ARTÍCULO: METAHEURISTIC-SUPPORTED IMAGE ENCRYPTION FRAMEWORK BASED ON BINARY SEARCH TREE AND DNA ENCODING

INTEGRANTES:

- PAMELA BLÁCIDO
- CARLA CHAVEZ
- PAOLA CONDOR

INTRODUCCIÓN

Problemática

- Importancia de proteger datos sensibles en transmisión digital.
- Imágenes con información visual son objetivos frecuentes de ataques cibernéticos.
- Métodos de cifrado tradicionales (DES, AES) no son efectivos.
- Alta correlación de píxeles expone secciones de la imagen.
- Propuesta de un enfoque novedoso:
  - Basado en árboles de búsqueda binarios (BST).
  - Codificación de ADN para superar limitaciones.

Objetivo del Artículo

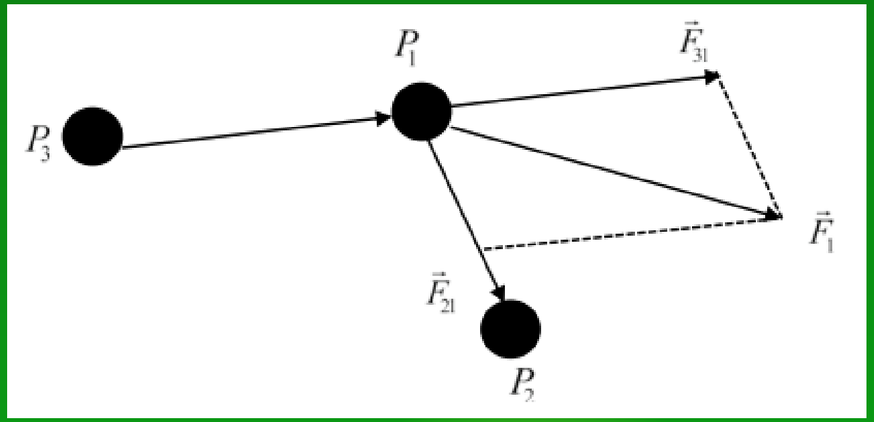
Proponer un framework de cifrado robusto que:

- Utilice la estructura de árboles de búsqueda binarios (BST) para organizar y cifrar los píxeles de la imagen.
- Emplee codificación ADN para asegurar una mayor complejidad.
- Optimice la estructura del árbol usando un enfoque metaheurístico basado en electromagnetismo (EMO).

Metodología empleada

- Organización de píxeles en el árbol.
- Codificación con ADN.
- Optimización de la estructura para mejorar la seguridad y eficiencia.

Fig. 1 Demonstration of computing resultant force using superposition principle



Uso de Árboles de Búsqueda Binarios en el Cifrado

Los árboles de búsqueda binaria (BST) se utilizan para:

- Asignar cada píxel a un nodo del árbol.
- Definir el orden de cifrado de los píxeles.
- Minimizar la predictibilidad de los patrones de píxeles.
- Cada nodo representa un valor de píxel que es codificado en ADN utilizando combinaciones de bases (A, C, G, T). El recorrido del BST determina cómo se organiza la imagen antes de realizar el cifrado final.

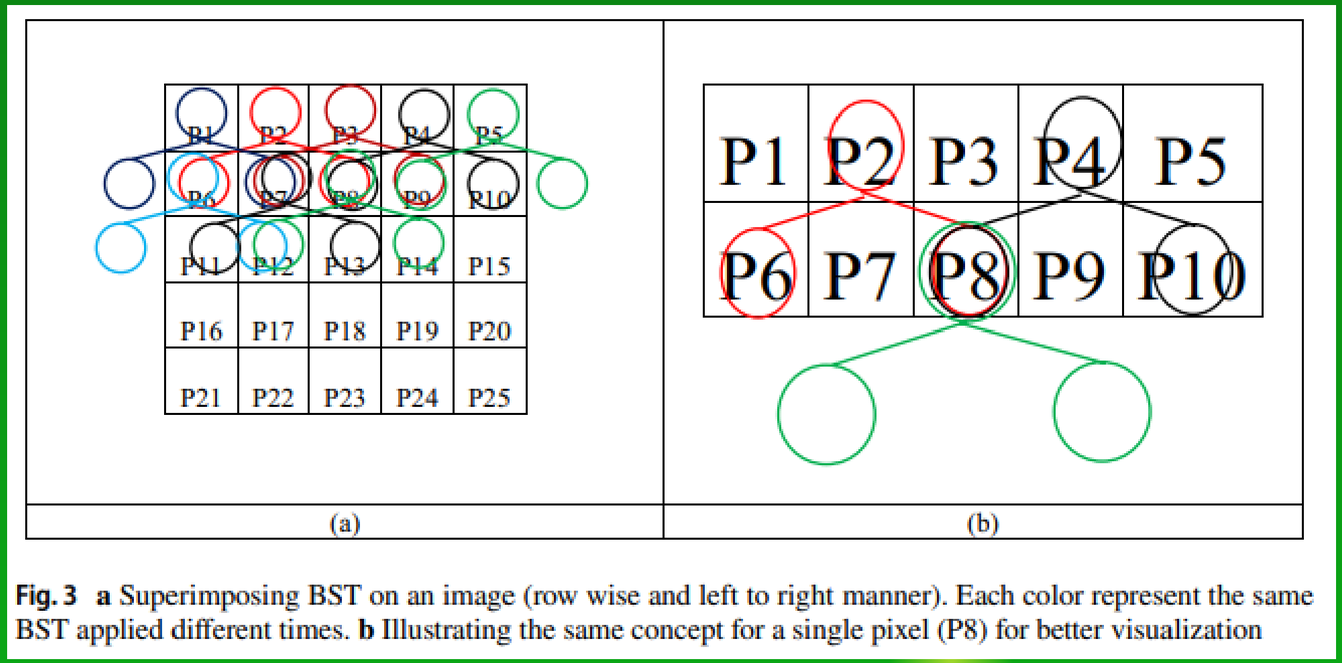


Fig. 3 **a** Superimposing BST on an image (row wise and left to right manner). Each color represent the same BST applied different times. **b** Illustrating the same concept for a single pixel (P8) for better visualization

Codificación ADN y su Uso

El cifrado implica la codificación de píxeles a cadenas de ADN:

- Se convierte cada valor de píxel en bases de ADN: adenina (A), citosina (C), guanina (G) y timina (T).
- Operaciones biológicas como intercambio de bases y mutación aseguran la complejidad del cifrado.

Ventajas de la codificación ADN:

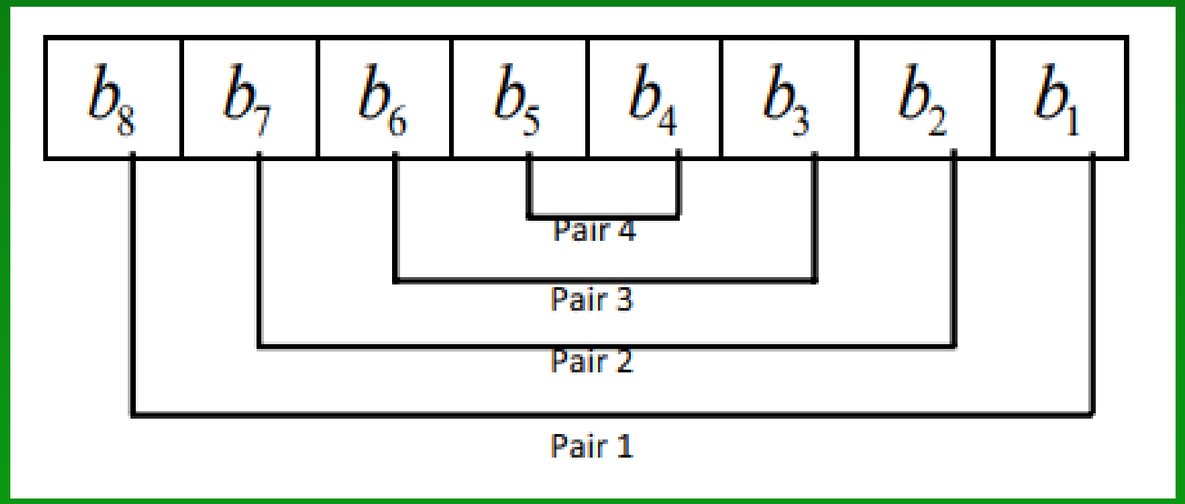
- Alta entropía en los datos.
- Dificultad para realizar análisis estadístico.
- Resistencia a ataques de fuerza bruta.

Table 1 Eight possible rules for DNA encoding and decoding rules

Binary combinations	R #1	R #2	R #3	R #4	R #5	R #6	R #7	R #8
00	A	A	T	T	C	C	G	G
01	C	G	C	G	A	T	A	T
10	G	C	G	C	T	A	T	A
11	T	T	A	A	G	G	C	C

Optimización Metaheurística con Electromagnetismo

Fig. 4 Initial pairing of bits for the scrambling algorithm



Se utiliza un algoritmo de optimización metaheurística basado en la teoría del electromagnetismo (EMO) para seleccionar la estructura óptima del BST:

- Las partículas cargadas en el campo electromagnético simulan soluciones candidatas.
- Las partículas con cargas similares se repelen y con cargas opuestas se atraen.
- Se exploran configuraciones de nodos en el árbol para maximizar la entropía y minimizar la vulnerabilidad del cifrado.

Implementación Técnica

Framework de cifrado implementado en un entorno de programación compatible con Python

Ideal:

01 Operaciones matemáticas complejas

02 Manejo de grandes volúmenes de datos

La optimización de la estructura del árbol y las operaciones sobre ADN se beneficiaron del uso de bibliotecas

01 NumPy y SciPy: Cálculos numéricos

02 Sklearn: Optimización metaheuristica

Ofrece:

- Flexibilidad para manipular cadenas
- Integrar técnicas de inteligencia artificial (optimización basada en electromagnetismo).

Resultados Experimentales y Análisis de Eficiencia

El artículo habla sobre cómo se probó un método para cifrar y descifrar imágenes, es decir, para protegerlas de forma segura sin perder información. Veamos los puntos clave:

01 ¿Qué se probó?

- Se realizaron experimentos para ver qué tan bien funcionaba este método tanto en seguridad como en eficiencia.
- Se usaron imágenes de diferentes tamaños y características (por ejemplo, diferentes tamaños de píxel y colores) para ver si el método funcionaba en distintos escenarios.

02 ¿Dónde se probó?

- Las pruebas se realizaron en un computador con un procesador Intel i5 y 8 GB de RAM.
- El resultado fue positivo: el método pudo cifrar y descifrar las imágenes sin perder ningún dato. Es decir, se podía recuperar la imagen original sin errores.

Seguridad del método

Uno de los aspectos más interesantes de este método es lo seguro que es contra posibles ataques.

01 ¿POR QUÉ ES TAN SEGURO?

- Se utiliza una estructura llamada árbol de búsqueda binaria (BST), que está optimizada para crear muchísimas configuraciones posibles.
- ¿Qué significa esto? Que es muy difícil para alguien adivinar o "romper" el código porque hay demasiadas opciones posibles.

02 ¿QUÉ LO HACE MÁS DIFÍCIL DE DESCIFRAR?

- Además del BST, este método usa técnicas de codificación de ADN. ¡Sí, ADN! Esto añade una capa extra de dificultad, haciendo que los atacantes necesiten saber tanto la estructura del BST como las reglas de codificación de ADN, algo extremadamente difícil de adivinar.

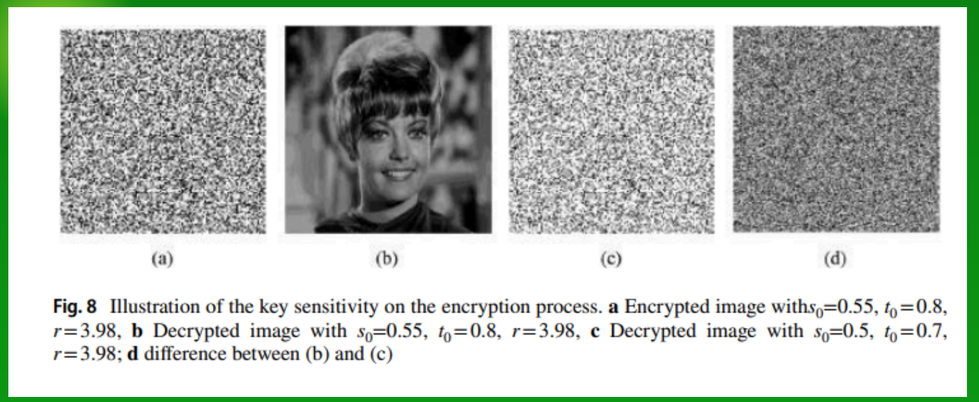


Fig. 8 Illustration of the key sensitivity on the encryption process. **a** Encrypted image with s_0=0.55, t_0=0.8, r=3.98, **b** Decrypted image with s_0=0.55, t_0=0.8, r=3.98, **c** Decrypted image with s_0=0.5, t_0=0.7, r=3.98; **d** difference between (b) and (c)

En resumen, el método no solo es rápido y eficiente para cifrar y descifrar imágenes, sino que también es muy seguro contra ataques tradicionales, gracias a la combinación del BST optimizado y la codificación de ADN.

Robustez Contra Ataques Diferenciales

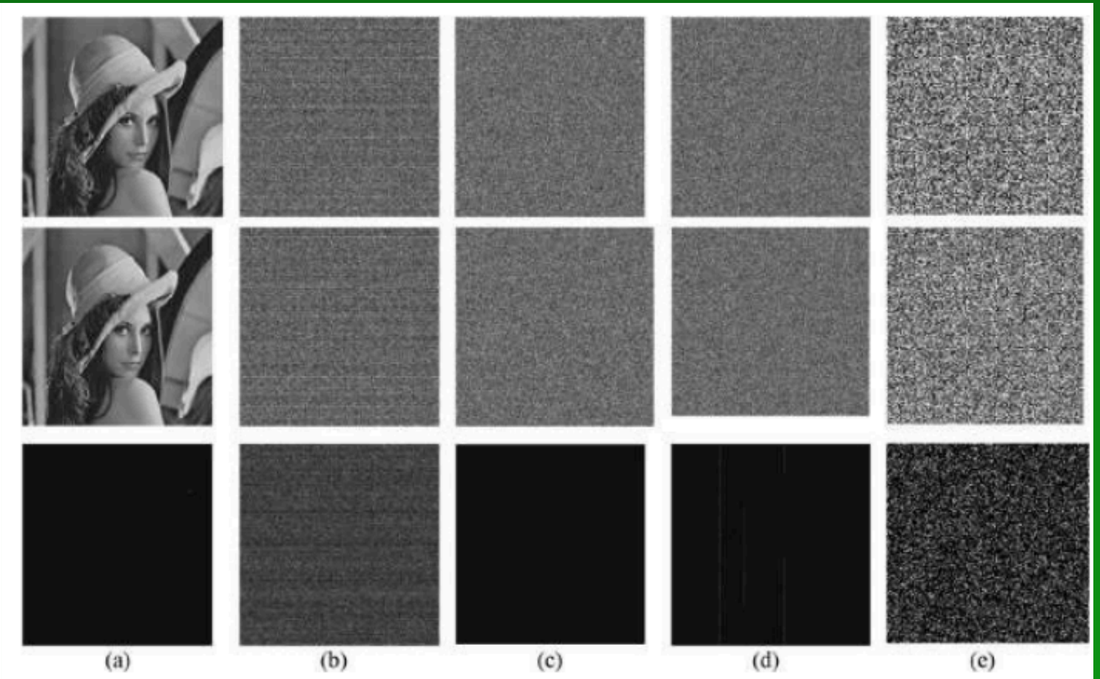


Fig. 11 Qualitative demonstration of the differential attack (**a**) from top to bottom: the original Lena test image and the same Lena image with a single bit difference, and the difference between these two images. (b)-(e) encrypted images and their difference that are generated by applying (**b**) DecomCrypt [55], (**c**) Zhou's approach [54], (**d**) Zhu's approach [56], (**e**) proposed approach

Esto demuestra la efectividad del método para resistir ataques basados en pequeñas modificaciones de la entrada

ANÁLISIS DE COMPLEJIDAD

Complejidad Temporal

- Tiempo de ejecución:
  ->Depende del tamaño del árbol de búsqueda binaria (BST) y el número de iteraciones de optimización.
  ->Complejidad temporal: O(nlogn), donde n es el número de píxeles de la imagen.
  ->Ideal para: Aplicaciones de gran escala, como cifrado de imágenes de alta definición.

Complejidad Espacial

- **Requiere almacenar:**
  -> Imagen original.
  -> Estructura del BST.
  -> Cadenas de ADN generadas en el proceso.
- **Optimización:**
  -> Uso de estructuras de datos eficientes en Python.
  ->Resultado: Mantiene la complejidad espacial en niveles aceptables.

CONCLUSIONES Y FUTURAS APLICACIONES

Cifrado de Imágenes

- Propuesta basada en árboles de búsqueda binarios y codificación de ADN.
- Optimización mediante metaheurística de electromagnetismo.

Aplicaciones

- Eficaz para entornos sensibles como la comunicación militar.
- Nuevas posibilidades en:
  -> Cifrado de video.
  -> Protección de datos biométricos.

Futuras Investigaciones

- Adaptación para hardware especializado (e.g., FPGA) para mejorar velocidad y eficiencia.