Private Membership Aggregation

Mohamed Nomeir Sajani Vithana Sennur Ulukus
Department of Electrical and Computer Engineering
University of Maryland, College Park, MD 20742
mnomeir@umd.edu *spallego@umd.edu* *ulukus@umd.edu*

***Abstract*—We consider the problem of private membership aggregation (PMA), in which a user counts the number of times a certain element is stored in a system of independent parties that store arbitrary sets of elements from a universal alphabet. The parties are not allowed to learn which element is being counted by the user. Further, neither the user nor the other parties are allowed to learn the stored elements of each party involved in the process. PMA is a generalization of the recently introduced problem of K private set intersection (K-PSI). The K-PSI problem considers a set of M parties storing arbitrary sets of elements, and a user who wants to determine if a certain element is repeated at least at K parties out of the M parties without learning which party has the required element and which party does not. To solve the general problem of PMA, we dissect it into four categories based on the privacy requirement and the collusions among databases/parties. We map these problems into equivalent private information retrieval (PIR) problems. We propose achievable schemes for each of the four variants of the problem based on the concept of cross-subspace alignment (CSA). The proposed schemes achieve *linear* communication complexity as opposed to the state-of-the-art K-PSI scheme that requires *exponential* complexity even though our PMA problems contain more security and privacy constraints.**

I. Introduction

Multi-party computation (MPC) is used in a wide range of applications such as secure voting, privacy-preserving data analysis, collaborative machine learning, secure social networks, etc [1]. Private set intersection (PSI) is one of the most fundamental multi-party computations [2]–[8]. In PSI, there are multiple parties, each storing a set of elements coming from an alphabet. It is required to find the intersection of the sets of all parties without leaking any information about the remaining elements in each party beyond the intersection. [7] formulates the two-party PSI problem from an information-theoretic point of view, finds the optimal download cost and proposes an optimum achievable scheme. [8] considers the multi-party version of PSI, determines the optimal download cost and gives a capacity-achieving scheme. The schemes in [7], [8] are based on concepts from information-theoretic private information retrieval (PIR).

A new variation of the multi-party PSI problem, called K-PSI, is recently introduced in [9]. In K-PSI, there are M parties storing arbitrary sets of elements out of an alphabet. A user wishes to know if a certain element is repeated K times or not among the M parties. In this problem, the parties do not want to leak any information about their datasets to the other parties or to the user; the user should not learn which parties contain the queried element and which parties do not; and the parties should not learn any information about the element being queried. In [9], a scheme is designed to solve the K-PSI problem with an exponential communication complexity, i.e., $\mathcal{O}(M^K(K-1))$. The same complexity can be achieved with weaker privacy using the existing schemes on PIR-based-PSI [7], [8] if we allow each party to have N databases. This motivates to look at K-PSI and its extensions through the lens of PIR, as it provides the elemental privacy and security requirements of any multi-server system [10]–[20].

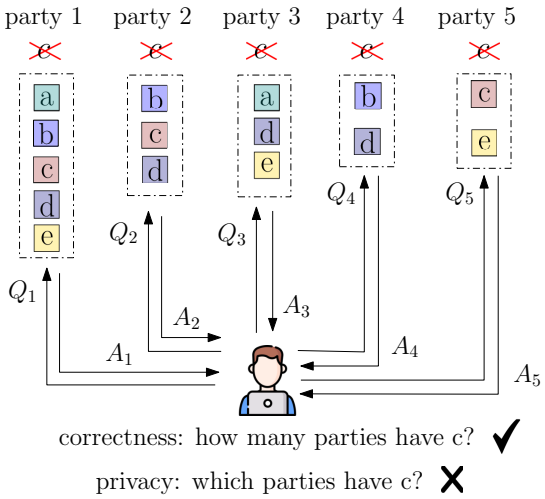

Fig. 1: Private membership aggregation (PMA) system model.

In this paper, we generalize the problem of K-PSI by computing the *exact number* of parties storing a certain element, without revealing the user any information about the elements stored in each party, and without letting the databases know which element is being checked. In addition, we do not allow the user to know which parties have the required element and which do not. We coin this problem as private membership aggregation (PMA); see Fig. 1. This is a *fine-grained* version of K-PSI, as instead of asking if an element is repeated more than K times in the parties, we ask how many times an element is repeated in the parties. The main applications of PMA include multiple identity detection and anomaly detection. For example, in the health insurance industry, companies want to make sure that a person with a certain social security number does not have another account in another company. Another application is to check the validity of certain information by making sure that it exists in some of the other parties as well.

In this work, we consider four different variants of PMA, described by the cases where: 1) different parties are allowed to eavesdrop on the answers from other parties, 2) the user is

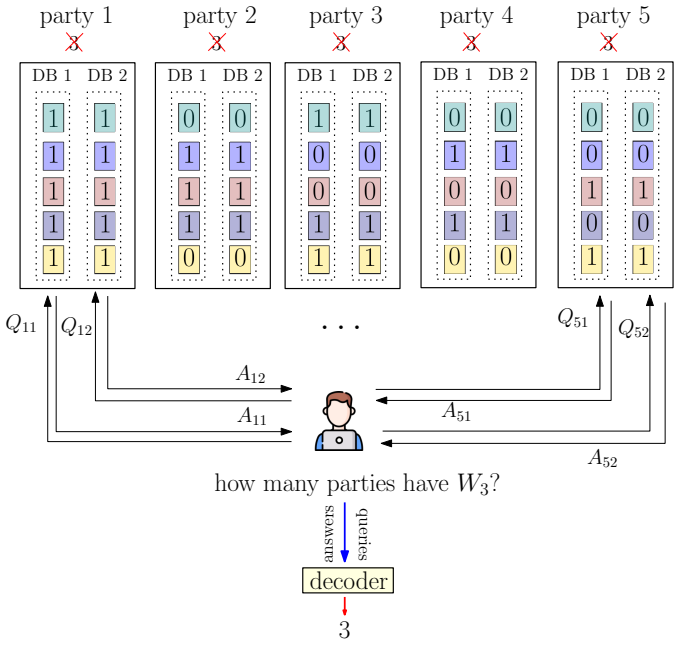

Fig. 2: PMA model: from elements to incidence vectors.

not allowed to learn any information about the elements other than what is being checked, which is coined as symmetric PMA (SPMA), 3) certain subsets of databases within each party are allowed to collude (type I collusion), and 4) certain subsets of parties are allowed to collude (type II collusion). We formulate each problem in the context of PIR, and use concepts from cross subspace alignment (CSA) [16] to solve each problem. We modify the basic CSA scheme to achieve the additional privacy requirement in SPMA by using a masking technique to keep the user from obtaining any information on the elements other than what is checked. We provide schemes that perform each variant of PMA stated above with a *linear* communication complexity, which is a significant improvement compared to the complexity required by the existing state-of-the-art K-PSI schemes, which is *exponential*.

II. Problem Formulation

We consider M parties, each containing N servers. There are E elements in total in the universal set each of which can be mapped to a separate message W_k, $k \in \{1, \dots, E\}$; see incidence vectors in Fig. 2. Each message W_k has a probability p_k to be in the message set of any given party, i.e.,

$$\mathbb{P}(W_k \in \mathcal{P}_i) = p_k, \quad i \in [M],\ k \in [E], \tag{1}$$

where $\mathcal{P}_i$ is the set of messages in the ith party. Each message W_i is generated uniformly at random, independent of other messages, and independent from the shared common randomness between the parties, i.e.,

$$H(W_{[E]}, \mathcal{S}) = EH(W_1) + H(\mathcal{S}), \tag{2}$$

where $\mathcal{S}$ is the shared randomness among the parties.

Each party wishes to keep their message contents $\mathcal{P}_i$ and the indices of the messages available at their datasets hidden from other parties, i.e., for each $i, j \in [M]$, $i \neq j$,

$$I(\mathcal{P}_i; \mathcal{E}_{ij} | \mathcal{P}_j, \mathcal{E}_{1j}, \dots, \mathcal{E}_{i-1,j}, \mathcal{E}_{i+1,j}, \dots, \mathcal{E}_{Mj}) = 0, \tag{3}$$
$$I(\mathcal{U}_i; \mathcal{E}_{ij} | \mathcal{P}_j, \mathcal{E}_{1j}, \dots, \mathcal{E}_{i-1,j}, \mathcal{E}_{i+1,j}, \dots, \mathcal{E}_{M,j}) = 0, \tag{4}$$

where $\mathcal{U}_i = \bigcup_{j=1}^{E} \{\mathbb{1}(W_j \in \mathcal{P}_i)\}$ and $\mathcal{E}_{ij}$ is all possible communications between the ith and the jth parties.

The user chooses an index $\theta \in [E]$ uniformly at random and wishes to compute how many parties store W_θ by sending a query Q_{ij}^θ to the jth database in the ith party, which satisfies,

$$I(\theta; Q_{ij}^\theta | \mathcal{P}_i) = 0, \quad i \in [M],\ j \in [N]. \tag{5}$$

After receiving the queries, each database replies truthfully with an answer string A_{ij}^θ which is a deterministic function of the received query Q_{ij}^θ, the messages available at each party $\mathcal{P}_i$, and shared randomness between parties $\mathcal{S}$, i.e.,

$$H(A_{ij}^\theta | Q_{ij}^\theta, \mathcal{P}_i, \mathcal{S}) = 0, \quad i \in [M],\ j \in [N]. \tag{6}$$

If the ith party is eavesdropping on $Y_{i\ell}$ links in the ℓth party, then, party i should not be able to obtain any information on the message index being checked, contents of the ℓth party, or the required answer by the user, $\kappa^\theta = \sum_{i \in [M]} \mathbb{1}(W_\theta \in \mathcal{P}_i)$, i.e., for $i, \ell \in [M]$, $i \neq \ell$, $\kappa^\theta \in \{0, \dots, M\}$, $\theta \in [E]$,

$$I(\theta; A_{\ell \mathcal{Y}_\ell}^\theta, Q_{\ell \mathcal{Y}_\ell}^\theta | \mathcal{S}, \mathcal{P}_i) = 0, \tag{7}$$
$$I(\mathcal{U}_\ell; A_{\ell \mathcal{Y}_\ell}^\theta, Q_{\ell \mathcal{Y}_\ell}^\theta | \mathcal{S}, \mathcal{P}_i) = 0, \tag{8}$$
$$I(\kappa^\theta; A_{\ell \mathcal{Y}_\ell}^\theta, Q_{\ell \mathcal{Y}_\ell}^\theta | \mathcal{S}, \mathcal{P}_i) = 0, \tag{9}$$

where $\mathcal{Y}_\ell$ is the set of databases in party ℓ whose communication links are eavesdropped on by party i, such that $|\mathcal{Y}_\ell| \leq \max_i(Y_{i\ell})$. Given the answer strings from all parties, the user can apply a decoding scheme that generates the required answer with no error, i.e.,

$$\hat{\kappa}^\theta = g(A_{ij}^\theta, Q_{ij}^\theta,\ i \in [M],\ j \in [N]), \quad \theta \in [E], \tag{10}$$

where g is the decoding scheme, and

$$\mathbb{P}(\hat{\kappa}^\theta = \kappa^\theta) = 1. \tag{11}$$

In addition, PMA requires that the user should not be able to infer any information about the parties containing the required message, except by random guessing, i.e., no information about the locations of the required message W_θ can be extracted from the answer strings, i.e., for $\theta \in [E]$, $\kappa \in \{0, \dots, M\}$, $i \in [M]$,

$$\mathbb{P}(W_\theta \in \mathcal{P}_i | A^\theta, \kappa^\theta = \kappa) = \frac{\kappa}{M}, \tag{12}$$

where $A^\theta = \bigcup_{i,j} \{A_{ij}^\theta\}$. Moreover, for any $\{i_1, \dots, i_n\} \subset [M]$ such that $n \leq \kappa$,

$$\mathbb{P}(W_\theta \in \{\mathcal{P}_{i_1}, \dots, \mathcal{P}_{i_n}\} | A^\theta, \kappa^\theta = \kappa) = \frac{\binom{\kappa}{n}}{\binom{M}{n}}. \tag{13}$$

Finally, in SPMA, it is required that no information about the messages other than the one being checked is allowed to leak to the user, i.e., for $\theta \in [E]$,

$$I(\mathcal{W}_{\theta^C}; A^\theta | Q^\theta, \kappa^\theta) = 0, \tag{14}$$

$$I(\Gamma^\theta; A^\theta | Q^\theta, \kappa^\theta) = 0, \tag{15}$$

where $\mathcal{W}_{\theta^C} = \{W_i : i \in [E], i \neq \theta\}$, $Q^\theta = \bigcup_{i,j}\{Q_{ij}^\theta\}$, and $\Gamma^\theta = \bigcup_{i=1}^{M}\bigcup_{\substack{j=1\\ j\neq\theta}}^{E}\{\mathbb{1}(W_j \in \mathcal{P}_i)\}$.

A scheme that satisfies (3)-(13) is called a PMA scheme, and a PMA scheme that satisfies (14), (15) is called an SPMA scheme. The download cost D of any of these schemes is

$$D = \mathbb{E}\Big[\sum_i \sum_j H(A_{ij}^\theta)\Big], \tag{16}$$

where the expectation is taken over θ.

We further separate the problem into two types, namely, 1) collusions within the databases in each party, i.e., the parties do not collude but the databases within each party are allowed to collude (type I collusion), and 2) collusions among the parties, i.e., the databases within each party are colluding with other databases from other parties (type II collusion).

III. Main Results

Theorem 1 *Consider a PMA system with type I collusions consisting of M parties, each of which has N databases with any T of them colluding. Each party is allowed to eavesdrop on Y links of the other parties. The optimal download cost of this case $D^*_{\text{PMA-I}}$ must satisfy,*

$$D^*_{\text{PMA-I}} \leq M(\max(T, Y) + 1), \tag{17}$$

with $N \geq \max(T, Y) + 1$.

Theorem 2 *For the same setting as in Theorem 1 with the additional condition of symmetric privacy, the optimal download cost of SPMA $D^*_{\text{SPMA-I}}$ must satisfy,*

$$D^*_{\text{SPMA-I}} \leq M(\max(T, Y) + 1), \tag{18}$$

with $N \geq \max(T, Y) + 1$.

Theorem 3 *Consider a PMA system with type II collusions consisting of M parties, each of which has N databases. All databases in any T out of the M parties can collude, and the ith party is able to listen to Y_i links of any other party. The optimal download cost of non-symmetric and symmetric variants of this case $D^*_{\text{PMA-II}}$ and $D^*_{\text{SPMA-II}}$ must satisfy,*

$$D^*_{\text{PMA-II}} \leq N + \max(TN, Y_1, \ldots, Y_M) + 1, \tag{19}$$

$$D^*_{\text{SPMA-II}} \leq N + \max(TN, Y_1, \ldots, Y_M) + 1, \tag{20}$$

with $MN \geq N + \max(TN, Y_1, \ldots, Y_M) + 1$.

Remark 1 *The bounds on the download costs in Theorems 1-3 do not depend on the number of messages in the system E.*

Remark 2 *The achievable schemes for Theorems 1 and 2 result in the same download cost for PMA type I and SPMA type I. This is because the modified CSA scheme that achieves symmetric privacy does not require any additional downloads.*

Remark 3 *The related work on K-PSI [9] achieves exponential communication complexity, which is significantly reduced in this work, as the download costs in Theorems 1, 2, 3 are all linear in the number of parties M, the number of databases per party N, and the number of colluding parties T.*

IV. Proposed Schemes

The schemes proposed for both PMA and SPMA are based on CSA coding [16], with further modifications to achieve symmetric privacy and security. In both problems, each party generates a private incidence vector P_i, $i \in [M]$. For the example shown in Fig. 1 and Fig. 2, where $E = 5$ and the alphabet is $\{a, b, c, d, e\}$, equivalently, $\{W_1, W_2, W_3, W_4, W_5\}$. Since $\mathcal{P}_1 = \{W_1, W_2, W_3, W_4, W_5\}$, the incidence vector of party 1 is $P_1 = [1, 1, 1, 1, 1]^t$, since $\mathcal{P}_2 = \{W_2, W_3, W_4\}$, the incidence vector of party 2 is $P_2 = [0, 1, 1, 1, 0]^t$, and so on.

Remark 4 *Using the incidence vector to reply to the user's queries instead of the messages explicitly satisfies* (3)*.*

A. Proposed Scheme for PMA Type I

In PMA type I, there are M parties, with N databases each, out of which any T can be colluding. Each party is allowed to eavesdrop on Y communication links of any other party. Let the number of databases per party be $N = \max(T, Y) + 1$. The vectors P_i, $i \in [M]$ are replicated in all the databases of each party. The user, who wishes to know how many times W_θ is repeated among the M parties sends queries Q_{ij}^θ,

$$Q_{ij}^\theta = e_\theta + \sum_{\ell=1}^{\mu} (1 + \alpha_j)^\ell Z_{i\ell}, \tag{21}$$

where $\mu = \max(T, Y)$, e_θ is a vector of length E with 1 at the θth index and zeros otherwise, Z_{i1}s are independent noise vectors, with the same length, chosen uniformly at random, and α_js are globally known distinct constants. After receiving the queries, each database responds with an answer A_{ij}^θ,

$$A_{ij}^\theta = P_i^t Q_{ij}^\theta + S_{ij}, \tag{22}$$

where $S_i = [S_{i1}, \ldots, S_{iN}]^t$ is the masking vector corresponding to the ith party, unknown to the user. The masking vectors, $S_1, \ldots, S_M$, are chosen, independent of the incidence vectors, such that $\sum_{i=1}^{M} S_i = 0_N$, where 0_N is the zero vector of size $N \times 1$. The answers from the ith party are given by,

$$A_i^\theta = \big[A_{i1}, A_{i2}, \ldots, A_{iN}\big]^t = \Upsilon_N \Lambda_i + S_i, \tag{23}$$

where

$$\Upsilon_N = \begin{bmatrix} 1 & 1+\alpha_1 & (1+\alpha_1)^2 & \ldots & (1+\alpha_1)^{N-1} \\ 1 & 1+\alpha_2 & (1+\alpha_2)^2 & \ldots & (1+\alpha_2)^{N-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & 1+\alpha_N & (1+\alpha_N)^2 & \ldots & (1+\alpha_N)^{N-1} \end{bmatrix}, \tag{24}$$

and $\Lambda_i = \big[\mathbb{1}(W_\theta \in \mathcal{P}_i), I_1, I_2, \ldots, I_{N-1}\big]^t$ with $I_1, I_2, \ldots, I_{N-1}$ being interference symbols. To find the required

answer, the user adds all the received answers, i.e.,

$$\sum_{i=1}^{M} A_i^{\theta} = \Upsilon_N \sum_{i=1}^{M} \Lambda_i = \Upsilon \begin{bmatrix} \sum_{i=1}^{M} \mathbb{1}(W_\theta \in \mathcal{P}_i) \\ \tilde{I}_1 \\ \tilde{I}_2 \\ \vdots \\ \tilde{I}_{N-1} \end{bmatrix}, \quad (25)$$

and use the invertibility of Υ to obtain $\sum_{i=1}^{M} \mathbb{1}(W_\theta \in \mathcal{P}_i)$.

Remark 5 *The number of databases per party required for the proposed scheme is $N \geq \max(Y, T) + 1$ and the optimal number of databases per party that satisfies the minimum download cost, with a fixed T and Y, for this scheme is $N = \max(T, Y) + 1$.*

Remark 6 *In the proposed scheme given in this section, there is no exchange of information between parties except for the masking, which is independent of the messages and indices. Thus,* (3)*,* (4) *are both satisfied.*

Remark 7 *This scheme does not satisfy the symmetric privacy constraint in* (15) *since the interference symbols may carry information about $\sum_i \mathbb{1}(W_{\theta^C} \in \mathcal{P}_i)$. A modified version presented in Section IV-B satisfies symmetric privacy.*

Remark 8 *The total communication complexity of the system, considering the sum of the user's upload cost, download cost, and the cost of sharing randomness between parties is given as $(M-1)N + EMN + MN$, where it is assumed that the masking vectors are generated by a single party, which are then sent to the rest of the parties.*

B. Proposed Scheme for SPMA Type I

In this section, we assume that the number of databases in each party is $N = \max(T, Y) + 1$, similar to the previous section, as the modification proposed for CSA to achieve symmetric privacy does not require additional databases. In contrast to PMA type I, SPMA type I hides any information about the availability of messages other than the one being checked from the user. Intuitively, if the parties can utilize random noise in the scheme such that the noise hides the contents of the interference symbols in (25), the scheme becomes private in both directions. The core difference between this scheme and the scheme in Section IV-A is that the databases within each party in this scheme share common randomness Z' which is generated independently from the messages, the incidence vector, and the masking variables. As in the previous section, the ith party stores its incidence vector P_i in a replicated manner in all N databases, i.e.,

$$P_{ij} = P_i, \quad i \in [M],\ j \in [N]. \quad (26)$$

The user sends the same queries as in (21), to which the databases send the corresponding answers given by,

$$A_{ij}^{\theta} = P_{ij}^t Q_{ij}^{\theta} + \sum_{\ell=1}^{N-1} (1+\alpha_j)^{\ell} Z'_{i\ell} + S_{ij}, \quad (27)$$

where $Z'_{i\ell}$s are random noise variables initialized and shared by the N databases in each party i. Thus, the user obtains,

$$A_i^{\theta} = \Upsilon_N \Lambda_i + S_i, \quad (28)$$

where Υ is the same as in (24), S_i is the masking vector, and Λ_i is given by,

$$\Lambda_i = \begin{bmatrix} \mathbb{1}(W_\theta \in \mathcal{P}_i) \\ I_1 + Z'_{i1} \\ I_2 + Z'_{i2} \\ \vdots \\ I_{N-1} + Z'_{i,N-1} \end{bmatrix}, \quad (29)$$

where $I_1, I_2, \ldots I_{N-1}$ are interference symbols. By applying the same decoding scheme as in the previous section, the user retrieves the required answer.

Remark 9 *The total communication complexity of this scheme is given by $(M-1)N + EMN + N - 1 + MN$, where we assume that one party generates the masking vectors and sends them to the rest of the parties.*

Remark 10 *If, in addition, we assume that T_2 parties are communicating, i.e., sharing their datasets to figure out the datasets of the remaining $M - T_2$ parties, the schemes presented in the previous sections still maintain the same download cost since the schemes do not require the parties to share their datasets, nor the incidence vectors.*

C. Proposed Scheme for SPMA Type II

In this case, there are M parties, each with N databases, and all databases in $T < M$ parties are allowed to collude. The main issue here is that the efficient PIR schemes cannot be applied separately to each party, as all the databases in each party collude with each other. This requires any information exchange among the parties to be secure against any N communicating databases, which motivates the mapping of this problem to an XSTPIR problem [16] with the number of colluding databases, i.e., databases that share information of the users, $T' = NT$, and the number of communicating databases, i.e., databases that share their contents, $X = N$. We adopt a variant of the CSA scheme to solve this problem as in the previous sections. For this scheme, N is chosen such that $MN = N + \max(TN, Y_1, \ldots, Y_M) + 1$, and the proposed approach is defined in the following steps:

Step 1: Initialization and Distribution: Each party with its message set, $\mathcal{P}_i$, has its corresponding incidence vector P_i that needs to be secure against any N communicating databases. Thus, it is encoded as,

$$\tilde{P}_{ij} = P_i + \sum_{\ell=1}^{N} (1+\alpha_j)^{\ell} X_{i\ell},\ i \in [M],\ j \in [MN], \quad (30)$$

where $X_{i\ell}$s are independent random noise vectors. The ith party sends $\tilde{P}_{ij}$ to the jth database. After receiving $\tilde{P}_{ij},\ i \in$

$[M]$, the jth database adds all the received vectors as,

$$\tilde{P}_j = \sum_{i=1}^{M} P_i + (1+\alpha_j)\tilde{X}_1 + \ldots + (1+\alpha_j)^N \tilde{X}_N, \quad (31)$$

for $j \in [MN]$, where $\tilde{X}_n = \sum_{i=1}^{M} X_{in},\ n \in [N]$.

Step 2: Queries and Answers Structure: The user who wants to know how many times W_θ is repeated in the M parties, sends the following query to the nth database,

$$Q_n^\theta = e_\theta + \sum_{\ell=1}^{\mu} (1+\alpha_n)^\ell Z_\ell,\ n \in [MN], \quad (32)$$

where $Z_k,\ k \in [\mu]$ are uniform independent noise vectors, and $\mu = \max(NT, Y_1, \ldots, Y_M)$. The parties agree on uniform random noise variables $Z'_1, Z'_2, \ldots, Z'_{MN-1}$, and generate the answers to achieve symmetric privacy as,

$$A_n^\theta = \tilde{P}_n^t Q_n^\theta + \sum_{i=1}^{MN-1} (1+\alpha_n)^i Z'_i, \quad n \in [MN]. \quad (33)$$

Decoding Structure: After retrieving all the answers, the user has the following answer vector

$$A^\theta = \begin{bmatrix} A_1^\theta \\ A_2^\theta \\ \vdots \\ A_{MN}^\theta \end{bmatrix} = \Upsilon_{MN} \begin{bmatrix} \sum_{i=1}^{M} \mathbb{1}(W_\theta \in \mathcal{P}_i) \\ I_1 + Z'_1 \\ \vdots \\ I_{MN-1} + Z'_{MN-1} \end{bmatrix}, \quad (34)$$

where $I_1, \ldots, I_{MN-1}$ are the interference symbols. The user multiplies the answer vector by Υ_{MN}^{-1} to obtain the required information. The download cost in the proposed scheme is MN with $MN \geq N + TN + 1$, which concludes the proof of the upper bound in Theorem 3.

Remark 11 *The total communication cost in this scheme is equal to* $(E+1)(N+TN+1) + N + NT$.

Remark 12 *If* $MN > N + \max(TN, Y_1, \ldots, Y_M) + 1$*, we can drop the extra databases. Interestingly, this shows that cooperation between parties, even though their datasets are secure from each other, can save some databases.*

Remark 13 *In this scheme, if there are* T_2 *communicating parties, then the optimal download cost is upper bounded by* $T_2 N + \max(TN, Y_1, \ldots, Y_M) + 1$.

V. Conclusions

In this paper, we introduced PMA which is a generalization and refinement of K-PSI. In PMA, the user wishes to know how many times a certain message appears in all parties. We consider different cases of the problem, based on the privacy requirements (user-privacy and symmetric privacy) and database collusions. We proposed achievable schemes for all cases considered, focusing on the behavior of the communication complexity as a function of system parameters. Compared to the previous work in K-PSI that achieves *exponential* complexity, the schemes proposed here achieve *linear* complexity with enhanced privacy and security guarantees.

VI. Appendix: Proofs

In this section, we prove important lemmas that collectively prove the security and privacy requirements for the developed schemes. More precisely, Lemmas 2, 4, and 5 prove Theorem 1; Lemmas 1, 2, 4, and 5 prove Theorem 2; and Lemmas 1, 3, 4, and 6 prove Theorem 3, in terms of the claims made on the levels of privacy and security achieved.

Lemma 1 *The schemes proposed for SPMA type I and SPMA type II provide symmetric privacy.*

Proof: For the scheme proposed for the SPMA type I problem, since every party has its own independent random variables, $Z'_1(i), \ldots, Z'_{N-1}(i),\ i \in [M]$, it suffices to consider each party independently. Let $\Gamma_{\theta^C}(i) = \bigcup_{\substack{k=1 \\ k \neq \theta}}^{E} \{\mathbb{1}(W_k \in \mathcal{P}_i)\}$, thus we need to show that $I(\Gamma_{\theta^C}(i); A^\theta_{i[N]}(i) | Q^\theta_{i[N]}(i), \kappa^\theta) = 0$. The answers received by the user from the ith party are contaminated with noise terms independent of the messages along the interference terms, which is simply a random noise symbol Z' unknown to the user. Thus,

$$\begin{aligned} &I(\Gamma_{\theta^C}(i); A^\theta_{i[N]} | \kappa^\theta, Q^\theta_{[MN]}, \theta) \\ &= H(\Gamma_{\theta^C}(i) | \kappa^\theta, Q^\theta_{[MN]}, \theta) \\ &\quad - H(\Gamma_{\theta^C}(i) | A^{[\theta]}_{i[N]}, \kappa^\theta, Q^\theta_{[MN]}, \theta) \quad (35) \\ &\leq H(\Gamma_{\theta^C}(i)) - H(\Gamma_{\theta^C}(i) | \kappa^\theta, Z'_{[N-1]}, Q^\theta_{[MN]}, \theta) \quad (36) \\ &= 0. \quad (37) \end{aligned}$$

For the scheme presented for the SPMA type II, the same approach is used, however, the interference terms in the answers are contaminated with independent random noise symbols collectively, thus we use $Z'_1, \ldots, Z'_{MN-1}$ in the proof and consider the answers collectively. ■

Lemma 2 *The masking used in PMA type I and SPMA type I schemes guarantees blind estimation requirements* (12)-(13).

Proof: Assume that the random vector $\Omega(W_\theta)$ represents the information that the user requires about W_θ and any possible side information about the presence of the same message in a subset of parties $\mathcal{M}$ such that $|\mathcal{M}| \leq M-1$. Let $A^\theta_{\mathcal{M}}$ be the set of answers received from those parties and Z be the noise terms used to ensure user privacy, then

$$\begin{aligned} I(\Omega(W_\theta); A^\theta_{\mathcal{M}} | \theta, Z) &= I(\Omega(W_\theta); A^\theta_{i_1}, \ldots, A^\theta_{i_{M-1}}) \quad (38) \\ &= I(\Omega(W_\theta); S_{i_1}, \ldots, S_{i_{M-1}}) = 0 \quad (39) \end{aligned}$$

where the last equality is due to the independence between the incidence vectors and the masking vectors.

Now, since the answer vectors from each party, $A_1, \ldots, A_M$, are aligned, the answers of all M parties cannot give any information about any subset of parties with cardinality less than M. ■

Lemma 3 *The scheme proposed for SPMA type II is secure against any* N *communicating databases.*

Proof: Let $P := \tilde{P}_j$ given in (30) for any j, and $P(\ell)$ is the ℓth element of P. Define the vector of any N observations for the ℓth element of P as U_ℓ. Then, U_ℓ can be written as

$$U_\ell = [U_\ell(1), \dots, U_\ell(N)]^t \tag{40}$$

$$= P(\ell) \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} + \begin{bmatrix} 1+\alpha_{i_1} & \dots & (1+\alpha_{i_1})^N \\ 1+\alpha_{i_2} & \dots & (1+\alpha_{i_2})^N \\ \vdots & & \vdots \\ 1+\alpha_{i_N} & \dots & (1+\alpha_{i_N})^N \end{bmatrix} \begin{bmatrix} X_1(\ell) \\ \vdots \\ X_N(\ell) \end{bmatrix} \tag{41}$$

$$= P(\ell)\mathbf{1} + \text{diag}(\mathbf{1} + \alpha_{[i_1:i_N]})\ \Upsilon_N [X_1(\ell), \dots, X_N(\ell)]^t. \tag{42}$$

In (42), the matrices $\text{diag}(\mathbf{1} + \alpha_{[i_1:i_N]})$ and Υ_N are invertible, which makes their product invertible as well. Then, the following inequalities hold

$$I(P(\ell); U_\ell)$$

$$= I(P(\ell); P(\ell)\mathbf{1} + \text{diag}(1 + \alpha_{[i_1:i_N]})\ \Upsilon_N X(\ell)) \tag{43}$$

$$= I(P(\ell); P(\ell)(\text{diag}(1 + \alpha_{[i_1:i_N]})\ \Upsilon_N)^{-1}\mathbf{1} + X(\ell)) \tag{44}$$

$$= I(P(\ell); X(\ell)) = 0, \tag{45}$$

where $X(\ell) = [X_1(\ell), \dots, X_N(\ell)]^t$. ■

Lemma 4 *The query structure defined in the schemes for PMA type I, SPMA type I and SPMA type II are secure against any T colluding databases.*

Proof: Let θ be the required user index, then for any set of colluding servers $\mathcal{T}$ such that $|\mathcal{T}| \leq T$, the collective observations $Q_\mathcal{T}$ can be written as

$$Q_\mathcal{T} = [Q^t_{i_1}, \dots, Q^t_{i_T}] = E_\theta + B \left[Z^t_1, \dots, Z^t_T\right]^t, \tag{46}$$

where $E_\theta = [e^t_\theta, \dots, e^t_\theta]^t$. Note that $H(\theta|E_\theta) = 0$, and the matrix $B = \text{diag}(\mathbf{1} + \alpha_{[i_1:i_T]})\Upsilon_T$ is invertible as proven in Lemma 3. Now, to ensure privacy given the queries $Q_\mathcal{T}$, we proceed as,

$$I(\theta; Q_\mathcal{T}) = I(\theta; E_\theta + BZ) \tag{47}$$

$$= I(\theta; B^{-1}E_\theta + Z) = I(\theta; Z) = 0, \tag{48}$$

which concludes the proof. ■

Lemma 5 *The schemes proposed for PMA type I and SPMA type I are secure against an eavesdropper who has access to any $\mathcal{Y}$, where $|\mathcal{Y}| \leq Y$ answers from the other parties.*

Proof: We note that the incidence vectors for each party are independent from each others, i.e., $I(P_i; P_\ell) = 0,\ i \neq \ell$. Thus, $I(A_{i\mathcal{Y}}; P_i|P_\ell) = I(A_{i\mathcal{Y}}; P_i)$. Now, using the same proof as in Lemma 4 with $\mathcal{Y}$ instead of $\mathcal{T}$ and P_i instead of θ, the proof follows. ■

Lemma 6 *The scheme proposed for SPMA type II protects the contents of the answers from any party that eavesdrops on $\mathcal{Y}_i$ links of other parties.*

Proof: First, note that $I(S_n; \mathcal{P}_i) = 0$, where S_n is the storage in the jth database. This means that the dataset is secure against any party. Let $\mathcal{Y} = \max(\mathcal{Y}_i)$. Thus,

$$I(\sum_{i=1}^{M} P_i; A_\mathcal{Y}|Q_\mathcal{Y}) = I(\sum_{i=1}^{M} P_i; A_\mathcal{Y}) \tag{49}$$

$$= I(e^t_\theta \sum_{i=1}^{M} P_i; A_\mathcal{Y}) + I(\Gamma^\theta; A_\mathcal{Y}) = 0 \tag{50}$$

where $A_\mathcal{Y}$ are the answers from any $\mathcal{Y}$ databases. The first term on the right hand side of (50) is equal to zero by the same method as in the proof of Lemma 4, where we replace the queries with the answers and θ with $e^t_\theta \sum_{i=1}^{M} P_i$. The second term is equal to zero as a direct consequence of Lemma 1. ■

REFERENCES

[1] S. Ulukus, S. Avestimehr, M. Gastpar, S. A. Jafar, R. Tandon, and C. Tian. Private retrieval, computing and learning: Recent progress and future challenges. *IEEE JSAC*, 40(3):729–748, March 2022.

[2] B. Pinkas, T. Schneider, and M. Zohner. Faster private set intersection based on OT extension. In *USENIX Security Symposium*, August 2014.

[3] B. Pinkas, T. Schneider, G. Segev, and M. Zohner. Phasing: Private set intersection using permutation-based hashing. In *USENIX Security Symposium*, August 2015.

[4] P. Rindal and M. Rosulek. Improved private set intersection against malicious adversaries. In *Advances in Cryptology, EUROCRYPT*, April 2017.

[5] V. Kolesnikov, N. Matania, B. Pinkas, M. Rosulek, and N. Trieu. Practical multi-party private set intersection from symmetric-key techniques. In *ACM SIGSAC Conference on Computer and Communications Security*, October 2017.

[6] B. Pinkas, M. Rosulek, N. Trieu, and A. Yanai. Spot-light: lightweight private set intersection from sparse ot extension. In *Advances in Cryptology, CRYPTO*, August 2019.

[7] Z. Wang, K. Banawan, and S. Ulukus. Private set intersection: A multi-message symmetric private information retrieval perspective. *IEEE Trans. Info. Theory*, 68(3):2001–2019, November 2022.

[8] Z. Wang, K. Banawan, and S. Ulukus. Multi-party private set intersection: An information-theoretic approach. *IEEE Jour. Selected Areas in Info. Theory*, 2(1):366–379, February 2021.

[9] A. Elkordy, Y. Ezzeldin, and S. Avestimehr. Federated K-private set intersection. In *ACM CIKM*, October 2022.

[10] H. Sun and S. A. Jafar. The capacity of private information retrieval. *IEEE Trans. Info. Theory*, 63(7):4075–4088, July 2017.

[11] H. Sun and S. A. Jafar. The capacity of symmetric private information retrieval. *IEEE Trans. Info. Theory*, 65(1):322–329, June 2018.

[12] S. Vithana, Z. Wang, and S. Ulukus. Private information retrieval and its applications: An introduction, open problems, future directions. *arXiv preprint arXiv:2304.14397*, 2023.

[13] A. Heidarzadeh, B. Garcia, S. Kadhe, S. Rouayheb, and A. Sprintson. On the capacity of single-server multi-message private information retrieval with side information. In *Allerton Conference*, October 2018.

[14] K. Banawan and S. Ulukus. The capacity of private information retrieval from coded databases. *IEEE Trans. Info. Theory*, 64(3):1945–1956, January 2018.

[15] K. Banawan and S. Ulukus. The capacity of private information retrieval from Byzantine and colluding databases. *IEEE Trans. Info. Theory*, 65(2):1206–1219, September 2018.

[16] Z. Jia, H. Sun, and S. A. Jafar. Cross subspace alignment and the asymptotic capacity of X-secure T-private information retrieval. *IEEE Trans. Info. Theory*, 65(9):5783–5798, May 2019.

[17] S. Vithana and S. Ulukus. Private federated submodel learning with sparsification. In *IEEE ITW*, November 2022.

[18] S. Vithana and S. Ulukus. Efficient private federated submodel learning. In *IEEE ICC*, May 2022.

[19] S. Vithana and S. Ulukus. Private read update write (PRUW) with storage constrained databases. In *IEEE ISIT*, June 2022.

[20] S. Vithana and S. Ulukus. Rate-privacy-storage tradeoff in federated learning with top r sparsification. In *IEEE ICC*, May 2023.