



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

**FACULTAD DE INGENIERÍA
INGENIERÍA EN TELECOMUNICACIONES**

**SIMULACIÓN DE LA CUANTIZACIÓN VECTORIAL
EN UN SISTEMA DE COMUNICACIONES**

T E S I S

**QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN TELECOMUNICACIONES**

**PRESENTA
RAMÍREZ GUTIÉRREZ EMMANUEL**

**ASESOR
DR. MIGUEL MOCTEZUMA FLORES**



MÉXICO, D.F.

2010



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

SIMULACIÓN DE LA CUANTIZACIÓN VECTORIAL EN UN SISTEMA DE COMUNICACIONES



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

*A mi familia, que a lo largo de estos años
nunca dejó de confiar en mí...*

*A la UNAM por darme la oportunidad y el orgullo
de pertenecer a esta honorable Universidad.*

*A mis profesores que a lo largo de estos años
ayudaron a mi formación y siempre brindaron su conocimiento.*

*A mis amigos que siempre compartieron
los buenos y malos momentos conmigo.*

*Al Dr. Miguel Moctezuma, por haber
confiado en mí para la realización de este trabajo.*



La presente tesis fue financiada mediante beca recibida del PROGRAMA DE APOYO A PROYECTOS DE INVESTIGACIÓN E INNOVACIÓN TECNOLÓGICA, PAPIIT, proyecto IN104708.

PRÓLOGO

El presente trabajo se enfoca a guiar al lector desde las bases más simples y básicas de la *Teoría de Probabilidad*, hasta conceptos más avanzados en donde se puede apreciar la aplicación de éstos para procesos que toman importancia en la transmisión real de señales por un sistema de comunicaciones, de forma más específica, en todos aquellos aspectos que afectan la transformación de señales analógicas a digitales y los procesos, algoritmos, problemáticas y posibles soluciones en estas etapas.

El contenido da pie a analizar las ventajas y desventajas de ciertos algoritmos que permiten realizar el proceso de cuantización de una manera óptima cuyas características finales satisfagan las condiciones iniciales del sistema. Para ello se incluyen simulaciones que representan a dos de los algoritmos más eficientes en cuanto a cuantización vectorial se refiere, y sus resultados correspondientes los cuales serán capaces de representar cualquier señal de entrada sin importar su distribución de probabilidad.

Se pretende que el contenido de cada uno de los capítulos sirva de consulta para alumnos y profesores interesados en entender de forma detalladamente la metodología y forma de aplicación de estos algoritmos de cuantización y las posibles variaciones que afecten su diseño y las repercusiones que estas se vean incluidas en los resultados finales.

La información está organizada de tal forma que al inicio se conozcan las bases y conceptos probabilísticos, así como la problemática a resolver en el canal de comunicaciones, para dar paso a la metodología de los algoritmos seleccionados para dar solución al planteamiento del problema inicial.

Además, se propone un trabajo práctico ya que a lo largo del contenido se presentarán las características específicas para obtener los resultados buscados, lo cual permitirá realizar variedad de simulaciones y a su vez, se cuente con gran diversidad de resultados aplicables de forma real a diferentes tipos de sistemas. Para tal fin se adjuntan los *anexos* los cuales contienen los resultados en forma numérica y gráfica.

LISTA DE FIGURAS

Capítulo 2

2.1 Muestreo de una Variable Aleatoria	9
2.2 Variable Aleatoria Continua	10
2.3 Variable Aleatoria Discreta	10
2.4 Función Acumulativa de Probabilidad, CDF	12
2.5 Función de Densidad de Probabilidad, PDF	13
2.6 Distribución Gaussiana (Normal)	17
2.7 Procesos estocásticos	20

Capítulo 3

3.1 Sistema de Comunicaciones	27
3.2 Gráfica correspondiente a la Entropía $H(X)$	30
3.3 Información Mutua $I(E_1;E_2)$	31
3.4 Diagrama de bloques de un Sistema de Comunicaciones	32
3.5 Canal Binario Simétrico	33

Capítulo 4

4.1 Proceso de Digitalización	44
4.2 Ejemplo de Cuantizador de 3 bits	45
4.3 Cuantizador Uniforme Midrise	46
4.4 Probabilidades mayores cercanas al origen	47
4.5 Cuantización Vectorial	50
4.6 Regiones de Voronoi	52
4.7 Principio del Vecino Más Cercano	54
4.8 Principio del Centroide	55
4.9 Descripción del algoritmo LGB	57
4.10 Descripción del algoritmo COVQ	59

Capítulo 5

5.1 Diagrama representativo del Algoritmo LGB	64
5.2.a) Secuencia aleatoria de 10 000 números en el intervalo [0,100]	66
5.2.b) Secuencia aleatoria de 100 números en el intervalo [0,1]	66
5.3.a) Secuencia aleatoria de 10 000 números en el intervalo [0,100]	67
5.3.b) Secuencia aleatoria de 100 números en el intervalo [0,1]	67
5.4.a) Secuencia Original	68
5.4.b) Centroides Iniciales	68
5.4.c) Frontera Inicial	68
5.5.1 Representación gráfica del codebook final	68
5.5.2 Centroides en el intervalo [0,1]	69
5.6 Representación del Algoritmo COVQ	72
5.7 Vectores aleatorios en dimensión 2	74
5.8 Vectores aleatorios en dimensión 2	74
5.9 Vectores aleatorios en dimensión 2 con el centroide inicial	75
5.10 Resultado de la primera iteración de Lloyd	76
5.11 Representación gráfica del algoritmo COVQ	77
5.12 Elementos generados en el intervalo [0,1]	77
5.13 Relación Distorsión vs elementos finales	81
5.14 Relación elementos finales vs iniciales para $D=0.003$	81
5.15 Relación elementos finales vs iniciales para $D=0.5$	82
5.16 Relación elementos finales vs iniciales para $D=1$	83
5.17 Elementos finales por intervalo	83
5.18 Relación Distorsión vs elementos finales para [0,100]	84
5.19 Relación Distorsión vs elementos finales para [0,10]	85
5.20 Relación Distorsión vs elementos finales para [0,1]	85
5.21 Distribución normal de 1000 elementos	87
5.22 Distribución normal de 1000 elementos	88
5.23.a) Centroides para una distribución normal	89
5.23.b) Centroides para una distribución uniforme	89
5.24 Centroides para una distribución normal	90
5.25 Centroides para una distribución normal [0,1]	91
5.26 Centroides para una distribución normal [0,3]	92

5.27 Centroides para una distribución normal de 500 elementos [-3,3]	92
5.28 Centroides para una distribución normal de 500 elementos [0,4]	93
5.29 Centroides para una distribución normal de 500 elementos [0,2]	94
5.30 Relación distorsión vs elementos finales para [0,5]	96
5.31.a) Cuantización LGB correcta	96
5.31.b) Cuantización LGB incorrecta	96
5.32.a) Cuantización COVQ correcta	97
5.32.b) Cuantización COVQ incorrecta	97
5.33 Relación distorsión vs elementos finales para [0,1]	98
5.34 Secuencia con distribución normal de 1000 elementos en 2D	99
5.35 Diccionario para 1000 elementos y $D=0.003$	100
5.36 Diccionario para 1000 elementos y $D=0.003$ en [-1,1]	101
5.37 Secuencia con distribución normal de 500 elementos	102
5.38 Diccionario para 500 elementos y $D=0.003$	102

LISTA DE TABLAS

Capítulo 3

3.1 Código de Shannon	38
-----------------------	----

Capítulo 5

5.1 Resultados finales del algoritmo LGB	80
5.2 Características de los diccionarios seleccionados	86
5.3 Resultados finales del algoritmo COVQ	95
5.4 Características de los diccionarios seleccionados	98

GLOSARIO DE SÍMBOLOS

A_i	Conjunto
C	Capacidad del canal
c_i	Centroide
d_{min}	Distancia mínima
$d_H(,)$	Distancia de Hamming
$d_E(,)$	Distancia de Euclidean
$E\{\}$	Media de un evento
f	Frontera
$f\{\}$	Función de densidad
$F\{\}$	Función de distribución
$H()$	Entropía
$I(,)$	Información mutua
k	Información del bit
$K_x()$	Autocovarianza
L^a	Longitud media
n	Símbolos del canal
p	Probabilidad de ocurrencia
$P\{ \}$	Probabilidad de un evento
$R(D)$	función de tasa de distorsión entre la fuente y el canal
$R_S(D)$	función de distorsión de la fuente
R_S	Velocidad de transferencia
R_X	Rango de una variable
R_{XY}	Correlación
R_{x1x2}	Autocorrelación
θ_x	Principio del vecino más cercano
U	Vectores de entrada
V	Vectores de salida
V_Y	Principio del centroide

LISTA DE ABREVIATURAS

<i>A/D</i>	Conversión Analógico Digital
<i>BER</i>	Tasa de error
<i>BSC</i>	Canal simétrico binario
<i>CDF</i>	Función acumulativa de distribución de probabilidad
<i>CE</i>	Cuantización escalar
<i>COSQ</i>	esquema de canal optimizado por cuantización escalar
<i>COVQ</i>	esquema de canal optimizado por cuantización vectorial
<i>CV</i>	Cuantización vectorial
<i>D/A</i>	Conversión Digital Analógico
<i>LBG</i>	Algoritmo Linde, Buzo y Gray
<i>MSE</i>	Error medio cuadrático
<i>OPTA</i>	funcionalidad teórica óptima
<i>PDF</i>	Función de densidad de probabilidad
<i>SNR</i>	Relación señal a ruido
<i>VA</i>	Variable aleatoria

CONTENIDO

Prólogo.....	i
Lista de figuras.....	ii
Lista de tablas.....	iv
Glosario de símbolos.....	v
Lista de abreviaturas.....	vi
Contenido.....	vii
I. INTRODUCCIÓN	1
1.1 Objetivo.....	1
1.1.1 Planteamiento del problema.....	2
1.1.2 Esquema de análisis.....	2
1.1.3 Campo de aplicación.....	3
1.1.4 Aportaciones de la tesis.....	4
II. PROCESOS ALEATORIOS	6
2.1 Axiomas de probabilidad.....	6
2.2 Variables aleatorias.....	7
2.2.1 Definición.....	8
2.2.2 Funciones de densidad y de distribución.....	11
2.2.3 Funciones de distribución de probabilidad.....	15
2.3 Ley de los grandes números.....	17
2.4 Procesos estocásticos.....	19
2.4.1 Conceptos.....	22
2.4.2 Estacionaridad y ergodicidad.....	24

III. SISTEMAS DE COMUNICACIONES	27
3.1 Medidas de información.....	28
3.1.1 Entropía.....	28
3.1.2 Información Mutua.....	31
3.2 Consideraciones y esquemas de análisis.....	32
3.3 Capacidad del canal.....	34
3.4 Codificación de fuente.....	36
3.4.1 Conceptos de Shannon.....	37
3.4.2 Medidas de distorsión.....	39
3.5 Funcionalidad teóricamente práctica.....	41
IV. CUANTIZACIÓN VECTORIAL	43
4.1 Fundamentos.....	44
4.2 El algoritmo Lloyd Max generalizado.....	50
4.2.1 Conceptos.....	51
4.2.2 El principio del vecino más cercano.....	53
4.2.3 El principio del centroide.....	55
4.3 Esquema de diseño LGB.....	56
4.4 Esquemas COSQ y COVQ.....	58
4.5 Esquema comparativo.....	59
V. SIMULACIÓN DE LA CUANTIZACIÓN ESCALAR Y VECTORIAL	62
5.1 Algoritmo para la Cuantización Escalar.....	63
5.2 Algoritmo para la Cuantización Vectorial.....	71
5.3 Esquema comparativo.....	78
5.3.1 Cuantización escalar.....	78
5.3.2 LGB: Distribución normal.....	86
5.3.3 Cuantización vectorial.....	95
5.3.4 COVQ: Distribución normal.....	99

5.4 Resultados.....	103
VI. CONCLUSIONES.....	106
ANEXOS	
Anexo 1: Cuantización Escalar (Distribución Uniforme).....	111
Anexo 2: Algoritmo LGB (Distribución Uniforme).....	120
Anexo 3: Cuantización Escalar (Distribución Normal).....	132
Anexo 4: Algoritmo LGB (Distribución Normal).....	134
Anexo 5: Cuantización Vectorial (Distribución Uniforme).....	144
Anexo 6: Algoritmo COVQ (Distribución Uniforme).....	153
Anexo 7: Cuantización Vectorial (Distribución Normal).....	159
Anexo 8: Algoritmo COVQ (Distribución Normal).....	168
BIBLIOGRAFÍA.....	178

Capítulo 1

INTRODUCCIÓN

La ‘cuantización’ es un proceso que ha tomado gran importancia en las comunicaciones en general desde el momento en que las comunicaciones analógicas comenzaron a ser desplazadas dándole paso a las comunicaciones digitales ya que dicho proceso forma parte del método de digitalización de las señales, el cual comprende a su vez los procesos conocidos como muestreo y codificación. La ‘cuantización’ es un proceso que puede llevarse a cabo mediante una gran cantidad de métodos basados en diferentes algoritmos dependiendo de las condiciones que se establezcan para el funcionamiento del sistema.

Además es uno de los procesos que se realizan desde el inicio del sistema y que se mantiene a lo largo de su vida útil.

1.1 OBJETIVO

El siguiente trabajo de tesis tiene como objetivos primordiales investigar y comprender los términos, métodos, algoritmos y características de la cuantización de tipo vectorial, entendiendo como primer paso el proceso más simple que lo antecede, la cuantización escalar.

También se pretende comparar los resultados de los diferentes algoritmos de cuantización utilizados y a partir de ellos, seleccionar aquel resultado que muestre las características más óptimas para su implementación dentro de un canal de comunicaciones.

Finalmente se tiene como objetivo presentar al lector todos los resultados obtenidos a partir de las simulaciones correspondientes a los algoritmos seleccionados con la finalidad de que éste

pueda ser capaz de analizarlos y decidir cual de ellos se ajusta a las características que se busquen en algún sistema en específico.

1.1.1 PLANTEAMIENTO DEL PROBLEMA

La transmisión de información en un canal de comunicaciones enfrenta diferentes formas de degradación generada por una gran cantidad de factores que pueden ser ocasionados tanto en la propia generación de la señal a enviar, como en el receptor y trasmisor, sin dejar de lado los factores que la afectan a lo largo del canal de comunicaciones. Desde el punto de vista de Ingeniería, la degradación se modela como ruido y los esfuerzos de solución se agrupan en técnicas de codificación de fuente y codificación de canal. El proceso confiable de transmisión y recepción en un canal de comunicaciones es hoy en día, un problema solo parcialmente resuelto.

La *cuantización vectorial* es un procedimiento utilizado para eliminar redundancia, ampliamente usado en técnicas de compresión de datos. Esta emplea con efectividad cuatro propiedades relacionadas de representaciones: dependencia lineal (correlación), dependencias no lineales, forma de la función de densidad de probabilidad y dimensión del vector.

Por lo tanto, se plantea establecer las condiciones óptimas para obtener mediante diferentes algoritmos la cuantización que muestre un mejor y eficiente funcionamiento para cualquier canal de comunicaciones y que al mismo tiempo muestre una disminución en las degradaciones que la señal de entrada pueda sufrir a lo largo de su paso por el sistema de comunicaciones.

1.1.2 ESQUEMA DE ANÁLISIS

Los trabajos de Shannon establecieron los fundamentos modernos en el estudio de los sistemas de comunicaciones. La cuantización de la información y la medida de la capacidad de un canal de comunicaciones adquirieron sentido a partir de 1948.

El grado de incertidumbre y la ocurrencia probabilística de los símbolos de un mensaje tomaron forma en los teoremas de codificación de fuente y canal de Shannon, aplicados hoy en día a la codificación de señales digitales de audio y video. Las referencias al estudio formal en el diseño de la cuantización vectorial iniciaron con el trabajo de Linde, Buzo y Gray (LGB). Históricamente a la cuantización vectorial se le conoce como el algoritmo generalizado de Lloyd (GLA) o el algoritmo de k-medias.

Con aplicación a canales ruidosos de comunicación, esta propuesta se enfoca en el estudio de las técnicas de codificación de fuente empleando la cuantización tanto escalar como vectorial. El objetivo es realizar una compresión de datos que permita una transmisión a baja tasa o que permita un mejor almacenamiento en medios tradicionales. El empleo de los métodos para determinar la cuantización vectorial en lugar de la cuantización escalar resulta en una mejora en la funcionalidad al comparar ambas alternativas en términos de menor tasa de bits (*bit rate*).

1.1.3 CAMPO DE APLICACIÓN

El presente trabajo estará enfocado a mostrar todos los elementos y conceptos básicos de probabilidad para entender de una forma sencilla el proceso de digitalización de señales así como su importancia en las comunicaciones, y de manera más específica, el proceso de la cuantización.

Se conocerán los detalles del funcionamiento de algoritmos de cuantización tanto escalar y vectorial, refiriéndose de manera más específica a los algoritmos LGB para el caso escalar y al algoritmo COVQ para la vectorial.

Se busca que, a partir del conocimiento de la definición y metodología de cada uno de estos algoritmos se pueda diseñar una base de datos a partir de las condiciones más óptimas posibles para el correcto funcionamiento del sistema de comunicaciones.

También se buscará inicialmente simular el proceso de funcionamiento del algoritmo LGB de la cuantización escalar a partir de secuencias aleatorias de entrada con diferentes distribuciones de probabilidad (distribuciones uniforme y normal), observado de forma detallada los resultados y características obtenidas para cada uno de los casos, señalando en todo caso las posibles ventajas y desventajas, indicando al final cual de ellos cumplió con las características solicitadas.

Una vez establecidas las bases de la cuantización escalar, se buscará ampliar el campo de investigación creando una nueva base de datos que corresponda al algoritmo que representa a la cuantización vectorial.

Esta simulación se llevará a cabo siguiendo la misma metodología que se utilice en el caso de la cuantización escalar manejando en cada caso las dos diferentes formas de distribución de probabilidad mencionadas anteriormente. Además se realizará una comparación exhaustiva entre los resultados obtenidos en cada caso y entre la escalar contra la vectorial.

Finalmente se adjuntarán los resultados obtenidos después de la variación de los parámetros de entrada para ambos casos, a fin de que puedan ser de utilidad al lector y puedan llegar a ser utilizados en una aplicación real.

1.1.4 APORTACIONES DE LA TESIS

El siguiente trabajo de tesis pretende realizar las siguientes aportaciones una vez que se hayan obtenido los resultados buscados:

- Se pretende que la tesis sea capaz de explicar de forma clara y detallada la problemática que existe en la codificación de fuente y canal y las diversas perturbaciones que se adhieren a la señal en su paso por el sistema de comunicaciones, así como presentar las posibles soluciones a estos problemas.
- Presentar de forma detallada las simulaciones que representen el comportamiento de las señales en los diferentes algoritmos estudiados de la cuantización escalar y vectorial, LGB y COVQ respectivamente.
- Partiendo de los resultados observados, seleccionar aquel que sea considerado el mejor a partir de sus características iniciales y finales y observar detalladamente las características para su implementación.
- Presentar los resultados obtenidos para hacer hincapié en las ventajas y desventajas de cada uno.
- Basándose en los conceptos básicos de probabilidad, se explicará el funcionamiento y la importancia de la digitalización y por lo tanto el de la cuantización en dicho proceso.
- Presentar un reporte final que contenga la mayoría de las posibles variaciones que puedan tener los procesos de cuantización escalar y vectorial en forma numérica y en forma gráfica.

Capítulo 2

PROCESOS ALEATORIOS

Los '*procesos aleatorios*' son importantes porque en casi todos los aspectos de la vida se presentan este tipo de situaciones en donde el comportamiento de un fenómeno o evento no puede ser predicho, por tal motivo, solo puede especularse acerca de los posibles efectos que tendrá. A continuación se describirán las principales características de estos procesos.

Además, la importancia de la probabilidad dentro de los procesos aleatorios radica en que estos son una extensión de la probabilidad misma.

La mayoría de los procesos son dependientes al tiempo, por tal motivo, una señal aleatoria es una función del tiempo, y debido a esto, aquellas funciones que presentan esta dependencia son llamados '*procesos aleatorios*'.

En el presente capítulo se hablará de los principales conceptos que describen el comportamiento de los procesos aleatorios, así como sus principales características.

2.1 AXIOMAS DE PROBABILIDAD

Los axiomas dentro de la teoría de probabilidad tienen gran importancia, debido a que con estas leyes, puede explicarse toda la teoría de probabilidad. Por tal motivo es importante inicial este capítulo enunciándolos, ya que serán utilizados a lo largo del presente tema.

Para poder entender los axiomas, se debe tener en cuenta la siguiente situación: si se tienen los conjuntos A_1, A_2, \dots, A_n se dice que para que estos sean disjuntos o exclusivos se debe cumplir que:

$$A_i \cap A_j = \emptyset \quad \forall i \neq j \quad (2.1)$$

Los axiomas de probabilidad se enumeran a continuación:

- *Axioma 1 (No negatividad)*: La probabilidad de cualquier evento A siempre se encontrará contenida entre 0 y 1, por lo que no existen las probabilidades negativas:

$$0 \leq P\{A\} \leq 1 \quad (2.2)$$

- *Axioma 2 (Unidad)*: La probabilidad total de un evento tiene una probabilidad unitaria:

$$P\{S\} = 1 \quad (2.3)$$

- *Axioma 3 (Adición finita)*: Si se cumple que A_1, A_2, \dots, A_n son eventos disjuntos o que no presentan intersección alguna, entonces

$$P\{A_1 \cup A_2 \cup \dots \cup A_n\} \triangleq P\left\{\bigcup_{i=1}^n A_i\right\} = \sum_{i=1}^n P\{A_i\} \quad (2.4)$$

- *Axioma 3' (Adición contable)*: Si se cumple que A_1, A_2, \dots son eventos disjuntos, entonces

$$P\left\{\bigcup_{i=1}^{\infty} A_i\right\} = \sum_{i=1}^{\infty} P\{A_i\} \quad (2.5)$$

2.2 VARIABLE ALEATORIA

El concepto de variable aleatoria puede definirse de diferentes maneras, es decir, se dice que una variable aleatoria es una *descripción numérica* de los resultados de un experimento, cuyo valor es incierto hasta el momento antes de realizar el experimento. Es una *función* de los elementos de cierto espacio, representado de forma general por una letra mayúscula, (dicho espacio está formado por un conjunto de muestras, descrito por letras minúsculas).

Además, una variable aleatoria es una función que *asigna un número a cada resultado* $s \in \mathcal{S}$ del experimento. Dicha función *mapea* todos los elementos del espacio de muestreo en puntos de la recta numérica. Donde \mathcal{S} está formado por todos los posibles valores de la variable aleatoria X .

$$S = \{x: x = 0,1,2, \dots \dots \dots, n\} \quad (2.6)$$

2.2.1 DEFINICIÓN

De acuerdo a la teoría de probabilidad, si una variable dependiente $u=f(v)$, donde f representa una cierta función de la variable independiente v , donde a cada valor de v corresponde uno de la variable u , por lo tanto $f()$ se obtiene asignando valores de v .

Una variable aleatoria es una variable dependiente como una función de una variable independiente, es decir, corresponde a las salidas que se producen al realizar un experimento. Por lo que es una descripción numérica de dichas salidas.

Mas específicamente, para cada salida w de una experimento aleatorio, se le asigna un único valor $X(w)$. La función $X()$ está definida como una *función aleatoria*. El valor $x=X(w')$ para una salida w' se denomina realización o el valor que la variable aleatoria X toma. A continuación se presenta una definición más general del concepto de variable aleatoria:

Una variable aleatoria X es una función con dominio S (el mismo espacio) y el co-dominio \mathbf{R} (conjunto de los números reales), escrito $X : S \rightarrow \mathbf{R}$. Además se deben cumplir dos propiedades:

- $P\{X = \infty\} = 0, P\{X = -\infty\} = 0$
- El conjunto $\{X \leq \alpha\}$ es un evento para cada número real α .¹

Además, para que cualquier función \mathcal{S} sea considerada como una variable aleatoria, debe cumplir con ciertas condiciones:

1. Dicha función no debe estar multivalorada, es decir, a cada punto en \mathcal{S} deberá corresponder un solo valor de $X(s)$.

¹ Las variables aleatorias se consideran función ya que asocian un número definido a cada punto muestra. Además, éstas permiten definir la probabilidad como una función numérica en lugar de conjuntos, logrando con esto, muestrear a la variable con una expresión específica.

2. El conjunto $\{X \leq x\}$ debe ser un evento para todo número real x . Tal conjunto corresponde a aquellos elementos del espacio \mathcal{S} para los cuales la variable aleatoria $X(s)$ no excede el valor de x .
3. $P\{X \leq x\}$ denota la suma de las probabilidades de los eventos individuales correspondientes a $\{X \leq x\}$.

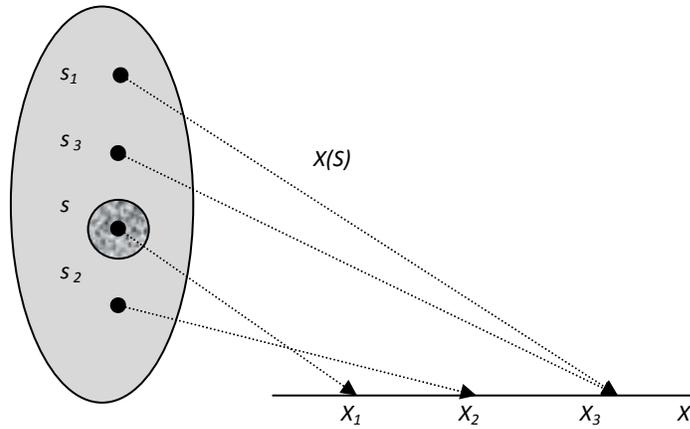


FIG. 2.1 MUESTREO DE UNA VARIABLE ALEATORIA

Existen diferentes tipos de variables aleatorias, dependiendo de su rango. Esta clasificación es la siguiente:

- Variables aleatorias *discretas*: Es aquella en donde se toman solo valores discretos.
- Variables aleatorias *continuas*: Es la que posee un rango continuo de valores.
- Variable aleatoria *mixta*: Este tipo de variable puede combinar un rango discreto con uno continuo.

A continuación se presentan algunos ejemplos de este tipo de variables:

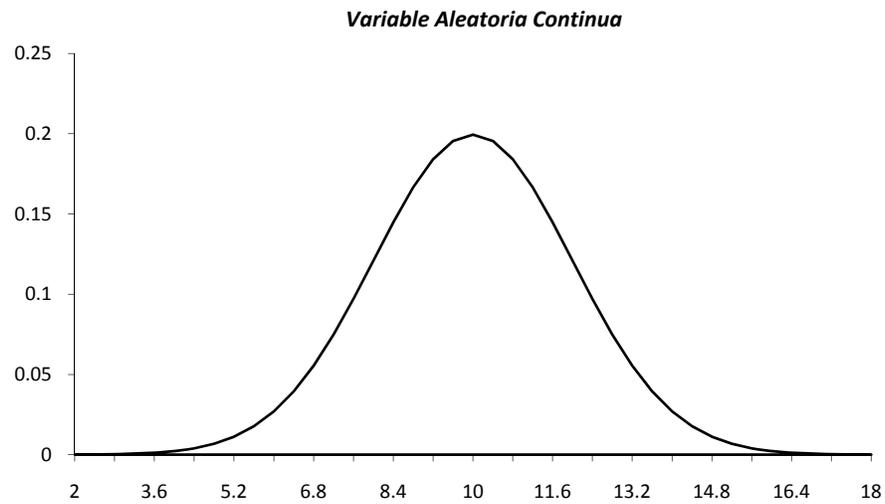


FIG. 2.2 VARIABLE ALEATORIA CONTINUA

En general suelen utilizarse las letras X , Y o Z para denotar a una variable aleatoria, mientras que la forma para denotar los elementos del rango de cada una de estas se realiza con letras minúsculas (x , y , z).

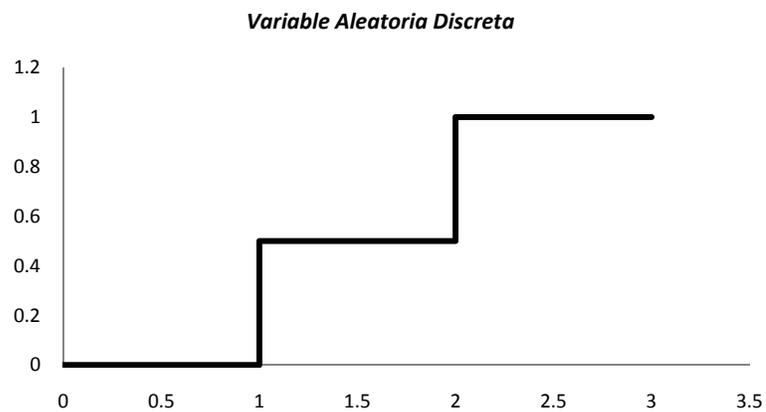


FIG. 2.3 VARIABLE ALEATORIA DISCRETA

2.2.2 FUNCIONES DE DENSIDAD Y DE DISTRIBUCIÓN

FUNCIÓN ACUMULATIVA DE DISTRIBUCIÓN (por sus siglas en inglés CDF)

Si $\{X \leq x\}$ es un evento que corresponde a un conjunto de salida w_i para cada $X(w_i) \leq x$. Entonces la función de distribución de una variable aleatoria se define como:

$$F(x) = F_X(x) = P\{X \leq x\} \quad \text{con } -\infty \leq x \leq \infty \quad (2.7)$$

Por lo tanto $F_X(x)$ es la probabilidad que la variable aleatoria X toma en un valor no mayor a x . Esto se puede ver a continuación:

- Una variable aleatoria discreta puede tener una forma de tipo escalera cuya función de distribución será discreta.
- Una variable aleatoria continua tendrá una función de distribución completamente continua.

El símbolo $\{X \leq x\}$ representa “el valor de la variable aleatoria X es menos o igual a x ”, y x es un número cualquiera.

El dominio de F_X es el rango de la variable aleatoria X . La variable aleatoria es un mapeo del espacio de muestras S del rango de R_X . La función F_X es un mapeo de R_X del rango de F_X . La función F_X es una probabilidad, por lo que su dominio debe ser un conjunto de eventos.

En otras palabras, la función de distribución es una función convencional de valor real. Si todos los valores de esta función pueden ser contados se tiene una función discreta, pero si no, es una continua.

Esta función de distribución es una función que distribuye todos los valores posibles de la variable X . Para describir por completo la variable aleatoria no es suficiente con especificar el posible conjunto de valores que puede tomar, ya que también es necesario describir la probabilidad de cada uno de los valores. Por lo tanto es de suma importancia tener el valor correspondiente de probabilidad que la variable puede tomar en un intervalo arbitrario $(a, b]$.

Una variable aleatoria compleja $z = x + jy$ no tiene función de distribución por que la desigualdad $x + jy \leq x + jy$ no tiene sentido. Por lo que las propiedades estadísticas de una variable Z se definen mediante una función de distribución conjunta.

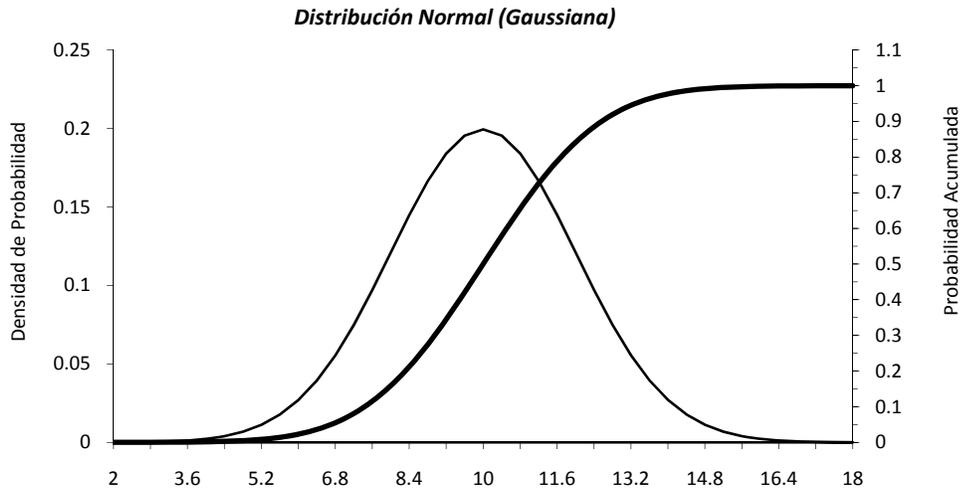


FIG. 2.4 FUNCIÓN ACUMULATIVA DE PROBABILIDAD, CDF

➤ *Propiedades de la Función Acumulativa de Distribución:*

Las siguientes propiedades pueden utilizar para saber si la función de distribución corresponde a una variable aleatoria.

1. $0 \leq F(x) \leq 1, \forall x$, desde $F(x)$ es probabilidad.
2. $F(-\infty) = P\{X \leq -\infty\} = P\{X = -\infty\} = 0$
3. $F(\infty) = P\{X \leq \infty\} = 1$, ya que desde $\{X \leq \infty\}$, se tiene un evento seguro.
4. $F(x)$ es no decreciente y x creciente:

$$F(x_2) \leq F(x_1) \quad \forall x_1 < x_2 \quad (2.8)$$

5. $P\{x_1 < X \leq x_2\} = F(x_2) - F(x_1)$

Un caso especial puede darse cuando:

$$P\{X > x\} = P\{x < X \leq \infty\} = F(\infty) - F(x) = 1 - F(x) \quad (2.9)$$

6. La función de distribución siempre es definida por la derecha:

$$F(x^+) \triangleq \lim_{\varepsilon \rightarrow 0^+} F(x + \varepsilon) = F(x) \quad (2.10)$$

FUNCIÓN DE DENSIDAD DE PROBABILIDAD (por sus siglas en inglés PDF)

El siguiente concepto que se tratará es la función de densidad, en donde el término densidad hace referencia a la suma o integración de dicha función, esto dentro de matemáticas o ingenierías, mientras que en aplicaciones como la mecánica, se integra con la finalidad de encontrar la masa de vigas.

En el procesamiento de señales, ayuda a conocer cual es la potencia de las señales. A continuación se presenta la definición:

La función de densidad para una variable aleatoria X continua esta dada por

$$f(x) = f_X(x) = \frac{dF(x)}{dx} \tag{2.11}$$

Esta función es de densidad por que indica en que valores la variable aleatoria está más o menos concentrada.

Si la variable aleatoria X es discreta, los valores x_i con probabilidades p_i estarán dados por:

$$f(x) = \sum_i p_i \delta(x - x_i) \quad p_i = P\{X = x_i\} \tag{2.12}$$

donde $\delta(x - x_i)$ es un impulso.

En general, la función de densidad es más conveniente que la de distribución para describir a una variable aleatoria, ya que estas indican la concentración de la masa. Sin embargo, a pesar de que cada variable aleatoria cuenta con una función de distribución, no es así con las de densidad, ya que no están bien definidas en toda región debido a la derivada que contiene, lo cual provocará que en ciertos puntos no exista valor alguno, por ejemplo, en esquinas, o curvas no suaves.

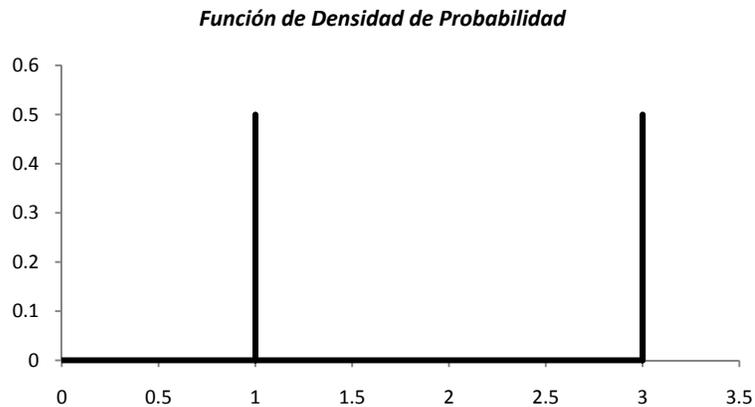


FIG. 2.5 FUNCIÓN DE DENSIDAD DE PROBABILIDAD, PDF

➤ *Propiedades de la Función de Densidad:*

1. No negatividad: $f(x) \geq 0, \forall x$ desde $F(x)$ es no decreciente) $\left(\frac{dF(x)}{dx}\right) \geq 0, \forall x$.
2. Relación con la función de Distribución: Integrando la función de densidad sobre el campo $(-\infty, x]$ de la función de distribución;

$$F(x) = \int_{-\infty}^x f(v)dv = \text{área bajo } f(x) \text{ en el intervalo } (-\infty, x] \quad (2.13)$$

3. Propiedad de normalización:

$$\int_{-\infty}^{\infty} f(x)dx = F(\infty) = 1 \quad (2.14)$$

4. El área bajo $f(x)$ en el intervalo $(x_1, x_2]$ es igual a $P\{x_1 < X \leq x_2\}$:

$$P\{x_1 < X \leq x_2\} = \int_{x_1}^{x_2} f(x)dx \quad (2.15)$$

El tipo de una variable aleatoria puede ser determinado fácilmente mediante la función de distribución y la de densidad:

- La función de densidad de una variable discreta debe ser una suma de funciones delta.
- La función de densidad de una variable continua no puede incluir ninguna función delta.
- La función de densidad de una variable mixta consiste en dos funciones delta y otras funciones.

Para el caso de la función de distribución también hay equivalentes:

- La función de distribución de una variable discreta debe consistir en puntos discontinuos y porciones constantes.
- La función de distribución de una variable continua no puede incluir ninguna discontinuidad.
- La función de distribución de una variable mixta consiste en dos puntos discontinuos y porciones suaves.

Además existe una relación entre las funciones de distribución y de densidad, ya que la función de densidad (PDF) f_X es la derivación de la función de distribución (CDF) F_X , la CDF deberá ser la integral de la PDF. A continuación se presenta esta relación en forma numérica:

$$F_X(x) = \int_{-\infty}^x f_X(\lambda)d\lambda \quad (2.16)$$

2.2.3 FUNCIONES DE DISTRIBUCIÓN DE PROBABILIDAD

En probabilidad, la distribución de probabilidad de una variable aleatoria está representada por una función la cual asigna a cada muestra del evento realizado la probabilidad correspondiente de éxito. Existen diferentes distribuciones de probabilidad tanto para funciones discretas como continuas:

DISTRIBUCIONES PARA VARIABLES ALEATORIAS DISCRETAS

Algunas de las distribuciones para variables aleatorias discretas son las siguientes:

- *Distribución binomial*: Es una distribución de probabilidad discreta que mide el número de éxitos en una secuencia de n ensayos independientes de 'Bernoulli ²' con una probabilidad fija p de ocurrencia del éxito entre los ensayos. En esta distribución, el anterior experimento se repite n veces, de forma independiente, y se trata de calcular la probabilidad de un determinado número de éxitos. La representación matemática de esta distribución es la siguiente:

$$X \sim B(n, p) \quad (2.17)$$

- *Distribución binomial negativa*: Esta distribución incluye la distribución de Pascal. El número de experimentos de Bernoulli de parámetro θ independientes realizados *hasta la consecución del k -ésimo éxito* es una variable aleatoria que tiene una distribución binomial negativa con parámetros k y θ .
- *Distribución Poisson*: Expresa la probabilidad de un número k de eventos ocurriendo en un tiempo fijo si estos eventos ocurren con una frecuencia media conocida y son independientes del tiempo discurrido desde el último evento. Se expresa matemáticamente mediante:

$$f(k; \lambda) = \frac{e^{-\lambda} \lambda^k}{k!} \quad (2.18)$$

- *Distribución geométrica*: Esta distribución puede definirse de dos formas distintas:

² Un experimento de Bernoulli se caracteriza tener únicamente dos posibles resultados. A uno de estos se denomina éxito y tiene una probabilidad de ocurrencia p y al otro, fracaso, con una probabilidad $q = 1 - p$.

1. La distribución de probabilidad del número X de experimentos de Bernoulli necesaria para obtener un éxito, contenido en el conjunto $\{1, 2, 3, \dots\}$ o,
 2. La distribución de probabilidad del número $Y = X - 1$ de fallos antes del primer éxito, contenido en el conjunto $\{0, 1, 2, 3, \dots\}$.
- *Distribución hipergeométrica:* Para definir esta distribución es necesario suponer que se tiene una población de N elementos, de los cuales d pertenecen a A y $N-d$ a B . La distribución hipergeométrica mide la probabilidad de obtener x ($0 \leq x \leq d$) elementos de la categoría A en una muestra de n elementos de la población original.
 - *Distribución de Bernoulli:* Es una distribución de probabilidad discreta, que toma valor 1 para la probabilidad de éxito (p) y valor 0 para la probabilidad de fracaso ($q = 1 - p$).
 - *Distribución uniforme discreta:* En ella, todos los elementos de un conjunto finito son equiprobables, es decir, es una distribución que asume un número finito de valores con la misma probabilidad.

DISTRIBUCIONES PARA VARIABLES ALEATORIAS CONTINUAS

Algunas de las distribuciones para variables aleatorias discretas son las siguientes:

- *Distribución exponencial:* Es una distribución con un parámetro $\lambda > 0$ cuya expresión matemática es la siguiente:

$$f(x) = \begin{cases} \lambda e^{-\lambda x}, & x \geq 0 \\ 0, & \text{de otro modo} \end{cases} \quad (2.19)$$

- *Distribución normal:* O también conocida como distribución Gaussiana. La gráfica de su función de densidad tiene una forma acampanada y es simétrica respecto de un determinado parámetro. Esta curva se conoce como campana de Gauss.

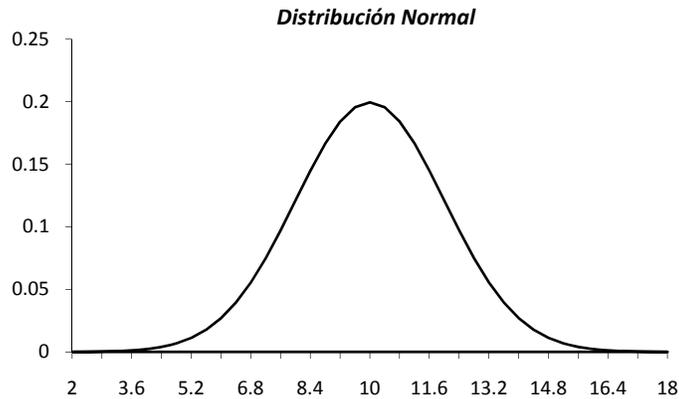


FIG. 2.6 DISTRIBUCIÓN GAUSSIANA (NORMAL)

Se llama distribución normal "estándar" a aquella en la que sus parámetros toman los valores $\mu=0$ y $\sigma=1$.

- *Distribución Gamma:* La distribución gamma es una distribución con dos parámetros k y λ cuya función de densidad para valores $x > 0$ es:

$$f(x) = \lambda^k e^{-\lambda x} \frac{x^{k-1}}{\Gamma(k)} \quad (2.20)$$

- *Distribución Beta:* La distribución beta es una distribución con dos parámetros a y b cuya función de densidad para valores $0 < x < 1$ es:

$$f(x) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} x^{a-1} (1-x)^{b-1} \quad (2.21)$$

2.3 LEY DE LOS GRANDES NÚMEROS

La ley de los grandes números es un teorema en probabilidades que describe el comportamiento del promedio de una sucesión de variables aleatorias según el número total de variables aumenta. El teorema describe hipótesis suficientes para afirmar que dicho promedio converge al promedio de las esperanzas de las variables aleatorias involucradas.

En particular, si todas las variables son idénticamente distribuidas e independientes, el promedio tiende al valor de la esperanza individual.

Las leyes de los grandes números implican que el promedio de una muestra al azar de una población de gran tamaño tenderá a estar cerca de la media de la población completa. Varias formulaciones de la ley de los grandes números (y sus condiciones asociadas) especifican la convergencia de formas distintas.

Cuando las variables aleatorias tienen una varianza finita, el teorema del límite central extiende nuestro entendimiento de la convergencia de su promedio describiendo la distribución de diferencias estandarizadas entre la suma de variables aleatorias y el valor esperado de esta suma. Sin importar la distribución subyacente de las variables aleatorias, esta diferencia estandarizada converge a una variable aleatoria normal estándar.

La frase '*ley de los grandes números*' es también usada ocasionalmente para referirse al principio de que la probabilidad de cualquier evento posible (incluso uno improbable) ocurra *al menos una vez* en una serie incrementa con el número de eventos en la serie. Por ejemplo, la probabilidad de que un individuo gane la lotería es bastante baja; sin embargo, la probabilidad de que *alguien* gane la lotería es bastante alta, suponiendo que suficientes personas comprasen boletos de lotería.

Ley débil de los grandes números:

La *ley débil de los grandes números* establece si se tiene una secuencia infinita de variables aleatorias X_i , donde todas las variables aleatorias tienen el mismo valor esperado μ y varianza σ^2 , además de que son independientes unas con otras. Las probabilidades de éxito o fallo son:

$$X_i = \begin{cases} 1 & \text{éxito} \\ 0 & \text{error} \end{cases} \quad (2.22)$$

entonces el promedio de una muestra:

$$\bar{X}_n = (X_1 + \dots + X_n)/n \quad (2.23)$$

converge en probabilidad a μ . En otras palabras: para cualquier número positivo ε , sin importar cuan pequeño, se tiene

$$\lim_{n \rightarrow \infty} P(|\bar{X}_n - \mu| < \varepsilon) = 1 \quad \text{para } \varepsilon > 0 \quad (2.24)$$

La *ley débil de los grandes números* es una afirmación acerca de los probables valores de una suma de N variables aleatorias distribuidas de la misma forma. Específicamente, si esta suma está dividida entre N .

Esta ley ayudar a tener una justificación si un experimento está representado por grandes números de tiempos, la frecuencia relativa de una salida en particular debería ser cercana a la

probabilidad de esa salida. La ley débil se utiliza para establecer un valor probable de una gran secuencia de variables aleatorias.

Ley fuerte de los grandes números:

La *ley fuerte de los grandes números* es contraria a la descrita anteriormente, establece que si un conjunto de variables aleatorias X_1, X_2, X_3, \dots es una secuencia infinita de variables aleatorias que son independientes e idénticamente distribuidas con $E(|X_i|) < \infty$ (y donde el valor esperado es μ), entonces

$$P(\lim_{n \rightarrow \infty} \bar{X}_n = \mu) = 1 \quad (2.25)$$

es decir, el promedio de las variables aleatorias converge a μ casi seguramente (en un conjunto de probabilidad 1). Esta ley es una simplificación y fuerte afirmación de la ley débil.

2.4 PROCESOS ESTOCÁSTICOS

Como se mencionó al inicio del capítulo, cuando una señal aleatoria es una función dependiente del tiempo, se dice que se trata de un proceso *aleatorio* o un proceso *estocástico*. A continuación se presenta una definición más específica de proceso aleatorio o estocástico:

Un proceso estocástico $X(t, \tau)$ es una función de dos variables, t y τ , donde τ es un elemento del espacio de muestras. Además, para cualquier tiempo fijo t , la función $X(t, \tau)$ debe satisfacer la definición de variable aleatoria que se describió anteriormente³.

³ Un proceso estocástico es un concepto matemático que sirve para caracterizar una sucesión de variables aleatorias que evolucionan en función de otra variable, generalmente, el tiempo. Cada una de las variables aleatorias del proceso tiene su propia función de distribución de probabilidad y, entre ellas, pueden estar correlacionadas o no.

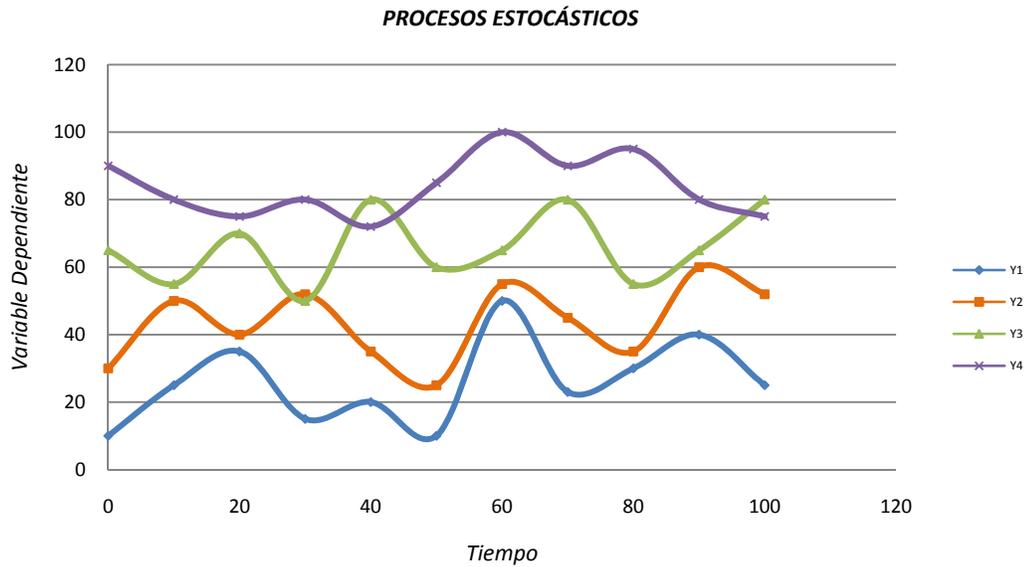


FIG. 2.7 PROCESOS ESTOCÁSTICOS

Ahora se tiene una función que depende de dos variables, en donde t es arbitraria, y generalmente se relaciona con el tiempo, sin embargo, se puede hablar de tiempo continuo o tiempo discreto. La diferencia ahora radica en que estas dos variables van a determinar la forma de onda de la salida. La palabra '*estocástico*' se deriva del griego '*stochastikos*' que hace referencia a la variabilidad.

Existen cuatro posibles interpretaciones de $X(t, \tau)$ dependiendo de la naturaleza de las variables:

1. Si t y τ son fijas, entonces $X(t, \tau)$ es un número.
2. Si t es fijo y τ es variable, entonces $X(t, \tau)$ es una variable aleatoria.
3. Si t es variable y τ es fija, entonces $X(t, \tau)$ es una función del tiempo (llamada función muestra).
4. Si ambas t y τ son variables, $X(t, \tau)$ entonces es un proceso estocástico.

Un proceso $X(t)$ estocástico puede ser definido en general por:

- Una variable aleatoria variando en el tiempo en el sentido de que el valor que toma la variable aleatoria es variable con el tiempo, o

- Una función variable en el tiempo en el sentido de que la función entera, además de un número, es asignada a cada salida.

Sin embargo, se debe denotar que:

- $X(t)$ en un tiempo específico t_1 es una variable aleatoria $X(t_1)$,
- Una realización de $X(t)$ es una función del tiempo, llamada capa muestra (*simple path*), función muestra (*function path*), o función miembro (*member function*), pero no número.
- Un espacio muestra es un conjunto de funciones del tiempo,
- $X(t)$ en diferentes instantes puede tener una infinita cantidad de distribuciones diferentes.

Existe una gran cantidad de procesos aleatorios, cada uno con diferentes características. Se dice que un proceso aleatorio puede ser:

- Continuo si $X(t_1)$ para muchos tiempos fijos t_1 es una variable aleatoria continua.
- Discreto si $X(t_1)$ para muchos t_1 es una variable aleatoria discreta.
- Mixto si no es necesariamente continuo o discreto.
- Estrictamente estacionario (este concepto de abarcará de una forma más detallada al final del presente capítulo).
- Estacionario si tampoco su auto correlación depende de la elección del tiempo de inicio, esto es, si:

$$E[X(t)] \text{ no depende de } t, y$$

$$R_x(t_1, t_2) = R_x(t_1 - t_2) = R_x(\tau), \quad \tau = t_1 - t_2 \quad (2.26)$$

para cada t_1, t_2 , esto es, que la auto correlación depende solo del la diferencia de los tiempos.

- No estacionario si no es estacionario.
- Blanco si los valores en diferentes tiempos no están correlacionados, esto es, si la auto varianza es siempre cero (o igualmente, su auto correlación es el producto de los valores esperados) para diferentes instantes de tiempo:

$$C_x(t_1, t_2) = 0 \quad [o \quad R_x(t_1, t_2) = \bar{x}(t_1)\bar{x}(t_2)] \quad \forall t_1 \neq t_2 \quad (2.27)$$

- Ergódico si el valor promedio conjunto \bar{x} es igual al tiempo promedio para cada una de las muestras de la función $x(t)$, definida por

$$\bar{x}(\omega) = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T x(t) dt, \quad \text{dada } \omega \quad (2.28)$$

El concepto de estacionaridad en los procesos aleatorios es similar al estado fijo de los procesos determinísticos; las características de los procesos son tiempo invariante incluso pensando que el proceso en si mismo es variable en el tiempo.

Un proceso estacionario $X(t)$ es blanco si su auto covarianza es siempre cero para cualquier valor no nulo de τ :

$$C_x(\tau) \triangleq E\{[X(t + \tau) - \bar{x}][X(t) - \bar{x}]\} = 0 \quad \forall \tau \neq 0 \quad (2.29)$$

Su auto correlación claramente siempre es cero para cualquier valor positivo de τ :

$$R_x(\tau) \triangleq E[X(t + \tau)X(t)] = 0 \quad \forall \tau \neq 0 \quad (2.30)$$

2.4.1 CONCEPTOS

Antes de comenzar a describir las principales características de los procesos aleatorios, es importante conocer algunos conceptos que serán mencionados a lo largo de los siguientes tópicos.

Momentos:

Para un proceso estocástico, los momentos que se definen para una sola variable aleatoria son muy similares, la diferencia radica en que ahora son funciones dependientes del tiempo. Por lo que el k-ésimo momento de $X(t, \tau)$ esta dado por:

$$m_k(t) = E[X^k(t, \tau)] = \int_{-\infty}^{\infty} \alpha^k f_X(\alpha; t) d\alpha \quad (2.31)$$

así mismo, el k-ésimo momento central está dado por:

$$\mu_k = E\{[X^k(t, \tau) - m_k(t)]^k\} = \int_{-\infty}^{\infty} [\alpha - m_k(t)]^k f_X(\alpha; t) d\alpha \quad (2.32)$$

La principal diferencia entre un proceso estocástico y las variables aleatorias es que ahora se tiene una forma de onda completamente diferente. La salida experimental τ determina una completa forma de onda $X(t, \tau)$, no solo un número.

Media:

La media de un proceso estocástico X en el tiempo t , se define como la media de la variable aleatoria $X(t)$ y se denota como:

$$E\{X(t)\} \triangleq m_x(t) \quad (2.33)$$

Autocorrelación:

La autocorrelación probabilística de un proceso aleatorio X , en dos tiempos t_1 y t_2 se refiere a la correlación de dos señales aleatorias $X(t_1)$ y $X(t_2)$ y se denota por:

$$E\{X(t_1)X(t_2)\} \triangleq R_{X(t_1)X(t_2)} \quad (2.34)$$

Autocovarianza:

Análogamente a la autocorrelación, la autocovarianza de un proceso aleatorio X en dos tiempos t_1 y t_2 está dado por:

$$E\{[X(t_1) - m_x(t_1)][X(t_2) - m_x(t_2)]\} \triangleq K_{X(t_1,t_2)} \quad (2.35)$$

Funciones de Distribución y de Densidad para procesos estocásticos:

Para los procesos estocásticos, las funciones de distribución y de densidad están definidas de una forma distinta a las de las variables aleatorias

$$F_{XX}(\alpha_1, \alpha_2; t_1, t_2) = P\{X(t_1) \leq \alpha_1, X(t_2) \leq \alpha_2\} \quad (2.36)$$

$$f_{XX}(\alpha_1, \alpha_2; t_1, t_2) = \frac{\partial^2}{\partial \alpha_1 \partial \alpha_2} F_{XX}(\alpha_1, \alpha_2; t_1, t_2) \quad (2.37)$$

Estas fórmulas conectan los valores de dos procesos estocásticos en dos tiempos distintos. Cuando se trate de tiempo discretos, se maneja la nomenclatura de n_1 y n_2 en lugar de t_i .

Funciones de correlación:

Las funciones de correlación ayudan a saber y cuantificar la similitud entre dos diferentes señales, además de que permite estimar el valor de una forma de onda en cualquier tiempo basándose en un tiempo dado.

Dentro de los procesos aleatorios, existen dos clases de correlación:

1. La función de correlación cruzada (*crosscorrelation function CCF*) de dos procesos aleatorios $X(t)$ y $Y(t)$ está definida como la correlación promedio de dos procesos:

$$R_{xy}(t + \tau, t) \triangleq E[X(t + \tau)Y(t)] = \text{la media conjunta de } X(t + \tau)Y(t) \quad (2.38)$$

2. La función de auto correlación (*autocorrelation function ACF*) de un proceso aleatorio $X(t)$ está definido por la autor relación promedio del proceso:

$$R_x(t + \tau, t) \triangleq E[X(t + \tau)X(t)] = \text{la media conjunta de } X(t + \tau)X(t) \quad (2.39)$$

Estas funciones de correlación no son variables, son un conjunto equivalente de las definidas para las que se aplican a los procesos determinísticos.

2.4.2 ESTACIONARIDAD Y ERGODICIDAD

PROCESO ESTACIONARIO

Un proceso estocástico es llamado estrictamente estacionario si sus propiedades estadísticas no cambian con el tiempo de origen, lo cual implica que la media, la varianza y la PDF de primer orden para $X(t_1, \tau)$ son las mismas para $X(t_2, \tau)$ para todo t_1 y t_2 . Esto quiere decir que la estadística determinada para $X(t, \tau)$ es igual para $X(t + \varepsilon, \tau)$ para todo valor de ε , pero solo corresponde al primer orden de estacionaridad, por lo que es necesario mencionar otras definiciones.

Un proceso estocástico es estacionario de orden k si

$$f_X(\alpha_1, \dots, \alpha_k; t_1, \dots, t_k) = f_X(\alpha_1, \dots, \alpha_k; t_1 + \varepsilon, \dots, t_k + \varepsilon) \quad \text{para todo } \varepsilon \quad (2.40)$$

De esta igualdad es posible darse cuenta que la función correspondiente al orden uno debe ser independiente de t y se define por:

$$f_X(\alpha; t) = f_X(\alpha) \quad (2.41)$$

De forma similar $f_X(\alpha_1, \alpha_2; t_1, t_2) = f_X(\alpha_1, \alpha_2; t_1 + \varepsilon, t_2 + \varepsilon)$ es independiente de ε para todo ε . Por eso se puede llegar a la conclusión de que:

$$f_X(\alpha_1, \alpha_2; t_1, t_2) = f_X(\alpha_1, \alpha_2; \tau) \quad \tau = t_1 - t_2 \quad (2.42)$$

PROCESO ERGÓDICO

La idea de ergodicidad surge si se tiene una sola función muestra de un proceso estocástico en lugar del conjunto entero. Esta función muestra nos puede dar poca información de la estadística del proceso. Pero si el proceso es ergódico, toda la información estadística puede ser derivada de esta sola función muestra.

Por lo tanto, ‘un proceso ergódico es aquel donde los promedios estadísticos son iguales a los temporales’⁴.

Cuando un proceso es ergódico, una sola función muestra representa el proceso entero. Por lo tanto, la ergodicidad implica al mismo tiempo estacionaridad, y así como hay grados de estacionaridad, también los hay de ergodicidad. A continuación solo se presentan los niveles de media y correlación.

Nivel 1: *Un proceso es ergódico si la media si*

$$\langle x(t) \rangle = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T x(t) dt = E[X(t, \tau)] \quad (2.43)$$

donde $m_X = E[X(t, \alpha)]$.

La ergodicidad de la media implica estacionaridad de la misma. Pero la estacionaridad de la media no implica su ergodicidad.

Nivel 2: *Un proceso es ergódico en autocorrelación si*

$$r(\tau) = \langle x(t)x(t - \tau) \rangle = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T x(t)x(t - \tau) dt = E[X(t)X(t - \tau)] = R_{XX}(\tau) \quad (2.44)$$

La variable r representa la función de tiempo de autocorrelación mientras que R indica la función estadística de autocorrelación.

⁴ *Un proceso aleatorio se considera ergódico si el promedio calculado sobre una función muestras es igual al promedio calculado sobre el ensamble.*

Capítulo 3

SISTEMAS DE COMUNICACIONES

Un *sistema de comunicaciones* es un conjunto de dispositivos que son utilizados con la finalidad de transmitir, emitir y recibir señales de todo tipo, como voz, datos, audio, video, etc., además dichas señales pueden ser del tipo digital o analógica.

Un sistema de comunicaciones puede describirse fácilmente mediante tres elementos básicos; un transmisor, el cual se encarga de generar la señal que se desea y acoplarla de tal forma que pueda viajar a través del canal, mediante procedimientos como modulación, filtrado, codificación, etc.; un medio de transmisión, el cual será el canal mediante el cual la señal va a viajar, y puede ser desde fibras ópticas, cables coaxiales, hasta el mismo aire; y finalmente un receptor, que realiza el procedimiento inverso del transmisor con la finalidad de reconstruir la señal y que esta sea lo mas parecida a la original.

En el canal se presentan fenómenos que deterioran la forma de la señal generando errores en la recepción.

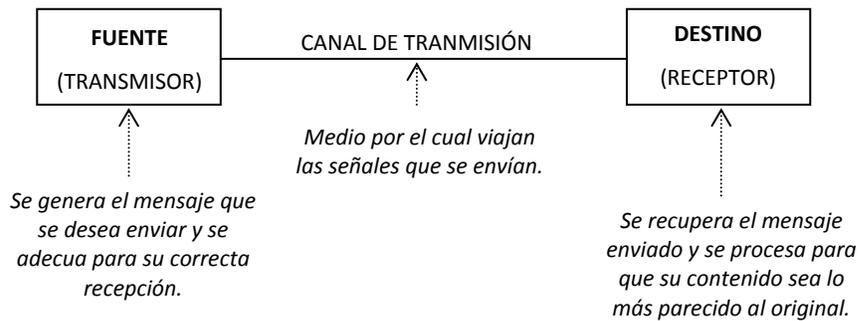


FIG. 3.1 SISTEMA DE COMUNICACIONES

Debido a que la presente tesis está enfocada a uno de los procedimientos que se llevan a cabo en el transmisor, el paso de una señal analógica a digital mediante cuantización y codificación, a continuación se tratarán la teoría básica de codificación.

3.1 MEDIDAS DE INFORMACIÓN

Primeramente nos concierne saber ¿qué es información? De forma general se puede definir a la información de la siguiente manera:

‘La información se define como el conjunto organizado de datos procesados, el cual crea un mensaje sobre un determinado fenómeno. Los datos se perciben, se integran y generan la información necesaria para producir el conocimiento que es el que finalmente permite tomar. La información también indica mediante códigos y conjuntos de datos, los modelos del pensamiento humano. La información por lo tanto, procesa y genera el conocimiento humano’¹.

Por lo tanto, deben existir métodos que permitan medir la ‘cantidad de información’ a lo largo de una transmisión. Para realizar estos procedimientos se utilizan los conceptos de entropía e información mutua. Empezaremos por describir el concepto de entropía.

3.1.1 ENTROPÍA

El concepto de entropía fue utilizado al inicio como una medida de información en un canal de transmisión común. El concepto de entropía o también llamado ‘variabilidad’ se da únicamente dentro de experimentos, por que siempre que se haga referencia a entropía se hace referencia al mismo tiempo a un experimento.

Para definirla, tomemos en cuenta que se tiene un experimento E y tres campos (S, F, P) , donde S representa el espacio muestra, F representa el campo de eventos y P representa la asignación de probabilidades de F . Estos parámetros ayudarán a medir la entropía.

¹ La importancia del concepto ‘información’ comenzó a tornarse importante en probabilidad cuando se desarrolló la rama matemática conocida como Teoría de la Información, la cual fue iniciada Claude E. Shannon a través de un artículo publicado en el Bell System Technical Journal en 1948, titulado ‘Una teoría matemática de la comunicación’.

De una forma general (sobre todo para ciencias como la Termodinámica), la entropía puede definirse como:

‘Entropía: Medida de la distribución aleatoria de un sistema’².

La ecuación que define a la entropía es la siguiente:

$$H(E) = \sum_i P_i \log\left(\frac{1}{P_i}\right) \quad (3.1)$$

En esta ecuación se utiliza la letra **H** para representar a la entropía. Una de las ventajas de utilizar los logaritmos en la definición radica en que al trabajar en experimentos combinados se tendrá que la entropía total es igual a la suma de las entropías individuales.

Además, generalmente se utiliza la base logarítmica 10, la base *e* y base 2. Para cada una de estas, la unidad de medida tiene nombres diferentes:

Base 10 – *hartley*

Base *e* – *nat* (de unidades naturales)

Base 2 – *bit* (de unidades binarias)

Además se puede pasar cómodamente de una unidad a otra mediante:

$$\log_2(x) = \frac{\log_e(x)}{\log_e(2)} = \frac{\log_{10}(x)}{\log_{10}(2)} \quad (3.2)$$

➤ *Propiedades de la entropía*

A continuación se presentan las principales propiedades de la entropía:

1. $H(E)$ es una función continua de p_i .
2. $H(E)$ es máxima para resultados igualmente probables. Si se tienen N elementos en el espacio muestra con $p_1 = \dots = p_N = 1/N$, ningún otro conjunto de probabilidades proporciona una gran entropía.
3. Si $p_1 = p_2 = \dots = p_N$, entonces $H(E)$ es una función creciente de N .
4. $H(E_1, E_2) \leq H(E_1) + H(E_2)$ con igualdad si y solo si E_1 y E_2 son independientes.

² La palabra entropía procede del griego y significa evolución o transformación y dicho concepto se relaciona ampliamente con el nivel de incertidumbre que existe en cualquier experimento o señal aleatoria. También puede ser considerado como la cantidad de ruido o desorden en el sistema.

La *propiedad 1* quiere decir que un pequeño cambio en la probabilidad de alguna de las salidas no provocará un gran cambio en la entropía del experimento. Si se comidera un experimento E con dos posible salidas con probabilidades p y $1-p$, la entropía esta dada por:

$$H(E) = -p \log p - (1 - p) \log (1 - p) \quad (3.3)$$

A continuación se muestra la gráfica que representa este experimento, en donde se puede observar las propiedades uno y dos, ya que se ve que dicha curva es continua en p , y el máximo ocurre cuando las dos salidas son igualmente probables, es decir, cuando $p=1-p$.

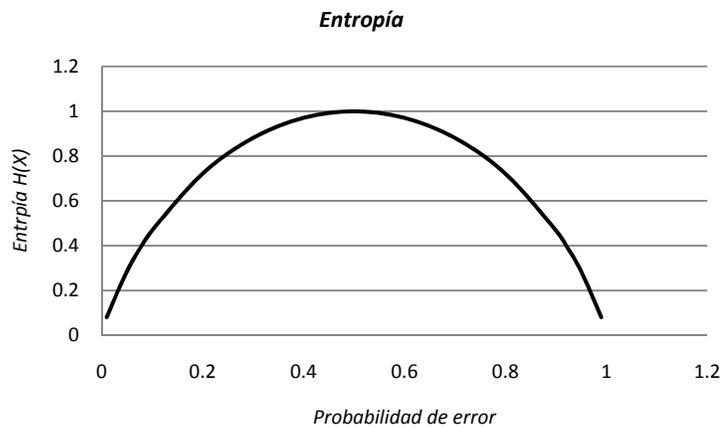


FIG. 3.2 GRÁFICA CORRESPONDIENTE A LA ENTROPÍA H(X)

En cuanto a la *propiedad 3*, debido a la definición de entropía, en donde se describió como una medida de variabilidad, y un incremento en las salidas igualmente probables hace que la entropía aumente.

Estas propiedades son muy importantes ya que pueden ser útiles para un gran número de aplicaciones. Uno muy importante es el método de la entropía máxima (MEM). Este método maximiza la entropía en un experimento sujeto a cualquier limitación. La única limitación es que la suma de las probabilidades es 1.

El problema de este método radica que escoger las probabilidades desconocidas para poder maximizar la entropía y poder satisfacer las limitaciones. Por otro lado se encuentra el inconveniente de encontrar las limitaciones que guían a obtener las probabilidades.

3.1.1 INFORMACIÓN MUTUA

La información mutua de dos experimentos E_i es una cantidad que mide la dependencia mutua de estos dos experimentos.³

Sin embargo, para poder definir este concepto de forma numérica primero se debe hacer referencia al concepto de ‘entropía restante’ la cual se conoce después de observar otro evento, es decir, la entropía restante de E_1 después de ver E_2 . Esta debe relacionarse con la probabilidad condicional $P(U_i|M_j)$ de la misma forma que la entropía inicial está relacionada con $P(U_i)$. Además tenemos que $H(E_1)$ es igual al $\log[1/P(U_i)]$. Por lo que la media del $\log[1/P(U_i|M_j)]$. Por lo que la entropía restante es la siguiente:

$$H(E_1|E_2) = - \sum_{i,j} P(U_i, M_j) \log P(U_i|M_j) \quad (3.4)$$

Esta es la entropía restante de E_1 después de observar a E_2 , y esto es llamado entropía condicional de E_1 dado E_2 . El cambio en la entropía es la ganancia de la información y es escribe como $I(E_1, E_2)$. Esta escritura tiene el siguiente significado:

$$I(E_1, E_2) = H(E_1) - H(E_1|E_2) \quad (3.5)$$

Cabe mencionar que se cumple la igualdad: $I(E_1, E_2) = I(E_2, E_1)$, o

$$I(E_1, E_2) = H(E_2) - H(E_2|E_1) \quad (3.6)$$

Este término se conoce como ‘información mutua’ por que es simétrica. El concepto de información mutua y la entropía restante son útiles al momento de diseñar sistemas.

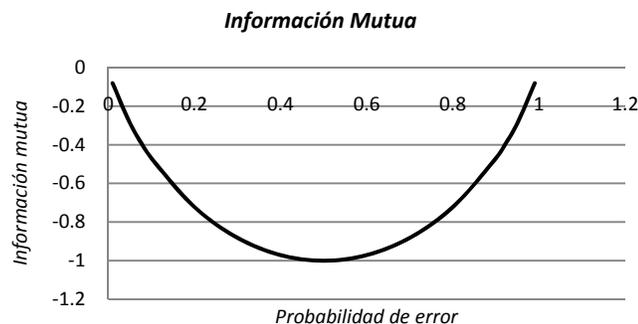


FIG. 3.3 INFORMACIÓN MUTUA $I(E_1;E_2)$

³ La información mutua es una medida de gran importancia, dentro de la teoría de la información. Define la información aportada por una variable aleatoria sobre otra.

3.2 CONSIDERACIONES Y ESQUEMAS DE ANÁLISIS

Como se mencionó al inicio del capítulo, un sistema de comunicaciones esta formado por una fuente, que genera los datos que se desean transmitir; un destino que se encarga de recibir los datos de la fuente y reconstruirlos para que sean lo más fieles a los originales; y un canal, que es el medio por donde viajan los datos.

A continuación se presenta un diagrama de bloques explicando la ubicación de cada uno de estos componentes:

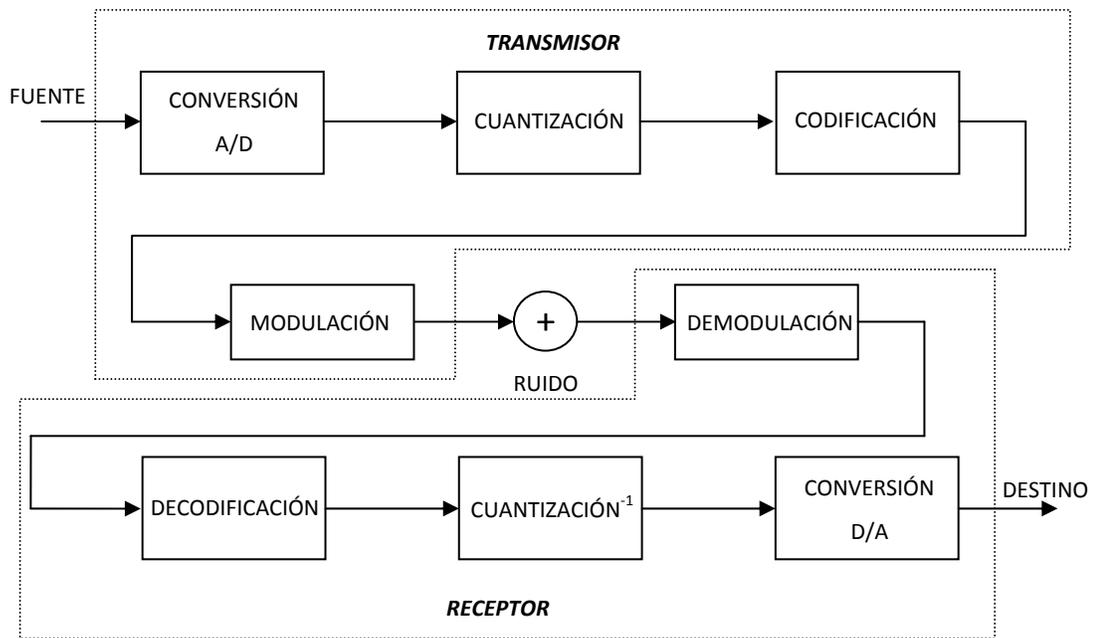


FIG. 3.4 DIAGRAMA DE BLOQUES DE UN SISTEMA DE COMUNICACIONES

Sin embargo, al momento de realizar la transmisión entre la fuente y el destino, se pueden presentar varios problemas en el canal como son:

- *Atenuación:* Es la reducción de nivel o potencia de una señal cuando pasa a través del canal. Esta se mide en decibeles, y por lo general depende de la frecuencia a que se este trabajando en el sistema.
- *Ruido:* Se considera ruido a todas las perturbaciones eléctricas que interfieren sobre las señales transmitidas o procesadas.

- *Distorsión:* Es la diferencia entre señal que entra a un equipo o sistema y la señal de salida del mismo. Por tanto, puede definirse como la "deformación" que sufre una señal tras su paso por un sistema.
- *Interferencia:* La interferencia es cualquier proceso que altera, modifica o destruye una señal durante su trayecto en el canal existente entre el emisor y el receptor.
- *Diafonía:* Es una forma de interferencia causada por las señales en las cercanías de conductores.

Además, existe gran variedad de canales, cada uno con diferentes características las cuales se explicarán a continuación.

Antes de comenzar a hablar de canales se deben de describir los diferentes tipos de canales que se pueden estudiar. La definición de canal nos dice que *un canal es el medio mediante el cual las señales viajan entre el transmisor y el receptor. Es decir, es el medio mediante el cual la información viaja desde un punto A hasta un punto B*⁴.

La velocidad de transferencia de un canal R_S se denota por:

$$R_S = \frac{k}{n} \frac{\text{información del bit}}{\text{símbolos del canal}} \quad (3.7)$$

El mensaje a ser transmitido es un proceso estacionario X y el proceso Y es la salida. La salida Y depende únicamente de la entrada X pero también de la naturaleza del canal. Para definir la naturaleza del canal se puede hacer la siguiente clasificación:

1. El canal es *sin memoria*, si el valor de Y depende únicamente del valor presente de X .
2. La entrada X presenta ruido blanco que se expande en todas las frecuencias.
3. El canal es *binario (BSC)*, si las entradas y salidas X, Y solo pueden tomar el valor de 0 y 1.

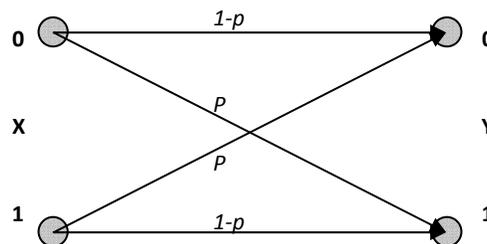


FIG. 3.5 CANAL BINARIO SIMÉTRICO

⁴ Un canal de comunicaciones es el medio físico utilizado para transportar información entre dos extremos. Este medio físico puede ser cableado, inalámbrico (es decir, por medio del espectro radioeléctrico), o una combinación de ambos.

En la transmisión de un vector de longitud n la probabilidad condicional de la salida Y conociendo X es:

$$p(Y|X) = p^d(1 - p)^{n-d} \quad (3.8)$$

donde $d = d_H(X, Y)$ es la distancia de *Hamming* entre el canal de entrada y salida.

Canal sin ruido:

El canal es sin ruido si existe una correspondencia *uno-a-uno* entre las entradas X_n y salidas Y_n , para un canal binario esto significa que $x_n=0$, entonces $y_n=0$.

Canal con ruido:

Un canal con ruido es un sistema variable que establece una relación estática entre las entradas X_n y las salidas Y_n .

Para un canal sin memoria, esta relación se da por medio de matrices cuyos elementos π_{ij} son las probabilidades condicionales entre los estados de entradas y salidas. Por ejemplo, para un canal binario:

$$\Pi = \begin{bmatrix} \pi_{00} & \pi_{01} \\ \pi_{10} & \pi_{11} \end{bmatrix} \quad \text{donde} \quad \pi_{00} + \pi_{01} = 1 \text{ y } \pi_{10} + \pi_{11} = 1 \quad (3.9)$$

Se dice que el canal es simétrico si $\pi_{10} = \pi_{01} = \beta$, y $\pi_{00} = \pi_{11} = 1 - \beta$ por lo que en un canal simétrico:

$$\Pi = \begin{bmatrix} 1 - \beta & \beta \\ \beta & 1 - \beta \end{bmatrix} \quad (3.10)$$

3.3 CAPACIDAD DEL CANAL

La capacidad del canal se mide en unidades de bit por símbolo, y se define como la información mutua máxima $I(X, Y)$ obtenida al variar la probabilidad de las entradas.

Esta información debe ser lo más fiable posible, es decir, que su probabilidad de error sea la mínima para su correcta recuperación. Existen dos definiciones más particulares, cada una de las cuales está enfocada a canales discretos y continuos. A continuación se presentan ambas definiciones:

Capacidad del Canal en canales discretos: Un canal es discreto *sin memoria* si cumple con las siguientes especificaciones:

- El conjunto $X = \{x_1, x_2, \dots, x_n\}$ en la entrada puede ser transmitido por el canal,
- El conjunto $Y = \{y_1, y_2, \dots, y_m\}$ pueden ser recibidos en la salida del canal,
- Las probabilidades de ambos conjuntos están dados por las probabilidades condiciones del canal (p_{ij}) donde $p_{ij} = p(y_i|x_j)$.

Se define la entropía de entrada, entropía de salida y entropía de entrada condicionada por la salida como,

$$H(X) = \sum_{i=1}^n p(x_i) \log \frac{1}{p(x_i)}; \quad H(Y) = \sum_{j=1}^m p(y_j) \log \frac{1}{p(y_j)} \quad (3.11)$$

$$(3.12)$$

$$H(X|Y) = \sum_{j=1}^m p(y_j) \sum_{i=1}^n p(x_i, y_j) \log \frac{1}{p(x_i|y_j)} \quad (3.13)$$

Por lo que la información mutua estaría dada por:

$$I(E_1, E_2) = H(E_2) - H(E_2|E_1) \quad (3.14)$$

y la capacidad del canal es:

$$C = \max_{p(x_i)} I(X, Y) \quad (3.15)$$

Capacidad del Canal en canales continuos: La capacidad de un canal continuo esta sumamente relacionado al teorema del muestreo, por lo que está definido de la siguiente forma:

$$C = B \log(1 + SNR) \quad (3.16)$$

donde B representa el ancho de banda, y SNR es la relación señal a ruido.

Este concepto es aplicable a todos los tipos de canales que se mencionaron con anterioridad, por ejemplo, la *capacidad del canal* para un *canal binario simétrico* esta dada por:

$$C = 1 - H_2(p) \quad (3.17)$$

donde H_2 es la función de entropía binaria.

Teorema

La información puede ser enviada a través de una canal con ruido a una tasa cercana a la capacidad del canal C con una pequeña probabilidad de error.

$$H(X) = I(X, Y) + H(X|Y) \geq I(X, Y) \tag{3.18}$$

*Lo cual demuestra que es posible encontrar una fuente con entropía muy cercana a C , por lo tanto:*⁵

$$H(X) < C \tag{3.19}$$

3.4 CODIFICACIÓN DE FUENTE

Supongamos que se tiene una fuente que genera un mensaje que debe ser transmitido de forma digital, sin embargo, el mensaje es analógico. Para poder realizar una transmisión de forma exitosa primero se debe llevar a cabo el proceso de muestreo, cuantización y codificación.

A continuación nos enfocaremos en el proceso de codificación abarcando diferentes clases de códigos que ayudan a obtener la menos cantidad posible de errores en el receptor. Para comenzar, ¿qué es un código?, ¿qué es la codificación?

*Un código es la forma de encapsular cierta información para que pueda ser transmitido con cierta seguridad. La codificación es el método que permite convertir un carácter de un lenguaje natural en un símbolo de otro sistema de representación, como un número o una secuencia de pulsos eléctricos en un sistema electrónico, aplicando normas o reglas de codificación*⁶.

⁵ Este teorema permite conocer el valor de la información mutua aún si solo se conocen los valores de entropía de las variables por separado o en su forma conjunta.

⁶ Existen una gran cantidad de algoritmos para encontrar el código adecuado en cada caso, sin embargo por lo general se busca que este sea óptimo.

Existen diferentes tipos de códigos dependiendo el uso y aplicaciones que se requiera, además de que para su elección deben tomarse en cuenta consideraciones como la longitud de los códigos, su funcionalidad y seguridad.

Generalmente se buscan códigos óptimos para minimizar la probabilidad de error. Pero, ¿qué es un código óptimo?

Un código óptimo es aquel código que permite codificar todas las palabras del mensaje con la menor longitud media, es decir, este tipo de códigos permite utilizar la menos cantidad de palabras de código y así disminuir la probabilidad de error en el receptor⁷.

Para esto se hace uso de la distancia mínima de Hamming entre cualesquiera dos palabras de código en el canal y se denota mediante d_{min} . El número de errores de bits que el canal es capaz de corregir t es expresado mediante la siguiente expresión:

$$d_{min} \geq 2t + 1 \quad (3.20)$$

3.4.1 CONCEPTOS DE SHANNON

A continuación se presenta uno de los códigos más conocidos. En el código de Shannon, las probabilidades p_i son números decimales, entonces la longitud del código $l_i = -\log p_i$ es óptima.

A partir de esto es posible crear códigos para diversa clase de casos.

Cada p_i especifica un número entero n_i

$$\frac{1}{2^{n_i}} \leq p_i < \frac{1}{2^{n_i - 1}} \quad (3.21)$$

donde $p_i > 1/2^{n_i}$ para al menos una p_i . Con n_m el mayor de los enteros n_i , se tiene que

$$\sum_{i=1}^N \frac{1}{2^{n_i}} \leq 1 - \frac{1}{2^{n_m}} \quad (3.22)$$

⁷ La importancia de crear un código óptimo es que con el, al disminuir su longitud de palabra y darle mayor importancia a las palabras con mayor probabilidad de ocurrencia, se logra que la velocidad de transmisión en el sistema aumente y por lo tanto su eficacia.

Por que el lado izquierdo es un entero binario más pequeño que 1. Si después n_m es cambiado por n_m-1 , el resultado de la suma del valor no excederá 1. Por lo que la simplificación es la siguiente:

$$\sum_{i=1}^N \frac{1}{2^{n_i}} = 1 \quad l_i \leq n_i \tag{3.23}$$

Por lo que con este conjunto de enteros es posible crear el código denotando que L^a es la longitud media. Entonces tenemos

$$L^a = \sum_{i=1}^N p_i l_i \leq \sum_{i=1}^N p_i n_i \tag{3.24}$$

Por lo que el código de Shannon debe cumplir con la siguiente condición:

$$H(X) \leq L^a \leq H(X) + 1 \tag{3.25}$$

Teorema

Una forma sencilla de encontrar el código de Shannon es mediante el siguiente método:

- Se ordenan los valores x_i de mayor a menor probabilidad.
- Se divide el resultado en dos grupos, procurando que la probabilidad de cada grupo esté lo más próxima posible.
- Un grupo se codificará como cero ("0") y el otro como uno ("1").
- Seguidamente se repite el proceso en cada grupo, de forma recursiva, mientras cada grupo tenga más de un dato x_i .

Ejemplo: Si se desea obtener el código de Shannon para el siguiente conjunto de probabilidades 0.4, 0.2, 0.2, 0.1, 0.07, 0.03, se tiene que:

0.4	0	0		
0.2	0	1		
0.2	1	0		
0.1	1	1	0	
0.07	1	1	1	0
0.03	1	1	1	1

TABLA 3.1 CÓDIGO DE SHANNON

El código Shannon para esta fuente sería: 00, 01, 10, 110, 1110, 1111.

Código de Huffman: Este código es un código óptimo ya que es el código de menor longitud media.

La construcción de este código se fundamenta en asignar a cada símbolo del alfabeto fuente una secuencia de bits cuya longitud esté relacionada de forma directa con la probabilidad de aparición de ese símbolo. De esta forma, a los símbolos con mayor frecuencia de aparición se les asignarán las palabras de código de menor longitud. En el proceso de construcción de este código es:

- Ordenar el conjunto de símbolos del alfabeto fuente en orden decreciente de probabilidades de aparición.
- A continuación se juntan los dos símbolos con menor probabilidad de aparición en un único símbolo cuya probabilidad será la suma de las probabilidades de los símbolos que dieron origen a este nuevo símbolo.
- Se repite este proceso hasta que sólo tengamos dos símbolos.
- A continuación se realiza el proceso de codificación. Primeramente asignamos un 1 a uno de los dos símbolos que tenemos y un 0 al otro.
- Posteriormente recorreremos la estructura que hemos construido hacia atrás de forma que cuando dos símbolos hayan dado origen a un nuevo símbolo, estos dos símbolos "heredarán" la codificación asignada a este nuevo símbolo y a continuación se le añadirá un 1 a la codificación de uno de los símbolos y un 0 a la del otro símbolo.

3.4.2 MEDIDAS DE DISTORSIÓN

Sin embargo, después de realizar la codificación de las palabras del alfabeto que se desean transmitir, nos topamos con un problema llamado ‘distorsión’. Literalmente, la definición de distorsión es la siguiente:

Distorsión es la diferencia entre señal que entra a un equipo o sistema y la señal de salida del mismo. Por tanto, puede definirse como la "deformación" que sufre una señal tras su paso por un sistema. La distorsión puede ser lineal o no lineal⁸.

⁸ La distorsión es un factor que tiene gran relevancia en las comunicaciones en general, ya que no importa que los sistemas cuenten con medidas de corrección muy efectivas, estas variaciones afectarán a la señal en cualquiera de los puntos del sistema de comunicaciones, desde la propia señal original y en cada uno de los bloques que conforman al transmisor, al receptor y al sistema de comunicaciones.

Además, en teoría de la información la distorsión debe ser medida para saber que tanto se aleja la palabra transmitida de la original y tenerlo en cuenta al momento de recuperar la información.

Hay dos clases de medidas de distorsión: medidas subjetivas y medidas objetivas de distorsión. Las primeras están basadas generalmente en percepciones humanas analizando las señales mediante métodos de observación, mientras que las segundas permiten obtener información más detallada que da la pauta para comparar el comportamiento de diferentes señales y sistemas.

La distorsión D puede calcularse de la misma forma que en la codificación de fuente, tomando en consideración que la distorsión temporal se crea entre el canal y la fuente:

$$D = \frac{1}{m} E(d(U, V)) \quad (3.26)$$

donde U y V representan los vectores de la fuente y la salida respectivamente, y el término $d(U, V)$ la medida de distancia entre ambos vectores. Y m representa el número de bits que componen el vector fuente (longitud del vector fuente).

- *Distancia de Hamming:* Usando la distancia de Hamming se tiene

$$\delta = \frac{1}{m} E[d_H(U, V)] \quad (3.27)$$

donde δ es la medida de la probabilidad de error del bit, que representa la media del número de errores entre el vector de entrada y salida, normalizada por la longitud del vector.

- *Distancia de Euclidean:* Usando la distancia de Euclidean se tiene

$$D = \frac{1}{m} E[d_E(U, V)] \quad (3.28)$$

donde $d_E = \|U - V\|^2$. Esta medida es llamada error medio cuadrático.

3.5 FUNCIONALIDAD TEÓRICA ÓPTIMA

La funcionalidad teórica óptima, o por sus siglas en inglés OPTA (*Optimal Performance Theoretically Attainable*), hace referencia a la tasa de transferencia como una función de la menor

probabilidad de error, cuando se hace una transmisión con una fuente y un canal dados. De acuerdo a la teoría de Shannon, la curva OPTA denotada por $R(D)$ está dada por:

$$R(D) = \frac{R_S(D)}{C} \frac{\text{símbolo del canal}}{\text{símbolo de la fuente}} \quad (3.29)$$

donde $R(D)$ representa la función de tasa de distorsión entre la fuente y el canal, y $R_S(D)$ es la función de distorsión de la fuente y C es la capacidad del canal.

Capítulo 4

CUANTIZACION VECTORIAL

Como se mencionó en el capítulo tres, un sistema de comunicaciones está formado por un transmisor, un canal de transmisión y un receptor. Debido a que el tema de 'cuantización' concierne al transmisor, solo nos enfocaremos a estudiar dicho componente.

Una *señal analógica* es aquella en donde su amplitud que toma diferentes valores, es decir, su nivel en cualquier muestra no está limitado a un conjunto finito de niveles predefinidos como es el caso de las señales cuantificadas. La mayoría de las señales ya sean audio, voz, etc. generalmente tienen una naturaleza analógica.

Por otro lado, las *señales digitales* son aquellas cuyas dimensiones tanto en tiempo como en amplitud son discretas, lo que determina que la señal deberá tomar solo ciertos valores establecidos. Si se desea transmitir una señal analógica por un medio digital, primero se debe convertir dicha señal en una señal digital, es decir, llevar a cabo una transformación analógico-digital, o mejor dicho, digitalizar la señal.

La digitalización o conversión analógica-digital consiste en realizar de forma periódica medidas de la amplitud de una señal, redondear sus valores a un conjunto finito de niveles preestablecidos de tensión (*conocidos como niveles de cuantificación*) y registrarlos como números enteros en cualquier tipo de memoria o soporte. Para esto se realizan tres diferentes procesos:

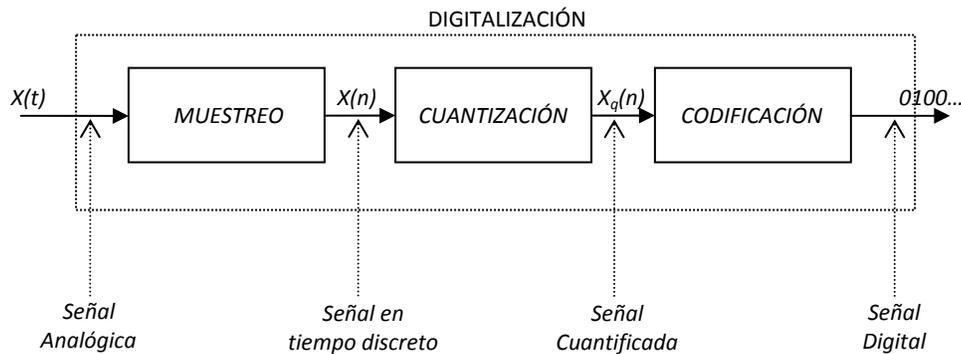


FIG. 4.1 PROCESO DE DIGITALIZACIÓN

1. *Muestreo*: O en inglés ‘*sampling*’ consiste en tomar muestras periódicas de la amplitud de onda. La velocidad con que se toma esta muestra se conoce como frecuencia de muestreo.
2. *Cuantización*: Es el proceso mediante el cual se mide el nivel de voltaje de cada una de las muestras. Consiste en asignar un margen de valor de una señal analizada a un único nivel de salida.
3. *Codificación*: Como se mencionó en el capítulo anterior, la codificación consiste en traducir los valores obtenidos durante la cuantificación a los códigos que se vayan a utilizar.

4.1 FUNDAMENTOS

Como se mencionó anteriormente, la cuantización consiste en establecer niveles de voltaje a las diferentes muestras que se hagan de la señal original, con la finalidad de enviar información solo en un número finito de valores de voltaje.

Además, existen diferentes tipos de cuantizadores, cada uno de ellos con diferentes características y propiedades. A continuación se mencionarán algunos de los tipos de cuantización más utilizados de forma simplificada:

Cuantización Escalar

Este tipo de cuantización es un mapeo de los números reales a un determinado conjunto representado por $C = \{y_1, y_2, \dots, y_N\}$, donde $y \in \mathbb{R}$, y $0 < N < \infty$. Asociado al conjunto C se tienen regiones R_i tales que:

$$R_i = \{x; Q(x) = y_i\} \quad (4.1)$$

Las cuales cumplen con las siguientes condiciones:

1. $R_i \cap R_j = \emptyset, \quad i \neq j, \quad i, j = 1, \dots, N$
2. $\bigcup_{i=1}^N R_i = R$

En la cuantización escalar la fuente se considera muestra por muestra y por lo general se asume que el canal es libre de ruido y otro tipo de distorsiones. En este tipo de cuantización se calculan *niveles de decisión* U_i , y los niveles de salida se conocen como *niveles de reconstrucción* V_i , mientras que la distancia entre U_i y U_{i+1} es llamada intervalo de cuantización.

Para generar de una manera precisa los niveles de decisión se toman las siguientes consideraciones:

- Sea una señal de entrada modelada por una variable aleatoria X ,
- Sea $F_x(x)$ la función de densidad de probabilidad (*PDF*) de X ,
- Para un cuantizador de M intervalos, al que representamos como la función $Q(x)$, por lo que se deben conocer:

Niveles de decisión:

Son los $M + 1$ puntos límites de los intervalos

Se representan por $\{b_i\}_{i=0}^M$

Niveles de reconstrucción:

Son los valores representativos de los intervalos (salidas)

Se representan por $\{y_i\}_{i=0}^M$

Ejemplo: A continuación se presenta un ejemplo de un cuantizador de 3 bits, el cual permite codificar en 8 niveles de voltaje:

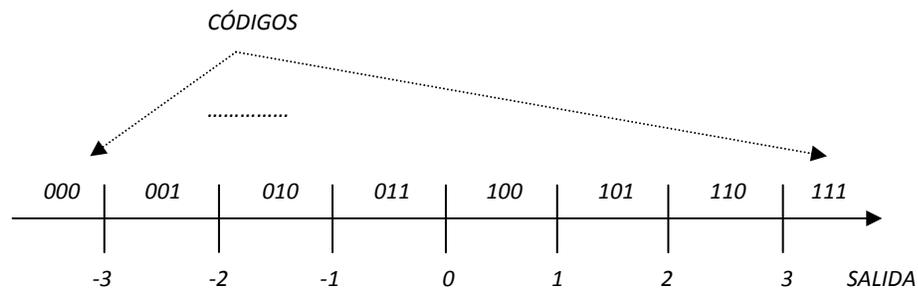


FIG. 4.2 EJEMPLO DE CUANTIZADOR DE 3 BITS

En los *cuantizadores escalares* existen dos diferentes regiones que son: *regiones granulares*, que tienen un intervalo finito; y las *regiones de sobrecarga*, que tienen un intervalo infinito.

En este tipo de cuantizadores, la *distorsión* depende de la partición en intervalos elegida y los valores representativos elegidos, mientras que la *longitud media* depende de la partición y de la longitud de las *palabras código* elegido.

Cuantización Uniforme

La cuantización uniforme es el tipo de cuantización más simple que hay, ya que todos los intervalos tienen un mismo tamaño (tamaño de escalón) Δ , excepto quizá los intervalos extremos.

Todos los niveles de reconstrucción están uniformemente espaciados, y en los intervalos internos son los puntos medios de los intervalos.

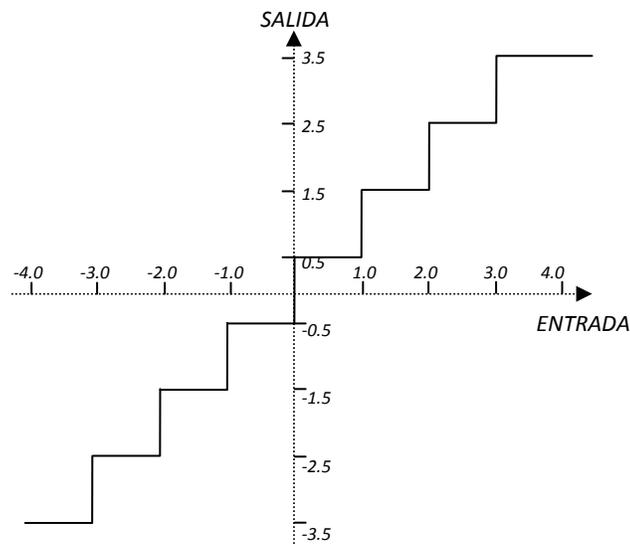


FIG. 4.3 CUANTIZADOR UNIFORME MIDRISE

Al igual que los cuantizadores escalares, existen dos tipos de cuantizadores uniformes: *midrise*, en los cuales el cero no es considerado como un nivel de reconstrucción, por lo que tiene un número par de niveles; y los *midtread*, donde el cero sí forma parte de los niveles de reconstrucción.

Cuantización Adaptativa

Este tipo de cuantización se encarga de adaptar dinámicamente las características del cuantizador a la estadística de la entrada.

Y hay dos tipos de cuantización: *adaptativa hacia adelante* (*off-line o forward-adaptive*), donde la salida de la fuente se divide en bloques de datos, analizando posteriormente cada uno para modificar los parámetros del cuantizador antes de cuantizar el bloque, y finalmente, los nuevos parámetros del cuantizador se transmiten al receptor; y la cuantización *adaptativa hacia atrás* (*on-line o backward-adaptive*), donde los parámetros del cuantizador se modifican en función de las características de los últimos datos ya cuantizados.

Cuantización no uniforme

Este tipo de cuantizadores tienen su utilización cuando las mayores probabilidades se den en torno al origen, y una manera de disminuir la distorsión es aproximar mejor la entrada en las zonas de mayor probabilidad. Esto se logra (sin aumentar el número de intervalos) utilizando intervalos más pequeños en las zonas de alta probabilidad.

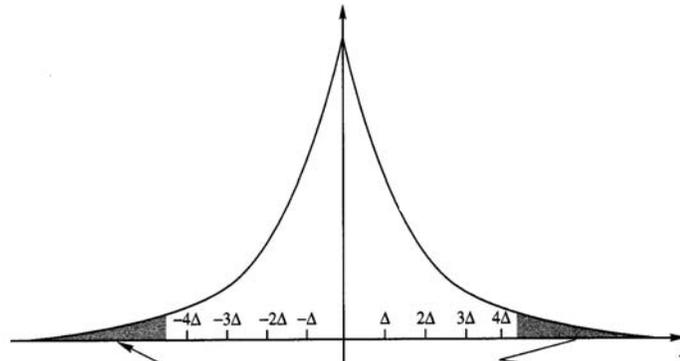


FIG. 4.4 PROBABILIDADES MAYORES CERCANAS AL ORIGEN

Este tipo de cuantizadores están formados por:

- *Compresor*: El cual se encarga de amplificar el nivel de las muestras que caen en regiones de alta probabilidad y reducir el nivel de las que caen en zonas de baja probabilidad.
- *Cuantizador uniforme*: Una vez que se soluciona el problema de las zonas de probabilidad, el sistema se puede manejar con un cuantizador uniforme.
- *Descompresor (expander)*: El cual, como su nombre lo indica, implementa la función inversa del compresor.

Cuantización Vectorial

El último tipo de cuantización es la cuantización vectorial. A partir de este momento, el presente trabajo estará enfocado únicamente a describir los fundamentos y características de este tipo de cuantización, así como los métodos que existen para su implementación, y las posibles optimizaciones que se le puedan agregar.

La *cuantización vectorial* es un tipo de cuantización que trabaja en bloques de datos, en lugar de datos aislados, y dichos bloques pueden ser representados por vectores, de ahí el nombre.

La ventaja de trabajar sobre bloques de datos radica en que es más fácil lograr una buena compresión sin tener tantas pérdidas, y por lo tanto, se puede lograr que la longitud de palabra sea menor para una distorsión dada, o una menor distorsión para una longitud de palabra dada.

A continuación se presenta la definición matemática de la cuantización vectorial:

Un cuantizador vectorial de dimensión \mathbf{K} y tamaño N es un mapeo del espacio \mathbb{R}^K , $\bar{x} = [x_1 \dots x_K]$ a un conjunto con N elementos $C = \{\bar{y}_1, \dots, \bar{y}_N\}$ donde y_i pertenece al conjunto de los números reales. Asociado a cada \bar{y}_i se tienen regiones R_i tales que:

$$R_i = \{x; Q(x) = y_i\} \quad (4.2)$$

Las cuales cumplen con las siguientes condiciones:

1. $R_i \cap R_j = \emptyset, \quad i \neq j, \quad i, j = 1, \dots, N$
2. $\bigcup_{i=1}^N R_i = \mathbb{R}^K$

Construcción de un cuantizador vectorial:

La creación de un cuantizador vectorial cuenta de tres pasos específicos, los cuales son: crear un *codebook* (o libro de palabras o diccionario), un codificador y finalmente el decodificador.

1. **Codebook:** Se genera con la finalidad de poseer todos los vectores que van a representar la información transmitida y este se genera únicamente al inicio del diseño, por lo que en general, no se modifica su contenido, pero eso depende de la aplicación. Para su construcción existen diversos algoritmos, siendo el algoritmo LGB de los más usados, el cual se basa en división binaria de códigos (más adelante se explicarán las características y funcionamiento de este procedimiento).

Algunos otros métodos para crear el *codebook* inicial son los siguientes:

Random Coding: Proceso mediante el cual se elige aleatoriamente N vectores de la secuencia de entrenamiento.

Algoritmo de Pruning: Es un algoritmo poco costoso que se adecua a la estadística de la secuencia de entrenamiento en donde los vectores del *codebook* hallado pertenecen a la secuencia de entrenamiento y todos los vectores de la secuencia de entrenamiento distan menos de un cierto *umbral* al *codebook* hallado.

Clustering: Este proceso se inicia con todos los vectores de la secuencia de entrenamiento como *codevectors*, se agrupan de a dos y son sustituidos por su centroide. Aquí, los vectores del *codebook* no son generalmente elementos de la secuencia de entrenamiento, pero son muy pesados computacionalmente.

Splitting: Aquí se comienza con el centroide de la secuencia de entrenamiento como único integrante del *codebook*. Luego se divide en dos *codevectors* perturbándolo; y se aplica la iteración de Lloyd. Se repite este procedimiento hasta llegar al tamaño de *codebook* buscado. Este procedimiento es muy utilizado en el algoritmo LGB.

- Codificación:** Este es el proceso que carga con la mayor dificultad, ya que se encarga de realizar una búsqueda para poder asociar un vector del diccionario creado con anterioridad, el cual debe ser lo más cercano posible y que sea capaz de representar al vector de entrada.

Es decir, el proceso consiste en tomar un vector X que corresponda a la entrada y compararlo con el contenido del diccionario, y de esta forma saber con cual de ellos se tiene la menor distorsión posible (es decir, la menor distancia). Una vez localizado se marca su posición mediante los índices i y manda esta información al decodificador.

A continuación se presenta el funcionamiento general de la cuantización vectorial:

El primer paso consiste en agrupar los datos que provienen de la fuente en bloques, o vectores. El codificador y el decodificador contienen un conjunto de vectores L -dimensionales llamado *codebook* (libro de códigos, o simplemente, diccionario). Los vectores de este 'diccionario' se llaman 'vectores de código' y se seleccionan de tal manera que sean una representación de las muestras emitidas.

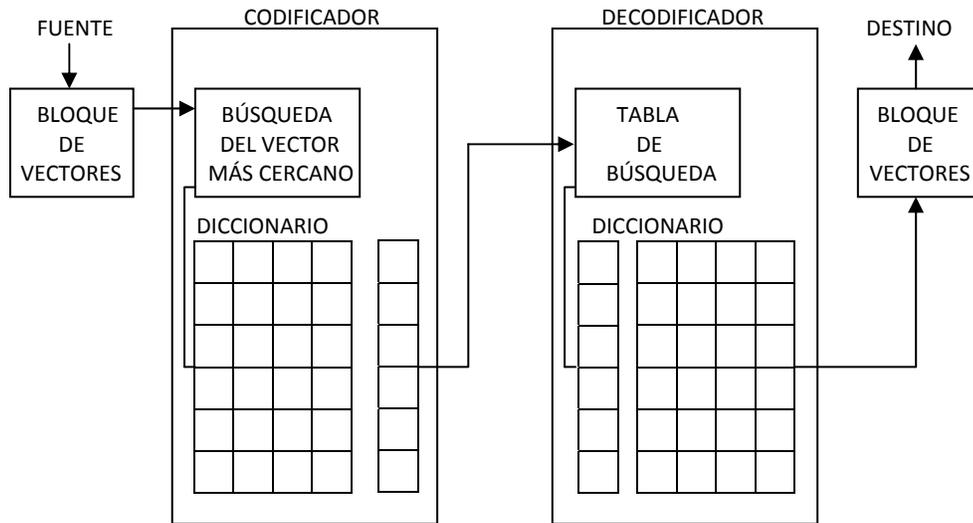


FIG. 4.5 CUANTIZACIÓN VECTORIAL

El diccionario asigna índices a cada vector código, y el codificador busca el vector código más cercano al vector de entrada. Después se transmite el índice de este vector código y el decodificador genera el mismo vector código, produciendo como salida la secuencia de componentes del vector código.

Por ejemplo, si el tamaño del diccionario es K y los vectores poseen una dimensión L , el índice requiere $\log_2(K)$ bits y cada vector contiene valores para reconstruir L muestras consecutivas de señal. Por tanto, la longitud media está dada por:

$$r = \frac{\log_2 K}{L} \quad (4.3)$$

3. **Decodificación:** Este proceso se encarga de identificar la posición del vector en el diccionario, entregando los datos que posee dicho vector, con lo que finalmente se recupera la información.

4.2 EL ALGORITMO LLOYD MAX GENERALIZADO

El algoritmo de Lloyd está basado en iteraciones optimizadas en el codificador y decodificador. El algoritmo de Lloyd es un proceso que se encarga de discretizar o crear particiones de un cierto espacio. Esto se realiza mediante la generación de zonas de 'Voronoi' definidas mediante una medida de distancia y un conjunto de ejemplos representativos o prototipos.

El objetivo de este algoritmo es generar el conjunto de prototipos que minimice una determinada medida de distorsión o error.

Después de terminar con cada una de las iteraciones se calcula el valor de la distorsión entre el codificador y el canal de transmisión. Esta distorsión está dada por:

$$D = \frac{1}{m} E(D_X(U)) \quad (4.4)$$

donde $D_X(U)$ representa la distorsión cuando se transmite un vector de entrada U sobre el canal.

4.2.1 CONCEPTOS

El algoritmo de Lloyd generalizado, o también conocido como algoritmo de *k-medias*, consiste en el siguiente proceso:

- Se tiene un alfabeto inicial C_1 . Se tiene que $m=1$.
- Dado el alfabeto C_m , se realiza la iteración de Lloyd para obtener un nuevo alfabeto C_{m+1} .
- Después es necesario calcular la distorsión de este nuevo alfabeto mediante la siguiente expresión:

$$D = \frac{1}{M} \sum_{j=1}^M \min_{\vec{y} \in C} (d(\vec{x}_j, \vec{y})), \quad \vec{y} \in C \quad (4.5)$$

- Finalmente, si se nota que la variación del valor de la nueva distorsión disminuyó en un valor muy pequeño, se dejan de realizar las iteraciones, pero si el cambio es mayor, se debe de modificar el valor de m por $m+1$ y se debe repetir nuevamente todo el proceso.

Iteración de Lloyd

Pero, ¿qué es una iteración de Lloyd? Para poder llevar a cabo el algoritmo de Lloyd, y así poder encontrar los valores de cuantización que general la menor distorsión posible, se deben realizar mediciones cíclicas que son conocidas como Iteración de Lloyd. Estas siguen el siguiente método.

- Dado un alfabeto $C_m = \{\vec{y}_i; i = 1, \dots, N\}$, dividir el conjunto de entrada T en regiones R_i mediante la siguiente condición:

$$R_i = \{\vec{x} \in T: d(\vec{x}, \vec{y}_i) \leq d, \text{ para todo } j \neq i\} \quad (4.6)$$

- Después de que se tengan las regiones necesarias, se calculan los centroides de dichas regiones para recalcular el alfabeto. Y finalmente hacer $C_{m+1} = \{\text{cent}(R_i)\}$.

NOTA: Sin embargo, si al realizar este proceso se genera una celda vacía, se debe asignar un vector alternativo para dicha celda:

$$\text{cent}(R) = \frac{1}{\|R\|} \sum_{\vec{x}_i \in R} \vec{x}_i \quad (4.7)$$

Regiones de Voronoi

Las regiones de Voronoi son un método de interpolación, basado en la distancia euclidiana. Se crean al unir los puntos entre sí, trazando las mediatrices de los segmentos de unión. Las intersecciones de estas mediatrices determinan una serie de regiones en un espacio bidimensional alrededor de un conjunto de puntos de control¹.

Un diagrama de Voronoi es una estructura geométrica que representa información de proximidad acerca de un conjunto de puntos u objetos es decir, el diagrama de Voronoi de un conjunto de objetos geométricos es una partición del espacio en celdas cada una de la cuales contiene una colindancia con sus puntos más cercanos.

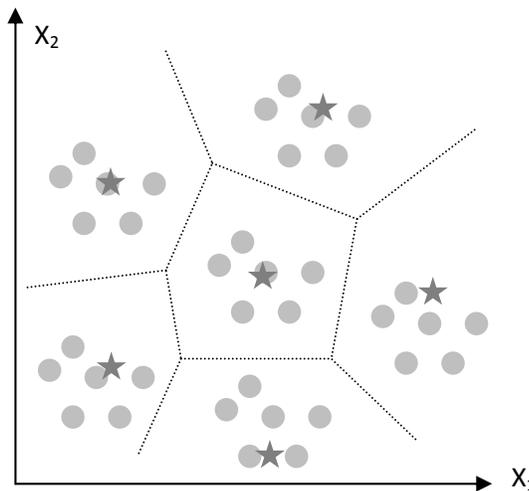


FIG. 4.6 REGIONES DE VORONOI

¹ Las regiones de Voronoi permiten conocer de forma gráfica las zonas que comprenden a cada uno de los centroides generados.

Si se tiene un codificador cualquiera, es posible disminuir el valor de la distorsión si a cada uno de los vectores que se desean representar se les asigna un centroide. Es decir, el centroide es un punto promedio que se localiza en las regiones de Voronoi, por lo tanto, la existencia de estas regiones permite una compresión más efectiva ya que el diccionario se localiza tanto en el codificador como en el decodificador, permitiendo de esta forma enviar una cantidad menos de vectores que estén asociados a las palabras código de cada *codebook*.

Construcción del diagrama de Voronoi

Para construir óptimamente el diagrama de Voronoi de un conjunto de puntos S , se aplica el algoritmo conocido como *divide y vencerás*. Las regiones se denotan por $Vor(S)$ y el procedimiento es el siguiente:

1. Particionar el conjunto S en dos subconjuntos S_1 y S_2 , de igual tamaño aproximadamente, usando la media de la coordenada x ,
2. Construir $Vor(S_1)$ y $Vor(S_2)$ recursivamente,
3. Finalmente, se repiten estos pasos hasta que se tengan las regiones que se desean o de acuerdo a la utilidad que se les desee dar, en el caso de la cuantización vectorial, el número de regiones depende del valor de la distorsión generada.

4.2.2 EL PRINCIPIO DEL VECINO MÁS CERCANO

Este principio se utiliza en el caso de querer codificar una serie de datos mediante palabras códigos que resulten en la menor distorsión posible.

Si se tienen varios vectores que se deseen codificar dentro de una región R_i específica, *el principio del vecino más cercano* se encargará de determinar en cuál de esas regiones debe colocarse el vector en cuestión basándose principalmente en la distorsión producida, es decir, en la distancia a dicha región, ya que a mayor distancia entre el vector y la región, la distorsión aumentará, y de igual forma, disminuirá si la distancia se reduce.

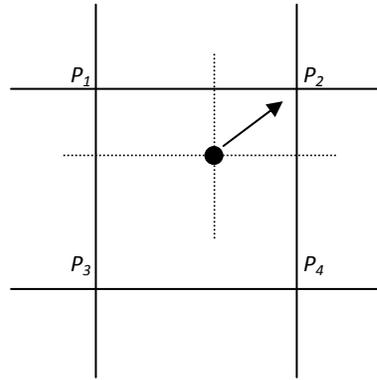


FIG. 4.7 PRINCIPIO DEL VECINO MÁS CERCANO

Para reducir la distancia es necesario minimizar el valor de $D_x(u)$ para todas las u dadas. Este principio ayuda a encontrar las regiones de codificación denotadas por Θ_x :

$$u \in \Theta_x \Leftrightarrow x = \text{Arg min}_x D_x(u) \quad (4.8)$$

es decir:

$$\Theta_x = \{u | D_x(u) \leq D_\ell(u); \forall \ell\} \quad (4.9)$$

donde U representa los vectores de entrada en el canal, y $D_x(U)$ denota la distorsión cuando las entradas U pasan a través del canal.

Generalización:

Debido a que en muchos casos es posible considerar a la distorsión incluyendo el concepto del error medio cuadrático como:

$$D = \frac{1}{m} E \|U - V\|^2 \quad (4.10)$$

Con la medida del error medio cuadrático, la formulación del vecino más cercano se da de la siguiente manera:

$$u \in \Theta_x \Leftrightarrow x = \text{Arg min}_x E_y (\|u - v_x\|^2) \quad (4.11)$$

Por lo que reformulando esta ecuación:

$$\Theta_x = \{u | E_y (\|u - v_x\|^2) \leq E_\ell (\|u - v_\ell\|^2); \forall \ell\} \quad (4.12)$$

4.2.3 EL PRINCIPIO DEL CENTROIDE

Estado en donde las representaciones de las salidas para las regiones dadas tienen la característica de minimizar la distorsión total.

Si se desea codificar una cierta cantidad de vectores en las regiones R_i dadas, *el principio del centroide* se encarga de calcular el promedio de los vectores contenidos dentro de una sola región, con la finalidad de encontrar un punto (o vector) con el cual se pueda disminuir la distorsión de toda la región con respecto al vector de entrada

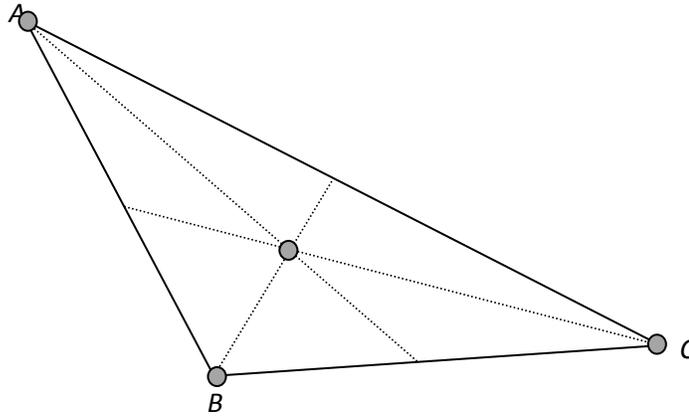


FIG. 4.8 PRINCIPIO DEL CENTROIDE

Para una salida y , el decodificador de la salida V_y^* que optimice la distorsión es:

$$V_y = E_{Y=y} D_x(u) \quad (4.13)$$

Generalización:

Conociendo la derivada de D con respecto a V_y , el valor de decodificación para cada una de las salidas se encuentra para ser la expresión matemática de la entrada sabiendo que la salida ha sido observada. La función de decodificación es la siguiente:

$$\begin{aligned} D &= E_y \left(\frac{1}{m} E_{u|Y=y} (\|u - V_y\|^2) \right) \\ \frac{\partial D}{\partial v_y} &= \frac{\partial \frac{1}{m} E_{u|Y=y} (\|u - V_y\|^2)}{\partial v_y} = 0 \\ &\Rightarrow \frac{2}{m} E_{u|Y=y} (u - V_y) = 0 \end{aligned} \quad (4.14)$$

$$V_y = E_u(u|Y = y) \quad (4.15)$$

Esta ecuación explica que el valor óptimo para cada uno de los centroides y es el promedio de todas las entradas sabiendo que el código recibido es y .

4.3 ESQUEMA DE DISEÑO LGB

La cuantización de vector fuente en un canal sin ruido fue propuesta en enero de 1980, y es conocida como LGB por las iniciales de las tres personas que la inventaron: Linde Yoseph, Buzo Andrés y Gray Robert.

Este algoritmo tiene como finalidad de este algoritmo es generar un buen *codebook*, o diccionario disminuyendo el valor de la distorsión, y es similar al algoritmo de *k-medias*. Este principio de cuantización es una variable del algoritmo de Lloyd y está basado en los principios mencionados anteriormente, el principio del vecino más cercano y el principio del centroide.

Además puede inicializarse tanto en el codificador como en el decodificador. Las iteraciones que se van realizando se detienen cuando el proceso empieza a converger a estos principios. Y el sub-espacio que es codificado con y es llamado, *regiones de Voronoi* o *células*

$$C_y = \{u|u \text{ es codificada con } y\} \quad (4.16)$$

y la reconstrucción para las salidas de las *regiones de Voronoi* son llamadas centroides, v_y .

En forma muy general, en cada una de las iteraciones realizadas, cada uno de los vectores es dividido en otros dos nuevos mediante el siguiente proceso:

- Primero se tiene un estado inicial $x(0)$, es decir, un centroide de la secuencia de entrenamiento;
- Después de la primera estimación se genera un diccionario de dimensión dos y una estimación final después de aplicar el algoritmo de Lloyd, es decir, se genera un diccionario óptimo de dos vectores. Este proceso se lleva a cabo mediante *splitting*;
- Finalmente, se vuelve a repetir el proceso para ir generando cada un diccionario óptimo que contenga una mayor cantidad de vectores.

Usando este algoritmos mediante una secuencia de entrenamiento, se inicia con un solo nivel de cuantización que consiste en el centroide de la secuencia de entrenamiento, a partir de donde estos vectores son divididos en otros dos nuevos vectores generando de esta manera un cuantizador de dos niveles de cuantización. Finalmente se realiza el mismo procedimiento para cada uno de estos vectores generando un cuantizador de cuatro niveles, y así sucesivamente, por lo que los niveles se dan en el siguiente orden: 1, 2, 4, 8, ... , N niveles. A continuación se presenta un diagrama que representa la metodología de este algoritmo:

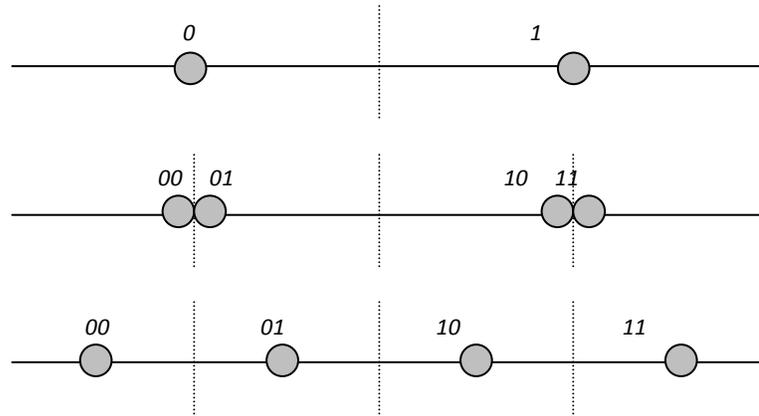


FIG. 4.9 DESCRIPCIÓN DEL ALGORITMO LGB

Como se mencionó con anterioridad, el algoritmo LGB está basado en los principios del vecino más cercano y del centroide, por lo que para este algoritmo se definen las siguientes ecuaciones:

Para el vecino más cercano se tiene:

$$\Theta_x = \{u | (\|u - v_x\|^2) \leq (\|u - v_\ell\|^2); \forall \ell\} \tag{4.17}$$

Y para el centroide:

$$V_y = E_{u \in \Theta_y}(u) \tag{4.18}$$

Como se puede observar, las expresiones obtenidas para el vecino más cercano y para los centroides son muy similares a las obtenidas en LGB que en algoritmo de Lloyd-Max, pero con notación de vector.

4.4 ESQUEMAS COSQ Y COVQ

Esquema COSQ

El esquema de canal optimizado por cuantización escalar o *COSQ* por sus siglas en inglés (*channel optimized scalar quantization*) fue propuesto ya que se tenía la necesidad de disminuir el valor de la distorsión debida al canal y considerando que se usa una fuente escalar.

Este esquema fue propuesto por Farvardin y Vaishampayan, quienes lograron las condiciones necesarias para una cuantización óptima escalar.

Las expresiones para el vecino más cercano y centroide son similares a las utilizadas en el algoritmo de *Lloyd-Max*, pero reaparecen los términos de expectativas en notación escalar como se muestra a continuación:

$$\Theta_x = \{u | E_y(\|u - v_x\|^2) \leq E_y(\|u - v_\ell\|^2); \forall \ell\} \quad (4.19)$$

$$V_y = E_u(u | Y = y) \quad (4.20)$$

Esquema COVQ

El esquema de canal optimizado por cuantización vectorial o *COVQ* por sus siglas en inglés (*channel optimized vector quantization*) es considerada como una generalización de trabajos en k -dimensiones, o también como una versión del algoritmo *LGB* descrito anteriormente, pero en un ambiente con ruido. Este esquema es capaz de minimizar la distorsión tomando en cuenta los errores de cuantización y los errores debidos a las perturbaciones del canal.

En *COVQ*, la fuente es cualquier vector con muestras de amplitud continua con o sin memoria. Y sus expresiones generales para el vecino más cercano y el centroide son las siguientes, las cuales manejan notación de vector, e incluyen los efectos del ruido:

$$\Theta_x = \{u | E_y(\|u - v_x\|^2) \leq E_y(\|u - v_\ell\|^2); \forall \ell\} \quad (4.21)$$

$$V_y = E_u(u | Y = y) \quad (4.22)$$

La metodología que utiliza este tipo de cuantización es muy similar a la mencionada en el algoritmo *LGB*, y esto se debe a como se mencionó, es una versión en un ambiente con ruido. Este algoritmo también se basa en el *splitting*, pero de una forma algo diferente, ya que ahora el proceso se realiza en dos dimensiones. En el siguiente diagrama se muestra el proceso que se realiza en este tipo de cuantización:

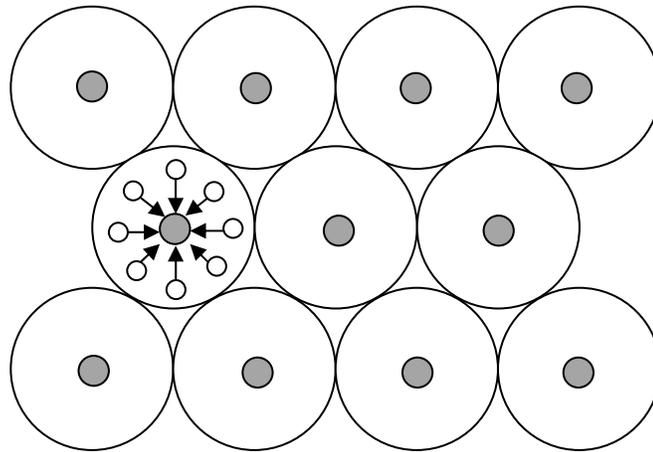


FIG. 4.10 DESCRIPCIÓN DEL ALGORITMO COVQ

Como se observa en la figura 4.10, el proceso de cuantización vectorial tiene como finalidad establecer ‘regiones de voronoi’ a lo largo de todo el plano. Cada una de estas regiones está representada por un punto central llamado centroide el cual es el promedio de todos los puntos que abarca dicha zona. Con este procedimiento, todos los puntos muestrales que caigan dentro de esa región serán representados por ese valor de centroide y así se estará disminuyendo de forma drástica el número de elementos que se envían en la transmisión.

Por tal motivo, este algoritmo de cuantización también está basado en los principios del centroide y del vecino más cercano.

4.5 ESQUEMA COMPARATIVO

Como se observó en las ecuaciones para el vecino más cercano y el centroide de los anteriores tres esquemas de diseño, LGB, COSQ y COVQ, es posible darse cuenta de que estas son muy similares, sin embargo la diferencia entre COSQ y COVQ radica en que el primer método es una representación *escalar* de una fuente, mientras que en el caso de COVQ, se tienen las mismas ecuaciones pero para el caso de una fuente de tipo *vectorial*.

Por otro lado, cabe mencionar que tanto los esquemas LGB como COVQ hacen referencia a una fuente vectorial, así que, ¿cuál es la diferencia entre estos esquemas?

La respuesta es la siguiente y basándonos en las siguientes ecuaciones correspondientes a los dos principios que hemos estado tratando, pero en su forma general (omitiendo por el momento el concepto de error medio cuadrático) tenemos que para el vecino más cercano,

$$\theta_x = \{u | D_x(u) \leq D_\ell(u); \forall \ell\}, \quad x \in \{1, \dots, L\} \quad (4.23)$$

y para el centroide,

$$V_y = \frac{\sum_{x=1}^L p(y|x) \int_{\theta_x} u p(u) du}{\sum_{x=1}^L p(y|x) \int_{\theta_x} p(u) du}, \quad y \in \{1, \dots, L\} \quad (4.24)$$

Y a partir de este par de ecuaciones correspondientes al esquema COVQ es posible generar las ecuaciones formuladas para LGB, asumiendo que el canal por el que se transmite es libre de ruido, es decir, haciendo $p=0$. Por lo tanto se puede decir que LGB únicamente es un caso especial del esquema COVQ.

Por el momento solo es posible conocer las diferencias entre estos tres esquemas basándonos únicamente en sus expresiones matemáticas y mediante sus características principales, por lo que no es fácil decidir cual de ellos resultan ser más adecuados.

Sin embargo, en el siguiente capítulo se abordarán más a fondo las diferencias entre los diferentes esquemas de diseño del *codebook* basándose en los esquemas vistos y mediante la realización de simulaciones que permitirán observar detalladamente como se realizan dichos procesos y con esto, llegar a una conclusión que defina que método es el más conveniente basándonos en el valor de la distorsión generada y la longitud de palabra.

Capítulo 5

SIMULACIÓN DE LA CUANTIZACIÓN ESCALAR Y VECTORIAL

Como se vio en cada uno de los capítulos anteriores, los principales métodos utilizados para la cuantización están basados en la creación de un *codebook*, en el cual se localizan todos los posibles valores correspondientes a la señal de entrada de un sistema. Este *codebook* se establece tanto en el transmisor como en el receptor, con la finalidad de comparar la muestra de entrada al transmisor con el diccionario, y así asignarle el valor correspondiente, el cual será enviado al receptor, en donde se especificará únicamente la ubicación de esa palabra de código. Esto permite que la distorsión que afecta a la señal sea mínima, ya que de inicio, el *codebook* se creará a partir de valores de distorsión, lo cual por si mismo disminuye la probabilidad de error en el receptor.

Debido a que estos temas fueron tratados con anterioridad en los demás capítulos a continuación se presentará la simulación realizada en *MATLAB* del proceso de cuantización por diferentes métodos y tomando diferentes consideraciones para su implementación. Es decir, se presentarán las simulaciones de los algoritmos LGB, COVQ basándonos en la generación de vectores aleatorios para la creación del *codebook*, y tomando como parámetro determinante la distorsión.

Además de presentar los algoritmos correspondientes tanto para la cuantización vectorial como la escalar, se presentarán los resultados obtenidos después de la generación de varios diccionarios variando las condiciones iniciales.

Finalmente se presenta un anexo en donde se incluyen si excepción los diccionarios generados para todos los casos, así como sus representaciones gráficas.

5.1 ALGORITMO PARA LA CUANTIZACIÓN ESCALAR

Algoritmo:

El proceso que se seguirá para realizar la simulación de la cuantización escalar estará basado en el esquema conocido como LGB, el cual a su vez, se basa en *splitting*, es decir, en ir dividiendo cada una de las regiones en dos partes. Este tema se explicó con mayor detenimiento en el capítulo 4. El algoritmo a seguir en la programación y generación del *codebook* es el siguiente:

1. Se debe generar una secuencia aleatoria de entrenamiento la cual será la representación de los datos que se desean enviar y con ella se generará el *codebook*.
2. Se ingresan valores de entrenamiento para poder realizar la iteración de Lloyd inicial.
3. Mediante la indicación '*lloyds*¹' se calculan una frontera y dos centroides, con lo cual se tiene el primer diccionario de 2 elementos (los centroides calculados).
4. A continuación, se debe calcular la distorsión generada, la cual se conocerá mediante la siguiente expresión,

$$D = \frac{1}{N} \sum_{n=1}^N (x_n - Q(x_n))^2 \quad (5.1)$$

para la cual es necesario tener la nueva secuencia cuantizada, y utilizando la instrucción '*quantiz*²' que genera una cuantización uniforme. Finalmente se calcula la distorsión.

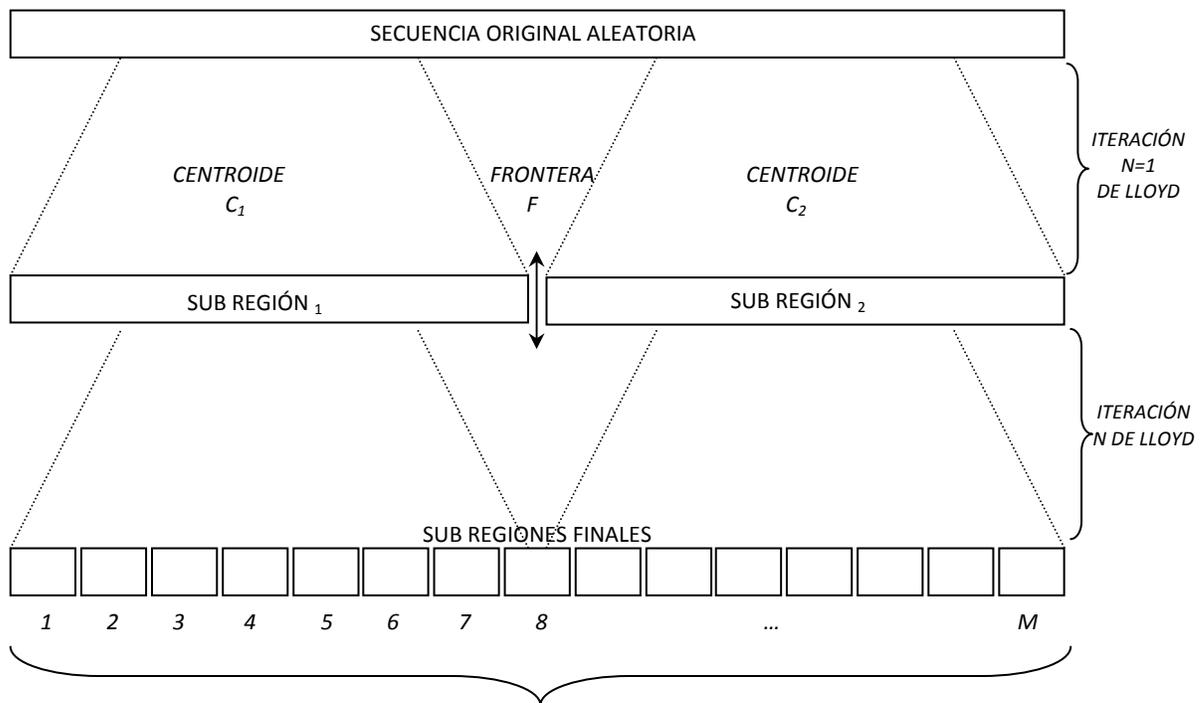
5. A partir de este punto, se entra a un ciclo repetitivo, en el cual el primer paso debe ser calcular los límites mínimo y máximo de cada una de las regiones, para lo cual se utilizan las fronteras que se generaron anteriormente.
6. Después, para cada una de las regiones se realiza el mismo procedimiento aplicando la iteración de '*lloyds*' para generar por cada una de las regiones originales, dos nuevas regiones con dos centroides y una nueva frontera.

¹ En el caso exclusivo de la programación decidí utilizar esta función solo por comodidad, ya que ésta realiza de manera automática el proceso de Iteración de Lloyd, la cual fue explicada en el capítulo correspondiente.

² La función *quantiz* se utilizó con la finalidad de comparar el valor de la distorsión generada por la función *lloyds* y la generada por una cuantización uniforme de la señal de entrada.

7. Una vez obtenidas estas dos nuevas regiones con sus centroides, se debe calcular el valor de la distorsión, la cual se comparará con un valor patrón de distorsión D_i establecido al inicio del ciclo.
8. Si la distorsión D calculada en cada región es menor a D_i , los valores de los centroides c_i deben guardarse en una nueva variable. Si la D calculada es mayor a D_i se debe volver a realizar el ciclo hasta que se tenga un valor de distorsión mínimo.
9. Finalmente, todos los centroides c_i guardados en la nueva variable, será el *codebook* buscado.

Ahora se presenta un diagrama que explica con más detalle el procedimiento que se espera realizar a partir de este algoritmo:



CADA UNO DE LOS CENTROIDES DE CADA REGIÓN FINAL FORMARÁ PARTE DEL CODEBOOK

FIG. 5.1 DIAGRAMA REPRESENTATIVO DEL ALGORITMO LGB

A continuación se presenta una parte del código utilizado para obtener los resultados solo para que se observe de forma detallada la aplicación del algoritmo antes mencionado.

Programa:

```

% CUANTIZADOR ESCALAR BASADO EN EL ALGORITMO LGB
%
% Secuencia de 10000 números aleatorios entre 0 y 100
    
```

```

% con una distribución uniforme:
%
a=100*rand(1,10000);
%
% Se realiza una función que tendrá como parámetro de
% entrada la secuencia aleatoria 'a' y como salida
% los valores correspondientes a dos nuevos sub-
% conjuntos y un valor de distorsión:
%
function [minimo,maximo,D]=div(a)
%
% Se establece un valor de distorsión inicial para
% realizar las comparaciones:
%
Di=10^-3;
co=[1,2];
minimo=[];
maximo=[];
%
% Se aplica la función 'lloyds' al conjunto original,
% generando para esta un valor de f (frontera), y dos
% de c (centroides):
%
[f,c,D]=lloyds(a,co);
%
% Se calcula la distorsión:
%
[inds,sq] = quantiz(minimo,f,c);
N=length(sq)
resta= minimo'-sq;
cuad=resta.^2;
D=mean(cuad)
%
% Se utiliza un ciclo 'for' para separar la secuencia
% original en dos nuevas regiones basándose en el
% valor de la frontera generada anteriormente:
%
for i=1:length(a)-1
    if a(i)<=f
        k=length(minimo);
        minimo(k+1)=a(i);
    else
        k=length(maximo);
        maximo(k+1)=a(i);
    end
end
minimo;
maximo;
%
% Se utiliza la instrucción 'if' para hacer que la
% función declarada se llame a sí misma y se comience

```

```

% así el ciclo repetitivo:
%
if D>Di
    div(minimo);
    div(maximo);
elseif D<=Di
% Se imprimen los valores de 'c' que cumplen con la
% condición:
    CODE=c;
end
% Se imprime el codebook final:
CODE

```

Inicialmente se generó una secuencia de 10 000 números aleatorios en el intervalo $I[0,100]$, con la finalidad de obtener un *codebook* de gran calidad que permita asignar un valor a cada una de las muestras de entrada y con la menor distorsión posible, facilitando de esta forma su recepción y evitando la existencia de errores en el receptor.

```

% Secuencia aleatoria de 10000 números en el intervalo
% de 0 a 100:
x=100*rand(1,10000);
x1=rand(1,100);
subplot(2,1,1); plot(x)
subplot(2,1,2); plot(x1)

```

A continuación se muestran de forma gráfica las secuencias generadas, la primera en el intervalo entre $[0,100]$, y la segunda entre $[0,1]$. Esto se realizó con la finalidad de observar de forma más detallada las muestras generadas.

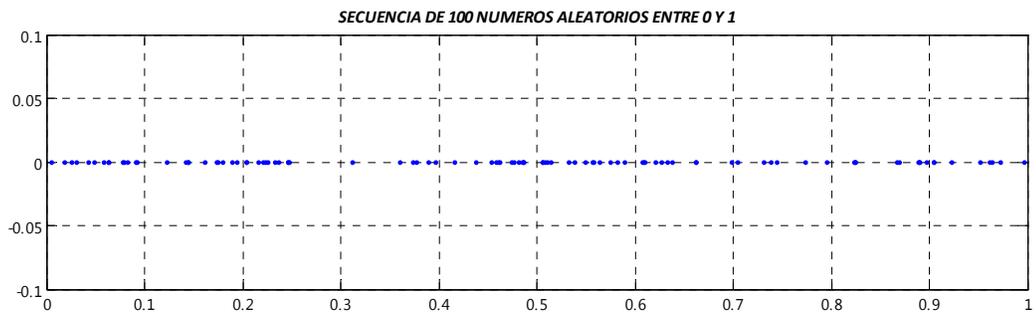
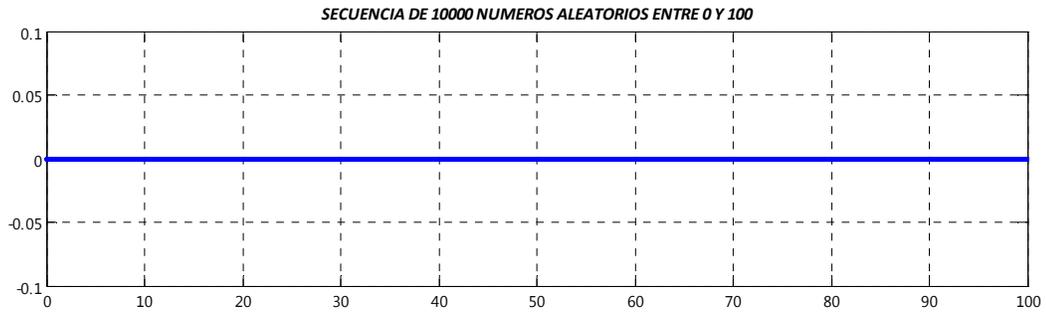


FIG. 5.2.A) SECUENCIA ALEATORIA DE 10000 NÚMEROS EN EL INTERVALO [0,100]

5.2.B) SECUENCIA ALEATORIA DE 100 NÚMEROS EN EL INTERVALO [0,1]

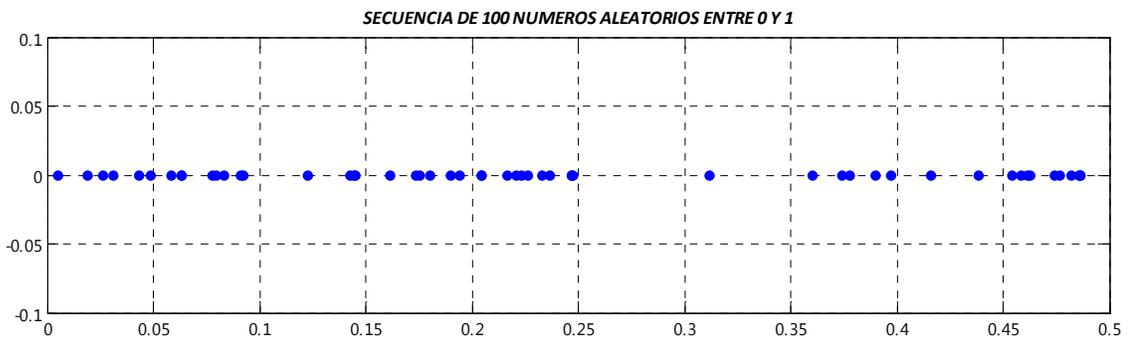
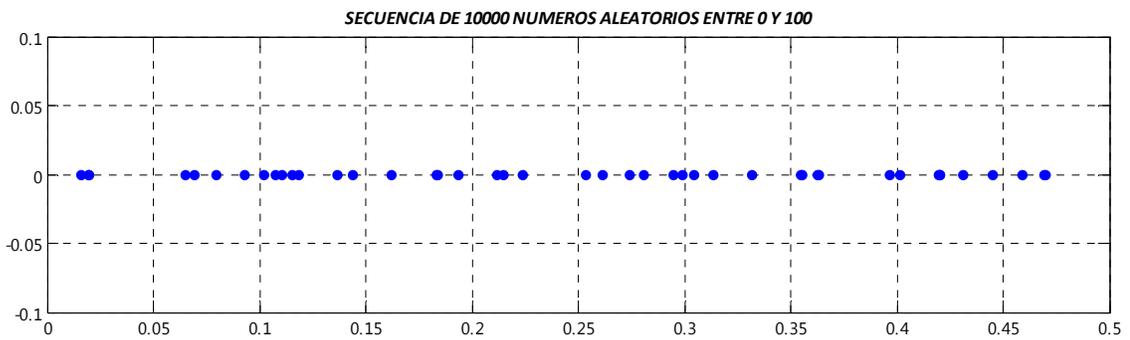


FIG. 5.3.A) SECUENCIA ALEATORIA DE 10000 NÚMEROS EN EL INTERVALO [0,100]

5.3.B) SECUENCIA ALEATORIA DE 100 NÚMEROS EN EL INTERVALO [0,1]

Después de haber aplicado el programa anterior para $N=1$, es decir para llevar a cabo únicamente la primera iteración de Lloyd, se pudieron obtener dos nuevos subconjuntos generados a partir de la secuencia aleatoria original, así como el valor que representa la frontera entre estas nuevas secuencia, y el valor medio (denominado centroide) de cada una de estas regiones.

```
% Valores para f y c iniciales:
f =
    49.7010
c =
    24.6902
    74.8592
% Gráfica de la secuencia con f y c:
x=100*rand(1,10000);
y=0*rand(2,10000);
xl=rand(1,100);
yl=0*rand(1,100);
    plot(x,y)
hold on
    plot(f,)
    plot(c,)
hold off
```

En la siguiente gráfica se muestran los valores de los centroides y de la frontera en comparación a la secuencia original:

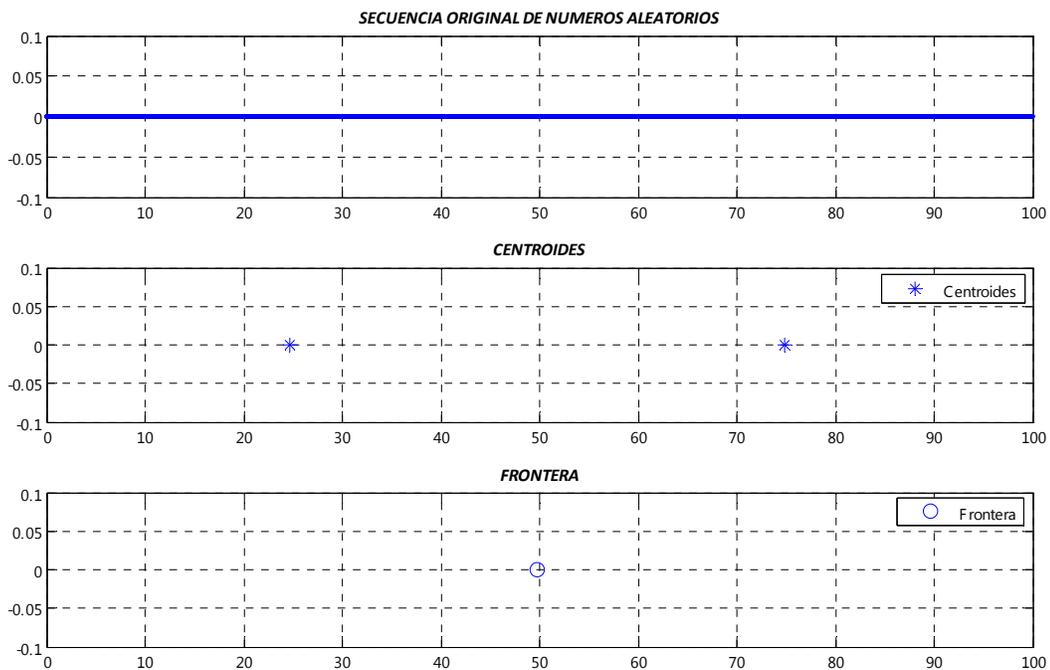


FIG. 5.4.A) SECUENCIA ORIGINAL, 5.4.B) CENTROIDES INICIALES, 5.4.C) FRONTERA INICIAL

Una vez aplicada la función para la generación del *codebook*, se obtuvo el resultado final para una distorsión de 10^{-3} que se muestra en la siguiente gráfica:

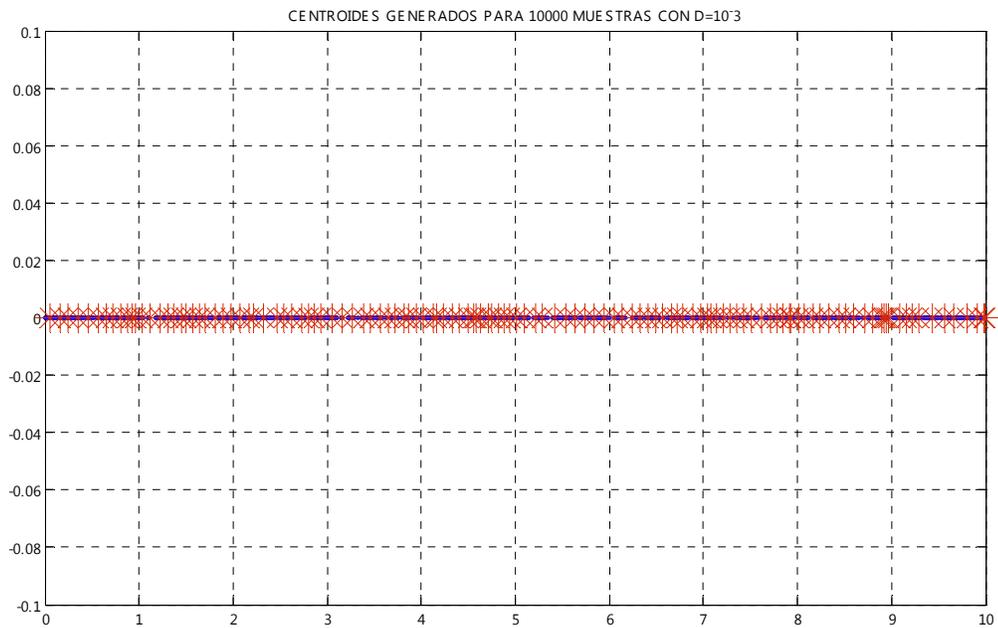


FIG. 5.5.1 REPRESENTACIÓN GRÁFICA DEL CODEBOOK FINAL

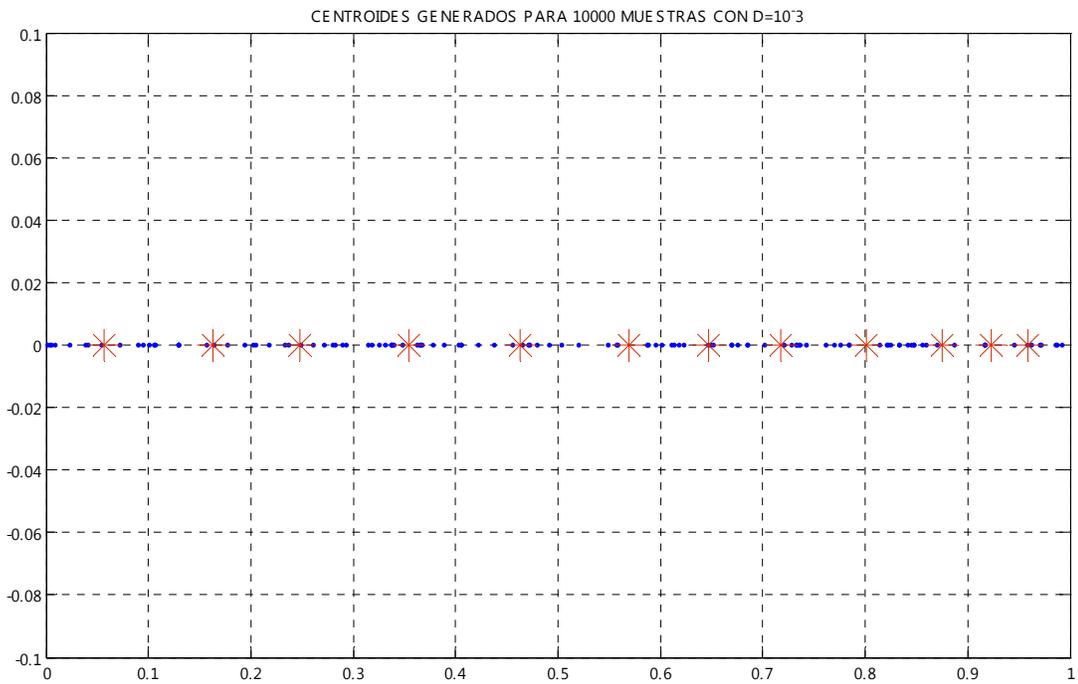


FIG. 5.5.2 CENTROIDES EN EL INTERVALO [0,1]

La tabla correspondiente al *codebook* generado a partir de las condiciones dadas se presenta al final en el Anexo 1 y 2, donde se podrá apreciar de forma detallada los 1 212 elementos que lo conforman, así como la representación gráfica en varios intervalos. Además, para fines comparativos también se presentan otras tablas generadas a partir de la modificación de las condiciones iniciales, como el valor de distorsión D , el número de elementos iniciales y el intervalo de trabajo.

Observaciones:

Como se puede observar con este diccionario, se consiguió reducir 10 000 números aleatorios a solo 1 212 elementos mediante la aplicación del programa utilizando y basándose en el algoritmo LGB y en el *splitting*.

Los resultados que se obtuvieron a partir de esta simulación corresponden a una distribución de probabilidad uniforme para los números aleatorios de entrada.

El código propuesto permite codificar cualquier palabra que ingrese al transmisor con gran eficacia, ya que esta creado a partir de una distorsión de 10^{-3} , es decir, casi una distorsión nula. Esto indica que cualquier palabra que entre será fácilmente asignada al centroide correspondiente de una manera sencilla y evitando cometer errores.

La generación de este diccionario fue un éxito, ya que permitió mostrar de forma analítica y gráfica los métodos propuestos al inicio del presente capítulo.

Con este diccionario de códigos, se reduce el valor de la distorsión a deseos del usuario, permitiendo con esto que los errores que puedan ocurrir en la señal se deban únicamente a las perturbaciones que se puedan dar en el medio de transmisión.

Es posible darse cuenta que la aplicación de este algoritmo es de suma facilidad, y esto depende a que este se aplica de forma única a una secuencia de números definidos conjuntamente en uno de los ejes, en este caso en el eje x (o simplemente recta numérica). Esto a su vez da la pauta para poder crear diccionarios muy variados, como los mostrados en los anexos 1 y 2 ya que con un valor mínimo de distorsión, la mayor longitud obtenida fue de 1 212 elementos, haciendo posible mostrarlo en su totalidad en el presente reporte.

Sin embargo, como se verá a continuación, para la cuantización vectorial es muy diferente, ya que al ocupar vectores que ahora estén alojados en un plano y que a cada uno de los elementos se les asigne coordenadas, nos daremos cuenta que el proceso se complica y el número de elementos finales que conforman el diccionario es del orden de x^2 en comparación con la cuantización escalar,

lo que obviamente limitará las intenciones de realizar iteraciones en conjuntos con un gran rango, ya que esto generará un *codebook* por encima de los 10 000 elementos.

5.2 ALGORITMO PARA LA CUANTIZACIÓN VECTORIAL

Algoritmo:

Para el caso de la cuantización vectorial, me basaré en el algoritmo de diseño *COVQ* (*channel optimized vector quantization*), la cual se explicó con anterioridad en otro capítulo. Lo que se busca al utilizar este medio de diseño, es poder crear a partir de una serie de vectores de dos dimensiones, un *codebook* que a cada una de las palabras de entrada le asigne automáticamente dos valores (de forma similar a unas coordenadas), es decir, que a cada palabra de entrada se le asigne un 'vector' que lo represente.

Se espera que el *codebook* generado sea similar al de la cuantización escalar, ya que se utilizará nuevamente *splitting*, pero en este caso se realizará en un espacio de dos dimensiones. El algoritmo a seguir será el siguiente:

1. Se debe generar una serie de vectores aleatorios de dos dimensiones (x,y) de entrenamiento como base del *codebook* que se espera.
2. Mediante la indicación 'lloyds' se calculan dos centroides en el plano dividiéndolo en dos nuevas regiones.

3. A continuación, se debe calcular la distorsión generada para cada uno de los puntos de la secuencia inicial, la cual se conocerá mediante la siguiente expresión,

$$D = \frac{1}{N} \sum_{n=1}^N (x_n - Q(x_n))^2 \quad (5.2)$$

4. Para cada una de las regiones generadas se deberá aplicar el mismo procedimiento para ir obteniendo en forma repetitiva los centroides $C_{x,y}$. La asignación de los valores a cada una de las regiones dependerá de la distancia que separe la muestra del centroide.
5. Se repite el procedimiento hasta que no se cumpla la condición (que dependerá del valor de la distorsión obtenida en cada iteración).
6. Finalmente, todos los centroides $c_{x,y}$ serán el *codebook* buscado.

A continuación se presenta un diagrama que explica con más detalle el procedimiento que se espera realizar a partir de este algoritmo:

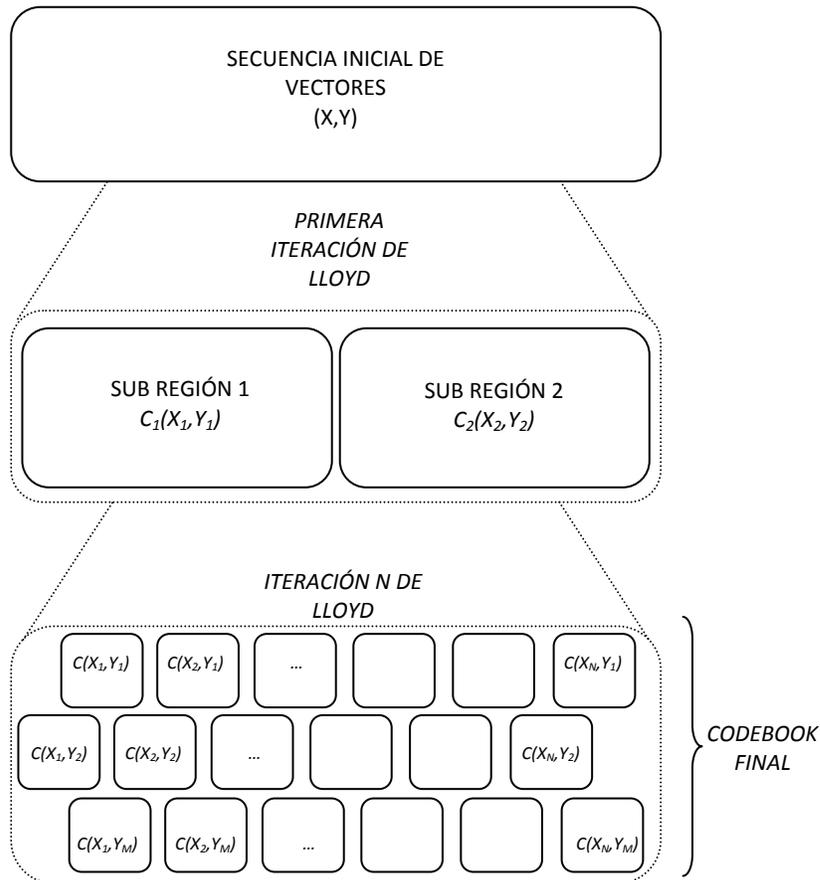


FIG. 5.6 REPRESENTACIÓN DEL ALGORITMO COVQ

Y aplicando el siguiente código, mediante el cual se imprimirán parejas ordenadas correspondientes a los centroides c_{xy} , se logrará generar un *codebook* que represente el algoritmo COVQ:

Programa:

```

% CUANTIZADOR VECTORIAL BASADO EN EL ALGORITMO COVQ
%
% Secuencia de 10000 vectores aleatorios entre 0 y 5
% con una distribución uniforme:
%
a=5*rand(2,10000);
%
function [minimo,maximo,D]=div(a)
%
Di=10^-3;
co=[1,2];
minimo=[];
maximo=[];
%
[f,c,D]=lloyds(a,co);
%
%
for i=1:length(a)-1
    if a(i)<=f
        k=length(minimo);
        minimo(k+1)=a(i);
    else
        k=length(maximo);
        maximo(k+1)=a(i);
    end
end
minimo;
maximo;
%
%
if D>Di
    div(minimo);
    div(maximo);
elseif D<=Di
    CODE=c;
end
% Se imprime el codebook final:
CODE

```

En las siguientes gráficas se muestra la generación de vectores de 2 dimensiones en los planos X y Y . En la figura 5.7 se muestran los 10 000 vectores aleatorios en el rango de 0 a 1 para cada uno de los ejes, mientras que en la figura 5.8 se tiene un acercamiento de la misma secuencia para observar más detalladamente la localización de los puntos:

```

a=rand(500,1);
b=rand(500,1);
co=[1 2];
plot(a,b)
hold on
[f1,c1]=lloyd(a,co);
[f2,c2]=lloyd(b,co);
    plot(f1,f2)
    plot(c1,c2)
hold off

```

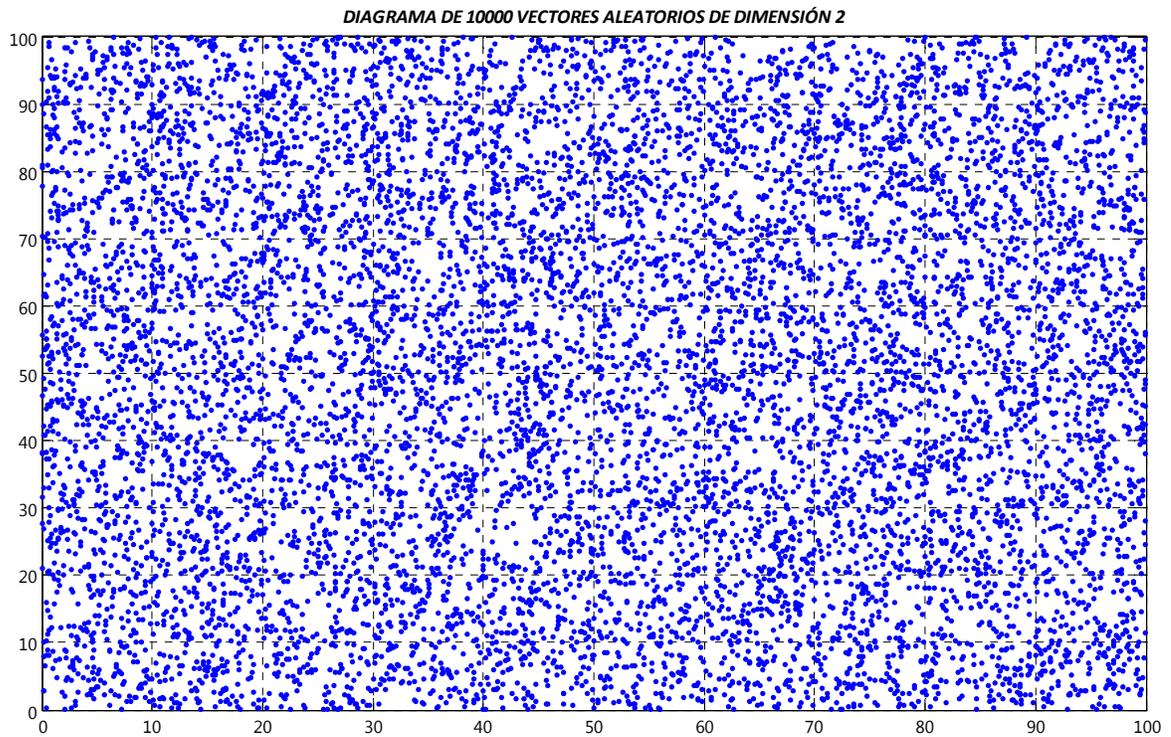


FIG. 5.7 VECTORES ALEATORIOS EN DIMENSIÓN 2

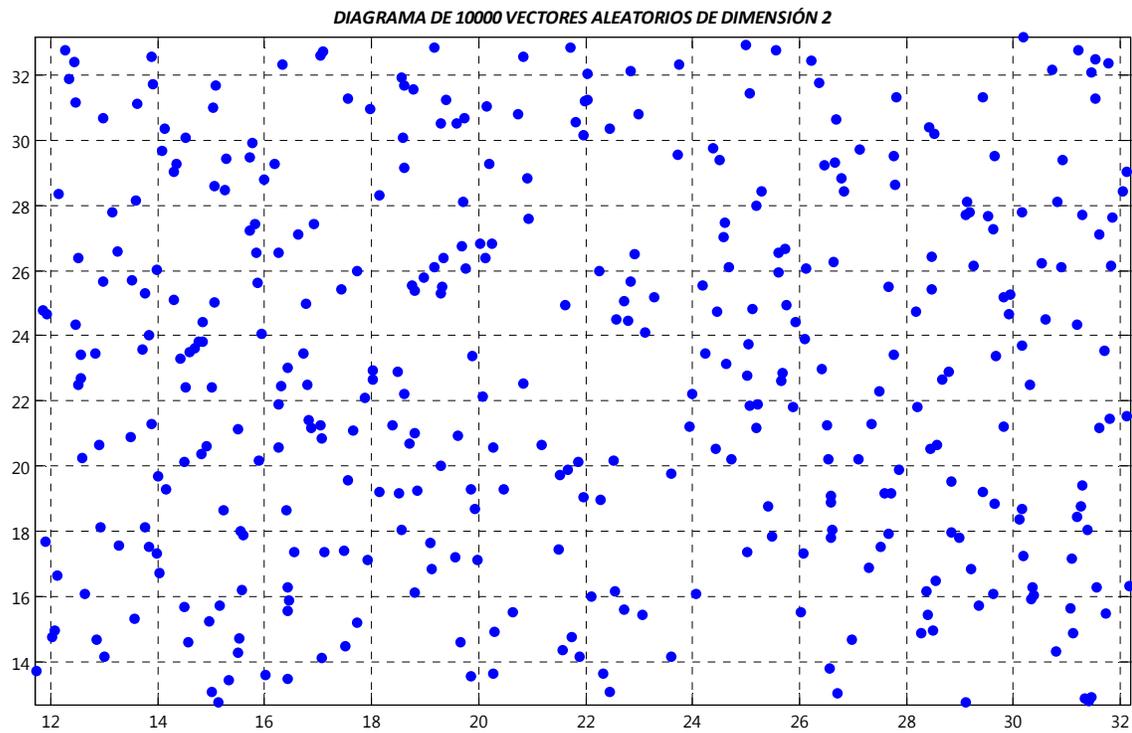


FIG. 5.8 VECTORES ALEATORIOS EN DIMENSIÓN 2

Inicialmente, al contar con los vectores en los ejes X y Y , se optó por conocer primeramente la localización del centroide inicial C_i , con la finalidad de obtener el promedio general de toda la secuencia. Esto se muestra en la siguiente gráfica:

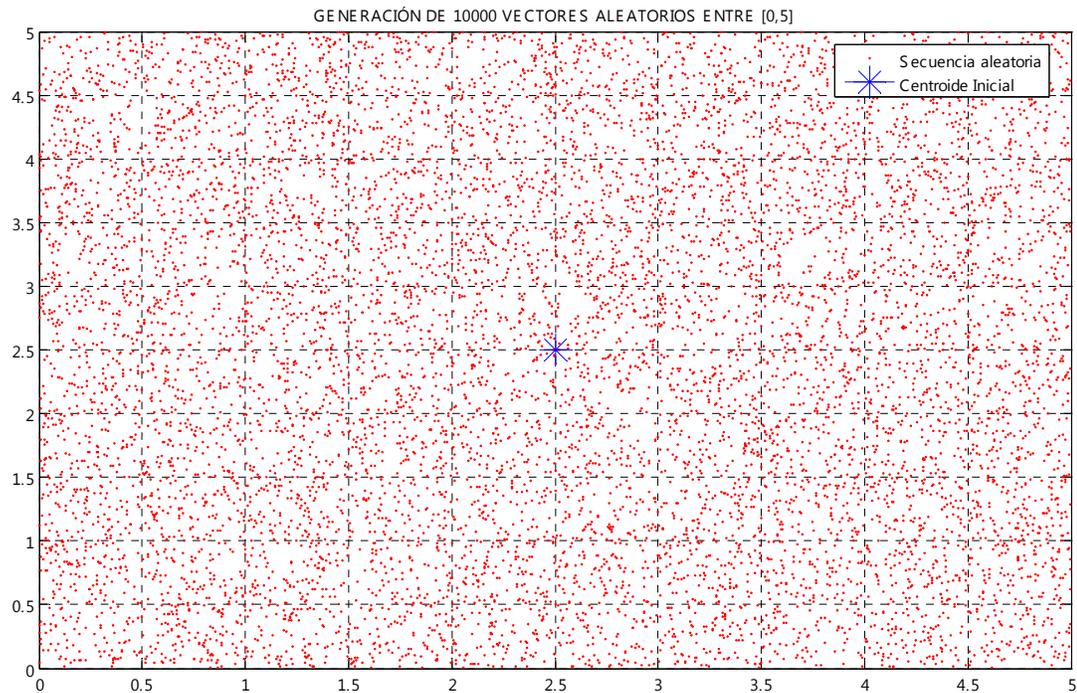


FIG. 5.9 VECTORES ALEATORIOS EN DIMENSIÓN 2 CON EL CENTROIDE INICIAL

Como se indica en el código utilizado, el *codebook* que se generará para describir el algoritmo COVQ tendrá las siguientes condiciones iniciales:

1. Se utilizarán 10 000 vectores iniciales con una distribución uniforme, esto con la finalidad de que los centroides obtenidos tengan mayor fidelidad al poder representar a un mayor cantidad de puntos en el plano.
2. La distorsión que se manejará será la misma que para el algoritmo LGB mostrado anteriormente, es decir, un valor de 10^{-3} , esto para crear una lista de datos con la menor probabilidad de error al momento de asignarle valores a la señal de entrada.
3. Para el caso de la cuantización escalar, se tiene que el número de elementos que forman parte del *codebook* es prácticamente el cuadrado del obtenido para la escalar, es decir, que si en este caso se manejará el mismo intervalo utilizado en el LGB en donde se obtuvieron 1 200 elementos, el *codebook* generado para este caso rebasaría los el millón de elementos, lo cual haría imposible plasmar dichos valores en el presente trabajo. Por tales razones se optó por manejar intervalos pequeños que a su vez fueran representativos y mostraran el resultado en una forma 'manejable' para el lector. Los intervalos que se usarán variarán entre [0,5] y [0,1].

En la siguiente gráfica se puede observar el resultado después de la primera iteración de Lloyd, la cual generó los dos primeros centroides dividiendo la región inicial en dos nuevos subconjuntos:

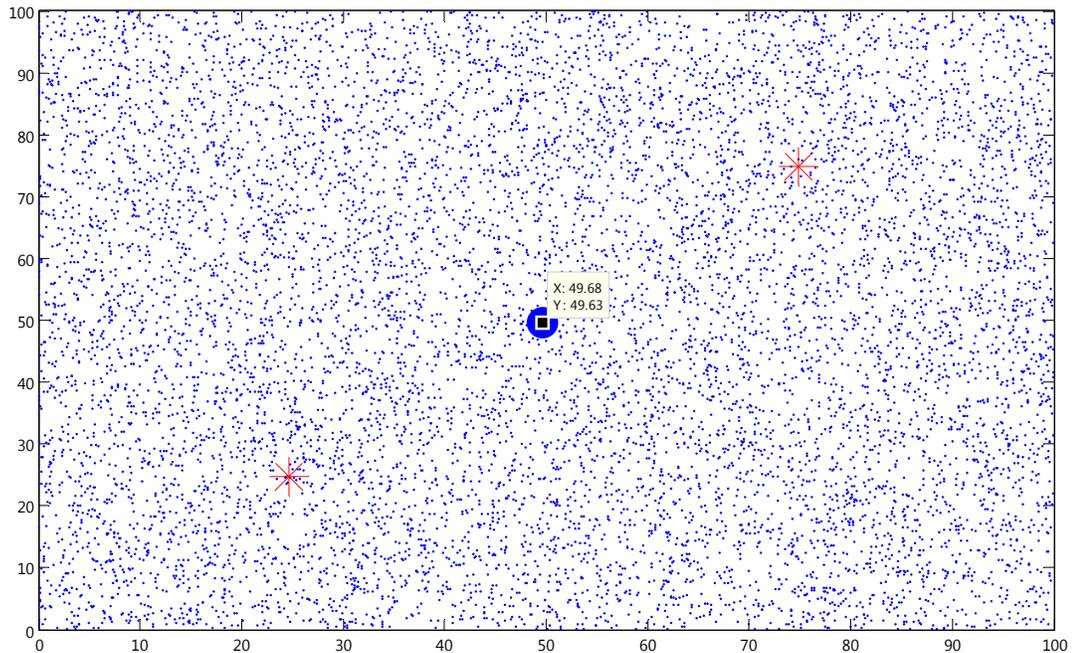


FIG. 5.10 RESULTADO DE LA PRIMER ITERACIÓN DE LLOYD

Como se puede apreciar, a partir del centroide inicial, se crean dos nuevas regiones, las cuales a su vez cuentan con su propio centroide, lo que indica que en ese momento existen dos regiones en el plano, y cada una contiene los elementos más cercanos. En este método es posible apreciar la aplicación del teorema del '*vecino más cercano*', ya que cada pareja ordenada se agregará al conjunto cuya distancia sea mínima en comparación en el otro. Este método iterativo se repetirá nuevamente, generando regiones con la siguiente periodicidad $N=1, 2, 4, 8, 16\dots$

Después de la aplicación del programa creado, se generó de manera gráfica el *codebook* final, mostrado a continuación:

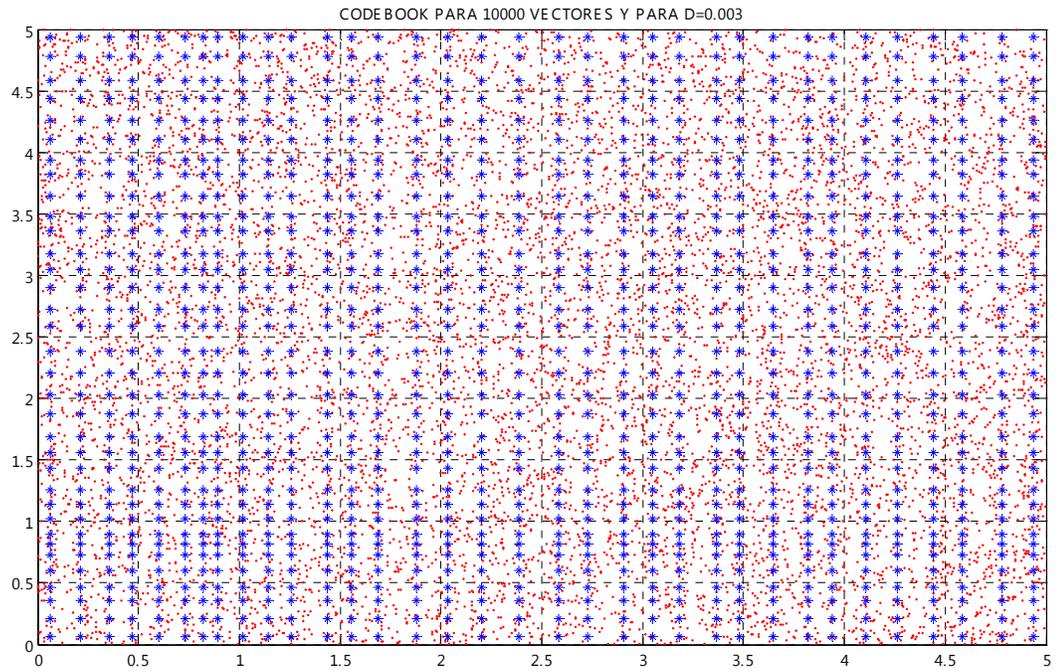


FIG. 5.11 REPRESENTACIÓN GRÁFICA DEL ALGORITMO COVQ

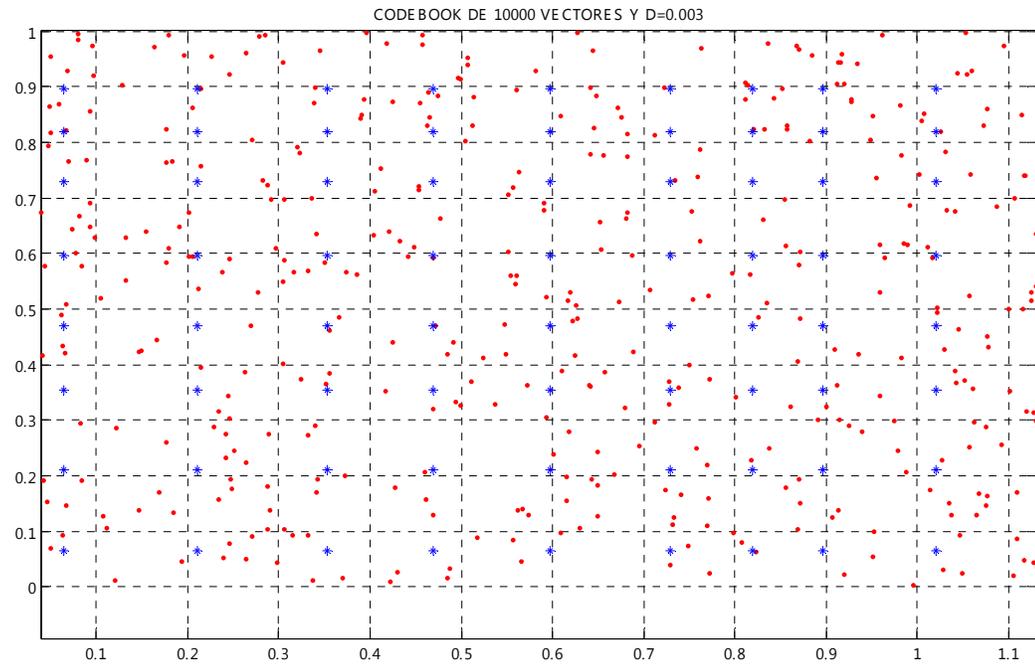


FIG. 5.12 ELEMENTOS GENERADOS EN EL INTERVALO $[0,1]$

Observaciones:

Como se puede observar en ambas gráficas, se generan una gran cantidad de centroides que pueden representar a los 10 000 vectores de entrada. En la figura 5.11 se puede observar el diccionario completo, el cual resulta estar formado de 1 156 parejas ordenadas C_{xy} , es decir, para cada muestra de entrada al transmisor, se le asignará una pareja (x,y) que la representará al momento de enviarla al receptor.

Además en esta misma gráfica se puede ver como los centroides forman una especie de 'reja' la cual no es totalmente simétrica, esto debido a que se generó a partir de vectores cuya posición fue aleatoria, justo como se observa en la figura 5.12 donde se tiene un acercamiento al intervalo de $[0,1]$. En ella nos podemos percatar de que los centroides (puntos azules) se generaron únicamente en las zonas de mayor concentración de muestras iniciales (puntos rojos).

Cabe mencionar que estas gráficas corresponden al caso más óptimo obtenido, ya que con los más de mil elementos (en parejas ordenadas) y en un intervalo de $[0,5]$ con $D=0.003$ es posible representar cualquier muestra de la señal de entrada.

Para observar los resultados de una manera más detallada, la tabla correspondiente a estas gráficas se localizan en el Anexo 5, además de otras tablas para fines comparativas obtenidas a partir de la variación del intervalo de trabajo y el valor nominal de D . Finalmente, en el anexo 6 se muestran las representaciones gráficas correspondientes a las tablas mencionadas.

5.3 ESQUEMA COMPARATIVO

5.3.1 CUANTIZACIÓN ESCALAR

De acuerdo a los resultados obtenidos en la simulación escalar basada en el algoritmo LGB se pueden llevar a cabo una serie de comparación mediante la cual será posible determinar cual de todos los *codebooks* calculados resultará ser el código más óptimo que permita el intercambio adecuado de información entre un transmisor y un receptor. A continuación se llevará a cabo una comparación basándose en los siguientes aspectos:

1. Distorsión,

Como se mencionó en los capítulos anteriores, la distorsión es una medida que indica la diferencia que existe entre una señal que ha sufrido algún tratado (ya sea que se hayan modificado algunas de sus características por necesidad o por que hayan sido creadas por alteraciones en el medio), y la señal original.

En el caso de las simulaciones realizadas para el algoritmo LGB correspondiente a la cuantización escalar, fue posible determinar una serie de resultados a partir de la variación de este parámetro de forma manual, lo que permitió crear un *codebook* único para cada valor de distorsión fijado.

2. Numero de elementos finales de codebook,

Otro punto de comparación de los resultados obtenidos radica en el número de elementos finales (o códigos de palabra) que forman el *codebook* o diccionario. Este aspecto es de suma importancia, ya que además de tener en cuenta que este parámetro es dependiente del valor de distorsión asignado, en una aplicación real debe tenerse en cuenta la capacidad de memoria y la velocidad con que se cuente en el sistema, ya que al tenerse un diccionario muy grande, puede que este supere la capacidad de memoria que esta reservada para esta tarea, tanto en el transmisor como en el receptor, ya que en ambos dispositivos, la tarea va más allá de solo almacenar el diccionario.

En ambos elementos, el sistema debe tener la capacidad de almacenar el *codebook*, pero al mismo tiempo, debe contar con una gran velocidad para poder asignarle a cada una de las muestras de entrada un valor fijo, y enviarlo. Lo mismo sucede en la recepción. Por este motivo, el número de elementos finales debe estar acoplado a las características iniciales del sistema. En este caso, solo se hará la comparación para fines prácticos, ya que los diccionarios propuestos están diseñados con la finalidad de mostrar la variabilidad de resultados al modificar las condiciones iniciales de un sistema cualquiera, es decir, no están especificadas las condiciones de un sistema único.

3. Numero de iteraciones que lograron ese codebook,

Este también resulta ser un parámetro importante, y que al igual que el anterior, depende en gran parte de la distorsión. Su importancia radica en que también depende de la capacidad del sistema, ya que el sistema puede no soportar el programa, y por lo tanto no permitir que el resulta final se conozca.

Como se mencionó, ninguno de estos parámetros son independientes, sino que son dependientes uno con otro, siendo aquel código que posea las condiciones más óptimas en todos los casos el diccionario mejor desarrollado.

En la siguiente tabla se muestran de una manera concentrada los resultados obtenidos después de las simulaciones:

Distorsión	Numero de elementos finales	Numero de elementos iniciales	Intervalo [0,]
10 ⁻³	1212	10000	100
10 ⁻³	136	1000	10
10 ⁻³	16	10000	1
10 ⁻³	16	1000	1
10 ⁻³	14	100	1
0.5	64	10000	100
0.5	6	1000	10
0.5	2	10000	1
0.5	2	1000	1
0.5	2	100	1
1	32	10000	100
1	4	1000	10
1	2	10000	1
1	2	1000	1
1	2	100	1
10	16	10000	100
10	2	1000	10
50	8	10000	100

TABLA 5.1 RESULTADOS FINALES DEL ALGORITMO LGB

A continuación se muestra de forma gráfica la relación entre la distorsión y el número de elementos que formarán el *codebook* buscado:

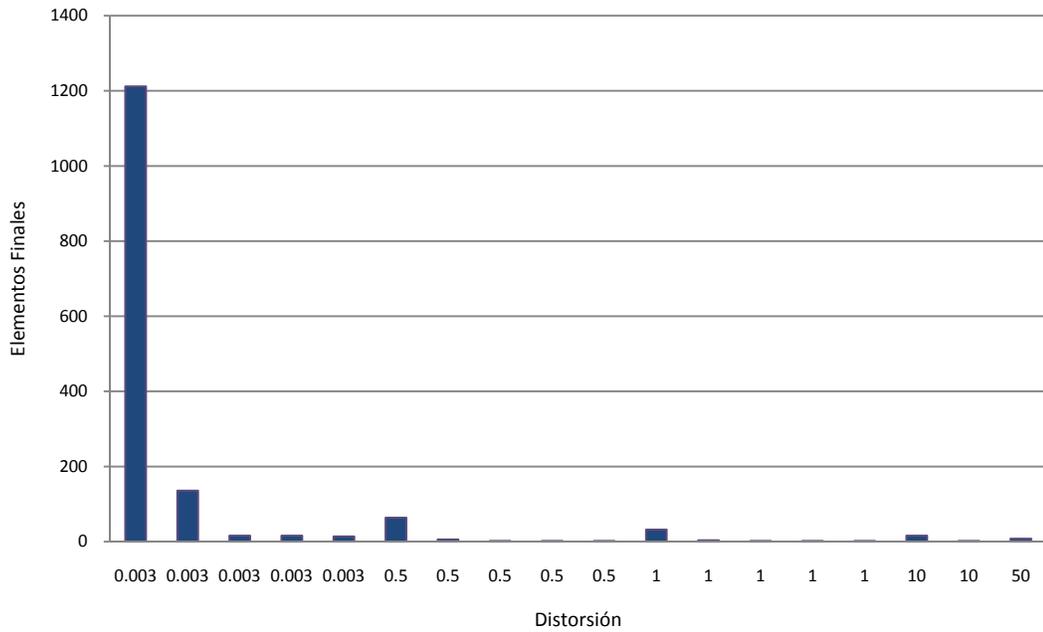


FIG. 5.13 RELACIÓN DISTORSIÓN VS ELEMENTOS FINALES

Sin embargo, para poder apreciar de una forma más detallada, se verán las gráficas por separada a partir del valor de distorsión. Se comenzará tomando como patrón $D=0.003$:

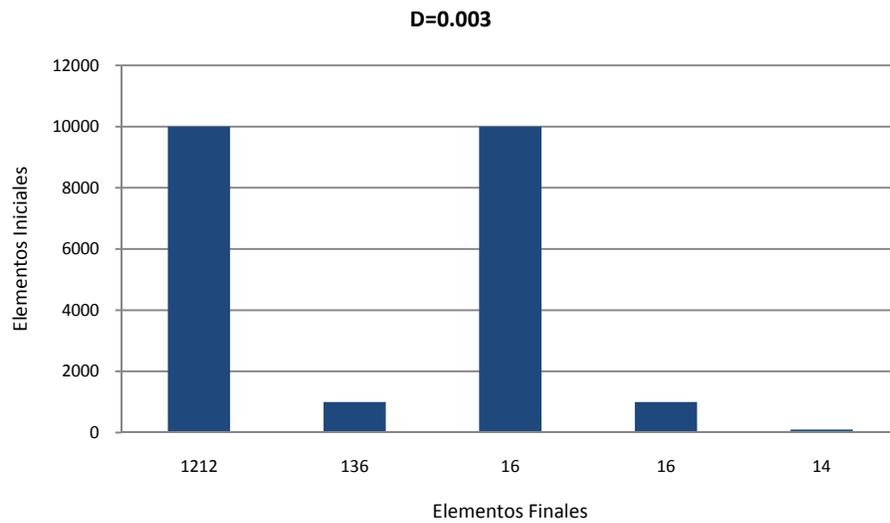


FIG. 5.14 RELACIÓN ELEMENTOS FINALES VS INICIALES PARA $D=0.003$

Como podemos observar, para una distorsión de 0.003, el número de elementos que forman el *codebook* es muy grande, sobre todo cuando el rango en que se generan los números aleatorios aumenta.

Esto se debe a que mientras en valor de distorsión disminuye, el rango de 'efectividad' de cada una de las muestras disminuye, provocando a la vez que su número aumente. Es decir, el aumento de la distorsión generará un diccionario de una longitud menor y viceversa.

En la gráfica se observa que el *codebook* que presenta mayor cantidad de elementos es aquel que se propuso realizar en un intervalo mayor, de [0,100], esto con la finalidad de darles a las muestras de entrada la capacidad de encontrar un punto que las represente con mayor eficacia, y al mismo tiempo que disminuya su probabilidad de error.

Sin embargo, como se mencionó al inicio de esta comparación de resultados, para la aplicación de este código se debe contar con un sistema que cuente con una memoria que soporte la cantidad de datos y que a su vez tenga una velocidad considerable para localizar el punto que represente a la señal de entrada.

A continuación se presenta la gráfica que corresponde a una distorsión $D=0.5$:

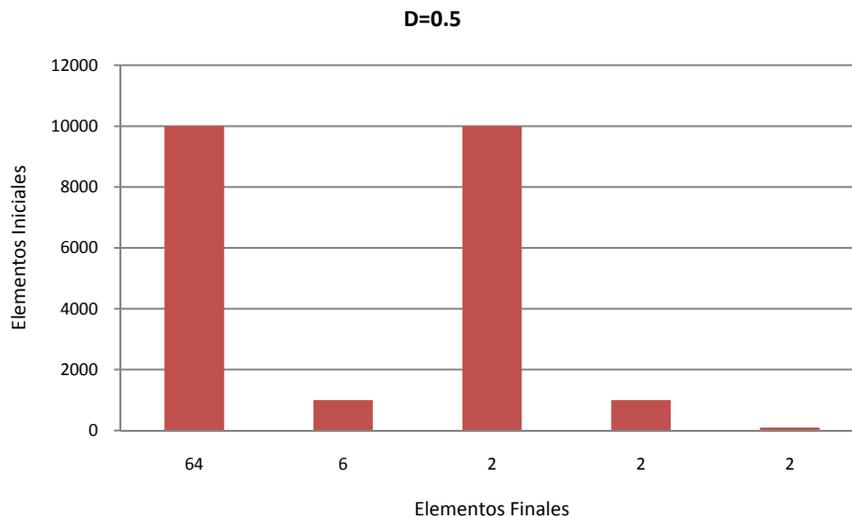


FIG. 5.15 RELACIÓN ELEMENTOS FINALES VS INICIALES PARA $D=0.5$

Como se puede observar, el número de elementos del diccionario disminuye de forma considerable en comparación con la anterior, esto se debe a que este valor de D cuenta con un rango mayor para representar las muestra, pero al mismo tiempo disminuye la calidad con que lo hace, es decir, aumenta la probabilidad de error. Esto se repite para la siguiente gráfica en donde se muestran los resultados para $D=1$:

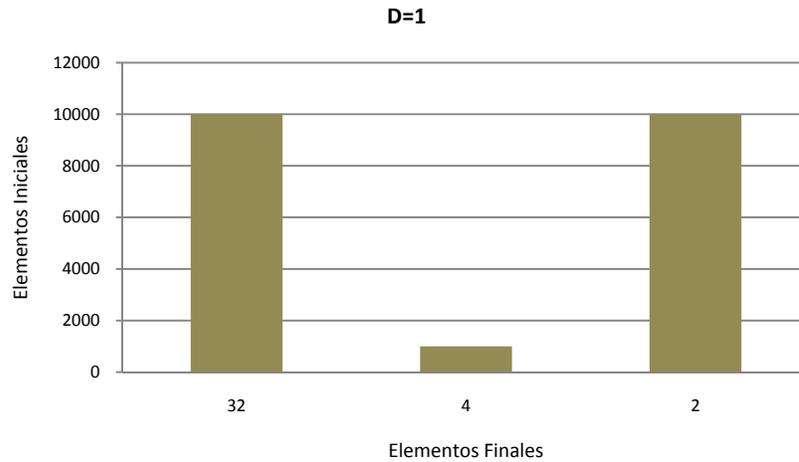


FIG. 5.16 RELACIÓN ELEMENTOS FINALES VS INICIALES PARA D=1

De esta comparación se puede concluir que mientras menor sea la distorsión, el número de elementos generados aumenta, por lo que la probabilidad de error al tratar de recuperar la señal original en el receptor es menor, ya que los errores ahora solo dependen de las perturbaciones que se generen en el medio de transmisión. Por lo que es de esperarse que para los casos de $D=10$ y $D=50$ el número de elementos será demasiado pequeño (dividirá la secuencia en 2 o 4 muestras) teniendo con esto que la señal de entrada quede, justo en el momento de la cuantización, con una forma muy diferente a la original, lo que dificulte su correcta recuperación.

A continuación se presentan los resultados pero desde el punto de vista del intervalo inicial en donde se generaron las muestras. Esto con la finalidad de determinar cual es el rango más óptimo en el que la señal de entrada puede cuantizarse de forma correcta:

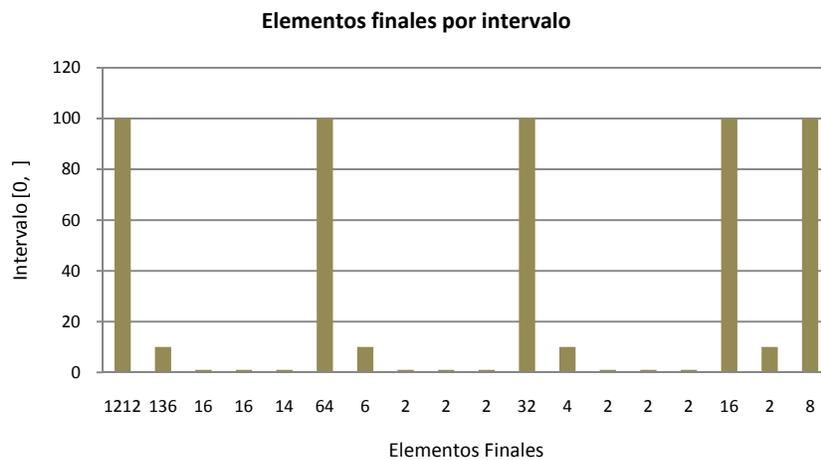


FIG. 5.17 ELEMENTOS FINALES POR INTERVALO

El primer intervalo a examinar comprende números generados entre 0 y 100. Como se puede observar en la siguiente gráfica, y tomando en consideración que el intervalo es muy amplio para poder tomar muestras que representen a la señal de entrada, es posible percatarse que la mayor concentración de estos se da cuando la distorsión es muy pequeña, es decir, que con un amplio rango y una distorsión casi nula, se generan 1212 elementos a partir de 10 000 elementos para cuantizar las señales (esto se observa en la primer barra de la gráfica).

Sin embargo, si el valor de la distorsión se varía, en este caso, si se aumenta, el número de elementos que conformen el diccionario tiende a disminuir de manera drástica, por lo que para este intervalo en concreto, conviene utilizar el *codebook* que se propone al inicio de este capítulo, ya que cualquiera de los demás en este intervalo podría provocar mayor cantidad de errores.

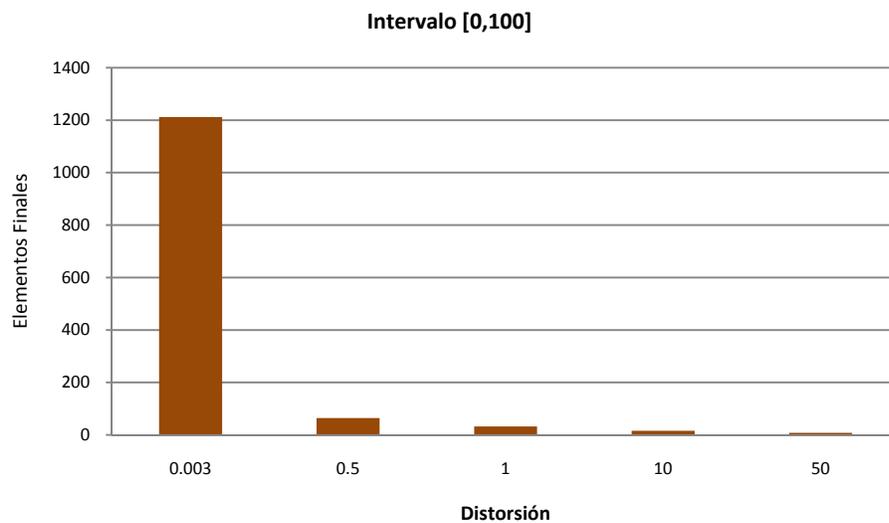


FIG. 5.18 RELACIÓN DISTORSIÓN VS ELEMENTOS FINALES PARA [0,100]

En la siguiente gráfica se muestran los resultados correspondientes para el intervalo de números entre 0 y 10 en donde es posible darse cuenta que ocurre lo mismo que en caso anterior, la mayor concentración de elementos se localiza cuando la distorsión es mínima, y estos van disminuyendo conforme la distorsión es mayor.

Además, como se mencionó, la diferencia entre los dos valores mínimos de D generan un diccionario con una cantidad de elementos que varía de una forma drástica, por lo que es recomendable utilizar el diccionario en con la menos distorsión.

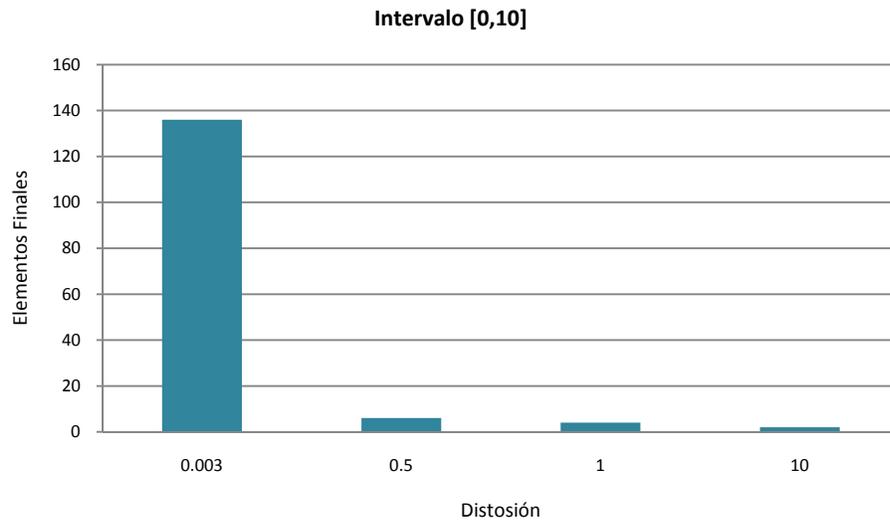


FIG. 5.19 RELACIÓN DISTORSIÓN VS ELEMENTOS FINALES PARA [0,10]

Finalmente, sucede lo mismo para el intervalo de 0 y 1. En este caso el número de elementos para todos los valores de distorsión resultó ser muy pequeña en comparación a todos los demás resultados:

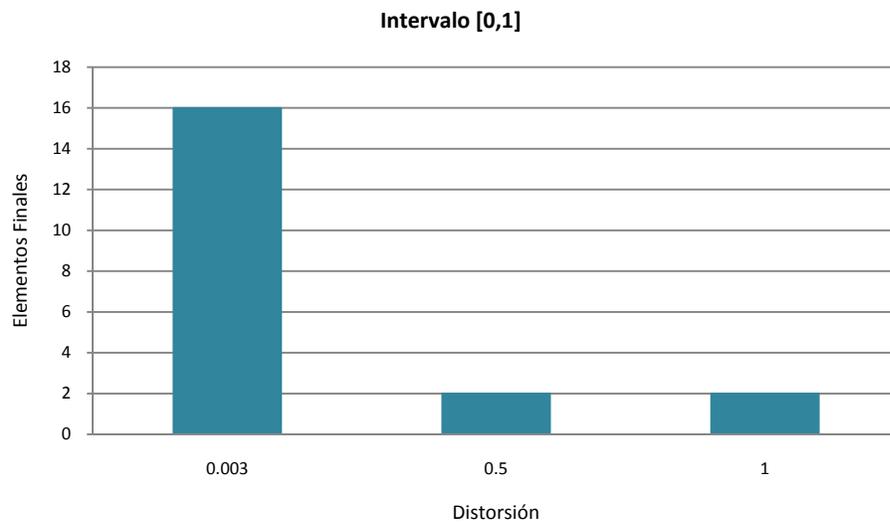


FIG. 5.20 RELACIÓN DISTORSIÓN VS ELEMENTOS FINALES PARA [0,1]

De esta comparación puedo concluir que el *codebook* que cumple con los requerimientos necesarios para llevar a cabo una cuantización escalar más óptima fue el propuesto al inicio, es

decir, el diccionario formado por 1 212 generado a partir de una distorsión $D=0.003$ y en un intervalo de $[0,100]$.

Esto debido a que presenta un cantidad impresionante de elementos que permitirán a cualquier valor de entrada al sistema encontrar un valor que lo represente y que prácticamente mantenga el 100% de fidelidad, evitando que aparezcan errores en el transmisor, y por ende en el receptor, y los errores que surjan se deberán únicamente al medio por el que se envía la señal.

<i>Distorsión</i>	<i>Numero de elementos finales</i>	<i>Numero de elementos iniciales</i>	<i>Intervalo [0,]</i>
10^{-3}	1212	10000	100
10^{-3}	136	1000	10

TABLA 5.2 CARACTERÍSTICAS DE LOS DICCIONARIOS SELECCIONADOS

Otro código que puede resultar útil, es el generado a partir de 1 000 números con $D=0.003$ y en un intervalo de 0 y 10. Esto debido a que el número de elementos finales es una décima parte del total, lo cual en cierta forma garantiza una cuantización buena.

Cabe señalar que no se eligió ningún diccionario con una distorsión mayor, ya que aunque se facilite la búsqueda, no se garantiza una comunicación adecuada y de calidad.

Hasta este punto se ha realizado la comparación para los diccionarios obtenidos al aplicar la función de Lloyd a una secuencia de números aleatorios con una distribución uniforme, pero ¿y si se utiliza una distribución normal?

5.3.2 LGB: DISTRIBUCIÓN NORMAL

En este apartado se generará un diccionario a partir de una secuencia de números aleatorios pero con una distribución normal, es decir, una secuencia donde se localicen en el centro una concentración mayor de muestras (véase capítulo 2).

Para realizar la simulación se utilizará una secuencia con una distribución Gaussiana estándar, es decir, una distribución cuyos parámetros son media $\mu = 0$ y varianza $\sigma = 1$. A continuación se presenta la secuencia generada formada por 1 000 elementos:

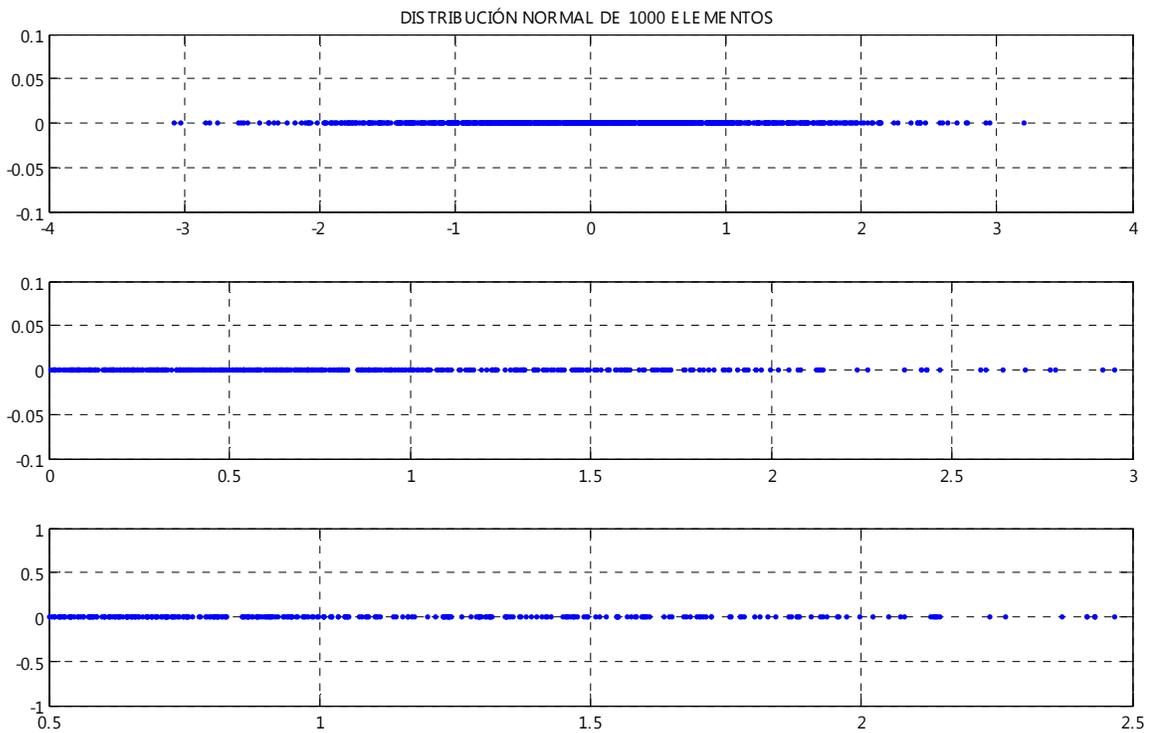


FIG. 5.21 DISTRIBUCIÓN NORMAL DE 1000 ELEMENTOS

Como se puede apreciar en la gráfica anterior, los números generados pueden ser tanto positivos como negativos, además de que la mayor concentración de ellos se localiza cerca del origen, mientras que en los costados los puntos generados son mínimos. Una vez más se aplica la función diseñada para obtener los primeros centroides:

```
a=randn(1000,1);
[f,c]=lloyd(a,co)
f =
    0.0298
c =
   -0.7833    0.8515
```

En la siguiente gráfica se observa la localización de los centroides C en comparación con la secuencia original con distribución normal. Los centroides generados en la primera iteración se localizan cerca del origen, es decir en donde la concentración de números es mayor:

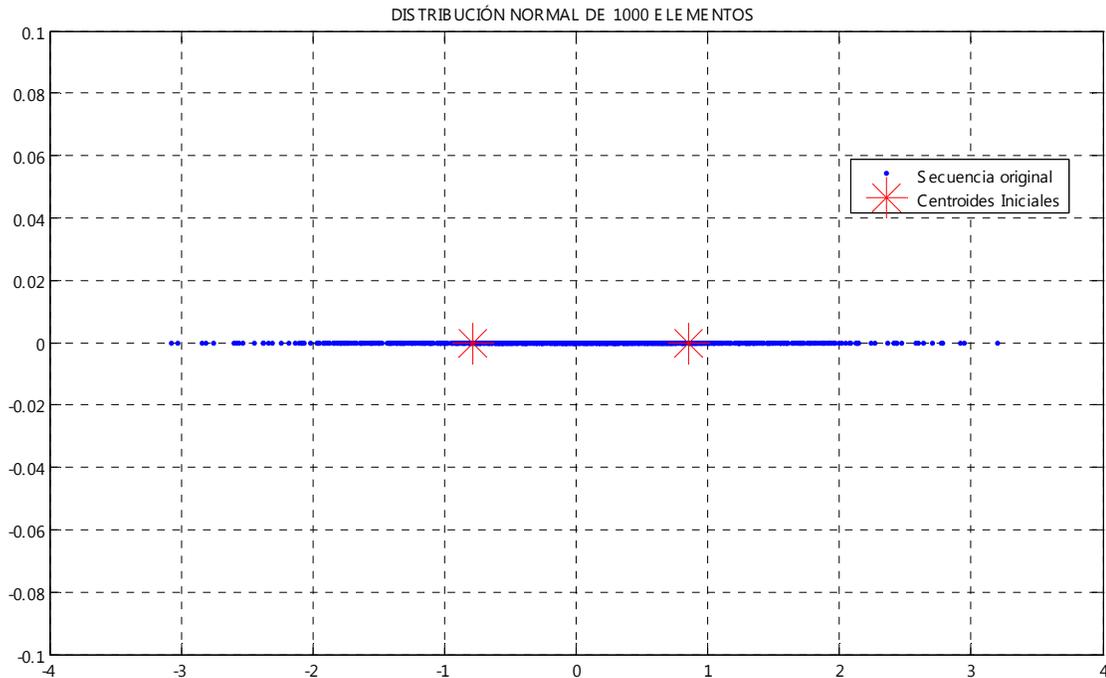


FIG. 5.22 DISTRIBUCIÓN NORMAL DE 1000 ELEMENTOS

Aquí se observan los primero dos centroides generados a partir de la secuencia original de 1000 elementos. A diferencia de los resultados obtenidos en la distribución uniforme, aquí es posible darse cuenta que los centroides no se encuentran distribuidos de una manera equidistante, es decir, en este caso se localizan cerca de donde se encuentra la mayor concentración de elementos, mas cerca del origen, por lo tomando en cuenta que la frontera esta casi en 0, el centroide no está precisamente en el centro de la nueva subregión.

Para observar las diferencias con mayor apreciación, se muestra en la gráfica de abajo la localización de los dos primeros centroides correspondientes tanto a la distribución normal como uniforme, esto con la finalidad de mostrar el efecto que tiene la distribución de la secuencia original para la determinación de dichos puntos:

```
x=randn(1000,1);
subplot(2,1,1); plot(x)
hold on
    co=[2,1];
    [f,c]=lloyds(x,co);
    c=[-0.7833 0.8515];
    plot(c)
hold off
a=rand(1000,1);
subplot(2,1,2); plot(a)
```

```

hold on
co=[2,1];
[f,c]=lloyds(a,co);
c=[0.2475 0.7489];
plot(c)
hold off

```

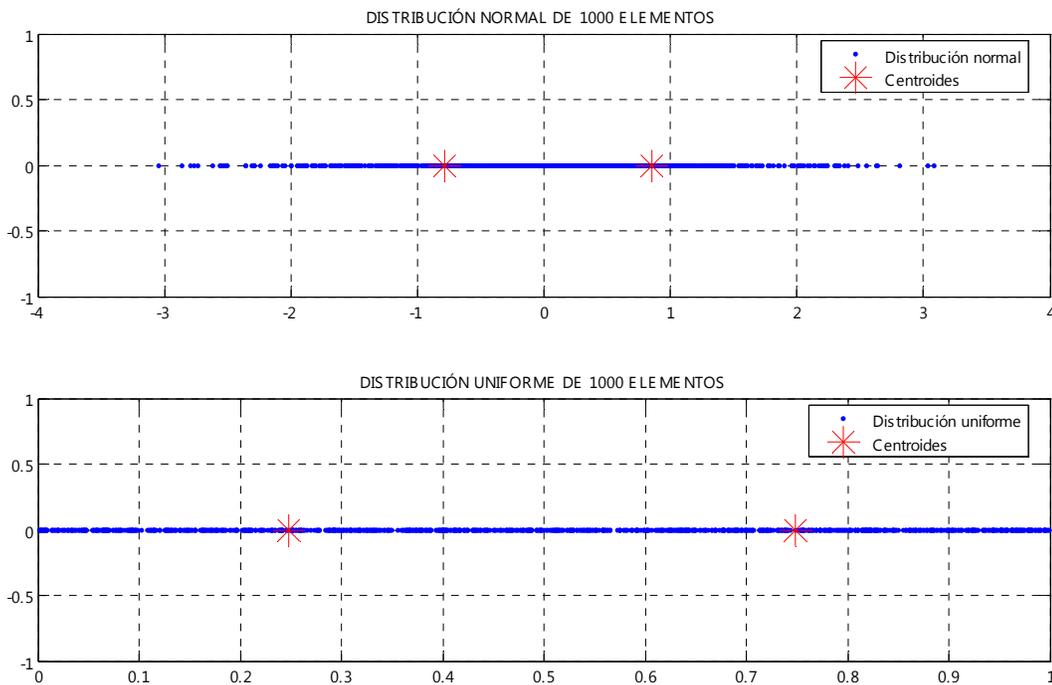


FIG. 5.23.A) CENTROIDES PARA UNA DISTRIBUCIÓN NORMAL
 5.23.B) CENTROIDES PARA UNA DISTRIBUCIÓN UNIFORME

Según la gráfica 5.23 y como se había predicho anteriormente, la localización de estos puntos iniciales C varía en gran medida para cada una de las distribuciones utilizadas, teniendo para una distribución uniforme zonas equidistantes donde la probabilidad de que ocurra un evento es casi la misma para todos los puntos, mientras que para una distribución normal se localizan en la zona con mayor concentración de punto los cuales cuentan con una probabilidad mayor de que sucedan.

Finalmente, aplicando la función correspondiente se tiene el siguiente diccionario para la distribución normal mostrado a continuación para 1 000 elementos aleatorios con distribución normal y una distorsión mínima de 0.003:

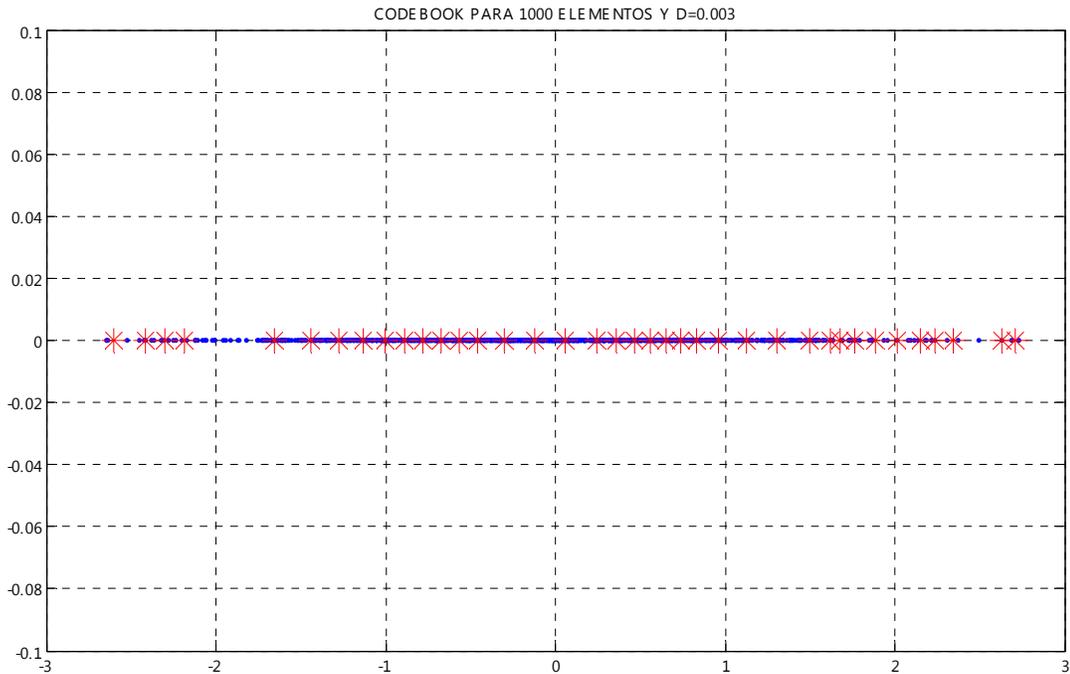


FIG. 5.24 CENTROIDES PARA UNA DISTRIBUCIÓN NORMAL

Como se observa en la figura 5.24 la distribución del diccionario final cuenta con una gran cantidad de elementos al igual que el caso uniforme, pero se puede observar también que no están distribuidos de uniformemente, sino que la mayor cantidad de puntos se localizan en el centro de la gráfica, y en los extremos hay, pero en menos cantidad.

Esto se debe a que en los extremos de la secuencia original las probabilidades de ocurrencia son menores a las del centro, pero aún así el *codebook* debe estar condicionado a que se ocupen con frecuencia o no, debe contener elementos que puedan representar a las muestras ocasionales que puedan surgir en esas regiones. Esto se verá con mayor claridad en el algoritmo COVQ con esta distribución, ya que deberán existir forzosamente centroides en las regiones en donde ni siquiera la secuencia original aparezca, sin embargo el que esta no lo haga no evita que la señal de entrada que es totalmente aleatoria pueda llegar a caer en esos lugares.

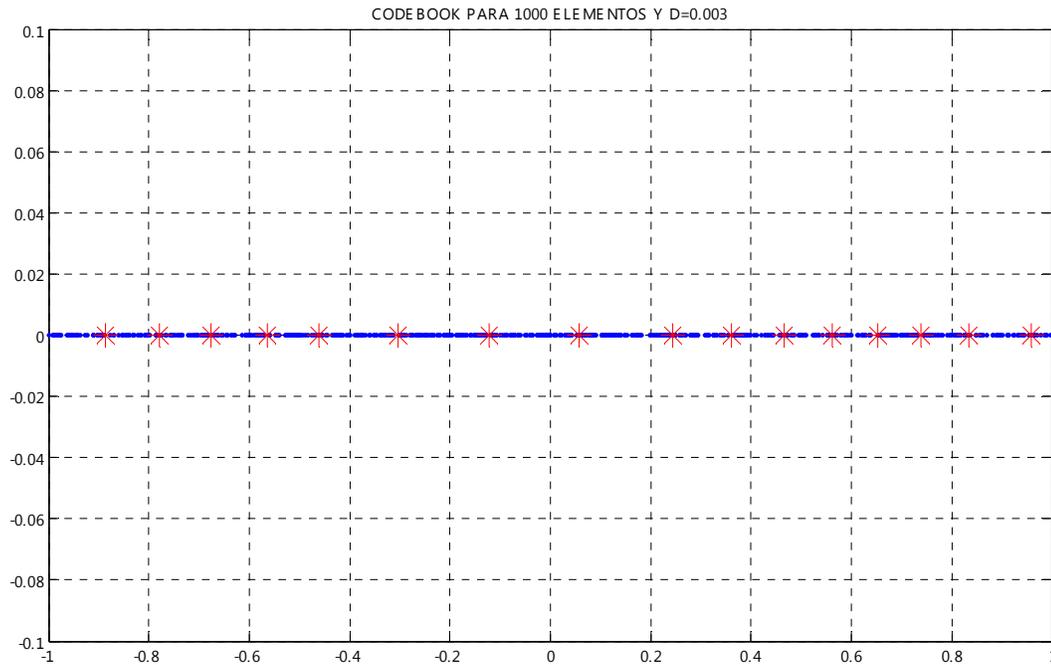


FIG. 5.25 CENTROIDES PARA UNA DISTRIBUCIÓN NORMAL [0,1]

En la figura 5.25 se puede apreciar el mismo diccionario generado pero con un acercamiento en el intervalo positivo [0,1] con la finalidad de observar el comportamiento de los centroides. Como se observa la distancia de cada uno de ellos no es la misma debido a que la señal de entrada tiene una función de probabilidad que hace las muestras más dispersas en ciertas zonas. Sin embargo para este intervalo en particular, donde la secuencia podría tener un comportamiento cercano al de una distribución uniforme, la localización de los centroides es muy similar a las que se observaron para el primer caso del algoritmo LGB con la secuencia uniforme debido a que en esta zona se localizan la mayor cantidad de muestras de la señal original.

Por otro lado, si se observa el comportamiento del diccionario en un intervalo mayor [0,3] es posible percatarse de las diferencias, justo como se muestra en la figura 5.26 donde la cantidad de centroides generados fue menor en el extremo positivo que en el centro (cerca del origen) por las razones que se han venido explicado. Además es fácil comprender que la existencia de estos centroides en específico es de suma importancia, ya que no se puede garantizar que la señal de entrada no vaya a caer específicamente en esas zonas, y si se omiten centroides únicamente por la poca probabilidad de ocurrencia que se tenga, pueden generarse problemas más adelante ya que se tendrían que colocar las muestras de acuerdo a los centroides generados en el resto de la distribución, aumentando significativamente el valor de la distorsión.

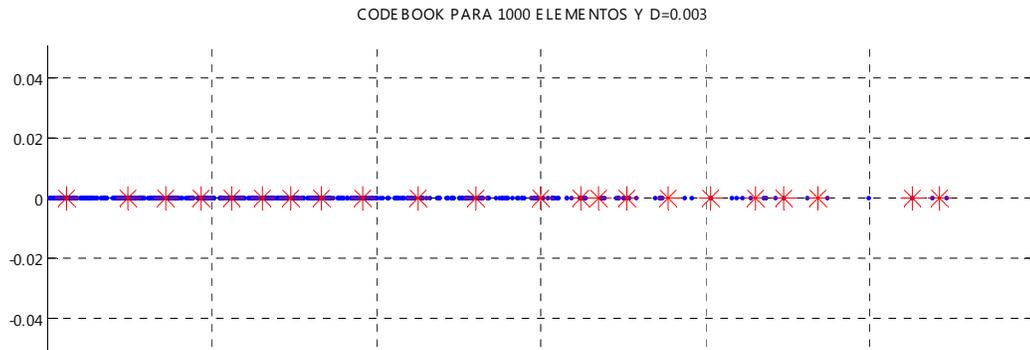


FIG. 5.26 CENTROIDES PARA UNA DISTRIBUCIÓN NORMAL [0,3]

Hasta este momento se han presentado únicamente los resultados obtenidos a partir de una secuencia original de 1 000 elementos y una distorsión de $D=0.003$ con una distribución normal. Para este caso se obtuvo un *codebook* de 64 elementos tanto negativos como positivos (véase anexo 3) en el intervalo de $I[-3,3]$ y de igual forma se han presentado sus representaciones gráficas.

A continuación se presentan los resultados obtenidos para el mismo experimento variando únicamente el número de elementos que componen la secuencia original de 500 elementos, logrando con esto que el número de elementos por zona disminuya a la mitad, esto con la finalidad de observar el comportamiento de los centroides que se generan en las zonas de menos de probabilidad (en los extremos de la secuencia).

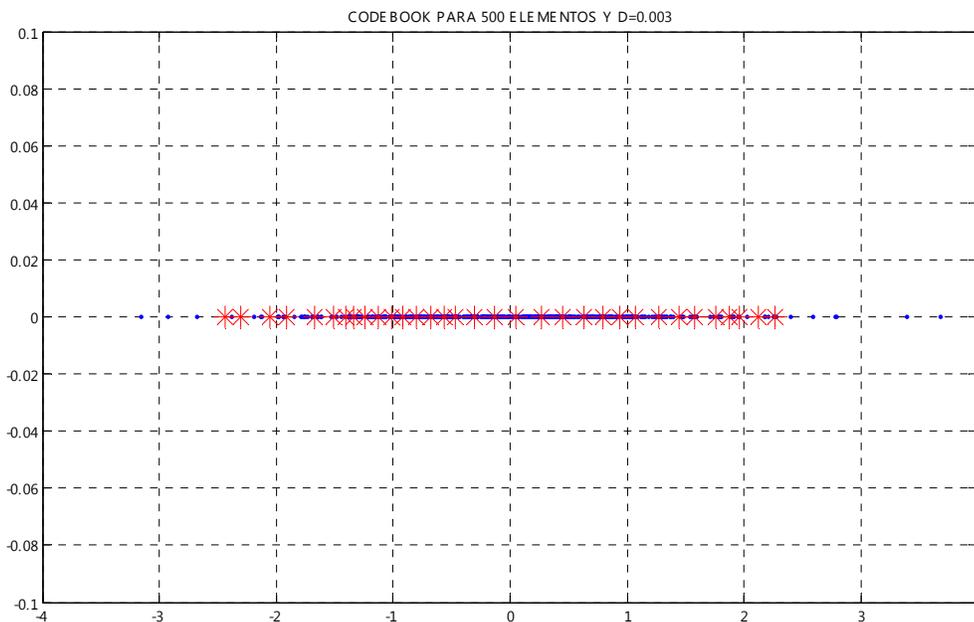


FIG. 5.27 CENTROIDES PARA UNA DISTRIBUCIÓN NORMAL DE 500 ELEMENTOS [-3,3]

En la figura 5.27 se muestra el diccionario generado para las condiciones mencionadas con anterioridad y es posible darse cuenta de que el comportamiento presenta ciertos cambios en comparación con las características anteriores, sobre todo en los extremos donde de inicio, el número de elementos iniciales disminuyó a la mitad. El cambio más significativo es que ahora no hay centroides en las regiones de poca probabilidad, lo cual es un gran problema ya que no se garantiza que esas regiones nunca vayan a ser ocupadas. A continuación se presenta un acercamiento de estas regiones:

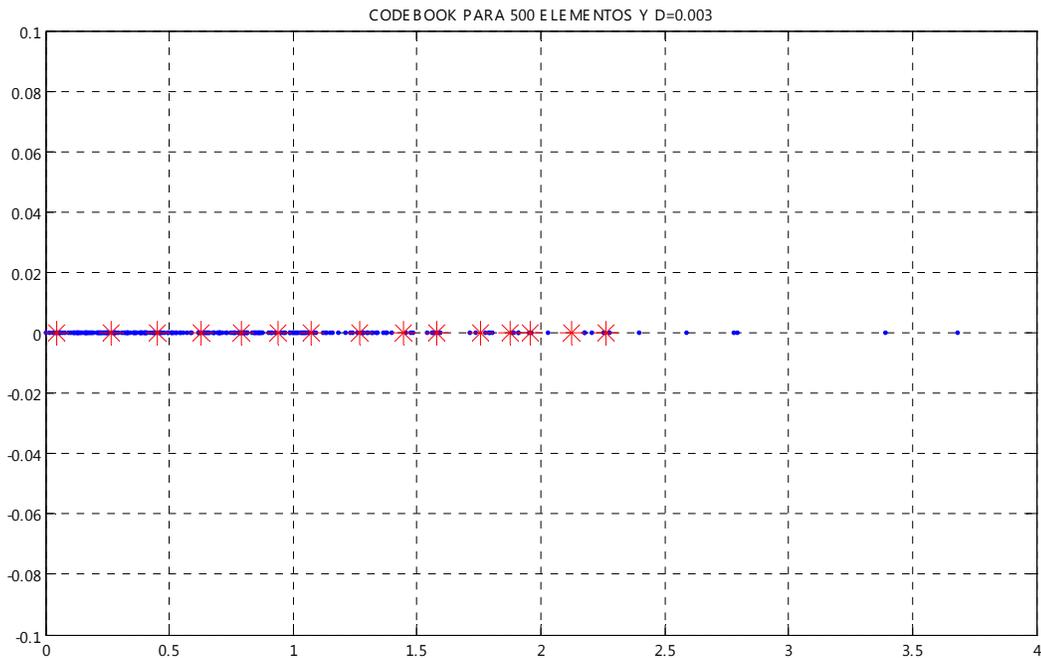


FIG. 5.28 CENTROIDES PARA UNA DISTRIBUCIÓN NORMAL DE 500 ELEMENTOS [0,4]

De acuerdo a los resultados, si una muestra de la señal de entrada se posiciona en el intervalo $[2.5, 4]$ el diccionario no tendrá posibilidades de asignarle un valor con el que pueda garantizar una distorsión mínima, por lo que a todas las muestras en este intervalo no tendrá más opción de asignar el valor de $c = 2.2609$, aunque en el resto de la secuencia se tenga un buen comportamiento.

En la siguiente gráfica se observa el diccionario en el intervalo positivo $[0, 2]$, intervalo que se podría denominar ‘óptimo’ ya que en él, el proceso de cuantización se da de manera normal ya que todos los centroides garantizan una distorsión máxima de 0.003 tanto para el eje positivo como el negativo:

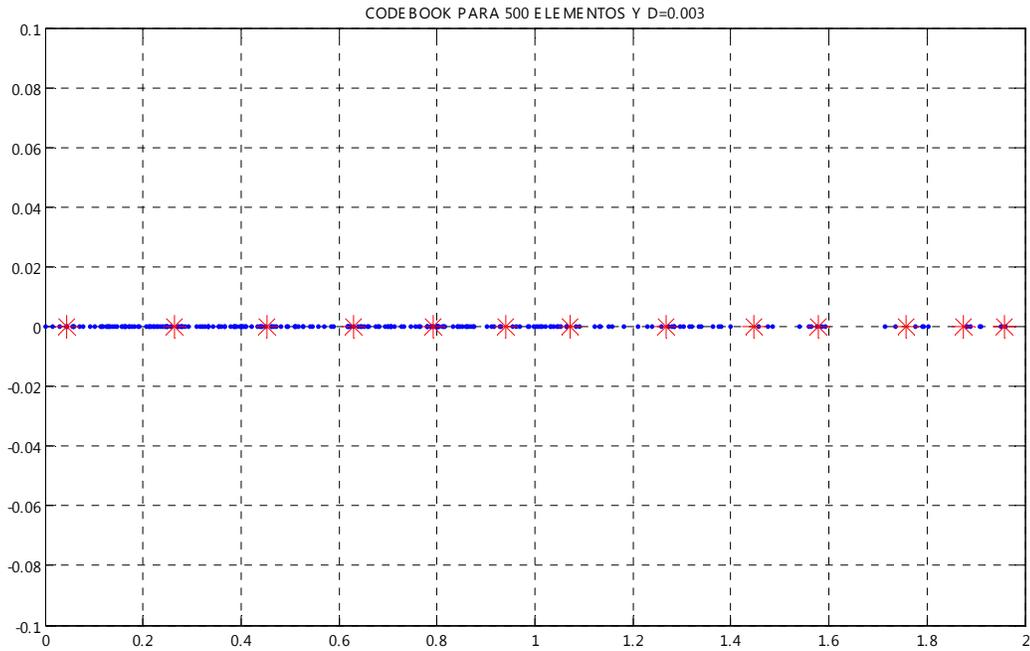


FIG. 5.29 CENTROIDES PARA UNA DISTRIBUCIÓN NORMAL DE 500 ELEMENTOS [0,2]

Finalmente, las características de este diccionario son: se creó a partir de 500 elementos con distribución normal y con una distorsión máxima de 0.003 y el *codebook* final estuvo formado de 64 elementos en el eje positivo y negativo (véase anexo 4).

De esto se puede ver fácilmente que de los dos diccionarios creados para este tipo de distribución, el más óptimo fue el primero generado a partir de 1 000 elementos. Se podría pensar que si de ahorrar elementos se trata, el caso de 500 elementos es el mejor ya que ambos generan la misma cantidad de elementos finales, sin embargo el primer caso es el de mayor eficiencia debido a que permite que las muestras de poca probabilidad sean cuantizadas de manera óptima respetando el margen de distorsión mínimo, a diferencia del segundo caso donde la mayor concentración de centroides se localiza al centro de la distribución dejando los extremos sin puntos que representen a las posibles muestras de entrada.

5.3.3 CUANTIZACIÓN VECTORIAL

Para saber cual de los diccionarios es el más óptimo, a continuación se realizará una comparación basándose en los mismos aspectos que se trataron con la cuantización escalar para el algoritmo LGB, es decir, en el número de elementos finales que se crearon dependiendo del valor de la distorsión.

Como se puede ver en los resultados del Anexo 5, para la cuantización basada en el algoritmo COVQ únicamente se utilizaron de manera inicial 10 000 vectores y solo existen dos intervalos de comparación [0,5] y [0,1], esto debido a que, como se mencionó anteriormente, si se maneja un intervalo más amplio, por ejemplo entre 0 y 10, el número de elementos finales crece de manera exponencial, sin mencionar que el caso de 0 y 100 generaría un diccionario imposible de plasmar.

La tabla que sume los resultados obtenidos es la siguiente:

<i>Distorsión</i>	<i>Numero de elementos finales</i>	<i>Numero de elementos iniciales</i>	<i>Intervalo [0,]</i>
10^{-3}	1156	10000	5
0.5	16	10000	5
1	4	10000	5
10^{-3}	64	10000	1
0.5	4	10000	1
1	4	10000	1

TABLA 5.3 RESULTADOS FINALES DEL ALGORITMO COVQ

Y en la siguiente gráfica se muestra la relación entre los elementos generados a partir de la variación de la distorsión, recordando que un aumento en la distorsión provoca que cada uno de los centroides generados abarque en este caso, un área mayor, por lo que algunos puntos que se encuentren alejados sí podrán asociarse a una pareja ordenada, pero esto provocará que las probabilidades de error aumenten ya que el margen es mayor. La gráfica corresponde al intervalo [0,1]:

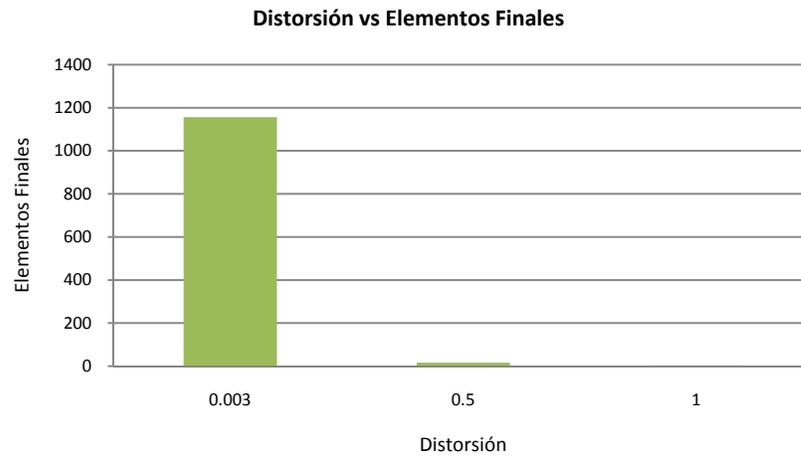


FIG. 5.30 RELACIÓN DISTORSIÓN VS ELEMENTOS FINALES PARA [0,5]

Como se puede observar, para un valor de distorsión casi nulo ($D=0.003$) y para el intervalo de 0 a 5, el número de elementos que conforman el *codebook* es muy parecido al generado en el algoritmo LGB para un intervalo de 0 a 100. Esto da la pauta para evitar hacer iteraciones en intervalos mayores.

Además se observa que al variar la distorsión el número de elementos finales disminuye en gran cantidad, lo que indica que existe un gran margen de error.

Cabe mencionar que para el caso de la cuantización escalar, la distorsión se media a lo más en dos direcciones, por lo que el error generado consistía en pasar de un punto C_1 a un punto C_2 .

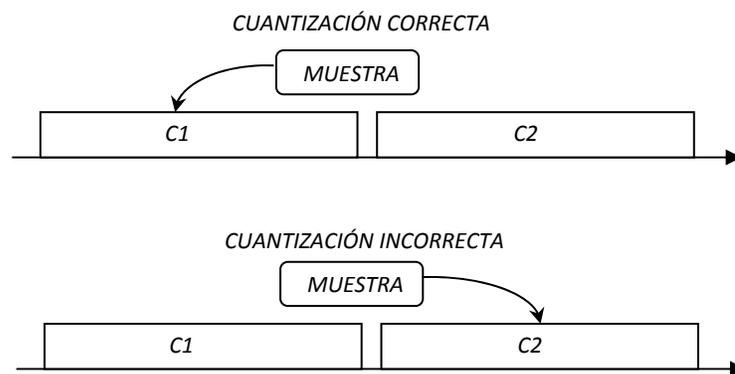


FIG. 5.31.A) CUANTIZACIÓN LGB CORRECTA

5.31.B) CUANTIZACIÓN LGB INCORRECTA

Pero para el caso de la cuantización vectorial un error no consiste en solo dar un punto diferente, sino varias al mismo tiempo, ya que el transmisor puede equivocarse en asignar ambas

coordenadas y terminar por proporcionar un dato que no represente en lo absoluto a la muestra que entra:

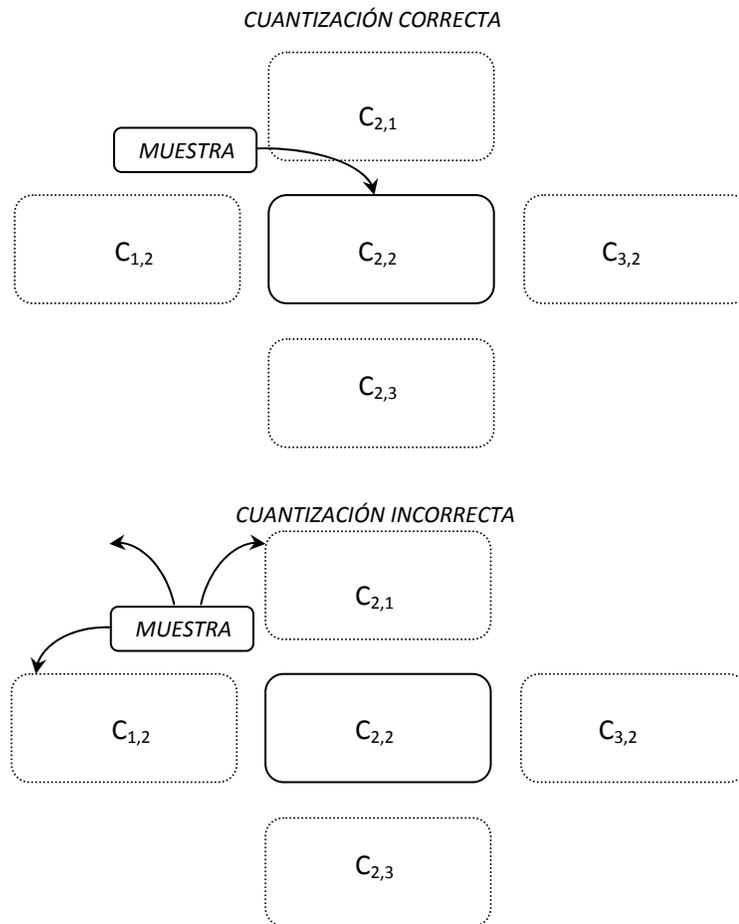


FIG. 5.32.A) CUANTIZACIÓN COVQ CORRECTA
 5.32.B) CUANTIZACIÓN COVQ INCORRECTA

Por tal motivo es considerablemente bueno pensar que a pesar de tener un *codebook* con muchos elementos puede disminuir la velocidad de proceso, pero al mismo tiempo ahorra problemas al evitar crear errores que sucederían si el proceso fuera más rápido y tuviera que buscar la asociación en una base de datos más pequeña.

En la siguiente gráfica se muestran los resultados obtenidos para la comparación de los diccionarios correspondientes al intervalo $[0,1]$:

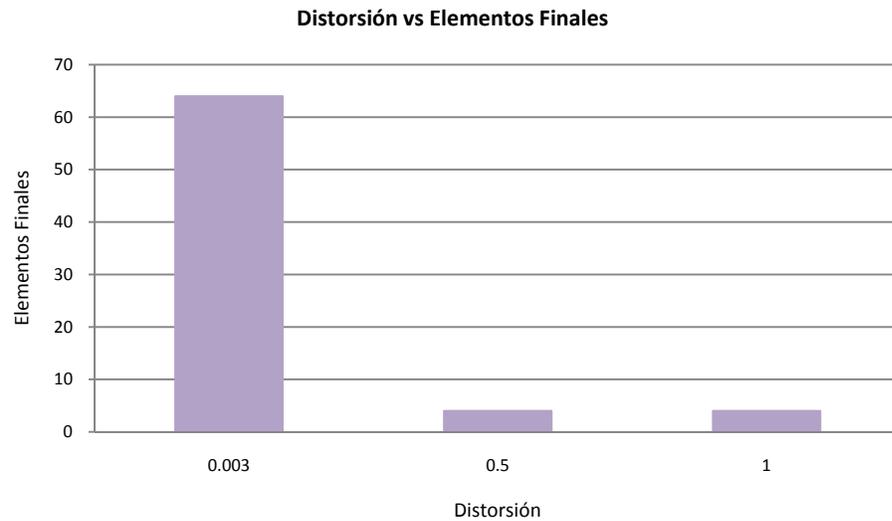


FIG. 5.33 RELACIÓN DISTORSIÓN VS ELEMENTOS FINALES PARA [0,1]

Como se aprecia, el resultado es similar al anterior, es decir, para el valor mínimo de distorsión se generó un diccionario con un número elevado de elementos, y este fue disminuyendo conforme el valor de D aumenta.

Sin embargo también se puede ver la modificación de este parámetro deja de tener efectos significativos, es decir, debido a que el intervalo es pequeño, no es necesario crear muchos centroides ya que para codificar tanto 10 000 vectores como 100 en el intervalo de [0,1] es posible hacer mediante los mismos centroides, esto debido a que ya el margen de error permitido es muy grande.

De este conjunto de diccionarios creados para la representación del algoritmo COVQ, puedo decir que los siguientes son los más óptimos:

<i>Distorsión</i>	<i>Numero de elementos finales</i>	<i>Numero de elementos iniciales</i>	<i>Intervalo [0,]</i>
10^{-3}	1156	10000	5
10^{-3}	64	10000	1

TABLA 5.4 CARACTERÍSTICAS DE LOS DICCIONARIOS SELECCIONADOS

El resultado es claro, de inicio se deben evitar diccionarios que presenten mayor vulnerabilidad a los errores, por lo tanto los creados a partir de $D=0.5$ y $D=1$ quedan omitidos. Esto reduce la lista a los dos presentados.

En ellos se observan las mejores características ya que ambos ofrecen gran cantidad de elementos para representar cualquier valor de las señales de entrada. Para el caso del primero, su cantidad de elementos es mayor, pero debe considerarse que su intervalo también lo es. Mientras que para el segundo, su número de elementos disminuye, pero su intervalo puede perfectamente contener todo tipo de señales. Además de que ambos fueron diseñados a partir de 10 000 vectores aleatorios, lo que sirve como referencia para saber que se realizaron las iteraciones de forma correcta y cada uno de los centroides representan la mayor cantidad de puntos posibles.

5.3.4 COVQ: DISTRIBUCIÓN NORMAL

Para el caso del algoritmo COVQ también se optó por hacer muestras variando la distribución de la secuencia de entrada con la finalidad de observar el comportamiento y localización de los centroides finales del diccionario. Inicialmente se utilizaron 1 000 vectores de dos dimensiones con distribución normal (gaussiana). A continuación se presenta la secuencia inicial:

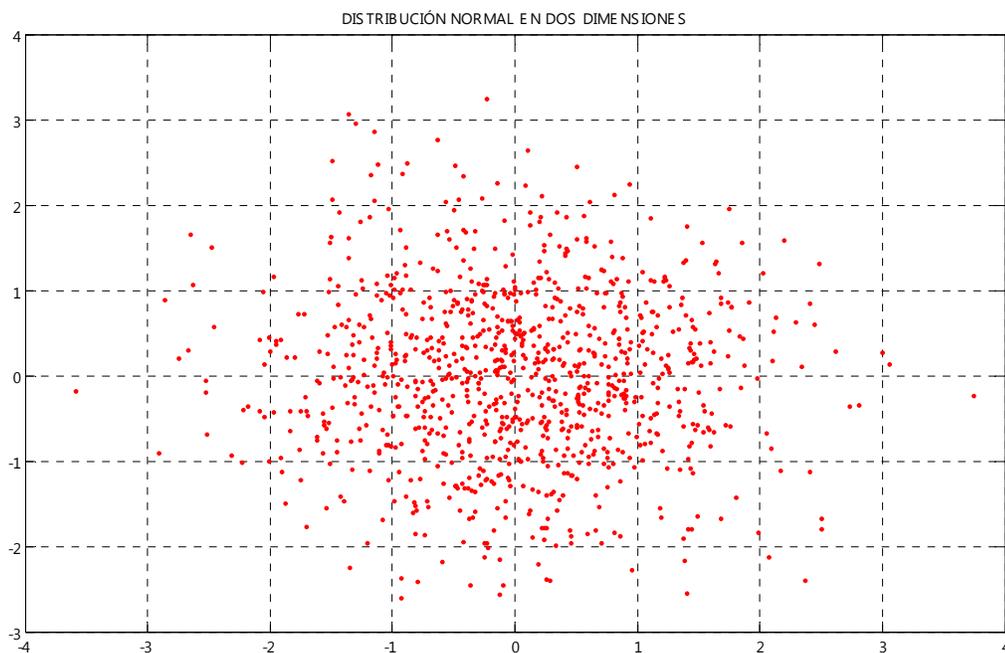


FIG. 5.34 SECUENCIA CON DISTRIBUCIÓN NORMAL DE 1000 ELEMENTOS EN 2D

Debido a que en este caso se está trabajando en dos dimensiones es fácil darse cuenta de varios aspectos que en el caso de LGB para la misma distribución no se apreciaban a simple vista,

el primer aspecto importante es que la distribución entre los cuatro cuadrantes no es uniforme ni simétrica, es decir, solo en los intervalos $[-1,1]$ tanto para el eje X y Y se parecía un comportamiento parecido, pero fue de estos la distribución varía mucho.

Otros aspectos importantes es que la mayor concentración esta alrededor del origen y esta va disminuyendo conforme se aleja de el en todas las direcciones.

Al igual que al resto de las distribuciones, se aplica la función creada y se obtiene la siguiente representación del diccionario buscado para los 1 000 vectores de entrada:

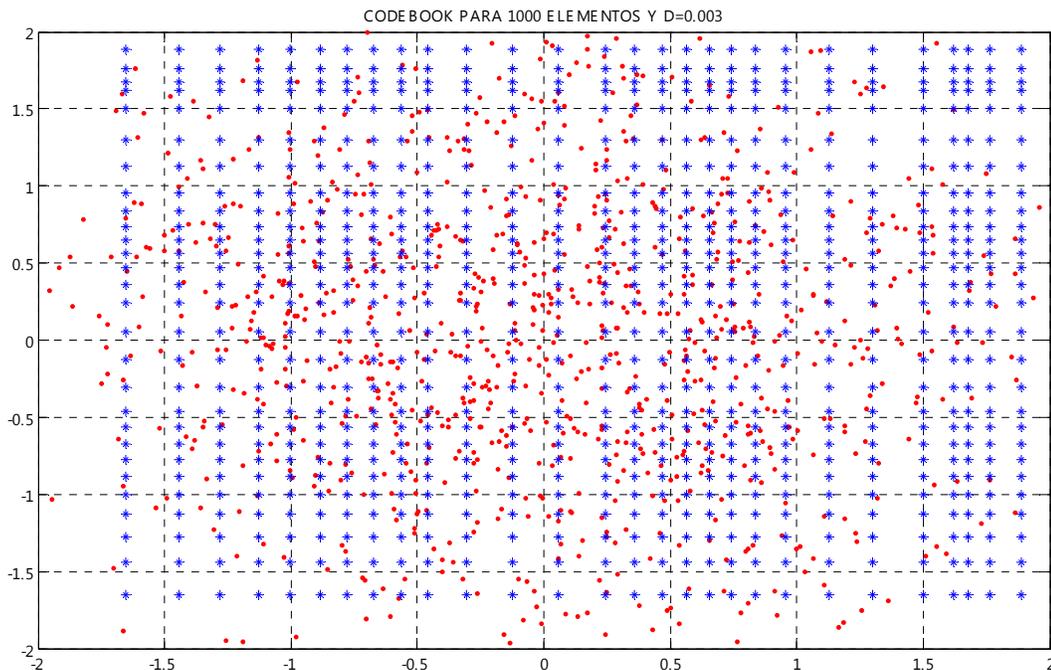


FIG. 5.35 DICCIONARIO PARA 1000 ELEMENTOS Y $D=0.003$

En la figura anterior se muestra el *codebook* generado a partir de la secuencia original de 1000 vectores. Como se observa el resultado es muy similar al obtenido para el caso de la distribución uniforme, pero si existen pequeñas diferencias que se explicarán a continuación.

La diferencia más notable que se puede ver a simple vista del resultado obtenido es que existe una menor presencia de centroides en las orillas de la gráfica, es decir, en los extremos de la señal original donde el número de muestras es menor debido a la probabilidad de ocurrencia. Además de que nuevamente se tiene un comportamiento similar al de una reja que cubre cada uno de los puntos del plano en su mayoría de forma uniforme en los intervalos $[-2,2]$.

Al igual que se mencionó en el algoritmo LGB para esta distribución, después de ver el resultado obtenido se generaron centroides en zonas del plano en donde de forma inicial no había

señal de entrada, pero esto se debió a que son necesarios, ya que para una señal de entrada cualquiera deben existir todas las posibilidades para poder cuantizarla de forma correcta. En la gráfica 5.36 se observa el diccionario obtenido en un intervalo más pequeño de valores:

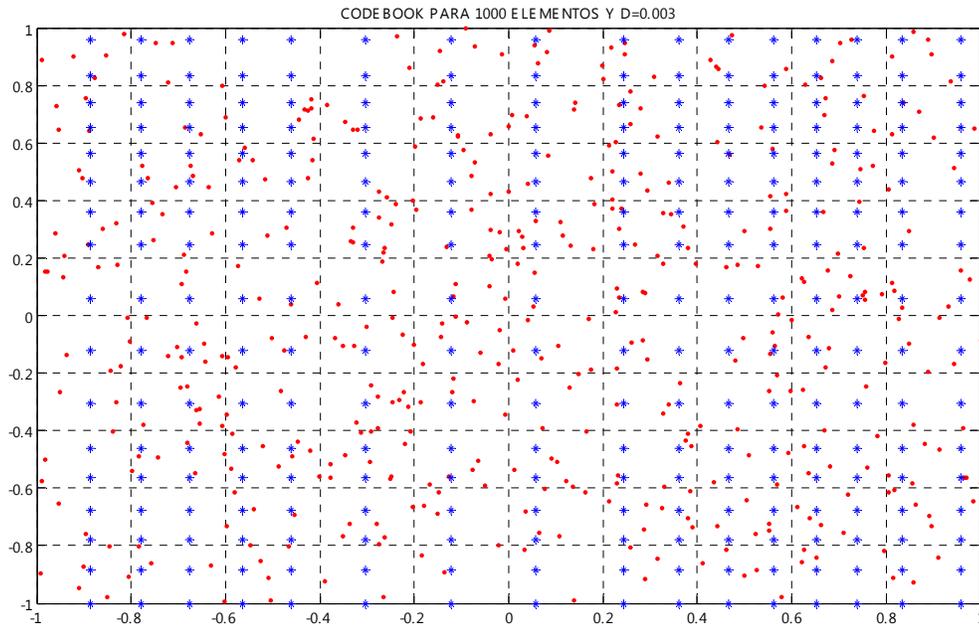


FIG. 5.36 DICCIONARIO PARA 1000 ELEMENTOS Y D=0.003 EN [-1,1]

Para los 1 000 vectores con distribución normal de entrada y con $D=0.003$ se obtuvo un diccionario final de 1 156 elementos, sin embargo para el caso de 500 vectores aleatorios de obtuvieron las misma cantidad de parejas ordenadas (véase anexo 7). A continuación se presentan los resultados para este segundo caso:

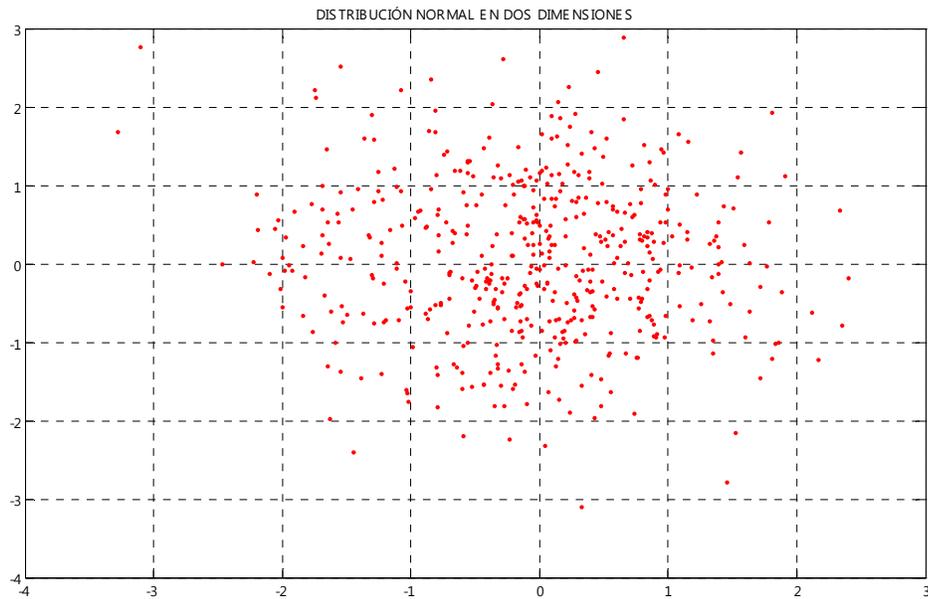


FIG. 5.37 SECUENCIA CON DISTRIBUCIÓN NORMAL DE 500 ELEMENTOS

A diferencia de la secuencia de 1 000 vectores, en esta figura se puede observar una distribución mucho más dispersa y con más ‘huecos’, además de que las zonas de probabilidad baja presentan menos cantidad de muestras.

Y el diccionario se obtuvo mediante la aplicación de la función diseñada y el resultado final se muestra a continuación:

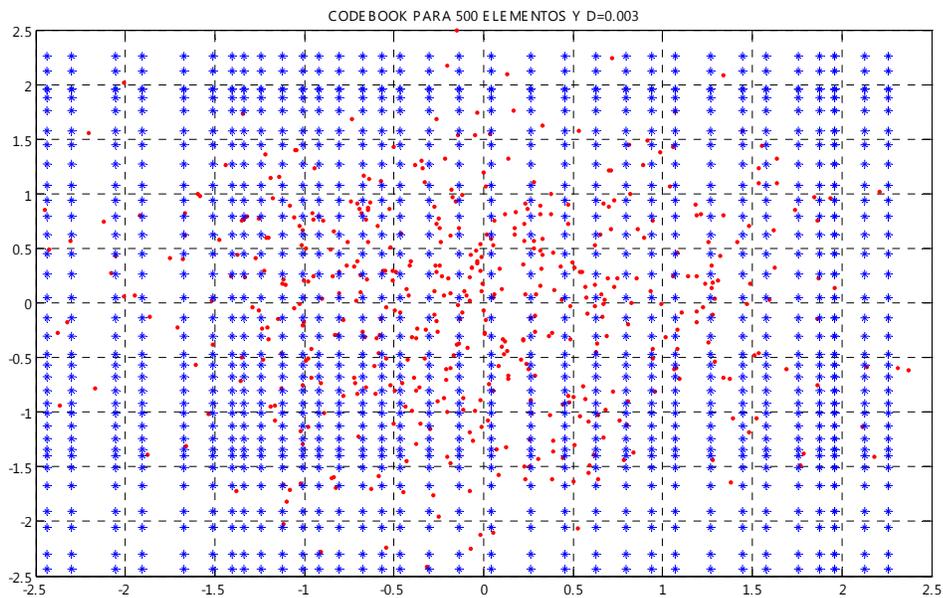


FIG. 5.38 DICCIONARIO PARA 500 ELEMENTOS Y D=0.003

Para este caso es más factible utilizar el diccionario obtenido con estas características ya que además de tener el mismo número de elementos que el del primer caso, ahora si permite la correcta cuantización de muestras posiblemente localizadas en los extremos de la distribución aún proviniendo de una distribución con una cantidad pequeña de muestras en estas zonas.

5.4 RESULTADOS

En el desarrollo del presente capítulo, se realizaron las simulaciones correspondientes para obtener un *codebook* representativo de la cuantización escalar (algoritmo LGB) y vectorial (algoritmo COVQ). Además se incluyó el *codebook* seleccionado como el más óptimo a partir de las especificaciones dadas en la comparación.

Sin embargo, para cuestiones de comparación, a continuación se presentan los diferentes *codebook* obtenidos con cada una de estas variaciones. Estos se presentan en forma de anexos, divididos en ocho apartados.

Para establecer cual de todos los resultados obtenidos, se compararon varios aspectos como el intervalo de trabajo, el número de elementos finales que conformaron el diccionario, el valor de distorsión a partir del cual estos se crearon.

En el caso de cuantización escalar del algoritmo LGB se seleccionó el diccionario creado a partir de un valor de distorsión $D=0.003$, es decir, un valor muy pequeño. Además se manejó en el, un intervalo mayor $[0,100]$ para dar más oportunidad a las señales de entrada a ser representadas con más detalle, y el número de elementos finales fue de 1212. Estas tablas se localizan en el anexo 1 de forma numérica y de forma gráfica están contenidos en el anexo 2 para el caso de la distribución uniforme.

Los primeros dos anexos corresponden exclusivamente a los resultados del algoritmo LGB, en donde se presentan las tablas con diferentes *codebook* generados a partir de diferentes valores de distorsión, desde $D=[10^{-3}, 0.5, 1, 10, 50]$. Otro parámetro importante fue el intervalo en que se generaron las muestras $I=\{[0,100], [0,10], [0,1]\}$, e incluso el número total de muestras iniciales. Estos dos primeros anexos corresponden a una distribución de probabilidad uniforme para las secuencias de entrada al sistema. El anexo dos incluye estos mismo resultados, pero de una forma gráfica, con la finalidad de observar detalladamente la localización de cada uno de los valores.

También se agregan en los anexos 3 y 4 las tablas y gráficas correspondientes para la distribución normal en el caso de cuantización escalar.

En el caso de la cuantización vectorial del algoritmo COVQ, se seleccionó el diccionario creado con las siguientes condiciones iniciales: una distorsión $D=0.003$, en un rango de $[0,5]$ y con 1156 parejas ordenadas, las cuales pueden ser consultadas en los Anexos 5 y 6 tanto de forma numérica como gráfica.

Finalmente, los anexos 7 y 8 contienen los resultados correspondientes a la cuantización vectorial del algoritmo COVQ pero con una distribución de probabilidad normal para los valores de entrada.

Capítulo 6

CONCLUSIONES

En el presente capítulo se darán a conocer las conclusiones a las que se llegó al momento de realizar cada uno de los capítulos estudiados, primero de una forma particular, para dar paso a las conclusiones generales.

En cada uno de los capítulos anteriores se dio a conocer de una manera detallada las principales bases que componen la teoría de Probabilidad. En el primer capítulo teórico se mencionaron conceptos que tuvieron gran importancia en la realización de la parte práctica del presente trabajo. Además se conocieron las características más importantes de diferentes tipos de fuentes de datos así como sus distribuciones de probabilidad para las muestras, esto tomó relevancia en la parte de simulación pues algunas de estas distribuciones fueron utilizadas para la generación de los datos buscados.

En este mismo capítulo se hizo un énfasis en que la mayoría de los procesos que se dan en la realidad son aleatorios y es casi imposible predecir su comportamiento y mucho menos representarlo mediante una expresión bien definida, y para ejemplificar este tipo de comportamientos, se utilizaron secuencia aleatorias (aunque en este caso se trato de distribución que cuenta con un modelo matemático) para la realización de las simulaciones correspondientes al capítulo 5.

En el siguiente capítulo se mencionaron los principales elementos que conforman el sistema de comunicaciones así como la problemática que este supone. Es decir, todas aquellas perturbaciones que pueden afectar a la señal a través de su paso por el canal de comunicaciones y en general, a través de todos los elementos que conforman el sistema, ya sea en el transmisor, receptor,

medio de comunicaciones, hasta las generadas en la misma fuente que genera la señal a enviar. Este tipo de perturbaciones son muy variadas y se deben tanto a los procesos intrínsecos como extrínsecos del sistema. La variedad de perturbaciones es muy amplia, ya sea ruido, distorsión, atenuación, interferencia, etc., pero en este trabajo se hizo hincapié en un solo factor que pudo ser llevado al mínimo, la distorsión. Se eligió este parámetro ya que es una de las perturbaciones más comunes y que causa más daño a las señales, además de que se pudo implementar de forma satisfactoria como una de las condiciones iniciales en el proceso de cuantización, lo que permitió que los resultados obtenidos estuvieran protegidos desde un inicio contra este fenómeno, dando la pauta para que las interferencias que pudieran llegar a aparecer no estuvieran relacionadas con este proceso.

Sin embargo, para poder lograr minimizar el efecto de este parámetro, se realizaron las simulaciones correspondientes al proceso de cuantización utilizando condiciones iniciales que fueran lo más adecuadas posibles para lograr dicho propósito. La metodología utilizada a seguir estuvo basada en los diferentes algoritmos de cuantización tanto escalar como vectorial que se mencionaron en el capítulo correspondiente. Inicialmente se optó por una cuantización más sencilla, es decir, la cuantización del tipo escalar la cual estuvo basada en el algoritmo de diseño LGB. Esta simulación se llevó a cabo variando las características iniciales del diseño con la finalidad de obtener varios resultados con los cuales fuera posible realizar una comparación a detalle, y que al mismo tiempo permitiera entender las ventajas y desventajas de cada uno de estos, para que, finalmente, se pudiera determinar cual de todos estos resultados cumplía con las mejores características de implementación.

A partir de estos resultados se pudieron observar algunas características muy importantes:

- Una desventaja que surgió rápidamente radicó en que mientras que el valor de distorsión utilizado al inicio del diseño se hacía más pequeño, el número de elementos finales que conformaron el resultado fue muy grande, siendo el diccionario seleccionado como el más óptimo, aquel que conto con una cantidad muy grande de elementos. Sin embargo se tomó como buen ya que además de tener un valor de distorsión mínima permitió describir el comportamiento de señales totalmente aleatorias en un rango de trabajo muy amplio. Esto sucedió para cada una de las distribuciones de probabilidad manejadas.
- Otro punto muy importante fue el caso contrario, en donde al aumentar poco a poco el valor de distorsión antes de generar un resultado, se obtuvieron resultados cuya cantidad de elementos fue muy pequeña, es decir, se obtuvieron resultados formados por 2 o 4 elementos únicamente, lo cual se traduce en una cantidad mínima de centroides funcionales

que permiten una correcta representación de las señales de entrada de una manera eficiente, y al mismo generará una mayor cantidad de errores incluso antes de llevarse a cabo una transmisión.

- Finalmente se observó que los resultados obtenidos para las distribuciones usadas (uniforme y normal) fueron muy similares cuando la distorsión era mínima, es decir, ambas convergieron al mismo resultado variando de forma significativa únicamente al elevar el valor de este parámetro de diseño debido a las características de cada una de estas distribuciones.

Con esta simulación inicial se obtuvieron resultados satisfactorios, sin embargo aún no cumplían con las características ni los objetivos propuesto, por tal motivo se realizó el procedimiento correspondiente para poder definir la metodología de un tipo de cuantización que resulta ser más eficiente ya que permite mejor manejo de las señales de entrada: la cuantización vectorial. Para ejemplificar este procedimiento se decidió utilizar el algoritmo de diseño COVQ el cual es muy similar al realizado anteriormente, pero ahora se amplió el campo de estudio a dos dimensiones.

En estas simulaciones se utilizaron nuevamente las diversas formas de distribución de probabilidad de la señal de entrada y las variaciones de las condiciones iniciales antes de la obtención de los resultados para que al final se fuera capaz de determinar cual resultó ser el mejor. Algunas de las principales observaciones fueron las siguientes:

- En primera instancia se observó que el intervalo de trabajo tenía que disminuir ya que la generación de parejas ordenadas se elevó en un factor casi cuadrático con respecto a la cuantización escalar del algoritmo LGB, ya que se trabajó en una dimensión más.
- A pesar de obtener una gran cantidad de parejas ordenadas, se eligió el resultado obtenido a partir del mínimo de distorsión ya que al ser en dos dimensiones la variación de la señal es mayor y por lo tanto existen zonas donde la probabilidad de ocurrencia de las muestras iniciales es menor, pero eso no garantiza que una señal de entrada cualquiera no vaya a caer en dichas zonas.
- De igual forma que en el LGB, los resultados para ambas distribuciones fueron muy similares al converger al mismo valor de distorsión, y fueron muy diferentes cuando este valor se aumentó.
- Finalmente uno de los aspectos que podrían considerarse de mayor importancia en estas simulaciones se dio en los errores, ya que como se mencionó en el capítulo

correspondiente, la probabilidad de que un error en el caso escalar ocurra no afecta en gran parte al mensaje transmitido, y menos con el valor de distorsión diseñado, ya que las muestras del diccionario son muy cercanas y una pequeña variación no representa grandes pérdidas de datos, sin embargo, para el caso vectorial, esta variación no solo es en un eje, sino en dos, lo que provoca que un error por mínimo que sea, no solo tiene dos posibles valores, sino que tiene ocho valores en los cuales puede caer la muestra, y si este fenómeno se realiza con frecuencia, no se garantiza una buena transmisión.

Para ambos casos, tanto la cuantización escalar y vectorial se generaron resultados con dos tipos de distribuciones, sin embargo no fue posible determinar cual de ellas tiene mayor funcionalidad ya ambos casos convergieron a los mismo resultados, por lo que ambos resultados sirven para representar de forma adecuada a cualquier señal de entrada.

Al término de esta tesis es posible decir que se cumplieron los objetivos planteados al inicio ya que se pudo aislar el problema en el sistema de comunicaciones y se plantearon las posibles soluciones para ello. Además de que se investigaron y comprendieron los términos, métodos, algoritmos y características de la cuantización de tipo vectorial pariendo de la antecesora cuantización escalar.

Se compararon los resultados de los diferentes algoritmos de cuantización utilizados y se seleccionaron aquellos cuyas características resultaron en las más óptimas para su implementación dentro de un canal de comunicaciones. También se presentaron todos los resultados obtenidos en cada una de las simulaciones para su análisis e implementación en un sistema real.

Finalmente, el siguiente trabajo de tesis cumplió con las siguientes aportaciones:

- Se explicó de forma clara y detallada la problemática que existe en la codificación de fuente y canal y las diversas perturbaciones que se adhieren a la señal en su paso por el sistema de comunicaciones, y se dio solución a ello.
- Se presentaron las simulaciones a diferentes algoritmos de cuantización escalar y vectorial, LGB y COVQ.
- Se seleccionaron aquellos resultados denominados óptimos a partir de sus características iniciales y finales.
- Se dieron los fundamentos para determinar cuales debieron ser elegidos y porque.
- Se presentó la base de datos final la cual contiene los resultados obtenidos en la simulación de los procesos de cuantización escalar y vectorial en forma numérica y en forma gráfica.

ANEXO 1: CUANTIZACIÓN ESCALAR (DISTRIBUCIÓN UNIFORME)

Parámetros de Diseño: Codebook generado a partir de 10 000 números aleatorios entre [0,100] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.003.

0.0562	3.7205	7.1393	10.3369	13.6383
0.1625	3.8133	7.2130	10.4757	13.7285
0.2482	3.8623	7.2860	10.5925	13.8111
0.3547	3.9461	7.3555	10.6518	13.8893
0.4634	4.0165	7.4230	10.6732	13.9476
0.5688	4.1325	7.5149	10.7222	14.0084
0.6465	4.1616	7.6163	10.8148	14.0659
0.7173	4.2366	7.7119	10.9194	14.1404
0.8015	4.3290	7.7780	11.0048	14.2305
0.8748	4.4067	7.8420	11.1101	14.3346
0.9234	4.4946	7.9190	11.2034	14.3981
0.9583	4.5525	7.9292	11.3110	14.4407
1.0311	4.5861	8.0109	11.3818	14.6539
1.1162	4.6238	8.0740	11.4766	14.7399
1.2233	4.7098	8.1614	11.5636	14.8379
1.3023	4.7491	8.2392	11.6394	14.9430
1.3726	4.8207	8.3379	11.6980	15.0519
1.4507	4.8861	8.4352	11.7363	15.1497
1.5049	4.9379	8.5091	11.7960	15.2038
1.5746	5.0107	8.5969	11.8492	15.2504
1.6371	5.0999	8.7116	11.9071	15.3018
1.7065	5.2116	8.8014	12.0015	15.3690
1.8150	5.2859	8.8948	12.1102	15.4335
1.8969	5.4129	8.9201	12.1964	15.5224
1.9734	5.5341	8.9480	12.2767	15.5702
2.0764	5.6374	8.9480	12.4088	15.6537
2.1679	5.7364	8.9701	12.4542	15.7565
2.2145	5.8368	9.0897	12.5189	15.7931
2.3290	5.9434	9.1636	12.6020	15.8370
2.4637	6.0467	9.2201	12.6630	15.9217
2.5447	6.1306	9.2857	12.7005	16.0099
2.6277	6.2372	9.4285	12.7693	16.0959
2.7347	6.3309	9.5695	12.8564	16.2268
2.8002	6.4083	9.6575	12.9262	16.2792
2.8617	6.4888	9.7993	13.0001	16.3738
2.9302	6.5651	9.9105	13.0402	16.4649
3.0542	6.6570	9.9792	13.0572	16.5955
3.1601	6.7253	9.9969	13.0839	16.6734
3.2653	6.7799	10.0423	13.1561	16.7928
3.3815	6.8834	10.1075	13.2351	16.8498
3.4624	6.9768	10.1677	13.3247	16.9138
3.5658	7.0451	10.2058	13.4596	16.9785
3.6494	7.0783	10.2674	13.5498	17.0435

17.1221	21.3404	25.5815	30.0028	34.5503
17.2059	21.4622	25.6429	30.0988	34.6398
17.2872	21.5820	25.7241	30.1904	34.7171
17.3637	21.6859	25.8137	30.3022	34.8202
17.4326	21.7687	25.8903	30.4588	34.9194
17.5341	21.8451	26.0178	30.5277	35.0115
17.5671	21.9743	26.1370	30.6070	35.1150
17.6179	22.0642	26.2408	30.6923	35.2259
17.6976	22.1669	26.3288	30.7516	35.2881
17.8093	22.2541	26.4491	30.8474	35.3957
17.9044	22.3574	26.5378	30.9437	35.4819
17.9701	22.4423	26.6280	31.0037	35.5976
18.0786	22.5578	26.7269	31.0733	35.6993
18.1920	22.6420	26.8531	31.1496	35.7831
18.2965	22.7145	26.9551	31.2523	35.8933
18.4121	22.7920	27.0378	31.3284	35.9990
18.4395	22.8397	27.1191	31.4259	36.0748
18.4675	22.8917	27.1915	31.5220	36.1985
18.5597	22.9742	27.2242	31.6168	36.2612
18.6936	23.0750	27.2680	31.7071	36.3723
18.7980	23.1522	27.3512	31.8646	36.4672
18.8971	23.1999	27.4230	31.9824	36.5767
18.9895	23.2501	27.4905	32.0913	36.6632
19.0898	23.3131	27.5883	32.1775	36.7251
19.1654	23.3848	27.6495	32.2556	36.7649
19.2505	23.4607	27.7225	32.3485	36.8073
19.3104	23.6040	27.8255	32.4128	36.8706
19.4186	23.6831	27.8773	32.5054	36.9607
19.4897	23.7606	27.9466	32.6015	36.9963
19.5651	23.8208	28.0453	32.6880	37.0718
19.6268	23.8938	28.1466	32.8020	37.1490
19.8192	23.9832	28.2236	32.8807	37.2881
19.8830	24.0539	28.3456	32.9649	37.3701
19.9709	24.1196	28.4087	33.0318	37.5106
20.0173	24.2217	28.4535	33.1241	37.5711
20.0786	24.3097	28.7189	33.2420	37.6530
20.1609	24.4165	28.7935	33.3386	37.7566
20.2405	24.5121	28.8727	33.4774	37.8692
20.3285	24.6255	28.9656	33.5504	37.9918
20.4300	24.7308	29.0450	33.6616	38.0816
20.4992	24.8440	29.1389	33.7622	38.1239
20.5656	24.9159	29.2311	33.8872	38.1688
20.6331	24.9907	29.3287	33.9367	38.2306
20.6966	25.0329	29.4250	34.0068	38.3322
20.7763	25.0670	29.5185	34.0649	38.4111
20.8842	25.1563	29.5837	34.1107	38.4678
20.9205	25.2714	29.6970	34.2027	38.5430
20.9585	25.3576	29.7437	34.2683	38.6148
21.0664	25.4278	29.7836	34.3200	38.6814
21.1396	25.4825	29.8418	34.4055	38.7741
21.2510	25.5237	29.9215	34.4646	38.7964

38.8359	42.9065	46.9521	51.0000	55.2130
38.8462	43.0355	47.0666	51.0755	55.2937
38.8939	43.1330	47.2051	51.1136	55.4142
38.9696	43.2382	47.2966	51.1765	55.5215
39.0347	43.2856	47.3695	51.2524	55.6339
39.0817	43.3520	47.4071	51.3371	55.6740
39.1428	43.4331	47.4470	51.4222	55.7643
39.2096	43.5189	47.5463	51.5116	55.8494
39.3133	43.5632	47.5803	51.6267	55.9576
39.3773	43.6589	47.6113	51.7153	56.0551
39.4439	43.7497	47.6617	51.7770	56.1682
39.5615	43.8606	47.7575	51.8658	56.2654
39.6624	43.9290	47.8441	51.9854	56.3366
39.7575	44.0089	47.9539	52.0619	56.4032
39.8987	44.0765	48.0559	52.1522	56.4750
39.9832	44.1917	48.1320	52.2574	56.5318
40.0629	44.2689	48.2165	52.3470	56.6604
40.1716	44.3369	48.2933	52.4314	56.7451
40.2810	44.4041	48.3842	52.5099	56.8660
40.3460	44.5282	48.4497	52.6074	56.9450
40.4479	44.6069	48.5166	52.7009	57.0316
40.5489	44.7243	48.5888	52.7815	57.1192
40.6267	44.8670	48.6749	52.8951	57.2163
40.7069	44.9921	48.7302	53.0171	57.2924
40.8139	45.0970	48.8201	53.0743	57.3419
40.9321	45.2203	48.9299	53.1603	57.3905
41.0374	45.2948	48.9987	53.2391	57.5278
41.1173	45.3799	49.0563	53.3222	57.6139
41.1889	45.4306	49.1186	53.4252	57.7274
41.2721	45.4916	49.1850	53.4963	57.8181
41.3467	45.5548	49.2664	53.5960	57.8661
41.4277	45.5980	49.3876	53.6766	57.9060
41.5535	45.6489	49.4830	53.7792	57.9624
41.6236	45.6736	49.5495	53.8688	58.0825
41.6693	45.6804	49.6044	53.9313	58.1496
41.7236	45.6996	49.6532	54.0006	58.2106
41.8225	45.7736	49.7193	54.0632	58.2562
41.8869	45.8569	49.7764	54.1339	58.3265
41.9846	45.9377	49.8649	54.2339	58.4025
42.0615	46.0175	49.9176	54.3123	58.4888
42.1415	46.1128	49.9918	54.3950	58.5477
42.1599	46.2259	50.0821	54.4925	58.6331
42.1965	46.2655	50.1884	54.5661	58.7086
42.2870	46.3172	50.2800	54.6278	58.8099
42.3797	46.3975	50.3902	54.6723	58.9092
42.4580	46.4463	50.4631	54.7458	59.0091
42.5335	46.5219	50.5713	54.7881	59.0925
42.5935	46.5812	50.6676	54.8548	59.1382
42.6873	46.6724	50.7842	54.9698	59.2375
42.7933	46.7601	50.8655	55.0607	59.3017
42.8827	46.8444	50.9304	55.1300	59.3553

59.4174	63.4638	67.9262	72.5435	76.1845
59.4729	63.4886	68.0116	72.6207	76.2524
59.5050	63.5388	68.1200	72.7487	76.3408
59.5731	63.6155	68.2368	72.8570	76.4029
59.6369	63.6968	68.3136	72.9679	76.4712
59.6973	63.7736	68.4231	73.0471	76.5259
59.7340	63.9288	68.5228	73.1332	76.5742
59.7728	64.0153	68.5861	73.1618	76.6232
59.8571	64.1315	68.6592	73.2382	76.6821
59.9135	64.2514	68.7000	73.2563	76.7653
59.9422	64.3736	68.7476	73.2684	76.8053
59.9654	64.4350	68.8366	73.2684	76.8751
59.9654	64.5133	68.9232	73.2892	76.9325
60.0114	64.6044	69.0593	73.3779	77.0491
60.0922	64.6923	69.1543	73.4170	77.1202
60.1456	64.7706	69.2206	73.4883	77.1859
60.2114	64.8741	69.2983	73.5813	77.3033
60.3253	64.9566	69.3606	73.6496	77.3924
60.4127	65.0352	69.4542	73.7358	77.4933
60.5200	65.1625	69.5732	73.8384	77.6064
60.6421	65.2814	69.6756	73.9294	77.6654
60.7590	65.3573	69.7829	73.9750	77.7426
60.8468	65.4636	69.8871	74.0793	77.8204
60.9965	65.5328	70.0193	74.1522	77.9197
61.0914	65.6293	70.1381	74.2391	77.9988
61.1567	65.7403	70.2420	74.2779	78.0912
61.2176	65.8341	70.3409	74.3232	78.1803
61.2884	65.9158	70.4549	74.3831	78.2349
61.4121	65.9796	70.5629	74.5032	78.2480
61.4737	66.0486	70.6914	74.5469	78.2670
61.5357	66.1503	70.7852	74.6056	78.3481
61.6294	66.1948	70.8614	74.6716	78.4190
61.7414	66.2745	70.9166	74.7552	78.4665
61.8798	66.3638	71.0237	74.8344	78.5046
61.9721	66.4653	71.1380	74.9318	78.5736
62.0951	66.5411	71.2223	75.0265	78.6854
62.2269	66.6496	71.3082	75.1203	78.7834
62.3232	66.7081	71.4174	75.1456	78.8566
62.4026	66.7317	71.5096	75.2231	78.9397
62.5054	66.8027	71.5980	75.3320	78.9646
62.5337	66.8792	71.6913	75.4276	79.0098
62.5774	66.9619	71.7689	75.4920	79.1164
62.6580	67.0678	71.8425	75.5513	79.2643
62.7417	67.1725	71.9004	75.6485	79.3561
62.8584	67.2559	71.9366	75.7639	79.4252
62.9489	67.3564	71.9722	75.8412	79.6135
63.0609	67.4579	72.0470	75.8986	79.6893
63.1792	67.5702	72.1187	75.9380	79.7507
63.2476	67.6835	72.2468	75.9860	79.7699
63.3239	67.7761	72.3782	76.0662	79.8401
63.4045	67.8415	72.4613	76.1258	79.9127

80.0350	84.1590	87.8512	91.9586	96.0999
80.1282	84.2690	87.9439	92.0501	96.1068
80.2380	84.3452	88.0512	92.1719	96.1906
80.3205	84.4503	88.1220	92.2931	96.2603
80.4307	84.5000	88.2206	92.3683	96.3438
80.5127	84.5321	88.2727	92.4877	96.4386
80.6721	84.5420	88.3295	92.5914	96.5096
80.7920	84.5580	88.3889	92.7247	96.5777
80.8928	84.6016	88.5113	92.8254	96.6908
80.9384	84.7061	88.6071	92.9089	96.7881
81.0497	84.8556	88.6977	93.0193	96.8888
81.1293	84.9525	88.8188	93.1237	96.9752
81.1984	85.0369	88.9394	93.2333	97.0604
81.2673	85.1255	89.0093	93.3290	97.1478
81.3465	85.2723	89.0486	93.4266	97.2408
81.4445	85.3339	89.1116	93.5005	97.3376
81.5471	85.3906	89.2085	93.6012	97.4298
81.6383	85.4701	89.2922	93.6911	97.4960
81.7467	85.5615	89.3955	93.7907	97.6112
81.8288	85.6599	89.4982	93.8565	97.7153
81.9606	85.7630	89.6102	93.9506	97.7777
82.0438	85.8441	89.7121	94.0193	97.8841
82.1159	85.8835	89.8221	94.1008	98.0018
82.2102	85.9323	89.8853	94.1645	98.1038
82.3131	86.0103	89.9835	94.2360	98.1652
82.3560	86.0873	90.0774	94.3190	98.2138
82.4019	86.1328	90.1452	94.3646	98.2804
82.4995	86.1682	90.2324	94.4268	98.3661
82.5617	86.2257	90.3489	94.5187	98.4420
82.6078	86.3078	90.4512	94.6021	98.5206
82.6744	86.3863	90.5380	94.6771	98.6205
82.7568	86.4631	90.5994	94.7570	98.7445
82.8418	86.5830	90.6777	94.8139	98.8755
82.9226	86.6540	90.7698	94.8619	98.9820
83.0194	86.7168	90.9499	94.9204	99.0791
83.0517	86.8243	91.0511	94.9943	99.1716
83.0890	86.8949	91.1139	95.0777	99.2600
83.1693	86.9634	91.1678	95.1745	99.3978
83.2802	87.0306	91.2167	95.2778	99.5084
83.4131	87.0901	91.2957	95.3785	99.5797
83.5318	87.1880	91.3642	95.4610	99.6653
83.6381	87.2693	91.4110	95.5537	99.7529
83.7412	87.3773	91.4836	95.6405	99.8339
83.8108	87.4503	91.5505	95.7472	99.9223
83.8900	87.5398	91.6682	95.8383	
83.9852	87.6525	91.7526	95.9243	
84.0770	87.7268	91.8740	96.0245	

Parámetros de Diseño: Codebook generado a partir de 1 000 números aleatorios entre [0,10] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.003

0.0620	2.0454	3.8415	5.8596	8.0678
0.1845	2.0454	3.8716	5.9120	8.1428
0.2963	2.0640	3.8976	5.9746	8.2389
0.3896	2.1525	3.9070	6.0777	8.2781
0.4908	2.1784	3.9349	6.1907	8.3508
0.5529	2.2389	4.0093	6.2484	8.4355
0.6439	2.3167	4.1401	6.3418	8.5393
0.6966	2.3880	4.1916	6.4086	8.6203
0.7474	2.4768	4.3021	6.5119	8.6778
0.7968	2.5606	4.3791	6.5665	8.7549
0.8629	2.6502	4.5460	6.6245	8.8443
0.9442	2.7239	4.6418	6.7237	8.9235
1.0294	2.7940	4.7251	6.8110	8.9644
1.1035	2.8378	4.8002	6.8799	9.0332
1.1596	2.8845	4.8653	6.9422	9.1182
1.2111	2.9530	4.9329	7.0105	9.2316
1.2581	3.0303	4.9953	7.0988	9.3320
1.2959	3.0303	5.0750	7.1489	9.4304
1.3725	3.0643	5.1777	7.2541	9.5054
1.4478	3.1353	5.2538	7.3244	9.5786
1.4997	3.2100	5.3076	7.4467	9.6976
1.5865	3.2946	5.4063	7.4876	9.8157
1.6599	3.3585	5.4395	7.5279	9.8777
1.7459	3.4112	5.4772	7.6028	9.9552
1.8566	3.5167	5.5118	7.7095	
1.9104	3.6110	5.5935	7.7812	
1.9995	3.6857	5.6974	7.8797	
2.0218	3.7563	5.7620	7.9764	

Parámetros de Diseño: Codebook generado a partir de 10 000 números aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.003

0.0289	0.2848	0.5388	0.7831
0.0917	0.3494	0.6033	0.8436
0.1555	0.4134	0.6636	0.9087
0.2212	0.4765	0.7251	0.9708

Parámetros de Diseño: Codebook generado a partir de 1 000 números aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.003

0.0265	0.3015	0.5407	0.7878
0.1033	0.3633	0.6028	0.8456
0.1744	0.4212	0.6718	0.9194
0.2380	0.4777	0.7332	0.9737

Parámetros de Diseño: Codebook generado a partir de 100 números aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.003

0.0383	0.2912	0.5603	0.8973
0.1042	0.3972	0.6603	0.9656
0.1641	0.4590	0.7620	
0.2218	0.4962	0.8323	

Parámetros de Diseño: Codebook generado a partir de 10 000 números aleatorios entre [0,100] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 1.

1.4770	26.0957	51.3178	76.5020
4.4742	28.9874	54.3577	79.5641
7.5688	32.2235	57.5741	82.9357
10.8196	35.5636	60.8949	86.0782
13.7660	38.8067	64.0692	89.2308
16.7518	41.7411	67.1328	92.3925
19.8395	44.7240	70.3080	95.4737
23.0422	48.1122	73.3192	98.5175

Parámetros de Diseño: Codebook generado a partir de 1 000 números aleatorios entre [0,10] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 1.

1.3136	6.4760
3.8184	8.8335

Parámetros de Diseño: Codebook generado a partir de 10 000 números aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 1.

0.2496	0.7546
--------	--------

Parámetros de Diseño: Codebook generado a partir de 1000 números aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 1.

0.2737	0.7645
--------	--------

Parámetros de Diseño: Codebook generado a partir de 100 números aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 1.

0.3095	0.7981
--------	--------

Parámetros de Diseño: Codebook generado a partir de 10 000 números aleatorios entre [0,100] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 10.

2.9494	27.6949	52.7838	77.6896
8.8619	34.0141	58.9213	84.0201
15.1375	40.1433	65.1422	90.6601
21.6467	46.6064	71.3457	97.0260

Parámetros de Diseño: Codebook generado a partir de 1 000 números aleatorios entre [0,10] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 10.

2.6236

7.5368

Parámetros de Diseño: Codebook generado a partir de 10 000 números aleatorios entre [0,100] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 50.

6.1374

56.0744

18.6490

68.4769

31.4025

80.6741

43.4924

93.3034

Parámetros de Diseño: Codebook generado a partir de 10 000 números aleatorios entre [0,100] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.5.

0.8079	24.6809	49.7750	75.2656
2.3129	26.1701	51.2073	76.9269
3.7602	27.6393	52.8233	78.6187
5.2310	29.0834	54.3330	80.1409
6.5464	30.5472	55.8132	81.6865
7.8735	32.0718	57.2851	83.2275
9.4892	33.7219	58.9484	84.5998
11.1147	35.2377	60.7360	86.0962
12.6692	36.8388	62.3360	87.5530
14.1004	38.5502	63.8672	88.9641
15.4717	40.2053	65.3980	90.5140
16.9208	41.6815	66.9560	92.1249
18.4530	43.3233	68.7120	93.8223
19.9647	44.7379	70.2787	95.5463
21.4445	46.3973	71.8753	97.2749
23.0992	48.0660	73.5789	99.0542

Parámetros de Diseño: Codebook generado a partir de 1 000 números aleatorios entre [0,10] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.5.

0.6129

4.3358

1.7599

6.1619

3.0252

8.7248

Parámetros de Diseño: Codebook generado a partir de 10 000 números aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.5.

0.2510

0.7520

Parámetros de Diseño: Codebook generado a partir de 1000 números aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.5.

0.2456

0.7444

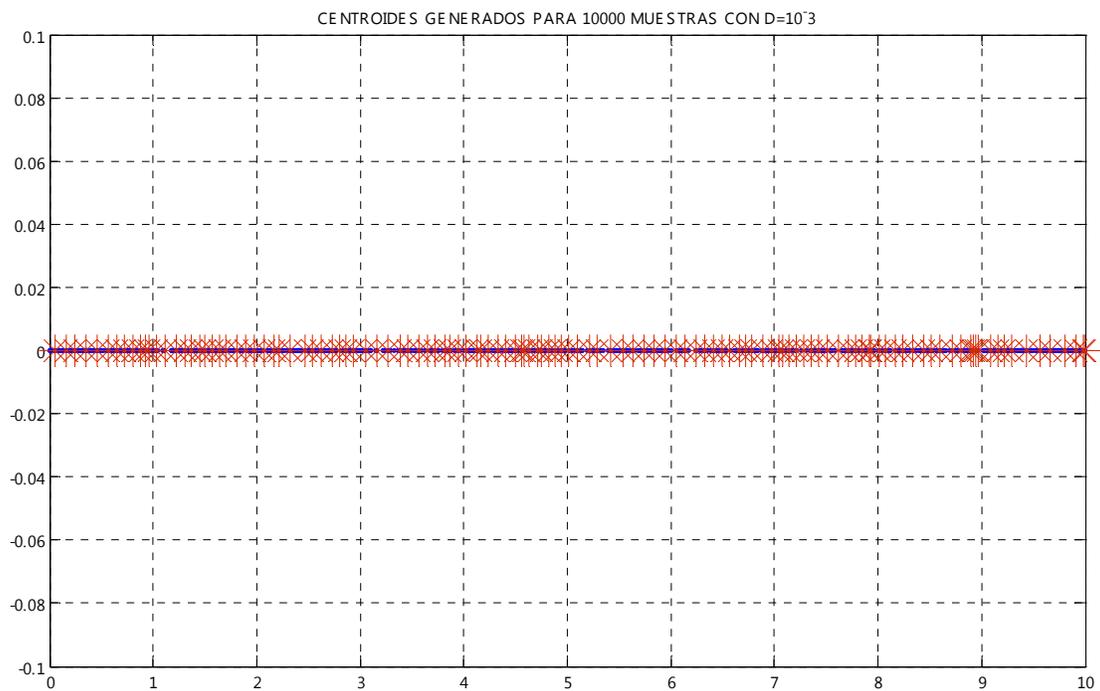
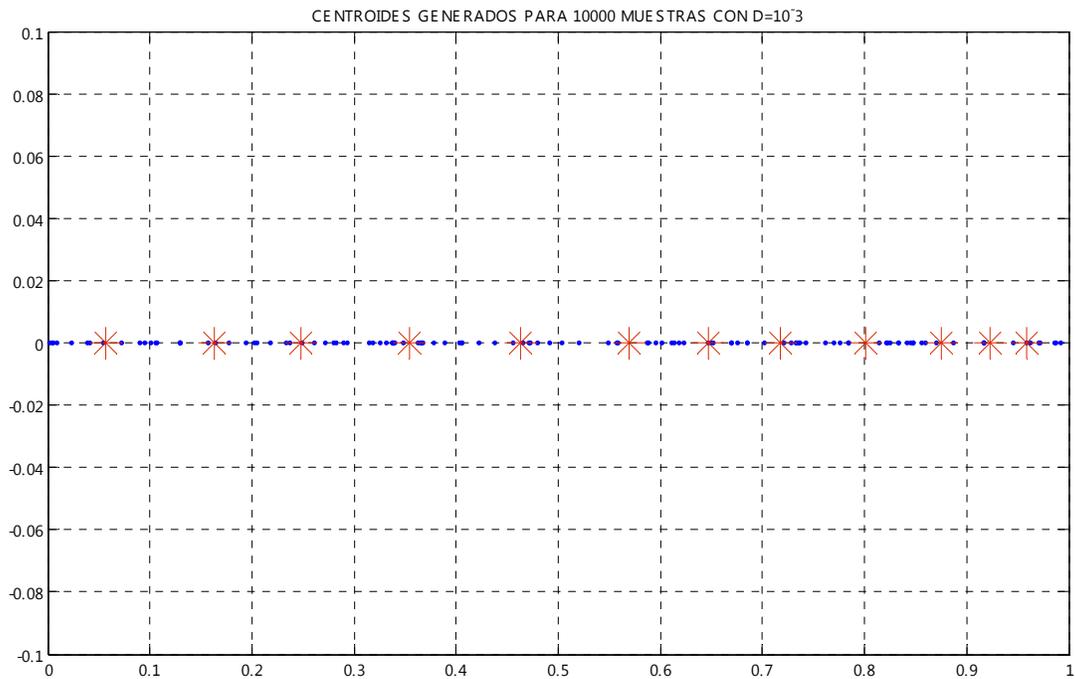
Parámetros de Diseño: Codebook generado a partir de 100 números aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.5.

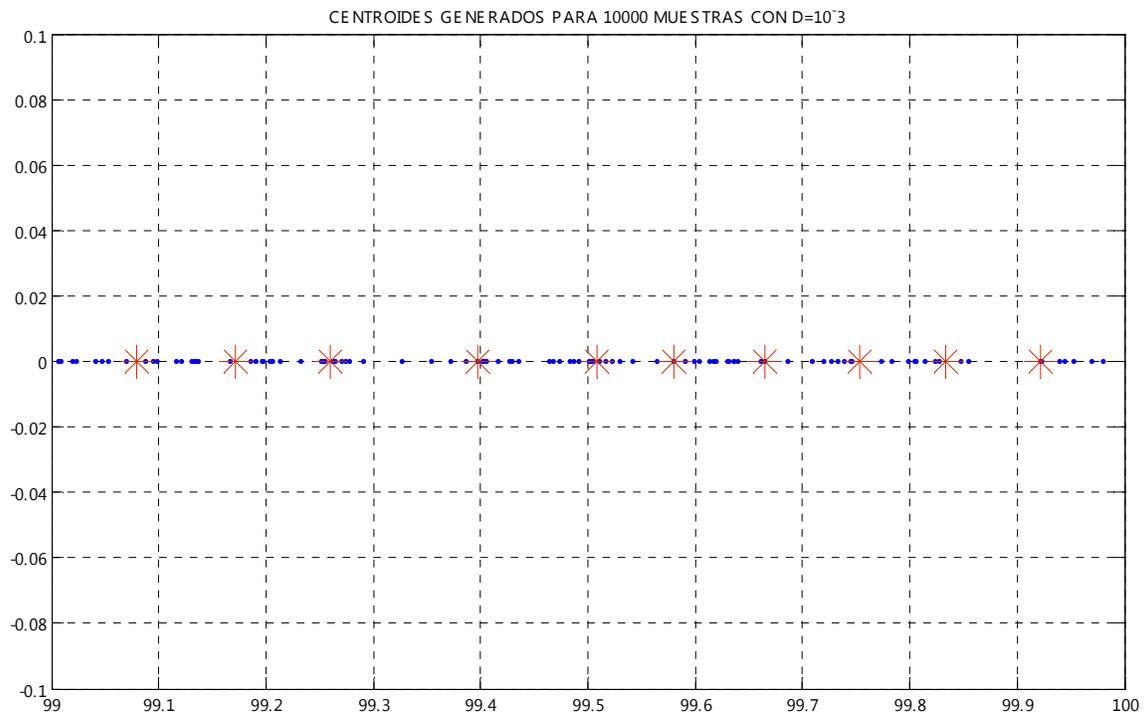
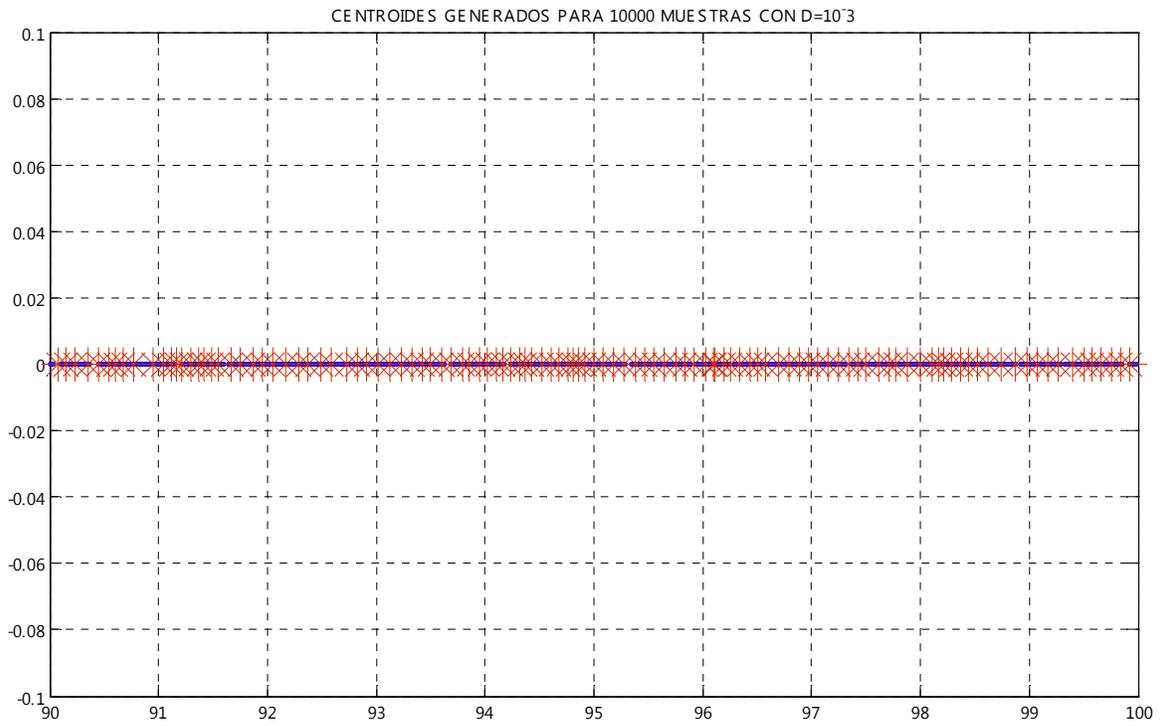
0.2459

0.8009

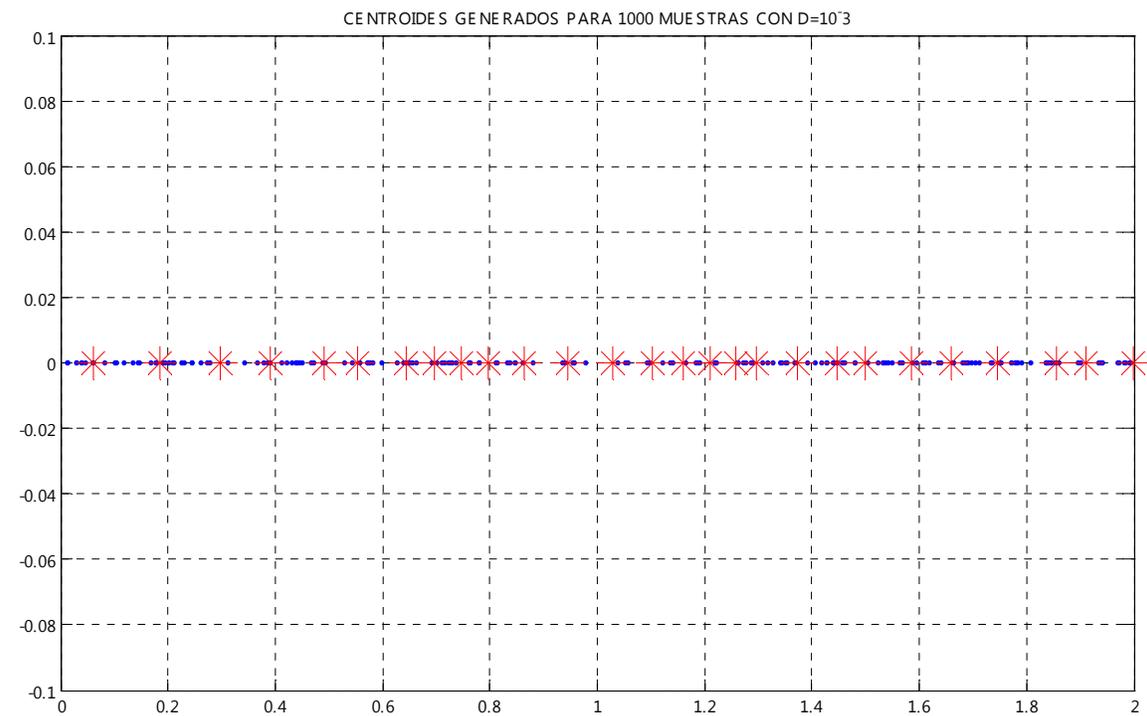
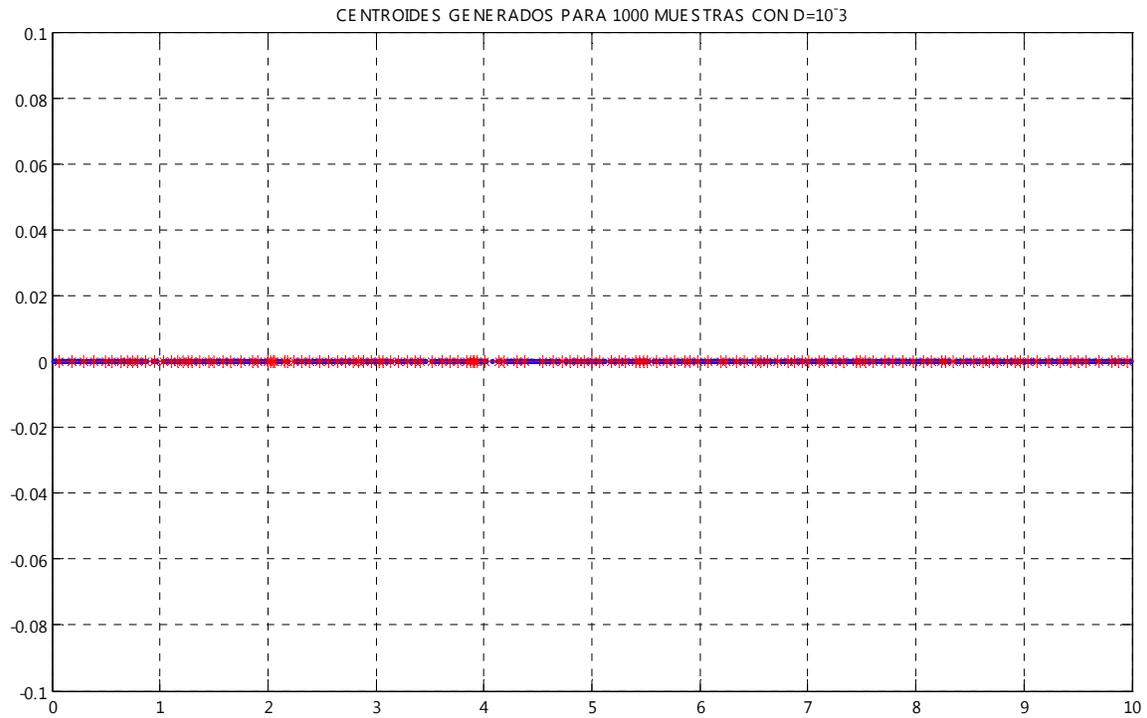
ANEXO 2: ALGORITMO LGB (DISTRIBUCIÓN UNIFORME)

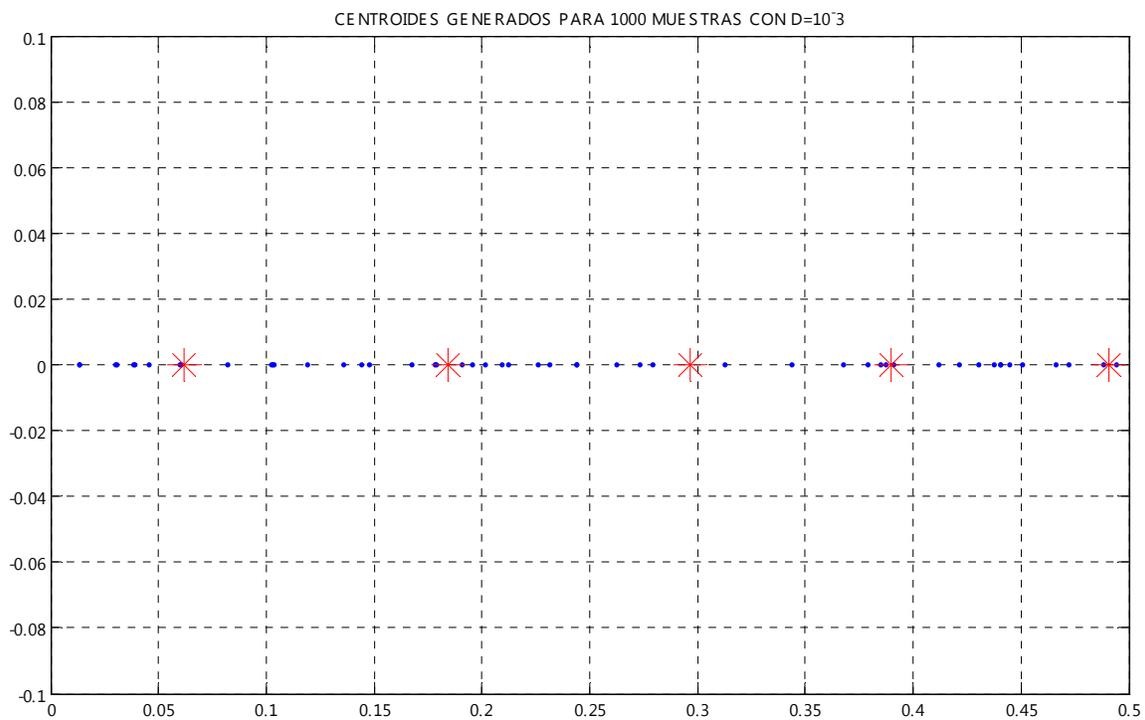
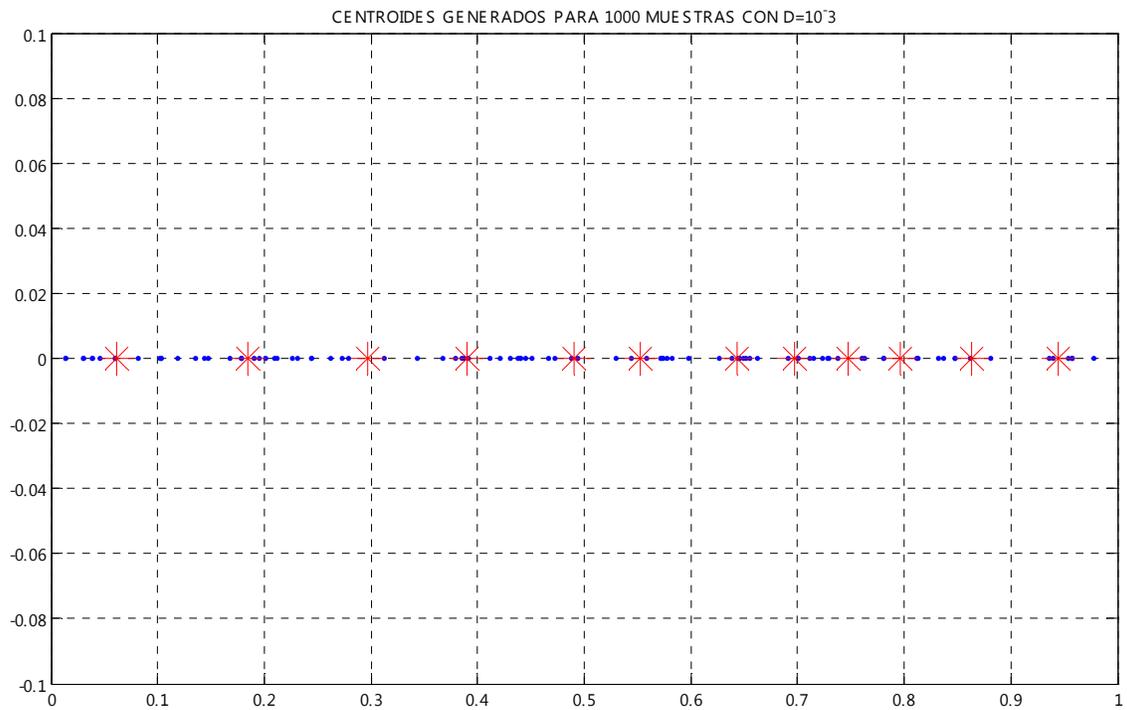
Parámetros de Diseño: Centroides generados a partir de 10 000 números aleatorios entre [0,100] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.003.



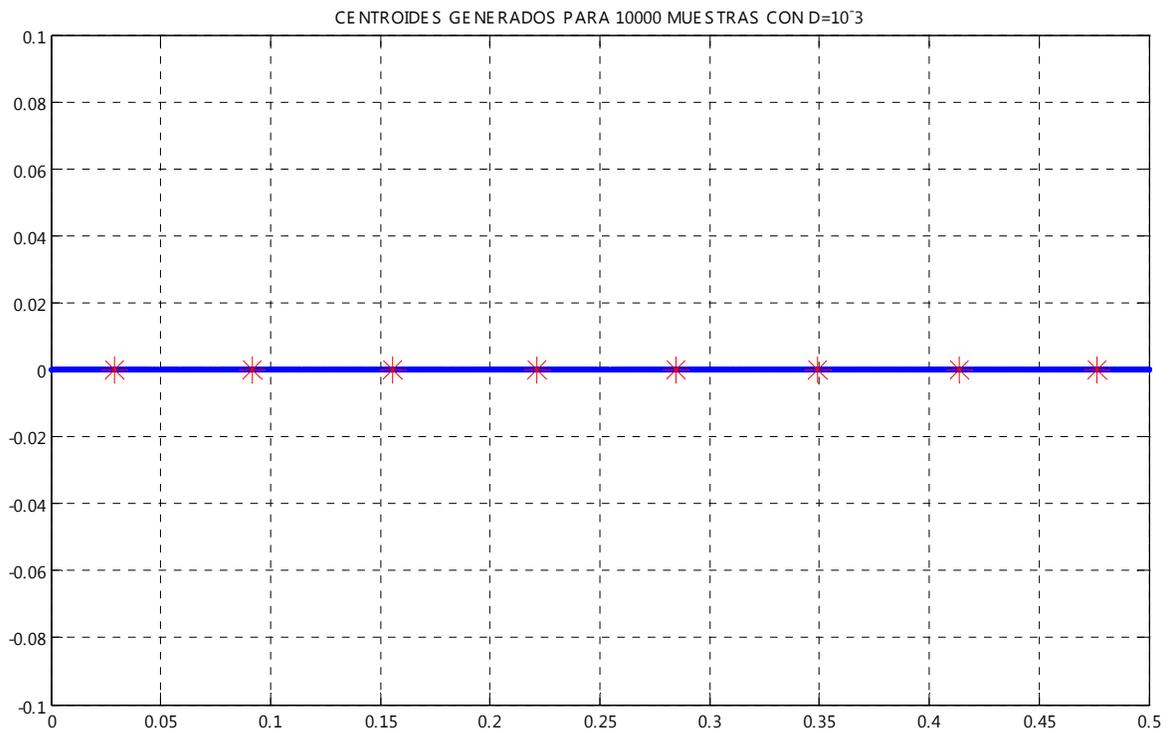
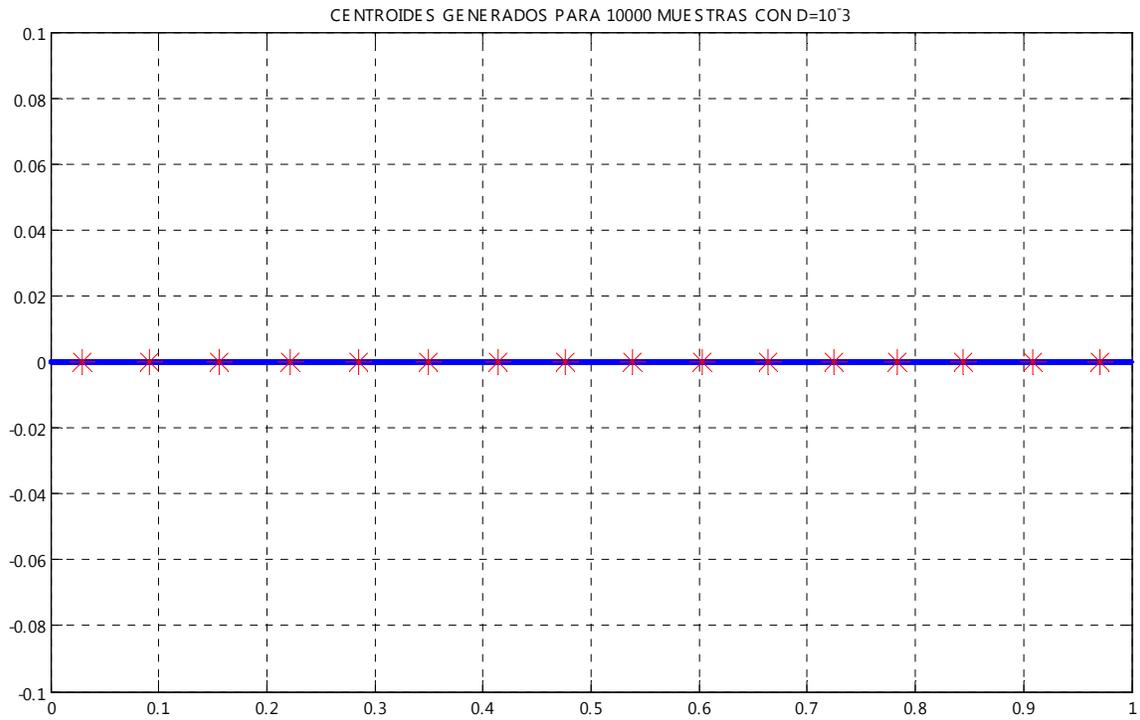


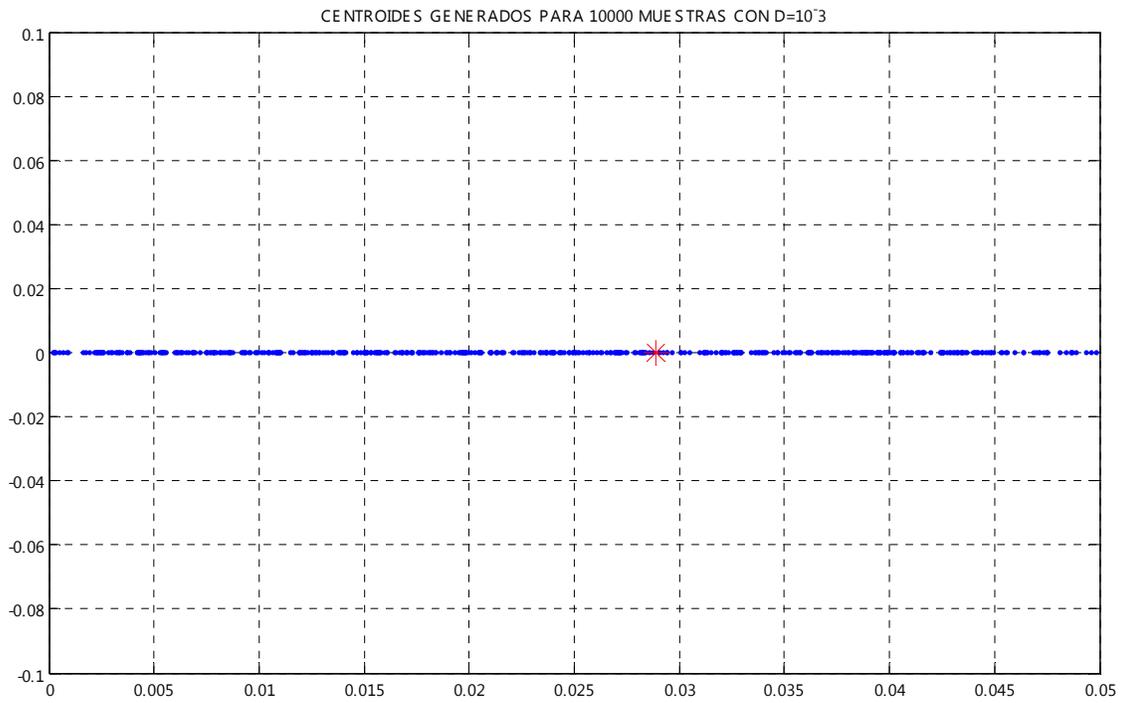
Parámetros de Diseño: Centroides generados a partir de 1000 números aleatorios entre [0,10] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.003.



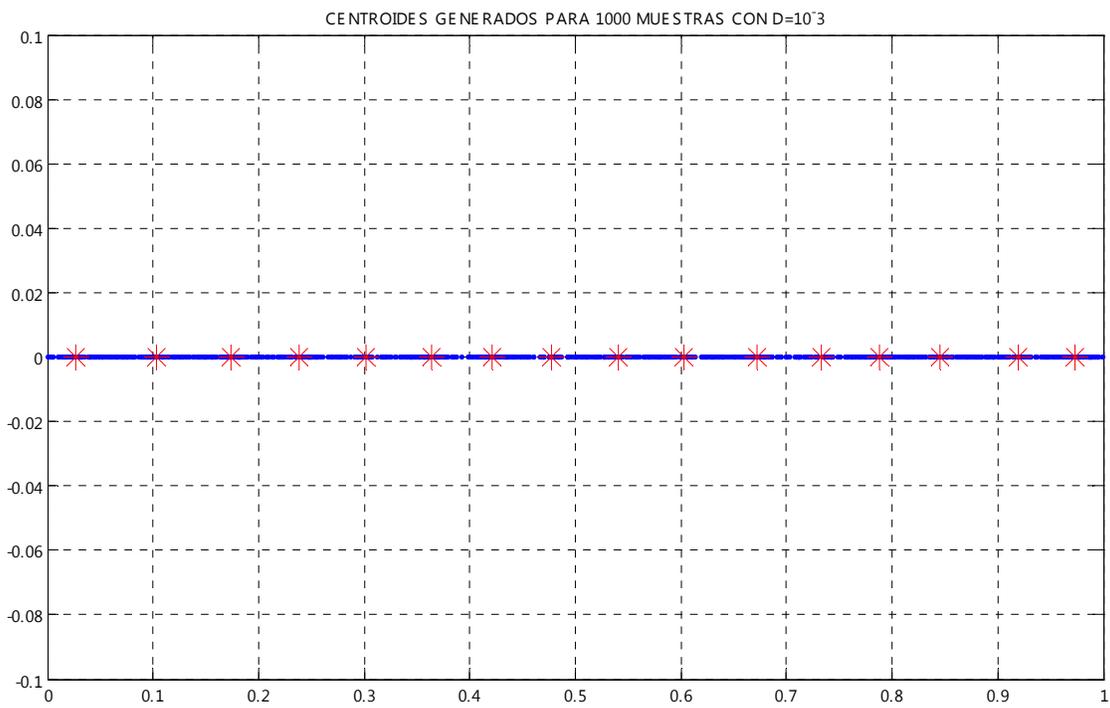


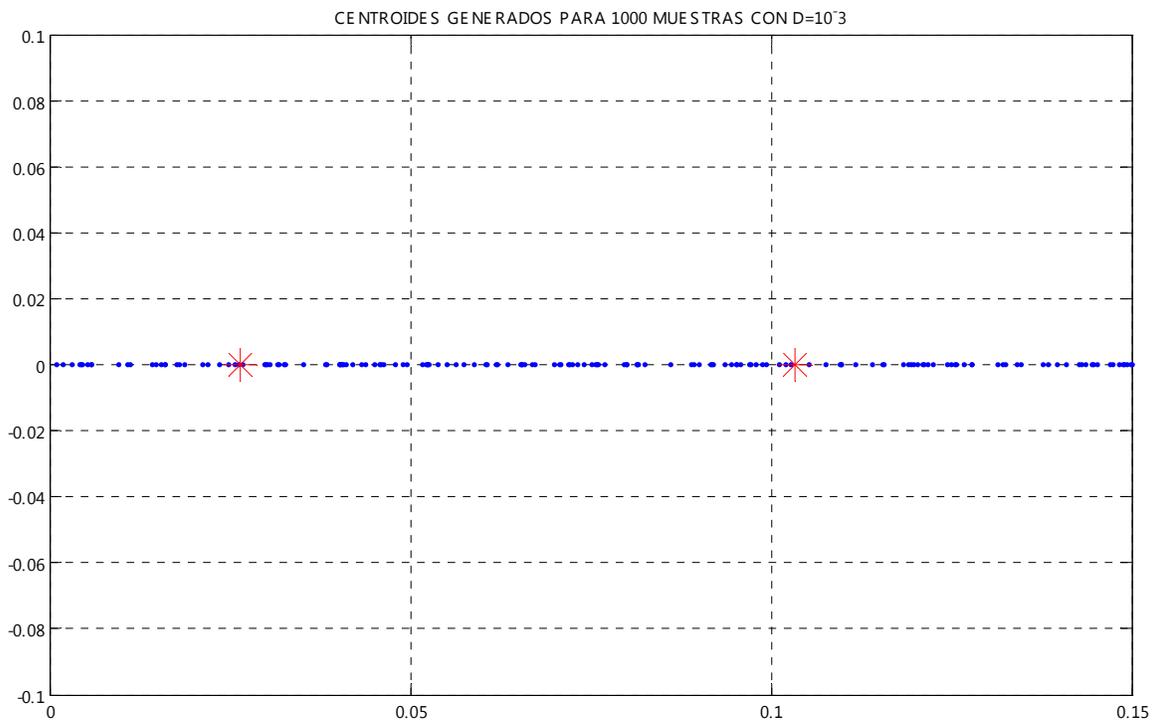
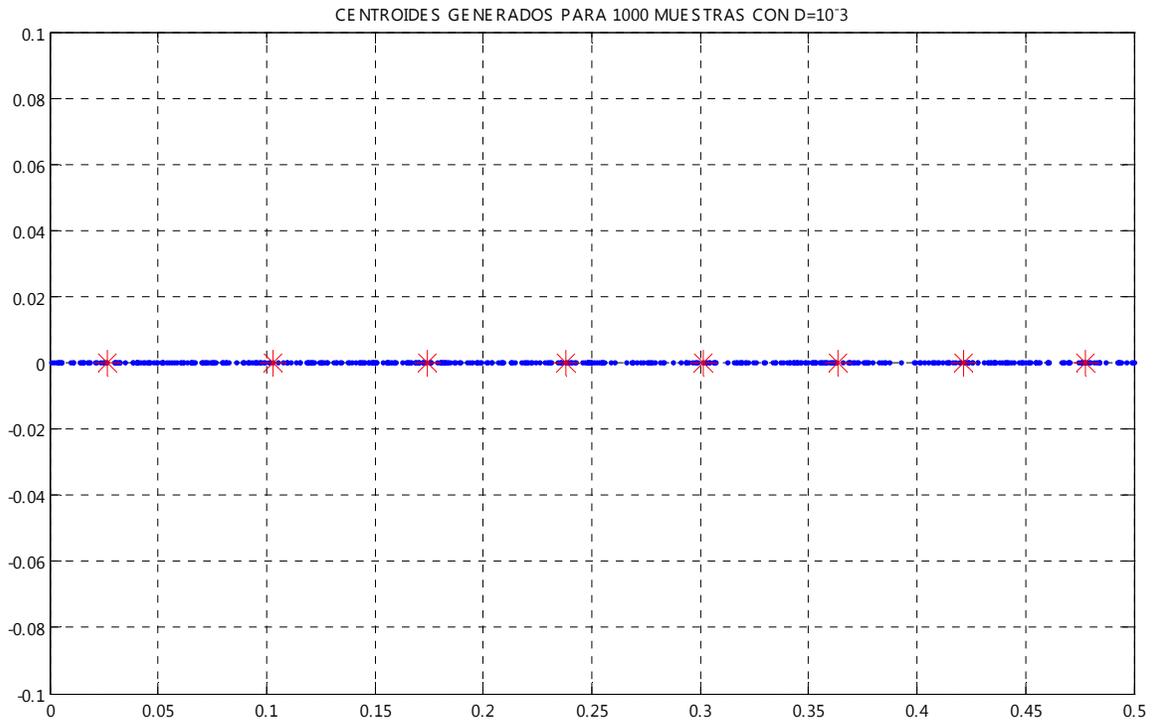
Parámetros de Diseño: Centroides generados a partir de 10 000 números aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.003.



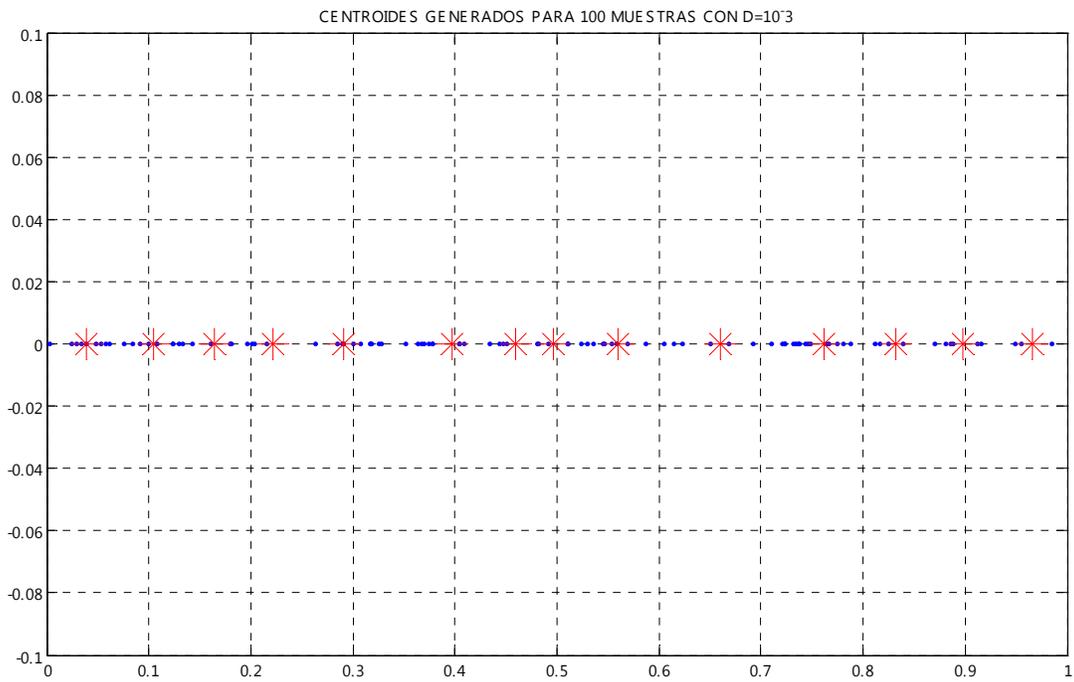


Parámetros de Diseño: Centroides generados a partir de 1 000 números aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.003.

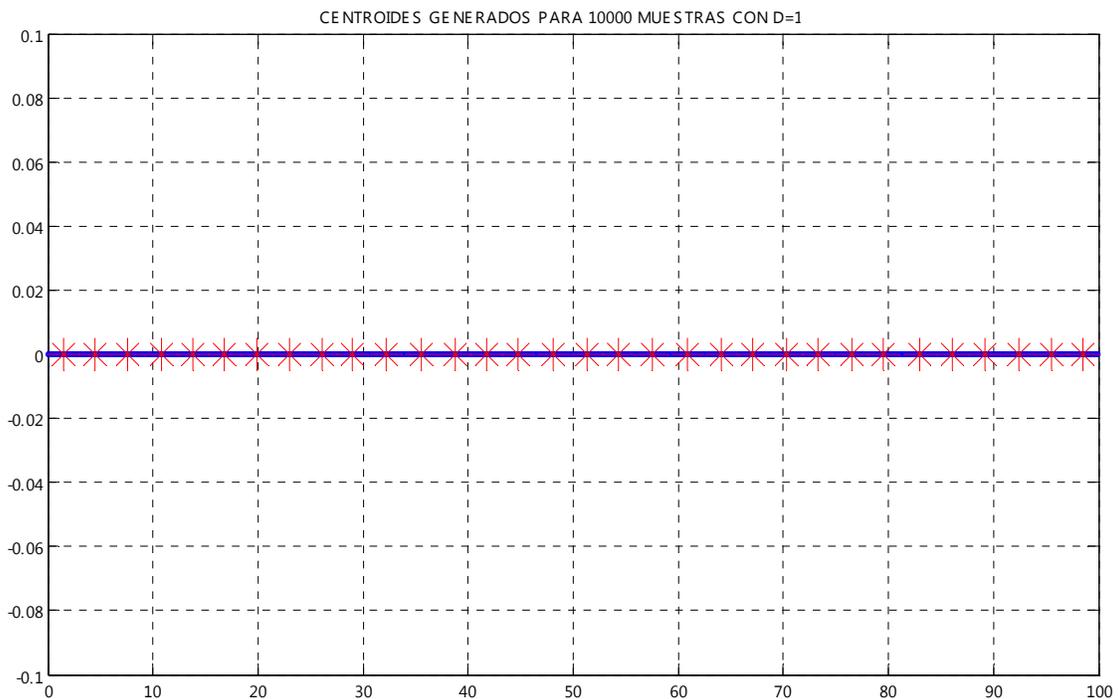


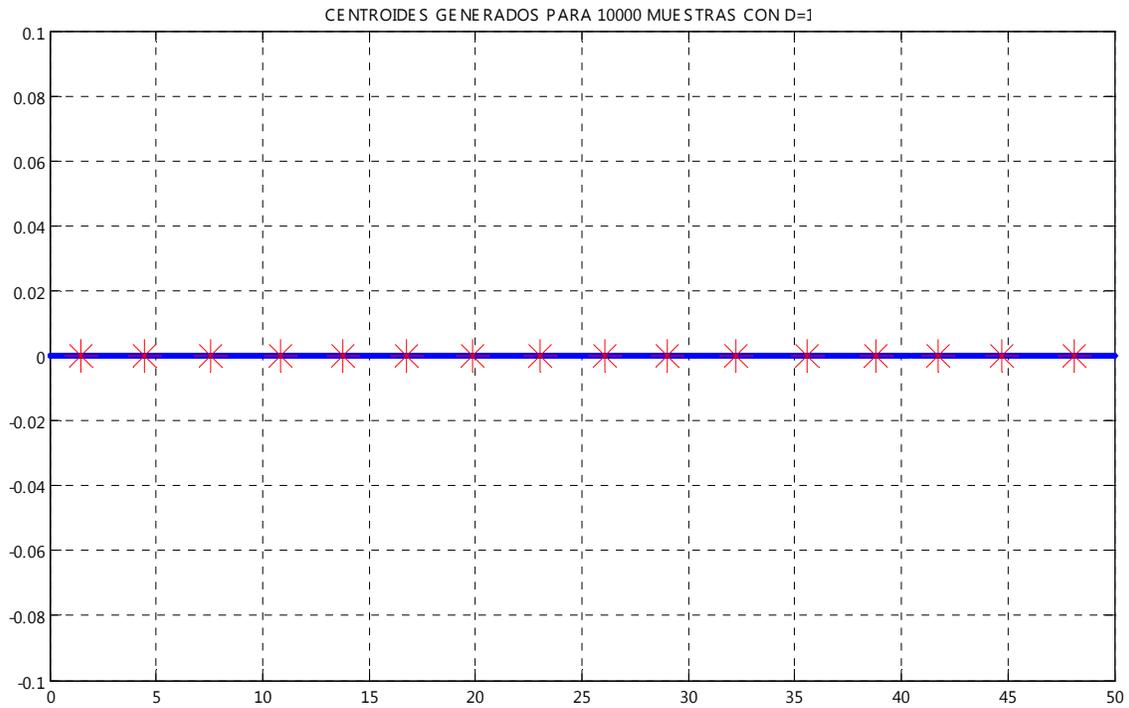


Parámetros de Diseño: Centroides generados a partir de 100 números aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.003.

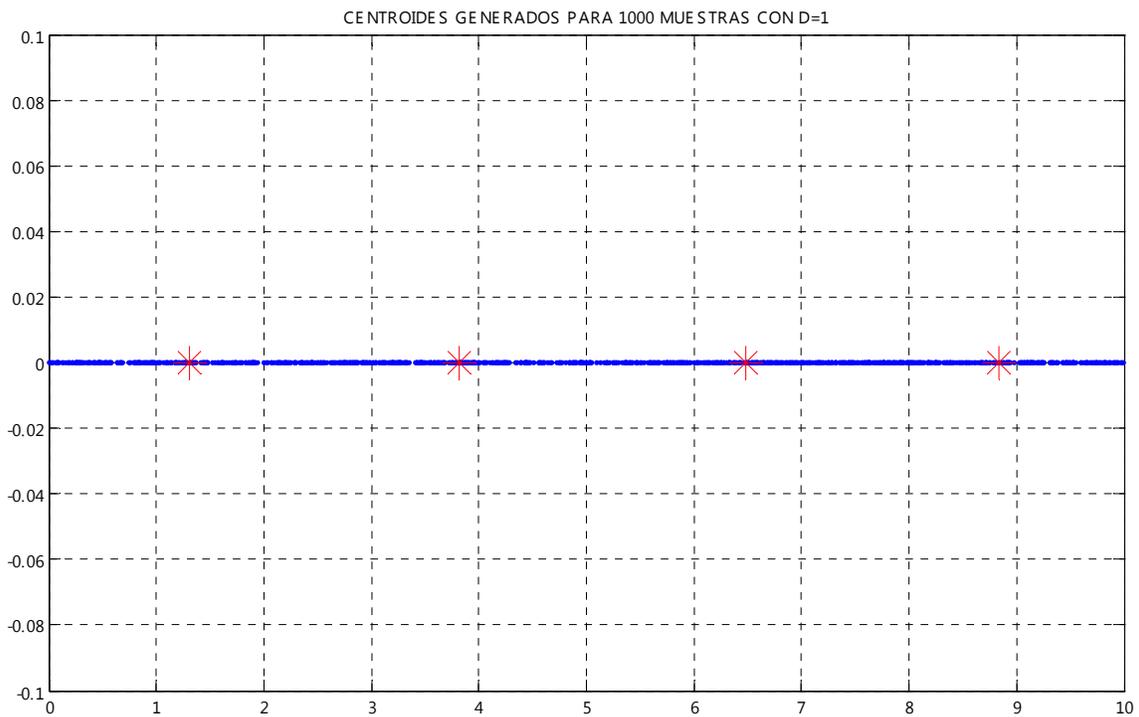


Parámetros de Diseño: Centroides generados a partir de 10 000 números aleatorios entre [0,100] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 1.

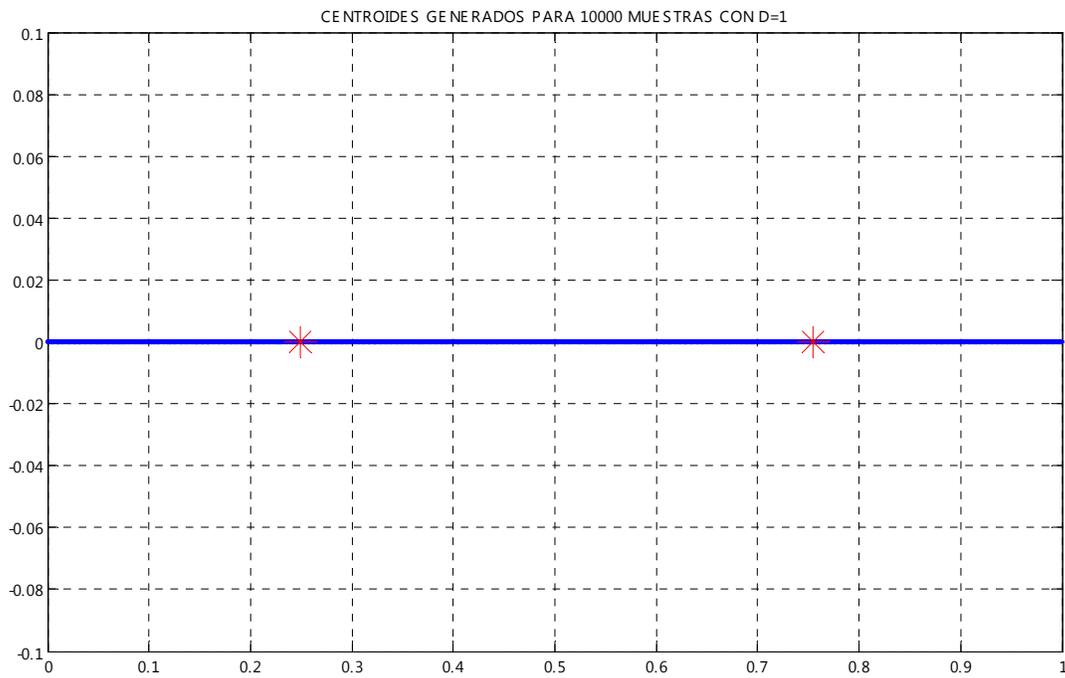




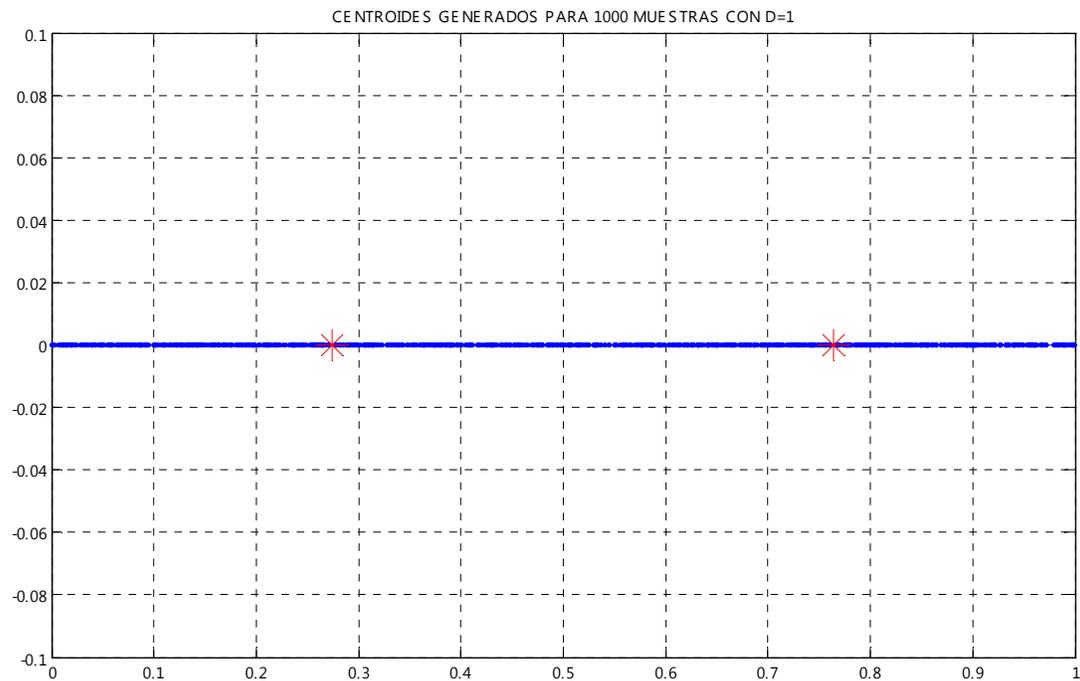
Parámetros de Diseño: Centroides generados a partir de 1 000 números aleatorios entre [0,10] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 1.



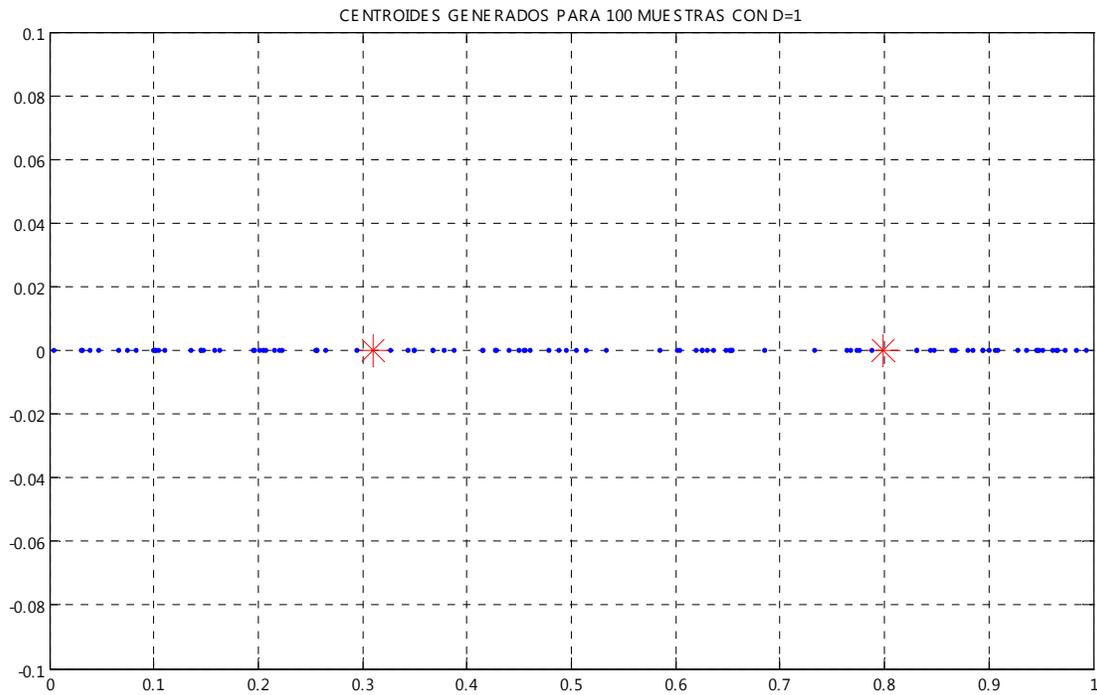
Parámetros de Diseño: Centroides generados a partir de 10 000 números aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 1.



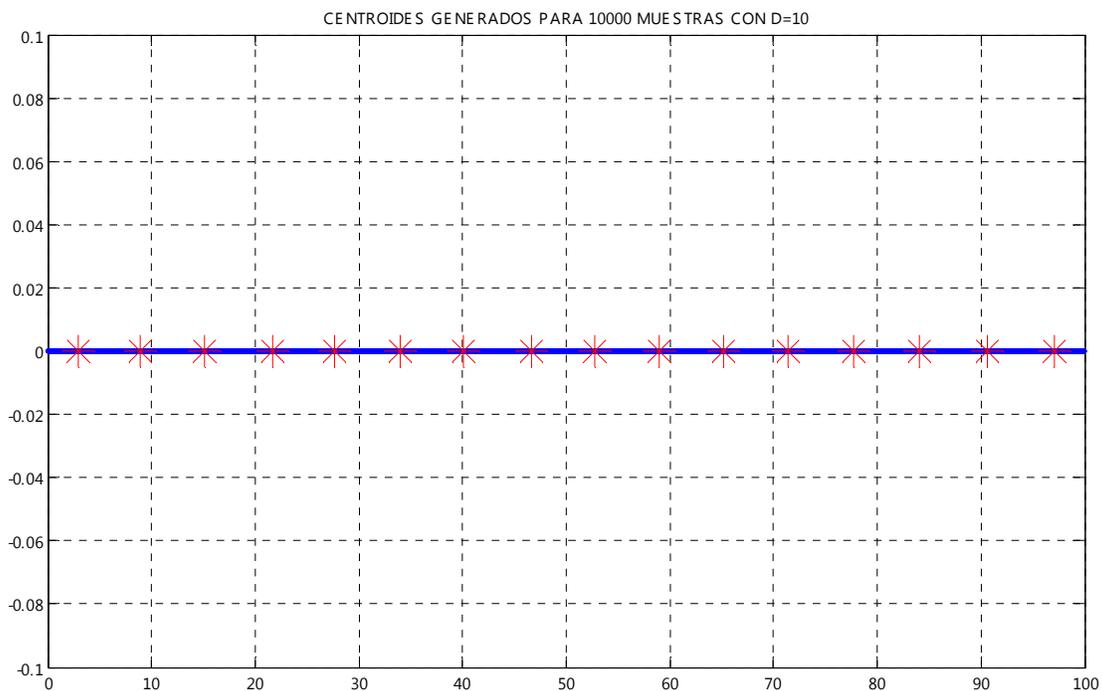
Parámetros de Diseño: Centroides generados a partir de 1 000 números aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 1.



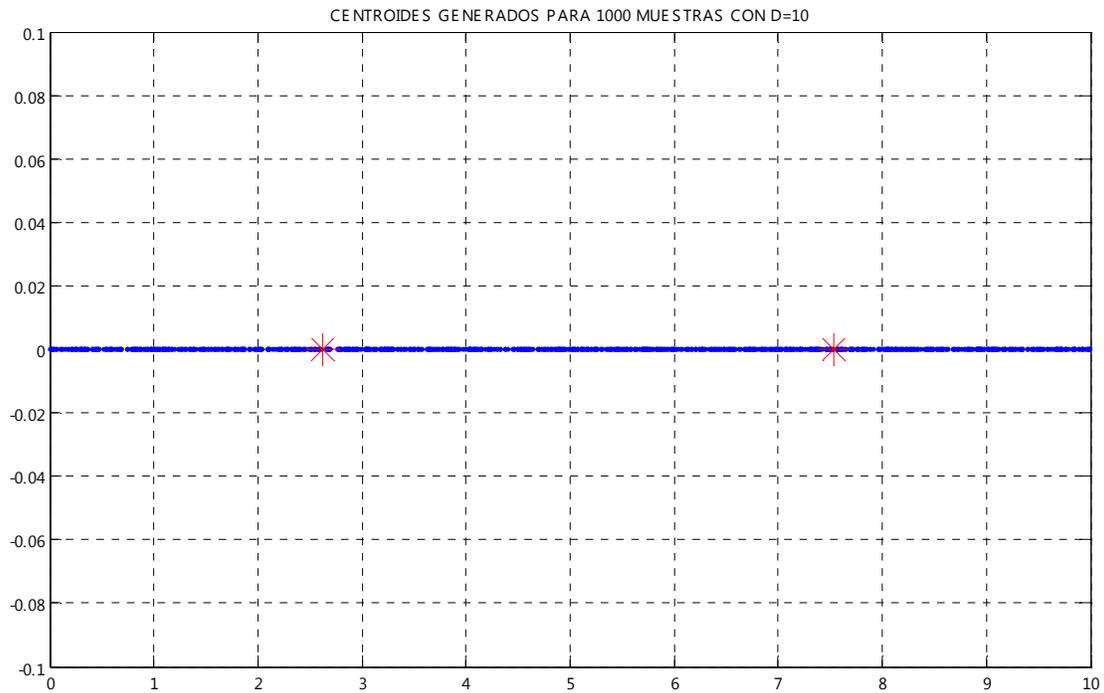
Parámetros de Diseño: Centroides generados a partir de 100 números aleatorios entre $[0,1]$ con una distribución uniforme y tomando como parámetro fundamental una distorsión de 1.



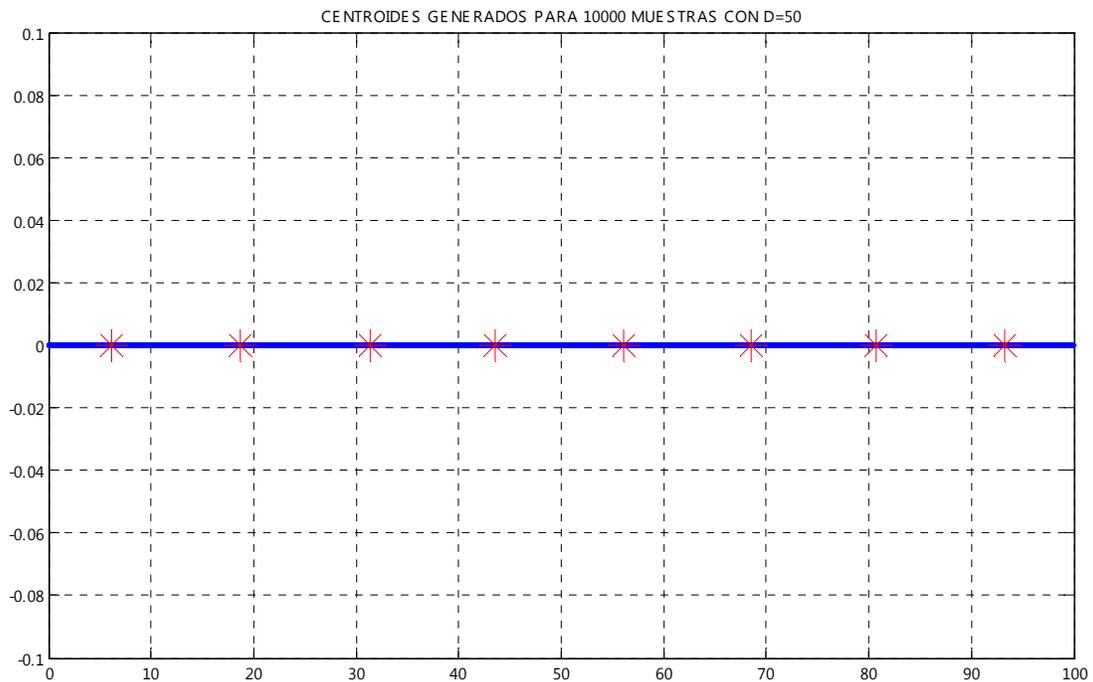
Parámetros de Diseño: Centroides generados a partir de 10 000 números aleatorios entre $[0,100]$ con una distribución uniforme y tomando como parámetro fundamental una distorsión de 10.



Parámetros de Diseño: Centroides generados a partir de 1 000 números aleatorios entre [0,10] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 10.



Parámetros de Diseño: Centroides generados a partir de 10 000 números aleatorios entre [0,100] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 50.



ANEXO 3: CUANTIZACIÓN ESCALAR (DISTRIBUCIÓN NORMAL)

Parámetros de Diseño: Codebook generado a partir de 1 000 números aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.003.

-2.5997	-0.7788	0.5622	1.7621
-2.4167	-0.6755	0.6534	1.8863
-2.3040	-0.5649	0.7394	2.0157
-2.1865	-0.4601	0.8350	2.1524
-1.6524	-0.3045	0.9572	2.2368
-1.4440	-0.1227	1.1279	2.3410
-1.2786	0.0585	1.3013	2.6289
-1.1291	0.2449	1.5009	2.7110
-1.0016	0.3607	1.6218	
-0.8857	0.4671	1.6759	

Parámetros de Diseño: Codebook generado a partir de 1 000 números aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.1.

-1.7868	-0.6191	0.1465	0.9562
-1.0768	-0.2124	0.5224	1.6646

Parámetros de Diseño: Codebook generado a partir de 1 000 números aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.5.

-0.8112	0.7009
---------	--------

Parámetros de Diseño: Codebook generado a partir de 500 números aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.003.

-2.4373	-1.1250	0.0443	1.5785
-2.2994	-1.0125	0.2642	1.7568
-2.0532	-0.9202	0.4518	1.8744
-1.9083	-0.8043	0.6302	1.9574
-1.6744	-0.6768	0.7924	1.9574
-1.5105	-0.5648	0.9393	2.1242
-1.4013	-0.4680	1.0718	2.2609
-1.3394	-0.3060	1.2686	
-1.2400	-0.1368	1.4472	

Parámetros de Diseño: Codebook generado a partir de 500 números aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.1.

-1.8209	-0.6211	0.1838	0.9448
-1.0588	-0.2145	0.5410	1.6728

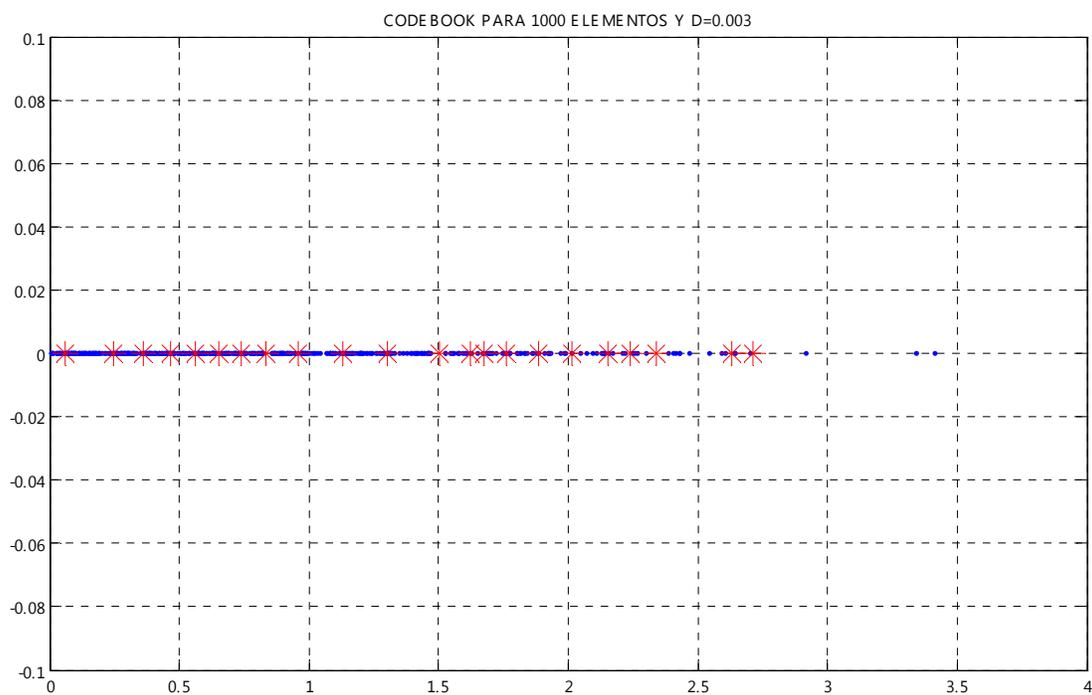
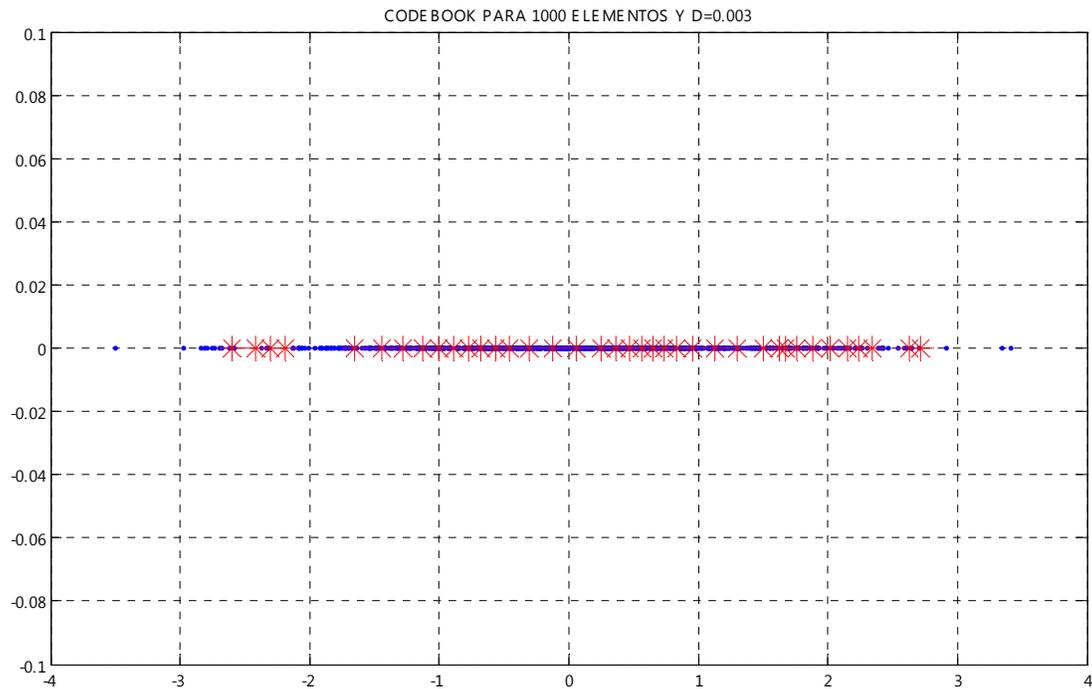
Parámetros de Diseño: Codebook generado a partir de 500 números aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.5.

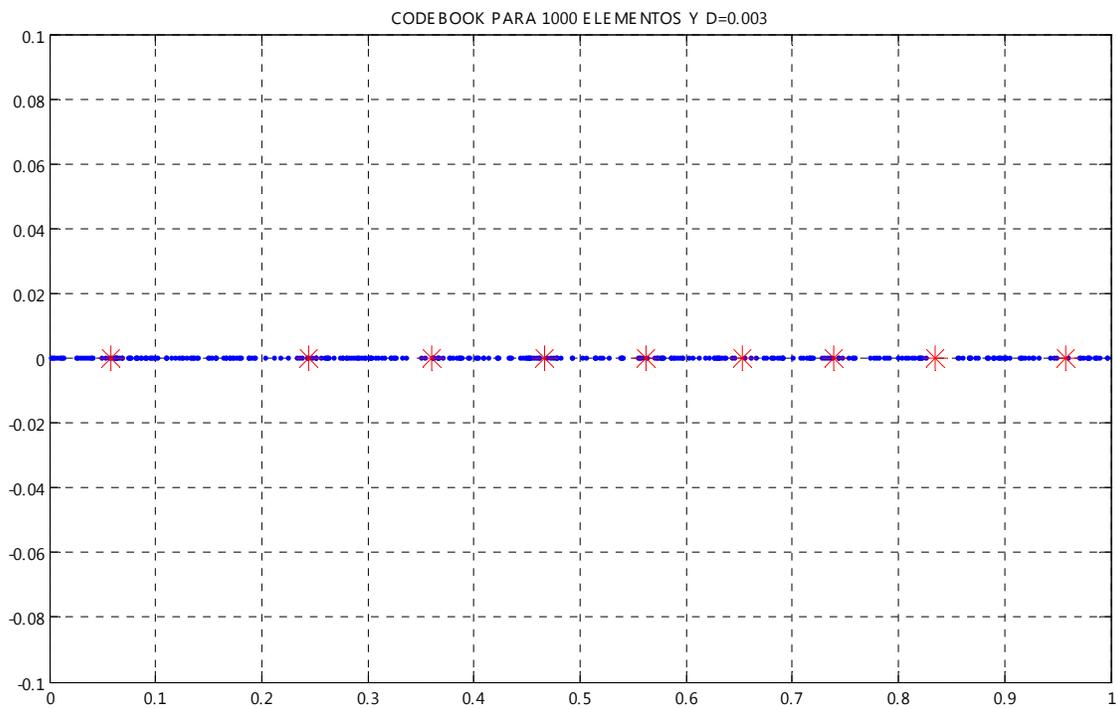
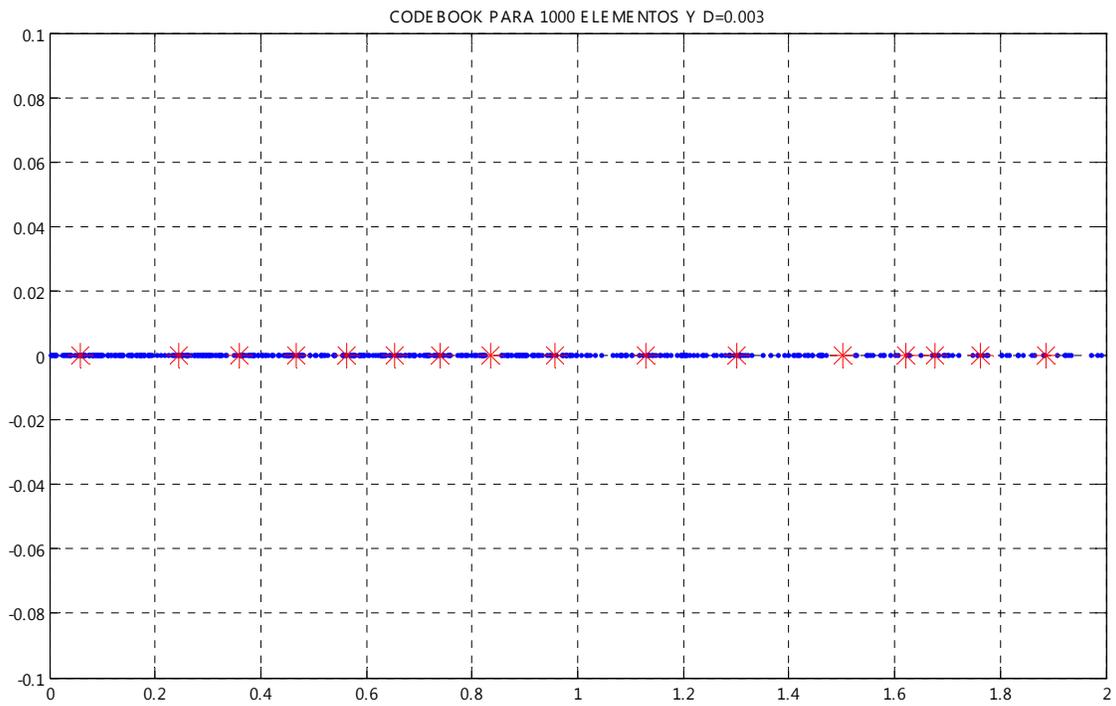
-0.7753

0.8518

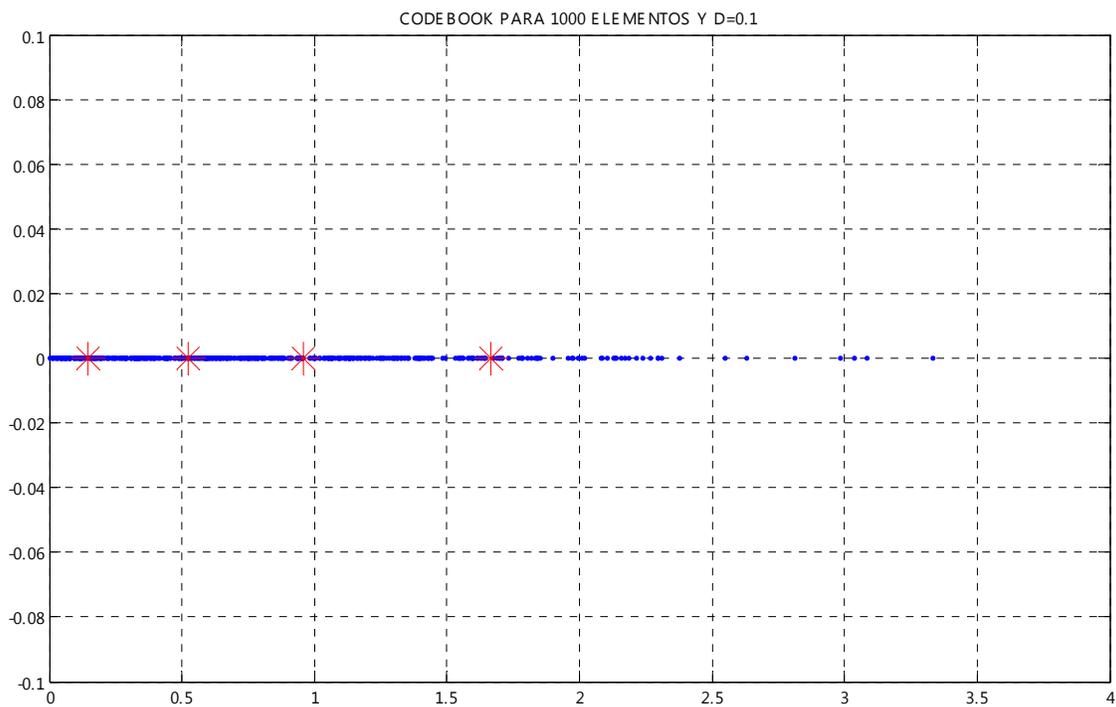
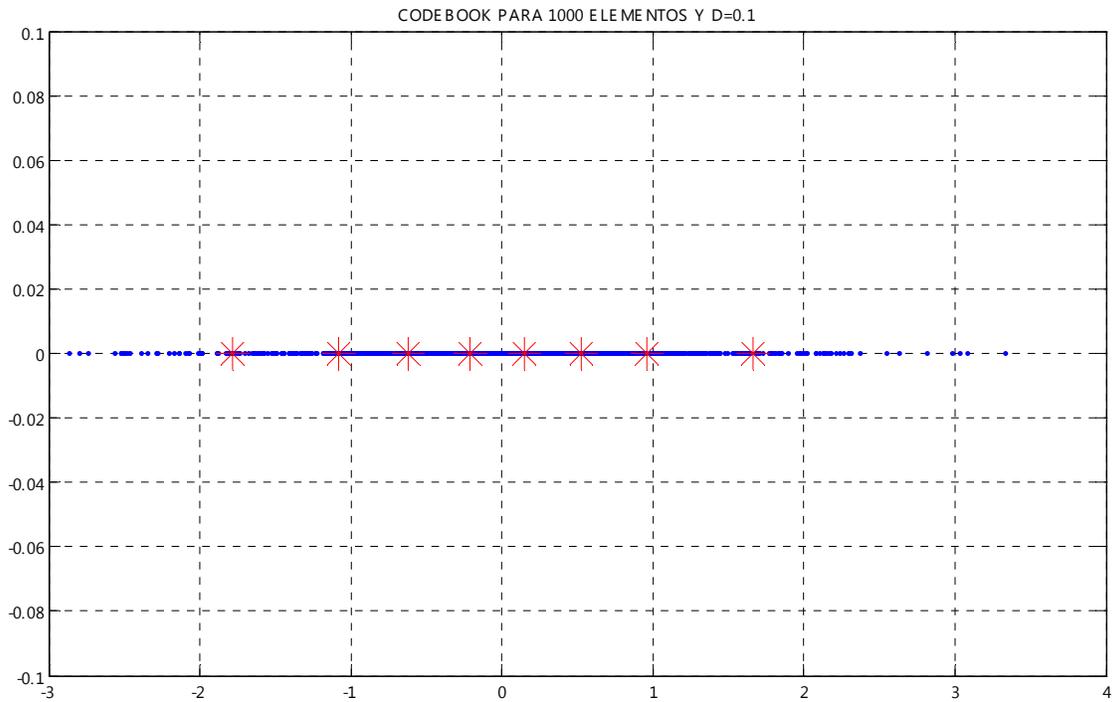
ANEXO 4: ALGORITMO LGB (DISTRIBUCIÓN UNORMAL)

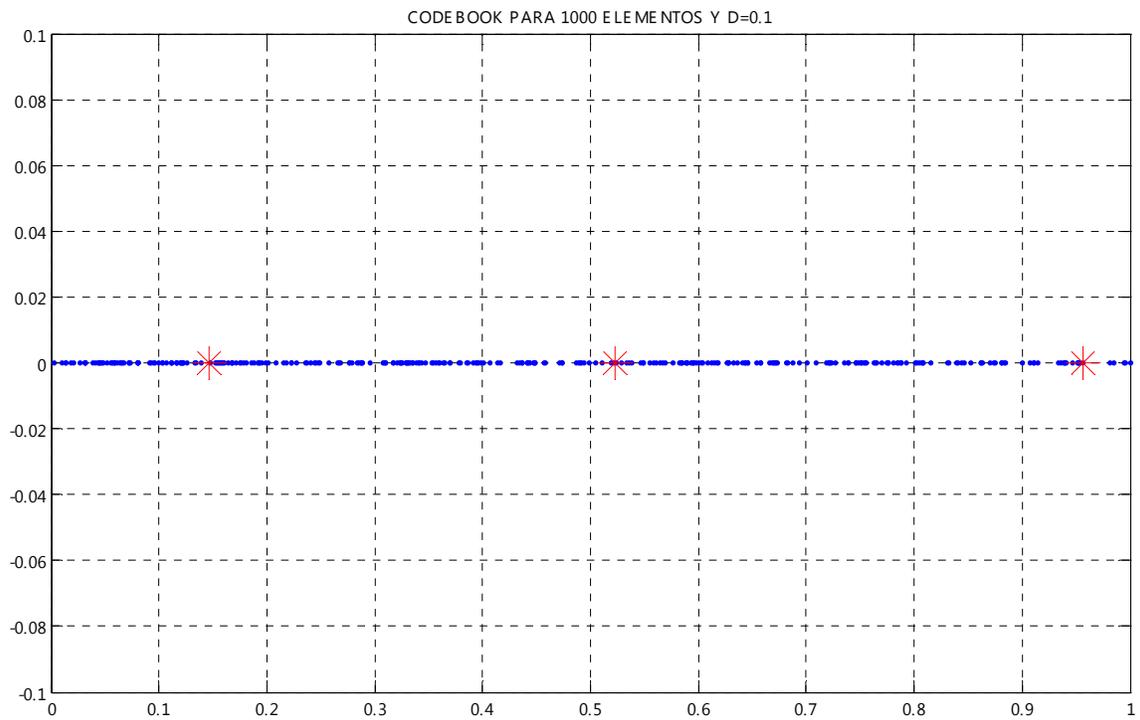
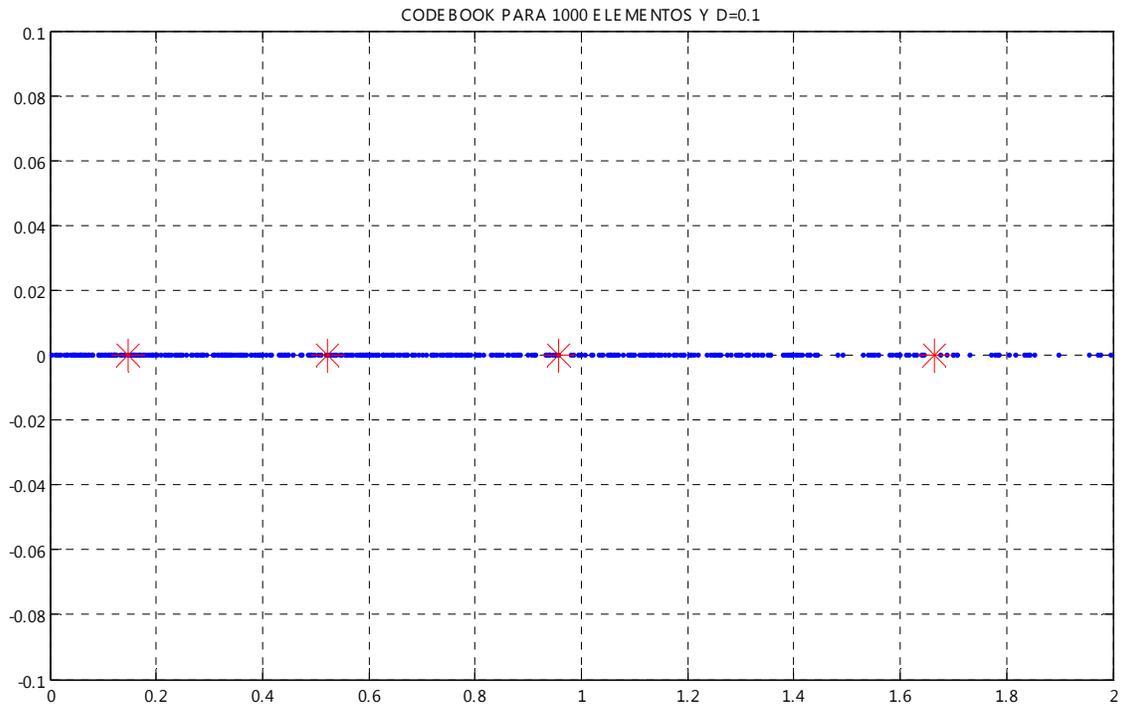
Parámetros de Diseño: Centroides generados a partir de 1 000 números aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.003.



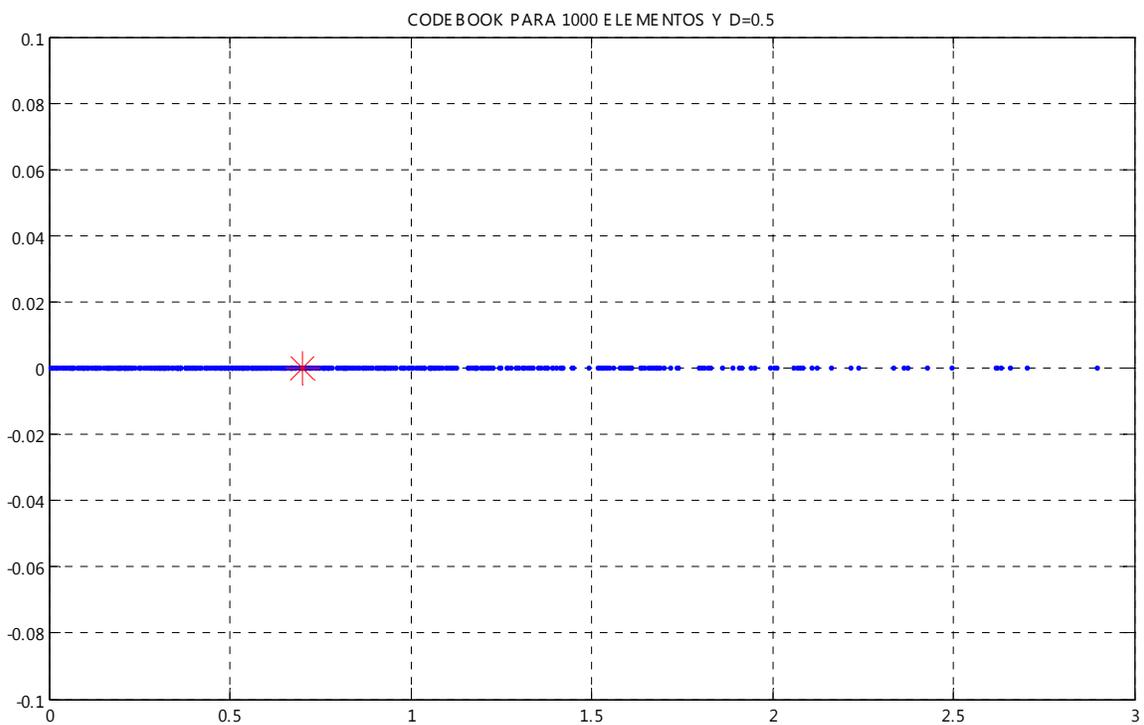
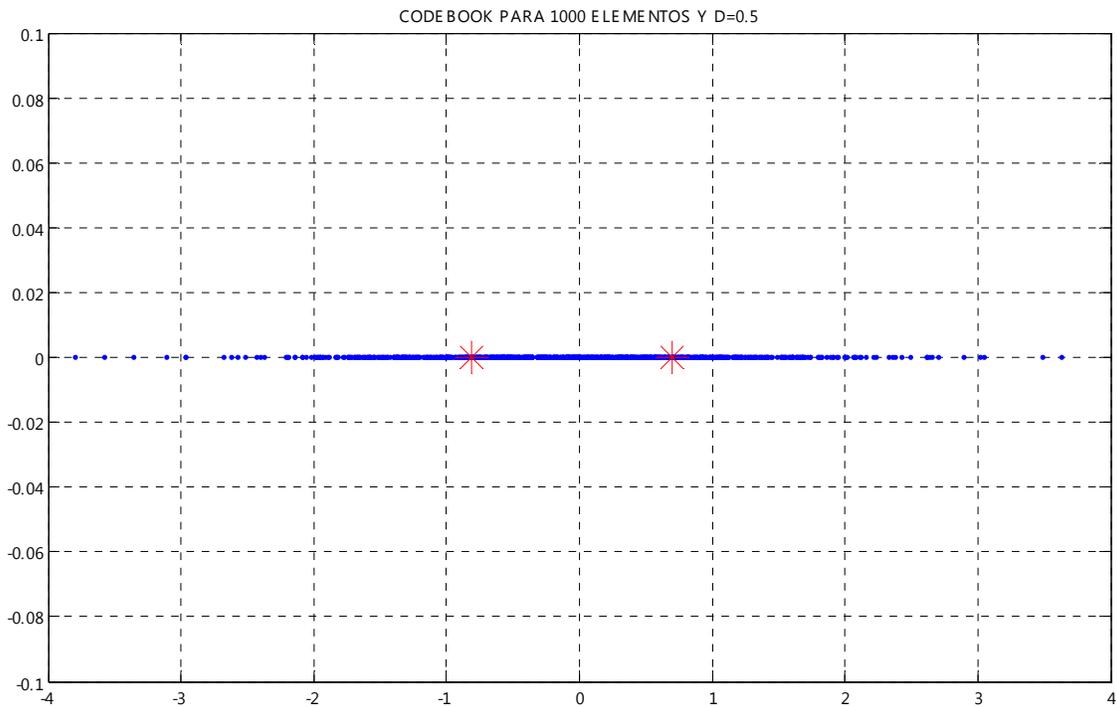


Parámetros de Diseño: Centroides generados a partir de 1 000 números aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.1.

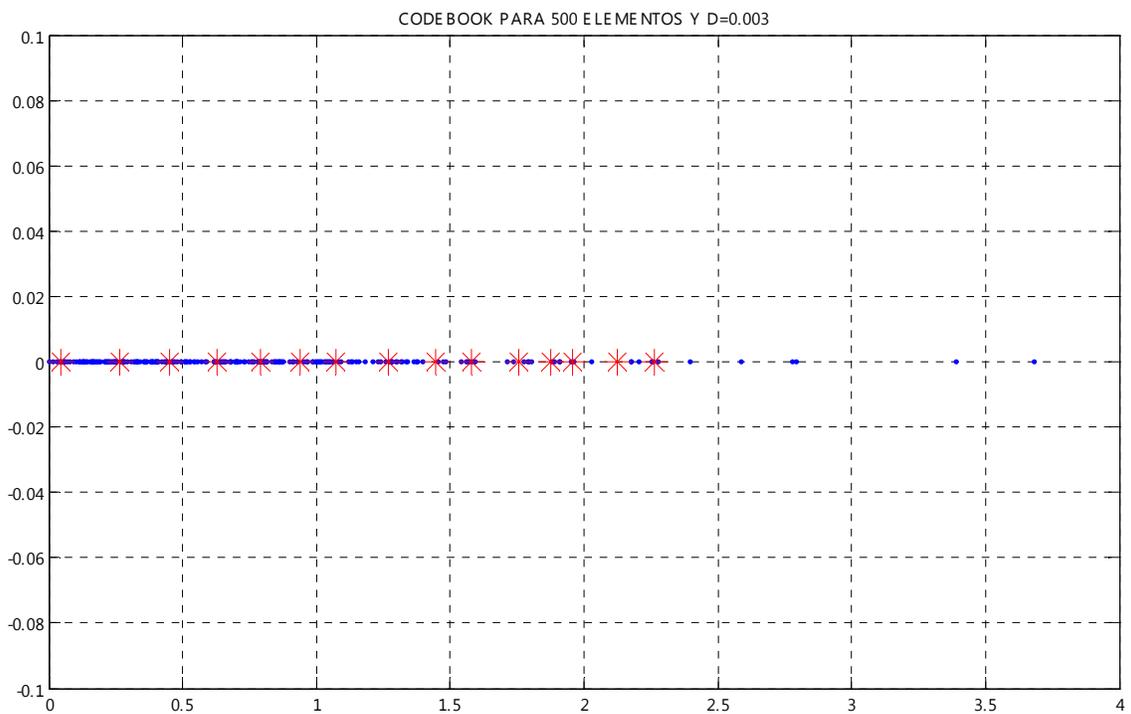
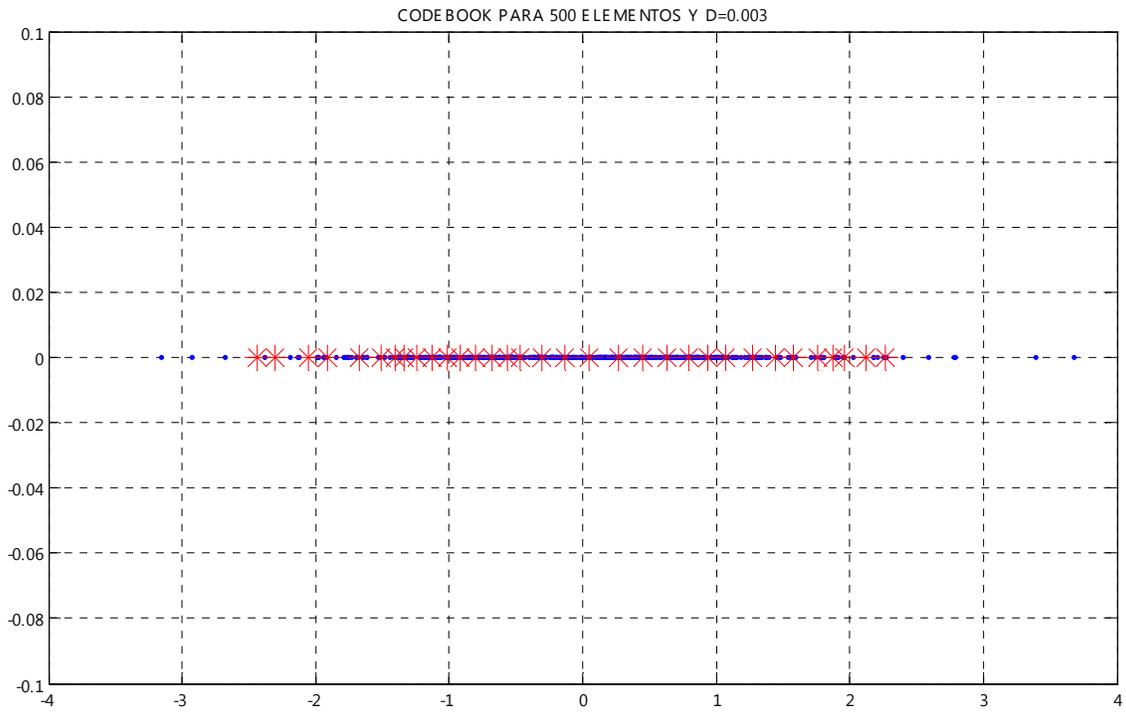


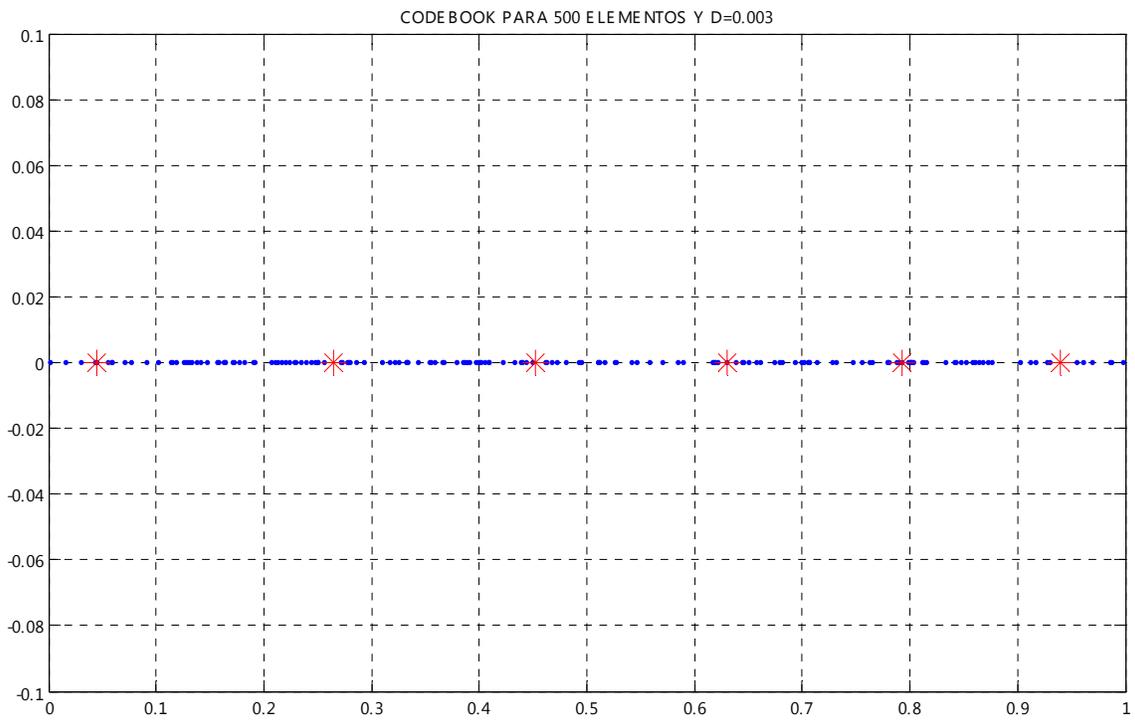
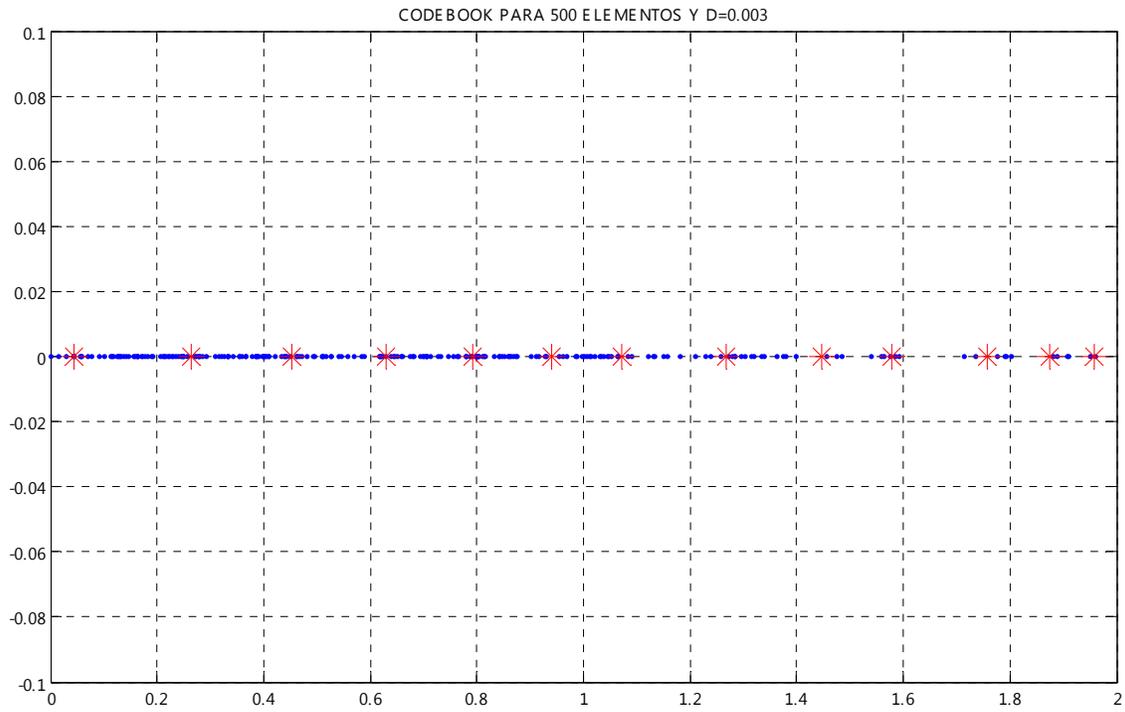


Parámetros de Diseño: Centroides generados a partir de 1 000 números aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.5.

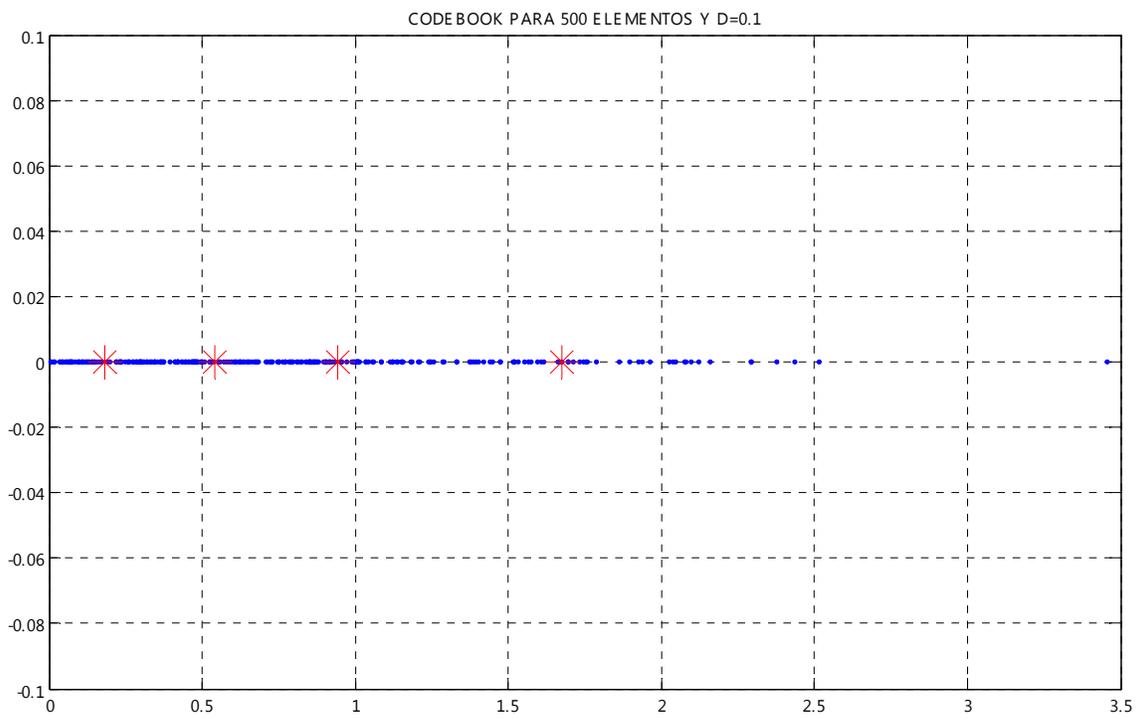
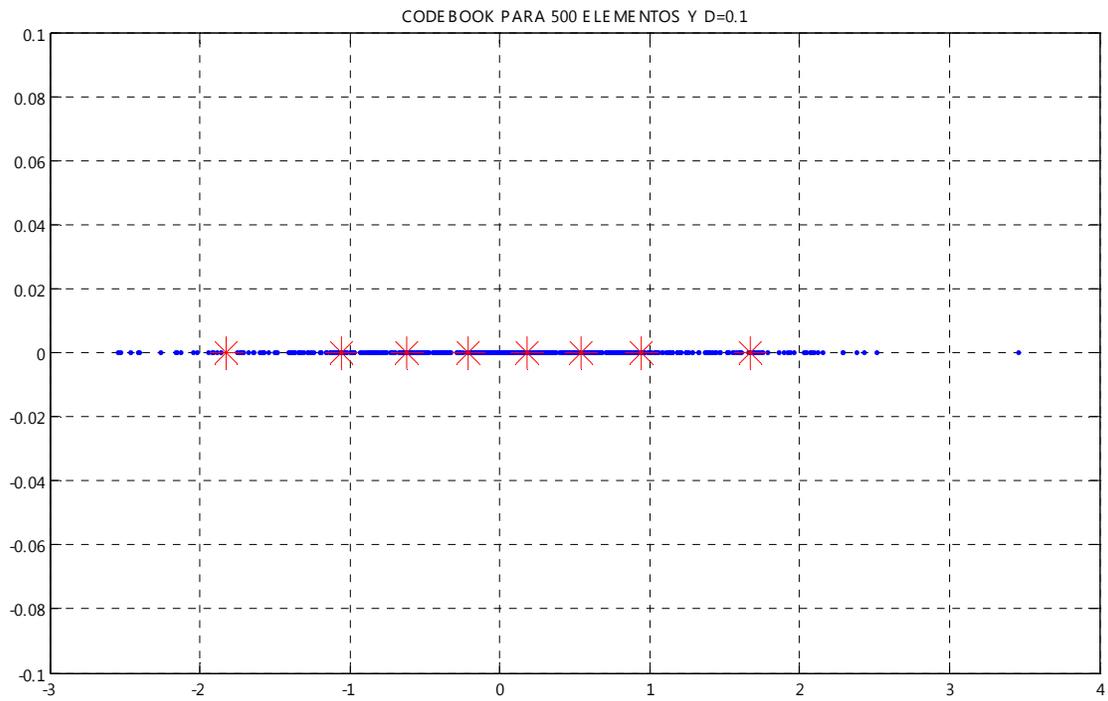


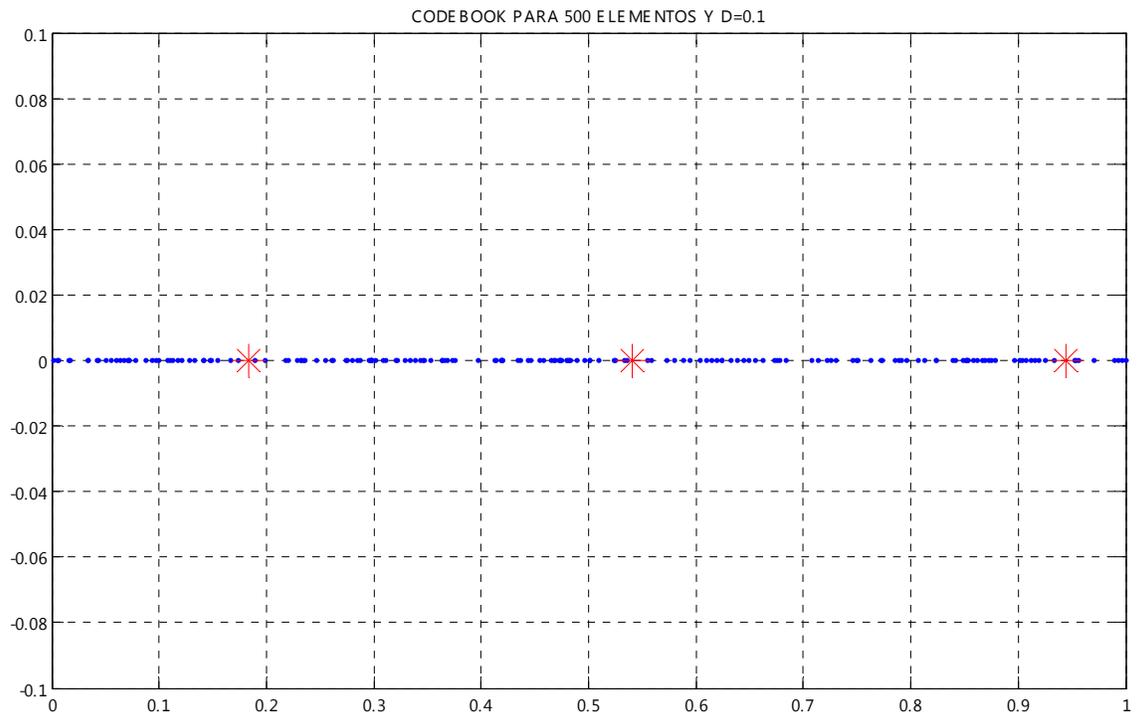
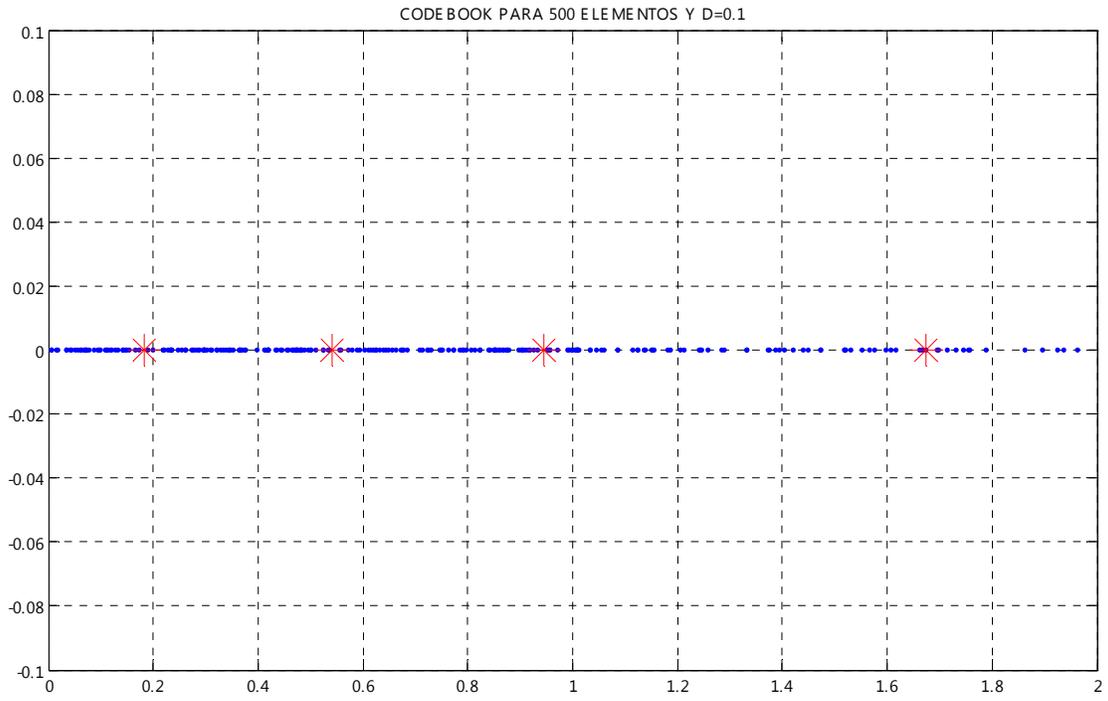
Parámetros de Diseño: Centroides generados a partir de 500 números aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.003.



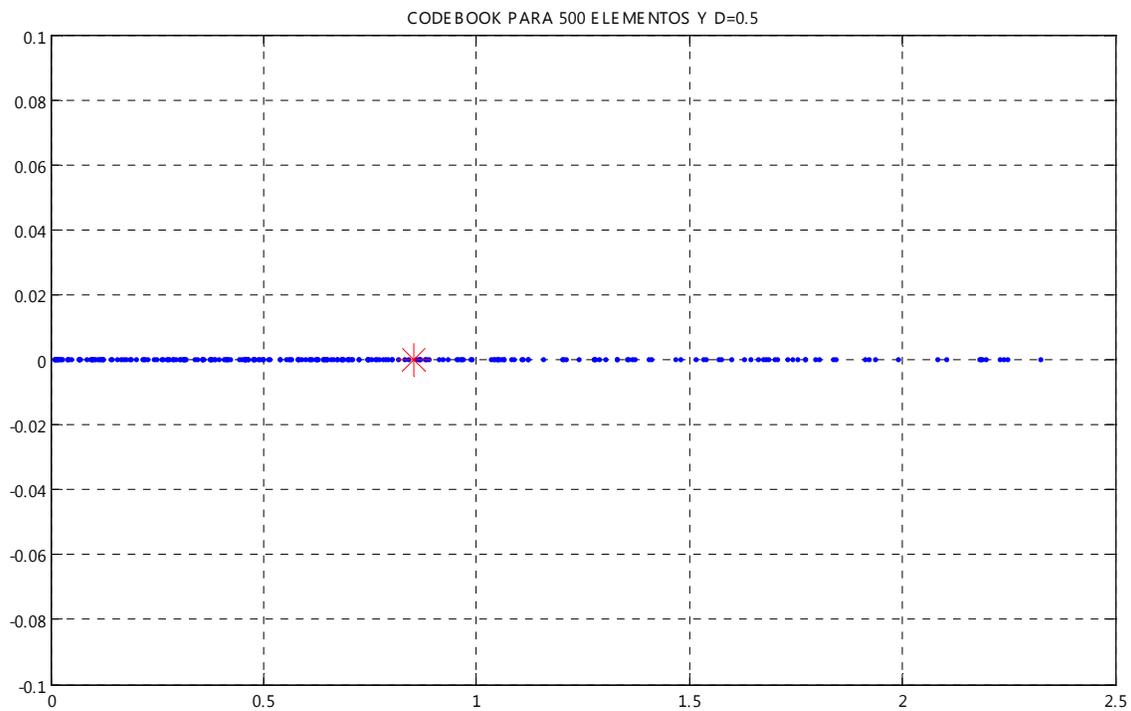
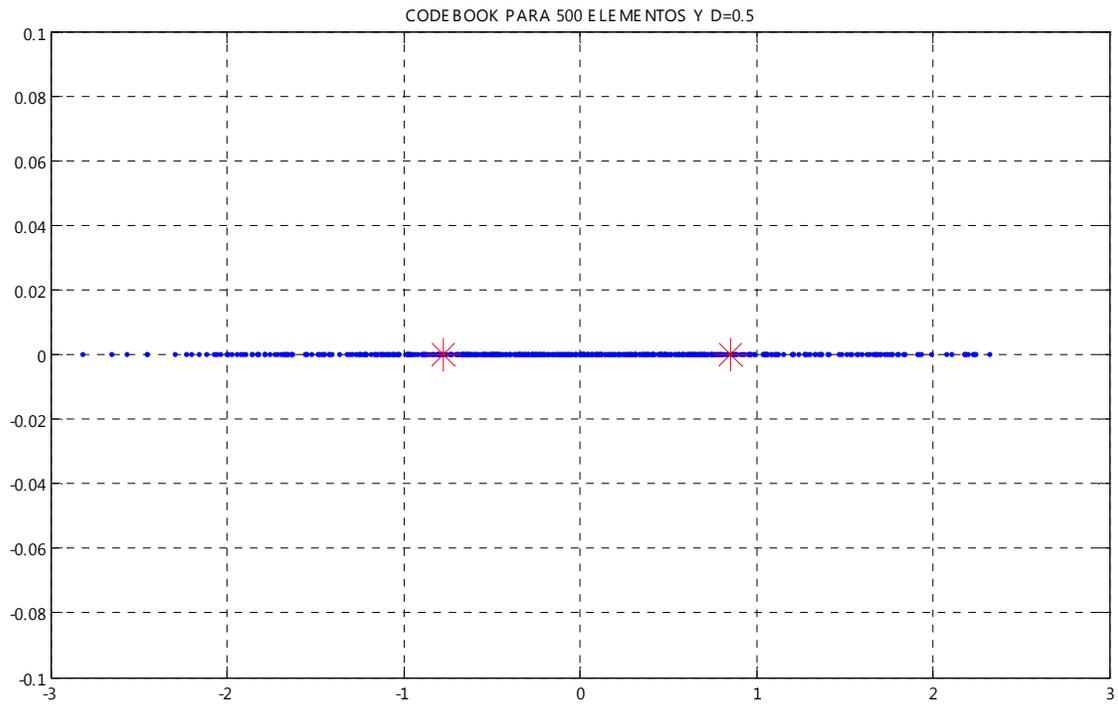


Parámetros de Diseño: Centroides generados a partir de 500 números aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.1.





Parámetros de Diseño: Centroides generados a partir de 500 números aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.5.



**ANEXO 5: CUANTIZACIÓN VECTORIAL
(DISTRIBUCIÓN UNIFORME)**

Parámetros de Diseño: Codebook generado a partir de 10 000 vectores aleatorios entre [0,5] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.003.

0.0645	0.0645	0.2106	1.4348	0.3534	3.1787
0.0645	0.2106	0.2106	1.5566	0.3534	3.3653
0.0645	0.3534	0.2106	1.6879	0.3534	3.4769
0.0645	0.4693	0.2106	1.8777	0.3534	3.6453
0.0645	0.5976	0.2106	2.0312	0.3534	3.8181
0.0645	0.7294	0.2106	2.2034	0.3534	3.9419
0.0645	0.8190	0.2106	2.3868	0.3534	4.1071
0.0645	0.8958	0.2106	2.5851	0.3534	4.2633
0.0645	1.0202	0.2106	2.7279	0.3534	4.4385
0.0645	1.1423	0.2106	2.9044	0.3534	4.5848
0.0645	1.2551	0.2106	3.0506	0.3534	4.7832
0.0645	1.4348	0.2106	3.1787	0.3534	4.9343
0.0645	1.5566	0.2106	3.3653	0.4693	0.0645
0.0645	1.6879	0.2106	3.4769	0.4693	0.2106
0.0645	1.8777	0.2106	3.6453	0.4693	0.3534
0.0645	2.0312	0.2106	3.8181	0.4693	0.4693
0.0645	2.2034	0.2106	3.9419	0.4693	0.5976
0.0645	2.3868	0.2106	4.1071	0.4693	0.7294
0.0645	2.5851	0.2106	4.2633	0.4693	0.8190
0.0645	2.7279	0.2106	4.4385	0.4693	0.8958
0.0645	2.9044	0.2106	4.5848	0.4693	1.0202
0.0645	3.0506	0.2106	4.7832	0.4693	1.1423
0.0645	3.1787	0.2106	4.9343	0.4693	1.2551
0.0645	3.3653	0.3534	0.0645	0.4693	1.4348
0.0645	3.4769	0.3534	0.2106	0.4693	1.5566
0.0645	3.6453	0.3534	0.3534	0.4693	1.6879
0.0645	3.8181	0.3534	0.4693	0.4693	1.8777
0.0645	3.9419	0.3534	0.5976	0.4693	2.0312
0.0645	4.1071	0.3534	0.7294	0.4693	2.2034
0.0645	4.2633	0.3534	0.8190	0.4693	2.3868
0.0645	4.4385	0.3534	0.8958	0.4693	2.5851
0.0645	4.5848	0.3534	1.0202	0.4693	2.7279
0.0645	4.7832	0.3534	1.1423	0.4693	2.9044
0.0645	4.9343	0.3534	1.2551	0.4693	3.0506
0.2106	0.0645	0.3534	1.4348	0.4693	3.1787
0.2106	0.2106	0.3534	1.5566	0.4693	3.3653
0.2106	0.3534	0.3534	1.6879	0.4693	3.4769
0.2106	0.4693	0.3534	1.8777	0.4693	3.6453
0.2106	0.5976	0.3534	2.0312	0.4693	3.8181
0.2106	0.7294	0.3534	2.2034	0.4693	3.9419
0.2106	0.8190	0.3534	2.3868	0.4693	4.1071
0.2106	0.8958	0.3534	2.5851	0.4693	4.2633
0.2106	1.0202	0.3534	2.7279	0.4693	4.4385
0.2106	1.1423	0.3534	2.9044	0.4693	4.5848
0.2106	1.2551	0.3534	3.0506	0.4693	4.7832

0.4693	4.9343	0.7294	2.2034	0.8190	4.9343
0.5976	0.0645	0.7294	2.3868	0.8958	0.0645
0.5976	0.2106	0.7294	2.5851	0.8958	0.2106
0.5976	0.3534	0.7294	2.7279	0.8958	0.3534
0.5976	0.4693	0.7294	2.9044	0.8958	0.4693
0.5976	0.5976	0.7294	3.0506	0.8958	0.5976
0.5976	0.7294	0.7294	3.1787	0.8958	0.7294
0.5976	0.8190	0.7294	3.3653	0.8958	0.8190
0.5976	0.8958	0.7294	3.4769	0.8958	0.8958
0.5976	1.0202	0.7294	3.6453	0.8958	1.0202
0.5976	1.1423	0.7294	3.8181	0.8958	1.1423
0.5976	1.2551	0.7294	3.9419	0.8958	1.2551
0.5976	1.4348	0.7294	4.1071	0.8958	1.4348
0.5976	1.5566	0.7294	4.2633	0.8958	1.5566
0.5976	1.6879	0.7294	4.4385	0.8958	1.6879
0.5976	1.8777	0.7294	4.5848	0.8958	1.8777
0.5976	2.0312	0.7294	4.7832	0.8958	2.0312
0.5976	2.2034	0.7294	4.9343	0.8958	2.2034
0.5976	2.3868	0.8190	0.0645	0.8958	2.3868
0.5976	2.5851	0.8190	0.2106	0.8958	2.5851
0.5976	2.7279	0.8190	0.3534	0.8958	2.7279
0.5976	2.9044	0.8190	0.4693	0.8958	2.9044
0.5976	3.0506	0.8190	0.5976	0.8958	3.0506
0.5976	3.1787	0.8190	0.7294	0.8958	3.1787
0.5976	3.3653	0.8190	0.8190	0.8958	3.3653
0.5976	3.4769	0.8190	0.8958	0.8958	3.4769
0.5976	3.6453	0.8190	1.0202	0.8958	3.6453
0.5976	3.8181	0.8190	1.1423	0.8958	3.8181
0.5976	3.9419	0.8190	1.2551	0.8958	3.9419
0.5976	4.1071	0.8190	1.4348	0.8958	4.1071
0.5976	4.2633	0.8190	1.5566	0.8958	4.2633
0.5976	4.4385	0.8190	1.6879	0.8958	4.4385
0.5976	4.5848	0.8190	1.8777	0.8958	4.5848
0.5976	4.7832	0.8190	2.0312	0.8958	4.7832
0.5976	4.9343	0.8190	2.2034	0.8958	4.9343
0.7294	0.0645	0.8190	2.3868	1.0202	0.0645
0.7294	0.2106	0.8190	2.5851	1.0202	0.2106
0.7294	0.3534	0.8190	2.7279	1.0202	0.3534
0.7294	0.4693	0.8190	2.9044	1.0202	0.4693
0.7294	0.5976	0.8190	3.0506	1.0202	0.5976
0.7294	0.7294	0.8190	3.1787	1.0202	0.7294
0.7294	0.8190	0.8190	3.3653	1.0202	0.8190
0.7294	0.8958	0.8190	3.4769	1.0202	0.8958
0.7294	1.0202	0.8190	3.6453	1.0202	1.0202
0.7294	1.1423	0.8190	3.8181	1.0202	1.1423
0.7294	1.2551	0.8190	3.9419	1.0202	1.2551
0.7294	1.4348	0.8190	4.1071	1.0202	1.4348
0.7294	1.5566	0.8190	4.2633	1.0202	1.5566
0.7294	1.6879	0.8190	4.4385	1.0202	1.6879
0.7294	1.8777	0.8190	4.5848	1.0202	1.8777
0.7294	2.0312	0.8190	4.7832	1.0202	2.0312

1.0202	2.2034	1.1423	4.9343	1.4348	2.2034
1.0202	2.3868	1.2551	0.0645	1.4348	2.3868
1.0202	2.5851	1.2551	0.2106	1.4348	2.5851
1.0202	2.7279	1.2551	0.3534	1.4348	2.7279
1.0202	2.9044	1.2551	0.4693	1.4348	2.9044
1.0202	3.0506	1.2551	0.5976	1.4348	3.0506
1.0202	3.1787	1.2551	0.7294	1.4348	3.1787
1.0202	3.3653	1.2551	0.8190	1.4348	3.3653
1.0202	3.4769	1.2551	0.8958	1.4348	3.4769
1.0202	3.6453	1.2551	1.0202	1.4348	3.6453
1.0202	3.8181	1.2551	1.1423	1.4348	3.8181
1.0202	3.9419	1.2551	1.2551	1.4348	3.9419
1.0202	4.1071	1.2551	1.4348	1.4348	4.1071
1.0202	4.2633	1.2551	1.5566	1.4348	4.2633
1.0202	4.4385	1.2551	1.6879	1.4348	4.4385
1.0202	4.5848	1.2551	1.8777	1.4348	4.5848
1.0202	4.7832	1.2551	2.0312	1.4348	4.7832
1.0202	4.9343	1.2551	2.2034	1.4348	4.9343
1.1423	0.0645	1.2551	2.3868	1.5566	0.0645
1.1423	0.2106	1.2551	2.5851	1.5566	0.2106
1.1423	0.3534	1.2551	2.7279	1.5566	0.3534
1.1423	0.4693	1.2551	2.9044	1.5566	0.4693
1.1423	0.5976	1.2551	3.0506	1.5566	0.5976
1.1423	0.7294	1.2551	3.1787	1.5566	0.7294
1.1423	0.8190	1.2551	3.3653	1.5566	0.8190
1.1423	0.8958	1.2551	3.4769	1.5566	0.8958
1.1423	1.0202	1.2551	3.6453	1.5566	1.0202
1.1423	1.1423	1.2551	3.8181	1.5566	1.1423
1.1423	1.2551	1.2551	3.9419	1.5566	1.2551
1.1423	1.4348	1.2551	4.1071	1.5566	1.4348
1.1423	1.5566	1.2551	4.2633	1.5566	1.5566
1.1423	1.6879	1.2551	4.4385	1.5566	1.6879
1.1423	1.8777	1.2551	4.5848	1.5566	1.8777
1.1423	2.0312	1.2551	4.7832	1.5566	2.0312
1.1423	2.2034	1.2551	4.9343	1.5566	2.2034
1.1423	2.3868	1.4348	0.0645	1.5566	2.3868
1.1423	2.5851	1.4348	0.2106	1.5566	2.5851
1.1423	2.7279	1.4348	0.3534	1.5566	2.7279
1.1423	2.9044	1.4348	0.4693	1.5566	2.9044
1.1423	3.0506	1.4348	0.5976	1.5566	3.0506
1.1423	3.1787	1.4348	0.7294	1.5566	3.1787
1.1423	3.3653	1.4348	0.8190	1.5566	3.3653
1.1423	3.4769	1.4348	0.8958	1.5566	3.4769
1.1423	3.6453	1.4348	1.0202	1.5566	3.6453
1.1423	3.8181	1.4348	1.1423	1.5566	3.8181
1.1423	3.9419	1.4348	1.2551	1.5566	3.9419
1.1423	4.1071	1.4348	1.4348	1.5566	4.1071
1.1423	4.2633	1.4348	1.5566	1.5566	4.2633
1.1423	4.4385	1.4348	1.6879	1.5566	4.4385
1.1423	4.5848	1.4348	1.8777	1.5566	4.5848
1.1423	4.7832	1.4348	2.0312	1.5566	4.7832

1.5566	4.9343	1.8777	2.2034	2.0312	4.9343
1.6879	0.0645	1.8777	2.3868	2.2034	0.0645
1.6879	0.2106	1.8777	2.5851	2.2034	0.2106
1.6879	0.3534	1.8777	2.7279	2.2034	0.3534
1.6879	0.4693	1.8777	2.9044	2.2034	0.4693
1.6879	0.5976	1.8777	3.0506	2.2034	0.5976
1.6879	0.7294	1.8777	3.1787	2.2034	0.7294
1.6879	0.8190	1.8777	3.3653	2.2034	0.8190
1.6879	0.8958	1.8777	3.4769	2.2034	0.8958
1.6879	1.0202	1.8777	3.6453	2.2034	1.0202
1.6879	1.1423	1.8777	3.8181	2.2034	1.1423
1.6879	1.2551	1.8777	3.9419	2.2034	1.2551
1.6879	1.4348	1.8777	4.1071	2.2034	1.4348
1.6879	1.5566	1.8777	4.2633	2.2034	1.5566
1.6879	1.6879	1.8777	4.4385	2.2034	1.6879
1.6879	1.8777	1.8777	4.5848	2.2034	1.8777
1.6879	2.0312	1.8777	4.7832	2.2034	2.0312
1.6879	2.2034	1.8777	4.9343	2.2034	2.2034
1.6879	2.3868	2.0312	0.0645	2.2034	2.3868
1.6879	2.5851	2.0312	0.2106	2.2034	2.5851
1.6879	2.7279	2.0312	0.3534	2.2034	2.7279
1.6879	2.9044	2.0312	0.4693	2.2034	2.9044
1.6879	3.0506	2.0312	0.5976	2.2034	3.0506
1.6879	3.1787	2.0312	0.7294	2.2034	3.1787
1.6879	3.3653	2.0312	0.8190	2.2034	3.3653
1.6879	3.4769	2.0312	0.8958	2.2034	3.4769
1.6879	3.6453	2.0312	1.0202	2.2034	3.6453
1.6879	3.8181	2.0312	1.1423	2.2034	3.8181
1.6879	3.9419	2.0312	1.2551	2.2034	3.9419
1.6879	4.1071	2.0312	1.4348	2.2034	4.1071
1.6879	4.2633	2.0312	1.5566	2.2034	4.2633
1.6879	4.4385	2.0312	1.6879	2.2034	4.4385
1.6879	4.5848	2.0312	1.8777	2.2034	4.5848
1.6879	4.7832	2.0312	2.0312	2.2034	4.7832
1.6879	4.9343	2.0312	2.2034	2.2034	4.9343
1.8777	0.0645	2.0312	2.3868	2.3868	0.0645
1.8777	0.2106	2.0312	2.5851	2.3868	0.2106
1.8777	0.3534	2.0312	2.7279	2.3868	0.3534
1.8777	0.4693	2.0312	2.9044	2.3868	0.4693
1.8777	0.5976	2.0312	3.0506	2.3868	0.5976
1.8777	0.7294	2.0312	3.1787	2.3868	0.7294
1.8777	0.8190	2.0312	3.3653	2.3868	0.8190
1.8777	0.8958	2.0312	3.4769	2.3868	0.8958
1.8777	1.0202	2.0312	3.6453	2.3868	1.0202
1.8777	1.1423	2.0312	3.8181	2.3868	1.1423
1.8777	1.2551	2.0312	3.9419	2.3868	1.2551
1.8777	1.4348	2.0312	4.1071	2.3868	1.4348
1.8777	1.5566	2.0312	4.2633	2.3868	1.5566
1.8777	1.6879	2.0312	4.4385	2.3868	1.6879
1.8777	1.8777	2.0312	4.5848	2.3868	1.8777
1.8777	2.0312	2.0312	4.7832	2.3868	2.0312

2.3868	2.2034	2.5851	4.9343	2.9044	2.2034
2.3868	2.3868	2.7279	0.0645	2.9044	2.3868
2.3868	2.5851	2.7279	0.2106	2.9044	2.5851
2.3868	2.7279	2.7279	0.3534	2.9044	2.7279
2.3868	2.9044	2.7279	0.4693	2.9044	2.9044
2.3868	3.0506	2.7279	0.5976	2.9044	3.0506
2.3868	3.1787	2.7279	0.7294	2.9044	3.1787
2.3868	3.3653	2.7279	0.8190	2.9044	3.3653
2.3868	3.4769	2.7279	0.8958	2.9044	3.4769
2.3868	3.6453	2.7279	1.0202	2.9044	3.6453
2.3868	3.8181	2.7279	1.1423	2.9044	3.8181
2.3868	3.9419	2.7279	1.2551	2.9044	3.9419
2.3868	4.1071	2.7279	1.4348	2.9044	4.1071
2.3868	4.2633	2.7279	1.5566	2.9044	4.2633
2.3868	4.4385	2.7279	1.6879	2.9044	4.4385
2.3868	4.5848	2.7279	1.8777	2.9044	4.5848
2.3868	4.7832	2.7279	2.0312	2.9044	4.7832
2.3868	4.9343	2.7279	2.2034	2.9044	4.9343
2.5851	0.0645	2.7279	2.3868	3.0506	0.0645
2.5851	0.2106	2.7279	2.5851	3.0506	0.2106
2.5851	0.3534	2.7279	2.7279	3.0506	0.3534
2.5851	0.4693	2.7279	2.9044	3.0506	0.4693
2.5851	0.5976	2.7279	3.0506	3.0506	0.5976
2.5851	0.7294	2.7279	3.1787	3.0506	0.7294
2.5851	0.8190	2.7279	3.3653	3.0506	0.8190
2.5851	0.8958	2.7279	3.4769	3.0506	0.8958
2.5851	1.0202	2.7279	3.6453	3.0506	1.0202
2.5851	1.1423	2.7279	3.8181	3.0506	1.1423
2.5851	1.2551	2.7279	3.9419	3.0506	1.2551
2.5851	1.4348	2.7279	4.1071	3.0506	1.4348
2.5851	1.5566	2.7279	4.2633	3.0506	1.5566
2.5851	1.6879	2.7279	4.4385	3.0506	1.6879
2.5851	1.8777	2.7279	4.5848	3.0506	1.8777
2.5851	2.0312	2.7279	4.7832	3.0506	2.0312
2.5851	2.2034	2.7279	4.9343	3.0506	2.2034
2.5851	2.3868	2.9044	0.0645	3.0506	2.3868
2.5851	2.5851	2.9044	0.2106	3.0506	2.5851
2.5851	2.7279	2.9044	0.3534	3.0506	2.7279
2.5851	2.9044	2.9044	0.4693	3.0506	2.9044
2.5851	3.0506	2.9044	0.5976	3.0506	3.0506
2.5851	3.1787	2.9044	0.7294	3.0506	3.1787
2.5851	3.3653	2.9044	0.8190	3.0506	3.3653
2.5851	3.4769	2.9044	0.8958	3.0506	3.4769
2.5851	3.6453	2.9044	1.0202	3.0506	3.6453
2.5851	3.8181	2.9044	1.1423	3.0506	3.8181
2.5851	3.9419	2.9044	1.2551	3.0506	3.9419
2.5851	4.1071	2.9044	1.4348	3.0506	4.1071
2.5851	4.2633	2.9044	1.5566	3.0506	4.2633
2.5851	4.4385	2.9044	1.6879	3.0506	4.4385
2.5851	4.5848	2.9044	1.8777	3.0506	4.5848
2.5851	4.7832	2.9044	2.0312	3.0506	4.7832

3.0506	4.9343	3.3653	2.2034	3.4769	4.9343
3.1787	0.0645	3.3653	2.3868	3.6453	0.0645
3.1787	0.2106	3.3653	2.5851	3.6453	0.2106
3.1787	0.3534	3.3653	2.7279	3.6453	0.3534
3.1787	0.4693	3.3653	2.9044	3.6453	0.4693
3.1787	0.5976	3.3653	3.0506	3.6453	0.5976
3.1787	0.7294	3.3653	3.1787	3.6453	0.7294
3.1787	0.8190	3.3653	3.3653	3.6453	0.8190
3.1787	0.8958	3.3653	3.4769	3.6453	0.8958
3.1787	1.0202	3.3653	3.6453	3.6453	1.0202
3.1787	1.1423	3.3653	3.8181	3.6453	1.1423
3.1787	1.2551	3.3653	3.9419	3.6453	1.2551
3.1787	1.4348	3.3653	4.1071	3.6453	1.4348
3.1787	1.5566	3.3653	4.2633	3.6453	1.5566
3.1787	1.6879	3.3653	4.4385	3.6453	1.6879
3.1787	1.8777	3.3653	4.5848	3.6453	1.8777
3.1787	2.0312	3.3653	4.7832	3.6453	2.0312
3.1787	2.2034	3.3653	4.9343	3.6453	2.2034
3.1787	2.3868	3.4769	0.0645	3.6453	2.3868
3.1787	2.5851	3.4769	0.2106	3.6453	2.5851
3.1787	2.7279	3.4769	0.3534	3.6453	2.7279
3.1787	2.9044	3.4769	0.4693	3.6453	2.9044
3.1787	3.0506	3.4769	0.5976	3.6453	3.0506
3.1787	3.1787	3.4769	0.7294	3.6453	3.1787
3.1787	3.3653	3.4769	0.8190	3.6453	3.3653
3.1787	3.4769	3.4769	0.8958	3.6453	3.4769
3.1787	3.6453	3.4769	1.0202	3.6453	3.6453
3.1787	3.8181	3.4769	1.1423	3.6453	3.8181
3.1787	3.9419	3.4769	1.2551	3.6453	3.9419
3.1787	4.1071	3.4769	1.4348	3.6453	4.1071
3.1787	4.2633	3.4769	1.5566	3.6453	4.2633
3.1787	4.4385	3.4769	1.6879	3.6453	4.4385
3.1787	4.5848	3.4769	1.8777	3.6453	4.5848
3.1787	4.7832	3.4769	2.0312	3.6453	4.7832
3.1787	4.9343	3.4769	2.2034	3.6453	4.9343
3.3653	0.0645	3.4769	2.3868	3.8181	0.0645
3.3653	0.2106	3.4769	2.5851	3.8181	0.2106
3.3653	0.3534	3.4769	2.7279	3.8181	0.3534
3.3653	0.4693	3.4769	2.9044	3.8181	0.4693
3.3653	0.5976	3.4769	3.0506	3.8181	0.5976
3.3653	0.7294	3.4769	3.1787	3.8181	0.7294
3.3653	0.8190	3.4769	3.3653	3.8181	0.8190
3.3653	0.8958	3.4769	3.4769	3.8181	0.8958
3.3653	1.0202	3.4769	3.6453	3.8181	1.0202
3.3653	1.1423	3.4769	3.8181	3.8181	1.1423
3.3653	1.2551	3.4769	3.9419	3.8181	1.2551
3.3653	1.4348	3.4769	4.1071	3.8181	1.4348
3.3653	1.5566	3.4769	4.2633	3.8181	1.5566
3.3653	1.6879	3.4769	4.4385	3.8181	1.6879
3.3653	1.8777	3.4769	4.5848	3.8181	1.8777
3.3653	2.0312	3.4769	4.7832	3.8181	2.0312

3.8181	2.2034	3.9419	4.9343	4.2633	2.2034
3.8181	2.3868	4.1071	0.0645	4.2633	2.3868
3.8181	2.5851	4.1071	0.2106	4.2633	2.5851
3.8181	2.7279	4.1071	0.3534	4.2633	2.7279
3.8181	2.9044	4.1071	0.4693	4.2633	2.9044
3.8181	3.0506	4.1071	0.5976	4.2633	3.0506
3.8181	3.1787	4.1071	0.7294	4.2633	3.1787
3.8181	3.3653	4.1071	0.8190	4.2633	3.3653
3.8181	3.4769	4.1071	0.8958	4.2633	3.4769
3.8181	3.6453	4.1071	1.0202	4.2633	3.6453
3.8181	3.8181	4.1071	1.1423	4.2633	3.8181
3.8181	3.9419	4.1071	1.2551	4.2633	3.9419
3.8181	4.1071	4.1071	1.4348	4.2633	4.1071
3.8181	4.2633	4.1071	1.5566	4.2633	4.2633
3.8181	4.4385	4.1071	1.6879	4.2633	4.4385
3.8181	4.5848	4.1071	1.8777	4.2633	4.5848
3.8181	4.7832	4.1071	2.0312	4.2633	4.7832
3.8181	4.9343	4.1071	2.2034	4.2633	4.9343
3.9419	0.0645	4.1071	2.3868	4.4385	0.0645
3.9419	0.2106	4.1071	2.5851	4.4385	0.2106
3.9419	0.3534	4.1071	2.7279	4.4385	0.3534
3.9419	0.4693	4.1071	2.9044	4.4385	0.4693
3.9419	0.5976	4.1071	3.0506	4.4385	0.5976
3.9419	0.7294	4.1071	3.1787	4.4385	0.7294
3.9419	0.8190	4.1071	3.3653	4.4385	0.8190
3.9419	0.8958	4.1071	3.4769	4.4385	0.8958
3.9419	1.0202	4.1071	3.6453	4.4385	1.0202
3.9419	1.1423	4.1071	3.8181	4.4385	1.1423
3.9419	1.2551	4.1071	3.9419	4.4385	1.2551
3.9419	1.4348	4.1071	4.1071	4.4385	1.4348
3.9419	1.5566	4.1071	4.2633	4.4385	1.5566
3.9419	1.6879	4.1071	4.4385	4.4385	1.6879
3.9419	1.8777	4.1071	4.5848	4.4385	1.8777
3.9419	2.0312	4.1071	4.7832	4.4385	2.0312
3.9419	2.2034	4.1071	4.9343	4.4385	2.2034
3.9419	2.3868	4.2633	0.0645	4.4385	2.3868
3.9419	2.5851	4.2633	0.2106	4.4385	2.5851
3.9419	2.7279	4.2633	0.3534	4.4385	2.7279
3.9419	2.9044	4.2633	0.4693	4.4385	2.9044
3.9419	3.0506	4.2633	0.5976	4.4385	3.0506
3.9419	3.1787	4.2633	0.7294	4.4385	3.1787
3.9419	3.3653	4.2633	0.8190	4.4385	3.3653
3.9419	3.4769	4.2633	0.8958	4.4385	3.4769
3.9419	3.6453	4.2633	1.0202	4.4385	3.6453
3.9419	3.8181	4.2633	1.1423	4.4385	3.8181
3.9419	3.9419	4.2633	1.2551	4.4385	3.9419
3.9419	4.1071	4.2633	1.4348	4.4385	4.1071
3.9419	4.2633	4.2633	1.5566	4.4385	4.2633
3.9419	4.4385	4.2633	1.6879	4.4385	4.4385
3.9419	4.5848	4.2633	1.8777	4.4385	4.5848
3.9419	4.7832	4.2633	2.0312	4.4385	4.7832

4.4385	4.9343	4.7832	0.0645	4.9343	0.2106
4.5848	0.0645	4.7832	0.2106	4.9343	0.3534
4.5848	0.2106	4.7832	0.3534	4.9343	0.4693
4.5848	0.3534	4.7832	0.4693	4.9343	0.5976
4.5848	0.4693	4.7832	0.5976	4.9343	0.7294
4.5848	0.5976	4.7832	0.7294	4.9343	0.8190
4.5848	0.7294	4.7832	0.8190	4.9343	0.8958
4.5848	0.8190	4.7832	0.8958	4.9343	1.0202
4.5848	0.8958	4.7832	1.0202	4.9343	1.1423
4.5848	1.0202	4.7832	1.1423	4.9343	1.2551
4.5848	1.1423	4.7832	1.2551	4.9343	1.4348
4.5848	1.2551	4.7832	1.4348	4.9343	1.5566
4.5848	1.4348	4.7832	1.5566	4.9343	1.6879
4.5848	1.5566	4.7832	1.6879	4.9343	1.8777
4.5848	1.6879	4.7832	1.8777	4.9343	2.0312
4.5848	1.8777	4.7832	2.0312	4.9343	2.2034
4.5848	2.0312	4.7832	2.2034	4.9343	2.3868
4.5848	2.2034	4.7832	2.3868	4.9343	2.5851
4.5848	2.3868	4.7832	2.5851	4.9343	2.7279
4.5848	2.5851	4.7832	2.7279	4.9343	2.9044
4.5848	2.7279	4.7832	2.9044	4.9343	3.0506
4.5848	2.9044	4.7832	3.0506	4.9343	3.1787
4.5848	3.0506	4.7832	3.1787	4.9343	3.3653
4.5848	3.1787	4.7832	3.3653	4.9343	3.4769
4.5848	3.3653	4.7832	3.4769	4.9343	3.6453
4.5848	3.4769	4.7832	3.6453	4.9343	3.8181
4.5848	3.6453	4.7832	3.8181	4.9343	3.9419
4.5848	3.8181	4.7832	3.9419	4.9343	4.1071
4.5848	3.9419	4.7832	4.1071	4.9343	4.2633
4.5848	4.1071	4.7832	4.2633	4.9343	4.4385
4.5848	4.2633	4.7832	4.4385	4.9343	4.5848
4.5848	4.4385	4.7832	4.5848	4.9343	4.7832
4.5848	4.5848	4.7832	4.7832	4.9343	4.9343
4.5848	4.7832	4.7832	4.9343		
4.5848	4.9343	4.9343	0.0645		

Parámetros de Diseño: Codebook generado a partir de 10 000 vectores aleatorios entre [0,5] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.5.

0.7219	0.7219	3.4319	0.7219
0.7219	2.0982	3.4319	2.0982
0.7219	3.4319	3.4319	3.4319
0.7219	4.4557	3.4319	4.4557
2.0982	0.7219	4.4557	0.7219
2.0982	2.0982	4.4557	2.0982
2.0982	3.4319	4.4557	3.4319
2.0982	4.4557	4.4557	4.4557

Parámetros de Diseño: Codebook generado a partir de 10 000 vectores aleatorios entre [0,5] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 1.

1.3468	1.3468	3.8717	1.3468
1.3468	3.8717	3.8717	3.8717

Parámetros de Diseño: Codebook generado a partir de 10 000 vectores aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.003.

0.0659	0.0659	0.2992	0.7946	0.6594	0.5520
0.0659	0.1795	0.2992	0.9327	0.6594	0.6594
0.0659	0.2992	0.4314	0.0659	0.6594	0.7946
0.0659	0.4314	0.4314	0.1795	0.6594	0.9327
0.0659	0.5520	0.4314	0.2992	0.7946	0.0659
0.0659	0.6594	0.4314	0.4314	0.7946	0.1795
0.0659	0.7946	0.4314	0.5520	0.7946	0.2992
0.0659	0.9327	0.4314	0.6594	0.7946	0.4314
0.1795	0.0659	0.4314	0.7946	0.7946	0.5520
0.1795	0.1795	0.4314	0.9327	0.7946	0.6594
0.1795	0.2992	0.5520	0.0659	0.7946	0.7946
0.1795	0.4314	0.5520	0.1795	0.7946	0.9327
0.1795	0.5520	0.5520	0.2992	0.9327	0.0659
0.1795	0.6594	0.5520	0.4314	0.9327	0.1795
0.1795	0.7946	0.5520	0.5520	0.9327	0.2992
0.1795	0.9327	0.5520	0.6594	0.9327	0.4314
0.2992	0.0659	0.5520	0.7946	0.9327	0.5520
0.2992	0.1795	0.5520	0.9327	0.9327	0.6594
0.2992	0.2992	0.6594	0.0659	0.9327	0.7946
0.2992	0.4314	0.6594	0.1795	0.9327	0.9327
0.2992	0.5520	0.6594	0.2992		
0.2992	0.6594	0.6594	0.4314		

Parámetros de Diseño: Codebook generado a partir de 10 000 vectores aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.5.

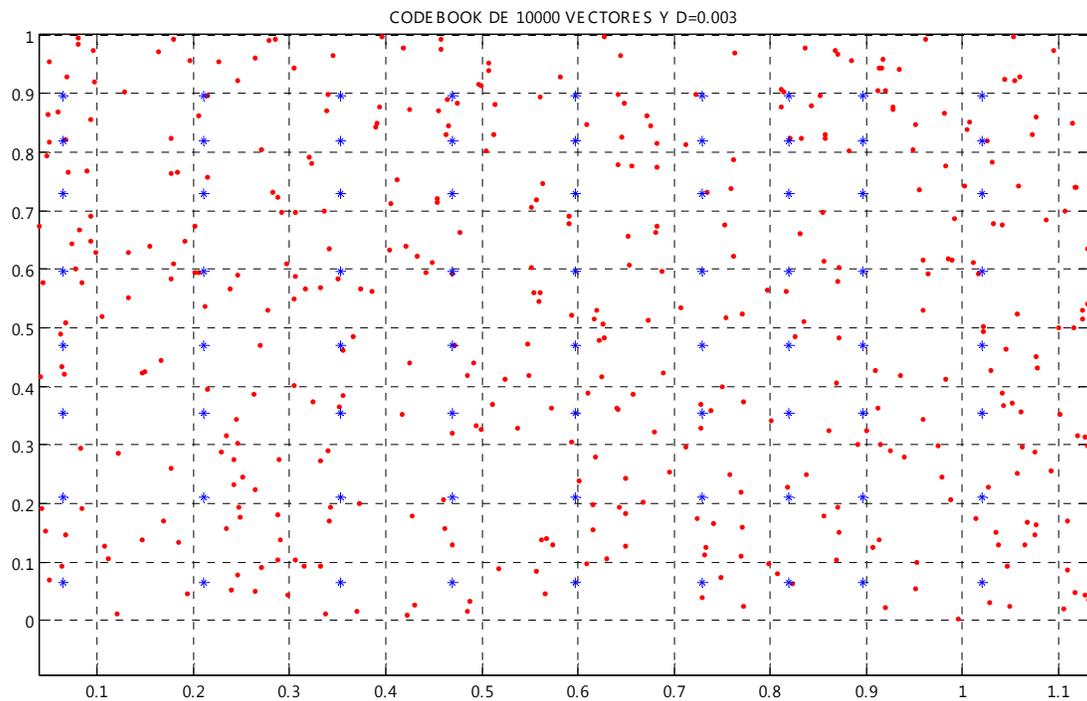
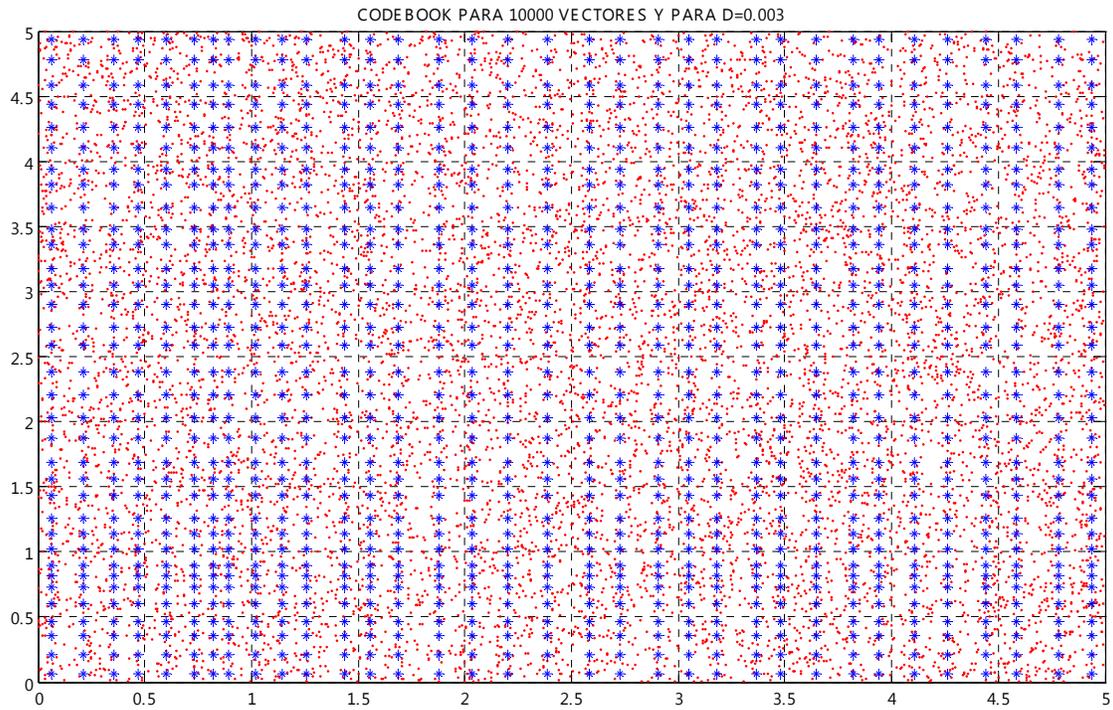
0.2373	0.2373	0.7483	0.2373
0.2373	0.7483	0.7483	0.7483

Parámetros de Diseño: Codebook generado a partir de 10 000 vectores aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 1.

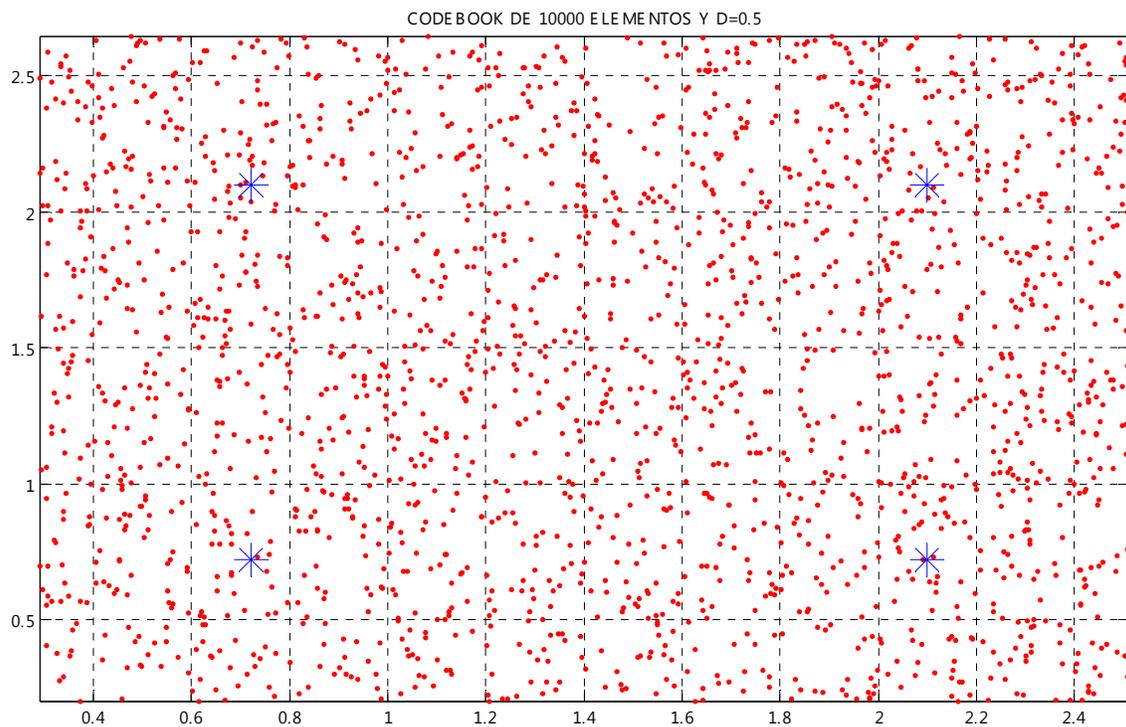
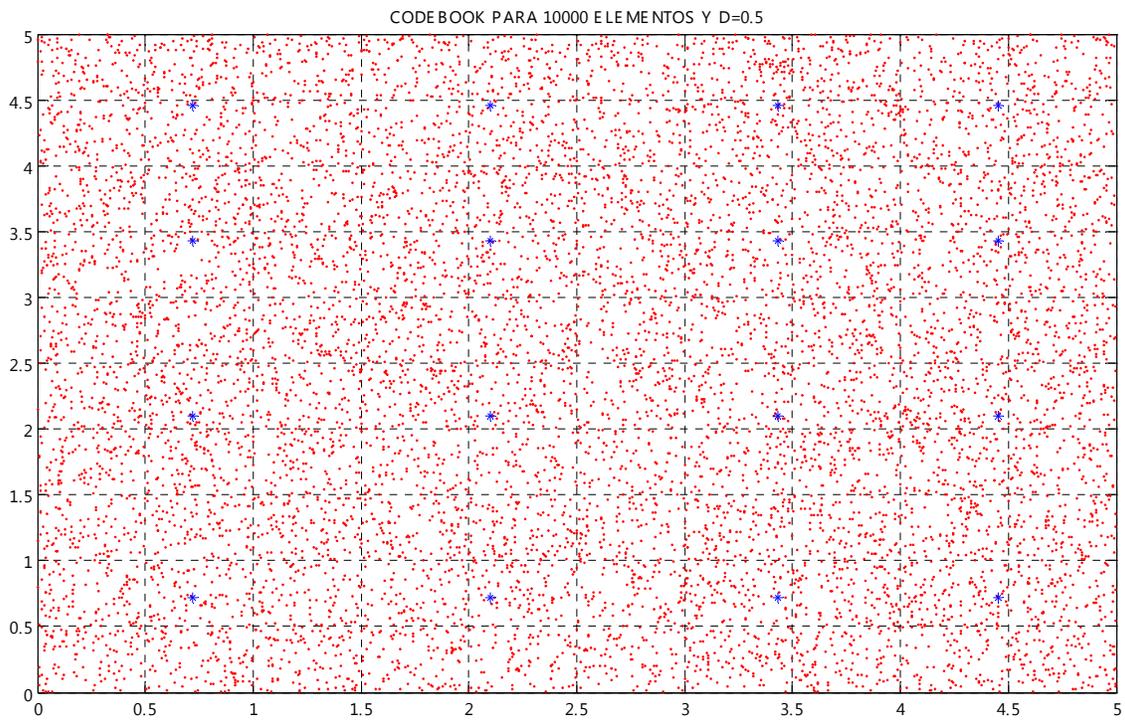
0.2845	0.2845	0.7701	0.2845
0.2845	0.7701	0.7701	0.7701

ANEXO 6: ALGORITMO COVQ (DISTRIBUCIÓN UNIFORME)

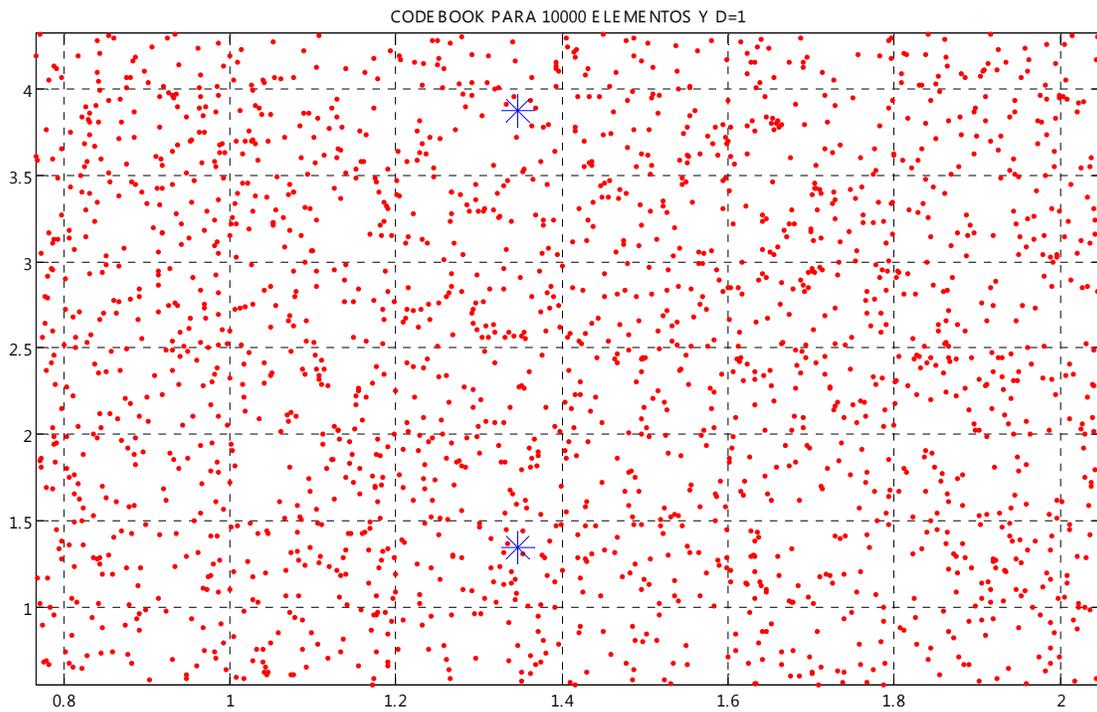
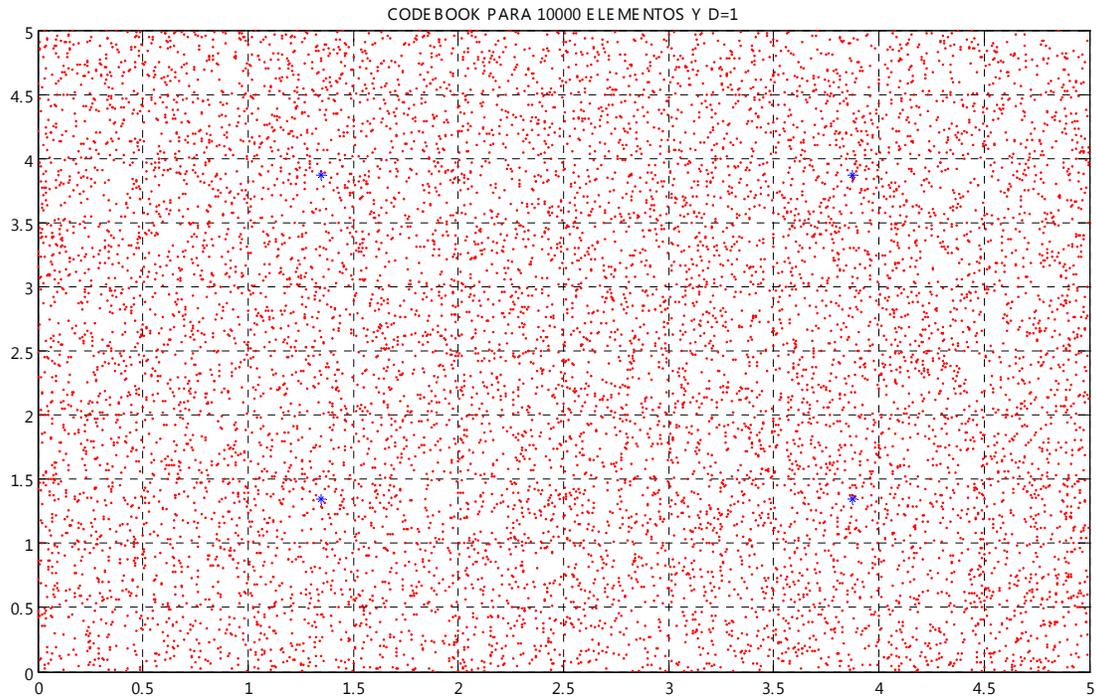
Parámetros de Diseño: Centroides generados a partir de 10 000 vectores aleatorios entre [0,5] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.003.



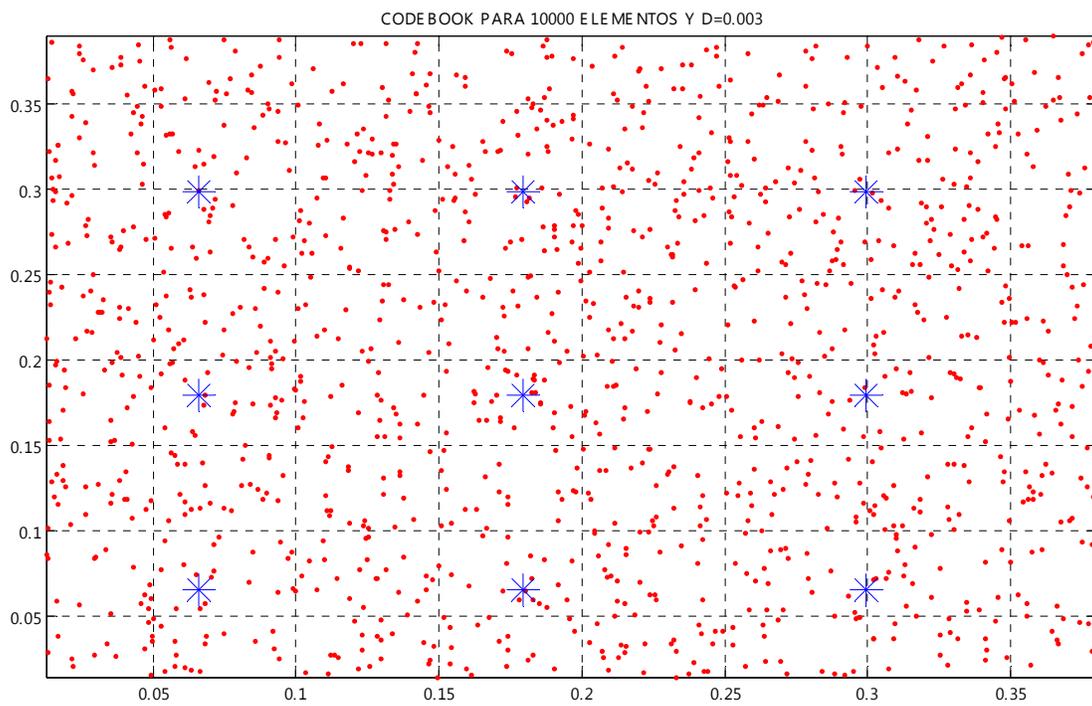
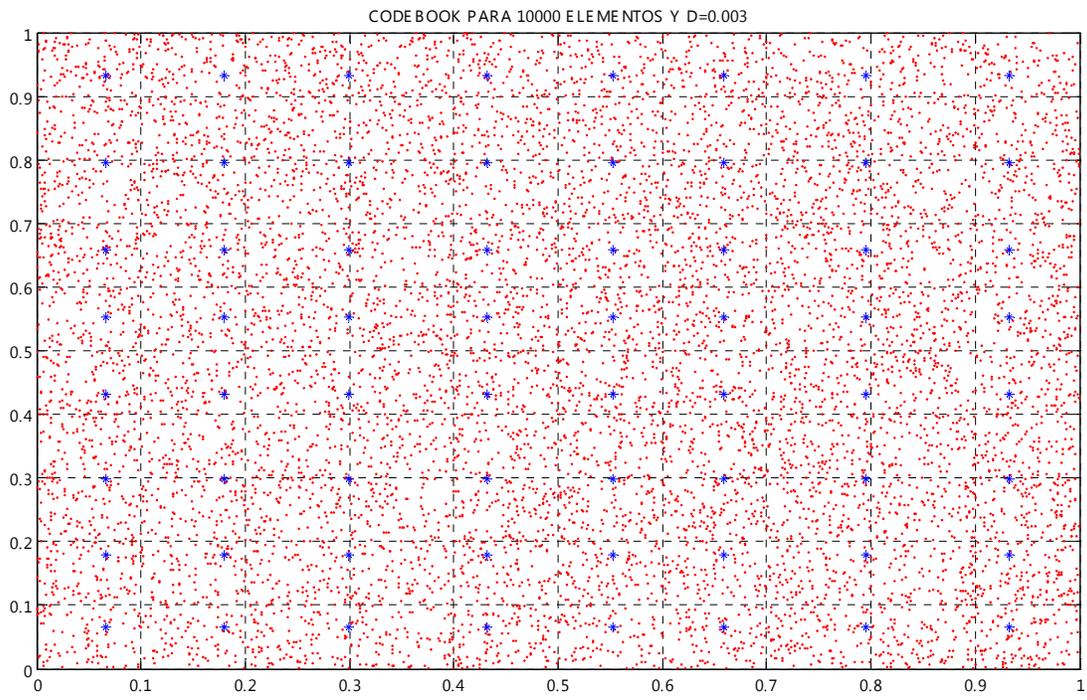
Parámetros de Diseño: Centroides generados a partir de 10 000 vectores aleatorios entre [0,5] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.5.



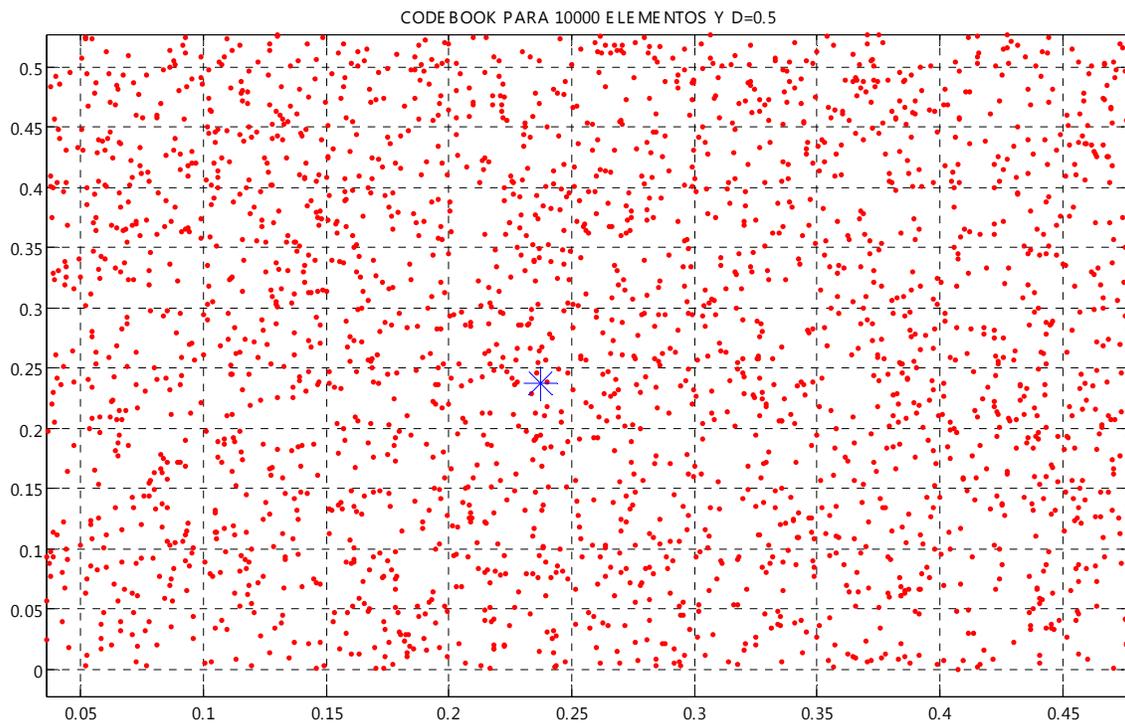
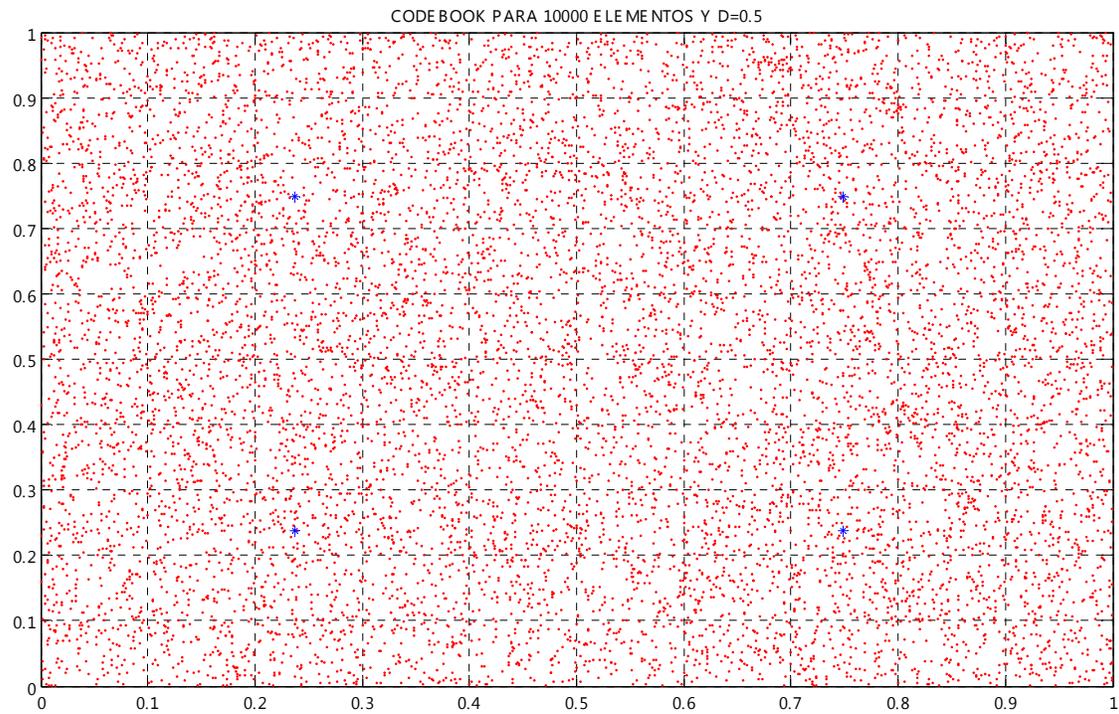
Parámetros de Diseño: Centroides generados a partir de 10 000 vectores aleatorios entre [0,5] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 1.



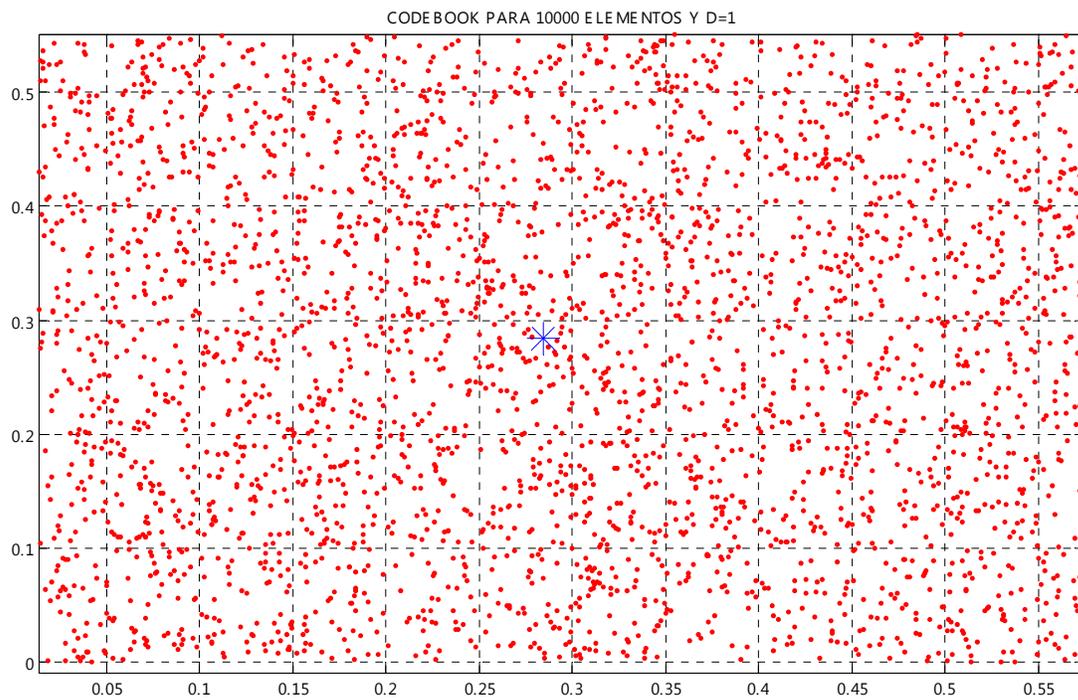
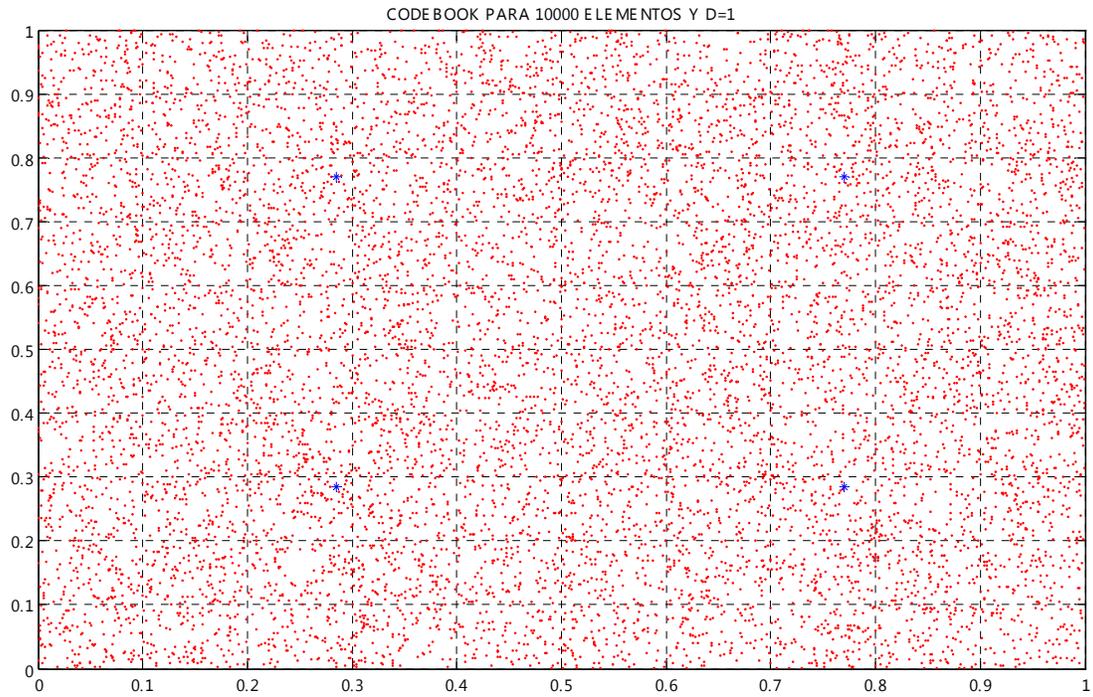
Parámetros de Diseño: Centroides generados a partir de 10 000 vectores aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.003.



Parámetros de Diseño: Centroides generados a partir de 10 000 vectores aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 0.5.



Parámetros de Diseño: Centroides generados a partir de 10 000 vectores aleatorios entre [0,1] con una distribución uniforme y tomando como parámetro fundamental una distorsión de 1.



ANEXO 7: CUANTIZACIÓN VECTORIAL (DISTRIBUCIÓN NORMAL)

Parámetros de Diseño: Codebook generado a partir de 1 000 vectores aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.1.

-1.7868	-1.7868	-0.6191	0.9562	0.5224	0.1465
-1.7868	-1.0768	-0.6191	1.6646	0.5224	0.5224
-1.7868	-0.6191	-0.2124	-1.7868	0.5224	0.9562
-1.7868	-0.2124	-0.2124	-1.0768	0.5224	1.6646
-1.7868	0.1465	-0.2124	-0.6191	0.9562	-1.7868
-1.7868	0.5224	-0.2124	-0.2124	0.9562	-1.0768
-1.7868	0.9562	-0.2124	0.1465	0.9562	-0.6191
-1.7868	1.6646	-0.2124	0.5224	0.9562	-0.2124
-1.0768	-1.7868	-0.2124	0.9562	0.9562	0.1465
-1.0768	-1.0768	-0.2124	1.6646	0.9562	0.5224
-1.0768	-0.6191	0.1465	-1.7868	0.9562	0.9562
-1.0768	-0.2124	0.1465	-1.0768	0.9562	1.6646
-1.0768	0.1465	0.1465	-0.6191	1.6646	-1.7868
-1.0768	0.5224	0.1465	-0.2124	1.6646	-1.0768
-1.0768	0.9562	0.1465	0.1465	1.6646	-0.6191
-1.0768	1.6646	0.1465	0.5224	1.6646	-0.2124
-0.6191	-1.7868	0.1465	0.9562	1.6646	0.1465
-0.6191	-1.0768	0.1465	1.6646	1.6646	0.5224
-0.6191	-0.6191	0.5224	-1.7868	1.6646	0.9562
-0.6191	-0.2124	0.5224	-1.0768	1.6646	1.6646
-0.6191	0.1465	0.5224	-0.6191		
-0.6191	0.5224	0.5224	-0.2124		

Parámetros de Diseño: Codebook generado a partir de 1 000 vectores aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.5.

-0.8112	-0.8112	0.7009	-0.8112
-0.8112	0.7009	0.7009	0.7009

Parámetros de Diseño: Codebook generado a partir de 500 vectores aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.003.

-2.4373	-2.4373	-2.4373	-0.8043	-2.4373	1.0718
-2.4373	-2.2994	-2.4373	-0.6768	-2.4373	1.2686
-2.4373	-2.0532	-2.4373	-0.5648	-2.4373	1.4472
-2.4373	-1.9083	-2.4373	-0.4680	-2.4373	1.5785
-2.4373	-1.6744	-2.4373	-0.3060	-2.4373	1.7568
-2.4373	-1.5105	-2.4373	-0.1368	-2.4373	1.8744
-2.4373	-1.4013	-2.4373	0.0443	-2.4373	1.9574
-2.4373	-1.3394	-2.4373	0.2642	-2.4373	1.9574
-2.4373	-1.2400	-2.4373	0.4518	-2.4373	2.1242
-2.4373	-1.1250	-2.4373	0.6302	-2.4373	2.2609
-2.4373	-1.0125	-2.4373	0.7924	-2.2994	-2.4373
-2.4373	-0.9202	-2.4373	0.9393	-2.2994	-2.2994

-2.2994	-2.0532	-2.0532	0.2642	-1.6744	-2.0532
-2.2994	-1.9083	-2.0532	0.4518	-1.6744	-1.9083
-2.2994	-1.6744	-2.0532	0.6302	-1.6744	-1.6744
-2.2994	-1.5105	-2.0532	0.7924	-1.6744	-1.5105
-2.2994	-1.4013	-2.0532	0.9393	-1.6744	-1.4013
-2.2994	-1.3394	-2.0532	1.0718	-1.6744	-1.3394
-2.2994	-1.2400	-2.0532	1.2686	-1.6744	-1.2400
-2.2994	-1.1250	-2.0532	1.4472	-1.6744	-1.1250
-2.2994	-1.0125	-2.0532	1.5785	-1.6744	-1.0125
-2.2994	-0.9202	-2.0532	1.7568	-1.6744	-0.9202
-2.2994	-0.8043	-2.0532	1.8744	-1.6744	-0.8043
-2.2994	-0.6768	-2.0532	1.9574	-1.6744	-0.6768
-2.2994	-0.5648	-2.0532	1.9574	-1.6744	-0.5648
-2.2994	-0.4680	-2.0532	2.1242	-1.6744	-0.4680
-2.2994	-0.3060	-2.0532	2.2609	-1.6744	-0.3060
-2.2994	-0.1368	-1.9083	-2.4373	-1.6744	-0.1368
-2.2994	0.0443	-1.9083	-2.2994	-1.6744	0.0443
-2.2994	0.2642	-1.9083	-2.0532	-1.6744	0.2642
-2.2994	0.4518	-1.9083	-1.9083	-1.6744	0.4518
-2.2994	0.6302	-1.9083	-1.6744	-1.6744	0.6302
-2.2994	0.7924	-1.9083	-1.5105	-1.6744	0.7924
-2.2994	0.9393	-1.9083	-1.4013	-1.6744	0.9393
-2.2994	1.0718	-1.9083	-1.3394	-1.6744	1.0718
-2.2994	1.2686	-1.9083	-1.2400	-1.6744	1.2686
-2.2994	1.4472	-1.9083	-1.1250	-1.6744	1.4472
-2.2994	1.5785	-1.9083	-1.0125	-1.6744	1.5785
-2.2994	1.7568	-1.9083	-0.9202	-1.6744	1.7568
-2.2994	1.8744	-1.9083	-0.8043	-1.6744	1.8744
-2.2994	1.9574	-1.9083	-0.6768	-1.6744	1.9574
-2.2994	1.9574	-1.9083	-0.5648	-1.6744	1.9574
-2.2994	2.1242	-1.9083	-0.4680	-1.6744	2.1242
-2.2994	2.2609	-1.9083	-0.3060	-1.6744	2.2609
-2.0532	-2.4373	-1.9083	-0.1368	-1.5105	-2.4373
-2.0532	-2.2994	-1.9083	0.0443	-1.5105	-2.2994
-2.0532	-2.0532	-1.9083	0.2642	-1.5105	-2.0532
-2.0532	-1.9083	-1.9083	0.4518	-1.5105	-1.9083
-2.0532	-1.6744	-1.9083	0.6302	-1.5105	-1.6744
-2.0532	-1.5105	-1.9083	0.7924	-1.5105	-1.5105
-2.0532	-1.4013	-1.9083	0.9393	-1.5105	-1.4013
-2.0532	-1.3394	-1.9083	1.0718	-1.5105	-1.3394
-2.0532	-1.2400	-1.9083	1.2686	-1.5105	-1.2400
-2.0532	-1.1250	-1.9083	1.4472	-1.5105	-1.1250
-2.0532	-1.0125	-1.9083	1.5785	-1.5105	-1.0125
-2.0532	-0.9202	-1.9083	1.7568	-1.5105	-0.9202
-2.0532	-0.8043	-1.9083	1.8744	-1.5105	-0.8043
-2.0532	-0.6768	-1.9083	1.9574	-1.5105	-0.6768
-2.0532	-0.5648	-1.9083	1.9574	-1.5105	-0.5648
-2.0532	-0.4680	-1.9083	2.1242	-1.5105	-0.4680
-2.0532	-0.3060	-1.9083	2.2609	-1.5105	-0.3060
-2.0532	-0.1368	-1.6744	-2.4373	-1.5105	-0.1368
-2.0532	0.0443	-1.6744	-2.2994	-1.5105	0.0443

-1.5105	0.2642	-1.3394	-2.0532	-1.2400	0.2642
-1.5105	0.4518	-1.3394	-1.9083	-1.2400	0.4518
-1.5105	0.6302	-1.3394	-1.6744	-1.2400	0.6302
-1.5105	0.7924	-1.3394	-1.5105	-1.2400	0.7924
-1.5105	0.9393	-1.3394	-1.4013	-1.2400	0.9393
-1.5105	1.0718	-1.3394	-1.3394	-1.2400	1.0718
-1.5105	1.2686	-1.3394	-1.2400	-1.2400	1.2686
-1.5105	1.4472	-1.3394	-1.1250	-1.2400	1.4472
-1.5105	1.5785	-1.3394	-1.0125	-1.2400	1.5785
-1.5105	1.7568	-1.3394	-0.9202	-1.2400	1.7568
-1.5105	1.8744	-1.3394	-0.8043	-1.2400	1.8744
-1.5105	1.9574	-1.3394	-0.6768	-1.2400	1.9574
-1.5105	1.9574	-1.3394	-0.5648	-1.2400	1.9574
-1.5105	2.1242	-1.3394	-0.4680	-1.2400	2.1242
-1.5105	2.2609	-1.3394	-0.3060	-1.2400	2.2609
-1.4013	-2.4373	-1.3394	-0.1368	-1.1250	-2.4373
-1.4013	-2.2994	-1.3394	0.0443	-1.1250	-2.2994
-1.4013	-2.0532	-1.3394	0.2642	-1.1250	-2.0532
-1.4013	-1.9083	-1.3394	0.4518	-1.1250	-1.9083
-1.4013	-1.6744	-1.3394	0.6302	-1.1250	-1.6744
-1.4013	-1.5105	-1.3394	0.7924	-1.1250	-1.5105
-1.4013	-1.4013	-1.3394	0.9393	-1.1250	-1.4013
-1.4013	-1.3394	-1.3394	1.0718	-1.1250	-1.3394
-1.4013	-1.2400	-1.3394	1.2686	-1.1250	-1.2400
-1.4013	-1.1250	-1.3394	1.4472	-1.1250	-1.1250
-1.4013	-1.0125	-1.3394	1.5785	-1.1250	-1.0125
-1.4013	-0.9202	-1.3394	1.7568	-1.1250	-0.9202
-1.4013	-0.8043	-1.3394	1.8744	-1.1250	-0.8043
-1.4013	-0.6768	-1.3394	1.9574	-1.1250	-0.6768
-1.4013	-0.5648	-1.3394	1.9574	-1.1250	-0.5648
-1.4013	-0.4680	-1.3394	2.1242	-1.1250	-0.4680
-1.4013	-0.3060	-1.3394	2.2609	-1.1250	-0.3060
-1.4013	-0.1368	-1.2400	-2.4373	-1.1250	-0.1368
-1.4013	0.0443	-1.2400	-2.2994	-1.1250	0.0443
-1.4013	0.2642	-1.2400	-2.0532	-1.1250	0.2642
-1.4013	0.4518	-1.2400	-1.9083	-1.1250	0.4518
-1.4013	0.6302	-1.2400	-1.6744	-1.1250	0.6302
-1.4013	0.7924	-1.2400	-1.5105	-1.1250	0.7924
-1.4013	0.9393	-1.2400	-1.4013	-1.1250	0.9393
-1.4013	1.0718	-1.2400	-1.3394	-1.1250	1.0718
-1.4013	1.2686	-1.2400	-1.2400	-1.1250	1.2686
-1.4013	1.4472	-1.2400	-1.1250	-1.1250	1.4472
-1.4013	1.5785	-1.2400	-1.0125	-1.1250	1.5785
-1.4013	1.7568	-1.2400	-0.9202	-1.1250	1.7568
-1.4013	1.8744	-1.2400	-0.8043	-1.1250	1.8744
-1.4013	1.9574	-1.2400	-0.6768	-1.1250	1.9574
-1.4013	1.9574	-1.2400	-0.5648	-1.1250	1.9574
-1.4013	2.1242	-1.2400	-0.4680	-1.1250	2.1242
-1.4013	2.2609	-1.2400	-0.3060	-1.1250	2.2609
-1.3394	-2.4373	-1.2400	-0.1368	-1.0125	-2.4373
-1.3394	-2.2994	-1.2400	0.0443	-1.0125	-2.2994

-1.0125	-2.0532	-0.9202	0.2642	-0.6768	-2.0532
-1.0125	-1.9083	-0.9202	0.4518	-0.6768	-1.9083
-1.0125	-1.6744	-0.9202	0.6302	-0.6768	-1.6744
-1.0125	-1.5105	-0.9202	0.7924	-0.6768	-1.5105
-1.0125	-1.4013	-0.9202	0.9393	-0.6768	-1.4013
-1.0125	-1.3394	-0.9202	1.0718	-0.6768	-1.3394
-1.0125	-1.2400	-0.9202	1.2686	-0.6768	-1.2400
-1.0125	-1.1250	-0.9202	1.4472	-0.6768	-1.1250
-1.0125	-1.0125	-0.9202	1.5785	-0.6768	-1.0125
-1.0125	-0.9202	-0.9202	1.7568	-0.6768	-0.9202
-1.0125	-0.8043	-0.9202	1.8744	-0.6768	-0.8043
-1.0125	-0.6768	-0.9202	1.9574	-0.6768	-0.6768
-1.0125	-0.5648	-0.9202	1.9574	-0.6768	-0.5648
-1.0125	-0.4680	-0.9202	2.1242	-0.6768	-0.4680
-1.0125	-0.3060	-0.9202	2.2609	-0.6768	-0.3060
-1.0125	-0.1368	-0.8043	-2.4373	-0.6768	-0.1368
-1.0125	0.0443	-0.8043	-2.2994	-0.6768	0.0443
-1.0125	0.2642	-0.8043	-2.0532	-0.6768	0.2642
-1.0125	0.4518	-0.8043	-1.9083	-0.6768	0.4518
-1.0125	0.6302	-0.8043	-1.6744	-0.6768	0.6302
-1.0125	0.7924	-0.8043	-1.5105	-0.6768	0.7924
-1.0125	0.9393	-0.8043	-1.4013	-0.6768	0.9393
-1.0125	1.0718	-0.8043	-1.3394	-0.6768	1.0718
-1.0125	1.2686	-0.8043	-1.2400	-0.6768	1.2686
-1.0125	1.4472	-0.8043	-1.1250	-0.6768	1.4472
-1.0125	1.5785	-0.8043	-1.0125	-0.6768	1.5785
-1.0125	1.7568	-0.8043	-0.9202	-0.6768	1.7568
-1.0125	1.8744	-0.8043	-0.8043	-0.6768	1.8744
-1.0125	1.9574	-0.8043	-0.6768	-0.6768	1.9574
-1.0125	1.9574	-0.8043	-0.5648	-0.6768	1.9574
-1.0125	2.1242	-0.8043	-0.4680	-0.6768	2.1242
-1.0125	2.2609	-0.8043	-0.3060	-0.6768	2.2609
-0.9202	-2.4373	-0.8043	-0.1368	-0.5648	-2.4373
-0.9202	-2.2994	-0.8043	0.0443	-0.5648	-2.2994
-0.9202	-2.0532	-0.8043	0.2642	-0.5648	-2.0532
-0.9202	-1.9083	-0.8043	0.4518	-0.5648	-1.9083
-0.9202	-1.6744	-0.8043	0.6302	-0.5648	-1.6744
-0.9202	-1.5105	-0.8043	0.7924	-0.5648	-1.5105
-0.9202	-1.4013	-0.8043	0.9393	-0.5648	-1.4013
-0.9202	-1.3394	-0.8043	1.0718	-0.5648	-1.3394
-0.9202	-1.2400	-0.8043	1.2686	-0.5648	-1.2400
-0.9202	-1.1250	-0.8043	1.4472	-0.5648	-1.1250
-0.9202	-1.0125	-0.8043	1.5785	-0.5648	-1.0125
-0.9202	-0.9202	-0.8043	1.7568	-0.5648	-0.9202
-0.9202	-0.8043	-0.8043	1.8744	-0.5648	-0.8043
-0.9202	-0.6768	-0.8043	1.9574	-0.5648	-0.6768
-0.9202	-0.5648	-0.8043	1.9574	-0.5648	-0.5648
-0.9202	-0.4680	-0.8043	2.1242	-0.5648	-0.4680
-0.9202	-0.3060	-0.8043	2.2609	-0.5648	-0.3060
-0.9202	-0.1368	-0.6768	-2.4373	-0.5648	-0.1368
-0.9202	0.0443	-0.6768	-2.2994	-0.5648	0.0443

-0.5648	0.2642	-0.3060	-2.0532	-0.1368	0.2642
-0.5648	0.4518	-0.3060	-1.9083	-0.1368	0.4518
-0.5648	0.6302	-0.3060	-1.6744	-0.1368	0.6302
-0.5648	0.7924	-0.3060	-1.5105	-0.1368	0.7924
-0.5648	0.9393	-0.3060	-1.4013	-0.1368	0.9393
-0.5648	1.0718	-0.3060	-1.3394	-0.1368	1.0718
-0.5648	1.2686	-0.3060	-1.2400	-0.1368	1.2686
-0.5648	1.4472	-0.3060	-1.1250	-0.1368	1.4472
-0.5648	1.5785	-0.3060	-1.0125	-0.1368	1.5785
-0.5648	1.7568	-0.3060	-0.9202	-0.1368	1.7568
-0.5648	1.8744	-0.3060	-0.8043	-0.1368	1.8744
-0.5648	1.9574	-0.3060	-0.6768	-0.1368	1.9574
-0.5648	1.9574	-0.3060	-0.5648	-0.1368	1.9574
-0.5648	2.1242	-0.3060	-0.4680	-0.1368	2.1242
-0.5648	2.2609	-0.3060	-0.3060	-0.1368	2.2609
-0.4680	-2.4373	-0.3060	-0.1368	0.0443	-2.4373
-0.4680	-2.2994	-0.3060	0.0443	0.0443	-2.2994
-0.4680	-2.0532	-0.3060	0.2642	0.0443	-2.0532
-0.4680	-1.9083	-0.3060	0.4518	0.0443	-1.9083
-0.4680	-1.6744	-0.3060	0.6302	0.0443	-1.6744
-0.4680	-1.5105	-0.3060	0.7924	0.0443	-1.5105
-0.4680	-1.4013	-0.3060	0.9393	0.0443	-1.4013
-0.4680	-1.3394	-0.3060	1.0718	0.0443	-1.3394
-0.4680	-1.2400	-0.3060	1.2686	0.0443	-1.2400
-0.4680	-1.1250	-0.3060	1.4472	0.0443	-1.1250
-0.4680	-1.0125	-0.3060	1.5785	0.0443	-1.0125
-0.4680	-0.9202	-0.3060	1.7568	0.0443	-0.9202
-0.4680	-0.8043	-0.3060	1.8744	0.0443	-0.8043
-0.4680	-0.6768	-0.3060	1.9574	0.0443	-0.6768
-0.4680	-0.5648	-0.3060	1.9574	0.0443	-0.5648
-0.4680	-0.4680	-0.3060	2.1242	0.0443	-0.4680
-0.4680	-0.3060	-0.3060	2.2609	0.0443	-0.3060
-0.4680	-0.1368	-0.1368	-2.4373	0.0443	-0.1368
-0.4680	0.0443	-0.1368	-2.2994	0.0443	0.0443
-0.4680	0.2642	-0.1368	-2.0532	0.0443	0.2642
-0.4680	0.4518	-0.1368	-1.9083	0.0443	0.4518
-0.4680	0.6302	-0.1368	-1.6744	0.0443	0.6302
-0.4680	0.7924	-0.1368	-1.5105	0.0443	0.7924
-0.4680	0.9393	-0.1368	-1.4013	0.0443	0.9393
-0.4680	1.0718	-0.1368	-1.3394	0.0443	1.0718
-0.4680	1.2686	-0.1368	-1.2400	0.0443	1.2686
-0.4680	1.4472	-0.1368	-1.1250	0.0443	1.4472
-0.4680	1.5785	-0.1368	-1.0125	0.0443	1.5785
-0.4680	1.7568	-0.1368	-0.9202	0.0443	1.7568
-0.4680	1.8744	-0.1368	-0.8043	0.0443	1.8744
-0.4680	1.9574	-0.1368	-0.6768	0.0443	1.9574
-0.4680	1.9574	-0.1368	-0.5648	0.0443	1.9574
-0.4680	2.1242	-0.1368	-0.4680	0.0443	2.1242
-0.4680	2.2609	-0.1368	-0.3060	0.0443	2.2609
-0.3060	-2.4373	-0.1368	-0.1368	0.2642	-2.4373
-0.3060	-2.2994	-0.1368	0.0443	0.2642	-2.2994

0.2642	-2.0532	0.4518	0.2642	0.7924	-2.0532
0.2642	-1.9083	0.4518	0.4518	0.7924	-1.9083
0.2642	-1.6744	0.4518	0.6302	0.7924	-1.6744
0.2642	-1.5105	0.4518	0.7924	0.7924	-1.5105
0.2642	-1.4013	0.4518	0.9393	0.7924	-1.4013
0.2642	-1.3394	0.4518	1.0718	0.7924	-1.3394
0.2642	-1.2400	0.4518	1.2686	0.7924	-1.2400
0.2642	-1.1250	0.4518	1.4472	0.7924	-1.1250
0.2642	-1.0125	0.4518	1.5785	0.7924	-1.0125
0.2642	-0.9202	0.4518	1.7568	0.7924	-0.9202
0.2642	-0.8043	0.4518	1.8744	0.7924	-0.8043
0.2642	-0.6768	0.4518	1.9574	0.7924	-0.6768
0.2642	-0.5648	0.4518	1.9574	0.7924	-0.5648
0.2642	-0.4680	0.4518	2.1242	0.7924	-0.4680
0.2642	-0.3060	0.4518	2.2609	0.7924	-0.3060
0.2642	-0.1368	0.6302	-2.4373	0.7924	-0.1368
0.2642	0.0443	0.6302	-2.2994	0.7924	0.0443
0.2642	0.2642	0.6302	-2.0532	0.7924	0.2642
0.2642	0.4518	0.6302	-1.9083	0.7924	0.4518
0.2642	0.6302	0.6302	-1.6744	0.7924	0.6302
0.2642	0.7924	0.6302	-1.5105	0.7924	0.7924
0.2642	0.9393	0.6302	-1.4013	0.7924	0.9393
0.2642	1.0718	0.6302	-1.3394	0.7924	1.0718
0.2642	1.2686	0.6302	-1.2400	0.7924	1.2686
0.2642	1.4472	0.6302	-1.1250	0.7924	1.4472
0.2642	1.5785	0.6302	-1.0125	0.7924	1.5785
0.2642	1.7568	0.6302	-0.9202	0.7924	1.7568
0.2642	1.8744	0.6302	-0.8043	0.7924	1.8744
0.2642	1.9574	0.6302	-0.6768	0.7924	1.9574
0.2642	1.9574	0.6302	-0.5648	0.7924	1.9574
0.2642	2.1242	0.6302	-0.4680	0.7924	2.1242
0.2642	2.2609	0.6302	-0.3060	0.7924	2.2609
0.4518	-2.4373	0.6302	-0.1368	0.9393	-2.4373
0.4518	-2.2994	0.6302	0.0443	0.9393	-2.2994
0.4518	-2.0532	0.6302	0.2642	0.9393	-2.0532
0.4518	-1.9083	0.6302	0.4518	0.9393	-1.9083
0.4518	-1.6744	0.6302	0.6302	0.9393	-1.6744
0.4518	-1.5105	0.6302	0.7924	0.9393	-1.5105
0.4518	-1.4013	0.6302	0.9393	0.9393	-1.4013
0.4518	-1.3394	0.6302	1.0718	0.9393	-1.3394
0.4518	-1.2400	0.6302	1.2686	0.9393	-1.2400
0.4518	-1.1250	0.6302	1.4472	0.9393	-1.1250
0.4518	-1.0125	0.6302	1.5785	0.9393	-1.0125
0.4518	-0.9202	0.6302	1.7568	0.9393	-0.9202
0.4518	-0.8043	0.6302	1.8744	0.9393	-0.8043
0.4518	-0.6768	0.6302	1.9574	0.9393	-0.6768
0.4518	-0.5648	0.6302	1.9574	0.9393	-0.5648
0.4518	-0.4680	0.6302	2.1242	0.9393	-0.4680
0.4518	-0.3060	0.6302	2.2609	0.9393	-0.3060
0.4518	-0.1368	0.7924	-2.4373	0.9393	-0.1368
0.4518	0.0443	0.7924	-2.2994	0.9393	0.0443

0.9393	0.2642	1.2686	-2.0532	1.4472	0.2642
0.9393	0.4518	1.2686	-1.9083	1.4472	0.4518
0.9393	0.6302	1.2686	-1.6744	1.4472	0.6302
0.9393	0.7924	1.2686	-1.5105	1.4472	0.7924
0.9393	0.9393	1.2686	-1.4013	1.4472	0.9393
0.9393	1.0718	1.2686	-1.3394	1.4472	1.0718
0.9393	1.2686	1.2686	-1.2400	1.4472	1.2686
0.9393	1.4472	1.2686	-1.1250	1.4472	1.4472
0.9393	1.5785	1.2686	-1.0125	1.4472	1.5785
0.9393	1.7568	1.2686	-0.9202	1.4472	1.7568
0.9393	1.8744	1.2686	-0.8043	1.4472	1.8744
0.9393	1.9574	1.2686	-0.6768	1.4472	1.9574
0.9393	1.9574	1.2686	-0.5648	1.4472	1.9574
0.9393	2.1242	1.2686	-0.4680	1.4472	2.1242
0.9393	2.2609	1.2686	-0.3060	1.4472	2.2609
1.0718	-2.4373	1.2686	-0.1368	1.5785	-2.4373
1.0718	-2.2994	1.2686	0.0443	1.5785	-2.2994
1.0718	-2.0532	1.2686	0.2642	1.5785	-2.0532
1.0718	-1.9083	1.2686	0.4518	1.5785	-1.9083
1.0718	-1.6744	1.2686	0.6302	1.5785	-1.6744
1.0718	-1.5105	1.2686	0.7924	1.5785	-1.5105
1.0718	-1.4013	1.2686	0.9393	1.5785	-1.4013
1.0718	-1.3394	1.2686	1.0718	1.5785	-1.3394
1.0718	-1.2400	1.2686	1.2686	1.5785	-1.2400
1.0718	-1.1250	1.2686	1.4472	1.5785	-1.1250
1.0718	-1.0125	1.2686	1.5785	1.5785	-1.0125
1.0718	-0.9202	1.2686	1.7568	1.5785	-0.9202
1.0718	-0.8043	1.2686	1.8744	1.5785	-0.8043
1.0718	-0.6768	1.2686	1.9574	1.5785	-0.6768
1.0718	-0.5648	1.2686	1.9574	1.5785	-0.5648
1.0718	-0.4680	1.2686	2.1242	1.5785	-0.4680
1.0718	-0.3060	1.2686	2.2609	1.5785	-0.3060
1.0718	-0.1368	1.4472	-2.4373	1.5785	-0.1368
1.0718	0.0443	1.4472	-2.2994	1.5785	0.0443
1.0718	0.2642	1.4472	-2.0532	1.5785	0.2642
1.0718	0.4518	1.4472	-1.9083	1.5785	0.4518
1.0718	0.6302	1.4472	-1.6744	1.5785	0.6302
1.0718	0.7924	1.4472	-1.5105	1.5785	0.7924
1.0718	0.9393	1.4472	-1.4013	1.5785	0.9393
1.0718	1.0718	1.4472	-1.3394	1.5785	1.0718
1.0718	1.2686	1.4472	-1.2400	1.5785	1.2686
1.0718	1.4472	1.4472	-1.1250	1.5785	1.4472
1.0718	1.5785	1.4472	-1.0125	1.5785	1.5785
1.0718	1.7568	1.4472	-0.9202	1.5785	1.7568
1.0718	1.8744	1.4472	-0.8043	1.5785	1.8744
1.0718	1.9574	1.4472	-0.6768	1.5785	1.9574
1.0718	1.9574	1.4472	-0.5648	1.5785	1.9574
1.0718	2.1242	1.4472	-0.4680	1.5785	2.1242
1.0718	2.2609	1.4472	-0.3060	1.5785	2.2609
1.2686	-2.4373	1.4472	-0.1368	1.7568	-2.4373
1.2686	-2.2994	1.4472	0.0443	1.7568	-2.2994

1.7568	-2.0532	1.8744	0.2642	1.9574	-2.0532
1.7568	-1.9083	1.8744	0.4518	1.9574	-1.9083
1.7568	-1.6744	1.8744	0.6302	1.9574	-1.6744
1.7568	-1.5105	1.8744	0.7924	1.9574	-1.5105
1.7568	-1.4013	1.8744	0.9393	1.9574	-1.4013
1.7568	-1.3394	1.8744	1.0718	1.9574	-1.3394
1.7568	-1.2400	1.8744	1.2686	1.9574	-1.2400
1.7568	-1.1250	1.8744	1.4472	1.9574	-1.1250
1.7568	-1.0125	1.8744	1.5785	1.9574	-1.0125
1.7568	-0.9202	1.8744	1.7568	1.9574	-0.9202
1.7568	-0.8043	1.8744	1.8744	1.9574	-0.8043
1.7568	-0.6768	1.8744	1.9574	1.9574	-0.6768
1.7568	-0.5648	1.8744	1.9574	1.9574	-0.5648
1.7568	-0.4680	1.8744	2.1242	1.9574	-0.4680
1.7568	-0.3060	1.8744	2.2609	1.9574	-0.3060
1.7568	-0.1368	1.9574	-2.4373	1.9574	-0.1368
1.7568	0.0443	1.9574	-2.2994	1.9574	0.0443
1.7568	0.2642	1.9574	-2.0532	1.9574	0.2642
1.7568	0.4518	1.9574	-1.9083	1.9574	0.4518
1.7568	0.6302	1.9574	-1.6744	1.9574	0.6302
1.7568	0.7924	1.9574	-1.5105	1.9574	0.7924
1.7568	0.9393	1.9574	-1.4013	1.9574	0.9393
1.7568	1.0718	1.9574	-1.3394	1.9574	1.0718
1.7568	1.2686	1.9574	-1.2400	1.9574	1.2686
1.7568	1.4472	1.9574	-1.1250	1.9574	1.4472
1.7568	1.5785	1.9574	-1.0125	1.9574	1.5785
1.7568	1.7568	1.9574	-0.9202	1.9574	1.7568
1.7568	1.8744	1.9574	-0.8043	1.9574	1.8744
1.7568	1.9574	1.9574	-0.6768	1.9574	1.9574
1.7568	1.9574	1.9574	-0.5648	1.9574	1.9574
1.7568	2.1242	1.9574	-0.4680	1.9574	2.1242
1.7568	2.2609	1.9574	-0.3060	1.9574	2.2609
1.8744	-2.4373	1.9574	-0.1368	2.1242	-2.4373
1.8744	-2.2994	1.9574	0.0443	2.1242	-2.2994
1.8744	-2.0532	1.9574	0.2642	2.1242	-2.0532
1.8744	-1.9083	1.9574	0.4518	2.1242	-1.9083
1.8744	-1.6744	1.9574	0.6302	2.1242	-1.6744
1.8744	-1.5105	1.9574	0.7924	2.1242	-1.5105
1.8744	-1.4013	1.9574	0.9393	2.1242	-1.4013
1.8744	-1.3394	1.9574	1.0718	2.1242	-1.3394
1.8744	-1.2400	1.9574	1.2686	2.1242	-1.2400
1.8744	-1.1250	1.9574	1.4472	2.1242	-1.1250
1.8744	-1.0125	1.9574	1.5785	2.1242	-1.0125
1.8744	-0.9202	1.9574	1.7568	2.1242	-0.9202
1.8744	-0.8043	1.9574	1.8744	2.1242	-0.8043
1.8744	-0.6768	1.9574	1.9574	2.1242	-0.6768
1.8744	-0.5648	1.9574	1.9574	2.1242	-0.5648
1.8744	-0.4680	1.9574	2.1242	2.1242	-0.4680
1.8744	-0.3060	1.9574	2.2609	2.1242	-0.3060
1.8744	-0.1368	1.9574	-2.4373	2.1242	-0.1368
1.8744	0.0443	1.9574	-2.2994	2.1242	0.0443

2.1242	0.2642	2.2609	-2.0532	2.2609	0.2642
2.1242	0.4518	2.2609	-1.9083	2.2609	0.4518
2.1242	0.6302	2.2609	-1.6744	2.2609	0.6302
2.1242	0.7924	2.2609	-1.5105	2.2609	0.7924
2.1242	0.9393	2.2609	-1.4013	2.2609	0.9393
2.1242	1.0718	2.2609	-1.3394	2.2609	1.0718
2.1242	1.2686	2.2609	-1.2400	2.2609	1.2686
2.1242	1.4472	2.2609	-1.1250	2.2609	1.4472
2.1242	1.5785	2.2609	-1.0125	2.2609	1.5785
2.1242	1.7568	2.2609	-0.9202	2.2609	1.7568
2.1242	1.8744	2.2609	-0.8043	2.2609	1.8744
2.1242	1.9574	2.2609	-0.6768	2.2609	1.9574
2.1242	1.9574	2.2609	-0.5648	2.2609	1.9574
2.1242	2.1242	2.2609	-0.4680	2.2609	2.1242
2.1242	2.2609	2.2609	-0.3060	2.2609	2.2609
2.2609	-2.4373	2.2609	-0.1368		
2.2609	-2.2994	2.2609	0.0443		

Parámetros de Diseño: Codebook generado a partir de 500 vectores aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.1.

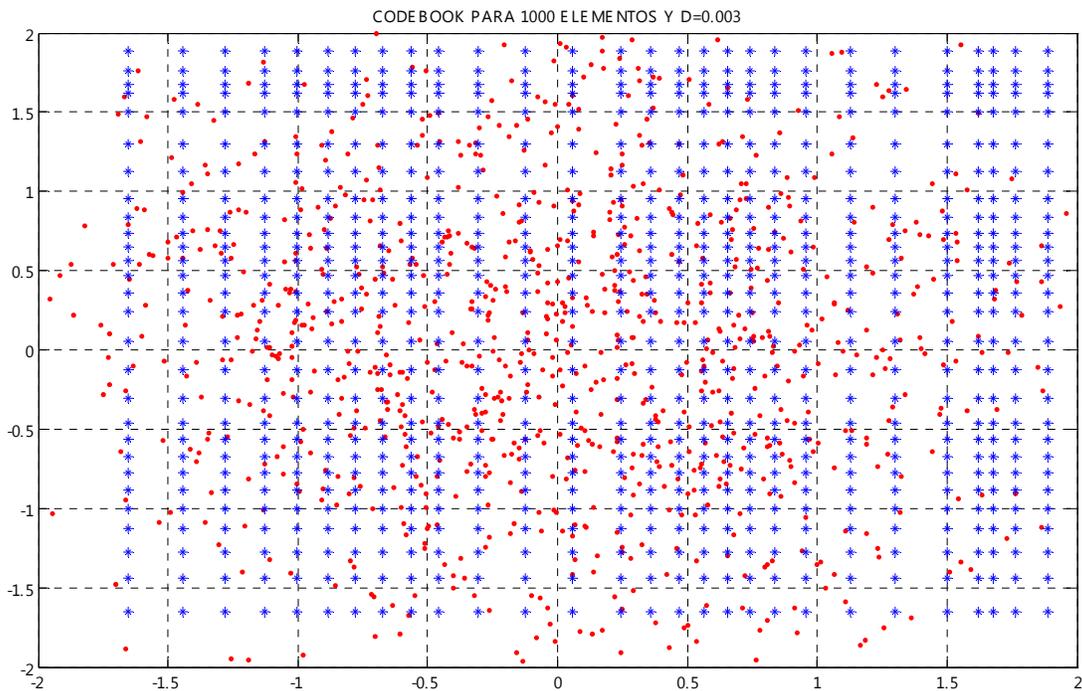
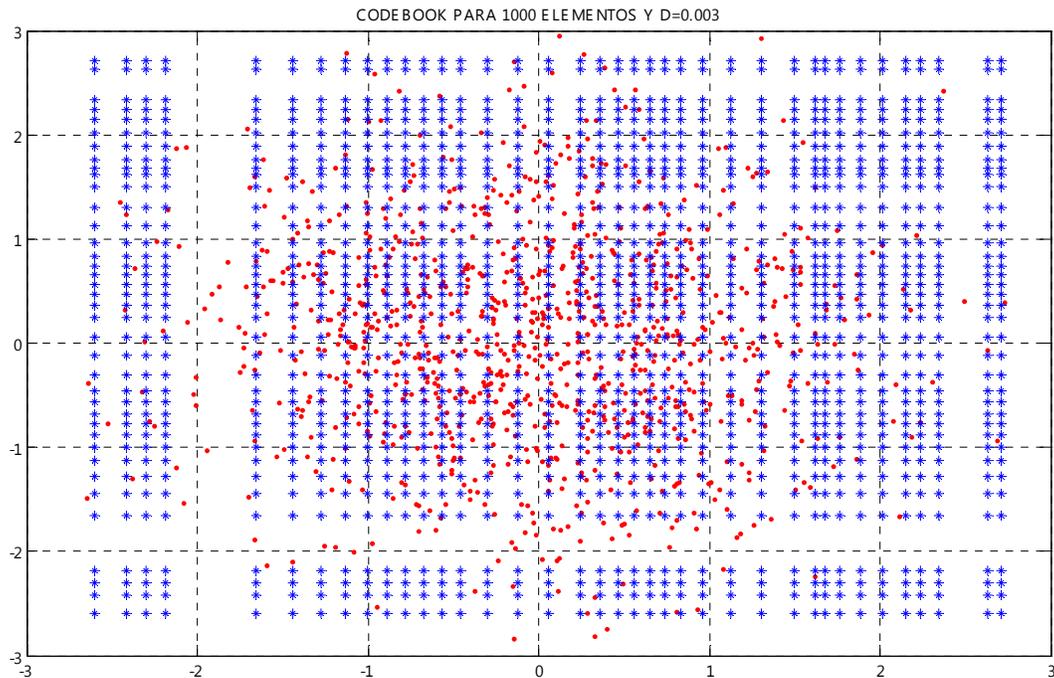
-1.8209	-1.8209	-0.6211	0.9448	0.5410	0.1838
-1.8209	-1.0588	-0.6211	1.6728	0.5410	0.5410
-1.8209	-0.6211	-0.2145	-1.8209	0.5410	0.9448
-1.8209	-0.2145	-0.2145	-1.0588	0.5410	1.6728
-1.8209	0.1838	-0.2145	-0.6211	0.9448	-1.8209
-1.8209	0.5410	-0.2145	-0.2145	0.9448	-1.0588
-1.8209	0.9448	-0.2145	0.1838	0.9448	-0.6211
-1.8209	1.6728	-0.2145	0.5410	0.9448	-0.2145
-1.0588	-1.8209	-0.2145	0.9448	0.9448	0.1838
-1.0588	-1.0588	-0.2145	1.6728	0.9448	0.5410
-1.0588	-0.6211	0.1838	-1.8209	0.9448	0.9448
-1.0588	-0.2145	0.1838	-1.0588	0.9448	1.6728
-1.0588	0.1838	0.1838	-0.6211	1.6728	-1.8209
-1.0588	0.5410	0.1838	-0.2145	1.6728	-1.0588
-1.0588	0.9448	0.1838	0.1838	1.6728	-0.6211
-1.0588	1.6728	0.1838	0.5410	1.6728	-0.2145
-0.6211	-1.8209	0.1838	0.9448	1.6728	0.1838
-0.6211	-1.0588	0.1838	1.6728	1.6728	0.5410
-0.6211	-0.6211	0.5410	-1.8209	1.6728	0.9448
-0.6211	-0.2145	0.5410	-1.0588	1.6728	1.6728
-0.6211	0.1838	0.5410	-0.6211		
-0.6211	0.5410	0.5410	-0.2145		

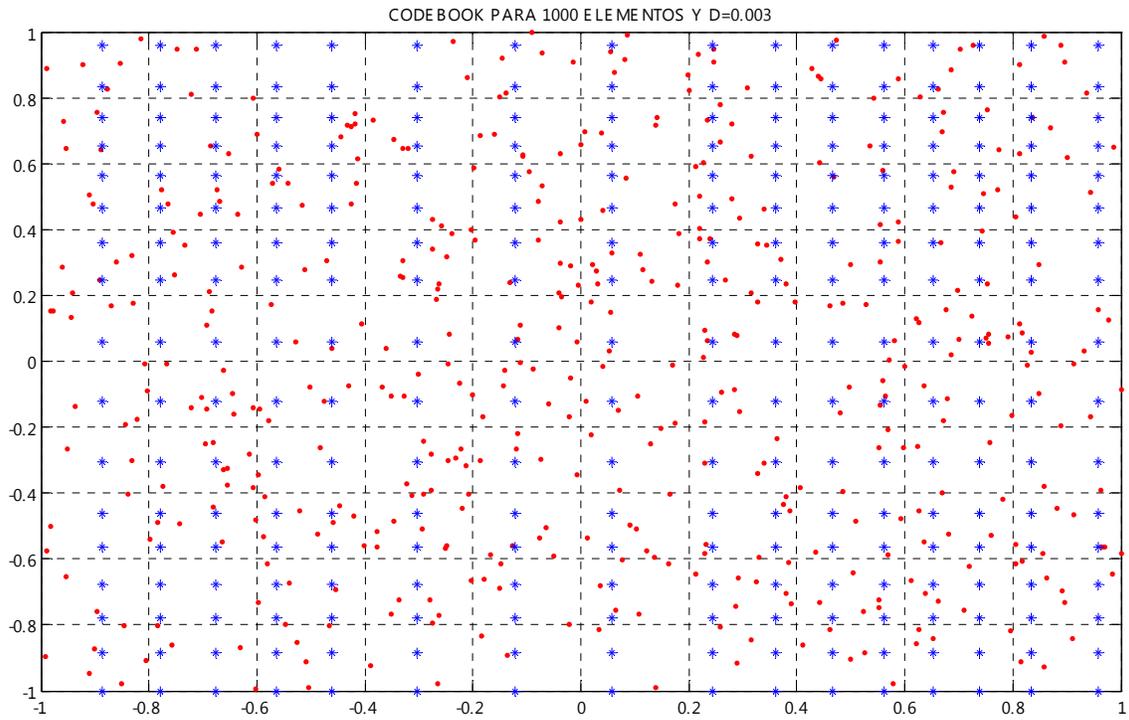
Parámetros de Diseño: Codebook generado a partir de 500 vectores aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.5.

-0.7753	-0.7753	0.8518	-0.7753
-0.7753	0.8518	0.8518	0.8518

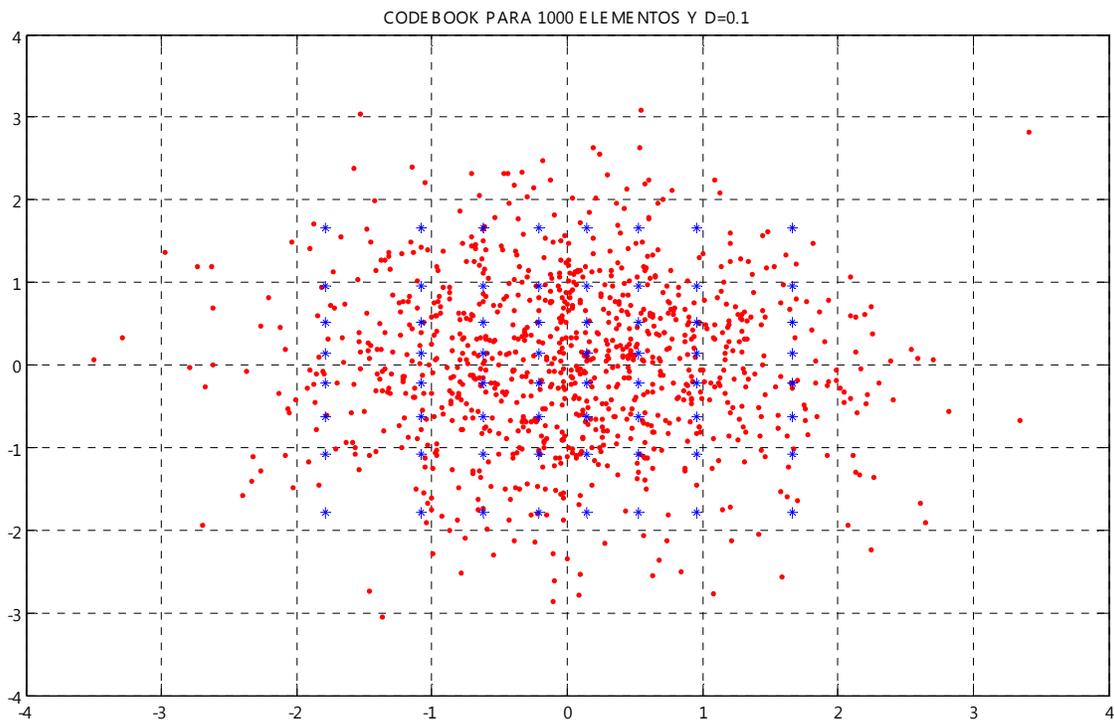
ANEXO 8: ALGORITMO COVQ (DISTRIBUCIÓN NORMAL)

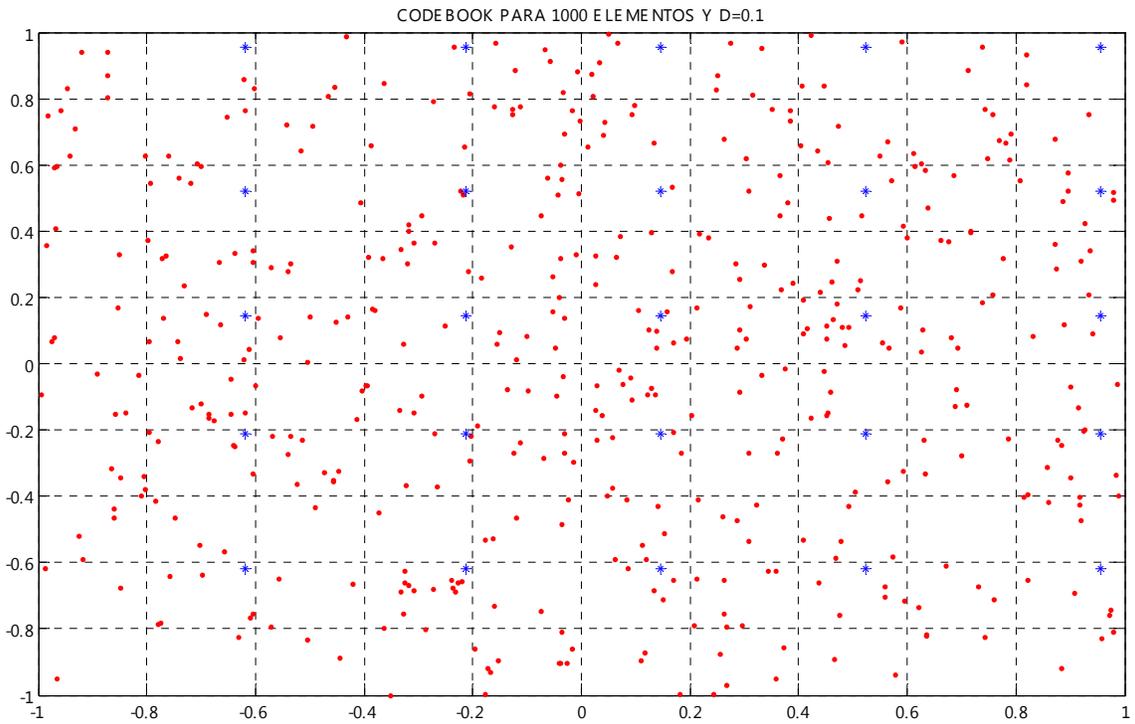
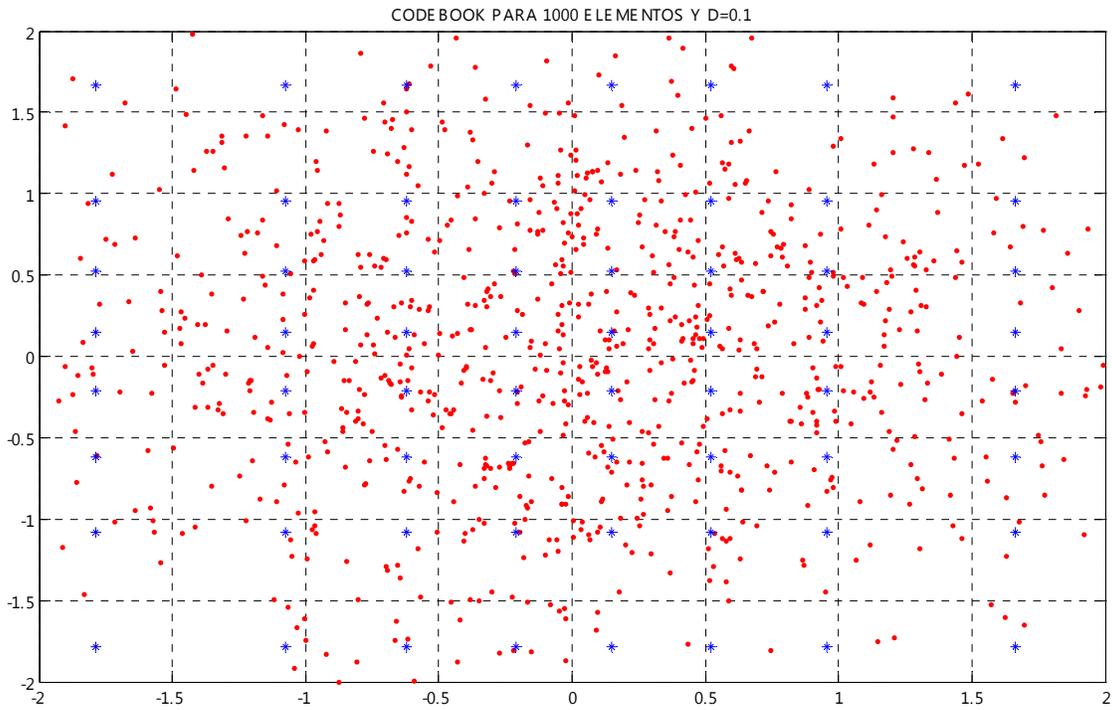
Parámetros de Diseño: Centroides generados a partir de 1 000 vectores aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.003.



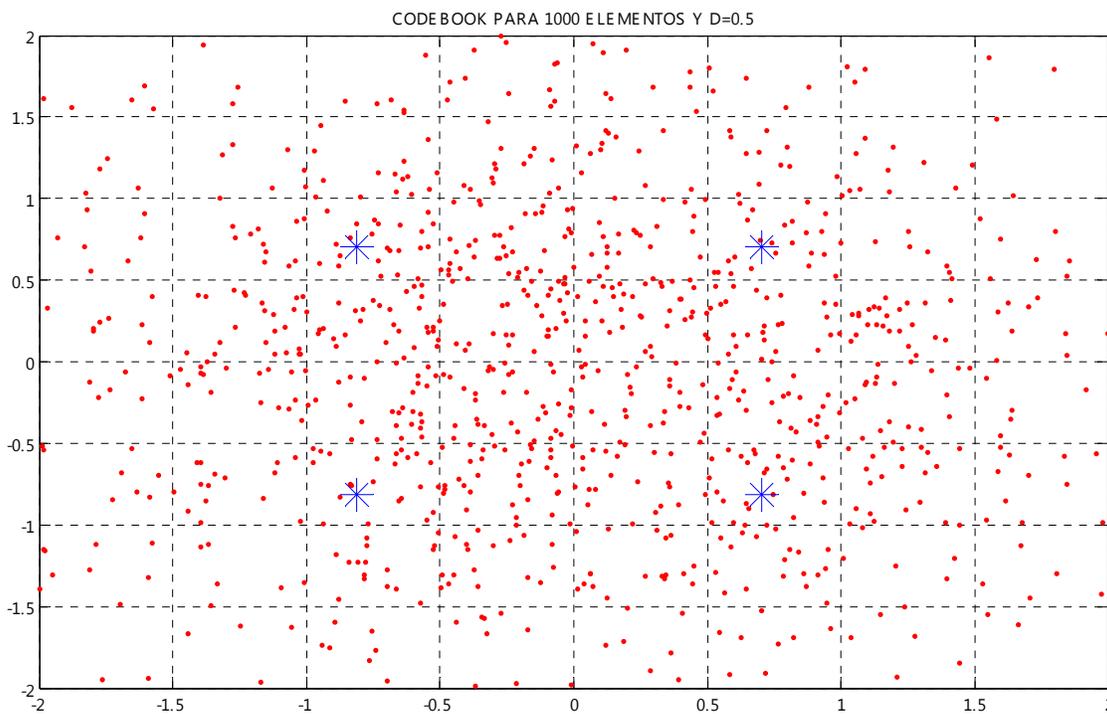
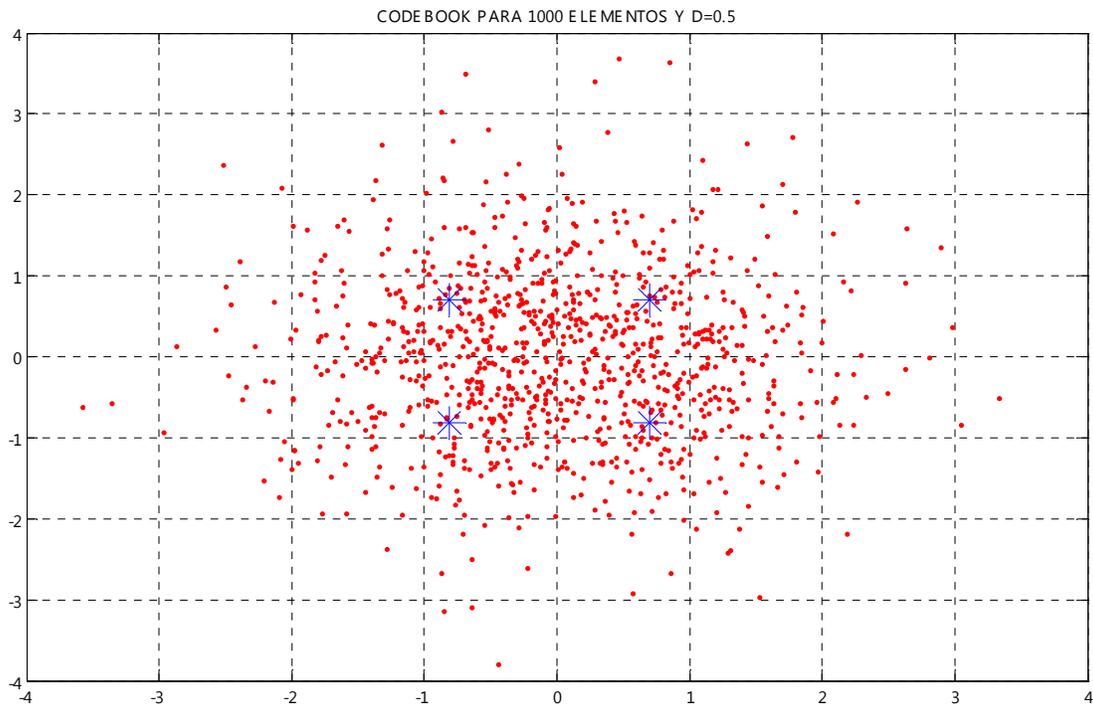


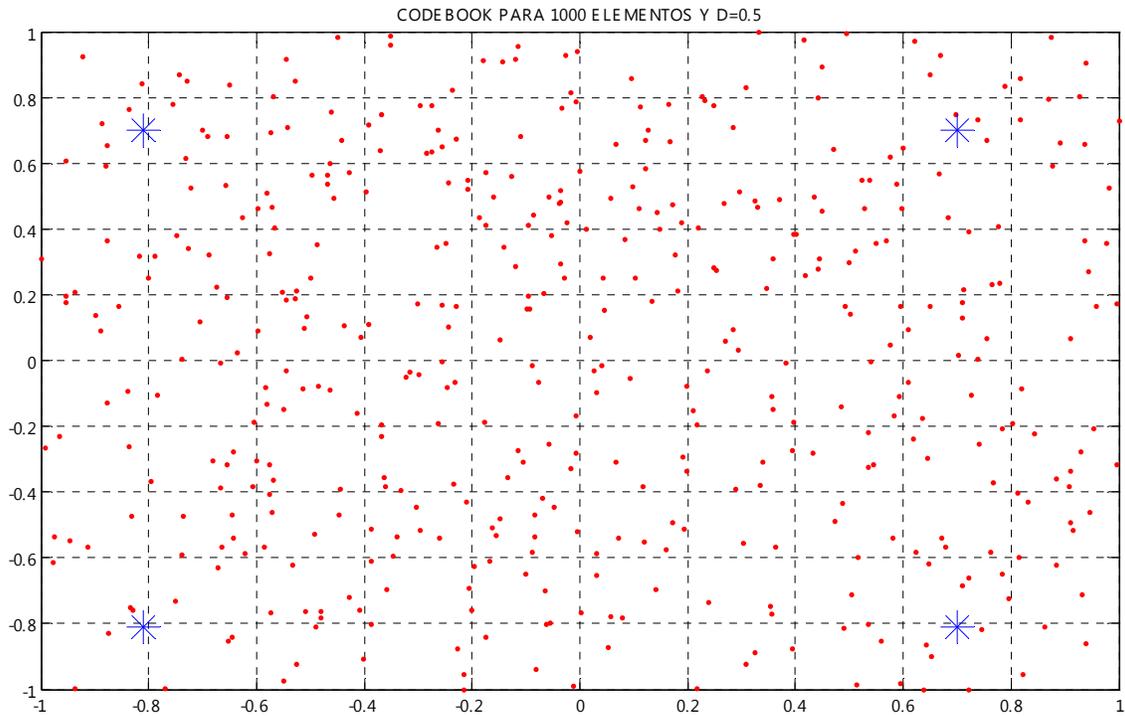
Parámetros de Diseño: Centroides generados a partir de 1 000 vectores aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.1.



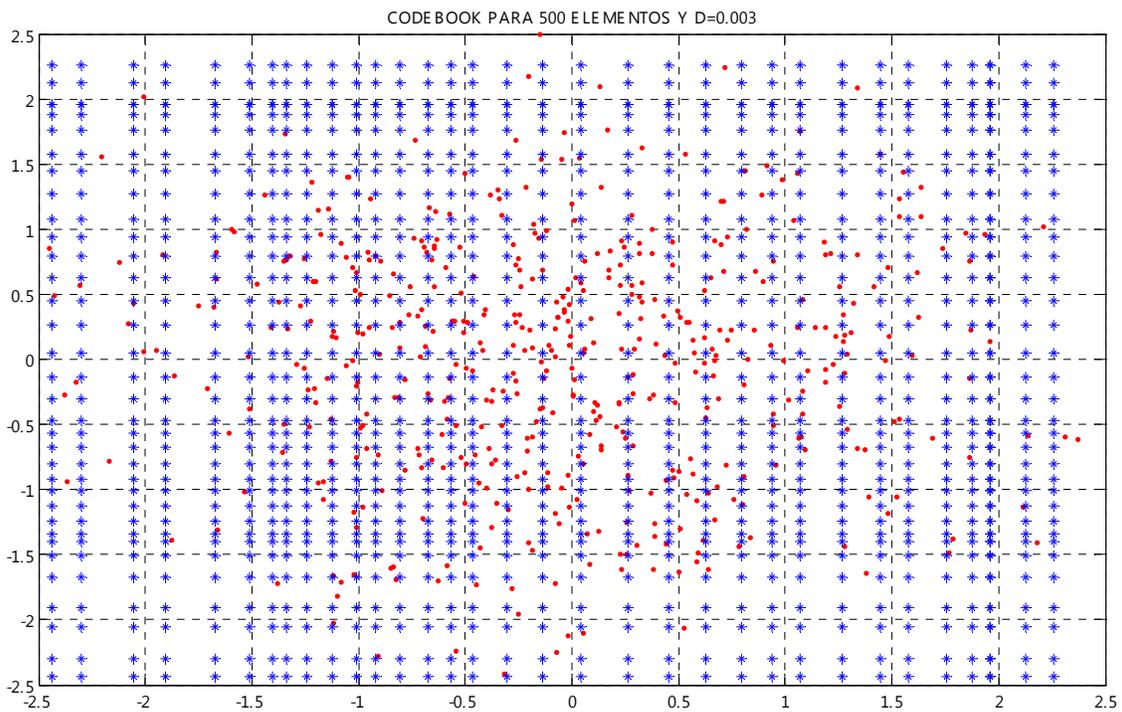


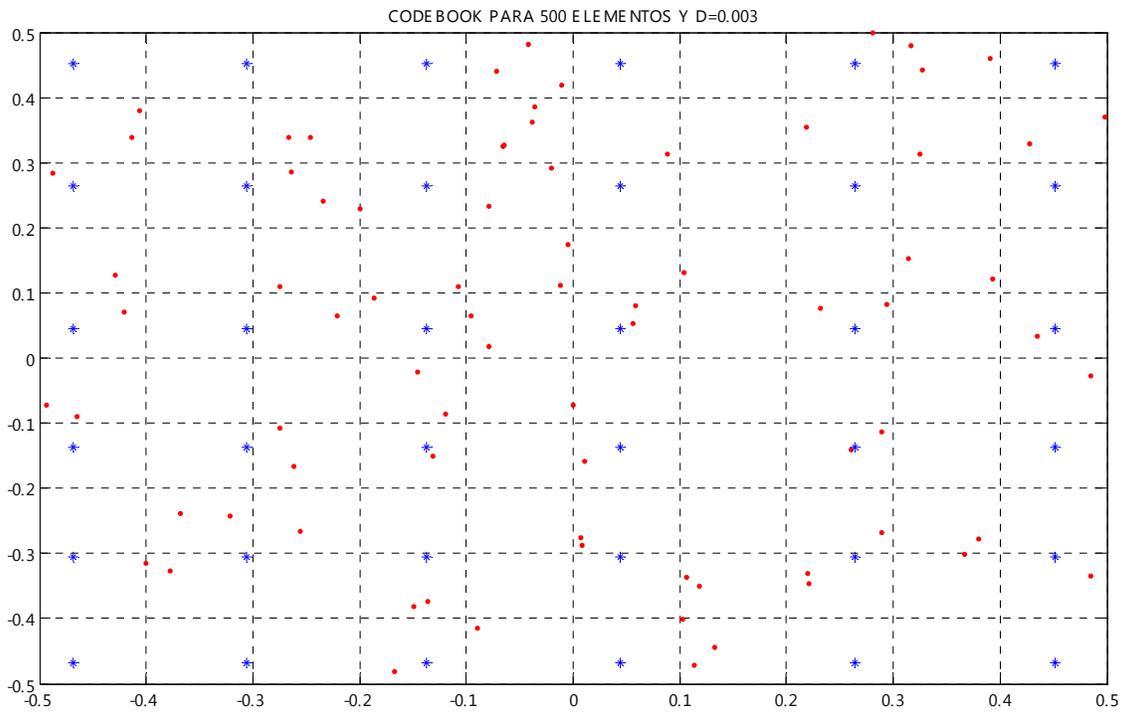
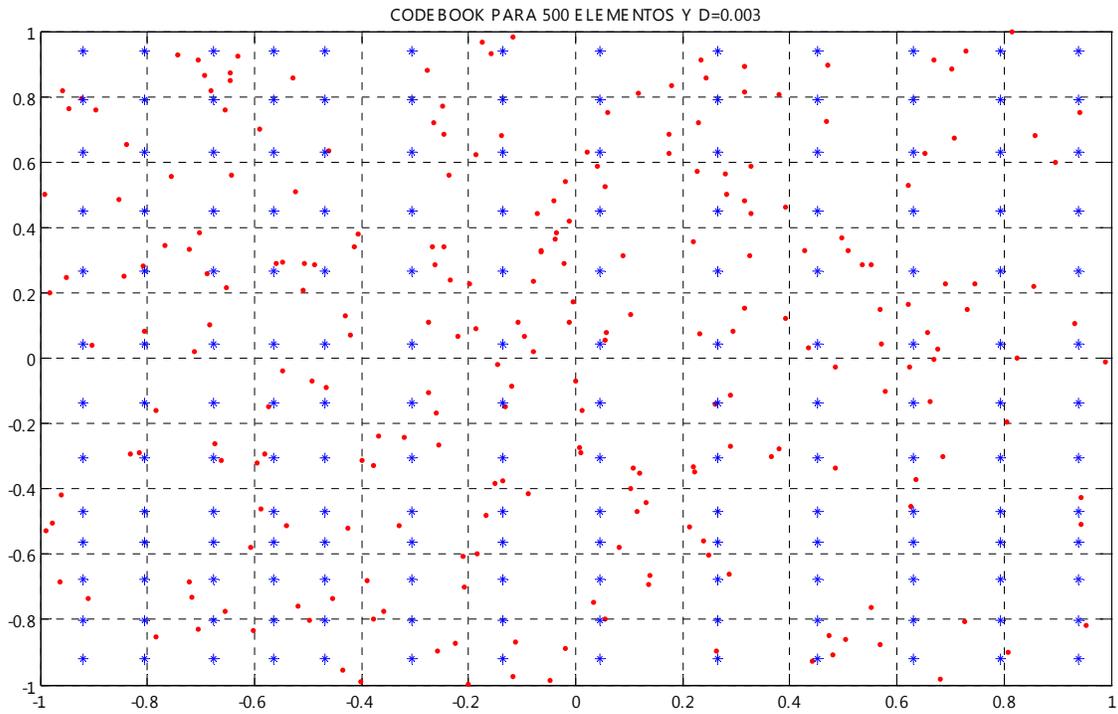
Parámetros de Diseño: Centroides generados a partir de 1 000 vectores aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.5.



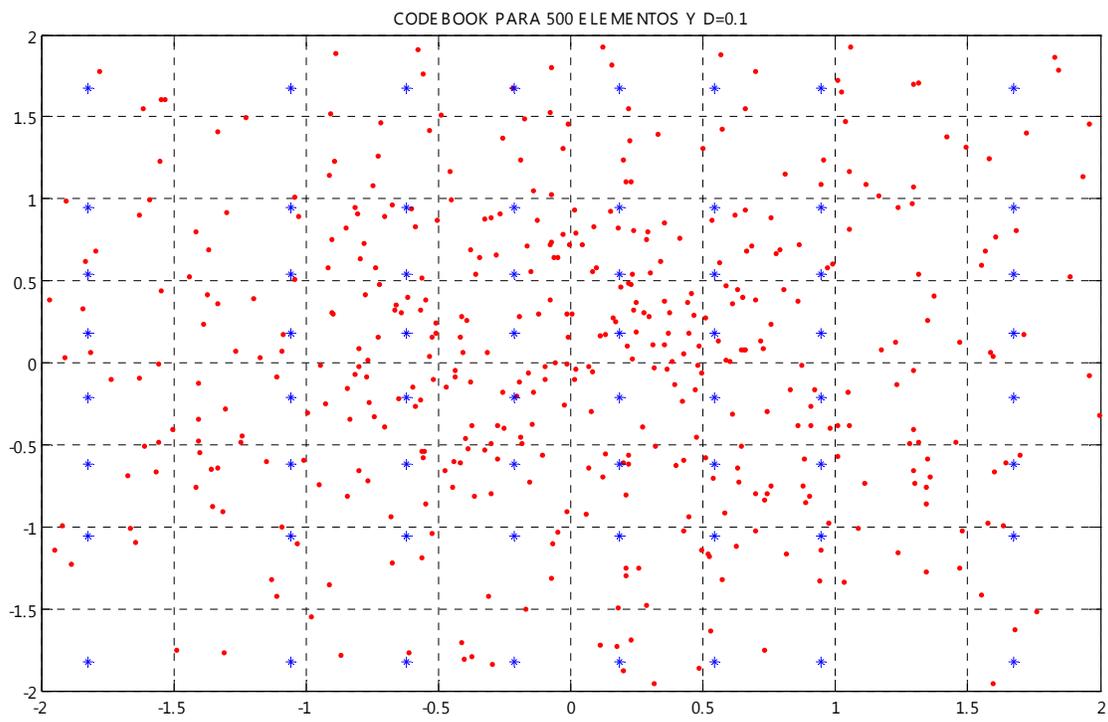
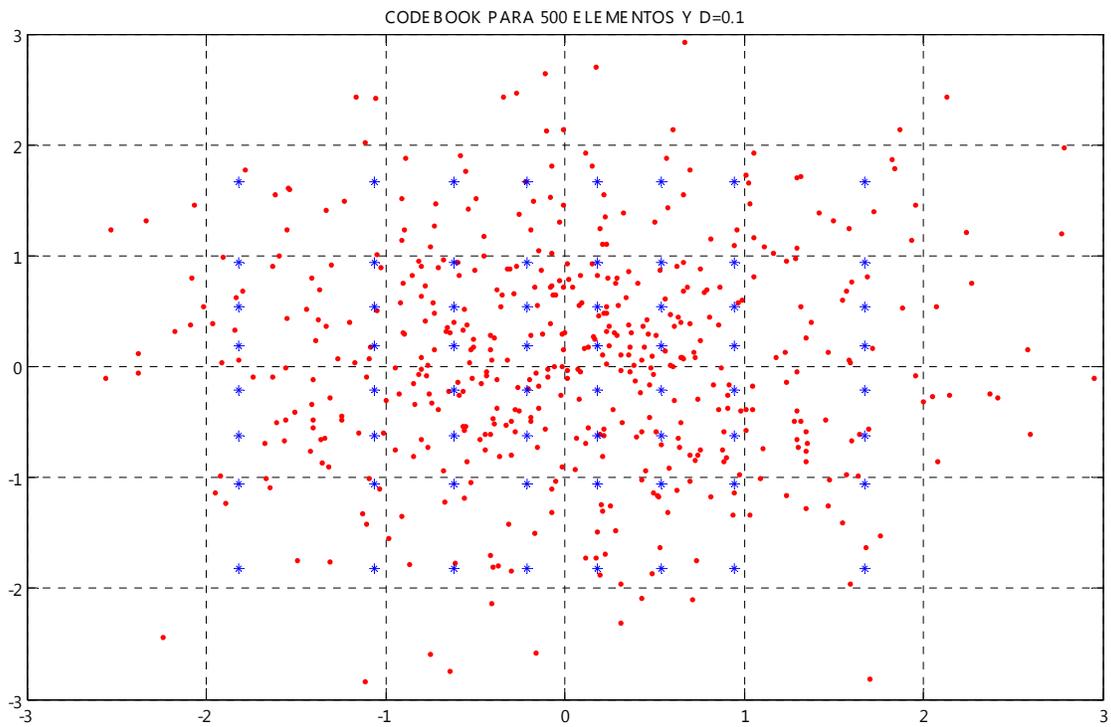


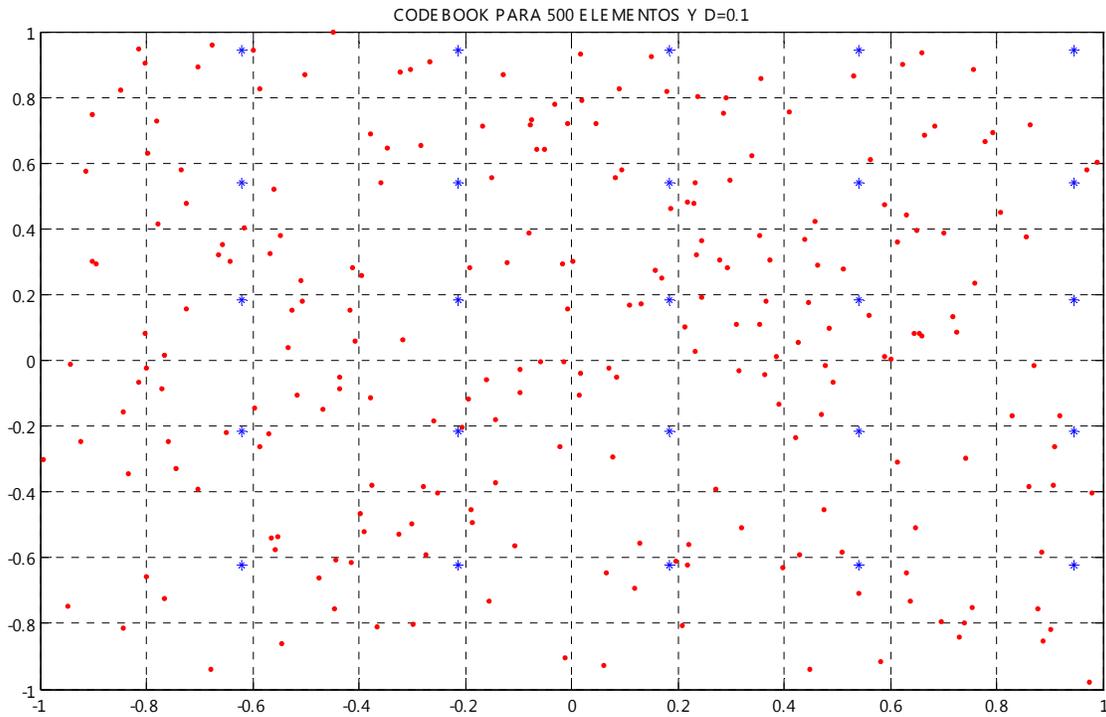
Parámetros de Diseño: Centroides generados a partir de 500 vectores aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.003.



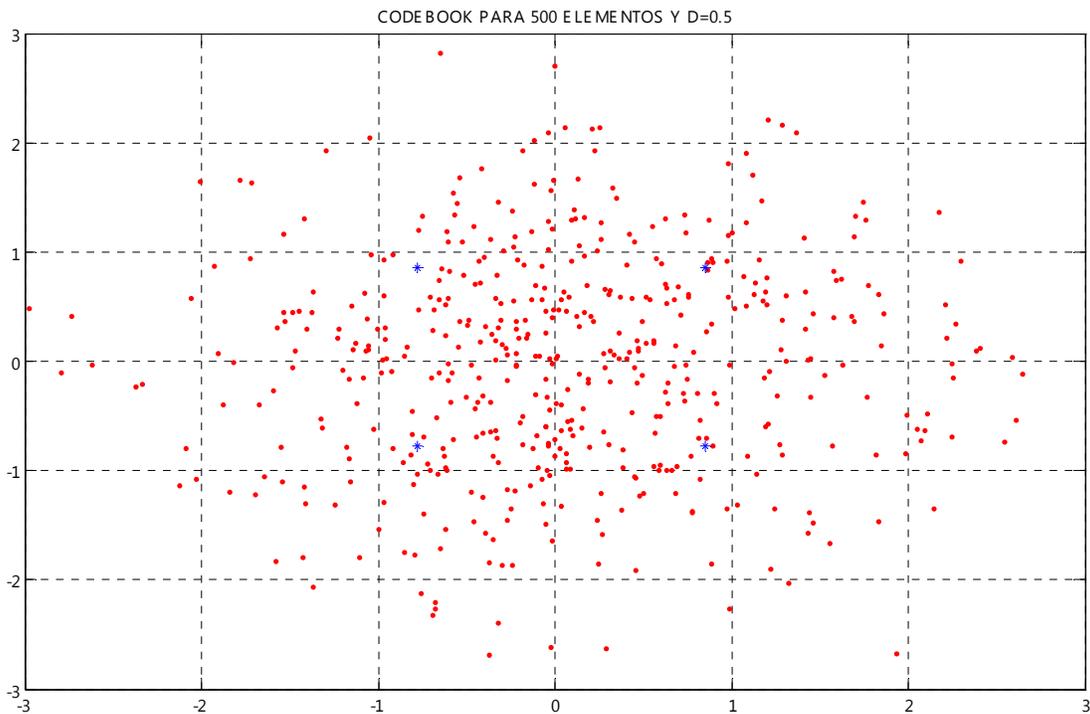


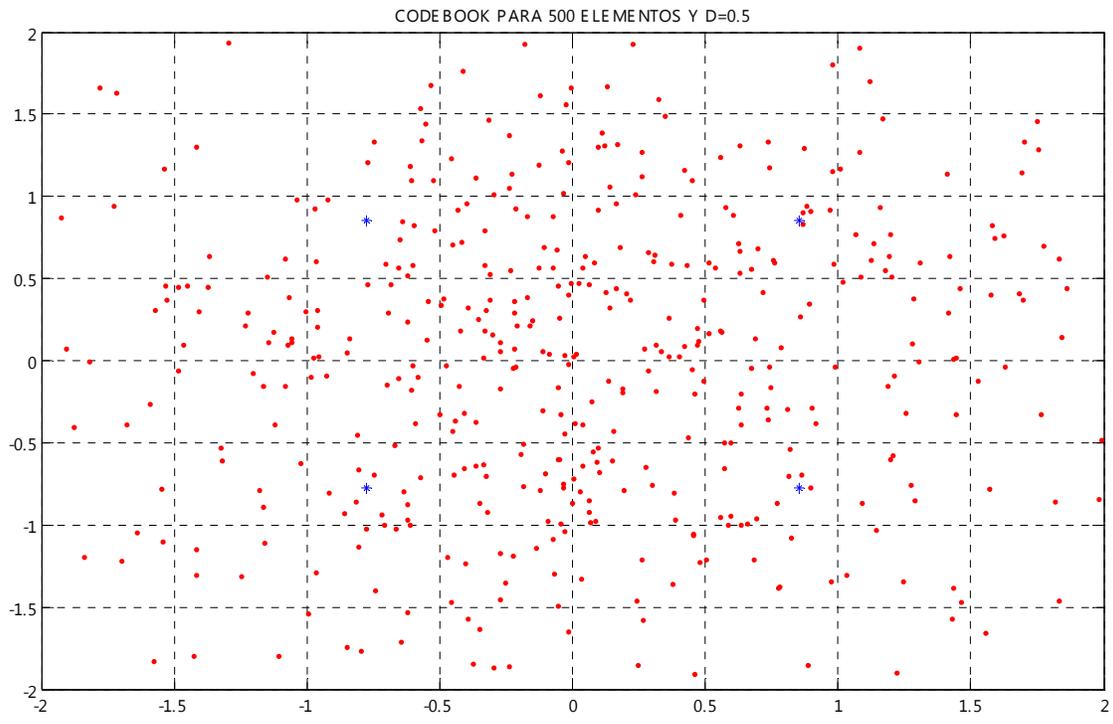
Parámetros de Diseño: Centroides generados a partir de 500 vectores aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.1.





Parámetros de Diseño: Centroides generados a partir de 500 vectores aleatorios con una distribución normal y tomando como parámetro fundamental una distorsión de 0.5.





BIBLIOGRAFÍA

LIBROS:

- [1] BRZEZNIAK, Zdzilaw; ZASTAWNIAK, Tomasz. *Basic Stochastic Processes, a course through exercises*. SPRINGER, Great Britain, 1999, pp. 222.
- [2] BUCKDAHN, Rainer; ENGELBERT, Hans-Jurgen. *Stochastic Processes and related topics*. Ed. Taylor and Francis. A series of books edited by Saul Jacka, Department of Statistics, University of Warwick, UK. Great Britain, 2002, pp. 276.
- [3] CHORNEI, Ruslan K.; DADUNA, Hans; KNOPOV, Pavel S. *Control of spatially structured random processes and random fields with applications*. SPRINGER, United States of America, 2006, pp. 261.
- [4] CINLAR, Erhan. *Introduction to stochastic processes*. Prentice-Hall, Inc. Englewood Cliffs, N.J. United States of America, 1975, pp. 402.
- [5] CLARKE, A. Bruce; DISNEY, Ralph L. *Probability and random processes for engineers and scientist*. John Wiley & Sons, Inc. United States of America, 1970, pp. 346.
- [6] COX, D.R.; MILLER H.D. *The theory stochastic processes*. Chapman & Hall. Londres, 1996, pp. 398.
- [7] EARL, Boyd. *Introduction to probability. A programmer unit in modern mathematics*. McGraw-Hill. United States of America, 1963, pp. 257.
- [8] FELDMAN, Richard M.; VALDEZ-FLORES, Ciriaco. *Applied Probability & Stochastic Processes*. PWS Publishing Company, 1996, pp. 308.
- [9] GARDNER, William A. *Introduction to Random Processes with applications to Signals and Systems*, 2nd edition. McGraw-Hill, Inc. United States of America, 1990, pp. 546.
- [10] GILAT, Amos. *MATLAB, an introduction with applications*. . John Wiley & Sons Inc., 3rd edition. United States of America, 2008, pp. 374.
- [11] INGLE, Vinay K.; PROAKIS, John G. *Digital signal processing using MATLAB*.

- Brooks/Cole Publishing Company, Canada, 2000, pp. 418.
- [12] KARLIN, Samuel. *A first course in stochastic processes*. Academic Press. New York, 1968, pp. 502.
- [13] LI, X. Rong. *Probability, random signals and statistics*. CRC PRESS. United States of America, 1999, pp. 455.
- [14] LUDEMAN, Lonnie C. *Random Processes. Filtering, estimation and detection*. John Wiley & Sons Inc. New Jersey, 2003, pp. 608.
- [15] MEYER, Paul L. *Introductory probability and statical applications*. Addison-Wesley, 1966, pp. 339.
- [16] PAPOULIS, Athanasios. *Probability, random variables and stochastic processes*. McGraw-Hill series in electrical engineering, 3rd edition. International edition 1999, pp. 666.
- [17] PAPOULIS, Athanasios. *Probability & Statistics*, Prentice Hall, Inc. A division of Simon & Shuster Englewood Cliffs. United States of America, 1990, New Jersey, pp. 454.
- [18] PARZEN, Emanuel. *Modern probability theory and its applications*. John Wiley & Sons Inc. United States of America, 1990, pp. 464.
- [19] PARZEN, Emanuel. *Teoría moderna de probabilidades y sus aplicaciones*. LIMUSA, 1993, pp. 502.
- [20] ROBINSON, Enders A. *Probability theory and applications*. International Human Resources Development Corporation. United States of America, 1985, pp. 420.
- [21] WOODROOFE, Michael. *Probability with Applications*. McGraw-Hill Kogakusha LTD. Edición Internacional de Estudiantes, 1975, pp. 372.
- [22] YATES, Roy D.; GOODMAN, David J. *Probability and Stochastic Processes. A friendly introduction for electrical and computer engineers*. John Wiley & Sons, Inc. United States of America, 1999, pp. 454.

TESIS:

- [23] ZAHIR AZAMI, Seyed Bahram. *Joint Source/Channel Coding Hierarchical Protection*. Ecole Nationale Supérieure des Telecommunications, Paris. Digital communications group, COMELEC department, 1999.

ARTÍCULOS:

- [24] LINDE Yoseph, BUZO Andrés, GRAY Robert M. *An algorithm for Vector Quantizer Design*. IEEE Transactions on communications, Vol. Com-28, No.1, January 1980.
- [25] MARÍN Roque. *Compresión de la información: Cuantización*. Dpto. de Ingeniería de la Información y las Comunicaciones, Facultad de Informática, Universidad de Murcia, Campus de Espinardo, 30071 – Espinardo (Murcia).
<http://perseo.dif.um.es/aike/roque/index.htm>
- [26] SÁNCHEZ CÉSPEDES Juan Manuel, Bernal Ruiz Giovanni A. *Compresión de la Señal Electrocardiográfica (E.G.C)*. Umbral Científico, junio, número 004. Fundación Universitaria Manuela Beltrán. Bogotá, Colombia.
umbralcientifico@umb.edu.co
- [27] *Codificación de imágenes y video: Cuantificación*. Instituto de Ingeniería Eléctrica (IIE), Facultad de Ingeniería de la Universidad de la República.
<http://iie.fing.edu.uy>
- [28] *Técnicas de Agrupación*. Grupo de Investigación en Planificación y Aprendizaje Automático (Planning and Learning research Group).
<http://www.plg.inf.uc3m.es>
- [29] *Guía de uso de MATLAB*. Universia y MIT: Ingeniería Electrónica y Ciencias Computacionales.
<http://mit.ocw.universia.net>
- [30] *MATLAB How to*. Grupo de inteligencia artificial, Depto. de arquitectura y Tecnología de computadores y ciencia de la computación e inteligencia artificial, Escuela Técnica Superior de Ingeniería Informática, Universidad Rey Juan Carlos.
<http://zenon.etsii.urjc.es/~jmbuena/VC/>