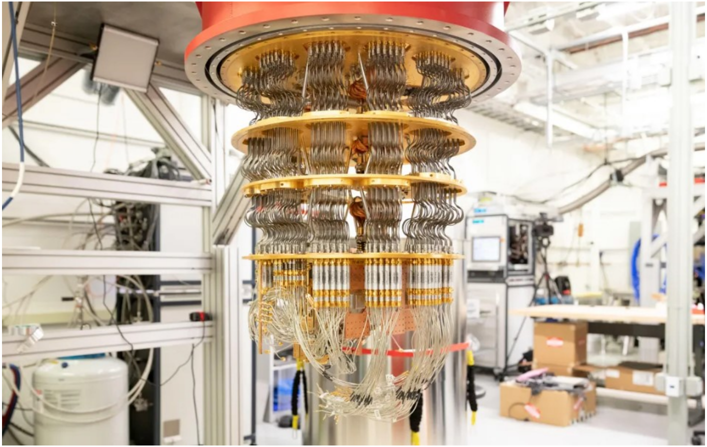
The McEliece Cryptosystem and Error-Correcting Codes

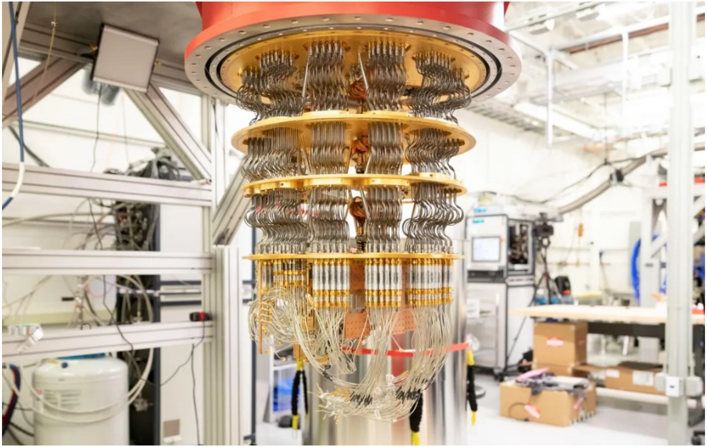
Mario Nonog Jr.
Mathematics Department,
US Naval Academy, Annapolis, Maryland

Background



- 1994: RSA was broken using Shor's Algorithm on a Quantum Computer.

Background



- 1994: RSA was broken using Shor's Algorithm on a Quantum Computer. But, don't worry $15 = 3 \times 5$.

Background

- 2016: However, it took 22 years for the National Institute of Standards and Technology (NIST) to begin the search for a new Post-Quantum Cryptography standard, in order to protect our secrets against adversaries.

Background

- 2016: However, it took 22 years for the National Institute of Standards and Technology (NIST) to begin the search for a new Post-Quantum Cryptography standard, in order to protect our secrets against adversaries.
- 2016: NIST has evaluated over 50 cryptosystems.

Background

- 2016: However, it took 22 years for the National Institute of Standards and Technology (NIST) to begin the search for a new Post-Quantum Cryptography standard, in order to protect our secrets against adversaries.
- 2016: NIST has evaluated over 50 cryptosystems.
- July 2020: only four are left standing. One of these is the McEliece Cryptosystem, based on error-correcting codes.

Background

- 2016: However, it took 22 years for the National Institute of Standards and Technology (NIST) to begin the search for a new Post-Quantum Cryptography standard, in order to protect our secrets against adversaries.
- 2016: NIST has evaluated over 50 cryptosystems.
- July 2020: only four are left standing. One of these is the McEliece Cryptosystem, based on error-correcting codes.

Goal:

The goal of this project is to understand the McEliece Cryptosystem, which utilizes Goppa Codes. Goppa Codes are a subclass of Linear Codes, and are related to Cyclic Codes and BCH Codes, which what we will focus on today.

Background on Error-Correcting Codes

The McEliece Cryptosystem based on Error-Correcting Codes

Alice	Public	Bob
	Eve	

The McEliece Cryptosystem based on Error-Correcting Codes

Alice	Public	Bob
		G
	Eve	

Bob: Chooses $k \times n$ generating matrix G,

The McEliece Cryptosystem based on Error-Correcting Codes

Alice	Public	Bob
		GP
	Eve	

Bob: Chooses $k \times n$ generating matrix G, and $n \times n$ permutation matrix P,

Alice	Public	Bob
		SGP
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P,

The McEliece Cryptosystem based on Error-Correcting Codes

Alice	Public	Bob
		$G_1 = SGP$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P,

The McEliece Cryptosystem based on Error-Correcting Codes

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

The McEliece Cryptosystem based on Error-Correcting Codes

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
G_1		
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
xG_1		
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.
Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$,

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e$		
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

The McEliece Cryptosystem based on Error-Correcting Codes

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$		
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.
Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow$	
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y$	
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) y
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) yP^{-1}
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1}$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.
Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.
Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= (SGP)$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP)$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP) \quad + e$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

The McEliece Cryptosystem based on Error-Correcting Codes

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP)P^{-1} + eP^{-1}$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP)P^{-1} + eP^{-1}$ $= \qquad + eP^{-1}$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

The McEliece Cryptosystem based on Error-Correcting Codes

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP)P^{-1} + eP^{-1}$ $= (xS) \quad + eP^{-1}$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP)P^{-1} + eP^{-1}$ $= (xS)G + eP^{-1}$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP)P^{-1} + eP^{-1}$ $= (xS)G + eP^{-1}$
		2) Compute syndrome $r_1 H^T$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP)P^{-1} + eP^{-1}$ $= (xS)G + eP^{-1}$ 2) Compute syndrome $r_1 H^T$ 3) Lookup the codeword
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

The McEliece Cryptosystem based on Error-Correcting Codes

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP)P^{-1} + eP^{-1}$ $= (xS)G + eP^{-1}$
		2) Compute syndrome $r_1 H^T$
		3) Lookup the codeword $c_1 = (xS)G$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP)P^{-1} + eP^{-1}$ $= (xS)G + eP^{-1}$
		2) Compute syndrome $r_1 H^T$
		3) Lookup the codeword $c_1 = (xS)G = x_1$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP)P^{-1} + eP^{-1}$ $= (xS)G + eP^{-1}$
		2) Compute syndrome $r_1 H^T$
		3) Lookup the codeword $c_1 = (xS)G = x_1 G$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.
Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

The McEliece Cryptosystem based on Error-Correcting Codes

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP)P^{-1} + eP^{-1}$ $= (xS)G + eP^{-1}$ 2) Compute syndrome $r_1 H^T$ 3) Lookup the codeword $c_1 = (xS)G = x_1 G$ associated with received word r_1
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP)P^{-1} + eP^{-1}$ $= (xS)G + eP^{-1}$
		2) Compute syndrome $r_1 H^T$
		3) Lookup the codeword $c_1 = (xS)G = x_1 G$ associated with received word r_1
		4) Extract message $x_1 = (xS)$
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP)P^{-1} + eP^{-1}$ $= (xS)G + eP^{-1}$ 2) Compute syndrome $r_1 H^T$ 3) Lookup the codeword $c_1 = (xS)G = x_1 G$ associated with received word r_1 4) Extract message $x_1 = (xS)$ associated with code word c_1 (first k bits)
	Eve	

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.
Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

The McEliece Cryptosystem based on Error-Correcting Codes

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1} = x(SGP)P^{-1} + eP^{-1} = (xS)G + eP^{-1}$
		2) Compute syndrome $r_1 H^T$
		3) Lookup the codeword $c_1 = (xS)G = x_1 G$ associated with received word r_1
		4) Extract message $x_1 = (xS)$ associated with code word c_1 (first k bits)
	Eve	5) Compute $x_1 S^{-1}$

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.
Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP)P^{-1} + eP^{-1}$ $= (xS)G + eP^{-1}$
		2) Compute syndrome $r_1 H^T$
		3) Lookup the codeword $c_1 = (xS)G = x_1 G$ associated with received word r_1
		4) Extract message $x_1 = (xS)$ associated with code word c_1 (first k bits)
	Eve	5) Compute $x_1 S^{-1} = (xS)S^{-1}$

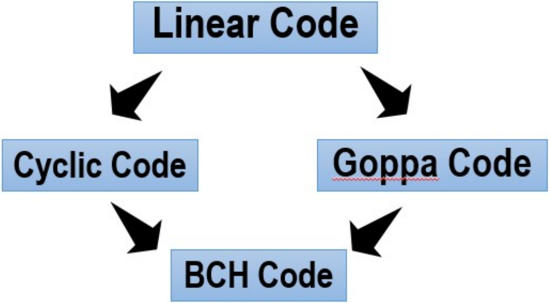
Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n

Alice	Public	Bob
	$G_1 \leftarrow$	$G_1 = SGP$
$xG_1 + e = y$	$\rightarrow y \rightarrow$	1) $r_1 = yP^{-1} = (xG_1 + e)P^{-1}$ $= x(SGP)P^{-1} + eP^{-1}$ $= (xS)G + eP^{-1}$
		2) Compute syndrome $r_1 H^T$
		3) Lookup the codeword $c_1 = (xS)G = x_1 G$ associated with received word r_1
		4) Extract message $x_1 = (xS)$ associated with code word c_1 (first k bits)
	Eve	5) Compute $x_1 S^{-1} = (xS)S^{-1} = x$

Bob: Chooses $k \times n$ generating matrix G, $k \times k$ invertible matrix S, and $n \times n$ permutation matrix P, calculates public key $G_1 = SGP$.

Alice: Chooses $1 \times k$ message $x \in \mathbb{Z}_2^k$, and error vector e of weight t, length n



The McEliece Cryptosystem

The McEliece Cryptosystem uses a Goppa code of length 1024 that can correct 50 errors. In this case, Eve has $\binom{1024}{50} \approx 3x10^{85}$ possible locations of the errors.

Linear Code

- **Definition:** An $[n, k, d]$ linear code is a vector space with dimension k and length n over a field $\mathbb{F}$, where the combination of any two codewords is always a codeword, with minimum distance d between two codewords.

Linear Code

- **Definition:** An $[n, k, d]$ linear code is a vector space with dimension k and length n over a field $\mathbb{F}$, where the combination of any two codewords is always a codeword, with minimum distance d between two codewords.
- **Parity Check Matrix (PCM Hamming Code):** Given generating matrix $G = [I_k, P] \in C$, then $H = [-P^T, I_{n-k}]$ is a parity check matrix for C if and only if $rH^T = 0$

Linear Code

- **Definition:** An $[n, k, d]$ linear code is a vector space with dimension k and length n over a field $\mathbb{F}$, where the combination of any two codewords is always a codeword, with minimum distance d between two codewords.
- **Parity Check Matrix (PCM Hamming Code):** Given generating matrix $G = [I_k, P] \in C$, then $H = [-P^T, I_{n-k}]$ is a parity check matrix for C if and only if $rH^T = 0$

Linear Code

- **Definition:** An $[n, k, d]$ linear code is a vector space with dimension k and length n over a field $\mathbb{F}$, where the combination of any two codewords is always a codeword, with minimum distance d between two codewords.
- **Parity Check Matrix (PCM Hamming Code):** Given generating matrix $G = [I_k, P] \in C$, then $H = [-P^T, I_{n-k}]$ is a parity check matrix for C if and only if $rH^T = 0$
- **Coset:** Given linear code C, and n-dimensional vector r, a coset is the set of $r + C$. The vector with minimum Hamming weight is the

Linear Code

- **Definition:** An $[n, k, d]$ linear code is a vector space with dimension k and length n over a field $\mathbb{F}$, where the combination of any two codewords is always a codeword, with minimum distance d between two codewords.
- **Parity Check Matrix (PCM Hamming Code):** Given generating matrix $G = [I_k, P] \in C$, then $H = [-P^T, I_{n-k}]$ is a parity check matrix for C if and only if $rH^T = 0$
- **Coset:** Given linear code C, and n-dimensional vector r, a coset is the set of $r + C$. The vector with minimum Hamming weight is the **coset leader**.

Linear Code

- **Definition:** An $[n, k, d]$ linear code is a vector space with dimension k and length n over a field $\mathbb{F}$, where the combination of any two codewords is always a codeword, with minimum distance d between two codewords.
- **Parity Check Matrix (PCM Hamming Code):** Given generating matrix $G = [I_k, P] \in C$, then $H = [-P^T, I_{n-k}]$ is a parity check matrix for C if and only if $rH^T = 0$
- **Coset:** Given linear code C, and n-dimensional vector r, a coset is the set of $r + C$. The vector with minimum Hamming weight is the **coset leader**.
- Given a linear code C, s errors detected if minimum distance $d(C) \geq s + 1$

Linear Code

- **Definition:** An $[n, k, d]$ linear code is a vector space with dimension k and length n over a field $\mathbb{F}$, where the combination of any two codewords is always a codeword, with minimum distance d between two codewords.
- **Parity Check Matrix (PCM Hamming Code):** Given generating matrix $G = [I_k, P] \in C$, then $H = [-P^T, I_{n-k}]$ is a parity check matrix for C if and only if $rH^T = 0$
- **Coset:** Given linear code C, and n-dimensional vector r, a coset is the set of $r + C$. The vector with minimum Hamming weight is the **coset leader**.
- Given a linear code C, s errors detected if minimum distance $d(C) \geq s + 1$
- Given a linear code C, t errors corrected if $d(C) \geq 2t + 1$

Linear Code

- **Definition:** An $[n, k, d]$ linear code is a vector space with dimension k and length n over a field $\mathbb{F}$, where the combination of any two codewords is always a codeword, with minimum distance d between two codewords.
- **Parity Check Matrix (PCM Hamming Code):** Given generating matrix $G = [I_k, P] \in C$, then $H = [-P^T, I_{n-k}]$ is a parity check matrix for C if and only if $rH^T = 0$
- **Coset:** Given linear code C, and n-dimensional vector r, a coset is the set of $r + C$. The vector with minimum Hamming weight is the **coset leader**.
- Given a linear code C, s errors detected if minimum distance $d(C) \geq s + 1$
- Given a linear code C, t errors corrected if $d(C) \geq 2t + 1$
- **Syndrome:** This is defined as $S(r) = rH^T$

Linear Hamming Code

- **How to DECODE a received word r?**

Linear Hamming Code

- **How to DECODE a received word r?**
  1. Calculate the syndrome, $S(r) = rH^T$.

Linear Hamming Code

- **How to DECODE a received word r?**
  1. Calculate the syndrome, $S(r) = rH^T$.
  2. Find which coset the syndrome belongs.

Linear Hamming Code

- **How to DECODE a received word r?**
  1. Calculate the syndrome, $S(r) = rH^T$.
  2. Find which coset the syndrome belongs.
  3. Look for the coset leader.

Linear Hamming Code

- **How to DECODE a received word r?**
  1. Calculate the syndrome, $S(r) = rH^T$.
  2. Find which coset the syndrome belongs.
  3. Look for the coset leader.
  4. message $= r -$ coset leader.

Linear Hamming Code

Example: Consider a $[4, 2, 2]$ linear code with G .

Linear Hamming Code

Example: Consider a $[4, 2, 2]$ linear code with G .

generating matrix $G = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}$,

Here is the lookup table for decoding received words.

$$\begin{array}{l} (0,0,0,0)(1,0,0,1)(0,1,0,1)(1,1,0,0) \\ (1,0,0,0)(0,0,0,1)(1,1,0,1)(0,1,0,0) \\ (0,0,1,0)(1,0,1,1)(0,1,1,1)(1,1,1,0) \\ (0,0,1,1)(1,0,1,0)(0,1,1,0)(1,1,1,1) \end{array}$$

Linear Hamming Code

Example: Consider a $[4, 2, 2]$ linear code with G, H.

generating matrix $G = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}$,

Here is the lookup table for decoding received words.

$$
\begin{array}{c}
(0,0,0,0)(1,0,0,1)(0,1,0,1)(1,1,0,0) \\
(1,0,0,0)(0,0,0,1)(1,1,0,1)(0,1,0,0) \\
(0,0,1,0)(1,0,1,1)(0,1,1,1)(1,1,1,0) \\
(0,0,1,1)(1,0,1,0)(0,1,1,0)(1,1,1,1)
\end{array}
$$

Linear Hamming Code

Example: Consider a $[4, 2, 2]$ linear code with G, H.

generating matrix $G = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}$,

parity check matrix $H = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}$

Here is the lookup table for decoding received words.

$$\begin{array}{l}
(0,0,0,0)(1,0,0,1)(0,1,0,1)(1,1,0,0) \\
(1,0,0,0)(0,0,0,1)(1,1,0,1)(0,1,0,0) \\
(0,0,1,0)(1,0,1,1)(0,1,1,1)(1,1,1,0) \\
(0,0,1,1)(1,0,1,0)(0,1,1,0)(1,1,1,1)
\end{array}$$

Linear Hamming Code

Alice encodes message $x = [1, 1]$ by computing $xG = [1, 1, 0, 0]$. She sends to Bob through a noisy channel, and Bob receives $r = (1, 1, 1, 0)$.
Next, Bob must **DECODE**, by calculating the syndrome $S(r) = rH^T$.

Coset Leader	Syndrome
$(0, 0, 0, 0)$	$(0, 0)$
$(1, 0, 0, 0)$	$(0, 1)$
$(0, 0, 1, 0)$	$(1, 0)$
$(0, 0, 1, 1)$	$(1, 1)$

$$S(r) = (1, 1, 1, 0) \begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} = (1, 0).$$

Thus, code word $=$

Linear Hamming Code

Alice encodes message $x = [1, 1]$ by computing $xG = [1, 1, 0, 0]$. She sends to Bob through a noisy channel, and Bob receives $r = (1, 1, 1, 0)$.
Next, Bob must **DECODE**, by calculating the syndrome $S(r) = rH^T$.

Coset Leader	Syndrome
$(0, 0, 0, 0)$	$(0, 0)$
$(1, 0, 0, 0)$	$(0, 1)$
$(0, 0, 1, 0)$	$(1, 0)$
$(0, 0, 1, 1)$	$(1, 1)$

$$S(r) = (1, 1, 1, 0) \begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} = (1, 0).$$

Thus, code word $= \underbrace{(1, 1, 1, 0)}_{\textbf{received word}}$

Linear Hamming Code

Alice encodes message $x = [1, 1]$ by computing $xG = [1, 1, 0, 0]$. She sends to Bob through a noisy channel, and Bob receives $r = (1, 1, 1, 0)$.

Next, Bob must **DECODE**, by calculating the syndrome $S(r) = rH^T$.

Coset Leader	Syndrome
$(0, 0, 0, 0)$	$(0, 0)$
$(1, 0, 0, 0)$	$(0, 1)$
$(0, 0, 1, 0)$	$(1, 0)$
$(0, 0, 1, 1)$	$(1, 1)$

$$S(r) = (1, 1, 1, 0) \begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} = (1, 0).$$

Thus, code word $= \underbrace{(1, 1, 1, 0)}_{\textbf{received word}} - \underbrace{(0, 0, 1, 0)}_{\textbf{coset leader}}$

Linear Hamming Code

Alice encodes message $x = [1,1]$ by computing $xG = [1,1,0,0]$. She sends to Bob through a noisy channel, and Bob receives $r = (1,1,1,0)$.
Next, Bob must **DECODE**, by calculating the syndrome $S(r) = rH^T$.

Coset Leader	Syndrome
$(0,0,0,0)$	$(0,0)$
$(1,0,0,0)$	$(0,1)$
$(0,0,1,0)$	$(1,0)$
$(0,0,1,1)$	$(1,1)$

$$S(r) = (1,1,1,0)\begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} = (1,0).$$

$$\textbf{Thus, code word} = \underbrace{(1,1,1,0)}_{\textbf{received word}} - \underbrace{(0,0,1,0)}_{\textbf{coset leader}} = \underbrace{(\underbrace{1,1}_{}, 0, 0)}_{}$$

Linear Hamming Code

Alice encodes message $x = [1, 1]$ by computing $xG = [1, 1, 0, 0]$. She sends to Bob through a noisy channel, and Bob receives $r = (1, 1, 1, 0)$.
Next, Bob must **DECODE**, by calculating the syndrome $S(r) = rH^T$.

Coset Leader	Syndrome
$(0, 0, 0, 0)$	$(0, 0)$
$(1, 0, 0, 0)$	$(0, 1)$
$(0, 0, 1, 0)$	$(1, 0)$
$(0, 0, 1, 1)$	$(1, 1)$

$$S(r) = (1, 1, 1, 0) \begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} = (1, 0).$$

$$\textbf{Thus, code word} = \underbrace{(1, 1, 1, 0)}_{\textbf{received word}} - \underbrace{(0, 0, 1, 0)}_{\textbf{coset leader}} = \underbrace{(\underbrace{1, 1}, 0, 0)}_{\textbf{codeword}}$$

Linear Hamming Code

Alice encodes message $x = [1, 1]$ by computing $xG = [1, 1, 0, 0]$. She sends to Bob through a noisy channel, and Bob receives $r = (1, 1, 1, 0)$.

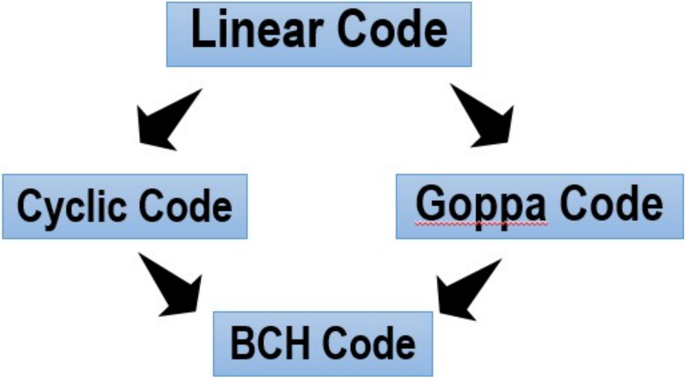
Next, Bob must **DECODE**, by calculating the syndrome $S(r) = rH^T$.

Coset Leader	Syndrome
$(0,0,0,0)$	$(0,0)$
$(1,0,0,0)$	$(0,1)$
$(0,0,1,0)$	$(1,0)$
$(0,0,1,1)$	$(1,1)$

$$S(r) = (1,1,1,0)\begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} = (1,0).$$

Thus, code word $= \underbrace{(1,1,1,0)}_{\textbf{received word}} - \underbrace{(0,0,1,0)}_{\textbf{coset leader}} = \underbrace{(\underbrace{1,1}_{\textbf{message}},0,0)}_{\textbf{codeword}}$

Cyclic Code



Cyclic Code

A code is said to be a **cyclic code** if it contains the property

$$(c_1, c_2, ..., c_{n-1}, c_n) \in C \Longleftrightarrow (c_n, c_1, c_2, \ldots, c_{n-1}) \in C$$

Given $g(X) = g_0 + g_1 X + \cdots + g_{n-1} X^{n-1} + g_{n-k} X^{n-k}$, and $h(X) = h_0 + h_1 X + \cdots + h_{k-1} X + h_k X^k$ (where $g(X)h(X) = X^n - 1$), we formulate $k \times n$ generating matrix G

Cyclic Code

A code is said to be a **cyclic code** if it contains the property

$$(c_1, c_2, ..., c_{n-1}, c_n) \in C \iff (c_n, c_1, c_2, \dots, c_{n-1}) \in C$$

Given $g(X) = g_0 + g_1 X + \cdots + g_{n-1} X^{n-1} + g_{n-k} X^{n-k}$, and $h(X) = h_0 + h_1 X + \cdots + h_{k-1} X + h_k X^k$ (where $g(X)h(X) = X^n - 1$), we formulate $k \times n$ generating matrix G

$$\underbrace{\begin{bmatrix} g_0 & g_1 & \cdots & g_{n-k} & 0 & \cdots & 0 \\ 0 & g_0 & g_1 & \cdots & g_{n-k} & 0 & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & g_0 & g_1 & \cdots & g_{n-k} \end{bmatrix}}_{G}$$

Cyclic Code

A code is said to be a **cyclic code** if it contains the property

$$(c_1, c_2, ..., c_{n-1}, c_n) \in C \iff (c_n, c_1, c_2, \ldots, c_{n-1}) \in C$$

Given $g(X) = g_0 + g_1 X + \cdots + g_{n-1} X^{n-1} + g_{n-k} X^{n-k}$, and $h(X) = h_0 + h_1 X + \cdots + h_{k-1} X + h_k X^k$ (where $g(X)h(X) = X^n - 1$), we formulate $k \times n$ generating matrix G and $(n-k) \times n$ parity check matrix H.

$$\underbrace{\begin{bmatrix} g_0 & g_1 & \cdots & g_{n-k} & 0 & \cdots & 0 \\ 0 & g_0 & g_1 & \cdots & g_{n-k} & 0 & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & g_0 & g_1 & \cdots & g_{n-k} \end{bmatrix}}_{G}$$

Cyclic Code

A code is said to be a **cyclic code** if it contains the property

$$(c_1, c_2, ..., c_{n-1}, c_n) \in C \iff (c_n, c_1, c_2, \ldots, c_{n-1}) \in C$$

Given $g(X) = g_0 + g_1 X + \cdots + g_{n-1} X^{n-1} + g_{n-k} X^{n-k}$, and $h(X) = h_0 + h_1 X + \cdots + h_{k-1} X + h_k X^k$ (where $g(X)h(X) = X^n - 1$), we formulate $k \times n$ generating matrix G and $(n-k) \times n$ parity check matrix H.

$$\underbrace{\begin{bmatrix} g_0 & g_1 & \cdots & g_{n-k} & 0 & \cdots & 0 \\ 0 & g_0 & g_1 & \cdots & g_{n-k} & 0 & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & g_0 & g_1 & \cdots & g_{n-k} \end{bmatrix}}_{G} \underbrace{\begin{bmatrix} h_k & h_{k-1} & \cdots & h_0 & 0 & \cdots & 0 \\ 0 & h_k & h_{k-1} & \cdots & h_0 & 0 & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h_k & h_{k-1} & \cdots & h_0 \end{bmatrix}}_{H}$$

Cyclic Code

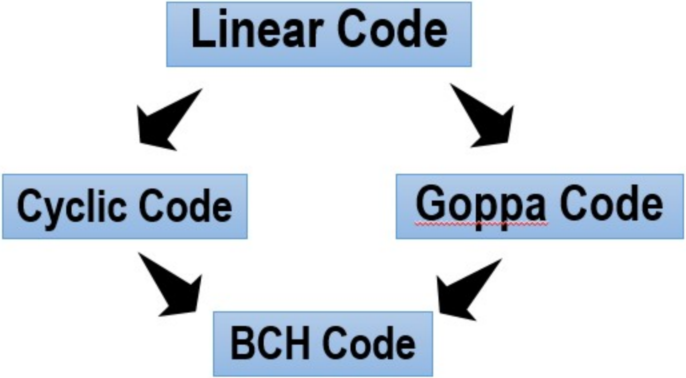
Example: Here is an $[n, k, d] = [7, 3, 4]$ cyclic code C.

$$X^7 - 1 = g(X)h(X) = \underbrace{(X^4 + X^2 + X + 1)}_{g(X)}\underbrace{(X^3 + X + 1)}_{h(X)}.$$

Then 3×7 generating matrix G and 4×7 parity check matrix H are:

$$G = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{bmatrix} \quad H = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

BCH (Bose-Chaudhuri-Hocquenhem) Code



BCH (Bose-Chaudhuri-Hocquenhem) Code

Theorem

Let C be an $[n, k, d]$ cyclic code over $\mathbb{F}_{q=p^m}$, where $p \nmid n$. Let α be a primitive n-th root of unity, and let $g(X)$ be a generating polynomial for C. Suppose there exist integers ℓ and δ such that

$$g(\alpha^{\ell}) = g(\alpha^{\ell+1}) = \cdots = g(\alpha^{\ell+\delta}) = 0$$

Then the minimum distance $d \geq \delta + 2$.

The parity check matrix H is

$$H = \begin{bmatrix} 1 & \alpha^{k+1} & \alpha^{2(k+1)} & \dots & \alpha^{(n-1)(k+1)} \\ 1 & \alpha^{k+2} & \alpha^{2(k+2)} & \dots & \alpha^{(n-1)(k+2)} \end{bmatrix}$$

BCH (Bose-Chaudhuri-Hocquenhem) Code

How to DECODE a received word r for one error?

1. Calculate $rH^T = (s_1, s_2)$.
2. If $s_1 = 0$, then no error (r is a codeword).
3. If $s_1 \neq 0$, compute $\frac{s_2}{s_1} = \alpha^{j-1}$, where j is position of error.
4. $r - e =$ codeword

BCH (Bose-Chaudhuri-Hocquenhem) Code

Example: Consider a $[7, 1, 7]$ BCH code with generating polynomial $g(X) = X^6 + X^5 + X^4 + X^3 + X^2 + X + 1$. There are two codewords: $(0, 0, 0, 0, 0, 0, 0)$ and $(1, 1, 1, 1, 1, 1, 1)$. Suppose Bob receives $r = (1, 1, 1, 0, 1, 1, 1)$. Detect and correct the error!

Solution: Since $rH^T = (s_1, s_2)$, we see

$$rH^T = (1, 1, 1, 0, 1, 1, 1) \begin{bmatrix} 1 & 1 \\ \alpha & \alpha^2 \\ \alpha^2 & \alpha^4 \\ \vdots & \vdots \\ \alpha^6 & \alpha^{12} \end{bmatrix} = (s_1, s_2) = \left(\alpha^3, \alpha^6\right)$$

Since $s_1 \neq 0$, we calculate $\frac{s_2}{s_1} = \frac{\alpha^6}{\alpha^3} = \alpha^3$. Therefore, $j - 1 = 3$, so the error position is at $j = 4$. Finally, we see

$$r - e = \underbrace{(1, 1, 1, 0, 1, 1, 1)}_{\text{received word}} - \underbrace{(0, 0, 0, 1, 0, 0, 0)}_{\text{error vector}} = \underbrace{(\underbrace{1}_{\text{message}}, 1, 1, 1, 1, 1, 1)}_{\text{codeword}}$$

References

- *Introduction to Cryptography with Coding Theory*, by W. Trappe, L. Washington, Pearson, 2nd edition, 2006
- *Fundamentals of Error-Correcting Codes*, by V. Pless, W.C. Huffman, Cambridge University Press, 2003

Conclusion