Privacy-Preserving for Dynamic Real-Time Published Data Streams Based on Local Differential Privacy

Wen Gao and Siwang Zhou

Abstract—**Real-time data collected from users can help various applications provide services, but there is a risk that sensitive information will be leaked. Existing local differential privacy (LDP)-based approaches mainly perturb each data point, which severely affects the time-series patterns, leading to sensitive information being leaked out from multiple consecutive significant patterns. In this article, we focus on dynamic data streams under honest but curious servers and propose a privacy-preserving method called PP-LDP to protect data privacy while preserving data stream patterns and improving the utility of published data. To this end, our approach consists of three main parts. First, we sample the points that can represent the data stream patterns by improving the popular least squares segmented linear fit method. Then, we use an adaptive budget allocation method to perturb the sampling points and provide *w*-event level privacy. Finally, we perform post-processing optimization of the data streams with Kalman filters to further improve the utility of the data streams. Extensive experimental results on realistic data sets show that our proposed scheme can not only protect the data streams' privacy but also effectively preserve patterns and guarantee the utility of private data.**

Index Terms—**Local differential privacy (LDP), linear fit, post-process, real-time published data streams, time series patterns.**

I. Introduction

In recent years, with the increase in the number and computing power of smart devices (such as mobile phones, wearable devices, household appliances, and cars), service providers can continuously collect data with time series (data streams) and aggregate them to analyze relevant information and then provide corresponding services (personalized services and public services) [1], [2], [3]. For example, wearable devices can predict the user's physical health by analyzing their heart rate, sleep quality, etc. [4]. GPS service providers gather users' location, speed, mobility, and other data for traffic monitoring to provide route planning [5]. Although these personalized services simplify consumers' lives, they also pose serious privacy leakage risks. Due to the time-dependent nature of the data stream, attackers can infer relevant information about users based on continuous data, resulting in sensitive personal information being leaked during the release process. For example, a smart ammeter's power consumption will reveal whether or not the user is at home and the usage of household appliances [6]. If social networks continue to publish the number of users on the current topic, advertisers will be able to reach a more specific audience [7]. Therefore, it is important to protect the privacy of data streams.

Recently, differential privacy (DP) [8] has been widely used as a framework with strong privacy guarantees in privacy-preserving. DP makes the supposition that the server is trustworthy. Actually, the server will pry into the user's sensitive information due to curiosity or commercial interests, which leaks privacy. To overcome the risk of privacy leakage caused by the server, local DP (LDP) [9] was proposed and is widely used. This protocol allows users to perturb data locally before uploading it, and users can guarantee their data privacy without the trust of any third party. The study of perturbation mechanisms and the study of statistical data release are the two main research areas of LDP. The research on perturbation mechanisms mainly includes random response [10], information compression, and distortion [11]. The research on statistical data release mainly includes frequency statistics [12] and mean value statistics [13].

However, the current LDP-based privacy-preserving methods in the data stream randomly perturb data points [14]. Although these methods protect the privacy of the data stream, they are unable to effectively preserve the patterns, which will reduce the private data utility, resulting in the service providers not being able to analyze the effective information based on the data stream and affecting the effectiveness of the corresponding services. Wang et al. [15] first proposed the PatternLDP approach to protect privacy and preserve patterns by only perturbing some remarkable points (peaks, valleys, and breakpoints). However, we found that PatternLDP has three main limitations. First, it uses the piecewise linear approximation (PLA) method to normalize the data stream and then selects the farthest remarkable point as the sampling point within a fixed fault tolerance threshold. However, this sampling method requires the constant redefinition of the normalization results, which violates the real-time and one-time requirements of dynamic data streams and is not applicable to real-time dynamical data streams. As shown in Fig. 1, for the Hong Kong Hang Seng index (HKHS) [16],

Manuscript received 1 April 2023; revised 14 August 2023 and 23 October 2023; accepted 22 November 2023. Date of publication 29 November 2023; date of current version 9 April 2024. This work was supported in part by the National Science Foundation of China under Grant 62172153. *(Corresponding author: Siwang Zhou.)*

The authors are with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China (e-mail: wgao@hnu.edu.cn; swzhou@hnu.edu.cn).

Digital Object Identifier 10.1109/JIOT.2023.3337397

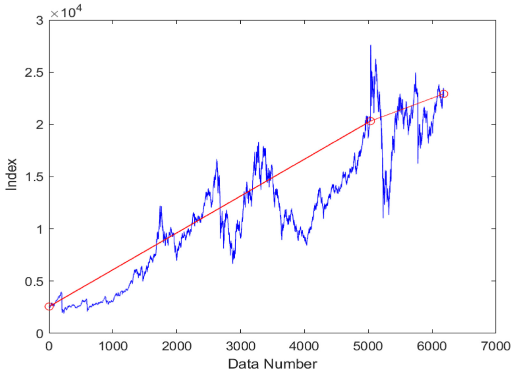


Fig. 1. PatternLDP only perturbs a small number of points in the HKHS, which does not provide privacy-preserving for this data set.

the data stream itself has certain trends. Since the PLA presets a fixed threshold in advance, resulting in only a very few points being sampled, perturbing these points is not enough to provide privacy to the data stream. Second, the method's consideration of the dynamic real-time data stream pattern in the budget allocation procedure was limited to the speed of data stream fluctuations, ignoring the trend of data stream fluctuations. Actually, the trend of the fluctuation is an important indicator for characterizing the dynamic real-time data stream pattern. Finally, preserving data stream patterns violates privacy to some extent; the authors did not utilize post-processing optimization during the process of balancing privacy and utility, which results in less-than-ideal accuracy of the output data. Therefore, a new, more effective privacy-preserving method has to be developed. This method should enable users to perturb data locally before real-time uploading it and meet the real-time and one-time requirements of the dynamic data stream. Which can effectively preserve patterns and achieve a balance between privacy and utility.

In this article, we focus on privacy-preserving in real-time dynamic published data streams on honest but curious servers, aiming to protect user privacy, improve private data utility, and preserve the data stream patterns. To achieve the above goals, we mainly face two challenges: The first challenge is how to adopt a suitable sampling method to select the remarkable points that can effectively represent the dynamic real-time data stream patterns. Since the fluctuation trend and speed of the dynamic real-time data streams are unknown and unpredictable, random sample perturbations can modify the pattern of the data stream. It is difficult to effectively select points that represent patterns without knowing the full picture of the data stream. The second challenge is to effectively balance the privacy-preserving and utility. Since the preservation of data stream patterns violates privacy to some extent, a smaller budget would provide better privacy but, in turn, reduce utility and lose more patterns. It is difficult to ensure data utility and preserve patterns while protecting privacy.

To address the above challenges, this article proposes an LDP-based privacy-preserving method for dynamic real-time data stream patterns called LDP-based pattern preserve (PP-LDP), which aims to protect data stream privacy while improving data utility and preserving data stream patterns. Specifically, for the first challenge, we employ the adaptive remarkable point sampling approach, which uses the dynamic real-time data stream's fluctuation trend and fluctuation speed to adaptively sample if the current remarkable point can accurately represent the pattern. This method can be applied to dynamic real-time data streams without knowing the global view. It also does not necessitate normalization. For the second challenge, we employ an adaptive budget allocation approach that focuses on protecting the privacy of remarkable points that have a greater impact on the patterns and provides w-event [17] privacy-preserving. Based on this, we use the property of post-processing immunity [39] to introduce Kalman filters for post-processing optimization of data streams to ensure the utility of the data stream.

The main contributions of this article are as follows.

1) Considering the purpose of preserving the patterns, we use the adaptive sampling method to sample the remarkable points that can effectively represent the patterns and can be applied to any dynamic real-time data stream.
2) Considering the mutual constraints of privacy-preserving level and data utility, we adaptively perturb the sampling points according to the trend and speed of data stream fluctuations. And then, we utilize the property of post-processing immunity to filter the privacy data in order to improve its utility and preserve useful patterns.
3) We compare our approach with other existing methods using real-world data sets, and the results show that it not only provides a good level of privacy-preserving but also effectively preserves the data stream patterns.

The rest of this article is organized as follows. Section II reviews related work on privacy-preserving in the data stream in recent years. Section III introduces our system model and gives related theoretical definitions. Section IV gives the design of PP-LDP and its privacy analysis in detail. Section V presents the related experimental evaluations. Finally, Section VI summarizes our work.

II. Related Work

Nowadays, protecting privacy has become a growing concern, and a large number of privacy-preserving related studies have emerged in different fields. For example, improving vehicular services and enhancing the driving experience in a secure and privacy-preserving manner in intelligent transportation systems (ITSs) [18]. There is a need to limit the data security and privacy leakage caused by untrusted third parties in the IoT [19]. To avoid the privacy and security concerns related to forged faces in bioinformatic authentication, facial information should be protected [20]. In this article, we focus on the work related to privacy-preserving data streams by DP. Many approaches have been proposed to protect time-series data privacy. We first review privacy-preserving methods for time series data, then introduce LDP-based privacy-preserving methods, and finally introduce existing LDP-based privacy-preserving levels.

Time-Series Data Privacy Preserving: According to different requirements and goals, privacy-preserving data streams can be divided into two categories: 1) privacy-preserving toward aggregated statistical analysis and 2) privacy-preserving toward time-series analysis. For the first objective, Benhamouda et al. [21] used a variant of the smooth projective hashing paradigm to construct privacy-preserving encryption schemes for time series aggregation. Song and Chaudhuri [22] investigated a form of inferential privacy called Pufferfish, which is applied to protect the aggregation properties of time series data. For the second objective, Zheng et al. [23] proposed a privacy-preserving time series similarity range query scheme to resist cloud inference attacks. Liu et al. [24] combined cryptography and data mining domain techniques to ensure the accuracy of privacy analysis of time series. The above privacy-preserving methods mainly use encryption to protect time series without considering DP with strong privacy guarantees. Literature [25] evaluated various methods such as generalization, hiding, and perturbation in time series privacy-preserving, and the final results proved that DP methods have better accuracy.

LDP in Data Streams: Dwork et al. [8] first introduced the concept of DP. LDP [9] was proposed and widely used to overcome the threat of leaking sensitive information from untrusted servers in DP. This protocol allows users to perturb data locally. Wang et al. [26] proposed a method for determining thresholds using an exponential mechanism with a mass function that approximates the utility objective well while maintaining low sensitivity. Ren et al. [27] designed a baseline method for infinite streams that avoided the high sensitivity of LDP noise to budget partitioning. The method then used the population division framework and the sparsity of the data stream to improve accuracy. Wang et al. [28] adopts the DP with an improved Kalman filter to protect the data stream, which improves the privacy-preserving level of the data stream. Xue et al. [29] proposed the dynamic difference reporting mechanism (DDRM), which is a new scheme with stronger privacy guarantees for continuous frequency estimation under LDP. They also applied an optimal privacy budget allocation scheme to improve the accuracy of the estimation. Gu et al. [30] developed a data perturbation mechanism based on LDP that can both support aggregate and single queries and has high utility. All of the above approaches used LDP to protect the privacy of the data stream, but none of them considered the potential loss of privacy brought by continuous patterns in the real-time dynamic published data stream. While [31] studied metric-based time series purification that preserves the unique pattern of time series, the approach only considered statically published data streams and is not applicable to dynamic real-time published data streams. Therefore, a privacy-preserving method that preserves the unique patterns of real-time dynamic data streams is needed.

Privacy-Preserving Level: Privacy-preserving in data streams initially focused on two concepts: 1) user-level privacy and 2) event-level privacy [32]. Xu et al. [33] introduced user-level LDP to process multidimensional analytical queries in order to formalize and protect the privacy of users when data tuples joined by users are published. Perrier et al. [34] truncated data streams with contribution limitation techniques to improve utility and provide event-level privacy. However, in dynamic real-time published data streams, user-level privacy can lead to increased privacy losses over time, and event-level privacy can lead to increased privacy losses over time. The concept of *w*-event privacy was first proposed by Kellaris et al. [17] to balance user-level and event-level privacy. Ren et al. [35] proposed a new average *w*-event privacy concept and an infinite stream privacy-preserving scheme based on Lyapunov optimization, which guarantees data privacy and high utility. Ye et al. [36] proposed the privacy-preserving method of the data stream based on LDP, which improves data privacy and utility while satisfying the *w*-event privacy requirement. Errounda et al. [37] used a sliding window approach based on the *w*-event privacy technique to address the issue of publishing location statistics with LDP over multiple time stamps. The method guarantees the utility of statistical data while preserving the privacy of the location. In order to minimize privacy loss and improve data utility, privacy-preserving schemes for dynamic real-time data streams should satisfy *w*-event privacy.

TABLE I
PARAMETERS TABLE

$S[i], v[i]$	the i$^t h$ data point and its corresponding value
α	the slope feasible domain
k	the slope of the fitted line
K_p, K_s, K_d	the proportional coefficients, the integral coefficients, and the differential coefficients, respectively
$p_k, p_\gamma, p_{k\gamma}$	the percentage of budget allocation based on the magnitude of fluctuations, fluctuation rate, and potential sampling points, respectively
p_γ	the percentage of budget allocation based on the fluctuation rate
$\widehat{x}_t, \widetilde{x}_t, \widetilde{x}_t$	the posterior estimates, measured values, and predicted values of the data in time t, respectively
k_t	the Kalman gain
Q	the covariance of the system process
$\widetilde{P}_t$	the covariance matrix between the true value and prior estimate in time t
$\widehat{P}_t$	the covariance matrix between the true value and posterior estimate in time t
R	the covariance of measurement noise

Based on the related work reviewed above, our work will meet the real-time and one-time requirements of the dynamic data stream by satisfying *w*-event, which allows users to perturb the data locally before uploading it in real-time. Our work reduces privacy loss and improves data utility while effectively preserving the unique patterns of dynamic real-time published time series.

III. System Model and Problem Definition

In this section, we introduce the system model. And then, we give the relevant theoretical definitions required in this article to facilitate the measurement of the degree of privacy-preserving and utility. As shown in Table I, we list the important symbols and their meanings.

A. System Model

The real-time data stream published model mainly includes two entities: 1) the user and 2) the data service provider (DSP). In this article, we assume that the DSP is honest but curious. Although the DSP will honestly perform various operations on the data, it will still pry into the user's sensitive information out of curiosity or for other purposes, resulting in the leakage of privacy. Our system model is shown in Fig. 2. First, each user will generate various data in their daily lives, and smart devices will collect the data in real time. Since the DSP is

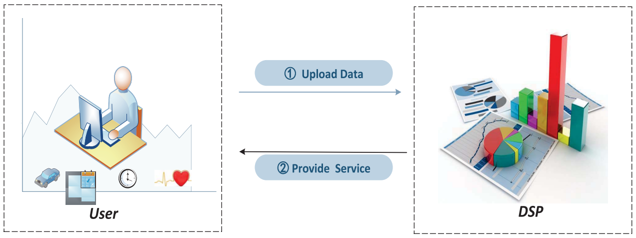


Fig. 2. System model.

untrusted, users will perturb the data locally to protect their privacy and then upload the private data to the DSP to prevent the DSP from inferring sensitive information. Second, the DSP aggregates the perturbed data and then performs data analysis on the aggregated data, such as summing the data, averaging the data, etc. And then, the DSP provides more appropriate services to users based on the results of the data analysis.

B. Privacy Definition

In this section, we will introduce the background knowledge involved in this article, including time series (data stream), LDP, w-event privacy, and Kalman filter. The specific related definitions are as follows:

Definition 1 (Time Series): The time sequence S is an ordered time point sequence, $S = \{s[1], s[2], \dots\}$, where each data point $s[i]$ consists of a corresponding tuple $(i, v\{i\})$, where i denotes the timestamp.

Definition 2 (Pattern): A pattern P in time series can be defined as a sequence of limited data points (e.g., $P = s[i], s[i+1], \dots, s[i+k-1]$), which can characterize some meaningful trends of time series.

Definition 3 (Least Squares Fit Method [38]): A set of 2-D data points $(x_i, y_i)(i = 1, 2, \dots, n)$, its empirical equation can be described by $F(x)$. The equation always contains some pending coefficients a_n. Substitute x_i into the equation to find the difference $y_i - F(x_i)$. When the overall error $Q = \sum(y_i - F(x_i))^2$ is the smallest, the best-fit function of the set of data can be found.

Definition 4 (ϵ-LDP [9]): Given a privacy algorithm $M(\cdot) : \mathcal{D} \longrightarrow \widetilde{\mathcal{D}}$, where $\mathcal{D}$ denotes the domain of all possible inputs, $\widetilde{\mathcal{D}}$ denotes the domain of all possible outputs. $M(\cdot)$ satisfies ϵ-local differential privacy, if and only if for any input $v_1, v_2 \in \mathcal{D}$,has

$$\forall T \subseteq \widetilde{\mathcal{D}} : \Pr[M(v_1) \in T] \leq e^{\epsilon} \times \Pr[M(v_2) \in T]. \quad (1)$$

w-Event Privacy: The w-event is a privacy model proposed for infinite streams that provides privacy guarantees for any time series that occurs in any continuous w timestamps. It expands event-level privacy and balances utility and privacy. Before we introduce the concept of w-event privacy, we first introduce the concept of neighboring stream prefixes.

Definition 5 (w-Neighboring [17]): Let w be a positive integer. Two stream prefixes $[S]_t$, S'_t are w-neighboring, if:

1) for each $S_t[i]$, ${S'}_t[i]$ such that $i \in [t]$ and $S_t[i] \neq {S'}_t[i]$ it holds that $S_t[i]$, ${S'}_t[i]$ are neighboring;
2) for each $S_t[i_1], S_t[i_2], {S'}_t[i_1], {S'}_t[i_2]$ with $i_1 < i_2, S_t[i_1] \neq S_t[i_2]$ and ${S'}_t[i_1] \neq {S'}_t[i_2]$, it holds that $i_2 - i_1 + 1 \leq w$.

Among them, S_t denotes the neighboring stream prefixes of S at time t, $S_t = \{s_t[1], s_t[2], \dots\}$. According to Definition 5, if S_t, ${S'}_t$ are w-neighboring, then: 1) their elements are pairwise the same or neighboring and 2) all their neighboring databases can fit in a window of up to w timestamps. It can be deduced that the relevant definition of w-event is as follows:

Definition 6 (w-Event [17]): Let M be a mechanism that takes as input a stream prefix of arbitrary size. Also let $\mathcal{O}$ be the set of all possible outputs of M. We say that M satisfies w-event ϵ-differential privacy (or, simply, w-event privacy) if for all sets $O \subseteq \mathcal{O}$, all w-neighboring stream prefixes S_t, ${S'}_t$, and all t, it holds that

$$\Pr[M(S_t) \in O] \leq e^{\epsilon} \times \Pr\left[M\left({S'}_t\right) \in O\right]. \quad (2)$$

Post Processing [39]: Given an algorithm A that satisfies ϵ-DP, $g(A(X))$ still satisfies ϵ-DP for any g release. That is, post-processing the output of the DP algorithm does not result in any additional privacy loss.

Kalman Filter [40]: Filtering refers to the derivation of posterior estimates based on a series of noise measurements in the hope of removing noise from the signal. Kalman filtering is mainly divided into two steps: 1) prediction and 2) update.

Prediction: The state of the present time (time t) is estimated based on the posterior estimated value (updated value) of the preceding time (time $t-1$), and the prior estimate (predicted value) at time t is derived. The process is

$$\widetilde{x}_t = A\widehat{x}_{t-1} + Bu_{t-1} \quad (3)$$
$$\widetilde{P}_t = A\widehat{P}_{t-1}A^T + Q \quad (4)$$

where $\widehat{x}_{t-1}$ is the filtered result, also called the best estimate. A denotes the state transition matrix, and B denotes the input control matrix. u_{t-1} denotes the external operation at time $t-1$. Q is the covariance of state transitions.

Update: The filter uses the measured values at the current time to correct the prior estimates in the prediction step to obtain the posterior estimates at the current time. The process is

$$K_t = \frac{\widetilde{P}_t H^T}{H\widetilde{P}_t H^T + R} \quad (5)$$
$$\widehat{x}_t = \widetilde{x}_t + K_t(\overline{x}_k - H\widetilde{x}_t) \quad (6)$$
$$\widehat{P}_t = (1 - K_t H)\widetilde{P}_t \quad (7)$$

where H denotes the transformation matrix of state variables. z_k denotes the measured value.

Therefore, the research problem in this article is formulated as follows. Assuming the time series is $S = \{s[1], s[2], \dots\}$, the objective is to obtain a private and filtered time series $S^* = \{s^*[1], s^*[2], \dots\}$ through LDP and Kalman filtering that can be sent to an untrusted DSP in real time. The pattern and utility of S^* are still guaranteed while satisfying the privacy requirements of LDP.

IV. Privacy-Preserving Mechanism

In this section, we first describe the process of PP-LDP, then give the specific steps, and finally, we perform the corresponding privacy analysis.

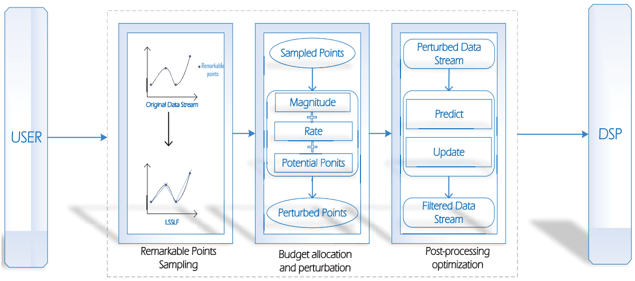


Fig. 3. PP-LDP process.

A. Process Overview for PP-LDP

This article aims to preserve data stream patterns while preserving data stream privacy and further improving data stream utility. As shown in Fig. 3, PP-LDP consists of three main parts: 1) remarkable point sampling; 2) budget allocation and perturbation; and 3) post-processing optimization. First, we use the LSSLF to sample remarkable points that can effectively represent the pattern. Second, according to the fluctuation magnitude, fluctuation rate, and number of potential sampling points, we adaptively assign budgets to the sampling points and obtain the perturbed data stream. Finally, according to the post-processing optimization, the perturbed data stream is predicted and updated with the Kalman filter to improve accuracy.

B. Remarkable Points Sampling

In this section, we propose a method based on the improved LSSLF to sample remarkable points that effectively represent the data stream pattern. The essential idea behind the method is to first use least squares to find a mathematical representation of the fitted line, and then determine whether the upcoming points can be fitted to the previous points based on the feasible slope domain. The method fits as many data points as possible onto the best straight line. The method can fit the fluctuation pattern of the data stream well in order to sample the remarkable points that are important to the pattern. Moreover, the method only needs to judge the current data, which is in line with the one-time and real-time characteristics of real-time dynamic data streams and has general applicability.

1) Remarkable Points: The commonly used method is using the first-order difference method to determine the remarkable points. As long as the point at time $t+1$ is known, this method can determine whether the point at time t is a remarkable point or not, which is consistent with the properties of real-time data streams. The steps are approximated as follows. First, the first-order derivative corresponding to each data point in the published data stream $S = \{s[1], s[2], \ldots\}$ is represented by the difference between that data and the latter data, i.e., $\text{Diff}_{s[i]} = w[i+1] - v[i]$. The remarkable point property has the following representation:

$$\begin{cases} \text{Diff}_{s[i]} \geq 0\&\&\text{Diff}_{s[i-1]} < 0 \\ \text{Diff}_{s[i]} > 0\&\&\text{Diff}_{s[i-1]} \geq 0 \\ \text{Diff}_{s[i]} \leq 0\&\&\text{Diff}_{s[i-1]} > 0 \\ \text{Diff}_{s[i]} < 0\&\&\text{Diff}_{s[i-1]} \leq 0. \end{cases} \tag{8}$$

When $s[i]$ satisfies any of the above conditions, it means that the point is a remarkable point in the published data stream. Correspondingly, when the slope of the line connecting $s[i]$ with $s[i-1]$ and $s[i+1]$ changes, $s[i]$ is a remarkable point. Corresponding to the fitted line, when the slope of the fitted line changes positively or negatively at point $s[i]$, it means that $s[i]$ is a remarkable point after fitting and can be chosen as a sampling point.

2) Linear Fit: The linear fitting equation in LSSLF can be expressed as $F(x) = kx + b$. The least-squares fit model is geometrically the one that minimizes the orthogonal distance from the actual data points to the fitted line, and the fitting criterion is the sum of the squares of the orthogonal distances from all points to the fitted line. Therefore, the corresponding sum of squared orthogonal distances can be obtained as $Q = \sum_{i=1}^{n}(y_i - kx_i - b)^2$. According to the fitting criterion, (9) can be obtained for the parameters k and b

$$\begin{cases} \sum_{i=1}^{n} 2(y_i - kx_i - b) \times (-x_i) = 0 \\ \sum_{i=1}^{n} 2(y_i - kx_i - b) \times (-1) = 0. \end{cases} \tag{9}$$

According to (9), we can obtain the expression

$$\begin{cases} k = \frac{2\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \times \sum_{i=1}^{n} y_i}{2\sum_{i=1}^{n} (x_i)^2 - \sum_{i=1}^{n} x_i \times \sum_{i=1}^{n} x_i} \\ b = \frac{1}{2}\left(\sum_{i=1}^{n} y_i - k\sum_{i=1}^{n} x_i\right). \end{cases} \tag{10}$$

Our fitting process is basically as follows when combined with the least squares method: First, we select the two adjacent remarkable points, $A(a, v[a])$ and $B(b, v[b])$. Then, (9) is combined with (10). One can thus get the fitted line $y = kx+b$, where the slope is $k' = (v[b] - v[a])/(b - a)$. For the next remarkable point $C(c, v[c])$, the equation of the line connecting points B and C is represented as $y' = kx' + b$, where the slope is $k = (v[c] - v[b])(c - b)$. We assume from the slope feasible domain when the angle between y and y' does not exceed α, i.e.,

$$\tan\theta = \left|\frac{k - k'}{1 + kk'}\right| \leq \tan\alpha. \tag{11}$$

That is, it can be considered that C can be fitted with A and B. The optimal fitted line to the three points of ABC can then be discovered using (9) and (10). Once the angle between the most recent fitted line and the newly added point P exceeds α or the angle is obtuse, a new linear fitting process begins with point P as the starting point.

Two adjacent fitted lines with the same trend may occasionally appear in fitted results acquired directly using the above method. Point P is not suitable as a point to represent the pattern at this time.

The fitted process is shown in Algorithm 1. The fitted line is first updated based on the initial point (line 3). Then the adaptive slope feasible domain is calculated according to (11)–(13) (lines 8 and 9), where x_j denotes the timestamp corresponding to the jth remarkable point. Then the process determines whether the next point can be fitted to the previous point. Once the slope exceeds the feasible domain, it outputs the current fitted line and the positivity and negativity of the slope. After that, the endpoint of the fitted line is used as the initial point of the new fitted line (lines 10–18). When the trend of two adjacent fitted lines is the same, the points

Algorithm 1: Fitted Process

```
Input: angle threshold $\tan\alpha$
Output: fitted liner $y = kx + b$
1  Initialize $i = 1$;
2  for $j \in [i + 1, n]$ do
3      Calculate $k$ by using Eq. (9), (10);
4      if $j + 1 > n$ then
5          return $y = kx + b$;
6          break;
7      end
8      Calculate $k' = \frac{v[j+1]-v[j]}{x_{j+1}-x_j}$;
9      Calculate $\tan\theta = \left|\frac{k-k'}{1+kk'}\right|$;
10     if $\tan\theta \leq \tan\alpha$ then
11         $j \leftarrow j + 1$;
12     else
13         $y = kx + b$;
14         $KS(j) \leftarrow sign(k)$;
15         $i_1 \leftarrow i$;
16         $i \leftarrow j$;
17         $j \leftarrow j + 1$;
18         $i_2 \leftarrow i$;
19         if $KS(i_1) == KS(i_2)$ then
20             Calculate $k$ between points $i_1$ to $i_2$ by using
               Eq. (9), (10);
21             return $y = kx + b$;
22         end
23     end
24 end
25 end
```

contained in the two fitted lines are refitted to create the new fitted line (lines 19–22).

3) Adaptive Angle Threshold α: For computational convenience, we denote α in Algorithm 1 by $\alpha = (\lambda\pi/2)$, where λ denotes the change ratio parameter of the angle. The fluctuation of the data stream determines the value of α. The k in the fitted linear model represents the fluctuating trend of the data stream. It is not enough to only consider the trend. The rate of the data stream's fluctuation is typically taken into account while describing it. Therefore, we introduce PID control to express the fluctuation rate of the data stream.

PID control is a common form of feedback control [41], which is often applied to represent the fluctuation rate. This article uses the PID to measure the dynamic performance of the data stream at the sampling point. The calculation is as

$$\gamma(t_i) = K_p e(t_i) + \frac{K_s}{\pi}\sum_{i=n-\pi-1}^{n} e(t_i) + \frac{K_d}{t_i - t_{i-1}}\left[e(t_i) - e(t_{i-1})\right] \tag{12}$$

where K_p, K_s, and K_d are the standard scale factors of PID, K_p represents the percentage of proportional error of the current dynamic data, K_s represents the percentage of past cumulative errors in the dynamic data, and π denotes the number of errors in the cumulative integration error. K_d represents the percentage of dynamic errors expected for future data. The feedback value $e(t_i)$ denotes the error between the fitted and actual values and is expressed as $e(t_i) = |v(t_i) - v'(t_i)|$. t_i denotes the time stamp corresponding to the ith sampling point. We use the fitted value as the predicted value $v'(t_i)$.

The risk of privacy leakage becomes higher in dynamic data streams when: 1) when $k > 0$, k is larger, it means that the rise of the data stream is larger and the risk of privacy leakage is higher, and α needs to be reduced; 2) when $k < 0$, k is smaller, which means that the decrease of the data stream is larger and the risk of privacy leakage is higher, and α needs to be reduced; and 3) when $\gamma(t_i)$ is smaller, the data stream fluctuates slowly and the interval to the next sampling point is larger, indicating that the risk of privacy leakage is higher and α needs to be reduced. According to $\alpha = (\lambda\pi/2)$, the change of λ causes the change of α. Therefore, α can be changed by dynamically changing λ. Since λ should be limited to the range $(0, 1)$. Therefore, we still choose the exponential function to determine the value of λ. Based on the above requirements, λ can be defined as

$$\lambda = 1 - \exp\left(-\left(\frac{1}{|k|} + \gamma(t_i)\right)\right). \tag{13}$$

Thus, depending on the trend and rate of the data stream, we can dynamically change the angle threshold to adaptively change the sampling interval to balance privacy and utility.

C. Budget Allocation and Perturbation

This section proposes an adaptive budget allocation method. In the process of budget allocation, the method considers both the dynamic changes of data streams near the sampling points and the number of potential sampling points. To provide stronger w-event privacy, the privacy budget cannot exceed ϵ per w timestamp.

1) Remaining Budget: The total maximum budget that can be allocated to the current sample point is related to the budget consumed by the prior $w-1$ timestamps due to the restriction of w-event privacy. Therefore, the budget available for the current sampling point can be expressed as

$$\epsilon' = \epsilon - \sum_{j=i-w+1}^{j-1} \epsilon_j \tag{14}$$

where ϵ denotes the total number of budgets. The budget available for the current sampling point is the total budget minus the budgets of the remaining points within w timestamps.

2) Budget Allocation: Definition 4 states that when the budget increases, the perturbation decreases and the utility of data improves, but it will affect the degree of privacy preserving. Therefore, when allocating the budget to the sampling points, it is necessary to consider how to balance the degree of privacy preserving and utility. However, in the actual budget allocation, ϵ' cannot be fully allocated to the current data point. Since: 1) the remarkable points have a greater effect on patterns with a large magnitude than on patterns with a small magnitude; 2) the remarkable points have a greater impact on the short-term pattern than on the long-term; and 3) it is needed to ensure that potential sampling points have a sufficient budget to satisfy w-event privacy. Next, we will explain how to adaptively allocate the budget ϵ' on the above three aspects.

Data Stream Magnitude: When slop $k > 0$, a larger value of k indicates that the current sampling point is experiencing rapid growth. At this point, it has a greater impact on the pattern, better utility needs to be guaranteed, and therefore a larger budget needs to be allocated. And when $k < 0$, a smaller value of k indicates that the current sampling point experiences a rapid decline, which has a greater impact on the pattern and requires more budget allocation. Therefore, depending on the magnitude of fluctuations in the data stream, the budget can be allocated according to the following principles:

$$p_k = \begin{cases} 1 - e^{-k}, k \geq 0 \\ 1 - e^{k}, k < 0. \end{cases} \tag{15}$$

Data Stream Fluctuation Rate: The larger $\gamma(t_i)$ in (12) proves that the data stream is experiencing rapid fluctuations, which has a greater impact on the pattern and requires a better privacy budget. In this way, the proportion of the privacy budget can be described as

$$p_\gamma = 1 - e^{-\gamma(t_i)}. \tag{16}$$

Number of Potential Sampling Points: According to the aforementioned requirements, when one of the following three scenarios occurs: 1) $k > 0$, the larger k; 2) $k < 0$, the smaller k; or 3) the greater γ; the higher the likelihood of sampling points within the w window, it is required to provide enough budget for potential sampling points. The percentage of the privacy budget that is allocated to the current point can be defined as

$$p_{k\gamma} = \begin{cases} 1 - \exp\left(-\frac{1}{k\gamma(t_i)}\right), & k \geq 0 \\ 1 - \exp\left(-\frac{k}{\gamma(t_i)}\right), & k < 0. \end{cases} \tag{17}$$

With this strategy, we have two budget allocation requirements: one is to ensure sensitivity to sampling points' significance, and the other is to ensure that potential sampling points have sufficient budgets. Therefore, we must balance the above budgets and ensure that the budget ratio is between 0 and 1. To this end, the proportional budget allocation is defined as

$$p = 1 - \exp\left(-\left(p_k + p_\gamma\right)/p_{k\gamma}\right)$$
$$p = \begin{cases} 1 - \exp\left(-\left(2 - \left(e^{-k} + e^{-\gamma(t_i)}\right)\right)/\left(1 - e^{-\frac{1}{k\gamma(t_i)}}\right)\right), k \geq 0 \\ 1 - \exp\left(-\left(2 - \left(e^{k} + e^{-\gamma(t_i)}\right)\right)/\left(1 - e^{-\frac{k}{\gamma(t_i)}}\right)\right), k < 0. \end{cases} \tag{18}$$

In this way, the privacy budget being assigned to the current point can be expressed as $\epsilon_i = p\epsilon'$.

3) Perturbation Mechanism: To reduce the perturbation and improve the accuracy, we introduce the SW [42] mechanism to perturb the sampling points. It extends the random response perturbation under traditional LDP; values that are close to the true value will be reported with high probability, while those far from it will be reported with low probability.

First, we define the "close" measure. When the output data $\widetilde{v}[i]$ is in the range of $\widetilde{v}[i] \in [v[i] - b[i], v[i] + b[i]]$, it is considered "close" and the true result will be reported with a high probability p_i. SW selects a value of $b[i]$ that is independent of distribution and only correlated with ϵ_i in order to make $b[i]$ perform well in any distribution. This value is defined as

$$b[i] = \left(\epsilon_i e^{\epsilon_i} - e^{\epsilon_i} + 1\right)/\left(2e^{\epsilon_i}\left(e^{\epsilon_i} - 1 - \epsilon_i\right)\right). \tag{19}$$

Equation (19) shows that $b[i]$ is a function that decreases with ϵ_i. Assume that $b[i]$ tends to 0 when ϵ_i goes to ∞. $b[i]$ is only $(1/2)$ when ϵ_i is 0, which results in an output domain that is at most twice the size of the input domain. Thus, for each input value, the output value can be considered as "close." By maximizing the difference between p_i and q_i, while satisfying the total probability, which adds up to 1, we can derive the probability formula as

$$\begin{cases} p_i = e^{\epsilon_i}/(2b[i]e^{\epsilon_i} + 1), & \text{if } |v[i] - \widetilde{v}[i]| \leq b[i] \\ q_i = (2b[i]e^{\epsilon_i} + 1)^{-1}, & \text{otherwise.} \end{cases} \tag{20}$$

Thus, given the input $v[i]$, we can report values closer to $v[i]$ with a higher probability than values far from $v[i]$. This ensures privacy while reducing perturbations.

In summary, if we allocate a sufficient budget for sampling points with a higher current fluctuation magnitude and a faster fluctuation rate, it ensures better utility but affects the degree of privacy preserving. Meanwhile, if the remaining budget is insufficient and the budget allocated to potential sampling points is too small, although their privacy-preserving degree is high, it will seriously affect utility and not be conducive to the analysis and usage of data. The adaptive budget allocation method takes into full consideration these factors, which can not only allocate appropriate budgets for current sampling points but also guarantee that potential sampling points have sufficient budgets, which well balances the utility of data streams and the degree of privacy-preserving.

D. Post-Processing Optimization

We consider the budget allocation in terms of both data stream dynamics and potential sampling points in Section IV-C. Although a smaller privacy budget ensures better privacy preservation, our approach adds more noise to the sampling points, which affects the data stream due to the mutual constraints on privacy preserving and utility. Due to the post-processing immune character of DP, we use Kalman filters to release posterior estimates to improve the accuracy of the perturbed data. The posterior estimates are obtained from both predicted values and measured values, as shown in

$$\widehat{x}_t = \widetilde{x}_t + K_t(\bar{x}_t - \widetilde{x}_t) \tag{21}$$

where $\bar{x}_t = x_t + \mathrm{Lap}((1/\epsilon_n))$. Since the data in the data stream varies naturally, there is no additional operational control and it can be seen as $A = 1$, $u_{t-1} = 0$. Combined with (3), we can see that $\widetilde{x}_t = \widehat{x}_{t-1}$. K_t is adjusted at each time stamp to minimize the posterior error variance, as shown in (21)

$$K_t = \widetilde{P}_t\left(\widetilde{P}_t + R_t\right)^{-1} \tag{22}$$

where $\widetilde{P}_t = E[(x_t - \widetilde{x}_t)(x_t - \widetilde{x}_t)^T]$ denotes the covariance of the a priori estimate. $\widetilde{P}_t$ at time t is related to the covariance $\widehat{P}_{t-1}$ of the posterior error at time $t-1$, $\widetilde{P}_t = \widehat{P}_{t-1} + Q$. The covariance of the optimal posterior estimate is $\widehat{P}_t = (1 - K_t)\widetilde{P}_t$. The process of post-processing optimization is shown by Algorithm 2.

Algorithm 2: Post-Processing Optimization

```
Input: perturbed value $\bar{x}_t$; published value of the previous moment $\widehat{x}_{t-1}$
Output: posterior estimates $\widehat{x}_t$
/* prediction process                                  */
$\widetilde{x}_t = \widehat{x}_{t-1}$;
$\widetilde{P}_t = \widehat{P}_{t-1} + Q$;
/* update process                                      */
$K_t = \widetilde{P}_t(\widetilde{P}_t + R_t)^{-1}$;
$\widehat{x}_t = \widetilde{x}_t + K_t(\bar{x}_t - \widetilde{x}_t)$;
$\widehat{P}_t = (1 - K_t)\widetilde{P}_t$;
```

E. Theoretical Analysis

In this section, we prove that PP-LDP satisfies ϵ-LDP and provides w-event privacy.

Theorem 1: PP-LDP makes the data point at timestamp i satisfies ϵ_i-LDP.

Proof: For two possible data points $s[i]$, $s'[i]$, their point values are $v[i]$, $v'[i]$, respectively. We write our mechanism as P in short to facilitate the proof, and the possible output domain is $T \subseteq \widetilde{\mathcal{D}}$. The output probability density function $P_v(\widetilde{v}) = \Pr[P(v) = \widetilde{v}]$ is

$$\begin{cases} P_{v[i]}(\widetilde{v}[i]) = q_i, \quad \text{for} \quad |v[i] - \widetilde{v}[i]| > b[i]; \\ \int_{-b[i]}^{b[i]} P_{v[i]}(\widetilde{v}[i]) d(v[i] - \widetilde{v}[i]) = 1 - q_i, \text{otherwise.} \end{cases} \tag{23}$$

Thus, for the possible output set $T \subseteq \widetilde{\mathcal{D}}$, there is $(\Pr[P(v[i]) \in T]/\Pr[P(v'[i]) \in T]) = (\Pr[P(v[i]) = \widetilde{v}[i]]/\Pr[P(v'[i]) = \widetilde{v}[i]]) = [(\int_{\widetilde{v}[i] \in T} P_{v[i]} (\widetilde{v}[i]) d(\widetilde{v}[i])) / (\int_{\widetilde{v}[i] \in T} P_{v'[i]} (\widetilde{v}[i]) d(\widetilde{v}[i]))]$, according to (20), one can get

$$\begin{aligned} \frac{\Pr[P(v[i]) \in T]}{\Pr[P(v'[i]) \in T]} &= \frac{\int_{\widetilde{v}[i] \in T} 1 - q_i d(\widetilde{v}[i])}{\int_{\widetilde{v}[i] \in T} q_i d(\widetilde{v}[i])} \\ &= \frac{\int_{\widetilde{v}[i] \in T} (2b[i]e^{\epsilon_i})/(2b[i]e^{\epsilon_i} + 1) d(\widetilde{v}[i])}{\int_{\widetilde{v}[i] \in T} (2b[i]e^{\epsilon_i} + 1)^{-1} d(\widetilde{v}[i])} \\ &= 2b[i]e^{\epsilon_i}. \end{aligned} \tag{24}$$

According to the analysis of (19), the range of $b[i]$ satisfies $b[i] \in (0, (1/2)]$. Thus, we can get $(\Pr[P(v[i]) \in T]/\Pr[P(v'[i]) \in T]) \leq e^{\epsilon_i}$. Therefore, PP-LDP satisfies ϵ_i-LDP. ■

Theorem 2: PP-LDP provides w-event privacy.

Proof: Since all mechanisms use independent randomness, the following contents for the time series S and any mechanism output $(r[1], \ldots, r[t]) \in R$. Here, our mechanism is written as P in short to facilitate the proof

$$\Pr[P(S) = (r[1], \ldots, r[t])] = \prod_{k=1}^{t} \Pr[P_k(s[k]) = r[k]]. \tag{25}$$

Similarly, for any w-neighboring stream prefixes S' and $(r[1], \ldots, r[t])$ have similar representations

$$\Pr\big[P(S') = (r[1], \ldots, r[t])\big] = \prod_{k=1}^{t} \Pr\big[P_k(s'[k]) = r[k]\big]. \tag{26}$$

According to Definition 5, it exits $i \in [t]$, such that $s[k] = r[k]$ for $i \leq k \leq i - w$ and $i + 1 \leq k \leq t$. Thus

$$\frac{\Pr[P(S) = (r[1], \ldots, r[t])]}{\Pr[P(S') = (r[1], \ldots, r[t])]} = \prod_{k=1}^{t} \frac{\Pr[P_k(s[k]) = r[k]]}{\Pr[P_k(s'[k]) = r[k]]}. \tag{27}$$

From Definition 5, $s[k]$, $s'[k]$ are neighboring with $i - w + 1 \leq k \leq i$. According to Theorem 1, our mechanism satisfies ϵ_i-LDP and, therefore, it can be deduced that

$$\begin{aligned} \frac{\Pr[P(S) = (r[1], \ldots, r[t])]}{\Pr[P(S') = (r[1], \ldots, r[t])]} &\leq \prod_{k=i-w+1}^{t} e^{\epsilon_k} \\ &= \exp\left(\sum_{k=i-w+1}^{i} \epsilon_k\right). \end{aligned} \tag{28}$$

Summing the probabilities $\Pr[P(S) = (r[1], \ldots, r[t])]$ on any R gives $(\Pr[P(S) \in R/\Pr[P(S') \in R]) \leq \exp(\sum_{k=i-w+1}^{i} \epsilon_k)$. Since $\sum_{k=i-w+1}^{i} \epsilon_k \leq \epsilon_i$, we can get $(\Pr[P(S) \in R/\Pr[P(S') \in R]) \leq e^{\epsilon_i}$. Thus, PP-LDP satisfies w-event privacy. ■

V. Performance Evaluation

This section evaluates the performance of the PP-LDP on three real-world time-series data sets. We evaluate three aspects: 1) evaluation of linear fitting methods to estimate the degree of preserving patterns; 2) evaluation of the utility of statistical estimation to estimate the impact of privacy-preserving methods on utility; and 3) evaluation of the utility of time series to evaluate the impact of privacy-preserving methods on preserving patterns in the data stream.

A. Evaluation Settings

1) Data Sets: The four real-world data sets are as follows.

Heart Rate Analysis (HRA) [43]: The data set contains heart rate time series data for a person during six days of normal activity, which has a total of 42 963 heart rate records.

HKHS [17]: The data set includes the daily Hang Seng Index from 1986 to 2013.

Electric Devices (EDs) [44]: The data set contains 16 637 electrical devices, which can be classified into seven categories, and each time series consists of 96 data points. We chose the subset EDTR(ElectricDevices_TRAIN) for simulations.

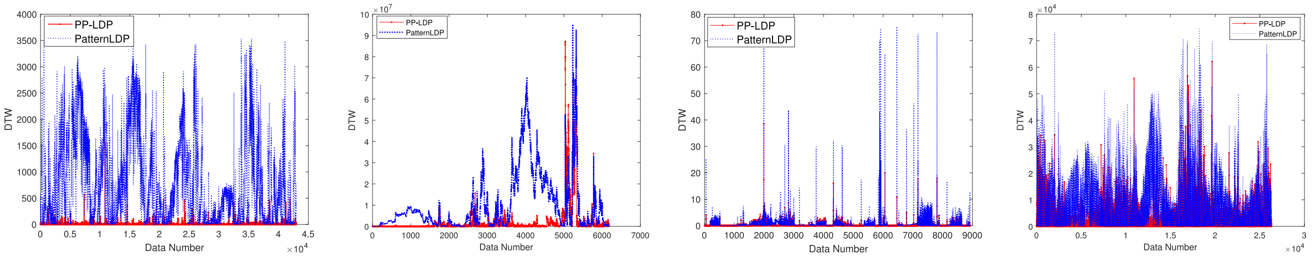
Electricity Consumption (EC) [45]: The data set contains EC in kWh recorded hourly between 2012 and 2014.

2) Compared Approaches: In this section, we not only compare the PP-LDP with PatternLDP but also compare it with the following three approaches.

LDP Budget Distribution (LBD) [27]: The method allocated the budget in the w window in an exponentially decreasing manner, and random perturbation values were responded to in the distribution $\widetilde{v}[i] \in [v[i] - (d - 2 + e^{\epsilon})/\big((e^{\epsilon} - 1)^2\big), v[i] + (d - 2 + e^{\epsilon})/\big((e^{\epsilon} - 1)^2\big)]$ with the following probability, where d denotes the scale of the selected value in the w window

$$\begin{cases} p = e^{\epsilon}/(e^{\epsilon} + d - 1), \widetilde{v}[i] = v[i]; \\ q = (e^{\epsilon} + d - 1)^{-1}, \widetilde{v}[i] \neq v[i]. \end{cases} \tag{29}$$

Piecewise Mechanism (PM) [46]: The method provided event-level privacy for data streams and returned a noise



(a) (b) (c) (d)

Fig. 4. Comparison of fitting methods. (a) HRA. (b) HKHS. (c) EDTR. (d) EC.

value interval of $(4/3)e^{(\epsilon/2)}/(e^{(\epsilon/2)}-1)^2$. The probability of perturbation is shown as

$$\begin{cases} p=\left(e^{\epsilon}-e^{\frac{\epsilon}{2}}\right)/\left(2e^{\frac{\epsilon}{2}}+2\right) \\ q=\left(e^{2\epsilon}-e^{\epsilon}\right)/\left(2e^{\frac{\epsilon}{2}}+2\right). \end{cases} \tag{30}$$

Duchi [47]: This method provided event-level privacy for data streams and defined the perturbation value $\widetilde{v}[i]$ as $[(e^{\epsilon}+1)/(e^{\epsilon}-1)]$ or $-[(e^{\epsilon}+1)/(e^{\epsilon}-1)]$. Its corresponding probability is shown in (31), where t_i denotes an input tuple that takes values in the range $t_i \in [-1,1]$

$$\begin{cases} p=t_i(e^{\epsilon}-1)/(2e^{\epsilon}+2)+\frac{1}{2}, x=\frac{e^{\epsilon}+1}{e^{\epsilon}-1} \\ q=-t_i(e^{\epsilon}-1)/(2e^{\epsilon}+2)+\frac{1}{2}, x=-\frac{e^{\epsilon}+1}{e^{\epsilon}-1}. \end{cases} \tag{31}$$

3) Indicators: We use dynamic time warping distance (DTW) to evaluate the fit similarity and mean-squared error (MSE) to evaluate the data utility.

DTW: DTW is used to measure the similarity of two sequences. Suppose two time series with length quantiles n, m, i.e., $Q=(q_1,q_2,\ldots,q_n)$, $C=(c_1,c_2,\ldots,c_m)$. The corresponding DTW distance is $\mathrm{DTW}(Q,C)=D(n,m)$. The mathematical expression is

$$\mathrm{dis}\left(q_i,c_j\right)=\left(q_i-c_j\right)^2$$
$$D(i,j)=\mathrm{dis}\left(q_i,c_j\right)+\min\begin{cases} D(i-1,j) \\ D(i,j-1) \\ D(i-1,j-1). \end{cases} \tag{32}$$

Mean Relative Error (MRE): MRS is used to measure the data series utility, the mathematical expression is as follows:

$$\mathrm{MSE}=\frac{1}{T}\sum_{i\in T}\left|\frac{\widehat{v}[i]-v[i]}{v[i]}\right|. \tag{33}$$

B. Performance Comparison

In our evaluation, we set the parameters $K_p=0.8$, $K_s=0.1$, $K_d=0.1$ in PID. In evaluating the privacy approach, we assume that the privacy budget is in the range of $\epsilon \in [0.1,1]$, and that the rest of the cases default $\epsilon=1$. In evaluating the privacy level, we assume that the range of the sliding window is $w \in [80,260]$, in the rest of the cases, the default is $w=160$. Evaluate the utility of methods in real-time dynamic data streams by gradually increasing the number of data points in each data set. In this section, we give the results of the performance comparison, and the results are the average of the results of 100 runs.

1) Comparison of Fitting Methods: As shown in Fig. 4, the DTW distances between the fitted line of sampling points and the original data stream obtained by the PatternLDP and PP-LDP, respectively, show the degree to which the two sampling methods preserve the pattern. It can be seen that the DTW distances obtained by PP-LDP in the four data sets are much smaller than those obtained by PatternLDP. The smaller the DTW distance, the closer the fitted line is to the original data stream, i.e., the sampling points selected by PP-LDP can better represent the pattern. In addition, according to Fig. 4(b), the DTW distances obtained by PatternLDP are much smaller than those obtained by PP-LDP in data streams that show a certain trend in general. This further indicates that the sampling method of PatternLDP has some limitations and cannot be widely applied to dynamic data streams. Therefore, we can conclude that the sampling points of PP-LDP can represent the pattern of the data stream more effectively, ensuring that the original pattern can be basically preserved even if these points are perturbed.

2) Comparison of Data Stream Utility: As shown in Fig. 5, the effect of the privacy budget on the MRE of the four data sets, i.e., the impact of the privacy budget on the utility. As the budget rises, it is clear that the MRE decreases. This is due to the fact that the perturbation decreases as the privacy budget rises. Among them, PP-LDP is always in a tiny steady state, which shows that the metric-based privacy statistics offered by this method are more stable and less noisy. PP-LDP takes advantage of post-processing to ensure accuracy in statistical analysis while ensuring privacy guarantees.

As shown in Fig. 6, the effect of the privacy budget on the similarity of data streams before and after privacy-preserving, i.e., the impact of the privacy budget on the preserved pattern. When the privacy budget rises, it is clear that the trend of DTW is decreasing. This is because as the budget increases, the random perturbations become smaller and the similarity of the data streams will be greater. Among them, the PP-LDP uses the sensible sampling approach, the budget allocation approach, and post-processing optimization, effectively ensuring similarity and good stability to ensure utility.

In addition to comparing the impact of the above methods on utility and the degree of pattern preservation when the privacy budget changes, we also simulated the impact of other factors on the data stream. First, as shown in Fig. 7, we evaluate the impact of sliding window size on the data stream in the w-event privacy. Second, as shown in Fig. 8, since the four data sets we used are already existing fixed-size data sets,

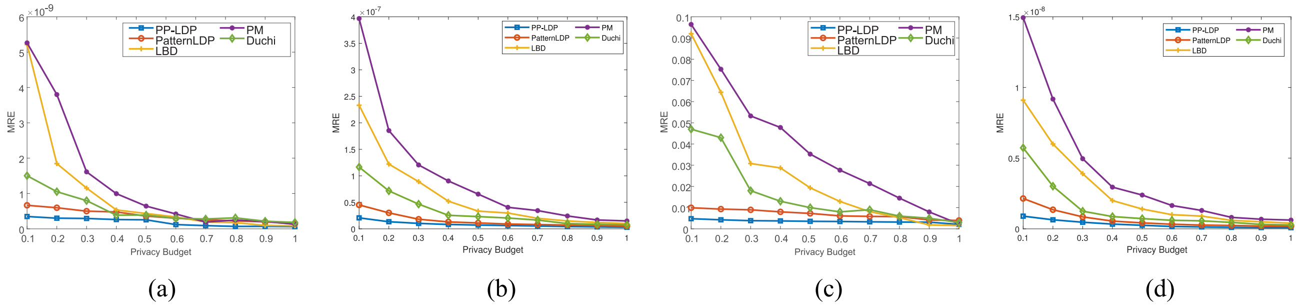


Fig. 5. Impact of privacy budgets on statistical analysis. (a) HRA. (b) HKHS. (c) EDTR. (d) EC.

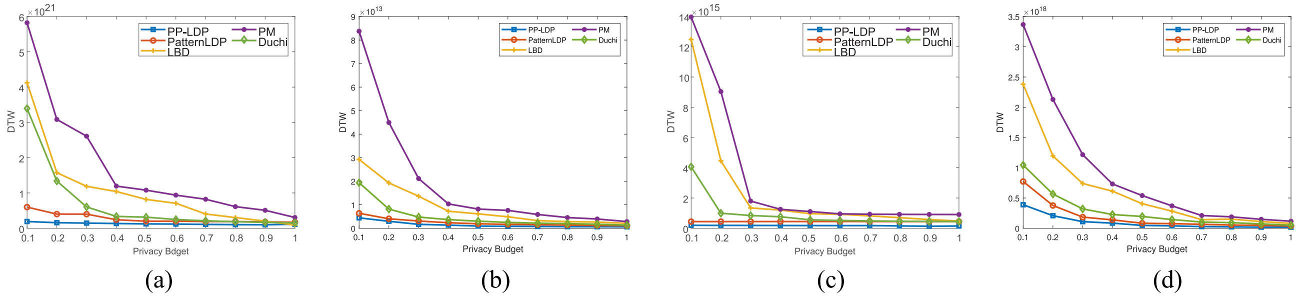


Fig. 6. Impact of privacy budgets on pattern. (a) HRA. (b) HKHS. (c) EDTR. (d) EC.

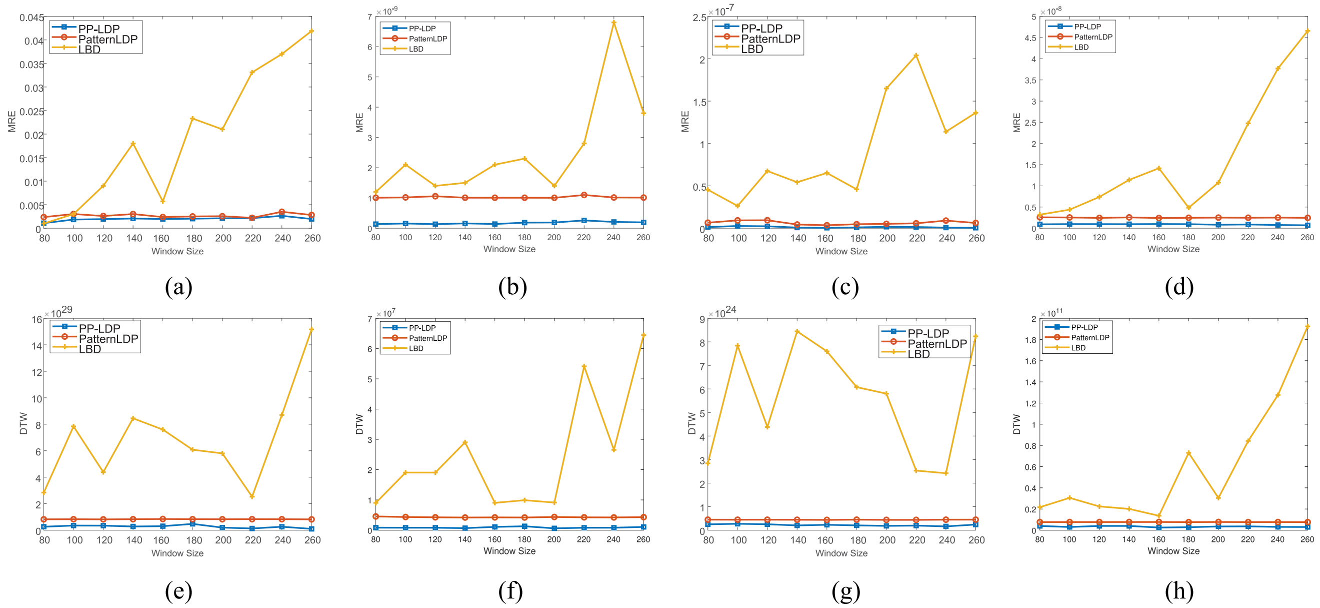
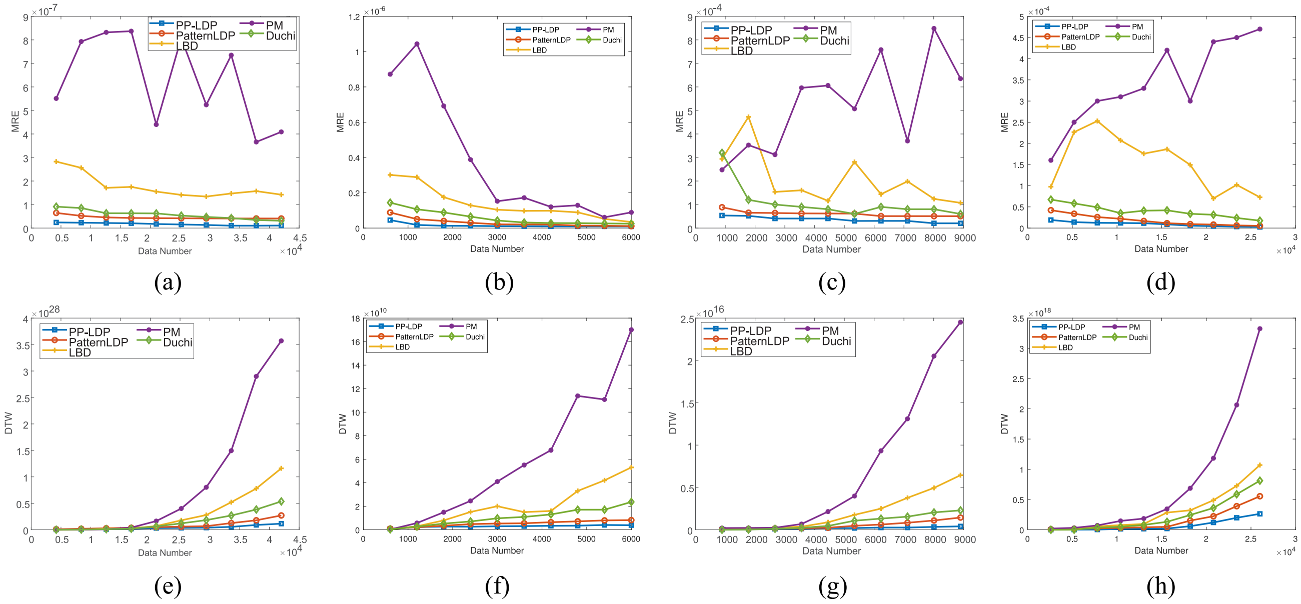


Fig. 7. Impact of sliding windows on utility. (a) and (e) HRA. (b) and (f) HKHS. (c) and (g) EDTR. (d) and (h) EC.

in order to better model the characteristic that the data points in real-time dynamic data streams increase in number over time, we also compared the impact of the five methods on utility and the degree of pattern preservation as the number of data points increased.

Since only LBD, PP-LDP, and PatternLDP provide *w*-event privacy, we only compare the impact of these methods on the data stream utility and the degree of pattern preservation when the sliding window size is changed. As shown in Fig. 7(a)–(d), the impact of sliding windows on the utility of data streams after privacy-preserving, respectively. As can be seen, the MRE of the LBD method is larger regardless of how the sliding window's size is changed. This is because the LBD uses exponentially decreasing ways to allocate the budget, which causes more data perturbation in each window and lower utility. PP-LDP and PatternLDP can still maintain a small error, and PP-LDP is better than PatternLDP. This is because PP-LDP allocates a higher budget to the sampling points within the sliding window in terms of data stream change rate, change trend, and the potential number of sampling points, and it better ensures the data error through post-processing optimization. As a result, PP-LDP provides a better average relative error and preserves the pattern well even when the sliding window is changed.



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 8. Impact of data number on utility. (a) and (e) HRA. (b) and (f) HKHS. (c) and (g) EDTR. (d) and (h) EC.

As shown in Fig. 7(e)–(h), the effect of sliding windows on the degree of preserved patterns in the four data sets, respectively, no matter how the sliding window changes, it is obvious that the LBD preserves the patterns poorly due to budget allocation. Meanwhile, the sampling approach, the perturbation approach, and the post-processing optimization of PP-LDP can effectively preserve the patterns while providing better utility.

Fig. 8(a)–(d) shows the effect of the data number on MRE. It can be seen that the PM and LBD are in an unsteady fluctuation state, which is due to the larger range of random perturbations in the PM, resulting in larger errors in the perturbed values and making it impossible to maintain a smooth trend in the average relative error. The perturbation range in LBD is related to the data number in the sliding window, so a change in the data number causes a change in the perturbation range, resulting in a fluctuating average relative error. Meanwhile, in PP-LDP, PatternLDP, and Duchi, the MRE is decreasing as the data number increases. The fact that PP-LDP consistently maintains a low MRE implies that the accuracy is dependent on a large amount of data and that its utility increases as the number of data points increases. And better utility is obtained with PP-LDP when the data stream length is longer and the data number is larger.

Fig. 8(e)–(h) shows the effect of data number on the DTW distance, i.e., the effect of methods on the degree of preserved pattern in the real-time dynamic data stream. The fact that the DTW grows indicates that as the data number rises, more points are perturbed, which reduces the data stream's similarity. Among them, PP-LDP still has better stability than other methods in the case of increasing data numbers and preserving patterns better.

To summarize, by changing the privacy budget and the size of the sliding window, PP-LDP always provides better data utility and effectively preserves patterns when compared with other methods. And in real-time dynamic data streaming, PP-LDP provides good data utility and effectively preserves patterns, even if the amount of data is getting larger over time.

VI. Conclusion

In this article, we examine privacy-preserving techniques for real-time data streams published on honest but observant servers, with the goal of preserving data stream patterns while enhancing utility and preserving privacy. We use the improved LSSLF combined with adaptive thresholding to sample remarkable points that can effectively represent data stream patterns, which can be applied to a variety of dynamic data streams and effectively balance privacy and utility. Based on this, we adaptively perturb sampling points according to the trend and rate of data stream fluctuations and take full advantage of the DP post-processing immunity to filter the data in order to improve the utility of data streams and preserve useful patterns. We compare our method to existing methods with four realistic data sets and demonstrate that it not only provides a good level of privacy preserving but also effectively preserves the original patterns of the data stream.

References

[1] S. Zhou, Y. He, S. Xiang, K. Li, and Y. Liu, "Region-based compressive networked storage with lazy encoding," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 6, pp. 1390–1402, Jun. 2019.

[2] S. Zhou, Y. Lian, D. Liu, H. Jiang, Y. Liu, and K. Li, "Compressive sensing based distributed data storage for mobile crowdsensing," *ACM Trans. Sens. Netw.*, vol. 18, no. 2, pp. 1–25, 2022.

[3] S. Zhou, X. Zhang, Y. Liu, H. Jiang, and K. Li, "Decentralized and compressed data storage for mobile crowdsensing," *IEEE Trans. Mobile Comput.*, early access, Jul. 13, 2023, doi: 10.1109/TMC.2023.3294969.

[4] A. Latvala et al., "Association of resting heart rate and blood pressure in late adolescence with subsequent mental disorders: A longitudinal population study of more than 1 million men in Sweden," *JAMA Psychiat.*, vol. 73, no. 12, pp. 1268–1275, 2016.

[5] Z. Qin, J. Weng, Y. Cui, and K. Ren, "Privacy-preserving image processing in the cloud," *IEEE Cloud Comput.*, early access, Jan. 12, 2018, doi: 10.1109/MCC.2018.111121403.

[6] D. Chen, P. Bovornkeeratiroj, D. Irwin, and P. Shenoy, "Private memoirs of IoT devices: Safeguarding user privacy in the IoT era," in *Proc. IEEE Int. Conf. Distrib. Comput. Syst. (ICDCS)*, 2018, pp. 1327–1336.

[7] S. Hu, Q. Wang, J. Wang, Z. Qin, and K. Ren, "Securing SIFT: Privacy-preserving outsourcing computation of feature extractions over encrypted image data," *IEEE Trans. Image Process.*, vol. 25, pp. 3411–3425, 2016.

[8] C. Dwork, "Differential privacy," in *Proc. Int. Colloq. Automata, Languages, Program.*, 2006, pp. 1–12.

[9] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Local privacy and statistical minimax rates," in *Proc. IEEE Annu. Symp. Found. Comput. Sci.*, 2013, pp. 429–438.

[10] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Local privacy, data processing inequalities, and statistical minimax rates," in *Proc. Comput. Sci.*, 2013, pp. 429–438.

[11] S. Xiong, A. D. Sarwate, and N. B. Mandayam, "Randomized requantization with local differential privacy," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, 2016, pp. 2189–2193.

[12] G. Fanti, V. Pihur, and Ú. Erlingsson, "Building a RAPPOR with the unknown: Privacy-preserving learning of associations and data dictionaries," *Proc. Privacy Enhanc. Technol.*, vol. 2016, no. 3, pp. 41–61, 2016.

[13] B. Ding, J. Kulkarni, and S. Yekhanin, "Collecting telemetry data privately," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 3571–3580.

[14] T. Wang, Z. Li, N. Li, M. Lopuhaä-Zwakenberg, and B. Skoric, "Consistent and accurate frequency oracles under local differential privacy," 2020, *arXiv:1905.08320*.

[15] Z. Wang, W. Liu, X. Pang, J. Ren, Z. Liu, and Y. Chen, "Towards pattern-aware privacy-preserving real-time data collection," in *Proc. IEEE Conf. Comput. Commun.*, 2020, pp. 109–118.

[16] T.-C. Fu, F.-L. Chung, K.-Y. Kwok, and C.-M. Ng, "Stock time series visualization based on data point importance," *Eng. Appl. Artif. Intell.*, vol. 21, no. 8, pp. 1217–1232, 2008.

[17] G. Kellaris, S. Papadopoulos, X. Xiao, and D. Papadias, "Differentially private event sequences over infinite streams," *VLDB Endow.*, vol. 7, no. 12, pp. 1155–1166, 2014.

[18] Y. Lin et al., "DRL-based adaptive sharding for blockchain-based federated learning," *IEEE Trans. Commun.*, vol. 71, no. 10, pp. 5992–6004, Oct. 2023.

[19] X. Deng, B. Chen, X. Chen, X. Pei, S. Wan, and S. K. Goudos, "Trusted edge computing system based on intelligent risk detection for smart IoT," *IEEE Trans. Ind. Informat.*, early access, May. 17, 2023, doi: 10.1109/TII.2023.3245681.

[20] F. Ding et al., "Securing facial bioinformation by eliminating adversarial perturbations," *IEEE Trans. Ind. Informat.*, vol. 19, no. 5, pp. 6682–6691, May 2022.

[21] F. Benhamouda, M. Joye, and B. Libert, "A new framework for privacy-preserving aggregation of time-series data," *ACM Trans. Inf. Syst. Secur.*, vol. 18, no. 3, pp. 1–21, 2016.

[22] S. Song and K. Chaudhuri, "Composition properties of inferential privacy for time-series data," in *Proc. Annu. Allerton Conf. Commun., Control, Comput.*, 2017, pp. 814–821.

[23] Y. Zheng, R. Lu, Y. Guan, J. Shao, and H. Zhu, "Efficient and privacy-preserving similarity range query over encrypted time series data," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 4, pp. 2501–2516, Jul./Aug. 2022.

[24] X. Liu, Y. Zheng, X. Yi, and S. Nepal, "Privacy-preserving collaborative analytics on medical time series data," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 3, pp. 1687–1702, May/Jun. 2022.

[25] A. Pyrgelis, C. Troncoso, and E. D. Cristofaro, "Measuring membership privacy on aggregate location time-series," *Proc. ACM Meas. Anal. Comput. Syst.*, vol. 4, no. 2, pp. 1–28, 2020.

[26] T. Wang et al., "Continuous release of data streams under both centralized and local differential privacy," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2021, pp. 1237–1253.

[27] X. Ren, L. Shi, W. Yu, S. Yang, C. Zhao, and Z. Xu, "LDP-IDS: Local differential privacy for infinite data streams," in *Proc. Int. Conf. Manag. Data*, 2022, pp. 1064–1077.

[28] J. Wang et al., "Improved Kalman filter based differentially private streaming data release in cognitive computing," *Future Gener. Comput. Syst.*, vol. 98, pp. 541–549, Sep. 2019.

[29] Q. Xue, Q. Ye, H. Hu, Y. Zhu, and J. Wang, "DDRM: A continual frequency estimation mechanism with local differential privacy," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 7, pp. 6784–6797, Jul. 2023.

[30] X. Gu, M. Li, Y. Cao, and L. Xiong, "Supporting both range queries and frequency estimation with local differential privacy " in *Proc. IEEE Conf. Commun. Netw. Secur. (CNS)*, 2019, pp. 124–132.

[31] L. Fan and L. Bonomi, "Time series sanitization with metric-based privacy," in *Proc. IEEE Int. Congr. Big Data*, 2018, pp. 264–267.

[32] C. Dwork, M. Naor, T. Pitassi, and G. N. Rothblum, "Differential privacy under continual observation," in *Proc. 42nd ACM Symp. Theory Comput.*, 2010, pp. 715–724.

[33] M. Xu, B. Ding, T. Wang, and J. Zhou, "Collecting and analyzing data jointly from multiple services under local differential privacy," *Proc. VLDB Endow.*, vol. 13, no. 12, pp. 2760–2772, 2020.

[34] V. Perrier, H. J. Asghar, and D. Kaafar, "Private continual release of real-valued data streams," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, 2019, pp. 1–13.

[35] X. Ren, S. Wang, X. Yao, C.-M. Yu, and X. Yang, "Differentially private event sequences over infinite streams with relaxed privacy guarantee," in *Proc. Int. Conf. Wireless Algorithms, Syst., Appl.*, 2019, pp. 272–284.

[36] Q. Ye, H. Hu, N. Li, X. Meng, H. Zheng, and H. Yan, "Beyond value perturbation: Local differential privacy in the temporal setting," in *Proc. IEEE Conf. Comput. Commun.*, 2021, pp. 1–10.

[37] F. Z. Errounda and Y. Liu, "Collective location statistics release with local differential privacy," *Future Gener. Comput. Syst.*, vol. 124, pp. 174–186, Nov. 2021.

[38] D. Eberly, *Least Squares Fitting of Data*. Chapel Hill, NC, USA: Magic Softw., 2000, pp. 1–10.

[39] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Found. Trends Theor. Comput. Sci.*, vol. 9, nos. 3–4, pp. 211–407, 2014.

[40] L. Fan and L. Xiong, "An adaptive approach to real-time aggregate monitoring with differential privacy," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 9, pp. 2094–2106, Sep. 2014.

[41] M. King, *Process Control: A Practical Approach*. Hoboken, NJ, USA: Wiley, 2010.

[42] Z. Li, T. Wang, M. Lopuhaä-Zwakenberg, N. Li, and B. Škoric, "Estimating numerical distributions under local differential privacy," in *Proc. ACM SIGMOD Int. Conf. Manag. Data*. 2020, pp. 621–635.

[43] "JenniferLing/heart_rate_analysis." 2017. [Online]. Available: https://github.com/JenniferLing/heart_rate_analysis

[44] Y. Chen et al., Oct. 2018, "The UCR time series classification archive," UCR. [Online]. Available: https://www.cs.ucr.edu/eamonn/time_series_data_2018/

[45] 2015, "ElectricityLoadDiagrams20112014," UCI. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/ElectricityLoadDiagrams20112014

[46] N. Wang et al., "Collecting and analyzing multidimensional data with local differential privacy," in *Proc. IEEE Int. Conf. Data Eng. (ICDE)*, 2019, pp. 638–649.

[47] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Minimax optimal procedures for locally private estimation," *J. Amer. Stat. Assoc.*, vol. 113, no. 521, pp. 182–201, 2018.

Wen Gao received the B.S. degree in computer science and technology from Changsha University, Changsha, China, in 2016, and the M.S. degree in software engineering from Central South University, Changsha, in 2019. She is currently pursuing the Ph.D. degree with the College of Computer Science and Electronic Engineering, Hunan University, Changsha.

Her research interests include privacy-preserving in social network, and incentive mechanism in crowd sensing and compressed sensing.

Siwang Zhou received the B.S. degree from Fudan University, Shanghai, China, in 1995, the M.S. degree from Xiangtan University, Xiangtan, China, in 2004, and the Ph.D. degree from Hunan University, Changsha, China, in 2007.

He has been a Professor with the College of Computer Science and Electronic Engineering, Hunan University. His research interests include image compressive sensing, deep learning, and Internet of Things.