支持 *K*-近邻搜索的移动社交网络隐私保护方案

李宇溪[1)] 周福才[2)] 徐紫枫[2)]

[1)](东北大学计算机科学与工程学院 沈阳 110819)

[2)](东北大学软件学院 沈阳 110819)

摘 要 聚焦移动社交网络特征和用户隐私保护的多元需求,本文首次提出一种支持 *K*-近邻搜索的移动社交网络隐私保护方案.方案首先构建融合细粒度访问控制的位置隐私安全模型,在此模型下设计面向移动终端的轻量级位置加密算法,并基于同态加密机制以及安全多方计算思想设计位置密文重加密协议以及 *K*-近邻搜索协议,从而构建安全可信的协同搜索架构,保证服务提供商在无需解密位置的前提下,对用户与好友之间距离进行安全计算并排序,在保护用户位置隐私的同时满足其近邻搜索服务的可用性;除此之外,为了满足细粒度访问控制,方案提出基于公钥广播加密的好友动态管理机制,用户无需为原有好友更新各自的密钥集合的情况下即可赋予或移除好友搜索其位置的权力,实现常数级好友身份认证.安全性方面,方案在随机预言模型下满足自适应$\mathcal{L}$-语义安全性以及撤销安全性.性能方面,与单服务器架构的相关工作相比,本方案降低了用户与服务器之间通信代价的同时,减少了向服务器泄露的位置信息与搜索模式,实现移动社交网络隐私性与可用性的有效平衡.

关键词 移动社交网络;隐私保护;协同架构;密文搜索;安全多方计算;同态加密

中图法分类号 TP309 **DOI 号** 10.11897/SP.J.1016.2021.01481

Privacy-Preserving *K*-Nearest-Neighbor Search over Mobile Social Network

LI Yu-Xi[1)] ZHOU Fu-Cai[2)] XU Zi-Feng[2)]

[1)](*School of Computer Science and Engineering, Northeastern University, Shenyang* 110819)

[2)](*Software College, Northeastern University, Shenyang* 110819)

Abstract Focusing on the features of MSNs (Mobile Social Networks) and the diversified demands of user privacy protection, we propose a privacy-preserving *K*-nearest-neighbor (KNN) search scheme over MSNs. First, we construct a location-privacy security model that incorporates access control, and design a lightweight location encryption algorithm for mobile terminals; we also propose a location re-encryption protocol and an encrypted KNN search protocol based on secure multi-party computation and homomorphic encryption mechanism, which achieves accurate nearby friends retrieving while preventing the geo-location and the distance order from revealing them to the servers. To satisfy the fine-grained access control requirement, we propose a light-weight dynamic friend management mechanism based on public-key broadcast encryption that enables the user to grant or remove the revoked friend's search right of his location without updating the key sets for his original friends, which achieves constant-time friend identity authentication. In terms of security, our scheme satisfies adaptive $\mathcal{L}$-semantic security and revocation security under the random oracle model. In terms of performance, compared with the works with the single server architecture, our scheme not only reduces the communication cost between users and the

收稿日期:2020-04-28;在线发布日期:2021-01-27.本课题得到国家自然科学基金(61772127,61532007,61472184)资助.**李宇溪**,博士,讲师,中国计算机学会(CCF)会员,主要研究方向为可搜索加密、安全多方计算.E-mail:eliyuxi@gmail.com.**周福才**,博士,教授,主要研究领域为网络空间安全、可信计算.**徐紫枫**,博士,讲师,主要研究方向为可验证计算.

server, but also reduces the leakage of the location information and search model, which achieves the trade-off of the location availability and privacy.

Keywords mobile social networks; privacy-preserving; collaboration architecture; encrypted search; secure multi-party computation; homomorphic encryption

1 引 言

随着 5G 和大数据等信息技术的迅猛发展以及智能移动设备的大面积普及,即时通信与搜索所代表的移动在线社交已成人们生活常态.艾媒咨询数据显示①,2019 年中国移动社交用户规模达 8.62 亿,预计到 2020 年底用户规模突破 9 亿.随着移动应用的开发,基于位置的应用使用率达到 96.9%.基于智能手机的定位功能,移动社交服务一般通过手机的 GPS 定位功能来为用户提供基于位置的附近场所推荐和近邻好友搜索服务等,如 Facebook 的"查看附近好友"服务、Foursquare 的"Swarm"应用以及悦跑团的跑友实时竞赛等.在上述服务中,人们通过借助移动设备的 GPS 功能登录到特定位置,服务提供商能够根据获得的位置来查询附近的好友,为现实世界提供一个移动社交界面.

然而,移动社交网络平台在向用户提供基于位置的近邻搜索服务的同时,也造成了严重的用户隐私泄露隐患.例如,通过用户和好友的位置信息,敌手可以以一定的概率推断出用户正在某地参加一个重要的仪式,从而获取用户的职业和社会关系,甚至预测出用户和好友的宗教信仰,以及从搜索历史中推断出个人的行踪等.全国移动 APP 安全性研究报告统计 50 个社交网络服务平台中,有 35 个平台泄露用户的位置数据给广告商或数据分析服务商②.随着数据挖掘技术的快速发展,服务提供商和恶意敌手获取社交网络中人们的隐私信息(如移动用户的位置、个人资料和通信细节等)越来越便利[1],从而更容易推断、跟踪和预测用户的日常活动.近年来许多研究工作也利用特定的大数据分析方法和机器学习相关技术,从社交网络的用户位置信息中获得丰富和大量的用户敏感信息,其中对 Foursquare 上的用户搜索模式(例如,搜索频率、匹配的好友和搜索相似度等)进行分析已被证实可预测用户与好友之间的位置转换模式[2-3];Facebook 的用户历史时空轨迹则泄露了用户与其频繁查询的地理位置之间的距离和访问的概率[4].

造成位置隐私泄露的首要原因是社交网络平台服务器的安全管理不到位导致外部非法用户(黑客等)对社交平台进行的攻击,或者来自社交网络服务提供商内部具有较高权限的管理员的权限滥用或者恶意破坏;另一方面原因是服务提供商默认在无需获得用户的特殊许可的情况下,可以获取用户的当前位置以进行跟踪①.为了避免服务提供商和黑客未经授权访问位置信息和搜索模式,用户可以在本地对位置信息进行加密再上传至社交网络平台.然而,传统加密方法限制了服务器为用户提供基于位置的近邻搜索服务,用户需要将所有好友的位置密文下载到本地,再与好友交互协助其进行解密,从而搜索距离较近的好友信息,这种操作不仅带来了庞大的通信代价和本地存储代价,而且用户也需要为解密和搜索操作付出巨大的计算代价.理论上私有数据检索[5]和可搜索加密[6]等密码学原语在一定程度上能够实现在位置密文数据中进行搜索,但是往往需要对数多项式的计算和通信代价,实现起来效率不理想.而且,在社交网络隐私保护系统中,用户各自拥有不同的密钥,若允许用户与好友共享位置加密密钥,当用户想要搜索其好友的位置信息时,需要向平台发起多次搜索请求,在增删好友时也需要用户同所有好友同步更新本地存储的密钥,使得方法的灵活性较差,不适用于现实中具有一定可扩展需求的移动社交网络平台.因此,如何保证社交网络用户位置数据以及搜索隐私等敏感信息不发生外泄的情况下,提供高效且精确的基于位置的近邻搜索服务,并实现细粒度的访问控制,成为当今移动社交网络隐私保护技术研究领域中具有挑战的研究课题之一.

基于对上述问题,本文提出支持 K-近邻搜索的移动社交网络隐私保护方案,在支持高效近邻好友排序搜索的前提下,在不可信环境中对用户的搜索模式和位置信息进行有效的保护,实现移动社交

① 艾媒报告|2019－2020 年中国移动社交行业年度研究报告. https://www.iimedia.cn/c400/70165, 2020,3,20

② 全国移动 APP 安全性研究报告:约 70%的 APP 都存在安全漏洞. https://www.freebuf.com/articles/paper/202843, 2019,5,7

网络隐私性与可用性的有效平衡. 本文主要贡献如下：

(1) 针对移动设备计算能力受限以及位置更新需求，设计基于伪随机函数的轻量级位置加密算法，降低用户的计算代价和通信代价；除此之外，为了满足用户的 K-近邻搜索需求，方案设计位置重加密协议，在无需解密的前提下，将好友的对称位置密文转化为搜索用户的 Paillier 密文，并将其作为计算、排序和返回搜索结果的依据，从而避免搜索时需要好友在线参与计算的局限性，达到较高的执行效率.

(2) 聚焦近邻搜索可用性需求以及用户位置隐私泄露问题，方案构建安全可信的双服务器协同搜索架构，并融合安全多方计算思想和同态加密机制设计面向位置密文的 K-近邻搜索协议，保证服务器在密文层面下实现对用户与好友之间距离的有效计算与排序，在保护用户位置隐私的同时满足用户的近邻搜索需求.

(3) 为了满足细粒度访问控制，方案设计基于公钥广播加密的好友动态管理机制，用户在无需为原有好友更新各自的密钥集合的情况下即可赋予/移除好友的搜索权力，从而实现轻量化好友身份认证及权限管理；同时，此机制在安全性方面满足敌手无法通过与已撤销的好友共谋来获取用户的位置信息，从而进一步提高方案的整体安全性.

(4) 在诚实并好奇的威胁场景下构建方案的安全模型，并对方案的正确性、自适应 $\mathcal{L}$-语义安全性以及撤销安全性进行严格数学分析与证明；从存储、计算以及通信三个维度对方案进行了代价分析，并与其他相关方案进行了特性对比；对方案进行实验分析来验证 K-近邻搜索的性能表现，实验结论证明方案具有较强的可扩展性和稳定性.

2　相关工作

2.1　基于位置的社交网络隐私保护技术

总体来讲，保证位置安全的重要手段包括对位置信息进行访问控制、隐藏和加密三类：早期的社交网络隐私保护技术大多通过强制系统执行访问控制策略来实现[7-8]，访问控制能够防止陌生人和恶意敌手对用户的位置信息进行非授权访问，但是这种方法主要依赖于行政监管，很难有效地保护用户数据以及搜索的隐私. 位置隐藏方法侧重于在系统中通过匿名技术[9-10]和混淆区域技术[11-12]向服务器隐藏用户的真实位置信息. 此类技术一般对用户身份和位置信息进行匿名化或模糊处理，使得服务器无法执行基于位置的准确查询. 随着安全理论技术的发展，许多研究学者致力于从不同角度研究针对移动社交网络位置泄露问题的隐私保护技术. 例如，文献[13]提出了基于布隆过滤器的用户位置相似性匹配方案，方案将两个用户之间的位置属性交集显式地传输到同一个布隆过滤器中来匹配双方的距离远近. 然而，布隆过滤器使得方案的匹配结果具有一定的误报率. 文献[14]和[15]基于差分隐私技术来实现隐私保护位置搜索. 差分隐私通过添加噪声使位置数据和搜索请求具有不可区分性，从而抵抗恶意敌手的背景知识攻击，提供了严格的隐私保障机制，但是噪声的注入造成搜索准确性的损失. 文献[16]和[17]结合机器学习聚类方法，分别根据社交网络的自然社区结构以及位置信息对用户进行聚类并分组，在保证高效位置搜索的同时显著降低了噪声值，但是无法提供精确距离远近排序. 然而，文献[14-17]中的方案的安全性大多依赖于可信第三方来维护用户的真实位置信息，但是如果可信第三方被恶意敌手攻击，则敌手可以访问所有的用户位置信息，这将会给用户带来更加严重的隐私威胁.

除此之外，依赖可证安全的密码学加密技术，一些提供较强安全模型的位置隐私保护方法被相继提出. 文献[18]利用私有信息检索技术(Private Information Retrieval，PIR)实现了路网近邻搜索. 文献[19]基于私有集合交集协议来设计具有隐私保护性质的用户属性匹配方案，方案允许两个用户(各自拥有私有属性集合)私下计算属性的密文集合交集来匹配目标用户相似度，但匹配过程不会泄露双方的隐私信息. 文献[20]提出了移动社交网络中基于内积相似性的隐私信息匹配协议，用户将隐私信息映射为向量并对向量进行加密，利用相似度函数度量不同用户密文向量相似程度. 文献[21]设计动态位置网格索引结构加密方法，在不向第三方泄露位置隐私的前提下，实现近邻兴趣点搜索. 上述基于加密的方案中由于通过服务器对用户隐私数据重加密或其他复杂运算而导致计算效率不理想，甚至需要对数级多轮交互的通信代价. 除此之外，安全多方计算(Multi-Party Computation，MPC)[22]的设计理念能给移动社交网络中的距离计算提供较高的安全保证. 然而，在已有的基于 MPC 的社交网络隐私保护方案中[23-25]，用户一般需要在线参与计算过程，这意味着用户发起搜索请求时需要与好友进行交互并要在好友客户端完成相应的安全计算过程，然而实

际主流社交网络部署架构中服务器为负责计算的实体,用户仅为信息上传或接收实体,因此这种分布式架构不满足实际部署需求.

基于上述研究现状可以看出,现有的基于位置的社交网络隐私保护方法中,大多基于较高的系统模型假设(例如,引入可信第三方)或较低的安全目标(例如,允许平台获得位置信息和用户搜索模式),而且大多包含复杂的通信与计算代价,在实际应用中具有局限性.因此,保证移动社交网络位置搜索服务中用户位置数据以及搜索隐私等敏感信息不发生外泄,并且实现高效精确的近邻搜索,是社交网络隐私保护方法的一个具有挑战性的问题.

2.2 协同架构隐私保护技术

协同架构隐私保护技术是近年来在密码学理论研究界和安全产业界上比较活跃的一个研究课题,目的是通过构建安全可信的协同工作机制,实现系统在性能和安全性之间的深度融合.在理论研究中,基于协同架构的安全协议一般应用于面向高安全需求的隐私保护应用场景中,例如可搜索加密、安全多方计算和隐私保护机器学习等.文献[26]提出了一种双服务器协同可搜索加密方案,在该方案中,两台服务器分别负责存储加密数据或用户密钥,实现面向多密钥加密的密文数据搜索.文献[27]利用 Paillier 同态加密方案在协同架构下设计支持搜索结果排序的可搜索加密方法,协同架构能够防止任意服务器学习到实际的用户访问模式.文献[28]提出了一种基于保序加密方法[29]的双服务器排序密文搜索方案,方案利用基于 TF-IDF 加权规则计算密文权值,使用保序加密对权值加密,服务器将搜索到的权值密文排序并返回给用户,用户在本地解密得到排序后的匹配结果.文献[30]在协同隐私保护环境中基于安全多方计算以及 KNN 聚类思想对加密数据库进行安全分类.

除此之外,在现实云环境中,很多外包数据存储与计算服务通过部署协同架构(例如,Cryptomator①、Boxcryptor6②或 pCloud Crypto5③)来确保用户敏感外包数据的安全性.此类协同架构一般由云服务提供商和云加密平台组成,前者通常由公有云或者大数据存储平台提供,后者通常配备了存储密钥的安全处理器,公司间由于商业利益等原因不会互相共谋.例如,Cryptomator 为用户的外包至大数据存储平台的文件提供透明的客户端加密机制,从而保护用户的密文文件免受未经授权的访问.Boxcryptor 与 Dropbox、谷歌 Drive 和坚果云等云存储服务平台能无缝结合为个人、团队以及企业用户提供各种加密解决方案.

3 预备知识

3.1 伪随机函数

设函数 $F:\{0,1\}^k\times\{0,1\}^*\to\{0,1\}^k$ 是伪随机函数,则给定密钥 $K\in\{0,1\}^k$ 以及输入 $x\in\{0,1\}^n$,函数 $F_k(x)$ 可以被有效计算出.给定 $((x_1,F_k(x_1)),\cdots,(x_m,F_k(x_m)))$,对于任意 x_{m+1},对于概率多项式时间算法 A,对于所有参数 k 和函数 F,存在一个可忽略函数 $negl$ 满足:

$$|\Pr[A^{F_k(\cdot)}=0|k\leftarrow\{0,1\}^k]-|\Pr[A^g=0|g\leftarrow F:\{0,1\}^k\times\{0,1\}^*\to\{0,1\}^k]|\leq negl(1^k).$$

3.2 公钥广播加密

本文利用满足 CPA(Chosen-Plaintext Attack)安全的公钥广播加密方案 BE[31]作为用户位置访问控制的构建模块.设 $\mathcal{U}$ 表示系统用户集合,$\mathcal{G}\subseteq\mathcal{U}$ 表示一组授权用户,$\mathcal{K}$ 表示密钥空间,$\mathcal{C}$ 表示密文空间,$\mathcal{M}$ 表示明文空间.BE 方案包含以下算法:

(1) $(msk,pk)\leftarrow\text{BE.KeyGen}(1^k)$.密钥生成算法,用于生成系统密钥.算法以安全参数 1^k 为输入,输出主密钥 msk 和初始公钥 pk;

(2) $sk_u\leftarrow\text{BE.Join}_{msk}(u)$.加入算法,用于为用户生成私钥.算法输入主密钥 msk 和用户 u,输出该用户的会话密钥 $sk_u\in\mathcal{K}$;

(3) $c\leftarrow\text{BE.Enc}_{pk}(\mathcal{G},m)$.加密算法,算法输入公钥 pk,接收用户集合 $\mathcal{G}\subseteq\mathcal{U}$,以及明文 $m\in\mathcal{M}$,输出密文 $c\in\mathcal{C}$;

(4) $(m/\perp)\leftarrow\text{BE.Dec}_{sk_u}(c)$.解密算法,用于对接收到的广播密文进行解密.算法输入会话密钥 sk_u、密文 $c\in\mathcal{C}$.如果是有效用户,则算法输出明文 m,否则输出 $\perp$.

3.3 同态加密机制

同态加密是一种无需解密即可在密文数据上进行计算的加密方法,与在原始数据上计算能够获得相同的结果,本节阐述应用于协议构建的 Paillier 同态加密方案[32]以及 GM 同态加密方案[33].

① Cryptomator — Free client-side encryption for your cloud files. Open source software; No backdoors, no registration [EB/OL]. https://cryptomator.org/

② Boxcryptor — Security for your Cloud. https://www.boxcryptor.com/en/

③ Dropbox challenger pCloud just became profitable. https://techcrunch.com/2019/04/16/dropbox-challenger-pcloud-just-became-profitable/

3.3.1　Paillier 同态加密

设$[m]$表示 Paillier 同态加密方案中 m 的密文. $\mathrm{lcm}(a,b)$指整数 a 和整数 b 的最小公倍数. Paillier 同态加密方案由三个多项式时间算法组成：

(1) $(pk_i,sk_i)\leftarrow \mathrm{P.KeyGen}(1^k)$. 输入安全参数 1^k，选择两个随机 k 比特素数 P 和 Q，并置 $N=PQ$. 选择 g，满足 $\gcd(L(g^{\lambda(N)} \bmod N^2),N)=1$，其中函数 $L(u)=(u-1)/N$，$\lambda(N)=\mathrm{lcm}(P-1,Q-1)$，选择一个随机数 $a\in Z_{N^2}^*$，令 $h=a^N \bmod N^2$，可知 h 在模 N^2 下的阶为 $\lambda(N)$. 算法输出公钥 $pk_i=(N,g,h)$，私钥 $sk_i=(P,Q)$.

(2) $[m]\leftarrow \mathrm{P.Enc}_{pk_P}(m)$. 输入消息 $m\in Z_N$，选择随机数 $r\in Z_N$，输出密文 $[m]=g^m h^r \bmod N^2$.

(3) $m\leftarrow \mathrm{P.Dec}_{sk_P}([m])$. 输入密文 $[m]\in Z_{N^2}^*$，输出明文 $m=(L([m]^{\lambda(N)} \bmod N^2)/L(g^{\lambda(N)} \bmod N^2)) \bmod N$.

Paillier 同态加密的同态加性质如下：

$\forall m_1,m_2\in Z_N,[m_1]\times[m_2]=[m_1+m_2]$.

除此之外，本方案利用 Lipmaa 等人在 2005 年提出 Paillier 双重加密的方法[34]，设$[\![m]\!]$为双重加密之后的密文，其中明文 m_1 满足 $m_1\in Z_{N^2}$，则 Paillier 双重加密方法满足如下性质：

$$[\![m_1]\!]^{m_2}=[\![[m_1][m_2]]\!]=[\![m_1+m_2]\!].$$

3.3.2　Goldwasser-Micali 同态加密

设 Goldwasser-Micali(GM)同态加密的明文空间 M 为$\{0,1\}$，$\mathcal{J}_p(x)$为 x 对 p 的 Jacobi 符号，$\|m\|$为 $m\in\{0,1\}$加密之后的 GM 密文. GM 同态加密方案由三个多项式时间算法组成：

(1) $(pk_{\mathrm{GM}},sk_{\mathrm{GM}})\leftarrow \mathrm{GM.KeyGen}(1^k)$. 输入安全参数 1^k，选择两个随机 k 比特素数 P 和 Q，并置 $N=PQ$；选取模 n 的一个二次非剩余 $\delta\in Z_n^*$ 满足雅可比符号$(\delta/n)=1$. 输出公钥 $pk_{\mathrm{GM}}=(\delta,n)$，私钥是 $sk_{\mathrm{GM}}=(P,Q)$.

(2) $\|m\|\leftarrow \mathrm{GM.Enc}_{pk_{\mathrm{GM}}}(m)$. 输入消息$m\in(0,1)$，选择随机数 $r\in Z_N$，输出密文 $\|m\|=r^2\delta^m \bmod n$.

(3) $m\leftarrow \mathrm{GM.Dec}_{sk_{\mathrm{GM}}}(\|m\|)$. 输入密文 $\|m\|\in Z_{N^2}^*$，计算 $\mathcal{J}_p(m)$和 $\mathcal{J}_q(m)$，如果 $\mathcal{J}_p(m)=J_q(m)=+1$，输出 m 为 0，否则 m 为 1.

GM 同态加密的同态加性质可表示为

$$|m_1||m_2|=|m_1\oplus m_2|.$$

4　方案描述

4.1　方案架构

图 1 为方案的整体架构，其中主要包含以下三类实体：

(1) $\mathcal{U}$：用户集合，包含 n 个用户$(u_1,\cdots,u_n)$. 每个用户 $u_i\in\mathcal{U}$ 能够添加系统中其他用户成为好友，并发起 K-近邻搜索请求.

(2) $\mathcal{S}_1$：主服务器，作为服务提供商，存储社交网络图结构 $\mathcal{G}=(\mathcal{V},\mathcal{E})$并响应用户搜索请求.

(3) $\mathcal{S}_2$：协同服务器，协助 $\mathcal{S}_1$ 根据用户搜索请求完成搜索结果的排序.

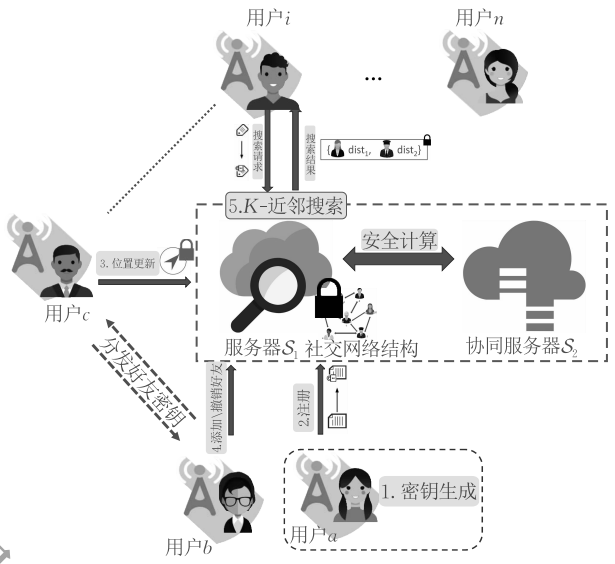


图 1　方案架构

如图 1 所示，本方案以现实中基于位置的集中式社交服务架构为设计依据进行架构设计，并部署可信定位基础设施来确定移动用户的物理位置. 服务器 $\mathcal{S}_1$ 负责初始化社交网络系统，构建社交网络图结构 $\mathcal{G}=(\mathcal{V},\mathcal{E})$用来存储用户信息. 系统内的任意用户 $u_i\in\mathcal{U}$ 都在本地生成密钥信息，并和服务器 $\mathcal{S}_1$ 交互来注册系统. 用户利用本地设备接收定位基础设施发送的位置信息，生成当前位置信息密文，并与服务器 $\mathcal{S}_1$ 交互执行位置更新协议来更新位置密文. 当用户 u_i 和 u_j 有意互相添加对方为其好友时，需要同服务器 $\mathcal{S}_1$ 交互来执行添加好友协议来赋予彼此搜索位置信息的权限；任意用户可以向服务器 $\mathcal{S}_1$ 发起 K-近邻搜索请求，服务器 $\mathcal{S}_1$ 在服务器 $\mathcal{S}_2$ 的协助下搜索距离用户最近的 K 个好友并返回给用户. 如果用户想要解除与某用户之间的好友关系，同样需要同服务器 $\mathcal{S}_1$ 交互来执行撤销好友协议来解除此用户的搜索权限.

4.2　设计思路

本节给出方案的设计思路. 方案的设计聚焦于针对移动终端的处理能力以及通信负载等限制因素，旨在将距离搜索和基于同态加密算法的安全协议进行深度融合的同时，针对移动社交网络场景对

位置加密、距离计算以及细粒度访问控制方面进行进一步优化与创新.由于针对不同地理位置以及时空场景,距离量算方法有所不同,因此针对细粒度场景区分设计出更精确或更高效的距离量算方法这一主题超出了本方案的侧重讨论范围.

以下为 K-近邻搜索的形式化定义:

定义 1. K-近邻搜索.设 $\mathcal{U}=(u_1,\cdots,u_n)$ 为社交网络中的用户集合,F_i 是用户 $u_i\in\mathcal{U}$ 的好友集合.$Q=(u,l,K)$ 为 K-近邻搜索请求,其中 $u\in\mathcal{U}$ 指发起搜索请求的用户,l 表示用户当前地理位置.D_i 表示 u 和 u_i 之间的平方欧几里得距离:$D_i=(l-l_i)^2$.K-近邻搜索结果 R_K 可表示为一个有序数组 $R_K=\{(u_1,D_1),\cdots,(u_K,D_K)\}$ 满足:

(1) R_K 中所有用户均为搜索用户的好友,即不存在用户 u_i,满足 $u_i\in R_K$ 且 $u_i\notin F_i$;

(2) R_K 包含距离最近的 K 个好友,即不存在用户 $u_i\in\{F_i\backslash R_K\}$ 满足 $D_i<D_j$,其中 D_j 为 R_K 中所有距离值中的最大值;

(3) R_K 中的距离值集合 $\{D_1,\cdots,D_K\}$ 满足有序性,即 $D_1\leq\cdots\leq D_K$.

① 位置加密.由于用户的地理位置 l 属于用户的隐私信息,为了向服务器隐藏真实的地理位置,用户需要在本地对位置 l 进行加密再上传至社交网络平台.由于距离搜索服务需要服务器能够有能力对位置密文进行计算并排序,因此方案首先假设用户可利用 Paillier 同态加密方法将位置信息在本地进行加密,但是考虑位置的频繁更新性以及移动终端计算能力限制,大量同态加密运算以及 Paillier 同态密文的较大膨胀率对于本地计算和处理能力提出较高要求.因此,本文针对上述问题,设计一种基于伪随机函数的轻量级位置加密算法,用户 u 在本地利用私钥 k 将位置信息 l 进行对称加密生成轻量级位置密文 L,并将 L 发送给服务器.然而,由于服务器端存储的位置密文是利用不同用户的私钥加密的密文,服务器无法根据位置密文判定用户与好友之间的距离.因此,为了满足用户的 K-近邻搜索需求,设计位置重加密协议,将用户好友 u_i 位置密文 L_i 利用搜索用户的 Paillier 公钥 pk 进行重新加密,从而转化为搜索用户的 Paillier 密文 L^i,并将其作为计算、排序和返回搜索结果的依据;从密文大小方面以及计算代价方面考虑,与朴素的方法相比,上述方法降低了用户端计算代价的同时,也解决了密文过大也使用户从客户端上传加密位置到服务器端会面临传输带宽瓶颈,从而使得服务器无需对距离密文进行解密即可实现 K-近邻搜索.

② 距离计算.由于欧几里得距离是较常用的距离量算方式,因此本文以平方欧几里得距离为实例,来阐述距离计算的设计思路.设用户 u 和好友 u_i 位置分别为 l 和 l_i,则双方的平方欧几里得距离 D_i 可表示为 $D_i=(l_i-l)^2$.根据 Paillier 加密算法的同态加性质,D_i 的密文可以按照如下方式计算得出:$[D_i]=[(l_i-l)^2]=[l_i^2][-2l_il][l^2]$.为了计算距离 $[D_i]$,服务器 $\mathcal{S}_1$ 和协同服务器 $\mathcal{S}_2$ 首先执行位置密文重加密协议得到位置密文 $[l_i]\in L^i$,通过计算 $[-2l_il]$ 并进行密文乘法运算 $[l_i^2][-2l_il][l^2]$ 得到用户与好友之间距离的 Paillier 密文 $[D_i]$.然而,Paillier 密文的随机性导致距离密文并不会表达好友之间的远近顺序信息,所以 $\mathcal{S}_1$ 并不能在上述过程得到排序之后的 K-近邻搜索结果.为了实现距离排序,$\mathcal{S}_1$ 需要同 $\mathcal{S}_2$ 执行面向 Paillier 距离密文的安全排序协议得到 K-近邻排序结果.协议中通过比较距离密文计算结果中每对 Paillier 距离密文的大小,并将每次比较过程作为黑盒,融合具有连贯层级的并行排序算法——Batcher 排序网络[35] 实现排序.K-近邻搜索协议执行完毕后,$\mathcal{S}_1$ 返回给用户根据距离排序并重加密的 K-近邻搜索结果 R_K,用户可以利用 Paillier 解密算法解密得到按照距离远近排序的 K-近邻好友信息.

③ 动态管理.针对用户位置的细粒度访问控制需求,方案构建好友动态管理机制,基于动态公钥广播加密方法设计好友添加与撤销协议实现针对用户好友的灵活位置访问权限增删.具体来讲,用户 u 在本地构建好友列表作为广播加密方法的授权用户组 F,生成自己的当前状态值 st,并将状态值 st 利用广播加密算法进行广播加密,将生成的状态密文 cst 更新至平台.在好友添加阶段,用户 u 首先为好友 u_i 分配其会话密钥 k_i,之后根据包含 u_i 的好友列表 F' 生成新的广播加密状态值 cst',并将 cst' 更新至社交网络平台,之后好友列表中的所有好友均有权解密用户的状态密文值来生成正确的搜索令牌,进而执行位置搜索请求;在好友撤销阶段,用户在本地将待删好友 u_j 从好友列表中删除,之后利用广播加密方法,根据当前的好友列表 F'' 更新状态密文 cst'',并将新的状态密文 cst'' 更新至社交网络平台.在效率方面,上述基于动态公钥广播加密方法设计的好友动态管理机制使方案在好友添加和撤销过程中无需为原有好友列表中的用户更新各自的密钥集合,从而有效降低了用户的计算代价和用户与好友间的通信

代价.在安全性方面,此机制满足撤销后的好友则无法成功解密用户信息状态密文,从而没有权限执行位置搜索请求;即使其密钥被敌手非法窃取,敌手仍不能获取用户的位置信息,从而进一步提高方案的整体安全性.

4.3　方案形式化描述

本节对提出的方案进行形式化描述,方案由7个多项式时间算法与协议组成,具体如下:

(1) $\mathcal{G}\leftarrow$Setup(1^k). 初始化算法,由 $\mathcal{S}_1$ 执行,输入安全参数 1^k,输出非定向图结构 $\mathcal{G}$.

(2) $(K_i,PK_i,SK_i)\leftarrow$KeyGen($1^k$). 密钥生成算法,由用户 u_i 执行,算法输入输入安全参数 1^k,输出用户密钥 K_i 以及公私钥 (PK_i,SK_i).

(3) $(F_i;\mathcal{G}')\leftarrow$Register($u_i(PK_i);\mathcal{S}_1(\mathcal{G})$). 注册协议,由用户 u_i 和服务器 $\mathcal{S}_1$ 之间交互执行,用于执行用户注册.协议中用户 u_i 输入公钥 PK_i,$\mathcal{S}_1$ 输入 $\mathcal{G}$.协议执行结束后,$\mathcal{S}_1$ 输出更新的图结构 $\mathcal{G}'$,用户输出好友集合 F_i.

(4) $(\perp;\mathcal{G}')\leftarrow$LocUpdate($u_i(K_i,l_i);\mathcal{S}_1(\mathcal{G})$). 位置更新协议,由用户 u_i 和服务器 $\mathcal{S}_1$ 之间交互执行,用于在系统中更新用户当前位置密文.协议中用户 u_i 输入密钥 K_i 和当前位置 l_i,$\mathcal{S}_1$ 输入 $\mathcal{G}$.协议执行结束后,$\mathcal{S}_1$ 输出更新的图结构 $\mathcal{G}'$.

(5) $((F_i',K_i');\mathcal{G}')\leftarrow$AddF($u_i(u_j,K_i,PK_i,F_i);\mathcal{S}_1(\mathcal{G})$). 添加好友协议,由用户 u_i 和服务器 $\mathcal{S}_1$ 之间交互执行,用于在系统中添加好友信息.协议中用户 u_i 输入密钥 (K_i,PK_i),$\mathcal{S}_1$ 输入 $\mathcal{G}$.协议执行结束后,$\mathcal{S}_1$ 输出更新的图结构 $\mathcal{G}'$,用户 u_i 输出更新的好友集合和密钥 (F_i',K_i').

(6) $(R_K;\perp;\perp)\leftarrow$Search($u_s(K_s,F_s);\mathcal{S}_1(\mathcal{G});\mathcal{S}_2(K_s)$). K-近邻搜索协议,由用户 u_s、服务器 $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 执行,用于执行近邻搜索.协议中用户 u_s 输入 K_s,$\mathcal{S}_1$ 输入 $\mathcal{G}$,$\mathcal{S}_2$ 输入 SK_s,协议执行结束后,用户 u_s 输出搜索结果 R_K.

(7) $(F_i';\mathcal{G}')\leftarrow$RevokeF($u_i(u_j,PK_i,F_i);\mathcal{S}_1(\mathcal{G})$). 撤销好友协议,由用户 u_i 和服务器 $\mathcal{S}_1$ 之间交互执行,用于在系统中删除好友.算法的输入为待删除的好友 u_j、密钥 PK_i 以及好友列表 F_i,$\mathcal{S}_1$ 输入 $\mathcal{G}$;协议执行结束后,用户 u_i 输出更新后的好友列表 F_i',$\mathcal{S}_1$ 输出更新的图结构 $\mathcal{G}'$.

定义 2.　正确性.方案的正确性意味着,对于所有的安全参数 1^k、任意由 Setup(1^k) 初始化的图结构 $\mathcal{G}$,以及所有用户通过算法 KeyGen(1^k) 生成的密钥 (K_i,PK_i,SK_i),以及任意顺序的位置更新操作 LocUpdate、添加好友 AddF、撤销好友 RevoleF 操作,K-近邻搜索协议 Search($u_s(K_s,F_s);\mathcal{S}_1(\mathcal{G});\mathcal{S}_2(SK_s)$)总会得到正确的结果 R_K,即满足:

(1) R_K 为有序集合,满足 $D_1<\cdots<D_K$;

(2) 不存在 $u_i\in R_K$,满足 $u_i\notin F_s$;

(3) 不存在 $u_i\in\{F_s\backslash R_K\}$ 满足 $D_i<D_j$,其中 D_j 为 R_K 中所有距离值中的最大值.

4.4　安全性定义

4.4.1　自适应 $\mathcal{L}$-语义安全性

本方案的安全模型中包括两个满足“诚实并好奇”模型的半可信实体:主服务器 $\mathcal{S}_1$ 以及协同服务器 $\mathcal{S}_2$.主服务器 $\mathcal{S}_1$ 会诚实地为用户提供 K-近邻搜索服务,不会主动更改或损坏用户的数据,但可能试图查看、分析或使用用户的位置数据和搜索令牌信息.协同服务器 $\mathcal{S}_2$ 配备了用于保存密钥的密码处理器,仅仅进行排序的辅助工作,不会获得任何有用数据,并且两个实体不会共谋.针对上述模型,构造博弈实验 Real(1^k) 与 Ideal(1^k) 来刻画自适应 $\mathcal{L}$-语义安全性.具体来说,设 $\mathcal{A}_1$ 和 $\mathcal{A}_2$ 为两个自适应敌手,分别代表方案中的服务器 $\mathcal{S}_1$ 与协同服务器 $\mathcal{S}_2$.Sim_1 和 Sim_2 为两个具有概率多项式时间计算能力的模拟器,$\mathcal{L}_1$、$\mathcal{L}_2$、$\mathcal{L}_3$ 和 $\mathcal{L}_4$ 为方案向 $\mathcal{A}_1$ 和 $\mathcal{A}_2$ 泄露的信息构成的泄漏函数.根据敌手和模拟器的博弈过程,实验具体构造如下:

(1) Real(1^k). 定义 Real(1^k) 为敌手 $\mathcal{A}_1$、$\mathcal{A}_2$ 和用户 $\mathcal{U}$ 之间的博弈实验,使用真实的方案算法与协议.$\mathcal{A}_1$ 初始化一个空的图结构 $\mathcal{G}$,每个用户 $u_i\in\mathcal{U}$ 利用 KeyGen 算法计算密钥 (K_i,PK_i,SK_i),并与 $\mathcal{A}_1$ 交互,执行任意顺序的位置更新协议 LocUpdate、添加好友协议 AddF、撤销好友协议 RevoleF.$\mathcal{A}_1$ 根据上述协议收到的信息更新图结构 $\mathcal{G}$.随后,$\mathcal{U}$ 中的任意用户向服务器发起至多 t 次的质询 $(q_1,\cdots,q_t)$.针对每次质询 q_i,$\mathcal{A}_1$ 在 $\mathcal{A}_2$ 协助下生成搜索结果 R_i,并将搜索结果返回给质询用户 u_i.在所有 t 次质询结束后,将 $\mathcal{A}_1$ 得到的信息记作 $view_{\mathcal{A}_1}$.$\mathcal{A}_2$ 仅参与与 $\mathcal{A}_1$ 的交互环节,将其从中得到的信息记作 $view_{\mathcal{A}_2}$.

实验结束后,$\mathcal{A}_1$ 输出 $\mathrm{Output}_{\mathcal{A}_1}^{\mathrm{Real}(1^k)}$,$\mathcal{A}_2$ 输出 $\mathrm{Output}_{\mathcal{A}_2}^{\mathrm{Real}(1^k)}$.

(2) Ideal(1^k). 定义 Ideal(1^k) 为模拟器 Sim_1、Sim_2 和用户 $\mathcal{U}$ 之间的博弈实验.该实验与 Real(1^k) 实验的区别是模拟器不运行真实方案中的算法与协议,而是利用泄漏函数 $\mathcal{L}_1$ 到 $\mathcal{L}_4$ 来作为唯一输入,来生成随机化的数据来回应用户搜索请求.根据泄露

函数$\mathcal{L}_1$和$\mathcal{L}_2$，Sim_1模拟图结构$\widetilde{\mathcal{G}}$. 用户向Sim_1发起至多t次的质询$(q_1,\cdots,q_t)$，其中t为多项式级. 针对每次质询q_i，Sim_1和Sim_2利用$\mathcal{L}_3$和$\mathcal{L}_4$模拟Search$(\mathcal{S}im_1^{\mathcal{L}_3,\mathcal{L}_4}(1^k),\mathcal{S}im_2^{\mathcal{L}_4}(1^k))$并且得到模拟结果$\widetilde{R}_i$. t次质询结束后，将Sim_1和Sim_2得到的信息记作$view_{Sim_1}$和$view_{Sim_2}$.

实验结束后，Sim_1输出$\text{Output}_{Sim_1}^{\text{Ideal}(1^k)}$，$Sim_2$输出$\text{Output}_{Sim_2}^{\text{Ideal}(1^k)}$.

定义 3. 自适应$\mathcal{L}$-语义安全性. 设1^k为安全参数，$\mathcal{A}_1$和$\mathcal{A}_2$为两个概率多项式时间敌手，本方案满足自适应$\mathcal{L}$-语义安全性，当且仅当Real(1^k)与Ideal(1^k)两个实验在多项式时间内具有不可区分性，即对于$\mathcal{A}_1$和$\mathcal{A}_2$，存在概率多项式时间的模拟器Sim_1和Sim_2，满足$\text{Output}_{\mathcal{A}_{1/2}}^{\text{Real}(1^k)}$与$\text{Output}_{Sim_{1/2}}^{\text{Ideal}(1^k)}$具有不可区分性：

$$\text{Output}_{\mathcal{A}_{1/2}}^{\text{Real}(1^k)}\approx\text{Output}_{Sim_{1/2}}^{\text{Ideal}(1^k)}.$$

4.4.2 撤销安全性

撤销安全性保证方案满足已撤销的好友无法提供有效搜索令牌，即使其密钥被敌手非法窃取，敌手仍不能获取用户的位置信息. 本节构造博弈实验$\text{Exp}_{\mathcal{A}_{rev}}^{\text{Revoke}}(1^k)$来刻画撤销安全性. $\text{Exp}_{\mathcal{A}_{rev}}^{\text{Revoke}}(1^k)$由挑战者$\mathcal{C}$和敌手$\mathcal{A}_{rev}$交互执行，敌手$\mathcal{A}_{rev}$拥有真实方案中添加好友、执行搜索和撤销好友的能力，经过多轮博弈后，挑战者$\mathcal{C}$撤销敌手$\mathcal{A}_{rev}$添加进好友列表的用户；敌手$\mathcal{A}_{rev}$使用被撤销的好友身份继续生成搜索令牌并向服务器发起搜索请求，服务器给出结果. 经过多轮质询与应答过程，证明任意概率多项式时间计算能力的敌手赢得自适应撤销博弈实验的概率可忽略，从而证明本方案满足用户撤销安全. $\text{Exp}_{\mathcal{A}_{rev}}^{\text{Revoke}}(1^k)$的形式化描述如过程 1 所示.

过程 1. $\text{Exp}_{\mathcal{A}_{rev}}^{\text{Revoke}}(1^k)$实验.

$\text{Exp}_{\mathcal{A}}^{\text{Revoke}}(1^k)$

$(K_i,PK_i,SK_i)\leftarrow_{\$}\text{KeyGen}(1^k)$

$st\leftarrow\mathcal{A}(1^k,PP)$

$\mathcal{G}\leftarrow_{\$}\text{Setup}(1^k,K_{u_i},PP)$

$st\leftarrow_{\$}\mathcal{A}^{\mathcal{O}}(st)$

FOR $u\in F_i$

$(K_{u_i},PP)\leftarrow_{\$}\text{RevokeF}(\cdot,\mathcal{G},u_j,PK_i,F_i)$

$\tau\leftarrow_{\$}\mathcal{A}^{\mathcal{O}}(st)$

$R_K\leftarrow_{\$}\text{Search}(\tau,k_{s_1})$

IF $R_K\neq\perp$ RETURN 1

ELSE RETURN 0

具体来讲，挑战者$\mathcal{C}$运行 Setup 创建$\mathcal{G}$初始化系统，利用 KeyGen 和 Register 生成用户密钥(K_i,PK_i,SK_i)和状态密文cst_i等. 挑战者$\mathcal{C}$发送$\mathcal{G}$、cst_i给敌手$\mathcal{A}_{rev}$. 随后敌手可以访问如下预言机，其中由“·”所表示的参数由敌手提供：

$\mathcal{O}_{\text{AddF}}(\cdot,\mathcal{G},u_j,PK_i,F_i)$：此预言机允许敌手发起添加好友的请求. 预言机通过敌手提供的输入运行 AddF，如果u_j未被添加为u_i好友，则向敌手返回 AddF(u_j,K_u,PP)的输出，如果用户u_j被添加为u_i好友，则返回$\perp$.

$\mathcal{O}_{\text{RevokeF}}(\cdot,\mathcal{G},u_j,PK_i,F_i)$：此预言机允许敌手发起撤销好友$u_j$的请求. 如果$u_j$为$u_i$好友，预言机通过敌手提供的输入运行 RevokeF. 如果用户u_j不是u_i好友，预言机返回$\perp$.

$\mathcal{O}_{\text{Search}}(\cdot,\mathcal{G},PK_s,F_s)$：此预言机允许敌手在$\mathcal{G}$中发起搜索请求. 敌手通过 Search 计算搜索令牌并将其发送给预言机作为输入，预言机仅仅通过敌手提供的输入运行 Search，并将结果输出给敌手.

在进行多项式次质询之后，挑战者$\mathcal{C}$撤销所有访问过$\mathcal{O}_{\text{AddF}}(\cdot,\mathcal{G},u_j,PK_i,F_i)$但是没访问过$\mathcal{O}_{\text{RevokeF}}(\cdot,\mathcal{G},u_j,PK_i,F_i)$的用户. 敌手在 Search 中生成一个搜索令牌τ，如果 Search 的输出不为$\perp$，则返回 1，否则返回 0.

定义 4(撤销安全性). 设1^k为安全参数，$\mathcal{A}$为概率多项式时间敌手，敌手赢得$\text{Exp}_{\mathcal{A}}^{\text{Revoke}}(1^k)$的优势为

$$\text{Adv}_{\mathcal{A}_{rev}}^{\text{Revoke}}(1^k)=|\Pr[\text{Exp}_{\mathcal{A}_{rev}}^{\text{Revoke}}(1^k)=1]|.$$

本方案满足撤销安全性，当且仅当对于所有概率多项式时间计算能力的敌手$\mathcal{A}_{rev}$，存在可忽略的函数$\text{negl}(1^k)$，使敌手在$\text{Exp}_{\mathcal{A}_{rev}}^{\text{Revoke}}(1^k)$中获胜的优势满足：

$$\text{Adv}_{\mathcal{A}_{rev}}^{\text{Revoke}}(1^k)\leq\text{negl}(1^k).$$

4.5 详细描述

系统在运行之前首先确定系统公共参数，包括如下内容：选择满足 CPA 安全的公钥广播加密方案 BE={BE.KeyGen, BE.Join, BE.Enc, BE.Dec}、Paillier 同态加密方法 P={P.KeyGen, P.Enc, P.Dec}以及 GM 同态加密方法 GM={GM.KeyGen, GM.Enc, GM.Dec}；选择伪随机函数$F:\{0,1\}^k\times\{0,1\}^*\rightarrow\{0,1\}^k$. 具体的算法及协议描述如下：

(1) 初始化算法. 服务器$\mathcal{S}_1$在初始化过程中负责初始化包含n个节点的社交网络图结构$\mathcal{G}$，其中每个节点$v_i\in\mathcal{V}$用来存储服务器获取的关于用户

$u_i \in \mathcal{U}$ 的相关信息，节点 v_i 和 v_j 之间的边 $e_{ij} \in \mathcal{E}$ 存储 u_i 和 u_j 之间的好友关系.

(2) 密钥生成算法. 为了加入系统，用户 $u_i \in \mathcal{U}$ 需要在本地执行密钥生成算法生成自己的密钥. 算法首先随机选择一个长度为 k 的比特串 k_i 作为伪随机函数 F 的密钥；通过调用 BE.KeyGen、P.KeyGen 以及 GM.KeyGen 算法，生成用户对称密钥 K_i、用户公钥 PK_i 以及用户私钥 SK_i. 算法结束后，用户 u_i 将公钥 PK_i 发布在社交网络平台中.

(3) 注册协议. 用户 u_i 与服务器 $\mathcal{S}_1$ 交互执行注册协议来注册系统. 协议中用户 u_i 调用 BE.Join 算法生成服务器 $\mathcal{S}_1$ 的会话密钥 $k_{\mathcal{S}_1}^i$；构建长度为 d 的表 F_i 用来存储用户 u_i 的好友以及相关信息：对于每个好友 u_j，F_i 中的对应元素以 u_j 为键，以对应的会话密钥 k_j^i 为值. 之后将服务器 $\mathcal{S}_1$ 添加进其中；随机选择一个长度为 k 的比特串作为当前的状态值 st_i；调用 BE.Enc 算法将状态值 st_i 面向 F_i 进行广播加密，生成状态密文 cst_i；发送注册请求 R 至服务器 $\mathcal{S}_1$，其中 u_i 表示用户名. 服务器 $\mathcal{S}_1$ 在图 $\mathcal{G}$ 中为用户 u_i 选取节点 v_i，并将其赋值 R.

(4) 位置更新协议. 用户 u_i 和服务器 $\mathcal{S}_1$ 交互执行位置更新协议生成当前位置密文并更新到系统中. 用户 u_i 首先提取出地理位置信息 l_i，并将 l_i 映射到 Z_k 中的整数 x_i，计算其平方值 x_i. 为了向服务器 $\mathcal{S}_1$ 隐藏真实地理位置，用户利用伪随机函数 F 和密钥 k_i 对 x_i 和 x_i 进行加密，生成位置密文 $L_i=(c_{x_i}, c_{x_i^2})$，并将 L_i 发送至服务器 $\mathcal{S}_1$. 服务器 $\mathcal{S}_1$ 在图 $\mathcal{G}$ 中提取节点 v_i，更新节点 v_i 的赋值：$v_i \leftarrow v_i \| L_i$.

(5) 添加好友协议. 用户 u_i 添加 u_j 为其好友时，需要同服务器 $\mathcal{S}_1$ 交互来执行添加好友协议. 用户 u_i 首先在本地将 u_j 添加进好友表 F_i；调用 BE.Join 算法生成好友 u_j 的会话密钥 k_j^i，并通过安全信道发送给 u_j；u_i 随后更新本地状态 st_i，调用 BE.Enc 算法为新集合 F_i' 生成状态密文 cst_i，随后将 $cst_i \| u_j$ 发送给 $\mathcal{S}_1$. 服务器 $\mathcal{S}_1$ 检查图 $\mathcal{G}$ 中节点 v_i 和 v_j 之间是否存在边，如果不存在则在 v_i 和 v_j 之间添加边 e_{ij}，并更新 v_i 中存储的状态密文值 cst_i.

(6) K-近邻搜索协议. 用户 u_s 同服务器 $\mathcal{S}_1$ 交互来执行 K-近邻搜索协议来搜索距离最近的 K 个好友，具体如过程 2 所示. 首先，用户 u_s 在 F_s 中的所有好友 $(u_1,\cdots,u_d)$ 中提取出对应的状态密文 $(cst_1,\cdots,cst_d)$，调用 BE.Dec 算法，利用会话密钥 $(k_s^1,\cdots,k_s^d)$ 解密得到 $(st_1',\cdots,st_d')$；随后 u_s 将解密结果整合为搜索令牌 $\tau=(st_1',\cdots,st_d')$，并发送给服务器 $\mathcal{S}_1$.

过程 2.　K-近邻搜索协议.

用户 $u_s(\boldsymbol{K_s},\boldsymbol{F_s})$		服务器 $\mathcal{S}_1(\boldsymbol{PK_s},\mathcal{G})$
1. FOR $1\le i\le d$ DO：		
2. $st_i \leftarrow$ BE.Dec(cst_i)		
3. $\tau \leftarrow (st_1',\cdots,st_d')$		
4.	$\xrightarrow{\tau}$	**过程 3.** 安全排序协议 SS.
		服务器 $\mathcal{S}_1(\tau,\boldsymbol{PK_s},\mathcal{G})$ 服务器 $\mathcal{S}_2(\boldsymbol{SK_s})$
		1. $(st_1',\cdots,st_d') \leftarrow \tau$
		2. FOR $1\le i\le d$ DO：
		3. $st_i \leftarrow$ BE.Dec(cst_i)
		4. IF $st_i'=st_i$
		5. $L_i \leftarrow v_i$
		6. $L_i^s \longleftarrow$ RE$(\mathcal{S}_1(L_i,PK_s);\mathcal{S}_2(SK_s)\longrightarrow\perp$
		7. $D_i \longleftarrow$ EDC$(\mathcal{S}_1(L_i,L_i^s,PK_s);\mathcal{S}_2(SK_s)\longrightarrow\perp$
		8. $I \leftarrow \{(u_1,[D_1]),\cdots,(u_d,[D_d])\}$
		9. FOR $1\le i\le d$ DO：
		10. $[u_i] \leftarrow$ P.Enc(u_i)
		11. $\widetilde{R} \leftarrow \{([u_1][D_1]),\cdots,([id][D_d])\}$
		12. $\widetilde{R}^i \leftarrow \widetilde{R}$
		13. FOR $1\le i\le(\log d)^2$ DO：
		14. $\widetilde{R}^{j+1} \leftarrow \widetilde{R}^j$
		15. FOR $1\le i\le d$ DO：
		16. $(([u_x]^i,[D_x]^i),([u_y]^i,[D_y]^i)) \leftarrow P_i$
		17. $P_{i+1} \longleftarrow$ SC$(\mathcal{S}_1(P_i.next),PK_s);\mathcal{S}_2(SK_s)\longrightarrow\perp$
		18. $R \leftarrow \widetilde{R}$
5.	$\xleftarrow{R_K}$	$R_K \leftarrow \{([u_1],[D_1]),\cdots,([u_k],[D_K])\}$
u_s 输出：R_K		$\mathcal{S}_1$ 输出：$\perp$

服务器$\mathcal{S}_1$收到搜索令牌τ之后，在v_s的每个邻接点v_i中提取状态密文cst_i，调用BE.Dec算法，利用u_i发送给服务器的会话密钥$k^i_{S_1}$解密状态cst_i，得到st_i；将st_i与令牌τ中的st'_i进行对比：若相等，则证明u_s和u_i之间好友关系有效，即u_s有权搜索u_i的位置；若无法解密，则证明u_i已被u_s移除好友. 对于有效的好友u_i，在节点v_i中提取出u_i的位置密文L_i，并进行如下步骤：

步骤1. 为了计算搜索用户与好友之间的距离，$\mathcal{S}_1$和$\mathcal{S}_2$执行位置重加密协议LRE(LRE协议的详细过程如附录A(1)所示)，将以u_i的密钥k_i加密的位置密文L_i转换为以u_s的密钥sk_s加密的Paillier密文，LRE协议满足双方仅需一次交互便获得好友位置的重加密密文. 随后，$\mathcal{S}_1$针对用户u_s的所有好友构造包含“用户名-位置密文”的键值对集合$L_s=\{(u_1,L_1),\cdots,(u_d,L_d)\}$.

步骤2. 用户u_s和u_i的平方欧几里得距离D_i可表示为$(x_i-x_s)^2=x_i^2-2x_ix_s+x_s^2$，根据Paillier加密算法的同态加性质，$D_i$的密文可以按照如下方式计算得出：$[D_i]=[(x_i-x_s)^2]=[x_i^2-2x_ix_s+x_s^2]=[x_i^2][-2x_ix_s][x_s^2]$. $\mathcal{S}_1$和$\mathcal{S}_2$执行平方欧几里得距离密文计算协议EDC计算$[D_i]$(EDC协议的详细过程如附录A(2)所示). 服务器$\mathcal{S}_1$和协同服务器$\mathcal{S}_2$执行位置密文重加密协议得到距离密文$[x_i]$，继而$\mathcal{S}_1$进行Paillier密文的同态加运算$[x_s][x_i]^{-1}$得到$[D_i]$. EDC协议执行完毕后，$\mathcal{S}_1$构建包含“好友用户名-距离密文”的键值对集合$I=\{(u_1,[D_1]),\cdots,(u_d,[D_d])\}$.

步骤3. 随后，服务器$\mathcal{S}_1$通过协同服务器$\mathcal{S}_2$的协助得到最终有序的搜索结果. 为了不泄露排序信息，对于每个好友的用户名u_i，$\mathcal{S}_1$利用Paillier加密算法P.Enc(u_i)将其加密为$[u_i]$，生成$\tilde{R}=\{([u_1],[D_1]),\cdots,([id],[D_d])\}$. 服务器$\mathcal{S}_1$进而以$D_i$为主键对$\tilde{R}$中的元素进行两两比较并排序，排序过程中$\mathcal{S}_1$同$\mathcal{S}_2$执行基于可证安全的DGK协议[36]的安全比较协议SC来比较$([u_x],[D_x])$和$([u_y],[D_y])$中的D_x和D_y大小(SC协议的详细过程如附录A(3)所示). SC协议满足$\mathcal{S}_1$得到基于D_x和D_y的大小排序的有序$([u_x],[D_x])$和$([u_y],[D_y])$，但是不会获得D_x和D_y的明文、真实距离大小以及比较结果v.

随后，$\mathcal{S}_1$与$\mathcal{S}_2$以$\tilde{R}$作为输入并以SC协议作为黑盒执行安全排序协议SS(如过程3所示). 基于Batcher排序网络思想，将$\tilde{R}$序列分为前半部分和后半部分，将两个子序列以距离密文为键进行奇偶排序(以SC协议作为黑盒)，随后对整个序列进行奇偶归并，最后将相邻元素，即对键为$2l$和$2l+1$($1\leqslant l\leqslant(d/2)-1$)进行邻近排序，得到最终的排序之后的序列$R=\{([u_1],[D_1]),\cdots,([u_d],[D_d])\}$.

步骤4. 步骤3后，$\mathcal{S}_1$得到根据距离排序并重加密的搜索结果$R=\{([u_1],[D_1]),\cdots,([u_d],[D_d])\}$. $\mathcal{S}_1$将R中前K个元素作为最终结果$R_K=\{([u_1],[D_1]),\cdots,([u_K],[D_K])\}$返回给用户$u_s$，用户$u_s$可以利用Paillier解密算法P.Dec$[u_i]$依次解密结果中的用户名密文$[u_i]$，从而得到按照距离远近排序的$K$个好友$(u_1,\cdots,u_K)$. 除此之外，用户同时可以利用Paillier解密算法P.Dec$[D_i]$解密结果中的距离密文$[D_i]$，得到和好友u_i之间的实际的距离.

(7) 撤销好友协议. 用户u_i同服务器$\mathcal{S}_1$运行撤销好友协议来撤销好友集合中u_j的搜索特权. 首先，用户u_i从好友集合F_i中删除u_j；随机选择一个长度为k的比特串作为新状态值st_i. 随后调用广播加密算法BE.Enc，输入新的用户集合F'_i和状态值st_i，生成服务器端的状态密文cst_i，并发送cst_i给服务器. 服务器$\mathcal{S}_1$在图结构$\mathcal{G}$中更新v_i中存储的状态密文值cst_i.

5 安全性分析

本节首先对本文提出的方案的正确性和安全性进行概述，并分别对其进行分析与数学证明.

5.1 正确性

根据定义2，方案的正确性意味着对于所有系统中的用户，在正确执行位置更新、添加好友以及撤销好友操作的前提下，与服务器$\mathcal{S}_1$交互执行的K-近邻搜索协议总会得到正确的结果R_K，即R_K符合定义2中描述的K-近邻搜索结果特性，不但满足搜索请求，而且按照距离排序.

证明. 方案的正确性主要体现在以下两个方面：安全排序协议的任意层级中每对距离之间的比较的正确性以及基于Batcher排序网络的安全排序协议的正确性.

(1) 距离比较正确性

首先分析安全排序协议SS的任意层级中每对元素之间的比较的正确性. 设搜索结果为$R=\{([u_1],[D_1]),\cdots,([id],[D_d])\}$，安全比较协议SC的正确性体现在对于输入$\{([u_x],[D_x],[u_y],$

$[D_y]$)}，协议输出的两个距离值 D_x 和 D_y 的顺序是正确的.

假设{($[u_x]$,$[D_x]$,$[u_y]$,$[D_y]$)}中 D_x 和 D_y 是两个 l 比特长的整数，分别表示 u_x 和 u_y 的真实距离. $z=D_y-D_x+2^l$ 是一个 $l+1$ 比特长的整数. 可以看出 z 的第 $l+1$ 位比特值如果 1，则代表 $D_x\leq D_y$；z 的第 $l+1$ 位比特值如果 0，则代表 $D_x>D_y$. 如果将 z 的第 $l+1$ 位比特值表示为 $z\div 2^l$，则可得出 $z=2^l(z\div 2^l)+(z \bmod 2^l)$，其中 $0\leq(z \bmod 2^l)\leq 2^l$. 因此，方案中 $\mathcal{S}_1$ 计算 $[z]$ 为 $[2^l]D_xD_y^{-1} \bmod n^2 (l>k)$，并且通过计算 $[d]:=[z][r] \bmod n^2$ 来隐藏 $[z]$. 因此 $[z]$ 被随机数 r 盲化为新的值 d：$d=z+r$. 所以 $d=2^l(d\div 2^l)+(d \bmod 2^l)=2^l((z\div 2^l)+(r\div 2^l))+((z \bmod 2^l)+(r \bmod 2^l))$.

设 d_l 表示 d 的第 l 位，r_l 表示 r 的第 l 位，λ 表示 d_l 和 r_l 的比较结果：如果 $d_l<r_l$，那么 $\lambda=1$. 由于 GM 方案的同态性，所以 $\|d_l\|\cdot\|r_l\|\cdot\|\lambda\|=|d_l\oplus r_l\oplus\lambda|$. z 的第 $l+1$ 位比特值的明文可计算为 $d_l\oplus r_l\oplus\lambda$. 只要密文不超过阈值范围，即 $l+1+\lambda\leq\log_2 N$，那么密文计算正确性同样保证.

如果 $(z \bmod 2^l)+(r \bmod 2^l)<2^l$，那么 $d\div 2^l=z\div 2^l+r\div 2^l$；如果 $(z \bmod 2^l)+(r \bmod 2^l)\geq 2^l$，那么 $d\div 2^l=z\div 2^l+r\div 2^l+1$. 因此上述等式可表示为 $d\div 2^l=z\div 2^l+r\div 2^l+\lambda$，其中 $\lambda=0\leftrightarrow(z \bmod 2^l)+(r \bmod 2^l)<2^l$.

如果 $\lambda=0$，那么 $d \bmod 2^l=(z \bmod 2^l)+(r \bmod 2^l)$；如果 $\lambda=1$，那么 $d \bmod 2^l=(z \bmod 2^l)+(r \bmod 2^l)-2^l$. 因此 $\lambda=0\leftrightarrow d \bmod 2^l=(z \bmod 2^l)+(r \bmod 2^l)\leftrightarrow d \bmod 2^l\geq(r \bmod 2^l)$. 另一方面，如果 $d \bmod 2^l=(z \bmod 2^l)+(r \bmod 2^l)-2^l<r \bmod 2^l$，那么 $d\div 2^l=z\div 2^l+(r\div 2^l)$. 因为 $(z \bmod 2^l)+(r \bmod 2^l)<2^l$，所以可以得出 $d \bmod 2^l>r \bmod 2^l\leftrightarrow d_l>r_l$，因此 $\lambda=0$.

综上所述，z 的第 $l+1$ 位比特值的明文可计算为 $d_l\oplus r_l\oplus\lambda$ 满足正确性，由于 GM 方案的同态性，因此排序协议中计算 z 的第 $l+1$ 位比特值的 GM 密文 $|d_l||r_l||\lambda|$ 计算的正确性得到保证. 即安全比较协议 SC 正确比较了{($[u_x]$,$[D_x]$,$[u_y]$,$[D_y]$)}中 $[D_x]$ 和 $[D_y]$ 对应的明文距离大小.

(2) 距离排序正确性

上述证明分析了安全排序协议 SS 的任意层级中每对元素之间的比较的正确性，本小节证明基于 Batcher 排序网络的安全排序协议的正确性，即进行 Batcher 排序网络中的层级网络比较之后，协议 SS 输出正确排序的搜索结果，即 $R=\{([u_1],[D_1]),\cdots,([u_d],[D_d])\}$ 满足 $D_1\leq\cdots\leq D_d$.

对于长度为 d 的搜索结果 $\widetilde{R}=\{([u_1],[D_1]),\cdots,([id],[D_d])\}$，安全排序协议 SS 首先将序列分为前半部分和后半部分，两个子序列同时以距离密文为键进行奇偶排序，随后，对整个序列进行奇偶归并(对索引为 $2l$ 和 $2l+1(l=1,2,\cdots,(d/2)-1)$ 进行排序，最后将相邻元素进行邻近排序，得到最终的排序之后的序列$=\{([u_1],[D_1]),\cdots,([id],[D_d])\}$.

在 $\widetilde{R}$ 前后两部分分别排序完成后，可以得出对于 1 和 d 之间的所有 q(除了 1 和 $n/2+1$)，第 $(q-1)$ 个元素肯定小于第 q 个元素. 换句话说，每一个偶数索引对应的距离值都有一个比它的前一个距离值小的数字(因为任何偶数索引对应的距离值和它的前一个距离值的输入相同)，所以偶数索引对应的距离最小值必须大于至少 1 个奇数索引对应的距离值. 类似地，每个奇数索引(除了 1 和 $n/2+1$)的前一个索引对应的距离值比它本身的距离值要小，因此第 $l+1$ 小的奇数索引对应的距离值一定大于至少 $l-1$ 个偶数索引对应的距离值.

设 D_s 表示对前后部分以及奇偶排序后的第 q 个索引对应的距离值，满足 $D_s=\mathrm{P.Dec}_{sk_p}([D_s])$. 之前我们论述了对于任何适当的 l，满足 $D_{2l-1}\leq D_{2l}$ 和 $D_{2l-2}\leq D_{2l+1}$. 因为已经对偶数索引对应的距离值和奇数索引对应的距离值进行了排序，所以可得出 $D_{2l-1}\leq D_{2l}$ 和 $D_{2l-2}\leq D_{2l+1}$. 因此，如果将集合中的元素进行分对：(D_{2l},D_{2l+1})，则可得出每个对的两个元素的距离值都大于或等于前一个对的两个元素对应的距离值. 因此，为了在对奇数和偶数排序后完成奇偶归并，只需将第 $(l+1)$ 小的奇数索引对应的距离值 D_{2l+1} 与第 l 小的偶数索引对应的距离值 D_{2l} 进行比较来进行奇偶归并，即完成方案中排序的最后一步. 因此可以得出 $D_1\leq D_2\leq\cdots\leq D_d$. 即安全排序协议 SS 的输出 $R=\{([u_1],[D_1]),\cdots,([u_d],[D_d])\}$ 是一个正确的基于距离排序的搜索结果，满足 $D_1\leq\cdots\leq D_d$.　　证毕.

5.2　自适应 $\mathcal{L}$-语义安全

在安全性分析中，首先分析在方案的执行过程中泄露给服务器的信息，然后给出泄露函数的形式化定义.

(1) 在注册协议以及任意顺序的多项式 q 次的添加好友和撤销好友协议中，给定 u_i 的第 j 次更新的状态密文值 cst_i^j，$\mathcal{S}_1$ 能够学到当前 cst_i^j 的大小：$|cst_i^j|$，以及状态更新历史. 将这些信息记为 $\mathcal{L}_1$：

$$\mathcal{L}_1=\{|cst_i^1|,\cdots,|cst_i^q|\}.$$

(2)在多项式 q 次位置更新协议中,给定 u_i 的第 j 次更新协议中当前位置密文 $L_i^j=(c_{x_i}^j,c_{x_i^2}^j)$,$\mathcal{S}_1$ 能够学到 L_i^j 的更新历史以及 $c_{x_i}^j$ 和 $c_{x_i^2}^j$ 的长度信息.将这些信息记为 $\mathcal{L}_2$:

$$\mathcal{L}_2=\{(|c_{x_i}^j|,|c_{x_i^2}^j|)_{1\le j\le q}\}.$$

(3)在多项式 q 次搜索协议中,$\mathcal{S}_1$ 得到 q 次搜索的搜索令牌 $\tau_1,\cdots,\tau_q$,并得到搜索结果 $\boldsymbol{R}_i$.$\mathcal{S}_1$ 可以从上述内容中学习到每个搜索结果 $\boldsymbol{R}_i$ 和搜索令牌 τ_i 之间的关联以及用户的搜索历史.设 Q 为 q 次搜索的搜索令牌 $\tau_1,\cdots,\tau_q$ 的集合 $Q=\tau_1,\cdots,\tau_q$,$l_{\max}$ 表示在集合 Q 中长度最长的搜索令牌,n_q 次搜索之后的搜索历史可表示为一个二进制四阶的矩阵 $\boldsymbol{Q}_{n_q\times n_q\times l_{\max}\times l_{\max}}$.将这些信息记为 $\mathcal{L}_3$:

$$\mathcal{L}_3=\{\boldsymbol{Q}_{n_q\times n_q\times l_{\max}\times l_{\max}}\}.$$

(4)在对每次搜索结果进行排序的过程中,需要 $\mathcal{S}_1$ 与 $\mathcal{S}_2$ 进行多轮交互,每轮交互中,$\mathcal{S}_2$ 能够学习到的知识包括和 $\mathcal{S}_1$ 运行两方安全密文比较协议中的得到的消息($[\![z]\!]$,$[\lambda]$),$\mathcal{S}_1$ 能够学习到的知识包括和 $\mathcal{S}_2$ 运行两方安全密文比较协议中的得到的消息($[D_x]$,$[D_y]$,l).之后 $\mathcal{S}_2$ 能够学习到交互轮数 $(\log d)^2$,在每对比较过程中,$\mathcal{S}_2$ 得到($[u_x]$,$[D_x]$),($[u_y]$,$[D_y]$),$\mathcal{S}_1$ 得到 $\{([\![z]\!],[\lambda])\}$.上述信息记为 $\mathcal{L}_4$:

$$\mathcal{L}_4^{\mathcal{S}_1}=\{([\![z]\!]_i,[\lambda]_i)_{1<i<(\log d)^2}\},$$

$$\mathcal{L}_2^{\mathcal{S}_2}=\{([D_y]_i,[D_x]_i,l)_{1<i<(\log d)^2}\}.$$

定理 1. 若 Paillier 加密方案以及 GM 加密方案是 CPA 安全的,F 为伪随机函数,并且 DGK 协议在随机预言模型下具有语义安全性,则本方案在随机预言模型下针对自适应敌手 $\mathcal{A}_{\mathcal{S}_1}$ 与 $\mathcal{A}_{\mathcal{S}_2}$ 攻击满足定义 3 中的自适应 $\mathcal{L}$-语义安全性.

证明. 给定泄漏函数 $\mathcal{L}_1$ 的信息,模拟器 Sim_1 可以学习到状态密文值 cst_i^j 的大小 $|cst_i^j|$ 以及状态更新历史 $\{|cst_i^1|,\cdots,|cst_i^q|\}$.随后,可以使用随机值来模拟状态密文值 $(\widetilde{cst}_i^j)$.由于广播加密方案的 CPA 安全性且密文 cst_i^j 是由 BE.Enc 算法生成的,所以 cst_i^j 具有随机性,即模拟值 $\widetilde{cst}_i^j$ 与真实值 cst_i^j 是无法区分的.因此,模拟器不能从状态更新历史 $\{cst_i^1,\cdots,cst_i^q\}$ 中学到额外信息,满足:

$$\text{Output}_{\mathcal{A}_{\mathcal{S}_1}}^{\text{Real}}(cst_i^1,\cdots,cst_i^q)\approx\text{Output}_{Sim_1}^{\text{Ideal}}(\widetilde{cst}_i^1,\cdots,\widetilde{cst}_i^q).$$

给定泄漏函数 $\mathcal{L}_2$ 的信息,模拟器 Sim_1 获取 u_i 的第 j 次位置更新协议中当前位置密文 $L_i^j=(c_{x_i}^j,c_{x_i^2}^j)$,可以学习到 L_i^j 的大小 $|L_i^j|$ 以及 $c_{x_i}^j$ 和 $c_{x_i}^j$ 的长度信息.随后,可以使用随机值来模拟密文位置,并输出模拟值 $(\widetilde{L}_i^j=\tilde{c}_{x_i}^j,\tilde{c}_{x_i^2}^j)$.由于 Paillier 方案的 CPA 安全性以及伪随机函数 F 的伪随机性,又由于密文 L_i^j 是由 P.Enc 算法以及伪随机函数 F 生成的,所以 L_i^j 具有随机性,即模拟值 $\widetilde{L}_i^j$ 与真实值 L_i^j 是无法区分的.因此,模拟器不能从位置历史 $\{L_i^1,\cdots,L_i^q\}$ 中学到额外信息,满足:

$$\text{Output}_{\mathcal{A}_{\mathcal{S}_1}}^{\text{Real}}(\{L_i^1,\cdots,L_i^q\})\approx\text{Output}_{Sim_1}^{\text{Ideal}}(\{\widetilde{L}_i^1,\cdots,\widetilde{L}_i^q\}).$$

给定泄漏函数 $\mathcal{L}_3$ 的信息,模拟器 Sim_1 可以学习到 q 次搜索的搜索令牌信息 $\tau_1,\cdots,\tau_q$.随后,使用随机值来构建模拟令牌,并输出作为模拟值 $\tilde{\tau}_i$.由于搜索令牌 $\tau_i=(st_1',\cdots,st_d')$ 是由 BE.Dec 算法利用添加私钥 $(k_s^1,\cdots,k_s^d)$ 解密状态 $(cst_1,\cdots,cst_d)$ 生成的.由于密钥 k_s^i 为随机数,因此令牌 τ_i 中任意 st_j' 具有随机性.因此,模拟值 $\tilde{\tau}_i$ 与真实值 τ_i 是无法区分的,满足:

$$\text{Output}_{\mathcal{A}_{\mathcal{S}_1}}^{\text{Real}}(\tau_1,\cdots,\tau_q)\approx\text{Output}_{Sim_1}^{\text{Ideal}}(\tilde{\tau}_1,\cdots,\tilde{\tau}_q).$$

给定泄漏函数 $\mathcal{L}_4$ 的信息,Sim_2 能够学习到每轮两方安全密文比较协议中的泄露的消息($[\![z]\!]$,$[\lambda_i]$)以及需要进行的轮数 $(\log d)^2$.所以,Sim_2 在每轮中能够学习到的知识为 $[D_x]$,$[D_y]$,l.由于 $\mathcal{A}_{\mathcal{S}_1}$ 与 $\mathcal{A}_{\mathcal{S}_2}$ 的交互为多次密文比较协议组成的 $(\log d)^2$ 层 Batcher 排序网络,所以通过证明 Sim_1 和 Sim_2 能够通过泄漏函数来模拟密文比较协议中的 $\mathcal{A}_{\mathcal{S}_1}$ 与 $\mathcal{A}_{\mathcal{S}_2}$,进而模拟 $(\log d)^2$ 层 Batcher 排序网络.

在进行安全比较协议 SC 时,$\mathcal{A}_{\mathcal{S}_1}$ 得到的信息可表示为 $view_{\mathcal{A}_{\mathcal{S}_1}}=([D_x],[D_y],l,pk_i',pk_i,r;[\![\lambda]\!],[z_l])$.给定 $([D_x],[D_y],l,pk_i',pk_i)$,构建一个模拟 $\mathcal{A}_{\mathcal{S}_1}$ 的 Sim_1:选择 $\tilde{r}\leftarrow(2,2^{\lambda+l})\cap Z$ 以及 $\tilde{z}_l\leftarrow(2,2^{\lambda+l})\cap Z$,利用 GM.Enc 算法将 $\tilde{\lambda}$,$\tilde{z}_l$ 加密为 $[\![\tilde{\lambda}]\!]$,$[\![\tilde{z}]\!]$;输出 $view_{Sim_1}=([D_x],[D_y],l,pk_i',pk_i,\tilde{r},[\tilde{z}_l])$.

对于 $view_{\mathcal{A}_{\mathcal{S}_1}}$ 和 $view_{Sim_1}$,r 为从均匀分布 $(0,2^{\lambda+l})\cap Z$ 中提取的随机值,而 $[\![\tilde{z}]\!]$ 为 Paillier 加密方案的二重密文,也具有随机性.因此,

$$([D_x],[D_y],l,pk_i',pk_i)=$$
$$([D_x],[D_y],l,pk_i',pk_i,r,[\tilde{z}_l]).$$

除此之外,r 和 $\tilde{r}$ 都是从同一个均匀分布中提取的,基于 GM 方案的 CPA 安全性可以得出:$\mathcal{A}_{\mathcal{S}_1}$ 得到的信息和 Sim_1 得到的信息具有计算不可区分性.

同理,$\mathcal{A}_{\mathcal{S}_2}$ 得到的信息可表示为 $view_{\mathcal{A}_{\mathcal{S}_2}}=(sk_i',sk_i,[\![z]\!],[\![\lambda]\!])$,其中 sk_i' 是 GM 加密方案的私钥,sk_i 是 Paillier 加密方案的私钥.给定 $(sk_i',sk_i,[\![z]\!],$

$[\lambda]$)，构建一个模拟 $\mathcal{A}_{S_2}$ 的 Sim_2：计算 $\tilde{\lambda}$ 的 GM 方案的密文 $\|\tilde{\lambda}\|$，用于表示 $\|\tilde{\lambda}\| \approx \|[D_x] \leq [D_y]\|$；随机选择 $\tilde{z} \leftarrow (0, 2^{\lambda+l}) \cap Z$；利用 P.Enc 算法将 $\tilde{z}$ 加密为 $[\tilde{z}]$；输出 $view_{Sim_2} = (sk'_i, sk_i, l, [\tilde{z}], \|\tilde{\lambda}\|)$.

在 Real(1^k)中，$z = x + r$，其中 x 是一个 l 比特长的整数并且 r 是一个 $l+\lambda$ 比特长的整数，所以 $\tilde{z}$ 的分布和 z 具有不可区分性. 因此可得出 $(sk_i, [\tilde{z}]) = (sk_i, [z])$. 因为 $\tilde{z}$ 和 z 的分布独立于 t，而且 $(sk'_i, sk_i, [\tilde{z}], \|\tilde{\lambda}\|) = (sk'_i, sk_i, l, [\tilde{z}], \|\tilde{\lambda}\|)$，所以 $(sk'_i, sk_i, l, [\tilde{z}], \|\tilde{\lambda}\|) = (sk'_i, sk_i, l, [z], \|\lambda\|)$.

除此之外，由于 $(sk'_i, \|\tilde{\lambda}\|) = (sk'_i, \|D_x \leq D_y\|)$，所以 $(sk'_i, sk_i, l, [z], \|\tilde{\lambda}\|) = (sk'_i, sk_i, l, [z], \|D_x \leq D_y\|)$. 由于 DGK 协议为随机语言模型下具有语义安全性[30]，所以

$$\mathrm{Output}^{\mathrm{Real}}_{\mathcal{A}_{S_2}}(sk'_i, sk_i, \|\lambda\|, [z]) \approx$$

$$\mathrm{Output}^{\mathrm{Ideal}}_{Sim_2}(sk'_i, sk_i, \|\tilde{\lambda}\| \approx \|D_x \leq D_y\|, [\tilde{z}], l).$$

由于 Sim_2 能够从泄漏函数 $\mathcal{L}_4$ 中获取需要排序的轮数 $(\log d)^2$，所以 Sim_1 和 Sim_2 能够模拟每次安全比较协议中的 $\mathcal{A}_{S_1}$ 与 $\mathcal{A}_{S_2}$，进而模拟 $(\log d)^2$ 轮 Bathcer 排序网络. 所以，给定泄漏函数 $\mathcal{L}_4$ 的信息，Sim_1 和 Sim_2 能够像真实方案一样应答用户的质询，不同的是其使用的是模拟值.

综上所述，对于敌手 $\mathcal{A}_{S_1}$，存在一个概率多项式时间的模拟器 Sim_1 满足下面两个分布式具有计算不可区分性：$\mathrm{Output}^{\mathrm{Real}}_{\mathcal{A}_{S_1}} \approx \mathrm{Output}^{\mathrm{Ideal}}_{Sim_1}$.

对于敌手 $\mathcal{A}_{S_2}$，存在一个概率多项式时间的模拟器 Sim_2 满足下面两个分布式具有计算不可区分性：$\mathrm{Output}^{\mathrm{Real}}_{\mathcal{A}_{S_2}} \approx \mathrm{Output}^{\mathrm{Ideal}}_{Sim_1}$.

因此，本方案在随机预言模型下满足定义 3 中的自适应 $\mathcal{L}$-语义安全性，定理 1 得证.　　证毕.

5.3　撤销安全性

定理 2.　若公钥广播加密方案 BE 是 CPA 安全的，则本方案在随机预言模型下针对自适应敌手 $\mathcal{A}_{rev}$ 满足定义 4 中的撤销安全性.

证明.　设敌手 $\mathcal{A}_{rev}$ 在定义 4 中赢得博弈实验 $\mathrm{Exp}^{\mathrm{Revoke}}_{\mathcal{A}}(1^k)$ 的优势 $\mathrm{Adv}^{\mathrm{Revoke}}_{\mathcal{A}_{rev}}(1^k)$ 为 δ，证明主要思路是首先假设 δ 不可忽略，那么可以构造一个敌手 $\mathcal{A}_{be}$，利用 $\mathcal{A}_{rev}$ 作为工具，攻破广播加密方案 BE 的 CPA 安全性.

为了使 $\mathrm{Exp}^{\mathrm{Revoke}}_{\mathcal{A}_{rev}}(1^k)$ 的输出为 1，敌手 $\mathcal{A}_{rev}$ 需要提供一个有效的搜索令牌. 值得说明的是，每次撤销好友之后系统都会更新 st_i 并将进行加密：$\mathrm{BE.Enc}_{bpk_i}(st_i, F_i \backslash u_j)$，其中 $F_i \backslash u_j$ 是新的好友集合，随后将密文广播给系统中所有用户. 广播加密的安全性保证只有用户 u_i 的好友能够解密状态密文值. 因此敌手 $\mathcal{A}_{rev}$ 必须为用户 u_i 的有效好友才能提供有效的搜索令牌，否则与广播加密方案 BE 的 CPA 安全性相违背，即敌手只能以可忽略的概率 2^{-k} 提供一个有效的令牌.

设敌手 $\mathcal{A}_{be}$ 和挑战者 $\mathcal{C}$ 运行广播加密方案 BE，敌手 $\mathcal{A}_{be}$ 和 $\mathcal{A}_{rev}$ 运行实验 $\mathrm{Exp}^{\mathrm{Revoke}}_{\mathcal{A}_{rev}}(1^k)$，其中敌手 $\mathcal{A}_{be}$ 充当实验 $\mathrm{Exp}^{\mathrm{Revoke}}_{\mathcal{A}}(1^k)$ 中的挑战者角色：

(1) 挑战者 $\mathcal{C}$ 运行 $\mathrm{BE.KeyGen}(1^k)$ 生成密钥 (msk_{be}, bpk_i). 敌手 $\mathcal{A}_{be}$ 初始化 F_i，选择随机值 st_i，并将 (st_i, F_i) 发送给挑战者 $\mathcal{C}$. 挑战者 $\mathcal{C}$ 运行 $\mathrm{BE.Enc}_{bpk_i}(st_i, F_i)$ 生成 st_{S_1}，并发送给敌手 $\mathcal{A}_{be}$. 敌手 $\mathcal{A}_{be}$ 运行 KeyGen 生成密钥 K_{u_i}，运行 Register 生成密钥 $k^i_{S_1}$，其中 K_{u_i} 不包含 k_{be}.

(2) 敌手 $\mathcal{A}_{be}$ 质询挑战者 $\mathcal{C}$ 关于 $\mathcal{A}_{rev}$ 的密钥. 挑战者 $\mathcal{C}$ 运行 $\mathrm{BE.Join}_{msk_{be}}(\mathcal{A}_{rev})$ 生成 $k_{\mathcal{A}_{rev}}$，发送 $k_{\mathcal{A}_{rev}}$ 给敌手 $\mathcal{A}_{be}$. 为了将 $\mathcal{A}_{rev}$ 加入系统并使其成为用户 u_i 的好友，敌手 $\mathcal{A}_{be}$ 需要同时更新状态密文. 敌手 $\mathcal{A}_{be}$ 发送 F_i 和新生成的 st_i 给挑战者 $\mathcal{C}$，挑战者 $\mathcal{C}$ 运行 $\mathrm{BE.Enc}_{bpk_i}(st_i, F_i)$ 生成新的状态密文 cst_i. 敌手 $\mathcal{A}_{be}$ 运行 AddF 生成 $\mathcal{A}_{rev}$ 的密钥 $k^i_{\mathcal{A}_{rev}}$.

(3) 敌手 $\mathcal{A}_{be}$ 运行 Setup 算法生成图结构 $\mathcal{G}$，并且将 $k^i_{\mathcal{A}_{rev}}$ 和 $\mathcal{G}$ 发送给 $\mathcal{A}_{rev}$. $\mathcal{A}_{rev}$ 可以访问预言机 $\mathcal{O}_{\mathrm{AddF}}$ 和 $\mathcal{O}_{\mathrm{RevokeF}}$.

(4) 敌手 $\mathcal{A}_{be}$ 通过运行 RevokeF 来对 $\mathcal{A}_{rev}$ 撤销其好友身份，$\mathcal{A}_{be}$ 生成针对 st_i 的两个状态值 $st_{i0} \leftarrow \{0,1\}^k$ 和 $st_{i1} \leftarrow \{0,1\}^k$，并将 st_{i0} 和 st_{i1} 作为敌手 $\mathcal{A}_{be}$ 的挑战值发送给挑战者 $\mathcal{C}$，同时发送已撤销 $\mathcal{A}_{rev}$ 的好友集合 F_i.

(5) 挑战者 $\mathcal{C}$ 选择比特 $b \in \{0,1\}$，调用 $\mathrm{BE.Enc}_{bpk_i}(st_{ib}, F_i)$ 算法加密 st_{ib} 生成 cst_i，发送 cst_i 给敌手 $\mathcal{A}_{be}$ 作为其对广播加密方案 BE 的 CPA 安全性的挑战结果，$\mathcal{A}_{be}$ 发送 cst_i 给敌手 $\mathcal{A}_{rev}$ 作为 $\mathrm{Exp}^{\mathrm{Revoke}}_{\mathcal{A}}(1^k)$ 的挑战结果.

(6) 敌手 $\mathcal{A}_{rev}$ 生成令牌 τ，发送 τ 给敌手 $\mathcal{A}_{be}$. 因为敌手 $\mathcal{A}_{rev}$ 在定义 4 中赢得实验 $\mathrm{Exp}^{\mathrm{Revoke}}_{\mathcal{A}}(1^k)$ 的优势不可忽略，所以 τ 有效的概率不可忽略.

(7) 如果 $t_0 \neq \perp$，那么 Search 协议中止，根据以下两种情况，$\mathcal{A}_{be}$ 输出对 b 的猜测：

① 如果 $t_0 \neq \perp$，那么 $\mathcal{A}_{be}$ 用的是 st_{i0} 作为状态值生成的令牌，$\mathcal{A}_{be}$ 输出对 b 的猜测 $b' = 0$；

② 如果 $t_1 \neq \perp$，那么 $\mathcal{A}_{be}$ 用的是 st_{i1} 作为状态值

生成的令牌，$\mathcal{A}_{be}$输出对b的猜测$b'=1$.

通过以上分析，计算$\mathcal{A}_{be}$攻破BE方案CPA安全性的优势$\mathrm{Adv}^{\mathrm{BE}}_{\mathcal{A}_{be}}(1^k)$为

$$\begin{aligned}\mathrm{Adv}^{\mathrm{BE}}_{\mathcal{A}_{be}}(1^k)&=|[(\Pr[(t_0\vee t_1)\neq\perp]\cdot 1-1/2)+\\&\quad(\Pr[(t_0\wedge t_1)\neq\perp]\cdot 1/2-1/2)|\\&=|[\delta\cdot 1+(1-\delta)\cdot 1/2-1/2]|\\&=|[(\delta+1)/2-1/2]|=\delta/2.\end{aligned}$$

因为$\mathcal{A}_{rev}$在定义4中赢得实验$\mathrm{Exp}^{\mathrm{Revoke}}_{\mathcal{A}}(1^k)$的优势$\delta$不可忽略，所以$\mathcal{A}_{be}$攻破公钥广播加密方案BE的CPA安全性的优势$\delta/2$不可忽略，这与公钥广播加密方案BE的CPA安全性相矛盾，所以不存在敌手$\mathcal{A}_{rev}$能够以不可忽略的概率赢得实验$\mathrm{Exp}^{\mathrm{Revoke}}_{\mathcal{A}}(1^k)$，因此本文方案满足定义4的撤销安全性，定理2得证. 证毕.

6 性能分析与实验

本节首先从理论层面对方案的各个阶段的性能进行整体评估，并与相关方案进行安全性以及特性对比，最后通过原型系统实验来进一步对方案的性能以及可扩展性进行分析.

6.1 代价分析

针对方案构建过程中的存储代价、计算代价以及通信代价三个维度分析方案性能，如表1所示.

表1 代价分析

	$Stor_u$	$Stor_{\mathcal{S}_1}$	$Stor_{\mathcal{S}_2}$	$Comp_u$	$Comp_{\mathcal{S}_1}$	$Comp_{\mathcal{S}_2}$	$Comm_u$	$Comm_{\mathcal{S}_1}$	$Comm_{\mathcal{S}_2}$
注册	$\mathcal{O}(1)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(1)$			$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$
添加好友	$\mathcal{O}(d)$	$\mathcal{O}(nd)$		$\mathcal{O}(d)$			$\mathcal{O}(d)$	$\mathcal{O}(nd)$	
删除好友				$\mathcal{O}(1)$	$\mathcal{O}(1)$		$\mathcal{O}(1)$	$\mathcal{O}(1)$	
位置更新		$\mathcal{O}(n)$		$\mathcal{O}(1)$			$\mathcal{O}(1)$		
K-近邻搜索				$\mathcal{O}(d)$	$\mathcal{O}(d+(\log d)^2)$	$\mathcal{O}(d+(\log d)^2)$	$\mathcal{O}(K)$	$\mathcal{O}((\log d)^2)$	$\mathcal{O}((\log d)^2)$

$Stor$：存储代价；$Comp$：计算代价；$Comm$：通信代价

设n表示系统中承受的最大用户数，d为每个用户最多添加的好友数量. 存储代价方面，如果n个用户完成注册且每个用户添加d个好友，那么服务器$\mathcal{S}_1$存储的图结构包含n个节点以及$nd/2$条边，存储代价为$\mathcal{O}(nd)$. 由于注册和添加好友时用户仅需要在本地存储私钥以及好友密钥，故用户端的存储代价为$\mathcal{O}(d)$. 计算代价方面，注册以及位置更新协议中用户的计算代价为常数级$\mathcal{O}(1)$，添加d个好友的计算代价为$\mathcal{O}(d)$. 搜索协议中用户生成搜索令牌的计算代价为$\mathcal{O}(d)$，服务器$\mathcal{S}_1$在社交网络图结构中提取用户好友位置的代价与好友数量呈线性关系$\mathcal{O}(d)$，两个服务器密文距离计算的计算代价均为$\mathcal{O}(d)$，距离排序的代价为$\mathcal{O}((\log d)^2)$，因此总计算代价为$\mathcal{O}(d+(\log d)^2)$.

除此之外，在通信代价方面，注册协议中用户和服务器$\mathcal{S}_1$和$\mathcal{S}_2$之间的通信复杂度为$\mathcal{O}(1)$；用户添加d个好友需要和服务器$\mathcal{S}_1$之间的通信复杂度为$\mathcal{O}(d)$. 搜索阶段，用户需要发送的搜索令牌大小与好友数量呈线性关系，即与服务器$\mathcal{S}_1$之间通信复杂度为$\mathcal{O}(d)$；服务器$\mathcal{S}_1$和服务器$\mathcal{S}_2$需要进行$\mathcal{O}(d)$轮交互来计算距离密文，并进行$\mathcal{O}(d(\log d)^2)$轮交互对距离密文进行排序. 如果部署并行Batcher排序网络，那么$\mathcal{S}_1$和服务器$\mathcal{S}_2$的搜索阶段通信复杂度可优化至$\mathcal{O}((\log d)^2)$.

6.2 方案对比

本节将方案与支持隐私保护的位置搜索方案[14,18,21]和支持排序的密文搜索方案[28,37-38]进行对比，其中文献[21,28,38]中的方案均为双服务器协同架构. 需要说明的是，由于本方案与现有方案在应用场景、安全模型、评价指标等因素差异性较大，难以进行性能上的量化比较，因此本文着重进行了特性以及安全性的对比，比较结果如表2所示.

表2 特性比较

	精确性	度量方法	灵活更新性	密码学工具	搜索隐私	位置隐私	访问控制	排序实体
文献[14]	×	线性规划	√	HMAC	√	√	×	—
文献[18]	√	欧氏距离/锚点	×	PIR/Paillier	√	√	×	—
文献[21]	×	动态网格	×	HMAC	√	×	×	用户
文献[28]	√	Tf-idf	√	保序加密	√	×	×	服务器协同
文献[37]	×	内积相似度	×	LWE-Brakerski	√	√	×	用户
文献[38]	×	协调匹配	×	—	×	×	×	服务器
本方案	√	平方欧氏距离	√	Paillier/GM/BE	√	√	√	服务器协同

在结果精确性方面，文献[14]中的方案通过线性规划技术实现了位置差分隐私性. 文献[21]中的方案基于动态网格结构来聚类位置距离相近的用户. 文献[37]通过内积相似度(Inner Product Similarity, IPS)对相似的元素进行聚类，文献[38]通过协调匹配计算每个条目中匹配搜索令牌的数量，通过计算内积向量乘积对条目进行排序. 然而，上述方法的搜索结果具有一定的误报率，适用于相似性搜索，故无法提供精确的距离密文计算和安全排序. 本方案与文献[18]中的方案均基于欧几里德距离来计算精确距离信息，但文献[18]侧重于搜索一定位置区域内的兴趣点个数，本方案提供可证安全的安全比较协议对搜索到的距离进行比较，从而实现了精确的距离密文排序.

安全性方面，文献[38]利用了启发式方法来隐藏位置信息和搜索模式. 文献[18]利用 PIR 实现了近邻搜索的隐私保护，方案引入锚点技术来提高搜索效率，但会带来一定程度的泄漏. 文献[38]基于 LWE-Brakerski 同态加密方法构建搜索索引，假设服务器仅仅执行搜索功能，并将搜索结果全部返回给用户，用户在本地对结果进行解密并排序，导致通信负载和用户端计算负载过大. 文献[21]和[28]中引入不共谋双服务器模型，但是双方交互过程中其中一方能够完全访问搜索结果的顺序信息，造成顺序泄露，安全性较弱. 本方案基于同态加密算法对用户位置密文进行重加密，使得服务器无需解密即可对距离进行计算并比较，从而在不获取额外信息的前提下对搜索结果进行排序.

除此之外，本方案与其他方案相比具有灵活访问控制机制，用户在进行好友更新与位置加密时均实现常数级计算代价与通信代价，并且用户端仅需存储密钥相关的信息. 在如今移动社交网络环境下，应尽可能保证轻量级用户存储与计算代价. 因此本方案在性能方面更适用于面向瘦客户端的实际移动社交网络部署场景.

6.3　实验分析

本工作对提出的方案进行了编程实现，并通过性能测试来分析方案的性能表现. 实验在 Linux Ubuntu 操作系统执行，用户端和服务器端均采用 Intel Core i7-2600 四核处理器，3.4 GHz，8 GB 内存. 在仿真实验中，安全参数 k 设定为 256 位，广播加密、Paillier 和 GM 公钥加密方法中的密钥长度为 1024 位. 实验使用 OpenSSL 库[①]来实现基础的密码学算法，例如使用 AES-CBC-256 和 SHA256 的 HMAC 算法来实现对称加密和伪随机函数. 除此之外，对于非对称加密算法，实验使用 Relic 库来实现 Paillier 和 GM 同态加密，并利用 BGW2 算法[31]实现公钥广播加密.

本实验数据集来自真实世界中的数据——Enron 邮件数据集[②]. 在数据处理阶段，实验首先在 Enron 邮件数据集中随机选择 1000 个账户作为社交网络总用户集合，其中电子邮件的发送用户和接收用户被表示为社交网络中的好友关系，在(10,50)中选择随机整数作为用户的位置的模拟值. 将上述数据初始化为包含 1000 个节点和 3831 条边的社交网络图结构 $\mathcal{G}$，每个点(用户)由 Z_k 域中一个唯一的值进行标识. 方案部署一个远程主机模拟社交网络服务器，同时部署多台主机模拟和服务器交互的用户客户端. 服务器端运行协同作业分配机制：首先利用一个主服务器存储社交网络图结构数据，并使用线程来模拟执行辅助任务的协同服务器. 为了提交搜索请求，用户客户端只与主服务器进行通信. 实验中每个数据均为实验重复 50 次并取平均值后得到的结果.

6.3.1　存储开销

本小节分析本方案的存储开销. 理论上来讲，服务器端负责存储的密态社交网络图结构 $\mathcal{G}$，每个节点包含用户上传的位置密文以及用于好友访问控制的状态值密文，分别由位置加密算法以及广播加密算法生成，会随着密文的膨胀以及用户节点数的变化而变化；用户端仅需存储密钥，因此存储代价为常数级别. 表 3 展示了实验中密态社交网络图结构的生成时间以及服务器存储代价随着用户节点数变化的变化趋势. 由表 3 可知，服务器端密态社交网络图结构 $\mathcal{G}$ 的存储开销随图中用户节点规模增长近似呈线性方式增长，且对称加密导致的密文膨胀率会带来服务器端存储开销的进一步增加，由于本方案采用对称加密进行位置加密，对称密文相较于 Paillier 同态密文膨胀率有明显减弱，与理论分析一致. 因此，本方案以额外的可接受范围内的存储开销换取用户位置的机密性以及好友信息的隐私性.

① The OpenSSL Project. OpenSSL: The Open Source toolkit for SSL/TLS. http://www.openssl.org/, 2015

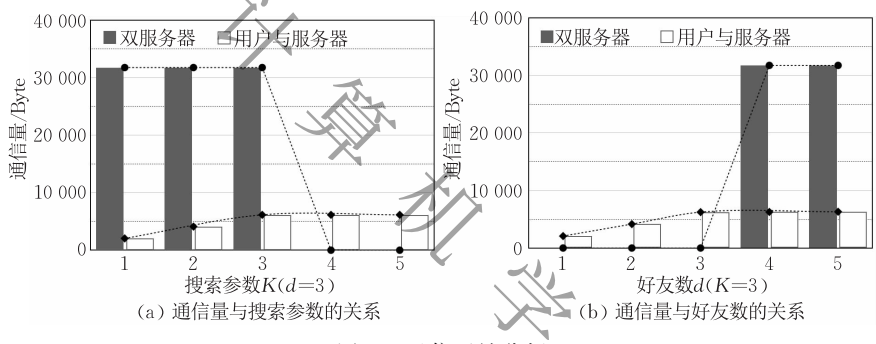
② Cohen W W. Enron Email Dataset. https://www.cs.cmu.edu/~enron/, 2015

表 3 存储开销分析

节点数	社交网络图结构		密态社交网络图结构		膨胀率/%
	存储代价/KB	生成时间/s	存储代价/KB	生成时间/s	
200	18.752	0.423	57.506	2.359	306.665
400	38.101	0.477	115.302	2.941	302.622
600	57.460	0.514	174.379	3.316	303.478
800	74.677	0.538	225.039	3.770	301.349
1000	95.988	0.575	289.864	4.113	301.979

6.3.2 通信开销

通信开销主要分析搜索协议运行时用户与服务器之间以及双服务器之间通信数据量.从理论上来讲,用户发起从 d 个好友中搜索 K 近邻好友的请求时,用户与服务器通信的数据规模为 $\mathcal{O}(K)$;当服务器从总数 d 个好友中获取位置密文并计算与用户之间的距离时,服务器之间通信的数据规模为 $\mathcal{O}((\log d)^2)$.图 2 展示了实验中两类通信开销分别与好友数量 d 和参数 K 之间的关系,其中 y 轴代表了通信数据量(单位为 Byte),x 轴分别代表用户的好友数量 d 和参数 K.总体来说,搜索协议随着 K 的增加,发起搜索的用户需要进行的数据传输量会增加,当参数 K 增加到一定值时(大于 d)数据通信量趋于稳定.服务器之间通信开销主要与用户的好友数 d 正相关,不会随着 K 的增加而变化,且距离计算需要进行多次交互使得服务器间通信数据量相对较大,与理论分析一致,且符合实际移动社交网络架构的瘦客户端部署需求.



(a) 通信量与搜索参数的关系 (b) 通信量与好友数的关系

图 2 通信开销分析

6.3.3 搜索耗时开销

本小节分析 K-近邻搜索协议的主要耗时来源.首先设置仿真实验的用户好友数量为{1,2,3,4,5},并将协议中服务器的计算拆分为位置搜索和距离排序两个子过程.图 3 展示了方案中搜索协议耗时与好友数量之间的关系,其中 y 轴代表了搜索协议耗时(单位为 μs),x 轴代表用户的好友数 d.

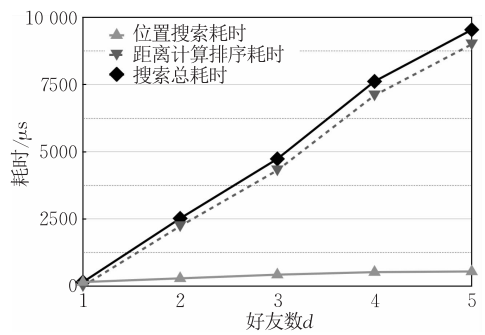


图 3 搜索协议耗时来源

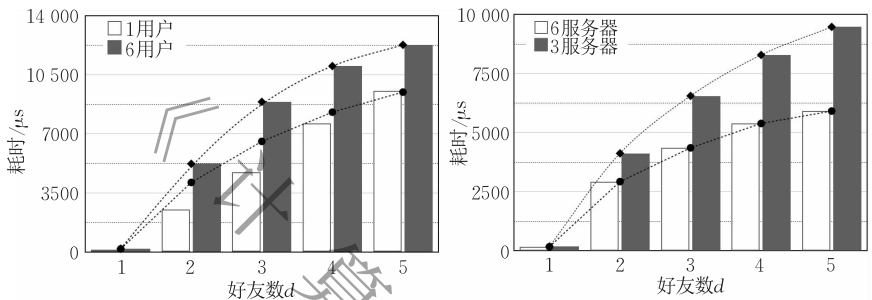
实验首先记录了针对不同数量的好友搜索协议总时间消耗;其次记录了服务器提取好友位置密文并重加密所需的时间消耗;以及服务器对距离密文进行计算和排序所需的时间消耗.可以看出,K-近邻搜索协议中两个子过程的时间消耗大体都随着好友数量的增加而增加,其中位置搜索耗时整体远低于距离计算与排序耗时,且随着好友数量增至 4 之后增速变缓,而距离计算与排序耗时与好友数量数总体呈近似稳定的线性关系.因此可得出结论,协议中服务器对搜索用户和其好友之间的密文距离计算与比较是 K-近邻搜索协议的主要耗时来源,与理论分析一致.

6.3.4 可扩展性

可扩展性分析部分主要分析扩展实体数量对方案稳定性及耗时的影响.实验首先分析了扩展搜索用户数量对 K-近邻搜索协议耗时的影响.具体来说,首先部署 1 台主机模拟 1 个用户执行 K-近邻搜索协议,记录总时间消耗;然后部署 6 台主机模拟 6 个用户重复相同的实验,并比较结果.值得说明的是,记录多用户搜索的时间消耗时,多个用户主机同时将搜索请求发送到主服务器,并记录服务器收到

搜索请求到为每个用户完成搜索为止的起止时间. 图 4(a)展示了方案中同时发起搜索的用户数量与 K-近邻搜索协议耗时之间的关系. 从图中可以看出,1 个用户搜索时协议的运行耗时比 6 个用户搜索时协议运行耗时略低,其中前者耗时近似地与好友的个数呈近似稳定的线性关系;后者耗时较前者有所增加,但是从曲线走势可以看出,6 个用户搜索耗时随着好友数增加而增速变缓且趋近恒定. 因此可以得出,在好友数量较多的情况下,搜索用户数量的增加对 K-近邻搜索协议中的耗时的影响较弱,即系统可承载的用户数量具有可扩展性空间.

除此之外,分析了扩展远程服务器数量对 K-近邻搜索协议耗时的影响. 首先,部署 3 个服务器同时为 6 个用户执行 K-近邻搜索协议,记录总时间消耗;然后部署 6 个服务器重复相同的实验,并比较结果. 图 4(b)展示了服务器数量与 K-近邻搜索协议耗时之间的关系,从图中可以看出,部署 6 个服务器的运行时间比 3 个服务器运行耗时整体显著降低,且前者增速较缓,好友数增至 4 之后耗时趋近恒定,但是后者耗时近似地与好友的个数呈稳定的线性关系. 因此可以得出,通过部署多个服务器执行并行搜索,不仅能显著减少搜索耗时,且能进一步削弱好友数量增加对 K-近邻搜索协议耗时的影响.



(a) 搜索耗时与用户数量的关系　　(b) 搜索耗时与服务器数量之间的关系

图 4　可扩展性分析

值得指出的是,方案搜索过程的主要的计算耗时都用在了同态加解密运算和广播加密的解密运算上. 由于这些计算的效率与所选用的参数和底层算法有很大关系,因此随着参数和算法的不断优化,其效率可以进一步得到提升. 此外,服务端的算法还可以进行进一步优化,例如对距离排序使用多线程来计算,和使用近似排序算法等,从而可以大大降低搜索阶段所花费的时间. 为了忠实地反映方案的原始执行效率,在实验中并未采用任何优化方法,而是让服务端每次都以单线程完整地进行所有的计算步骤.

7　总　结

聚焦于移动社交网络的特征和用户位置隐私保护的多元需求,本文提出一种支持 K-近邻搜索的移动社交网络隐私保护方案. 方案利用安全多方计算思想以及 Paillier 和 GM 同态加密方法设计面向加密位置的 K-近邻搜索协议,协议保证服务器无需解密的前提下可以对用户与好友之间距离进行计算并排序,在保护用户及其好友位置隐私的同时满足用户的近邻搜索需求. 除此之外,方案提出基于广播加密的好友动态管理机制,设计添加好友协议和撤销好友协议来实现赋予或移除好友搜索用户位置的权力,实现对于好友的灵活安全增删以及用户位置隐私的细粒度访问控制. 安全性方面,方案在随机预言模型下满足自适应 $\mathcal{L}$-语义安全性以及撤销安全性;性能方面,本方案的协同架构降低了用户与服务器之间通信代价的同时,减少了向服务器泄露的位置信息与搜索模式. 本方案在进行距离密文计算以及排序时的计算代价具有提升空间,因此在未来的研究工作中,会在保证现有安全性不变的情况下,重点关注如何提高方案的距离排序执行效率. 除此之外,未来考虑使方案满足有隐私保护需求的更多社交网络搜索场景,例如基于兴趣的推荐服务以及车联网路况查询等.

参　考　文　献

[1] Beigi G, Shu K, Zhang Y, Liu H. Securing social media user data: An adversarial approach//Proceedings of the 29th on Hypertext and Social Media. Baltimore, Maryland, 2018: 165-173

[2] Noulas A, Scellato S, Mascolo C, Pontil M. An empirical study of geographic user activity patterns in foursquare//Proceedings of the 5th International AAAI Conference on Weblogs and Social Media. Barcelona, Spain, 2011

[3] Cheng Z, Caverlee J, Lee K, Sui D. Exploring millions of footprints in location sharing services//Proceedings of the International Conference on Weblogs and Social Media. Barcelona, Spain, 2011: 81-88

[4] Preotiuc-Pietro D, Cohn T. Mining user behaviors: A study of check-in patterns in location-based social network//Proceedings of the Conference on ACM Web Science. Paris, France, 2013: 306-315

[5] Chor B, Goldreich O, Kushilevitz E, Sudan M. Private information retrieval//Proceedings of the IEEE 36th Annual Foundations of Computer Science. Milwaukee, Wisconsin, 1995: 41-50

[6] Curtmola R, Garay J, Kamara S, Ostrovsky R. Searchable symmetric encryption: Improved definitions and efficient constructions. Journal of Computer Security, 2016, 19(5): 895-934

[7] Park J, Sandhu R, Cheng Y. A user-activity-centric framework for access control in online social networks. IEEE Internet Computing, 2011, 15(5): 62-65

[8] Cheng Y, Park J, Sandhu R. Relationship-based access control for online social networks: Beyond user-to-user relationships//Proceedings of the 2012 International Conference on Privacy, Security, Risk and Trust. Amsterdam, The Netherlands, 2012: 646-655

[9] Bamba B, Liu L, Pesti P, Wang T. Supporting anonymous location queries in mobile environments with PrivacyGrid//Proceedings of the 17th International Conference on World Wide Web. New York, USA, 2008: 237-246

[10] Chow C Y, Mokbel M F, Liu X. A peer-to-peer spatial cloaking algorithm for anonymous location-based services//Proceedings of the 14th Annual ACM International Symposium on Advances in Geographic Information Systems. Arlington, USA, 2006: 171-178

[11] Beresford A R, Stajano F. Location privacy in pervasive computing. IEEE Pervasive Computing, 2003, 2(1): 46-55

[12] Fu Y, Wang Y. BCE: A privacy-preserving common-friend estimation method for distributed online social networks without cryptography//Proceedings of the 7th International Conference on Communications and Networking. Kun Ming, China, 2013: 212-217

[13] Sun J, Zhang R, Zhang Y. Privacy-preserving spatiotemporal matching//Proceedings of the IEEE INFOCOM. Turin, Italy, 2013: 800-808

[14] Bordenabe N E, Chatzikokolakis K, Palamidessi C. Optimal geo-indistinguishable mechanisms for location privacy//Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (CCS'14). Scottsdale, USA, 2014: 251-262

[15] Elaine S, Richard C, Hubert C, et al. Privacy-preserving aggregation of time-series data//Proceedings of the Network and Distributed System Security Symposium (NDSS 2011). San Diego, USA, 2011: 1-17

[16] Qin Z, Yu T, Yang Y, et al. Generating synthetic decentralized social graphs with local differential privacy//Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. Dallas, USA, 2017: 425-438

[17] Kanjanatarakul O, Kuson S, Denoeux T. An evidential K-nearest neighbor classifier based on contextual discounting and likelihood maximization//Proceedings of the International Conference on Belief Functions. Compiègne, France, 2018: 155-162

[18] Zhou Chang-Li, Chen Yong-Hong, Tian Hui, Cai Shao-Bin. Location privacy and query privacy-preserving method for K-nearest neighbor query in road networks. Journal of Software, 2020, 31(2): 471-492(in Chinese)
(周长利, 陈永红, 田晖, 蔡绍滨. 保护位置隐私和查询内容隐私的路网 K 近邻查询方法. 软件学报, 2020, 31(2): 471-492)

[19] Niu B, Li X, Zhu X, et al. Are you really my friend? The exactly spatiotemporal matching scheme in privacy-preserving mobile social networks//Proceedings of the International Conference on Security and Privacy in Communication Systems. Beijing, China, 2014: 33-40

[20] Li Z, Wang C, Yang S, et al. Lass: Local-activity and social-similarity based data forwarding in mobile social networks. IEEE Transactions on Parallel and Distributed Systems, 2014, 26(1): 174-184

[21] Schlegel R, Chow C, Huang Q, Wong D. User-defined privacy grid system for continuous location-based services. IEEE Transactions on Mobile Computing, 2015, 14(10): 2158-2172

[22] Lindell Y. Secure Multiparty Computation (MPC). IACR Cryptol. ePrint Arch., 2020: 300

[23] Huang Q, Ma Z, Yang Y, et al. Secure data sharing and retrieval using attribute-based encryption in cloud-based OSNs. Chinese Journal of Electronics, 2014, 23(3): 557-563

[24] Rajtmajer S, Squicciarini A, Griffin C, et al. Constrained social-energy minimization for multi-party sharing in online social networks//Proceedings of the 2016 International Conference on Autonomous Agents Multiagent Systems. Singapore, 2016: 680-688

[25] Ali S, Rauf A, Islam N, Farman H. A framework for secure and privacy-protected collaborative contents sharing using public OSN. Cluster Computing, 2019, 22(3): 7275-7286

[26] Van Rompay C, Molva R, Onen M. Multi-user searchable encryption in the cloud//Proceedings of the International Conference on Information Security. Trondheim, Norway, 2015: 299-316

[27] Elmehdwi Y, Samanthula B K, Jiang W. Secure k-nearest neighbor query over encrypted data in outsourced environments//Proceedings of the 2014 IEEE 30th International Conference on Data Engineering. Chicago, USA, 2014: 664-675

[28] Xia Zhihua, Wang Xinhui, Sun Xingming, Wang Qian. A secure and dynamic multi-keyword ranked search scheme over

encrypted cloud data. IEEE Transactions on Parallel and Distributed Systems, 2015, 27(2): 340-352

[29] Agrawal R, Kiernan J, Srikant R, Xu Y. Order preserving encryption for numeric data//Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data. New York, USA, 2004: 563-574

[30] Bost R, Popa R A, Tu S, Goldwasser S. Machine learning classification over encrypted data//Proceedings of the Network and Distributed System Security Symposium (NDSS 2015). San Diego, USA, 2015

[31] Boneh D, Craig G, Brent W. Collusion resistant broadcast encryption with short ciphertexts and private keys//Proceedings of the Advances in Cryptology (CRYPTO 2005). Santa Barbara, USA, 2005: 258-275

[32] Paillier P. Public-key cryptosystems based on composite degree residuosity classes//Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques. Prague, Czech Republic, 1999: 223-238

[33] Goldwasser S, Micali S. Probabilistic encryption. Journal of Computer and System Sciences, 1984, 28(2): 270-299

[34] Damgard I, Mads J, Jesper B N. A generalization of Paillier's public-key system with applications to electronic voting. International Journal of Information Security, 2010, 9(6): 371-385

[35] Batcher K E. Sorting networks and their applications// Proceedings of the Spring Joint Computer Conference. New York, USA, 1968: 307-314

[36] Veugen T. Improving the DGK comparison protocol// Proceedings of the International Workshop on Information Forensics and Security. Tenerife, Spain, 2012: 49-54

[37] Strizhov M, Indrajit R. Multi-keyword similarity search over encrypted cloud data//Proceedings of the IFIP International Information Security Conference. Marrakech, Morocco, 2014: 52-65

[38] Cao N, Wang C, Li M, et al. Privacy-preserving multi-keyword ranked search over encrypted cloud data. IEEE Transactions on Parallel and Distributed Systems, 2014, 25(1): 222-233

附录 A.

K-近邻搜索协议包含位置重加密协议 LRE、距离密文计算协议 EDC 以及安全比较协议 SC 三个子协议.

(1) 位置重加密协议 LRE

过程 4. 位置重加密协议 LRE.

服务器 $\mathcal{S}_1(\boldsymbol{L}_i)$		服务器 $\mathcal{S}_2(\boldsymbol{SK}_s)$
1. $(c_{x_i}, c_{x_i}) \leftarrow L_i$		
2. $(c_{x_i}^1, c_{x_i}^2) \leftarrow c_{x_i}$		
3. $p^* \leftarrow_{\mathcal{R}} \{0,1\}^k$		
4. $c_{x_i}^* = c_{x_i}^1 + p_i^*$		
5.	$\xrightarrow{\{c_{x_i}^*, c_{x_i}^2\}}$	
6.		$d_{x_i} \leftarrow c_{x_i}^* - F_{k_i}(c_{x_i}^2)$
7.		$[d_{x_i}] \leftarrow \text{P.Enc}(d_{x_i})$
8.	$\xleftarrow{[d_{x_i}]}$	
9. $[x_i] = [d_{x_i}][p^*]^{-1}$		
10. $[x_i^2] \leftarrow$ ——$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 重复中 3～9 重加密 c_{x_i}		
$\mathcal{S}_1$ 输出: L_i^s		$\mathcal{S}_2$ 输出: $\perp$

(2) 距离密计算协议 EDC

过程 5. 距离密计算协议 EDC.

服务器 $\mathcal{S}_1(\boldsymbol{L}_i^s, \boldsymbol{L}_i)$		服务器 $\mathcal{S}_2(\boldsymbol{SK}_s)$
1. $r_i \leftarrow_{\mathcal{R}} \{0,1\}^k$		
2. $r_i \leftarrow_{\mathcal{R}} \{0,1\}^k$		
3. $([x_i], [x_i^2]) \leftarrow L_i^s$		
4. $([x_s], [x_s^2]) \leftarrow L_s$		
5. $x_i' \leftarrow [x_i] * [r_i]$		
6. $x_s' \leftarrow [x_s] * [r_s]$		
7.	$\xrightarrow{\{x_i', x_s'\}}$	
8.		$h_i \leftarrow \text{P.Dec}([x_i'])$
9.		$h_i \leftarrow h_i * h_s \bmod n$
10. $[h] \leftarrow \text{P.Enc}(h)$		
11. $s \leftarrow [h] * [x_i]^{n-r_*} \bmod n$		
12. $[D_i] \leftarrow [x_i^2] * [n-2][x_i x_s] * [x_s^2]$		
$\mathcal{S}_1$ 输出: $[D_i]$		$\mathcal{S}_2$ 输出: $\perp$

(3) 安全比较协议 SC

过程 6. 安全比较协议 SC.

服务器 $\mathcal{S}_1(([\boldsymbol{u}_x], [\boldsymbol{D}_x]), ([\boldsymbol{u}_y], [\boldsymbol{D}_y]), \boldsymbol{PK}_s)$		服务器 $\mathcal{S}_2(\boldsymbol{SK}_s)$
1. $[z] \leftarrow [2^l][D_x][D_y]^{-1} \bmod n^2$		
2. $r \leftarrow_{\mathcal{R}} (0,1)^{l+k}$		
3. $r_l \leftarrow r \bmod 2^l$		
4. $[d] \leftarrow [z] \cdot [r] \bmod n^2$		
5.	$\xrightarrow{[d]}$	
6.		$d \leftarrow \text{P.Dec}([d])$
7.		$d_l \leftarrow d \bmod 2^l$
8.		$\|\lambda\| \leftarrow \text{DGK}(\mathcal{S}_1(r_l, PK_S); \mathcal{S}_2(d_l, SK_S)) \rightarrow \perp$
9.		$\|d_l\| \leftarrow \text{GM.Enc}(d_l)$
10.	$\xleftarrow{\|d_l\|}$	
11. $\|r_l\| \leftarrow \text{GM.Enc}(r_l)$		
12. $\|v\| \leftarrow \|d_l\| \|r_l\| \|\lambda\|$		
----------比特重加密过程----------		
13. $r_s \leftarrow_{\mathcal{R}} \{0,1\}$		
14. $\|s_{r_s}\| \leftarrow \|v\| \cdot \|0\|$		
15. $\|s_{1-r_s}\| \leftarrow \|v\| \cdot \|1\|$		
16.	$\xrightarrow{\|s_0\|, \|s_1\|}$	
17.		$s_0 \leftarrow \text{GM.Dec}(\|s_0\|)$
18.		$s_1 \leftarrow \text{GM.Dec}(\|s_1\|)$
19.		$[[s_0]] \leftarrow \text{P.Enc}(s_0)$
20.		$[[s_1]] \leftarrow \text{P.Enc}(s_1)$
21.	$\xleftarrow{[[s_0]], [[s_1]]}$	
----------密文选择过程----------		
22. $[[D_x']] \leftarrow ([1][[s_r]]^{-1})^{[D_x]}[[s_r]]^{[D_y]}$		
23. $r' \leftarrow_{\mathcal{R}} \{0,1\}^{(l+1)}$		
24. $[[\text{P.Dec}[D_x]' + r']] \leftarrow [[D_x]']^{[r']}$		
25. $[[\widetilde{D}_x']] \leftarrow [[\text{P.Dec}[D_x]' + r']]$		
26.	$\xrightarrow{[[\widetilde{D}_x]]}$	
27.		$[\widetilde{D}_x] \leftarrow \text{P.Dec}([[\widetilde{D}_x']])$
28.	$\xleftarrow{[\widetilde{D}_x]}$	
29. $[D_x]' \leftarrow [\widetilde{D}_x][r']^{-1}$		
30. $[D_y]' \leftarrow$ ——$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 重复中 22～29 行计算 $[D_y]'$		
$\mathcal{S}_1$ 输出: L_i^s		$\mathcal{S}_2$ 输出: $\perp$

LI Yu-Xi, Ph. D., lecturer. Her research interests include searchable encryption and secure multi-party computation.

ZHOU Fu-Cai, Ph. D., professor. His research interests include cyber security and trusted computing.

XU Zi-Feng, Ph. D., lecturer. His research interests include verifiable computation.

Background

Mobile social networks (MSNs) are mobile online services that people use to build social relations with others. They have exploded in popularity by offering many services related with user's life. In recent years, the problem of location privacy has received many attentions. To the best of our knowledge, how to design a practical privacy-preserving location-based MSNs scheme is an incredibly challenging task and remains open.

To satisfy the need of user location privacy protection in MSNs, we propose a privacy-preserving K-nearest-neighbor (KNN) search scheme over MSNs. In this scheme, we design a lightweight location encryption algorithm using the pseudo-random function. We also build a secure collaborative search architecture and propose an encrypted KNN search protocol in this architecture, which achieves accurate nearby friends retrieving while preventing the privacy of geo-location and the distance order from revealing them to the servers. We propose a lightweight dynamic friend management mechanism based on public key broadcast encryption, which enables users to grant/remove the location search right without updating the key sets for their original friends, thus realizing constant-time friend identity authentication. Both security and performance analysis demonstrate that our scheme has both a light user workload and a moderate server workload while being secure against user-server collusion.

As future work, our aim is to extend our scheme to efficiently support more functions in MSNs. We also plan to improve its efficiency to reduce the computation cost associated with the query and sort phase. In terms of security, how to achieve forward and backward privacy is treated as one of the vital research directions.

This research is financially supported by the National Natural Science Foundation of China under Grant Nos. 61772127, 61532007 and 61472184.