



Ingeniería en Seguridad Informática y Redes

Ingeniería en Sistemas Computacionales

Algoritmos de Solución Numérica

Proyecto final

IS727272 - Marco Ricardo Cordero Hernández

SI727576 - Edgar Guzmán Claustro

Jal., 24 de noviembre de 2023

## Índice

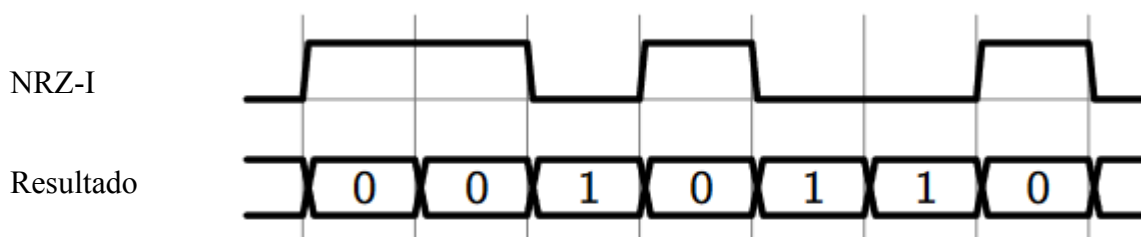
Introducción.....	1
Fundamento.....	2
Desarrollo.....	2
Función de generación para NRZ.....	3
Función auxiliar para conversión a binario.....	3
Manejador y gráfica.....	4
Resultados.....	5
Primera ejecución.....	5
Segunda ejecución.....	5
Tercera ejecución.....	6
Cuarta ejecución.....	6
Conclusiones.....	7
Bibliografía.....	8

## Introducción

El mundo moderno necesita estar en constante comunicación, es un hecho innegable que incluso ha creado carreras y profesiones nuevas, existiendo todo un campo de tecnologías de la información que atienden a este propósito. Las cosas no son tan sencillas como antes, en donde alguno de los métodos más primitivos consistía en lanzar una paloma con un papiro atado a la misma, y después realizar una oración refiriendo a la deidad divina de preferencia. Esto, al resultar extremadamente ineficiente y evidentemente voluble, obligó a las cosas a cambiar para prescindir de métodos aviares o de señales de humo, aunque quizás sí hubo humo en las pruebas que se realizaron para llevar a cabo la siguiente etapa de la humanidad.

Con la llegada de las telecomunicaciones y las computadoras en siglos pasados, muchos de los problemas de la electrónica también comenzaban a introducirse de manera informal, tales como la inducción y la interferencia, lo que dificultaría el progreso de los nuevos métodos emergentes. A pesar de estas adversidades, los pioneros de antaño comenzaron a idear sistemas de transmisión para dar pie a lo que llegaría a convertirse en muchas de las maravillas tecnológicas contemporáneas. Entre estos sistemas básicos se encuentra el **código NRZ**.

Non-Return-to-Zero, mejor conocido como NRZ por sus iniciales, es un método análogo de transmisión de bits, el cual representa la transmisión de bits (dígito binario con posibles valores [0, 1]) a través de un conjunto de condiciones suficientes, usualmente un pulso que ocupa todo un periodo  $T_b$  (Spolaor et al, 2016). La forma en que se representan los datos es mediante una transmisión continua en donde un voltaje positivo puede representar un 1 (NRZ-L) o un 0 (NRZ-I), y su contraparte negativa representa el opuesto (RF Wireless World, 2012).



La principal ventaja de este tipo de codificación reside en la facilidad con la que puede ser implementado, requiriendo tan solo unos cuantos parámetros para su demostración. Por el gran significado histórico con el que cuenta y el hecho de que visualización puede conseguirse en poco tiempo (lo cual se valora inmensamente estos días), el proyecto actual tiene el objetivo de realizar una implementación integral y parametrizable de la codificación NRZ. Esta decisión también está soportada por el hecho de que los integrantes del presente pertenecen a campos de la ingeniería relacionados en gran medida con el manejo de la información.

## Fundamento

Al pertenecer mayoritariamente al área de las telecomunicaciones, la aplicación del código NRZ requiere de algunos parámetros para realizar una correcta demostración. Una lista de los más básicos de estos luce como la siguiente:

- Tasa de transmisión: también encontrada como velocidad de los datos, esta métrica define cuántas unidades son transmitidas por segundo, usualmente con unidades de bits por segundos (bps).
- Número de puntos: también conocida como el número de muestras, es la cantidad que define la “resolución” de cada bit del dato a transmitir, es decir, a mayor número de muestras, mejor fidelidad de la transmisión.
- Amplitud de señal: formalmente definida como la distancia entre el punto más alejado de una onda y el punto de equilibrio o medio, este componente define el rango de operación de la señal, por ejemplo, para un sistema que opera a 5 voltios, la amplitud oscilaría entre 0 y 5.
- Duración de señal: aunque puede resultar la variable más compleja, este componente simplemente indica el tiempo de transmisión sobre el cual se produce la señal.

Con estos elementos conocidos, es posible recrear una simulación fidedigna por medio de herramientas de cómputo, algo de lo cual se hablará en la siguiente sección.

## Desarrollo

Como era de esperar y del mismo modo en que se ha venido manejando a lo largo del curso, la herramienta predilecta para el desarrollo y demostración formal será Matlab (The MathWorks Inc., 2023), ya que su versatilidad y su sistema de indexación basado en 1 seguramente propiciarán la facilidad para llevar a cabo el objetivo planteado.

### Función de generación para NRZ

```
function nrz = NRZ(D, T, P, V, A, N)
%{
    D = Cadena binaria de datos para la señal
    T = Duración de la señal
    P = Puntos de muestra
    V = Velocidad de los datos
    A = Amplitud de la señal
    N = Agregar ruido a la señal (booleano)
%}

% Variables locales
Tb = 1 / V;           % Tiempo de duración del bit
PT = Tb / P;          % Paso de tiempo
t = 0:PT:T;           % Segmentación del tiempo de la señal
NP = length(t);       % Total de puntos en la señal
nrz = zeros(1, NP);   % Señal base

% Ajuste de la señal
for n = 0:length(D)-1
    if D(n+1) == '1' || D(n+1) == 1
        nrz(P*n+1:P*n+P) = 1;
    end
end

nrz = nrz .* A;

% Agregar ruido a la señal
if (N == true)
    SNR = floor(rand()*10);
    nrz = awgn(nrz, SNR, 'measured');
end
```

### Función auxiliar para conversión a binario

```
function str = str2bin(s)
%{
    s = Cadena de entrada
%}

holder = reshape(dec2bin(s, 8).-'0',1,[]); % Convertir a binario
str = '';                                % Cadena contenedora

% Regresar arreglo de binarios en cadena
```

```

for c = holder
    str = strcat(str, int2str(c));
end

```

## Manejador y gráfica

```

% Restablecer entorno
clear, clc, close all

% Definición de valores de entrada
str = input('Ingresa el dato a transmitir -> ', 's');
binstr = str2bin(str); % Convertir entrada a binario
T = input('Ingresa la duración de la señal (s) -> ');
P = input('Ingresa puntos de muestra -> ');
V = input('Ingresa la velocidad de los datos (bps) -> ');
A = input('Ingresa amplitud de la señal -> ');
N = upper(input('¿Deseas agregar ruido a la señal? -> ', 's'));
t = 0:1/(V*P):T; % Representación gráfica de la duración
valid_options = {'Y' 'S' 'TRUE' '1'};

% Verificar variable de ruido
for o = 1:length(valid_options)
    if (N == valid_options{o})

        if (N == o)
            N = true; break
        end
    end
end

if (N ~= true)
    N = false;
end

% Llamada a función
nrz = NRZ(binstr, T, P, V, A, N);

% Gráfica
figure('name', 'Demostración de NRZ', 'NumberTitle','off');
plot(t, nrz);
xlim([0 0.1]);
title(strcat('Transmisión de texto: ', str));
xlabel(binstr);

```

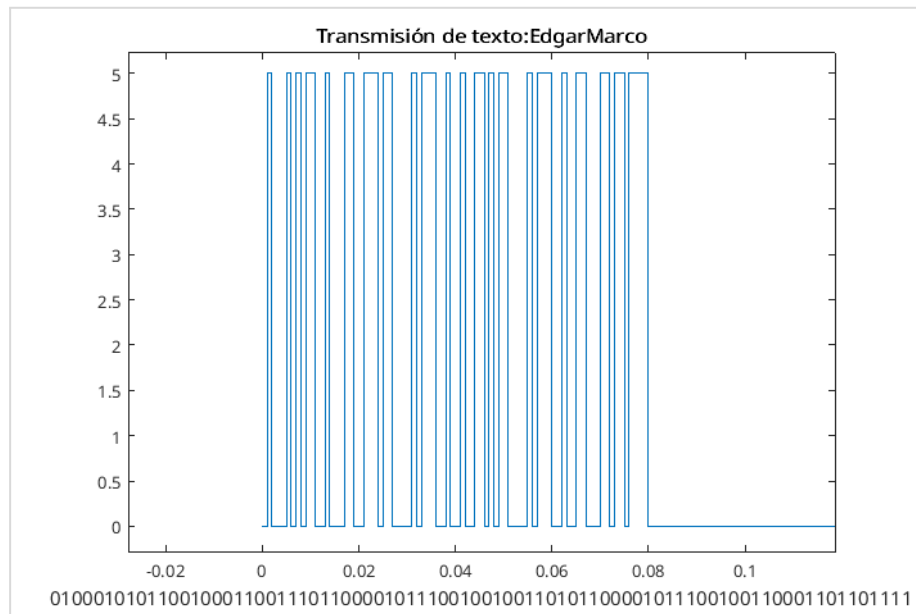
Los comentarios y la ejecución del programa deberían ser lo suficientemente descriptivos para establecer el funcionamiento de la simulación, sin embargo, como ya se comentaba en la sección anterior, se requiere el conocimiento de varios datos previo a la ejecución, los cuales son solicitados al momento de correr el código.

La siguiente sección muestra algunas corridas junto con su resultado gráfico.

## Resultados

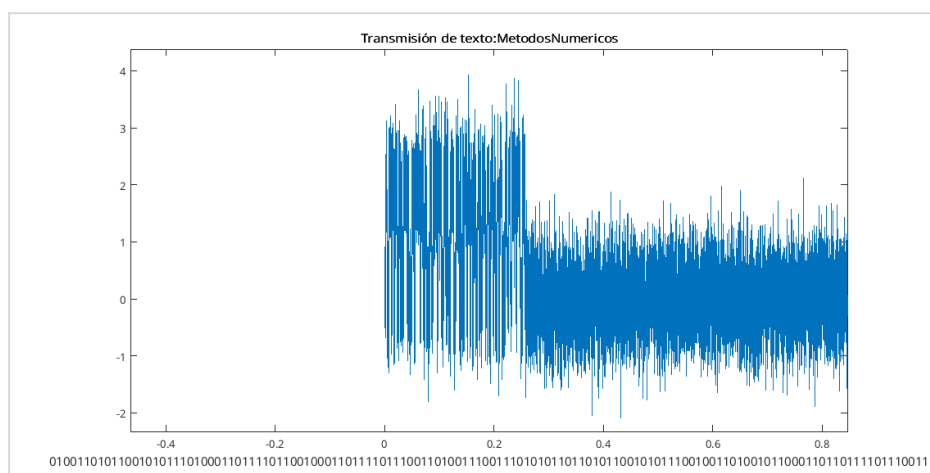
### Primera ejecución

```
Ingresa el dato a transmitir -> EdgarMarco
Ingresa la duración de la señal (s) -> 2
Ingresa puntos de muestra -> 100
Ingresa la velocidad de los datos (bps) -> 1000
Ingresa amplitud de la señal -> 5
¿Deseas agregar ruido a la señal? -> N
```



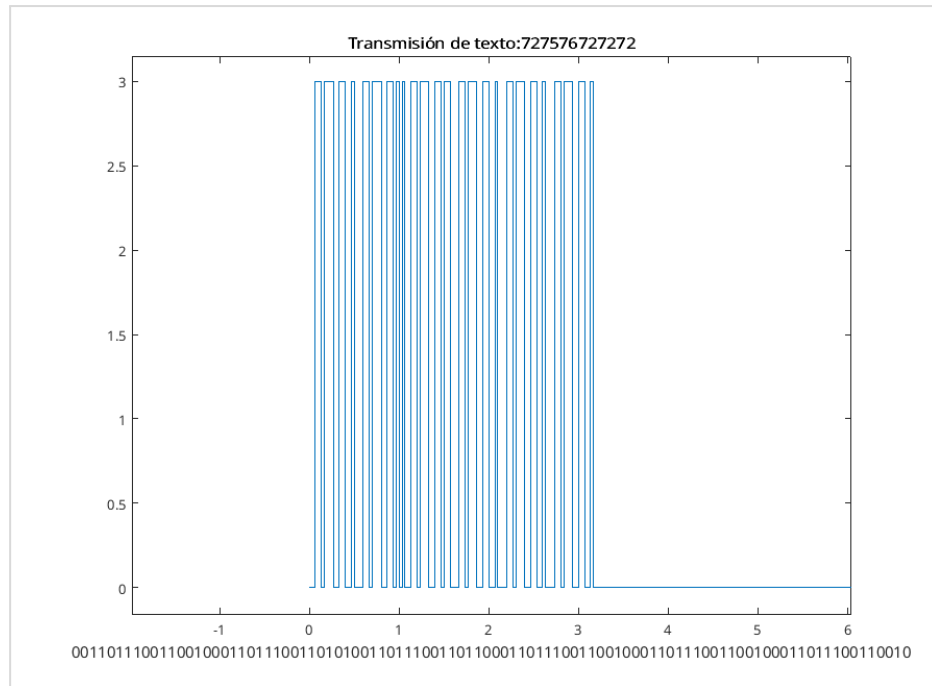
### Segunda ejecución

```
Ingresa el dato a transmitir -> MetodosNumericos
Ingresa la duración de la señal (s) -> 1.234
Ingresa puntos de muestra -> 30
Ingresa la velocidad de los datos (bps) -> 500
Ingresa amplitud de la señal -> 2
¿Deseas agregar ruido a la señal? -> S
```



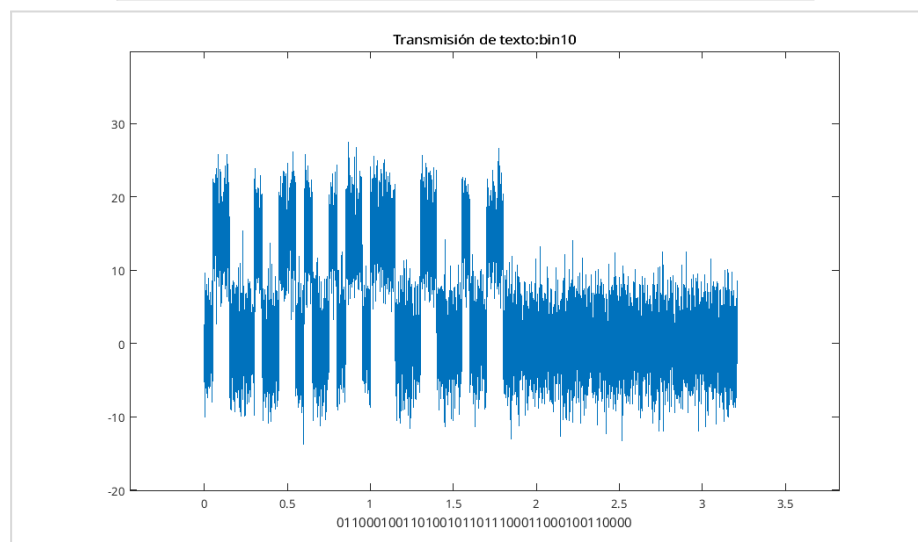
### Tercera ejecución

```
Ingresa el dato a transmitir -> 727576727272
Ingresa la duración de la señal (s) -> 30
Ingresa puntos de muestra -> 2842
Ingresa la velocidad de los datos (bps) -> 30
Ingresa amplitud de la señal -> 3
¿Deseas agregar ruido a la señal? -> N
```



### Cuarta ejecución

```
Ingresa el dato a transmitir -> bin10
Ingresa la duración de la señal (s) -> 3.21
Ingresa puntos de muestra -> 200
Ingresa la velocidad de los datos (bps) -> 20
Ingresa amplitud de la señal -> 15
¿Deseas agregar ruido a la señal? -> true
```





## Conclusiones

Guzmán Claustro, Edgar

Dentro de este proyecto se pudo apreciar un ejemplo de una aplicación “con sentido” dentro del ámbito de la ingeniería, donde se demostró que se pueden utilizar dichos conocimientos para generar automatizaciones funcionales. Como “casi” Ingeniero en Seguridad Informática y Redes, me fue de gran placer realizar este proyecto, tomando conocimientos de diversas materias para concebirlo. El lenguaje utilizado fue Matlab y sin duda alguna, puedo decir que nunca se puede acabar de conocer algo o convertirse en experto. A pesar que el manejo del lenguaje es “sencillo” encaramos diversos retos durante la realización de esto.

No creo volver a tocar el lenguaje durante un largo tiempo, a menos que el destino me lo ponga otra vez. Ahora tengo un conocimiento más amplio y sinceramente creo que me será de provecho recordar todas estas implementaciones creadas. Fue un curso agradable y aunque a veces sentía que íbamos “lento”, en otras ocasiones no lo era así.

Cordero Hernández, Marco R.

Todo lo bueno tiene que acabar, sin embargo, en esta ocasión, lo que llega a su fin es el uso de Matlab, al menos por el momento. La utilización de este software deja muchas enseñanzas, aunque aún queda en tela de juicio su calidad, ya que el lenguaje que utiliza es bastante esotérico.

Dejando de lado la tan extenuante parte técnica, llegar al final del curso con un desarrollo como el actual deja más dudas que respuestas de las que existían al inicio, ya que a pesar de los malos momentos pasados en este planteamiento o en el uso general del lenguaje, el sentimiento de deseo por querer conocer más prevalece por encima de todas las cosas. Al tomar esta materia como curso complementario no se tienen expectativas mayúsculas más que las de esperar contenido fructíferos y la menor cantidad posible de tareas; aunque la materia sólo satisfizo uno de los mencionados, lo hizo de manera espectacular, puesto que la conjunción de las matemáticas y la computación siempre será algo interesante de analizar, al final, la informática es solo una rama más de las matemáticas aplicadas.

Realizar este proyecto logró lo anterior y además pudo presentar una aplicación adicional de la ingeniería llevada a la realidad, y aunque se haya hecho de forma rudimentaria y quizá un poco burda, la enseñanza es la que queda y persistirá mucho más allá de los contenidos de este documento e inclusive de la carrera dentro de la cual se tiene la oportunidad de presentar este proyecto.

Gracias.

## Bibliografía

Spolaor, R., Abudahi, L., Moonsamy, V., et al. (2016). *No Free Charge Theorem: a Covert Channel via USB Charging Cable on Mobile Devices*. Recuperado de <https://www.researchgate.net/publication/307984756>.

RF Wireless World. (2012). *Advantages of NRZ line coding | Disadvantages of NRZ encoding*. Recuperado de <https://www.rfwireless-world.com/Terminology/Advantages-and-Disadvantages-of-NRZ-encoding.html>.

The MathWorks Inc. (2023). MATLAB version: 9.14.0 (R2023a), Natick, Massachusetts: The MathWorks Inc. <https://www.mathworks.com>.