NIZK from Bilinear Maps

BIU Winter School 2019

Jens Groth, DFINITY

Bilinear maps

Background

Bilinear maps

Setup describing $(p, G_1, G_2, G_T, e, g, h)$

- Prime p
  - Size of prime related to security level, could for instance choose $|p| \approx 256$
- Cyclic groups G_1, G_2, G_T of order p
  - Written multiplicatively with neutral elements 1 in this talk
  - Generators g, h such that $G_1 = \langle g \rangle, G_2 = \langle h \rangle$
- Map $e: G_1 \times G_2 \to G_T$
  - Non-degenerate: $e(g, h) \neq 1$
  - Bilinear: For all $a, b \in \boldsymbol{Z}_p: e(g^a, h^b) = e(g, h)^{ab}$

Generic bilinear group operations

- Canonical representation of group elements
  - So easy to determine whether $u = v$

- Efficient algorithms to
  - Decide membership in the three groups, e.g., $u \in G_1$
  - Compute group operations in the three groups, e.g., $u \cdot v$ in G_2
  - Evaluate the bilinear map, e.g., $e(u, v)$

- We refer to these as the *generic* group operations

Types of bilinear maps

- In pairing-based cryptography, usually the source groups G_1 (G_2) are subgroups of elliptic curves over a finite field $\boldsymbol{F}_q$ $(\boldsymbol{F}_{q^e})$, the target group G_T a multiplicative subgroup of $\boldsymbol{F}^*_{q^k}$, and the bilinear map a pairing $e: G_1 \times G_2 \rightarrow G_T$
- The underlying mathematical details of the groups and the bilinear map will not be important for these lectures, but it is worth noting the classification of Galbraith, Paterson and Smart [GPS04]
  - Type I: Symmetric setting where $G_1 = G_2$
  - Type II: Asymmetric setting $G_1 \neq G_2$ with an efficiently computable isomorphism $\psi: G_2 \rightarrow G_1$
  - Type III: Asymmetric setting $G_1 \neq G_2$ where there is no efficiently computable isomorphism in either direction

Efficiency

- Type III pairings are currently the most efficient
  - So unless otherwise specified we work in the type III setting
- Size of group element representations
  - For $a \in \boldsymbol{Z}_p, u \in G_1, v \in G_2, w \in G_T$ expect $|a| < |u| < |v| < |w|$
- Cost of operations
  - Multiplications in G_1 cheaper than multiplications in G_2 cheaper than multiplications in G_T
  - Exponentiations in G_1 cheaper than exponentiations in G_2 cheaper than exponentiations in G_T
  - Bilinear map the most expensive

Getting used to bilinear maps

- Recall $e: G_1 \times G_2 \rightarrow G_T$
  - Non-degenerate: $e(g, h) \neq 1$
  - Bilinear: For all $a, b \in \boldsymbol{Z}_p$: $e\left(g^a, h^b\right) = e(g, h)^{ab}$
- Exercises
  - What does the equation $e(u, v)e(u, w) = y^a z$ implicitly assume about which groups u, v, w, y, z, a belong to?
  - If you see the equation $e(u, u) = z$ are you in a type I, II or III setting?
  - Reduce $e(g^a, h)e\left(g^b, h\right)$, $e(g, h^a)e\left(g^b, h\right)$, $e\left(g^a, h^{-b}\right)e(u, v)e(f, h)^c$, $\prod_{i=1}^{n} e(g, h^{a_i})^{b_i}$
  - Reduce $e(u, v)e(u, w)$, $e(u, v^a)e(u^b, v)$, $e\left(g^a, v^{-b}\right)e(f, w)e(u, v)^c$, $\prod_{i=1}^{n} e\left(u^a, v_i^b\right)^{\frac{c_i}{ab}}$
  - Show that if $e(u, v) = 1$ then $u = 1$ or $v = 1$

Answers

- What does the equation $e(u,v)e(u,w) = y^a z$ implicitly assume about which groups u, v, w, y, z, a belong to?

  $u \in G_1, v, w \in G_2, y, z \in G_T, a \in \boldsymbol{Z}_p$

- If you see the equation $e(u,u) = z$ are you in a type I, II or III setting?

  Type I because $u \in G_1, u \in G_2$ indicates $G_1 = G_2$

- Reduce $e(g^a,h)e(g^b,h)$, $e(g,h^a)e(g^b,h)$, $e(g^a,h^{-b})e(u,v)e(g,h)^c$, $\prod_{i=1}^n e(g,h^{a_i})^{b_i}$

  $$e(g^a,h)e(g^b,h) = e(g,h)^a e(g,h)^b = e(g,h)^{a+b}$$
  $$e(g,h^a)e(g^b,h) = e(g,h)^a e(g,h)^b = e(g,h)^{a+b}$$
  $$e(g^a,h^{-b})e(u,v)e(g,h)^c = e(g,h)^{-ab} e(g,h)^c e(u,v) = e(g,h)^{c-ab} e(u,v)$$
  $$\prod_{i=1}^n e(g,h^{a_i})^{b_i} = \prod_{i=1}^n e(g,h)^{a_i b_i} = e(g,h)^{\sum_{i=1}^n a_i b_i}$$

- Interesting follow-up question, is $e(g^{a+b},h)$ or $e(g,h^{a+b})$ or $e(g,h)^{a+b}$ more "reduced"?

  - Recall cost hierarchy expo in $G_1 \leq$ expo in $G_2 \leq$ expo in $G_T \leq$ pairing
  - So maybe $e(g^{a+b},h)$ cheaper to compute at cost of 1 expo in G_1 and 1 pairing
  - However, if $e(g,h)$ used often, precompute to get $e(g,h)^{a+b}$ at amortized cost of 1 expo in G_T

Answers

- Reduce $e(u,v)e(u,w)$, $e(u,v^a)e(u^b,v)$, $e(g^a,v^{-b})e(f,w)e(u,v)^c$, $\prod_{i=1}^{n} e(u^a,v_i^b)^{\frac{c_i}{ab}}$

Because g generates G_1 we can write any $u \in G_1$ as $u = g^x$
Similarly, we can write any $v,w \in G_2$ as $v = h^y$ and $w = h^z$
- All we know is such $x,y,z \in \boldsymbol{Z}_p$ exist, we may not know what they are

$$e(u,v)e(u,w) = e(g^x,h^y)e(g^x,h^z) = e(g,h)^{x(y+z)} = e(u,vw)$$
$$e(u,v^a)e(u^b,v) = e(u,v)^a e(u,v)^b = e(u,v)^{a+b}$$
$$e(g^a,v^{-b})e(f,w)e(u,v)^c = e(g,v)^{-ab}e(g^x,v)^c e(f,w) = e(g^{-ab}u^c,v)e(f,w)$$
$$\prod_{i=1}^{n} e(u^a,v_i^b)^{\frac{c_i}{ab}} = \prod_{i=1}^{n} e(u,v_i)^{ab\cdot\frac{c_i}{ab}} = \prod_{i=1}^{n} e(u,v_i)^{c_i} = e\left(u,\prod_{i=1}^{n} v_i^{c_i}\right)$$

- Show that if $e(u,v) = 1$ then $u = 1$ or $v = 1$

$e(u,v) = e(g^x,h^y) = e(g,h)^{xy}$ is the same as $1 = e(g,h)^0$
Since $e(g,h) \neq 1$ it generates G_T so we have $xy = 0$ implying $x = 0$ or $y = 0$

Decisional Diffie-Hellman assumption

- We will assume the DDH problem is hard in both G_1 and G_2
  - Also known as the Symmetric External DH (SXDH) assumption
- The DDH assumption in G_1 over setup $(p, G_1, G_2, G_T, e, g, h)$
  - Define for adversary A the following experiment

$$b \leftarrow \{0,1\}$$
$$x, y, z \leftarrow \boldsymbol{Z}_p^*$$
$$u = g^x, v = g^y$$
$$w = g^{bxy+(1-b)z}$$
$$b^* \leftarrow A(p, G_1, G_2, G_T, e, g, h, u, v, w)$$

  - The assumption says that for any realistic (computationally bounded) adversary $\Pr[b = b^*] \approx \frac{1}{2}$
- The DDH assumption in G_2 over setup $(p, G_1, G_2, G_T, e, g, h)$ is defined similarly

ElGamal encryption

- Key generation in group G_1 assuming setup $(p, G_1, G_2, G_T, e, g, h)$
  - Pick $x \leftarrow Z_p$ and let this be the secret key. Let the public key be $y = g^x$
- Encryption of $m \in G_1$
  - Pick $r \leftarrow Z_p$ and return ciphertext $c = \mathrm{Enc}(y, m; r) := (g^r, y^r m)$
- Decryption of $c = (u, v) \in G_1^2$
  - Return plaintext $m = \mathrm{Dec}(x, u, v) := v u^{-x}$
- IND-CPA secure under DDH assumption in G_1
- ElGamal encryption in G_2 similar

Pairing-based proofs

Statements we want to prove

Groth-Sahai proofs

- Two computationally indistinguishable types of common reference string
  - Binding common reference string $\qquad g, u, g', u' \in G_1, h, v, h', v' \in G_2$
    - Perfect completeness
    - Perfect soundness

$$\Updownarrow ? \quad \text{DDH}$$

  - Hiding common reference string $\qquad g, u, g', u' \in G_1, h, v, h', v' \in G_2$
    - Perfect completeness
    - Perfect zero-knowledge

Statements

- Instance $\phi = \{eq_1, \dots, eq_q\}$, equations over variables $X_i \in G_1, Y_j \in G_2, x_i, y_j \in \boldsymbol{Z}_p$
  - Pairing product equation defined by $A_j \in G_1, B_i \in G_2, \gamma_{ij} \in \boldsymbol{Z}_p$

$$\prod_{j\in[n]} e(A_j, Y_j) \cdot \prod_{i\in[m]} e(X_i, B_i) \cdot \prod_{i\in[m]} \prod_{j\in[n]} e(X_i, Y_j)^{\gamma_{ij}} = 1$$

  - Multi-exponentiation equation in G_1 defined by $A_j, T \in G_1, b_i, \gamma_{ij} \in \boldsymbol{Z}_p$ (analogous for G_2)

$$\prod_{j\in[n']} A_j^{y_j} \cdot \prod_{i\in[m]} X_i^{b_i} \cdot \prod_{i\in[m]} \prod_{j\in[n']} X_i^{\gamma_{ij} y_j} = T$$

  - Quadratic equations defined by $a_j, b_i, \gamma_{ij}, t \in \boldsymbol{Z}_p$

$$\sum_{j\in[n']} a_j y_j + \sum_{i\in[m']} x_i b_i + \sum_{i\in[m']} \sum_{j\in[n']} x_i \gamma_{ij} y_j = t$$

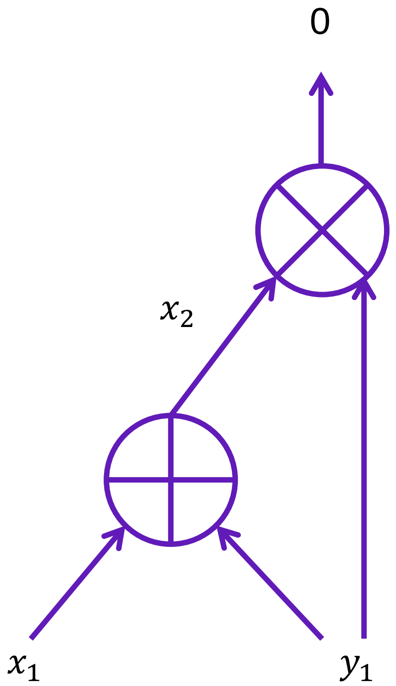
- Witness $X_1, \dots, X_m \in G_1, Y_1, \dots, Y_n \in G_2, x_1, \dots, x_{m'}, y_1, \dots, y_{n'} \in \boldsymbol{Z}_p$ satisfying all eq_k

NP completeness

- SAT formula ϕ: $(x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_3 \vee x_4 \vee x_5) \wedge \cdots$
- Witness $x_1 = \text{true}, x_2 = \text{false}, \ldots$

- Can rewrite ϕ as a set of quadratic equations
  - Encode true as 1 and false as 0 in $\boldsymbol{Z}_p$
  - For each variable x_i have the quadratic equations $x_i \cdot 1 + 1 \cdot y_i = 0$ and $x_i \cdot 1 + x_i \cdot y_i = 0$
    The first equation gives us $y_i = -x_i$
    The second equation gives us $x_i \cdot (1 - x_i) = 0$ so $x_i \in \{0,1\}$, i.e., it encodes true or false
  - Translate each clause into a quadratic equation that involves an extra variable y'
    Example $(x_1 + (1 - x_2) + x_3) \cdot y_1' = 1$, $\left((1 - x_3) + x_4 + x_5\right) \cdot y_2' = 1$, …
    Such inverses y_1', y_2', ... exist in $\boldsymbol{Z}_p$ if and only if the clauses are satisfied

Arithmetic circuit



- Arithmetic circuit over $\boldsymbol{Z}_p$
- Instance describes circuit wiring, gates and some of the inputs and outputs
- Witness is values on the wires that satisfy all gates

- Can reduce an arithmetic circuit to quadratic equations

$$x_1 \cdot 1 + x_2 \cdot (-1) + 1 \cdot y_1 = 0$$

$$x_2 \cdot y_1 = 0$$

Practical cryptography

- When constructing cryptographic protocols more likely to encounter statement like "This is a ciphertext encrypting a signature on m"
  - Suppose we have an ElGamal ciphertext $(u, v) \in G_1$ under public key $y \in G_1$
  - Suppose the claim is it encrypts a weak Boneh-Boyen signature $m \in \boldsymbol{Z}_p$ of the form $\sigma = g^{\frac{1}{x+m}}$, which satisfies the verification equation $e(\sigma, wh^m) = e(g, h)$ where the public key is $w = h^x$
  - Instance defined by setup $(p, G_1, G_2, G_T, e, g, h)$ and $u, v, y \in G_1, w \in G_2, m \in \boldsymbol{Z}_p$
    Witness is randomness $r \in \boldsymbol{Z}_p$ used in encryption and secret signature $\sigma \in G_1$
- Exercise
  - Rewrite statement as a set of pairing-product, multi-exponentiation and quadratic equations

A solution

- Equations over variables $\sigma, f \in G_1, r \in \boldsymbol{Z}_p$
  - Pairing-product equation defined by $wh^m, h \in G_2$
    $$e(\sigma, wh^m)e(f, h) = 1$$
  - Multi-exponentiation equations
    $$f^1 = g^{-1}$$
    $$g^r = u$$
    $$y^r\sigma = v$$
- When all equations satisfied, then indeed (u, v) is an ElGamal ciphertext encrypting a weak Boneh-Boyen signature σ on $m \in \boldsymbol{Z}_p$ satisfying the verification equation $e(\sigma, wh^m) = e(g, h)$

Why not $e(g, wh^m) = e(g, h)$?
Because Groth-Sahai proofs only guarantee zero-knowledge when the target element is 1
(Can be generalized to ZK for this equation though [G-Escala 2013])

Writing the top equation in full, it is
$$1^r \cdot \sigma^0 f^1 \cdot \sigma^{0r} f^{0r} = g^{-1}$$
where with the previous notation
$$A_1 = 1, b_1 = 0, b_2 = 1$$
$$\gamma_{11} = 0, \gamma_{12} = 0, T = g^{-1}$$

A warm-up proof system

Perfect soundness, but modest privacy

Extended bilinear map

- We define an extended map $E: G_1^2 \times G_2^2 \to G_T^4$ by

$$E\left(\begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, (d_1, d_2)\right) = \begin{pmatrix} e(c_1, d_1) & e(c_1, d_2) \\ e(c_2, d_1) & e(c_2, d_2) \end{pmatrix}$$

- Exercise
  - Show the map is bilinear on the left hand side, i.e.,

$$E\left(\begin{pmatrix} a_1 \\ a_2 \end{pmatrix}\begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, (d_1, d_2)\right) = E\left(\begin{pmatrix} a_1 \\ a_2 \end{pmatrix}, (d_1, d_2)\right) E\left(\begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, (d_1, d_2)\right)$$

    using entry-wise product for the vectors and matrices

  - And the same for the right hand side

Extended bilinear map

- We define an extended map $E: G_1^2 \times G_2^2 \to G_T^4$ by

$$E\left(\begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, (d_1, d_2)\right) = \begin{pmatrix} e(c_1, d_1) & e(c_1, d_2) \\ e(c_2, d_1) & e(c_2, d_2) \end{pmatrix}$$

- Exercise solution
  - Show the map is bilinear on the left hand side, i.e.,

$$E\left(\begin{pmatrix} a_1 \\ a_2 \end{pmatrix}\begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, (d_1, d_2)\right) = \begin{pmatrix} e(a_1 b_1, d_1) & e(a_1 b_1, d_2) \\ e(a_2 b_2, d_1) & e(a_2 b_2, d_2) \end{pmatrix} = \begin{pmatrix} e(a_1, d_1)e(b_1, d_1) & e(a_1, d_2)e(b_1, d_2) \\ e(a_2, d_1)e(b_2, d_2) & e(a_2, d_2)e(b_2, d_2) \end{pmatrix}$$

$$= \begin{pmatrix} e(a_1, d_1) & e(a_1, d_2) \\ e(a_2, d_1) & e(a_2, d_2) \end{pmatrix}\begin{pmatrix} e(b_1, d_1) & e(b_1, d_2) \\ e(b_2, d_1) & e(b_2, d_2) \end{pmatrix} = E\left(\begin{pmatrix} a_1 \\ a_2 \end{pmatrix}, (d_1, d_2)\right) E\left(\begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, (d_1, d_2)\right)$$

using entry-wise product for the vectors and matrices

Warm-up proof system

- Common reference string consists of setup and random $u \in G_1, v \in G_2$
- Suppose we have an instance with a single pairing-product equation

$$e(X, Y) = T$$

- The prover encrypts X as $(c_1, c_2) = (g^r, u^r X)$ and Y as $(d_1, d_2) = (h^s, v^s Y)$
- Let us apply the extended bilinear product to the ciphertexts

$$E\left(\begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, (d_1, d_2)\right) = E\left(\begin{pmatrix} g \\ u \end{pmatrix}^r \begin{pmatrix} 1 \\ X \end{pmatrix}, (d_1, d_2)\right)$$

$$= E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (d_1, d_2)^r\right) E\left(\begin{pmatrix} 1 \\ X \end{pmatrix}, (d_1, d_2)\right)$$

Warm-up proof system

$$= E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (d_1, d_2)^r\right) E\left(\begin{pmatrix} 1 \\ X \end{pmatrix}, (h, v)^s (1, Y)\right)$$

$$= E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (d_1, d_2)^r\right) E\left(\begin{pmatrix} 1 \\ X \end{pmatrix}^s, (h, v)\right) E\left(\begin{pmatrix} 1 \\ X \end{pmatrix}, (1, Y)\right)$$

$$= E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (d_1, d_2)^r (h, v)^t\right) E\left(\begin{pmatrix} 1 \\ X \end{pmatrix}^s \begin{pmatrix} g \\ u \end{pmatrix}^{-t}, (h, v)\right) \begin{pmatrix} 1 & 1 \\ 1 & e(X, Y) \end{pmatrix}$$

using random $t \leftarrow \boldsymbol{Z}_p$

- The prover sets $(\pi_1, \pi_2) = (d_1^r h^t, d_2^r v^t)$ and $(\theta_1, \theta_2) = \left(g^{-t}, X u^{-t}\right)$ and returns the full proof $(c_1, c_2, d_1, d_2, \pi_1, \pi_2, \theta_1, \theta_2)$

Verification

- The verifier given the proof $(c_1, c_2, d_1, d_2, \pi_1, \pi_2, \theta_1, \theta_2)$ for $e(X, Y) = T$ accepts if and only if

$$E\left(\begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, (d_1, d_2)\right) = E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (\pi_1, \pi_2)\right) E\left(\begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}, (h, v)\right) \begin{pmatrix} 1 & 1 \\ 1 & T \end{pmatrix}$$

- Perfect completeness when $e(X, Y) = T$ follows from the calculations
- Exercise
  - Show that the proof system gives a proof of knowledge of X, Y such that $e(X, Y) = T$
  - Hint: suppose you know the knowledge extraction keys a, b such that $u = g^a, v = h^b$. Now decrypt the columns with a and the rows with b

Knowledge soundness

- Solution
  - Let us define the knowledge extractor to return $X = c_1^{-a} c_2$ and $Y = d_1^{-b} d_2$
  - Recall that by definition

$$E\left(\begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, (d_1, d_2)\right) = \begin{pmatrix} e(c_1, d_1) & e(c_1, d_2) \\ e(c_2, d_1) & e(c_2, d_2) \end{pmatrix}$$

  - Decrypting the columns with $a \in \boldsymbol{Z}_p$ gives us

$$\left(e(c_1, d_1)^{-a} e(c_2, d_1)\,, e(c_1, d_2)^{-a} e(c_2, d_2)\right) = \left(e(c_1^{-a} c_2, d_1), e(c_1^{-a} c_2, d_2)\right)$$

  - Decrypting the row with $b \in Z_p$ gives us

$$e(c_1^{-a} c_2, d_1)^{-b} e(c_1^{-a} c_2, d_2) = e(c_1^{-a} c_2, d_1^{-b} d_2)$$

  - So vertical and horizontal decryption gives us $e(X, Y)$

Analyzing the verification equation

- The verification equation is

$$E\left(\begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, (d_1, d_2)\right) = E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (\pi_1, \pi_2)\right) E\left(\begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}, (h, v)\right) \begin{pmatrix} 1 & 1 \\ 1 & T \end{pmatrix}$$

- We just saw the left hand side decrypts to $e(X, Y)$
- The matrix $E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (\pi_1, \pi_2)\right)$ decrypts to $e\left(g^{-a}u, \pi_1^{-b}\pi_2\right) = e\left(1, \pi_1^{-b}\pi_2\right) = 1$
- The matrix $E\left(\begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}, (h, v)\right)$ decrypts to $e\left(\theta_1^{-a}\theta_2, h^{-b}v\right) = e(\theta_1^{-a}\theta_2, 1) = 1$
- And the matrix $\begin{pmatrix} 1 & 1 \\ 1 & T \end{pmatrix}$ decrypts to T so we get $e(X, Y) = 1 \cdot 1 \cdot T$

Generalizing to more complex equation

- For a pairing-product equation defined by $A_j \in G_1, B_i \in G_2, \gamma_{ij} \in \boldsymbol{Z}_p, T \in G_T$

$$\prod_{j\in[n]} e(A_j, Y_j) \cdot \prod_{i\in[m]} e(X_i, B_i) \cdot \prod_{i\in[m]} \prod_{j\in[n]} e\left(X_i, Y_j\right)^{\gamma_{ij}} = T$$

- The prover ElGamal encrypts each variable

$$\left(c_{1,i}, c_{2,i}\right) = (g^{r_i}, u^{r_i} X) \qquad \left(d_{j,1}, d_{j,2}\right) = (h^{s_j}, v^{s_j} Y_j)$$

- The prover computes

$$(\pi_1, \pi_2) = \prod_{i\in[m]} (1, B_i)^{r_i} \cdot \prod_{i\in[m]} \prod_{j\in[n]} \left(d_{j,1}, d_{j,2}\right)^{\gamma_{ij} r_i} \cdot (h, v)^{-t}$$

$$(\theta_1, \theta_2) = \prod_{j\in[n]} \left(1, A_j\right)^{s_j} \cdot \prod_{i\in[m]} \prod_{j\in[n]} (1, X_i)^{\gamma_{ij}} \cdot (g, u)^t$$

Generalizing to more complex equation

- The verifier accepts the proof if and only if

$$\prod_{j\in[n]} E\left(\begin{pmatrix}1\\A_j\end{pmatrix},(d_{j,1},d_{j,2})\right)\cdot\prod_{i\in[m]} E\left(\begin{pmatrix}c_{1,i}\\c_{2,i}\end{pmatrix},(1,B_j\,)\right)\cdot\prod_{i\in[m]}\prod_{j\in[n]} E\left(\begin{pmatrix}c_{1,i}\\c_{2,i}\end{pmatrix},(d_{j,1},d_{j,2})\right)^{\gamma_{ij}}$$

$$= E\left(\begin{pmatrix}g\\u\end{pmatrix},(\,\pi_1,\pi_2)\right)\cdot E\left(\begin{pmatrix}\theta_1\\\theta_2\end{pmatrix},(\,h,v)\right)\cdot\begin{pmatrix}1&1\\1&T\end{pmatrix}$$

- Perfect completeness
  - Many calculations, home exercise
- Perfect soundness
  - Proof of knowledge, as before by decrypting on both dimensions

Multi-exponentiation equations

- Multi-exponentiation equation in G_1 defined by $A_j, T \in G_1, b_i, \gamma_{ij} \in \boldsymbol{Z}_p$

$$\prod_{j\in[n']} A_j^{y_j} \cdot \prod_{i\in[m]} X_i^{b_i} \cdot \prod_{i\in[m]} \prod_{j\in[n']} X_i^{\gamma_{ij} y_j} = T$$

-

  Can be mapped to pairing product equation by instead proving

$$\prod_{j\in[n']} e\left(A_j, h^{y_j}\right) \cdot \prod_{i\in[m]} e\left(X_i, h^{b_i}\right) \cdot \prod_{i\in[m]} \prod_{j\in[n']} e(X_i, h^{y_j})^{\gamma_{ij}} = e(T, h)$$

- Multi-exponentiation equation in G_2 similar

Quadratic equations

- Quadratic equation defined by $a_j, b_i, \gamma_{ij}, t \in \boldsymbol{Z}_p$

$$\sum_{j \in [n']} a_j y_j + \sum_{i \in [m']} x_i b_i + \sum_{i \in [m']} \sum_{j \in [n']} x_i \gamma_{ij} y_j = t$$

- Can be mapped to pairing product equation by instead proving

$$\prod_{j \in [n']} e(g^{a_j}, h^{y_j}) \cdot \prod_{i \in [m]} e\left(g^{x_i}, h^{b_i}\right) \cdot \prod_{i \in [m]} \prod_{j \in [n']} e(g^{x_i}, h^{y_j})^{\gamma_{ij}} = e(g, h)^t$$

Multiple equations

- Instance $\phi = \{eq_1, \dots, eq_q\}$, equations over variables $X_i \in G_1, Y_j \in G_2, x_i, y_j \in \boldsymbol{Z}_p$
- Witness $X_1, \dots, X_m \in G_1, Y_1, \dots, Y_n \in G_2, x_1, \dots, x_{m'}, y_1, \dots, y_{n'} \in \boldsymbol{Z}_p$ satisfying all eq_k

- The prover encrypts all variables in the witness as

$$(c_{1,i}, c_{2,i}) = (g^{r_i}, u^{r_i} X_i) \qquad (d_{j,1}, d_{j,2}) = (h^{s_j}, v^{s_j} Y_j)$$

$$(c'_{1,i}, c'_{2,i}) = \left(g^{r'_i}, u^{r'_i} g^{x_i}\right) \qquad (d'_{j,1}, d'_{j,2}) = \left(h^{s'_j}, v^{s'_j} Y_j\right)$$

- For each equation eq_k the prover generates proof elements $\pi_{k,1}, \pi_{k,2}, \theta_{k,1}, \theta_{k,2}$
- The full proof for all equations being simultaneously satisfiable is $(c_{1,1}, \dots, \theta_{q,2})$

- The verifier checks verification equations for $k = 1, \dots, q$

  - Note the verification equations reuse the commitments $(c_{1,1}, c_{2,1}, \dots, d'_{n',1}, d'_{n',2})$ to variables but each equation has a separate quadruple $(\pi_{k,1}, \pi_{k,2}, \theta_{k,1}, \theta_{k,2})$

Security

- Perfect completeness
- Perfect soundness
  - Each commitment decrypts to unique X_i, Y_j or g^{x_i}, h^{y_j}
  - Decrypting the verification equations horizontally and vertically shows each equation satisfied
- Privacy?
  - Witness-indistinguishable in the generic group model where attacker can only do generic group operations [Deshpande-G-Smeets]
  - Provably not zero-knowledge in the generic group model [Deshpande-G-Smeets]
- But we want zero-knowledge under standard assumptions (DDH)!

Groth-Sahai proofs

Soundness and witness-indistinguishability/zero-knowledge

Commitments

- Let us extend the setup to include $g, u, g', u' \in G_1, h, v, h', v' \in G_2$
- Now the prover will make commitments to $X \in G_1$ and $Y \in G_2$ of the form

$$\left(g^r (g')^{r'}, u^r (u')^{r'} X\right) \text{ and } \left(h^s (h')^{s'}, v^s (v')^{s'} Y\right)$$

- More precisely, for $X \in G_1$ the prover picks random $r, r' \leftarrow \mathbf{Z}_p$ and computes a commitment as $(c_1, c_2) = (g, u)^r (g', u')^{r'} (1, X)$
- The core observation to make is that we can now have two setups
  - Binding setup $(g', u') = (g^\alpha, u^\alpha)$
  - Hiding setup $(g', u') = \left(g^\alpha, u^\alpha g^{-1}\right)$

?

Indistinguishable under DDH

- Exercise: Show commitments are perfectly binding and hiding, respectively

Commitments

- Let us extend the setup to include $g, u, g', u' \in G_1, h, v, h', v' \in G_2$
- For $X \in G_1$ the prover picks random $r, r' \leftarrow \boldsymbol{Z}_p$ and computes a commitment as $(c_1, c_2) = (g, u)^r (g', u')^{r'} (1, X)$
- We now have two computationally indistinguishable setups
  - Binding setup $(g', u') = (g^\alpha, u^\alpha)$
  - Hiding setup $(g', u') = \left(g^\alpha, u^\alpha g^{-1}\right)$
- Exercise solution

  - In the binding setup $(c_1, c_2) = \left(g^{r+\alpha r'}, u^{r+\alpha r'} X\right)$ embeds unique X

  - In the hiding setup $(c_1, c_2) = \left(g^{r+\alpha r'}, u^{r+\alpha r'} \left(g^{-r'} X\right)\right)$ is random for all X

Proof example

- Common reference string with $g, u, g', u' \in G_1, h, v, h', v' \in G_2$
- Suppose we have an instance with a single pairing-product equation

$$e(X, Y) = T$$

- Prover commits to X and Y as

$$(c_1, c_2) = (g^r (g')^{r'}, u^r (u')^{r'} X) \quad \text{and} \quad (d_1, d_2) = \left(h^s (h')^{s'}, v^s (v')^{s'} Y\right)$$

- Let us apply the extended bilinear map to the commitments

$$E\left(\begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, (d_1, d_2)\right) = E\left(\begin{pmatrix} g \\ u \end{pmatrix}^r \begin{pmatrix} g' \\ u' \end{pmatrix}^{r'} \begin{pmatrix} 1 \\ X \end{pmatrix}, (d_1, d_2)\right)$$

$$= E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (d_1, d_2)^r\right) E\left(\begin{pmatrix} g' \\ u' \end{pmatrix}, (d_1, d_2)^{r'}\right) E\left(\begin{pmatrix} 1 \\ X \end{pmatrix}, (d_1, d_2)\right)$$

Proof example

$$= E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (d_1, d_2)^r\right) E\left(\begin{pmatrix} g' \\ u' \end{pmatrix}, (d_1, d_2)^{r'}\right) E\left(\begin{pmatrix} 1 \\ X \end{pmatrix}, (h, v)^s (h', v')^{s'} (1, Y)\right)$$

$$= E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (d_1, d_2)^r\right) E\left(\begin{pmatrix} g' \\ u' \end{pmatrix}, (d_1, d_2)^{r'}\right) E\left(\begin{pmatrix} 1 \\ X \end{pmatrix}^s, (h, v)\right) E\left(\begin{pmatrix} 1 \\ X \end{pmatrix}^{s'}, (h', v')\right) E\left(\begin{pmatrix} 1 \\ X \end{pmatrix}, (1, Y)\right)$$

$$= E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (\pi_1, \pi_2)\right) E\left(\begin{pmatrix} g' \\ u' \end{pmatrix}, (\pi_1', \pi_2')\right) E\left(\begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}, (h, v)\right) E\left(\begin{pmatrix} \theta_1' \\ \theta_2' \end{pmatrix}, (h', v')\right) \begin{pmatrix} 1 & 1 \\ 1 & e(X, Y) \end{pmatrix}$$

- The proof elements are then randomized using $t, t', t'', t''' \leftarrow \boldsymbol{Z}_p$

$$(\pi_1, \pi_2) \mapsto (\pi_1, \pi_2)(h, v)^t (h', v')^{t'} \qquad (\theta_1, \theta_2) \mapsto (\theta_1, \theta_2)(g, u)^{-t} (g', u')^{-t''}$$

$$(\pi_1', \pi_2') \mapsto (\pi_1', \pi_2')(h, v)^{t''} (h', v')^{t'''} \qquad (\theta_1', \theta_2') \mapsto (\theta_1', \theta_2')(g, u)^{-t'} (g', u')^{-t'''}$$

Security

- The verifier given the proof $(c_1, c_2, d_1, d_2, \pi_1, \pi_2, \pi_1', \pi_2', \theta_1, \theta_2, \theta_1', \theta_2')$ for $e(X, Y) = T$ accepts if and only if

$$E\left(\begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, (d_1, d_2)\right) = E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (\pi_1, \pi_2)\right) E\left(\begin{pmatrix} g' \\ u' \end{pmatrix}, (\pi_1', \pi_2')\right) E\left(\begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}, (h, v)\right) E\left(\begin{pmatrix} \theta_1' \\ \theta_2' \end{pmatrix}, (h', v')\right) \begin{pmatrix} 1 & 1 \\ 1 & T \end{pmatrix}$$

- Perfect completeness when $e(X, Y) = T$ follows from the calculations

- On a binding setup, where $(g', u') = (g, u)^\alpha$ and $(h', v') = (h, v)^\beta$, decryption vertically and horizontally shows the proof system is perfectly sound

Privacy

- On a hiding setup, where $(g', u') = (g^\alpha, u^\alpha g^{-1})$ and $(h', v') = \left(h^\beta, v^\beta h^{-1}\right)$, the proof system is perfectly witness indistinguishable
  - Commitments $(c_1, c_2), (d_1, d_2)$ are uniformly random
  - Proof elements $\pi_1, \pi_2, \pi_1', \pi_2'$ are uniformly random due to the rerandomization
  - Conditioned on these the verification equation uniquely determines $\theta_1, \theta_2, \theta_1', \theta_2'$
  - So impossible to tell whether the prover used a witness (X, Y) such that $e(X, Y) = T$ or used another witness (X', Y') also satisfying $e(X', Y') = T$
- What about zero-knowledge? Given T can we simulate a proof?
  - Hard in general, given arbitrary T it is infeasible to find solution to $\prod_{i=1}^{n} e(A_i, B_i) = T$ so the simulator cannot satisfy the verification equation
  - But if $T = 1$ the problem is easy, just pick $X = 1, Y = 1$ and we have $e(X, Y) = T$
    And because the proof is witness indistinguishable, this witness is as good as any other

Statements – witness indistinguishability

- Instance $\phi = \{eq_1, \dots, eq_q\}$, equations over variables $X_i \in G_1, Y_j \in G_2, x_i, y_j \in \boldsymbol{Z}_p$
  - Pairing product equation defined by $A_j \in G_1, B_i \in G_2, \gamma_{ij} \in \boldsymbol{Z}_p$

$$\prod_{j\in[n]} e(A_j, Y_j) \cdot \prod_{i\in[m]} e(X_i, B_i) \cdot \prod_{i\in[m]} \prod_{j\in[n]} e(X_i, Y_j)^{\gamma_{ij}} = T$$

  - Multi-exponentiation equation in G_1 defined by $A_j, T \in G_1, b_i, \gamma_{ij} \in \boldsymbol{Z}_p$ (analogous for G_2)

$$\prod_{j\in[n']} A_j^{y_j} \cdot \prod_{i\in[m]} X_i^{b_i} \cdot \prod_{i\in[m]} \prod_{j\in[n']} X_i^{\gamma_{ij} y_j} = T$$

  - Quadratic equations defined by $a_j, b_i, \gamma_{ij}, t \in \boldsymbol{Z}_p$

$$\sum_{j\in[n']} a_j y_j + \sum_{i\in[m']} x_i b_i + \sum_{i\in[m']} \sum_{j\in[n']} x_i \gamma_{ij} y_j = t$$

- Witness $X_1, \dots, X_m \in G_1, Y_1, \dots, Y_n \in G_2, x_1, \dots, x_{m'}, y_1, \dots, y_{n'} \in \boldsymbol{Z}_p$ satisfying all eq_k

Statements – zero-knowledge

- Instance $\phi = \{eq_1, \dots, eq_q\}$, equations over variables $X_i \in G_1, Y_j \in G_2, x_i, y_j \in \boldsymbol{Z}_p$
  - Pairing product equation defined by $A_j \in G_1, B_i \in G_2, \gamma_{ij} \in \boldsymbol{Z}_p$

$$\prod_{j\in[n]} e(A_j, Y_j) \cdot \prod_{i\in[m]} e(X_i, B_i) \cdot \prod_{i\in[m]} \prod_{j\in[n]} e(X_i, Y_j)^{\gamma_{ij}} = 1$$

  - Multi-exponentiation equation in G_1 defined by $A_j, T \in G_1, b_i, \gamma_{ij} \in \boldsymbol{Z}_p$ (analogous for G_2)

$$\prod_{j\in[n']} A_j^{y_j} \cdot \prod_{i\in[m]} X_i^{b_i} \cdot \prod_{i\in[m]} \prod_{j\in[n']} X_i^{\gamma_{ij} y_j} = T$$

  - Quadratic equations defined by $a_j, b_i, \gamma_{ij}, t \in \boldsymbol{Z}_p$

$$\sum_{j\in[n']} a_j y_j + \sum_{i\in[m']} x_i b_i + \sum_{i\in[m']} \sum_{j\in[n']} x_i \gamma_{ij} y_j = t$$

- Witness $X_1, \dots, X_m \in G_1, Y_1, \dots, Y_n \in G_2, x_1, \dots, x_{m'}, y_1, \dots, y_{n'} \in \boldsymbol{Z}_p$ satisfying all eq_k

Commitments to field elements

- Setup includes $g, u, g', u' \in G_1, h, v, h', v' \in G_2$
- Now the prover will make commitments to $x \in \boldsymbol{Z}_p$ and $y \in \boldsymbol{Z}_p$ of the form

$$(g^r (g')^x, u^r (u'g)^x) \text{ and } (h^s (h')^y, v^s (v'g)^x)$$

- More precisely, for $x \in \boldsymbol{Z}_p$ the prover picks random $r \leftarrow \boldsymbol{Z}_p$ and computes a commitment as $(c_1, c_2) = (g, u)^r (g', u'g)^x$
- Recall the two setups
  - Binding setup $(g', u') = (g^\alpha, u^\alpha)$
  - Hiding setup $(g', u') = (g^\alpha, u^\alpha g^{-1})$

?

Indistinguishable under DDH

- So on binding setup $(c_1, c_2) = (g^{r+\alpha x}, u^{r+\alpha x} g^x)$, an encryption of g^x
- And on hiding setup $(c_1, c_2) = (g^{r+\alpha x}, u^{r+\alpha x})$, where r perfectly hides x

Proof example for quadratic equation

- Common reference string with $g, u, g', u' \in G_1, h, v, h', v' \in G_2$
- Suppose we have an instance with a single quadratic equation

$$xy = t$$

- Prover commits to x, y as

$$(c_1, c_2) = (g^r (g')^x, u^r (u'g)^x) \quad \text{and} \quad (d_1, d_2) = (h^s (h')^y, v^s (v'h)^y)$$

- Let us apply the extended bilinear map to the commitments

$$E\left(\begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, (d_1, d_2)\right) = E\left(\begin{pmatrix} g \\ u \end{pmatrix}^r \begin{pmatrix} g' \\ u'g \end{pmatrix}^x, (d_1, d_2)\right)$$

$$= E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (d_1, d_2)^r\right) E\left(\begin{pmatrix} g' \\ u'g \end{pmatrix}^x, (d_1, d_2)\right)$$

Proof example

$$= E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (d_1, d_2)^r\right) E\left(\begin{pmatrix} g' \\ u'g \end{pmatrix}^x, (h, v)^s (h', v'h)^y\right)$$

$$= E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (d_1, d_2)^r\right) E\left(\begin{pmatrix} g' \\ u'g \end{pmatrix}^{xs}, (h, v)\right) E\left(\begin{pmatrix} g' \\ u'g \end{pmatrix}^x, (h', v'h)^y\right)$$

$$= E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (d_1, d_2)^r (h, v)^t\right) E\left(\begin{pmatrix} g' \\ u'g \end{pmatrix}^{xs} \begin{pmatrix} g \\ u \end{pmatrix}^{-t}, (h, v)\right) E\left(\begin{pmatrix} g' \\ u'g \end{pmatrix}, (h', v'h)\right)^{xy}$$

for any $t \in \boldsymbol{Z}_p$

- The prover computes the proof elements as (using uniformly random $t \leftarrow \boldsymbol{Z}_p$)

$$(\pi_1, \pi_2) = (d_1, d_2)^r (h, v)^t \quad \text{and} \quad (\theta_1, \theta_2) = (g', u'g)^{xs} (g, u)^{-t}$$

Verification

- The verifier given the proof $(c_1, c_2, d_1, d_2, \pi_1, \pi_2, \theta_1, \theta_2)$ for $xy = t$ accepts if and only if

$$E\left(\begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, (d_1, d_2)\right) = E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (\pi_1, \pi_2)\right) E\left(\begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}, (h, v)\right) E\left(\begin{pmatrix} g' \\ u'g \end{pmatrix}, (h', v'h)\right)^t$$

- Perfect completeness when $xy = t$ follows from the calculations

- On a binding setup, where $(g', u') = (g, u)^\alpha$ and $(h', v') = (h, v)^\beta$, decryption vertically and horizontally shows the proof system is perfectly sound
  - It is not a proof of knowledge though, decryption gives you g^x and h^y instead of x, y
    Take for instance $(c_1, c_2) = (g^r (g')^x, u^r (u'g)^x) = (g^{r+\alpha x}, u^{r+\alpha x} g^x)$ and all you get is g^x

Witness indistinguishability

- On a hiding setup, where $(g', u') = (g^{\alpha}, u^{\alpha} g^{-1})$ and $(h', v') = \left(h^{\beta}, v^{\beta} h^{-1}\right)$, the proof system is perfectly witness indistinguishable
  - Commitments $(c_1, c_2), (d_1, d_2)$ are uniformly random
  - The proof elements $\pi_1, \pi_2, \theta_1, \theta_2$ are uniformly random conditioned on satisfying the verification equation

$$E\left(\begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, (d_1, d_2)\right) = E\left(\begin{pmatrix} g \\ u \end{pmatrix}, (\pi_1, \pi_2)\right) E\left(\begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}, (h, v)\right) E\left(\begin{pmatrix} g' \\ u' g \end{pmatrix}, (h', v' h)\right)^{t}$$

    - Randomization ($(\pi_1, \pi_2) = (d_1, d_2)^{r} (h, v)^{t}$) makes π_1 uniformly random
    - The top left corner of the verification equation then uniquely determines θ_1, the bottom left corner uniquely determines θ_2, and now the right top corner uniquely determines π_2

Proof size

- The common reference string has 8 elements $g, u, g', u' \in G_1, h, v, h', v' \in G_2$
- For a system of equations $\{eq_1, \dots, eq_q\}$ over variables X_i, Y_j, x_i, y_j

Variable/equation	Elements in G_1	Elements in G_2
$X \in G_1, x \in \boldsymbol{Z}_p$	2	0
$Y \in G_2, y \in \boldsymbol{Z}_p$	0	2
Pairing product	4	4
Multi-exponentiation in G_1	2	4
Multi-exponentiation in G_2	4	2
Quadratic	2	2

- Proofs may in some cases be smaller than the instance

  - For instance for q pairing-product equations over $X_1, \dots, X_m, Y_1, \dots, Y_n$ with many non-trivial γ_{ij} instance size is around mnq and proof size is $2m + 2n + 8q$

Statements – witness indistinguishability

- Instance $\phi = \{eq_1, \dots, eq_q\}$, equations over variables $X_i \in G_1, Y_j \in G_2, x_i, y_j \in \boldsymbol{Z}_p$
  - Pairing product equation defined by $A_j \in G_1, B_i \in G_2, \gamma_{ij} \in \boldsymbol{Z}_p$

$$\prod_{j\in[n]} e(A_j, Y_j) \cdot \prod_{i\in[m]} e(X_i, B_i) \cdot \prod_{i\in[m]} \prod_{j\in[n]} e\left(X_i, Y_j\right)^{\gamma_{ij}} = T$$

  - Multi-exponentiation equation in G_1 defined by $A_j, T \in G_1, b_i, \gamma_{ij} \in \boldsymbol{Z}_p$ (analogous for G_2)

$$\prod_{j\in[n']} A_j^{y_j} \cdot \prod_{i\in[m]} X_i^{b_i} \cdot \prod_{i\in[m]} \prod_{j\in[n']} X_i^{\gamma_{ij} y_j} = T$$

  - Quadratic equations defined by $a_j, b_i, \gamma_{ij}, t \in \boldsymbol{Z}_p$

$$\sum_{j\in[n']} a_j y_j + \sum_{i\in[m']} x_i b_i + \sum_{i\in[m']} \sum_{j\in[n']} x_i \gamma_{ij} y_j = t$$

- Witness $X_1, \dots, X_m \in G_1, Y_1, \dots, Y_n \in G_2, x_1, \dots, x_{m'}, y_1, \dots, y_{n'} \in \boldsymbol{Z}_p$ satisfying all eq_k

Statements – zero-knowledge

- Instance $\phi = \{eq_1, \ldots, eq_q\}$, equations over variables $X_i \in G_1, Y_j \in G_2, x_i, y_j \in \boldsymbol{Z}_p$
  - Pairing product equation defined by $A_j \in G_1, B_i \in G_2, \gamma_{ij} \in \boldsymbol{Z}_p$

$$\prod_{j\in[n]} e(A_j, Y_j) \cdot \prod_{i\in[m]} e(X_i, B_i) \cdot \prod_{i\in[m]} \prod_{j\in[n]} e\left(X_i, Y_j\right)^{\gamma_{ij}} = 1$$

  - Multi-exponentiation equation in G_1 defined by $A_j, T \in G_1, b_i, \gamma_{ij} \in \boldsymbol{Z}_p$ (analogous for G_2)

$$\prod_{j\in[n']} A_j^{y_j} \cdot \prod_{i\in[m]} X_i^{b_i} \cdot \prod_{i\in[m]} \prod_{j\in[n']} X_i^{\gamma_{ij} y_j} = T$$

  - Quadratic equations defined by $a_j, b_i, \gamma_{ij}, t \in \boldsymbol{Z}_p$

$$\sum_{j\in[n']} a_j y_j + \sum_{i\in[m']} x_i b_i + \sum_{i\in[m']} \sum_{j\in[n']} x_i \gamma_{ij} y_j = t$$

- Witness $X_1, \ldots, X_m \in G_1, Y_1, \ldots, Y_n \in G_2, x_1, \ldots, x_{m'}, y_1, \ldots, y_{n'} \in \boldsymbol{Z}_p$ satisfying all eq_k

Simulation strategy

- Use trivial witness

$$X_1 = 1, \ldots, X_m = 1 \qquad Y_1 = 1, \ldots, Y_n = 1 \qquad x_1 = 0, \ldots, y_{n'} = 0$$

- Works well for the pairing-product equations

$$\prod_{j \in [n]} e(A_j, Y_j) \cdot \prod_{i \in [m]} e(X_i, B_i) \cdot \prod_{i \in [m]} \prod_{j \in [n]} e\left(X_i, Y_j\right)^{\gamma_{ij}} = 1$$

- Maybe not so well for the other equations? For instance

$$\sum_{j \in [n']} a_j y_j + \sum_{i \in [m']} x_i b_i + \sum_{i \in [m']} \sum_{j \in [n']} x_i \gamma_{ij} y_j = t$$

with non-trivial $t \neq 0$

Zero-knowledge for non-trivial targets

- A quadratic equation with $t \neq 0$ can be rewritten as

$$\sum_{j \in [n']} a_j y_j + \sum_{i \in [m']} x_i b_i + 1 \cdot (-t) + \sum_{i \in [m']} \sum_{j \in [n']} x_i \gamma_{ij} y_j = 0$$

- Observe $(g', u'g) = (g, u)^0 (g', u'g)^1$ is commitment to 1 with $r = 0$
  - On a binding string $(g', u'g)$ is perfectly binding to 1, so we have perfect soundness
  - On a hiding string, $(g', u'g) = (g, u)^\alpha (g', u'g)^0$ so it is also a commitment to 0
    - The simulator can use $x_1 = \cdots = y_{n'} = 0$ and "$1 = 0$" to simulate proof
    - By perfect witness indistinguishability, the simulated proof looks exactly like a real proof

Zero-knowledge for non-trivial targets

- A multi-exponentiation equation in G_1 with $T \neq 1$ can be rewritten as

$$\prod_{j\in[n']} A_j^{y_j} \cdot (T^{-1})^1 \cdot \prod_{i\in[m]} X_i^{b_i} \cdot \prod_{i\in[m]} \prod_{j\in[n']} X_i^{\gamma_{ij} y_j} = 1$$

- Using $(h', v'h) = (h, v)^0 (h', v'h)^1$ is commitment to 1 with $s = 0$
  - On a binding string it is unconditionally binding, so we have perfect soundness
  - On a hiding string also commitment to 0 since $(h', v'h) = (h, v)^\beta (h', v'h)^0$, so we can simulate a proof using the trapdoor β
- Btw, the proofs you prove/simulate are exactly the same as in the WI case

Statements – zero-knowledge

- Instance $\phi = \{eq_1, \ldots, eq_q\}$, equations over variables $X_i \in G_1, Y_j \in G_2, x_i, y_j \in \boldsymbol{Z}_p$
    - Pairing product equation defined by $A_j \in G_1, B_i \in G_2, \gamma_{ij} \in \boldsymbol{Z}_p$

$$\prod_{j \in [n]} e(A_j, Y_j) \cdot \prod_{i \in [m]} e(X_i, B_i) \cdot \prod_{i \in [m]} \prod_{j \in [n]} e\left(X_i, Y_j\right)^{\gamma_{ij}} = 1$$

    - Multi-exponentiation equation in G_1 defined by $A_j, T \in G_1, b_i, \gamma_{ij} \in \boldsymbol{Z}_p$ (analogous for G_2)

$$\prod_{j \in [n']} A_j^{y_j} \cdot \prod_{i \in [m]} X_i^{b_i} \cdot \prod_{i \in [m]} \prod_{j \in [n']} X_i^{\gamma_{ij} y_j} = T$$

    - Quadratic equations defined by $a_j, b_i, \gamma_{ij}, t \in \boldsymbol{Z}_p$

$$\sum_{j \in [n']} a_j y_j + \sum_{i \in [m']} x_i b_i + \sum_{i \in [m']} \sum_{j \in [n']} x_i \gamma_{ij} y_j = t$$

- Simulate all proofs using $X_i = 1, Y_j = 1$, $x_i = 0, y_j = 0$ and trapdoors α, β