# AUXILIARY INFORMATION IN DATA PRIVACY AND ATTACKS FOR SECURE MULTIPARTY PROTOCOLS

**S.Rajesh*** 

**Baskar****

**Abstract-**

Secure multiparty protocols have been proposed to enable non-colluding parties to cooperate without a trusted server. Even though such protocols prevent information disclosure other than the objective function, they are quite costly in computation and communication. The high overhead motivates parties to estimate the utility that can be achieved as a result of the protocol beforehand. Here propose a look-ahead approach, specifically for secure multiparty protocols to achieve distributed k-anonymity, which helps parties to decide if the utility benefit from the protocol is within an acceptable range before initiating the protocol. The look-ahead operation is highly localized and its accuracy depends on the amount of information the parties are willing to share. Experimental results show the effectiveness of the proposed methods

* AP /CSE, Dr Pauls Engineering college

** Hod/CSE SKP Engineering college

## 1 INTRODUCTION

SECURE multiparty computation (SMC) protocols are one of the first techniques used in privacy preserving data mining in distributed environments [17]. The idea behind these protocols is based on the theoretical proof that two or more parties, both having their own private data, can collaborate to calculate any function on the union of their data [8]. While doing so, the protocol does not reveal anything other than the output of the function or anything that can be computed from it in polynomial time. Moreover, the protocol does not require a trusted third party. While these properties are promising for privacy preserving applications, SMC may be prohibitively expensive. In fact, many SMC protocols for privacy preserving data mining suffer from high computation and communication costs. Furthermore, those that are closest to be practical are designed for the semihonest model, which assumes that parties will not deviate from the protocol. Theoretically, it is possible to convert protocols in the semihonest model into protocols in the malicious model. However, the resulting protocols are even more costly.

The high overhead of SMC protocols raises the question of whether the information gain (increase in utility) after the protocol execution is worth the cost. This is a valid question or protocols working on horizontally or vertically partitioned data (but especially crucial for horizontally partitioned data where an objective function is well defined on the partitions ). More specifically, for private table $T_\sigma$ of party $P_\sigma$ and an objective function O; initiating the SMC protocol is meaningful only if the information gain from O; $I_\sigma = I(O(T_u)) - I(O(T_\sigma))$ where $T_\sigma$ is the union of all private tables, is more than a user defined threshold. In most cases, it is not possible to calculate $I_\sigma$ without executing the protocol. However, it may be possible to estimate it by knowing some prior (and nonsensitive) information about $T_u$.

To the best of our knowledge, this is the first work that looks ahead of an SMC protocol and gives an estimate for $I_\sigma$ . We state that an ideal look ahead satisfies the following:

1. The methodology is highly localized in computation, it is fast and requires little communication cost (at least asymptotically better than the SMC protocol).

2. The methodology relies on nonsensitive data, or better, data that would be implied from the output of the objective function.

We state that an ideal look ahead will benefit the parties in answering the following:

1. How likely is it that the information gain Iσ will be within an acceptable range?

2. Since efficiency of SMC depends heavily on data, what size of private data would be enough to get an acceptable $I_\sigma$ ?

Our focus is the SMC protocol for distributed k-anonymity previously studied in [31], [11], [10]. k-Anonymity is a well-known privacy preservation technique proposed in [27], [24] to prevent linking attacks on shared databases. Linking attacks are performed by adversaries who know some attributes (quasi-identifier attributes) of an individual to identify him/her in the data set. A database is said to be k- anonymous if every tuple projected over the quasi-identifier attributes appears at least k times in the database. k-Anonymization is the process of enforcing the k-anonymity property on a given database by using generalization and suppression of values. Works in [11], [10] assume that data are vertically partitioned between two parties and they share a common key making a join possible. Authors in [11] propose a semihonest SMC solution to create a k-anonymization of the join without revealing anything else (The protocol takes around 2 weeks time to execute for k = 100 and 30,162 tuples.). Work in [31] assumes horizontally partitioned data. The motivation behind k-anonymity or distributed k-anonymity as a privacy notion has been studied extensively in the literature. Many extensions to k anonymity have been proposed that address various weaknesses of the notion against different types of adversaries [22], [16], [18], [20], [29], [30], [19], [3]. ℓ-Diversity [18] is one such extension that enforces constraints on the distribution of the sensitive values. We first focus on the k-anonymization process and show later how the proposed methodology can be extended for ℓ-diversity. Our contribution can be

summarized as follows:

1. We introduce and formally define secure look-ahead protocols.

2. We design a fast look ahead of k-anonymization of horizontally partitioned data. The look ahead returns an upper bound on the probability that k-anonymity will be achieved at a certain utility. Utility is quantified by commonly used metrics from

the anonymization literature.

3. Look ahead exploits prior information such as total data size, attribute distributions, or attribute correlations, all of which require simple SMC operations. Look ahead returns tighter bounds as the security constraints allow more prior information.

4. We show how look ahead can be extended to enforce diversity on sensitive attributes as in [16], [18].

5. To the best of our knowledge, this work is the first attempt in making a probabilistic analysis of k-anonymity given only statistics on the private data. More specifically, given only statistics on the private data set, we show how to calculate the μ-probability;the the probability that a mapping of values to generalization will make a private data set k-anonymous.

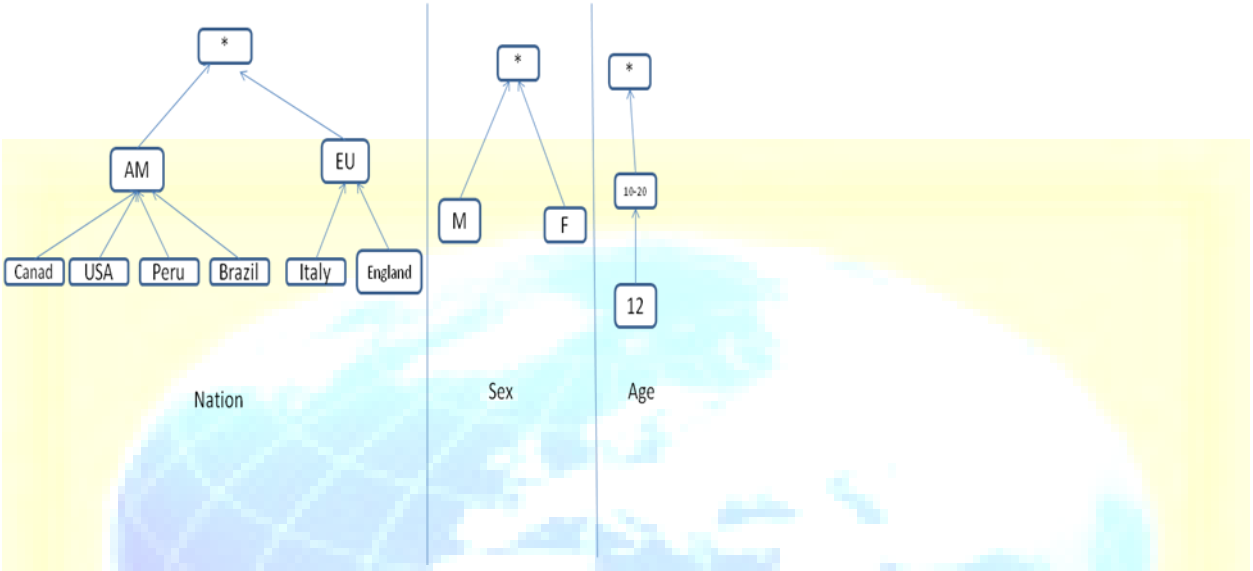


Fig 1: DGH structures.

## 2 BACKGROUND

### 2.1 k-Anonymity and Table Generalizations

Given a data set (table) T, T[c][r] refers to the value of column c, row r of T. T[C] refers to the projection of set of columns C on T and T[.][r] refers to selection of row r on T. Although there are many ways to generalize a given data value, in this paper, we stick to generalizations according to

domain generalization hierarchies (DGH) given in Fig. 1 since they are widely used in the literature.

**Definition 1 (i-Gen Function).** For two data values v* and v from some attribute A, we write $v^*=\Delta_i(v)$ if and only if v* is the ith parent of v in the DGH for A. Similarly for tuples $t,t^*,t^*=\Delta_{i1,\ldots\ldots in(t)}$ iff $t^*[c] = \Delta_{ic}t[c]$ for all columns c. Function Δ without a subscript returns all

possible generalizations of a value v. We also abuse notation and write $\Delta^1(v^*)$ to indicate the children of v* at the leaf nodes.

For example, given DGH structures in Fig. 1

$\Delta 1(USA) = AM,$

$\Delta 2(Canada) = *, \Delta 01(<M,USA>) = <M,AM>,$

$\Delta(USA) = \{USA,AM,*\}, \Delta^{-1}(AM)$

$= \{USA,Canada,Peru,Brazil\}$

**Definition 2 (μ-Generalization).** A generalization mapping μ is any surjective function that maps tuples from domain D to a generalized domain D* such that for t € Dand t*€ D*.we have

TABLE 1

Home party and Remote party Data Sets and Their Local and Global Anonymizations.

| Name Sex Nation | Salary |
|---|---|
| q1 F England | >40K |
| q2 M Canada | ≤40K |
| q3 M USA | ≤40K |
| q4 F Peru | ≤40K |

$T_\sigma$

| Name Sex Nation | Salary |
|---|---|

| | | |
|---|---|---|
| q5 Canada | M | >40K |
| q6 USA | M | >40K |
| q7 Brazil | F | >40K |
| q8 Italy | F | ≤40K |

$T_1$

| Name Nation | Sex | Salary |
|---|---|---|
| q1 * | F | >40K |
| q2 * | M | ≤40K |
| q3 * | M | ≤40K |
| q4 * | F | ≤40K |

$T^*_\sigma$

$\mu(t) = t^*$ (we also use notation $\Delta_\mu(t) = \mu(t)$ for consistency) only if $t^* \in \Delta(t)$. We say a table $T^*$ is a $\mu$-generalization of a

table T with respect to a set of attributes QI and write $\Delta_\mu(T) = T^*$, if and only if records in $T^*$ can be ordered in such a way that $\Delta_\mu(T[QI][r]) = T^*[QI][r]$ for every row r.

From now on, we use the superscript * in table notations to indicate generalizations.

**Definition 3 (Single Dimensional Generalization).** We say a mapping _ is $\mu$ is $[i_1; \ldots, i_n]$ single dimensional with respect to set of attributes $QI = \{A_1; \ldots; A_n\}$ iff given $\mu(t) = t^*$, we have $t^* = \Delta_{i1 \ldots in}(t)$.

For example, in Table 1, tables $T^*_{\sigma}$,$T^*_1$ are [0, 2] generalizations of Tσ and T1, (Tσ)respectively,w.r,t. attributes sex and nation. Similarly $T^*_{u,\sigma} = \Delta_{0,1}$ , $T^*_{u,1} = \Delta_{0,1}(T1)$.

**Definition 4 (Multidimensional Generalization).** We say a mapping μ is multidimensional iff the following conditions are satisfied. we have μ(t) = t* only if t* € Δ(t) and whenever we have μ(t) = t*, we also have for every $\mu(t_i) = t^*$, for every ti € $\Delta^{-1}(t^*)$.

Every single dimensional mapping is also multidimensional

| Name | Sex | Salary |
|---|---|---|
| Nation | | |
| q5 | M * | >40K |
| q6 | M * | >40K |
| q7 | F * | >40K |
| q8 | F * | ≤40K |

$sT^*_1$

| Name | Sex | Salary |
|---|---|---|
| Nation | | |
| q1 | F | >40K |
| EU | | |
| q2 | M | ≤40K |
| AM | | |
| q3 | M | ≤40K |
| AM | | |
| q4 | F | ≤40K |
| AM | | |

| | | |
|---|---|---|
| q5 | M | >40K |
| AM | | |

| | | |
|---|---|---|
| q6 AM | M | >40K |
| q7 AM | F | >40K |
| q8 EU | F | ≤40K |

$$T^*_u = T^*_{u,\sigma} \cup T^*_{u,1}$$

**Definition 5.** Given two generalization mappings $\mu_1$ and $\mu_2$, we say $\mu_1$ is a more general or higher mapping than $\mu_2$ and write $\mu_1 \, C \, \mu_2$ iff $\Delta_{\mu 1}(T)$ is a generalization of $\Delta_{\mu 2}(T)$ for all possible T.

For single dimensional mappings $\Delta_{\mu 1} = [i_1,\ldots..,i_n]$ and $\mu_2 = [j_1,\ldots.,j_n]$ iff $\mu_{1\neq} \mu_2$ and $i_s \geq j_s$ for all s 2 ½1 n_. For example, [0, 2] is a higher mapping than [0, 1].We now revisit briefly k-anonymity definitions.

While publishing person specific sensitive data, simply removing uniquely identifying information (SSN, name) from data is not sufficient to prevent identification because partially identifying information, quasi identifiers, (age, sex,nation, . . . ) can still be mapped to individuals (and possibly
to their sensitive information such as salary) by using an external knowledge [26]. (Even though Tσ of Table 1 does not contain information about names, releasing Tσ is not safe when external information about QI attributes ispresent. If an adversary knows some person Alice is a British female and is in T_; she can map Alice to tuple q1 thus to salary >40K.) The goal of privacy protection based on k-anonymity is to limit the linking of a record from a set of released records to a specific individual even when adversaries can link individuals via QI.

**Definition 6 (k-Anonymity).** A table T* is k-anonymous with respect to a set of quasi-identifier attributes QI if each record in T*[QI] appears at least k times.

For example, $T^*_\sigma$,$T^*_1$ are 2-anonymous generalizations of $T_\sigma$ and $T_1$, respectively. Note that given $T^*_\sigma$ , the same adversary can at best link Alice to tuples {q1 and q4}.

**Definition 7 (Equivalence Class).** The equivalence class of tuple t in data set T* is the set of all tuples in T* with identical quasi-identifier values to t.

For example, in data set $T^*_\sigma$ , the equivalence class for tuple q1 is {q1; q4}.

There may be more than one k-anonymizations of a given data set, and the one with the most information

content is desirable. Previous literature has presented many metrics to measure the utility of a given anonymization [9],[21], [13], [4], [1]. We revisit Loss Metric (LM) defined in [9]. LM penalizes each generalization value v* proportional to

$|\Delta(v^*)|$ and returns an average penalty for the generalization. Let a be the number of attributes, then

$$LM(T^*) = \frac{1}{|T^* \cdot a|} \sum_{i,j} \frac{|\Delta^{-1}(T[i][j]|-1}{|\Delta^{-1}(*)|-1} \qquad (1)$$

For example, LM penalizes the value EU in Nation

attribute of Table $T^*_u$ as (2-1)/(6-1) = 1/5 since there are two children of EU node in Fig. 1. Similarly, penalties for Canada, AM, and * are 0/5, 3/5, and 5/5, respectively. Note that as the number of children of a generalized value increases, the amount of uncertainty due to generalization increases as well resulting in more LM penalty. LM cost of

the table is the average penalty in all data cells in $T^*_u$

Another metric we will be using is the _-cost metric

defined as follows.

**Definition 8 (_μ-Cost).** Given a generalization T* of a table T, μ-cost function returns the generalization mapping of T*:for example, for single dimensional mappings $\mu(T^*) = [i_1,\ldots,i_n]$ iff $T^* = \Delta_{i1,\ldots,in}(T)$.

μ-Cost of $T^*_{u,1}$ is [0,1].

**Theorem 1**. Given mappings $\mu_1$ C $\mu_2$ and $T^* = \Delta_{\mu 1(T),}$

$T^*_2 = \Delta\mu 2(T)$;T*2 is utilized at least as well as T*1

(e.g.,T*2 is at least as informative as T*1)

The above theorem is true because $T^*_1$ can be simulated from $T^*_2$ . For example, in Table 1, $T^*_{u,\sigma}$ is utilized at least as well as $T_\sigma$. Also note that, as expected, $LM(T^*_{u,\sigma}) < LM(T^*_\sigma)$.The theorem gives us a way to use μ-cost as a partial order

to quantify utility. However, note that unlike previously proposed metrics, not all mappings are comparable with respect to μ-cost. Specifically, if $\mu_1 \not\subset \mu_2$ and $\mu_2 \not\subset \mu_1$, we cannot compare the inherit utility in generalizations $T^*_1$ and $T^*_2$ . For example, mappings [0,1] and [1, 0] are not comparable

with respect to μ-cost. We now define, for each

aforementioned types of mappings, a separate distance function that will allow us to quantify the change in utility when using two comparable mappings.

**Definition 9 (Single Dimensional Distance Function).**

Given two single dimensional mappings $\mu 1 = [i_1,\ldots i_n]$ and $\mu_2 = [j_1,\ldots,j_n]$ with $\mu_1 C \mu_2$. We define $\mu_1 - \mu_2 = \Sigma s\ i_s - j_s$.

For example,[0,1] – [0,2] = 1,[0,0] –[0,2] = 2. Note that [0,0] is further away then [0,2]and more utilized than [0,1].

**Definition 10 (Multidimensional Distance Function).**

Given a private table T, let $\mu_1$ and $\mu_2$ be two multidimensional mappings and $D_1$ and $D_2$ be the domains of $\Delta_{\mu 1}(T)$ and $\Delta_{\mu 2}(T)$ respectively. Then, $\mu_2 - \mu_1 = |D_2| - |D_1|$

For example, in Table 1, let $\mu_1$ and $\mu_u$ be two multidimensional mappings with $\Delta_{\mu 1}(T) = T^*_1$ and $\Delta_{\mu u}(T) = T^*_u$.The domains for the tables are given as Du = {<M,AM>,<M,EU>,<F,AM>,<F,EU>}

Thus, $\mu_u - \mu_1 = |D_u| - |D_1| = 2$.

Single dimensional and multidimensional mappings

both split the original data domain into several partitions. Both distance functions depend on the number of these partitions created by the mappings. This is not arbitrary as the number of partitions in the anonymization mostly correlates with utility.

In the rest of the paper, we only operate on the partitions and refer to the underlying utility metric (e.g., μ-cost or LM). Thus, the discussion applies to both types of mappings. But for the sake of discussion, we will use single dimensional generalizations as examples.

Since k-anonymity does not enforce constraints on the sensitive attributes, sensitive information disclosure is still possible in a k-anonymization. (e.g., in $T^*_\sigma$ both tuples of equivalence class {q2,q3} have the same sensitive value.) This problem has been addressed in [18], [16], [22] by enforcing diversity on sensitive attributes within a given equivalence class. We also show, how to extend the look-ahead process to support diversity on sensitive attributes. It should be noted that even extensions to k-anonymity have

TABLE 2

Descendant Preserving K-Anonymization $T^*_u$

| Name | Sex | Nation | Salary | Map |
|---|---|---|---|---|
| q1 | F | EU | >40K | [0,2] |
| q2 | M | AM | ≤40K | [0,2] |
| q3 | M | AM | ≤40K | [0,2] |
| q4 | F | AM | ≤40K | [0,2] |
| q5 | M | AM | >40K | [0,2] |
| q6 | M | AM | >40K | [0,2] |
| q7 | F | AM | >40K | [0,2] |
| q8 | F | EU | ≤40K | [0,2] |

vulnerabilities in the case of external knowledge (e.g., anonymizations can be subject to intersection attacks when there are multiple releases for the same individual [6].) As our focus in this paper is the look-ahead process, we do not present further detail.

For the sake of simplicity, from now on we assume data sets contain only QI attributes unless noted otherwise.

### 2.2 Distributed k-Anonymity

Even though k-anonymization of data sets by a single data owner has been studied extensively; in real world, databases may not reside in one source. Data might be horizontally or vertically partitioned over multiple parties all of which may be willing to participate to generate a kanonymization of the union. The main purpose of the participation is using a larger data set to create a better utilized k anonymization.

As reported in [7], as we increase data size, fewer tuples need to be suppressed or generalized to satisfy k-anonymity, in other words k-anonymization can be satisfied with lower level mappings. Suppose in Table 1, two parties $P_\sigma$ and $P_1$ have $T_\sigma$ and $T_1$ as private data sets and agree to release a 2-anonymous union. Since data are horizontally partitioned, one solution is to 2-anonymize locally and takea union. $T^*_\sigma$,$T^*_1$ are optimal (with minimal distortion) 2- anonymous full-domain generalizations of $T_\sigma$ and T1, respectively. However, optimal 2-anonymization of $T_\sigma$ U $T_1$; $T^*_U$ is better utilized than $T^*_\sigma$U $T^*_1$ .So there is a clear benefit in working on the union of the data sets instead of working separately on each private data set.

As mentioned above, in most cases, there is no trusted party to make a secure local anonymization on the union. So SMC protocols are developed in [11], [10], [31] among parties to securely compute the anonymization among semihonest parties.

TABLE 3

Notations for an SMC Protocol

| | |
|---|---|
| Pσ | Home party. |
| UiPi | Set of remote parties i € {i,….,n}. |
| Tj | Private table of Pj, j € {σ,1,…,n}. |
| T*j | Local k-anonymization of Tj,j €{σ,1,….,n}. |
| Tu | Global union: $T_\sigma u U_i T_i$ . |
| T*u | Global k-anonymization of $T_u$. |
| T*u,j | The portion of $T^*_U$ the generalization from Tj, j € {σ,1,….,n}. |

In this paper, we assume data are horizontally partitioned. A look ahead on vertically partitioned data involves a comparative quantification of utility over different projections of the data set thus requires the design of correlation aware cost metrics. We leave such an analysis as a future work and focus on horizontally partitioned data sets. We assume we have n + 1 parties $P_\sigma$, $P_1$,… , $P_n$ with private tables Tσ,$T_1$, . . . ., $T_n$. The home party $P_\sigma$ is looking ahead of the SMC protocol and remote parties $P_1$; . . . ; $P_n$ are supplying statistical information on the union of their private tables,$U_iT_i$. We use the notation $T_u$ for the global union (e.g., $T_U = T_\sigma uUiTi$). We use the notation T*u,j to indicate the portion of $T^*_u$ that is generalized from $T_j$ (see Table 1), thus $T^*_u = T^*_{u,\sigma}$ u $U_iT^*_{u,i}$..In Table 3, we summarize the notations we use in later sections.

In this paper, we securely look ahead of SMC protocols for the following two functionalities:

**Definition 11 (Optimal k-Anonymization Function $O^o_k$).**

Given a set of tables {$T_\sigma$; $T_1$, . . . $T_n$}, and a cost metric, $O^o_k$ returns a single dimensional k-anonymization of $T_u$ with the minimum cost.

The above definition can be rewritten for multidimensional generalizations as well. In Table 1, $O^o_k [T_\sigma,T_1,….,T_n) = T^*_u$.

**Definition 12 (Descendant Preserving Optimal k-Anonymization Function $O^d_k$).**

Let $T^*_\sigma,T_1,….,T_n$ be the optimal k-anonymization of set of tables ST = {$T_\sigma,T_1,….,T_n$} with mappings $\mu_\sigma,\mu_1,…,\mu_n$ respectively. Suppose {$Q_1$, . . ..$Q_m$}
be the set of (quasi identifier) attributes in each T € ST. GivenST, $O^d_k$ returns $T^*_u$ with m + 1 attributes such that

- $T^*_u[1 – m]$ is k-anonymous with a single dimensional mapping $\mu_u$
- Each mapping in $\{\mu_\sigma,\mu_1,…,\mu_n\}$ is a higher mapping than $\mu_u$.
- Among all generalizations satisfying the above criteria,$T^*_u$[1-m] minimizes the utility cost metric.
- $T^*_u[m = 1][j] = \mu_i$ if T[1-m][j]€$T^*_{u,i}$ for I € σ, 1,…,n.

Informally $O^{d}_{k}$ returns an anonymization for which the generalization mapping is a descendant of all local mappings. In addition, the returned anonymization contains the local mappings but does not link the mappings to the parties. It should be noted that the resulting anonymization is still k-anonymous since the new attribute is not a quasi identifier(e.g., the attribute cannot be used to link individuals to tuples).$T^{*}_{u}$ of Table 2 is a descendant preserving optimal k-anonymization of $T_{\sigma}$ and $T_1$ of Table 1.

### 2.3 k-Anonymity Extensions

Many extensions to k-anonymity have been proposed to deal with potential disclosure problems in the basic

definition [22], [16], [18], [20], [29], [30], [19], [3]. Problems arise mostly because k-anonymity does not enforce diversity on the sensitive values within an equivalence class. Even though, there is no distributed protocol proposed for the k-anonymity extensions yet, there is strong motivation in doing so. In Appendix B, available in the online supplemental material, we design a look ahead for recursive $(c,\ell)$-diversity protocol.

## 3 SECURE LOOK AHEAD

In this paper, we follow the standard simulation based definition of security for semihonest parties as can be read in [8]. The parties have private inputs $T_1, \ldots, T_n$, and wish to securely compute a function f on these inputs. Briefly, a protocol $\prod_f$ for computing f is a set of Turing machines (one per party). The Turing machines are connected pair wise with communication tapes on which they can send and receive (private) messages. A protocol is executed

by running the Turing machines where each Turing

machine gets the private input of the corresponding party, we write $\prod_f [T_1, \ldots, T_n]$. The list of all messages received by the ith Turing machine on all it's communication tapes during the execution of the protocol is called the view of party i. A protocol for functionality f is said to be secure if, for all parties i, the view of party i can be efficiently simulated from the input, $T_i$, the output $f(T_1, \ldots, T_n)$,[1] and any background information of the ith party. Put simply, simulating the view of party i means that there exists a polynomial time probabilistic algorithm which can

generate tuples with a statistical distribution that is indistinguishable (either statistically close, or computationally indistinguishable in polynomial time, depending on the desired security model) from the view of the party. For more details on this definition the reader is referred to [8].

In look-ahead SMC the parties have a main functionality, f1, which they wish to compute. However, since the protocol, $\prod_{f1}$ , for computing f1 may be too costly in some cases, the parties agree on a decision predicate, d. If the decision predicate evaluated on the inputs of the parties is 1, the benefit of executing $\prod_{f1}$ is well worth the cost. However, if the predicate evaluates to 0, the cost of running

$\prod_{f1}$ is expected to exceed the value of the output. In this case, the parties have already obtained some information about the inputs: namely the result of the decision predicate, so our security model needs to allow for some information to be released even when the predicate is false. In general we allow the parties to agree on a fall-back functionality, f0. If the decision predicate on their inputs is 0 the parties run a protocol for the fall-back functionality instead of the protocol for the main functionality. It is

required that there are efficient protocols, $\prod_{d}$ and $\prod_{f0}$ , forboth the decision predicate and the fall-back functionality.

**Definition 14 (Secure Decision Computation Protocol).**

Given functionalities f0 and f1, and decision predicate d, a protocol $\prod_{d}$ is a secure decision computation protocol for f1 with fall-back f0 if the following is satisfied:

- Correctness: $\prod_{b}[T_1, \ldots, T_n] = d[T_1,\ldots,T_n]$.
- Implied by main functionality: The view of party I during the execution of $\prod_{d}$ can be simulated from the input of party i, the prior knowledge of party i, and $f_1(T_1,\ldots,T_n)$.
- Implied by fall-back functionality: The view of party i during the execution of $\prod_{d}$ can be simulated from the input of party i, the prior knowledge of party i, and $f_o(T_1,\ldots,T_n)$.

The difference between a normal secure protocol for d, and a secure decision computation protocol is that in the latter the simulator is allowed to use the outputs of either the main functionality or the fall-back functionality instead of just the 1-bit output of the predicate. This gives us more freedom while creating the protocol for d, since we allow $\prod_{d}$ to leak much more information. However, the extra information leaked will be leaked by either the main

functionality or the fall-back functionality once they are executed, so it does not give an attacker any added value .Also note that the security definition is not arbitrary. The standard security definitions of SMC require that, once a protocol has been computed (be it the protocol for f1 or the protocol for f2) the views should be simulatable from the output of the function (f1 OR f2, we don't know yet). In order to guarantee that this is the case, no matter which protocol is computed next, the initial decision protocol has to be simulatable in both cases.

Given the three protocols $\prod_{f1}$ ,$\prod_{f0}$ , and $\prod_{d}$ we now define the secure look-ahead SMC protocol:

1. Compute b = $\prod_{d}[T_1,\ldots,T_n]$.
2. If b = 1, run protocol _f1 for the main functionality,
3. else run protocol $\prod_{f0}$ for the fall-back functionality.

Our claim is that the look-ahead SMC is secure, as long as $\prod_{f0}$ and $\prod_{f1}$ are secure SMC protocols, and $\prod_{d}$ is a secure decision computation protocol.

**Theorem 2**. Given main functionality f1 with SMC protocol $\prod_{f1}$ , fall-back functionality f0 with SMC protocol $\prod_{f0}$ , and decision predicate d with secure decision computation protocol $\prod_{d}$, the above protocol is a secure multiparty computation for the Functionality.

$F(T_1,\ldots,T_n) = [d(T_1,\ldots,T_n), f_{d(T1,\ldots,Tn)}(T_1,\ldots,T_n)]$.

**Proof**. Let $S_{f1}$ and $S_{f0}$ be simulators of $\prod$f1 and $\prod$f0 , respectively, and let Sd,0 and Sd,1 be the simulators from Definition 14. The simulator, S, for F works as follows: The simulator is given F(T1,…,Tn) = [δ,γ]as input. The value _ is used as input to the simulator Sfb to compute s = Sfb(γ),and then s is used in the simulator Sd,b to compute $s^1$ = Sd,b(s), the output of S is ($s^1$,s)

. We now argue that the simulation is indistinguishable from the view of the adversary. Since the parties are semihonest, we are guaranteed that the view, ($v^1$,v), of the adversary is from a protocol run with outputs($\delta^1$,$\gamma^1$), where $\delta^1$ = d(T1,…,Tn), and $\gamma^1$ = $f\delta^1$(T1,..Tn). In particular, $\delta^1$ and $\gamma^1$ are generated from the same input, T1,….,Tn, and $\gamma^1$ is obtained by running protocol

$\prod f\delta^1$. The security of the protocol $\prod f\delta^1$ ensures us that the simulated view v is indistinguishable from the view of running $\prod f\delta^1$. The security (secure decision computation) of $\prod d$ means that the view of $\prod d$ can be simulated from $f\delta^1$ (since $\delta^1$ is guaranteed to be equal to $\prod d[T1,\ldots,Tn]$.

The definitions and results in this section can be applied to both computational and unconditional security. The proof for Theorem 2 works for both computational and statistical indistinguishability as long as the underlying protocols are secure in the corresponding model.

## 4 PROBLEM DEFINITION

Recall from Section 4, distributed k-anonymity protocol is c, p-sufficient for party σ iff

$$P(\mu(O_k(T_u)) - \mu(O_k(T_\sigma)) \geq c(K_f) \geq p$$

$\mu^\sigma = \mu(O_k(T_\sigma))$ requires local input and can be computed by party σ

Let $S\mu = \{\mu_1^{=c},\ldots,\mu_m^{=c}\}$ be the mappings that are exactly c distance beyond μσ and $\{\mu_1^{>c},\ldots,\mu_m^{>c}\}$ be the mappings that are more than c distance beyond $\mu^\sigma$. Let also $A_\mu$ be the event that $\Delta_\mu(T_U)$ is k-anonymous. Then, we have

$$P(\mu(O_k(T_u)) - \mu^\sigma) \geq c|K_F)$$
$$= P((U_i A_{\mu i}^{=c}) \cup (U_i A_{\mu i}^{>c})|K_F$$
$$\geq P(U_i A_{\mu i}^{=c})|K_F)$$
$$\geq \text{Max}_i\, P(A_{\mu i}^{=c})|K_F)$$

This follows from the monotonicity of k-anonymity. So the problem of sufficiency reduces to proving that, for at least one $\mu \in S_{\mu^1}$.

$$P(A\mu|K_F \geq p.$$

From now on, we will name $P(A_\mu|KF)$ (the probability that $\Delta_\mu T$ is k-anonymous given KF ) as the μ-probability.

Suppose in Table 1, party σ needs to check for (1,p)- sufficiency. Optimal 2anonymization for party σ's private table $T_\sigma$ is $T_\sigma^*$ with $\mu(T_\sigma^*) = [0,2]$. There is only one lower mapping [0, 1] which is 1 away from [0, 2]. So we need to check if $P(\Delta_{0,1}(T_u)$ is 2-

anonymous |KF)≥p. Note that we do not need to check also for the mapping [0, 0] since if $\Delta_{0,1}(T_u)$ violates k-anonymity so does $\Delta_{0,0}(T_u)$.

Note that we take the safe road and require one mapping in the set Sμ to have a sufficient probability of producing kanonymization. This is not a necessary requirement as the probability of producing a k-anonymization from some mapping in Sμ is always higher than producing it from any given mapping within the set. However, it is difficult to calculate the exact probability of the union as the events $A_{\mu i}^{=c}$ are not independent, thus just summing up each individual probability would not work. Note that this assumption harms the accuracy of the look-ahead process but will have no effect on the privacy. The resulting tables will satisfy k-anonymity no matter what decision is returned by the look ahead.

As we increase c, it becomes less likely that $\mu_i(O_k(T_u)$ will be k-anonymous for any $\mu_i$ € $S_\mu$. We will observe in that it is unlikely to get a k -anonymization with c > 2. For c = 2, the number of mappings for which we should calculate μ-probability (e.g.,| $S_\mu$|) is bounded by n2. However, even if the parties agree to check for a high c, the
parties can choose to analyze the set of mappings
$\{\mu_1^{<=c},\ldots,\mu_m^{<=c}\}$. In this case, the number of mappings that require a look ahead, in practice, is far less than |$S_\mu$|. This is due to antimonotonicity. The μ-probability for a child mapping cannot be any bigger than that of the parent mapping. So if the parties see a low μ-probability for a μ, they can safely prune its children and grand children without calculating the μ-probability for them.

We also want to note that the underlying framework would still apply even if we use a utility metric other than μ-cost. The set Sμ, in that case, would contain those mappings that give better utilization than $\mu^\sigma$.

In the next section, we show how to calculate $P(A_\mu|K_F)$, the μ-probability, for a distributed k-anonymity protocol.

## 5 LOOK-AHEAD PROTOCOLS

In this section, we describe secure look-ahead protocols for the two functionalities mentioned in previous Section . Assuming we have n remote parties $P_1$,...., Pn, for each protocol, we state what information is being transferred to party σ and also prove that the proposed protocol is secure with respect to the definitions given in Section 3.

In each one of the protocols, the information shared with $P_\sigma$ can be simulated from the output. The security proof at the end of this section will exploit this fact.

### 5.1 Optimal Distributed k-Anonymization

In ODK, party σ only learns the total size of

$U_{i=1}^{n}T_i$(e.g., $K_N = |U_{i=1}^{n}T_i|$ . Party σ assumes the distribution of tuples in $U_{i=1}^{n}T_i$ Ti are similar to those in $T_\sigma$. To avoid over fitting, instead of using the exact statistics from $T_\sigma$, party σ extracts histograms from Tσ, normalizes them such σ that the total number of tuples is $K_N$, and proceeds as if histograms were extracted from $U_{i=1}^{n}T_i$. Party σ calculates $K_F$ from the histograms assuming attribute independence.

Algorithm 1 shows how the home party gets $K_N = |U_{i=1}^{n}T_i|$ securely. In line 3, each party i creates a random number ri sends it to Pσ. In line 4, party i adds the size of its private data along with ri to a random sum and the last party sends the final sum back to party σ. Party σ finds the total size by subtracting the sum of all random numbersfrom the final sum.

**Algorithm 1**. Secure Look Ahead Protocol for ODK

R**equire**: Parties Pσ,P1, . . . ,Pn agree on an integer m which cannot be smaller than $|U_{i=1}^{n}T_i|$

**Ensure:** Pσ gets $|U_{i=1}^{n}T_i|$ and calculates the decision predicate for a given c and p.

1: $sum_0 = 0$

2: while i _ n do

3: Pi generates a random integer ri such that $r_i \in [0,m-1]$ sends it to Pσ.

4: Pi calculates $sum_i = sum_{i-1} + r_i + |T_i| \bmod m$ and if $i \neq n$ sends it to $P_{i+1}$

5: i++.

6: $P_n$ sends $sum_n$ to $P_\sigma$.

7: $P_\sigma$ calculates $K_N = sum_n - \Sigma_j r_j \bmod m$.

8: $P_\sigma$ extracts histograms from $T_\sigma$, calculates $K_F$ from the histograms assuming attribute independence and $K_N$.

9: $P_\sigma$ calculates the decision predicate $P(I_\sigma \geq c|K_F)$ To avoid over fitting, Algorithm 1 extracts histograms from $T_\sigma$. However, if the home party has a high confidence that the distributions of private tables $T_i$s are very similar to that of $T_\sigma$, she can choose to extract other statistics that better describe.

### 5.2 Descendant Preserving Distributed k-Anonymization

In DPK, party σ gets a distorted global histogram (e.g., $K_{F^1}$ ) on $U_{i=1}^{n}T_i$. Each party i contributes in forming the histogram; however the amount of information shared by each party is bounded by their local anonymizations. We achieve this by having parties extract histograms from their local anonymizations rather than their private tables. We use SMC protocols to calculate the a distorted global distribution $K_{F^1}$ . Thus, for more than two parties, private shares inherent in the global $K_{F^1}$ are indistinguishable. Fortunately, such a protocol is not costly for noncolluding parties.

Algorithm 2 shows how parties can calculate global $K_{F^1}$ securely. In line 7, each party i creates a random number $r_{i,j}$ for each domain value $v_j$ and sends it to party . In line 8, each party adds its private share (which we explain shortly) along with $r_{i,j}$ to a random sum and the last party sends the final sum back to party σ. Party σ finds the global distribution by subtracting the sum of all random numbers from the final sum.

**Algorithm** 2. Secure Look Ahead Protocol for DPK

**Require:** Parties $P_\sigma, P_1, \ldots, P_n$ agree on an integer m which cannot be smaller than $|T_U|$

**Ensure:** $P_\sigma$ gets the set of distorted distribution functions $K_F{}^1 = U_a f^1{}_a$ on $U_{i=1}^{n}T_i$ and calculates the decision predicate for a given c and p.

1: for all attributes a do

2: Let set of values $\{v_1,\ldots,v_s\}$be the domain of a

3: i = 1

4: $SUM_0 = \{sum_{0,0},\ldots, sum_{0,s}\}$, $sum_{0,j} = 0$ for all j.

5: while $i \leq n$ do

6: $P_i$ calculates the distorted distribution function

Gi = getUniform(a).

7: Pi generates a vector of random integers

Ri $\{r_{i,1}, \ldots ; r_{i,s}\}$ such that $r_{i,j} \in [0, m-1]$ for all j

and sends it to Pσ.

8: Pi calculates $SUM_i = SUM_{i-1} + R_i +$

$\{Gi(v_i), \ldots, Gi(v_s)\} \bmod m$ and if $i \neq n$ sends it to $P_{i+1}$.

9: i++.

10: $P_n$ sends $SUM_n$ to $P_\sigma$.

11: Pσ calculates the distorted distribution for a;

$f^1_a(v_i) = sum_{ni} - \Sigma_j r_{I,j} \bmod m$..

12: $P_\sigma$ calculates the decision predicate $P(I_\sigma \geq c | K_F^1) \geq p$

The important point here is that private shares of parties do not contain the exact frequency of $v_i$. Parties distort the frequency of values The algorithm

Get Uniform returns new distributions from the local kanonymization rather than the private table. The anonymized distribution (of values of possibly coarser granularity) is first extracted and a new distribution on atomic values (e.g., G) that respects the anonymized distribution is returned randomly. Randomization should enforce symmetry thus indistinguishability between each atomic value in the same equivalence class ($P(G(v_i) = x) = P(G(vj) = x)$ for all i; j; x).

## 6 CONCLUSION AND FUTURE WORK

Most SMC protocols are expensive in both communication and computation. We introduced a look-ahead approach for SMC protocols that helps involved parties to decide whether the protocol will meet the expectations before initiating it. We presented a look-ahead protocol specifically for the distributed k-anonymity by approximating the probability that the output of the SMC will be more utilized than their local anonymizations. Experiments on real data showed the effectiveness of the approach. Designing look aheads for other SMC protocols stands as a future work. A wide variety of SMC protocols have been proposed especially for privacy preserving data mining applications [12], [17], [28] each requiring a unique lookahead approach. As for the look-ahead process on distributed anonymization protocols, definitions of k-anonymity definitions can be revisited, more efficient techniques can be developed and experimentally evaluated.

**REFERENCES**

[1] R.J. Bayardo and R. Agrawal, "Data Privacy through Optimal KAnonymization," Proc. 21st Int'l Conf. Data Eng. (ICDE '05), pp. 217-228, 2005.

[2] C. Blake and C.J. Merz, "UCI Repository of Machine Learning Databases," http://www.ics.uci.edu/mlearn/MLRepository.

html, Univ. of California, Irvine, Dept. of Information and Computer Sciences, 2012.

[3] B.-C. Chen, K. LeFevre, and R. Ramakrishnan, "Privacy Skyline: Privacy with Multidimensional Adversarial Knowledge," Proc.33rd Int'l Conf. Very Large Data Bases (VLDB '07), pp. 770-781, 2007.

[4] J. Domingo-Ferrer and V. Torra, "Ordinal, Continuous and Heterogeneous K-Anonymity through Microaggregation," Data Mining and Knowledge Discovery, vol. 11, no. 2, pp. 195-212, 2005.

[5] W. Feller, An Introduction to Probability Theory and Its Applications,vol. 1, Wiley, 1968.

[6] S.R. Ganta, S.P. Kasiviswanathan, and A. Smith, "Composition Attacks and Auxiliary Information in Data Privacy," Proc. 14th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining

(KDD '08), pp. 265-273,http://doi.acm.org/ 10.1145/1401890.1401926, 2008.

[7] G. Ghinita, P. Karras, P. Kalnis, and N. Mamoulis, "Fast Data Anonymization with Low Information Loss," Proc. 33rd Int'l Conf.

Very Large Data Bases (VLDB '07), pp. 758-769, 2007.

[8] O. Goldreich, The Foundations of Cryptography, vol. 2, Cambridge Univ. Press, http://www.wisdom.weizmann. ac.il/~oded/ PSBookFrag/enc.ps, 2004.

[9] V.S. Iyengar, "Transforming Data to Satisfy Privacy Constraints,"Proc. Eighth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data

Mining (KDD '02), pp. 279-288, 2002.

[10] W. Jiang and C. Clifton, "Privacy-Preserving Distributed k- Anonymity," Proc. 19th Ann. IFIP WG 11.3 Working Conf. Database and Applications Security, Aug. 2005.

[11] W. Jiang and C. Clifton, "A Secure Distributed Framework for Achieving k-Anonymity," VLDB J., special issue on privacy preserving data management, vol. 15, pp. 316-333, Sept. 2006.

[12] M. Kantarclu and C. Clifton, "Privacy-Preserving Distributed Mining of Association Rules on Horizontally Partitioned Data," IEEE Trans. Knowledge and Data Eng., vol. 16, no. 9, pp. 1026-1037, Sept. 2004.

[13] D. Kifer and J. Gehrke, "Injecting Utility into Anonymized Datasets," Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD '06), pp. 217-228, 2006.

[14] S.N. Lahiri, A. Chatterjeea, and T. Maiti, "Normal Approximation to the Hypergeometric Distribution in Nonstandard Cases and a Sub-Gaussian Berryesseen Theorem," J. Statistical Planning and Inference, vol. 137, no. 11, pp. 3570-3590, http://dx.doi.org/10. 1016/j.jspi.2007.03.033, Nov. 2007.

[15] B. Levin, "A Representation for Multinomial Cumulative Distribution Functions," The Annals of Statistics, vol. 9, no. 5, pp. 1123- 1126, http://www.jstor.org/stable/2240628, 1981.

[16] N. Li and T. Li, "T-Closeness: Privacy Beyond K-Anonymity and L-Diversity," Proc. IEEE 23rd Int'l Conf. Data Eng. (ICDE '07), Apr. 2007.

[17] Y. Lindell and B. Pinkas, "Privacy Preserving Data Mining," J. Cryptology, vol. 15, pp. 36-54, 2000.

[18] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam,'-Diversity: Privacy beyond k-Anonymity," Proc. IEEE 22nd Int'l Conf. Data Eng. (ICDE '06), Apr. 2006.

[19] D.J. Martin, D. Kifer, A. Machanavajjhala, J. Gehrke, and J.Y. Halpern, "Worst-Case Background Knowledge for Privacy-Preserving Data Publishing," Proc. IEEE 23rd Int'l Conf. Data Eng. (ICDE '07), Apr. 2007.

[20] M.E. Nergiz, M. Atzori, and C. Clifton, "Hiding the Presence of Individuals in Shared Databases," Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD '07), June 2007.

[21] M.E. Nergiz and C. Clifton, "Thoughts on K-Anonymization," Data and Knowledge Eng., vol. 63, no. 3, pp. 622-645, http:// dx.doi.org/10.1016/j.datak.2007.03.009, Dec. 2007.

[22] A. Øhrn and L. Ohno-Machado, "Using Boolean Reasoning to Anonymize Databases," Artificial Intelligence in Medicine, vol. 15, no. 3, pp. 235-254, http://dx.doi.org/10.1016/S0933 3657(98)00056-6, Mar. 1999.

[23] V. Poosala and Y.E. Ioannidis, "Selectivity Estimation without the Attribute Value Independence Assumption," Proc. 23rd Int'l Conf. Very Large Data Bases (VLDB '97), pp. 486-495, 1997.

[24] P. Samarati, "Protecting Respondent's Identities in Microdata Release," IEEE Trans. Knowledge and Data Eng., vol. 13, no. 6, pp. 1010-1027, Nov./Dec. 2001.

[25] S.J. Schwager, "Bonferroni Sometimes Loses," The Am. Statistician, vol. 38, no. 3, pp. 192-197, http://www.jstor.org/stable/2683651, 198

[26] L. Sweeney, "Achieving k-Anonymity Privacy Protection Using Generalization and Suppression," Int'l J. Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 10, no. 5, pp. 571-588, 2002.

[27] L. Sweeney, "k-Anonymity: A Model for Protecting Privacy," Int'l J. Uncertainty, Fuzziness Knowledge-Based Systems, vol. 10, no. 5,
pp. 557-570, 2002.

[28] J. Vaidya, "Privacy Preserving Data Mining Over Vertically Partitioned Data," PhD dissertation, Dept. of Computer Sciences, Purdue Univ., West Lafayette, Indiana, http://www.cs.purdue. edu/homes/jsvaidya/thesis.pdf, 2004.

[29] R.C.-W. Wong, J. Li, A.W.-C. Fu, and K. Wang, "(_, k)-Anonymity: An Enhanced K-Anonymity Model for Privacy Preserving Data Publishing," Proc. 12th ACM SIGKDD Int'l Conf. Knowledge
Discovery and Data Mining (KDD '06), pp. 754-759, 2006.

[30] X. Xiao and Y. Tao, "M-Invariance: Towards Privacy Preserving Re-Publication of Dynamic Datasets," Proc. ACM SIGMOD Int'l
Conf. Management of Data (SIGMOD '07), pp. 689-700, 2007.

[31] S. Zhong, Z. Yang, and R.N. Wright, "Privacy-Enhancing KAnonymization of Customer Data," Proc. 24th ACM SIGMODSIGACT- SIGART Symp. Principles of Database Systems (PODS '05),pp. 139-147, 2005.