Operación XOR en dos Renyis Acoplados Y aplicación de pruebas NIST.

Marcos Daniel Calderón Calderón

Abstract

En este reporte se presentan a pruebas NIST de dos Mapas Renyi cuyos resultados son combinados por medio de una operación XOR.

1. Ejemplo 1.

1.1. Procedimiento.

EL mapa caótico Renyi tiene la siguiente forma:

$$f(k) = \left(q2^{n-i}k + \lfloor \frac{k}{2^j} \rfloor \right) \quad \text{mód } 2^n \tag{1}$$

para que el mapa sea invertible, es necesario que se cumpla la siguiente condición:

$$0 < i = j \leq n \tag{2}$$

por lo tanto, hemos elegido los siguientes parámetros:

- Mapa 1: $i = j = 9$.
- Mapa 2: $i = j = 13$.

Ahora, es necesario elegir un valor de q adecuado, en este ejemplo, utilizamos los siguientes:

- Mapa 1: $q = 13$.
- Mapa 2: $q = 19$.

Ahora, es necesario calcular el parámetro que se forma de la expresión:

$$\beta = q2^{n-i} \tag{3}$$

con base en lo anterior, se encontraron los siguientes parámetros para el mapa 1 y el mapa 2 respectivamente:

Email address: marcos.calderon@cimat.mx (Marcos Daniel Calderón Calderón)

- $\beta_1 = 109051904$.
- $\beta_2 = 9961472$.

También es importante elegir el tipo de dato que se va a utilizar para almacenar los valores generados por los mapas, en este caso se eligió el tipo de dato **unsigned long** (se utilizó un equipo con arquitectura de 32 bits, donde este tipo de dato tiene un tamaño de 32 bits).

Se generaron 80,000 valores a la hora de relacionar los valores de los dos mapas con la operación XOR. Como cada valor está formado de 32 bits, se obtiene un total de $2,560,000$ bits para la aplicación de pruebas NIST.

1.2. Resultados.

Se utilizó el siguiente código para la ejecución de las pruebas NIST:
Se ejecutó el siguiente código:

- **./assess 2560000**
- User Prescribed Input File: **dosRenyis.dat**
- Enter 0 if you DO NOT want to apply all of the statistical tests to each sequence and 1 if you DO. Enter chice: **1**
- How many bitstreams? **1**
- Input File Format: [0] ASCII - A sequence of ASCII 0's and 1's [1] Binary - Each byte in data file contains 8 bits of data

  Select input mode: **1**

LOs resultados obtenidos son los siguientes:

Cuadro 1: Resultados de las pruebas de aleatoriedad NIST a los datos dosRenyis.dat .

Prueba Aplicada	P-Valor	Exito?
Aproximate Entropy	0.751203	√
Block Frecuency	0.818897	√
Cumulative Sums	Forward test:0.557353, Reverse test: 0.508345	√
FFT	0.535638	√
Frecuency	0.412929	√
Linear Complexity	0.893675	√
Longest Run	0.864540	√
Non Overlapping Template	P-valores aceptados: 148 de 148	√
Overlapping Template	0.583516	√
Random Excursions	P-valores aceptados: 8 de 8	√
Random Excursions Variant	P-valores aceptados: 18 de 18	√
Rank	0.416165	√
Runs	0.052786	√
Serial	P-valores aceptados: 2 de 2	√
Universal	0.509933	√

2. Ejemplo 2.

2.1. Procedimiento.

Para el ejemplo 2, se eligieron los siguientes parámetros:

- Mapa 1: $i = j = 5$.
- Mapa 2: $i = j = 14$.

Ahora, es necesario elegir un valor de q adecuado, en este ejemplo, utilizamos los siguientes:

- Mapa 1: $q = 29$.
- Mapa 2: $q = 31$.

Ahora, es necesario calcular el parámetro que se forma de la expresión:

$$\beta = q2^{n-i} \tag{4}$$

con base en lo anterior, se encontraron los siguientes parámetros para el mapa 1 y el mapa 2 respectivamente:

- $\beta_1 = 3892314112$.
- $\beta_2 = 8126464$.

2.2. Resultados.

Se utilizó el siguiente código para la ejecución de las pruebas NIST:
Se ejecutó el siguiente código:

- **./assess 2560000**
- User Prescribed Input File: **dosRenyis2.dat**
- Enter 0 if you DO NOT want to apply all of the statistical tests to each sequence and 1 if you DO. Enter chice: **1**
- How many bitstreams? **1**
- Input File Format: [0] ASCII - A sequence of ASCII 0's and 1's [1] Binary - Each byte in data file contains 8 bits of data

  Select input mode: **1**

LOs resultados obtenidos son los siguientes:

Cuadro 2: Resultados de las pruebas de aleatoriedad NIST a los datos dosRenyis2.dat .

Prueba Aplicada	P-Valor	Exito?
Aproximate Entropy	0.427962	√
Block Frecuency	0.766879	√
Cumulative Sums	Forward test:0.368550, Reverse test: 0.541795	√
FFT	0.613759	√
Frecuency	0.298921	√
Linear Complexity	0.609575	√
Longest Run	0.080531	√
Non Overlapping Template	P-valores aceptados: 148 de 148	√
Overlapping Template	0.992376	√
Random Excursions	P-valores aceptados: 8 de 8	√
Random Excursions Variant	P-valores aceptados: 18 de 18	√
Rank	0.827165	√
Runs	0.260303	√
Serial	P-valores aceptados: 2 de 2	√
Universal	0.652881	√

3. Conclusiones.

Si en los mapas acoplados $j_1 = j_2$, puede ocurrir que no se aprueben las pruebas NIST, pero si ocurre que $j_1 \neq j_2$, los p-valores son muy buenos y las pruebas NIST son exitosas. Se obtienen buenos valores para las pruebas NIST si se eligen de manera decuada los parámetros.

4. Anexos.

4.1. Código del ejemplo 1.

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>


#define RENYI_MAP(var, parametro, j) ((var)*(parametro)+((var)>>(j)))
#define ITtotales 80000 /*Iteraciones totales para NIST.*/


/*
 File:   main.c
 Author: daniel
En este programa, se ejecutan de manera simultánea dos mapas de tipo Renyi,
 estos no
son acoplados, las salidas de estos mapas se utilizan para una
operacion XOR.
*/



unsigned long calcular_parametro(unsigned long q, unsigned int n, unsigned int i){
    unsigned long potencia =1;
    unsigned int j;
    /*Calculo de la potencia (calculamos 2(a la)( n-i) ): */
    for(j =0; j<(n-i); j++){
        potencia*=2;
    }

    /*Ahora, lo que hacemos es la operacion q*2(a la )(n-i): */
    unsigned long parametro = q*(potencia);
    printf("  EL parametro es: %lu \n \n \n",parametro);
    return parametro;
}

int main(){


   /*Declaramos los arreglos que vamos a utilizar para guardar esto.
    EL tipo de dato será
   unsigned long, cuyo tamaño es de 32 bits
```

```
 (arquitectura de 32 bits).*/

unsigned long Xtotal[ITtotales];
FILE *  archivobin;
unsigned long Xn1 = 7;
unsigned long Xn2 = 9;
unsigned int n = 32;

/*Valores de i, j y q para el mapa 1.*/
unsigned int i1=9;
unsigned int j1=9;
unsigned long q1=13;
/*Valores de i, j y q para el mapa 2.*/
unsigned int i2=13;
unsigned int j2=13;
unsigned long q2=19;

/*Calculo de los parametros para cada mapa: 1 y 2 respectivamente.*/
unsigned long param1 =calcular_parametro(q1, n, i1);
unsigned long param2 =calcular_parametro(q2, n, i2);

unsigned int iteraciones=0;
unsigned int IT = 80000;

/* Apertura del fichero de destino, para escritura en binario.*/
archivobin = fopen ("dosRenyis.dat", "wb");
if (archivobin==NULL)
{
perror("No se puede abrir dosRenyis.dat");
return -1;
}

printf("\n\n\n Operacion XOR en dos Renyis \n EL
tamanio de unsigned long en maquina de 32 bits es:  %d", sizeof(long));



while (iteraciones < IT) {

          Xn1= RENYI_MAP(Xn1,param1,j1);
          Xn2= RENYI_MAP(Xn2,param2,j2);
          Xtotal[iteraciones++] = Xn1^Xn2;
}
```

```
   /*Escribimos la informacion.*/
   fwrite(Xtotal,4,80000,archivobin);

   if(!fclose(archivobin)){
      printf( "\nArchivo binario cerrado\n" );
   }
   else{
      printf( "\nError: Archivo binario no cerrado \n" );
      return 1;
   }



return 0;
}
```

4.2. Código del criterio 2.

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>


#define RENYI_MAP(var, parametro, j) ((var)*(parametro)+((var)>>(j)))
#define ITtotales 80000 /*Iteraciones totales para NIST.*/


/*
 File:   main.c
 Author: daniel
En este programa, se ejecutan de manera simultánea dos mapas de tipo Renyi, estos no
son acoplados, las salidas de estos mapas se utilizan para una operacion XOR.
*/



unsigned long calcular_parametro(unsigned long q, unsigned int n, unsigned int i){
    unsigned long potencia =1;
    unsigned int j;
    /*Calculo de la potencia (calculamos 2(a la)( n-i) ): */
    for(j =0; j<(n-i); j++){
        potencia*=2;
```

```
    }

    /*Ahora, lo que hacemos es la operacion q*2(a la )(n-i): */
    unsigned long parametro = q*(potencia);
    printf("  EL parametro es: %lu \n \n \n",parametro);
    return parametro;
}

int main(){


   /*Declaramos los arreglos que vamos a utilizar para guardar esto
   . EL tipo de dato será
   unsigned long, cuyo tamaño es de 32 bits (arquitectura de 32 bits).*/
   unsigned long Xtotal[ITtotales];
   FILE *  archivobin;
   unsigned long Xn1 = 13;
   unsigned long Xn2 = 19;
   unsigned int n = 32;

   /*Valores de i, j y q para el mapa 1.*/
   unsigned int i1=5;
   unsigned int j1=5;
   unsigned long q1=29;
   /*Valores de i, j y q para el mapa 2.*/
   unsigned int i2=14;
   unsigned int j2=14;
   unsigned long q2=31;

   /*Calculo de los parametros para cada mapa: 1 y 2 respectivamente.*/
   unsigned long param1 =calcular_parametro(q1, n, i1);
   unsigned long param2 =calcular_parametro(q2, n, i2);

   unsigned int iteraciones=0;
   unsigned int IT = 80000;

  /* Apertura del fichero de destino, para escritura en binario.*/
   archivobin = fopen ("dosRenyis2.dat", "wb");
   if (archivobin==NULL)
   {
   perror("No se puede abrir dosRenyis2.dat");
   return -1;
   }
```

```
   printf("\n\n\n Operacion XOR en dos Renyis \n EL tamanio de
   unsigned long en maquina de 32 bits es:  %d", sizeof(long));



   while (iteraciones < IT) {

            Xn1= RENYI_MAP(Xn1,param1,j1);
            Xn2= RENYI_MAP(Xn2,param2,j2);
            Xtotal[iteraciones++] = Xn1^Xn2;

   }


   /*Escribimos la informacion.*/
   fwrite(Xtotal,4,80000,archivobin);

   if(!fclose(archivobin)){
      printf( "\nArchivo binario cerrado\n" );
   }
   else{
      printf( "\nError: Archivo binario no cerrado \n" );
      return 1;
   }



return 0;
}
```