Sistema de Votación Electrónica Basado en ElGamal Homomórfico

Universidad del Norte
Octubre de 2025

Luis Cabarcas
Matemáticas
lcabarcas@uninorte.edu.co

Andres España
Ciencia de Datos
coreoSpain@uninorte.edu.co

Ashley Mercado
Matemáticas
agmercado@uninorte.edu.co

Resumen

Este proyecto implementa un sistema de votación electrónica de aula basado en criptografía homomórfica ElGamal multiplicativa. El protocolo garantiza privacidad del voto, integridad del conteo y verificabilidad pública mediante pruebas de conocimiento cero no-interactivas (NIZK). Se describe la arquitectura criptográfica, el flujo de emisión con tokens de elegibilidad, la validación de votos y el mecanismo de acumulación y conteo agregado que explota las propiedades homomórficas del esquema ElGamal. Se proporciona un análisis de seguridad, limitaciones del diseño y consideraciones para implementación práctica en un contexto educativo.

votación electrónica, criptografía homomórfica, ElGamal, pruebas de conocimiento cero, conteo agregado

1. Introducción

La votación electrónica representa un desafío criptográfico fundamental: recopilar preferencias de múltiples votantes de manera que se satisfagan simultáneamente la privacidad individual y la verificabilidad colectiva del resultado. Los sistemas de votación tradicionales basados en urnas físicas no escalan eficientemente en contextos digitales, donde la verificación manual es imposible y la confianza debe depositarse en algoritmos.

La solución propuesta en este trabajo aprovecha las propiedades homomórficas del esquema de cifrado ElGamal. A diferencia de esquemas donde cada voto se cifra y descifra independientemente, el homomorfismo multiplicativo permite que el Centro de Conteo (VTC) agregue los votos cifrados algebraicamente, revelando únicamente la suma total sin exponer información sobre votos individuales.

Este proyecto cubre el diseño completo de un protocolo de votación electrónica para contextos de aula, integrando:

- Cifrado homomórfico ElGamal multiplicativo
- Pruebas no-interactivas de conocimiento cero (NIZK) para validación de votos
- Flujo de elegibilidad mediante tokens firmados

- Mecanismo de conteo agregado que preserva privacidad
- Análisis de seguridad bajo supuestos estándar (DDH, oráculo aleatorio)

El objetivo es proporcionar una implementación educativa que demuestre cómo principios criptográficos rigurosos pueden resolver el problema de votación electrónica de forma práctica y verificable.

2. Estado del Arte

2.1. Características de sistemas de votación electrónica de los Sistemas de Votación Electrónica

Los sistemas de votación electrónica han evolucionado desde ser simples sistemas de recolección digital hasta convertirse en complejos y eficientes protocolos criptográficos en los cuales se garantiza la privacidad del voto y su verificabilidad publica. Las características principales que identifican estos tipos de sistemas son:

- **Verificabilidad**: Se puede verificar que un voto fue contabilizado correctamente.
- **Privacidad:** El contenido del voto debe permanecer secreto
- **Eligibilidad:** Solo votantes autorizados pueden votar.
- ***Fairness:*** No se pueden revelar resultados parciales antes del cierre.

2.2. Criptografía Homomórfica en Votación Electrónica

La criptografía homomorfica es un esquema dinámico en el cual dado dos textos cifrados, se pueden realizar operaciones. Por ejemplo, una operación homomorfica de *"suma"* retornaría un texto cifrado el cual al ser descifrado mostraría la suma de los dos elementos originales. Esto permite hacer manipulación de los datos sin exponerlos a brechas de seguridad generados porque se tenían que desencriptar primero, como la contabilidad de un sistema de votación.

La palabra homomorfismo es un concepto matemático, que formalmente es una mapeo en una estructura matemática a otra, esto es:

2.2.1. ElGamal Multiplicativo para Votación Electrónica

Suponga que tiene n votantes cuyas opcones de votacion son *"si"* o *"no"*, o para ser mas simples, 0 o 1. Si queremos mantener los votos privados un esquema propuesto es la varieante multiplicativa de ElGamal [1].

Definición 2.1 (ElGamal Multiplicativo)**.** Sea $\mathbb{G}$ un grupo cíclico de orden primo q generado por $g \in \mathbb{G}$.

Considere una variante simple del sistema de cifrado ElGamal $\mathcal{E}_{\text{MFG}} = (G, E, D)$ que está definido sobre $(\mathbb{G}, \mathbb{G}^2)$. El algoritmo de generación de claves G es el mismo que en el ElGamal estandar, pero el cifrado y descifrado funcionan como sigue:

- para una clave pública dada $pk = u \in \mathbb{G}$ y mensaje $m \in \mathbb{G}$:

$$E(pk, m) := \quad \beta \xleftarrow{\$} \mathbb{Z}_q, \quad v \leftarrow g^{\beta}, \quad e \leftarrow u^{\beta} \cdot m, \quad \text{output } (v, e)$$

- para una clave secreta dada $sk = \alpha \in \mathbb{Z}_q$ y un texto cifrado $(v, e) \in \mathbb{G}^2$:

$$D(sk, (v, e)) := e/v^{\alpha}$$

Propiedad de Homomorfismo Multiplicativo: $\mathcal{E}_{\text{MFG}}$ tiene la siguiente propiedad: dada una clave pública pk, y dos textos cifrados $c_1 = (v_1, e_1) \xleftarrow{\$} E(pk, m_1)$ y $c_2 = (v_2, e_2) \xleftarrow{\$} E(pk, m_2)$, es posible crear un nuevo texto cifrado $c = (v_1 \cdot v_2, e_1 \cdot e_2)$ que es el cifrado de $m_1 \cdot m_2$. Esta propiedad se llama **homomorfismo multiplicativo**.[1]

2.3. Problema de la Validez del Voto

Una forma de hacer trampa en este sistema criptográfico es votar algo diferente a 0 o 1. Suponga que alguien malicioso en vez de encriptar g^0 o g^1 encripta el elemento del grupo ciclico g^{100}, que en nuestro sistema es equivalente a realizar 100 votos de 1. Para evitar eso, una forma es tener un metodo de verificacion que el voto encriptado sea de un formato valido. Tal verificación se puede hacer aplicando una transformación Fiat-Shamir, la cual convierte protocolos interactivos en no-interactivos mediante hash del *commitment*. Despues de aplicar la transformacion se obtiene una propiedad de *Zero-Knowledge*, es decir, se puede probar que un voto es valido sin revelar su contenido y ademas se dice que la prueba *sound*, esto es, solo se puede demostrar que se cifro 0 o 1 si realmente fue el caso.[1]

2.4. Propiedades del Sistema del ElGamal Multiplicativo

El sistema de cifrado ElGamal multiplicativo mediante la transformacion Fiat-Shamir utilizado en el protocolo de votación satisface las siguientes propiedades definidas anteriormente:

- **Sistema de Prueba No Interactivo:** Mediante la transformada de Fiat-Shamir, los votantes pueden generar pruebas no interactivas de que sus votos cifrados son válidos (0 o 1).
- ***Soundness* No Interactiva:** Bajo el modelo de oráculo aleatorio, es computacionalmente inviable para un votante corrupto crear una prueba válida de que su voto es 0 o 1 cuando en realidad no lo es.

- ***Zero Knowledge* No Interactivo:** Las pruebas generadas no revelan información adicional sobre el voto real del votante, más allá de confirmar que es válido. Esto preserva la privacidad del voto.

Estas propiedades, combinadas con el homomorfismo multiplicativo del esquema ElGamal, permiten crear un protocolo de votación que es tanto verificable (los votos pueden ser validados) como privado (los votos individuales permanecen secretos).

3. Diseño del Sistema

4. Diseño del Sistema

El diseño del sistema de votación electrónica se fundamenta en la combinación de cifrado homomórfico ElGamal multiplicativo y pruebas de conocimiento cero no-interactivas (NIZK). Esta sección detalla las decisiones arquitectónicas y los componentes criptográficos que garantizan privacidad del voto, integridad del conteo y verificabilidad pública.

4.1. Arquitectura Criptográfica

4.1.1. Selección del Grupo Cíclico

La seguridad del esquema ElGamal depende críticamente de la elección del grupo cíclico subyacente $\mathbb{G}$. Se consideran dos opciones principales:

Opción 1: Grupo Multiplicativo Módulo p Sea $p = 2q + 1$ un primo seguro, donde q es también primo. El grupo $\mathbb{G} = \langle g \rangle \subset \mathbb{Z}_p^*$ tiene orden primo q.

- **Ventajas**: Implementación directa, ampliamente documentada en literatura criptográfica, compatible con bibliotecas estándar.
- **Desventajas**: Requiere tamaños de clave grandes ($\geq$ 2048 bits para seguridad de 112 bits), operaciones modulares costosas.

Opción 2: Grupo de Curva Elíptica Utilizando Curve25519 o NIST P-256, donde la operación de grupo es la suma de puntos.

- **Ventajas**: Claves de 256 bits proveen seguridad equivalente a RSA-3072, operaciones más eficientes, menor ancho de banda.
- **Desventaja**: Requiere biblioteca especializada de curvas elípticas.

4.1.2. Esquema ElGamal Multiplicativo

Configuración: Una autoridad de confianza (Vote Tallying Center, VTC) ejecuta:

1. Selecciona $\alpha \xleftarrow{\$} \mathbb{Z}_q$ como clave privada
2. Computa $u = g^\alpha \in \mathbb{G}$ como clave pública
3. Publica $(\mathbb{G}, q, g, u)$ y mantiene α en secreto

Cifrado de voto $b \in \{0,1\}$: Para cifrar un voto $b \in \{0,1\}$:

$$m = g^b \quad \text{(codificación del voto)} \tag{1}$$

$$\beta \xleftarrow{\$} \mathbb{Z}_q \quad \text{(aleatoriedad fresca)} \tag{2}$$

$$v = g^\beta \tag{3}$$

$$e = u^\beta \cdot m = g^{\alpha\beta+b} \tag{4}$$

El voto cifrado es el par $(v, e) \in \mathbb{G} \times \mathbb{G}$.

Propiedad homomórfica: Para votos cifrados (v_i, e_i) con $i = 1, \ldots, n$:

$$\left(\prod_{i=1}^{n} v_i, \prod_{i=1}^{n} e_i\right) = \text{Enc}\left(g^{\sum_{i=1}^{n} b_i}\right) \tag{5}$$

Esto permite acumular votos sin descifrarlos individualmente.

Descifrado del agregado: Dado el cifrado acumulado $(v^*, e^*) = (\prod_i v_i, \prod_i e_i)$:

$$m^* = \frac{e^*}{(v^*)^\alpha} = g^\sigma \quad \text{donde } \sigma = \sum_{i=1}^{n} b_i \tag{6}$$

Recuperación de σ: Como $\sigma \in [0, n]$ es pequeño (contexto de aula), se recupera mediante:

- **Búsqueda exhaustiva**: Computar $g^0, g^1, \ldots, g^n$ hasta encontrar coincidencia
- **Tabla de lookup**: Precalcular $\{g^i : i \in [0, n]\}$ para búsqueda en tiempo constante
- **Baby-step giant-step**: Para n grandes (no aplicable en aula)

4.2. Prueba NIZK de Validez del Voto

La componente central del sistema es la prueba no-interactiva que certifica que un voto cifrado (v, e) es válido, es decir, cifra g^0 o g^1, sin revelar cuál de los dos.

4.2.1. Declaración a Probar

***Statement* público:** $y = (u, v, e) \in \mathbb{G}^3$

***Witness* secreto:** $x = (b, \beta) \in \{0, 1\} \times \mathbb{Z}_q$

Relación: $R = \{((b, \beta), (u, v, e)) : v = g^\beta \wedge e = u^\beta \cdot g^b\}$

El *prover* debe demostrar:

$$(v = g^\beta \wedge e = u^\beta \cdot g^0) \vee (v = g^\beta \wedge e = u^\beta \cdot g^1) \tag{7}$$

sin revelar cuál disyunción es verdadera.

4.2.2. Protocolo Sigma Disjuntivo

El protocolo se basa en pruebas OR: demostrar conocimiento de uno de dos *witnesses* sin revelar cuál.

Idea clave:

- **Rama verdadera**: Generar *commitment* honestamente
- **Rama falsa**: Simular hacia atrás usando ecuaciones de verificación invertidas

4.3. Consideraciones Criptográficas

4.3.1. Función Hash Criptográfica

Selección: SHA-256 o SHA3-256 (FIPS 202) modelado como oráculo aleatorio en el análisis de seguridad.

4.3.2. Generación de Aleatoriedad

Requisitos críticos:

- β debe ser generado por un generador de números pseudoaleatorios criptográficamente seguro (CSPRNG)
- Distribución uniforme en $\mathbb{Z}_q$
- Reusar β permite recuperar la clave secreta α mediante análisis de diferencias

4.3.3. Validación de Entradas

Antes de procesar cualquier prueba, el verificador debe validar:

1. $v, e \in \mathbb{G}$ (pertenecen al grupo)

2. $v, e \neq \mathcal{O}$ (no son el elemento neutro)
3. $\text{order}(v) = q$ y $\text{order}(e) = q$ (tienen el orden correcto)
4. $\beta_{z0}, \beta_{z1} \in \mathbb{Z}_q$ (están en el rango válido)
5. c_0, c_1 tienen la longitud esperada según la función hash utilizada

Limitaciones conocidas:

- **No *receipt-freeness***: El votante puede conservar (β, b) como recibo coercible.
- **Dependencia del VTC**: La autoridad puede descifrar votos individuales (requiere *threshold decryption* para mitigar)
- **No robustez ante VTC malicioso**: Se asume confianza en el VTC (extensiones futuras pueden eliminar esto mediante técnicas de *distributed key generation*)

4.3.4. Justificación de Decisiones de Diseño

4.3.5. ¿Por qué ElGamal?

- ElGamal es más estándar en literatura académica de votación electrónica
- Compatible con curvas elípticas (mayor eficiencia computacional)
- ElGamal tiene análisis de seguridad más establecido en el contexto de protocolos Sigma

4.3.6. ¿Por qué NIZK y no protocolos interactivos?

- Votación requiere asincronía (votantes en diferentes momentos)
- Verificabilidad pública (cualquier observador puede auditar sin interactuar)
- Menor complejidad de coordinación
- No requiere mantener estado del verificador

4.3.7. ¿Por qué Fiat-Shamir?

- Estándar de la industria ampliamente analizado
- Seguridad bien entendida en el modelo de oráculo aleatorio
- Balance óptimo entre eficiencia y seguridad
- Transformación directa desde protocolos Sigma bien estudiados

5. Flujo de Emisión con Token

El flujo de emisión con token define el mecanismo mediante el cual los votantes autorizados pueden emitir un voto único y válido, preservando la **privacidad** y garantizando la **elegibilidad**. Este componente conecta la autenticación del votante con la emisión del voto cifrado bajo el esquema **ElGamal homomórfico multiplicativo**, incorporando la **Prueba No Interactiva de Conocimiento Cero (NIZK)** como requisito de validez.

5.1. Arquitectura de las Entidades

La emisión de tokens y el registro del voto se basan en la existencia de tres entidades con responsabilidades separadas para asegurar la privacidad y la integridad del proceso:

Entidad	**Responsabilidad Principal**	**Objetivo Criptográfico**
Autoridad de Registro (RA)	Elegibilidad y Unicidad	Emisión de tokens firmados (T_i)
Centro de Votación (VC)	Verificabilidad del Voto	Valida el token y la prueba NIZK.
Centro de Conteo (VTC)	Privacidad y Totalización	Posee la clave secreta para el descifrado.

Cada votante legítimo obtiene de la RA un **token electrónico** T_i que acredita su derecho a sufragar. Dicho token se genera con una firma digital de la RA y no contiene información identificable del votante, asegurando así el anonimato y la unicidad.

5.2. Etapas del Proceso

El flujo de emisión se implementa mediante tres fases:

5.2.1. Emisión del Token de Elegibilidad

El votante se autentica ante la RA. Tras verificar la elegibilidad del votante, la RA genera un identificador aleatorio y único T_i, que firma digitalmente con su clave privada. El votante recibe T_i y la RA marca el estado del votante como *autenticado*, evitando la emisión de tokens duplicados. Este paso desacopla la identidad del votante de su acto de voto.

5.2.2. Cifrado del Voto y Generación de la Prueba NIZK

El votante procede localmente utilizando la clave pública $pk = u$ (Sección 4.1.2):

1. **Cifrado de ElGamal:** El votante codifica su decisión $b_i \in \{0, 1\}$ como $m_i = g^{b_i}$ y la cifra con un valor aleatorio fresco $\beta \in \mathbb{Z}_q$:

$$\mathbf{c}_i = (v_i, e_i) = E(pk, g^{b_i}) \tag{8}$$

2. **Generación de la Prueba NIZK:** El votante utiliza el *witness* secreto $x = (b_i, \beta)$ y el protocolo Sigma Disjuntivo para generar la prueba no interactiva π^*. Dicha prueba certifica que el par $\mathbf{c}_i$ pertenece a la relación de validez $\mathcal{R}$, es decir, que cifra g^0 o g^1 sin revelar cuál.

5.2.3. Verificación y Registro en el VC

El votante envía el paquete $\left(T_i, \mathbf{c}_i, \pi^*\right)$ al Centro de Votación (VC) para su registro:

1. **Validación de T_i:** El VC verifica la firma de la RA en T_i y comprueba que el token no haya sido usado previamente, garantizando la **Unicidad** del sufragio.

2. **Validación Criptográfica:** El VC ejecuta el algoritmo de verificación $VrfyPrf(\mathbf{c}_i, \pi^*)$. Si $\mathbf{c}_i$ es un cifrado válido y la prueba cumple la propiedad de *Soundness No Interactiva*, la verificación es exitosa.

3. **Publicación:** Si las verificaciones son válidas, el VC publica $\left(\mathbf{c}_i, \pi^*\right)$ en el Boletín de Votos y marca el token T_i como consumido.

6. Formato del Voto Cifrado

El formato del voto cifrado define la estructura del mensaje que el votante transmite al Centro de Votación (VC), garantizando que cada sufragio sea verificable, anónimo y acumulable bajo el esquema homomórfico descrito anteriormente.

6.1. Estructura del Paquete de Voto

Cada votante i genera y envía un paquete digital:

$$\mathcal{P}_i = (T_i, \mathbf{c}_i, \pi_i^*)$$

donde:

- T_i: Token de elegibilidad firmado por la Autoridad de Registro (RA).
- $\mathbf{c}_i = (v_i, e_i)$: Cifrado de ElGamal del voto $b_i \in \{0, 1\}$, con $v_i = g^{\beta_i}$ y $e_i = u^{\beta_i} \cdot g^{b_i}$.
- π_i^*: Prueba no interactiva de conocimiento cero (NIZK) que certifica que $\mathbf{c}_i$ cifra un valor válido g^0 o g^1.

6.2. Validación en el Centro de Votación

El VC ejecuta las siguientes verificaciones:

1. Comprueba la firma de la RA sobre T_i y que no haya sido reutilizado.

2. Verifica la prueba π_i^* mediante $VrfyPrf(\mathbf{c}_i, \pi_i^*)$.

3. Si ambas validaciones son correctas, publica $(\mathbf{c}_i, \pi_i^*, \text{hash}(T_i))$ en el boletín público y marca el token como consumido.

7. Acumulación y Conteo Agregado

7.1. Fundamento Teórico del Homomorfismo en la Agregación

La propiedad fundamental que permite el conteo agregado es el homomorfismo multiplicativo del esquema ElGamal. Esta propiedad establece que el producto de dos cifrados es equivalente al cifrado del producto de los mensajes, lo que permite realizar operaciones algebraicas en el espacio cifrado sin acceso a la clave privada.

Formalmente, si $c_1 = (v_1, e_1)$ es un cifrado de m_1 y $c_2 = (v_2, e_2)$ es un cifrado de m_2 bajo la misma clave pública u, entonces el par $(v_1 \cdot v_2, e_1 \cdot e_2)$ representa un cifrado válido del producto $m_1 \cdot m_2$. Esta propiedad se generaliza a n cifrados, permitiendo agregar múltiples votos mediante multiplicación repetida.

En el contexto de votación, cada voto se codifica como $b_i \in \{0, 1\}$ y se transforma al mensaje de grupo $m_i = g^{b_i}$. La agregación posterior de estos mensajes en el espacio cifrado produce un resultado que, al ser descifrado, revela la suma total $\sigma = \sum_{i=1}^{n} b_i$ sin exponer información sobre votos individuales.

7.2. Etapa de Agregación de Cifrados

Una vez que todos los votos han sido emitidos, validados y publicados en el boletín de votación, el Centro de Conteo de Votos (VTC) ejecuta la etapa de agregación. Este proceso es completamente determinístico y público, es decir, cualquier observador puede reproducirlo y verificar su correctitud sin acceso a información secreta.

Sea n el número total de votantes. Cada votante i ha producido un cifrado válido:

$$\mathbf{c}_i = (v_i, e_i) = (g^{\beta_i}, u^{\beta_i} \cdot g^{b_i}) \tag{9}$$

donde $\beta_i \in \mathbb{Z}_q$ es elegido aleatoriamente por el votante durante el cifrado, y $b_i \in \{0, 1\}$ es su voto.

El VTC agrega todos estos cifrados mediante la multiplicación componente a componente:

$$v^* := \prod_{i=1}^{n} v_i = \prod_{i=1}^{n} g^{\beta_i} = g^{\sum_{i=1}^{n} \beta_i} \tag{10}$$

$$e^* := \prod_{i=1}^{n} e_i = \prod_{i=1}^{n} \left(u^{\beta_i} \cdot g^{b_i}\right) = u^{\sum_{i=1}^{n} \beta_i} \cdot g^{\sum_{i=1}^{n} b_i} \tag{11}$$

Definiendo $\Gamma := \sum_{i=1}^{n} \beta_i$ como la suma acumulada de los parámetros de aleatoriedad y $\sigma := \sum_{i=1}^{n} b_i$ como el total de votos afirmativos, se obtiene:

$$(v^*, e^*) = (g^\Gamma, u^\Gamma \cdot g^\sigma) \tag{12}$$

Este par es un cifrado válido bajo el esquema ElGamal con aleatoriedad acumulada Γ y mensaje plano g^σ. La corrección de esta agregación proviene directamente de las propiedades algebraicas del grupo $\mathbb{G}$ y de la definición del esquema de cifrado.

7.3. Desciframiento del Resultado Agregado

Una vez completada la agregación, el VTC procede a descifrar el par agregado (v^*, e^*) utilizando su clave privada α. El algoritmo de desciframiento ElGamal es:

$$g^\sigma = \frac{e^*}{(v^*)^\alpha} \tag{13}$$

Sustituyendo los valores agregados:

$$g^\sigma = \frac{u^\Gamma \cdot g^\sigma}{(g^\Gamma)^\alpha} \tag{14}$$

$$= \frac{g^{\alpha\Gamma} \cdot g^\sigma}{g^{\alpha\Gamma}} \tag{15}$$

$$= g^\sigma \tag{16}$$

El resultado es el elemento del grupo g^σ, donde σ es el total de votos. Esta operación es la única en la que se utiliza la clave privada del VTC; ningún voto individual es descifrado en ningún momento del proceso.

7.4. Recuperación del Total de Votos

El desciframiento produce el elemento del grupo g^σ, pero el resultado final debe ser el número entero σ. Dado que en el contexto de una votación de aula n es pequeño, se tiene que $\sigma \in [0, n]$, lo que hace que la recuperación del exponente sea computacionalmente trivial.

El VTC dispone de dos estrategias principales:

7.4.1. Tabla de Precálculo

Durante la fase de inicialización, el VTC genera y almacena la tabla:

$$\text{Tabla} = \{(g^j, j) : j \in [0, n]\} \tag{17}$$

Esta tabla es independiente de los votantes específicos y se construye una única vez. Durante el desciframiento, el VTC busca la coincidencia de g^σ en esta tabla y recupera el exponente correspondiente en tiempo $O(1)$ con búsqueda indexada.

7.4.2. Búsqueda Exhaustiva

Alternativamente, el VTC itera computando:

$$g^0, g^1, g^2, \ldots, g^n \tag{18}$$

hasta encontrar el valor que coincide con g^σ. Esta estrategia tiene complejidad temporal $O(n)$ pero no requiere almacenamiento previo de la tabla.

En ambos casos, la recuperación es instantánea para valores pequeños de n típicos en una votación de aula (por ejemplo, $n = 50$ estudiantes).

7.5. Publicación y Verificabilidad del Resultado

Una vez recuperado σ, el VTC lo publica acompañado de evidencia que permite a cualquier observador verificar su corrección. La publicación típicamente incluye:

1. El valor σ (el total de votos afirmativos)
2. El valor $n - \sigma$ (el total de votos negativos)
3. El agregado cifrado (v^*, e^*) que fue descifrado
4. El boletín completo de cifrados publicados (v_i, e_i) para $i = 1, \ldots, n$

Un verificador independiente puede:

1. Replicar la agregación: computar $\prod_{i=1}^{n} v_i$ y $\prod_{i=1}^{n} e_i$
2. Verificar que el resultado coincide con (v^*, e^*) publicado
3. Verificar que el desciframiento es correcto al computar $\frac{e^*}{(v^*)^\alpha}$

Esta verificabilidad algebraica es una propiedad fundamental del esquema ElGamal y no depende de la confianza en el VTC para la agregación (aunque sí en el VTC para la guarda de la clave privada).

8. Conclusiones

Este proyecto ha desarrollado un protocolo completo de votación electrónica que integra criptografía homomórfica ElGamal multiplicativa con pruebas de conocimiento cero no-interactivas. El diseño propuesto garantiza privacidad del votante, integridad del conteo mediante validación NIZK de votos, elegibilidad a través de tokens firmados, y verificabilidad pública de la agregación algebraica. Bajo supuestos criptográficos estándar (DDH, oráculo aleatorio), los votos individuales permanecen privados mientras se permite la auditoría transparente del resultado total. El protocolo es particularmente adecuado para votaciones de aula y contextos institucionales de escala pequeña a mediana ($n \leq 10000$), donde la confianza en una autoridad central es razonable. Aunque

presenta limitaciones como la dependencia del VTC, estas pueden mitigarse mediante técnicas avanzadas como *threshold cryptography* en trabajo futuro. El trabajo demuestra cómo la criptografía moderna proporciona herramientas prácticas para resolver el problema de votación electrónica preservando simultáneamente privacidad y verificabilidad.

Referencias

[1] D. Boneh and V. Shoup, *A Graduate Course in Applied Cryptography*. 2023. Version 0.6.