Lecture 09: Shamir Secret Sharing (Lagrange Interpolation)
Recall: Goal

We want to

- Share a secret $s \in \mathbb{Z}_p$ to $n$ parties, such that $\{1, \ldots, n\} \subseteq \mathbb{Z}_p$,
- Any two parties can reconstruct the secret $s$, and
- No party alone can predict the secret $s$
Recall: Secret Sharing Algorithm

SecretShare(s, n)

- Pick a random line $\ell(X)$ that passes through the point $(0, s)$
  - This is done by picking $a_1$ uniformly at random from the set $\mathbb{Z}_p$
  - And defining the polynomial $\ell(X) = a_1X + s$
- Evaluate $s_1 = \ell(X = 1)$, $s_2 = \ell(X = 2)$, $\ldots$, $s_n = \ell(X = n)$
- Secret shares for party 1, party 2, $\ldots$, party $n$ are $s_1$, $s_2$, $\ldots$, $s_n$, respectively
Recall: Reconstruction Algorithm

SecretReconstruct($i_1, s^{(1)}, i_2, s^{(2)}$)

- Reconstruct the line $\ell'(X)$ that passes through the points $(i_1, s^{(1)})$ and $(i_2, s^{(2)})$
  - We will learn a new technique to perform this step, referred to as the Lagrange Interpolation
- Define the reconstructed secret $s' = \ell'(0)$
We want to

- Share a secret $s \in \mathbb{Z}_p$ to $n$ parties, such that $\{1, \ldots, n\} \subseteq \mathbb{Z}_p$,
- Any $t$ parties can reconstruct the secret $s$, and
- Less than $t$ parties cannot predict the secret $s$.

Shamir Secret Sharing
Shamir’s Secret Sharing Algorithm

SecretShare($s, n$)

- Pick a polynomial $p(X)$ of degree $\leq (t - 1)$ that passes through the point $(0, s)$
  - This is done by picking $a_1, \ldots, a_{t-1}$ independently and uniformly at random from the set $\mathbb{Z}_p$
  - And defining the polynomial
    \[ \ell(X) = a_{t-1}X^{t-1} + a_{t-2}X^{t-2} + \ldots + a_1X + s \]
- Evaluate $s_1 = p(X = 1)$, $s_2 = p(X = 2)$, \ldots, $s_n = p(X = n)$
- Secret shares for party 1, party 2, \ldots, party $n$ are $s_1$, $s_2$, \ldots, $s_n$, respectively
SecretReconstruct\((i_1, s^{(1)}, i_2, s^{(2)}, \ldots, i_t, s^{(t)})\)

- Use Lagrange Interpolation to construct a polynomial \(p'(X)\) that passes through \((i_1, s^{(1)}), \ldots, (i_t, s^{(t)})\) (we describe this algorithm in the following slides)
- Define the reconstructed secret \(s' = p'(0)\)
Consider the example we were considering in the previous lecture.

The secret was $s = 3$.

Secret shares of party 1, 2, 3, and 4, were 0, 2, 4, and 1, respectively.

Suppose party 2 and party 3 are trying to reconstruct the secret.

- Party 2 has secret share 2, and
- Party 3 has secret share 4.

We are interested in finding the line that passes through the points $(2, 2)$ and $(3, 4)$. 
Subproblem 1:

- Let us find the line that passes through \((2, 2)\) and \((3, 0)\)
  - Note that at \(X = 3\) this line evaluates to 0, so \(X = 3\) is a root of the line
  - So, the line has the equation \(\ell_1(X) = c \cdot (X - 3)\), where \(c\) is a suitable constant
  - Now, we find the value of \(c\) such that \(\ell_1(X)\) passes through the point \((2, 2)\)
  - So, we should have \(c \cdot (2 - 3) = 2\), i.e., \(c = 3\)
  - \(\ell_1(X) = 3 \cdot (X - 3)\) is the equation of that line
Subproblem 2:

- Let us find the line that passes through (2, 0) and (3, 4)
  - Note that at $X = 2$ this line evaluates to 0, so $X = 2$ is a root of the line
  - So, the line has the equality $\ell_2(X) = c \cdot (X - 2)$, where $c$ is a suitable constant
  - Now, we find the value of $c$ such that $\ell_2(X)$ passes through the point (3, 4)
  - So, we should have $c \cdot (3 - 2) = 4$, i.e. $c = 4$
  - $\ell_2(X) = 4 \cdot (X - 2)$
Putting Things Together:

- Define $\ell'(X) = \ell_1(X) + \ell_2(X)$
- That is, we have

$$\ell'(X) = 3 \cdot (X - 3) + 4 \cdot (X - 2)$$

- Evaluation of $\ell'(X)$ at $X = 0$ is

$$s' = \ell'(X = 0) = 3 \cdot (-3) + 4 \cdot (-2) = 3 \cdot 2 + 4 \cdot 3 = 1 + 2 = 3$$
We shall prove the following result

**Theorem**

There is a unique polynomial of degree at most $d$ that passes through $(x_1, y_1), (x_2, y_2), \ldots, (x_{d+1}, y_{d+1})$

- If possible, let there exist two distinct polynomials of degree $\leq d$ such that they pass through the points $(x_1, y_1), (x_2, y_2), \ldots, (x_{d+1}, y_{d+1})$
- Let the first polynomial be
  \[ p(X) = a_d X^d + a_{d-1} X^{d-1} + \cdots + a_1 X + a_0 \]
- Let the second polynomial be
  \[ p'(X) = a'_d X^d + a'_{d-1} X^{d-1} + \cdots + a'_1 X + a'_0 \]
Let \( p^*(X) \) be the polynomial that is the difference of the polynomials \( p(X) \) and \( p'(X) \), i.e.,

\[
p^*(X) = p(X) - p'(X) = (a_d - a'_d)X^d + \ldots (a_1 - a'_1)X + (a_0 - a'_0)
\]

**Observation.** The degree of \( p^*(X) \) is \( \leq d \)
For $i \in \{1, \ldots, d+1\}$, note that at $X = x_i$ both $p(X)$ and $p'(X)$ evaluate to $y_i$.

So, the polynomial $p^*(X)$ at $X = x_i$ evaluates to $y_i - y_i = 0$, i.e. $x_i$ is a root of the polynomial $p^*(X)$.

Observation. The polynomial $p^*(X)$ has roots $X = x_1$, $X = x_2$, $\ldots$, $X = x_{d+1}$.
We will use the following result

**Theorem (Schwartz–Zippel, Intuitive)**

*A non-zero polynomial of degree \(d\) has at most \(d\) roots (over any field)*

**Conclusion.**

- Based on the two observations above, we have a \(\leq d\) degree polynomial \(p^*(X)\) that has at least \((d + 1)\) distinct roots \(x_1, \ldots, x_{d+1}\).
- This implies, by Schwartz–Zippel Lemma, that the polynomial is the zero-polynomial.
- That is, \(p^*(X) = 0\).
- This implies that \(p(X)\) and \(p'(X)\) are identical.
- This contradicts the initial assumption that there are two distinct polynomials \(p(X)\) and \(p'(X)\).
The proof in the previous slides proves that

- Given a set of points \((x_1, y_1), \ldots, (x_{d+1}, y_{d+1})\)
- There is a unique polynomial of degree at most \(d\) that passes through all of them!
Suppose we are interested in constructing a polynomial of degree $\leq d$ that passes through the points $(x_1, y_1), \ldots, (x_{d+1}, y_{d+1})$.
Subproblem $i$:
- We want to construct a polynomial $p_i(X)$ of degree $\leq d$ that passes through $(x_i, y_i)$ and $(x_j, 0)$, where $j \neq i$.
- So, $\{x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_{d+1}\}$ are roots of the polynomial $p_i(X)$.
- Therefore, the polynomial $p_i(X)$ looks as follows:

$$p_i(X) = c \cdot (X - x_1) \cdots (X - x_{i-1})(X - x_{i+1}) \cdots (X - x_{d+1})$$

- Tersely, we will write this as:

$$p_i(X) = c \cdot \prod_{j \in \{1, \ldots, d+1\} \text{ such that } j \neq i} (X - x_j)$$
Now, to evaluate $c$ we will use the property that $p_i(x_i) = y_i$

Observe that the following value of $c$ suffices

$$c = \frac{y_i}{\prod_{j \in \{1,\ldots,d+1\} \text{ such that } j \neq i} (x_i - x_j)}$$

So, the polynomial $p_i(X)$ that passes through $(x_i, y_i)$ and $(x_j, 0)$, where $j \neq i$ is

$$p_i(X) = \frac{y_i}{\prod_{j \in \{1,\ldots,d+1\} \text{ such that } j \neq i} (x_i - x_j)} \cdot \prod_{j \in \{1,\ldots,d+1\} \text{ such that } j \neq i} (X - x_j)$$

Observe that $p_i(X)$ has degree $d$
Lagrange Interpolation IV

- **Putting Things Together:**
  - Consider the polynomial

\[ p(X) = p_1(X) + p_2(X) + \ldots + p_{d+1}(X) \]

- This is the desired polynomial that passes through \((x_i, y_i)\)

**Claim**

*The polynomial \(p(X)\) passes through \((x_i, y_i)\), for \(i \in \{1, \ldots, d + 1\}\)*

**Proof.**

- Note that, for \(j \in \{1, \ldots, d + 1\}\), we have

\[ p_j(x_i) = \begin{cases} y_i, & \text{if } j = i \\ 0, & \text{otherwise} \end{cases} \]

- Therefore, \(p(x_i) = \sum_{j=1}^{d+1} p_j(x_i) = y_i\)
Summary of Interpolation

- Given points \((x_1, y_1), \ldots, (x_{d+1}, y_{d+1})\)
- Lagrange Interpolation provides one polynomial of degree \(\leq d\) polynomial that passes through all of them
- Theorem 1 states that this \(\leq d\) degree polynomial is unique
Let us find a degree $\leq 2$ polynomial that passes through the points $(x_1, y_1), (x_2, y_2),$ and $(x_3, y_3)$

Subproblem 1:

- We want to find a degree $\leq 2$ polynomial that passes through the points $(x_1, y_1), (x_2, 0),$ and $(x_3, 0)$
- The polynomial is

$$p_1(X) = \frac{y_1}{(x_1 - x_2)(x_1 - x_3)}(X - x_2)(X - x_3)$$
Subproblem 2:
- We want to find a degree \( \leq 2 \) polynomial that passes through the points \((x_1, 0), (x_2, y_2), \) and \((x_3, 0)\).
- The polynomial is

\[
p_2(X) = \frac{y_2}{(x_2 - x_1)(x_2 - x_3)}(X - x_1)(X - x_3)
\]

Subproblem 3:
- We want to find a degree \( \leq 2 \) polynomial that passes through the points \((x_1, 0), (x_2, 0), \) and \((x_3, y_3)\).
- The polynomial is

\[
p_2(X) = \frac{y_3}{(x_3 - x_1)(x_3 - x_2)}(X - x_1)(X - x_2)
\]
Putting Things Together: The reconstructed polynomial is

\[ p(X) = p_1(X) + p_2(X) + p_3(X) \]
This completes the description of Shamir’s Secret Sharing algorithm. In the following lectures we will argue its security.