On the Simulatability Condition in Key Generation Over a Non-authenticated Public Channel

Wenwen Tu and Lifeng Lai

Abstract—Simulatability condition is a fundamental concept in studying key generation over a non-authenticated public channel. Using this condition, Maurer and Wolf showed a remarkable “all or nothing” result: if the simulatability condition does not hold, the key capacity over the non-authenticated public channel will be the same as that of the case with a passive Eve, while the key capacity over the non-authenticated channel will be zero if the simulatability condition holds. However, several questions remain open so far: 1) For a given joint probability mass function (PMF), are there polynomial complexity algorithms for checking whether the simulatability condition holds or not? 2) If the simulatability condition holds, are there efficient algorithms for finding the corresponding attack strategy? and 3) How sensitive is this condition on the knowledge about Eve’s observations? In this paper, we fully answer these open questions. In particular, for a given joint PMF, we construct a linear programming (LP) problem and show that the simulatability condition holds if and only if the optimal value obtained from the constructed LP is zero. In addition, we construct another LP and show that the minimizer of the newly constructed LP is a valid attack strategy. Both LPs can be solved with a polynomial complexity. We further show that the simulatability condition is not sensitive on the knowledge about Eve’s observations.

Index Terms—Complexity, Key generation, Linear programming, Non-authenticated channel, Simulatability condition.

I. INTRODUCTION

Simulatability condition, introduced by Maurer and Wolf in [2]–[5], is a fundamental concept in studying the key generation via public discussion through a non-authenticated public channel, in which Eve is active and can intercept, modify or falsify any message exchanged through the public channel. Using this condition, Maurer and Wolf established a remarkable “all or nothing” result. In particular, they showed that for the secret key generation via a non-authenticated public channel with two legitimate terminals in the presence of an active adversary: 1) if the simulatability condition holds, the two legitimate terminals will not be able to establish a secret key, and hence the key capacity is 0; and 2) if the simulatability condition does not hold, the two legitimate terminals can establish a secret key and furthermore the key capacity will be the same as that of the case when Eve is passive considered in [6], [7].

Due to its importance, there have been significant efforts in designing efficient algorithms to check whether the simulatability condition holds or not for a given joint probability mass function (PMF). Using ideas from mechanical models, [4] made significant progress. In particular, [4] proposed to represent PMFs as mass constellations in a coordinate and designed a low-complexity algorithm to check a related concept called “more centered” condition. For some important special cases, which will be described precisely in Section II, [4] showed that the “more centered” condition is necessary and sufficient for the simulatability condition for these special cases. However, in the general case, the “more centered” condition is a necessary but not sufficient condition. As the result, despite the significant progress, [4] left the following questions regarding the simulatability condition for the general case as open questions:

1) For a given joint PMF, are there efficient algorithms (polynomial complexity algorithms) for checking whether the simulatability condition holds or not?

2) If the simulatability condition holds, are there efficient algorithms for finding the corresponding Eve’s attack strategy?

3) Suppose the PMF is not exactly known, how sensitive is the condition on the modeling uncertainty?

In this paper, we fully answer these open questions.

To answer the first open question, we construct a linear programming (LP) problem and show that the simulatability condition holds if and only if the optimal value obtained from this LP is zero. Since there exists polynomial complexity algorithms for solving LP problems [8]–[10], we thus find a polynomial complexity algorithm for checking the simulatability condition for a general PMF.

To answer the second open question, we construct another LP and show that the minimizer of this LP is a valid attack strategy. The proposed approach is very flexible in the sense that one can simply modify the cost function of the constructed LP to obtain different attack strategies. All these optimization problems with different cost functions can be solved with a polynomial complexity.

To answer the third open question, we show that if the simulatability condition does not hold for a given PMF, then the simulatability condition does not hold if the PMF related to Eve’s observation is changed up to a certain threshold. We fully characterize this threshold.

The remainder of the paper is organized as follows. In Section II, we introduce some preliminaries and the problem setup. In Section III, we present our main results. In Section IV, we present an approach to further reduce the computational
complexity. In Section V, we focus on the sensitivity analysis. In Section VI, we offer our concluding remarks.

II. PRELIMINARIES AND PROBLEM SETUP

Let $X = \{1, \cdots, |X|\}$, $Y = \{1, \cdots, |Y|\}$ and $Z = \{1, \cdots, |Z|\}$ be three finite sets. Consider three correlated random variables $(X, Y, Z)$, taking values from $X \times Y \times Z$, with joint PMF $P_{XYZ}$, the simulatability condition is defined as follows:

**Definition 1.** ([2]) For a given $P_{XYZ}$, we say $X$ is simulatable by $Z$ with respect to $Y$, denoted by $\text{Sim}_Y(Z \rightarrow X)$, if there exists a conditional PMF $P_{\bar{X}|Z}$ such that $P_{Y\bar{X}} = P_{YX}$, with

$$P_{Y\bar{X}}(y, x) = \sum_{z \in Z} P_{YZ}(y, z) \cdot P_{\bar{X}|Z}(x|z), \quad (1)$$

in which $P_{YX}$ and $P_{YZ}$ are the joint PMFs of $(Y, X)$ and $(Y, Z)$ under $P_{XYZ}$ respectively.

One can also define $\text{Sim}_X(Z \rightarrow Y)$ in the same manner. This concept of simulatability, first defined in [2], is a fundamental concept in the problem of secret key generation over a non-authenticated public channel [3]–[5], in which two terminals Alice and Bob would like to establish a secret key over a non-authenticated public channel [3]–[5], in which two terminals Alice and Bob can discuss with each other via a public channel, where the simulatability condition holds, the key rate will be zero. On the other hand, if $\text{Sim}_X(Z \rightarrow Y)$ holds or not? If $\text{Sim}_Y(Z \rightarrow X)$ holds, then $\text{Sim}_X(Z \rightarrow Y)$ holds or not? One can also define $\text{Sim}_Y(Z \rightarrow X)$, which is a fundamental concept in the problem of secret key generation over a non-authenticated public channel [3]–[5].

### Results

**Theorem 1.** ([3]) If $\text{Sim}_Y(Z \rightarrow X)$ or $\text{Sim}_X(Z \rightarrow Y)$, then $S^*(X; Y|Z) = 0$. Otherwise, $S^*(X; Y|Z) = S(X; Y|Z)$.

This significant result implies that, if the simulatability condition does not hold, one can generate a key with the same rate as if Eve were passive. On the other hand, if the simulatability condition holds, the key rate will be zero.

### Main Results

In this paper, we focus on $\text{Sim}_Y(Z \rightarrow X)$. The developed algorithm can be easily modified to check $\text{Sim}_X(Z \rightarrow Y)$. We rewrite (1) in the following matrix form

$$C = AQ,$$  \hspace{1cm} (5)

in which $C = [c_{ij}]$ is a $|Y| \times |X|$ matrix with $c_{ij} = P_{YX}(i, j)$, $A = [a_{ik}]$ is a $|Y| \times |Z|$ matrix with $a_{ik} = P_{YZ}(i, k)$, and $Q = [q_{kj}]$ is a $|Z| \times |X|$ matrix with $q_{kj} = P_{X|Z}(j|k)$ if such $P_{X|Z}$ exists.

Checking whether $\text{Sim}_Y(Z \rightarrow X)$ holds or not is equivalent to checking whether there exists a transition matrix $Q$ such that...
that (5) holds. As \( Q \) is a transition matrix, its entries \( q_{kj} \)'s must satisfy

\[
q_{kj} \geq 0, \quad \forall k \in [1 : |Z|], j \in [1 : |X|],
\]

(6)

\[
\sum_{j=1}^{|X|} q_{kj} = 1, \quad \forall k \in [1 : |Z|].
\]

(7)

We note that if \( q_{kj} \)'s satisfy (6) and (7), they will automatically satisfy \( q_{kj} \leq 1 \). Hence, we don’t need to state this requirement here.

If there exists at least one transition matrix \( Q \) satisfying (5), (6) and (7) simultaneously, we can conclude that the two main steps: 1) whether the system is consistent or not; 2) if it is consistent, whether there exists a nonnegative solution or not. Checking the consistency of (15) is straightforward: a necessary and sufficient condition for a system of non-homogenous linear equations to be consistent is

\[
\text{Rank}(A_c) = \text{Rank}((A_c|c)),
\]

(16)

where \((A_c|c)\) is the augmented matrix of \( A_c \). If (16) is not satisfied, it can be concluded that \( \text{Sim}_Y(Z \rightarrow X) \) does not hold. If (16) is satisfied, we need to further check whether there exists a nonnegative solution to (15) or not.

To proceed further, we will need the following definition of generalized inverse (g-inverse) of a matrix \( G \).

**Definition 2.** ([11]) For a given \( m \times n \) real matrix \( G \), an \( n \times m \) real matrix \( G^g \) is called a g-inverse of \( G \) if

\[ GG^g G = G. \]

The g-inverse \( G^g \) is generally not unique (If \( n = m \) and \( G \) is full rank, then \( G^g \) is unique and equal to the inverse matrix \( G^{-1} \)). A particular choice of g-inverse is called the Moore-Penrose pseudoinverse \( G^+ \), which can be computed using multiple different approaches. One approach is to use the singular value decomposition (SVD): by SVD, for a given \( G \) and its SVD decomposition

\[ G = U \Sigma V^T, \]

then, \( G^+ \) can be obtained as

\[ G^+ = V \Sigma^+ U^T, \]

in which \( \Sigma^+ \) is obtained by taking the reciprocal of each non-zero element on the diagonal of the diagonal matrix \( \Sigma \), leaving the zeros in place. One can easily check that the Moore-Penrose pseudoinverse \( G^+ \) obtained by SVD satisfies the g-inverse matrix definition and hence is a valid g-inverse.

With the concept of g-inverse, we are ready to state our main result regarding the first open question.

**Theorem 2.** Let \( h^g \) be any given g-inverse of \( A \) (e.g., it can be chosen as the Moore-Penrose pseudoinverse \( A^+ \)), and \( h^* \) be obtained by the following LP

\[ h^* = \min_t \{ t^T A^g c \}, \]

s. t. \[ t \succeq 0, \]

\[ (I - A^g A)^T t = 0. \]

Then \( \text{Sim}_Y(Z \rightarrow X) \) holds, if and only if \( h^* = 0 \) and (16) holds.

**Proof:** If (16) does not hold, then there is no solution to (15), and hence \( \text{Sim}_Y(Z \rightarrow X) \) does not hold.

In the remainder of the proof, we assume that (16) holds. If (16) holds, the general solution to (15) can be written in the following form (see, e.g., Theorem 2 a.d) of [12])

\[ q = h^g c + (h^g A - I) p, \]

(20)

in which \( h^g \) can be any given g-inverse of \( A_c \), and \( p \) is an arbitrary length-\( n \) vector.
As the result, the problem of whether there exists a nonnegative solution to (15) (i.e., \(q \succeq 0\)) is equivalent to the problem of whether there exists a solution \(p\) for the following system defined by

\[
(I - A^gA)p \leq A^g c. \tag{21}
\]

To check whether the system defined by (21) has a solution, we use Farkas’ lemma, a fundamental lemma in linear programming and related area in optimization. For completeness, we state the form of Farkas’ lemma used in our proof in Appendix A. To use Farkas’ lemma, we first write a LP related to the system defined in (21)

\[
\begin{align*}
  h^* &= \min_t \{ t^T A^g c \}, \\
  \text{s.t.} \quad & t \geq 0, \\
  & (I - A^gA)^T t = 0.
\end{align*}
\]

The above LP is always feasible since \(t = 0\) is a vector that satisfies the constraints, which results in \(t^T A^g c = 0\). Hence the optimal value \(h^* \leq 0\). Using Farkas’ lemma, we have that (21) has a solution if and if \(h^* = 0\). More specifically, if \(h^* = 0\), then there exists at least a solution \(p\) for (21), which further implies that there is a nonnegative solution to (15), and hence Sim\(_Y\)(\(Z \to X\)) holds. On the other hand, if \(h^* < 0\), then there is no solution \(p\) for (21), which further implies that there is no nonnegative solution to (15), and hence Sim\(_Y\)(\(Z \to X\)) does not hold.

As mentioned above, if \(\text{Rank}(A) = m = n\) holds, then \(A^g = A^{-1}\) is unique. For other cases, \(A^g\) might not be unique. One may wonder whether different choices of \(A^g\) will affect the result in Theorem 2 or not. The following proposition answers this question.

**Proposition 1.** Different choices of \(A^g\) will not affect the result on whether \(h^*\) equals 0 or not.

**Proof:** Let \(A^g_1\) and \(A^g_2\) be two different \(g\)-inverses of \(A\), and let \(h^*_1\) and \(h^*_2\) be the values obtained using \(A^g_1\) and \(A^g_2\) in (19) respectively. It suffices to show that if \(h^*_1 = 0\), then \(h^*_2 = 0\).

Assuming that \(h^*_1 = 0\), then there exists a vector \(p_1\) satisfying \((I - A^g_1A)p_1 \preceq A^g_1c\), we will show that there exists a vector \(p_2\) satisfying \((I - A^g_2A)p_2 \preceq A^g_2c\), which then implies \(h^*_2 = 0\).

First, we know that \(A^g_1c\) and \(A^g_2c\) are two solutions to the system \(Aq = c\), which can be easily verified by setting \(A^g\) as \(A^g_1\) and \(A^g_2\) in (20) respectively and setting \(p = 0\). This implies that

\[
A(A^g_2 - A^g_1)c = 0, \tag{22}
\]

and hence \(A^g_2c - A^g_1c\) is a solution to the system \(Aq = 0\).

Second, we know that any solution to the system \(Aq = 0\) can be written in the form \((I - A^gA)p\) [12]. As \(A^g_2c - A^g_1c\) is a solution to system \(Aq = 0\), there must exist a \(p_0\) such that

\[
(I - A^g_2A)p_0 = A^g_2c - A^g_1c. \tag{23}
\]

In addition, it is easy to check that \((I - A^g_1A)p_1 + (I - A^g_2A)p_0\) is also a solution to the system \(Aq = 0\). Thus, there exists a \(p_2\) such that

\[
(I - A^g_2A)p_2 = (I - A^g_1A)p_1 + (I - A^g_2A)p_0. \tag{24}
\]

Plugging (23) into (24), we have

\[
(I - A^g_2A)p_2 = (I - A^g_1A)p_1 + (I - A^g_2A)p_0
= (I - A^g_1A)p_1 + A^g_2c - A^g_1c
\leq A^g_2c, \tag{25}
\]

in which the last inequality comes from the assumption that \((I - A^g_1A)p_1 \preceq A^g_1c\). Hence, we have found a \(p_2\), such that \((I - A^g_2A)p_2 \preceq A^g_2c\). This implies that \(h^*_2 = 0\).

**Remark 1.** The proposed algorithm for checking whether Sim\(_Y\)(\(Z \to X\)) holds or not has a polynomial complexity. Among all operations required, computing the \(g\)-inverse and solving the LP defined by (19) require most computations. The complexity to obtain \(A^g\) is of order \(O(n^3)\) [13]. Furthermore, there exists polynomial complexity algorithms to solve the LP defined by (19). For example, [9] provided an algorithm to solve LP using \(O(n^3 L)\) operations, where \(L\) is number of binary bits needed to store input data of the problem (one can refer to Chapter 8 in [10] for more details about the complexity of algorithms for solving LP). Hence, the total operations of our algorithm for checking Sim\(_Y\)(\(Z \to X\)) is of order \(O(n^3 L)\). In addition, we note that we can terminate the LP algorithm earlier once the algorithm finds a \(t\) such that \(t^T A^g c < 0\), as this indicates that \(h^* < 0\). This can potentially further reduce the computational complexity.

Thus, we can conclude that the proposed algorithm can check whether Sim\(_Y\)(\(Z \to X\)) holds or not with a polynomial complexity. Algorithm 1 summarizes the main steps involved in our algorithm. In the following algorithm, we use \(Res = 0\) to denote that Sim\(_Y\)(\(Z \to X\)) does not hold and \(Res = 1\) to denote that Sim\(_Y\)(\(Z \to X\)) holds.

In the following, we provide our answer to the second open question, i.e., if Sim\(_Y\)(\(Z \to X\)) holds, how to find \(P_{X\mid Z}\) efficiently.

**Theorem 3.** Let \(e\) be any \(n \times 1\) vector with \(e \succ 0\), and \(q^*\) be the obtained from the following LP:

\[
\begin{align*}
  \min_{q} \quad & f(q) = e^T q, \\
  \text{s.t.} \quad & q \succeq 0, \\
  & Aq = c.
\end{align*}
\]

If Sim\(_Y\)(\(Z \to X\)) holds, then \(Q^* = \text{Reshape}(q^*, [\mid X\mid, \mid Z\mid])^T\) is a valid choice for \(P_{X\mid Z}\).

**Proof:** By assumption, Sim\(_Y\)(\(Z \to X\)) holds, which implies that the system defined by (15) is consistent and it
Algorithm 1 Checking Sim$_Y(Z \rightarrow X)$

1: **Input:** PMF $P_{X|Z}$;

2: **Initiate:**
3: a. Calculate matrices $A$ and $C$;
4: b. Construct $c$ and $A$ using (10) and (11) respectively;
5: c. Set $Res = 0$;

6: if $(\text{Rank}(A) \neq \text{Rank}(A|c))$ then
7: \hspace{1em} break;
8: else
9: \hspace{1em} d. Find a $\tilde{A}$, and calculate $\tilde{A}^g c, I - \tilde{A}^g \tilde{A}$;
10: \hspace{1em} e. Solve LP (19) and obtain $h^*$;
11: \hspace{1em} if $(h^* = 0)$ then
12: \hspace{2em} $Res = 1$;
13: else
14: \hspace{2em} break;
15: end if
16: end if

17: **Output:** $Res$.

Having nonnegative solutions. Hence, the following LP is feasible

$$\begin{align*}
\min_{q} & \quad e^T q \\
\text{s.t.} & \quad q \succeq 0, \\
& \quad \tilde{A} q = c,
\end{align*}$$

where $e \succ 0$. Hence, the minimizer $q^*$ is nonnegative and satisfies $\tilde{A} q^* = c$. We can then reshape $q^*$ into matrix $Q^*$ (see (12)). $Q^*$ is a valid choice for $P_{X|Z}$.

**Remark 2.** Since finding a suitable $P_{X|Z}$ using our approach is equivalent to solving a LP, the complexity is of polynomial order.

**Remark 3.** For a given distribution $P_{X|Z}$, there may be more than one possible $P_{X|Z}$ such that (1) holds. Different choices of $e$ in (27) give different values for $P_{X|Z}$.

**Remark 4.** The objective function $f(q)$ can be further modified to satisfy various design criteria of Eve. For example, let

$$\tilde{q} = \text{Vec}(\tilde{Q}[\tilde{q}_{kj}])^T$$

with $\tilde{q}_{kj} = P_{X|Z}(k|j)$, then setting

$$f(q) = ||q - \tilde{q}||^2_2$$

will minimize the amount of changes in the conditional PMF in the $l_2$ norm sense. This is a quadratic programming, which can still be solved efficiently.

IV. COMPLEXITY REDUCTION

In Proposition 1, we show that different choices of $A^g$ will not affect the result on whether $h^*$ equals zero or not. However, different choices of $A^g$ may affect the amount of computation needed. Primal-dual path-following method is one of the best methods for solving LP of the following form [10]:

$$\begin{align*}
\min_{t} & \quad t^T b \\
\text{s.t.} & \quad t \geq 0, \\
& \quad B t = d,
\end{align*}$$

in which $B$ is a matrix of size $m \times n$. The complexity is related to the size of $B$. In particular, in terms of $m$ and $n$, the complexity is $O((nm^2 + n L m))$ [14], [15]. In LP (19) constructed in the proof of Theorem 2, $B = (I - A^g A)^T$, which is an $n \times n$ matrix, and hence the complexity is $O(n^3 L)$ as mentioned in Section III.

In the following, we show that if we choose the g-inverse of $A$ to be $A^+$, the Moore-Penrose inverse, the problem size can be reduced by some further transformations. Let the SVD of $A$ be $U \Sigma V^T$. Then $A^+ = V \Sigma^+ U^T$. Suppose rank($P_{m \times n}$) = $r$ and set $s = n - r$. We have

$$A^+ A = V \Sigma^+ U^T U \Sigma V^T = V \begin{bmatrix} I_r & 0_{r \times s} \\ 0_{s \times r} & 0_{s \times s} \end{bmatrix} V^T.$$  (29)

As discussed in the proof of Theorem 2, checking Sim$_Y(Z \rightarrow X)$ holds or not is equivalent to checking whether

$$(I - A^+ A) p \preceq A^+ c$$  (30)

has a solution or not. We now perform some transformations on (30). First we have

$$I - A^+ A = V \begin{bmatrix} I_r & 0_{r \times s} \\ 0_{s \times r} & I_s \end{bmatrix} V^T - V \begin{bmatrix} I_r & 0_{r \times s} \\ 0_{s \times r} & 0_{s \times s} \end{bmatrix} V^T$$

$$= V \begin{bmatrix} 0_{r \times r} & 0_{r \times s} \\ 0_{s \times r} & I_s \end{bmatrix} V^T.$$  (31)

Hence, (30) is equivalent to

$$V \begin{bmatrix} 0_{r \times r} & 0_{r \times s} \\ 0_{s \times r} & I_s \end{bmatrix} V^T p \preceq A^+ c.$$  (32)

$V$ can be split into four blocks as

$$V = \begin{bmatrix} V_{r \times r} & V_{r \times s} \\ V_{s \times r} & V_{s \times s} \end{bmatrix}.$$  (33)

We use $w$ to denote the $n \times 1$ column vector $V^T p$, i.e.,

$$w = V^T p.$$  (34)

Note that $p \leftrightarrow w$ is a reversible bijection, since $V^T$ is a full rank matrix.

Then (32) is equivalent to

$$\begin{bmatrix} 0_{r \times r} & \bar{V}_{r \times s} \\ 0_{s \times r} & \bar{V}_{s \times s} \end{bmatrix} \begin{bmatrix} w_{r \times 1} \\ w_{s \times 1} \end{bmatrix} \preceq A^+ c,$$  (35)

which is equivalent to

$$\begin{bmatrix} \bar{V}_{r \times s} \\ \bar{V}_{s \times s} \end{bmatrix} \begin{bmatrix} w_{s \times 1} \end{bmatrix} \preceq A^+ c.$$  (36)

Hence, checking whether (30) has a solution or not is equivalent to checking whether (36) has a solution or not. To
check whether (36) has a solution or not, we can construct a new LP for (36) in the same way as in the proof in Theorem 2. However, the size of the newly constructed LP will be smaller than that of (19) constructed in the proof of Theorem 2. The complexity for the newly constructed LP will be \(O((ns^2 + n^{1.5})L)\). Since \(s\) is always less than or equal to \(n\) (sometimes, \(s\) can be much less than \(n\)) and that \(L\) doesn’t change, compared with the LP (19), the computational complexity for this new LP will be reduced.

V. Sensitivity Analysis

The simulatability condition \(\text{Sim}_Y(Z \to X)\) requires the precise information about the joint PMFs \(P_{XY}\) and \(P_{YZ}\). In practice, it’s reasonable to assume that \(P_{XY}\) is precisely known. However, in certain scenarios, we may not know \(P_{YZ}\) perfectly, as \(Z\) is the random variable observed at the adversary. In this section, we investigate the sensitivity of the simulatability condition \(\text{Sim}_Y(Z \to X)\) with regards to the modeling uncertainty about \(P_{YZ}\). The techniques can be applied to analyze \(\text{Sim}_X(Z \to Y)\).

In particular, assume that \(P_{XY}\) is perfectly known but \(P_{YZ}\) is known only to a certain precision. To be more precise, assume that the true joint PMF of Bob and Eve is \(P_{YZ}\), but the legitimate users know only an estimate \(P_{YZ}\) to a certain precision in the following sense:

\[
|\Delta a_{i,k}| \leq \delta, \quad \forall i \in [1 : |Y|], k \in [1 : |Z|],
\]

(37)

in which

\[
\Delta a_{i,k} \triangleq P_{YZ}(i,k) - P_{YZ}(i,k).
\]

(38)

As \(P_{XY}\) is perfectly known, which implies \(P_Y\) is perfectly known, we have

\[
P_Y(i) = \sum_{k=1}^{|Z|} P_{YZ}(i,k) = \sum_{k=1}^{|Z|} P_{YZ}(i,k).
\]

(39)

Similar to (5), we use \(A\) to denote \(P_{YZ}\), and \(\hat{A} = A + \Delta A\) to denote \(P_{YZ}\). Using these notation, we can rewrite (37) and (39) as

\[
|\Delta a_{i,k}| \leq \delta, \quad \forall i = 1, \ldots, |Y|, k = 1, \ldots, |Z|,
\]

(40)

and

\[
\sum_{k=1}^{|Z|} \Delta a_{i,k} = 0, \quad \forall i = 1, \ldots, |Y|.
\]

(41)

Suppose \(\text{Sim}_Y(Z \to X)\) does not hold with regards to \(P_{XYZ}\) (the perceived model by the legitimate users), we would like to know whether \(\text{Sim}_Y(Z \to X)\) holds or not regarding \(P_{XYZ}\) (the true underlying model).

From the discussion in Section III, we know that checking \(\text{Sim}_Y(Z \to X)\) is equivalent in checking whether there exists a \(Q\) satisfying (5), (6) and (7). In Section III, to facilitate the analysis of the algorithm complexity, we convert these equations to an LP problem. In this section, to facilitate the sensitivity analysis, we construct another optimization problem:

\[
\min_{q} ||\hat{A}q - c||_1
\]

(42)

s.t.

\[
q \succeq 0,
\]

(43)

\[
[I_{|Z|} \otimes 1_{1,X|Z|}]q = 1_{|Z|},
\]

(44)

where \(||\cdot||_1\) is the \(\ell_1\) norm, \(c \triangleq [A \otimes I_{|X|}]\) and \(q \triangleq \text{Vec}(Q^T)\). Here, (42) corresponds to (5), (43) corresponds to (6), and (44) corresponds to (7), respectively. It is clear that the simulatability condition holds iff the optimal value for (42) equals 0.

Now, suppose \(\text{Sim}_Y(Z \to X)\) does not hold with regards to \(P_{XYZ}\), that is the optimal value of (42) is \(\epsilon_0 > 0\), we have the following theorem.

**Theorem 4.** Suppose \(\text{Sim}_Y(Z \to X)\) does not hold with regards to \(P_{XYZ}\), then for any \(\delta < \frac{\epsilon_0}{\|Y||Z|}\), \(\text{Sim}_Y(Z \to X)\) does not hold with regards to \(P_{XYZ}\) neither.

**Proof:** To prove the simulatability condition doesn’t hold for \(P_{XYZ}\) is equivalent to show the optimal value for the following convex optimization problem is larger than 0. We have

\[
\min_{q} ||\hat{A}q - c||_1
\]

(45)

s.t.

\[
q \succeq 0,
\]

(43)

\[
[I_{|Z|} \otimes 1_{1,X|Z|}]q = 1_{|Z|},
\]

(44)

in which \(\hat{A} \triangleq [A \otimes I]\). We have

\[
\min_{q} ||\hat{A}q - c||_1
\]

(42)

\[
= \min_{q} ||(A + \Delta A) \otimes Iq - c||_1
\]

(43)

\[
\geq \min_{q} ||Aq + \Delta Aq - c||_1
\]

(44)

\[
\geq \min_{q} \{ ||Aq - c||_1 - ||\Delta Aq||_1 \}
\]

(45)

\[
\geq \min_{q} \{ ||Aq - c||_1 - \max_{q} ||\Delta Aq||_1 \}
\]

(46)

\[
\geq \epsilon_0 - |Y||Z|\delta
\]

(47)

\[
> 0,
\]

if \(\delta < \frac{\epsilon_0}{|Y||Z|}\). In the above derivation, step (a) holds, because the summation of each row of \(Q\) equals to 1. This completes the proof.

The bound obtained in Theorem 4 is sharp. In particular, there are examples in which once \(\delta = \frac{\epsilon_0}{|Y||Z|}\), we can find \(P_{XYZ}\) such that the simulatability condition holds although the condition does not hold for \(P_{XYZ}\). In the following, we give such an example.

**Assume**

\[
A = \begin{bmatrix}
1/4 & 1/4 \\
1/4 & 1/4
\end{bmatrix}, \quad C = \begin{bmatrix}
1/3 & 1/6 \\
1/6 & 1/3
\end{bmatrix}.
\]

(46)
Then, by setting $Q = \begin{bmatrix} \lambda_1 & 1 - \lambda_1 \\ \lambda_2 & 1 - \lambda_2 \end{bmatrix}$, we have
\[
\varepsilon_0 := \min_Q \| Aq - c \|_1 \quad (47)
\]
\[
= \min_Q \| \text{Vec}(AQ - C) \|_1 \quad (48)
\]
\[
= \min_{\lambda_1, \lambda_2} \left\| \begin{pmatrix} \frac{\lambda_1 + \lambda_2}{4} - \frac{1}{3} & \frac{\lambda_1 + \lambda_2}{4} - \frac{1}{6} \\ \frac{\lambda_1 + \lambda_2}{4} - \frac{1}{3} & \frac{\lambda_1 + \lambda_2}{4} - \frac{1}{6} \end{pmatrix} \right\|_1 \quad (49)
\]
\[
= 2 \min_{\lambda_1, \lambda_2} \left\{ \frac{\lambda_1 + \lambda_2}{4} - \frac{1}{3} + \frac{\lambda_1 + \lambda_2}{4} - \frac{1}{6} \right\} \quad (50)
\]
\[
= 2 \cdot \frac{1}{3} - \frac{1}{6} \quad (51)
\]
\[
= \frac{1}{3} \quad (52)
\]
which implies that $\text{Sim}_Y(Z \to X)$ does not hold for the given $P_{XYZ}$.

Now if
\[
\delta = \frac{\varepsilon_0}{||Y||_Z} = \frac{1}{2} \cdot \frac{\varepsilon_0}{12} = \frac{1}{12},
\]
then $\tilde{A}$ can be
\[
\tilde{A} = \begin{bmatrix} 1/4 + 1/12 & 1/4 - 1/12 \\ 1/4 - 1/12 & 1/4 + 1/12 \end{bmatrix} \quad (53)
\]
\[
= \begin{bmatrix} 1/3 & 1/6 \\ 1/6 & 1/3 \end{bmatrix} \quad (54)
\]
This is exactly the same as $C$, which obviously indicates that the simulatability condition holds for the perturbed PMF $P_{XYZ}$.

VI. CONCLUSION

In this paper, we have proposed an efficient algorithm to check the simulatability condition, an important condition in the problems of secret key generation using a non-authenticated public channel. We have also proposed a simple and flexible method to calculate a possible simulatability channel if the simulatability condition holds. The proposed algorithms have polynomial complexities. We have shown that the simulatability condition is not sensitive to modelling uncertainty. Finally, we have proposed an approach to further reduce the computational complexity.

APPENDIX A

FARKAS’ LEMMA

There are several equivalent forms of the Farkas’ lemma [16]. Here, we state a form that will be used in our proof.

Lemma 1. (Farkas’ Lemma [16]) Let $B$ be a matrix, and $b$ be a vector, then the system specified by $Bp \leq b$, has a solution $p$, if and only if $t^T b \geq 0$ for each column vector $t \geq 0$ with $B^T t = 0$.

REFERENCES