面向雾增强型工业物联网的多维安全查询方案

周由胜[1,2]，谭畅[1]，唐飞[2]
（1. 重庆邮电大学计算机科学与技术学院，重庆 400065；
2. 重庆邮电大学网络空间安全与信息法学院，重庆 400065）

摘 要：针对现有范围查询方案进行多维数据查询时缺乏隐私保护的问题，提出了一种带有隐私保护特性的面向雾增强型工业物联网多维安全查询方案。该方案首先将用户待查询的多个维度区间映射成一个查询矩阵；然后，构造辅助向量对查询矩阵进行分解，利用 BGN 同态加密对辅助向量进行处理形成查询陷门；最后，物联网设备终端利用同态特性将传感数据与查询陷门进行匹配。特殊辅助向量有效降低了方案空间复杂度，同态加密的自盲性保证了传感数据机密性和用户查询模式的隐私。仿真实验结果表明，所提方案的计算开销和通信开销较低。

关键词：雾增强；工业物联网；安全查询；隐私保护；BGN 同态加密
中图分类号：TP309.2
文献标识码：A
doi: 10.11959/j.issn.1000−436x.2020127

Multi-dimensional secure query scheme for fog-enhanced IIoT

ZHOU Yousheng[1,2], TAN Chang[1], TANG Fei[2]
1. College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China
2. College of Cyber Security and Information Law, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

Abstract: In view of the fact that most of the existing range query schemes for fog-enhanced IoT cannot achieve both multi-dimensional query and privacy protection, a privacy-preserving multi-dimensional secure query scheme for fog-enhanced IIoT was proposed. Firstly, the multiple ranges to be queried were mapped into a certain query matrix. Then, auxiliary vectors were constructed to decompose the query matrix, and then the auxiliary vector was processed by BGN homomorphic encryption to form a query trapdoor. Finally, with the homomorphic computation utilized by an IoT device terminal, the query trapdoor could be matched to its sensor data. Spatial complexity could be effectively reduced with the used specific auxiliary vectors. The confidentiality of sensor data could be ensured and the privacy of user's query mode could be protected by the homomorphic encryption property. Experiments results show that the computational and communication costs are relatively low.

Key words: fog-enhanced, IIoT, secure query, privacy-preserving, BGN homomorphic encryption

1 引言

工业物联网（IIoT, industrial Internet of things）是物联网系统与工业自动化系统的融合，具有全面感知、互联传输、智能处理等特点，是物联网技术未来的主要发展方向之一。工业物联网应用系统的重要任务之一是分析和处理物联网设备传感数据，如果将物联网设备的传感数据全部汇集到控制中心进行处理，不仅可能因网络拥塞引起系统较高时延，而且会使系统面临单点失效的风险。近年来兴起的边缘计算模式通过充分利用网络边缘侧设备的算力，可有效缓解传统集中式计算模式下存在的性能瓶颈和安全风险[1]。通过在工业物联网边缘部署雾设备，可以在网络边缘侧对物联网中传感数据

收稿日期：2020−01−06；**修回日期**：2020−04−18
通信作者：谭畅，tch1995@live.com
基金项目：国家自然科学基金资助项目（No.61702067）
Foundation Item: The National Natural Science Foundation of China (No.61702067)

进行预处理，提高服务质量[2-3]。

物联网设备的传感数据保护问题是雾增强型工业物联网系统需要考虑的重要问题之一。例如，在雾增强型工业互联网系统中，运维人员需要实时监控系统中的物联网设备是否正常运行，即查询物联网设备的传感数据是否处于合理的范围从而确定其运行状态。出于安全性考虑，查询过程中传感数据和查询结果均不能泄露给其他任何非授权方，因此必须设计安全的数据查询方案。在数据安全查询方面，目前诸多学者考虑了外包加密数据的查询，例如，Wang 等[4]使用布鲁姆过滤器构建查询索引，实现了对加密数据的模糊多关键词排序范围查询。Dai 等[5]针对双层无线传感器网络下的数据查询问题提出了一种融合桶分区、身份认证和校验码等技术的范围查询方法。Shen 等[6]研究了外包数据的多维度保密范围查询问题，提出了一种个体化的、权限灵活的查询方案。近年来，针对非外包加密数据的范围查询吸引了研究人员的关注，如 Lu[7]提出了一种范围查询矩阵化的方法，利用同态加密技术设计了单维安全范围查询方案。现有支持隐私保护特性的范围查询的研究主要集中在查询范围以及满足条件的查询子集的保密性上，不支持多维度查询，且通信开销较高。

由于在工业互联网中有些大型物联网设备有多种不同类型的传感器，因此对其进行状态监控需要获取归属该设备的多个传感数据，即进行多维数据查询，而采用传统的数据查询只能使用多次查询的方式实现，增加系统计算开销和通信开销，因此，有必要设计针对工业物联网的多维数据查询方案。此外，设计方案时还需要兼顾传感数据的机密性和用户查询模式保护问题。基于同态加密构造面向多维数据的查询陷门和陷门匹配是解决该问题的一个可行思路，如 BGN（Boneh- Goh-Nissim）同态加密算法[8]和 Paillier[9]提出的算法。用户将查询范围进行保密处理形成查询陷门，物联网设备在收到密文形式的查询陷门后与自身的传感数据进行匹配，利用同态加密算法的自盲性将传感数据形成新的密文并上报给雾设备。雾设备对其所属的物联网设备所提交的满足条件的传感数据密文进行汇集，并将汇集结果返回给查询用户进行计算和解密。

本文提出了一种面向雾增强型工业物联网的具有隐私保护特性的多维度安全范围查询方案，该方案采用查询区间矩阵化、基于辅助向量的矩阵分解和重构等方法降低存储开销和通信开销，利用同态加密实现隐私保护。本文方案首先将涉及多个不同量纲、不同起始点的 n 维数据的范围映射到一个查询矩阵，该矩阵的大小由总的查询区间大小而定，不受所需查询维度量纲的影响；其次，构造多个辅助向量将该矩阵分解和重构；再次，利用 BGN 同态加密设计查询陷门生成过程和陷门匹配过程；最后，通过仿真实验对方案的可行性和性能进行验证分析。

2 预备知识

本节介绍本文方案使用的 2 种基本数学方法，即大合数$\mathcal{N}=pq$ 阶双线性映射 $e:G\times G\to G_T$ 和 BGN 同态加密算法。

2.1 大合数阶双线性映射

令 p、q 是 2 个同长度的大素数，即其比特长度 $|p|=|q|$。令$\mathcal{N}=pq$，当在群 G 和 G_T 间存在满足如下 3 个属性的可计算的映射关系 $e:G\times G\to G_T$ 时，映射 $e(G, G_T)$可被称为合数阶双线性映射。

1) 双线性。对任意的$(g, h)\in G^2$，$a, b\in\mathbb{Z}_{\mathcal{N}}$，有 $e(g^a, h^b)=e(g, h)^{ab}$。

2) 非退化性。存在 $g\in G$，使 $e(g, g)$在$\mathcal{N}$阶群 G_T上。

3) 可计算性。存在一种高效算法，当$(g, h)\in G$ 时，所有 $e(g, h)\in G_T$都是可计算的。

大合数阶双线性参数生成器$\mathcal{C}$Gen($\mathcal{K}$)是一种概率算法，其以安全参数$\mathcal{K}$作为输入值，输出一个五元组($\mathcal{N}$, g, G, G_T, e)。其中，$\mathcal{N}=pq$，p 和 q 是 2 个 $\mathcal{K}$ bit 的素数；G 和 G_T是 2 个$\mathcal{N}$阶的群；$g\in G$ 是群 G 的一个$\mathcal{N}$阶生成元；$e:G\times G\to G_T$是一个非退化性的、可以高效计算的双线性映射。

令 g 是 G 的一个生成元，此时由 $h=g^q\in G$ 可以生成一个 p 阶子群 $G_p=\{g^0, g^1, \cdots, g^{p-1}\}$，而 $g'=g\in G$ 可以生成一个 q 阶子群 $G_q=G_p=\{g'^0, g'^1, \cdots, g'^{q-1}\}$。此时，$G$ 群上的子群区分（SGD, sub-group decision）难题[9]可以表述为：给定五元组($\mathcal{N}$, g, G, G_T, e)，如果元素 x 是随机从群 G 或其子群 G_q 中选取的，则确定 x 是否为 G_q 中元素是困难的。若此难题成立，则 BGN 同态加密算法的安全性得到保证[9]。

2.2 BGN 同态加密算法

BGN 同态加密算法是著名的全同态加密算法，由 3 个阶段组成，即密钥生成阶段、加密阶段和解

密阶段。

1) 密钥生成阶段

给定安全参数$\mathcal{K}$，合数阶双线性映射参数组($\mathcal{N}$, g, G, G_T, e)由生成器$\mathcal{G}$Gen($\mathcal{K}$)生成。此处$\mathcal{N}=pq$，其中 p, q 是 2 个$\mathcal{K}$ bit 的素数，G 和 G_T 是 2 个$\mathcal{N}$阶的群，$g\in G$ 是群 G 的一个$\mathcal{N}$阶生成元。设 $h=g^q$，h 是 G 的一个随机 p 阶生成元。此时，公钥 pk=($\mathcal{N}$, G, G_T, e, g, h)，私钥 sk=p。

2) 加密阶段

设仅包含整数的、容量由具体应用决定的消息空间 $\hat{S}=\{0, 1,\cdots, \Delta\}$，$\Delta\ll q$。当加密 $m\in\hat{S}$ 时，选取随机数 $r\in\mathbb{Z}_{\mathcal{N}}$，则密文 $c=E(m, r)=g^m h^r\in G$。

3) 解密阶段

给定密文 $c=E(m, r)=g^m h^r\in G$，明文可用密钥 sk 进行恢复。观察可知 $c^p=(g^m h^r)^p=(g^p)^m$，若要解密 m，相当于求解以 g^p 为底的 c^p 离散对数问题，而由于 $0\leqslant m\leqslant\Delta$，使用 Pollard lambda 算法求解这个问题的时间复杂度为 $O(\sqrt{\Delta})$。

此外，BGN 同态加密算法拥有自盲性，即，给定密文 $E(m, r)\in G$，有 $E(m, r+r')=E(m, r)h^{r'}\in G$ 是 m 的一个有效密文。

BGN 同态加密算法拥有以下同态特性。

① 群 G 上的加法同态性。给定 $E(m_1, r_1)\in G$ 和 $E(m_2, r_2)\in G$，有

$$E(m_1,r_1)E(m_2,r_2)=E(m_1+m_2,r_1+r_2)\in G \quad (1)$$

为了简洁表示，可以忽略随机数项，改写为 $E(m_1)E(m_2)=E(m_1+m_2)\in G$。

② 群 G 上的乘法同态性。给定 $E(m_1,r_1)\in G$，$m_2\in\hat{S}$，有 $E(m_1,r_1)^{m_2}=E(m_1m_2,r_1m_2)\in G$，即 $E(m_1)^{m_2}=E(m_1m_2)\in G$。

③ 群 G 到群 G_T 上的乘法同态性。给定 $E(m_1)\in G$ 和 $E(m_2)\in G$，有 $e(E(m_1), E(m_2))=E_T(m_1m_2)\in G_T$。

④ 群 G_T 上的加法同态性。给定 $E_T(m_1)\in G_T$ 和 $E_T(m_2)\in G_T$，有 $E_T(m_1)E_T(m_2)=E_T(m_1+m_2)\in G_T$。

⑤ 群 G_T 上的乘法同态性。给定 $E_T(m_1)\in G_T$，$m_2\in\hat{S}$，有 $E_T(m_1)^{m_2}=E_T(m_1m_2)\in G_T$。

3　带隐私保护特性的多维度查询方案

本文方案是一种面向各类雾增强工业物联网聚合式查询方案，本文中的多维度是指一个物联网设备的传感数据由多个不同维度的数据构成，如一个物联网设备的传感数据包含水温、电压、水量、材料余量等。根据实际需求，可能需要同时对该设备不同维度的传感数据进行查询。例如，在某个工厂中通过统计水温、电压、水量、材料余量的平均值，不仅可以及时了解并预测设备运行状态，还可以为优化生产工艺的流程提供依据。

需要说明的是，在实际的工业物联网环境中，不同维度的数据具有不同量纲和精度，例如电力消耗数据、水量消耗数据等可能会出现小数的情况。由于每个物联网设备在制造后不太可能通过 OTA（over the air）升级等软件手段来改变其探测精度，为了简化计算和适应多维度查询的需求，本文方案做如下处理。精度小于 1 的传感数据在参与计算时需要进行转换，如电力传感数据某个时间节点的值为 10.37 W，在计算时将其乘以 100，即变为 1 037 再进行范围查询。这样，可以在不损失精度的情况下实现多个维度的范围查询。事实上，某个维度的数据的扩增倍数可在物联网设备部署时写入设备。

3.1　系统模型

本文方案中有三类实体，即位于网络边缘侧的雾设备、位于雾设备管辖范围内的工业物联网设备 $D=\{D_1,D_2,\cdots,D_N\}$ 以及查询用户，方案模型如图 1 所示。

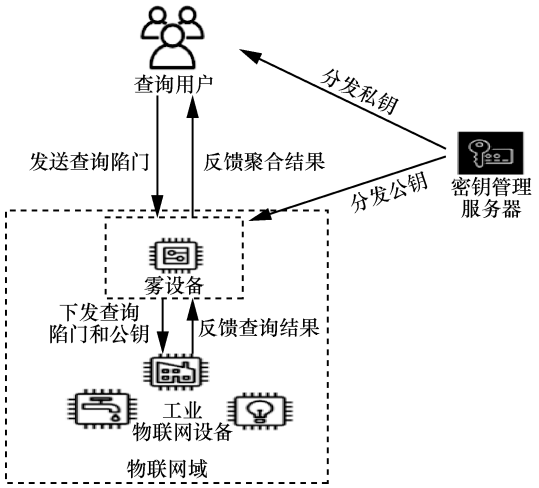

图 1　方案模型

1) 工业物联网设备 $D=\{D_1,D_2,\cdots,D_N\}$ 是一组物联网设备的集合，其分布于各个特定的物联网域中。每个物联网设备不只拥有探测和收集特定数据的能力，同时其也拥有数据传输的能力，使每个物联网设备 D_k 可以周期性地向其所属域内的雾设备上报传感数据。

2) 本文方案的模型中的雾设备均位于网络边

缘，每个雾设备有自身的管辖范围，这个管辖范围可以根据具体应用场景而定，如一台生产设备、一个车间，乃至一个厂区。雾设备可以在其管辖范围内对设备的传感数据进行收集和计算，并完成用户发来的查询请求。相对于物联网设备而言，雾设备拥有更强的计算和存储性能，并可以实时完成对物联网设备的传感数据收集和回应。

3) 本文方案的模型中，查询用户可以直接生成多个维度的范围查询并发送给雾设备，从雾设备上获得所需数据。例如，用户可能想要知晓有多少个雾设备范围内的设备传感数据可以同时满足维度 1，范围$[B_1, T_1]$，维度 2，范围$[B_2, T_2]$，…，维度 m，范围$[B_m, T_m]$；哪些设备的传感数据满足条件；满足条件的设备在各个维度上传感数据的均值是多少。雾设备可以收集到管辖范围内的所有设备反馈的满足条件的传感数据，然后根据其反馈数据生成相关度，最后将相关度和传感数据返回给用户。

3.2 多区间查询的矩阵化

本文方案在单查询区间矩阵化算法[7]的基础上进行了改进，使多维度范围查询得以矩阵化，且维持较低的通信开销。

通过观察可以发现，任意给定的查询区间均可分解成 2 种类型的行，分别为不完全行（PR, partial row）和连续行（CR, continuity row）。图 2 为 2 个查询区间映射后的查询矩阵，深灰色部分为 PR，浅灰色部分为 CR。这种矩阵结构为进一步压缩矩阵提供了可能。一般情况下，一个典型的查询区间可以分解得到 2 个 PR 和一个 CR。

0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	1
1	1	1	1	1	1	1	1
1	1	0	0	0	0	0	0
0	0	0	0	0	0	1	1
1	1	1	1	1	1	1	1
1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0

图 2 2 个查询区间映射后的查询矩阵

设总的查询区间数为 n。首先，定义 4 种特殊的辅助向量。

1) $\boldsymbol{X}_k=(x_{k1}, x_{k2},\cdots, x_{kn})$。其生成规则为，当第 k 个矩阵中第 i 行元素全为 1 时，设定 $x_{ki}=0$；否则 $x_{ki}=1$。

2) $\boldsymbol{Y}_k=(y_{k1}, y_{k2},\cdots, y_{kn})$。其生成规则为，当第 k 个矩阵中第 j 列元素有至少一个 1 时，设定 $y_{kj}=0$；否则 $y_{ki}=1$。

3) $\boldsymbol{X}_k'=(x'_{k1}, x'_{k2},\cdots, x_{kn})$。其生成规则为，当第 k 个矩阵中第 i 行元素含有至少一个 1 时，设定 $x'_{ki}=0$；否则 $x'_{ki}=1$。

4) $\boldsymbol{Y}_k'=(y'_{k1}, y'_{k2},\cdots, y'_{kn})$。其生成规则为，所有 $y'_{kj}=1$。

本文分别对矩阵中的不完全行和连续行进行处理。首先，将 n 个查询区间对应的不完全行独立拆分为多个矩阵 $\boldsymbol{R}_1, \boldsymbol{R}_2,\cdots, \boldsymbol{R}_{2n}$，将所有的连续行拆分成一个矩阵 $\boldsymbol{R}_C$，如图 3 所示。显然，原查询矩阵 $\boldsymbol{R}=\boldsymbol{R}_1\cup\boldsymbol{R}_2\cup\cdots\cup\boldsymbol{R}_{2n}\cup\boldsymbol{R}_C$。

构造所有拆分矩阵的向量 $\boldsymbol{X}_k, \boldsymbol{Y}_k, \boldsymbol{X}_k', \boldsymbol{Y}_k'$，并生成新的矩阵 $\mathbf{1}-\boldsymbol{X}^{\mathrm{T}}_k\boldsymbol{Y}_k$ 和 $\boldsymbol{X}'^{\mathrm{T}}_k\boldsymbol{Y}'_k$。根据向量的生成规则，新生成的 2 个矩阵中的每个元素均可表示为

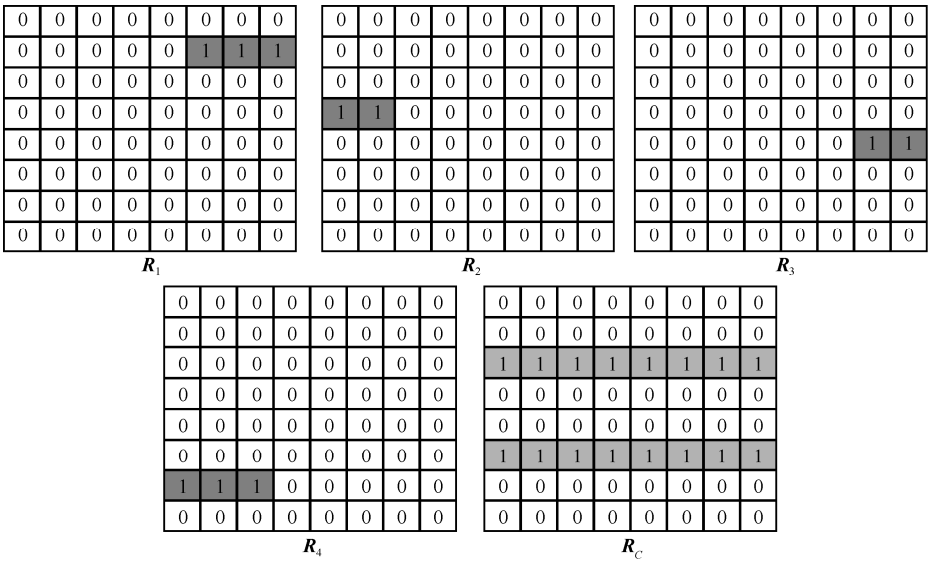

图 3 查询矩阵的拆分结果

$$\begin{cases} R_{\mathbf{1}-\boldsymbol{X}_k^{\mathrm{T}}\boldsymbol{Y}_k}(i,j)=1-x_{ki}y_{kj} \\ R_{\boldsymbol{X}_k^{\mathrm{T}}\boldsymbol{Y}_k'}(i,j)=x'_{ki}\,y'_{kj}=x'_{ki} \end{cases} \tag{2}$$

此时，矩阵中所有元素均可通过向量运算得到。用 $\boldsymbol{R}_k$ 表示与不完全行相关的矩阵，用 $\boldsymbol{R}_C$ 表示与连续行相关的矩阵，将矩阵的与运算表现为按位与运算，将矩阵的或运算表现为按位或运算，并整理为向量乘法和向量加法的形式，如式(3)所示。

$$\begin{aligned} &R(i,j)=(\vee_{k=1}^{2n}R_k(i,j))\vee R_C(i,j)= \\ &\vee_{k=1}^{2n}(R_{\mathbf{1}-\boldsymbol{X}_k^{\mathrm{T}}\boldsymbol{Y}_k}(i,j)\wedge R_{\boldsymbol{X}_k'^{\mathrm{T}}\boldsymbol{Y}_k'}(i,j))= \\ &\vee_{k=1}^{2n}((1-x_{ki}y_{kj})\vee x'_{ki}y'_{kj})\wedge((1-x_{Ci}y_{Cj})\vee x'_{Ci}y'_{Cj})= \\ &\sum_{k=1}^{2n}(1-x_{ki}y_{kj})x'_{ki}y'_{kj}+(1-x_{Ci}y_{Cj})x'_{Ci}y'_{Cj} \end{aligned} \tag{3}$$

在参与运算的 $8n$+4 个向量中，有 $4n$+1 个向量的元素均为 1，不需进行计算和存储，即{$\boldsymbol{X}_1, \boldsymbol{X}_2,\cdots, \boldsymbol{X}_{2n}, \boldsymbol{Y}'_1, \boldsymbol{Y}'_2,\cdots, \boldsymbol{Y}'_{2n}, \boldsymbol{Y}'_C$}。进一步地，$\boldsymbol{Y}_C$ 中的元素均为 0，不需存储和计算。因此仅需计算和存储{$\boldsymbol{Y}_1, \boldsymbol{Y}_2,\cdots, \boldsymbol{Y}_{2n}$, $\boldsymbol{X}'_1, \boldsymbol{X}'_2,\cdots, \boldsymbol{X}'_{2n}, \boldsymbol{X}'_C, \boldsymbol{X}_C$}共 $4n$+2 个向量即可完成计算和矩阵重构。对于矩阵中的任意元素 $R(i, j)$，由于 $y'_{kj}=1$，$x_{ki}=1$ ($k=1,2,\cdots,2n$)，且 $y_{C_j}=0, y'_{C_j}=1$，其值可通过式(4)计算得出。

$$R(i,j)=\sum_{k=1}^{2n}(1-y_{kj})x_{ki}+x_{Ci}=\begin{cases}1, R(i,j)\text{在查询区间中}\\ 0, \text{其他情况}\end{cases} \tag{4}$$

经过计算可知，向量 $\boldsymbol{X}_C$ 在重构中无作用，最终所需的向量个数为 $4n$+1 个。

3.3　数据查询流程

本文方案流程主要由三部分组成。首先，密钥管理服务器生成查询密钥，将私钥分发给用户，公钥分发给雾设备 FD_i。用户进行数据范围查询时，使用私钥对所查询范围和查询维度进行加密生成陷门，并将其发送给雾设备 FD_i。然后，雾设备在收到用户发来的查询陷门和查询维度密文后，将查询陷门下发给其所管辖的物联网设备进行查询。物联网设备在计算出本次查询的结果 ω 后，发送给雾设备进行聚合。雾设备通过聚合和计算得到 ζ 并反馈给用户。最后，用户将收到的 ζ 进行计算和解密，即可得到查询结果。整个流程如图 4 所示。下面将具体介绍本文方案的流程。

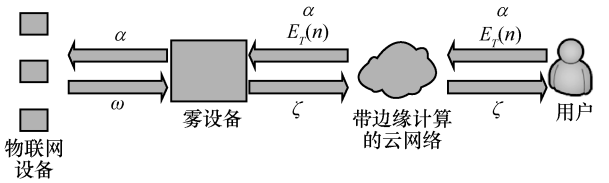

图 4　系统流程

为方便描述数据查询流程，将流程构造中使用到的主要参数在表1中列出。

表 1　　主要参数及其定义

参数	定义
α	查询陷门
σ	查询的匹配度，其值介于 0 到维度总数之间
γ	用户所发起查询的所在维度或物联网设备自身所在维度
β	查询偏移
ω	物联网设备反馈给雾设备的结果
ζ	雾设备向用户发送的查询结果
B_i	查询区间 i 的下界
T_i	查询区间 i 的上界
Build()	生成各类复合数据对象和向量等
Parse()	从各类复合数据对象和向量中提取数据

1) 密钥生成

给定安全参数$\mathcal{K}$，密钥管理服务器由生成器$\mathcal{C}$Gen($\mathcal{K}$)生成合数阶双线性映射参数组($\mathcal{N}$, g, G, G_T, e)。其中$\mathcal{N}=pq$, p, q 是 2 个$\mathcal{K}$ bit 的随机素数，G,G_T 是 2 个$\mathcal{N}$阶的群，$g\in G$ 是群 G 的一个$\mathcal{N}$阶生成元，e:$G\times G\rightarrow G_T$ 是大合数阶双线性映射。设 $h=g^q$，则 h 是 G 的一个随机 p 阶生成元。密钥管理服务器生成公钥 pk=($\mathcal{N}$, G, G_T, e, g, h)，私钥 sk=p。之后，密钥管理服务器分发私钥 sk 至查询用户，分发公钥 pk 至全体雾设备 FD_i，并由雾设备下发至其管辖域上的物联网设备。用户设定自己的消息空间为 $\hat{S}$={0, 1,$\cdots$, Δ}, $\Delta\ll q$。

2) 查询陷门生成

对于每个确定的使用场景，用户每次发起的查询涉及的维度数量和种类是确定的。本文方案中设定所查询数据的维度数量为 n，即用户查询的区间数量为 n。需要说明的是，流程中所指的查询区间[B_{query}, T_{query}]均为经过倍增处理的传感数据区间，物联网端也已对数据进行了相同的倍增处理，故不再赘述数据倍增过程。一次范围查询的陷门生成由以

下几个步骤构成。

① 数据映射和转换。将本次查询的任一查询维度设定为第一区间，之后按照以下规则确定每个区间在查询序列中的开始点。其中，l_i 为区间 i 的区间长度。

$$\begin{aligned}&P_{1_{\text{start}}}=0\\&P_{2_{\text{start}}}=l_1+l_1=2l_1\\&P_{3_{\text{start}}}=2l_1+l_2+l_2=2(l_1+l_2)\\&\quad\vdots\qquad\quad\vdots\\&P_{n_{\text{start}}}=2\sum_{i=1}^{n-1}l_i\end{aligned}\tag{5}$$

即任一区间 i 的起始点为区间 $i-1$ 的结束点后增加一个区间长度后的位置，以防止多个区间相连，形成太多 CR 区块。确定起始点后，任一区间 i 的结束点为

$$P_{i_{\text{end}}}=P_{i_{\text{start}}}+l_i=2\sum_{t=1}^{i-1}l_t+l_i\tag{6}$$

对于一个查询区间[B_{query^i}，T_{query^i}]而言，其查询区间内第 k 个元素 u_{ik} 满足

$$u_{ik}=B_{\text{query}^i}+k\tag{7}$$

该查询区间的数据偏移 β_i 可以由查询区间的开始点来确定，设偏移后该查询区间内第 k 个元素为 v_{ik}，则 β_i 和 v_{ik} 满足式(8)所示条件。

$$\begin{aligned}&\beta_i=B_{\text{query}^i}-P_{i_{\text{start}}}\\&v_{ik}=u_{ik}+\beta_i\end{aligned}\tag{8}$$

由最后一个查询区间的结束点确定矩阵阶数 m，然后构建 m 阶查询矩阵 $\boldsymbol{R}$。为了将查询区间映射到查询矩阵 $\boldsymbol{R}$ 中，将 v_{ik} 映射到矩阵元素 $R(i,j)$，如式(9)所示。

$$\begin{aligned}&m=\left\lceil\sqrt{P_{n_{\text{end}}}}\right\rceil\\&v_{ik}\to(i-1)m+j\\&R(i,j)=\begin{cases}1, B_{\text{query}^i}\leqslant(i-1)m+j\leqslant T_{\text{query}^i}\\0,\text{其他}\end{cases}\end{aligned}\tag{9}$$

矩阵的阶数 m 由最后一个查询区间的结束点而不是单个查询区间的上界决定，故每个向量的存储空间需求与多维度查询的单个区间的上界无关，只与所有查询区间长度之和有关，存储开销为 $O(m)$。

② 向量生成与加密。在生成查询矩阵 $\boldsymbol{R}$ 后，按照 3.2 节所述方法，由查询矩阵生成相应的向量 $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{X}', \boldsymbol{Y}'$，并筛选出所需的 $4n+1$ 个向量进行存储和加密。即

$$\begin{aligned}&2n\text{个}\boldsymbol{Y}_k=(y_{k1},y_{k2},\cdots,y_{km})\\&2n\text{个}\boldsymbol{X}'_k=(x'_{k1},x'_{k2},\cdots,x'_{km})\\&\boldsymbol{X}'_C=(x'_{C1},x'_{C2},\cdots,x'_{Cm})\end{aligned}\tag{10}$$

为便于计算，以向量 $\boldsymbol{Y}_k$ 进一步构造向量 $\overline{\boldsymbol{Y}}_k$，如式(11)所示。

$$\overline{\boldsymbol{Y}}_k=(\overline{y}_{k1}=1-y_{k1},\overline{y}_{k2}=1-y_{k2},\cdots,\overline{y}_{km}=1-y_{km})\tag{11}$$

根据式(4)，查询矩阵内所有元素 $R(i, j)$可以使用向量内元素表示

$$R(i,j)=\sum_{k=1}^{2n}(1-y_{kj})x'_{ki}+x'_{Ci}=\sum_{k=1}^{2n}\overline{y}_{kj}x'_{ki}+x'_{Ci}\tag{12}$$

用户选取 $2n+1$ 个随机数 $r_1, r_2,\cdots, r_{2n+1}$ 对所有的 $4n+1$ 个向量进行加密，得到

$$\begin{aligned}&E(\overline{\boldsymbol{Y}})=\{E(\overline{\boldsymbol{Y}}_1),E(\overline{\boldsymbol{Y}}_2),\cdots,E(\overline{\boldsymbol{Y}}_i)\},i=1,2,\cdots,2n\\&E(\boldsymbol{X}')=\{E(\boldsymbol{X}'_1),E(\boldsymbol{X}'_2),\cdots,E(\boldsymbol{X}'_i)\},i=1,2,\cdots,2n\\&E(\boldsymbol{X}'_C)=(g^{x'_{C1}}h^{r_{2n+1}},g^{x'_{C2}}h^{r_{2n+1}},\cdots,g^{x'_{Cm}}h^{r_{2n+1}})\end{aligned}$$

其中，

$$\begin{aligned}&E(\overline{\boldsymbol{Y}}_i)=\{E(\overline{\boldsymbol{Y}}_{i1}),E(\overline{\boldsymbol{Y}}_{i2}),\cdots,E(\overline{\boldsymbol{Y}}_{im})\}=\\&\{g^{\overline{y}_{i1}}h^{r_i},\ g^{\overline{y}_{i2}}h^{r_i},\cdots,g^{\overline{y}_{im}}h^{r_i}\}\\&E(\boldsymbol{X}'_i)=\{E(\boldsymbol{X}'_{i1}),E(\boldsymbol{X}'_{i2}),\cdots,E(\boldsymbol{X}'_{im})\}=\\&\{g^{x'_{i1}}h^{r_i},g^{x'_{i2}}h^{r_i},\cdots,g^{x'_{im}}h^{r_i}\}\end{aligned}\tag{13}$$

为了满足多维度查询的需求，还需要在原有的 $E(\overline{\boldsymbol{Y}}_i)$中添加 2 个值，即所查询的维度 γ_i 和该向量在运算时所需要的偏移量 β_i。

$$E'(\overline{\boldsymbol{Y}}_i)=(\gamma_i,\beta_i,E(\overline{\boldsymbol{Y}}_{i1}),E(\overline{\boldsymbol{Y}}_{i2}),\cdots,E(\overline{\boldsymbol{Y}}_{im}))\tag{14}$$

计算处理后的向量的哈希值 H_i。

$$\begin{aligned}&H=\{H_1,H_2,\cdots,H_i\}\\&H_i=\text{Hash}(E'(\overline{\boldsymbol{Y}}_i)\|E(\boldsymbol{X}'_i)\|E(\boldsymbol{X}'_C))\\&E'(\overline{\boldsymbol{Y}})=\{E'(\overline{\boldsymbol{Y}}_1),E'(\overline{\boldsymbol{Y}}_2),\cdots,E'(\overline{\boldsymbol{Y}}_i)\}\\&E(\boldsymbol{X}')=\{E(\boldsymbol{X}'_1),E(\boldsymbol{X}'_2),\cdots,E(\boldsymbol{X}'_i)\}\\&\alpha=\{E'(\overline{\boldsymbol{Y}}),E(\boldsymbol{X}'),E(\boldsymbol{X}'_C),H\}\end{aligned}\tag{15}$$

至此，查询陷门 α 生成完毕。将其发送至雾设备 FD_i，然后下发给其物联网域上的各物联网设备 D_k 进行查询。此外，用户还需要生成并向雾设备 FD_i 发送本次查询所需的维度密文 $E_T(n)$。

$$E_T(n)=e(E(n),E(1))\tag{16}$$

3) 物联网设备端的处理

与用户端发送的维度标识 γ 相对应，每个物联

网设备 D_k 拥有自己的维度标识 γ'_k。物联网设备依次提取用户发送的查询陷门 α 中的 $E'(\bar{\boldsymbol{Y}}_i)$和其对应的哈希值 H_i 来进行计算和比对。首先，物联网设备计算并比对用户发来的 H_i，若一致则从 $E'(\bar{\boldsymbol{Y}}_i)$中取出这条向量内的维度标识 γ_i 与设备自身的维度标识 γ'_k 比对。经过比对，物联网设备筛选出符合自身维度的查询，并将相关联的向量一同从查询陷门中提取出来并组成 **queryVector** 进行下一步计算。具体流程如算法 1 所示。算法 1 的时间复杂度为 $O(n)$ 与用户发来查询陷门内 $E(\bar{\boldsymbol{Y}}_i)$向量个数 n 有关。

算法 1　设备比较

输入　α, γ'_k

输出　**queryVector**

1) function DeviceCompare($E'(\bar{\boldsymbol{Y}})$, H)
2)　$I \leftarrow 0$
3)　**queryVector** $\leftarrow 0$
4)　while i=1 to n do
5)　　$H'_I \leftarrow$ Hash($E(\bar{\boldsymbol{Y}}_i) \| E(\boldsymbol{X}'_i) \| E(\boldsymbol{X}'_C)$)
6)　　if $H'_i = H_i$ then
7)　　if $\gamma'_k = \gamma_i$ then
8)　　**queryVector**←Build($E'(\bar{\boldsymbol{Y}}_i)$, $E(\boldsymbol{X}'_i)$, $E(\boldsymbol{X}'_{i+1})$, $E'(\bar{\boldsymbol{Y}}_{i+1})$, $E'(\boldsymbol{X}'_C)$)
9)　　break
10)　　end if
11)　　end if
12)　　i++
13)　end while
14)　return **queryVector**
15) end function

获取 **queryVector** 后，物联网设备首先从其中提取出本次查询的偏移量 β_k 对自身的传感数据 v^*_k 进行数据偏移，得到偏移后的值 v'_k，并使用 ElementShift 函数得到其在矩阵中的位置(i, j)。此时，物联网设备根据(i, j)从 **queryVector** 提取对应的向量进行计算。具体流程如算法 2 所示。每个物联网设备只需执行一次算法 2，时间复杂度为 $O(1)$。

算法 2　物联网设备处理

输入　**queryVector**, m_k, γ'_k

输出　ω_k

1) function DataShift(**queryVector**, m_k)
2)　$\beta_k \leftarrow$ Parse(**queryVector**)
3)　$v'_k \leftarrow v^*_k - \beta_k$
4) end function
5) function ElementShift(v'_k)
6)　while i=1 to m do
7)　　while j=1　to m do
8)　　if $(i-1)\ m + j = v'_k$ then
9)　　break
10)　　end if
11)　　end while
12)　end while
13) return i, j
14) end function
15) function DeviceCompute(v'_k, **queryVector**)
16)　i,$j \leftarrow$ ElementShift(v'_k)
17)　$E(\bar{\boldsymbol{Y}}_{1j})$,$E(\bar{\boldsymbol{Y}}_{2j})$, $E(x'_{1i})$, $E(x'_{2i})$, $E(x'_{Ci}) \leftarrow$ Parse(**queryVector**)
18)　r_{k1}, $r_{k2} \leftarrow$ Random()
19)　$c_k \leftarrow e(E(\bar{y}_{1j}), E(x'_{1i}))e(E(\bar{y}_{2j}),E(x'_{2i}))$, $e(E(\bar{y}_{Cj}), E(x'_{Ci}))\, e(g,h)^{r_{k1}}$
20)　$s_k \leftarrow c_k^{v_k^*} e(g,h)^{r_{k2}}$
21)　$\omega_k \leftarrow$ Build(c_k, s_k, γ'_k)
22)　return ω_k
23) end function

对于算法 2 中所述的 c_k 和 s_k，有

$$c_k = e(E(\bar{y}_{1j}), E(x'_{1i}))e(E(\bar{y}_{2j}), E(x'_{2i}))e(E(\bar{y}_{Cj}), E(x'_{Ci}))e(g,h)^{r_{k1}} = E_T(\bar{y}_{1j}x'_{1i} + \bar{y}_{2j}x'_{2i} + x'_{Ci}) = E_T(\boldsymbol{R}_k(i,j))$$

$$s_k = c_k^{v_k^*} e(g,h)^{r_{k2}} = E_T(\boldsymbol{R}_k(i,j))^{v_k} e(g,h)^{r_{k2}} = E_T(\boldsymbol{R}_k(i,j)v_k^*) \quad (17)$$

通过执行算法 1、算法 2，可以将物联网设备的传感数据 v^*_k 转换为查询矩阵对应位置的值在 G_T 群的映射 c_k，至此完成一次查询。此外，还可以通过 s_k 在查询匹配的情况下返回物联网设备的实际传感数据。完成计算后，将 ω_k 发送给物联网设备所属域的雾设备 FD_i。

4) 雾设备端的处理

雾设备 FD_i 接收到其下属的 k 个物联网设备发来的 ω_k 后，从中提取物联网设备对本次 n 维度查询的结果 c_k 并进行计算。雾设备将所有维度数据 c_k 相乘，根据 BGN 算法的同态性，得到的结果即 k 个维度上反馈的所有结果之和，FD_i 本次查询的匹配程度 σ_i 即

为此和值与用户发来的经加密的查询维度信息 $E_T(n)$ 的差值。FD_i 将 σ_i 和所有 ω_k 的值构建成 ζ_i 发送给用户。具体流程如算法 3 所示。算法 3 的时间复杂度 $O(n)$ 与雾设备 FD_i 所属物联网设备个数 k 有关。

算法 3 雾设备处理

输入 $\omega_1, \omega_2, \cdots, \omega_k, E_T(n)$

输出 ζ_i

1) function FDProcess($\omega_k, E_T(n)$)
2) while k=1 to n do
3) $c_k \leftarrow$ Parse(ω_k)
4) $\sigma_i \leftarrow \sigma_i c_k$
5) end while
6) $\sigma_i \leftarrow \dfrac{E_T(n)}{\sigma_i}$
7) $\zeta_i \leftarrow$ Build($\sigma_i, \omega_1, \omega_2, \cdots, \omega_k$)
8) return ζ_i
9) end function

5) 用户解析数据

用户收到 FD_i 发来的数据 ζ_i 后，首先提取查询匹配程度值 σ_i 并解密。当且仅当 σ_i=0 时，雾设备 FD_i 返回的结果完全与查询匹配。用户对雾设备返回的完全匹配的数据分维度进行累乘，并计算完全匹配的设备个数。具体流程如算法 4 所示。算法 4 的时间复杂度与雾设备 FD_i 的个数 n 有关，为 $O(n)$。

算法 4 用户数据解析

输入 $\zeta_1, \zeta_2, \cdots, \zeta_i$

输出 $S_1, S_2, \cdots, S_\gamma$, VD

1) function UserProcess($\zeta_1, \zeta_2, \cdots, \zeta_i$)
2) while i=1 to n do
3) $\sigma_i \leftarrow$ Parse(ζ_i)
4) $S_1, S_2, \cdots, S_\gamma \leftarrow 0$
5) VD $\leftarrow$ 0
6) if Decrypt(σ_i = 0)
7) $\omega_1, \omega_2, \cdots, \omega_k \leftarrow$ Parse(ζ_i)
8) while 1 to k do
9) $s_k, \gamma'_k \leftarrow$ Parse(ω_k)
10) $S_{\gamma'_k} \leftarrow S_{\gamma'_k} s_k$
11) end while
12) VD++
13) else
14) $\zeta_i \leftarrow$ NULL
15) end if
16) end while
17) return $S_1, S_2, \cdots, S_\gamma$, VD
18) end function

此时，VD 即为返回的有效数据的个数，将 S_γ 解密后即可得到维度 γ 下符合查询条件数据之和。经过处理后，不匹配的 ζ_i 被全部置 0，剩余的元素来自返回了有效数据的雾设备，即可定位返回了有效数据的具体雾设备。

4 安全性分析

本节对本文方案的安全性进行分析，包括前向安全性、后向安全性、隐私保护性、不可链接性等关键安全特性，并且介绍了近年来一些针对加密范围查询方案设计的攻击模式，分析了本文方案在面对这些攻击模式时的安全性。

1) 前向安全及后向安全

假定敌手 $\mathcal{A}$ 获得了某次查询中所使用的私钥 sk=p，其仅能对当次查询的密文进行解密。由于在每次进行查询时，用户均会使用新生成的公钥和私钥进行查询，敌手 $\mathcal{A}$ 使用私钥 sk=p 仅可得知用户生成密钥所用的安全参数 $\mathcal{K}$, 无法由此计算出用户此前所生成密钥的任何信息，从而无法解密除当次查询数据外此前任何一次查询的数据。由此，本文方案满足前向安全及后向安全。

2) 查询模式隐私保护

假定敌手 $\mathcal{A}$ 拦截了用户 U 在某次查询中使用的陷门 $\alpha=\{E(\bar{\boldsymbol{Y}}_i), E(\boldsymbol{X}'), E(\boldsymbol{X}'_C), H\}$，由于其在公开信息中仅能获得公钥 pk=($\mathcal{N}$, G, G_T, e, g, h)和对当次查询的维度总量的加密信息 $E_T(n)$，因此其无法解密出查询陷门的具体内容，也无法获知当次查询的查询维度总量。进一步地，由于 BGN 算法具有语义安全性，攻击者无法区分陷门中加密向量的内容，也无法通过多次拦截陷门分析出用户的查询规律。

除敌手 $\mathcal{A}$ 外，物联网设备 D_k 和雾设备 FD_i 也无法获知查询陷门的具体信息。由于本文方案与传统的查询方案不同，使用了同态加密计算来取代传统的比对过程，D_k 根据查询偏移量 β 来确定自身原始传感数据 v^*_k 并提取查询陷门内对应的加密向量进行计算。在这个过程中，D_k 并不能获取查询区间[B_k, T_k]的具体值。而 FD_i 在方案中仅对运行于其所在物联网域内的物联网设备发来的反馈 ω_k 进行聚合，也无法获知各维度上具体的查

询区间。因此，本文方案的查询陷门不会泄露任何有用的信息，如查询模式、查询区间等，查询陷门的隐私得到了保护。

3) 传感数据机密性

假定敌手$\mathcal{A}$拦截了物联网设备 D_k 在某次查询中发送到雾设备 FD_i 的消息 ω_k，由于其在公开信息中不能获取除公钥和当次查询维度总量以外的信息，其无法解密出 ω_k 的具体内容，无法获知 D_k 是否满足本次查询或得到 D_k 的原始传感数据 v^*_k。由于物联网设备向雾设备发送信息时使用了不同的随机数，敌手$\mathcal{A}$无法通过多次拦截 ω_k 来分析出任何有用的信息。

假定敌手$\mathcal{A}$拦截了雾设备 FD_i 发往用户的消息 ζ_i，由于其无法获得私钥 sk=p，其无法对 ζ_i 进行解密，无法获知满足查询条件的物联网设备及其传感数据 v^*_k。与前述相同，随机数的引入使敌手无法通过多次拦截 ζ_i 来分析获知查询规律等有用的信息。

除敌手$\mathcal{A}$外，雾设备 FD_i 也无法获知物联网设备 D_k 的原始传感数据 v^*_k 是否满足查询。FD_i 收到 D_k 的反馈 ω_k 后，由于其仅能掌握公钥 pk，其无法解密 ω_k 的具体内容，无法获知 D_k 是否满足本次查询或得到 D_k 的原始传感数据 v^*_k。因此，除查询用户和物联网设备外的任何一方均无法获知物联网设备的传感数据 v^*_k，传感数据的机密性得到保证。

4) 不可链接性

本文方案中，由于随机数的使用，所有由用户 U 生成的查询陷门具有随机性，即使由同一个用户发送的 2 个或多个查询陷门，外部攻击者无法确定这些查询是否来自同一个用户，也就意味着攻击者无法将两次不同的查询关联到某一个特定用户。因此，本文方案提供了不可链接性，攻击者无法通过拦截查询陷门的方式来关联查询者身份。

5) 查询陷门完整性

为了防止数据传输过程中可能出现的数据完整性损坏，本文方案中，用户 U 生成查询陷门时，将成对的查询向量 $E(\bar{\boldsymbol{Y}}_i)$、$E(\boldsymbol{X}'_i)$和 $E(\boldsymbol{X}'_C)$进行连接并计算消息指纹 $\mathrm{Hash}(E(\bar{\boldsymbol{Y}}_i)\|E(\boldsymbol{X}'_i)\|E(\boldsymbol{X}'_C))$，生成的哈希值被发送给各物联网设备 D_k。D_k 在提取某一组加密向量内容前，首先利用哈希值对消息进行校验，若校验未通过，该组向量被视作已损坏向量并丢弃。通过这种方式，查询陷门的数据完整性得到了保证。

6) 密钥更新

本文方案中，对物联网设备传感数据所进行的查询过程本质上是一个基于 BGN 的同态加密计算过程，其过程中不涉及解密操作，物联网设备和雾设备仅需获知系统公钥 pk，而私钥 sk 仅由查询用户掌握。因此，密钥管理和更新涉及实体较少，密钥管理成本较低。

7) 对其他攻击方式的防护

除上述常见的安全攻击类型外，近年来，出现了多种针对加密范围查询的攻击方式。其中，Islam 等[10]使用访问模式泄露的方式对加密数据查询进行攻击。Naveed 等[11]对属性加密方案中的 DTE（deterministic encryption）和 OPE（oder preserving encryption）加密方式实施攻击，并使用加密信息和公开的辅助数据完成明文恢复。Kellaris 等[12]定义了 2 种基本的泄露来源，如访问模式泄露和通信容量泄露，并开发了一种通用的、适用于任何支持范围查询且其访问模式或通信容量存在泄露的系统的重放攻击方式。Lacharité 等[13]使用范围查询中泄露的包括查询模式、排序信息在内的数据，提出了一种改进的对加密数据的重放攻击。

与文献[10-13]讨论的数据库查询场景有所不同，本文方案主要考虑雾增强工业物联网中进行具有隐私保护特性的范围查询，是一种聚合式的隐私保护查询。本文方案可以保护范围查询中的上下界与符合查询条件的设备子集的隐私，不会泄露查询模式及关键词频度等相关信息。此外，对于任何一次范围查询，本文方案返回的查询结果的数据体积均相同，不会有流量信息被泄露。因此，本文方案可以抵抗上述几种较为典型的对安全范围查询的攻击。

5　性能分析

本节将本文方案与现有同类方案的性能进行分析比较，主要包括查询陷门生成阶段、服务器查询阶段的计算开销和通信开销的比较。所有仿真实验均在一台配置为 Intel i7-9750H 2.60 GHz，内存为 16 GB，运行 Windows 10 1903 系统的笔记本电脑上进行，使用 Java.Math 的大数计算库和 JPBC 库[14]进行实现算法。此外，对加法运算、哈希运算等时间消耗较低的运算，在比较中忽略不计。

5.1　计算开销

1) 各阶段计算开销

方便起见，定义 T_m、T_o、T_p 和 T_{em} 分别表示单次整数乘法、整数模幂、双线性映射和椭圆曲线上点乘的时间开销。设 l 为每个维度上查询区间的长

表 2 各阶段计算开销对比

方案	原始数据加密阶段	陷门生成阶段	查询阶段
文献[7]方案	—	$10k\lceil\sqrt{l}\rceil(T_o)$	$2kT_{em}+3kT_p+kT_o$
文献[15]方案	$2klT_m+klT_o$	$4kT_m$	$2kT_m$
文献[16]方案	$2klT_o+klT_m$	$(2k+1)T_o$	$3kT_m+2kT_o$
文献[17]方案	$2klT_o+qT_{em}$	$5kT_m+6kT_o$	$2kT_{em}+kT_p$
本文方案	—	$(8k+2)\lceil\sqrt{l}\rceil(T_o)$	$2kT_{em}+3kT_p+kT_o$

度，k 为维度总数，q 为单个查询区间的上界。对于只支持单个维度查询的方案，如文献[7]方案，将一次多维度查询视作进行多次单维度查询来处理。各阶段计算开销对比如表 2 所示。

本文方案和文献[7]方案均是由物联网设备直接在网络边缘侧参与查询，故传感数据不需要加密存储于服务器上，其加密阶段不产生计算开销。

2) 全维度查询计算开销

本节考虑在维度总数一定的情况下，每次查询均对所有维度的数据进行查询时的计算开销。设维度总数 k=10，查询区间长度 l=36，查询区间上界 q=50，则使用文献[7]方案、文献[15]方案、文献[16]方案、文献[17]方案和本文方案完成一次全维度查询分别约需 22.641 ms、15.864 ms、20.191 ms、21.638 ms 和 20.360 ms。

图 5 显示了本文方案和对比方案进行指定维度的全维度查询计算开销的比较结果。其中，本文方案与文献[7]方案在查询过程中引入了双线性映射运算，计算开销相较于使用整数加密运算的文献[15]方案和文献[16]方案更高，但是文献[15]方案、文献[16]方案和文献[17]方案需预先将传感数据加密发送到第三方数据存储服务器，该过程必然存在时延，且产生加密开销。本文方案与文献[7]方案则不需要加密和发送传感数据，支持用户对传感数据进行实时查询。文献[7]方案由于仅支持单维度查询，在进行多维度查询时，其整体计算开销高于本文方案。

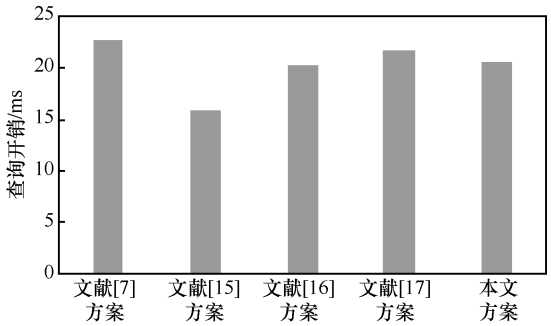

图 5 全维度查询计算开销比较

3) 部分维度查询计算开销

部分维度数量查询是指在维度总数一定的情况下，每次查询均仅对部分维度的传感数据进行查询。设维度总数 k=10，查询区间长度 l=36，查询区间上界 q=50，分别进行维度为 5、7、9 的部分维度查询。图 6 展示了本文方案与对比方案在进行部分维度查询时的计算开销对比结果。需要说明的是，考虑到即使进行部分维度查询，数据存储服务器存储的数据数量也应与进行全维度查询时一致，因此对于需要对传感数据加密存储的方案，其加密开销仍等价于全维度查询的计算开销。虽然本文方案与文献[7]方案、文献[17]方案都因使用双线性映射运算导致查询阶段开销较高，但由于本文方案与文献[7]方案是对传感数据进行实时查询，不需要对传感数据进行预先加密和存储，所以在给定维度总数的条件下进行较少维度查询时（维度为 5、7 查询），总体计算开销仍比需要加密存储传感数据类方案低。而文献[7]方案由于仅支持单维度查询，在进行多维度查询时需要重复发送查询陷门，其整体计算开销也高于本文方案。

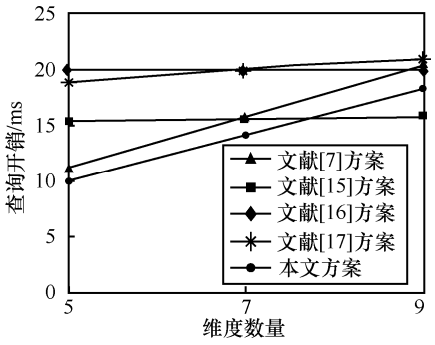

图 6 不同维度数量查询计算开销比较

4) 不同区间长度查询计算开销

不同区间长度的查询是指在维度总数、查询区间长度上限、查询区间上界一定的情况下，每次查询仅对有限区间长度进行查询。设维度总数 k=10，查询区间长度上限 l=100，查询区间上界 q=100，分别进行 25%、50%、75%长度区间查询时，将本文方案与对

比方案在完成一次查询时的计算开销进行比较，结果如图 7 所示。

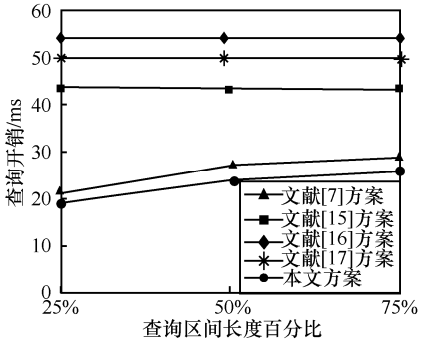

图 7　不同区间长度查询的计算开销比较

根据本文部分维度数量查询计算开销中分析可知，对于需要对原始传感数据加密存储类方案，其加密开销与全维度查询时计算开销相同。除本文方案和文献[7]方案外，其他 3 个查询方案的计算开销均与查询区间长度无关，而与查询维度总数有关。但是，由于文献[15]方案、文献[16]方案和文献[17]方案需要预先对原始传感数据进行加密存储，使其在查询区间长度较大、维度总数较高时，总体计算开销较高。文献[7]方案由于不支持多维度查询，在进行多维度查询时重复发送单维度查询陷门，其所使用的向量个数比本文方案多，计算开销较高。

5.2　通信开销

本节对本文方案和对比方案的通信开销进行比较。设定维度总数 k=10，查询区间长度 l=36，查询区间上界 q=50 的全维度查询，对不同方案全流程的通信开销进行比较。为了公平比较各方案的通信开销，做以下假设。

1) 各方案安全参数（或密钥长度）均为 128 bit。

2) 各方案所用哈希算法均为 SHA-1，因此，各方案的哈希摘要长度均为 160 bit。

3) 各方案所用随机数均为 64 bit。

本文方案的通信过程主要在于查询陷门发送阶段、物联网设备反馈雾设备阶段和雾设备反馈查询用户阶段。由于本文方案是将查询矩阵向量化进行查询，查询陷门通信开销较大，其余阶段开销较小。查询陷门发送阶段生成的查询陷门通信开销约为 31 016 B，物联网设备反馈雾设备阶段通信开销约为 1 365 B，雾设备反馈查询用户部分通信开销约为 1 370 B，共计约 33 751 B。表 3 给出了本文方案和对比方案的通信开销。

表 3　　通信开销比较

方案	通信开销/B
文献[7]方案	40 535
文献[15]方案	17 520
文献[16]方案	32 880
文献[17]方案	26 830
本文方案	33 751

文献[15]方案使用了改进自 SHE 加密方案的整数域上的同态加密算法，其在相同条件下密文长度较小，但其算法易受密文中噪声影响，当密文中噪声较大时可能影响解密。文献[16]方案较本文方案通信开销略低，但其与文献[15]方案均是对传感数据加密并传输到服务器的数据进行查询，且其方案需要 2 个服务器参与计算，传输时延较大，不能实现传感数据的实时查询。文献[17]方案查询陷门体积较小，使其通信开销较本文方案和文献[7]方案低，但其计算开销较高，且不支持传感数据的实时查询。支持实时数据查询的文献[7]方案由于只支持单维度查询，对个多维度传感数据查询时需要发起多次查询，其通信开销高于本文方案。

综上所述，本文方案在计算开销、通信开销以及多维度查询特性支持上实现了较好的平衡。

6　结束语

本文面向雾增强工业物联网设计了一种拥有较高通信效率的带有隐私保护特性的多维度安全查询方案。本文方案使用 BGN 同态加密算法对查询陷门进行加密，并融合了矩阵分解和重构技术，保护了查询模式的隐私，同时确保传感数据不会泄露给非授权方。安全性分析和仿真实验表明，本文方案在保证多种安全特性和多维查询功能的情况下，计算开销及通信开销均维持在较低水平。下一步的工作将进一步优化本文方案，例如进一步提高通信的效率、降低通信开销等。

参考文献：

[1] CHOO K K R, LU R X, CHEN L Q, et al. A foggy research future: advances and future opportunities in fog computing research[J]. Future Generation Computer Systems, 2018, 78(2): 677-679.

[2] LU R X, HEUNG K, LASHKARI A H, et al. A lightweight privacy-preserving data aggregation scheme for fog computing-enhanced IoT[J]. IEEE Access, 2017, 5: 3302-3312.

[3] LU R, LIANG X H, LI X, et al. EPPA: an efficient and privacy-preserving

aggregation scheme for secure smart grid communications[J]. IEEE Transactions on Parallel and Distributed Systems, 2012, 23(9): 1621-1631.

[4] WANG J, YU X, ZHAO M. Privacy-preserving ranked multi-keyword fuzzy search on cloud encrypted data supporting range query[J]. Arabian Journal for Science and Engineering, 2015, 40(8): 2375-2388.

[5] DAI H, YE Q Q, YI X, et al. VP2RQ: efficient verifiable privacy-preserving range query processing in two-tiered wireless sensor networks[J]. International Journal of Distributed Sensor Networks, 2016, 12(11): 1-15.

[6] SHEN Y, HUANG L S, YANG W. Achieving personalized and privacy-preserving range queries over outsourced cloud data[C]//2017 IEEE International Conference on Communications. Piscataway: IEEE Press, 2017: 1-6.

[7] LU R. A new communication-efficient privacy-preserving range query scheme in fog-enhanced IoT[J]. IEEE Internet of Things Journal, 2018, 6(2): 2497-2505.

[8] BONEH D, GOH E J, NISSIM K. Evaluating 2-DNF formulas on ciphertexts[C]//Theory of Cryptography Conference. Berlin: Springer, 2005: 325-341.

[9] PAILLIER P. Public-key cryptosystems based on composite degree residuosity classes[C]// International Conference on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 1999: 223-238.

[10] ISLAM M S, KUZU M, KANTARCIOGLU M. Inference attack against encrypted range queries on outsourced databases[C]//Proceedings of the 4th ACM conference on Data and application security and privacy. New York: ACM Press, 2014: 235-246.

[11] NAVEED M, KAMARA S, WRIGHT C V. Inference attacks on property-preserving encrypted databases[C]//Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2015: 644-655.

[12] KELLARIS G, KOLLIOS G, NISSIM K, et al. Generic attacks on secure outsourced databases[C]//Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2016: 1329-1340.

[13] LACHARITÉ M S, MINAUD B, PATERSON K G. Improved reconstruction attacks on encrypted data using range query leakage[C]// 2018 IEEE Symposium on Security and Privacy. Piscataway: IEEE Press, 2018: 297-314.

[14] DE C A, IOVINO V. jPBC: Java pairing based cryptography[C]//2011 IEEE symposium on computers and communication. Piscataway: IEEE Press, 2011: 850-855.

[15] HUANG B, LIANG S. A range search scheme based on encrypted index hiding order and access patterns[C]//2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData). Piscataway: IEEE Press, 2019: 340-347.

[16] GUO C, ZHUANG R, JIE Y, et al. Secure range search over encrypted uncertain IoT outsourced data[J]. IEEE Internet of Things Journal, 2018, 6(2): 1520-1529.

[17] BABU D S V. Efficient and privacy-preserving range query over outsourced cloud[D]. Fredericton: Fredericton University of New Brunswick, 2018.

[作者简介]

周由胜（1979– ），男，湖北恩施人，博士，重庆邮电大学副教授， 主要研究方向为数据安全、认证与密钥协商。

谭畅（1995– ），男，山东聊城人，重庆邮电大学硕士生，主要研究方向为安全查询、物联网安全等。

唐飞（1986– ），男，重庆垫江人，博士，重庆邮电大学副教授，主要研究方向为公钥密码理论与应用。