Fake-Hiding Deniable Secret Sharing from One-Way Functions

Kengo Sasaki[1], Yusuke Yoshida[1], and Keisuke Tanaka[1]

Institute of Science Tokyo
sasaki.k.3877@m.isct.ac.jp, yoshida.yusuke@comp.isct.ac.jp,
keisuke@comp.isct.ac.jp

Abstract. Deniable secret sharing (DSS), introduced by Canetti et al. (TCC 2025), considers an adversary who coerces or bribes shareholders into revealing their shares. Shareholders are either fakers, who submit fake shares of a target message, or snitches, who reveal their real shares. Fake hiding, the strongest security notion considered in their work, ensures that the adversary cannot tell whether any submitted shares are fake.

We focus on their coordinated shareholder-specified setting for threshold DSS, where fakers coordinate and choose the target fake message. When the fakers form a strict majority but remain unqualified to reconstruct the secret, Canetti et al. rule out information-theoretic fake-hiding DSS and give a construction based on indistinguishability obfuscation.

In this regime, we construct a DSS scheme with statistical fake hiding and computational privacy from one-way functions alone, and prove that one-way functions are also necessary for such schemes. We further define a property that strengthens fake hiding by guaranteeing deniability even when the fakers' original shares are revealed, and call it strong fake hiding. We show that public-key encryption is necessary and sufficient to construct DSS schemes satisfying this property.

Table of Contents

Fake-Hiding Deniable Secret Sharing from One-Way Functions 1
Kengo Sasaki, Yusuke Yoshida, and Keisuke Tanaka
1 Introduction 3
1.1 Background 3
1.2 Our Results 4
1.3 Technical Overview 4
1.4 Future Work 8
1.5 AI Disclosure 8
2 Preliminaries 8
2.1 Notation 8
2.2 Pseudorandom Generators 9
2.3 Public Key Encryption 9
3 Deniable Secret Sharing 9
3.1 Definitions 10
3.2 Information-Theoretic Building Block 11
4 Strong Fake Hiding and Public-Key Encryption 13
4.1 Strong Fake Hiding from Public-Key Encryption 14
4.2 Strong Fake Hiding Implies Public-Key Encryption 18
5 Fake-Hiding Deniable Secret Sharing from One-Way Functions 19
5.1 SKE with Ciphertext Replaceability from One-Way Functions . . . 19
5.2 Fake-Hiding DSS from SKE with Ciphertext Replaceability 23
5.3 One-Way Functions from Fake-Hiding DSS 26

1 Introduction

1.1 Background

Deniable secret sharing (DSS), introduced by Canetti et al. [CDK+25], extends secret sharing to settings in which an adversary bribes or coerces shareholders into revealing their shares. DSS equips shareholders with an additional *faking algorithm*, allowing them to submit fake shares instead.

Shareholders are either *fakers*, who submit fake shares, or *snitches*, who submit their real shares. Canetti et al. consider several notions of deniability. Fake-real indistinguishability hides which of two groups of shareholders submitted real shares and which submitted fake shares. The strongest notion considered in their work is *fake hiding*. Even when fake shares are mixed with real shares, the submitted shares are indistinguishable from a fresh sharing of a fake message, so the adversary cannot tell whether any shares are fake. This can reduce the adversary's incentive to coerce shareholders, since it cannot be sure that the shares it obtains correspond to the real secret.

Canetti et al. consider random, dealer-specified, and shareholder-specified faking, in which the fake message is respectively generated at random, chosen by the dealer, or chosen by the fakers. Faking may also be uncoordinated, with each faker acting locally, or coordinated, with the fakers cooperating to generate fake shares.

For coordinated shareholder-specified threshold DSS, they rule out fake hiding unless the fakers form a strict majority and the snitches are unqualified. On the positive side, they give an information-theoretic construction when the fakers are qualified. The remaining case is an unqualified majority of fakers. Canetti et al. show that information-theoretic fake-hiding DSS is impossible in this case and give a computational construction based on indistinguishability obfuscation. Figure 1 summarizes these feasibility results.

In subsequent work, Amores et al. [ABDC+26] study the communication required for faking and what the fakers learn about the secret. Their results apply to general access structures. For information-theoretic fake hiding, they prove lower bounds on faking message complexity and give protocols based on linear secret sharing. For linear secret sharing schemes, they further show that certain unqualified subsets of fakers necessarily learn the secret through faking.

For fake-real indistinguishability, they give two computational constructions achieving non-interactive faking and privacy against fakers, assuming that each faker knows the set of fakers. The first relies only on a PRG but has share size exponential in the number of shareholders. The second avoids this exponential overhead assuming virtual black-box obfuscation. Very recently, Nassar [Nas26] weakened the obfuscation assumption to indistinguishability obfuscation.

Despite these results, it remains open whether fake-hiding DSS with an unqualified majority of fakers can be constructed from assumptions weaker than indistinguishability obfuscation [CDK+25, Section 1.3]. A further question is whether fake hiding fully captures the deniability we expect from DSS.

1.2 Our Results

We study coordinated shareholder-specified threshold DSS with n shareholders and reconstruction threshold t. We consider the case where the set F of fakers forms a strict majority but remains unqualified to reconstruct the secret, i.e., $n/2 < |F| < t$.

- We introduce *strong fake hiding* to capture deniability even when the fakers' original shares are revealed. This property conceals which of two sets of the fakers' shares was used to fake the other, even given both sets and the snitches' real shares.
- We show that public-key encryption with perfect correctness is necessary and sufficient for DSS with perfect correctness, perfect strong fake hiding, and computational privacy. If we allow negligible decryption error in PKE, computational strong fake hiding already suffices to construct PKE.
- For ordinary fake hiding, we show that one-way functions are necessary and sufficient. From one-way functions, we construct a DSS scheme achieving computational privacy, statistical fake hiding, and perfect correctness. Conversely, any DSS scheme with computational privacy and computational fake hiding implies the existence of one-way functions.
- We introduce *ciphertext replaceability* for secret-key encryption as a key ingredient in our OWF-based DSS construction. This property allows anyone, given only a ciphertext and a target message, to replace the ciphertext with one that is indistinguishable from a fresh encryption of the target message under the same key. We construct a one-time secure secret-key encryption scheme achieving statistical ciphertext replaceability from one-way functions.

Together with prior work, these results map out the computational landscape of fake hiding in coordinated shareholder-specified threshold DSS. Figures 1 and 2 summarize this landscape.

1.3 Technical Overview

Fake hiding from public-key encryption. We begin with our construction of fake-hiding DSS from public-key encryption. The central idea is to encrypt the secret message and share the ciphertext and secret key separately, using different reconstruction thresholds. The ciphertext is shared with threshold $r = \lfloor n/2 \rfloor + 1$, while the secret key is shared with threshold t. Each shareholder receives a share of each component, together with the public key.

An unqualified set of at least r fakers can reconstruct the ciphertext but cannot reconstruct the secret key. To fake toward a target message, the fakers use the public key to encrypt that message and apply the information-theoretic faking algorithm to their ciphertext shares.

This construction extends the information-theoretic fake-hiding DSS scheme of Canetti et al. [CDK+25] from reconstruction threshold r to t, while still allowing any set of at least r shareholders to fake.

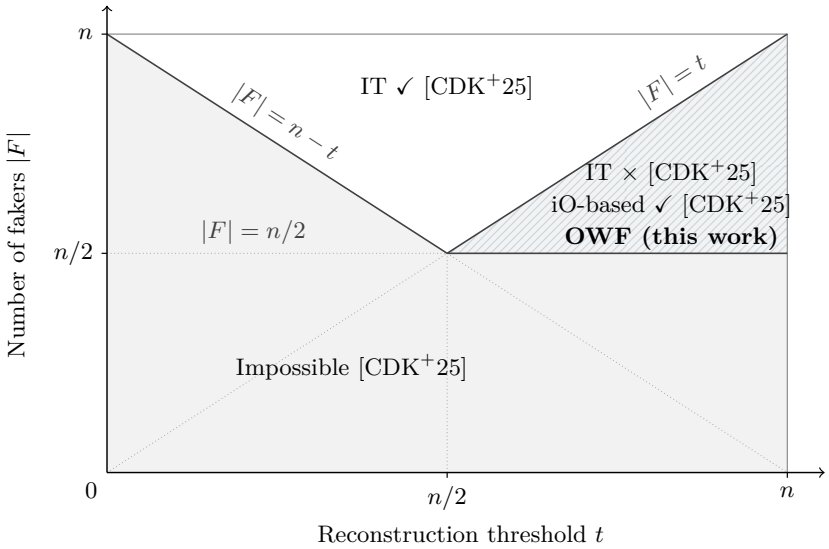

Fig. 1. Feasibility of fake hiding with coordinated shareholder-specified faking. n denotes the number of shareholders. The parameter-space presentation follows [CDK+25, Fig. 1(b)].

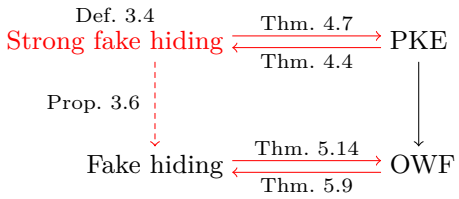

Fig. 2. Overview of our results for threshold-t DSS with coordinated shareholder-specified faking in the region $n/2 < |F| < t$. Our contributions are highlighted in red. The dashed arrow denotes a trivial implication.

Strong fake hiding. Fake hiding guarantees that the shares after faking are indistinguishable from an honest sharing of the target message. However, the definition does not consider the fake shares together with the fakers' original shares.

We therefore introduce *strong fake hiding*. Given two sets of the fakers' shares corresponding to two messages, together with the snitches' real shares, one cannot tell which set was used to fake the other. This property implies ordinary fake hiding.

To see how strong fake hiding can be useful, consider the following scenario. Suppose a secret is shared among shareholders in two groups, F and R, for later joint reconstruction, with neither group able to reconstruct it alone. Before the intended disclosure, an adversary obtains the shares held by F and presents

them to R. This allows R to reconstruct the secret without F's cooperation. The shareholders in F wish to deny that the message reconstructed by R was the original secret. They use Fake to produce shares for an alternative message and offer a competing explanation. They claim that these were their original shares and that the adversary obtained them and used Fake to produce the shares shown to R. The shareholders in R compare both sets using their own shares, which they do not reveal to the adversary. Strong fake hiding ensures that R cannot distinguish the two accounts on the basis of these shares.

F could also claim that the adversary generated the disputed shares using an efficient algorithm other than Fake. Our result on the necessity of PKE also holds when such explanations are allowed (Remark 4.9).

Our PKE-based DSS construction satisfies perfect strong fake hiding (Theorem 4.4). The ciphertext is shared using the information-theoretic DSS scheme of Canetti et al., which uses Shamir secret sharing [Sha79]. The original and target ciphertexts are generated independently under the same public key. Given the snitches' ciphertext shares, honest sharing and faking independently sample the remaining shares from the same distribution for each fixed ciphertext. Thus, sharing the first ciphertext and faking to the second gives exactly the same joint distribution as sharing the second and faking to the first. The public key and secret-key shares are unchanged by faking, so this symmetry extends to the full shares.

From strong fake hiding to public-key encryption. Conversely, strong fake hiding allows us to use the original faker shares as a public key and the faking algorithm as encryption. A key-generation algorithm shares the bit 0, publishes the shares of an unqualified faker set F, and retains the remaining shares as the secret key. To encrypt a bit b, one fakes the public shares toward b. Decryption reconstructs from the ciphertext and secret key.

Together with privacy of the unqualified faker set, strong fake hiding suffices to make this conversion secure (Theorem 4.7).

Fake hiding from one-way functions. In our PKE-based DSS construction, faking relies on replacing the original ciphertext with one that is indistinguishable from a fresh encryption of the target message under the same key. The fakers can then use the information-theoretic faking algorithm to replace their ciphertext shares with shares of this new ciphertext, while keeping their secret-key shares unchanged. Public-key encryption provides this functionality by allowing the fakers to encrypt the target message using the public key. A natural question is whether the same functionality can be achieved with secret-key encryption (SKE).

We formalize this functionality as *ciphertext replaceability* for SKE (Definition 5.3) and construct a scheme with statistical ciphertext replaceability from one-way functions.

Our construction is based on the following idea. Encryption chooses one of exponentially many indices uniformly at random, and replacement encrypts

the target message at the next index. A replaced ciphertext is therefore distributed like a fresh encryption of the target message, except that its index is shifted by one. Since the indices are uniform over exponentially many values, this shift changes the distribution only at the endpoints, giving negligible statistical distance.

We realize this idea by letting the indices correspond to the leaves of a GGM tree whose root seed is the secret key. At each index, the message is masked with a pseudorandom one-time pad derived from the corresponding leaf. The ciphertext contains the leaf index and seeds for subtrees to the right of that leaf. These seeds allow anyone to advance to the next leaf, replace the message using its pad, and update the subtree seeds, while the pad at the original leaf remains pseudorandom. Both the ciphertext size and the update time are polynomial in the tree depth.

Canetti et al. [CDK$^+$25, Section 5] use the same idea of a pseudorandom starting level and a one-step increase in their iO-based construction. Here, the GGM subtree seeds enable the required update without obfuscation.

We obtain our DSS construction by using SKE with ciphertext replaceability in place of public-key encryption. Statistical ciphertext replaceability yields statistical fake hiding, and one-time security suffices for computational t-privacy (Theorem 5.9).

Our OWF-based construction does not satisfy strong fake hiding. Since Fake increases the ciphertext index, comparing the ciphertext indices reveals which set of shares was used to fake the other.

From fake hiding to one-way functions. Canetti et al. [CDK$^+$25, Theorem 4] show that information-theoretic fake hiding is impossible when the fakers are unqualified. The construction above nevertheless achieves fake hiding from one-way functions, a minimal assumption for computational cryptography. This suggests that one-way functions may be necessary as well as sufficient.

To establish necessity, we compare what privacy and fake hiding imply about predicting the secret. Consider an honest sharing of a uniformly random bit. The goal is to predict this bit using only the shares of an unqualified set of fakers. We measure the remaining uncertainty by conditional entropy. Perfect privacy makes this entropy one, whereas perfect fake hiding forces it to be zero.

Let us see why the entropy is zero. By perfect correctness, the snitch shares resolve all remaining uncertainty about the secret once the honest faker shares are known. Consider the following experiment:

1. Sample an honest sharing of a uniformly random bit and give only the faker shares to a predictor.
2. The predictor chooses an independent uniform target bit.
3. The predictor applies Fake to the received shares toward this target.

By perfect fake hiding, the following pairs have the same joint distribution with the unchanged snitch shares:

– the shares output by Fake and the chosen random target bit,

– honest faker shares and their secret bit.

In this experiment, the predictor has used only the received faker shares and its own randomness, so it has learned nothing new about the snitch shares. Yet it has reproduced the situation of receiving honest faker shares together with their secret. Even if the predictor were also given the original secret, it would therefore learn nothing more about the snitch shares on average. The honest faker shares must therefore already determine the secret, so the conditional entropy is zero.

Allowing statistical error in both privacy and fake hiding gives approximate versions of these entropy bounds. These show that privacy and fake hiding cannot both have small statistical error. We combine the corresponding distribution pairs into a single pair of efficiently samplable distributions with noticeable statistical distance. Computational privacy and fake hiding make this pair computationally indistinguishable. Goldreich's theorem [Gol90] states that the existence of such a pair implies one-way functions.

1.4 Future Work

Following Canetti et al. [CDK+25], we focus on threshold secret sharing. We expect our results to extend to general access structures, but leave this extension for future work. Another direction is to explore applications of SKE with ciphertext replaceability to constructing other cryptographic primitives from one-way functions.

1.5 AI Disclosure

We used GPT-6 Astra and GPT-6.1 Sol to help translate drafts, suggest wording, and revise the text. These tools also helped us explore and discuss ideas for proofs and constructions. In particular, they suggested the use of GGM trees in the construction from one-way functions. We reviewed and revised all text, verified all proofs, and take full responsibility for the entire content of this paper.

2 Preliminaries

2.1 Notation

We denote the security parameter by λ. We denote the set $\{1, 2, \ldots, n\}$ by $[n]$. We write PPT for probabilistic polynomial-time algorithms.

For a finite set S, we write $x \leftarrow S$ for sampling x uniformly at random from S. For a probabilistic algorithm $\mathcal{A}$, $y \leftarrow \mathcal{A}(x)$ denotes its output on input x using fresh randomness. Let U_k denote the uniform distribution over $\{0,1\}^k$.

Let $\mathsf{bin}_k(i)$ denote the k-bit representation of i for $0 \leq i < 2^k$. We write ϵ for the empty string.

We write $X \approx_c Y$, $X \approx_s Y$, and $X \equiv Y$ for computational indistinguishability, statistical indistinguishability, and identical distributions, respectively. We write $\Delta(X, Y)$ for statistical distance and $\mathsf{negl}(\lambda)$ for a negligible function of the security parameter λ.

2.2 Pseudorandom Generators

Definition 2.1 (Pseudorandom generator, PRG).
A function $G : \{0,1\}^* \to \{0,1\}^*$ is a pseudorandom generator if:

- *There exists a polynomial l such that $l(\lambda) > \lambda$ for all $\lambda \in \mathbb{N}$, and $|G(x)| = l(|x|)$ for all $x \in \{0,1\}^*$.*
- *Given $x \in \{0,1\}^\lambda$, $G(x)$ can be computed in time polynomial in λ.*
- *For every PPT algorithm $\mathcal{A}$,*

$$\left|\Pr[\mathcal{A}(1^\lambda, G(U_\lambda)) = 1] - \Pr[\mathcal{A}(1^\lambda, U_{l(\lambda)}) = 1]\right| \leq \mathsf{negl}(\lambda).$$

2.3 Public Key Encryption

Definition 2.2 (Public Key Encryption Scheme). *Let $\ell_m : \mathbb{N} \to \mathbb{N}$ be a polynomially bounded length function. A public key encryption scheme is a tuple of PPT algorithms* $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ *such that:*

- $\mathsf{KeyGen}(1^\lambda) \to (\mathsf{pk}, \mathsf{sk})$ *takes a security parameter λ and outputs a public key* pk *and a secret key* sk.
- $\mathsf{Enc}(\mathsf{pk}, m) \to c$ *takes a public key* pk *and a message $m \in \{0,1\}^{\ell_m}$ and outputs a ciphertext c.*
- $\mathsf{Dec}(\mathsf{sk}, c) \to m$ *takes a secret key* sk *and a ciphertext c and outputs a message $m \in \{0,1\}^{\ell_m}$ or $\perp$.*

and satisfies perfect correctness: for all $m \in \{0,1\}^{\ell_m}$,

$$\Pr[\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, m)) = m] = 1,$$

where $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$.

Definition 2.3 (IND-CPA Security). *A public key encryption scheme is IND-CPA secure if the following holds. For any PPT adversary $\mathcal{A}$, consider the following game:*

1. $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$
2. *$\mathcal{A}$ is given* pk *and outputs two messages $m_0, m_1 \in \{0,1\}^{\ell_m}$.*
3. *Sample $b \leftarrow \{0,1\}$ and give $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m_b)$ to $\mathcal{A}$.*
4. *$\mathcal{A}$ outputs a guess $b' \in \{0,1\}$.*
5. *$\mathcal{A}$ wins if $b' = b$.*

Then,

$$\left|\Pr[\mathcal{A} \text{ wins}] - \frac{1}{2}\right| \leq \mathsf{negl}(\lambda).$$

3 Deniable Secret Sharing

In this paper, we consider only efficient threshold DSS with coordinated shareholder-specified faking. We recall the DSS model and existing properties of [CDK+25], and introduce a new property, strong fake hiding.

3.1 Definitions

Notation. We use the following public parameters:

- n: the number of shareholders.
- t: the reconstruction threshold, with $1 \leq t \leq n$.

The number of shareholders and message lengths are polynomially bounded in λ. Let $F \sqcup R = [n]$ be a partition of the shareholders into fakers, who provide fake shares, and snitches, who provide their real shares. For a share tuple $S = (s_1, \ldots, s_n)$ and $Q \subseteq [n]$, we write $S_Q = (s_i)_{i \in Q}$, retaining the party indices.

While the original definitions in [CDK+25] are game-based, we use a distribution-based formulation for the following definitions. We give computational definitions, except for correctness. The statistical variants require negligible statistical distance. The perfect variants require identical distributions.

Definition 3.1 (Threshold-t Deniable Secret Sharing).
A threshold-t deniable secret sharing scheme is a tuple of PPT algorithms $(\mathsf{Share}, \mathsf{Rec}, \mathsf{Fake})$ *such that:*

- $\mathsf{Share}(m) \to S$ *takes a secret* m *and outputs* n *shares.*
- $\mathsf{Rec}(S_Q) \to m$ *or* $\perp$ *takes shares indexed by* $Q \subseteq [n]$ *and outputs a message or* $\perp$*.*
- $\mathsf{Fake}(S_F, m^{\mathsf{fake}}) \to S_F^{\mathsf{fake}}$ *takes the shares of* F *and a fake message* m^{fake}*, and outputs fake shares.*

For each message length, each party has a share space whose elements have polynomial length in λ*.* Fake *takes and returns shares in the same spaces.*

The scheme satisfies the following properties:

Perfect correctness. *For every message* m *and every* $Q \subseteq [n]$ *such that* $|Q| \geq t$*,*

$$\Pr[\mathsf{Rec}(S_Q) = m] = 1,$$

where $S \leftarrow \mathsf{Share}(m)$*.*

t-Privacy. *For every partition* $F \sqcup R = [n]$ *such that* $|R| < t$ *and all messages* m_0, m_1, m' *such that* $|m_0| = |m_1| = |m'|$*, the output distributions* D_0, D_1 *below satisfy* $D_0 \approx_c D_1$*.*

D_0	D_1
$S \leftarrow \mathsf{Share}(m_0)$	$S \leftarrow \mathsf{Share}(m_1)$
$S_F \leftarrow \mathsf{Fake}(S_F, m')$	$S_F \leftarrow \mathsf{Fake}(S_F, m')$
return (m_0, m_1, m', F, S)	**return** (m_0, m_1, m', F, S)

Our definition differs from that of Canetti et al. [CDK+25] in that we explicitly include correctness and omit their requirement of $(\emptyset, [n])$-fake-real indistinguishability. This requirement follows from fake hiding (for $F = [n]$), which is the focus of this work.

Definition 3.2 (F-Fake Hiding). *Fix $F \subseteq [n]$. A deniable secret sharing scheme is F-fake-hiding if, for all messages m, m' s.t. $|m| = |m'|$, the output distributions $D_{\text{fake}}, D_{\text{real}}$ below satisfy $D_{\text{fake}} \approx_c D_{\text{real}}$.*

D_{fake}	D_{real}
$S \leftarrow \mathsf{Share}(m)$	$S \leftarrow \mathsf{Share}(m')$
$S_F \leftarrow \mathsf{Fake}(S_F, m')$	**return** (m, m', F, S)
return (m, m', F, S)	

Definition 3.3 (k-Fake Hiding). *Let $k \in \{0, \ldots, n\}$. A deniable secret sharing scheme is k-fake-hiding if it is F-fake-hiding for any $F \subseteq [n]$ such that $|F| > k$.*

Strong fake hiding. We introduce strong fake hiding, which hides which of two sets of the fakers' shares was used to fake the other, even given both sets and the snitches' real shares.

Definition 3.4 (F-Strong Fake Hiding).

Fix $F \subseteq [n]$ and let $R = [n] \setminus F$. A deniable secret sharing scheme satisfies F-strong fake hiding if, for all messages m_0, m_1 s.t. $|m_0| = |m_1|$, the output distributions D_0, D_1 below satisfy $D_0 \approx_c D_1$.

D_0	D_1
$S \leftarrow \mathsf{Share}(m_0)$	$S \leftarrow \mathsf{Share}(m_1)$
$X_0 \leftarrow S_F$	$X_1 \leftarrow S_F$
$X_1 \leftarrow \mathsf{Fake}(X_0, m_1)$	$X_0 \leftarrow \mathsf{Fake}(X_1, m_0)$
$Y \leftarrow S_R$	$Y \leftarrow S_R$
return $(m_0, m_1, F, X_0, X_1, Y)$	**return** $(m_0, m_1, F, X_0, X_1, Y)$

Definition 3.5 (k-Strong Fake Hiding). *Let $k \in \{0, \ldots, n\}$. A deniable secret sharing scheme satisfies k-strong fake hiding if it satisfies F-strong fake hiding for any $F \subseteq [n]$ such that $|F| > k$.*

Proposition 3.6 (Strong Fake Hiding Implies Fake Hiding). *For every $F \subseteq [n]$, F-strong fake hiding implies F-fake hiding. The same implication holds for the statistical and perfect variants.*

Proof. Discarding X_0 and combining X_1, Y in Definition 3.4 gives the two F-fake-hiding experiments. □

3.2 Information-Theoretic Building Block

For reconstruction threshold q, let ITDSS_q denote the information-theoretic DSS scheme of Canetti et al. [CDK+25], based on Shamir secret sharing [Sha79]. They showed that ITDSS_q achieves perfect F-fake hiding when the fakers are qualified and the snitches are unqualified, i.e., $n - |F| < q \leq |F|$. We prove perfect F-strong fake hiding in the same range. We also verify perfect correctness and perfect q-privacy.

Construction of ITDSS_q *[CDK+25].* Fix a reconstruction threshold $1 \le q \le n$ and a prime field $\mathbb{F}$ with more than n elements. The algorithms for messages in $\mathbb{F}$ are given in Figure 3.

$\mathsf{Share}(m)$

$p \leftarrow \{ f \in \mathbb{F}[x] \mid \deg f < q,\ f(0) = m \}$
return $(p(i))_{i \in [n]}$

$\mathsf{Rec}(S_Q)$

if $|Q| < q$ **then return** $\perp$
Interpolate p with $\deg p < q$ and $p(i) = s_i \quad (i \in Q)$
if no such p exists **then return** $\perp$
return $p(0)$

$\mathsf{Fake}(S_F, m^{\mathsf{fake}})$

$R \leftarrow [n] \setminus F$
if $|R| \ge q$ **then return** S_F
if $|F| \ge q$ **then**
  Interpolate p from S_F
  $p' \leftarrow \{ f \in \mathbb{F}[x] \mid \deg f < q,\ f(0) = m^{\mathsf{fake}}, f(j) = p(j) \quad (j \in R) \}$
  return $(p'(i))_{i \in F}$
$s_i^{\mathsf{fake}} \leftarrow \mathbb{F} \quad (i \in F)$
return $(s_i^{\mathsf{fake}})_{i \in F}$

Fig. 3. Information-theoretic DSS with reconstruction threshold q.

We encode bit-string messages as vectors over $\mathbb{F}$ and apply the construction coordinatewise.

We use the following lemma to prove Theorem 3.8.

Lemma 3.7. *Fix $F \subseteq [n]$ and let $R = [n] \setminus F$. If a DSS scheme satisfies the following conditions:*

- *Perfect F-fake hiding.*
- *For all messages m_0, m_1 s.t. $|m_0| = |m_1|$, let $S \leftarrow \mathsf{Share}(m_0)$. Then S_F and $\mathsf{Fake}(S_F, m_1)$ are conditionally independent given S_R.*

then it satisfies perfect F-strong fake hiding.

Proof. Fix messages m_0, m_1 s.t. $|m_0| = |m_1|$, and consider the experiments D_0, D_1 in Definition 3.4. Perfect F-fake hiding gives, for $i \in \{0,1\}$,

$$\Pr_{D_0}[X_i = x, Y = y] = \Pr_{D_1}[X_i = x, Y = y].$$

Taking the marginal distributions on Y, we obtain

$$\Pr_{D_0}[Y = y] = \Pr_{D_1}[Y = y] =: \mu(y).$$

For each y with $\mu(y) > 0$, define

$$K_i(x \mid y) := \Pr_{D_0}[X_i = x \mid Y = y] = \Pr_{D_1}[X_i = x \mid Y = y], \qquad i \in \{0,1\}.$$

For $b \in \{0,1\}$, conditional independence gives

$$\begin{aligned}&\Pr_{D_b}[X_0 = x_0, X_1 = x_1, Y = y] \\ &\quad = \Pr_{D_b}[Y = y] \Pr_{D_b}[X_0 = x_0 \mid Y = y] \Pr_{D_b}[X_1 = x_1 \mid Y = y] \\ &\quad = \mu(y) K_0(x_0 \mid y) K_1(x_1 \mid y).\end{aligned}$$

Thus $D_0 \equiv D_1$. □

Theorem 3.8 (Properties of ITDSS). *For every $1 \le q \le n$, the construction ITDSS_q above has the following properties:*

- *Perfect correctness.*
- *Perfect q-privacy.*
- *Perfect F-strong fake hiding for every $F \subseteq [n]$ with $n - |F| < q \le |F|$.*

Proof.

Perfect correctness. Perfect correctness follows from Lagrange interpolation.

Perfect q-privacy. Consider the q-privacy experiments D_0, D_1 for $F \sqcup R = [n]$ with $|R| < q$ and equal-length messages m_0, m_1, m'.

If $|F| \ge q$, perfect F-fake hiding gives

$$D_0 \equiv (m_0, m_1, m', F, \mathsf{Share}(m')) \equiv D_1.$$

If $|F| < q$, Fake samples the shares of F uniformly. Since $|R| < q$, the perfect privacy of Shamir secret sharing implies that the real shares of R are independent of the shared message. Thus their joint distributions are identical, giving $D_0 \equiv D_1$.

Perfect strong fake hiding. Fix $F \subseteq [n]$ with $n - |F| < q \le |F|$, and let $R = [n] \setminus F$. Perfect F-fake hiding follows from [CDK+25]. For messages m, m' s.t. $|m| = |m'|$, let $S \leftarrow \mathsf{Share}(m)$ and $S'_F \leftarrow \mathsf{Fake}(S_F, m')$. Since $|R| < q \le |F|$, Fake interpolates p from S_F and obtains $S_R = (p(j))_{j \in R}$. Conditional on $S_R = y$, it samples p' uniformly from

$$\big\{ f \in \mathbb{F}[x] \;\big|\; \deg f < q,\ f(0) = m',\ f(j) = y_j\ (j \in R) \big\}.$$

Since this set depends only on m' and S_R, fresh sampling makes $S'_F = (p'(i))_{i \in F}$ conditionally independent of S_F given S_R. Perfect F-strong fake hiding follows from Lemma 3.7. □

4 Strong Fake Hiding and Public-Key Encryption

We show that public-key encryption is necessary and sufficient for DSS with perfect strong fake hiding when the fakers are unqualified and form a strict majority, i.e., $n/2 < |F| < t$.

4.1 Strong Fake Hiding from Public-Key Encryption

Fix n, t such that $\lfloor n/2 \rfloor + 1 \leq t \leq n$. Let ℓ_m be the message length, and let $\mathsf{PKE} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be a public-key encryption scheme with the same message length. Let $r = \lfloor n/2 \rfloor + 1$.

Let ITDSS_r and ITDSS_t be information-theoretic DSS algorithms. Using them, we construct the algorithms $(\mathsf{Share}, \mathsf{Rec}, \mathsf{Fake})$ shown in Figure 4.

Without loss of generality, by padding, we may assume that equal-length messages yield equal-length ciphertexts under the same public key. We may also assume that secret keys have a fixed length $\ell_{\mathsf{sk}} = \ell_{\mathsf{sk}}(\lambda)$.

We use c_i for a share of the ciphertext, s_i for a share of the secret key, and $u_i = (\mathsf{pk}, i, c_i, s_i)$ for a full share of our scheme.

$\mathsf{Share}(m)$

$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{PKE.KeyGen}(1^\lambda)$
$c \leftarrow \mathsf{PKE.Enc}(\mathsf{pk}, m)$
$(c_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_r.\mathsf{Share}(c)$
$(s_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_t.\mathsf{Share}(\mathsf{sk})$
return $((\mathsf{pk}, i, c_i, s_i))_{i\in[n]}$

$\mathsf{Rec}((u_i)_{i\in Q})$

if $|Q| < t$ **then return** $\perp$
$(\mathsf{pk}, i, c_i, s_i) \leftarrow u_i \quad (i \in Q)$
$c^{\mathsf{rec}} \leftarrow \mathsf{ITDSS}_r.\mathsf{Rec}((c_i)_{i\in Q})$
$\mathsf{sk}^{\mathsf{rec}} \leftarrow \mathsf{ITDSS}_t.\mathsf{Rec}((s_i)_{i\in Q})$
if $c^{\mathsf{rec}} = \perp \vee \mathsf{sk}^{\mathsf{rec}} = \perp$ **then return** $\perp$
$m^{\mathsf{rec}} \leftarrow \mathsf{PKE.Dec}(\mathsf{sk}^{\mathsf{rec}}, c^{\mathsf{rec}})$
return m^{rec}

$\mathsf{Fake}((u_i)_{i\in F}, m^{\mathsf{fake}})$

$(\mathsf{pk}, i, c_i, s_i) \leftarrow u_i \quad (i \in F)$
if $|F| < r$ **then**
  $(s_i^{\mathsf{fake}})_{i\in F} \leftarrow \mathsf{ITDSS}_t.\mathsf{Fake}((s_i)_{i\in F}, 0^{\ell_{\mathsf{sk}}})$
  return $((\mathsf{pk}, i, c_i, s_i^{\mathsf{fake}}))_{i\in F}$
$c^{\mathsf{fake}} \leftarrow \mathsf{PKE.Enc}(\mathsf{pk}, m^{\mathsf{fake}})$
$(c_i^{\mathsf{fake}})_{i\in F} \leftarrow \mathsf{ITDSS}_r.\mathsf{Fake}((c_i)_{i\in F}, c^{\mathsf{fake}})$
return $((\mathsf{pk}, i, c_i^{\mathsf{fake}}, s_i))_{i\in F}$

Fig. 4. Algorithm for Deniable Secret Sharing using PKE.

Lemma 4.1 (Perfect correctness). *If* ITDSS_r *and* ITDSS_t *satisfy perfect correctness at thresholds* r *and* t*, respectively, then the algorithms in Figure 4 satisfy perfect correctness.*

Proof. Fix $Q \subseteq [n]$ with $|Q| \geq t$. Since $t \geq r$, perfect correctness of ITDSS_r and ITDSS_t ensures that Rec recovers the original ciphertext and secret key exactly. Hence, by perfect correctness of PKE,

$$\Pr[m^{\mathsf{rec}} = m] = 1.$$

□

Lemma 4.2 (Perfect strong fake hiding). *If* ITDSS_r *satisfies perfect* $\lfloor n/2 \rfloor$*-strong fake hiding, then the algorithms in Figure 4 satisfy perfect* $\lfloor n/2 \rfloor$*-strong fake hiding.*

Proof. Fix messages m_0, m_1 with $|m_0| = |m_1|$ and $F \subseteq [n]$ with $|F| \geq r$, and let $R = [n] \setminus F$. Then $|R| \leq n - r < r$. Let D_0, D_1 be the output distributions in Definition 3.4 for the faker set F. We show $D_0 \equiv D_1$ through a sequence of hybrids.

Throughout, write $u_i^{(b)} = (\mathsf{pk}, i, c_i^{(b)}, s_i)$.

H_0. Expand Share and Fake in D_0:

H_0

$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{PKE.KeyGen}(1^\lambda)$
$c_0 \leftarrow \mathsf{PKE.Enc}(\mathsf{pk}, m_0)$
$(c_i^{(0)})_{i\in[n]} \leftarrow \mathsf{ITDSS}_r.\mathsf{Share}(c_0)$
$(s_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_t.\mathsf{Share}(\mathsf{sk})$
$c_1 \leftarrow \mathsf{PKE.Enc}(\mathsf{pk}, m_1)$
$(c_i^{(1)})_{i\in F} \leftarrow \mathsf{ITDSS}_r.\mathsf{Fake}((c_i^{(0)})_{i\in F}, c_1)$
return $(m_0, m_1, F, (u_i^{(0)})_{i\in F}, (u_i^{(1)})_{i\in F}, (u_i^{(0)})_{i\in R})$

By construction, $H_0 \equiv D_0$.

H_1. Move the encryption of m_1 before the sharing of c_0:

Before (H_0)	After (H_1)
$(c_i^{(0)})_{i\in[n]} \leftarrow \mathsf{ITDSS}_r.\mathsf{Share}(c_0)$	$c_1 \leftarrow \mathsf{PKE.Enc}(\mathsf{pk}, m_1)$
$\vdots$	$(c_i^{(0)})_{i\in[n]} \leftarrow \mathsf{ITDSS}_r.\mathsf{Share}(c_0)$
$c_1 \leftarrow \mathsf{PKE.Enc}(\mathsf{pk}, m_1)$	$\vdots$

$H_0 \equiv H_1$. The moved step uses independent randomness, so $H_0 \equiv H_1$.

H_2. Reverse the direction of faking by sharing c_1 and faking to c_0.

H_2

$\vdots$
$(c_i^{(1)})_{i\in[n]} \leftarrow \mathsf{ITDSS}_r.\mathsf{Share}(c_1)$
$(s_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_t.\mathsf{Share}(\mathsf{sk})$
$(c_i^{(0)})_{i\in F} \leftarrow \mathsf{ITDSS}_r.\mathsf{Fake}((c_i^{(1)})_{i\in F}, c_0)$
return $(m_0, m_1, F, (u_i^{(0)})_{i\in F}, (u_i^{(1)})_{i\in F}, (u_i^{(1)})_{i\in R})$

$H_1 \equiv H_2$. By perfect strong fake hiding of ITDSS_r,

$$\big((c_i^{(0)})_{i\in F}, (c_i^{(1)})_{i\in F}, (c_i^{(0)})_{i\in R}\big) \equiv \big((c_i^{(0)})_{i\in F}, (c_i^{(1)})_{i\in F}, (c_i^{(1)})_{i\in R}\big),$$

where the left and right sides are generated in H_1 and H_2, respectively. Hence, $H_1 \equiv H_2$.

H_3. Move the encryption of m_0 immediately before faking to c_0:

Before (H_2)	After (H_3)
$c_0 \leftarrow \mathsf{PKE.Enc}(\mathsf{pk}, m_0)$	$\vdots$
$\vdots$	$c_0 \leftarrow \mathsf{PKE.Enc}(\mathsf{pk}, m_0)$
$(c_i^{(0)})_{i\in F} \leftarrow$	$(c_i^{(0)})_{i\in F} \leftarrow$
$\quad \mathsf{ITDSS}_r.\mathsf{Fake}((c_i^{(1)})_{i\in F}, c_0)$	$\quad \mathsf{ITDSS}_r.\mathsf{Fake}((c_i^{(1)})_{i\in F}, c_0)$

$H_2 \equiv H_3$. Again, the moved step uses independent randomness, so $H_2 \equiv H_3$. By construction, $H_3 \equiv D_1$. Therefore,

$$D_0 \equiv H_0 \equiv H_1 \equiv H_2 \equiv H_3 \equiv D_1,$$

proving perfect $\lfloor n/2 \rfloor$-strong fake hiding. □

Lemma 4.3 (Computational privacy). *If* PKE *is IND-CPA secure,* ITDSS_r *satisfies perfect* $\lfloor n/2 \rfloor$*-fake hiding, and* ITDSS_t *satisfies perfect* t*-privacy, then the algorithms in Figure 4 satisfy computational* t*-privacy.*

Proof. Fix a partition $F \sqcup R = [n]$ s.t. $|R| < t$, and messages m_0, m_1, m' s.t. $|m_0| = |m_1| = |m'|$. For $b \in \{0,1\}$, let D_b be the output distribution below, as in the definition of t-privacy for the faker set F. We show that $D_0 \approx_c D_1$.

D_b

$(u_i)_{i\in[n]} \leftarrow \mathsf{Share}(m_b)$
$(u_i)_{i\in F} \leftarrow \mathsf{Fake}((u_i)_{i\in F}, m')$
return $(m_0, m_1, m', F, (u_i)_{i\in[n]})$

Case $|F| \geq r$. Applying perfect F-fake hiding of ITDSS_r to the ciphertext shares with target $c^{\mathsf{fake}} \leftarrow \mathsf{PKE.Enc}(\mathsf{pk}, m')$ gives

$$D_0 \equiv (m_0, m_1, m', F, \mathsf{Share}(m')) \equiv D_1.$$

Case $|F| < r$. Expanding Share and the $|F| < r$ branch of Fake gives the following procedure for D_b.

D_b

$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{PKE.KeyGen}(1^\lambda)$
$c \leftarrow \mathsf{PKE.Enc}(\mathsf{pk}, m_b)$
$(c_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_r.\mathsf{Share}(c)$
$(s_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_t.\mathsf{Share}(\mathsf{sk})$
$(s_i)_{i\in F} \leftarrow \mathsf{ITDSS}_t.\mathsf{Fake}((s_i)_{i\in F}, 0^{\ell_{\mathsf{sk}}})$
return $(m_0, m_1, m', F, ((\mathsf{pk}, i, c_i, s_i))_{i\in[n]})$

Let H_b replace the sharing of sk in D_b by a sharing of $0^{\ell_{\mathsf{sk}}}$:

H_b

$\vdots$

$(s_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_t.\mathsf{Share}(0^{\ell_{\mathsf{sk}}})$

$\vdots$

$D_b \equiv H_b$. Since $|R| < t$, perfect t-privacy of ITDSS_t gives $D_b \equiv H_b$.

$H_0 \approx_c H_1$. Let $\mathcal{A}$ be a PPT distinguisher for H_0, H_1. With m_0, m_1, m', F fixed as above, define the IND-CPA adversary $\mathcal{B}$ as follows.

$\mathcal{B}(\mathsf{pk})$

Send (m_0, m_1) to the challenger
Receive the challenge ciphertext $c^\star$
$(c_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_r.\mathsf{Share}(c^\star)$
$(s_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_t.\mathsf{Share}(0^{\ell_{\mathsf{sk}}})$
$(s_i)_{i\in F} \leftarrow \mathsf{ITDSS}_t.\mathsf{Fake}((s_i)_{i\in F}, 0^{\ell_{\mathsf{sk}}})$
$v \leftarrow (m_0, m_1, m', F, ((\mathsf{pk}, i, c_i, s_i))_{i\in[n]})$
return $\mathcal{A}(1^\lambda, v)$

If the challenge bit is b, then v is distributed exactly as H_b. Hence, IND-CPA security implies $H_0 \approx_c H_1$. Therefore,

$$D_0 \equiv H_0 \approx_c H_1 \equiv D_1,$$

proving computational t-privacy. □

Theorem 4.4 (Strong Fake Hiding from PKE). *If there exists an IND-CPA secure public-key encryption scheme, then for every n, t such that $\lfloor n/2 \rfloor + 1 \le t \le n$, there exists a threshold-t deniable secret sharing scheme satisfying perfect correctness, computational t-privacy, and perfect $\lfloor n/2 \rfloor$-strong fake hiding.*

Proof. In Figure 4, take PKE to be an IND-CPA secure scheme and ITDSS_r, ITDSS_t from Section 3.2. We verify the required properties below.

Perfect correctness. By Theorem 3.8, both ITDSS_r and ITDSS_t satisfy perfect correctness. By Lemma 4.1, the algorithms in Figure 4 satisfy perfect correctness.

Perfect strong fake hiding. Since $n - r < r$, Theorem 3.8 gives perfect $\lfloor n/2 \rfloor$-strong fake hiding of ITDSS_r. By Lemma 4.2, the algorithms in Figure 4 satisfy perfect $\lfloor n/2 \rfloor$-strong fake hiding.

Computational privacy. By Proposition 3.6, ITDSS_r also satisfies perfect $\lfloor n/2 \rfloor$-fake hiding. Moreover, ITDSS_t satisfies perfect t-privacy by Theorem 3.8. Since PKE is IND-CPA secure, Lemma 4.3 gives computational t-privacy for the algorithms in Figure 4. □

4.2 Strong Fake Hiding Implies Public-Key Encryption

Let $\mathsf{DSS} = (\mathsf{Share}, \mathsf{Rec}, \mathsf{Fake})$ be a tuple of PPT algorithms with the DSS syntax. Fix $1 \leq t \leq n$ and $F \subseteq [n]$, and let $R = [n] \setminus F$. Using these algorithms, we construct $\mathsf{PKE} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ for single-bit messages as follows:

- $\mathsf{KeyGen}(1^\lambda)$: Sample $S \leftarrow \mathsf{DSS.Share}(0)$ and return $(\mathsf{pk}, \mathsf{sk}) = (S_F, S_R)$.
- $\mathsf{Enc}(\mathsf{pk}, b)$: Return $\mathsf{DSS.Fake}(\mathsf{pk}, b)$.
- $\mathsf{Dec}(\mathsf{sk}, c)$: Let $S_F = c$ and $S_R = \mathsf{sk}$. Return $\mathsf{DSS.Rec}(S)$ if it is a bit, and $\perp$ otherwise.

Let $S^{(b)} \leftarrow \mathsf{DSS.Share}(b)$ for $b \in \{0, 1\}$.

Lemma 4.5 (Perfect correctness). *If* DSS *satisfies perfect correctness and perfect F-fake hiding, then* PKE *satisfies perfect correctness.*

Proof. Fix $b \in \{0, 1\}$, and let S' be given by

$$S'_F \leftarrow \mathsf{DSS.Fake}(S_F^{(0)}, b), \qquad S'_R = S_R^{(0)}.$$

Perfect F-fake hiding gives

$$S' \equiv S^{(b)}.$$

By perfect correctness of DSS, we obtain

$$\begin{aligned} \Pr[\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, b)) = b] &= \Pr[\mathsf{DSS.Rec}(S') = b] \\ &= \Pr[\mathsf{DSS.Rec}(S^{(b)}) = b] \\ &= 1, \end{aligned}$$

where $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$. □

Lemma 4.6 (IND-CPA security). *If* DSS *satisfies computational t-privacy and perfect F-strong fake hiding, and $|F| < t$, then* PKE *satisfies IND-CPA security.*

Proof. Since $|F| < t$, applying t-privacy with faker set R and discarding the shares of R gives $S_F^{(0)} \approx_c S_F^{(1)}$. For each $b \in \{0, 1\}$,

$$\begin{aligned} (\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, b)) &\equiv (S_F^{(0)}, \mathsf{DSS.Fake}(S_F^{(0)}, b)) \\ &\equiv (\mathsf{DSS.Fake}(S_F^{(b)}, 0), S_F^{(b)}) \\ &\approx_c (\mathsf{DSS.Fake}(S_F^{(0)}, 0), S_F^{(0)}). \end{aligned}$$

The second $\equiv$ uses perfect strong fake hiding for $(0, b)$ and discards the snitch shares. The $\approx_c$ follows by applying the efficient randomized map $x \mapsto (\mathsf{DSS.Fake}(x, 0), x)$ to $S_F^{(b)} \approx_c S_F^{(0)}$. The final distribution is independent of b, proving IND-CPA security. □

Theorem 4.7 (PKE from Strong Fake Hiding). *If there exists a threshold-t DSS scheme satisfying perfect F-strong fake hiding for some $F \subseteq [n]$ with $|F| < t$, then an IND-CPA secure public-key encryption scheme exists.*

Proof. It suffices to construct a PKE scheme for single-bit messages. Let DSS be a threshold-t DSS scheme satisfying perfect F-strong fake hiding for some $F \subseteq [n]$ with $|F| < t$. Instantiate PKE with this DSS.

Perfect correctness. By Proposition 3.6, DSS satisfies perfect F-fake hiding. Together with perfect correctness of DSS, Lemma 4.5 gives perfect correctness of PKE.

IND-CPA security. Since DSS satisfies computational t-privacy and perfect F-strong fake hiding, and $|F| < t$, Lemma 4.6 gives IND-CPA security of PKE. □

Remark 4.8. If perfect correctness of PKE is relaxed to allow negligible decryption error, Theorem 4.7 holds with computational F-strong fake hiding.

Proof. By Proposition 3.6, computational F-strong fake hiding implies computational F-fake hiding. In the proof of Lemma 4.5, we therefore have $S' \approx_c S^{(b)}$. Perfect correctness of DSS and efficiency of $\mathsf{DSS.Rec}$ then give decryption success probability at least $1 - \mathsf{negl}(\lambda)$. In the proof of Lemma 4.6, the second $\equiv$ becomes $\approx_c$, and IND-CPA security still follows. □

Remark 4.9. Theorem 4.7 remains valid if Definition 3.4 is relaxed by replacing $\mathsf{DSS.Fake}(X_1, m_0)$ in D_1 with $\mathcal{A}(X_1, m_0, m_1)$ for some PPT algorithm $\mathcal{A}$. Use the construction above with $\mathsf{Enc}(\mathsf{pk}, b) = \mathcal{A}(\mathsf{pk}, b, 0)$. Correctness and IND-CPA security follow by the same argument.

5 Fake-Hiding Deniable Secret Sharing from One-Way Functions

From one-way functions, we construct SKE with a new property called ciphertext replaceability. We then use it to construct fake-hiding DSS. Finally, we show that one-way functions are necessary for fake-hiding DSS.

5.1 SKE with Ciphertext Replaceability from One-Way Functions

Definition 5.1 (Secret Key Encryption Scheme). *Let $\ell_m = \ell_m(\lambda)$ be a polynomially bounded message length. A secret key encryption (SKE) scheme is a tuple of PPT algorithms* $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ *such that:*

- $\mathsf{KeyGen}(1^\lambda) \to \mathsf{sk}$ *takes a security parameter λ and outputs a secret key* sk.
- $\mathsf{Enc}(\mathsf{sk}, m) \to c$ *takes a secret key* sk *and a message $m \in \{0,1\}^{\ell_m}$ and outputs a ciphertext c.*
- $\mathsf{Dec}(\mathsf{sk}, c) \to m$ *takes a secret key* sk *and a ciphertext c and outputs a message $m \in \{0,1\}^{\ell_m}$ or $\perp$.*

and satisfies perfect correctness: for all $m \in \{0,1\}^{\ell_m}$,

$$\Pr[\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{sk}, m)) = m] = 1,$$

where $\mathsf{sk} \leftarrow \mathsf{KeyGen}(1^\lambda)$.

Definition 5.2 (One-Time Security). *An SKE scheme is one-time secure if, for every* $m_0, m_1 \in \{0,1\}^{\ell_m}$,

$$\mathsf{Enc}(\mathsf{sk}, m_0) \approx_c \mathsf{Enc}(\mathsf{sk}, m_1),$$

where $\mathsf{sk} \leftarrow \mathsf{KeyGen}(1^\lambda)$.

One-time security suffices for our DSS construction.

Definition 5.3 (Ciphertext Replaceability). *An SKE scheme has statistical ciphertext replaceability if there exists a PPT algorithm* Replace *with the following syntax:*

- $\mathsf{Replace}(c, m') \to c'$ *takes a ciphertext* c *and a target message* $m' \in \{0,1\}^{\ell_m}$ *and outputs a ciphertext* c'.

The algorithm must satisfy, for every $m, m' \in \{0,1\}^{\ell_m}$,

$$\Delta((\mathsf{sk}, \mathsf{Replace}(\mathsf{Enc}(\mathsf{sk}, m), m')), (\mathsf{sk}, \mathsf{Enc}(\mathsf{sk}, m'))) \leq \mathsf{negl}(\lambda),$$

where $\mathsf{sk} \leftarrow \mathsf{KeyGen}(1^\lambda)$.

SKE with Ciphertext Replaceability from GGM Trees. Let $G : \{0,1\}^\lambda \to \{0,1\}^{2\lambda}$ and $G' : \{0,1\}^\lambda \to \{0,1\}^{\ell_m}$ be PRGs.[1] Write $G(s) = G_0(s) \| G_1(s)$, with $G_0(s)$, $G_1(s) \in \{0,1\}^\lambda$. Such G and G' exist assuming one-way functions [HILL99].

For a depth-λ GGM tree [GGM86] rooted at sk, define the label s_p of each node indexed by $p \in \{0,1\}^{\leq \lambda}$ recursively by

$$s_\epsilon = \mathsf{sk}, \qquad s_{pb} = G_b(s_p) \quad (|p| < \lambda,\ b \in \{0,1\}).$$

Let $N = 2^\lambda$. For $0 \leq i < N$, write $i_1 \cdots i_\lambda = \mathsf{bin}_\lambda(i)$ for the λ-bit binary representation of i. The corresponding GGM PRF on λ-bit integers is defined by

$$F_{\mathsf{sk}}(i) = G_{i_\lambda}(\cdots G_{i_1}(\mathsf{sk}) \cdots) = s_{i_1 \cdots i_\lambda}.$$

The label of leaf i is $F_{\mathsf{sk}}(i)$. Let $T_{\mathsf{sk}}(i)$ be the list obtained by following the path from the root to leaf i and appending the right sibling's label at each left turn. The subtrees rooted at these right siblings partition the leaves $i+1, \ldots, N-1$ (Figure 5), allowing evaluation at any later leaf.

The algorithms are given in Figure 6.

A ciphertext is a tuple (i, T, u) with index $i \in \{0, \ldots, N\}$, labels T for later evaluation, and body $u \in \{0,1\}^{\ell_m}$. We reserve $i = N$ for a terminal state with an empty list T.

Write $\mathsf{Enc}(\mathsf{sk}, m; i)$ for encryption with the sampled index fixed to i.

[1] If $\ell_m \leq \lambda$, we truncate the output of a PRG to ℓ_m bits.

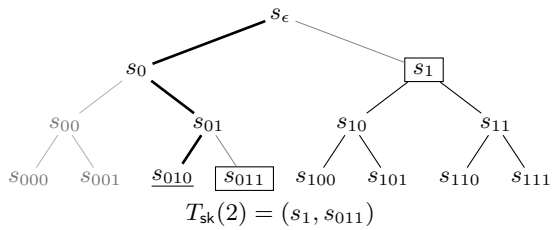

Fig. 5. GGM tree in our construction ($i = 2$).

$\underline{\mathsf{KeyGen}(1^\lambda)}$

$\mathsf{sk} \leftarrow \{0,1\}^\lambda$
return sk

$\underline{\mathsf{Enc}(\mathsf{sk}, m)}$

$i \leftarrow \{0, \ldots, N-1\}$
$(T, s) \leftarrow (T_{\mathsf{sk}}(i), F_{\mathsf{sk}}(i))$
return $(i, T, m \oplus G'(s))$

$\underline{\mathsf{Dec}(\mathsf{sk}, (i, T, u))}$

if $i = N$ **then return** u
return $u \oplus G'(F_{\mathsf{sk}}(i))$

$\underline{\mathsf{Replace}((i, T, u), m')}$

if $i \geq N-1$ **then**
  return $(N, (), m')$
$(T', s) \leftarrow \mathsf{Next}(i, T)$
return $(i+1, T', m' \oplus G'(s))$

$\underline{\mathsf{Next}(i, T) \quad (i < N-1)}$

$T' \leftarrow T$
$s \leftarrow$ last element of T'
remove the last element of T'
$i_1 \cdots i_\lambda \leftarrow \mathsf{bin}_\lambda(i)$
$\rho \leftarrow \max\{h : i_h = 0\}$
for $h = \rho+1, \ldots, \lambda$ **do**
  $(s, s') \leftarrow (G_0(s), G_1(s))$
  append s' to T'
return (T', s)

Fig. 6. SKE with Ciphertext Replaceability from GGM Trees.

Lemma 5.4. *For every* $\mathsf{sk} \in \{0,1\}^\lambda$ *and* $0 \leq i < N-1$,

$$\mathsf{Next}(i, T_{\mathsf{sk}}(i)) = (T_{\mathsf{sk}}(i+1), F_{\mathsf{sk}}(i+1)).$$

Moreover, for every key sk, *messages* $m, m' \in \{0,1\}^{\ell_m}$ *and* $0 \leq i < N$,

$$\mathsf{Replace}(\mathsf{Enc}(\mathsf{sk}, m; i), m') = \begin{cases} \mathsf{Enc}(\mathsf{sk}, m'; i+1), & i < N-1, \\ (N, (), m'), & i = N-1. \end{cases}$$

Proof. Fix sk and $0 \leq i < N-1$. Write $\mathsf{bin}_\lambda(i) = i_1 \cdots i_\lambda$ and put $\rho = \max\{h : i_h = 0\}$ and $p = i_1 \cdots i_{\rho-1}$. Then

$$\mathsf{bin}_\lambda(i) = p\,0\,1^{\lambda-\rho},$$
$$\mathsf{bin}_\lambda(i+1) = p\,1\,0^{\lambda-\rho}.$$

When the leaf index changes from i to $i+1$, the path from the root stays the same through the prefix p, then turns right instead of left and takes only left turns thereafter. Next updates the labels accordingly. Indeed, it removes the last element s_{p1} from T' and, starting from node $p1$, follows the leftmost path to leaf $i+1$, appending the right sibling's label at each step. Thus, $T' = T_{\mathsf{sk}}(i+1)$ and $s = F_{\mathsf{sk}}(i+1)$, so the algorithm returns $(T_{\mathsf{sk}}(i+1), F_{\mathsf{sk}}(i+1))$. The second claim follows from the definitions of Enc and $\mathsf{Replace}$. □

Theorem 5.5. *The above construction is an SKE scheme satisfying*

- *Perfect correctness.*
- *One-time security.*
- *Statistical ciphertext replaceability (Definition 5.3).*

Proof. **Perfect correctness.** Fix sk, m. For $0 \le i < N$,

$$\begin{aligned}\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{sk}, m; i)) &= (m \oplus G'(F_{\mathsf{sk}}(i))) \oplus G'(F_{\mathsf{sk}}(i)) \\ &= m.\end{aligned}$$

One-time security. Fix $m \in \{0,1\}^{\ell_m}$. For $h = 0, \ldots, \lambda$, define $H_h(m)$ by running $\mathsf{Enc}(\mathsf{sk}, m)$, replacing the outputs of the first h calls to G along the path to i with samples from $U_{2\lambda}$. Then $H_0(m) \equiv \mathsf{Enc}(\mathsf{sk}, m)$, where $\mathsf{sk} \leftarrow \mathsf{KeyGen}(1^\lambda)$. Finally, define $H_{\lambda+1}(m)$ by replacing the output of G' in $H_\lambda(m)$ with a sample from U_{ℓ_m}.

For $1 \le h \le \lambda$, $H_{h-1}(m)$ and $H_h(m)$ differ only at depth h. Given a PRG challenge $z \in \{0,1\}^{2\lambda}$, run $H_{h-1}(m)$ with the output of G at depth h replaced by z.

Since the label at depth $h-1$ on the path to i is sampled uniformly and used only in the hth call to G, the resulting ciphertext is distributed as $H_{h-1}(m)$ when $z \leftarrow G(U_\lambda)$, and as $H_h(m)$ when $z \leftarrow U_{2\lambda}$. Thus, the pseudorandomness of G gives

$$H_{h-1}(m) \approx_c H_h(m).$$

In $H_\lambda(m)$, the label of leaf i is uniform and independent of (i, T). By the pseudorandomness of G',

$$H_\lambda(m) \approx_c H_{\lambda+1}(m).$$

$H_{\lambda+1}(m)$ is independent of m. Combining these hybrids, for any $m_0, m_1 \in \{0,1\}^{\ell_m}$,

$$\begin{aligned}\mathsf{Enc}(\mathsf{sk}, m_0) \equiv H_0(m_0) &\approx_c H_{\lambda+1}(m_0) \\ &\equiv H_{\lambda+1}(m_1) \approx_c H_0(m_1) \equiv \mathsf{Enc}(\mathsf{sk}, m_1).\end{aligned}$$

Statistical ciphertext replaceability. By Lemma 5.4, replacement advances the ciphertext index by one. Fix sk, m, m' and define, for $0 \le j \le N$,

$$c_j = \begin{cases} \mathsf{Enc}(\mathsf{sk}, m'; j), & 0 \le j < N, \\ (N, (), m'), & j = N. \end{cases}$$

The ciphertexts $c_0, \ldots, c_N$ are distinct. $\mathsf{Replace}(\mathsf{Enc}(\mathsf{sk}, m), m')$ is uniform on $\{c_1, \ldots, c_N\}$, whereas $\mathsf{Enc}(\mathsf{sk}, m')$ is uniform on $\{c_0, \ldots, c_{N-1}\}$. Thus, their statistical distance is

$$\frac{1}{2}\left(\frac{1}{N} + \frac{1}{N}\right) = 2^{-\lambda}.$$

Since this holds for every sk, the same distance holds for the joint distributions including sk, establishing statistical ciphertext replaceability. □

5.2 Fake-Hiding DSS from SKE with Ciphertext Replaceability

Fix n, t such that $\lfloor n/2 \rfloor + 1 \leq t \leq n$. Let $\mathsf{RSKE} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Replace})$ be a secret-key encryption scheme with a ciphertext replacement algorithm. For RSKE, ciphertext replaceability is defined with respect to $\mathsf{RSKE.Replace}$. For each λ, we use fixed-length encodings of secret keys and ciphertexts, padding as needed. Let $\ell_{\mathsf{sk}} = \ell_{\mathsf{sk}}(\lambda)$ denote the secret-key length.

Let $r = \lfloor n/2 \rfloor + 1$. Let ITDSS_r and ITDSS_t be information-theoretic DSS algorithms. Using them, we construct the algorithms $(\mathsf{Share}, \mathsf{Rec}, \mathsf{Fake})$ shown in Figure 7. Write $u_i = (i, c_i, s_i)$ for a full share, where c_i and s_i are shares of the ciphertext and secret key, respectively.

$\mathsf{Share}(m)$

$\mathsf{sk} \leftarrow \mathsf{RSKE.KeyGen}(1^\lambda)$
$c \leftarrow \mathsf{RSKE.Enc}(\mathsf{sk}, m)$
$(c_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_r.\mathsf{Share}(c)$
$(s_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_t.\mathsf{Share}(\mathsf{sk})$
return $((i, c_i, s_i))_{i\in[n]}$

$\mathsf{Rec}((u_i)_{i\in Q})$

if $|Q| < t$ **then return** $\perp$
$(i, c_i, s_i) \leftarrow u_i \quad (i \in Q)$
$c^{\mathsf{rec}} \leftarrow \mathsf{ITDSS}_r.\mathsf{Rec}((c_i)_{i\in Q})$
$\mathsf{sk}^{\mathsf{rec}} \leftarrow \mathsf{ITDSS}_t.\mathsf{Rec}((s_i)_{i\in Q})$
if $c^{\mathsf{rec}} = \perp \vee \mathsf{sk}^{\mathsf{rec}} = \perp$ **then return** $\perp$
$m^{\mathsf{rec}} \leftarrow \mathsf{RSKE.Dec}(\mathsf{sk}^{\mathsf{rec}}, c^{\mathsf{rec}})$
return m^{rec}

$\mathsf{Fake}((u_i)_{i\in F}, m^{\mathsf{fake}})$

$(i, c_i, s_i) \leftarrow u_i \quad (i \in F)$
if $|F| < r$ **then**
  $(s_i^{\mathsf{fake}})_{i\in F} \leftarrow \mathsf{ITDSS}_t.\mathsf{Fake}((s_i)_{i\in F}, 0^{\ell_{\mathsf{sk}}})$
  return $((i, c_i, s_i^{\mathsf{fake}}))_{i\in F}$
$c^{\mathsf{rec}} \leftarrow \mathsf{ITDSS}_r.\mathsf{Rec}((c_i)_{i\in F})$
if $c^{\mathsf{rec}} = \perp$ **then return** $\perp$
$c^{\mathsf{fake}} \leftarrow \mathsf{RSKE.Replace}(c^{\mathsf{rec}}, m^{\mathsf{fake}})$
$(c_i^{\mathsf{fake}})_{i\in F} \leftarrow \mathsf{ITDSS}_r.\mathsf{Fake}((c_i)_{i\in F}, c^{\mathsf{fake}})$
return $((i, c_i^{\mathsf{fake}}, s_i))_{i\in F}$

Fig. 7. Algorithm for Deniable Secret Sharing using RSKE.

Lemma 5.6 (Perfect correctness). *If* ITDSS_r *and* ITDSS_t *satisfy perfect correctness at thresholds* r *and* t*, respectively, then the algorithms in Figure 7 satisfy perfect correctness.*

Proof. The correctness argument for Lemma 4.1 applies with RSKE in place of PKE. Since RSKE has perfect correctness, we obtain $\Pr[m^{\mathsf{rec}} = m] = 1$. □

Lemma 5.7 (Statistical fake hiding). *If* ITDSS_r *satisfies perfect correctness at threshold* r *and perfect* $\lfloor n/2 \rfloor$*-fake hiding, and* RSKE *satisfies statistical ciphertext replaceability, then the algorithms in Figure 7 satisfy statistical* $\lfloor n/2 \rfloor$*-fake hiding.*

Proof. Fix messages m, m' s.t. $|m| = |m'|$ and $F \subseteq [n]$ s.t. $|F| \geq r$. Then

$$|F| \geq r, \qquad n - |F| \leq n - r < r.$$

Let $D_{\mathrm{fake}}, D_{\mathrm{real}}$ be the output distributions in the definition of F-fake hiding.

H_0. Expand Share and Fake in D_{fake}, using $c^{\mathsf{rec}} = c$ by perfect correctness of ITDSS_r.

H_0

$\mathsf{sk} \leftarrow \mathsf{RSKE.KeyGen}(1^\lambda)$
$c \leftarrow \mathsf{RSKE.Enc}(\mathsf{sk}, m)$
$(c_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_r.\mathsf{Share}(c)$
$(s_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_t.\mathsf{Share}(\mathsf{sk})$
$c' \leftarrow \mathsf{RSKE.Replace}(c, m')$
$(c_i)_{i\in F} \leftarrow \mathsf{ITDSS}_r.\mathsf{Fake}((c_i)_{i\in F}, c')$
return $(m, m', F, ((i, c_i, s_i))_{i\in[n]})$

H_1. Replace the ciphertext sharing and faking in H_0 as follows.

H_1

$\vdots$
~~$(c_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_r.\mathsf{Share}(c)$~~
$(s_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_t.\mathsf{Share}(\mathsf{sk})$
$c' \leftarrow \mathsf{RSKE.Replace}(c, m')$
~~$(c_i)_{i\in F} \leftarrow \mathsf{ITDSS}_r.\mathsf{Fake}((c_i)_{i\in F}, c')$~~
$(c_i)_{i\in[n]} \leftarrow \mathsf{ITDSS}_r.\mathsf{Share}(c')$
return $(m, m', F, ((i, c_i, s_i))_{i\in[n]})$

$H_0 \equiv H_1$. By perfect fake hiding, $H_0 \equiv H_1$.

H_2. Sample c' as a fresh encryption of m'.

H_2

$\vdots$

~~$c' \leftarrow \mathsf{RSKE.Replace}(c, m')$~~

$c' \leftarrow \mathsf{RSKE.Enc}(\mathsf{sk}, m')$

$\vdots$

$\Delta(H_1, H_2) \leq \mathsf{negl}(\lambda)$. Let X_1, X_2 be the joint distributions of (sk, c') in H_1, H_2, respectively. They are

$$X_1 = (\mathsf{sk}, \mathsf{RSKE.Replace}(c, m')),$$
$$X_2 = (\mathsf{sk}, \mathsf{RSKE.Enc}(\mathsf{sk}, m')),$$

where $\mathsf{sk} \leftarrow \mathsf{RSKE.KeyGen}(1^\lambda)$ and $c \leftarrow \mathsf{RSKE.Enc}(\mathsf{sk}, m)$. Statistical ciphertext replaceability (Definition 5.3) gives $\Delta(X_1, X_2) \leq \mathsf{negl}(\lambda)$.

The only difference between H_1 and H_2 is the distribution of (sk, c'). Both generate the full shares from this pair in the same way. Applying this common randomized procedure cannot increase statistical distance, so

$$\Delta(H_1, H_2) \leq \Delta(X_1, X_2) \leq \mathsf{negl}(\lambda).$$

H_3. Remove the generation of c.

H_3

$\vdots$

~~$c \leftarrow \mathsf{RSKE.Enc}(\mathsf{sk}, m)$~~

$\vdots$

$H_2 \equiv H_3$. Since c is unused in H_2, we have $H_2 \equiv H_3 \equiv D_{\text{real}}$. Therefore,

$$\Delta(D_{\text{fake}}, D_{\text{real}}) = \Delta(H_1, H_2) \leq \mathsf{negl}(\lambda),$$

proving statistical $\lfloor n/2 \rfloor$-fake hiding.

□

Lemma 5.8 (Computational privacy). *If* ITDSS_r *satisfies perfect correctness at threshold* r *and perfect* $\lfloor n/2 \rfloor$*-fake hiding,* ITDSS_t *satisfies perfect* t*-privacy, and* RSKE *is one-time secure and satisfies statistical ciphertext replaceability, then the algorithms in Figure 7 satisfy computational* t*-privacy.*

Proof. Fix a partition $F \sqcup R = [n]$ s.t. $|R| < t$, and messages m_0, m_1, m' s.t. $|m_0| = |m_1| = |m'|$. For $b \in \{0, 1\}$, let D_b be the output distribution in the definition of t-privacy for the faker set F. We adapt the privacy proof of Lemma 4.3 to show $D_0 \approx_c D_1$.

Case $|F| \geq r$. By Lemma 5.7, statistical fake hiding gives $D_0 \approx_s D_1$.

Case $|F| < r$. Let H_b replace the sharing of sk in D_b with a sharing of $0^{\ell_{\mathsf{sk}}}$. Perfect t-privacy of ITDSS_t gives $D_b \equiv H_b$.

$H_0 \approx_c H_1$. The reduction shows $H_0 \approx_c H_1$ using the one-time security of RSKE in place of IND-CPA security, since H_b depends on sk only through $c \leftarrow \mathsf{RSKE.Enc}(\mathsf{sk}, m_b)$. Therefore,

$$D_0 \equiv H_0 \approx_c H_1 \equiv D_1,$$

proving computational t-privacy. □

Theorem 5.9 (Fake-Hiding DSS from OWFs). *If there exists a one-way function, then for every n, t such that $\lfloor n/2 \rfloor + 1 \leq t \leq n$, there exists a threshold-t deniable secret sharing scheme satisfying perfect correctness, computational t-privacy, and statistical $\lfloor n/2 \rfloor$-fake hiding.*

Proof. In Figure 7, take RSKE from Theorem 5.5 and $\mathsf{ITDSS}_r, \mathsf{ITDSS}_t$ from Section 3.2. We verify the required properties below.

Perfect correctness. By Theorem 3.8, both ITDSS_r and ITDSS_t satisfy perfect correctness. By Lemma 5.6, the algorithms in Figure 7 satisfy perfect correctness.

Statistical fake hiding. Since $n - r < r$, Theorem 3.8 and Proposition 3.6 give perfect $\lfloor n/2 \rfloor$-fake hiding of ITDSS_r. By Theorem 5.5, RSKE satisfies statistical ciphertext replaceability. By Lemma 5.7, the algorithms in Figure 7 satisfy statistical $\lfloor n/2 \rfloor$-fake hiding.

Computational privacy. By Theorem 3.8, ITDSS_t satisfies perfect t-privacy. Since RSKE is one-time secure by Theorem 5.5, Lemma 5.8 gives computational t-privacy for the algorithms in Figure 7. □

5.3 One-Way Functions from Fake-Hiding DSS

We write $H(X)$ and $H(X \mid Y)$ for Shannon entropy and conditional Shannon entropy, respectively, measured in bits. We denote the binary entropy function by h_2.

Lemma 5.10 (Continuity of conditional entropy [AS20]). *Let (U, V) and (U', V') be pairs of random variables taking values in the same finite set $\mathcal{U} \times \mathcal{V}$, and put*

$$\varepsilon = \Delta\big((U, V), (U', V')\big).$$

Then

$$\big|H(U \mid V) - H(U' \mid V')\big| \leq \varepsilon \log_2 |\mathcal{U}| + h_2(\varepsilon).$$

Proof. The case $\varepsilon = 0$ is immediate. For $0 < \varepsilon \leq 1 - 1/|\mathcal{U}|$, the claim follows from [AS20, Eq. (7)]. For $\varepsilon > 1 - 1/|\mathcal{U}|$, we have

$$h_2(\varepsilon) \geq -(1 - \varepsilon) \log_2 (1 - \varepsilon) \geq (1 - \varepsilon) \log_2 |\mathcal{U}|.$$

Thus

$$\big|H(U \mid V) - H(U' \mid V')\big| \leq \log_2 |\mathcal{U}| \leq \varepsilon \log_2 |\mathcal{U}| + h_2(\varepsilon).$$

□

Theorem 5.11 (Goldreich [Gol90]). *Let $\mathcal{D}_0 = \{D_{0,\lambda}\}$ and $\mathcal{D}_1 = \{D_{1,\lambda}\}$ be ensembles of strings whose lengths are polynomially bounded in λ. Suppose that:*

1. ***Efficient sampling.*** *For each $b \in \{0,1\}$, there is a PPT algorithm that, on input 1^λ, samples $D_{b,\lambda}$.*
2. ***Computational indistinguishability.*** *$\mathcal{D}_0 \approx_c \mathcal{D}_1$.*
3. ***Noticeable statistical distance.*** *There is a positive polynomial p such that, for all sufficiently large λ, $\Delta(D_{0,\lambda}, D_{1,\lambda}) \geq 1/p(\lambda)$.*

Then pseudorandom generators, and hence one-way functions, exist.

For the following two lemmas, fix a threshold-t DSS scheme $\mathsf{DSS} = (\mathsf{Share}, \mathsf{Rec}, \mathsf{Fake})$ for n shareholders. Fix a partition $F \sqcup R = [n]$ with $|F| < t$.

For $b \in \{0,1\}$, consider the following privacy experiment:

$\underline{D_b^{\mathrm{priv}}}$
$S \leftarrow \mathsf{DSS.Share}(b)$
return S_F

Write $\delta_{\mathrm{P}} = \Delta(D_0^{\mathrm{priv}}, D_1^{\mathrm{priv}})$.

Lemma 5.12. *For every λ, we have*

$$H(b \mid S_F) \geq 1 - \delta_{\mathrm{P}},$$

where $b \leftarrow \{0,1\}$ and $S \leftarrow \mathsf{DSS.Share}(b)$.

Proof. We have

$$\begin{aligned} H(b \mid S_F) &= \mathbb{E}_{S_F}[h_2(\Pr[b = 0 \mid S_F])] \\ &\geq 1 - \mathbb{E}_{S_F}[|\Pr[b = 0 \mid S_F] - \Pr[b = 1 \mid S_F]|] \\ &= 1 - \frac{1}{2} \sum_s |\Pr[S_F = s \mid b = 0] - \Pr[S_F = s \mid b = 1]| \\ &= 1 - \delta_{\mathrm{P}}, \end{aligned}$$

where the inequality uses $h_2(u) \geq 1 - |2u - 1|$, and the second equality follows from Bayes' rule and the uniformity of b. □

Consider the following fake-hiding experiments:

D_{fake}	D_{real}
$b, b' \leftarrow \{0,1\}$	$b \leftarrow \{0,1\}$
$S \leftarrow \mathsf{DSS.Share}(b)$	$S \leftarrow \mathsf{DSS.Share}(b)$
$S'_F \leftarrow \mathsf{DSS.Fake}(S_F, b')$	**return** (b, S_F, S_R)
return (b', S'_F, S_R)	

Write $\delta_{\mathrm{F}} = \Delta(D_{\text{fake}}, D_{\text{real}})$.

Lemma 5.13. *Let $L = L(\lambda) \geq 1$ be a polynomial upper bound on the length of a binary encoding of S_R. If* DSS *satisfies perfect correctness, then for every λ,*

$$H(b \mid S_F) \leq (L+1)\delta_{\mathrm{F}} + h_2(\delta_{\mathrm{F}}),$$

where $b \leftarrow \{0,1\}$ and $S \leftarrow \mathsf{DSS.Share}(b)$.

Proof. Sample $b, b' \leftarrow \{0,1\}$ and $S \leftarrow \mathsf{DSS.Share}(b)$. Set $S'_F \leftarrow \mathsf{DSS.Fake}(S_F, b')$. The distributions of (b', S'_F, S_R) and (b, S_F, S_R) are D_{fake} and D_{real}, respectively. We have

$$\begin{aligned} H(S_R \mid S_F) &= H(S_R \mid S_F, S'_F, b') \\ &\leq H(S_R \mid S'_F, b') \\ &\leq H(S_R \mid S_F, b) + (L+1)\delta_{\mathrm{F}} + h_2(\delta_{\mathrm{F}}). \end{aligned}$$

The equality holds because $\mathsf{DSS.Fake}$ uses only S_F, b', and independent randomness. The last inequality follows from Lemma 5.10, since S_R has fewer than 2^{L+1} possible encodings.

Perfect correctness of DSS gives $H(b \mid S_F, S_R) = 0$. Thus, by the chain rule,

$$\begin{aligned} H(b \mid S_F) &= H(S_R \mid S_F) - H(S_R \mid S_F, b) \\ &\leq (L+1)\delta_{\mathrm{F}} + h_2(\delta_{\mathrm{F}}). \end{aligned}$$

□

Theorem 5.14 (OWFs from Fake-Hiding DSS). *If, for some fixed n, t and $F \subseteq [n]$ with $|F| < t$, there exists a threshold-t DSS scheme* DSS *for n shareholders satisfying computational t-privacy and computational F-fake hiding, then one-way functions exist.*

Proof. We construct two distributions satisfying the conditions of Theorem 5.11. Using the experiments from Lemmas 5.12 and 5.13, define D_0 and D_1 as follows:

D_0	D_1
$c \leftarrow \{0,1\}$	$c \leftarrow \{0,1\}$
if $c = 0$ **then**	**if** $c = 0$ **then**
$\quad X \leftarrow D_0^{\text{priv}}$	$\quad X \leftarrow D_1^{\text{priv}}$
else	**else**
$\quad X \leftarrow D_{\text{fake}}$	$\quad X \leftarrow D_{\text{real}}$
return (c, X)	**return** (c, X)

Both distributions are efficiently samplable and have polynomial output length.

Computational indistinguishability. Applying t-privacy with faker set R and discarding S_R gives

$$D_0^{\text{priv}} \approx_c D_1^{\text{priv}}.$$

By F-fake hiding of DSS,

$$D_{\text{fake}} \approx_c D_{\text{real}}.$$

These comparisons apply to the branches $c = 0$ and $c = 1$, respectively. Hence $D_0 \approx_c D_1$.

Statistical separation. Let L be as in Lemma 5.13. For $b \leftarrow \{0,1\}$ and $S \leftarrow \mathsf{DSS.Share}(b)$, Lemmas 5.12 and 5.13 give

$$1 - \delta_{\text{P}} \leq H(b \mid S_F) \leq (L+1)\delta_{\text{F}} + h_2(\delta_{\text{F}}).$$

Using $h_2(x) \leq 1/2 + 2x$, we obtain

$$\frac{1}{2} \leq (L+3)(\delta_{\text{P}} + \delta_{\text{F}}).$$

Hence, by the definitions of D_0 and D_1,

$$\Delta(D_0, D_1) = \frac{\delta_{\text{P}} + \delta_{\text{F}}}{2} \geq \frac{1}{4(L+3)}.$$

Since L is polynomial, Theorem 5.11 yields one-way functions. □

References

ABDC+26. I. Amores, T. Ben David, M. Campanelli, S. Kolby, E. Omri, A. Paskin-Cherniavsky, and S. Yakoubov. Deniable secret sharing: Towards practical faking. In *TCC 2026, Part I*, *LNCS* 17191, pages 60–83. Springer, Cham, 2026.

AS20. M. A. Alhejji and G. Smith. A tight uniform continuity bound for equivocation. In *2020 IEEE International Symposium on Information Theory (ISIT)*, pages 2270–2274. IEEE, 2020.

CDK+25. R. Canetti, I. Damgård, S. Kolby, D. Ravi, and S. Yakoubov. Deniable secret sharing. In *TCC 2025, Part II*, *LNCS* 16269, pages 399–427. Springer, Cham, December 2025.

GGM86. O. Goldreich, S. Goldwasser, and S. Micali. How to construct random functions. *Journal of the ACM*, 33(4):792–807, October 1986.

Gol90. O. Goldreich. A note on computational indistinguishability. *Information Processing Letters*, 34(6):277–281, 1990.

HILL99. J. Håstad, R. Impagliazzo, L. A. Levin, and M. Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.

Nas26. S. Nassar. A note on non-interactive deniable secret sharing from indistinguishability obfuscation. Cryptology ePrint Archive, Paper 2026/2884, 2026.

Sha79. A. Shamir. How to share a secret. *Communications of the Association for Computing Machinery*, 22(11):612–613, November 1979.