Composition Properties of Inferential Privacy for Time-Series Data

Shuang Song  
University of California San Diego  
shs037@eng.ucsd.edu

Kamalika Chaudhuri  
University of California San Diego  
kamalika@eng.ucsd.edu

Abstract—With the proliferation of mobile devices and the internet of things, developing principled solutions for privacy in time series applications has become increasingly important. While differential privacy is the gold standard for database privacy, many time series applications require a different kind of guarantee, and a number of recent works have used some form of inferential privacy to address these situations.

However, a major barrier to using inferential privacy in practice is its lack of graceful composition – even if the same or related sensitive data is used in multiple releases that are safe individually, the combined release may have poor privacy properties. In this paper, we study composition properties of a form of inferential privacy called Pufferfish when applied to time-series data. We show that while general Pufferfish mechanisms may not compose gracefully, a specific Pufferfish mechanism, called the Markov Quilt Mechanism, which was recently introduced by [10], has strong composition properties comparable to that of pure differential privacy.

I. INTRODUCTION

With the proliferation of mobile devices and the internet of things, large amounts of time series data are being collected, stored and mined to draw inferences about the physical environment. Examples include activity recordings of elderly patients to determine the state of their health, power consumption data of residential and commercial buildings to predict power demand responses, location trajectories of users over time to deliver suitable advertisements, among many others. Much of this information is extremely sensitive – activity recordings yield information about what the patient is doing all day, power consumption of a residence can reveal occupancy, and location trajectories can reveal activities of the subjects. It is therefore imperative to develop principled and rigorous solutions that address privacy in these kinds of time series applications.

The gold standard for privacy in database applications has long been differential privacy [2]; the typical setting is that each record corresponds to the private value of a single person, and the goal is to design algorithms that can compute functions such as classifiers and clusterings on the sensitive data, while hiding the participation of a single person. Differential privacy has many good properties, such as post-processing invariance and graceful composition, which have led to its high popularity and practical use over the years.

Unfortunately, many of the time-series applications described above require a different kind of privacy guarantee. Consider the physical activity monitoring application for example, where the goal is to hide activity at small time intervals while revealing long-term activity patterns. Here the entire dataset is about a single patient, and hence hiding their participation will not be useful. An alternative is entry differential privacy, which hides the inclusion of activity at any given single time point in the data; since activities at close-by time points are highly correlated, this will not prevent an adversary from inferring the activity at the hidden time. To address these issues, a number of recent works [10], [8], [4], [11] have used the notion of inferential privacy, where the goal is to prevent an adversary who has some prior knowledge, from inferring the state of the time series at any particular time.

A clean and elegant framework for inferential privacy is Pufferfish [7], which is our privacy framework of choice. Pufferfish models a privacy problem through a triple \( \langle S, Q, \Theta \rangle \); here \( S \) is a set of secrets, which is a set of potential facts that we may wish to hide. \( Q \) is a set of tuples of the form \((s_t, s_j)\) where \(s_t, s_j \in S\) which represent which pairs of secrets should be indistinguishable to an adversary. Finally, \( \Theta \) is a set of distributions that can plausibly generate the data and describes prior beliefs of an adversary. A mechanism \( \mathcal{A} \) is said to satisfy \( \epsilon \)-Pufferfish privacy in the framework \( \langle S, Q, \Theta \rangle \) if an adversary’s posterior odds of every pair of secrets \((s_t, s_j)\) in \( Q \) is within a factor of \( e^\epsilon \) of its prior odds. Pufferfish models the physical activity monitoring application as follows – \( S \) consists of elements of the form \( s_t \), which represent patient has activity \( a \) at time \( t \), \( Q \) consists of tuples of the form \( (s_t^a, s_t^b) \) for all \( t \) and all activity pairs \((a, b)\), and \( \Theta \) consists of a set of Markov Chains that describe how activities transition across time.

However, a major limitation of Pufferfish privacy is that except under very special conditions, it often does not compose gracefully – even if the same or related sensitive data is used in multiple Pufferfish releases that are individually safe, the combined release may have poor privacy guarantees [7]. In many real applications, same or related data is often used across applications, and this forms a major barrier to the practical applicability of Pufferfish.

In this paper, we study this question, and we show a number of composition results for Pufferfish privacy for time series applications in the framework described above. Our results look at two scenarios – sequential and parallel composition; the first is when the same sensitive data is used across multiple computations, and the second is when disjoint sections of the Markov Chain are used in different computations. Note that while in differential privacy, composition in the second case is trivial, this does not apply to Pufferfish, as information about the state of one segment
of a Markov Chain can leak information about a correlated segment.

For sequential composition, we show that while in general we cannot expect any arbitrary Pufferfish mechanism to compose gracefully even for the time series framework described above, a specific mechanism, called the Markov Quilt Mechanism, that was recently introduced by [10], does compose linearly, much like pure differential privacy. For parallel composition, we provide two results; first, we show a general result that applies to any Pufferfish mechanism in the framework described above and shows that the privacy guarantee obtained from two releases on two disjoint segments A and B of the Markov Chain is the worse of the composition guarantees plus a correction factor that depends on the distance between A and B and properties of the chain. Second, we show that if the two segments of the chain are far enough, then, under some mild conditions, using a specific version of the Markov Quilt Mechanism can provide even better parallel composition guarantees, matching those of differential privacy. Our results thus demonstrate that the Markov Quilt Mechanism and its versions have strong composition properties when applied to Markov Chains, thus motivating their use for real time-series applications.

A. Related Work

Since graceful composition is a critical property of any privacy definition, there has been a significant amount of work on differential privacy composition [2], and it is known to compose rather gracefully. [2] shows that pure differential privacy composes linearly under sequential composition; for parallel composition, the guarantees are even better, and the combined privacy guarantee is the worst of the guarantees offered by the individual releases. [3] shows that a variant of differential privacy, called approximate differential privacy, has even better sequential composition properties than pure differential privacy. Optimal composition guarantees for both pure and approximate differential privacy are established by [6]. Finally, [1] provides a method for numerically calculating privacy guarantees obtained from composing a number of approximate differentially private mechanisms.

In contrast, little is known about the composition properties of inferential privacy. [7] provides examples to show that Pufferfish may not sequentially compose, except in some very special cases. [5] shows that a specialized version of Pufferfish, called Blowfish, which is somewhat closer to differential privacy does have graceful sequential composition properties; however, Blowfish does not apply to time-series data. [10] provides limited privacy guarantees for the Markov Quilt Mechanism under serial composition; however, these guarantees are worse than linear, and they only apply under much more stringent conditions – namely, if all the mechanisms use the same active Markov Quilt.

Two. Preliminaries

A. Time Series Data and Markov Chains

Many time-series data can be modeled using Markov chains.

Example 1. Suppose we have data tracking the physical activity of a subject: \((X_1, X_2, \ldots, X_T)\) where \(X_t\) denotes activity (e.g., running, sitting, etc) of the subject at time \(t\). Our goal is to provide the aggregate activity pattern of the subject by releasing (an approximate) histogram, while preventing an adversary from finding out what the subject was doing at a specific time \(t\).

Example 2. Suppose we have power consumption data for a house: \((X_1, X_2, \ldots, X_T)\), where \(X_t\) is the power level in Watts at time \(t\). Our goal is to output a general power consumption pattern of the household by releasing (an approximate) histogram of the power levels, while preventing an adversary from inferring the power level at a specific time \(t\); specific power levels may be sensitive information, as the presence or absence of family members at a given time can be inferred with the power level.

Markov Chains. Temporal correlation in this kind of time-series data is usually captured by a Markov chain \(X_1 \rightarrow X_2 \rightarrow \ldots \rightarrow X_T\), where \([k]\) represents all possible states and \(X_t \in [k]\) represents the state at time \(t\). In Example 1, \(X_t\) represents the activity performed by the subject at time \(t\) and \([k]\) represents all possible activities. In Example 2, \(X_t\) represents the power level of the house at time \(t\) and \([k]\) represents all possible power levels. The transition from one state to another is determined by a transition matrix \(P\), and state of \(X_1\) is drawn from an initial distribution \(q\).

B. The Pufferfish Privacy Framework

Pufferfish privacy framework captures the privacy in these examples. We next define it and specify how the examples mentioned fit in.

A Pufferfish framework is specified by three parameters – a set of secret \(S\), a set of secret pairs \(Q\) and a set of data distributions \(\Theta\). \(S\) consists of possible facts about the data that need to be protected. \(Q \subseteq S \times S\) is a set of secret pairs that we want to be indistinguishable. \(\Theta\) is a set of distributions that can plausibly generate the data and captures the correlation among records; each \(\theta \in \Theta\) represents one adversary’s belief of the data. The goal of Pufferfish framework is to ensure indistinguishability of the secrets pairs in \(Q\) under any belief in \(\Theta\). Now we define Pufferfish privacy under the framework \((S, Q, \Theta)\).

Definition 2.1 (Pufferfish Privacy): A privacy mechanism \(M\) is said to be \(\epsilon\)-Pufferfish private in a framework \((S, Q, \Theta)\) if for datasets \(X \sim \theta\) where \(\theta \in \Theta\), for all secret pairs \((s_i, s_j) \in Q\) and for all \(w \in \text{Range}(M)\), we have

\[
e^{-\epsilon} \leq \frac{P_{\Theta,M}(M(X) = w|s_i, \theta)}{P_{\Theta,M}(M(X) = w|s_j, \theta)} \leq e^{\epsilon} \tag{1}
\]

when \(s_i\) and \(s_j\) are such that \(P(s_i|\theta) \neq 0, P(s_j|\theta) \neq 0\).

Note that differential privacy is a special case of Pufferfish privacy when \(\Theta\) consists of all distributions where each record is independent of the rest [7].

Pufferfish Framework for Time-Series Data: We can model the time-series data described in the previous section with the following Pufferfish framework.

815
Let the database be a Markov chain \( X = (X_1 \rightarrow X_2 \rightarrow \ldots \rightarrow X_T) \), where each \( X_i \) lies in the state space \([k]\). Such a Markov Chain may be fully described by a tuple \((q, P)\) where \( q \) is an initial distribution and \( P \) is a transition matrix.

Let \( s_i^a \) denote the event that \( X_i \) takes value \( a \in [k] \). The set of secrets is \( S = \{ s_i^a : a \in [k], i \in [T] \} \), and the set of secret pairs is \( Q = \{ (s_i^a, s_i^b) : a, b \in [k], a \neq b, i \in [T] \} \). Each \( \theta = (q_\theta, P_\theta) \in \Theta \) represents a Markov chain of the above structure with transition matrix \( P_\theta \) and initial distribution \( q_\theta \).

In the first example, the state space represents the set of all possible activities and \( s_i^a \) or \( X_i = a \) represents the event that the subject is engaged in activity \( a \) at time \( i \). \( Q \) indicates that we do not want the adversary to distinguish whether the subject is engaged in activity \( a \) or \( b \) at a given time. In the second example, the state space represents the set of all possible power levels and \( s_i^a \) or \( X_i = a \) represents the event that the power level of the house is \( a \) at time \( i \). \( Q \) indicates that we do not want the adversary to distinguish whether the house is at power level \( a \) or \( b \) at a given time.

C. Notation

We use \( X \) with a lowercase subscript, for example, \( X_i \), to denote a single node in the Markov chain, and \( X \) with an uppercase subscript, for example, \( X_A \), to denote a set of nodes in the Markov chain. For a set of nodes \( X_A \) we use the notation \( \text{card}(X_A) \) to denote the number of nodes in \( X_A \).

For \( I \subseteq [T] \), we use \( X^I \) to denote the subchain \( \{X_i\}_{i \in I} \subseteq X \), and we use \( S^I \) to denote \( \{ (s_i^a, s_i^b) : a \in [k], i \in I \} \), \( Q^I \) to denote \( \{ (s_i^a, s_j^b) : a, b \in [k], a \neq b, i, j \in I \} \).

D. The Markov Quilt Mechanism

[10] proposes the Markov Quilt Mechanism (MQM). It can be used to achieve Pufferfish privacy in the case where \( \Theta \) consists of Bayesian networks, of which Markov chains are special cases. We restate the algorithm and the corresponding definitions in this section.

To understand the main idea of MQM, consider a Markov chain \( \theta \). Any two nodes in \( \theta \) are correlated to a certain degree, which means releasing the state of one node potentially provides information on the state of the other. However, the amount of correlation between two nodes usually decays as the distance between them grows. Consider a node \( X_i \) in the Markov chain. The nodes close to \( X_i \) can be highly influenced by its state, while the nodes that are far away are almost independent. Therefore to hide the effect of a node \( X_i \) on the result of a query, MQM adds noise that is roughly proportional to the number of nearby nodes, and uses a small correction term to account for the effect of the almost independent set.

To measure the amount of dependence, [10] defines max-influence.

**Definition 2.2 (max-influence):** The max-influence of a variable \( X_i \) on a set of variables \( X_A \) under \( \Theta \) is

\[
e_{\Theta}(X_A|X_i) = \sup_{\theta \in \Theta} \max_{a,b \in [k]} \max_{x_A \in [k]^{\text{card}(X_A)}} \log \frac{P(X_A = x_A|X_i = a, \theta)}{P(X_A = x_A|X_i = b, \theta)}. \tag{2}
\]

A higher max-influence means higher level of correlation between \( X_i \) and \( X_A \), and max-influence becomes 0 if \( X_i \) and \( X_A \) are independent. For simplicity, we would use \( e_{\theta} \) to denote \( e_{\{\theta\}} \).

In a Markov chain, the max-influence can be calculated exactly given the transition matrix \( P_\theta \) and initial distribution \( q_\theta \). It can also be approximated using properties of the stationary distribution and eigen-gap of the transition matrix if the Markov chain is irreducible and aperiodic. [10] shows the following upper bound of max-influence.

**Lemma 2.3:** For an irreducible and aperiodic Markov chain described by \( \theta = (q_\theta, P_\theta) \), \( P_\theta^k \) is the time reversal of \( P_\theta \). Let \( \pi_\theta \) be the stationary distribution of \( \theta \) and \( \pi_{\theta}^{\min} = \min_{x \in [k]} \pi_\theta(x) \) and let \( \theta^g = \min\{1 - |\lambda| : P_\theta P_\theta^g x = \lambda x, |\lambda| < 1\} \) be the eigen-gap of \( P_\theta P_\theta^* \). If \( \pi_\theta > 0 \), \( g_\theta > 0 \) and \( a, b \geq 2^{\log(1/\pi_{\theta}^{\min})} \), then for \( X_Q = \{X_{i-a}, X_{i+b}\} \),

\[
e_{\theta}(X_Q|X_i) \leq 2 \log \frac{\pi_{\theta}^{\min} + \exp(-g_\theta a/2)}{\pi_{\theta}^{\min} - \exp(-g_\theta b/2)} + \frac{2 \log(1/\pi_{\theta}^{\min})}{g_\theta}.
\]

To facilitate efficient search for an almost independent set, [10] then defines a Markov Quilt which takes into account the structure a Markov chain.

**Definition 2.4 (Markov Quilt):** A set of nodes \( X_Q, Q \subseteq [n] \) in a Markov chain \( X \) is a Markov Quilt for a node \( X_i \) if the following conditions hold:
1. Deleting \( X_Q \) partitions \( X \) into parts \( X_N \) and \( X_R \) such that \( X = X_N \cup X_Q \cup X_R \) and \( X_i \in X_N \).
2. For all \( x_R \in [k]^{\text{card}(X_R)} \), all \( x_Q \in [k]^{\text{card}(X_Q)} \) and for all \( a \in [k] \), \( P(X_R = x_R|X_Q = x_Q, X_i = a) = P(X_R = x_R|X_Q = x_Q) \).

Thus, \( X_R \) is independent of \( X_i \) conditioned on \( X_Q \).

Intuitively, \( X_R \) is a set of “remote” nodes that are far from \( X_i \), and \( X_N \) is the set of “nearby” nodes; \( X_N \) and \( X_R \) are separated by the Markov Quilt \( X_Q \).

A Markov Quilt \( X_Q \) (with corresponding \( X_N \) and \( X_R \)) of \( X_i \) is minimal if among all other Markov Quilts with the same nearby set \( X_N \), it has the minimal cardinality.

**Lemma 2.5:** In a Markov chain \( X = \{X_k\}_{k=1}^n \), the set of minimal Markov Quilts of a node \( X_i \)

\[
S_{Q,i} = \{ \{X_{i-a}, X_{i+b}\}, \{X_{i-a}, \}, \{X_{i+b}, \}, \emptyset \mid 1 \leq a \leq i - 1, 1 \leq b \leq T - i \}. \tag{5}
\]

That is, one node on its left and one node to its right can form a Markov Quilt for \( X_i \). Additionally, a Markov Quilt can also be formed by only one node \( X_{i-a} \) (or \( X_{i+b} \)), in which case \( X_N = \{X_j\}_{j=i-a+1}^T \) and \( X_R = \{X_j\}_{j=i+b-1}^T \); and the empty Markov Quilt is also allowed, with corresponding \( X_N \) as the whole chain and \( X_R \) as the empty set.

Now we restate the Markov Quilt Mechanism for Markov chain.
Algorithm 1 MQM(Dataset $D$, 1-Lipschitz query $F$, $\Theta$, privacy parameter $\epsilon$)

for all $\theta \in \Theta$ do
    for all $X_i$ do
        for all Markov Quilts $X_Q \in S_{Q,i}$ where $S_{Q,i}$ is in 5 do
            Calculate $\epsilon(\theta)(X_Q|X_i)$
            if $\epsilon(\theta)(X_Q|X_i) < \epsilon$ then
                $\sigma_i(X_Q) = \frac{\text{card}(X_Q)}{e^{\epsilon(\theta)(X_Q|X_i)}}$ //score of $X_Q$/
            else
                $\sigma_i(X_Q) = \infty$
            end if
        end for
        $\sigma^\theta_i = \min_{X_Q \in S_{Q,i}} \sigma_i(X_Q)$
    end for
    $\sigma^\theta_{max} = \max_i \sigma^\theta_i$
end for

$\sigma_{max} = \max_{\theta \in \Theta} \sigma^\theta_{max}$
return $F(D) + \sigma_{max} \cdot Z$, where $Z \sim \text{Lap}(1)$

For each node $X_i$, MQM searches over all of its Markov Quilts, finds the one with the least amount of noise needed, and finally adds an amount of noise that is enough to protect all nodes.

[10] shows that MQM guarantees $\epsilon$-Pufferfish privacy in the framework $(S, Q, \Theta)$ in Section II-A provided that $F$ is 1-Lipschitz. Note that any Lipschitz function can be scaled to 1-Lipschitz function.

Observe that Algorithm 1 does not specify how to compute max-influence. [10] proposes two versions of MQM—MQMExact which computes the exact max-influence using Definition 2.2, and MQMApprox which computes an upper bound of max-influence according to Lemma 2.3.

1) Previous Results on Composition: To design more sophisticated privacy preserving algorithms, we need to understand the privacy guarantee of the combination of two private algorithms, which is called composition.

There are two types of composition – parallel and sequential. The first describes the case where multiple privacy algorithms are applied on disjoint data sets, while the second describes the case where they are applied to the same data.

A major advantage of differential privacy is that it composes gracefully. [2] shows that applying $K$ differentially private algorithms, each with $\epsilon_k$-differential privacy, guarantees $\sum_k \epsilon_k$-differential privacy under parallel composition, and $\epsilon_k$-differential privacy under sequential composition. Better and more sophisticated composition results have been shown for approximate differential privacy [3], [6].

Unlike differential privacy, Pufferfish privacy does not always compose linearly [7]. However, we can still hope to achieve composition for special Pufferfish mechanisms or on special classes of data distributions $\Theta$.

[10] shows the following sequential composition for MQM on Markov chain.

Theorem 2.6: Let $\{F_k\}_{k=1}^K$ be a set of Lipschitz queries, $(S, Q, \Theta)$ be a Pufferfish framework as defined in Section II-
A, and $D$ be a database. Given fixed Markov Quilt sets $\{S_{Q,i}\}_{i=1}^n$ for all $X_i$, let $M_k(D)$ denote the Markov Quilt Mechanism that releases $F_k(D)$ with $\epsilon_k$-Pufferfish privacy under $(S, Q, \Theta)$ using Markov Quilt sets $\{S_{Q,i}\}_{i=1}^n$. Then releasing $(M_1(D), M_2(D), \ldots, M_K(D))$ guarantees $K \max_{k \in [K]} \epsilon_k$-Pufferfish privacy under $(S, Q, \Theta)$.

Notice that this result holds only when the same Markov Quilts are used for all releases. Moreover, the final privacy guarantee depends on the worst privacy guarantees $\max_k \epsilon_k$ over the $K$ releases. In practice, it might not be easy to enforce the MQM to use the same Markov Quilts at all releases; and if even one of the releases guarantees large $\epsilon_k$, the final privacy guarantee can be bad.

[10] does not provide any parallel composition result.

III. RESULTS

As discussed in the previous section, general Pufferfish mechanisms do not compose linearly. However, we can exploit the properties of data distributions – Markov chains, and properties of the specific Pufferfish mechanism – MQM to achieve parallel composition as well as sequential composition results that are better than [10].

A. Parallel Composition

Setup: Consider the Pufferfish framework $(S, Q, \Theta)$ as described in Section II-A. We can formulate parallel composition in the following way.

Suppose there exist $1 \leq T_1 < T_2 < T_3 < T_4 \leq T$. There are two subchains of the Markov chain $X^A = X[T_1:T_2]$ and $X^B = X[T_3:T_4]$; and correspondingly, let $S_A = S[T_1:T_2]$, $Q_A = Q[T_1:T_2]$ and $S_B = S[T_3:T_4]$, $Q_B = Q[T_3:T_4]$. Suppose Alice has access to subchain $X^A$ and wants to publish statistics $F_A$ while guaranteeing $\epsilon_A$-Pufferfish Privacy under framework $(S_A, Q_A, \Theta)$; and Bob has access to $X^B$ and wants to publish statistics $F_B$ while guaranteeing $\epsilon_B$-Pufferfish Privacy under framework $(S_B, Q_B, \Theta)$.

We further assume that both $F_A, F_B$ are Lipschitz.

Our goal is to determine what kind of Pufferfish privacy guarantee we can get if we release $(M_A(X^A), M_B(X^B))$.

1) A General Result for Markov Chains:

Theorem 3.1: Suppose $M_A, M_B$ are two mechanisms used by Alice and Bob such that $M_A(X^A)$ satisfies $\epsilon_A$-Pufferfish privacy under framework $(S_A, Q_A, \Theta)$ and $M_B(X^B)$ satisfies $\epsilon_B$-Pufferfish privacy under framework $(S_B, Q_B, \Theta)$. Then releasing $(M_A(X^A), M_B(X^B))$ guarantees $\max\{\epsilon_A + \epsilon_B, \epsilon_A + \epsilon_e(X_{T_2}[X_{T_3}]), \min(\epsilon_B + \epsilon_A + \epsilon_e(X_{T_2}[X_{T_3}]), \epsilon_B + \epsilon_e(X_{T_2}[X_{T_3}]))\}$-Pufferfish Privacy under framework $(S_A \cup S_B, Q_A \cup Q_B, \Theta)$.

Comparing with parallel composition for differential privacy, here we have the extra terms $\epsilon_e(X_{T_2}[X_{T_3}])$ and $\epsilon_e(X_{T_2}[X_{T_3}])$ which capture the correlation between $X_{T_2}$ and $X_{T_3}$ – the end point of the first subchain and the starting point of the second. This is to be expected, since there is correlation among states in the Markov chain. Intuitively, if the two subchains are close enough, releasing information on one can cause a privacy breach of the other. Note that as
differential privacy assumes independence between records, it does not suffer from privacy breach caused by correlation.

2) MQM on Markov Chains: Let $MQM(D, F, \epsilon, (S, Q, \Theta))$ denote the output of MQM on dataset $D$, query function $F$, privacy parameter $\epsilon$ and Pufferfish framework $(S, Q, \Theta)$. Suppose both Alice and Bob use MQMApprox

$MQM(X^A, F_A, \epsilon_A, (S_A, Q_A, \Theta))$ and $MQM(X^B, F_B, \epsilon_B, (S_B, Q_B, \Theta))$ respectively.

Before we establish a parallel composition result, we begin with a definition.

Definition 3.2: (Active Markov Quilt) Consider an instance of the Markov Quilt Mechanism $M$. We say that a Markov Quilt $X_Q$ (with corresponding $X_N$, $X_R$) for a node $X_i$ is active if $X_Q = \arg \min_{X_Q \in S_Q} \sigma_i(X_Q)$, and thus $\sigma_i(X_Q) = \sigma_i$. We denote this Markov Quilt as $X_{Q,i}$.

Theorem 3.3: Suppose we run MQMApprox to release

$MQM(X^A, F_A, \epsilon_A, (S_A, Q_A, \Theta)), MQM(X^B, F_B, \epsilon_B, (S_B, Q_B, \Theta))$. If the following conditions hold

1) for any $\theta \in \Theta$, there exists some $T_1 \leq i \leq T_2$ and $T_1 \leq i \leq T_4$ such that the active Markov Quilts of $X_i^A$ and $X_i^B$ are of the form $\{X_{i-a}^A, X_{i+b}^A\}$ and $\{X_{i-a}^B, X_{i+b}^B\}$ respectively for some $a, b, a', b'$.

2) $T_3 - T_2 \geq \max\{T_2 - T_1, T_4 - T_3\}$, i.e., $X^A, X^B$ are far from each other compared to their lengths,

then the release guarantees $\max(\epsilon_A, \epsilon_B)$-Pufferfish Privacy under the framework $(S_A \cup S_B, Q_A \cup Q_B, \Theta).

The main intuition is as follows. Note that we require the active Markov Quilt of some $X_j \in X^A$ to be of the form $\{X_{i-j-a}^A, X_{i+a+b}^A\}$. For any $X_i \in X^A$, the correction factor added to account for the effect of the nodes $\{X_k^A\}_{k \geq i + b}$ also automatically account for the effect of $X^B = X_{[T_3, T_4]}$, provided that $[T_3, T_4]$ does not overlap with $[i - a, i + b]$. This is ensured by the second condition in Theorem 3.3.

B. Sequential Composition

Consider the case when Alice and Bob have access to the entire Markov Chain $X = \{X_t\}_{t=1}^T$, and want to publish Lipschitz queries $F_A, F_B$ with Pufferfish parameters $\epsilon_A, \epsilon_B$ and Pufferfish framework $(S, Q, \Theta)$ as described in Section II-A.

1) General Results for Markov Chains: First, we show that any arbitrary Pufferfish mechanism does not compose linearly even when $\Theta$ consists of Markov chains.

Theorem 3.4: There exists a Markov chain $X$, a function $F$ and mechanisms $M_1, M_2$ such that both $M_1(X)$ and $M_2(X)$ guarantee $\epsilon$-Pufferfish privacy under framework $(S, Q, \Theta)$, yet releasing $(M_1(X), M_2(X))$ does not guarantee $2\epsilon$-Pufferfish privacy under framework $(S, Q, \Theta)$.

Now we show that arbitrary Pufferfish mechanisms compose with a correction factor that depends on the max-divergence between the joint and product distributions of $M_A(X)$ and $M_B(X)$. We define max-divergence first.

Definition 3.5 (max-divergence): Let $p$ and $q$ be two distributions with the same support. The max-divergence $D_\infty(p, q)$ between them is defined as:

$$D_\infty(X \parallel Y) = \sup_{x \in \text{support}(p)} \log \frac{p(x)}{q(x)}.$$ 

Now we can state the composition theorem.

Theorem 3.6: Suppose $\mathcal{M}_A, \mathcal{M}_B$ are two mechanisms used by Alice and Bob such that $\mathcal{M}_A(X)$ guarantees $\epsilon_A$-Pufferfish privacy under framework $(S, Q, \Theta)$ and $\mathcal{M}_B(X)$ guarantees $\epsilon_B$-Pufferfish privacy under framework $(S, Q, \Theta)$. If there exists $E$, such that for all $s_i \in S, \theta \in \Theta$,

$$D_\infty(\mathcal{M}_A(X), \mathcal{M}_B(X)|s_i, \theta) \parallel \mathcal{M}_A(X)|s_i, \theta) \mathcal{M}_B(X)|s_i, \theta) \leq E$$

then the releasing $(\mathcal{M}_A(X), \mathcal{M}_B(X))$ guarantees $(\epsilon_A + \epsilon_B + 2E)$-Pufferfish Privacy under framework $(S, Q, \Theta)$.

The max-divergence between the joint and product distributions of $\mathcal{M}_A(X)$ and $\mathcal{M}_B(X)$ measures the amount of dependence between the two releases. The more independent they are, the smaller the max-divergence would be and the stronger privacy the algorithm guarantees.

2) MQM on Markov Chains: We next show that we can further exploit the properties of MQM to provide tighter privacy guarantees.

Suppose Alice and Bob use MQM to achieve Pufferfish privacy under the same framework $(S, Q, \Theta)$. We show that when $\Theta$ consists of Markov chains, even if the two runs of MQM use different Markov Quilts, MQM still compose linearly. This result applies to both MQMExact and MQMApprox.

Theorem 3.7: Suppose releasing

$MQM(X, F_i, \epsilon_i, (S, Q, \Theta))$, guarantees $\epsilon_i$-Pufferfish privacy under framework $(S, Q, \Theta)$. Releasing

$MQM(X, F_1, \epsilon_1, (S, Q, \Theta)), MQM(X, F_2, \epsilon_2, (S, Q, \Theta))$ guarantees $(\epsilon_1 + \epsilon_2)$-Pufferfish privacy under framework $(S, Q, \Theta)$.

This result shows that MQM on Markov chain achieves the same composition guarantee as pure differential privacy. Comparing to the composition results provided in [10], i.e., Theorem 2.6, Theorem 3.7 provides better privacy guarantee under less restricted conditions. It does not require the same Markov Quilts to be used in the two runs of MQM. Moreover, the privacy guarantee is better when $\epsilon_k$’s are different $- \sum_k \epsilon_k$ as opposite to $K \max_k \epsilon_k$.

IV. CONCLUSION

In conclusion, motivated by emerging sensing applications, we study composition properties of Pufferfish, a form of inferential privacy, for certain kinds of time-series data. We provide both sequential and parallel composition results. Our results illustrate that while Pufferfish does not have strong composition properties in general, variants of the recently introduced Markov Quilt Mechanism that guarantees Pufferfish privacy for time series data, do compose well, and have strong composition properties comparable to pure
differential privacy. We believe that these results make these mechanisms attractive for practical time series applications.

ACKNOWLEDGMENT

We thank Joseph Geumlek, Sewoong Oh and Yizhen Wang for initial discussions. This work was partially supported by NSF under IIS 1253942, ONR under N00014-16-1-2616 and a Google Faculty Research Award.

REFERENCES

[1] Martín Abadi, Young-Deok Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pages 308–318. ACM, 2016.

[2] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In Theory of Cryptography, 2006.

[3] Cynthia Dwork, Guy N Rothblum, and Salil Vadhan. Boosting and differential privacy. In Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on, pages 51–60. IEEE, 2010.

[4] Arpita Ghosh and Robert Kleinberg. Inferential privacy guarantees for differentially private mechanisms. arXiv preprint arXiv:1603.01508, 2016.

[5] Xi He, Ashwin Machanavajjhala, and Bolin Ding. Blowfish privacy: tuning privacy-utility trade-offs using policies. In SIGMOD '14, pages 1447–1458, 2014.

[6] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. The composition theorem for differential privacy. In Francis Bach and David Blei, editors, Proceedings of the 32nd International Conference on Machine Learning, volume 37 of Proceedings of Machine Learning Research, pages 1376–1385, Lille, France, 07–09 Jul 2015. PMLR.

[7] Daniel Kifer and Ashwin Machanavajjhala. Pufferfish: A framework for mathematical privacy definitions. ACM Trans. Database Syst., 39(1):3, 2014.

[8] Changchang Liu, Supriyo Chakraborty, and Prateek Mittal. Dependence makes you vulnerable: Differential privacy under dependent tuples. In NDSS 2016, 2016.

[9] Shuang Song and Kamalika Chaudhuri. Composition properties of inferential privacy for time-series data. CoRR, abs/1707.02702, 2017.

[10] Shuang Song, Yizhen Wang, and Kamalika Chaudhuri. Pufferfish privacy mechanisms for correlated data. In Proceedings of the 2017 ACM International Conference on Management of Data, pages 1291–1306. ACM, 2017.

[11] Yonghui Xiao and Li Xiong. Protecting locations with differential privacy under temporal correlations. In Proceedings of the 22nd ACM SIGSAC CCS, pages 1298–1309. ACM, 2015.

APPENDIX

A. Parallel Composition Proofs

See full version [9] for the proof of Theorem 3.1.

Proof: (of Theorem 3.3) Denote the noises added by MQM for Alice and Bob by $Z_A, Z_B$ respectively. Consider any secret pair of the form $(X^A = a, X^B = b)$. We want to upper bound the following ratio for any $w_A, w_B, \theta, i, a, b$.

$$P(F_A(X^A) + Z_A = w_A, F_B(X^B) + Z_B = w_B | X^A = a, \theta)$$

By assumption, there exists some $X_j \in X^A$ whose active Markov Quilt is $X_{R,j} = \{(X_{j-a}, X_{j+a})\}$ with corresponding $X_{R,j}$ and $X_{N,j}$; and we have $\sigma^\theta_{\max} \geq \sigma_j^\theta = \text{card}(X_{N,j})/(\ell - e\theta_i |X_{R,j}|)$.

The main idea of the proof is that we can “borrow” the Markov Quilt of $X_j$ as the Markov Quilt for any $X_i \in X^A$ because doing so will not increase the noise scale $\sigma^\theta_{\max}$.

There are three cases:

1) If $i - a \geq 1$ and $i + b \leq T_2$, then let $X_Q = \{X_{i-a}, X_{i+a}\}$ (we omit the subscript $i$ for simplicity) with corresponding $X_R = \{X_k\}_{1 \leq k < i-a} \cup \{X_k\}_{i+a < k \leq T_2}$ and $X_N = \{X_k\}_{i-a < k < i+a}$.

2) If $i - a \geq 1$ and $i + b > T_2$, then let $X_Q = \{X_{i-a}, X_{i+a}\}$ with corresponding $X_R = \{X_k\}_{1 \leq k < i-a}$ and $X_N = \{X_k\}_{i-a < k < i+b}$.

3) If $i + b \leq T_2$ and $i - a \leq 0$, then let $X_Q = \{X_{i-b}\}$ with corresponding $X_N = \{X_k\}_{1 \leq k < i+b}$ and $X_R = \{X_k\}_{i+b < k \leq T_2}$.

Notice that when max-influence is approximated with Lemma 2.3, for any $i$ and $j$, we have $e\theta(X_{i-a}, X_{i+a}) = e\theta(\{X_{j-a}, X_{j+a}\})$, i.e., the max-influence is only affected by the relative distance between $X_i$ and its Markov Quilt.

Therefore, in the first two cases, we have $\sigma^\theta_j(X_{Q,j})$ since the max-influence and the size of nearby nodes are the same; in the last case, since $e\theta(X_Q | X_i) = e\theta(X_{Q,j} | X_j)$ and $\text{card}(X_Q) \leq \text{card}(X_{N,j})$, we have $\sigma^\theta_j(X_{Q,j}) \leq \sigma^\theta_j(X_{Q,j})$. Therefore we know that Lap($\sigma^\theta_{\max}$) suffices to protect $X_i$.

Let $X_{R,j} = X_R \cup X_Q$. We can split $X_{R,j}$ into two parts. $X_{mid} = \{X_j \in X_{R,j}, j > i\}$ which is closer to the middle of $X^A$ and $X^B$, and $X_{out} = \{X_j \in X_{R,j}, j < i\}$ which is closer to the boundary of the Markov chain. For the three cases respectively, we have

1) $X_{mid} = \{X_k\}_{1 \leq k < i-a}$ and $X_{out} = \{X_k\}_{1 \leq k < i-a}$.

2) $X_{mid} = \{X_{i+a}\}$ and $X_{out} = \{X_k\}_{1 \leq k < i-a}$.

3) $X_{mid} = \{X_k\}_{1 \leq k < i+b}$ and $X_{out} = \emptyset$.

By assumption, $X^A$ and $X^B$ are far enough, and thus $X_{i+a} \not\in X^B$ and $X_{mid} \cap X^B = \emptyset$.

Then we have

$$P(F_A(X^A) + Z_A = w_A, F_B(X^B) + Z_B = w_B | X^A = a, \theta) = P(F_A(X^A) + Z_A = w_A, F_B(X^B) + Z_B = w_B, X_{out} = x_{out}, X_{mid} = x_{mid}, X_i = a | x_{mid} = x_{mid}, \theta)$$

By Lemma 1.2, for any $a, b$ and and $x_{R,j}$.

$$P(F_A(X^A) + Z_A = w_A, X_{R,j} = x_{R,j} | X_i = a, \theta) < e^{\epsilon \theta}.$$

$$P(F_A(X^A) + Z_A = w_A, X_{R,j} = x_{R,j} | X_i = a, \theta)$$

$$P(F_A(X^A) + Z_A = w_A, X_{R,j} = x_{R,j} | X_i = a, \theta) < e^{\epsilon \theta}.$$
Therefore for any $X_{out}, X_{mid}$ we have

\[
P(\mathcal{F}_A(X^A) + Z_A = w_A, X_{out} = x_{out}, X_{mid} = x_{mid} | X_i = a, \theta) P(\mathcal{F}_B(B^B) + Z_B = w_B | X_{mid} = x_{mid}, \theta) \]

\[
P(\mathcal{F}_A(X^A) + Z_A = w_A, X_{out} = x_{out}, X_{mid} = x_{mid} | X_i = b, \theta) P(\mathcal{F}_B(B^B) + Z_B = w_B | X_{mid} = x_{mid}, \theta) \]

\[
\leq e^{c_A}.
\]

and therefore

\[
P(\mathcal{F}_A(X^A) + Z_A = w_A, \mathcal{F}_B(B^B) + Z_B = w_B | X_i = a, \theta) P(\mathcal{F}_A(X^A) + Z_A = w_A, X_{out} = x_{out}, X_{mid} = x_{mid} | X_i = b, \theta) P(\mathcal{F}_B(B^B) + Z_B = w_B | X_{mid} = x_{mid}, \theta) \]

\[
\leq \max_{x_{out}, x_{mid}} e^{c_A} = e^{c_A}.
\]

When the secret pair is of the form $(X^A = a, X^B = b)$, similar argument applies and the bound is $e^{c_B}$.

Therefore for Pufferish parameter $(\mathcal{S}_A \cup \mathcal{S}_B, Q \cup Q_B, \Theta)$, $(\text{MQM}(X^A, \mathcal{F}_A, \mathcal{A}_i, (S_A, Q_A, \Theta)), \text{MQM}(X^B, \mathcal{F}_B, \mathcal{B}_i, (S_B, Q_B, \Theta)))$ guarantees $\max(\epsilon_A, \epsilon_B)$-Pufferish Privacy.

B. Sequential Composition Proofs

See full version [9] for the proofs of Theorem 3.4 and 3.6.

Proof: (of Theorem 3.7) Consider any secret pair $(X_i = a, X_i = b)$ and any $\theta \in \Theta$. Let $X_1 = \{X_{i=1}, X_{i=1+1}\}$ be active Markov Quilt in the first publication (with $X_N = \{X_k\}_{i=1,k<i} \cup \{X_k\}_{i=1,k>i}$, $X_R = \{X_k\}_{i=1,k<i}$ be in the second (with $X_R = \{X_k\}_{i=1,k<i} \cup \{X_k\}_{i=1,k>i}$). Denote $X_R \cup X_Q$ as $X_{R\cup Q}$ for $j = 1, 2, \ldots, N$. Let $X_{R\cup Q} = \bigcup_{j=1}^{\infty} X_j \cup X_Q$, which is guaranteed to be non-empty since it contains at least $X_i$. Let $X_Q = \{X_{i=min(a_i,b_i)} \cup X_{i=max(b_i,a_i)}\}$, i.e., we pick from $X_{i=1}$ to $X_{i=1}$ and $X_{i=1}$ to $X_{i=1}$ (and also $X_{i=1}$ to $X_{i=1}$) the ones that are closer to $X_i$. Notice that this is a valid Markov Quilt of $X_i$, with corresponding secret node $X_i$ and remote set $X_{R\cup Q} \setminus X_Q$.

Let $Z_1, Z_2$ denote the Laplace noises added by MQM for the two releases respectively. For simplicity, let the $\theta$ term in the probabilities and assume all $X_i$s are distributed according to $\theta$. Then for any $w_1, w_2$, we have

\[
p(F_1(X) + Z_1 = w_1, F_2(X) + Z_2 = w_2 | X_i = a) \]

\[
p(F_1(X) + Z_1 = w_1, F_2(X) + Z_2 = w_2 | X_i = b) \]

\[
x_{R\cup Q} = x_{R\cup Q} | X_i = a) dx_{R\cup Q} \]

\[
x_{R\cup Q} = x_{R\cup Q} | X_i = b) dx_{R\cup Q} \]

\[
\leq \max_{x_{out}, x_{mid}} e^{c_A} \leq e^{c_A}.
\]

First, consider the first ratio in (6). Let $X_{N\setminus \{i\}} = X_{N\setminus \{i\}} \setminus X_i$ denote all “nearby” nodes except for $X_i$. Let $p_F(x_1, x_{N\setminus \{i\}}, x_{R\cup Q})$ denote the function value of $F_i$ when $x_i = x_{out}$, $X_{N\setminus \{i\}} = x_{N\setminus \{i\}}$ and $X_{R\cup Q} = x_{R\cup Q}$. We have

\[
p(F_i(X) + Z_1 = w_1, F_2(X) + Z_2 = w_2 | X_{R\cup Q} = x_{R\cup Q}, X_i = a) \]

\[
p(F_i(X) + Z_1 = w_1, F_2(X) + Z_2 = w_2 | X_{R\cup Q} = x_{R\cup Q}, X_i = b) \]

\[
x_{R\cup Q} = x_{R\cup Q} | X_i = x_{N\setminus \{i\}} \}

\[
\leq \max_{x_{out}, x_{mid}} e^{c_A} \leq e^{c_A}.
\]

Also, notice that $p(X_{N\setminus \{i\}} = x_{N\setminus \{i\}} | X_{R\cup Q} = x_{R\cup Q}, X_i = x_{N\setminus \{i\}}$ are probability distributions, and $X_{N\setminus \{i\}}$ is the random variable we are integrating over on. Therefore

\[
p(F_i(X) = x_{N\setminus \{i\}} | X_{R\cup Q} = x_{R\cup Q}, X_i = x_{N\setminus \{i\}} \}

\[
\leq \max_{x_{out}, x_{mid}} e^{c_A} \leq e^{c_A}.
\]
Then we consider the second ratio in (6).
\[
\frac{p(X_{R|Q} = x_{R|Q}|X_i = a)}{p(X_{R|Q} = x_{R|Q}|X_i = b)}
\]
\[= p(X_R = x_R|X_Q = x_Q, X_i = a)p(X_Q = x_Q|X_i = a)\]
\[p(X_R = x_R|X_Q = x_Q, X_i = b)p(X_Q = x_Q|X_i = b)
\]
\[= p(X_R = x_R|X_Q = x_Q)p(X_Q = x_Q|X_i = a)
\]
\[p(X_R = x_R|X_Q = x_Q)p(X_Q = x_Q|X_i = b)
\]
\[= p(X_Q = x_Q|X_i = a)\]
\[p(X_Q = x_Q|X_i = b)\]
\[
\leq e^{\epsilon_0(x_Q|x_i)},
\]
where the second equality comes from the fact that \(X_Q\) is a valid Markov Quilt of \(X_i\), and the last inequality follows from the definition of max-influence.

Now we show that
\[
e_{\theta}(X_Q^2|X_i) \geq e_{\theta}(X_Q|X_i).
\]

If \(X_Q\) is equal to one of \(X_Q^1\) and \(X_Q^2\), the inequality hold trivially. Otherwise, according to Lemma 1.1, we have
\[
e_{\theta}(\{x_s, x_b\}|X_i) + e_{\theta}(\{x_s^2, x_b^2\}|X_i)
\]
\[
\geq e_{\theta}(\{\max(a, a^2), \max(b, b^2)\}|X_i)
\]
\[
\leq e_{\theta}(\{\max(a, a^2), \min(b, b^2)\}|X_i).
\]

Combining with (7), we know that (6) is upper bounded by
\[
e^{\epsilon_1 + \epsilon_2 - e_{\theta}(x_Q^1|X_i) - e_{\theta}(x_Q^2|X_i)} e^{\epsilon_0(x_Q|x_i)} \leq e^{\epsilon_1 + \epsilon_2}.
\]

C. Other Lemmas and Proofs

Lemma 1.1: Let \(X_S\) and \(X_R\) be two sets of nodes in a Bayesian network such that \(X_S \subseteq X_R\). For any \(\theta\), we have
\[
e_{\theta}(X_S|X_i) \leq e_{\theta}(X_R|X_i).
\]

Proof: Let \(T = R \setminus S\), for any \(\theta\) we have
\[
\exp(e_{\theta}(X_R|X_i))
\]
\[= \max_{x_R, x_T, x_i'} p(X_R = x_R|X_i = x_i, \theta)
\]
\[p(X_R = x_R|X_i = x_i, \theta)
\]
\[= \max_{x_S, x_T, x_i'} p(X_S = x_S, X_T = x_T|X_i = x_i, \theta)
\]
\[p(X_S = x_S, X_T = x_T|X_i = x_i, \theta)
\]

Then we have
\[
e_{\theta}(X_S|X_i, \theta)
\]
\[= \max_{x_S, x_T, x_i'} \sum_{x_T = x_T'} p(X_S = x_S, X_T = x_T'|X_i = x_i, \theta)
\]
\[\sum_{x_T} \exp(e_{\theta}(X_R|X_i)) p(X_R = x_R)
\]
\[\sum_{x_T} \exp(e_{\theta}(X_R|X_i)) p(X_R = x_R)
\]
\[\leq \max_{x_S, x_T, x_i'} \sum_{x_T} p(X_S = x_S, X_T = x_T'|X_i = x_i, \theta)
\]
\[= \exp(e_{\theta}(X_R|X_i))
\]
where the inequality is from the definition of max-influence.

Since this hold for all \(\theta \in \Theta\), we have \(e_{\theta}(X_S|X_i, \theta) \leq e_{\theta}(X_R|X_i, \theta)\).

Here we also prove a useful lemma which is similar to the privacy guarantee of MQM proved in [10].

Lemma 1.2: For any secret pair \((X_i = a, X_i = b) \in \mathcal{Q}\) and any \(\theta \in \Theta\), let \(X_Q\) be the Markov Quilt for \(X_i\) which has the minimum score \(\sigma(x_Q)\), and suppose that deleting \(X_Q\) breaks up the underlying Bayesian network into \(X_N\) and \(X_R\) where \(X_i \in X_N\). Then for any \(w\) and any realization \(x_{R|Q}\) of \(X_R|Q\),
\[
p(F(X) + \sigma_{\max} \cdot Z = w, X_{R|Q} = x_{R|Q}|X_i = a, \theta)
\]
\[p(F(X) + \sigma_{\max} \cdot Z = w, X_{R|Q} = x_{R|Q}|X_i = b, \theta)\]
\[\leq e^{r'.}
\]

Proof: Pick a secret pair \((X_i = a, X_i = b) \in \mathcal{Q}\) and any \(\theta \in \Theta\). Let \(X_Q\) be the Markov Quilt for \(X_i\) which has the minimum score \(\sigma(x_Q)\), and suppose that deleting \(X_Q\) breaks up the underlying Bayesian network into \(X_N\) and \(X_R\) where \(X_i \in X_N\).

For any \(w\) and any realization \(x_{R|Q}\) of \(X_R|Q\), we can write
\[
p(F(X) + \sigma_{\max} \cdot Z = w, X_{R|Q} = x_{R|Q}|X_i = a, \theta)
\]
\[p(F(X) + \sigma_{\max} \cdot Z = w, X_{R|Q} = x_{R|Q}|X_i = b, \theta)\]
\[\leq e^{r'}.\]

Consider the first ratio of (9),
\[
p(F(X) + \sigma_{\max} Z = w|X_i = a, \theta, X_{R|Q} = x_{R|Q}, \theta)
\]
\[p(F(X) + \sigma_{\max} Z = w|X_i = b, \theta, X_{R|Q} = x_{R|Q}, \theta)
\]
\[\leq e^{r'}.\]

Since \(F\) is 1-Lipschitz, when \(X_{R|Q}\) is fixed, \(F(X)\) can vary by at most \(\text{card}(X_N)\) (potentially when all the variables in \(X_N\) change values). Since \(\sigma_{\max} \geq \frac{\text{card}(X_N)}{e^{\epsilon_0(x_Q|x_i)}}\) for any \(X_i\) with its best Markov Quilt \(X_Q\), and \(Z \sim \text{Lap}(1)\), we know that the above ratio is upper bounded by
\[
e^{r}\].

Then consider the second part of (9). We have
\[
p(X_{R|Q} = x_{R|Q}|X_i = a, \theta)
\]
\[p(X_{R|Q} = x_{R|Q}|X_i = b, \theta)
\]
\[\leq e^{r}\].

Since \(X_Q\) is a Markov Quilt for \(X_i\) and \(X_i \notin X_R\), we have \(p(X_R|X_Q, X_i = a, \theta) = p(X_R|X_Q, X_i = b, \theta)\); moreover, by definition of max-influence, \(\frac{p(X_R=x_Q|X_i=a,\theta)}{p(X_R=x_Q|X_i=b,\theta)} \leq e^{\epsilon_0(x_Q|x_i)}\). Therefore the above ratio is upper bounded by
\[
e^{\epsilon_0(x_Q|x_i)}.
\]

Combining the two ratios together, we can conclude that for any \(w\) and any secret pair \((s^a_i, s^b_i)\),
\[
p(F(X) + \sigma_{\max} \cdot Z = w, X_{R|Q} = x_{R|Q}|X_i = a, \theta)
\]
\[p(F(X) + \sigma_{\max} \cdot Z = w, X_{R|Q} = x_{R|Q}|X_i = b, \theta)\]
\[\leq e^{r}'.\]