PUCE Virtual

CLASE

7

Teoría de la Información y la Comunicación

Códigos Correctores de Errores

QS WORLD UNIVERSITY RANKINGS

Educación de calidad

INTRODUCCIÓN DE LA CLASE

GRADO / POSGRADO / TECNOLOGÍAS

Estudia a tu tiempo
son interrupciones

EDUCACIÓN EN LÍNEA DE CALIDAD

En los sistemas modernos de comunicación digital, la integridad de la información representa uno de los pilares fundamentales para garantizar procesos seguros y confiables. En escenarios donde los datos viajan a través de redes inalámbricas, plataformas en la nube o infraestructuras críticas, la presencia de ruido, interferencias o ataques informáticos puede alterar el contenido transmitido. Desde la perspectiva de la ciberseguridad, la protección de la información no depende únicamente del cifrado, sino también de mecanismos capaces de detectar y corregir errores durante la transmisión.

Durante esta semana se abordarán los fundamentos de los códigos correctores de errores, herramientas esenciales para asegurar la disponibilidad y confiabilidad de los datos en ambientes digitales. El estudio de estos mecanismos permitirá comprender cómo las organizaciones reducen riesgos asociados a corrupción de datos, pérdida de paquetes y alteraciones ocasionadas por fallos físicos o amenazas externas. Asimismo, estos conceptos son ampliamente utilizados en redes de telecomunicaciones, sistemas bancarios, satélites, dispositivos IoT y protocolos de seguridad informática.

Los temas relacionados con la distancia de Hamming y el código Hamming constituyen bases indispensables para el diseño de sistemas capaces de mantener la integridad de la información aun cuando existan perturbaciones en el canal de comunicación. El estudiante analizará cómo la redundancia controlada permite identificar errores y restaurar mensajes afectados sin necesidad de retransmisiones constantes, optimizando así el rendimiento de los sistemas digitales. El desarrollo de esta temática contribuye directamente al Resultado de Aprendizaje 3 (RDA3), orientado a la implementación de códigos de detección y corrección de errores en sistemas de transmisión de datos. De igual manera, los contenidos permitirán avanzar significativamente en el RETO 3, debido a que proporcionan las bases conceptuales y prácticas necesarias para diseñar un sistema de comunicación capaz de equilibrar redundancia y capacidad de corrección frente a escenarios de ruido e interferencia.

7. Códigos Correctores de Errores

Los códigos correctores de errores constituyen técnicas matemáticas utilizadas para detectar y corregir alteraciones en la información transmitida o almacenada. Su principal propósito es incrementar la confiabilidad de los sistemas digitales mediante la incorporación de bits adicionales denominados bits de redundancia. Estos bits permiten verificar la integridad de los datos y, en ciertos casos, reconstruir la información original cuando ocurre un error.

De acuerdo con Alencar (2015), los sistemas de comunicación modernos requieren mecanismos que reduzcan el impacto del ruido sobre la señal transmitida. En ausencia de códigos correctores, cualquier alteración producida durante la transmisión podría generar pérdidas de información, fallos en aplicaciones críticas o vulnerabilidades de seguridad.

En el ámbito de la ciberseguridad, los códigos correctores poseen gran relevancia debido a que ayudan a mantener la integridad de la información. Un sistema que no pueda identificar errores durante la transmisión puede permitir que datos corruptos ingresen a bases de datos, servidores o aplicaciones críticas. Esto puede ocasionar fallos operativos, inconsistencias de información e incluso vulnerabilidades explotables por atacantes.
La detección y corrección de errores fortalece la integridad y confiabilidad de los sistemas de comunicación digitales utilizados en entornos de ciberseguridad.

Los códigos correctores se clasifican en dos grandes categorías:

- Códigos detectores de errores.
- Códigos correctores de errores.

Los primeros permiten identificar la existencia de un error, mientras que los segundos además posibilitan recuperar el mensaje original. Entre los códigos detectores más conocidos se encuentran la paridad simple y el CRC. En cambio, los códigos Hamming, Reed-Solomon y BCH pertenecen a la categoría de códigos correctores.

7.1 Fundamentos de Códigos Correctores de Errores

Los fundamentos de los códigos correctores se apoyan en el principio de redundancia controlada. Esto significa que el transmisor agrega información adicional al mensaje original para que el receptor pueda verificar si los datos llegaron correctamente.

Según Csiszár y Körner (2011), un sistema de codificación eficiente debe equilibrar dos aspectos fundamentales:

1. Capacidad de detección y corrección.
2. Cantidad de redundancia añadida.

Agregar demasiados bits de redundancia incrementa el consumo de ancho de banda y almacenamiento. Sin embargo, una redundancia insuficiente disminuye la capacidad de recuperación ante errores.

Elementos de un sistema corrector de errores

Fuente de información

Es el origen de los datos que serán transmitidos. Puede tratarse de texto, imágenes, audio o información generada por sensores.

Codificador
Añade bits de control o redundancia al mensaje original.

Canal de comunicación
Es el medio por el cual viajan los datos. Puede verse afectado por ruido, interferencias electromagnéticas o ataques.

Decodificador
Analiza los datos recibidos y verifica si existe algún error.

Receptor
Obtiene el mensaje corregido o validado.

Redundancia y decodificación son conceptos esenciales para comprender el funcionamiento de los sistemas correctores.

Tipos de errores

Error de un solo bit
Ocurre cuando únicamente un bit cambia de valor.
Ejemplo:
Mensaje original: 10110010
Mensaje recibido: 10100010

Error por ráfaga
Se produce cuando varios bits consecutivos son alterados. Este tipo de error es frecuente en redes inalámbricas y canales afectados por interferencia.

Relación con la ciberseguridad
En sistemas financieros o plataformas de autenticación, un error en la transmisión podría modificar credenciales, registros o paquetes críticos. Los códigos correctores ayudan a prevenir que información alterada sea procesada por aplicaciones sensibles.

Enlaces complementarios
Título: Introducción a los Códigos de Corrección de Errores
Descripción: Documento académico en español, de acceso libre y sin necesidad de registro, que explica los fundamentos de los códigos detectores y correctores de errores, redundancia, distancia de Hamming y aplicaciones en transmisión y almacenamiento de datos. Incluye ejemplos prácticos y ejercicios de codificación y decodificación.
Enlace: https://riunet.upv.es/server/api/core/bitstreams/4ba871c1-4777-4b59-a0d6-4484559f62a3/content

7.2 Distancia de Hamming

La distancia de Hamming es un concepto matemático utilizado para medir la diferencia entre dos cadenas binarias de igual longitud. Se define como el número de posiciones en las cuales los bits correspondientes son distintos.

Según Gorzalka y Deloumeaux (2012), este concepto resulta fundamental en teoría de la información debido a que permite determinar la capacidad de un código para detectar y corregir errores.

Definición matemática

La distancia de Hamming entre dos palabras binarias se calcula comparando bit a bit ambas secuencias.

Ejemplo:

Palabra A: 101101

Palabra B: 100001

Comparación:

- Bit 1: igual
- Bit 2: igual
- Bit 3: diferente
- Bit 4: diferente
- Bit 5: igual
- Bit 6: igual

Distancia de Hamming = 2

Figura 1

Fórmula de la distancia de Hamming

$$d_H(A,B) = \sum_{i=1}^{n} (a_i \neq b_i)$$

Nota. Representación matemática de la distancia de Hamming utilizada para medir diferencias entre secuencias binarias. Adaptado de Csiszár y Körner (2011).

La expresión corresponde a la fórmula matemática de la Distancia de Hamming, utilizada para medir cuántos bits son diferentes entre dos cadenas binarias de igual longitud.

La fórmula:

significa lo siguiente:

- $d_H(A,B)$: representa la distancia de Hamming entre las cadenas binarias A y B.
- $\sum$: indica una suma total.
- $a_i \neq b_i$: verifica si los bits en la posición $i$son diferentes.
- n: cantidad total de bits comparados.

En términos simples, la fórmula cuenta cuántas posiciones cambian entre dos secuencias binarias.

Ejemplo práctico

Cadena A: 101101

Cadena B: 100001

Comparación:

Posición	A	B	¿Diferente?
1	1	1	No
2	0	0	No
3	1	0	Sí
4	1	0	Sí
5	0	0	No
6	1	1	No

Existen 2 posiciones diferentes, por eso:
Distancia de Hamming = 2

Esto significa que ocurrieron dos cambios de bits entre ambas cadenas binarias. En teoría de la información y ciberseguridad, esta medida permite determinar la capacidad de un código para detectar y corregir errores durante la transmisión de datos.

La distancia mínima de un código determina cuántos errores pueden detectarse y corregirse.

Capacidad de detección y corrección

Un código con distancia mínima d puede:

- Detectar hasta d − 1 errores.
- Corregir hasta (d − 1)/2 errores.

Por ejemplo, si un código posee distancia mínima igual a 3, puede detectar dos errores y corregir un error.
La distancia mínima de Hamming permite evaluar la confiabilidad de un sistema de transmisión frente a errores generados por ruido.

Importancia en redes y ciberseguridad

La distancia de Hamming es utilizada en sistemas de autenticación, almacenamiento distribuido y protocolos de comunicación segura. En centros de datos, este mecanismo contribuye a verificar integridad de bloques de memoria y paquetes de red.
Además, los sistemas RAID y memorias ECC utilizan principios basados en distancia de Hamming para prevenir corrupción de datos.
Distancia mínima y bit de paridad representan elementos fundamentales para comprender la corrección de errores.

Aplicación práctica

Supóngase un sistema de autenticación biométrica que transmite patrones binarios entre dispositivos. Si el canal introduce alteraciones, la distancia de Hamming permite identificar diferencias y validar si el patrón recibido corresponde al original.

Relación con el RETO 3
El diseño de un sistema corrector requiere analizar la distancia mínima necesaria para garantizar integridad de información sin generar exceso de redundancia. Este análisis aporta directamente al desarrollo del RETO 3.

7.3 Código Hamming

El código Hamming fue desarrollado por Richard Hamming con el propósito de detectar y corregir errores de un solo bit en sistemas digitales. Este código constituye uno de los mecanismos más importantes en teoría de la información debido a su simplicidad y eficiencia.

El funcionamiento del código Hamming se basa en la incorporación estratégica de bits de paridad dentro del mensaje original. Estos bits permiten identificar la posición exacta donde ocurre un error.

Según Alencar (2015), los códigos Hamming representan una solución eficiente para sistemas donde la probabilidad de error es moderada y se requiere minimizar retransmisiones.

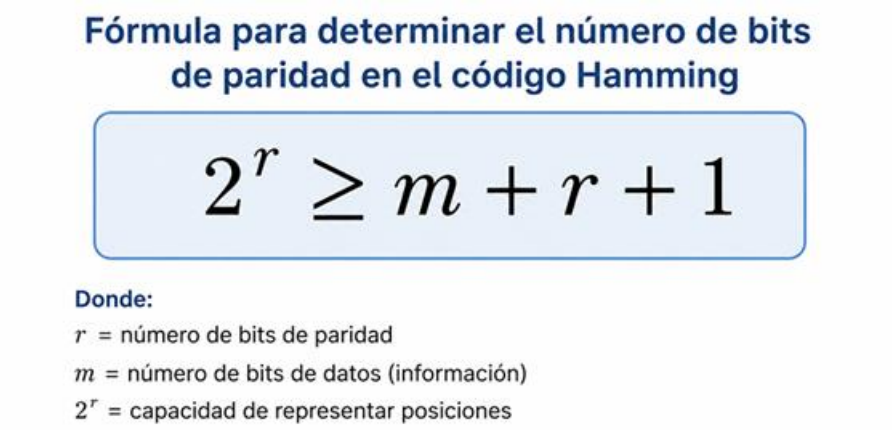
Estructura del código Hamming

El código se conoce como Hamming(posiciones,datos); por ejemplo, si se quiere trasmitir 4 bits de datos el código sería Hamming(7,4), la pregunta que nace es de donde sale el 7.

El 7 es el total de bits trasmitidos y esto esta en función de el numero de bits de paridad que hay que adicionar al dato trasmitido.

Para calcular el número de bits de paridad que se deben adicionar al mensaje trasmitido usamos la ecuación de la figura 1. Donde:

Figura 2
Fórmula para determinar el número de bits de paridad en el código Hamming.



Nota. Esta desigualdad garantiza que los bits de paridad sean suficientes para detectar y corregir errores de un solo bit. Elaboración propia basada en Csiszár y Körner (2011). Miguel Ortiz Navarrete

Para transmitir 4 bits de datos tendríamos

r	(2^r)	(m+r+1)	MAYOR
2	4	(4+2+1)=7	$7 \geq 8\ NO$
3	8	(4+3+1)=8	$8 \geq 8\ SI$

Entonces: para trasmitir 4 bits se necesita 3 bit de paridad esto sumado a los datos trasmitidos (4) nos da un total de 7 (Hamming(7,4))

Los bits de paridad se ubican en posiciones que corresponden a potencias de dos como se muestra en la Tabla 1.

Tabla 1
Ubicación de bits de paridad en código Hamming

Potencia	Posición
(2^0)	1
(2^1)	2
(2^2)	4
(2^3)	8

Nota. Elaboración propia con base en los principios de codificación Hamming descritos por Alencar (2015). Miguel Ortiz Navarrete

Los demás espacios contienen datos, al final la estructura queda como se muestra en la Tabla 2.

Tabla 2
Distribución de bits de datos y bits de paridad en un código Hamming de 8 bits

Posición	1	2	3	4	5	6	7	8	8	10	11	12
Tipo	P1	P2	D1	P4	D2	D3	D4	P8	D5	D6	D7	D8

Nota. Elaboración propia basada en la estructura de códigos Hamming descrita por Csiszár y Körner (2011) y Alencar (2015). Miguel Ortiz Navarrete

Cálculo de bits de paridad
Cada bit de paridad controla determinadas posiciones.

Paridad P1: Controla posiciones: 1, 3, 5, 7
Paridad P2: Controla posiciones: 2, 3, 6, 7
Paridad P4: Controla posiciones: 4, 5, 6, 7

¿Cómo sabe cada paridad qué posiciones revisar?
La regla es esta: Cada bit de paridad revisa posiciones cuyo número binario tenga un 1 en su posición correspondiente.

Las posiciones en binario se muestran en la Tabla 3

Tabla 3

Representación binaria de posiciones utilizadas en el código Hamming

Posición	Binario
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100

Nota. Elaboración propia basada en los principios de representación binaria y verificación de paridad empleados en códigos Hamming descritos por Gorzalka y Deloumeaux (2012). Miguel Ortiz Navarrete

¿Qué controla P1?

P1 corresponde al último bit binario: 0001

Entonces revisa todas las posiciones cuyo último bit sea 1 como se identifica en la Tabla 4.

Tabla 4

Posiciones controladas por el bit de paridad P1 en un código Hamming

Posición	Binario
1	0001
3	0011
5	0101
7	0111
9	1001
11	1011

Nota. Elaboración propia basada en el principio de verificación binaria utilizado por los códigos Hamming para el control de paridad (Alencar, 2015). Miguel Ortiz Navarrete

En términos generales si trasmitimos 8 bits como se define en la Tabla 2 necesitamos 4 bits de paridad y cada bit controlaría las posiciones como se establece en la Tabla 5

Tabla 5

Posiciones controladas por los bits de paridad en un código Hamming de 8 bits

Paridad	Posiciones que controla
P1	1, 3, 5, 7, 9, 11
P2	2, 3, 6, 7, 10, 11
P4	4, 5, 6, 7, 12
P8	8, 9, 10, 11, 12

Nota. Elaboración propia basada en la distribución de bits de control y verificación binaria utilizada en códigos Hamming para detección y corrección de errores (Csiszár & Körner, 2011; Alencar, 2015).

Enlace Complementario

Título: Simulación interactiva del Código Hamming para detección y corrección de errores

Descripción: Recurso educativo interactivo que permite visualizar cómo se generan los bits de paridad, cómo se codifican los datos y cómo el receptor detecta y corrige errores utilizando el código Hamming. El recurso facilita comprender la relación entre posiciones binarias, paridades y localización del error en sistemas digitales.

Enlace: https://www.geeksforgeeks.org/computer-networks/hamming-code-in-computer-network/

Ejemplo de estructura si queremos transmitir este mensaje: Mensaje original: 1011

Posición	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Tipo	P1	P2	D1	P4	D2	D3	D4
Valor	P1	P2	1	P4	0	1	1

En función de las posiciones que controla cada bit de paridad tendríamos

P1

Posición	**1**	**3**	**5**	**7**
Tipo	P1	D1	D2	D4
Valor	P1	1	0	1

Cantidad de unos (sin contar P1): 2
Como 2 es par en P1=0

P2

Posición	**2**	**3**	**6**	**7**
Tipo	P2	D1	D3	D4
Valor	P2	1	1	1

Cantidad de unos (sin contar P2): 3
Como 3 es par en P2=1

P4

Posición	**4**	**5**	**6**	**7**
Tipo	P4	D2	D3	D4
Valor	P4	0	1	1

Cantidad de unos (sin contar P4): 2
Como 2 es par en P4=0

Código Hamming final

Reemplazamos los valores:

Posición	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Tipo	P1	P2	D1	P4	D2	D3	D4
Valor	0	1	1	0	0	1	1

Después del cálculo de paridades nos queda el código completo: 0 1 1 0 0 1 1
El código completo es transmitido al receptor.

El código Hamming permite identificar la posición exacta de un error de un solo bit mediante el análisis de paridades.

7.4 Decodificación y Corrección con Código Hamming

La etapa de decodificación tiene como propósito identificar si existe un error dentro de la secuencia recibida y, posteriormente, corregirlo.
El receptor recalcula los bits de paridad y compara los resultados con los bits transmitidos. Si las paridades coinciden, el mensaje se considera correcto. En caso contrario, se genera un síndrome binario que indica la posición del error.

Síndrome de error

El síndrome corresponde a un número binario formado por los resultados de las verificaciones de paridad.

¿Cómo ayuda esto a corregir errores?

Cuando el receptor recibe el código, vuelve a calcular las paridades. Si una o más paridades fallan, sus posiciones se combinan y forman un número binario que indica exactamente dónde está el error.

Por ejemplo: Si recibió este código con error: 0110111

Al calcular las paridades encuentra que: P1 falla, P2 funciona, P4 falla

Como cada paridad representa una potencia en 2 se tiene

Paridad	Valor decimal que representa
P1	1
P2	2
P4	4

Entonces, si fallan P1 y P4: P1 = 1, P2 = 0, P4 = 1
Síndrome de error: 101

El decimal del binario correspondiente al síndrome de error es: 1 + 0 + 4 = 5
La posición 5 es la que falla

¿Qué hace el receptor después?

Si recibió este código con error: 0110111, y detecta que el error está en la posición 5, cambia ese bit.

Posición	1	2	3	4	5	6	7
Recibido	0	1	1	0	1	1	1
Corrección	0	1	1	0	0	1	1

Luego elimina los bits de paridad y se queda solo con los datos:

Posición	3	5	6	7
Datos	1	0	1	1

Mensaje original recuperado: 1011

Aplicaciones reales

Redes inalámbricas: Permiten reducir retransmisiones ocasionadas por interferencia.
Comunicaciones satelitales: Garantizan confiabilidad en enlaces de larga distancia.
Sistemas bancarios: Protegen transacciones y registros electrónicos.
Centros de datos: Previenen corrupción de información almacenada.
La decodificación Hamming incrementa la confiabilidad de los sistemas digitales al corregir errores antes de que afecten aplicaciones críticas.

Relación con el RDA3 y el RETO 3

La implementación de procesos de codificación y decodificación constituye parte esencial del desarrollo práctico solicitado en el RDA3. Estos conocimientos permiten diseñar un sistema capaz de detectar y corregir errores considerando la relación entre redundancia y eficiencia.
Desde la perspectiva del RETO 3, el estudiante ya dispone de herramientas para:

- Diseñar códigos correctores básicos.
- Evaluar integridad de información.
- Analizar eficiencia frente al ruido.
- Implementar mecanismos de corrección.

Con ello se alcanza aproximadamente el 50% del desarrollo conceptual y práctico requerido en el reto.
Síndrome de error y memoria ECC son conceptos relevantes asociados a la decodificación Hamming.

Alencar, M. S. (2015). Information theory. Momentum Press.

Csiszár, I., & Körner, J. (2011). Information theory: Coding theorems for discrete memoryless **memoryless systems (2nd ed.).** Cambridge University Press.

Gorzalka, J. D., & Deloumeaux, P. (2012). Information theory: New research. Nova Science Publishers.

PUCE Virtual

síguenos en:

www.pucevirtual.puce.edu.ec