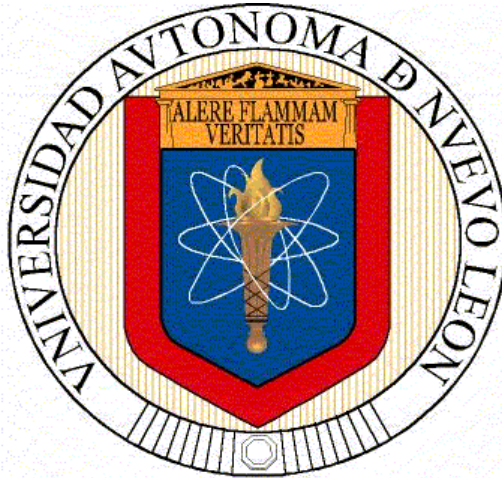


**UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN  
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA**



**TESIS**

**“MODELO DE CANAL Y FUENTE DE INFORMACIÓN PARA UN  
SISTEMA DE CONTROL VEHICULAR QUE UTILIZARÁ  
TECNOLOGÍA RFID.”**

**POR**

**CARLOS ALBERTO ONOFRE BARRÓN**

**COMO REQUISITO PARCIAL PARA OBTENER EL GRADO DE  
MAESTRÍA EN CIENCIAS DE LA INGENIERÍA ELÉCTRICA CON  
ORIENTACIÓN EN CONTROL AUTOMÁTICO.**

**JULIO 2014**

**UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN  
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA**



**TESIS**

**“MODELO DE CANAL Y FUENTE DE INFORMACIÓN PARA UN  
SISTEMA DE CONTROL VEHICULAR QUE UTILIZARÁ  
TECNOLOGÍA RFID.”**

**POR**

**CARLOS ALBERTO ONOFRE BARRÓN**

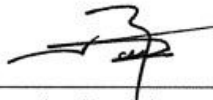
**COMO REQUISITO PARCIAL PARA OBTENER EL GRADO DE  
MAESTRÍA EN CIENCIAS DE LA INGENIERÍA ELÉCTRICA CON  
ORIENTACIÓN EN CONTROL AUTOMÁTICO.**

**JULIO 2014**

Universidad Autónoma de Nuevo León  
Facultad de Ingeniería Mecánica y Eléctrica  
Subdirección de estudios de posgrado

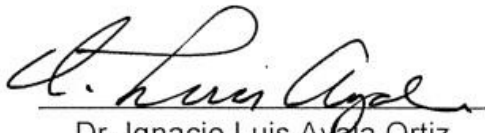
Los miembros del Comité de Tesis recomendamos que la Tesis " Modelo de canal y fuente de información para un Sistema de control vehicular que utilizará tecnología RFID", realizada por el alumno Carlos Alberto Onofre Barrón con número de matrícula 1338614, sea aceptada para su defensa como opción al Control Automático.

El comité de Tesis



---

Dr. José Ramón Rodríguez Cruz  
Asesor



---

Dr. Ignacio Luis Ayala Ortiz  
Revisor

---

Dr. Neale Ricardo Smith Cornejo  
Revisor

Vo. Bo.

---

Dr. Simón Martínez Martínez  
Subdirector de Estudios de Posgrado

San Nicolás de los Garza, Nuevo León, Julio 2014

## **Agradecimientos**

Agradezco a Dios por brindarme la oportunidad de estudiar una Maestría y permitirme terminar satisfactoriamente esta nueva etapa de mi vida.

A FIME y CONACyT por la oportunidad y el apoyo que me brindaron.

A mi asesor el Dr. Ramón por sus consejos y su guía a lo largo de este trabajo de tesis.

A mis padres y mi hermana por su apoyo incondicional en todo momento de mi vida.

A mis compañeros y amigos a quienes tuve el gusto de conocer a lo largo de éste camino y de los cuales me llevo grandes recuerdos, alegrías y enseñanzas.

## **Dedicatoria**

Dedico este trabajo especialmente a mis padres el Sr. José Benito Onofre Pérez y la Sra. Beatriz Barrón Nájera quienes han sido el pilar más importante, el factor común en todos los logros a lo largo de mi vida, y sin quienes no sería la persona que soy actualmente. Y a mi Dios, por llenarme de bendiciones y darme la dicha de tener una familia que me apoya y unos padres maravillosos que me motivan a superarme cada día y a ser una mejor persona.

# Índice de Contenido

## Contenido

1. INTRODUCCIÓN.....	1
1.1 Identificación por Radiofrecuencia.....	1
1.1.1 RFID en control vehicular.....	3
1.1.2 Estándar EPCglobal Class 1 Generation 2. ....	4
1.1.3 Organización de la memoria de la etiqueta Gen 2. ....	6
1.2 Número de Identificación Vehicular (NIV).....	8
1.3 Estructura del Trabajo.....	8
1.4 Contribuciones.....	10
1.4.1 Modelo de la fuente de información. ....	11
1.4.2 Entropía.....	11
1.4.3 Simulador de la fuente. ....	11
1.4.4 Codificación y compresión. ....	11
1.4.5 Detección y Corrección de errores.....	11
2. ANTECEDENTES.....	13
2.1 Antecedentes.....	13
2.1.1 Codificación conjunta canal-fuente.....	13
2.1.2 Codificación binaria de un NIV. ....	14
2.1.3 Análisis Teórico de un sistema RFID. ....	14
3. PLANTEAMIENTO DEL PROBLEMA.....	18
3.1 Planteamiento del problema. ....	18
3.1.1 Acotamiento del problema.....	19

3.2	Objetivos.....	20
4.	DEFINICIÓN DE LA FUENTE INFORMACIÓN.....	21
4.1	Fuente de información.....	21
4.1.1	Estructura del NIV.....	22
4.2	Almacenamiento de la información.....	24
4.3	Obtención de datos de la fuente.....	27
4.4	Clasificación de los datos.....	28
4.4.1	NIVs que cumplan con el estándar actual.....	28
4.4.2	Autos fabricados en Territorio Nacional.....	29
4.4.3	Autos circulando en el estado de Nuevo León.....	29
4.4.4	Autos con modelos del 2003-2013.....	30
5.	MODELOS PROPUESTOS.....	31
5.1	Modelo 1.....	31
5.1.1	Modelo probabilístico.....	31
5.1.2	Resultados del análisis Modelo 1.....	33
5.1.3	Cálculo de probabilidad por NIV.....	37
5.2	Modelo 2.....	38
5.2.1	Modelo de Markov.....	39
5.2.2	Resultados del análisis.....	40
5.2.3	Cálculo de probabilidad.....	53
5.2.4	Árbol de probabilidades.....	54
5.2.5	Entropía.....	57
5.2.6	Simulador de datos.....	60
5.2.7	Obtención de la Función de Masa de Probabilidad (PMF).....	63
6.	CODIFICACIÓN Y COMPRESIÓN.....	72

6.1	Métodos de Codificación y compresión. ....	72
6.1.1	ASCII.....	72
6.1.2	Método de codificación binaria de un NIV. ....	73
6.1.3	Métodos de compresión para texto. ....	77
6.1.4	Método de codificación binaria con la NOM-001-SSP-2008. ....	86
6.1.5	Método de codificación binaria con Modelo 2. ....	87
6.1.6	Método de codificación binaria con Huffman.....	92
6.1.7	Codificación Binaria Adaptada con Huffman. ....	98
6.2	Codificación y decodificación de la información.....	100
6.2.1	Codificador. ....	101
6.2.2	Decodificador. ....	102
6.3	Pruebas en Canal BSC.....	103
6.3.1	Canal BSC .....	103
6.3.2	Metodología .....	104
6.3.3	Detección y Corrección de errores (Grupo 1).....	107
6.3.4	Pruebas.....	112
7.	RESULTADOS. ....	116
7.1	Resultados.....	116
7.1.1	Modelo Matemático. ....	116
7.1.2	Programa Computacional.....	117
7.1.3	Función de masa de probabilidad. ....	117
7.1.4	Codificación y compresión. ....	117
7.1.5	Detección y corrección de errores.....	118
8.	CONCLUSIONES.....	119
8.1	Conclusiones .....	119



8.1.1	Modelo de la fuente.....	119
8.1.2	Codificación y compresión. ....	119
8.1.3	Robustez del método de codificación propuesto.....	120
8.1.4	Corrección de errores.....	120
9.	REFERENCIAS BIBLIOGRÁFICAS. ....	122
	Índice de Figuras .....	124
	Índice de TABLAS.....	126

## Capítulo 1

# 1. INTRODUCCIÓN.

### 1.1 Identificación por Radiofrecuencia.

La tecnología RFID (Identificación por radiofrecuencia) tiene como objetivo primordial la identificación de objetos de una manera rápida sin la necesidad de que exista línea de vista entre el lector y la etiqueta. Los elementos básicos de esta tecnología son un transmisor, al cual se le referirá como "lectora", y un receptor, al cual se le referirá "la etiqueta". El lector RFID es un transceptor: un transmisor y receptor que trabajan juntos para llevar a cabo la comunicación con la etiqueta. Estos elementos se comunican mediante ondas electromagnéticas cuyas frecuencias que pueden ir desde los 100 KHz hasta los 5 GHz dependiendo de la tecnología de los elementos y la aplicación en la que se vaya a implementar el sistema. Las bandas de frecuencia más usuales para el uso de la tecnología RFID son las de 125/134 KHz (LF), 13.56 MHz (HF), 860-960 MHz y 2.4-2.45 GHz (UHF) tal y como se puede observar en la Figura 1.1.1 tomada de [1].

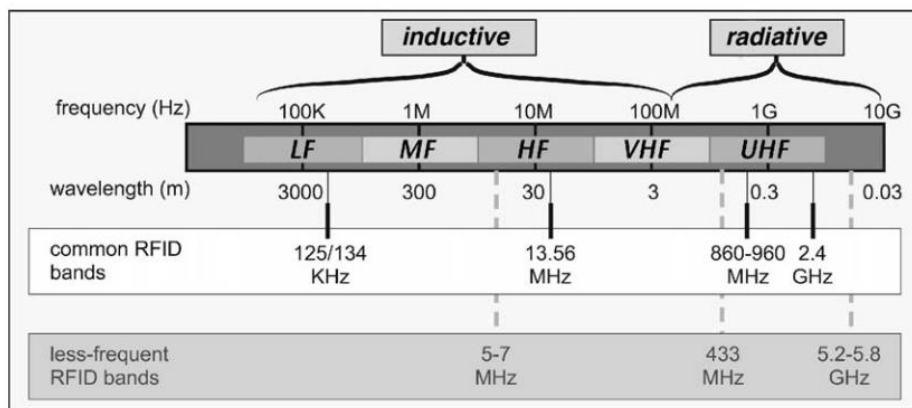
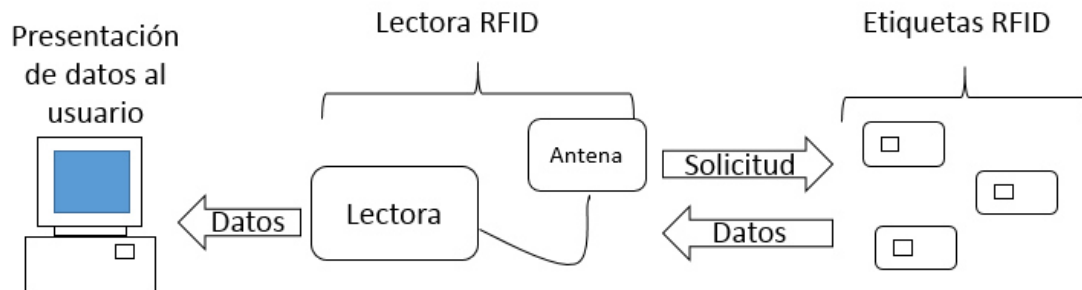


Figura 1.1.1 Bandas de frecuencia de RFID. (Referencia tomada de [1])

Las etiquetas RFID cuentan con un circuito integrado, el cual a su vez tiene bancos de memoria en los cuales se puede guardar información. Ésta información, por lo general, está relacionada al artículo al que la etiqueta esta adherido. Así mismo la información en los bancos de memoria es la que se busca transmitir a las lectoras cuando éstas lo requieran. Existen diferentes tipos de etiquetas de acuerdo a si cuentan o no con batería propia para alimentar su circuito; las pasivas, semi-pasivas y activas. Las etiquetas pasivas no cuentan con una batería para alimentar su circuito integrado, sino que utilizan la energía que reciben por medio de las ondas de radiofrecuencia y la transforman para poder utilizarla. Con ésta energía que reciben operan el circuito lógico y responden a la lectora con otra señal de RF. Por otro lado, las etiquetas activas cuentan con una batería para alimentar su circuito integrado y responder a la lectora. Una ventaja de éstas últimas con respecto a las pasivas, es que pueden almacenar una cantidad más grande de información en sus bancos de memoria, y una desventaja es que al trabajar con una batería de vida limitada, también se limita la vida útil de la etiqueta. Por último las etiquetas semi-pasivas, también conocidas como etiquetas pasivas asistidas por batería, poseen una batería local para abastecer la circuitería de la etiqueta, pero siguen usando la energía recibida para las comunicaciones etiqueta a lectora, es decir, usan parte de la energía que reciben para responder más no para alimentar su circuito.

En la Figura 1.1.2 se muestran de una forma gráfica los elementos de un sistema RFID. En la parte de presentación al usuario final, se hace referencia a que es necesario un software para traducir los datos recibidos por la etiqueta a un lenguaje, formato o presentación adecuada para el usuario final, que es quién está interesado en la información contenida en las etiquetas. La parte de la lectora RFID comprende a la lectora, que es un transceptor (al igual que las etiquetas) debido a que envía y recibe información. Dicha lectora por lo general ya tiene una antena integrada y en algunos casos tiene la opción de agregarle más antenas para poder elegir una con el patrón de radiación apropiado según sea el tipo de aplicación. Por último en la imagen se muestran las etiquetas RFID,

que es donde se encuentra guardada la información a la que el usuario final desea acceder. Así mismo se puede agregar nueva información a una etiqueta o modificar la información escrita en ellas, ya que algunas etiquetas tienen la capacidad de ser editables. Para la edición de la información (borrar/escribir) que está almacenada en una etiqueta con esta característica se necesita contar con la contraseña de escritura correcta.



*Figura 1.1.2 Elementos de un sistema RFID.*

El proceso ilustrado en la Figura 1.1.2 básicamente consiste en: (1) el usuario elige los parámetros o la información que desea obtener, (2) la aplicación le manda los comandos a la lectora RFID, (3) ésta a su vez establece una comunicación con la etiqueta y una vez entablado el enlace de comunicación solicita la información que requiere el usuario, (4) la etiqueta localiza en sus bancos de memoria la información que le fue solicitada, (5) la envía de regreso a la lectora que ahora funciona como receptora y (6) ésta envía los datos a la aplicación para que sean mostradas en el formato correcto al usuario final.

### **1.1.1 RFID en control vehicular.**

El área de investigación de la tecnología RFID es muy extensa, debido a las características de la misma ha sido ampliamente aceptada por las empresas en los procesos de la cadena de suministro, debido a que permite trabajar con una gran cantidad de datos de manera muy eficaz y a grandes velocidades. Características que han llevado a esta tecnología a considerarse la sucesora del código de barras.

Aunque el ámbito relacionado a la cadena de suministro es una de las áreas con más aceptación de la tecnología RFID, no es la única donde se emplea dicha tecnología. Ya que se han encontrado trabajos en los que se emplea tecnología RFID para aplicaciones relacionadas con el control de vehículos.

En la identificación vehicular implementando tecnología RFID por lo general se etiqueta al vehículo con una etiqueta RFID en una parte accesible para las lectoras, por ejemplo, en el parabrisas. De esta forma al entrar al área de cobertura de alguna de las lectoras que se encuentran cerca, estas pueden obtener la información contenida en las etiquetas. Dicha información por lo general está relacionada o contiene información ligada al vehículo. De esta forma las lectoras recopilan y utilizan esta información para el uso de la aplicación en cuestión, que puede ir desde llevar la bitácora de los automóviles que accedan a cierta área, hasta llevar a cabo tareas más complejas como la detección de embotellamientos y propuesta de rutas alternas.

Actualmente se han publicado una gran cantidad de trabajos en los cuales podemos encontrar; sistemas inteligentes de estacionamiento, los cuales optimizan los registros de entradas y salidas de autos, así como los cargos que se cobran por el uso de este servicio [2]. Sistemas para la prevención de accidentes con peatones, dichos sistemas funcionan en conjunto con sensores para avisar a los conductores sobre la presencia de peatones [3]. También se encuentran trabajos de monitoreo del tráfico en las ciudades, ayudando a detectar congestionamientos y proponiendo a los usuarios nuevas rutas para evitarlos [4]. Existen muchas otras aplicaciones que se pueden encontrar en el estado del arte y que están relacionadas con tecnología RFID y vehículos.

### **1.1.2 Estándar EPCglobal Class 1 Generation 2.**

Los distintos estándares utilizados en la tecnología RFID los podemos encontrar en [1] en donde se puede encontrar a detalle una explicación del funcionamiento y características de los distintos estándares de la tecnología RFID. Pero para el

desarrollo de este trabajo se hace mención de algunas características del estándar “EPCglobal Class 1 Generation 2” debido a que es el estándar con el cual trabaja la tecnología RFID que se pretende utilizar para los fines de esta tesis.

Como se menciona en el párrafo anterior para el desarrollo del trabajo usaremos el estándar de la Generación 2 (Gen 2), el cuál es diferente en muchos aspectos a los estándares de la primera generación, ya que incluye algunas mejoras de gran importancia en comparación con sus antecesores, entre las mejoras más importantes tenemos:

- ✓ Etiquetas con velocidades flexibles de comunicación de datos.
- ✓ Control del espectro de transmisión de lectoras y etiquetas para minimizar interferencia.
- ✓ Bits de control de protocolo separados, con declaración explícita de la longitud del EPC.
- ✓ Usan un esquema adaptable basado en el protocolo Aloha para la resolución de colisiones con un número de espacios fácilmente variable (el protocolo Q)
- ✓ Sesiones lógicas basadas en números aleatorios que permiten la singularización en la presencia de EPCs idénticos o ausencia de los mismos.
- ✓ Múltiples banderas persistentes que permiten inventarios quasi-simultáneos para diferentes lectoras.
- ✓ Comandos de longitud variable para la mejora de velocidad del inventario.
- ✓ Especificaciones explícitas de los mapas de memoria, disposiciones de candado y candado permanente, y procedimientos de programación.
- ✓ Codificación en el enlace para una programación segura de las etiquetas.
- ✓ Procedimiento de prueba de conformidad e interoperabilidad definido por EPCglobal.

Para información más detallada de los distintos estándares con los que trabaja RFID, así como del funcionamiento de los mismos se recomienda consultar [1]

en el capítulo 8 donde se explica los distintos protocolos con los que funciona la tecnología RFID. Debido a que de acuerdo a los alcances de este trabajo no se considera necesario entrar a fondo en el tema de la estandarización de la tecnología ya que dicha información no impacta en los resultados que se buscan obtener al final del trabajo.

### **1.1.3 Organización de la memoria de la etiqueta Gen 2.**

Las etiquetas Gen 2 cuentan con campos editables en los que se puede escribir, es decir, guardar información. Además, los campos individuales pueden ser bloqueados contra escritura (que es lo más usual) y en algunos casos también se pueden bloquear contra lectura. Otra característica importante en las etiquetas de Generación 2 es que se les puede dar la una instrucción llamada “kill” con el fin de dejar inutilizable la etiqueta, tanto para escritura como para lectura. Cabe mencionar que al hacer esto la etiqueta ya no podrá ser utilizada en un futuro y tendrá que ser desechada. A primera instancia se pudiera pensar que esto sugiere cierta vulnerabilidad por parte de las etiquetas. Pero la realidad es que las operaciones de “lock” (para bloquear ya se lectura o escritura) y “kill” (para dejar inutilizable la etiqueta), están protegidas por una contraseña de 32 bits, la cual es necesaria para poder llevar a cabo dichas instrucciones.

Otra característica importante de la Generación 2 es que especifica la organización de la memoria en la etiqueta, dicha organización está compuesta por dos campos obligatorios y dos campos opcionales. A continuación se presenta una breve descripción de cada uno de estos 4 bancos de información.

#### **Banco 00 (Reservado)**

El banco 0 es un banco de memoria reservado. Aquí se almacenan por lo menos las contraseñas de KILL y ACCESS para la etiqueta.

**Banco 01 (EPC)**

El banco 1 está destinado al EPC (Electronic Product Code) de la etiqueta. Almacena la palabra de control de protocolo (PC) de 32 bits la cual describe la longitud del EPC. Sirve para almacenar información opcional sobre la etiqueta. Contiene el CRC16 usado para revisión de errores del valor del EPC. El CRC es calculado por la etiqueta y no necesita ser escrito en la memoria por el lector RFID.

**Banco 10 (ID de la etiqueta)**

El banco 2 contiene información referente a la identificación de la etiqueta. Se brinda para identificar información relacionada a la misma, que sea distinta de cualquier otro objeto al cual pueda estar adherida. Por lo cual es útil para rastrear la fabricación del circuito integrado (IC) de la etiqueta y así como el inventario de la misma.

**Banco 11 (Usuario)**

El banco 3 está destinado para fines del usuario final. Su organización no está limitada, excepto por el número de bytes y palabras. Está disponible para cualquier tipo de datos específicos de una aplicación.

En la Figura 1.1.3 tomada de [1], se muestra de manera gráfica la organización y terminología de la memoria de una etiqueta RFID de Generación 2.



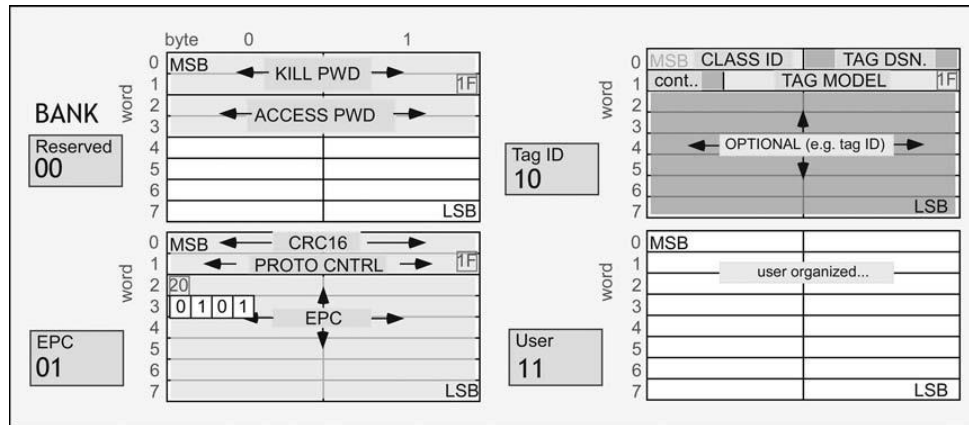


Figura 1.1.3 Organización y Terminología de la Memoria en una Etiqueta RFID Gen 2. (Referencia tomada de [1])

## 1.2 Número de Identificación Vehicular (NIV).

El número de identificación vehicular consta de 17 dígitos alfanuméricos. En México de acuerdo a la NOM 001-SSP-2008 se toman en cuenta los estándares ISO 3779 [5] e ISO 3780 [6] para la asignación del NIV [7]. Los caracteres que conforman el NIV deben ser seleccionados por el fabricante. Hasta el año de 1981 no existía un estándar para la composición de este identificador. Fue hasta principios de los años 80 cuando la Administración Nacional de Seguridad del Tráfico de E.U. estipuló que todos los vehículos terrestres deberían contener un NIV de 17 caracteres que no incluyera las letras I (i), O (o), o Q (Q) (para evitar la confusión con los números 1 y 0). Esto estableció el sistema de NIV estándar que actualmente usa la mayoría de fabricantes de vehículos [8].

## 1.3 Estructura del Trabajo.

En el Capítulo 1 de este trabajo se presentó una breve introducción de los temas y alcances que se abordaran a lo largo de la tesis. Se describen de una forma simple algunos de los términos más importantes que se estarán utilizando a lo largo del trabajo. Por último se presenta una tabla con algunas de las contribuciones más importantes que se obtuvieron con el trabajo realizado.

En el Capítulo 2 se presentan los antecedentes que se tomaron en cuenta para el desarrollo del este trabajo. Así mismo se hace una breve descripción de los temas que éstos abarcan y la relación que tienen con este trabajo.

En el Capítulo 3 se hace una breve descripción del problema de investigación que se abordará y los objetivos que quieren lograr con el trabajo realizado. Este capítulo ayuda a entender la finalidad de este trabajo de tesis y los resultados que se tiene pensado alcanzar.

En el Capítulo 4 se describe la fuente de información en estudio. Se hace una breve explicación de la relación que tiene la fuente de información con la tecnología RFID. Así mismo se describe de manera detallada como se realiza la elección de las muestras de la fuente que serán estudiadas y analizadas.

En el Capítulo 5 se describe el trabajo hecho en cuanto a los modelos matemáticos para describir la fuente de información. Se presentan dos modelos obtenidos en base a los resultados del estudio de las muestras tomadas de la fuente de información. Se describen las características de cada uno de los modelos, así como las ventajas o desventajas que puede conllevar el uso de cada uno de ellos. Cabe mencionar que la principal diferencia entre estos modelos es la generalidad. El primero es un modelo más general en el cual se toman en cuenta una mayor cantidad de números de serie posibles, mientras que el segundo se apega más a las características de condicionalidad que se observaron entre los elementos de la información de la fuente. Al tomar en cuenta estas características el modelo pierde generalidad pero al ser más específico ofrece mejores resultados en temas como la compresión.

En el Capítulo 6 se presentan los diversos métodos para codificación y compresión tomados en cuenta como posibles opciones para llevar a cabo la codificación de la fuente en estudio. Así mismo se encuentran los resultados obtenidos con cada uno de ellos. En esta sección podemos encontrar el método

elegido como mejor opción para la codificación de la fuente de información según las características y propiedades encontradas a lo largo del estudio.

En el Capítulo 7 se presentan de manera puntual los resultados obtenidos al final de la investigación. Para llevar un orden se presentan los resultados en base a los capítulos presentados en este trabajo. Así mismo se trata de presentar aquellos resultados que se consideran relevantes para la investigación.

En el Capítulo 8 se mencionan las conclusiones a las que se llega al final de la investigación. Para ello se toman en cuenta todos los aspectos que se presentaron a lo largo de la investigación así como los resultados obtenidos. La información contenida en este capítulo se puede decir que se obtiene en gran medida a lo presentado en el Capítulo 5.

En el Capítulo 9 se encuentra la lista de referencias que sirvieron como cimientos para fundamentar lo expuesto en este trabajo, y que pueden ser de gran ayuda al momento de querer entender más a fondo algunos de los temas que se tocaron en la elaboración de este trabajo.

#### **1.4 Contribuciones.**

Algunas de las contribuciones importantes consideradas en este trabajo se mencionan a continuación. De igual forma en la TABLA 1.4.1 se presentan estas mismas contribuciones de una manera resumida. En la primera columna se describe el tema o concepto relacionado a dichas contribuciones y en la segunda columna se presenta una breve descripción de dichos conceptos. Cabe mencionar que estos temas se encuentran de manera desarrollada y de una forma más explícita en el cuerpo de este trabajo.

#### **1.4.1 Modelo de la fuente de información.**

Se propone un modelo que describe la fuente de información en estudio en base el uso de cadenas de Markov. Esto mediante el uso de las propiedades condicionales observadas en los elementos de la fuente de información.

#### **1.4.2 Entropía**

Mediante el cálculo de la entropía se define la cota inferior posible para una compresión sin pérdidas, de acuerdo a la fuente de información representada por el modelo propuesto. Este dato es de suma importancia ya que servirá de referencia para evaluar el nivel de compresión obtenido con los métodos propuestos.

#### **1.4.3 Simulador de la fuente.**

En base al modelo propuesto elegido para la descripción matemática de la fuente de información, se realizó un programa computacional regido por las propiedades probabilísticas de dicho modelo. Con el fin de generar una serie de datos que asemejen de la mejor manera la información de la fuente. Simulando de esta manera la información que produce la fuente.

#### **1.4.4 Codificación y compresión.**

Se hace una adaptación a un método encontrado en la literatura que trata la codificación binaria de un NIV. Al adaptar las características del método a la fuente que se estudia se obtienen mejores resultados que al utilizar el método como tal. Además se propone una forma de codificación del NIV por partes, obteniendo un mejor resultado que el obtenido al hacer la codificación de todo el NIV con un mismo método.

#### **1.4.5 Detección y Corrección de errores.**

De acuerdo al método de codificación propuesto se desarrolla un método de detección y corrección de errores para la parte del código que corresponde a la codificación de las primeras 8 posiciones de un NIV.

*TABLA 1.4.1 Contribuciones del trabajo.*

<b>Tema/Concepto</b>	<b>Contribución</b>
Modelo de la fuente de información.	En base a las propiedades de la fuente de información se propone un modelo matemático para describirla.
Entropía	Se obtiene el valor de la entropía para tener una referencia al momento de comparar los niveles de compresión obtenidos mediante los modelos de codificación y compresión propuestos.
Simulador de la fuente.	Se realizó un programa computacional regido por las propiedades probabilísticas del modelo de fuente propuesto. A manera poder replicar la información que produce la fuente.
Codificación y compresión.	Se propone una forma de codificación del NIV por partes, obteniendo un mejor resultado que el obtenido al hacer la codificación de todo el NIV con un mismo método.
Detección y Corrección de errores.	Se desarrolla un método de detección y corrección de errores para la parte del código que corresponde a la codificación de las primeras 8 posiciones de un NIV.

## Capítulo 2

# 2. ANTECEDENTES

### 2.1 Antecedentes.

En la literatura se encuentran diversos trabajos que abarcan el tema de la tecnología RFID. Para este trabajo se tomaron en cuenta principalmente tres que son [9],[10] y [11]. En el primero de los trabajos se estudia la manera de codificar de manera conjunta canal y fuente en un sistema RFID en el cual la fuente de información comprende números de serie de vehículos españoles. Esto para obtener una mejor eficiencia en la transmisión de la información [9] . En [10] se propone una manera de codificar y decodificar de manera binaria cualquier NIV que cumpla con las normas [5] y [6]. Esto mediante el cálculo del número de bits necesarios para la representación de las combinaciones posibles. Además se ha trabajado también en el análisis matemático de un sistema RFID dónde existe comunicación entre una lectora y dos o más etiquetas. Realizando dicho análisis mediante el uso de fundamentos de teoría de la información y basados principalmente en el texto [11].

#### 2.1.1 Codificación conjunta canal-fuente.

En la literatura se tiene el estudio de la codificación conjunta canal-fuente para un sistema RFID UHF [9]. El objetivo del trabajo mencionado se basa en el aumento de la eficiencia de la transmisión de información en sistemas UHF RFID través de un esquema de codificación-conjunta. Que tiene como resultado el aumento en el número de lecturas sin error al utilizar codificación-conjunta. En este trabajo el autor desarrolla una metodología general para introducir los algoritmos de codificación específicos para las aplicaciones en el campo utilizado.

### **2.1.2 Codificación binaria de un NIV.**

Como ya se mencionó los caracteres que conforman un NIV pueden ser números del 0 al 9 y letras sin incluir la I, O y Q, con lo que el alfabeto total se reduce a 33 caracteres. Para la codificación de un NIV se tienen trabajos en la literatura como [10]. En él se considera que si un NIV contiene 17 dígitos y cada dígito puede tomar distintos valores de un alfabeto, se estaría hablando de que existe un total de  $33^{17}$  posibles combinaciones válidas. Al considerar cada combinación o cada NIV como un elemento, el autor propone un sistema de codificación basado en la cantidad de bits necesarios para codificar todos los elementos de manera binaria.

### **2.1.3 Análisis Teórico de un sistema RFID.**

En los antecedentes que se encuentran referentes al análisis de un sistema RFID existe uno cuyo desarrollo está hecho en base a los conceptos y terminologías del texto [12]. Nos referimos al antecedente [11], en el cual se estudia un red pasiva RFID y se modela el sistema físico como una cascada especial de ciertos canales de difusión (BCC) y canales de acceso múltiple (MAC)<sup>1</sup>.

En su trabajo el autor plantea un escenario que consta de dos unidades móviles (etiquetas), y menciona que puede extender este número de unidades a  $M$  usuarios mediante el uso de los mismos argumentos.

El sistema cuenta con un transceptor (dispositivo que funciona como transmisor y receptor), el cual envía un par de mensajes de difusión representados como  $(W_1, W_2) \in \mathcal{W}_1 \times \mathcal{W}_2$  hacia las unidades móviles.

Las unidades móviles decodifican el mensaje que les corresponde, por ejemplo: la unidad 1 decide que se le envió el mensaje  $\hat{W}_1$ , y eligen su mensaje de acuerdo

---

<sup>1</sup> Para mayor referencia de los canales BCC y MAC revisar 12. Cover, T.M. and J.A. Thomas, *Elements of information theory* 2006: Wiley-interscience.

a esta elección tomada, en el caso del ejemplo la unidad elige el mensaje  $M_1 \in \mathcal{M}_1$  y lo envía al transceptor.

En la Figura 2.1.1 se puede ver de manera gráfica este proceso. En la parte de la izquierda están representados los mensajes que la lectora envía a la etiqueta. Estos mensajes son procesados para ser enviados por el canal. A la salida del canal las etiquetas decodifican los mensajes y en base a la decisión que tomen sobre el mensaje recibido eligen la información correspondiente. Dicha información se encuentra almacenada en los campos de memoria de la etiqueta y es enviada de regreso a la lectora RFID.

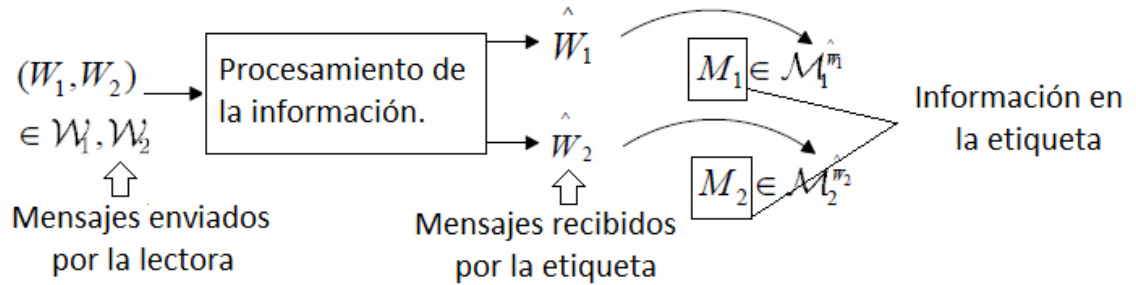


Figura 2.1.1 Recepción de mensajes y selección de información.

El sistema de comunicaciones descrito en [11] cuenta con los siguientes elementos:

Ocho conjuntos finitos que son  $\mathcal{X}, \mathcal{Y}_1, \mathcal{Y}_2, \mathcal{Q}_1, \mathcal{Q}_2, \hat{\mathcal{Q}}_1, \hat{\mathcal{Q}}_2, \mathcal{S}$ .

La parte del canal de difusión “BCC part” descrita por la función de masa de probabilidad (p.m.f.) condicional  $p(y_1, y_2 | x) \in \mathcal{Y}_1 \times \mathcal{Y}_2 \times \mathcal{X}$ . Para la enésima extensión de esta parte se tiene la siguiente expresión:

$$p(y_1^n, y_2^n | x^n) = \prod_{k=1}^n p(y_{1k}, y_{2k} | x_k) \quad (1.1)$$

Las imperfecciones del canal sin memoria, dadas por la p.m.f. condicional  $p(\hat{q}_i | q_i) \in \hat{\mathcal{Q}} \times \mathcal{Q}_i$ .

Para la enésima extensión para las imperfecciones  $i \in \{1, 2\}$



$$p(\hat{q}_i^n | q_i^n) = \prod_{k=1}^n p(\hat{q}_{i,k} | q_{i,k}) \quad (1.2)$$

La parte de canal de acceso múltiple “MAC part”, descrita por la p.m.f. condicional  $p(s | \hat{q}_1, \hat{q}_2) \in \mathcal{S} \times \hat{\mathcal{Q}}_1 \times \hat{\mathcal{Q}}_2$ . Para la  $n$ -ésima extensión de esta parte se tiene la siguiente expresión.

$$p(s^n | \hat{q}_1^n, \hat{q}_2^n) = \prod_{k=1}^n p(s_k | \hat{q}_{1,k}, \hat{q}_{2,k}) \quad (1.3)$$

Mensajes de difusión transmitidos a las unidades móviles o “mensajes del canal de difusión”  $(W_1, W_2) \in \mathcal{W}_1 \times \mathcal{W}_2$  donde  $W_i \triangleq \{1, \dots, 2^{nR_i^D}\}$  para  $i \in \{1, 2\}$ .

La función de codificación del transceptor, nombrada “codificador del canal de difusión” dado por  $X^{BCC}: \mathcal{W}_1 \times \mathcal{W}_2 \rightarrow \mathcal{X}^n$  tal que  $X^{BCC}(W_1, W_2) = x^n(W_1, W_2)$ .

Los decodificadores de las unidades móviles, o “decodificadores del canal de difusión” dados por  $g_i^{BCC}: \mathcal{Y}_i^n \rightarrow \mathcal{W}_i \cup \{0\}$  tal que  $g_i^{BCC}(\mathbf{Y}_i^n) = \hat{W}_i$  para  $i \in \{1, 2\}$ .

Los mensajes de las unidades móviles, o “mensajes del canal de acceso múltiple” correspondientes a los mensajes del canal de difusión decodificados, representados como  $M_i \in \mathcal{M}_i^{\hat{W}_i}$  donde  $M_i^{\hat{W}_i} \triangleq \{1, \dots, 2^{nR_i^{Data}}\}$  para  $i \in \{1, 2\}$ .

Cabe mencionar que es en esta parte en donde se ubica el trabajo realizado en esta tesis. Ya que se trabaja, entre otras cosas, en la forma de codificar y comprimir la información que será almacenada en la etiqueta RFID pasiva. Representada como  $M_i \in \mathcal{M}_i^{\hat{W}_i}$  de acuerdo a la notación que utiliza el autor en [11].

La función de codificación de las unidades móviles, o “codificadores del canal de acceso múltiple”  $Q_i^{MAC}: \mathcal{M}_i^{\hat{W}_i} \rightarrow \mathcal{Q}_i^n$  para  $i \in \{1, 2\}$  tal que  $Q_i^{MAC}(M_i) = q_{\hat{W}_i}^n(M_i)$ .

La función de decodificación del transceptor, o “decodificador del canal de acceso múltiple” dado por  $g^{MAC}: \mathcal{S}^n \rightarrow \mathcal{M}_1^{W_1} \times \mathcal{M}_2^{W_2}$ .

Los mensajes decodificados en el transceptor:  $(M_1, M_2) \in \mathcal{M}_1^{W_1} \times \mathcal{M}_2^{W_2}$ .

Con lo que la representación del diagrama de bloques para el sistema de comunicación multiusuario mencionado quedaría como se muestra en la Figura

2.1.2. En dicho diagrama únicamente se hace el análisis para dos etiquetas en la red pasiva RFID UHF, pero se hace mención que para un número mayor de etiquetas el análisis se hace siguiendo los mismos principios.

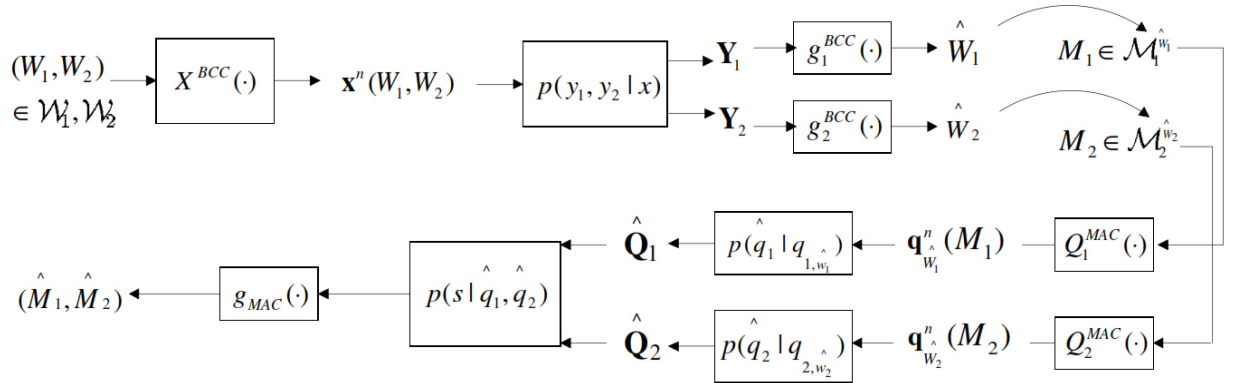


Figura 2.1.2 Diagrama de bloques para el sistema de comunicación multiusuario en [11]

Esta investigación estaría ubicada en la parte de los elementos  $M_1 \in \mathcal{M}_1$  del sistema de comunicaciones propuesto por el autor. Ya que son los bloques referentes a la información que está almacenada en las etiquetas RFID del sistema. Y es sobre esta información donde se requiere proponer un método para la codificación y compresión antes de ser grabada en las etiquetas.

El análisis de este escenario sirve para ubicar en que parte del sistema de comunicaciones se trabaja actualmente. Este trabajo se ubica en la parte de  $M_1$  y  $M_2$ . Es decir, en la información que se encuentra almacenada en la etiqueta. Ya que al tratarse de etiquetas pasivas para esta aplicación, la cantidad de almacenamiento en ellas se ve limitada. Por lo tanto se busca proponer un esquema de compresión para la información que se requiere almacenar, con la finalidad de ahorrar espacio en bits dentro de la etiqueta para otros fines que pudieran necesitarse.

## Capítulo 3

# 3. PLANTEAMIENTO DEL PROBLEMA

### 3.1 Planteamiento del problema.

Hablando del tema de parque vehicular y de acuerdo a un censo realizado en el año 2012, en México circulan más de 33 millones de vehículos. De éstos vehículos más de 2.1 millones lo hacen en el estado de Nuevo León [13]. Como ya se mencionó cada uno de estos vehículos cuenta con un NIV único (Número de Identificación Vehicular). Esto conlleva a que se tenga una gran cantidad de información formada por todos estos números de serie que representan e identifican los vehículos que componen el parque vehicular mexicano. A pesar de que se está consciente de ésta gran cantidad de información existente, no se ha encontrado en la literatura un modelo matemático que describa o exprese la estructura o comportamiento de esta fuente de información.

El NIV debido a que es un identificador único e irrepetible, pasa a ser un importante elemento al momento de querer identificar un automóvil. A este identificador único se le pueden asociar otros datos de interés relacionados al vehículo como pueden ser dueños anteriores, infracciones, irregularidades, etc. Inclusive aquella información confidencial que pueda ser útil a las autoridades para algún fin en específico.

Las bases de datos que tienen esta información sobre los vehículos que circulan en territorio nacional ya existen (Plataforma México y REPUVE-Registro Público

Vehicular) y la tecnología para poder explotar esta información también. En México de acuerdo a [14] todos los automóviles que circulen en territorio nacional contarán con una etiqueta con tecnología RFID pasiva pegada en los parabrisas. Estas etiquetas llevarán escrita información referente al automóvil al cual están adheridas. Debido a la tecnología que manejan las etiquetas, la cantidad de información que se puede almacenar es limitada, por lo cual no se puede almacenar toda la información que se tenga sobre el vehículo en dicha etiqueta. Aquí es donde entra el NIV, ya que al ser únicamente un identificador de longitud fija, no ocupa tanta memoria como lo haría grabar toda la información referente al automóvil. Teniendo el NIV identificado se puede hacer uso de otros recursos tecnológicos para hacer una búsqueda cruzada de la información referente al automóvil mediante su NIV. Así se podrá mostrar ésta información al usuario final. Por lo tanto, las posibilidades de crear un sistema de comunicaciones para control vehicular, mediante la implementación de una red RFID, se incrementan.

Aunque las posibilidades y alcances de crear dicho sistema sean muy viables en cuanto a infraestructura y tecnología, la falta de un modelo matemático que describa el comportamiento de la fuente de información, hace que se compliquen temas referentes al manejo de la misma. Por ejemplo la codificación y la compresión que tiene que hacerse para que exista una comunicación rápida y confiable.

### **3.1.1 Acotamiento del problema**

Debido en parte a la gran generalidad del problema que se tiene al considerar los más de 30 millones de vehículos, y el acceso limitado hacia las bases de datos oficiales con información sobre los vehículos, se decidió acotar el área geográfica del problema de investigación. Por lo cual este trabajo estará enfocado al estudio, análisis y propuestas para aquellos elementos (NIVs) que cumplan con las siguientes características:

- Automóviles fabricados en territorio nacional (México).
- Automóviles que circulen en el estado de Nuevo León.

- Automóviles de modelos 2003 a 2013.

### **3.2 Objetivos.**

El objetivo principal de este trabajo es el de proponer un modelo matemático en base al análisis y caracterización de una serie de datos reales tomados de la fuente de información de interés. Ésta fuente está formada por NIVs que actualmente circulan en el estado de Nuevo León, México. Se toma como objetivo principal ya que en base a dicho modelo se buscarán encontrar métodos para la codificación y compresión de la información. Los cuales ofrezcan mejores resultados en comparación a los que se pueden obtener tomando como referencia lo que actualmente se encuentra en la literatura.

Actualmente no se encuentra en la literatura algún trabajo que proponga un modelo matemático para describir la fuente de información que se estudiará en este trabajo. De igual forma no se encuentran trabajos referentes a temas de codificación y compresión. Debido a esto la importancia de este trabajo recae en encontrar un modelo matemático que describa la fuente de información de interés de la mejor manera posible. Después en base a este modelo proponer un esquema de codificación y compresión de la información que se va a transmitir. Con lo cual al final del trabajo se pueda tener un sistema de comunicaciones en el cual la transmisión y recepción de la información pueda llevarse a cabo de manera confiable. Lo anterior tomando en cuenta que la información ya se encuentre comprimida, y que dicha compresión se hará en base a las características de la fuente según el modelo matemático que se haya propuesto.

## Capítulo 4

# 4. DEFINICIÓN DE LA FUENTE INFORMACIÓN.

### 4.1 Fuente de información.

Como se mencionó en la sección 3.1 la fuente de información que tomaremos para este trabajo está formada por los números de identificación vehicular de vehículos que circulan en el estado de Nuevo León, que además, hayan sido fabricados en México y sean de modelos entre el año 2003 y el 2013. Esto con el fin de reducir la generalidad inicial del problema, en la que se tomaban en cuenta los más de 30 millones de vehículos registrados en la actualidad. Cabe mencionar que posteriormente se puede hacer una generalización de las dimensiones del problema. De forma que mediante de los mismos métodos utilizados en el estudio de esta población en particular, se pueda realizar un estudio para una fuente de información que incluya características más generales, por ejemplo una fuente que incluya todos los automóviles que circulen en territorio nacional, fabricados en México o en el extranjero y de cualquier año.

El escenario de este trabajo consiste en utilizar las etiquetas RFID pasivas adheridas a los vidrios parabrisas en los coches en México, en conjunto con el identificador único de cada uno de ellos (NIV) para una aplicación de control vehicular mediante el uso de esta tecnología y transmitiendo este valor único de cada coche.

#### 4.1.1 Estructura del NIV.

Históricamente las fábricas automotrices de Detroit empezaron plasmando números de identificación en carros y sus partes en el año de 1954. Hasta el año de 1981 no existía un estándar para la composición de estos números, por lo cual diferentes fabricantes usaban distintos formatos. Fue hasta principios de los años 80 cuando la Administración Nacional de Seguridad del Tráfico de E.U. estipuló que todos los vehículos terrestres deberían contener un NIV de 17 caracteres que no incluyeran las letras I (i), O (o), o Q (Q) (para evitar la confusión con los números 1 y 0). Esto estableció el sistema de NIV estándar que actualmente usa la mayoría de fabricantes de vehículos [8].

El número de identificación vehicular consta de 17 dígitos. Éstos a su vez se dividen en 4 secciones según la Norma Oficial Mexicana NOM-001-SSP-2008 [7]. La primera sección está compuesta por los caracteres 1º al 3º y está relacionada al código identificador del fabricante internacional. La segunda sección se compone de los caracteres 4º al 8º y se relaciona con la descripción del vehículo. El 9º carácter se calcula en base a los demás valores que conforman el NIV y se utiliza como dígito verificador para corroborar la validez del mismo. La cuarta sección corresponde a los caracteres 10º al 17º y está relacionada con la identificación individual del vehículo.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
<b>3</b>	<b>N</b>	<b>1</b>	<b>E</b>	<b>B</b>	<b>3</b>	<b>1</b>	<b>S</b>	<b>1</b>	<b>4</b>	<b>K</b>	<b>3</b>	<b>1</b>	<b>4</b>	<b>5</b>	<b>5</b>	<b>8</b>	
<b>1ª Sección</b>			<b>2ª Sección</b>					<b>3ª</b>	<b>4ª Sección</b>								

*Figura 4.1.1 Secciones del NIV según la NOM [7]*

En la TABLA 4.1.1 se muestra una descripción más detallada de las secciones que componen el NIV y la información asociada a cada sección.

*TABLA 4.1.1 Secciones que conforman el NIV [7]*

Dígitos	Sección	Descripción
1ro	1ra	Identificar mundialmente al fabricante o ensamblador. Opcionalmente el fabricante podrá incluir caracteres de esta sección en la segunda sección.
2do		Si el fabricante produce más de 500 unidades al año utilizara los primeros 3 caracteres.
3ro		Si el fabricante produce hasta 500 unidades al año, deberá además utilizar los caracteres 12,13 y 14.
4to	2da	Contiene información que describe las características generales de vehículos. Los caracteres y su secuencia se determinan por el fabricante o ensamblador. Los espacios no utilizados deben ser ocupados por los caracteres "Z" o "0".
5to		
6to		
7mo		
8vo		
9no	3era	Dígito de Verificación
10mo	4ta	Primer carácter hace referencia al año
11mo		Segundo carácter hace referencia a la planta de fabricación y es asignado por el fabricante o ensamblador.
12mo		Los últimos seis caracteres corresponden al número de serie consecutivo de producción del vehículo cuando se trata de un fabricante de más de 500 vehículos por año.
13ro		
14to		
15to		Cuando el fabricante produce 500 vehículos o menos por año, los caracteres de las posiciones 12 a la 14 serán los asignados por la autoridad competente o el organismo facultado en el país de origen; los caracteres del 15 al 17 corresponden al número de serie.
16to		
17mo		

En la TABLA 4.1.2 se puede ver la equivalencia del carácter que ocupe la 10ma posición del NIV y el año de fabricación del automóvil.



*TABLA 4.1.2 10ma Posición del NIV*

Año Modelo	Clave	Año Modelo	Clave	Año Modelo	Clave
2001	1	2011	B	2021	M
2002	2	2012	C	2022	N
2003	3	2013	D	2023	P
2004	4	2014	E	2024	R
2005	5	2015	F	2025	S
2006	6	2016	G	2026	T
2007	7	2017	H	2027	V
2008	8	2018	J	2028	W
2009	9	2019	K	2029	X
2010	A	2020	L		

Esta estructura del NIV sirve para identificar a los vehículos de modelos posteriores a 1998, en modelos anteriores a 1998 a este número se le conocía como número serie y no había una norma a nivel mundial que rigiera la estructura de dicho número de identificación [7]. En México de acuerdo a la NOM 001-SSP-2008 se toman en cuenta los estándares ISO 3779 [5] e ISO 3780 [6] para la asignación del NIV. Los caracteres que conforman el NIV deben ser seleccionados por el fabricante o ensamblador, utilizando el siguiente alfabeto:

Caracteres Alfabéticos: A B C D E F G H J K L M N P R S T U V W X Y Z

Caracteres Numéricos: 0 1 2 3 4 5 6 7 8 9

## **4.2 Almacenamiento de la información.**

### **Etiqueta RFID UHF Pasiva.**

En este trabajo lo que se quiere es proponer un sistema de codificación y compresión de los datos de la fuente para su transmisión por un sistema RFID UHF con tecnología pasiva en las etiquetas. Debido a la tecnología pasiva, en la cual las etiquetas no tienen una batería que las alimente, la capacidad de almacenamiento de las etiquetas pasivas es menor a que en las activas, en las cuales se cuenta con batería propia. Así que, por este motivo, se busca una manera eficaz de comprimir parte de la información que irá guardada en dichas

etiquetas para así dejar espacio disponible para otra información que se requiera almacenar, como lo es la información propia que ya tiene la etiqueta almacenada en ella.

Las etiquetas pasivas carecen de una batería que alimente su circuito integrado. Por lo tanto, utilizan circuitos de bajo costo, baja velocidad y con lógica simple para de esta forma permitir que el circuito integrado pueda ser alimentado por la pequeña cantidad de potencia (micro watts) logre captar del lector el cual por lo general se localiza a varios metros de distancia de la etiqueta.

Un diagrama simplificado de la estructura eléctrica de una etiqueta pasiva está descrito en la Figura 4.2.1. Aquí la señal del lector que oscila en los 900 MHz es convertida por la antena en corriente alterna, de esta corriente la etiqueta debe extraer tanto la información enviada como la corriente para alimentar su circuito. Una vez hecho esto la etiqueta debe interpretar la información recibida, extraer de su memoria no volátil la información requerida por la lectora, y modular la carga presentada en la antena para poder responder con otra señal de radiofrecuencia a la lectora RFID.<sup>2</sup>

En una etiqueta pasiva la estructura de la antena interactúa con campos electromagnéticos que inciden, produciendo voltaje de alta frecuencia (RF). Dicho voltaje es rectificado por un diodo (solo permite que la corriente fluya en una dirección), y la señal resultante se suaviza usando un capacitor de almacenamiento para crear un voltaje más o menos constante que después es usado para alimentar el circuito lógico de la etiqueta y el acceso a la memoria. Esto se muestra de manera ilustrada en la Figura 4.2.2 y de una forma más esquemática en la Figura 4.2.3. Ambas figuras fueron tomadas de la referencia [1].

---

<sup>2</sup> Para mayor referencia sobre el cambio de RF a CD para alimentación del circuito revisar 1. Dobkin, D.M., *The RF in RFID: passive UHF RFID in practice*.2008.

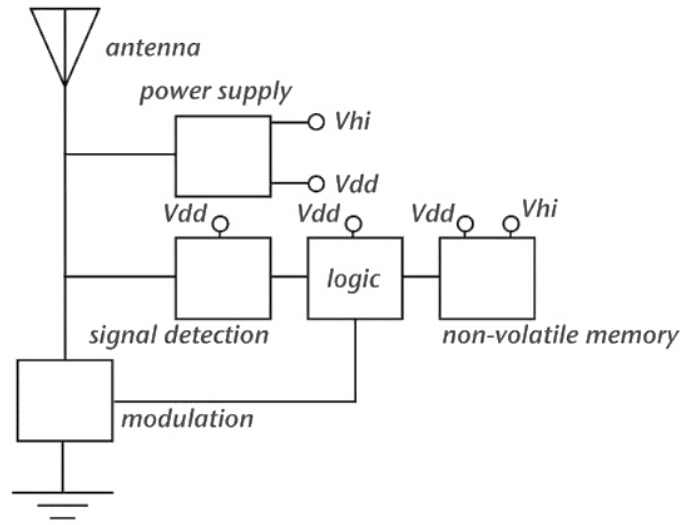


Figura 4.2.1 Elementos de una etiqueta UHF pasiva. (Referencia tomada de [1])

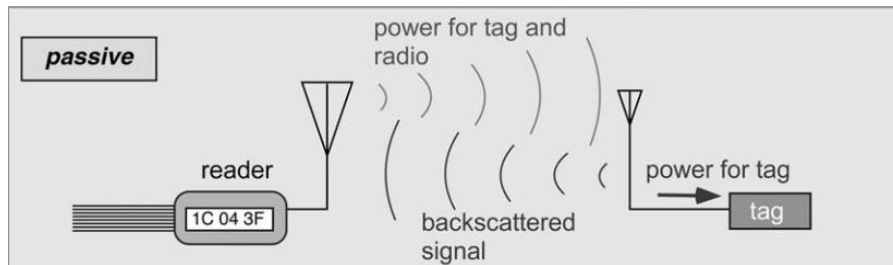


Figura 4.2.2 Etiqueta pasiva. (Referencia tomada de [1])

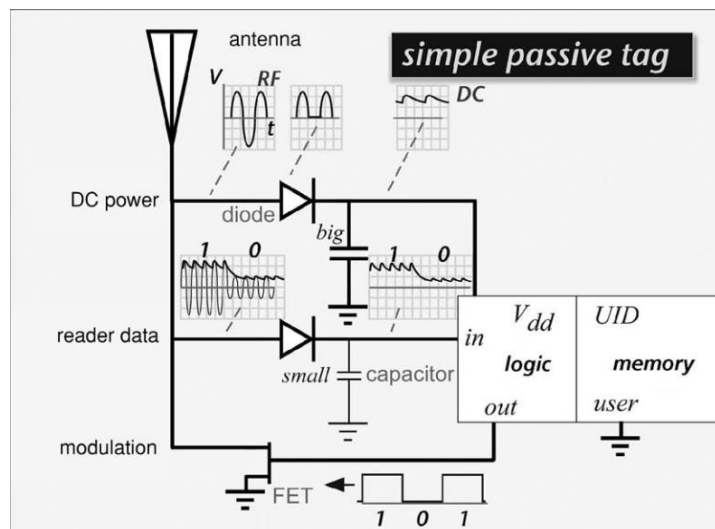


Figura 4.2.3 Representación esquemática de una etiqueta RFID pasiva simple. (Referencia tomada de [1])

### **4.3 Obtención de datos de la fuente.**

Para fines de estudio se buscó tener una serie de muestras reales de la fuente de interés para su posterior análisis. Estas bases de datos debían contener principalmente número de identificación vehicular de automóviles que circularan en el Estado de Nuevo León, ya que como se mencionó en el planteamiento del problema, es el área geográfica en la que se eligió hacer el estudio por fines prácticos. En México existen bases de datos, que como ya se mencionó, contienen información de los vehículos registrados en territorio nacional, algo así como un historial del vehículo desde su primer registro hasta la actualidad. Esta información al ser de carácter oficial está bajo administración de las autoridades competentes. Esto debido a que se maneja información confidencial de los propietarios actuales y anteriores de los vehículos. La cual, al caer en manos equivocadas, podría ser usada para fines indebidos. Por lo tanto, el acceso a esta información o a estas bases de datos es en cierta forma restringido y sólo se le da acceso a él a las autoridades competentes. Lo ideal para esta investigación habría sido contar con una base de datos oficial y actualizada la cual contara con los registros de todos los automóviles que circulan en el estado de Nuevo León y su información, en especial el NIV de cada uno de ellos. Pero debido a las limitaciones de acceso antes mencionadas, se decidió trabajar con una serie de datos obtenidos de una fuente de confianza como lo es la página de la Procuraduría General de Justicia del Estado de Nuevo León, dicha base de datos contiene, entre otra información, los números de serie o NIVs de automóviles que contaban con reporte de robo (en su momento) en el Estado de Nuevo León[15]. Se decide trabajar partiendo de que al ser información obtenida de una fuente confiable se puede asumir de igual manera que los datos obtenidos son verdaderos. Entonces para fines de estudio, avances y propuestas en este trabajo, se estará tomando esta base de datos como la serie de muestras representativas de la población de interés, con la posibilidad de adaptar los resultados obtenidos al momento de encontrar alguna fuente real que represente de mejor manera la población, esto mediante el uso y seguimiento de la misma

metodología que se explicará en este trabajo y permitiendo con ello que exista la posibilidad de una extrapolación de la metodología para diversas fuentes futuras.

#### **4.4 Clasificación de los datos.**

Una vez hecha la elección de la base de datos con la que se trabajará se debe hacer un filtrado y selección de aquellos que cumplan con las características siguientes: NIVs cuya estructura este basada en los ISO 3779 y 3780, autos que hayan sido fabricados en territorio nacional (México), autos que circulen en el estado de Nuevo León y por último autos cuyo modelo este entre los años 2003 y 2013, es decir, que tenga 10 años de antigüedad o menos.

##### **4.4.1 NIVs que cumplan con el estándar actual.**

La primera etapa de clasificación fue la selección de aquellos números de identificación que cumplieran con las características del NIV antes mencionadas, es decir, para esta etapa se conservaron aquellos datos cuya longitud fuera de 17 caracteres. Esto debido a que al ser una base de datos que probablemente fue llenada por personas, existe la posibilidad de que algunos de los datos en ella sean erróneos, ya sea por una transcripción equivocada por parte del encargado de llenar los bancos de información o error por parte de la persona que brindó la información. Por lo tanto la manera más sencilla de descartar todos aquellos números de serie cuya estructura no corresponda a lo estipulado en los estándares es verificando si su longitud es de 17 dígitos. Todos aquellos números cuya longitud sea diferente a 17 serán tomados como números no válidos y se descartarán para el estudio.

**3N1EB31S14K314558**  
└──────────────────┘  
**17 dígitos**

*Figura 4.4.1 Clasificación: longitud.*

#### 4.4.2 Autos fabricados en Territorio Nacional.

Después del primer paso de la clasificación se tiene una nueva base de datos con NIVs que en teoría ya cumplen con lo estipulado en las normas oficiales, por lo cual se procede a realizar el siguiente paso en la clasificación: *autos fabricados en territorio nacional*. Para esta parte de la clasificación se hace uso de lo estipulado en la norma ISO 3780, la cual define al identificador mundial de fabricante. Los primeros 3 dígitos del NIV están relacionados con el lugar de fabricación del automóvil, siendo el primer carácter el que indica el país de fabricación. Entonces, de acuerdo a la norma los autos fabricados en México deben tener un NIV que empiece con el dígito “3”. Por lo tanto la segunda parte de la clasificación es descartar todos aquellos NIVs cuyo primer dígito sea diferente de “3”. De esta forma nos aseguramos de, en teoría, conservar sólo aquellos NIVs de automóviles que fueron fabricados en territorio nacional mexicano.

Para el caso de autos fabricados en México, su NIV inicia con el dígito “3”, por lo cual en la primer parte de la clasificación se extraen todos aquellos NIVs que empiecen con el número “3”.

**3N1EB31S14K314558**  
↑  
**Lugar de fabricación**

*Figura 4.4.2 Clasificación: Lugar de Fabricación.*

#### 4.4.3 Autos circulando en el estado de Nuevo León.

Esta parte de la clasificación ya está implícita desde el momento que se elige una base de datos con información correspondiente a automóviles de esta entidad federativa de la República Mexicana. Por lo cual, en cierto modo, es la primera clasificación que se realizó.

#### 4.4.4 Autos con modelos del 2003-2013

Por último y una vez hechos los pasos anteriores, se procede a extraer únicamente aquellos NIVs cuya décima posición indique que fueron fabricados entre los años 2003 y 2013. Para esto nos basaremos en la TABLA 4.1.2 en donde aparece la nomenclatura que se le da a cada año y que ocupa la posición número 10 en el NIV. Un ejemplo de esto es el NIV mostrado en la Figura 4.4.3 que corresponde a un automóvil de modelo 2004 de acuerdo a la TABLA 4.1.2.

**3N1EB31S14K314558**  
                                  ↑  
**Año Modelo**

*Figura 4.4.3 Clasificación: Año(s).*

Esta selección se debe hacer ya que la base de datos mencionada en 4.3 está formada por elementos de todo tipo y que no precisamente cumplen con las características de interés que se desea que cumplan las muestras. Al hacer esto se obtendrá de resultado una serie de muestras representativas que cumplirán con las características de interés y sobre las cuales se realizarán las acciones de análisis y estudio para proponer un modelo matemático que pueda representar dicha fuente de información.

## Capítulo 5

# 5. MODELOS PROPUESTOS.

Una de las principales contribuciones de este trabajo de tesis es el diseño de un modelo matemático el cual sea capaz de describir el comportamiento de la fuente de información descrita anteriormente. Por lo cual esta sección puede ser considerada como una de las más relevantes de este trabajo de tesis.

En esta parte del trabajo se obtiene el modelo matemático con el cual se trabajará a lo largo de la investigación. Dicho modelo es de gran importancia ya que en base a él se desarrollaran los temas de codificación y compresión de la información. Por lo cual el modelo puede ser considerado la base de los resultados obtenidos en este trabajo de tesis. La importancia de diseñar un modelo matemático que describa el comportamiento de la fuente recae en el hecho de que actualmente no se encuentre en la literatura algún modelo lo haga.

El objetivo de este trabajo, como ya se mencionó, es proponer un modelo que describa una fuente de información a partir del estudio de una serie de muestras que representan esta fuente. A lo largo del estudio se plantearon diversos modelos que pudieran ayudar a describir el comportamiento de la fuente de información. A continuación se dará una explicación de los 2 modelos que fueron los que se consideraron que podrían resultar más útiles.

### 5.1 Modelo 1.

#### 5.1.1 Modelo probabilístico.

El modelo estadístico más simple para una fuente es asumir que cada símbolo generado por la fuente es independiente de cualquier otro símbolo, y que cada



uno ocurre con la misma probabilidad. Este modelo es llamado el modelo de ignorancia y sería generalmente útil en aquellos casos en los que no se conoce nada sobre la fuente de información. Después de este modelo, el que le sigue en complejidad es aquel en el que se preserva la independencia entre la ocurrencia de los símbolos, pero se quita la igualdad en probabilidad de ocurrencia [16]. En base a este tipo de modelos se propone una primera opción para representar la fuente de información. Se hace un análisis de las muestras que se tienen posición por posición. Se estudian los conjuntos de símbolos que existen en cada posición para obtener un alfabeto y una distribución de probabilidades de ocurrencia de cada uno de ellos. Este estudio se realiza sobre el conjunto de muestras representativas de la población de interés que se tomó y cuya selección se explica en la sección 4.4 de este trabajo.

Para esto se separan los NIVs que se tenían originalmente en conjuntos que incluyen los elementos de cada posición, es decir, se forma un conjunto de símbolos con todos aquellos que ocupan la segunda posición en las muestras estudiadas, un grupo donde se ponen todos los símbolos que ocupan la tercera posición, y así para cada una de las 17 posiciones. Con lo cual, al final se tienen 17 grupos, cada uno representando una posición en el NIV. De esta forma se procede al estudio de cada grupo, se examina cuáles son los distintos símbolos que aparecen y con qué frecuencia lo hacen, para con esto determinar el alfabeto correspondiente a la posición que representa el grupo en cuestión y a su vez determinar la probabilidad de ocurrir que tienen los símbolos en dicha posición. Es decir, se obtiene un modelo probabilístico que asume independencia entre los símbolos y distintas probabilidades de ocurrencia. Se obtiene un modelo de estas características para cada uno de los 17 grupos.

Una manera de expresar lo anterior sería representando a cada símbolo como  $A_k(s_k)$  en donde;

$A_k$  = Alfabeto para la posición  $k$ .

$n_k$  = Número de símbolos en  $A_k$ .

$$1 \leq s_k \leq n_k$$

para

$$k \in \{1, 2, \dots, 17\}$$

Por ejemplo  $A_1(1) = '3'$  debido a que de antemano se conoce que el alfabeto 1 ( $A_1$ ) únicamente está compuesto por el símbolo '3', por lo cual su tamaño es 1. En este caso  $k$  sólo puede tomar el valor de 1.

### **5.1.2 Resultados del análisis Modelo 1.**

Los resultados obtenidos de éste análisis se muestran más adelante en tablas, donde se pueden ver los elementos que conforman los alfabetos junto con sus probabilidades de ocurrencia.

En primer lugar se presentan las tablas en las que se encontró que existe mayor diferencia en cuanto a la distribución de probabilidades de los símbolos, es decir, en donde para cada posición existe uno o una serie de símbolos que ocurren con mucha más frecuencia que los demás (TABLA 5.1.1, TABLA 5.1.2, TABLA 5.1.3)

*TABLA 5.1.1 Probabilidad posiciones 2,3, y 4.*

<b>P2</b>	<b>Probabilidad</b>
N	57.474%
G	21.135%
V	10.182%
D	2.701%
H	1.112%
C	2.138%
F	3.100%
A	0.595%
T	1.025%
B	0.538%

<b>P3</b>	<b>Probabilidad</b>
1	60.195%
N	6.505%
W	10.182%
6	5.930%
C	4.289%
4	1.575%
G	0.532%
Z	1.076%
D	1.034%
E	1.031%
2	1.134%
K	0.595%
B	2.186%
T	1.049%
A	1.105%
M	1.025%
Y	0.557%

<b>P4</b>	<b>Probabilidad</b>
E	56.991%
F	1.595%
R	3.202%
Y	2.723%
W	1.629%
P	2.101%
C	4.720%
D	4.285%
B	2.750%
J	3.886%
1	0.539%
A	0.543%
K	2.674%
L	1.575%
N	1.055%
T	3.215%
H	1.081%
S	3.771%
M	1.118%
G	0.547%

*TABLA 5.1.2 Probabilidad posiciones 5,6 y 7.*

<b>P5</b>	<b>Probabilidad</b>	<b>P6</b>	<b>Probabilidad</b>	<b>P7</b>	<b>Probabilidad</b>
B	51.049%	3	51.579%	1	59.086%
C	10.903%	1	18.464%	6	8.060%
G	3.827%	4	5.868%	9	5.342%
V	5.342%	5	8.579%	E	1.577%
N	4.281%	6	4.772%	2	4.889%
K	3.231%	7	2.074%	4	6.465%
M	1.112%	0	1.591%	F	1.575%
E	1.076%	9	1.055%	8	1.076%
D	5.348%	C	0.503%	A	6.558%
F	5.823%	L	0.595%	C	1.648%
W	1.112%	2	3.243%	3	1.613%
S	0.495%	H	0.538%	7	0.510%
7	0.538%	A	0.580%	J	0.538%
A	2.698%	B	0.559%	5	1.063%
P	0.525%				
U	1.025%				
L	0.536%				
2	0.539%				
X	0.540%				

*TABLA 5.1.3 Probabilidad posiciones 8 y 11.*

<b>P8</b>		<b>P11</b>	<b>Probabilidad</b>
	<b>Probabilidad</b>		
S	52.195%	K	53.265%
J	2.201%	G	12.999%
T	9.106%	M	17.563%
K	4.899%	T	1.575%
M	5.342%	L	8.542%
D	2.682%	S	6.056%
5	1.561%		
V	0.540%		
3	0.532%		
7	1.612%		
C	1.085%		
L	1.026%		
		0	1.055%
		G	2.151%
		F	3.263%
		8	0.533%
		X	3.248%
		P	2.141%
		W	0.510%
		N	1.582%
		B	0.532%
		6	1.624%
		R	0.580%

En cuanto a los demás grupos, se decidió trabajar de manera distinta con ellos para fines prácticos y de ampliación de cobertura del modelo. En primer lugar

comenzaremos con los valores de la posición número 9. Este carácter corresponde a un dígito verificador, el cual se calcula en base a los otros 16 dígitos que componen el NIV, y cuyo cálculo no es de gran relevancia para este trabajo pero se puede ver claramente explicado en [7]. Los valores de éste dígito no se incluyen en las primeras tablas debido a que, a diferencia de los alfabetos ya presentados, las probabilidades de ocurrencia de los símbolos en esta posición son muy similares. Por eso, y para fines prácticos, se hace la suposición de que las probabilidades son igualmente distribuidas y pueden tomar valores del 0 al 9 y la letra X. Esto tomando como referencia lo estipulado en la Norma Oficial Mexicana [7].

*TABLA 5.1.4 Probabilidad posición 9.*

<b>P9</b>	<b>Probabilidad</b>
0	9.091%
1	9.091%
2	9.091%
3	9.091%
4	9.091%
5	9.091%
6	9.091%
7	9.091%
8	9.091%
9	9.091%
X	9.091%

Por otra parte, como ya se mencionó en 4.1.1, la cuarta sección del NIV corresponde a la identificación del vehículo. En esta sección los últimos seis dígitos (12<sup>o</sup> a 17<sup>o</sup>) corresponden al número de serie consecutivo de producción del vehículo, mientras que el primero (10<sup>o</sup>) corresponde al año de fabricación. El análisis para estos campos fue distinto, ya que (para dar mayor dimensión a la cobertura del modelo) se consideraron todos los valores válidos según la norma. Además se consideró una probabilidad de ocurrencia igual para cada uno de los

elementos en cada uno de los alfabetos. Por lo cual las tablas de probabilidades para estas posiciones (en este modelo) quedan de la siguiente manera:

*TABLA 5.1.5 Probabilidades Posiciones 10,12-17.*

<b>P10</b>	<b>Probabilidad</b>	<b>P12</b>	<b>Probabilidad</b>	<b>P13</b>	<b>Probabilidad</b>	<b>P14- P17</b>	<b>Probabilidad</b>
3	9.091%	0	9.091%	0	10.000%	0	10.000%
4	9.091%	1	9.091%	1	10.000%	1	10.000%
5	9.091%	2	9.091%	2	10.000%	2	10.000%
6	9.091%	3	9.091%	3	10.000%	3	10.000%
7	9.091%	4	9.091%	4	10.000%	4	10.000%
8	9.091%	5	9.091%	5	10.000%	5	10.000%
9	9.091%	6	9.091%	6	10.000%	6	10.000%
A	9.091%	7	9.091%	7	10.000%	7	10.000%
B	9.091%	8	9.091%	8	10.000%	8	10.000%
C	9.091%	9	9.091%	9	10.000%	9	10.000%
D	9.091%	A	9.091%				

El primer carácter, debido a cuestiones de clasificación siempre será “3”, tal y como se explicó en la sección 4.4.2. Por lo cual este valor no será tomado en cuenta al momento de codificar la información, ya que al ser un valor conocido no aporta información alguna para su transmisión.

### **5.1.3 Cálculo de probabilidad por NIV.**

En este modelo se hace la suposición de que para cada una de las diecisiete posiciones del NIV existe un alfabeto válido, por lo cual se tendrán diecisiete alfabetos con un conjunto de símbolos único (en teoría) para cada uno. A su vez cada uno de los símbolos que conforman los alfabetos tendrá una probabilidad de ocurrencia distinta, es decir, la aparición de un símbolo en la posición “k” (dónde  $k=1\dots 17$ ) dependerá únicamente de la probabilidad de ocurrencia que tenga el mismo en el alfabeto  $A_k$ . Para este modelo no importa que símbolo o combinación de símbolos haya en las posiciones  $k-1$  y anteriores.

De esta forma la probabilidad de ocurrencia de un NIV se obtendrá suponiendo que la aparición de un símbolo en cada una de las 17 posiciones es producto de un evento independiente. Por lo cual la probabilidad de ocurrencia del NIV se obtiene calculando la probabilidad conjunta de los 17 eventos que ocurrieron para dar origen a dicho NIV.

## **5.2 Modelo 2.**

Antes de comenzar a describir el segundo modelo que se diseñó en este trabajo, cabe mencionar que existen procesos probabilísticos como lo son el procesos de Bernoulli y el procesos de Poisson. Estos dos procesos reciben el nombre de procesos sin memoria, debido a que en ellos el futuro no depende del pasado. Es decir, la ocurrencia de nuevos eventos o "salidas" no depende de la historia pasada del proceso. Este tipo de procesos probabilísticos no serían de utilidad en nuestro modelo, debido a que la información con la que se trabajará en este trabajo está formada por una cadena de 17 símbolos. Y la probabilidad de ocurrencia de dichos símbolos está ligada a la combinación de símbolos que haya sucedido en el pasado. Es decir, el futuro depende y puede ser predicho por lo que sucedió en el pasado.

Debido a esto es que se decidió usar un modelo probabilístico basado en Cadenas de Markov, ya que es un tipo de proceso probabilístico con memoria, es decir, donde la ocurrencia de un evento depende de lo que sucedió en el pasado, por lo cual, este tipo de proceso se acerca más a las necesidades que se tienen en el modelado de la fuente de información.

A continuación se describe de manera breve la implementación de un modelo de Markov para describir el comportamiento de la fuente de información en estudio. En caso de que se requiera información más detallada sobre el tema de Cadenas de Markov se puede consultar [17] en su Capítulo 7.

### 5.2.1 Modelo de Markov.

Los modelos de Markov son una de las formas más populares de representar una fuente con dependencia entre sus datos. Las cadenas de Markov de tiempo discreto son un tipo específico de proceso de Markov el cual se utiliza en aquellos modelos usados en compresión sin pérdida. Suponga  $\{X_n\}$  es una secuencia de observaciones. Esta secuencia se dice que sigue un Modelo de Markov de orden  $k$  si

$$P(x_n | x_{n-1}, \dots, x_{n-k}) = P(x_n | x_{n-1}, \dots, x_{n-k}, \dots) \quad (5.1)$$

Es decir, el conocimiento de los pasados  $k$  símbolos es equivalente al conocimiento del pasado completo del proceso. Los valores tomados por el conjunto  $\{x_{n-1}, \dots, x_{n-k}\}$  son llamados los estados del proceso. Si el tamaño del alfabeto es  $l$ , entonces el número de estados es  $l^k$ . El modelo de Markov más comúnmente usado es el de primer orden, para el cual

$$P(x_n | x_{n-1}) = P(x_n | x_{n-1}, x_{n-2}, x_{n-3}, \dots) \quad (5.2)$$

Donde se puede ver que se puede conocer el pasado completo del proceso al conocer el símbolo anterior al actual.

Tratando de aprovechar lo antes dicho, se propone un segundo modelo matemático (Modelo 2) mediante el uso de Cadenas de Markov. Éste nuevo modelo propuesto parte de las mismas bases de análisis que el Modelo 1 pero presenta algunas ventajas en comparación a dicho modelo. El Modelo 1 al tomar en cuenta únicamente la probabilidad de ocurrencia de un símbolo de acuerdo a las tablas de probabilidades obtenidas, hace que cualquier combinación sea posible y cuente con una probabilidad de ocurrencia válida. Esto lo hace abarcar grandes dimensiones ya que acepta como válido cualquier NIV que se presente y se pueda formar en base a sus alfabetos, pero a la vez debido a esto hace que existan demasiadas combinaciones posibles, con probabilidades que, haciendo un análisis más a fondo, no representan lo que en realidad se observa en el estudio. Un ejemplo de ello es que algunas combinaciones que en la realidad no se presentan, o son de ocurrencia poco probable, tengan igual o más probabilidad



de ocurrencia según dicho modelo. Debido a esto se decidió llevar a cabo análisis más detallado sobre las características de las muestras estudiadas, con lo cual se llegó a una propuesta un tanto más atractiva del comportamiento de la fuente de información. Ya que para este nuevo modelo propuesto se toman en cuenta las relaciones o influencias que tienen los símbolos que ocurren en los que están por ocurrir, es decir, las probabilidades de los símbolos en cada uno de los 17 alfabetos ya no son valores fijos, sino que están condicionados al evento o conjunto de eventos que ocurrieron antes que él. Además se propone un modelo de la fuente mediante cadenas de Markov que hace fácil detectar aquellas combinaciones que no son válidas según el comportamiento de las muestras estudiadas. Esto le da un valor agregado en comparación al modelo 1, ya que gracias a esto en un futuro puede ser posible detectar errores en base a estas relaciones y validación de combinaciones.

### **5.2.2 Resultados del análisis.**

En esta parte del trabajo se presentan mediante tablas los resultados obtenidos del estudio de las muestras obtenidas de la fuente. Se presenta posición por posición como se van creando vínculos y dependencias entre los distintos valores que pueden ir ocurriendo al momento de formar un NIV.

De igual forma que en el modelo anterior, la primera posición no aporta información ya que es conocida con anterioridad por lo cual no se toma en cuenta ya que su probabilidad de ocurrencia siempre será 1.

#### **Posición 2**

Según el análisis hecho esta posición puede tomar valores dentro del siguiente alfabeto  $A_2 \{ A, B, C, D, F, G, H, N, T, V \}$  y debido a que es el inicio del NIV (sin tomar en cuenta el primer dígito) su probabilidad de ocurrencia no está condicionada. Al contrario es de ésta posición donde se empieza a condicionar la formación del Número de Identificación Vehicular creado por el Modelo 2 propuesto.

*TABLA 5.2.1 Posición 2 Modelo 2.*

P2	P1	Probabilidad
N	3	57.53
G	3	20.97
V	3	10.22
F	3	3.23

P2	P1	Probabilidad
D	3	2.69
C	3	2.15
H	3	1.08
T	3	1.08

P2	P1	Probabilidad
A	3	0.54
B	3	0.54

### Posición 3

La tercera posición puede tomar valores de  $A_3 \{1,2,4,6, A, B, C, D, E, G, K, M, N, T, W, Y, Z\}$ . La probabilidad de estos símbolos depende de la combinación que exista en las posiciones 1 y 2. Dicha probabilidad se muestra en la

TABLA 5.2.2 en donde la primera columna indica el símbolo que puede ocupar la posición 3 del NIV, la segunda columna indica las posibles combinaciones que puede haber antes de dicho símbolo, y la última columna indica la probabilidad de ocurrencia del símbolo según la combinación que haya ocurrido en las posiciones anteriores.

*TABLA 5.2.2 Posición 3 Modelo 2.*

P3	Si P1-P2	Probabilidad
1	3G	30.77
	3N	93.46
2	3G	5.13
4	3C	50
	3D	60
6	3C	25
	3D	40
	3N	6.54

P3	Si P1-P2	Probabilidad
A	3F	33.33
	3H	50
B	3G	10.26
C	3G	20.51
D	3F	33.33
E	3F	16.67
G	3H	50
K	3A	100

P3	Si P1-P2	Probabilidad
M	3T	100
N	3B	100
	3G	28.21
T	3F	16.67
	3G	2.56
W	3V	100
Y	3G	2.56
Z	3C	25

#### Posición 4

La cuarta posición puede tomar valores de  $A_4 \{1, A, B, C, D, E, F, G, H, J, K, L, M, N, P, R, S, T, W, Y\}$ . La probabilidad de estos símbolos depende de la combinación que exista en las posiciones 1, 2 y 3. Dicha probabilidad se muestra en la TABLA 5.2.3 en donde la primera columna indica el símbolo que puede ocupar la posición 4 del NIV, la segunda columna indica las posibles combinaciones que puede haber antes de dicho símbolo, y la última columna indica la probabilidad de ocurrencia del símbolo según la combinación que haya ocurrido en las posiciones anteriores.

TABLA 5.2.3 Posición 4 Modelo 2.

P4	Si P1-P3	Probabilidad	P4	Si P1-P3	Probabilidad	P4	Si P1-P3	Probabilidad
1	3VW	5.26	F	3FA	100	M	3GB	25
A	3N1	1		3GN	9.09		3HA	100
B	3BN	100		3VW	5.26	N	3GC	12.5
	3N1	3	G	3VW	5.26		3GT	100
	3VW	5.26	H	3VW	10.53	P	3C4	100
C	3GN	9.09	J	3AK	100		3D4	100
	3HG	100		3G2	100		3VW	5.26
	3N1	7		3TM	100	R	3CZ	100
D	3GN	9.09		3VW	10.53		3VW	21.05
	3N6	100	K	3FD	50	S	3G1	50
E	3FT	100		3FE	50		3VW	5.26
	3GC	87.5		3GB	75	T	3G1	50
	3GN	72.73	L	3FD	50	W	3D6	100
	3GY	100		3FE	50	Y	3C6	100
	3N1	89		3VW	5.26		3VW	21.05

#### Posición 5

La quinta posición puede tomar valores de  $A_5 \{2, 7, A, B, C, D, E, F, G, K, L, M, N, P, S, U, V, W, X\}$ . La probabilidad de estos símbolos depende de la combinación que exista en las posiciones 1 a la 4. Dicha probabilidad se muestra en la TABLA 5.2.4 en donde la primera columna indica

el símbolo que puede ocupar la posición 5 del NIV, la segunda columna indica las posibles combinaciones que puede haber antes de dicho símbolo, y la última columna indica la probabilidad de ocurrencia del símbolo según la combinación que haya ocurrido en las posiciones anteriores.

TABLA 5.2.4 Posición 5 Modelo 2.

P5	Si P1-P4	Probabilidad
2	3GTN	100
7	3GBM	100
A	3G1T	66.67
	3GND	100
B	3BNB	100
	3G2J	100
	3N1A	100
	3N1C	42.86
	3N1E	98.88
C	3G1T	16.67
	3GBK	100
	3GCE	71.43
	3GNE	87.5
	3GNF	100
	3N1B	100
D	3C4P	100
	3C6Y	100
	3N1E	1.12
	3N6D	100
E	3CZR	100

P5	Si P1-P4	Probabilidad
F	3FDK	100
	3FDL	100
	3FEK	100
	3FEL	100
	3FTE	100
	3G1S	100
G	3AKJ	100
	3D4P	100
	3VWH	100
	3VWJ	50
	3VWR	25
K	3GCE	28.57
	3GCN	100
	3GNE	12.5
	3GYE	100
	3N1C	14.29
L	3GNC	100
M	3HAM	100
	3HGC	100

P5	Si P1-P4	Probabilidad
N	3D6W	100
	3N1C	42.86
	3VWG	100
N	3VWP	100
P	3FAF	100
S	3VWF	100
U	3TMJ	100
V	3VW1	100
	3VWB	100
	3VWR	75
	3VWS	100
	3VWY	100
W	3VWJ	50
	3VWL	100
X	3G1T	16.67

### Posición 6

La sexta posición puede tomar valores de  $A_6 \{0,1,2,3,4,5,6,7,9, A, B, C, H, L\}$ . La probabilidad de estos símbolos depende de la combinación que exista en las posiciones 1 a la 5. Dicha probabilidad se muestra en la TABLA 5.2.5 en donde la primera columna indica el símbolo que puede ocupar la posición 6 del NIV, la segunda columna indica las posibles combinaciones que puede haber antes de

dicho símbolo, y la última columna indica la probabilidad de ocurrencia del símbolo según la combinación que haya ocurrido en las posiciones anteriores.

*TABLA 5.2.5 Posición 6 Modelo 2.*

P6	Si P1-P5	Probabilidad	P6	Si P1-P5	Probabilidad	P6	Si P1-P5	Probabilidad
0	3VW1V	100	2	3G1SF	33.33	5	3D4PG	50
	3VWBV	100		3GNDA	100		3D6WN	100
	3VWVRV	33.33		3N6DD	28.57		3G1SF	16.67
1	3BNBB	100		3VWSV	100		3G1TA	100
	3FTEF	100	3	3N1ED	100		3G1TC	100
	3G2JB	50		3FDKF	100		3G1TX	100
	3GCEC	100		3FEKF	100		3G2JB	50
	3GCEK	100		3GBKC	100		3GNCL	100
	3GNEC	100		3N1CB	33.33		3HGCM	100
	3GNEK	100		3N1CK	100		3N1CB	66.67
	3GNFC	100		3N1EB	100	6	3D4PG	50
	3N1BC	100		4	3CZRE		100	3G1SF
	3N6DD	71.43	3FAFP		100		3GYEK	100
	3VWGN	100	3FDLF		100		3N1AB	100
	3VWHG	100	3FELF		100		3TMJU	100
	3VWJG	100	3VWVRV		66.67	3VWLW	100	
	3VWJW	100	3VWYV		100	7	3N1CN	100
	3VWPN	100	9	A	3VWFS		100	
3VWRG	100	3GCNK			100			
		3GTN2			100			
		3HAMM			100			
		3C6YD			100			
		3C4PD			100			
		3GBM7			100			
		3AKJG			100			

### Posición 7

La séptima posición puede tomar valores de  $A_7 \{1, 2, 3, 4, 5, 6, 7, 8, 9, A, C, E, F, J\}$ . La probabilidad de estos símbolos depende de la combinación que exista en las posiciones 1 a la 6. Dicha probabilidad se muestra en la TABLA 5.2.6 en donde

la primera columna indica el símbolo que puede ocupar la posición 7 del NIV, la segunda columna indica las posibles combinaciones que puede haber antes de dicho símbolo, y la última columna indica la probabilidad de ocurrencia del símbolo según la combinación que haya ocurrido en las posiciones anteriores.

TABLA 5.2.6 Posición 7 Modelo 2.

P7	Si P1-P6	Probabilidad	P7	Si P1-P6	Probabilidad	P7	Si P1-P6	Probabilidad
1	3G1SF5	100	3	3GNCL5	100	A	3C6YDB	100
	3G1SF2	100		3GNDA2	100		3FAFP4	100
	3G1SF6	100		3N6DD1	20		3G1TA5	100
	3GBM7H	100	4	3GBKC3	100		3HAMMA	100
	3N1AB6	100		3GCEC1	100		3N1BC1	33.33
	3N1BC1	66.67		3GCEK1	100		3N1CN7	100
	3N1CB3	100		3N6DD1	40		3VVLW6	100
	3N1CB5	100	5	3N6DD2	100	C	3AKJGL	100
	3N1EB3	100	6	3D6WN5	66.67		3G1TC5	100
	3N1ED3	100		3FDKF3	100		3N1CK3	100
	3VWFS7	100		3FDLF4	100	E	3D6WN5	33.33
	3VWGN1	100		3FEKF3	100		3GCNK9	100
	3VWHG1	100		3FELF4	100		3GTN29	100
	3VWJG1	100		7	3GNEC1	100	F	3C4PDC
	3VWJW1	100	3GNFC1		100	3D4PG5		100
	3VWPN1	100	3HGCM5		100	3D4PG6		100
3VWRG1	100	8	3FTEF1	100	J	3BNBB1	100	
2	3G1TX5	100	8	3CZRE4	100			
	3G2JB1	100	9	3VW1V0	100			
	3G2JB5	100		3VWBV0	100			
	3GNEK1	100		3VWRV0	100			
	3GYEK6	100		3VWRV4	100			
	3N6DD1	40		3VWSV2	100			
	3	3TMJU6		100	3VWYV4	100		

### Posición 8

La octava posición puede tomar valores de  $A_8 \{0,3,5,6,7,8,B,C,D,F,G,J,K,L,M,N,P,R,S,T,V,W,X\}$ . La probabilidad de

estos símbolos depende de la combinación que exista en las posiciones 1 a la 7. Dicha probabilidad se muestra en la TABLA 5.2.7 en donde la primera columna indica el símbolo que puede ocupar la posición 8 del NIV, la segunda columna indica las posibles combinaciones que puede haber antes de dicho símbolo, y la última columna indica la probabilidad de ocurrencia del símbolo según la combinación que haya ocurrido en las posiciones anteriores.

TABLA 5.2.7 Posición 8 Modelo 2.

P8	Si P1-P7	Probabilidad
0	3GCNK9E	100
	3GTN29E	100
3	3HGCM56	100
5	3N1CB31	100
	3N1EB31	2.27
6	3BNBB1J	100
	3G1TA5A	25
	3G1TX52	100
7	3CZRE48	100
	3GNCL53	100
8	3N1EB31	1.14
B	3D4PG5F	100
C	3D6WN56	50
	3GBM7H1	100
D	3D6WN56	50
	3N1AB61	100
	3N1CN7A	100
F	3G1TA5A	75
	3G1TC5C	100
	3G2JB12	100
	3G2JB52	100
G	3C4PDCF	100
	3GBKC34	100

P8	Si P1-P7	Probabilidad
J	3FAFP4A	100
	3GCEK14	50
	3GNFC16	100
	3VWLW6A	100
K	3AKJGLC	100
	3VWFS71	100
	3VWGN11	100
	3VWHG11	100
	3VWJG11	100
	3VWJW11	100
	3VWPN11	100
L	3VWRG11	100
	3FDKF36	100
M	3FEKF36	100
	3VW1V09	100
	3VWBV09	100
	3VWRV09	100
	3VWRV49	100
	3VWSV29	100
N	3VWYV49	100
	3GYEK62	100
	3TMJU62	100

P8	Si P1-P7	Probabilidad
P	3C6YDBA	100
	3FDLF46	100
	3FELF46	100
	3GNDA23	100
R	3HAMMAA	100
S	3N1BC11	100
	3N1BC1A	100
	3N1CB51	100
	3N1CK3C	100
	3N1EB31	96.59
	3N1ED31	100
	3N6DD12	100
T	3N6DD13	100
	3N6DD14	100
T	3D6WN5E	100
	3GCEC14	100
	3GCEK14	50
	3GNEC16	100
	3GNEK12	100
	3N6DD25	100
V	3D4PG6F	100
W	3FTEF17	100
X	3G1SF51	100
	3G1SF21	100
	3G1SF61	100

### Posición 9

La novena posición puede tomar valores de  $A_9 \{0,1,2,3,4,5,6,7,8,9, X\}$ . Al igual que en el modelo 1 las probabilidades de ocurrencia para los símbolos de esta posición se suponen constantes e iguales. Y a diferencia de las posiciones anteriores, la probabilidad de ocurrencia de los símbolos para esta posición no está condicionada por alguna otra combinación.

*TABLA 5.2.8 Posición 9 Modelo 2.*

Posición9	Probabilidad
0	9.091
1	9.091
2	9.091
3	9.091
4	9.091
5	9.091
6	9.091
7	9.091
8	9.091
9	9.091
X	9.091

### Posición 10

La décima posición puede tomar valores de  $A_{10} \{3,4,5,6,7,8,9, A, B, C, D\}$ . Al igual que en el modelo 1 las probabilidades de ocurrencia para los símbolos de esta posición se suponen constantes e iguales. Y a diferencia de las posiciones anteriores, la probabilidad de ocurrencia de los símbolos para esta posición no está condicionada por alguna otra combinación.



*TABLA 5.2.9 Posición 10 Modelo 2.*

Posición10	Probabilidad
3	9.091
4	9.091
5	9.091
6	9.091
7	9.091
8	9.091
9	9.091
A	9.091
B	9.091
C	9.091
D	9.091

### **Posición 11**

La décimo primera posición puede tomar valores de  $A_{11} \{G, K, L, M, S, T\}$ . La probabilidad de estos símbolos depende de la combinación que haya antes, estas combinaciones son de distintas longitudes, ya que algunos están condicionados por combinaciones que se dan en las primeras 2 posiciones y en otros casos se condiciona a combinaciones de hasta 5 dígitos, es decir de la posición 1 a la 5. La probabilidad para los símbolos de este alfabeto se muestra en la

TABLA 5.2.10 en donde la primera columna indica el símbolo que puede ocupar la posición 11 del NIV, la segunda columna indica las posibles combinaciones que puede haber antes de dicho símbolo, y la última columna indica la probabilidad de ocurrencia del símbolo según la combinación que haya ocurrido en las posiciones anteriores.

TABLA 5.2.10 Posición 11 Modelo 2.

P11	Si P1-P(n)	Probabilidad	P11	Si P1-P(n)	Probabilidad	P11	Si P1-P(n)	Probabilidad
G	3C6	100	M	3F	100	L	3B	100
	3CZ	100		3GB	100		3G1T	100
	3D6	100		3GC	25		3HA	100
	3GC	75		3T	100		3N1A	100
	3GNE	100		3V	100		3N1B	33
	3GNF	100	S	3A	100		3N1CB	67
	3GT	100		3G1S	100		3N1CK	100
	3GY	100		3G2	100		3N1CN	100
	3HG	100		3GNC	100		K	3N1B
T	3C4	100		3GND	100	3N1CB		33
	3D4	100	3GN	100	3N1E	100		
					3N6	100		

### Posición 12

La décimo segunda posición puede tomar valores de  $A_{12} \{0,1,2,3,4,5,6,7,8,9,A\}$ .

Al igual que en el modelo 1 las probabilidades de ocurrencia para los símbolos de esta posición se suponen constantes e iguales. Y a diferencia de las posiciones anteriores, la probabilidad de ocurrencia de los símbolos para esta posición no está condicionada por alguna otra combinación.

TABLA 5.2.11 Posición 12 Modelo 2.

P12	Probabilidad
0	9.091
1	9.091
2	9.091
3	9.091
4	9.091
5	9.091
6	9.091
7	9.091
8	9.091
9	9.091
A	9.091

### Posiciones 13 a 17

Las últimas cinco posiciones puede tomar valores de  $A_{13} \dots A_{17} \{0,1,2,3,4,5,6,7,8,9\}$ . Al igual que en el modelo 1 las probabilidades de ocurrencia para los símbolos de esta posición se suponen constantes e iguales. Y a diferencia de las posiciones anteriores, la probabilidad de ocurrencia de los símbolos para esta posición no está condicionada por alguna otra combinación.

*TABLA 5.2.12 Posiciones 13-17 Modelo 2.*

<b>P13- P17</b>	<b>Probabilidad</b>
0	10
1	10
2	10
3	10
4	10
5	10
6	10
7	10
8	10
9	10

### Tabla de Alfabetos.

En la

TABLA 5.2.13 se pueden ver de manera gráfica los alfabetos correspondientes a cada una de las posiciones. Cada columna corresponde a un alfabeto, y el título de las columnas corresponde a cada una de las posiciones.

TABLA 5.2.13 Posibles valores según la posición en el NIV.

Número de Identificación Vehicular (NIV)																
1ro	2do	3ro	4to	5to	6to	7mo	8vo	9no	10mo	11mo	12mo	13ro	14to	15to	16to	17mo
3	A	1	1	2	0	1	0	0	3	G	0	0	0	0	0	0
	B	2	A	7	1	2	3	1	4	K	1	1	1	1	1	1
	C	4	B	A	2	3	5	2	5	L	2	2	2	2	2	2
	D	6	C	B	3	4	6	3	6	M	3	3	3	3	3	3
	F	A	D	C	4	5	7	4	7	S	4	4	4	4	4	4
	G	B	E	D	5	6	8	5	8	T	5	5	5	5	5	5
	H	C	F	E	6	7	B	6	9		6	6	6	6	6	6
	N	D	G	F	7	8	C	7	A		7	7	7	7	7	7
	T	E	H	G	9	9	D	8	B		8	8	8	8	8	8
	V	G	J	K	A	A	F	9	C		9	9	9	9	9	9
		K	K	L	B	C	G	X	D		A					
		M	L	M	C	E	J									
		N	M	N	H	F	K									
		T	N	P	L	J	L									
		W	P	S			M									
		Y	R	U			N									
		Z	S	V			P									
			T	W			R									
			W	X			S									
			Y				T									
							V									
							W									
							X									

### 5.2.3 Cálculo de probabilidad.

Para el cálculo de la probabilidad, este modelo toma en cuenta la condicional que existe entre la ocurrencia de un símbolo y sus antecesores. Es decir, propone describir el comportamiento de la fuente de información mediante el principio de funcionamiento de cadenas de Markov. Se propone un árbol de probabilidades en donde se van estableciendo relaciones y dependencias de los símbolos conforme van ocurriendo hasta que se logra formar un NIV.

La probabilidad de ocurrencia del símbolo  $A_k(s_k)$  está dada por la posición que ocupa en el NIV, y la cadena de símbolos que le precede  $\{A_{(k-1)}(s_{(k-1)}), \dots, A_1(1)\}$ .

$$p(A_k(s_k)|A_{(k-1)}(s_{(k-1)}), \dots, A_1(1))$$

$A_k$  = Alfabeto para la posición  $k$ .

$n_k$  = Cantidad de elementos de  $A_k$ .

$s_k$  = Símbolo de  $A_k$  (depende de  $n_k$  no del valor de  $k$  ).

$$1 \leq s_k \leq n_k$$

para

$$k \in \{1,2, \dots, 17\}$$

La probabilidad de ocurrencia de un NIV está dada por:

$$p(\text{NIV}) = \prod_{k=1}^{17} p(s_k) \quad (5.3)$$

Los resultados de estas probabilidades y dependencias, obtenidos después de realizar el estudio y análisis de las muestras representativas de la fuente, son los que se plasman en las tablas de la sección 5.2.2. Y que dado un NIV nos ayudan a obtener la probabilidad de ocurrencia del mismo, así como su validez, en base a este modelo propuesto.

#### 5.2.4 Árbol de probabilidades.

Gracias a la información obtenida y presentada en la sección 5.2.2 se propone un árbol de probabilidades en representación del modelo matemático. Dicho árbol consta de 17 niveles, en el que cada nivel hace referencia a una posición del NIV en el mismo orden. Es decir, el primer nivel corresponde a la posición uno del NIV, el segundo nivel corresponde a la dos, y así sucesivamente.

En la Figura 5.2.1 se muestra de manera gráfica y resumida. Se puede observar el desarrollo del árbol de probabilidades hasta alcanzar el último nivel correspondiente a la posición número diecisiete. En dicho diagrama se utiliza la nomenclatura  $P_k(e)$  en donde  $P_k$  hace referencia a la posición del NIV, y la  $e$  al

elemento que exista en el alfabeto correspondiente a dicha posición. Por ejemplo; si  $A_9 \{0,1,2,3,4,5,6,7,8,9, X\}$  entonces para este caso  $\ell$  puede tomar valores de  $e = 1, 2, 3, \dots, 11$ , por lo tanto  $P_9(11)$  hace referencia al onceavo elemento del alfabeto  $A_9$  que sería  $X$ , quedando entonces  $P_9(11) = X$ .

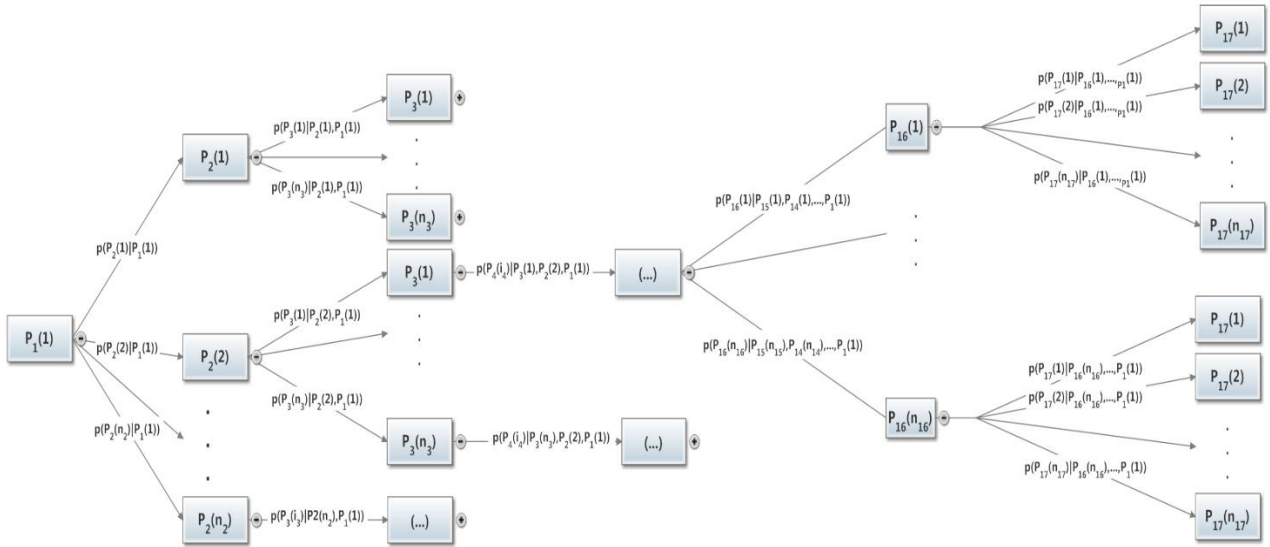


Figura 5.2.1 Diagrama de árbol de probabilidades.

Para fines de apreciación se dividió el diagrama antes mencionado en dos partes, la Parte A que representa el inicio de la formación del NIV, y la Parte B que representa la aparición de los últimos dígitos que constituyen el NIV.

En la Parte A se puede observar como el primer nivel está compuesto únicamente de un elemento, haciendo alusión al dígito '3' que para este trabajo se considera que es conocido y siempre ocupara dicha posición. De aquí para pasar al carácter que ocupa la segunda posición se tienen varias opciones, en el diagrama se puede observar que existen las posibilidades desde  $P_2(1)$  hasta  $P_2(n_2)$ , en este caso se utiliza la nomenclatura  $n_2$  para hacer referencia al número de elementos del alfabeto para la posición 2. Por lo tanto el subíndice en  $n_k$  puede tomar valores de  $k = 1, 2, 3, \dots, 17$ . En el caso de  $P_2(1)$  aunque la ramificación dice que su ocurrencia está condicionada a lo ocurrido en  $P_1(1)$ , en realidad no es así del todo,



ya que como se mencionó anteriormente se tiene completa certeza de lo ocurrido en  $P_1(1)$  por lo cual para este caso la probabilidad de que  $P_2(1)$  ocurra, está dada por la probabilidad de ocurrencia que tenga el primer elemento del alfabeto 2 en dicho alfabeto. Por otro lado al llegar al tercer nivel en donde  $P_3(1)$  aparece en un inicio, se puede ver que su aparición está condicionada por la combinación que haya salido antes. Por ejemplo si en la posición dos ocurrió  $P_2(1)$  y anteriormente había ocurrido  $P_1(1)$  entonces la probabilidad de que  $P_3(1)$  ocurra está dada por  $p(P_3(1)|P_2(1),P_1(1))$ . En cambio, si en la posición dos ocurrió  $P_2(2)$  y en la posición uno ocurrió  $P_1(1)$  la probabilidad de ocurrencia en la tercer posición del mismo símbolo  $P_3(1)$  es diferente y está dada por  $p(P_3(1)|P_2(2),P_1(1))$ .

De igual forma se sigue este mismo procedimiento para la creación del dato final (NIV) de la fuente de información, de acuerdo a este modelo, y se llega a la parte final del árbol (Parte B) que se ve representada en la Figura 5.2.3. Aquí se ve representada de manera gráfica como se toma la elección del último símbolo que aparecerá en el NIV. Esto basado en la combinación de símbolos que se ha creado con anterioridad desde la primera posición hasta la decimosexta.

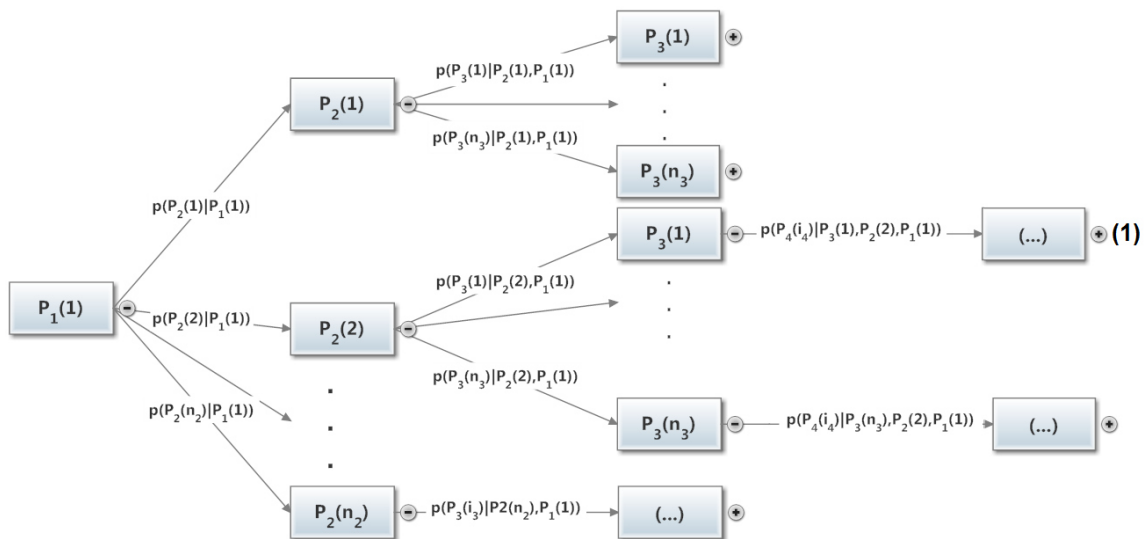


Figura 5.2.2 Parte A

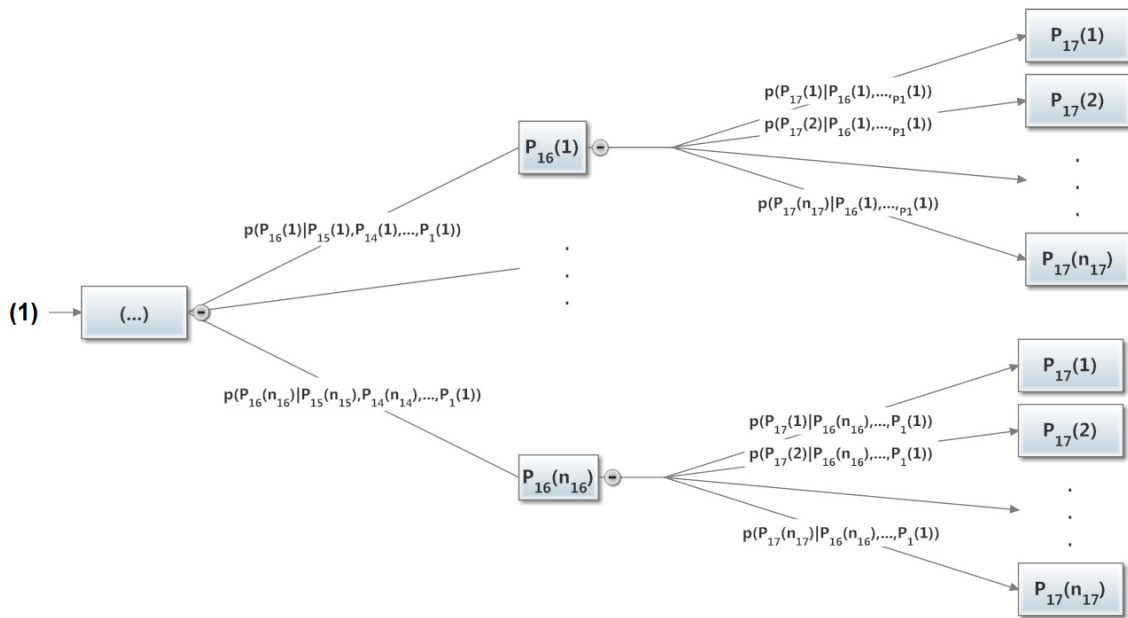


Figura 5.2.3 Parte B

## 5.2.5 Entropía

De manera general no es posible conocer la entropía de una fuente física, por lo cual se hace únicamente una estimación de la misma. Esta estimación depende de las suposiciones que se hacen en los modelos presentados de este trabajo para representar la estructura de la secuencia de la fuente. Al obtener una estimación de la entropía. El resultado de la entropía indica un promedio de la cantidad mínima de bits, bajo el mejor esquema de codificación, que son necesarios para codificar dicha fuente de información.

El conjunto de símbolos  $\mathcal{A}$  llamado "*alfabeto*" de la fuente, y a los símbolos se les llama "*letras*". De modo que, para una fuente  $\mathcal{S}$  con alfabeto  $\mathcal{A} = \{1, 2, \dots, m\}$ , donde  $m = 33$ , y que genera una secuencia  $\{X_1, X_2, \dots\}$ , la entropía esta dada por

$$H(\mathcal{S}) = \lim_{n \rightarrow \infty} \frac{1}{n} G_n \quad (5.4)$$

donde

$$G_n =$$

$$\sum_{i_1=1}^{i_1=m} \sum_{i_2=1}^{i_2=m} \dots \sum_{i_n=1}^{i_n=m} P(X_1 = i_1, X_2 = i_2, \dots, X_n = i_n) \log P(X_1 = i_1, X_2 = i_2, \dots, X_n = i_n) \quad (5.5)$$

y  $\{X_1, X_2, \dots, X_n\}$ , es una secuencia de longitud  $n$  de la fuente, donde  $n = 17$  [16].

Este cálculo es importante ya indica la cantidad mínima de bits en promedio que se necesitan para codificar la información de la fuente de información. Es decir, la entropía sirve como parámetro para comparar los resultados obtenidos luego de proponer un método de codificación y compresión. Los métodos de codificación y compresión estarán basados en el Modelo 2 propuesto ya que es el Modelo que más se asemeja a la fuente de información estudiada.

Mediante el árbol de probabilidades y las tablas de dependencia presentadas en 5.2.2 se procede a hacer el cálculo de la entropía de la fuente según el Modelo 2. Para esto se realiza el cálculo matemáticos por secciones para fines prácticos, es decir, se calcula la entropía de distintas partes del árbol probabilístico y al final se suman para obtener la entropía de la fuente de información.

### Posiciones 1 a la 8

Para obtener esta entropía se lleva a cabo una pequeña metodología que consiste en lo siguiente; primero se obtienen las combinaciones válidas en base al árbol de probabilidades, esto se hace ya que a simple vista y basándose únicamente en los alfabetos, el número de combinaciones que se pueden formar con ellos es relativamente extenso, pero al momento de verificar condicionalidades y dependencias, la mayor parte de estas posibles combinaciones tienen una probabilidad 0, es decir, no son combinaciones válidas según el modelo matemático que se está proponiendo. Una vez hecho esto, se conservan únicamente aquellas combinaciones válidas así como su probabilidad de ocurrencia y se procede al cálculo de la entropía. Esto es, se extrapola el

problema a otro escenario en el que se considera cada combinación (NIV) como un símbolo de otra fuente de información. Asumiendo que en esta nueva fuente los símbolos tienen independencia, y tomando las probabilidades de cada uno de ellos se procede al cálculo de la entropía para esta parte del modelo. Con esto se obtiene el mismo resultado de aplicar la ecuación (5.5) con los datos que se tienen.

Como resultado para estas primeras 8 posiciones se obtiene que la entropía de la fuente desde la posición 1 hasta la 8 es de 4.17 bits. Cabe mencionar que el primer carácter no aporta información por lo cual no se requieren bits para codificarlo por lo cual ésta entropía corresponde al cálculo que se hizo con las combinaciones formadas desde la segunda posición hasta la octava.

### **Posiciones 9, 10 y 12**

La siguiente agrupación de posiciones que se elige es la que corresponde a las posiciones 9, 10 y 12. Esto ya que en términos probabilísticos, y gracias a las suposiciones hechas, los símbolos que ocupan estas posiciones tienen independencia ya que no dependen de ningún otro valor para su aparición. Además son idénticamente distribuidos y pueden tomar 11 valores distintos valores. El cálculo de la entropía para esta sección es de 10.3782948559118 bits.

### **Posiciones 13-17**

La siguiente agrupación que se hace es la que corresponde a las posiciones 13, 14, 15,16 y 17. Esto ya que, de igual forma que la agrupación anterior, en términos probabilísticos y gracias a las suposiciones hechas, los símbolos que ocupan estas posiciones tienen independencia ya que no dependen de ningún otro valor para su aparición. Además son idénticamente distribuidos y pueden tomar 10 valores distintos valores. El cálculo de la entropía para esta sección es de 16.609640474418 bits.

### Posición 11

Por último se calcula la entropía para la onceava posición. Esta posición puede tomar 6 valores distintos, los cuales pueden depender tanto como de 2 hasta 5 valores anteriores. Si se consideraran estos 6 valores independientes y con una misma distribución de probabilidad, se tendría una entropía de 2.585 bits. Ahora, haciendo un análisis de las probabilidades de ocurrencia de los valores que condicionan a los símbolos en esta posición, se obtiene que la entropía tomando en cuenta la distintas probabilidades es de 2.0946 bits.

### Entropía total del NIV.

Al hacer una suma de los resultados de entropía obtenidos en esta sección, se tiene que en promedio la cantidad mínima de bits necesaria para realizar una codificación sin pérdidas de la fuente de información es de 34 bits/NIV. Lo cual nos proporciona la cota mínima que se puede alcanzar y en base a ella comparar los resultados obtenidos al momento de proponer una solución de codificación y compresión para los datos de la fuente en estudio.

En la TABLA 5.2.14 Entropía del NIV. Se pueden observar los distintos resultados obtenidos al momento de calcular la entropía en cada una de las secciones y al final la entropía total al sumar cada una de ellas.

*TABLA 5.2.14 Entropía del NIV.*

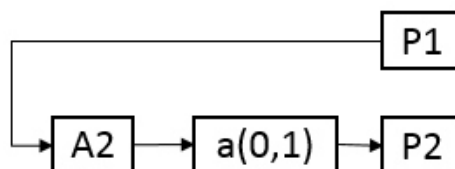
Entropía	Posiciones
4.17	1-8
10.3782	9 ,10 y 12
16.6096	13-17
2.0946	11
33.2524	NIV

### 5.2.6 Simulador de datos

Tomando como referencia los resultados del análisis obtenidos en la sección 5.2.2 se realizó un programa computacional cuyo fin es simular datos de la fuente

de información. El principio de dicho programa es que posición por posición vaya creando un NIV en base a las probabilidades y dependencias observadas en los datos reales, en otras palabras que siga el árbol de probabilidades propuesto conforme vayan apareciendo los símbolos para cada posición. Con esto se trata de poder generar una mayor cantidad de datos para futuras pruebas en caso de que sean necesarios.

El programa empieza creando el dígito '3' en la primera posición (P1) del nuevo NIV a crear. La segunda posición no está condicionada por lo que haya ocurrido antes, los símbolos de su alfabeto tienen una probabilidad fija ya calculada y que puede ser encontrada en la sección 5.2.2. Ahora, cada uno de los símbolos abarcará una región del espacio comprendido entre (0,1) de acuerdo a su probabilidad. Las regiones se irán formando según la probabilidad de ocurrencia de cada símbolo en un orden de mayor a menor. Por ejemplo, si el símbolo más probable tuviera una probabilidad de ocurrencia del 60%, la primera región correspondería a este símbolo y ocuparía el espacio (0,0.5). Habiendo explicado esto lo siguiente en el programa es generar un valor aleatorio entre (0,1) que servirá para decidir que símbolo irá en la segunda posición (P2) del NIV. Ocupando esta segunda posición aquél símbolo cuya región sea donde haya caído el valor aleatorio generado previamente.



*Figura 5.2.4 Inicio del programa. Selección de P2.*

Teniendo listos los primeros dos valores del NIV, (I) se hace la combinación de los elementos se tienen hasta el momento, (II) en base a esta combinación se hace un filtrado al alfabeto de la posición siguiente para obtener aquel que sea válido para la combinación de valores que se acaba de crear. La elección de este nuevo alfabeto válido se hace acorde al árbol de probabilidades y dependencias del modelo propuesto. (III) Una vez elegido el alfabeto adecuado, cuyos símbolos

tienen una probabilidad normalizada, cada uno de los símbolos abarcará una región del espacio comprendido entre (0,1) de acuerdo a su probabilidad. Las regiones se irán formando según la probabilidad de ocurrencia de cada símbolo en un orden de mayor a menor como ya se explicó anteriormente. (IV) Se genera un valor aleatorio entre (0,1) y (V) se decide que símbolo irá en la siguiente posición según la región donde haya caído el valor aleatorio generado. Una vez hecho esto se vuelve a repetir el proceso desde el Paso I hasta que al final la combinación que se tenga sea la de los 17 dígitos que conforman el NIV.

En la Figura 5.2.5 se muestra el diagrama de bloques que representa el funcionamiento básico del simulador de datos que se elaboró.

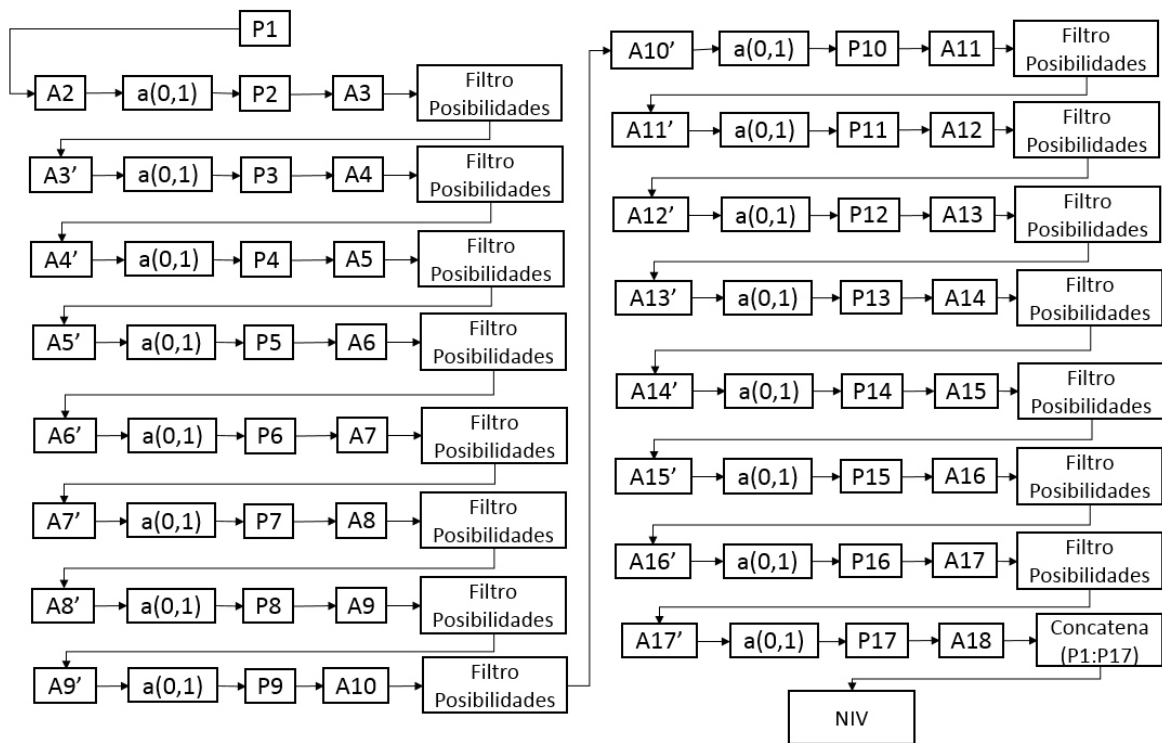


Figura 5.2.5 Diagrama de bloques del simulador de datos.

En la Figura 5.2.6 se puede ver el diagrama de flujo del simulador de datos, cuyo funcionamiento expliqué en esta sección y consiste básicamente en llevar al

lenguaje de programación el Modelo 2 propuesto para la simulación de la fuente de información, el cual está basado en el funcionamiento de cadenas de Markov.

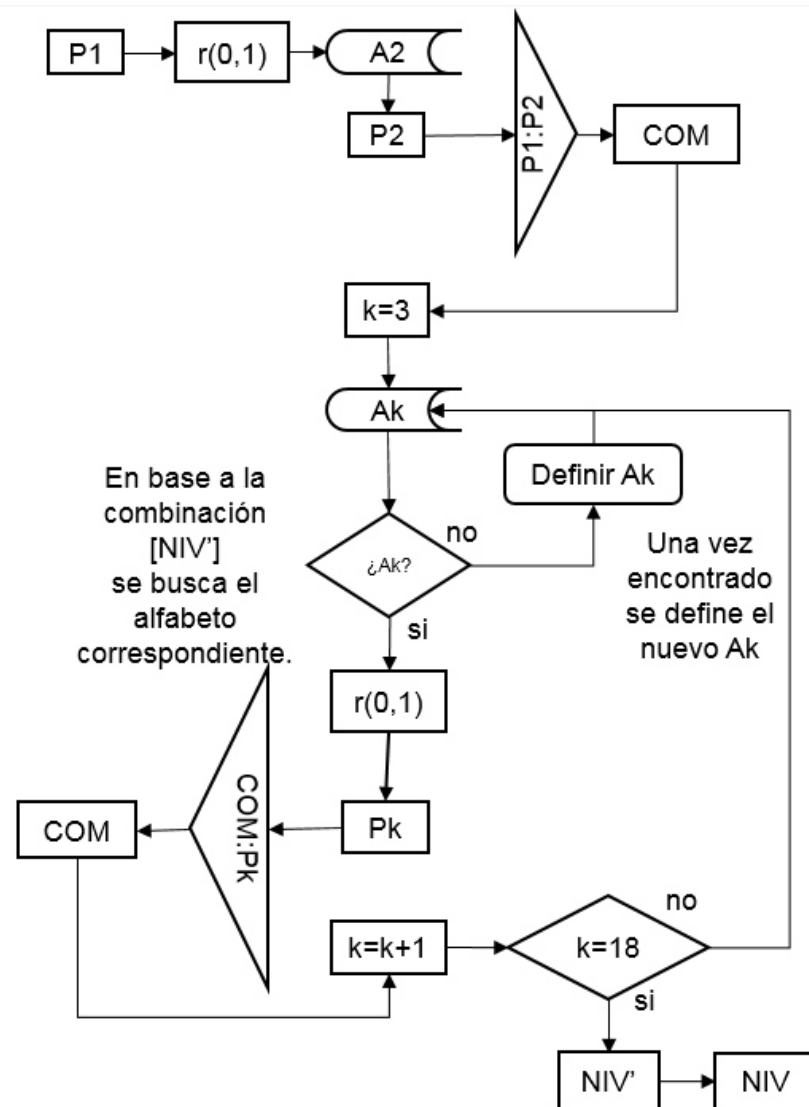


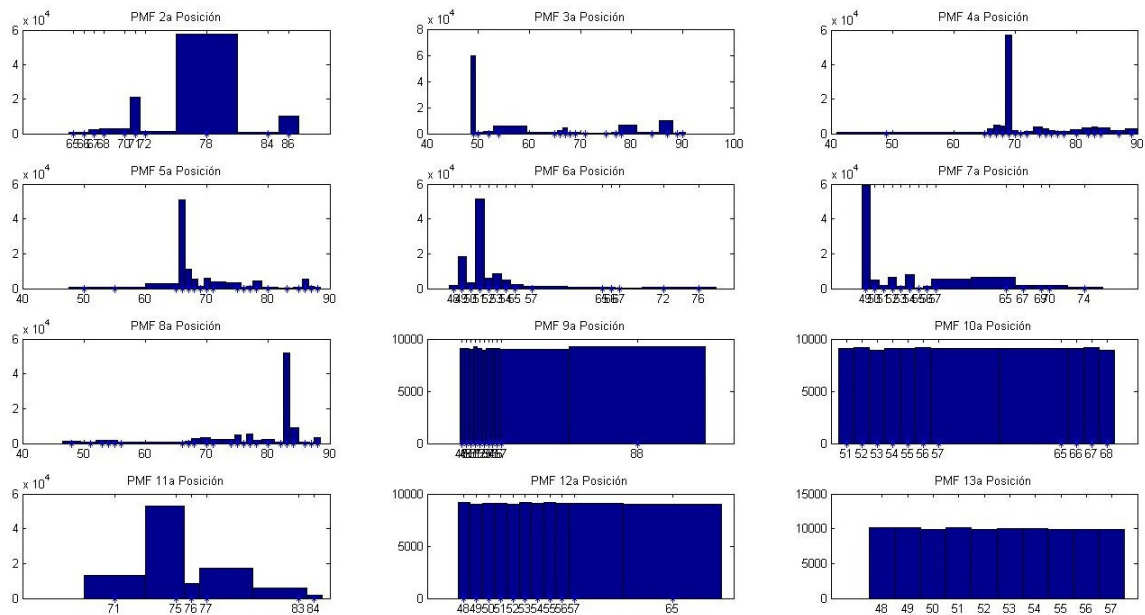
Figura 5.2.6 Diagrama de flujo del simulador.

### 5.2.7 Obtención de la Función de Masa de Probabilidad (PMF)

Una vez teniendo el simulador de la fuente, explicado en la sección anterior, se generó una serie cien mil muestras de las cuales se obtuvo la función de masa de probabilidad (PMF) de cada una de las posiciones del NIV. De acuerdo a lo estipulado en [18] una muestra de hasta un 4% del total de una población puede



ser tomada como muestra representativa de dicha población. Para la fuente de información en estudio las muestras generadas en teoría representan el 21.34%. Por lo cual se asume que esta serie de muestras representa a la población de interés. Los resultados del análisis de las muestras generadas mediante el simulador de datos, se muestran en la Figura 5.2.7 en la cual se tienen las funciones de masa de probabilidad desde la posición 2 hasta la posición 13 del NIV.



*Figura 5.2.7 Función de Masa de Probabilidad obtenida de las muestras de la simulación.*

Los valores del eje horizontal en las gráficas de la Figura 5.2.7 representan el valor en código ASCII de los símbolos de cada posición. Por lo tanto para la interpretación de dicha figura se toma como referencia la información presentada en la

TABLA 5.2.15.

*TABLA 5.2.15 Relación de valores del eje horizontal.*

Eje X	Símbolo	Eje X	Símbolo	Eje X	Símbolo
48	0	66	B	78	N
49	1	67	C	80	P
50	2	68	D	82	R
51	3	69	E	83	S
52	4	70	F	84	T
53	5	71	G	85	U
54	6	72	H	86	V
55	7	74	J	87	W
56	8	75	K	88	X
57	9	76	L	89	Y
65	A	77	M	90	Z

A continuación se muestran nuevamente las funciones de masa de probabilidad para cada una de las posiciones en las cuales se observa claramente una distribución no uniforme. Estas posiciones serían la 2<sup>a</sup>, 3<sup>a</sup>, 4<sup>a</sup>, 5<sup>a</sup>, 6<sup>a</sup>, 7<sup>a</sup>, 8<sup>a</sup> y 11<sup>a</sup>. Y las podemos observar a continuación desde la Figura 5.2.8 hasta la Figura 5.2.15.

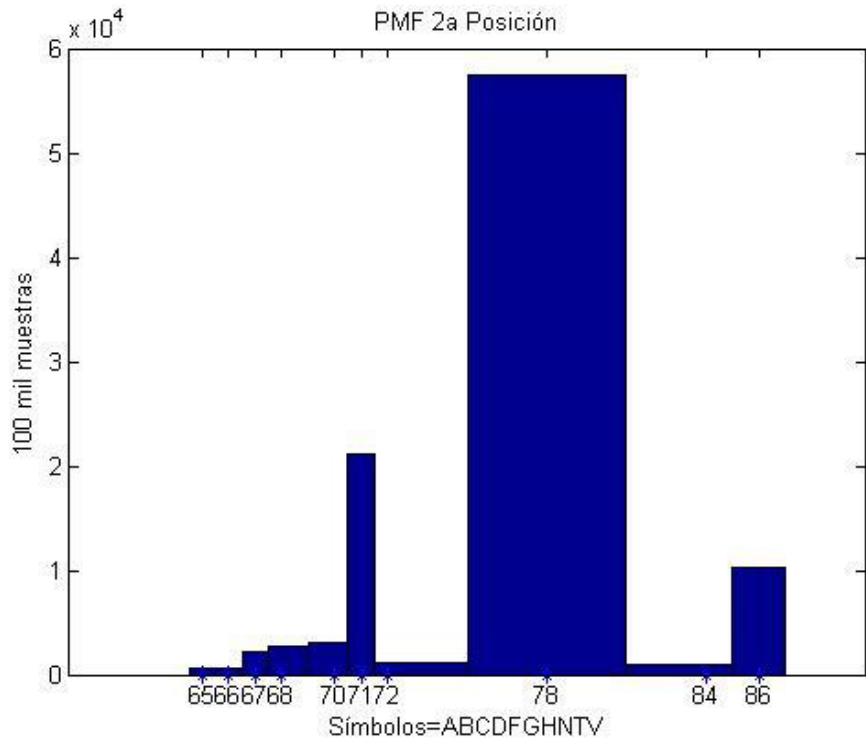


Figura 5.2.8 PMF 2ª Posición.

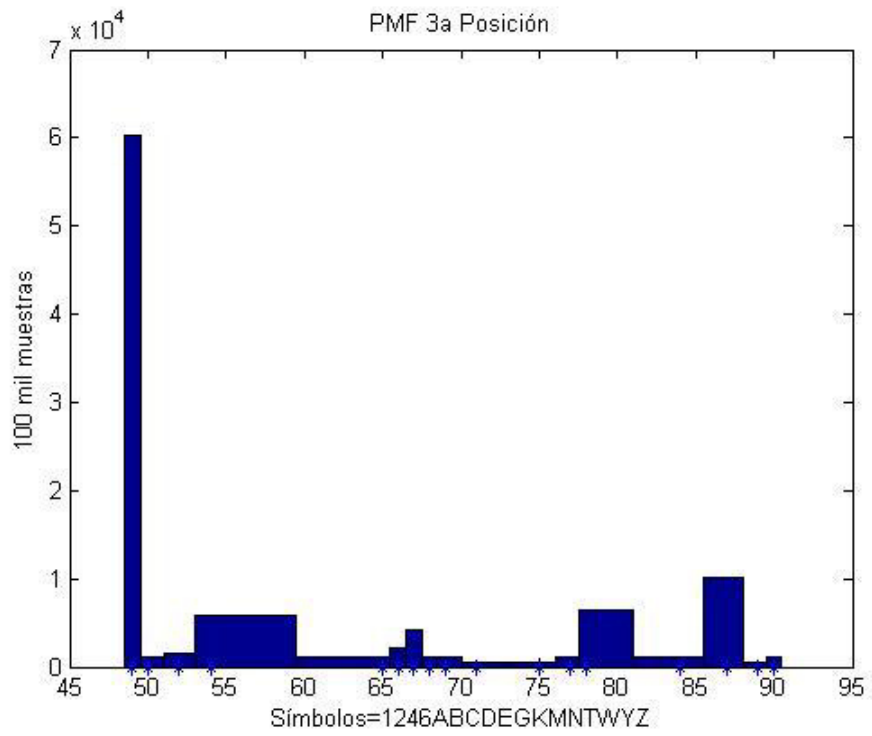


Figura 5.2.9 PMF 3ª Posición.

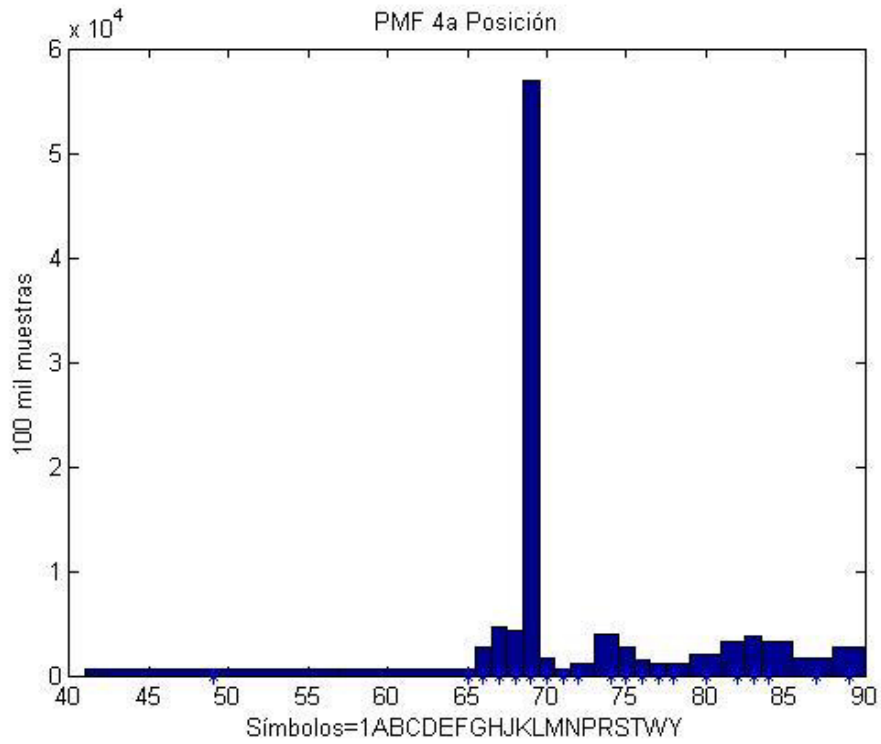


Figura 5.2.10 PMF 4ª Posición.

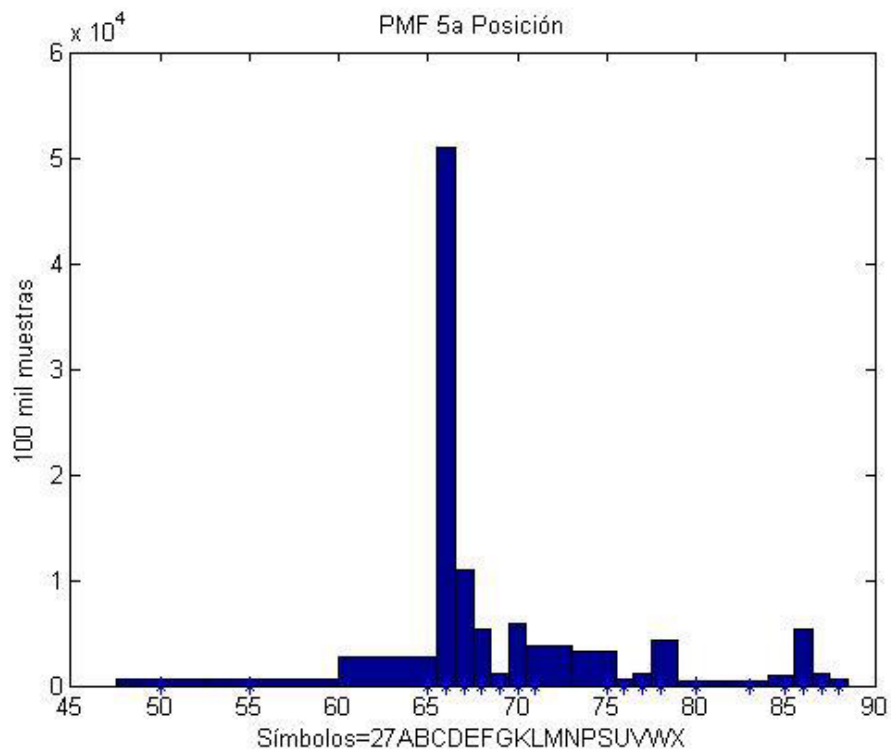


Figura 5.2.11 PMF 5ª Posición.

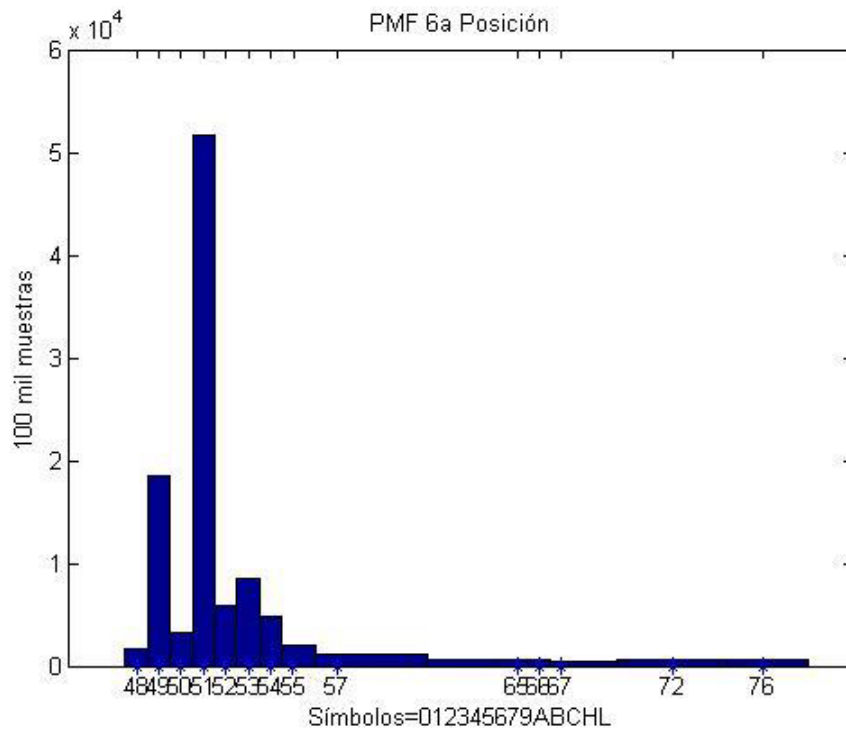


Figura 5.2.12 PMF 6ª Posición.

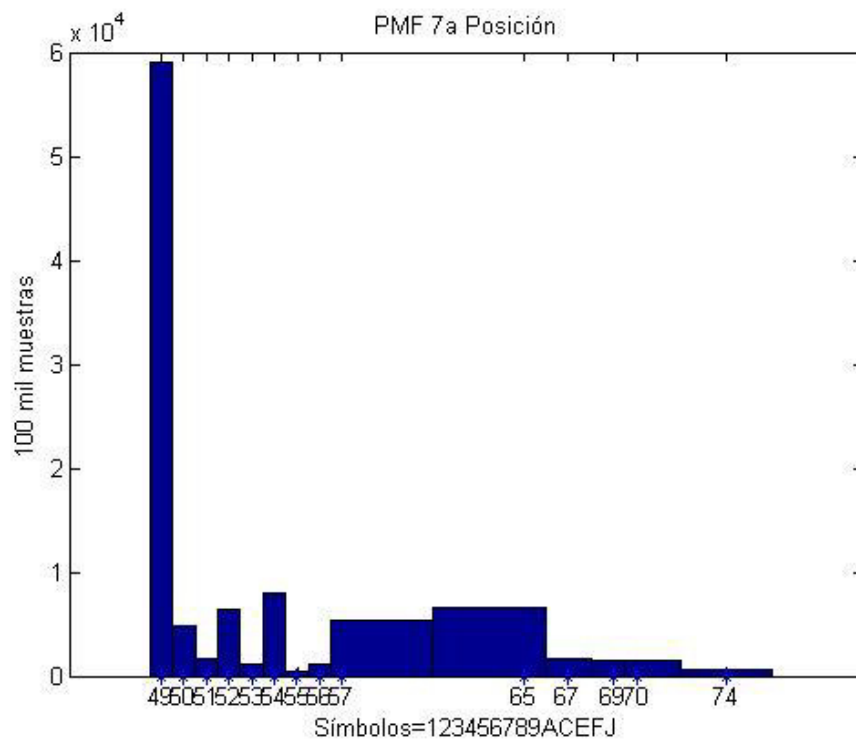


Figura 5.2.13 PMF 7ª Posición.

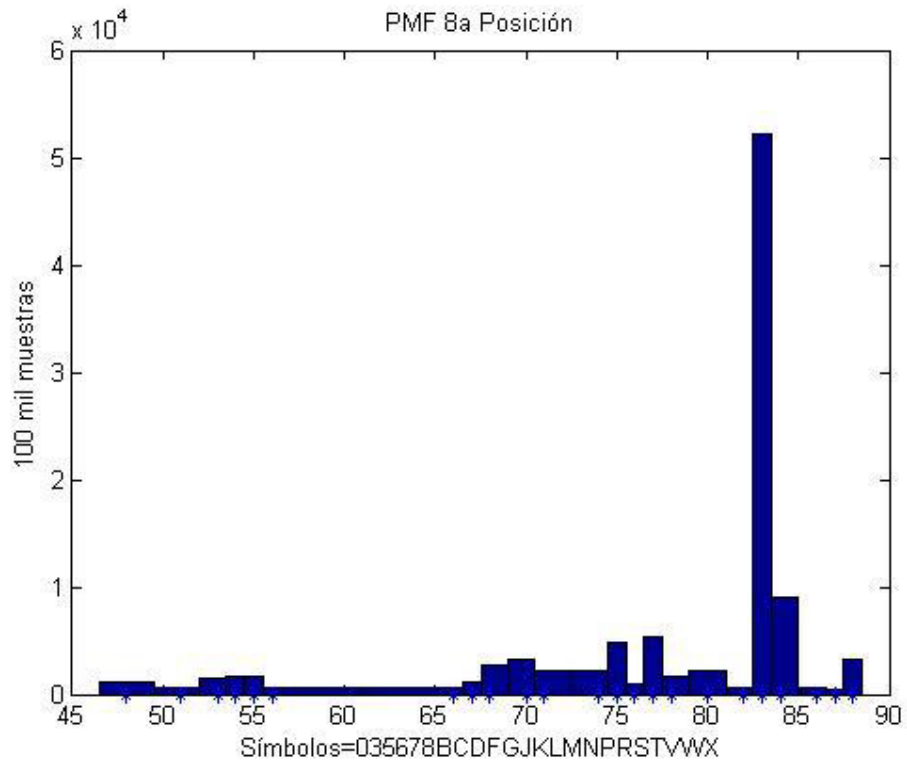


Figura 5.2.14 PMF 8ª Posición.

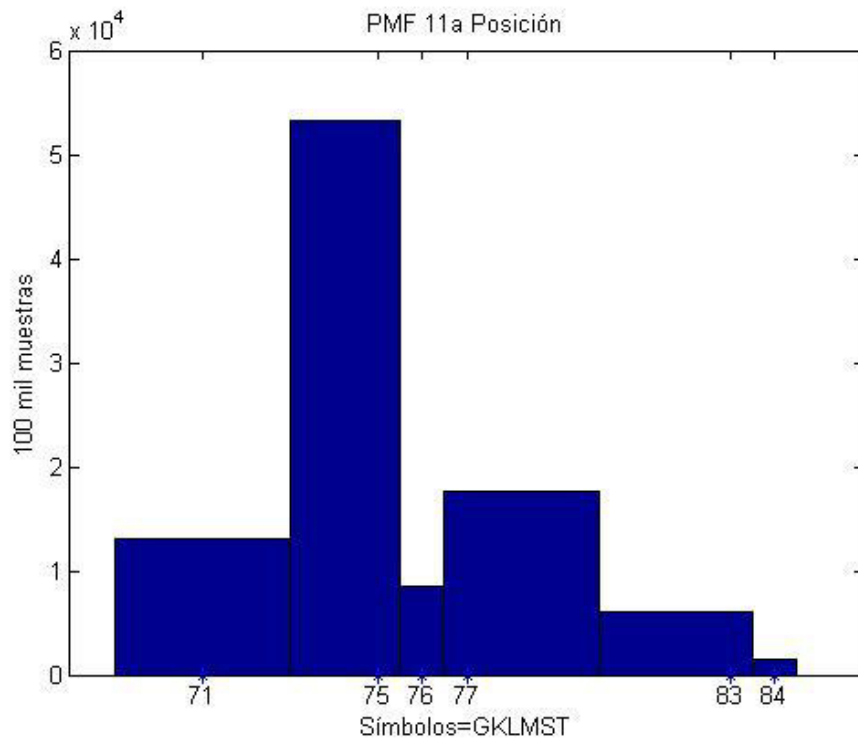


Figura 5.2.15 PMF 11ª Posición.

La posición número 1 no se ilustra debido a que según el modelo se están estudiando solamente aquellos números de serie que comienzan con el símbolo '3'. Por lo tanto esta gráfica tendría un único valor ocupando el 100% de la probabilidad. Por otro lado las posiciones 9, 10, 12 y 13 que se muestran tienen una distribución de probabilidad casi uniforme. Esto se explica brevemente a continuación:

Primero, como ya se mencionó, la posición 9 del NIV corresponde al dígito verificador, este valor se calcula en base a los otros 16 símbolos del NIV [7]. La manera de calcularlo no es de mucha importancia para nuestro trabajo, ya que la función de este símbolo es determinar si un NIV es válido. En este trabajo se hace la suposición de que éste dígito es independiente y no está relacionado con las características del automóvil. Debido a esto, para fines prácticos, en el simulador se consideró que esta posición puede ser ocupada por cualquier símbolo del alfabeto válido con la misma probabilidad de ocurrencia. Esto se explica en los resultados de la TABLA 4.1.1. En base a estos resultados se decide asignar a la tercera y cuarta sección del NIV cierta generalidad al momento de proponer el modelo. Los valores que conforman estas secciones hacen referencia a características del vehículo tales como el dígito de verificación del NIV, el año y planta de fabricación del vehículo, así como el número de serie consecutivo de producción del vehículo (Sección 4.1.1). Por lo tanto para las posiciones 9 a la 17 (a excepción de la 11) se supone una distribución de probabilidad casi uniforme por lo cual en la Figura 5.2.7 únicamente se muestran los resultados hasta la posición 13. Ya que de la posición 14 a la 17 se tendrán PMFs muy similares a las que se pueden observar en las posiciones 9, 10, 12 y 13.

En el análisis visual de los resultados mostrados en la Figura 5.2.7 se puede observar que es en las posiciones 2 a la 8 del NIV en donde se tienen distribuciones de probabilidades no uniformes, en las cuales hay elementos que tienen marcadamente más probabilidad de ocurrencia que los otros valores válidos de cada posición.



## Capítulo 6

# 6. CODIFICACIÓN Y COMPRESIÓN.

### 6.1 Métodos de Codificación y compresión.

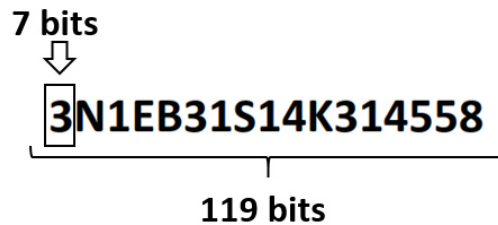
En este capítulo se abarcaran las distintas opciones para la codificación y compresión que se tomaron en cuenta. Se requiere encontrar una forma de codificación binaria que permita representar la información que emite la fuente según nuestro modelo propuesto. Para ello se estudiaron y compararon distintos métodos de la literatura en la actualidad. En base a los resultados se determinó cuál es el esquema de codificación que mejor se adapta para el escenario planteado.

#### 6.1.1 ASCII

La primera opción a tomar como referencia es el código ASCII. Esto debido a las características de la información transmitida. El NIV al tratarse de caracteres alfanuméricos (sin algún tipo de carácter especial) puede ser codificado sin problema alguno mediante el uso de este método de codificación. El código ASCII por su significado en inglés de *“American Standard Code for Information Interchange”* o Código Estándar Americano para el Intercambio de Información, es un código de caracteres creado en el año de 1963 por el Comité Estadounidense de Estándares y el cual utiliza 7 bits para la representación de los caracteres y en algunos casos un bit adicional (bit de paridad) para la detección de errores en la transmisión.

Tomando como referencia este método de codificación se necesitan 119 bits para codificar cualquier NIV que pudiese presentarse. Esta cantidad de bits se tomará como referencia a comparar con otras técnicas de codificación. Con el fin de tener

un parámetro para ver que tanto se puede comprimir la información con alguna de las técnicas que se seleccionen.



*Figura 6.1.1 Bits para codificación en ASCII.*

### **6.1.2 Método de codificación binaria de un NIV.**

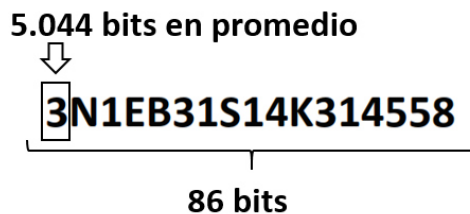
#### **Cálculo de bits necesarios para la codificación.**

Para el problema de la codificación binaria de un NIV que cumpla con las normas [5] y [6] se encuentra en la literatura el trabajo referente a [10]. En su trabajo el autor parte de que un NIV se puede almacenar de manera digital como una cadena codificada ASCII, para lo que son necesarios 17 bytes, uno para cada posición del NIV, lo que da un total de 136 bits de almacenamiento en la etiqueta para representar un NIV de manera binaria. Cabe mencionar que en [10] el autor considera la codificación ASCII utilizando 8 bits por carácter en vez de los 7 bits que se mencionan en el punto 6.1.1 de este trabajo. En el trabajo referenciado se menciona que algunas etiquetas RFID tienen alrededor de 80 bits de almacenamiento disponibles para el usuario, ya que el espacio restante está destinado para fines de comunicación y seguridad de la información que tiene la misma. Se podría pensar que una solución al limitado espacio de almacenamiento sería buscar etiquetas con más capacidad de almacenamiento, pero la compresión de la información sigue siendo muy importante para otros aspectos en la comunicación, como lo son la rapidez de lectura y la misma transmisión de información. Ya que entre menor sea la cantidad de información a enviar mayor será la velocidad con que se pueda trabajar con esta información.

Como se mencionó en la sección 4.1 los caracteres que conforman un NIV pueden ser números del 0 al 9 y letras sin incluir la I, O y Q, con lo que el alfabeto total se reduce a 33 caracteres o símbolos. En [10] se considera que cada carácter del NIV puede tomar cualquier valor del alfabeto, por lo cual existe un total de  $33^{17}$  combinaciones válidas posibles. Esto asumiendo que cada posición del NIV puede ser ocupada por alguno de los 33 símbolos del alfabeto. Para este caso, se puede considerar cada combinación como un elemento dentro de un universo, y la cantidad de bits necesarios para codificar dichos elementos está representada por la ecuación:

$$\frac{\ln 33^{17}}{\ln 2} = 85.7547 \quad (6.1)$$

Entonces asumiendo todas las combinaciones posibles de un NIV, que cumpla con el alfabeto de las normas oficiales, se necesitan 86 bits para poder codificar estas combinaciones. Con lo cual se estaría hablando de un promedio de 5.044 bits por carácter. Esto se puede ver ilustrado en la Figura 6.1.2.



*Figura 6.1.2 Codificación binaria de un NIV-ISO*

En el mismo trabajo se menciona que, al hacer un análisis más a fondo de la composición y estructura del NIV, es posible observar que la novena posición está ocupada por el carácter correspondiente al dígito verificador. Dicho valor junto con los últimos 3 caracteres del NIV, sólo pueden tomar valores numéricos (0 al 9). Con esta información es posible reducir el número de combinaciones con lo cual también se reduce el número de bits necesarios para su codificación. Aplicando la misma ecuación se tiene que:

$$\frac{\ln(33^{13} \times 10^4)}{\ln 2} = 78.8648 \quad (6.2)$$



Después de esto se identifica cada uno de las posiciones del NIV con los valores  $V_1, V_2, \dots, V_{17}$ . Dónde  $V_1$  corresponde al valor numérico del carácter que ocupa la posición número 1 en el NIV en cuestión. Lo mismo para el resto de las variables  $V$ . Por último se usa la ecuación (6.3), el resultado de esta operación da un valor decimal distinto para cada uno de los NIVs que se sustituyan en ella. Para obtener la cadena binaria de 79 bits con la que se codificará el NIV únicamente hay que transformar este resultado de base decimal a base binaria. La ecuación para obtener este valor representativo para cada NIV se da por:

$$\text{código} = ((((((((((((((V_1 \times 33 + V_2) \times 33 + V_3) \times 33 + V_4) \times 33 + V_5) \times 33 + V_6) \times 33 + V_7) \times 33 + V_8) \times 10 + V_9) \times 33 + V_{10}) \times 33 + V_{11}) \times 33 + V_{12}) \times 33 + V_{13}) \times 33 + V_{14}) \times 10 + V_{15}) \times 10 + V_{16}) \times 10 + V_{17} \quad (6.3)$$

En la ecuación (6.3) el valor de  $V_1$  se refiere al dígito (número o letra) que ocupe la posición número 1 del NIV. Una vez identificado este dígito se buscará en la TABLA 6.1.1 el valor decimal que le corresponde y se sustituirá éste valor en  $V_1$ . Lo mismo aplica para las demás posiciones ( $V_2$  a  $V_{17}$ ).

Este método es muy general y no está diseñado para un ambiente específico. Para el caso particular de este trabajo, al tratarse de un ambiente hostil RFID, existe la posibilidad que la información presente errores al momento de ser decodificada. Esto debido a que en este método propuesto para codificación y decodificación binaria, el cambio de un bit en la cadena enviada representa un error muy significativo al momento de la decodificación. Es decir, el cambio de un bit modifica totalmente el mensaje que se intentó transmitir. Entre más grande sea el peso del bit que se cambia en la cadena enviada, mayor será la cantidad de símbolos erróneos al momento de decodificarla. Al decir peso, nos referimos a que tan a la izquierda de la cadena se encuentra el bit. El bit de mayor peso es el primero de la cadena, mientras que el bit de menor peso es el último de la cadena.

### **6.1.3 Métodos de compresión para texto.**

Los métodos de compresión sin pérdidas de texto son bien conocidos en la literatura [16] y [19]. Debido a sus características, al tratarse de caracteres alfanuméricos, el NIV puede ser considerado como una cadena de texto. Por ello en este trabajo se emplean tres métodos de compresión de texto los cuales son aplicados a una serie de muestras de la fuente modelada para ver los resultados que se pueden obtener.

Para estas pruebas se empleó el simulador de datos que se propuso, el cual se explica en la sección 5.2.6. Mediante el simulador se generan cien mil muestras, lo que equivale a un millón setecientas mil letras, con las cuales se forma una cadena de caracteres la cual es introducida a los códigos de compresión descritos en esta sección. A continuación se presenta una breve descripción de los métodos empleados y su funcionamiento básico.

#### **Método de Huffman.**

Hablar de Huffman en el tema de la compresión de información, es hablar de uno de los referentes más importantes en el tema. El algoritmo que lleva su nombre fue desarrollado por el norteamericano David Albert Huffman en 1952 mientras hacía su doctorado en MIT. En el cual su profesor Robert M. Fano le asignó un trabajo bajo la premisa de encontrar el código binario más eficiente. En su trabajo de investigación Huffman solucionó la mayor parte de los errores en el algoritmo de codificación previamente propuesto por su profesor, el cual lleva por nombre el algoritmo Shannon-Fano, ya que fue desarrollado en conjunto por el profesor Robert M. Fano y el considerado inventor de la teoría de la información Claude Shannon. A raíz de esto Huffman desarrolló un trabajo el cual fue publicado en una revista con el nombre de "A Method for the Construction of Minimum-Redundancy Codes" y el resultado de utilizar dicho algoritmo es la obtención de un código prefijo óptimo.

Un código prefijo óptimo para una distribución de probabilidad dada se puede construir mediante este simple algoritmo descubierto por Huffman. Además está demostrado que cualquier otro código para el mismo alfabeto no puede tener una longitud esperada más pequeña que la que se obtiene al utilizar el algoritmo de Huffman. Esto se explica de manera más detallada en la sección 5.6 de [12].

En teoría de la información dicho método tiene uso para la compresión de datos. El resultado de este método es una tabla la cual contiene los códigos (de longitudes variables) para representar a cada uno de los símbolos que aparecen en el mensaje que se desea comprimir. En esta tabla los códigos de menor longitud corresponden a aquellos caracteres cuya probabilidad de ocurrencia es mayor. Y por otro lado utilizando los códigos de mayor longitud para aquellos caracteres cuya probabilidad de ocurrencia es menor al resto de los símbolos. Una de las características principales de éste método de codificación es que utiliza códigos prefijo. Al hablar de código prefijo se hace referencia a que la cadena de bits que representa a un símbolo, no puede ser prefijo a la cadena de bits que represente a otro símbolo en la misma tabla. Con esto se asegura que cada uno de los símbolos sea únicamente decodificable, y se puede realizar la decodificación símbolo por símbolo conforme se vaya recibiendo la cadena de bits.

Para el método de Huffman se usa una técnica que consiste en la creación de un árbol binario en el cual cada símbolo representa una hoja del final del árbol. A continuación se muestra de manera breve el procedimiento para el diseño de un código de Huffman:

- a) Suponga que  $L$  es la lista de probabilidades de los símbolos de la fuente, estos símbolos (letras) se considera están asociados con las hojas del árbol binario.
- b) Se toman las dos probabilidades más pequeñas de la lista  $L$  y se genera un nuevo nodo hermano cuya probabilidad será la suma de

estos dos probabilidades tomadas. Además se genera un nuevo nodo cuyas ramas irán a cada uno de los símbolos que se tomaron para la creación del nodo hermano. LA rama del nodo 1 irá a uno de estos símbolos y la rama del nodo 0 al otro.

- c) Ahora en la lista L se reemplazaran las dos probabilidades que se acaban de sumar, por el nuevo nodo con la suma de estas probabilidades. Si la lista L contiene ahora sólo un elemento, el proceso termina. De otra manera hay que pasar al punto “b” nuevamente.

En la Figura 6.1.4 se muestra un ejemplo de un árbol binario creado mediante el algoritmo de la codificación Huffman. Dicho ejemplo fue tomado de [19] y consta de una lista de 8 símbolos con distintas probabilidades, en la figura se muestran las probabilidades ya organizadas de manera descendente como  $P(7), \dots, P(0)$ . Se muestra de manera ilustrada como se van haciendo las relaciones y formación de nodos hasta llegar a la raíz del árbol, o a las últimas dos ramas donde la suma de las probabilidades de cada una de ellas es igual a 1.

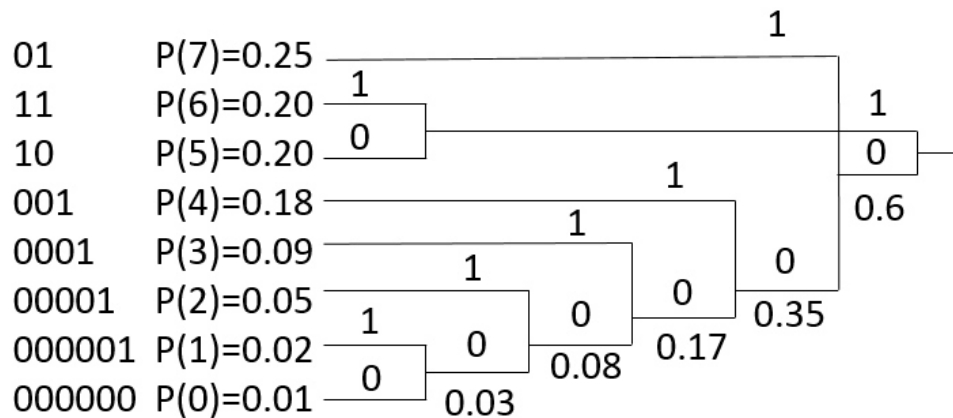


Figura 6.1.4 Un código de Huffman (Ejemplo tomado de [19])

Como se puede observar el codificador Huffman crea una estructura ordenada con todos los símbolos y la frecuencia con la que estos ocurren. Las ramas se construyen en forma recursiva comenzando con los símbolos menos frecuentes. La construcción del árbol se realiza ordenando en primer lugar los símbolos según la frecuencia de aparición. Los dos símbolos con menor frecuencia de



aparición se eliminan sucesivamente de la lista y se conectan a un nodo cuyo peso es igual a la suma de la frecuencia de los dos símbolos. El símbolo con mayor peso es asignado a la rama 1, el otro a la rama 0 y así sucesivamente, considerando cada nodo formado como un símbolo nuevo, hasta que se obtiene un nodo principal llamado raíz. El código de cada símbolo corresponde a la sucesión de códigos en el camino, comenzando desde este carácter hasta la raíz. De esta manera, cuanto más dentro del árbol esté el símbolo, más largo será el código.

### **Método de Huffman Adaptable.**

Como ya se mencionó para llevar a cabo la codificación de Huffman se necesita contar con la probabilidad de ocurrencia de cada símbolo que aparece en la fuente de información. Para en base a ello seguir con el algoritmo y generar el árbol de probabilidades que designará la codificación de cada uno de ellos.

Una variante del código de Huffman es el código de Huffman adaptativo. La principal característica de este algoritmo es que hace posible ir generando las probabilidades mientras se van presentando los símbolos. De esta forma también se va generando de manera adaptativa el árbol binario que caracteriza al código de Huffman. Dicho árbol contiene ciertas propiedades que lo caracterizan para que sea un árbol correspondiente a un código de Huffman.

Consideremos un árbol binario correspondiente a un alfabeto de tamaño  $n$ . En dicho árbol los símbolos del alfabeto serán representados como hojas. Con esto se deduce que el número de nodos del árbol es  $2n-1$ . Cada nodo del árbol cuenta con dos propiedades; una es el número de nodo y la otra el peso que ese nodo tiene. Cabe mencionar que el número de nodo será único, y cada nodo puede tomar un valor entre 1 y  $2n-1$ . En esta ocasión nombraremos a estos números como  $Y_1, \dots, Y_{2n-1}$ . Por otro lado está el peso del nodo, que puede ser de dos tipos; el peso de un nodo hoja y el peso de un nodo interno. El peso de un nodo hoja es simplemente el número de veces que aparece el símbolo de dicha

hoja. El peso del nodo interno es la suma de los pesos de sus nodos hijos. Una vez explicado esto, notaremos a los nodos del árbol binario como  $X_1, \dots, X_{2^{n-1}}$ .

Se puede probar que los pesos de los nodos quedan ordenados en un orden no decreciente al asignar números a éstos comenzando por el 1 y recorriendo el árbol por niveles. Haciendo esto de izquierda a derecha en cada nivel y de las hojas a la raíz (de abajo hacia arriba). Al hacer esto se obtiene un árbol binario correspondiente a un código de Huffman. La probabilidad de cada nodo hoja del árbol será igual a su peso dividido por la suma de los pesos. A esta propiedad se le conoce como propiedad de hermanos (sibling property).

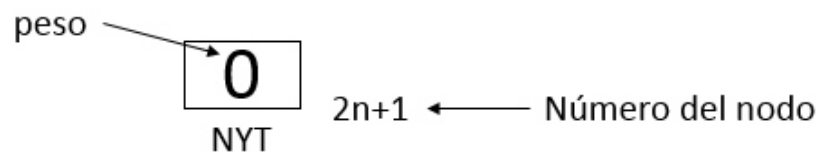
Debido a que se trata de un algoritmo en el cual las probabilidades se van calculando conforme aparecen los símbolos, ni el transmisor ni el receptor conocen en un inicio las probabilidades que tomarán dichos símbolos. Por lo tanto ambos comienzan con un árbol con un nodo único que corresponde a todos los símbolos que no han ocurrido, o no han sido transmitidos. A este nodo se le conoce como NYT por sus siglas "Not Yet Transmitted" (Aún sin transmitir), dicho nodo tiene peso cero. Más adelante conforme va progresando la aparición de los símbolos se añadirán al árbol los nodos correspondientes a ellos, se modificarán los pesos (tanto si el símbolo es nuevo o ya ha aparecido antes) y se actualizará el árbol para que siga cumpliendo la propiedad de hermanos, es decir, que siga cumpliendo con las características de un árbol de Huffman.

Hablando ahora sobre la codificación de los símbolos cabe mencionar que para los símbolos que aparecen por primera vez consta de la codificación que proporciona el árbol del nodo NYT seguido de un código fijo para el símbolo, el cual deben conocer tanto el codificador como el decodificador. Esto es para todos aquellos casos salvo que sea el primer símbolo de la secuencia. Si se trata del primer símbolo de la secuencia no es necesario transmitir el código de NYT. Por otro lado para la codificación de símbolos que ya hayan aparecido se utiliza la codificación que proporciona el árbol que se va construyendo.

El algoritmo de Huffman adaptativo se puede dividir en cuatro elementos que se describen a continuación en los puntos a, b, c y d.

**a) Inicialización del código.**

El árbol debe de codificar  $n+1$  símbolos, incluyendo el correspondiente a NYT. Por lo tanto se necesitan  $2(n+1)-1 = 2n+1$  números para la numeración de cada uno de los nodos. En un inicio el árbol de Huffman tiene un único nodo como se muestra en la Figura 6.1.5 en donde se observa que el primer elemento es el nodo NYT y tiene un peso cero.



*Figura 6.1.5 Codificación Huffman.*

**b) Algoritmo de codificación.**

En la parte de codificación, si se encuentra un símbolo nuevo se tiene que generar el código del nodo NYT seguido del código fijo del símbolo<sup>3</sup>. Después añadir el nuevo símbolo al árbol en formación. En caso de que el símbolo actual ya está presente, se debe generar su código usando el árbol. Una vez hecho esto se actualiza el árbol para que siga conservando la propiedad de hermanos.

**c) Algoritmo de decodificación.**

En la parte de decodificación para el método de Huffman Adaptativo se decodificará el símbolo usando el árbol actual. Si se encuentra el nodo NYT, se debe usar el código fijo para decodificar el símbolo que viene a continuación. De igual forma se añade el nuevo símbolo al árbol y se actualiza éste para que siga conservando la propiedad de hermanos.

**d) Actualización del árbol para mantener la propiedad de hermanos.**

---

<sup>3</sup> Nota: Recordemos que para el primer símbolo no hace falta transmitir el código de NYT. El código fijo es conocido al principio por codificador y decodificador.

La actualización del árbol para el código de Huffman Adaptativo es algo necesario que va ocurriendo cada vez que ocurre un nuevo símbolo, esto es necesario para seguir conservando las propiedades de un árbol de Huffman. Para ello supongamos que  $y$  es la hoja (símbolo) con un peso  $x$ . Si  $y$  es la raíz, se aumenta  $x$  en 1. Después se intercambia  $y$  con el nodo con número mayor que tenga de mismo peso que él, salvo que sea el nodo padre. Siguiendo a esto se aumenta  $x$  en 1 nuevamente. Si  $y$  es el padre tiene su peso  $x$ , y se vuelve al paso en que si  $y$  es la raíz, se aumenta  $x$  en 1, y se prosigue con la misma metodología de actualización.

### **Método Aritmético.**

El método de codificación aritmético es de utilidad para los casos en que las distribuciones de probabilidad son muy descompensadas. Ya que para estos casos el código de Huffman otorga resultados muy lejanos a la entropía (que como se mencionó en 5.2.5 nos da una cota mínima de bits que se puede alcanzar al momento de comprimir la información sin que haya pérdidas). En el método aritmético se puede llevar a cabo la codificación de una serie de símbolos. Para ello se distingue una secuencia de símbolos de otra mediante el etiquetado con un identificador único. El valor que pueden tomar las etiquetas para la representación de secuencias es el conjunto de números reales en el intervalo  $[0,1)$ . A este intervalo se le llamará intervalo unidad. Para este método se necesita una función que aplique secuencias de símbolos en este intervalo. Una función que aplica variables aleatorias en este intervalo es la función de distribución. Para comenzar se introducen algunos términos que se utilizarán para el desarrollo del tema de codificación aritmética.

Primero se asigna un número a cada posible símbolo del alfabeto. Es decir si  $A = a_1 \dots a_m$  asignamos  $A_i \longleftrightarrow i$ . De forma que  $P(a_i) = P(i)$  y se tiene la función de distribución

$$F_X(i) = \sum_{k=1}^i P(X = k) \quad (6.4)$$

donde  $X$  es la variable aleatoria correspondiente a la probabilidad de la fuente. Para generar la etiqueta se reduce el tamaño del intervalo en el que estará la etiqueta conforme se vayan recibiendo más símbolos. Se comienza dividiendo el intervalo unidad en intervalos de la forma  $(F_X(0), F_X(1)), (F_X(1), F_X(2)), \dots, (F_X(m-1), F_X(m))$ . Recordemos que  $F_X(0) = 0$  y  $F_X(m) = 1$ . Por lo tanto se tiene una partición del intervalo unidad en intervalos disjuntos.

Se supone que la secuencia que se va a codificar comienza por el símbolo  $a_k$ , entonces la etiqueta que generaremos para la secuencia estará en el intervalo  $(F_X(k-1), F_X(k))$ . Se divide este subintervalo en las mismas proporciones que el intervalo unidad (se lleva el intervalo  $[0, 1)$  al intervalo  $(F_X(k-1), F_X(k))$  y se sigue dividiendo este manteniendo las proporciones de  $F$  en el intervalo  $[0, 1)$ ).

Ejemplo  $P(a_1) = 0.5, P(a_2) = 0.25, P(a_3) = 0.25$

Observamos  $a_1$  y después  $a_2$

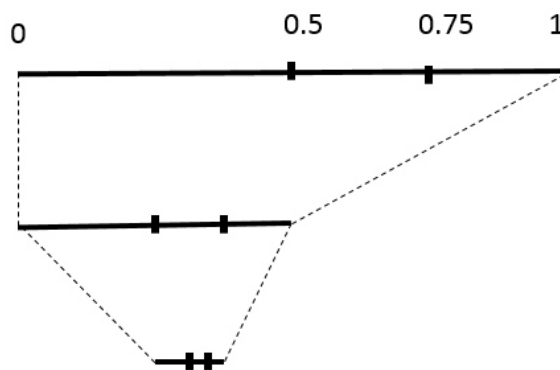


Figura 6.1.6 Ejemplo Código Aritmético.

Para llevar el intervalo  $[0, 1)$  al intervalo  $(F_X(k-1), F_X(k))$ . La transformación para  $x \in [0, 1)$  es

$$f(x) = (F_x(k) - F_x(k-1))x + F_x(k-1) \quad (6.5)$$

Si el segundo símbolo observado es  $a_j$  entonces la codificación de  $a_k a_j$  estará en el intervalo

$$(F_x(k) - F_x(k-1))F_x(j-1) + F_x(k-1), (F_x(k) - F_x(k-1))F_x(j) + F_x(k-1) \quad (6.6)$$

y así sucesivamente.

En resumen estos son los tres métodos empleados en codificación de texto que se usaron para realizar las pruebas con las muestras generadas. Una explicación más a detalle sobre el funcionamiento de cada uno de ellos puede encontrarse en [16].

### Resultados de métodos de compresión para texto.

Después de haber realizado las pruebas con los distintos métodos explicados con anterioridad, y en base a los resultados obtenidos, se observa que en resumen para la codificación y compresión de un NIV se necesitan entre 81 y 83 bits dependiendo del método empleado. La hipótesis que se tiene en cuanto a la similitud en los resultados obtenidos, es que al tratarse de una sola cadena de caracteres que no tienen una estructura definida, los métodos como el Huffman Adaptable no pueden aportar gran ventaja ya que no se generan combinaciones que vayan tomando más peso en cuanto a probabilidad. Y la cota de los tres métodos es muy parecida a la del código de Huffman, el cual básicamente asigna un código binario fijo a cada uno de los símbolos que aparecen en la cadena de texto, basándose en la probabilidad de ocurrencia de los símbolos y otorgándole menos bits de codificación a aquellos cuya probabilidad sea mayor.

**4.88 bits en promedio**



**3N1EB31S14K314558**

**82 ± 1 bits**

#### **6.1.4 Método de codificación binaria con la NOM-001-SSP-2008.**

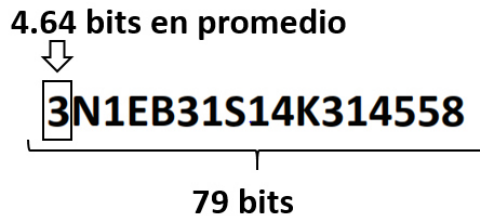
Una vez explicado el método de codificación binaria de un NIV encontrado en la literatura se presentan diversas variantes en cuanto a los alfabetos usados. Esto con el fin de determinar en qué grado puede esto ayudar a una mayor comprensión de la información. El primer análisis y experimentación que se hizo fue en base a la Norma Oficial Mexicana NOM-001-SSP-2008. En la cual se mencionan algunas características que pueden ser de utilidad para reducir el panorama general que presenta el autor en [10] a uno más específico, como lo es el problema de investigación de este trabajo.

Para este punto del trabajo se tiene que según lo descrito en la Norma Oficial Mexicana [7] la novena posición del NIV corresponde al dígito verificador. Dicho valor en vehículos fabricados en territorio Mexicano, puede tomar valores del 0 al 9 y la letra X (11 valores). Además las últimas 4 posiciones únicamente toman valores numéricos, desde el 0 hasta el 9 (10 valores).

Con esta información en conjunto con el método para codificación binaria descrito en 6.1.2, el número de bits requeridos para la codificación aquellos NIVs que cumplan con lo descrito en la NOM-001-SSP-2008 se puede obtener mediante la ecuación (6.3). Esto se logra dando a el conjunto de  $V_1, \dots, V_{17}$  los valores correspondientes al último NIV que se pueda formar con los alfabetos que se tienen. Es decir, sustituyendo el máximo valor numérico posible en cada posición desde  $V_1$  hasta  $V_{17}$  y aplicando la ecuación para obtener el número de bits necesarios para la codificación del resultado mayor. Realizando lo anterior se obtiene que para la codificación de un NIV con las características mencionadas en este punto se necesitarían 79 bits. Esto se muestra en la ecuación (6.7) en donde el numerador corresponde al resultado en decimal de sustituir los valores máximos posibles en  $V_1, \dots, V_{17}$ . Que en valores alfanuméricos es el símbolo "Z" y en valores numéricos el símbolo "9", ya que son los que tienen el un valor numérico mayor.

$$\frac{\ln(550.40 \times 10^{21})}{\ln(2)} = 78.8648 \text{ bits} \quad (6.7)$$

En resumen, usando el método de codificación binaria en combinación con las características de un NIV estipuladas en la NOM-001-SSP-2008, se necesitan 79 bits para la codificación binaria de los NIVs que se pueden obtener.



*Figura 6.1.7 Codificación binaria de un NIV-NOM*

En esta ocasión se ve que los resultados son los mismos a los obtenidos por el autor en [10] por lo cual no aporta un gran avance en cuanto el tema de compresión referente a lo que ya se tiene en la literatura. Aunque esto en gran medida se debe a la gran generalidad que sigue existiendo aún después de tomar estas especificaciones de la norma oficial mexicana.

### **6.1.5 Método de codificación binaria con Modelo 2.**

Para los resultados de este punto se parte de las mismas bases de la codificación binaria presentadas en 6.1.2, en donde el resultado de una operación matemática determina la cantidad de bits necesarios para una codificación.

En este punto se toman en cuenta las características que implica usar de referencia el modelo de la fuente propuesto en 5.2 (Modelo 2). Para conocer el número de bits necesarios para la codificación de un NIV, de la misma forma que en el punto anterior, se requiere encontrar el resultado de la ecuación (6.3) para aquellos valores de  $V_1$  hasta  $V_{17}$  que maximicen el resultado de la operación. Para hacer esto se requiere saber los alfabetos que componen cada posición de NIV según este modelo propuesto. Estos alfabetos se observan en la



TABLA 6.1.2. En ella se encuentran ordenados de manera ascendente cada una de las posiciones del NIV con sus posibles alfabetos. La columna indica de qué posición se trata, y los elementos de cada columna indican los posibles símbolos que se pueden encontrar en dicha posición. Se observa que el número de elementos en los alfabetos varía desde un elemento, en la primera posición, hasta veintitrés elementos, en la posición número ocho.

Recordando, de acuerdo con el método propuesto en [10] existe una tabla con los valores numéricos para cada símbolo del alfabeto del NIV. Con la ayuda de la

TABLA 6.1.2 se obtienen los valores máximos que pueden tomar  $V_1, \dots, V_{17}$ . Al sustituir estos valores en la ecuación (6.3) se obtiene que la cantidad de bits necesaria para la codificación binaria en este escenario es de 76 bits. Esto se puede ver en la ecuación (6.8) dónde el elemento dentro del logaritmo en el numerador representa el resultado de la operación (6.3) al sustituir los valores máximos de  $V_1$  a  $V_{17}$ , los cuales se toman en referencia a los alfabetos mostrados en la

TABLA 6.1.2.

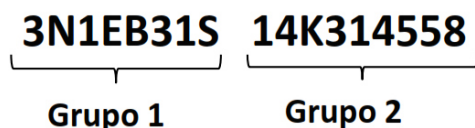
$$\frac{\ln(64.6933 \times 10^{21})}{\ln(2)} = 75.7760 \text{ bits} \quad (6.8)$$



### 6.1.6 Método de codificación binaria con Huffman.

Haciendo un análisis al árbol de probabilidades propuesto en 5.2.4 y a la distribución de probabilidades obtenidas en 5.2.7, se puede observar que entre las posiciones 1 y 8 del NIV es dónde existen dependencias y relaciones más marcadas entre los símbolos de una y otra posición. En estas posiciones la probabilidad de ocurrencia de los símbolos no está uniformemente distribuida, a diferencia de aquellas donde se propuso suponer esta uniformidad para fines prácticos y de generalidad del modelo. Gracias a esto se propone hacer una codificación de la información de la fuente (NIVs) por partes. Para esto se decide hacer una división del NIV en dos partes, la primera parte abarca la codificación de las primeras 8 letras del NIV, mientras que la segunda parte trata la codificación del resto del NIV, es decir, los últimos 9 caracteres.

Para la codificación del primer grupo se utilizan códigos obtenidos mediante el método de Huffman. El segundo grupo se codificará mediante el método de codificación binaria propuesto por [10], haciendo una adaptación para la codificación de los últimos 9 caracteres únicamente. A continuación se presenta la metodología que se sigue para la codificación de los grupos mencionados, que juntos dan como resultado la codificación binaria con compresión de un NIV.



*Figura 6.1.9 Codificación por grupos.*

#### **Primer Grupo (Posiciones 1 a 8 del NIV)**

Como se mencionó anteriormente, para realizar la codificación de este grupo se emplea el método de Huffman (Sección 6.1.3). Se tiene un modelo propuesto el cual se obtuvo en base al análisis de una serie de muestras representantes de la población de interés (Modelo 2). Mediante el uso de cadenas de Markov se busca representar las dependencias y probabilidades de los símbolos del NIV. Esto en base a la posición que ocupen y la combinación de símbolos que exista en las

posiciones anteriores. Recordemos que una cadena de Markov (también llamadas procesos de Markov) es un proceso estocástico en el cual los resultados dependen de resultados de la(s) etapa(s) anterior(es). Dichas cadenas tienen memoria ya que recuerdan los eventos anteriores y con ello condicionan las probabilidades de ocurrencia de los eventos futuros [12].

Gracias al modelo se pueden obtener las combinaciones válidas que es posible crear con las primeras 8 posiciones del NIV. Ya que es la primera parte que se quiere codificar de manera binaria. Como ya se mencionó, la elección de esta forma de codificación y la longitud del tramo del NIV a codificar, fue en base a que en esta parte es donde se observa que el uso de métodos de compresión sin pérdidas puede brindar mejores resultados. Esto gracias a las probabilidades no uniformes que presentan las combinaciones formadas hasta esta posición.

El primer paso para llevar a cabo esta codificación es obtener las posibles combinaciones válidas según el modelo propuesto. En teoría, con los alfabetos obtenidos y haciendo caso omiso de la condicionalidad de ocurrencia que tienen los símbolos con sus antecesores, se pueden crear  $291.2168 \times 10^6$  combinaciones. Cabe mencionar que aunque la cantidad de combinaciones es muy grande, no implica que todas ellas sean válidas. Es decir, en base al modelo propuesto (árbol de probabilidades) la mayoría de estas combinaciones no ocurren nunca. Mediante el simulador de datos, y de un filtrado para la eliminación de datos repetidos, se obtienen las combinaciones válidas que se pueden formar en las primeras 8 posiciones del NIV para el modelo de fuente propuesto.

Para la obtención de las combinaciones válidas se hizo uso del simulador de datos creado, el cual tiene un comportamiento basado en el modelo propuesto. Se generaron alrededor de 10,000 combinaciones, únicamente tomando las primeras 8 posiciones del NIV. Esto con la finalidad de que el programa genere todas las combinaciones válidas que hay, ya que al utilizar las decisiones que se le programaron, y usar probabilidades para cada una de ellas, al generar esta

cantidad de datos se asegura que se genere más de una ocasión cada una de las combinaciones posibles, desde la más probable hasta la que tiene menos probabilidad. Una vez hecho esto, se eliminaron aquellos datos repetidos. Con esto se obtiene una lista de combinaciones únicas, que de acuerdo al modelo son las combinaciones válidas que hay. Esto en las primeras 8 posiciones del NIV.

Una vez obtenidas estas combinaciones válidas, se procede a corroborar que la información se correcta. Para ello se realiza un programa el cual calcula la probabilidad de ocurrencia de un NIV en base a las probabilidades del modelo propuesto, dicho programa trabaja en base a las tablas de probabilidades presentadas en la sección 5.2.2. Después, usando este programa creado, se obtienen las probabilidades de cada una de las combinaciones válidas obtenidas mediante la metodología explicada, y al final se suman. El resultado de la suma de las probabilidades debe dar 1, indicando que no queda excluida ninguna combinación que tenga una probabilidad de ocurrencia representativa. Y corroborando con esto que la metodología usada para la obtención de las combinaciones válidas da los resultados esperados.

Una vez obtenidas las combinaciones válidas con sus probabilidades correspondientes se emplea el método de compresión y codificación Huffman. Para esto se hace una especie de extrapolación del problema, es decir, cada una de las combinaciones es tomada como un elemento con una probabilidad. Por lo cual, se tiene una serie de elementos con sus probabilidades de ocurrencia. Habiendo hecho esta simplificación del problema, se aplica el método tradicional de Huffman, para en base a las probabilidades de ocurrencia de los elementos, obtener un código prefijo para cada uno de los elementos. Así se obtiene la codificación de manera binaria de cada uno de los elementos del problema de acuerdo a su probabilidad de ocurrencia, requiriendo menos bits de codificación aquel elemento cuya probabilidad es mayor. Con lo cual, recordando que un elemento es una combinación, el resultado es tener la codificación binaria para cada una de las combinaciones en base a su probabilidad de ocurrencia. Todo

esto tomando como referencia las probabilidades y suposiciones que el modelo de fuente propuesto conlleva.

En la TABLA 6.1.3 se presentan los resultados obtenidos del el uso del algoritmo de Huffman para codificación y compresión. En la primera columna que lleva como nombre “NIV” se tienen las combinaciones válidas que se obtuvieron para el modelo propuesto. Cabe mencionar que no se muestra de manera explícita el primer dígito ya que de manera predeterminada se sabe que su valor siempre será “3”. En la columna con nombre “Probabilidad” se tienen las probabilidades relacionadas a la combinación de símbolos correspondiente a cada fila. Y por último en la tercera columna llamada “Código” se tiene la representación binaria que se le da a las combinaciones representadas en la primera columna.

*TABLA 6.1.3 Codificación 1-8.*

NIV	Probabilidad	Código
N1EB31S	0.4579	1
GNEC16T	0.0381	110
GCEC14T	0.0246	1100
VWYV49M	0.0192	1011
G1TA5AF	0.0184	1111
GBKC34G	0.0158	10001
G1SF61X	0.015	10011
N1CN7AD	0.0139	10100
FAFP4AJ	0.0132	10110
VWHG11K	0.0123	11010
G1SF21X	0.0114	1
N6DD12S	0.011	101
VWRV49M	0.0107	1000
N1EB315	0.0106	1001
C4PDCFG	0.0106	1010
TMJU62N	0.0105	1011
N1BC11S	0.0105	1100
N6DD25T	0.0104	10001
N6DD14S	0.0103	10010
N1CB51S	0.0096	11100
D4PG6FV	0.0089	100000
D4PG5FB	0.0079	100100
VWPN11K	0.0069	101010
GCEK14J	0.0065	101111

NIV	Probabilidad	Código
GYEK62N	0.0065	101110
N1AB61D	0.0061	110110
VWGN11K	0.0061	110111
GNDA23P	0.0061	111000
FDKF36L	0.006	111010
HGCM563	0.006	111001
GCEK14T	0.0059	111100
VW1V09M	0.0059	111011
FDLF46P	0.0059	111101
HAMMAAR	0.0058	0
FTEF17W	0.0058	111111
N6DD13S	0.0058	111110
GNFC16J	0.0057	100
G1TC5CF	0.0057	1
VWRG11K	0.0056	101
VWRV09M	0.0056	110
C6YDBAP	0.0055	1100
CZRE487	0.0055	1001
G1TX526	0.0055	111
VWFS71K	0.0055	1000
VWJW11K	0.0054	1101
N1CB315	0.0054	1111
BNBB1J6	0.0054	1110
GNCL537	0.0053	11010

NIV	Probabilidad	Código
G2JB12F	0.0052	11110
G2JB52F	0.0052	11101
VWSV29M	0.0052	100001
GNEK12T	0.0052	11111
GTN29E0	0.0052	100000
VWLW6AJ	0.0052	11100
VWBV09M	0.0051	100110
GBM7H1C	0.0049	101000
GCNK9E0	0.0049	100111
N1CK3CS	0.0048	111010
N1BC1AS	0.0048	101001
N1EB318	0.0048	101010
G1TA5A6	0.0048	101011
N1ED31S	0.0045	111011
AKJGLCK	0.0044	1000011
G1SF51X	0.0044	1000010
D6WN56C	0.0038	1001010
VWJG11K	0.0036	1001011
D6WN5ET	0.0034	1010110
D6WN56D	0.0032	1010111
FEKF36L	0.0028	110110
FELF46P	0.0024	110111



De los resultados obtenidos se puede ver que para la codificación de la combinación con menor probabilidad de ocurrencia se utilizan 9 bits. Lo que es poco más de un bit por símbolo (no se incluye al principio el número 3 ya que no aporta información) lo cual es un grado de compresión considerable si se toma como referencia el promedio de bits/símbolo que se necesitan emplear con cualquiera de los métodos propuestos en 6.1

**3N1EB31S14K314558**  
└──────────┘  
**1-9 bits**

*Figura 6.1.10 Codificación Grupo 1.*

### **Segundo Grupo (Posiciones 9 a 17 del NIV)**

Debido al espacio limitado con el que cuentan los bancos de memoria de las etiquetas RFID pasivas, [10] propone un método matemático para la representación de un NIV con un número menor de bits. De acuerdo a [10] un NIV contiene 17 caracteres y para la representación binaria de las posibles combinaciones se necesitarían 86 bits.

Lo anterior es tomando en cuenta que se trata de la codificación completa del NIV, los 17 caracteres, pero como en esta parte del trabajo lo que nos interesa es la codificación de las últimas 9 posiciones, se espera también que el número de bits necesarios para la codificación disminuya. Tomando en cuenta que sólo se requiere codificar las combinaciones formadas por los últimos 9 símbolos, y se sabe que cada una de las posiciones del NIV tiene un alfabeto particular según el modelo de fuente propuesto. Usando la misma tabla de valores numéricos expuesta en la sección 6.1.2, y haciendo una pequeña modificación a la ecuación presentada en esa misma sección, se logra codificar de manera binaria únicamente los últimos 9 valores del NIV, que es lo que se desea. Tomando en cuenta esto, y realizando el mismo procedimiento explicado en la sección mencionada, se tiene que para la representación de este segundo grupo de

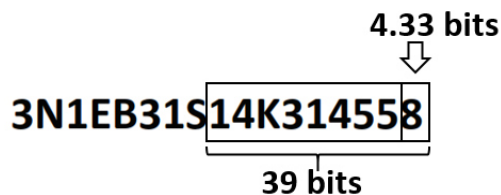
caracteres se necesitan 39 bits. Para esto se buscaron en la TABLA 6.1.4 aquellos valores de V9 a V17 que maximizaran el resultado de la ecuación (6.3) adaptada para este caso.

La TABLA 6.1.4 muestra los alfabetos de las posiciones 9 a la 17 del NIV. A partir de los cuales se eligen los elementos para determinar la cantidad de bits que se requieren para la codificación binaria de esta sección del NIV.

*TABLA 6.1.4 Alfabetos posiciones 9-17.*

Número de Identificación Vehicular (NIV)								
9no	10mo	11mo	12mo	13ro	14to	15to	16to	17mo
0	3	G	0	0	0	0	0	0
1	4	K	1	1	1	1	1	1
2	5	L	2	2	2	2	2	2
3	6	M	3	3	3	3	3	3
4	7	S	4	4	4	4	4	4
5	8	T	5	5	5	5	5	5
6	9		6	6	6	6	6	6
7	A		7	7	7	7	7	7
8	B		8	8	8	8	8	8
9	C		9	9	9	9	9	9
X	D		A					

En la Figura 6.1.11 se muestra de manera gráfica la cantidad de bits necesarios para la codificación del segundo grupo de símbolos de un NIV. Esto de acuerdo a la propuesta explicada en esta sección.



*Figura 6.1.11 Codificación Grupo 2.*

**Resultado conjunto de ambos grupos.**

Como resultado de este método propuesto se obtiene, que para la codificación de manera binaria de un NIV se pueden emplear desde 40 hasta 48 bits. Esto

gracias a la gran medida de compresión que se logra hacer en las primeras 8 posiciones tomando como base las características del modelo de fuente que se está proponiendo.

Con la reducción en la cantidad de bits para codificar la fuente, el espacio que se ahorre en la etiqueta RFID puede ser utilizado para almacenar alguna otra información de importancia, que sea adicional al número de serie del vehículo.

En la Figura 6.1.12 se muestra de manera gráfica la cantidad de bits necesarios para la representación de un NIV utilizando el método propuesto en esta sección, así como el promedio en bits que se estaría usando para cada uno de los símbolos representados.



*Figura 6.1.12 Codificación con Grupos.*

### **6.1.7 Codificación Binaria Adaptada con Huffman.**

En este punto se presenta la metodología y resultados de un análisis parecido al 6.1.6. Solo que en esta ocasión se hace una modificación a la tabla de valores numéricos y ecuación de obtención del código binario, presentados por [10] en su trabajo. La variación que se hace únicamente afecta a lo propuesto en el número anterior para la codificación binaria de las combinaciones posibles con los últimos 9 dígitos del NIV, es decir el segundo grupo. Por lo tanto, la codificación del primer grupo de caracteres (1 al 8) será la misma que se explicó en el punto 6.1.6.

A manera de recordar, en su trabajo Hashim propone una tabla con valores numéricos para cada símbolo del alfabeto para un NIV. El alfabeto que se

muestra contienen los 33 caracteres que marca la norma. En base a estos valores posibles, se obtiene un resultado decimal después de una operación matemática. Éste resultado es diferente para cada NIV dependiendo de los símbolos que lo compongan. Por último la codificación binaria del NIV se obtiene al pasar ese resultado de base decimal a base binaria.

Tomando en cuenta lo anterior, se propone reducir el alfabeto de acuerdo a los posibles valores que se pueden presentar, según el modelo propuesto, en las combinaciones que se hagan desde la posición 9 hasta la 17. De igual forma se reasigna el valor numérico a cada uno de los elementos del nuevo alfabeto y se reestructura la fórmula para la obtención del código. Estos nuevos valores propuestos se muestran en la TABLA 6.1.5.

*TABLA 6.1.5 Valores numéricos para Codificación Binaria\_2.*

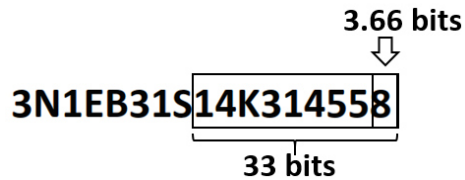
Caracter del NIV	0	1	2	3	4	5	6	7	8	9	A	B
Valor Numérico	0	1	2	3	4	5	6	7	8	9	10	11

Caracter del NIV	C	D	G	K	L	M	S	T	X
Valor Numérico	12	13	14	15	16	17	18	19	20

Y la ecuación para la obtención del código quedaría como se expresa en la ecuación (6.9). El resultado de la operación se pasa a base binaria y de esa forma sería como se representaría el segundo grupo de caracteres (9 al 17). Para la representación de este segundo grupo de caracteres, en base a esta nueva forma de codificación propuesta, se necesitan 33 bits para la codificación binaria de las combinaciones posibles. Lo que nos lleva a una compresión de 6 bits menos en comparación de usar el método descrito en el punto 6.1.6.

$$código = (((((((V1 \times 14 + V2) \times 20 + V3) \times 11 + V4) \times 10 + V5) \times 10 + V6) \times 10 + V7) \times 10 + V8) \times 10 + V9 \quad (6.9)$$

En la Figura 6.1.13 se muestra de manera gráfica la cantidad de bits necesarios para la codificación del segundo grupo de símbolos de un NIV. Esto de acuerdo a la propuesta explicada en esta sección.

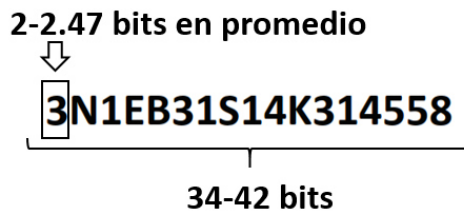


*Figura 6.1.13 Codificación Grupo 2 Propuesta.*

### **Resultado conjunto de ambos grupos.**

Como resultado del método propuesto en esta sección se obtiene que para la codificación de manera binaria de un NIV se pueden emplear desde 34 hasta 42 bits. Lo que implica 6 bits menos a diferencia de lo propuesto en el punto anterior. De igual manera esto es gracias a la gran medida de compresión que se logra hacer en las primeras 8 posiciones tomando como base las características del modelo de fuente que se está proponiendo.

En la Figura 6.1.14 se muestra de manera gráfica la cantidad de bits necesarios para la representación de un NIV utilizando el método propuesto en esta sección, así como el promedio en bits que se estaría usando para cada uno de los símbolos representados.



*Figura 6.1.14 Codificación con grupos (2)*

## **6.2 Codificación y decodificación de la información.**

Después de analizar las diversas formas de codificar la información se opta por elegir el método propuesto en 6.1.7 debido a que es el método propuesto que presenta mejores resultados de compresión.

### 6.2.1 Codificador.

En la parte de codificación de un NIV, se realizó un programa que realiza lo propuesto en la sección 6.1.7. El funcionamiento del programa computacional se puede ver en la Figura 6.2.1, primero el NIV es dividido en dos grupos, el grupo 1 contiene los símbolos desde la posición 1 hasta la 8, mientras que el grupo 2 contiene los símbolos desde la posición número 9 hasta la 17. Una vez identificados ambos grupos se procede a hacer la codificación de cada uno de ellos, el procedimiento de codificación del primer grupo se puede ver en la sección 6.1.6 en la parte de Primer Grupo (Posiciones 1 a 8 del NIV). Por otro lado, el procedimiento para la codificación del segundo grupo se tiene en la sección 6.1.7 con las adaptaciones al modelo propuesto. Después de este paso se tiene de manera separada la codificación binaria de cada uno de los grupos, por lo cual el siguiente paso en el proceso es la de unir estas cadenas de bits para obtener como resultado la cadena final que representará de manera binaria el NIV que haya ingresado al proceso de codificación. La Figura 6.2.1 muestra el diagrama de bloques de los pasos que sigue el codificador desde que se detecta el NIV a codificar.

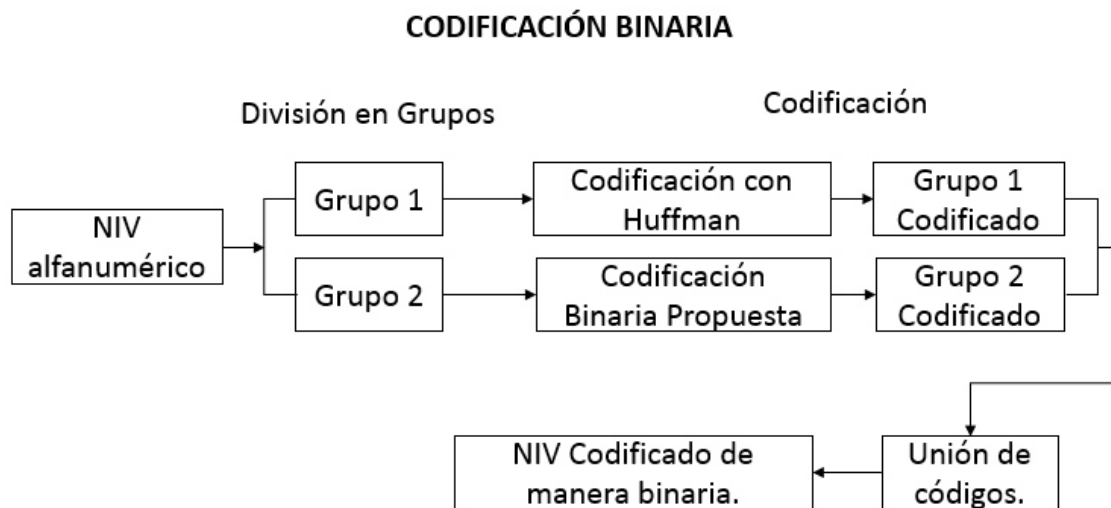


Figura 6.2.1 Diagrama de bloques del Codificador.

### **6.2.2 Decodificador.**

De igual forma que para la codificación, para la decodificación se realizó un programa computacional que realiza las operaciones y procedimientos necesarios para obtener el valor simbólico del NIV a partir de una cadena de bits. La Figura 6.2.2 muestra un diagrama de bloques con el procedimiento que lleva a cabo el decodificador al momento de recibir la cadena de bits correspondientes a un NIV previamente codificado. Lo primero, al igual que en la parte de la codificación, es identificar las cadenas correspondientes al grupo 1 (posiciones 1 al 8 del NIV) y grupo 2 (posiciones 9 a 17 del NIV). Para esto se hace uso del conocimiento previo que se tiene sobre la cantidad de bits que se necesitan para codificar al segundo grupo, ya que se emplean códigos de la misma longitud, en este caso 33 bits. Al conocer esto se puede hacer la división de los bits que corresponden al grupo 1 y los que corresponden al grupo 2. Después aparece la parte de detección y corrección de errores, esta parte va más enfocada a la detección y corrección en los bits que se tienen del primer grupo, ya que es donde es más factible y fácil encontrar errores y llevar a cabo una corrección en base al árbol de decisiones que se tiene. Esto debido a que el cambio de un bit en la cadena lleva a cadenas de bits que están fuera del libro de códigos propuesto, con lo que es posible detectar estos errores. A diferencia de esto, la cadena de bits que representa al Grupo 2, al sufrir alguna modificación en uno o más de sus bits, tienden a resultar en un resultado que al ser decodificado resulta válido, más no necesariamente correcto. La parte de detección y corrección de errores se explicará la siguiente sección de este trabajo. Continuando con el proceso de decodificación, y asumiendo que las cadenas de bits que se tienen para ambos grupos son correctas, se procede a hacer la decodificación de cada uno de los grupos. Primero, para el grupo uno, se utiliza la tabla con combinaciones presentada en la sección 6.1.6 en la parte de “Primer Grupo (Posiciones 1 a 8 del NIV)” donde aparece la TABLA 6.1.3 que muestra la codificación binaria que se le asigna a cada una de las combinaciones válidas que se pueden realizar de acuerdo al modelo de fuente propuesto. Se busca la combinación de bits recibida y en base a ella se decide que cadena de símbolos es la que se está

decodificando. De igual forma se lleva a cabo la decodificación de la cadena de bits correspondientes al segundo grupo. El cual representa la combinación de símbolos que hay desde la posición 9 hasta 17 del NIV. Para la decodificación de esta parte se realiza la operación inversa a la ecuación (6.9) para obtener los valores de  $V_9, V_{10}, \dots, V_{17}$  y en base a ellos determinar cada uno de los símbolos que se están decodificando, utilizando para ello la TABLA 6.1.5 de la sección 6.1.7.

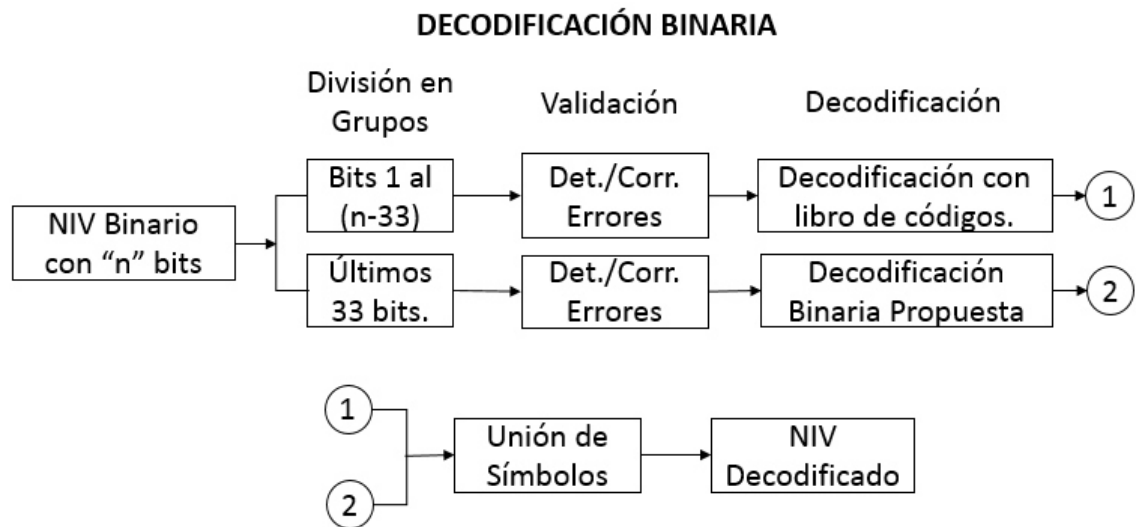


Figura 6.2.2 Diagrama de bloques del Decodificador.

## 6.3 Pruebas en Canal BSC

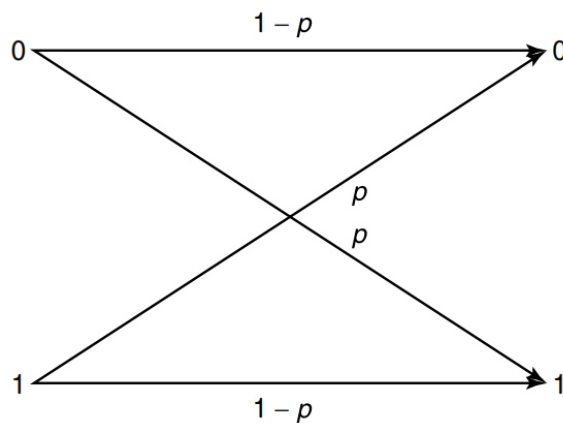
### 6.3.1 Canal BSC

Un canal simétrico binario (BSC) es un modelo de canal de comunicaciones usualmente usado en temas de teoría de la información y teoría de la codificación. En dicho modelo, el transmisor desea enviar un bit el cual será recibido por el receptor. Se asume que usualmente el bit será transmitido de manera correcta, pero existirá la probabilidad de el bit cambie de valor a la salida del canal. Se usa este canal con frecuencia en teoría de la información ya que es uno de los canales más simples de analizar. Se le llama canal binario debido a que únicamente puede transmitir dos símbolos (en este caso 0 y 1). En dicho canal la transmisión



no será perfecta, por lo cual, dependiendo de la probabilidad de error que se establezca, el receptor recibirá de manera errónea uno o más bits.

En la Figura 6.3.1 se muestra un diagrama del comportamiento de un canal simétrico binario. La probabilidad de error se representa como  $1-p$  y corresponde a la probabilidad de que el bit enviado sea cambiado por el canal. Por otro lado la probabilidad de que llegué correctamente está representada por  $p$  y quiere decir con que probabilidad se estarán transmitiendo los bits de manera correcta por el canal.



*Figura 6.3.1 Canal Simétrico Binario*

### 6.3.2 Metodología

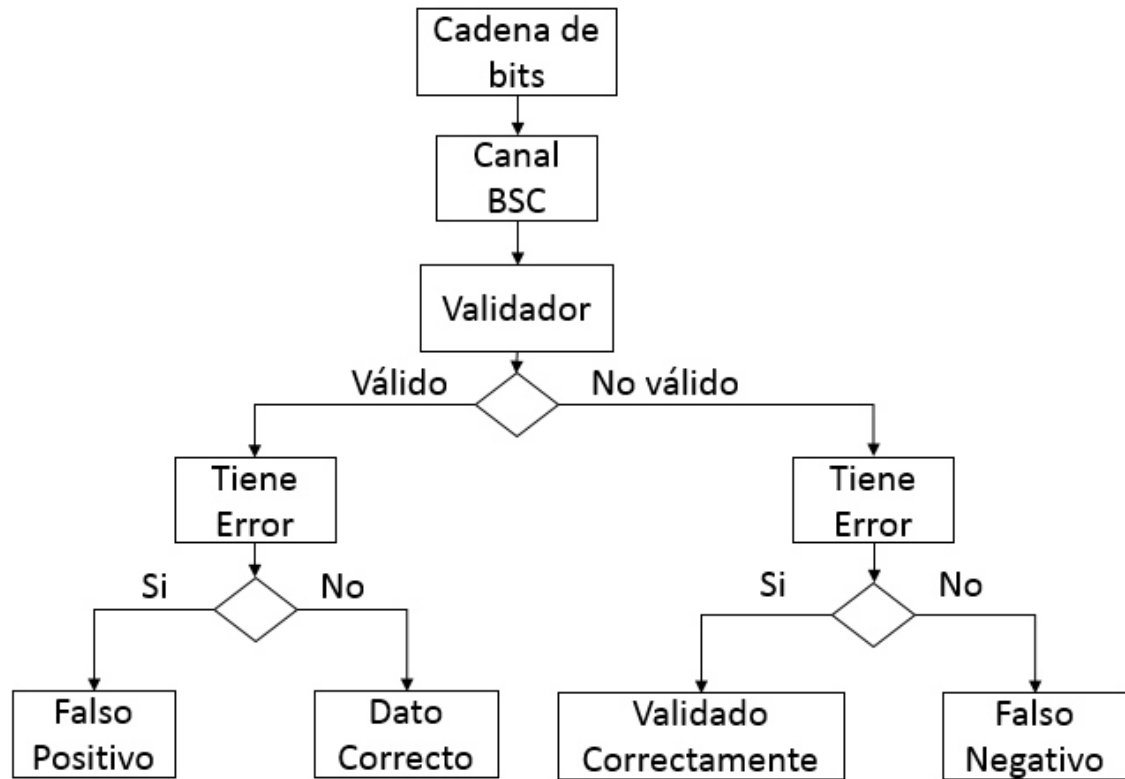
Para este punto ya se tienen programados de manera computacional los siguientes elementos:

- ✓ Simulador
- ✓ Codificador
- ✓ Decodificador

Con esto es posible ya realizar algunas pruebas en cuanto a la robustez que tienen los códigos. Para esto se emplea un canal simétrico binario (BSC) con el fin de observar el funcionamiento del decodificador y la sensibilidad de los códigos generados al momento de generar en ellos error con cierta probabilidad.

Para estas pruebas usaremos los elementos antes mencionados, generando datos por el simulador obtendremos NIVs que cumplan con las características que se plantearon estudiar en un inicio. Después esta información será codificada de manera binaria, usando el codificador explicado en la sección 6.2.1 lo que nos dará como resultado una cadena de bits que representará el NIV generado previamente por el simulador. Esta cadena de bits será pasada por el BSC el cual generará errores con una cierta probabilidad que se le dará, con esto algunos bits de la cadena que se tenía se cambiarán, por lo tanto es posible que ocurra errores en algunos de los NIVs al momento de decodificarlos. Por lo tanto, al momento de salir del canal BSC se realizará una verificación de validez de la cadena obtenida. Con esto se sabe en primera instancia si la cadena de bits puede ser decodificada de manera correcta, o si contiene algún error dentro de sus bits.

Cabe mencionar que debido a la forma en que se codifican los NIVs, se decide únicamente trabajar en la detección y corrección de errores para la parte que corresponde al primer grupo de caracteres (posiciones 1 a la 8) ya que es la parte donde gracias a las características del modelo en donde se tiene un árbol de decisiones y probabilidades, es más fácil hacer la detección y posible corrección de aquellas cadenas de bits que contengan un error. A diferencia del grupo 1, en el grupo 2 (posiciones 9 a 17) al cambiar un bit de la cadena es muy difícil detectar el error, ya que en la mayoría de los casos este cambio lo que ocasionará es que el decodificador dé como resultado una cadena de símbolos que sea válida pero que no sea la que se codificó. A esto le llamaremos un falso positivo, ya que ante el detector de errores la cadena de bits no tendrá error alguno, pero al momento de comparar la información que se ingresó con la que se decodificó después del canal se podrá ver que el error existe. De manera opuesta también existen los falsos negativos, esto sucede cuando el validador determina que hay un error en la cadena de bits que está analizando, cuando en realidad la información sea correcta. Estos términos se ilustran de mediante un diagrama de bloques en la Figura 6.3.2.



*Figura 6.3.2 Falsos Negativo y Positivo.*

El proceso que se llevará a cabo para las pruebas se muestra en la Figura 6.3.3 en donde primero se tiene el NIV que se va a transmitir, este NIV es generado previamente por el simulador de la fuente que se propone. El NIV será dividido en dos grupos para su posterior codificación de manera binaria, la cual se explica en la sección 6.2.1. Una vez codificado el NIV se pasa a través del canal BSC el cual modificará la cadena de bits con cierta probabilidad de error. Es aquí donde se pasa a la parte de detección y corrección de errores, que como ya se comentó tendrá un enfoque hacia el grupo de bits que representan el grupo 1.

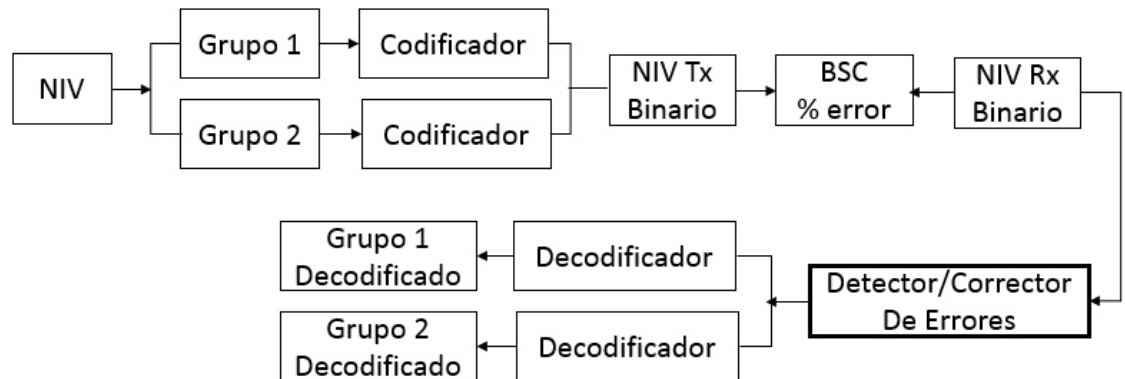


Figura 6.3.3 Diagrama de bloques con BSC.

Una vez dicho esto, se presenta a continuación la metodología para la detección y corrección de errores en la cadena de bits correspondientes a la combinación de símbolos desde la posición 1 hasta la posición 8 del NIV.

### 6.3.3 Detección y Corrección de errores (Grupo 1)

Después de pasar la cadena de bits a través de un canal BSC (Canal Simétrico Binario) con cierta probabilidad de error la cadena de bits saldrá del canal con algunos de sus elementos cambiados, esta nueva cadena entrará a un proceso de detección de errores. El factor principal para la decisión en alguna corrección que sea necesaria (al detectar un error) será la distancia de Hamming, por lo cual se describirá de manera breve a continuación.

#### Distancia Hamming.

En Teoría de la Información se denomina distancia de Hamming a la efectividad de los códigos de bloque y depende de la diferencia entre una palabra de código válida y otra. Cuanto mayor sea esta diferencia, menor es la posibilidad de que un código válido se transforme en otro código válido por una serie de errores. A esta diferencia se le llama distancia de Hamming, y se define como el número de bits que tienen que cambiarse para transformar una palabra de código válida en otra palabra de código válida.

Si dos palabras de código difieren en una distancia  $d$ , se necesitan  $d$  errores para convertir una en la otra.

La distancia de Hamming se denomina así gracias a su inventor Richard Hamming, profesor de la Universidad de Nebraska, que fue el que introdujo el término para establecer una métrica capaz de establecer un código para la detección y auto-corrección de códigos. Se emplea en la transmisión de información digitalizada para contar el número de desvíos en cadenas de igual longitud y estimar el error, por esto se denomina a veces como distancia de señal.

La distancia de Hamming tiene las siguientes propiedades:

1.  $d(a,b) = d(b,a)$
2.  $d(a,b) = 0$  si y sólo si  $a=b$
3.  $d(a,b) + d(b,c) \geq d(a,c)$

"d" es el nº de bits "p" en que son diferentes el mensaje emitido del recibido.

### **Proceso de detección y corrección de errores.**

Una vez teniendo la cadena de bits que sale del canal BSC el primer paso para la detección de errores en el Grupo 1 es aislar los bits que correspondan a dicho grupo, para esto se descartan los últimos 33 bits de la cadena a estudiar, que como ya se mencionó corresponden al Grupo 2 (posiciones 9 a 17 del NIV). Una vez teniendo identificada la cadena de bits con la que se trabajará se obtiene su longitud, es decir, de cuantos bits está formada dicha cadena. Esta longitud puede variar desde 1 hasta 9 bits, ya que es la cantidad de bits necesarios para codificar las combinaciones posibles con los primeros 8 símbolos de acuerdo al modelo propuesto. Así mismo, como ya se sabe, se tiene un libro de códigos en donde se encuentran las combinaciones válidas junto con su codificación binaria, los códigos en este libro también van desde 1 hasta 9 bits. El siguiente paso, una vez obtenida la longitud de la cadena de bits correspondiente al Grupo 1, es buscar en el libro de códigos todos aquellos códigos que tengan la misma longitud. Y se genera de manera atemporal un pequeño grupo de códigos en el cual todos tienen la misma longitud la cual corresponde a la longitud de bits del

grupo 1. Una vez teniendo estos elementos se procede a calcular la distancia Hamming de la cadena en validación con cada uno de los elementos del grupo generado de manera temporal. Si existe una distancia de Hamming igual 0 al momento de hacer la resta con alguno de los elementos, se asume que la cadena de bits, correspondiente al grupo 1, no sufrió cambio alguno al momento de pasar por el BSC y se toma esta parte del NIV como válida. En caso de que no existe alguna distancia Hamming cuyo valor haya sido 0, entonces se toma la cadena de bits validada como una cadena no válida, y se sustituye por la cadena de bits, en el grupo creado temporalmente, que haya dado la menor distancia Hamming. En caso de que existan 2 o más códigos en el grupo que den como resultado la misma distancia, se extraerán estos elementos para realizar un proceso de elección entre ellos. El proceso consiste en normalizar las probabilidades de cada uno de estos nuevos elementos seleccionados de manera que el total de sus sumas de 1. Cada uno de los elementos abarcará una región de 0 a 1 en base a su probabilidad, abarcando la región más cercana al 0 aquel código cuya probabilidad resulte mayor, y así se ocuparan de igual forma las regiones que vayan quedando. Por ejemplo, si las probabilidades de los nuevos elementos ya normalizadas son: elemento 1 un 60% y elemento 2 un 40%. La región del (0,0.6) la ocupará el elemento 1, mientras que la región (0.6, 1) la abarcará el elemento 2. Después de definir dichas regiones se genera un valor aleatorio (0,1) y en base al resultado se elige cuál de los códigos (elementos) reemplazará la cadena de bits con error.

En la Figura 6.3.4 se muestra un diagrama de flujo con la metodología de detección y corrección de errores para la cadena de bits correspondiente a los primeros 8 símbolos del NIV.

En la Figura 6.3.5 se presenta un diagrama el cual comienza con los bits que representan un NIV, los cuales pasan por un canal BSC el cual genera un error en la cadena de acuerdo a una probabilidad de error previamente asignada al canal. Después de esto son decodificados y validados con el fin de ver si se trata

de un NIV válido de acuerdo al modelo propuesto. Si resulta ser válido se comprobará si existe algún error entre los bits enviados y los recibidos y se contabilizará el error en caso de existir. En caso de no ser válido se realizará la corrección de errores necesaria de manera de que el resultado sea un NIV válido. Al corregirlo se revisará si la corrección fue correcta en base a los datos que se enviaron a través del BSC. Para ello se comprueba si existe algún error entre los bits enviados y los recibidos después de haber hecho la corrección y se contabiliza.

De esta forma se realiza así un programa computacional cuyo fin es detectar y corregir errores en los bits que se reciben correspondientes al grupo 1. Dicho programa funciona conforme lo presentado en esta sección y sirve para probar la eficiencia que tiene el detector/corrector de errores al momento de recibir trenes de bits que proceden de un canal con cierta probabilidad de error.

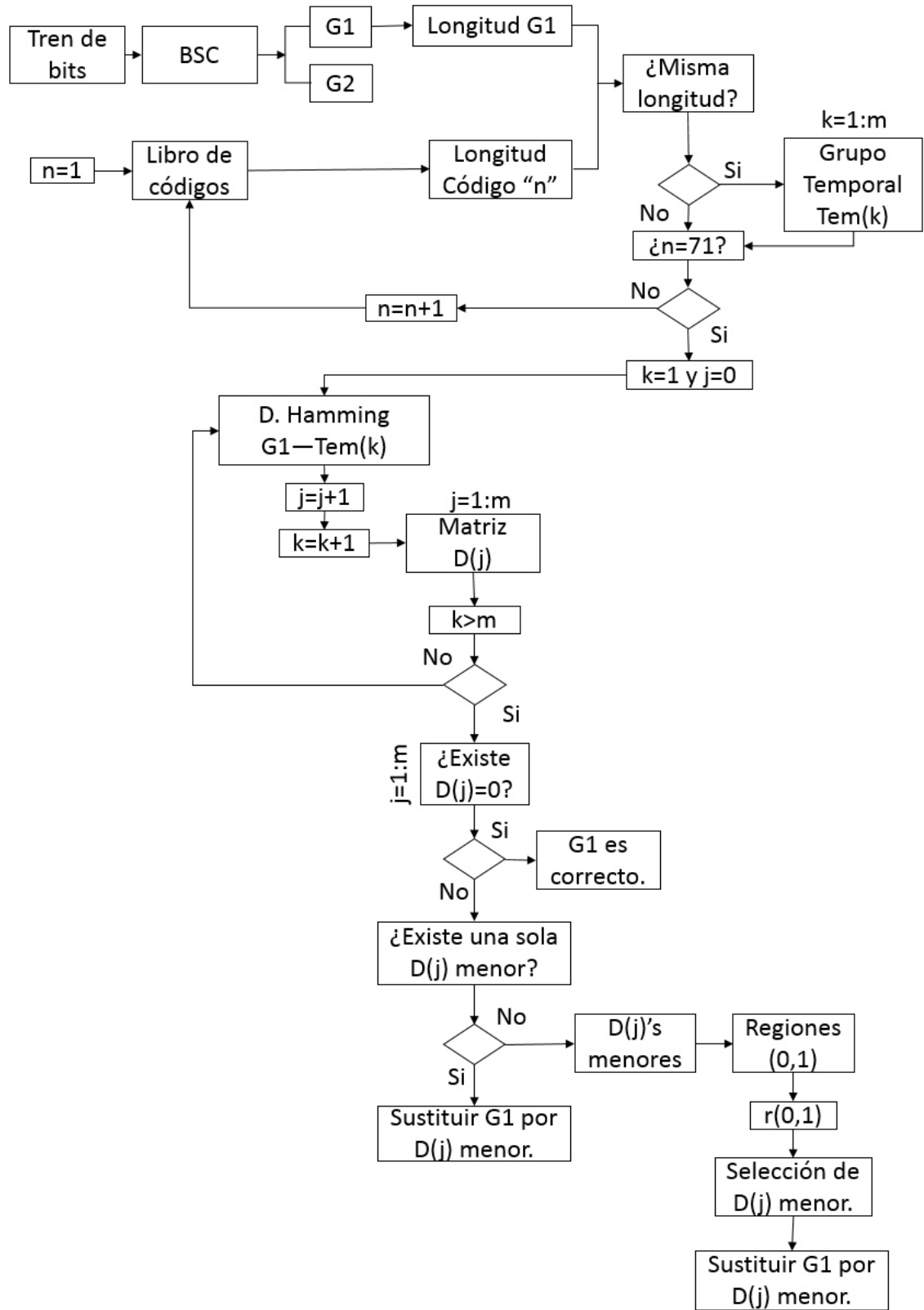


Figura 6.3.4 Diagrama de Detección y Corrección de Errores.



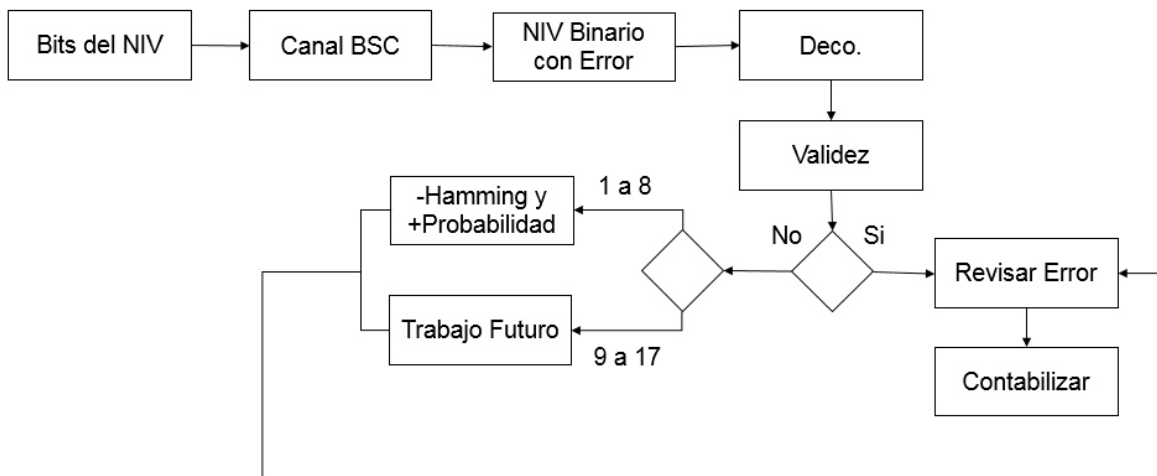


Figura 6.3.5 Diagrama de identificación de la etiqueta.

### Resumen.

En resumen para la detección de errores se sigue la siguiente metodología:

- ✓ Tomar el código binario después del BSC.
- ✓ Descartar los últimos 33 bits.
- ✓ Obtener la secuencia de bits del Grupo 1.
- ✓ Obtener la longitud.
- ✓ Obtener la distancia Hamming con aquellas palabras de igual longitud en el libro de códigos.
- ✓ Si existe un error ( $d > 0$ ) reemplazar la trama equivocada con aquella cuya distancia Hamming sea menor.

### 6.3.4 Pruebas

Para probar la eficiencia del corrector de errores para los primeros 8 dígitos se realizaron algunas pruebas. En dichas pruebas se pasaban una serie de NIVs generados por el simulador a través de un BSC al cual se le variaba la probabilidad de error. Primero se elige una probabilidad de error y a través de ese canal se pasa la trama de bits correspondiente a la codificación de un NIV. Después se toma el tren de bits que sale del canal BSC y se pasa por el detector de errores. En caso de que se detecte algún error en esta sección se procede a hacer la corrección del mismo. Al final se comprueba si la información que se

tenía antes del BSC es igual a la información al final del procedimiento de detección y corrección de errores. Si se detecta algún error se contabiliza. Este procedimiento se repite para todos aquellos NIVs que se hayan empleado en el experimento.

Después de realizar una serie de pruebas con distintas probabilidades de error, se obtiene como resultado que en promedio se corrigen alrededor del 50% o más de los errores iniciales. Resaltando que este porcentaje sólo indica la corrección en los primeros 8 bits, a pesar de que el tren de bits que pasa por el BSC es el del NIV completo.

**Ejemplo:**

Se hacen pasar 100 NIVs por un canal BSC con 10% de Ruido. En la TABLA 6.3.1 se muestran los resultados obtenidos de dicho experimento. La lectura de la tabla es la siguiente; de los 100 NIVs que se enviaron a través del BSC en los primeros 8 símbolos se detectaron 35 errores reales a la salida del canal, de estos 35 errores existentes 27 fueron detectados por el detector de errores, los otros 8 no se detectaron debido a que fueron falsos positivos, es decir, el tren de bits correspondiente a estos 8 casos presentó un cambio cuando salió del BSC pero siguió siendo un tren de bits válido de acuerdo al libro de códigos que se tiene. Los resultados de la tercera columna indican que, después de pasar el tren de bits que sale del BSC por el detector/corrector de errores, al momento de validarlos con la información antes del BSC se detectó que había 16 errores. Todos ellos falsos positivos ya que después del corrector de errores siempre se tendrá un tren de bits válido, más no correcto. Es decir de los 27 errores que se detectaron todos se corrigieron. Sólo que 19 se corrigieron de forma correcta y 8 se corrigieron como falsos positivos. Esto sucede cuando existe más de una distancia de hamming menor, ya que la elección del código que reemplazará la trama de bits errónea dependerá de una decisión aleatoria que toma como referencia las probabilidades de cada una de las posibles opciones de corrección.

TABLA 6.3.1 Resultados Ejemplo Detección/Corrección.

Errores:	Sin Corrección	Con Corrección
-Reales	35	16
-Detectados	27	0
-Falsos Positivos	8	16

En base a los resultados de la tabla anterior se puede ver que en promedio para este experimento en particular se corrigió un 54.28% de los errores netos al momento de usar el detector/corrector de errores para las primeras 8 posiciones del NIV.

Estos resultados anteriores son únicamente los obtenidos para el experimento del ejemplo, pero de igual forma se realizaron distintas pruebas variando únicamente el porcentaje de ruido en el Canal Simétrico Binario y se graficaron los resultados en la Figura 6.3.6, donde la línea azul (línea de arriba) representa los errores presentados antes del bloque de detección/corrección y la línea roja (línea de abajo) representa los errores después de pasar por dicho bloque.

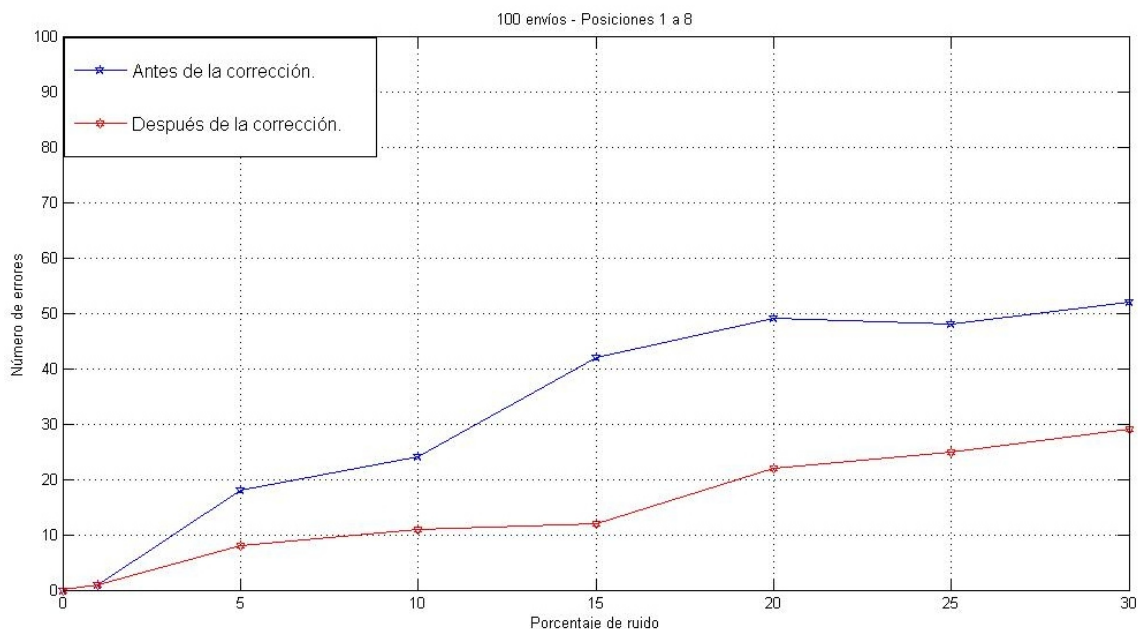


Figura 6.3.6 Errores antes y después de la corrección.

La Figura 6.3.7 muestra los mismos resultados que la Figura 6.3.6 sólo con un poco más de acercamiento en el eje vertical.

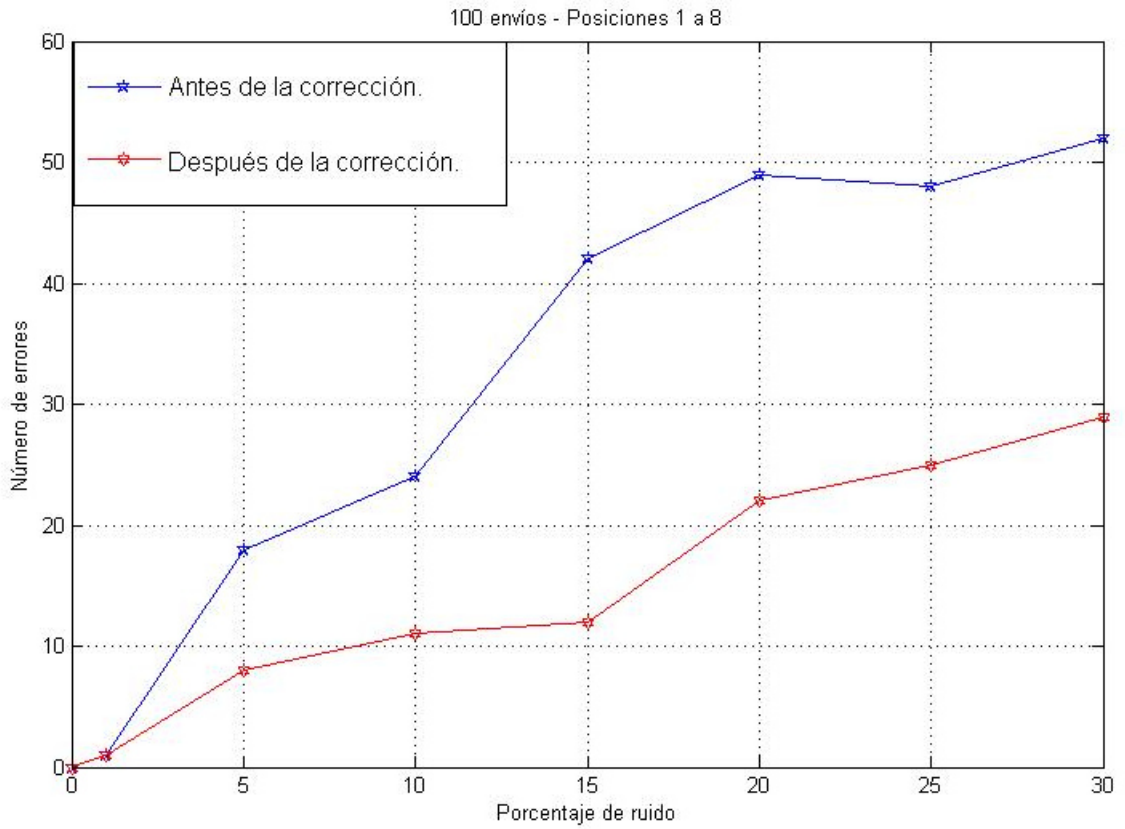


Figura 6.3.7 Errores antes y después de la corrección. (Acercamiento)

## Capítulo 7

# 7. RESULTADOS.

### 7.1 Resultados.

A continuación se muestra de manera breve una descripción de los resultados obtenidos a lo largo de este trabajo.

#### 7.1.1 Modelo Matemático.

Uno de los primeros resultados obtenidos a partir del análisis de las muestras realizado fue el diseño de modelos matemáticos para describir el comportamiento de la fuente de información. Los dos modelos presentados en este trabajo tienen como diferencia principal la generalidad.

El primer modelo (Modelo 1) obtenido no toma en cuenta condicionalidad ni dependencia entre los símbolos que conforman la estructura de un NIV, es decir, la ocurrencia de cualquier símbolo únicamente depende de la probabilidad que dicho símbolo tenga en el alfabeto del que forma parte, habiendo un alfabeto característico para cada una de las posiciones el NIV.

El segundo modelo (Modelo 2) a diferencia del primero, toma en cuenta la dependencia entre los símbolos que conforman el NIV. En este modelo la probabilidad de ocurrencia de un símbolo depende de dos cosas. El primer factor que determina la probabilidad de ocurrencia de un símbolo es la posición del NIV en donde se encuentre y el segundo factor que ayudará a determinar esta probabilidad será la combinación de símbolos que haya ocurrido antes de esta posición. Es decir, en una misma posición un mismo símbolo podrá tener distintas

probabilidades de ocurrencia e inclusive llegar a ser no válido de acuerdo al modelo, todo esto dependiendo del pasado que haya ocurrido en las combinaciones anteriores a dicho símbolo.

### **7.1.2 Programa Computacional.**

En base al Modelo 2 se desarrolló un programa computacional cuyo fin es simular la información que produce la fuente en estudio. Para ello se utilizaron las características obtenidas del análisis de una serie de muestras reales y se tomaron de ahí las probabilidades y dependencias. De esta manera en el plasmó el comportamiento de la fuente de información que se propuso en el Modelo 2, a manera de lograr generar muestras en base a simulación las cuales cumplieran de la mejor manera con las características presentadas en aquellas muestras obtenidas desde la fuente de información real.

### **7.1.3 Función de masa de probabilidad.**

Con la ayuda del programa computacional, desarrollado con el fin de simular la fuente de información, se genera una serie de muestras mayor a la que se tenía de las muestras reales (cien mil valores simulados). En base a estos datos generados se realiza un análisis de la función de masa de probabilidad obteniendo resultados que sirvieron como base para determinar una forma de codificación.

### **7.1.4 Codificación y compresión.**

En el tema de codificación y compresión se analizan y proponen diversos métodos que se considera pueden ser de utilidad según las características de la información de la fuente en estudio. Después del análisis de diversos métodos y en base a la comparación y resultados obtenidos de ellos se decide que el método con mejores resultados es el que lleva a cabo la codificación por grupos. El grupo uno comprende los caracteres 1 al 8 del NIV y se codifica mediante el método de Huffman. El grupo 2 comprende los caracteres de las posiciones 9 a la 17 y su codificación se lleva a cabo mediante una adaptación al método encontrado en

la literatura de manera que se obtengan mejores resultados para el problema en cuestión.

#### **7.1.5 Detección y corrección de errores.**

El tema de la detección y corrección de errores se trabajó para el primer grupo codificado (Posiciones 1 a la 8). Esto debido a que fue en este grupo de caracteres en donde se puede sacar mayor provecho del modelo propuesto. Debido a que se tiene un libro de códigos para cada una de las combinaciones válidas posibles con los primeros 8 caracteres, se utiliza la distancia Hamming como parámetro para la detección y corrección de errores en las cadenas de bits que corresponden a la codificación de esta parte del NIV.

## Capítulo 8

# 8. CONCLUSIONES.

### 8.1 Conclusiones

A continuación se muestran de manera breve algunas de las conclusiones obtenidas de este trabajo en base a los resultados obtenidos.

#### 8.1.1 Modelo de la fuente.

En la parte de modelo de la fuente se concluye que es posible crear un modelo matemático de la fuente de información tomando como base los resultados obtenidos del estudio de una serie de datos reales.

La aportación que se tiene en este apartado es que en la literatura no se encontraba un modelo el cual describiera el comportamiento de la fuente de información sobre la cual se desea trabajar. Por lo cual se llevó a cabo la tarea de diseñar dicho modelo para en base a él desarrollar los temas de codificación y compresión de ésta información para su manejo en un sistema de comunicaciones.

#### 8.1.2 Codificación y compresión.

En el tema de codificación y compresión de la información se puede concluir que la codificación propuesta que ofrece mejores resultados de compresión es aquella en la que se propone la codificación por grupos. Empleando el método de Huffman para el grupo 1, el cual comprende las combinaciones logradas con las posiciones 1 a la 8, y la adaptación del método encontrado en la literatura



(referente a codificación binaria de un NIV) para la codificación del grupo 2, que comprende las combinaciones posibles con las posiciones 9 a la 17 del NIV.

En resumen, se tomaron dos métodos de codificación y compresión que se tenían en la literatura. Se adaptaron estos métodos a las características de la fuente de información y se usaron en conjunto logrando con esto mejores resultados que los que se hubiesen obtenido al usar cualquiera de los dos métodos por separado para la fuente de información.

### **8.1.3 Robustez del método de codificación propuesto.**

De igual forma en el tema de codificación se observa que la codificación binaria propuesta para los NIVs tiene poca robustez. Por lo cual al existir un cambio en alguno de los bits que la conforman se pierde gran parte de la información al momento de llevar a cabo la decodificación. Para ello se hicieron pruebas utilizando un canal simétrico binario variando la probabilidad de error del mismo.

El método de codificación y compresión propuesto en este trabajo tiene poca robustez. Esto es debido a que no toma en cuenta las probabilidades de error que pudiesen existir en el canal sino que únicamente se enfoca en la manera de poder comprimir la información y decodificarla luego de haberla comprimido. Dejando de esta manera abierta la puerta para el estudio de los distintos métodos de protección de la información ante las inclemencias de un canal de comunicaciones.

### **8.1.4 Corrección de errores.**

En base a los resultados obtenidos se concluye que es posible corregir de manera correcta más del 50% de los errores presentados en el código correspondiente a las posiciones 1 a 8. Para realizar esta corrección se hace uso de la distancia de Hamming, sustituyendo la cadena de bits con errores por aquella cadena válida con la cual la distancia de Hamming sea menor.

Por otro lado también se llegó a la conclusión de que el cambio en alguno de los bits del grupo 2, por lo general, resulta en un falso positivo. Esto debido a que no se encontró una manera fiable de detectar errores al momento de obtener la cadena de bits después del canal BSC. Ya que la decodificación de estas cadenas resultan en combinaciones válidas, más no en aquellas combinaciones de símbolos que se intentó transmitir.

Como trabajo futuro se tiene diseñar un esquema de protección de la información ante las inclemencias de un canal de comunicaciones, de manera que sea posible utilizar los bits que se ahorraron al momento de la compresión para agregar redundancia y robustez a la información que se transmitirá. En este punto se cree que es posible realizar una codificación conjunta canal-fuente una vez modelado el canal de comunicaciones que se asemeje más a la realidad en la que se desea implementar la aplicación de identificación vehicular.

## Capítulo 9

# 9. REFERENCIAS BIBLIOGRÁFICAS.

1. Dobkin, D.M., *The RF in RFID: passive UHF RFID in practice*.2008.
2. Pala, Z.Y.Y., *Smart Parking Applications Using RFID Technology*. 2007.
3. Kubota, S., Y. Okamoto, and H. Oda. *Safety driving support system using RFID for prevention of pedestrian-involved accidents*. in *ITS Telecommunications Proceedings, 2006 6th International Conference on*. 2006. IEEE.
4. Wang, F., S. Liu, and P. Liu, *A temporal RFID data model for querying physical objects*. *Pervasive and Mobile Computing*, 2010. **6**(3): p. 382-397.
5. *ISO 3779:2009 Road vehicles -- Vehicle identification number (VIN) -- Content and structure*. 2009.
6. *ISO 3780:2009 Road vehicles -- World manufacturer identifier (WMI) code*. 2009.
7. *NOM-001-SSP-2008, Para la determinación, asignación e instalación del número de identificación vehicular*. Diario Oficial de la Federación, 2010.
8. *Vehicle Identification Numbers (VINs)*. National Highway Traffic Safety Administration, 2011.
9. Azpilicueta Fernández de las Heras, L., *Estudio sobre la codificación conjunta canal-fuente en sistema UHF RFID*. 2009.
10. Hashim, H., *Binary Encoding of Vehicle Identification Number*. Research & Advanced Engineering, Ford Motor Company, 2009.
11. Altug, Y., S.S. Kozat, and M.K. Mihcak, *An Information Theoretic Analysis of Single Transceiver Passive RFID Networks*. arXiv preprint arXiv:0901.0643, 2009.
12. Cover, T.M. and J.A. Thomas, *Elements of information theory*2006: Wiley-interscience.
13. INEGI, *Vehículos de motor registrados en circulación (excluye motocicletas)*. Estadísticas de Vehículos de Motor Registrados, 2012.
14. *PROY-NOM-001-SCT-2-2012. Placas metálicas, calcomanías de identificación y tarjetas de circulación empleadas en automóviles, autobuses, camiones, midibuses, motocicletas y remolques matriculados en la República Mexicana, licencia federal de conductor, calcomanía de verificación físico-mecánica, listado de series asignadas por tipo de vehículo, servicio y entidad federativa o dependencia de gobierno, especificaciones y método de prueba*. Diario Oficial de la Federación, 2013.

15. *Vehículos Robados*. Procuraduría General de Justicia, 2013. **Nuevo León**.
16. Sayood, K., *Introduction to data compression* 2012: Morgan Kaufmann.
17. Bertsekas Dimitri, T.J., *Introduction to Probability*. 2008. **2nd Edition**.
18. *NMX-Z-12/1-1987 Muestreo para la inspección por atributos. Parte 1: Información general y aplicaciones. Sampling for inspection by attributes. Part 1: General Information and Applications*. Normas Mexicanas. Dirección General de Normas, 1987.
19. Gersho, A. and R.M. Gray, *Vector quantization and signal compression* 1992: Springer.

# Índice de Figuras

Figura 1.1.1 Bandas de frecuencia de RFID. (Referencia tomada de [1]) .....	1
Figura 1.1.2 Elementos de un sistema RFID. ....	3
Figura 1.1.3 Organización y Terminología de la Memoria en una Etiqueta RFID Gen 2.(Referencia tomada de [1]).....	8
Figura 2.1.1 Recepción de mensajes y selección de información.....	15
Figura 2.1.2 Diagrama de bloques para el sistema de comunicación multiusuario en [11].....	17
Figura 4.1.1 Secciones del NIV según la NOM [7].....	22
Figura 4.2.1 Elementos de una etiqueta UHF pasiva.(Referencia tomada de [1]) .....	26
Figura 4.2.2 Etiqueta pasiva.(Referencia tomada de [1]) .....	26
Figura 4.2.3 Representación esquemática de una etiqueta RFID pasiva simple. (Referencia tomada de [1]) .....	26
Figura 4.4.1 Clasificación: longitud. ....	28
Figura 4.4.2 Clasificación: Lugar de Fabricación. ....	29
Figura 4.4.3 Clasificación: Año(s). ....	30
Figura 5.2.1 Diagrama de árbol de probabilidades. ....	55
Figura 5.2.2 Parte A.....	56
Figura 5.2.3 Parte B.....	57
Figura 5.2.4 Inicio del programa. Selección de P2.....	61
Figura 5.2.5 Diagrama de bloques del simulador de datos. ....	62
Figura 5.2.6 Diagrama de flujo del simulador. ....	63
Figura 5.2.7 Función de Masa de Probabilidad obtenida de las muestras de la simulación.....	64
Figura 5.2.8 PMF 2ª Posición. ....	67
Figura 5.2.9 PMF 3ª Posición. ....	67
Figura 5.2.10 PMF 4ª Posición. ....	68
Figura 5.2.11 PMF 5ª Posición. ....	68

Figura 5.2.12 PMF 6 <sup>a</sup> Posición. ....	69
Figura 5.2.13 PMF 7 <sup>a</sup> Posición. ....	69
Figura 5.2.14 PMF 8 <sup>a</sup> Posición. ....	70
Figura 5.2.15 PMF 11 <sup>a</sup> Posición. ....	70
Figura 6.1.1 Bits para codificación en ASCII.....	73
Figura 6.1.2 Codificación binaria de un NIV-ISO .....	74
Figura 6.1.3 Codificación binaria de un NIV-ISO_2 .....	75
Figura 6.1.4 Un código de Huffman (Ejemplo tomado de [19]) .....	79
Figura 6.1.5 Codificación Huffman.....	82
Figura 6.1.6 Ejemplo Código Aritmético.....	84
Figura 6.1.7 Codificación binaria de un NIV-NOM .....	87
Figura 6.1.8 Codificación binaria de un NIV-Modelo 2.....	91
Figura 6.1.9 Codificación por grupos. ....	92
Figura 6.1.10 Codificación Grupo 1. ....	96
Figura 6.1.11 Codificación Grupo 2. ....	97
Figura 6.1.12 Codificación con Grupos.....	98
Figura 6.1.13 Codificación Grupo 2 Propuesta. ....	100
Figura 6.1.14 Codificación con grupos (2) .....	100
Figura 6.2.1 Diagrama de bloques del Codificador. ....	101
Figura 6.2.2 Diagrama de bloques del Decodificador. ....	103
Figura 6.3.1 Canal Simétrico Binario .....	104
Figura 6.3.2 Falsos Negativo y Positivo.....	106
Figura 6.3.3 Diagrama de bloques con BSC.....	107
Figura 6.3.4 Diagrama de Detección y Corrección de Errores.....	111
Figura 6.3.5 Diagrama de identificación de la etiqueta. ....	112
Figura 6.3.6 Errores antes y después de la corrección.....	114
Figura 6.3.7 Errores antes y después de la corrección. (Acercamiento).....	115

# Índice de TABLAS

TABLA 1.4.1 Contribuciones del trabajo. ....	12
TABLA 4.1.1 Secciones que conforman el NIV [7].....	23
TABLA 4.1.2 10ma Posición del NIV .....	24
TABLA 5.1.1 Probabilidad posiciones 2,3, y 4. ....	34
TABLA 5.1.2 Probabilidad posiciones 5,6 y 7. ....	35
TABLA 5.1.3 Probabilidad posiciones 8 y 11. ....	35
TABLA 5.1.4 Probabilidad posición 9.....	36
TABLA 5.1.5 Probabilidades Posiciones 10,12-17.....	37
TABLA 5.2.1 Posición 2 Modelo 2. ....	41
TABLA 5.2.2 Posición 3 Modelo 2. ....	41
TABLA 5.2.3 Posición 4 Modelo 2. ....	42
TABLA 5.2.4 Posición 5 Modelo 2. ....	43
TABLA 5.2.5 Posición 6 Modelo 2. ....	44
TABLA 5.2.6 Posición 7 Modelo 2. ....	45
TABLA 5.2.7 Posición 8 Modelo 2. ....	46
TABLA 5.2.8 Posición 9 Modelo 2. ....	47
TABLA 5.2.9 Posición 10 Modelo 2. ....	48
TABLA 5.2.10 Posición 11 Modelo 2. ....	50
TABLA 5.2.11 Posición 12 Modelo 2. ....	50
TABLA 5.2.12 Posiciones 13-17 Modelo 2. ....	51
TABLA 5.2.13 Posibles valores según la posición en el NIV. ....	53
TABLA 5.2.14 Entropía del NIV. ....	60
TABLA 5.2.15 Relación de valores del eje horizontal. ....	66
TABLA 6.1.1 Valores numéricos para Codificación Binaria. ....	75
TABLA 6.1.2 Alfabetos por posición en el NIV.....	91
TABLA 6.1.3 Codificación 1-8. ....	95
TABLA 6.1.4 Alfabetos posiciones 9-17.....	97
TABLA 6.1.5 Valores numéricos para Codificación Binaria_2. ....	99

TABLA 6.3.1 Resultados Ejemplo Detección/Corrección.....	114
--	-----