds.

In summary, while examining the current reading r_m (safe or not) along with the previous $m-1$ readings, if three conditions hold (Equation 11, 12 and 13), then r_m is safe since the reading stream (with r_m) still satisfy $(\epsilon, \delta)^m$-Uncertainty. Such three conditions will be adopted by our stream algorithm to check whether a reading is safe or not.

B. Initializing the Smart Meter (Offline)

Before running the streaming algorithm, the smart meter should be initialized to recursively traverse all the $2^{|A|}$ subsets of A and then identify the candidate rate set G, candidate reading set $\mathcal{R}$ as well as candidate appliance sets $\forall \omega \in G, c(\omega)$. Notice that the initialization is a one-time offline process.

Most real-life households typically have a small or medium number of appliances (e.g., $|A| \leq 40$). Thus, it is feasible to find the exact candidate rate set G by traversing every element in the power set 2^A. In case that a large number of appliances are attached to a smart meter of a community, building or factory (e.g., $|A| = 1000$), the exponential number of possible appliance combinations (in A's power set 2^A) cannot be enumerated in polynomial time. To resolve this, we define a reasonably large number p (e.g., 10^9) as the maximum number of traversed elements in 2^A to terminate the recursive traversal in the smart meter initialization such that the approximated G, $\mathcal{R}$ and $\forall\omega \in G, c(\omega)$ are generated. As will be demonstrated in Section VI, such an approximation leads to satisfactory results in terms of generating candidate rate/reading sets.

C. Privacy Preserving Streaming (Online)

Assuming that the smart meter originally collects K readings in the input stream $\vec{R_{in}}$, our streaming algorithm privately streams K output readings $\vec{R_{out}}$. Our algorithm incrementally generates and outputs safe readings based on the input readings in $\vec{R_{in}}$ where billing and aggregation errors can be extremely low while reading errors can also be minimized to some extent in the output stream. The basic idea is – for each input reading, our algorithm first looks up the closest safe reading in $\mathcal{R}$ (which is *a key building block* of our streaming algorithm), and outputs it in different fashions to achieve good utility of the stream, shown as below.

1) Closest Safe Reading Lookup

Algorithm 1 presents the details of closest safe reading lookup. While the smart meter captures an input reading, it iteratively finds the closest reading r in $\mathcal{R}$ and examine whether r is safe or not: whether the output stream with r satisfies (ϵ, δ^m)-Uncertainty or not.

Algorithm 1: Closest Safe Reading Lookup

```
Input  : an input reading $\vec{R_{in}}[i]$; current output readings
         $\vec{R_{out}} = \langle \vec{R_{out}}[1], \ldots, \vec{R_{out}}[i-1] \rangle$; candidate
         reading set $\mathcal{R}$; privacy parameters $\epsilon, \delta, m$
Output: an output safe reading $\vec{R_{out}}[i]$
initialize $\vec{R_{out}}[i] = \text{argmin}_{\forall r \in \mathcal{R}} |r - \vec{R_{in}}[i]|$ (closest)
while $\vec{R_{out}} \cup \vec{R_{out}}[i]$ is not $(\epsilon, \delta^m)$-Uncertain do
    $\mathcal{R}' = \mathcal{R} - \vec{R_{out}}[i]$
    $\vec{R_{out}}[i] = \text{argmin}_{\forall r \in \mathcal{R}'} |r - \vec{R_{in}}[i]|$ (next closest)
    check whether $\vec{R_{out}} \cup \vec{R_{out}}[i]$ satisfies
      $(\epsilon, \delta^m)$-Uncertainty with Equations 11, 12 and 13
Return the safe reading $\vec{R_{out}}[i]$
```

2) Streaming Algorithm

In order to minimize the aggregation and billing errors, while streaming every safe reading (converted from the input reading), our algorithm rolls over the reading remainder ($\vec{R_{out}}[i] - \vec{R_{in}}[i]$ can be positive or negative) to the either (1) the last reading, or (2) the next input reading of the stream. We propose two roll over options as below:

- Cyclic Reading Conversion (CRC): find the closest safe reading for each input reading, aggregate all the input readings' remainders (either positive or negative) together and roll over the aggregated remainder to the last reading. "Cyclic" means every reading remainder is cyclically reset to 0 (does not affect the next reading) and the aggregated remainder will be subtracted in the last reading.
- Dynamic Reading Conversion (DRC): dynamically update every input reading with the previous reading remainder and then find the closest safe reading based on the updated input reading.

With these two options of handling the reading remainders (which result from the conversion from input readings to safe readings), we detail our streaming algorithm in Algorithm 2.

Algorithm 2: Privacy Preserving Reading Streaming

```
Input  : an input reading stream $\vec{R_{in}}$; candidate reading
         set $\mathcal{R}$; privacy parameters $\epsilon, \delta, m$
Output: output safe reading stream $\vec{R_{out}}$
initialize a reading remainder $\lambda = 0$
/* (1) if CRC (roll over remainders) */
foreach $\vec{R_{in}}[i] \in \vec{R_{in}}, i \in [1, K]$ do
    if $i = K$ then
        /* at the last reading */
        $\vec{R_{in}}[K] = \vec{R_{in}}[K] - \lambda$
        call Algorithm 1 to get $\vec{R_{in}}[K]$'s closest safe
          reading $closest$
        Return $closest$ as $\vec{R_{in}}[K]$
    else
        call Algorithm 1 to get $closest$ as $\vec{R_{in}}[i]$'s closest
          safe reading
        $\lambda + = (closest - \vec{R_{in}}[i])$
        Return $closest$ as $\vec{R_{out}}[i]$
/* (2) if DRC (roll over remainders) */
foreach $\vec{R_{in}}[i] \in \vec{R_{in}}, i \in [1, K]$ do
    $\vec{R_{in}}[i] = \vec{R_{in}}[i] - \lambda$
    run Algorihtm 1 to get $closest$ as $\vec{R_{in}}[i]$'s closest
      safe reading
    $\lambda = closest - \vec{R_{in}}[i]$
    Return $closest$ as $\vec{R_{out}}[i]$
```

CRC Option. Given the i^{th} reading $\vec{R_{in}}[i]$ in the stream, CRC first verifies whether it is the last reading. If yes, CRC will sum up the aggregated reading remainder λ to the current reading $\vec{R_{in}}[K]$ and return its closest safe reading. Otherwise, the algorithm returns the original input reading $\vec{R_{in}}[i]$'s closest safe reading $closest$, and λ is updated with the difference $closest - \vec{R_{in}}[i]$. In summary, Algorithm 2 with the CRC option has the following characteristics:

- **Aggregation Error**: since the aggregated reading remainder λ is integrated into the last reading, the overall aggregation error would be the difference between one single reading ($\vec{R_{in}}[K] - \lambda$) and its closest safe reading. It has high possibility of being close to 0, as shown in experimental section.
- **Billing Error**: assuming the time frame for $\forall i \in [1, K]$, $\vec{R_{in}}[i]$ is the billing cycle, if the tariff function $f(\cdot)$ of electricity bill is a constant, the difference between aggregating the readings in $\vec{R_{in}}$ and $\vec{R_{out}}$ results from the last reading (since reading remainders have been integrated in the aggregation). Thus, CRC option's billing error is identical to its aggregation error (close to 0).

- **Reading Error**: since all K readings in the stream except the last reading can achieve its local optimum towards minimizing the reading error with its closest safe reading, the reading error can be minimized to some extent.
- **Privacy**: $\vec{R_{out}}$ satisfies (ϵ, δ^m)-Uncertainty.

DRC Option. DRC dynamically updates each reading by integrating the previous reading remainder λ, then returns the updated reading's closest safe reading, and finally generates a new reading remainder for the next reading. Also, Algorithm 2 with the DRC option has the following characteristics:

- **Aggregation Error**: since every reading remainder is integrated into the next reading, the aggregation error is the last remainder which is well balanced by all the readings into a very small number. Then, the aggregation error is extremely close to 0.
- **Billing Error**: similar to the aggregation error, the billing error (with constant tariff) is also extremely close to 0.
- **Reading Error**: since each reading integrates the previous remainder, the reading error of DRC is relatively higher than CRC. Nevertheless, after integrating the previous remainder, each reading can also achieve its local optimum towards minimizing the reading error by converting the reading to the closest safe reading.
- **Privacy**: $\vec{R_{out}}$ satisfies (ϵ, δ^m)-Uncertainty.

Furthermore, our privacy preserving algorithm outputs safe readings which are associated with large number of possible combinations of appliances in real world (by ensuring (ϵ, δ^m)-Uncertainty). Thus, our proposed approach could also prevent the privacy risks against NILM algorithms (both supervised [9], [8], [25] and unsupervised [24], [22]) for two reasons. First, safe readings are converted from the true readings, and the aggregated consumption have been changed from the original readings. Second, the output safe readings in our algorithm are associated with large number of possible combinations of appliances in real world by satisfying the privacy notion, such large number of appliance combinations would increase the estimated consumption amount of more appliances (compared to the true readings) and greatly reduce the learning accuracy.

V. Analysis

A. Privacy Analysis

We now analyze the privacy leakage in the output $\vec{R_{out}}$, assuming that the adversary can possess the some or all of the background knowledge described in Section III-B3.

Lemma 1: The output reading stream of Algorithm 2: $\vec{R_{out}}$ satisfies (ϵ, δ^m)-Uncertainty.

Proof. Since $\forall i \in [1, K]$, $\vec{R_{out}}[i]$ are generated in time series sequence, and each newly streamed output reading together with the most recent $m-1$ readings in the stream strictly satisfy the three groups of conditions (Equations 11, 12 and 13), it is straightforward to see that all the output readings in the stream satisfies (ϵ, δ^m)-Uncertainty. ⊡

B. Complexity Analysis

Our approach consists of two phases (1) the offline smart meter initialization phase, and (2) the online streaming phase. First, the offline phase recursively traverses the power set of A (exponential) to identify the candidate reading set and possible rates' candidate appliance sets. For a large size A, the recursive traversal is terminated with a sufficiently large number of traversed elements in 2^A (denoted as p). Second, the online phase streams K readings: for each reading, it iteratively looks up a closest safe reading from $\mathcal{R}$ (say, $O(n)$ readings are returned to identify a safe reading) and examines all the appliances in every candidate reading's candidate appliance set (at most $|A|$ appliances) to check whether the information leakage meets (ϵ, δ^m)-Uncertainty or not for the most recent m output readings. Thus, the computational complexity of the online streaming phase is $O(K|A|nm)$, which is polynomial.

C. Implementation and Scalability

Our streaming algorithm can be easily integrated into a smart meter. Specifically, the appliance set A and privacy parameters ϵ, δ and m can be loaded into the smart meter via a web interface or a mobile application for generating G, $\mathcal{R}$ and $\forall c(\omega)$ in the initialization. Privacy parameter ϵ is specified to bound the information leakage in single readings while δ and m are specified to bound the information leakage from the correlations of energy usage in one or multiple readings. The CRC and DRC can be implemented as different privacy-aware running modes in the smart meter. Once a new reading is captured by the smart meter, a safe reading is generated immediately and transmitted to the utility company.

Specifically, the consumers can locally adjust their privacy parameters ϵ, δ and m based on their levels of privacy demand at any time, but they do not need to change the (CRC and DRC based) privacy preserving streaming algorithm (which is integrated in the smart meter). For better functioning some emerging services (e.g., energy saving recommendation, and non-intrusive load monitoring [15]), the utility company can keep a detailed inventory of each appliance, which will not pose additional privacy concerns to our streaming algorithm (since we assume that adversaries could possess the appliance list as background knowledge in our privacy model). Upon addition and/or removal of appliances (e.g., purchasing/replacing a new appliance, house owners move out, renting the houses to tenants, and houses with visitors who bring their own devices), the consumers can locally reset the smart meter with an updated list of appliance set A (running smart meter initialization once) and inform the utility company if necessary. Notice that the consumers do not necessarily change the streaming algorithm, and they can switch from CRC to DRC (and vice-versa), as well as specify a new group of privacy parameters ϵ, δ and m according to their privacy demand.

Finally, after implementing our privacy preserving streaming algorithm in the smart meters, the utility can offer such privacy-aware smart metering services to the consumers. Since both CRC and DRC based streaming algorithm would result in small billing errors (less than $\pm 4\%$ in general), if the consumers pay 4% more, this is the price traded for better privacy protection; if they pay less, the utility can charge a small amount of service fee to tackle such non-technical losses. In the meanwhile, there are two alternative approaches that can reduce the billing errors to as low as 0: (1) rolling over the remainder of the last reading in the current billing cycle to the next billing cycle (instead of discarding it), and (2) we can let the consumers locally calculate the billed amount without leaking private information. Such approaches would need relatively more trust on the consumers, and indeed match the fact that utilities in many countries (e.g., US and Canada) allow consumers to submit their readings by themselves.

VI. Experiments

A. Experimental Setup

Datasets. Richardson et al. [28] collected 22 dwellings' power consumption over two years in East Midlands, UK. Each of the 22 smart meters has reported 1,051,200 readings (1 reading per minute). Furthermore, in UMass Trace Repository (http://traces.cs.umass.edu/), Barker et al. [5] collected 3 smart meters' consumption data over three months in 2012, respectively (1 reading per second).

We conducted the experiments on these two datasets (denoted as "UK" and "UMass", respectively), and averaged the results of multiple smart meters in each dataset. The characteristics of the two datasets are presented in Table IV.

TABLE IV
Characteristics of the Datasets

Datasets	Meters #	Average # of Appliances	Readings #	Time
UK Data	22	24.0	1,051,200	2 years
UMass Data	3	35.7	7,776,000	3 months

Parameters. ϵ is selected from 0.01 to 0.2 while δ is selected from 0.01 to 0.1 in the experiments. In the evaluation of heuristic smart meter initialization, we run additional tests by letting the number of appliances be $|A| = 20, 40, 100$ and each appliance's consumption rate (watts) is selected from a real-world list of appliances and consumption rates [1]. For the tariffs of energy usage, we set the rates per the real world energy pricing plans offered by Pacific Gas and Electric Company (PG&E) [2].

Platform. All the experiments were performed on a DELL PC with Intel Core i7-4790 CPU 3.60GHz and 16G RAM running Microsoft Windows 8.1 Operating System.

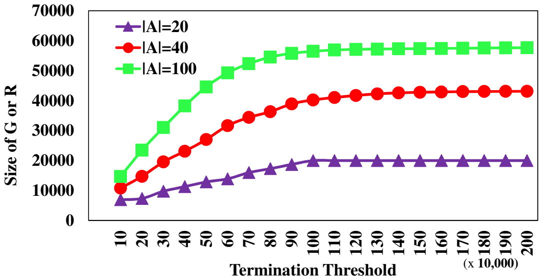


Fig. 1. Smart Meter Initialization

B. Smart Meter Initialization (Offline)

Both UK and UMass Data include specific appliances and their time series consumption information. We can obtain the exact candidate rate G, candidate reading set $\mathcal{R}$ and the candidate appliance sets $\forall \omega \in G, c(\omega)$. However, if the number of appliances m reaches 50, we may not be able to obtain the exact result within a reasonable time due to the exponential increase of the computational cost. In these cases, we set a termination threshold p in the heuristic safe candidate rate set generation, which runs only once for every smart meter and requires a *one-time offline* computational cost. Figure 1 presents the experimental results for smart meter initialization. As p tends close to 10^6, the exact results of G, $\mathcal{R}$ and $\forall \omega \in G, c(\omega)$ for $|A| = 20$ can be derived, and the results of $|A| = 40$ and $|A| = 100$ become relatively stable. Therefore, if $|A|$ increases, the approximated results G, $\mathcal{R}$ and $\forall \omega \in G, c(\omega)$ can be sufficiently accurate.

C. Utility Evaluation

In the experiments, we have evaluated all three error rates (billing, aggregation and reading) for both CRC and DRC options with constant tariff.[1] Specifically, we conduct experiments using both UK and UMass datasets to test CRC and DRC's utility on varying ϵ, δ and m, respectively.

We select $\epsilon \in [0.1, 0.3]$, $\delta \in [0.05, 0.15]$ and $m \in [10, 30]$. While testing every privacy parameter, the other two parameters are fixed to achieve the best privacy protection. For ϵ, we fix $\delta = 0.05$ and $m = 30$; for δ, we fix $\epsilon = 0.1$ and $m = 30$; for m, we fix $\epsilon = 0.1$ and $\delta = 0.05$. The experimental results of aggregation/billing error rates are plotted in Figure 2 (varying ϵ in Figure 2(a), varying δ in Figure 2(b) and varying m in Figure 2(c)). Furthermore, the corresponding experimental results of reading error rates are plotted in Figure 3. Then, we have the following observations:

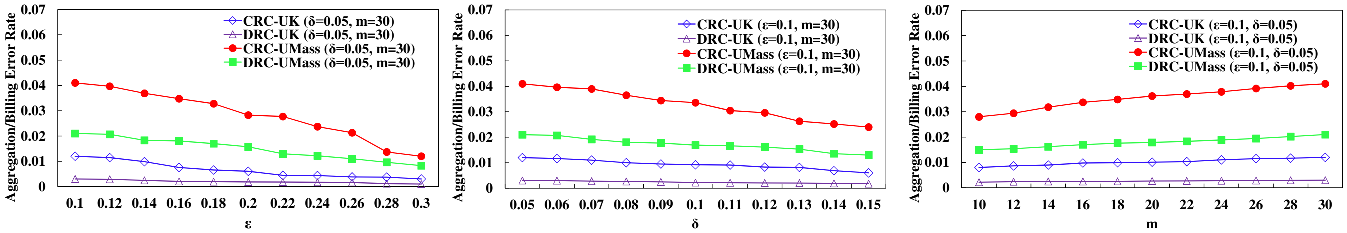
- Smaller ϵ and δ or greater m in the algorithm (both CRC and DRC) could stream outputs with lower aggregation/billing and reading errors.
- Aggregation/billing error rates are low (UK data: $< 1.2\%$ and UMass data: $< 4.1\%$).
- The CRC option in the streaming generates higher aggregation/billing errors but lower reading errors than the DRC option (for both UK and UMass data).
- Privacy parameter ϵ impacts utility more significantly than δ and m in our privacy model. Utility improves more quickly as ϵ increases (all the errors decrease more quickly) compared to increasing δ or reducing m.

In case of dynamic energy billing, e.g., TOU and TB plans [2], [27], [20], we have discussed a possible solution to ensure 0 billing error in Appendix A.

D. Efficiency Evaluation

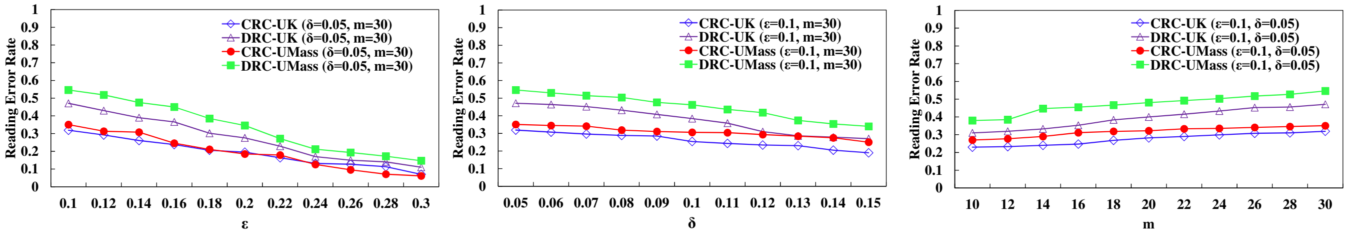
Computational Performance. Figure 4(a) presents the one-time offline runtime for the number of appliances $|A| = 20, 25, 30, \ldots, 1000$. When $|A| \geq 60$, we implement the heuristic smart meter initialization by specifying a large termination threshold $p = 10^9$. As expected, the one-time offline cost is tolerable for any smart meter (e.g., 120MHz [3]). On the other hand, Figure 4(b) presents the total online runtime

[1] Aggregation error rate err_a always equals the billing error rate err_b in case of constant tariff.



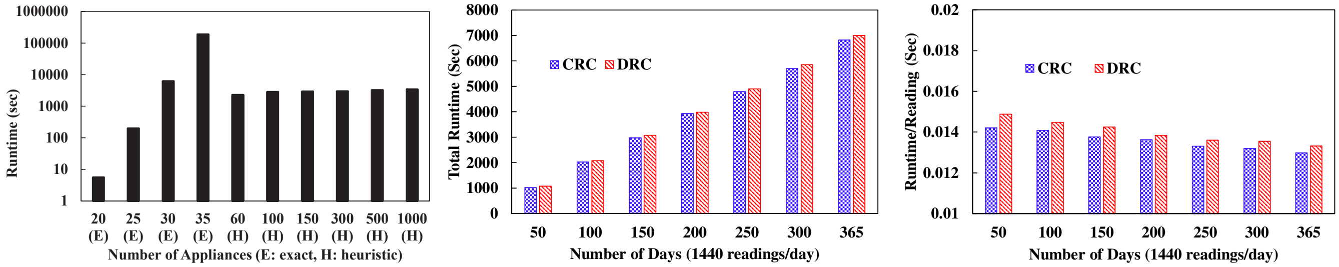
(a) Aggregation/Billing Error Rate vs. ϵ (b) Aggregation/Billing Error Rate vs. δ (c) Aggregation/Billing Error Rate vs. m

Fig. 2. Aggregation/Billing Error Rate (Constant Tariff) – UK and UMass Datasets



(a) Reading Error Rate vs. ϵ (b) Reading Error Rate vs. δ (c) Reading Error Rate vs. m

Fig. 3. Reading Error Rate – UK and UMass Datasets



(a) Offline (one-time) Computational Cost (b) Online Computational Cost (c) Online Runtime Per Reading

Fig. 4. Computational Performance (UK Data)

for a varying number of readings in the input stream (from 50 days to 1 year, with 1440 readings per day). Both CRC and DRC take less than 2 hours to convert and stream 1 year's readings in the UK data (525,600 readings in total). The CRC option is slightly more efficient than DRC in the streaming algorithm. Note that our streaming algorithm also has similar performance on the UMass data.

Streaming Latency. Figure 4(c) demonstrates the runtime consumed by streaming every single reading on average, where $\epsilon \in [0.1, 0.3], \delta \in [0.05, 0.15]$ and $m \in [10, 30]$. Such runtime is less than 0.016 second ($\ll 1$ second) for both CRC and DRC. As a result, although smart meters have relatively weaker computation power (e.g., 120MHz [3]) than an experimental PC (e.g., 3.60GHz), our privacy preserving streaming algorithm can be implemented in the smart meters for high resolution reading streams without any latency.

E. Case Study

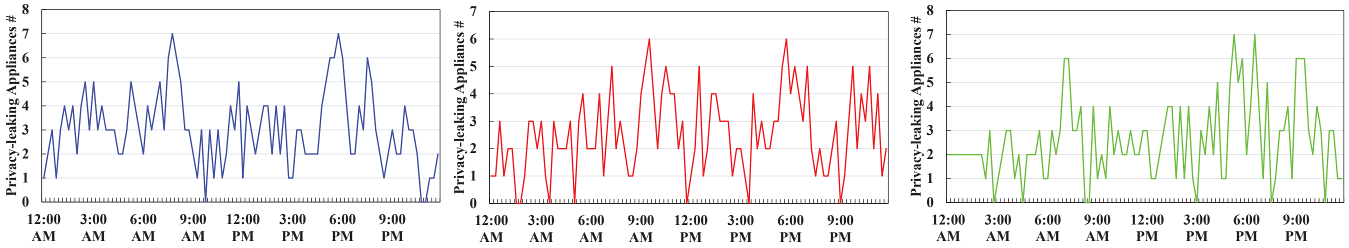
Besides conducting experiments on the overall metering datasets, we also study the cases of specific houses.

1) Case Study Setup

We select three sample houses with different types (terraced, semi-detached and detached) from the UK dataset [28] for the case study. In the three houses, 18, 26 and 33 appliances are installed, respectively. Specifically, we simulate the information leakage and the privacy preserving algorithm on the data collected from the three houses on Jan 1, 2008. Note that we aggregate the high-resolution readings into 15 minutes per reading, which is a commonly used reading frequency. Also, some non-electric appliances (e.g., Heating, Water Heating, and Gas Oven) do not consume electricity and are not considered as leaking privacy. In this case study, we specify $\epsilon = 0.3$, $\delta = 0.2$ and $m = 5$ (5 consecutive readings form one hour interval for testing usage patterns in multiple readings). Then, safe readings should make the information leakage of each appliance not exceed 0.3 and the information leakage of any appliance or combination of appliances in every 5 consecutive readings not exceed 0.2.

Due to fluctuated power quality, the actual reading may not equal the candidate reading derived from the labeled consumption rates of the appliances (they are indeed close to each other). For every reading, we find its closest candidate reading to derive the information leakage. This matches the fact that real-life adversaries also utilize the labeled consumption rates in their mind to learn the appliances' ON/OFF status.

To estimate the *information leakage of appliances from their*



(a) Terraced (b) Semi-Detached (c) Detached

Fig. 5. Number of Privacy-leaking Appliances vs. Time in One Day ($\epsilon = 0.3$, $\delta = 0.2$, $m = 5$)

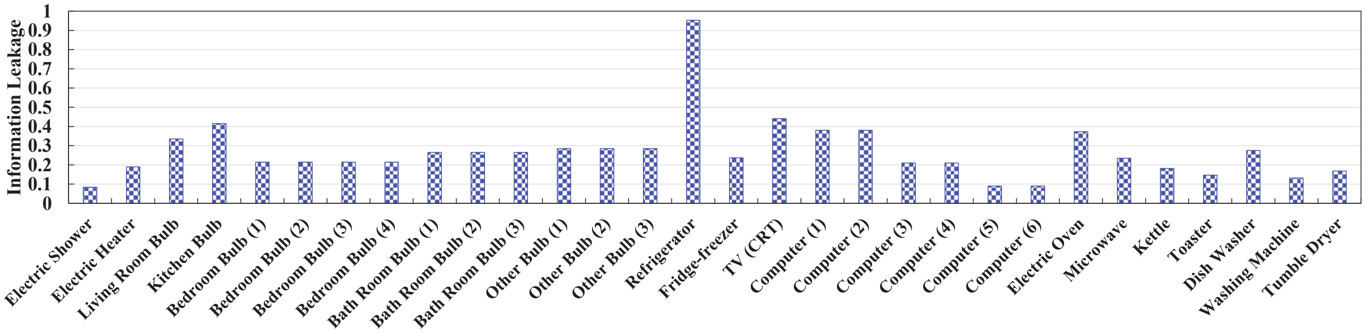


Fig. 6. Information Leakage in the Detached House at 6:30pm ($\epsilon = 0.3$, $\delta = 0.2$, $m = 5$)

temporal usage patterns $\mathcal{I}[t \to a_x] \in [0, 1]$ for the day Jan 1, 2008, we perform empirical study by surveying 10 students on campus. Every student estimates the likelihood of each of the appliances is ON at all the timestamps on Jan 1, 2008 (which is a generic estimate for all the households without looking at the reading), where 24 time slots are given and every 4 readings share the same result, e.g., the information leakage w.r.t. "TV is ON" in [8pm, 9pm) is 0.3. Finally, we average the results for each appliance and time in all the surveys.

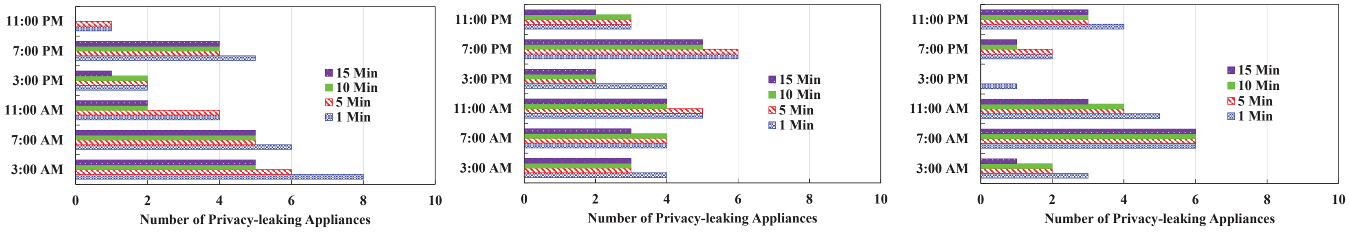
2) Information Leakage Analysis

From the reading stream of each house, if the information leakage of any appliance's ON status at time t is greater than $\epsilon = 0.3$, or the information leakage of any appliance or any combination of appliances' ON status in any 5 consecutive readings (including time t) is greater than $\delta = 0.2$, then such appliance(s) are considered as "privacy-leaking appliances at time t". Following this rule of information leakage analysis on the reading stream of three sample houses, we can identify the number of "privacy-leaking appliances" for each reading time, and then plot such results for three houses in a time series manner in Figure 5. It shows the number of privacy-leaking appliances at different times in such three houses. We can also observe which original reading is safe (no privacy-leaking appliances at a specific time), and which original reading is unsafe and how unsafe (based on the number of privacy-leaking appliances at a specific time). Specifically, we find out that some readings are originally safe in every house w.r.t. the given parameters ϵ, δ and m (e.g., 9:45am in the terraced house, 3:30am in the semi-detached house, and 10:45pm in the detached house on Jan 1, 2008). On the contrary, some readings in three houses are highly unsafe, e.g., 7:45am in the terraced house, 5:45pm in the semi-detached house, and 6:30pm in the detached house. The privacy-leaking appliances in those readings are listed as below:

- Terraced (7:45am): Bathroom Bulb, Kitchen Bulb, Bedroom Bulb (1), Bedroom Bulb (2), Refrigerator, Toaster, Kettle, Microwave.
- Semi-detached (5:45pm): Refrigerator, Kitchen Bulb and Living Room Bulb (as a combination), Electric Oven, Microwave, Kettle, Bathroom Bulb (1).
- Detached (6:30pm): Kitchen Bulb, Living Room Bulb, Refrigerator, TV (CRT), Computer (1), Computer (2), Electric Oven, Dishwasher.

Furthermore, we select an unsafe reading at 6:30pm in the detached house (6 appliances have information leakage higher than 0.3) to demonstrate the privacy risks of all the appliances. In Figure 6, we plot the information leakage of the 30 electric appliances, which are obtained by examining the candidate appliance set of the reading/consumption rate (4.94kW). Then, we have the following observations:

- Refrigerator is the most easily detected at any time (with an information leakage higher than 0.9 due to its high $\mathcal{I}[t \to a_x]$). This poses a challenge that an extremely high $\mathcal{I}[t \to a_x]$ of an appliance (such as refrigerator) would lead to a high overall information leakage all the time. Nevertheless, its ON status does not leak any private information of the consumer since almost all the households keep their such appliances (e.g., refrigerator) running all the time.
- Although the bulbs in different rooms of each house may have the same consumption rate, their information leakage at the same time might be different (since such bulbs may have different $\mathcal{I}[t \to a_x]$ at the same time).
- In the detached house, at 6:30pm, TV is also a privacy-



(a) Terraced (b) Semi-Detached (c) Detached

Fig. 7. Number of Privacy-leaking Appliances vs. Reading Frequency ($\epsilon = 0.3$, $\delta = 0.2$, $m = 5$)

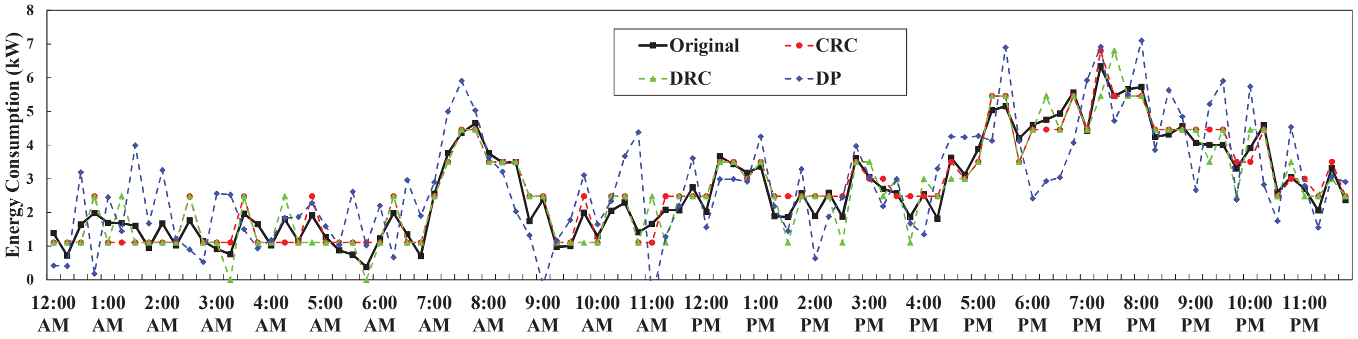


Fig. 8. CRC and DRC ($\epsilon = 0.3$, $\delta = 0.2$, $m = 5$) vs. Original Reading and Differential Privacy (5-DP)

leaking appliance with high information leakage whereas electric shower is the hardest to identify with the lowest information leakage. Similarly, in the semi-detached house, at 5:45pm, kitchen bulb is the easiest to detect (besides the refrigerator) whereas two computers are the hardest to detect. In the terraced house, at 7:45am, microwave is the easiest to detect (besides refrigerator) whereas tumble dryer is the hardest to detect. Notice that the ON status of some appliances can be leaked as a combination (e.g., kitchen bulb and living room bulb in the semi-detached house), and some appliances can be detected from the correlations of energy usage in multiple readings (e.g., dishwasher in the detached house).

Note that all the above observations match the ground truth of power consumption in households. The information leakage of each appliance's ON status can either increase or decrease over time due to the highly fluctuated consumption amount and the usage patterns of such appliance at different times.

Finally, in order to learn how reading frequency affect the privacy risks applied to different readings, we conducted experiments to examine the number of privacy-leaking appliances at 6 selected times in the same day (3 AM, 7AM, 11AM, 3PM, 7PM, 11PM) by varying the reading frequencies (from 1 reading per minute to 1 reading per 15 minutes). Then, we plot the number of privacy-leaking appliances at those 6 different times with 4 different reading frequencies in Figure 7 (the results obtained from each house is plotted in a subfigure). Therefore, we can learn that readings would leak more private information if they are reported more frequently (e.g., adversaries can identify the largest number of privacy-leaking appliances if the readings are reported with the highest frequency 1 minute/reading). This matches the fact that finer-grained readings would result in more privacy leakage.

3) *CRC and DRC*

To bound the privacy leakage in a reading stream, our privacy preserving algorithm has two streaming options CRC and DRC that satisfy (ϵ, δ^m)-Uncertainty. Due to space limit, we only demonstrate the reading conversion results in the sample detached house on Jan 1, 2008 in Figure 8. We can find out the safe readings w.r.t. to privacy parameters $\epsilon = 0.3$, $\delta = 0.2$ and $m = 5$ are close to the original readings, then the reading errors can be minimized in two different ways (note that the aggregation and billing errors are negligible). Meanwhile, we have plotted the reading conversion results for differential privacy [11] by adding the generic Laplace noise to ensure 5-DP for the reading stream in which the multiplicative differences between the probabilities of generating any identical output from two neighboring inputs are bounded by e^5. The results show that differentially private algorithm would lead to much higher errors and also greatly fluctuate the output readings.

4) *Phantom Load*

We also examined how phantom load (power consumption as some appliances are OFF, e.g., computers, microwave, electric oven and TV) affect the performance of information leakage and the privacy preserving algorithm in our case study. The phantom loads for such appliances are referred to sites such as http://standby.lbl.gov/summary-table.html. For instance, computer's phantom load is $\sim 3.84\%$ its regular consumption rate, TV is $\sim 3.53\%$, Microwave is $\sim 0.21\%$, and Washing Machine is $\sim 0.48\%$. In the case study, for the appliances with phantom load, we assign the phantom load to their OFF status and derive the number of privacy-leaking appliances (PL App #) and the reading errors of the CRC and DRC. Table V shows that phantom loads can slightly make the readings safer and lead to less errors in reading conversions.

TABLE V
PERFORMANCE VS. PHANTOM LOAD

	Terraced	Semi-Detached	Detached
PL App #	-0.27%	-0.31%	-0.43%
CRC Error	-0.094%	-0.113%	-0.186%
DRC Error	-0.093%	-0.109%	-0.179%

Finally, if some specific smart meters are deployed by consumers with high-resolution readings (e.g., microgrids [19]), e.g., the households established by UMass [5], information leakage may still exist if explicitly disclosing such readings to other parties. For instance, as the deployed smart meter is integrated into the main grid, the high-resolution readings might be analyzed in some applications (e.g., NILM [15], and regional statistics [7]). Given any reading in such applications (aggregated or fine-grained), adversaries can still learn the status of appliances with their background knowledge.

VII. LIMITATIONS AND CHALLENGES

Limitations. First, the reading errors of our CRC or DRC based streaming algorithm can be relatively high if specifying a small ϵ, δ and/or large m (for high privacy demand), compared to the aggregation and billing errors (which can be close to 0). Therefore, it may affect the accuracy of some real-time services based on the smart meter streams (e.g., load monitoring). Second, as discussed in Section VI-E2, if some appliances are very likely to be ON at most of the times ($\mathcal{I}[t \to a_x]$ lies close to 1) such as refrigerator and heating in winter, the information leakage of such appliances cannot be effectively bounded without sacrificing too much utility. Nevertheless, its ON status leaks very limited privacy of the consumers since almost all the households keep them running all the time. Finally, once a new appliance is connected to the home, smart meter needs to be re-initialized for the privacy model. Also, the reading conversion may also violate some regulations for guaranteeing the integrity of the bills in some countries/regions.

Challenges. First, smart meter initialization requires an exponential complexity (offline) to generate the candidate rate set and each possible consumption rate's candidate appliance set. For a small or medium number of appliances, the algorithm can be executed once to obtain the exact result. However, for a large number of appliances, we have to run a heuristic algorithm (e.g., specifying a terminating point for the algorithm) to obtain an approximated result. Second, the information leakage of an appliance is derived based on both the consumption rate and time. It is challenging to quantify the information leakage from the temporal usage patterns of different appliances $\mathcal{I}[t \to a_x]$. In the case study on Jan 1, 2008 (Section VI-E), we survey energy consumers to obtain such patterns (the likelihood that most energy consumers use each appliance at different times). Alternatively, we can use the probability distribution function in [10] to estimate such patterns and the corresponding information leakage.

VIII. CONCLUSION AND FUTURE WORK

Smart meter reading streams have posed severe privacy threats to electricity consumers on the power grid. Beyond the smart meter privacy issues tackled in literature, in this paper, we have quantitatively measured and mitigated the information leakage in such streaming data based on a wide variety of background knowledge, including appliances' consumption rates and temporal patterns of usage, other correlations/patterns of running the same or different appliances at different times. We have defined a novel privacy model for time series reading stream and developed a privacy preserving streaming algorithm that efficiently outputs safe readings with excellent utility. We have conducted experiments on real datasets to validate the performance of our approach.

We can extend our work in several directions. First, for an exponential number of candidate consumption rates and the corresponding appliance subsets in A's power set, we can try to develop other heuristic or approximation algorithms to generate the safe candidate rate set instead of simply setting the termination threshold for recursively traversing A's power set, e.g., designing rules to prune the search space. Second, for some real-time applications (e.g., load monitoring [15]) which have high demand on reducing the reading errors, we can explore other privacy preserving streaming algorithms for smart meters to further minimize such errors. Third, inspired from many state-of-the-art NILM solutions which start to use the transient of the power consumption signal or the transition between power consumption states (e.g., HMM [24], [22]) to estimate the specific-appliance's consumption, we plan to investigate the background knowledge of consumption transient and the corresponding privacy leakage, and define a rigorous privacy notion to quantify and bound such risks. Moreover, information leakage may also occur in other time series data, such as stock market data, and system/server logs. We plan to explore efficient privacy preserving solutions to tackle all of these problems in our future work.

ACKNOWLEDGMENT

This research is supported in part by the National Science Foundation under the Grant No. CNS-1618221. Authors with Concordia University are partially supported by the Natural Sciences and Engineering Research Council of Canada under Discovery Grant N01035. We also acknowledge the real world data support from the UMass Trace Repository and the Center for Renewable Energy Systems Technology at Loughborough University, UK. Meanwhile, we sincerely thank the anonymous reviewers for their very constructive comments.

REFERENCES

[1] *http://energy.gov/*.
[2] *http://www.pge.com/*.
[3] *http://www.atmel.com/products/smart-energy/power-metering/*.
[4] G. Ács and C. Castelluccia. I have a dream! (differentially private smart metering). In *Information Hiding*, pages 118–132, 2011.
[5] S. Barker, A. Mishra, D. Irwin, E. Cecchet, P. Shenoy, and J. Albrecht. Smart*: An open data set and tools for enabling research in sustainable homes. In *the 2012 Workshop on Data Mining Applications in Sustainability*, 2012.
[6] J. Cao, P. Karras, C. Raïssi, and K.-L. Tan. rho-uncertainty: Inference-proof transaction anonymization. *PVLDB*, 3(1):1033–1044, 2010.
[7] C.-K. Chu, J. K. Liu, J. W. Wong, Y. Zhao, and J. Zhou. Privacy-preserving smart metering with regional statistics and personal enquiry services. In *ASIACCS*, pages 369–380, 2013.

[8] R. Dong, L. J. Ratliff, H. Ohlsson, and S. S. Sastry. A dynamical systems approach to energy disaggregation. In *Proceedings of the 52nd IEEE Conference on Decision and Control*, pages 6335–6340, 2013.

[9] R. Dong, L. J. Ratliff, H. Ohlsson, and S. S. Sastry. Energy disaggregation via adaptive filtering. In *51st Annual Allerton Conference on Communication, Control, and Computing*, pages 173–180, 2013.

[10] R. Dong, L. J. Ratliff, H. Ohlsson, and S. S. Sastry. Fundamental limits of nonintrusive load monitoring. In *3rd International Conference on High Confidence Networked Systems*, pages 11–18, 2014.

[11] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *TCC*, pages 265–284, 2006.

[12] Z. Erkin, J. Troncoso-Pastoriza, R. Lagendijk, and F. Perez-Gonzalez. Privacy-preserving data aggregation in smart metering systems: An overview. *Signal Processing Magazine, IEEE*, 30(2):75–86, 2013.

[13] X. Fang, S. Misra, G. Xue, and D. Yang. Smart grid - the new and improved power grid: A survey. *IEEE Communications Surveys and Tutorials*, 14(4):944–980, 2012.

[14] S. Goel and Y. Hong. Security challenges in smart grid implementation. *SpringerBriefs in Cybersecurity*, pages 1–39, 2015.

[15] G. W. Hart. Nonintrusive appliance load monitoring. *Proceedings of the IEEE*, 80(12):1870–1891, Dec 1992.

[16] Y. Hong, J. Vaidya, H. Lu, and M. Wu. Differentially private search log sanitization with optimal output utility. In *Proceedings of Extending Database Technology*, pages 50–61, 2012.

[17] Y. Hong, J. Vaidya, and H. Lu. Secure and efficient distributed linear programming. *Journal of Computer Security*, 20(5):583–634, 2012.

[18] Y. Hong, J. Vaidya, H. Lu, P. Karras, and S. Goel. Collaborative search log sanitization: Toward differential privacy and boosted utility. *IEEE Trans. Dependable Sec. Comput.*, 12(5):504–518, 2015.

[19] Y. Hong, S. Goel, and W. Liu. An Efficient and Privacy Preserving Scheme for Energy Exchange among Smart Microgrids. *International Journal of Energy Research*, 40(3):313–331, 2016.

[20] M. Jawurek, M. Johns, and F. Kerschbaum. Plug-in privacy for smart metering billing. In *PETS*, pages 192–210, 2011.

[21] A. Korolova, K. Kenthapadi, N. Mishra, and A. Ntoulas. Releasing search queries and clicks privately. In *WWW*, pages 171–180, 2009.

[22] L. Mauch, K. S. Barsim, and B. Yang. How well can HMM model load signals. In *Proceedings of the 3rd International Workshop on Non-Intrusive Load Monitoring*, 2016.

[23] F. McSherry and I. Mironov. Differentially private recommender systems: building privacy into the net. In *KDD*, pages 627–636, 2009.

[24] O. Parson, S. Ghosh, M. Weal, and A. Rogers. Non-Intrusive Load Monitoring Using Prior Models of General Appliance Types. In *Proceedings of the Twenty-Sixth AAAI Conference*, 2012.

[25] D. Piga, A. Cominola, M. Giuliani, A. Castelletti, and A. E. Rizzoli. Sparse optimization for automated energy end use disaggregation. *IEEE Trans. Contr. Sys. Techn.*, 24(3):1044–1051, 2016.

[26] A. Reinhardt, D. Egarter, G. Konstantinou, and D. Christin. Worried about privacy? let your PV converter cover your electricity consumption fingerprints. In *2015 IEEE International Conference on Smart Grid Communications*, pages 25–30, 2015.

[27] G. Research. Understanding the potential of smart grid data analytics. *A GTM Research Whitepaper*.

[28] I. Richardson, M. Thomson, D. Infield, and C. Clifford. Domestic electricity use: A high-resolution energy demand model. *Energy and Buildings*, 42(10):1878 – 1887, 2010.

[29] C. Rottondi, G. Verticale, and A. Capone. Privacy-preserving smart metering with multiple data consumers. *Computer Networks*, 57(7):1699–1713, 2013.

[30] C. Rottondi, G. Verticale, and C. Krauss. Distributed privacy-preserving aggregation of metering data in smart grids. *IEEE Journal on Selected Areas in Communications*, 31(7):1342–1354, 2013.

[31] L. Sankar, S. R. Rajagopalan, S. Mohajer, and H. V. Poor. Smart meter privacy: A theoretical framework. *IEEE Trans. Smart Grid*, 4(2):837–846, 2013.

[32] M. Savi, C. Rottondi, and G. Verticale. Evaluation of the precision-privacy tradeoff of data perturbation for smart metering. *IEEE Trans. Smart Grid*, 6(5):2409–2416, 2015.

[33] E. Shi, T.-H. H. Chan, E. G. Rieffel, R. Chow, and D. Song. Privacy-preserving aggregation of time-series data. In *NDSS*, 2011.

[34] G. Smith. Quantifying information flow using min-entropy. In *Eighth International Conference on Quantitative Evaluation of Systems*, pages 159–167, 2011.

[35] R. Srikant and R. Agrawal. Mining sequential patterns: Generalizations and performance improvements. In *International Conference on Extending Database Technology*, pages 3–17, 1996.

[36] L. Sweeney. k-anonymity: a model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 10(5):557–570, 2002.

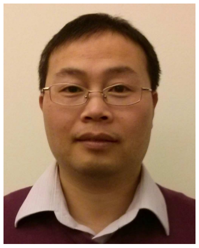
[37] W. Yang, N. Li, Y. Qi, W. H. Qardaji, S. E. McLaughlin, and P. McDaniel. Minimizing private data disclosures in the smart grid. In *Proceedings of ACM Conference on CCS*, pages 415–427, 2012.

[38] F. Zhang, L. He, W. He, and X. Liu. Data perturbation with state-dependent noise for participatory sensing. In *2012 Proceedings of IEEE INFOCOM*, pages 2246–2254, March 2012.

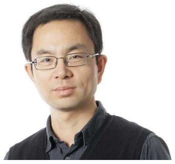
Appendix A
Privacy Preserving Streaming for Dynamic Energy Billing

Besides the constant tariff in a standard energy billing plan, two different dynamic pricing policies (TOU and TB) are widely adopted by utility companies [2], [27], [20]. To minimize the billing errors under such plans, the smart meter (e.g., household) can locally compute the bill using the input readings $\vec{R_{in}}$ (but without disclosing $\vec{R_{in}}$). At this time, billing can be separated from the CRC or DRC based privacy preserving streaming. Then, output reading stream $\vec{R_{out}}$ can be transmitted to the utility company in sequence while the smart meter can still privately use $\vec{R_{in}}$ and TOU or TB plan to calculate the bill (disclosing the bill would not leak any information in the vector $\vec{R_{in}}$ [17]). In this case, we assume that the smart meter (e.g., a household) is a trusted entity to report the true bill (dynamic pricing) to the utility company.

Yuan Hong is an Assistant Professor in the Department of Computer Science at Illinois Institute of Technology. He received his Ph.D. degree in Information Technology from Rutgers, the State University of New Jersey. His research interests primarily lie at the intersection of privacy, security, optimization, and data mining. His research is supported by the National Science Foundation. He is a member of the IEEE.



Wen Ming Liu received the Ph.D. degree in Computer Science and M.Sc. degree in Information Systems Security from Concordia University, Montreal, Canada. He is currently an Affiliated Research Associate in the Concordia Institute for Information Systems Engineering (CIISE). His main research interests include data privacy, cryptology, application security, and network security.



Lingyu Wang is an Associate Professor in the Concordia Institute for Information Systems Engineering (CIISE) at Concordia University, Montreal, Quebec, Canada. He received his Ph.D. degree in Information Technology from George Mason University. His research interests include data privacy, network security, security metrics, cloud computing security, and malware analysis. He has co-authored over 100 refereed publications on security and privacy.