# Conditional Disclosure of Secrets

## In the Context of General Secret-Sharing

Nikunj Gupta, Nikhil Sheoran
University of Illinois at Urbana-Champaign
{nikunj,sheoran2}@illinois.edu

## Abstract

In this report [1], we look at the recent advances around the cryptographic primitive *Conditional Disclosure of Secrets* and how these advances have led to breakthrough developments in the field of *General Secret Sharing*. The report glances over the construction of general secret sharing schemes and conditional disclosure of secrets (CDS). The report then discusses the advances in CDS along with other optimizations to break the 30-year-old complexity barrier for general secret sharing from $2^n$ to $2^{0.994n}$. Finally, we discuss the more recent works that bring this upper bound down to $2^{0.637n}$. A large gap between the lower and the upper bound for general secret sharing still persists and is an open research problem.

## 1 Introduction

Secret sharing is a formal method using which a dealer can distribute a secret $s$ to $n$ parties such that only the authorized subsets of parties can reconstruct the secret. Secret sharing forms the building blocks for various cryptographic protocols such as multiparty computations, byzantine agreement, threshold cryptography, and generalized oblivious transfer. More recently, advances in conditional disclosure of secrets have lead to an increased interest in tightening the upper bounds of the general secret sharing schemes.

Conditional disclosure of secret (CDS) allows a set of parties to conditionally (based on a predicate) disclose a secret to a third party. CDS finds its use in various general cryptographic design including but not limited to Private Information Retrieval (PIR).

The report is organized as follows. Section 2 discusses some basic concepts that will be used throughout the report. Section 3 formally explains the first secret sharing scheme introduced by Shamir [17] and its generalization proposed by Ito et al. [12]. We also look at the performance of these schemes in terms of communication complexity. Section 4 introduces the concept of conditional disclosure of secrets and discusses various properties and example constructions. Section 5 describes a breakthrough sub-exponential construction of CDS using concepts from Private Information Retrieval. Using the concepts introduced in Section 4 and Section 5, Section 6 describes the proof to tighten the secret sharing upper bound using slice functions and a clever decomposition of access structure. Finally, Section 7 discusses some other recent works in the field of conditional disclosure of secrets that have further improved the complexity upper bound.

[1] Majority of the content of this report was inspired from the seminal PhD Thesis of Tianren Liu, advised by Vinod Vaikuntanathan [13] and the presentations by Benny Applebaum in the 10th BIU Winter School on Cryptography [1]

## 2 Preliminaries

### 2.1 Monotone Functions

Let $x, y \in \{0,1\}^n$. A function $f : \{0,1\}^n \to \{0,1\}$ is said to be monotone iff $x \preceq y \implies f(x) \leq f(y)$. Here, the operation $\preceq$ is defined such that $x \preceq y \implies \forall i, x_i \leq y_i$.

### 2.2 Monotone Access Structure

An access structure $A$ is monotone if $\forall B, C$ if $B \in A$ and $B \subseteq C$ then $C \in A$.

### 2.3 Slice Functions

A function $f : \{0,1\}^n \to \{0,1\}$ is a slice function if there exists a threshold $t$ such that

$$x_1 + x_2 + .. + x_n < t \implies f(x) = 0$$
$$x_1 + x_2 + .. + x_n > t \implies f(x) = 1.$$

The behavior when $x_1 + x_2 + .. + x_n = t$ is not defined. In this report, unless specified otherwise, consider $t = n/2$.

### 2.4 Formula and Circuits

Given a function $f : \{0,1\}^n \to \{0,1\}$ that we want to compute. A boolean circuit is a Directed Acyclic Graph (DAG) with wires carrying a value in $\{0,1\}$ and these values are processed by gates computing certain boolean operations.

A formula is a circuit where each gate has a fan-out of at most 1. A key result here is that every boolean function with $n$-bit input has a formula in the basis of $(\wedge, \vee)$ of size at-most $2^n$.

## 3 Secret Sharing

Formally, a secret sharing scheme involves a dealer holding a secret $s$, a set of $n$ parties, and a collection $A$ of subsets of parties called the access structure. An access structure is a set of authorized set of parties where any authorized set of parties can reconstruct the secret $s$. Therefore, a secret sharing scheme for an access structure $A$ is a method by which the dealer distributes shares to the parties such that:

1. Any subset of parties $B \in A$ is able to reconstruct the secret $s$.

2. Any subset of parties $B \notin A$ cannot obtain any information about the secret $s$.

The performance parameter for any secret sharing schemes is the size of the secret share held by each party involved.

### 3.1 General Secret Sharing

Ito et al. [12] defined a general secret-sharing scheme realizing any monotone access structure. The secret sharing scheme involves any monotone access structure $A$, a dealer with a secret key $s \in \{0,1\}$ and parties $p_1, ..., p_n$. Mathematical construction for any authorized set $B \in A$ where $B$ contains parties $\{p_{B_1}, ..., p_{B_l}\}$ is as follows:

1. The dealer chooses $l-1$ random bits $r_1, ..., r_{l-1}$
2. The dealer computes $r_l = s \oplus r_1 \oplus ... \oplus r_{l-1}$
3. Gives party $p_{B_i}$ the bit $r_i$ where $p_{B_i}$ is the i-th party of the authorized set $B$

The reconstruction phase involves calculating XORs of the parties involved in the reconstruction phase. If the parties involved belongs to the access structure $A$, the computed XOR will reveal the secret key $s$. Otherwise, the XOR will reveal no secret. The number of bits held by each party $p_i$ in this secret sharing scheme is the number of authorized sets that contains the party $p_i$. A trivial optimization over this sharing scheme is to share the secret $s$ with the minimal authorized sets. For example, consider $n/2$-out-of-$n$ minimal access structure scenario, i.e.

$$A_{n/2} = \{B \subset \{p_1, ..., p_n\} : |B| \geq n/2\}$$

The number of bits that each party receives in the above described scheme can be written as ${}^{n-1}C_{n/2-1}$, i.e., $O(2^n/\sqrt{n})$ share size complexity.

### 3.2 Shamir Secret Sharing

Shamir Secret Sharing [17], also known as $(n,t) - threshold$ secret sharing, is a secret sharing scheme where any set of $t$ or more parties can recover the secret, and no subsets of fewer than $t$ parties can learn any information about the secret whatsoever. In the context of general secret sharing, the monotone $f$ can be viewed as a slice function such that the secret can be reconstructed using an addition function with an output 1 when at least $t$ access bits out of $n$ are 1. Shamir Secret Sharing proposed a construction by using a degree $t-1$ polynomial and reconstruction of the polynomial using t points through Lagrange interpolation.

### 3.3 Secret Sharing for a General Access Structure

As described earlier in Section 3, the major performance criterion for a secret sharing scheme is the communication complexity, i.e. the message size held by each party. While Shamir secret sharing has a secret share size of $O(n)$, the construction and reconstruction phase does not work on any general access structures. For any general access structure, the first known upper bound was given by Ito et al. [12] as $O(2^n)$ using monotonic functions. The share size can further improve to $O(2^n/\sqrt{n})$ [6] for $n/2$-out-of-$n$ minimal access structures . Similarly, there exists an explicit monotone function that requires a share size of $\Omega(n/\log n)$ [9]. No better lower bounds are currently known to this problem.

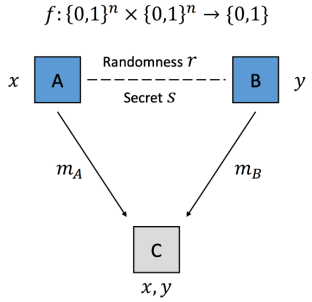



**Figure 1.** Conditional Disclosure of Secret

It can be observed that there is a drastic difference between the known lower and upper bounds. Recent advances in Conditional Disclosure of Secrets and Private Information Retrieval has led to an improved upper bounds on general secret sharing schemes. The next few sections discuss the advances and the improved upper bounds.

## 4 Conditional Disclosure of Secrets

The Conditional Disclosure of Secrets (CDS) primitive was introduced by Gertner et al. [11] in the context of Symmetrically Private Information Retrieval (SPIR) and more generally as a building block for designing more general cryptographic protocols. CDS allows a set of parties to disclose a shared secret to an external party, subject to the inputs of these parties satisfying a pre-defined predicate. The external party knows the inputs of all the parties to evaluate the pre-defined predicate whereas each individual party only has access to their own input.

More concretely, consider three parties Alice, Bob and Charlie. Alice and Bob have a shared secret $s$ and a shared randomness $r$. They also have inputs $x$ and $y$ respectively. The goal of the protocol is for Alice and Bob to send one message each to Charlie ($m_A$ and $m_B$) such that Charlie is able to recover the secret $s$ using $(x, y, m_A, m_B)$, iff the inputs $(x,y)$ satisfy a pre-defined predicate $f$ i.e. $f(x,y) = 1$. Note that no communication is allowed between Alice and Bob except for the shared randomness $r$. Figure 1 shows a schematic of the CDS primitive. The following properties must hold:

- **Correctness:** If $f(x,y) = 1$, then $\forall$ $s$ and $r$, Charlie should be able to recover $s$ by using $(x, y, m_A, m_B)$.
- **Privacy:** If $f(x,y) = 0$, then for any two secrets $s_0$ and $s_1$, the random variables $(m_A^{s_0}, m_B^{s_0})$ and $(m_A^{s_1}, m_B^{s_1})$ should be identically distributed. That is, $(x, y, m_A, m_B)$ should reveal no information about the secret $s$.

The performance of a CDS protocol is evaluated in terms of the maximal communication complexity i.e. the size of the messages $|m_A| + |m_B|$ sent in the protocol.
*Note:* In the definition above, the secret $s$ was shared across all the parties. Such a formulation is more precisely called *conditional disclosure of a common secret*. A more general version can be defined where only a subset of parties knows the secret $s$. For the context of this report, when we say *conditional disclosure of secret*, we refer to *conditional disclosure of common secret*.

### 4.1 Multi-Party CDS

The above defined 2-party CDS can naturally be extended to multiple parties. Consider an $n$-party CDS having $n$ senders with a shared randomness $r$ and a shared secret $s$ where sender $i$ has input $x_i$ and sends the message $m_i(x_i)$. Given $\mathbf{x} = [x_1, x_2, .., x_n]$ and $\mathbf{m} = [m_1(x_1), m_2(x_2), .., m_n(x_n)]$, Charlie should be able to recover the secret $s$ iff $f(\mathbf{x}) = 1$.

### 4.2 Examples

Let us look at a few example CDS constructions.

***Example 1 (XOR):*** Alice and Bob have 1-bit inputs $x$ and $y$ and a 1-bit secret $s$. The predicate $f$ is defined as XOR of $x$ and $y$, i.e., $f = x \oplus y$.

Let the shared randomness $r$ be a 2-bit vector, $r = (r_0, r_1)$. Let $m_A = s \oplus r_x$ and $m_B = r_{1-y}$.
If $x \oplus y = 1 \implies x = 1 - y \implies r_x = r_{1-y}$.
$m_A \oplus m_B = s \oplus r_x \oplus r_{1-y} = s$ (XOR of same bit).
If $x \oplus y = 0 \implies x = y$.
$m_A \oplus m_B = s \oplus r_x \oplus r_{1-y} = s \oplus r_0 \oplus r_1$ (Random Bit).

Hence, satisfying both the correctness and privacy properties. The communication cost is 2-bits.

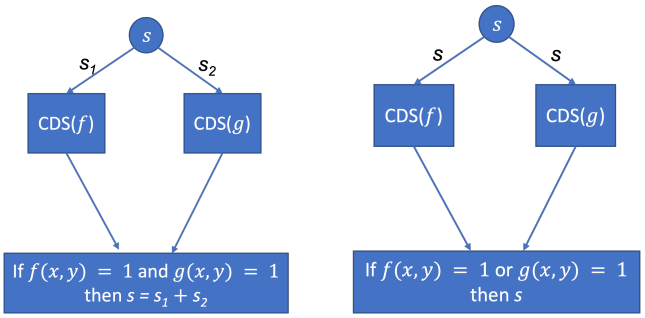
***Example 2 (Equality):*** Alice and Bob have $n$-bit inputs $x$ and $y$ and a $k$-bit secret $s$. The predicate $f$ is defined as Equality of $x$ and $y$ i.e. $f = (x == y)$.

Let the shared randomness $r$ be a hash function $h$ chosen from a family of hash functions $H$ with $k$-bit output. Let $m_A = h(x)$ and $m_B = h(y) \oplus s$.
If $x = y$, then $m_A \oplus m_B = h(x) \oplus h(y) \oplus s = s$. (Same Hash)
If $x \neq y$, then $m_A \oplus m_B = h(x) \oplus h(y) \oplus s$ (Random k-bits)

Hence, satisfying both the correctness and privacy properties. The communication cost is $2k$ bits.

***Example 3 (INDEX):*** Alice has a database $D$ of size $N = 2^n$ with 1-bit information and Bob has an $n$-bit index $x$. Both of them have a 1-bit secret $s$. The predicate $f$ is defined as the INDEX on the database $D$ i.e. $f = D(x)$. If the entry in the database at index $x$ is 1, then Charlie should be able to recover the secret $s$, otherwise not.

Let the shared randomness be of size $N$. Let us define $m_A = \{r_i \oplus s : D_i = 1\}$ and $m_B = r_x$.
If $D(x) = 1$, then $r_x \oplus s$ would be present in $m_A$. Charlie will obtain $s$ using $r_x \oplus s \oplus m_B = s$.




**(a)** Closure Under AND **(b)** Closure Under OR

**Figure 2.** Closure Properties of CDS

If $D(x) = 0$, then all bits of $m_A$ and $m_B$ are random bits.

Hence, satisfying both the correctness and privacy properties. The communication cost is $2^n$ bits from Alice and 1-bit from Bob. Gay et al. [10] more generally show that for any $t \leq 2^n$, there exists a linear CDS construction with communication complexity of $t$ and $2^n/t$. To optimize total communication complexity, we set $t = 2^{n/2}$ giving a total communication complexity of $O(2^{n/2})$ bits.

### 4.3 Closure Properties

Let us look at a few basic closure properties of CDS.

***Closure under AND:*** Given two CDS implementations corresponding to predicate $f$ and predicate $g$, we want to implement the predicate $f \wedge g$ such that the secret $s$ should now be shared iff both the predicates $f$ and $g$ evaluate to true on the inputs $x$ and $y$. Figure 2a shows a simple way to achieve this. The original secret $s$ can be split into two shares $s_1$ and $s_2$ such that $s = s_1 + s_2$. Now, the individual CDS implementing predicates $f$ and $g$ will have $s_1$ and $s_2$ as their secrets. If both $f$ and $g$ evaluate to true, the client can obtain $s_1$ and $s_2$ and thus reconstruct $s$. Otherwise, the client obtains none or only one share of the secret ($s_1$ or $s_2$). Regarding communication complexity, complexity of $CDS(f \wedge g) \leq CDS(f) + CDS(g)$.

***Closure under OR:*** Given two CDS implementations corresponding to predicate $f$ and predicate $g$, we want to implement the predicate $f \vee g$ such that the secret $s$ should now be shared if either of the predicates $f$ and $g$ evaluate to true on the inputs $x$ and $y$. Figure 2b shows a simple way to achieve this. The original secret $s$ can be duplicated across both the CDS implementing $f$ and $g$. If either of $f$ and $g$ evaluate to true, the client will obtain the secret $s$. Regarding communication complexity, complexity of $CDS(f \vee g) \leq CDS(f) + CDS(g)$.

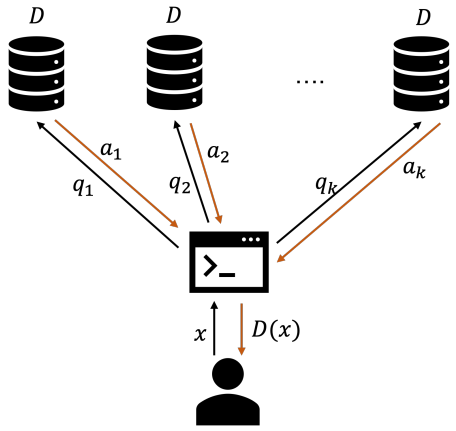



**Figure 3.** Schematic of $k$-server PIR

### 4.4 CDS for a General Boolean Function $f$

Let us look at how we can implement CDS for a general boolean function $f$. Consider a function $f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$.

***Construction 1:*** Gertner et al. [11] show that for a function $f$ with a formula of size $F$, a CDS protocol can be constructed to share a 1-bit secret $s$ with both the communication and randomness complexity bounded by $F$. For a boolean function with $2n$ bits of input, the formula size (total number of leaves) is bounded by $2^{2n}$. Using the construction for AND and OR seen in the previous section, one can construct a linear CDS with $CDS(f) \leq 2^{2n}$.

***Construction 2:*** An optimal linear construction can be obtained by a standard reduction of any general predicate $f$ to the INDEX (Example 3 from Sec. 4.2) predicate on $2^n$-dimensional vectors. Alice treats the truth table $f(x, \cdot)$ as a vector of length $2^n$ and Bob treats $y \in \{0,1\}^n$ as an index, so that the INDEX predicate returns $f(x, y)$. Then, we can use the $(t, 2^n/t)$-linear CDS construction for INDEX predicate from [10]. With such a construction, we can construct a linear CDS with $CDS(f) \leq 2^{n/2}$.

The above construction is the optimal linear CDS construction based on the product bound of $|m_A|.|m_B| = \Omega(2^n)$ shown by Gay et al. [10] on the message complexity for linear reconstruction. In the next section, we will see a sub-exponential CDS construction with a communication complexity of $2^{O(\sqrt{n})}$. This construction "breaks the linear reconstruction" barrier by using a non-linear reconstruction developed in the context of information-theoretic private information retrieval [15] and serves as a building block for general secret-sharing.

## 5 Sub-Exponential CDS using PIR

### 5.1 Private Information Retrieval

Private Information Retrieval (PIR) [8] allows a client to retrieve a single entry at index $x$ of a database $D$ of size $N$ held by $k$-servers without letting the individual servers being able to identify the index $x$. Mathematically, a $k$-server PIR scheme for a database $D$ of length $N$ consists of:
1. $k$ query functions: mapping the input index $x$ to $k$ queries $\{q_1, q_2, .., q_k\}$ where $q_i \in \{0,1\}^{l_q}$,
2. $k$ answer functions that returns the answers $\{a_1, a_2, .., a_k\}$ where $a_i \in \{0,1\}^{l_a}$ corresponding to each of the $k$ queries, and
3. a reconstruction function that takes the input $x$, the set of generated queries and the received answers to reconstruct the entry located at the index $x$ in the database $D$. Figure 3 shows a schematic of the $k$-server PIR. The following properties must hold:

- **Correctness:** For every $D \in \{0,1\}^N$, $x \in [N]$ and randomness $r$, the reconstruction function should be able to recover $D[x]$.
- **Privacy:** For any two indexes $i$ and $j$ ($i \neq j$), the distribution of the obtained query $q \in \{0,1\}^{l_q}$ should be identical.

In the context of this report, when we refer to PIR, we refer to 2-server PIR where the database $D$ is replicated at the two servers (which do not communicate) and the client obtains the value at the index $x$ in the database $D$ using one round of communication (total 2 queries and 2 answers).

### 5.2 PIR Encoding

Based on prior PIR protocols, Liu et al. [13] extract a structure called PIR Encoding. An $(N, l)$-PIR encoding is a collection of $N$ length-$l$ vectors $\{\mathbf{u_1}, \mathbf{u_2}.., \mathbf{u_N}\}$ such that for any database $D$, there exists a length preserving function $H_D$ and $\forall x$ and $\mathbf{r}$, we have $\langle H_D(\mathbf{u_x} + \mathbf{r}), \mathbf{u_x}\rangle - \langle H_D(\mathbf{r}), \mathbf{u_x}\rangle = D(x)$ where $\langle , \rangle$ denotes the inner product. The above equation with a 1-bit secret $s$ implies

$$\langle H_D(s\mathbf{u_x} + \mathbf{r}), \mathbf{u_x}\rangle - \langle H_D(\mathbf{r}), \mathbf{u_x}\rangle = s.D(x)$$

### 5.3 2-Party CDS based on PIR Encoding

Based on the above $(N, l)$-PIR encoding, a CDS protocol with communication complexity $O(l)$ can be constructed. Let the shared randomness be $\mathbf{r_1}$ and $\mathbf{r_2}$. Alice sends $\mathbf{m_A} = H_D(\mathbf{r_1}) + \mathbf{r_2}$. Bob sends a tuple of message $m_B = (\mathbf{m_{B1}}, m_{B2})$ where $\mathbf{m_{B1}} = s\mathbf{u_x} + \mathbf{r_1}$ and $m_{B2} = \langle \mathbf{u_x}, \mathbf{r_2}\rangle$. Charlie can compute $s$ by evaluating $sD(x)$ as follows:

$$sD(x) = \langle H_D(\mathbf{m_{B1}}), \mathbf{u_x}\rangle - \langle H_D(\mathbf{m_A})\mathbf{u_x}\rangle + m_{B2}$$

When $D(x) = 1$, Charlie obtains the secret $s$, otherwise it obtains 0. The additional randomness $\mathbf{r_2}$ is added to preserve the privacy property of CDS.

Liu et al. [15] showed different length PIR-encoding and corresponding CDS construction with communication complexity $O(2^{n/2})$ (linear reconstruction), $O(2^{n/3})$ complexity (quadratic reconstruction) and $2^{O(\sqrt{nlogn})}$ (using matching-vector families). For more details about PIR-encoding of sub-exponential length (sub-polynomial in N), one can refer to Section 3.4 of [15].

## 6 Breaking the Secret-Sharing Barrier [2]

With the knowledge of CDS and the recent breakthroughs involving sub-exponential CDS, the goal now is to implement General Secret Sharing for any monotone function $f$ with an upper-bound complexity better than $2^n$. The general idea is to decompose a general family of functions to a more restricted family of functions such that any function in the former family can be computed by combining several functions in the latter family with basic boolean operations. The reduction step is continued till we reach a function family for which secret sharing can be implemented using a CDS scheme. First we look at a secret sharing scheme for monotone slice functions using CDS, and then we look at the access structure decomposition that enabled $2^{cn}$ size secret shares.

### 6.1 Secret Sharing for Monotone Slice Functions

For every monotone slice access structure, there exists a secret sharing scheme with total share size $2^{O(\sqrt{n})}$ [7, 14].

First, we will look at the notion of a partition and being balanced with respect to a partition. Let $\Pi$ be an even partition of $n$ parties into $k = \sqrt{n}$ buckets each of size $n/k = \sqrt{n}$. A set $T$ is balanced with respect to a partition of parties $\Pi$ if each bucket of $\Pi$ contains the same number of parties from $T$. Since, it is a slice function, we have $|T| = n/2$. Hence, each partition should contain $n/2k = \sqrt{n}/2$ parties from $T$.

Given the monotone slice function $F$, a monotone slice function on a partition $\Pi$ is defined as:

$$F_\Pi(T) = \begin{cases} 0, & \text{if } |T| = n/2, \text{ and } T \text{ is not balanced wrt } \Pi \\ F(T), & \text{otherwise} \end{cases}$$

Liu et. al.[13] argue that there is a collection of $L = 2^{O(\sqrt{n})}$ partitions such that every subset $T$ with $|T| = n/2$ is balanced wrt some partition in the collection $L$. Therefore, the monotone slice function $F$ can be written as $F = \vee F_{\Pi^l}$.

One can construct a secret sharing scheme for each of the functions $F_{\Pi^l}$ and use them to construct a secret sharing scheme for $F$. Each of the function $F_{\Pi^l}$ can be implemented using a $k = \sqrt{n}$ parties CDS scheme where each party has an input string $x \in \{0,1\}^{n/k}$ as follows:

- Dealer picks a random bit $\sigma_j$ for each of the $k$ buckets and shares it using a $b/2$-out-of-$b$ threshold secret sharing scheme where $b$ is the number of parties in each bucket. *This is done so that if a partition $j$ is balanced (or have more than $b/2$), then only we would be able to recover the secret bit $\sigma_j$.*
- Dealer defines $\sigma^* = s \oplus \sigma_1 \oplus .... \oplus \sigma_k$ and for each bucket $j$ for all possible input strings $x$, uses the same randomness $R$ and $\sigma^*$ and additively shares it with each of the parties indexed by $x$ in the bucket $j$. *This is done so that for each bucket $j$ we are able to obtain the $j$th CDS message for it if it is a valid $x$ string. These messages combined across all $k$-parties (hence a valid $T$) can be used to obtain $\sigma^*$ and then $s$.*
- Dealer shares secret $s$ using an $(n/2+1)$-out-of-n threshold secret sharing scheme. *This is done for cases where $|T| > n/2$ and slice function evaluates to 1.*

Using the construction given above, secret sharing can be implemented for monotone slice functions with the share size of complexity $2^{O(\sqrt{n})}$.

To extend secret sharing of monotone slice functions to all monotone functions, the idea is to partition the access structure $A$ into two access structures - (1) A slice access structure $A_1$ and (2) A top-and-bottom access structure $A_2$ which captures sets of size either less than or more than $n/2$. The idea is to use the above defined secret sharing scheme for $A_1$ and a trivial secret sharing scheme with the share size proportional to the number of sets for $A_2$. One major problem with this approach is that the size of $A_2$ is about $2^n.(1 - 1/\sqrt{n})$. To counter this, Liu et al. introduce what is called the *fat-slice* function. A *fat-slice* function with $a \le b$ assigns 0 to all sets with size less than $a$ and 1 to all sets with size greater than $b$. A slice-function can be seen as a special case of fat-slice function with $a = b = n/2$.

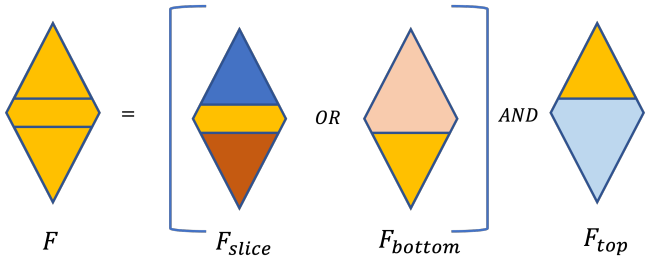



**Figure 4.** Access Structure Decomposition [1]

### 6.2 Access Structure Decomposition

Based on the fat-slice function defined above, we can now for any $\delta \in (0, n/2)$ decompose the original access structure into (1) A fat-slice access structure $A_1$ with $a = n/2 - \delta$ and $b = n/2 + \delta$ and (2) top-and-bottom access structures. Figure 4 shows the decomposition of $F$ into $F_{slice}$, $F_{bottom}$ and $F_{top}$. The functions are defined as follows:

$$F_{top}(T) = \text{Largest monotone } F(T) \text{ where } |T| > n/2 + \delta.$$

[2] The section name inspired from [14]

$$F_{slice}(T) = \begin{cases} 1, & \text{it } |T| > n/2 + \delta \\ F(T), & \text{it } |T| \in [n/2 - \delta, n/2 + \delta] \\ 0, & \text{it } |T| < n/2 - \delta \end{cases}$$

$F_{bottom}(T)$ = Smallest monotone $F(T)$ where $|T| < n/2 - \delta$.

With this decomposition, the original monotone function $F$ can be evaluated as $F = (F_{slice} \vee F_{bottom}) \wedge F_{top}$. To verify the correctness of this decomposition, consider the three cases:

***Case 1:*** $|T| > n/2 + \delta$ : We have $F_{top}(T) = F(T)$ and $F_{slice}(T) = 1$, giving, $F = 1 \wedge F_{top}(T) = F(T)$.

***Case 2:*** $|T| \in [n/2 - \delta, n/2 + \delta]$ : We have $F_{slice} = F(T)$ and since $F_{bottom} \leq F \leq F_{top}$, $F = (F(T) \vee F_{bottom}(T)) \wedge F_{top}(T) = F(T)$.

***Case 3:*** $|T| < n/2 - \delta$ : We have $F_{bottom} = F(T)$ and $F_{slice}(T) = 0$. Also, $F_{top} \geq F$, so, $F = (0 \vee F(T)) \wedge (F_{top}(T)) = F(T)$.

Hence, secret sharing for $F$ can be performed with a share-size $SS(F) \leq SS(F_{slice}) + 2.\binom{n}{(n/2-\delta)}$. The term $\binom{n}{(n/2-\delta)}$ corresponds to the share size with a trivial sharing scheme for $F_{top}$ and $F_{bottom}$. Liu et. al. [14] further use a 4-step reduction to reduce the fat-slice function into a multi-party CDS protocol with sub-exponential communication complexity. Refer Section 3.2-3.5 of [14] for more details of the reduction steps.

Based on these reductions, for a general $\delta(n) = o(n/logn)$, and the slice $(n/2-\delta(n), n/2+\delta(n))$, they obtain $SS(F_{slice}) \leq 2^{O(\sqrt{n\delta(n)logn}+\sqrt{n}logn)}$. This combined with the $SS(F_{top})$ and $SS(F_{bottom})$ and minimized over the choice of $\delta$ gives $SS(F) \leq O(2^{0.994n})$. Note that, there is a trade-off involved with the choice of $\delta$ as the share size of $F_{slice}$ increases as $\delta$ increases whereas the share size of $F_{top}$ and $F_{bottom}$ increases as $\delta$ decreases.

## 7 Recent Advances

Since the breakthrough work by Liu et. al. [14] in 2018, various extensions and improvements have been proposed. [4] further improved the upper bound from $2^{0.994n}$ to $2^{0.892n}$. This was achieved by recursively performing general secret sharing on the slices $F_{top}$ and $F_{bottom}$ as opposed to using a trivial sharing scheme for these slices.

Applebaum et al. [5] further reduced the upper bound down to $2^{0.637n}$ for general SS (and $2^{0.762n}$ for linear SS). They define a new type of gates called *somewhat-regular* gates characterized by $(a, b)$ and block size $B$ such that $a \leq b \leq B$. While the work by Liu et al. [14] in realizing slice functions applied the constraint that each bucket in the partition $\Pi$ should have equal size requiring exponential number of gates (size of $L$ in Sec. 6.1). [5] modified this requirement by having each bucket in the partition have a hamming weight between $a$ and $b$, thus significantly reducing the number of gates. To realize this *somewhat-regular* gate, [5] proposes a variant of CDS called **robust CDS**. A CDS protocol is robust if it provides information-theoretic privacy even when it is invoked on a finite number of multiple inputs using the same randomness. Applebaum et al. provide a general transformation called "immunization" that takes a standard CDS and transforms it into a robust CDS.

Almost all the protocols that we saw in this report were for 1-bit secret. Applebaum et al. [3] in another work show that for an $m$-bit secret where $m$ is sufficiently large (double-exponent of input size $n$ i.e. $m \geq exp(exp(n))$), there exists a perfect linear CDS with a total communication complexity of $O(nm)$. They call it **amortization over long secrets**. Note that, this complexity is better than the complexity of having $m$ 1-bit CDS. This was further improved by [2], with a CDS protocol for an $m$-bit secret with a communication complexity of size $4m$.

Another recent work [16] studies **function-privacy** in the context of CDS. The basic idea is that the pre-decided condition ($f$) is never revealed to Charlie. They use a function secret-sharing scheme introduced earlier along with a threshold distributed point-functions to split the function such that at least a threshold number of shares are required to evaluate the function at any given input.

There are other interesting works closely related to CDS on topics like Private Information Retrieval (PIR), Private Simultaneous Messages (PSM). These works are not surveyed in this report.

## Conclusion

In this report, we first looked at the problem of *General Secret Sharing* on a monotone function. We looked at the decades old upper bound of $2^n$ on the share size. We then looked at an interesting primitive *Conditional Disclosure of Secrets* that enables multiple parties to reveal a secret to a third party iff their inputs satisfy a pre-defined predicate. We looked at various properties, examples and constructions of CDS including a recent advancement that led to sub-exponential CDS using PIR. We then looked at how these breakthroughs coupled with other advancements have helped "break" the barrier for general secret sharing from $2^n$ to $2^{0.637n}$. The huge gap between the lower bound and upper bound still remains an open research problem.

## References

[1] The 10th biu winter school on cryptography. http://cyber.biu.ac.il/event/the-10th-biu-winter-school-on-cryptography/, 2020. Accessed: 2022-04-18.

[2] Applebaum, B., and Arkis, B. On the power of amortization in secret sharing: $d$-uniform secret sharing and cds with constant information rate. Cryptology ePrint Archive, Report 2018/001, 2018. https://ia.cr/2018/001.

[3] Applebaum, B., Arkis, B., Raykov, P., and Vasudevan, P. N. Conditional disclosure of secrets: Amplification, closure, amortization, lower-bounds, and separations. In *Annual International Cryptology Conference* (2017), Springer, pp. 727–757.

[4] Applebaum, B., Beimel, A., Farràs, O., Nir, O., and Peter, N. Secret-sharing schemes for general and uniform access structures. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques* (2019), Springer, pp. 441–471.

[5] Applebaum, B., Beimel, A., Nir, O., and Peter, N. Better secret sharing via robust conditional disclosure of secrets. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing* (2020), pp. 280–293.

[6] Beimel, A. Secret-sharing schemes: A survey. In *International conference on coding and cryptology* (2011), Springer, pp. 11–46.

[7] Beimel, A., Kushilevitz, E., and Nissim, P. The complexity of multiparty psm protocols and related models. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques* (2018), Springer, pp. 287–318.

[8] Chor, B., Goldreich, O., Kushilevitz, E., and Sudan, M. Private information retrieval. In *Proceedings of IEEE 36th Annual Foundations of Computer Science* (1995), IEEE, pp. 41–50.

[9] Csirmaz, L. The size of a share must be large. *Journal of cryptology 10*, 4 (1997), 223–231.

[10] Gay, R., Kerenidis, I., and Wee, H. Communication complexity of conditional disclosure of secrets and attribute-based encryption. https://ia.cr/2015/665.

[11] Gertner, Y., Ishai, Y., Kushilevitz, E., and Malkin, T. Protecting data privacy in private information retrieval schemes. *Journal of Computer and System Sciences 60*, 3 (2000), 592–629.

[12] Ito, M., Saito, A., and Nishizeki, T. Secret sharing scheme realizing general access structure. *Electronics and Communications in Japan (Part III: Fundamental Electronic Science) 72*, 9 (1989), 56–64.

[13] Liu, T. Breaking barriers in secret sharing. https://dspace.mit.edu/handle/1721.1/124114, 2019.

[14] Liu, T., and Vaikuntanathan, V. Breaking the circuit-size barrier in secret sharing. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing* (New York, NY, USA, 2018), STOC 2018, Association for Computing Machinery, p. 699–708.

[15] Liu, T., Vaikuntanathan, V., and Wee, H. Conditional disclosure of secrets via non-linear reconstruction. In *Advances in Cryptology – CRYPTO 2017* (Cham, 2017), Springer International Publishing, pp. 758–790.

[16] Miranda, N., Yeo, F. Y., and Sehrawat, V. S. Function-private conditional disclosure of secrets and multi-evaluation threshold distributed point functions. In *International Conference on Cryptology and Network Security* (2021), Springer, pp. 334–354.

[17] Shamir, A. How to share a secret. *Communications of the ACM 22*, 11 (1979), 612–613.