ISSN 1673-9418 CODEN JKYTA8
Journal of Frontiers of Computer Science and Technology
1673-9418/2016/10(04)-0481-14
doi: 10.3778/j.issn.1673-9418.1507074

E-mail: fcst@vip.163.com
http://www.ceaj.org
Tel: +86-10-89056056

基于矩阵机制的差分隐私连续数据发布方法*

蔡剑平,吴英杰+,王晓东
福州大学 数学与计算机科学学院,福州 350116

Method Based on Matrix Mechanism for Differential Privacy Continual Data Release*

CAI Jianping, WU Yingjie+, WANG Xiaodong
College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350116, China
+ Corresponding author: E-mail: yjwu@fzu.edu.cn

CAI Jianping, WU Yingjie, WANG Xiaodong. Method based on matrix mechanism for differential privacy continual data release. Journal of Frontiers of Computer Science and Technology, 2016, 10(4): 481-494.

Abstract: The vast majority of the literature on differential privacy algorithms focuses on one time static release of datasets, while many applications of data analysis involve the continual data release. This paper proposes a method based on matrix mechanism for differential privacy continual data release. The key idea of the proposed method is to firstly construct the strategy matrix of the continual data release problem using the binary indexed tree, and then optimize the strategy matrix to boost the accuracy of the published data. After that, aiming at the high time complexity of existing optimization algorithm based on matrix mechanism, this paper puts forward a fast diagonal matrix optimization algorithm (FDA) with $O(\lg N)$ time complexity, which can be applied to the situation of large-scale continuous data publishing effectively. This paper compares and analyzes FDA and the traditional algorithms on the accuracy of the released data by experiments. The experimental results show that FDA is effective and feasible.
Key words: differential privacy; matrix mechanism; binary indexed tree; continual data release

摘 要:现有绝大多数差分隐私算法只考虑数据的一次静态发布,而实际许多数据分析应用却涉及连续数据发布。为此,提出了一种基于矩阵机制的差分隐私连续数据发布方法。该方法的核心思想是首先利用树状数组构建连续数据发布问题的策略矩阵,然后对策略矩阵进行优化以提高发布数据的精确性。随后,进一步针

* The National Natural Science Foundation of China under Grant No. 61300026 (国家自然科学基金); the Natural Science Foundation of Fujian Province under Grant No. 2014J01230 (福建省自然科学基金).
Received 2015-06, Accepted 2015-08.
CNKI网络优先出版: 2015-08-27, http://www.cnki.net/kcms/detail/11.5602.TP.20150827.1411.002.html

对现有基于矩阵机制的优化算法复杂度极高的问题,提出了时间复杂度为$O(\lg N)$的快速对角阵优化算法(fast diagonal matrix optimization algorithm,FDA),以有效应用于大规模的连续数据发布。通过实验比较分析了FDA算法与同类算法所发布数据的精确度,结果表明FDA算法是有效可行的。

关键词:差分隐私;矩阵机制;树状数组;连续发布

文献标志码:A **中图分类号**:TP309.2

1 引言

随着数字技术的发展,数据越来越多地充斥于现实生活当中。数据给人们生活带来的好处不言而喻,人们不仅可以利用数据进行评估、分析和预测,还可以从中寻找有价值的信息,如啤酒与尿布的故事。然而,在享受数据带来的好处的同时,也应该注意到数据中包含的个人隐私信息可能存在隐私泄露的风险。特别是当攻击者带有恶意时,他就有可能利用已掌握的知识分析所发布的数据,并从中挖掘出数据所对应用户的隐私信息。例如,只需根据4个时空点就能使95%的人泄露其位置信息[1]。因此,如何在发布数据的同时避免数据中包含的隐私被泄露是数据时代亟待解决的问题之一。针对这一问题,各种隐私保护模型被提出。其中,以提供严格数据保护为特点的差分隐私模型[2-4]得到了广泛的认可。该模型被提出后,人们基于该模型开展了很多研究工作,内容涉及直方图发布[5-8]、空间划分发布[9-10]、智能数据分析[11-12]等。相比于基于k-匿名[13]和划分[14]的隐私保护方法该模型有效克服了需要事先对攻击做出假设的不足。差分隐私数据发布研究的关键问题在于如何在保证差分隐私的前提下,同时提高所发布数据的可用性。

现有关于差分隐私的数据发布方法大多关注静态发布问题,而现实应用中更多情况下需要发布方法具有连续数据发布的能力。然而,经研究表明,这些方法无法应用于连续数据发布问题。为此,本文对差分隐私下的连续数据发布问题展开研究。例如,某医疗数据库中记录了每个月的入院病人的信息,其中病人感染HIV的情况为敏感信息。表1展示了其中3个月的数据示例。同时,出于某研究目的,医院将按月统计并公布入院的HIV病人数。公布的数据形如表2所示,医院将当前入院并且感染HIV的病人累计并于当月发布最新数据。与数据静态发布不同,在医院发布完每个月的统计信息后,该数据并非不再改变,而是在下个月将得到更新。更重要的是,在发布每一次数据的过程中,以后需要发布的数据是无法被预知的。该问题的核心是在满足差分隐私的条件下,寻找更精确且更高效率的连续数据发布方法。

Table 1 Medical records

表1 医学记录表

Name	HIV+	Name	HIV+	Name	HIV+
Alice	Yes	Alan	No	Andy	No
Bob	No	Ben	Yes	Bill	No
Carol	Yes	Cari	No	Chen	Yes
…		…		…	

Table 2 Statistics on HIV+ patients

表2 HIV+病人数统计表

Month	HIV+	Sum of HIV+
1	531	531
2	392	923
3	426	1 349
…		

以上连续发布问题的一种朴素解决方案[15]是直接在前一个月发布的HIV病人加上本月新增的HIV病人数,然后再添加噪声使其满足差分隐私。该方案导致每一次发布数据的噪声的均方误差线性累加,最终发布的数据失去可用性。文献[15]针对该问题提出了一种基于二叉树的发布方法。然而,此方法仅仅引入二叉树模拟发布,并未对精确性提出有效优化。为此,本文以提高发布数据的精确性为主要目标,将矩阵机制引入差分隐私连续数据发布问

题中，以期设计出高效的基于矩阵机制的差分隐私连续数据发布方法，有效满足大规模连续数据发布的要求。

2 相关工作

在差分隐私的数据发布中，为提高数据发布的精度，Hay[7]和Xiao等人[8]分别提出的基于一致性调节的区间树方法和小波变换方法实现了较高精度的数据发布。然而，上述两种方法只适用于差分隐私下的数据静态发布，无法应用于差分隐私下的连续数据发布。Chan等人[15]提出了两种利用二叉树结构进行连续数据发布的方法。第一种方法构建一棵叶节点数量为 2^m 的完全二叉树，然后利用模拟二叉树统计发布的过程进行连续数据发布。第二种方法是第一种方法的改进版本，它试图通过调整二叉树各层节点的隐私预算分配来达到无限发布的效果。研究表明，虽然第一种方法相比于朴素方法，数据发布的精确性有显著的提升，然而该方法仅仅引入二叉树结构来模拟发布过程，并未做进一步改进，因此数据发布的精确性仍有较大的提升空间；第二种方法的隐私预算分配并不合理，导致发布数据的误差远大于第一种方法。

为了解决差分隐私下的线性查询问题，Li等人[16]提出了基于矩阵机制的批量查询方法。其基本思想是通过寻找策略矩阵对线性查询进行优化，进而提高发布数据的精确性。然而，该文献提出的矩阵机制仅能满足小规模数据集和查询负载的要求。此外，它还很容易产生次优化的查询策略，使得结果往往并不理想。为此，Yuan等人[17]利用负载矩阵低秩的性质进行优化，提出了低秩矩阵机制，在一定程度上改善了原有矩阵机制的不足，提升了数据发布效率与精确性。然而，该文献提出的优化查询使用半正定规划算法，同样只能适用小规模数据集和查询负载的要求。本文研究的连续数据发布问题本质上也是线性查询问题，因此拟利用矩阵机制，结合连续数据发布问题本身具有的一些特性，设计出精度更高且效率更高的算法，使之具有大规模连续数据发布的能力。

3 基础知识与问题

3.1 差分隐私

Dwork等人[2]首次提出差分隐私保护模型。该模型保证了在数据发布过程中，无论攻击者具有何种背景知识，都无法泄露隐私数据。其形式化定义如下：

定义1（ε－差分隐私[2]）设 A 表示在查询 Q 的基础上通过某种方式添加随机噪声的随机算法。将其分别作用于兄弟数据集 $\boldsymbol{D}$ 和 $\boldsymbol{D}'$ 时的概率密度满足以下条件，则算法 A 满足 ε-差分隐私。其中 $\boldsymbol{O}$ 表示可能的输出集合。

$$\forall \boldsymbol{X}\in \boldsymbol{O},\ \mathrm{e}^{-\varepsilon}\leqslant \frac{P(A(\boldsymbol{D})=\boldsymbol{X})}{P(A(\boldsymbol{D}')=\boldsymbol{X})}\leqslant \mathrm{e}^{\varepsilon} \tag{1}$$

定义2（敏感度）对数据集 $\boldsymbol{D}$ 和 $\boldsymbol{D}'$ 进行统计得：$Q(\boldsymbol{D})=(x_1,x_2,\cdots,x_n)^{\mathrm{T}}$，$Q(\boldsymbol{D}')=(x_1',x_2',\cdots,x_n')^{\mathrm{T}}$。那么查询集合 Q 的敏感度 Δ_Q 满足以下定义：

$$\Delta_Q=\max_{\boldsymbol{D}\cong\boldsymbol{D}'}\left\|Q(\boldsymbol{D})-Q(\boldsymbol{D}')\right\|_p \tag{2}$$

本文主要使用1-范数，即 $p=1$。

文献[18]提出了拉普拉斯机制。该机制通过添加拉普拉斯噪音来实现差分隐私保护。为了表示方便，本文用 $\tilde{\boldsymbol{L}}_n$ 表示满足分布 Laplace(0,1) 的 n 维列向量。

3.2 矩阵机制

矩阵机制是一种针对差分隐私下线性查询问题的优化方法。它通过将查询集 $\boldsymbol{Q}$ 转换成负载矩阵 $\boldsymbol{W}$，然后寻找最优策略矩阵 $\boldsymbol{M}$ 实现差分隐私下线性查询的优化。其中查询集 $\boldsymbol{Q}$ 是一组线性查询的集合，满足 $\boldsymbol{Q}=\{q_1,q_2,\ldots,q_n\}$。每条线性查询表示如下：

$$q=\sum_{i=1}^{m}w_i x_i$$

其中，$\boldsymbol{X}=(x_1,x_2,\cdots,x_m)^{\mathrm{T}}$ 为数据向量；w_i 为该查询于分量 x_i 的权重。负载矩阵 $\boldsymbol{W}$ 由每组线性查询的权重组成，并满足：

$$\boldsymbol{Q}=\boldsymbol{W}\boldsymbol{X}=\left(\sum_{j=1}^{m}W_{1j}x_j,\sum_{j=1}^{m}W_{2j}x_j,\cdots,\sum_{j=1}^{m}W_{nj}x_j\right)^{\mathrm{T}}$$

原始的矩阵机制[16]通过直接寻找策略矩阵 $\boldsymbol{M}$ 的方式求解问题，这种做法的效率和优化效果都不够理想。而低秩矩阵机制[17]则是采用分解负载矩阵的方法来寻找优化策略。该机制下，将 $\boldsymbol{W}$ 分解成两个

矩阵 $\boldsymbol{B}$、$\boldsymbol{M}$。其中 $\boldsymbol{M}$ 表示低秩矩阵下的策略矩阵，且满足 $\boldsymbol{W}=\boldsymbol{BM}$。通过对中间结果 $\boldsymbol{MX}$ 添加噪声的方式减少误差。其形式化表示如下：

$$A(\boldsymbol{W},\boldsymbol{X})=\boldsymbol{B}\left(\boldsymbol{MX}+\frac{\Delta_{\boldsymbol{M}}}{\varepsilon}\tilde{\boldsymbol{L}}_n\right) \tag{3}$$

文献[17]指出，式(3)的敏感度 $\Delta_{\boldsymbol{M}}$ 与策略矩阵的列范式相等，即 $\Delta_{\boldsymbol{M}}=\boldsymbol{M}_1$。而低秩矩阵的均方误差由下式求得：

$$err=\frac{2}{\varepsilon^2}\text{trace}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})\Delta_{\boldsymbol{M}}{}^2 \tag{4}$$

研究表明，差分隐私下的数据连续发布问题能够被转换成基于矩阵机制的优化问题。只需将每一次发布视为一个查询，然后将所有发布过程视为查询负载，并转换成相应的矩阵，利用矩阵机制进行求解。

3.3 问题提出

考虑一个随着时间增长，记录会不断产生并被添加其中的记录流，该记录流是数据发布的来源。记录流的每条记录都有需要保护的属性 σ，满足 $\sigma\in\{0,1\}$。并假设记录集 $\mathbb{A}_i$ 表示第 $i-1$ 次和第 i 次发布之间记录流中被添加的记录的集合。由 $\mathbb{A}_i$ 可求出第 i 次发布的数据增量 $a_i=|\{\sigma|\sigma\in\mathbb{A}_i\text{ and }\sigma=1\}|$。

定义3（连续数据发布）对于记录流，数据发布者随着记录流的记录增长按照某种规则多次发布当前记录流中满足 $\sigma=1$ 的记录数的行为即为连续数据发布。假设第 i 次发布的累计数据为 s_i，那么 s_i 满足：

$$s_i=s_{i-1}+a_i=\sum_{j=1}^{i}a_j \tag{5}$$

差分隐私下的连续数据发布行为即通过某种隐私算法 $A(*)$ 发布添加了噪声的累计数据 $\tilde{s}_i$，从而使数据连续发布的结果满足 ε-差分隐私。同时，在此基础上，本文还要求所提出的算法能够精确而且高效地发布数据。

为了避免数据连续发布的敏感度过高而影响数据发布精度，本文主要考虑了给定次数下的满足 ε-差分隐私的次数受限的数据连续发布算法。

定义4（次数受限的数据连续发布算法）如果隐私算法 $A(*)$ 至多接受并发布 N 次满足 ε-差分隐私的统计数据，则称该算法为次数受限的数据连续发布算法。

上述问题能够转换成线性查询问题，而矩阵机制能够对线性查询问题进行优化。为了提出更精确的数据连续发布算法，本文将这一问题与矩阵机制相结合，并在此基础上寻找快速发布算法。而且，矩阵机制是成熟的且经过严格检验的满足 ε-差分隐私[17]的隐私保护机制，因此基于矩阵机制的线性查询算法只要符合式(3)的形式就保证了该算法满足 ε-差分隐私。

利用矩阵机制优化前，需要将式(5)转换成负载矩阵 $\boldsymbol{W}$ 如下：

$$\boldsymbol{W}=\begin{pmatrix}1&0&0&\cdots\\1&1&0&\cdots\\1&1&1&\cdots\\\cdots&\cdots&\cdots&\end{pmatrix} \tag{6}$$

可以看出，数据连续发布下的负载矩阵 $\boldsymbol{W}$ 为下三角矩阵，且满足 $W_{ij}=1,i<j$。

根据矩阵机制的特征及一般研究过程[16-17]。本文将通过以下步骤对该问题展开研究：

(1)寻找初始的策略矩阵 $\boldsymbol{M}$，将 $\boldsymbol{W}$ 分解为矩阵 $\boldsymbol{B}$ 和 $\boldsymbol{M}$，使矩阵机制能够较为精确地发布数据；

(2)对策略矩阵 $\boldsymbol{M}$ 进行研究，寻找优化策略以优化数据发布的精确性；

(3)综合分析矩阵 $\boldsymbol{B}$、$\boldsymbol{M}$ 以及优化策略的性质，在保证数据发布的精确性得到优化的前提下，提出高效的优化算法。

4 数据连续发布算法

4.1 利用树状数组构造策略矩阵

由于数据随着发布过程动态产生，未来数据无法得知，只能根据当前以及过往的数据进行优化。这一特征反映到矩阵机制时，就要求矩阵机制所应用的策略矩阵为下三角矩阵。通过各种数据结构的对比研究，本文发现树状数组[19]更加适合于构造基于矩阵机制的数据连续发布方法的策略矩阵。它能够自然并且快速求解数列的前 k 项和，符合连续数据发布的基本特征。同时，初步研究表明，将它与差分隐私结合而提出的隐私保护模型能够达到与基于二叉树的连续数据发布方法[15]相当的精确性。同时，深入研究表明，结合树状数组的差分隐私模型在实现的巧妙性以及进一步优化的潜力方面，与后者相比均略胜一筹。

树状数组主要针对以下问题：给定 N 个实数，记为 $a_1,a_2,\cdots,a_n$，要求快速求出前 k 项的和。记前 k 项的和为 s_k，则 $s_k=\sum_{j=1}^{k}a_j$。

针对该问题，树状数组提供如下解决方法。该方法计算了中间统计量 c_i，而 c_i 由以下公式求得：

$$c_i=\sum_{j=i-lowbit(i)+1}^{i}a_j \tag{7}$$

其中，函数 $lowbit(x)$ 表示将正整数 x 写成二进制形式后，将该二进制数的值为1的最低位的数置为1，其余位均置为0。例如，当 $x=10$ 时，其对应的二进制数为 $(1010)_2$，从低往高有 2^0 位为0，2^1 位为1。那么，$lowbit(x)$ 的输出值就将该位置为1，其他位均置为0，即 $(0010)_2=2$。再将其转成十进制就得到 $lowbit(10)=2$。

该函数可以表示如下：

算法1 $lowbit(x)$

输入：正整数 x

输出：x 对应的 $lowbit$ 值

1. $p\leftarrow 0,y\leftarrow 1$;
2. while $x \bmod 2=0$
3. $y\leftarrow 2\times y;x\leftarrow\left\lfloor\frac{x}{2}\right\rfloor$;
4. wend
5. return y

结合算法1以及式(7)，本文按照树状数组求解中间统计量的方式构造策略矩阵 $\boldsymbol{M}$。

算法2 求解策略矩阵 $\boldsymbol{M}$

输入：发布次数 N

输出：策略矩阵 $\boldsymbol{M}$

1. $\boldsymbol{M}\leftarrow\boldsymbol{O}_{N\times N}$; //初始化为零矩阵
2. for $p=1$ to N do
3. $pt\leftarrow p$;
4. while $pt<N$
5. $M_{pt,p}\leftarrow 1$; //更新矩阵元素
6. $pt\leftarrow pt+lowbit(pt)$;
7. wend
8. end for
9. return $\boldsymbol{M}$

下面，需要进一步讨论矩阵 $\boldsymbol{B}$ 的求解。由 $\boldsymbol{W}=\boldsymbol{BM}$ 以及 $\boldsymbol{M}_N$ 的可逆性可得，$\boldsymbol{B}=\boldsymbol{W}_N\boldsymbol{M}_N^{-1}$，因此只需根据树状数组的性质就能够快速地求解 $\boldsymbol{B}$。而树状数组的求和操作按照以下公式进行求解：

$$s_i=c_i+s_{i-lowbit(i)} \tag{8}$$

算法3 求解矩阵 $\boldsymbol{B}$

输入：发布次数 N

输出：矩阵 $\boldsymbol{B}$

1. $\boldsymbol{B}\leftarrow\boldsymbol{O}_{N\times N}$; //初始化为零矩阵
2. for $p=1$ to N do
3. $pt\leftarrow p$;
4. while $pt>0$
5. $B_{pt,p}\leftarrow 1$; //更新矩阵元素
6. $pt\leftarrow pt-lowbit(pt)$;
7. wend
8. end for
9. return $\boldsymbol{B}$

上述算法结合低秩矩阵的表达式，即可得到基于树状数组的数据连续发布的表达式：

$$A(\boldsymbol{W},\boldsymbol{D})=\boldsymbol{B}\left(\boldsymbol{M}_N\boldsymbol{X}+\frac{\Delta_{\boldsymbol{M}_N}}{\varepsilon}\tilde{\boldsymbol{L}}_n\right) \tag{9}$$

接下来，本文将分析该策略矩阵所产生的均方误差的情况。关于 $\boldsymbol{M}_N$ 和 $\boldsymbol{B}$ 有如下两个相关定理。

定理1 $\|\boldsymbol{M}_N\|_1=\|\boldsymbol{M}_N(:,1)\|_1\geq\|\boldsymbol{M}_N(:,j)\|_1(j>1)$，且 $\|\boldsymbol{M}_N\|_1=\lfloor \mathrm{lb}\,N\rfloor+1$。

证明 通过算法2研究矩阵 $\boldsymbol{M}_N(:,1)$ 的构造情况。可得当第一次迭代 $p=1$ 时，有 $pt=p=1=2^0$。设第 t 次迭代 $pt=2^{t-1}$，由更新表达式有 $pt=pt-lowbit(pt)=2^{t-1}+2^{t-1}=2^t$。而根据步骤6的判断条件，有 $pt>N$ 时，该列构造结束。因此有，$2^{t-1}\leq N\Rightarrow t\leq \mathrm{lb}\,N+1$。此时 $M_{pt,p}=1$ 更新了 $\lfloor \mathrm{lb}\,N\rfloor+1$ 次，$\|M_N(:,1)\|_1=\lfloor \mathrm{lb}\,N\rfloor+1(j>1)$。

对于 $j>1$ 的情况，假设第 t 次迭代 $pt>2^{t-1}$ 且 $lowbit(pt)\geq 2^{t-1}$，由第一次迭代有 $pt=j>1$，可知满足该条件。根据 $lowbit$ 函数的性质，$lowbit(pt')=lowbit(pt+lowbit(pt))\geq lowbit\,(2lowbit(pt))=2lowbit(pt)\geq 2^t$，从而 $pt'=pt+lowbit(pt)>2^t$。很显然，根据 $pt>2^{t-1}$ 可推论，$j>1$ 时的更新次数不大于 $j=1$。因此，$\|\boldsymbol{M}_N(:,1)\|_1\geq\|\boldsymbol{M}_N(:,j)\|_1(j>1),\|\boldsymbol{M}_N\|_1=\max_j\|\boldsymbol{M}_N(:,j)\|_1=\lfloor \mathrm{lb}\,N\rfloor+1$。 □

定理2 构造矩阵 $\boldsymbol{B}$ 的第 p 行的迭代次数为将 p 表示为二进制 $(p)_2$ 时，$(p)_2$ 中包含的1的个数。

证明 根据算法3的步骤6有操作 $pt=pt-lowbit(pt)$，该操作的结果是将 $(pt)_2$ 中值为1的最低位置为0。而由步骤3有 pt 由 p 进行初始化。说明 $(p)_2$ 中包含多少个1，迭代就进行了多少次。 □

由式(4)推出该策略矩阵所产生的均方误差为：

$$err=\frac{2}{\varepsilon^2}\mathrm{trace}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})\Delta_{\boldsymbol{M}_N}{}^2=\frac{2}{\varepsilon^2}\mathrm{trace}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})(\lfloor \mathrm{lb}\,N\rfloor+1)^2$$

同时，结合定理2可知，$\boldsymbol{B}$ 的每一行至多有 $O(\lg N)$ 个元素为1，其余皆为0。因此，B 中元素为1的个数为 $O(N\lg N)$。即 $\mathrm{trace}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})$ 的数值复杂度也为 $O(N\lg N)$。又由定理1可知，$\boldsymbol{M}_N$ 的列范数 $\lfloor \mathrm{lb}\,N\rfloor+1$，其复杂度为 $O(\lg N)$。因而，根据式(4)可以求得总体的均方误差为 $O(N\lg^3 N)$。而对于每条查询的均方误差则为 $O(\lg^3 N)$。

综上所述，由树状数组构造的策略矩阵满足低敏感性以及均方误差复杂度低的特点，初步具备了在差分隐私下较为精确地进行连续数据发布的能力。然而，仅仅利用树状数组构造的策略矩阵并不能达到本文的精确性要求。接下来，本文将在此基础上寻找更精确的数据发布算法。

一般而言，可以通过发布数据的一致性调节[7]和策略矩阵的权重系数调节等方法提高数据发布的精确性。然而，研究表明该问题已经满足线性一致性，具体分析如下。

定义5（线性一致性）对于负载矩阵 $\boldsymbol{W}$，记未加噪的查询结果为 $\boldsymbol{Y}=\boldsymbol{W}Q(\boldsymbol{D})$，通过低秩矩阵机制获得的查询结果为 $\boldsymbol{Y}'=A(\boldsymbol{W},\boldsymbol{D})$。$A(\boldsymbol{W},\boldsymbol{D})$ 满足线性一致性当且仅当对于任意可以用 $\boldsymbol{Y}$ 线性表示的统计量 $\boldsymbol{z}$，对任意可以表示成 $\boldsymbol{z}=\boldsymbol{v}\boldsymbol{Y}$ 的行向量 $\boldsymbol{v}$ 都有 $\boldsymbol{v}\boldsymbol{Y}'$ 为定值。同时，线性一致性满足定理3。

定理3 当式(3)中的矩阵 $\boldsymbol{M}$ 为行满秩矩阵时，低秩矩阵机制满足线性一致性。

证明 当矩阵 $\boldsymbol{L}$ 为行满秩矩阵时，求得其右逆矩阵 $\boldsymbol{M}^{+}=\boldsymbol{M}^{\mathrm{T}}(\boldsymbol{M}\boldsymbol{M}^{\mathrm{T}})^{-1}$，满足 $\boldsymbol{M}\times\boldsymbol{M}^{+}=\boldsymbol{I}$。

由于 $\boldsymbol{W}=\boldsymbol{B}\boldsymbol{M}$，则 $\boldsymbol{B}=\boldsymbol{W}\boldsymbol{M}^{+}$。

任取统计查询 $\boldsymbol{z}$，有多个 v_i 满足 $\boldsymbol{z}=v_i\boldsymbol{W}\boldsymbol{X}$（其中 v_i 之间互不相等）。

将其代入式(3)中，可得统计后的结果，令 z_i' 表示由 v_i 求得的查询结果：

$$z_i'=v_i\boldsymbol{B}\left(\boldsymbol{M}\boldsymbol{X}+\frac{\Delta_{\boldsymbol{M}}}{\varepsilon}\tilde{\boldsymbol{L}}_n\right)=v_i\boldsymbol{W}\boldsymbol{M}^{+}\left(\boldsymbol{M}\boldsymbol{X}+\frac{\Delta_{\boldsymbol{M}}}{\varepsilon}\tilde{\boldsymbol{L}}_n\right)=\boldsymbol{z}\boldsymbol{M}^{+}\left(\boldsymbol{M}\boldsymbol{X}+\frac{\Delta_{\boldsymbol{M}}}{\varepsilon}\tilde{\boldsymbol{L}}_n\right)$$

经过化简可以看出，z_i' 是与 v_i 无关的噪声统计量。因此，可以得出 z_i' 的值是相等的。 □

$\boldsymbol{M}_N$ 为可逆矩阵，结合定理3，可得出推论：由树状数组构造基于矩阵机制的数据连续发布方法满足线性一致性。因此无法从线性一致性的角度提高数据发布的一致性。4.2节将对策略矩阵的权重系数调节问题进行研究。

4.2 利用对角矩阵提高精确性

上文分析了差分隐私下的连续数据发布的性质，并通过树状数组构造出矩阵机制的策略矩阵。而根据3.3节所描述的步骤，本文将在此基础上进行优化。进一步研究矩阵 $\boldsymbol{M}_N$，可发现该矩阵未饱和。以 $\boldsymbol{M}_3$ 为例，表示如下：

$$\boldsymbol{M}_3=\begin{pmatrix}1&0&0\\1&1&0\\0&0&1\end{pmatrix}\Rightarrow\boldsymbol{M}_3'=\begin{pmatrix}1&0&0\\1&1&0\\0&0&2\end{pmatrix}$$

计算可得 $\|\boldsymbol{M}_3\|_1=2$。若将 $\boldsymbol{M}_3$ 的第3行乘以2，得到 $\boldsymbol{M}_3'$，依旧满足 $\|\boldsymbol{M}_3'\|_1=2$，并不会影响整体的敏感度。同时，矩阵 $\boldsymbol{B}$ 也应转换为 $\boldsymbol{B}'$。

$$\boldsymbol{B}=\begin{pmatrix}1&0&0\\0&1&0\\0&1&1\end{pmatrix}\Rightarrow\boldsymbol{B}'=\begin{pmatrix}1&0&0\\0&1&0\\0&1&0.5\end{pmatrix}$$

根据式(4)，可以直接求出转换前后两者之间的均方误差。转换前为 $32/\varepsilon^2$，转换后为 $26/\varepsilon^2$。经过转换，均方误差降低了。这说明直接由树状数组构造的矩阵 $\boldsymbol{M}_N$ 优化还不够彻底。经研究发现，可以通过在 $\boldsymbol{M}_N$ 前面乘上一个对角阵的方式提高精确性。

令对角阵 $\boldsymbol{\Sigma}_N=\begin{pmatrix}\lambda_1&&&\\&\lambda_2&&\\&&\ddots&\\&&&\lambda_N\end{pmatrix}$，表示 $N\times N$ 的系数对角阵。运用该公式，可将式(9)拓展如下：

$$A(\boldsymbol{W},\boldsymbol{D})=\boldsymbol{B}\boldsymbol{\Sigma}_N^{-1}\left(\boldsymbol{\Sigma}_N\boldsymbol{M}_N\boldsymbol{X}+\frac{\Delta_{\boldsymbol{\Sigma}_N\boldsymbol{M}_N}}{\varepsilon}\tilde{\boldsymbol{L}}_n\right)\tag{10}$$

式(10)即为添加系数对角阵后的隐私保护机制。当 $\boldsymbol{\Sigma}_N=\boldsymbol{I}_N$ 时，该公式与式(9)等价。对应的均方误差公式如下所示：

$$err=\frac{2}{\varepsilon^2}\text{trace}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{\Sigma}_N^{-2})\Delta_{\boldsymbol{\Sigma}_N\boldsymbol{M}_N}^{\ 2} \tag{11}$$

根据文献[17]的结论，令 $\boldsymbol{B}'=\alpha\boldsymbol{B}\boldsymbol{\Sigma}_N^{-1}$，$L'=\alpha^{-1}\boldsymbol{\Sigma}_N\boldsymbol{M}_N$，则有 $\frac{2}{\varepsilon^2}\text{trace}(\boldsymbol{B}'^{\mathrm{T}}\boldsymbol{B}')\Delta_{L'}^{\ 2}=\frac{2}{\varepsilon^2}\text{trace}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{\Sigma}_N^{-2})\Delta_{\boldsymbol{\Sigma}_N\boldsymbol{M}_N}^{\ 2}$。因此，可将 $\Delta_{\boldsymbol{\Sigma}_N\boldsymbol{M}_N}$ 限制为 $\left|\boldsymbol{\Sigma}_N\boldsymbol{M}_N\right|_1\leqslant 1$，最小化 $\text{trace}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{\Sigma}_N^{-2})$。则该优化问题可表示为如下形式：

$$\begin{cases}\text{opt: }\min\limits_{\boldsymbol{\Sigma}_N} f(\boldsymbol{\Sigma}_N)=\frac{2}{\varepsilon^2}\text{trace}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{\Sigma}_N^{-2})\\ \text{s.t. }\left|\boldsymbol{\Sigma}_N\boldsymbol{M}_N\right|_1\leqslant 1\end{cases}$$

当上式取得最优解时，即等价于式(12)取得最优解。为简化推理过程，忽略式(12)的常数 $2/\varepsilon^2$。实际计算时加上该常数即可。

$$\begin{cases}\text{opt: }\min\limits_{\boldsymbol{\Sigma}_N} f(\boldsymbol{\Sigma}_N)=\sum\limits_{i=1}^{N}\frac{\boldsymbol{B}(:,i)^{\mathrm{T}}\boldsymbol{B}(:,i)}{\lambda_i^{\ 2}}\\ \text{s.t. }\boldsymbol{M}_N^{\mathrm{T}}\begin{pmatrix}\lambda_1\\ \lambda_2\\ \vdots\\ \lambda_N\end{pmatrix}\leqslant\boldsymbol{I}_{N\times 1},\ \lambda_i>0\end{cases} \tag{12}$$

其中，$\boldsymbol{B}(:,i)$ 表示矩阵 $\boldsymbol{B}$ 的第 i 列。

由分析可知，式(12)是一个线性约束下的凸优化问题。针对该问题，可直接采用SQP(sequential guadratic programming)方法[20]求得最优解。然而SQP方法是一种时间复杂度很高的算法，无法满足大规模数据的要求。实验表明，对于一般计算机，该方法最多只能满足 N 小于1 000的求解规模。因此，需要进一步研究更快速的方法求得式(12)的最优解。

4.3 快速求解最小误差

由于SQP方法求解复杂度很高，对于大规模数据，对角阵 $\boldsymbol{\Sigma}_N$ 求解是无法完成的。因此，有必要对 $\boldsymbol{\Sigma}_N$ 的求解进一步优化，并提出高效的解决方案。利用 $\boldsymbol{M}_N$ 和 $\boldsymbol{B}$ 之间的特殊性质，本文提出了一种高效的求解最小误差的算法——快速对角阵优化算法(fast diagonal matrix optimization algorithm, FDA)。当 $N=2^m-1$ 时，该算法可以在 $O(\lg N)$ 的时间复杂度下求解 $\boldsymbol{\Sigma}_N$ 的任意系数值 λ_i。该算法与未使用 $\boldsymbol{\Sigma}$ 前的方法有相当的求解效率，因此它保证了在不影响算法复杂度的前提下提高隐私数据发布的精确性。该算法是基于以下定理提出的。

定理4 令 $\boldsymbol{\Sigma}_N^{\ *}$ 表示 $\boldsymbol{\Sigma}_N$ 最优系数矩阵，则存在待定系数 α 使得 $\boldsymbol{\Sigma}_{2^m-1}^{\ *}$ 和 $\boldsymbol{\Sigma}_{2^{m-1}-1}^{\ *}$ 之间满足以下递推关系：

$$\boldsymbol{\Sigma}_{2^m-1}^{\ *}=\begin{pmatrix}\alpha\boldsymbol{\Sigma}_{2^{m-1}-1}^{\ *} & & \\ & 1-\alpha & \\ & & \boldsymbol{\Sigma}_{2^{m-1}-1}^{\ *}\end{pmatrix} \tag{13}$$

证明 对于矩阵 $\boldsymbol{B}_{2^m-1}$ 进一步分析可发现它满足以下特性：令 $a<2^{m-1}$，$b=2^{m-1}+a$。将其写成二进制形式可以描述为 $a=(x_{m-2}x_{m-3}\cdots x_0)_2$ 和 $b=(1x_{m-2}x_{m-3}\cdots x_0)_2$。根据算法2，不难发现 $\boldsymbol{B}_{2^m-1}(a,t)=1(t>0)$ 当且仅当 $\boldsymbol{B}_{2^m-1}(b,2^{m-1}+t)=1$。同时，由于 b 的 2^{m-1} 位为1，因此 $\boldsymbol{B}_{2^m-1}(b,2^{m-1})=1$

通过以上分析，可将 $\boldsymbol{B}_{2^m-1}$ 写成如下形式：

$$\boldsymbol{B}_{2^m-1}=\begin{pmatrix}\boldsymbol{B}_{2^{m-1}-1} & \boldsymbol{O}_{2^{m-1}-1\times 1} & \boldsymbol{O}_{2^{m-1}-1\times 2^{m-1}-1}\\ \boldsymbol{O}_{1\times 2^{m-1}-1} & 1 & \boldsymbol{O}_{1\times 2^{m-1}-1}\\ \boldsymbol{O}_{2^{m-1}-1\times 2^{m-1}-1} & \boldsymbol{I}_{2^{m-1}-1\times 1} & \boldsymbol{B}_{2^{m-1}-1}\end{pmatrix} \tag{14}$$

通过式(14)得：

$$\boldsymbol{B}_{2^m-1}(:,b)^{\mathrm{T}}\boldsymbol{B}_{2^m-1}(:,b)=\begin{pmatrix}\boldsymbol{O}_{2^{m-1}\times 1}\\ \boldsymbol{B}_{2^{m-1}-1}(:,b-2^{m-1})\end{pmatrix}^{\mathrm{T}}\begin{pmatrix}\boldsymbol{O}_{2^{m-1}\times 1}\\ \boldsymbol{B}_{2^{m-1}-1}(:,b-2^{m-1})\end{pmatrix}=\boldsymbol{B}_{2^{m-1}-1}(:,a)^{\mathrm{T}}\boldsymbol{B}_{2^{m-1}-1}(:,a)$$

下面将分析 $\boldsymbol{M}_{2^m-1}$ 与 $\boldsymbol{M}_{2^{m-1}-1}$ 之间的关系。

根据算法4，有 $\boldsymbol{M}_{2^m-1}(t,a)=1(1\leqslant t\leqslant 2^{m-1}-1)$ 当且仅当 $\boldsymbol{M}_{2^m-1}(2^{m-1}+t,b)=1$。满足 $\forall 1\leqslant t\leqslant 2^{m-1}-1$，$\boldsymbol{M}_{2^m-1}(2^{m-1},t)=1$。

因此，将 $\boldsymbol{M}_{2^m-1}$ 与 $\boldsymbol{M}_{2^{m-1}-1}$ 写成如下递推关系：

$$\boldsymbol{M}_{2^m-1}=\begin{pmatrix}\boldsymbol{M}_{2^{m-1}-1} & \boldsymbol{O}_{2^{m-1}-1\times 1} & \boldsymbol{O}_{2^{m-1}-1\times 2^{m-1}-1}\\ \boldsymbol{I}_{1\times 2^{m-1}-1} & 1 & \boldsymbol{O}_{1\times 2^{m-1}-1}\\ \boldsymbol{O}_{2^{m-1}-1\times 2^{m-1}-1} & \boldsymbol{O}_{2^{m-1}-1\times 1} & \boldsymbol{M}_{2^{m-1}-1}\end{pmatrix} \tag{15}$$

令 $\boldsymbol{R}_N$ 表示 $\boldsymbol{\Sigma}_N$ 求对角线元素组成的列向量，$\boldsymbol{R}_N=(\lambda_1\lambda_2\cdots\lambda_N)^{\mathrm{T}}$。当 $N=2^m-1$ 时，可将 $\boldsymbol{R}_{2^m-1}$ 拆分成3个部分，即：

$$\boldsymbol{R}_{2^m-1}=\begin{pmatrix}\boldsymbol{R}_{2^m-1}^{(1)\mathrm{T}} & \lambda_{2^{m-1}} & \boldsymbol{R}_{2^m-1}^{(2)\mathrm{T}}\end{pmatrix}^{\mathrm{T}} \tag{16}$$

其中 $\boldsymbol{R}_{2^m-1}^{(1)}=(\lambda_1\lambda_2\cdots\lambda_{2^{m-1}-1})^{\mathrm{T}}$

$\boldsymbol{R}_{2^m-1}^{(2)}=(\lambda_{2^{m-1}+1}\lambda_{2^{m-1}+2}\cdots\lambda_{2^m-1})^{\mathrm{T}}$

对于式(12),亦可将其拆分为以下3个子部分:

$$f(\boldsymbol{R}_{2^m-1})=\sum_{i=1}^{2^{m-1}-1}\frac{\boldsymbol{B}_{2^m-1}(:,i)^{\mathrm{T}}\boldsymbol{B}_{2^m-1}(:,i)}{\lambda_i^2}+ \quad ①$$

$$\frac{\boldsymbol{B}_{2^m-1}(:,2^{m-1})^{\mathrm{T}}\boldsymbol{B}_{2^m-1}(:,2^{m-1})}{\lambda_{2^{m-1}}^2}+ \quad ②$$

$$\sum_{i=1}^{2^{m-1}-1}\frac{\boldsymbol{B}_{2^m-1}(:,i)^{\mathrm{T}}\boldsymbol{B}_{2^m-1}(:,i)}{\lambda_{2^{m-1}+i}^2} \quad ③$$

令 $f^{(i)}(*)$ 分别表示这3个子部分,从而将上述表达式转换为3个子部分的和:

$$f(\boldsymbol{R}_{2^m-1})=f^{(1)}(\boldsymbol{R}_{2^m-1}^{(1)})+f^{(2)}(\lambda_{2^{m-1}})+f^{(3)}(\boldsymbol{R}_{2^m-1}^{(2)})$$

而对于其限制条件,有 $\boldsymbol{M}_{2^m-1}^{\mathrm{T}}\times\boldsymbol{R}_{2^m-1}\leqslant\boldsymbol{I}_{2^m-1\times1}$。依照式(15)、(16)展开得:

$$\begin{pmatrix}\boldsymbol{M}_{2^{m-1}-1}^{\mathrm{T}}\boldsymbol{R}_{2^m-1}^{(1)}+\lambda_{2^{m-1}}\boldsymbol{I}_{2^{m-1}-1\times1}\\ \lambda_{2^{m-1}}\\ \boldsymbol{M}_{2^{m-1}-1}^{\mathrm{T}}\boldsymbol{R}_{2^m-1}^{(2)}\end{pmatrix}\leqslant\boldsymbol{I}_{2^m-1\times1} \quad (17)$$

由式(17)可将限制条件分解成以下3个子条件:

$$\begin{cases}\boldsymbol{M}_{2^{m-1}-1}^{\mathrm{T}}\boldsymbol{R}_{2^m-1}^{(1)}\leqslant(1-\lambda_{2^{m-1}})\boldsymbol{I}_{2^{m-1}\times1} & ①\\ \lambda_{2^{m-1}}\leqslant1 & ②\\ \boldsymbol{M}_{2^{m-1}-1}^{\mathrm{T}}\boldsymbol{R}_{2^m-1}^{(2)}\leqslant\boldsymbol{I}_{2^{m-1}-1\times1} & ③\end{cases}$$

通过以上3个子限制条件,可知子条件①受限于子条件②中 $\lambda_{2^{m-1}}$ 的取值。因此,先假设 $\lambda_{2^{m-1}}$ 为待定系数,令 $\lambda_{2^{m-1}}=1-\alpha(0<\alpha<1)$。

式(12)取最优时,子部分①满足:

$$f(\boldsymbol{R}_{2^m-1}^{*})=\min_{\Sigma_{2^m-1}}f(\boldsymbol{R}_{2^m-1})\Rightarrow f^{(1)}(\boldsymbol{R}_{2^m-1}^{*(1)})=\min_{\Sigma_{2^m-1}}f^{(1)}(\boldsymbol{R}_{2^m-1}^{(1)})$$

$$\boldsymbol{M}_{2^m-1}^{\mathrm{T}}\times\boldsymbol{R}_{2^m-1}\leqslant\boldsymbol{I}_{2^m-1\times1}\Rightarrow\boldsymbol{M}_{2^{m-1}-1}^{\mathrm{T}}\boldsymbol{R}_{2^m-1}^{(1)}\leqslant(1-\lambda_{2^{m-1}})\boldsymbol{I}_{2^{m-1}\times1}$$

由于满足:

$$\boldsymbol{M}_{2^{m-1}-1}^{\mathrm{T}}\boldsymbol{R}_{2^m-1}^{(1)}\leqslant\alpha\boldsymbol{I}_{2^{m-1}\times1}\Leftrightarrow\boldsymbol{M}_{2^{m-1}-1}^{\mathrm{T}}\left(\frac{1}{\alpha}\boldsymbol{R}_{2^m-1}^{(1)}\right)\leqslant\boldsymbol{I}_{2^{m-1}\times1}$$

令 $\mu_i=\frac{1}{\alpha}\lambda_i$,$\boldsymbol{Q}_N=\frac{1}{\alpha}\boldsymbol{R}_N=(\mu_1\mu_2\cdots\mu_N)$,并将其代入式①后有:

$$f^{(1)}(\boldsymbol{R}_{2^m-1}^{(1)})=\sum_{i=1}^{2^{m-1}-1}\frac{\boldsymbol{B}_{2^m-1}(:,i)^{\mathrm{T}}\boldsymbol{B}_{2^m-1}(:,i)}{\lambda_i^2}=\frac{1}{\alpha^2}\sum_{i=1}^{2^{m-1}-1}\frac{\boldsymbol{B}_{2^m-1}(:,i)^{\mathrm{T}}\boldsymbol{B}_{2^m-1}(:,i)}{(\mu_i)^2}=\frac{1}{\alpha^2}f^{(1)}(\boldsymbol{Q}_{2^m-1}^{(1)})$$

通过以上分析,可将第①部分的子问题描述如下:

$$\begin{cases}\text{opt: }\min\limits_{\boldsymbol{Q}_{2^m-1}^{(1)}}\frac{1}{\alpha^2}f^{(1)}(\boldsymbol{Q}_{2^m-1}^{(1)})\Leftrightarrow\text{opt: }\min\limits_{\boldsymbol{Q}_{2^m-1}^{(1)}}f^{(1)}(\boldsymbol{Q}_{2^m-1}^{(1)})\\ \text{s.t. }\boldsymbol{M}_{2^{m-1}-1}^{\mathrm{T}}(\boldsymbol{Q}_{2^m-1}^{(1)})\leqslant\boldsymbol{I}_{2^{m-1}\times1},\mu_i>0\end{cases}$$

将 $\boldsymbol{Q}_{2^m-1}^{(1)}$ 用 $\boldsymbol{R}_{2^{m-1}-1}$ 代入,则问题等价于求解 $\boldsymbol{R}_{2^{m-1}-1}^{*}$,即 $\boldsymbol{\Sigma}_{2^{m-1}-1}^{*}$。由此可得 $\boldsymbol{Q}_{2^m-1}^{*(1)}=\boldsymbol{R}_{2^{m-1}-1}^{*}=\frac{1}{\alpha}\cdot\boldsymbol{R}_{2^m-1}^{*(1)}$,即 $\boldsymbol{R}_{2^m-1}^{*(1)}=\alpha\boldsymbol{R}_{2^{m-1}-1}^{*}$。

而第③部分可被看成是 α 等于1的特殊情况。因此,可以得出 $\boldsymbol{R}_{2^m-1}^{*(2)}=\boldsymbol{R}_{2^{m-1}-1}^{*}$。

综上所述,式(13)成立。 □

由定理4,可得形如式(12)的 $\boldsymbol{\Sigma}_N^{*}$ 递推关系,从而可由 $\boldsymbol{\Sigma}_{2^{m-1}-1}^{*}$ 的结果来求解关于 $\boldsymbol{\Sigma}_{2^m-1}^{*}$ 的最优结果。

假设已经求得 $N=2^{m-1}-1$ 下的最小均方误差 $err_{m-1}=\min\limits_{\Sigma_{2^{m-1}-1}}f(\Sigma_{2^{m-1}-1})$ 及 $\boldsymbol{\Sigma}_{2^{m-1}-1}^{*}$。将上述问题转化为关于 α 的最优化问题:

$$\begin{cases}\text{opt: }h(\alpha)=\min\limits_{\alpha}\left(\frac{err_{m-1}}{\alpha^2}+\frac{2^{m-1}}{(1-\alpha)^2}\right)+err_{m-1}\\ \text{s.t. }0<\alpha<1\end{cases} \quad (18)$$

定理5 当且仅当 $\alpha=\frac{\sqrt[3]{err_{m-1}}}{\sqrt[3]{err_{m-1}}+\sqrt[3]{2^{m-1}}}$ 时,$h(\alpha)$ 取得最小值,$h(\alpha)$ 最小值为 $(\sqrt[3]{err_{m-1}}+\sqrt[3]{2^{m-1}})^3+err_{m-1}$。

证明 首先对 $h(\alpha)$ 进行求导得:

$$h'(\alpha)=-2\frac{err_{m-1}}{\alpha^3}+2\frac{2^{m-1}}{(1-\alpha)^3} \quad (19)$$

令 $h'(\alpha)=0$,求得 $\frac{\alpha^3}{(1-\alpha)^3}=\frac{err_{m-1}}{2^{m-1}}$。

令 $g(\alpha)=\frac{\alpha^3}{(1-\alpha)^3}$,由于 α^3 单调递增,$(1-\alpha)^3$ 单调递减,从而 $g(\alpha)$ 单调递增。同时 $g(0)=0$,$g(1)=+\infty$,因此 $g(\alpha)=\frac{err_{m-1}}{2^{m-1}}$ 必有解,且唯一。

而 $\alpha=\frac{\sqrt[3]{err_{m-1}}}{\sqrt[3]{err_{m-1}}+\sqrt[3]{2^{m-1}}}$ 满足 $g(\alpha)=\frac{err_{m-1}}{2^{m-1}}$。因此 $\alpha=\frac{\sqrt[3]{err_{m-1}}}{\sqrt[3]{err_{m-1}}+\sqrt[3]{2^{m-1}}}$ 为 $h'(\alpha)=0$ 的唯一解。满足 $h\left(\frac{\sqrt[3]{err_{m-1}}}{\sqrt[3]{err_{m-1}}+\sqrt[3]{2^{m-1}}}\right)=\min\limits_{\alpha}h(\alpha)$。

然后将 $\alpha=\frac{\sqrt[3]{err_{m-1}}}{\sqrt[3]{err_{m-1}}+\sqrt[3]{2^{m-1}}}$ 代入得：

$$h\left(\frac{\sqrt[3]{err_{m-1}}}{\sqrt[3]{err_{m-1}}+\sqrt[3]{2^{m-1}}}\right)=\sqrt[3]{err_{m-1}}\left(\sqrt[3]{err_{m-1}}+\sqrt[3]{2^{m-1}}\right)^2+err_{m-1}+\sqrt[3]{2^{m-1}}\left(\sqrt[3]{err_{m-1}}+\sqrt[3]{2^{m-1}}\right)^2=\left(\sqrt[3]{err_{m-1}}+\sqrt[3]{2^{m-1}}\right)^3+err_{m-1}$$ □

因此，从以上结论可以得出以下关于均方误差 err_m 的递推公式：

$$err_m=\begin{cases}1, m=1\\ \left(\sqrt[3]{err_{m-1}}+\sqrt[3]{2^{m-1}}\right)^3+err_{m-1}, m>1\end{cases}$$

由该结果进一步推导出优化后均方误差，如下：

$$err_{\mathrm{FDA}}(2^m-1)=\frac{2}{\varepsilon^2}err_m \tag{20}$$

记 $N=2^{m-1}-1$ 时的迭代优化变量为 α_m，则由上式可得 $\alpha_m=\frac{\sqrt[3]{err_{m-1}}}{\sqrt[3]{err_{m-1}}+\sqrt[3]{2^{m-1}}}$。

而根据 $\alpha_i(1\leqslant i\leqslant m)$ 即可求得 $\boldsymbol{\Sigma}_m$ 中所有 λ_k 的值。具体计算步骤如算法4所示。

算法4 求解最优对角阵系数 $\lambda_k=coef(k,m)$

输入：系数的标号 k，数据规模 m

输出：λ_k 的值

1. $\lambda_k\leftarrow 1$; //初始化 λ_k
2. $kt\leftarrow k, t\leftarrow m; div\leftarrow 2^{m-1}$; // div 表示子问题的分割中点
3. while $div\neq kt$
4. if $kt<div$ then $\lambda_k\leftarrow\lambda_k\times\alpha_t$;
5. if $div<kt$ then $kt\leftarrow kt-div$;
6. $div\leftarrow\frac{div}{2}; t\leftarrow t-1$; //更新子问题
7. wend
8. $\lambda_k\leftarrow\lambda_k\times(1-\alpha_t)$;
9. return λ_k

通过算法4的步骤6可知，每次迭代过程都会将 div 除以2。因此，算法4的时间复杂度为 $O(\lg N)$。

基于上述理论，本文提出了完整的快速对角阵优化算法，如算法5所示。该算法利用 $\phi_{lowbit(t)}$ 代替 c_t 对算法进行了优化，达到空间重复利用的效果，进一步提高效率。

算法5 快速对角阵优化算法

输入：连续发布的上限 $T\in 2^m-1$，数据增量 $a_i, 1\leqslant i\leqslant T$；隐私预算 ε

输出：每次的发布结果 $\tilde{s}_t(1\leqslant t\leqslant T)$

1. for $t=1$ to T do //循环每次发布过程
2. $p\leftarrow \mathrm{lb}(lowbit(t))$;
3. $\phi_p\leftarrow a_i+\sum_{i=0}^{p-1}\phi_i$; //更新实际统计量
4. For $i=1$ to $p-1$ do $\phi_i=0$;
5. $\lambda_t\leftarrow coef(t,m)$; //由算法4计算系数
6. $\tilde{\phi}_1\leftarrow\lambda_t\times\phi_p+\frac{\tilde{\boldsymbol{L}}_1}{\varepsilon}$; //添加噪声
7. $k\leftarrow t, \tilde{s}_t\leftarrow 0$; //初始化发布值
8. while $k>0$
9. $q\leftarrow \mathrm{lb}(lowbit(k))$;
10. $\lambda_k\leftarrow coef(k,m)$
11. $\tilde{s}_t\leftarrow\tilde{s}_t+\tilde{\phi}_q/\lambda_k$
12. output $\tilde{s}_t$; //发布隐私数据
13. $k\leftarrow k-lowbit(k)$;
14. wend
15. end for

4.4 优化效果分析

针对本文提出的FDA算法，将通过分析对优化前后的均方误差进行对比，从而对优化效果进行评估。

考虑数据量 $N=2^m-1$ 的情况。根据递推公式(20)，定量求出优化后均方误差 err_{opt}；同时，结合定理2与矩阵 $\boldsymbol{B}$ 的性质，可推导出优化前的均方误差 err_{org} 递推式：

$$\begin{cases}err_m=\begin{cases}1, m=1\\ 2err_{m-1}+2^{m-1}, m>1\end{cases}\\ err_{\mathrm{org}}(N)=\frac{2m^2}{\varepsilon^2}err_m\end{cases} \tag{21}$$

根据式(20)及(21)，本文分别求出在不同规模下两者的均方误差。同时求出了它们均方误差之比 $r=err_{\mathrm{org}}/err_{\mathrm{FDA}}$ 来表示优化策略的优化效果。结果如图1所示。

通过两者的对比可以发现，无论多大规模的数据量，优化后方法的均方误差总是低于优化前的方法。说明了优化后的数据精确性取得了较大的改善。而从图1(b)可以发现，它们之间的均方误差之

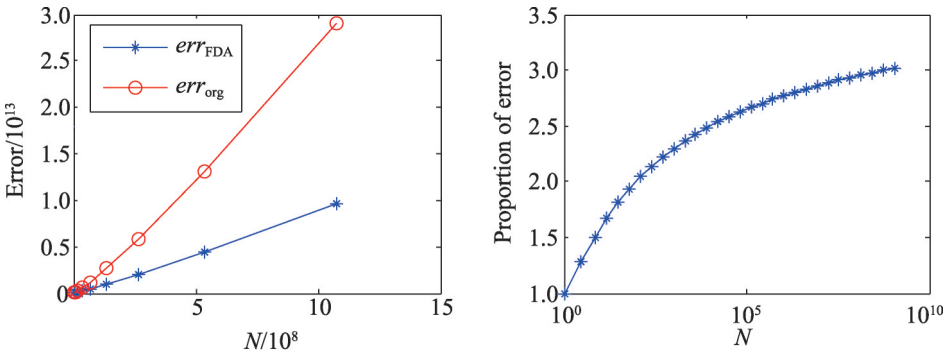


Fig.1 Comparison before and after improvement

图1 改进前后的比较

比呈单调递增曲线。说明了随N取值越大FDA算法的效果越好。

5 实验

为了测试FDA算法的效果,本文主要在差分隐私下对数据发布的精确度进行分析。首先,将前言中提到了一种数据连续发布的朴素方法[15]与FDA算法进行比较,以此来论证本文方法是有效的。其次,将FDA算法与基于二叉树的次数受限的连续数据发布方法(BM)[15]进行对比,以说明本文方法能够提高数据连续发布问题的精确性。由于静态数据发布问题是数据连续发布问题的一种特例,FDA算法也能应用于静态数据发布。为了说明FDA算法在静态数据发布领域也是有效的,本文将其应用于静态数据发布,与现有比较成熟的基于一致性优化区间树方法(Boost)和小波变换方法(Privelet)进行对比。

很显然,本文涉及的隐私保护模型发布数据的精确性与实际数据具有相对独立性。为了能进行更大规模的实验测试,本文主要采用虚拟数据进行实验,同时结合各个方法已有的理论分析,以保证实验的准确性。本次实验的环境为奔腾双核CPU T4200 2.00 GHz的计算机。实验所使用的语言为C++和Matlab。由于实验过程中ε的取值不会对实验结果产生重大影响,实验中都统一将ε设置为1。即满足1-差分隐私。另外,为了确保实验结果符合实际的误差期望,均进行了500次重复实验,然后取平均值作为最终的实验结果。

5.1 FDA算法与朴素方法

本实验对比了FDA算法和朴素方法的效果。该实验模拟数据规模为4 095的数据集对两者进行对比。实验结果如图2及图3所示。图2分别用蓝线和红线展示了FDA算法和朴素方法每次发布数据的均方误差结果。图3则分别展示了它们之间的均方误差之比$r=err_{simple}/err_{FDA}$以体现两者之间的效果差距。该比值越高说明FDA算法与朴素方法比较的效果越好。

由图2可发现朴素方法所产生的均方误差以直线方式增长;而FDA算法产生的均方误差随着更新次数的增长呈上下波动的现象,均方误差并不随着更新次数的增长而增长。其结果经过一段时间的数据发布,朴素方法由于均方误差增长过快而失去可

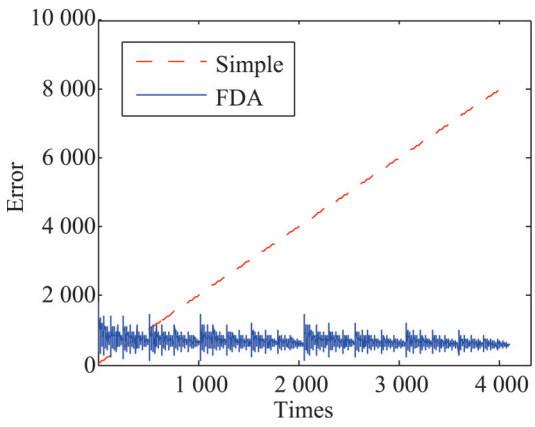


Fig.2 Effect of simple method and FDA

图2 朴素方法与FDA算法的效果

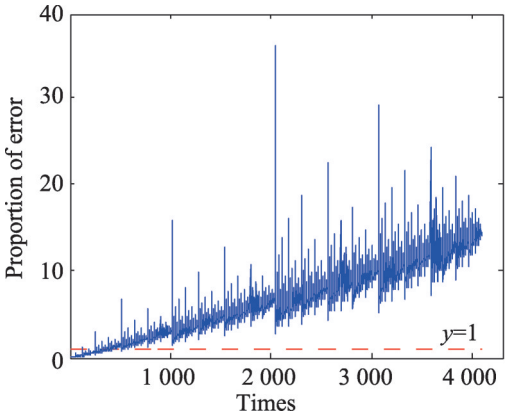


Fig.3 Proportion of error between simple method and FDA

图3 朴素方法与FDA算法的均方误差比

用性，而FDA算法仍维持在一个可控范围内。图3表明了当更新次数增多时，朴素方法在发布初期误差更低，而在数据发布达到一定次数后，FDA算法的误差则远低于朴素方法。进一步观察实验结果可知，它们之间效果好坏的分割点大约在400次数据发布前后。

5.2 FDA算法与二叉树方法

本实验对二叉树方法与FDA算法的效果进行了对比，并采用与上一个实验相同的数据集进行实验，效果如图4~图6所示。图4是利用二叉树方法进行数据连续发布时的均方误差；图5是FDA算法所产生的均方误差；图6则取两者之间的均方误差之比 $r=err_{BM}/err_{FDA}$ 以比较它们之间的效果。

由图6可以看出在绝大多数情况下，FDA算法所发布的数据比BM方法更加精确。这说明了相比于BM方法，FDA算法在精确性上取得了较大的提高。由图可知，FDA算法相比于BM方法精确性的提高集中在2至4倍之间，最高可达6倍。而分别比较图6和图5可知，通过FDA算法发布的数据产生的均方误差更加集中，而BM方法发布的数据波动幅度较大。这从侧面说明了FDA算法发布的数据更加稳定。

5.3 与静态数据发布方法对比

本实验对FDA算法与两种静态发布方法(Boost, Privelet)进行对比。实验采用的虚拟数据为离线数据，数据集规模为 $N=2^m$，$m=1,2,\cdots,25$；而FDA算法则减少一个数据点进行实验以符合算法本身的要求。本文主要描述了差分隐私下的数据连续发布问题，因此在对比时采用的查询为前 N 项和。为保证实验结果尽可能反映真实情况，本文结合文献[7-8,21]

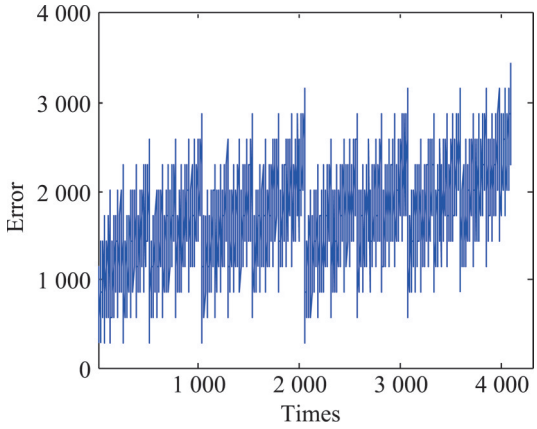


Fig.4 Effect of BM method

图4 BM方法的效果

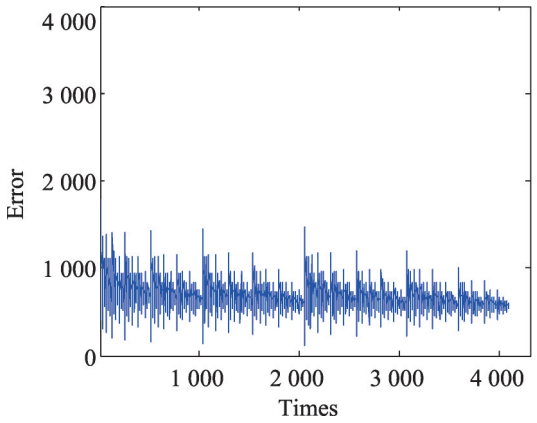


Fig.5 Effect of FDA

图5 FDA算法的效果

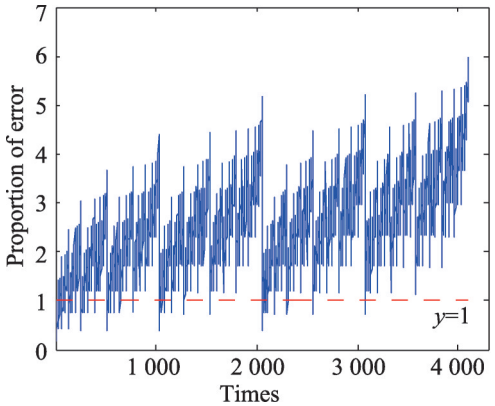


Fig.6 Proportion of error between BM method and FDA

图6 BM方法与FDA算法的均方误差比

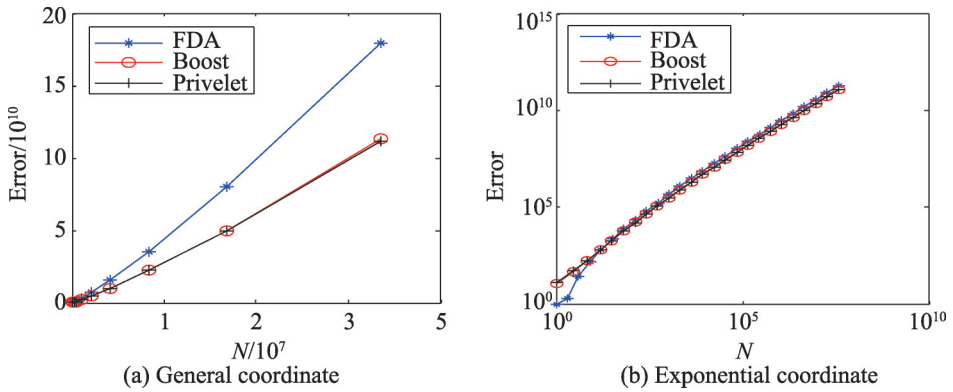


Fig.7 Comparison of three methods

图7 3种方法实验结果对比

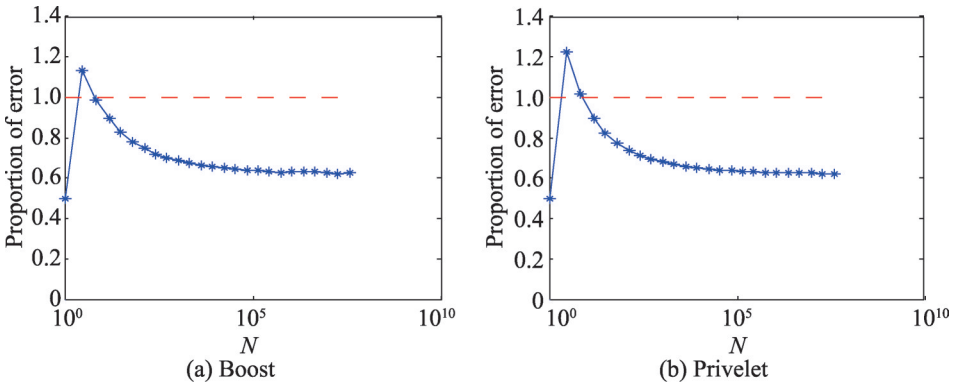


Fig.8 Proportion of error of static release methods and FDA

图8 FDA与静态发布方法的均方误差比

的相关理论分析进行验证。实验结果如图7和图8。图7用一般坐标系以及指数坐标系表示三者的均方误差。而在图8中,分别就区间树方法以及小波变换方法所产生的均方误差与FDA算法的均方误差做比较,求出不同规模数据下,它们与FDA算法的均方误差之比。

由图7可以看出,小波变换方法和区间树方法所产生的均方误差相当,且都低于FDA算法。然而,从图8可以看出,FDA算法在静态数据发布领域与这两者之间的精确性差距是有限的,并且稳定在一定的数值范围内。从图中可以看出,FDA算法的精确性大约为其他两种方法的0.625倍。这说明了FDA算法在静态数据发布领域也能有效地发布数据,但是在发布的精确性方面并不如专门发布静态数据的方法。

6 总结

本文描述了FDA算法,它可以快速并精确地处理差分隐私下的数据连续发布问题。通过理论分析和实验比较,FDA算法极大地改善了现有的连续数据发布方法的精确性,并且具有发布大规模数据的能力。然而,在静态数据发布领域,FDA算法目前还未能超越已有的成熟方法。说明FDA算法在一定程度上还有进一步提高的空间。

同时,FDA算法是针对一般连续数据发布问题提出的改进方法,因此该算法可以进一步应用于数据连续发布算法的拓展性问题上,从而进一步提高这些方法的效果,比如基于滑动窗口的流数据保护问题[22]、衰减累加的隐私保护问题[23]、无限数据流问题[24]等。而这也说明了本文提出的FDA算法具有较强的应用价值。

References:

[1] De Montjoye Y A, Hidalgo C A, Verleysen M, et al. Unique in the crowd: the privacy bounds of human mobility[R]. 2013.

[2] Dwork C, Mcsherry F, Nissim K, et al. Calibrating noise to sensitivity in private data analysis[C]//LNCS 3876: Proceedings of the 3rd Theory of Cryptography Conference, New York, USA, Mar 4-7, 2006. Berlin, Heidelberg: Springer, 2006: 265-284.

[3] Dwork C. Differential privacy: a survey of results[C]// LNCS 4978: Proceedings of the 5th International Conference on Theory and Applications of Models of Computation, Xi'an, China, Apr 25-29, 2008. Berlin, Heidelberg: Springer, 2008: 1-19.

[4] Dwork C, Lei Jing. Differential privacy and robust statistics[C]//Proceedings of the 41st Annual ACM Symposium on Theory of Computing, Bethesda, USA, May 31-Jun 2, 2009. New York, USA: ACM, 2009: 371-380.

[5] Acs G, Castelluccia C, Chen Rui. Differentially private histogram publishing through lossy compression[C]//Proceedings of the 2012 IEEE 12th International Conference on Data Mining, Brussels, Dec 10-13, 2012. Piscataway, USA: IEEE, 2012: 1-10.

[6] Xu Jia, Zhang Zhenjie, Xiao Xiaokui, et al. Differentially private histogram publication[J]. The VLDB Journal, 2013, 22(6): 797-822.

[7] Hay M, Rastogi V, Miklau G, et al. Boosting the accuracy of differentially private histograms through consistency[J]. Proceedings of the VLDB Endowment, 2010, 3(1/2): 1021-1032.

[8] Xiao Xiaokui, Wang Guozhang, Gehrke J. Differential Privacy via Wavelet Transforms[J]. IEEE Transactions on Knowledge and Data Engineering, 2011, 23(8): 1200-1214.

[9] Cormode G, Procopiuc C, Srivastava D, et al. Differentially private spatial decompositions[C]//Proceedings of the 2012 IEEE 28th International Conference on Data Engineering, Washington, USA, Apr 1-5, 2012. Piscataway, USA: IEEE, 2012: 20-31.

[10] Qardaji W, Yang Weining, Li Ninghui. Differentially private grids for geospatial data[C]//Proceedings of the 2013 IEEE 29th International Conference on Data Engineering, Brisbane, Australia, Apr 8-12, 2013. Piscataway, USA: IEEE, 2013: 757-768.

[11] Smith A. Privacy-preserving statistical estimation with optimal convergence rates[C]//Proceedings of the 43rd Annual ACM Symposium on Theory of Computing, San Jose, USA, Jun 6-8, 2011. New York, USA: ACM, 2011: 813-822.

[12] Zhang Jun, Zhang Zhenjie, Xiao Xiaokui, et al. Functional mechanism: regression analysis under differential privacy[J]. Proceedings of the VLDB Endowment, 2012, 5(11): 1364-1375.

[13] Sweeney L. *k*-anonymity: a model for protecting privacy [J]. International Journal on Uncertainty, Fuzziness and Knowledge Based Systems, 2002, 10(5): 557-570.

[14] Machanavajjhala A, Kifer D, Gehrke J, et al. *l*-diversity: privacy beyond *k*-anonymity[J]. ACM Transactions on Knowledge Discovery from Data, 2007, 1(1): 3.

[15] Chan T H H, Shi E, Song D. Private and continual release of statistics[J]. ACM Transactions on Information and System Security, 2011, 14(3): 26.

[16] Li Chao, Hay M, Rastogi V, et al. Optimizing linear counting queries under differential privacy[C]//Proceedings of the 29th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, Indianapolis, USA, Jun 6-11, 2010. New York, USA: ACM, 2010: 123-134.

[17] Yuan Ganzhao, Zhang Zhenjie, Winslett M, et al. Low-rank mechanism: optimizing batch queries under differential privacy[J]. Proceedings of the VLDB Endowment, 2012, 5(11): 1352-1363.

[18] Dwork C, McSherry F, Nissim K, et al. Calibrating noise to sensitivity in private data analysis[C]//LNCS 3876: Proceedings of the 3rd Theory of Cryptography Conference, New York, USA, Mar 4-7, 2006. Berlin, Heidelberg: Springer, 2006: 265-284.

[19] Fenwick P M. A new data structure for cumulative frequency tables[J]. Software: Practice and Experience, 1994, 24(3): 327-336.

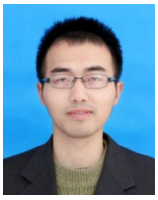
[20] Boyd S, Vandenberghe L. Convex optimization[M]. Cambridge, UK: Cambridge University Press, 2004.

[21] Qardaji W, Yang Weining, Li Ninghui. Understanding hierarchical methods for differentially private histograms[J]. Proceedings of the VLDB Endowment, 2013, 6(14): 1954-1965.

[22] Cao Jianneng, Xiao Qian, Ghinita G, et al. Efficient and accurate strategies for differentially-private sliding window queries[C]//Proceedings of the 16th International Conference on Extending Database Technology, Genoa, Italy, Mar 18-22, 2013. New York, USA: ACM, 2013: 191-202.

[23] Bolot J, Fawaz N, Muthukrishnan S, et al. Private decayed predicate sums on streams[C]//Proceedings of the 16th International Conference on Database Theory, Genoa, Italy, Mar 18-22, 2013. New York, USA: ACM, 2013: 284-295.

[24] Kellaris G, Papadopoulos S, Xiao X, et al. Differentially private event sequences over infinite streams[J]. Proceedings of the VLDB Endowment, 2014, 7(12): 1155-1166.



CAI Jianping was born in 1990. He is an M.S. candidate at College of Mathematics and Computer Science, Fuzhou University. His research interest is differential privacy.

蔡剑平(1990—),男,福建漳州人,福州大学数学与计算机科学学院硕士研究生,主要研究领域为差分隐私。

WU Yingjie was born in 1979. He received the Ph.D. degree in computer application technology from Southeast University in 2012. Now he is an associate professor at Fuzhou University. His research interests include data mining, data security and privacy protection, etc.

吴英杰(1979—),男,福建泉州人,2012年于东南大学计算机应用技术专业获得博士学位,现为福州大学副教授,主要研究领域为数据挖掘,数据安全,隐私保护等。

WANG Xiaodong was born in 1957. He received the M.S. degree from Fuzhou University in 1985. Now he is a professor at Fuzhou University. His research interests include data structures, design and analysis of algorithms, etc.

王晓东(1957—),男,1985年于福州大学获得硕士学位,现为福州大学教授,主要研究领域为数据结构、算法设计与分析等。