



Arquitectura de Computadores Avançada

Design principles for Hadamard codes

António Rui Borges

Computation of the code generating matrix for Hadamard code $[8,4,4]_2$

$$\mathbf{x} = \mathbf{m} \times \mathbf{G} = \parallel m_1, m_2, \dots, m_4 \parallel \times \mathbf{G}$$

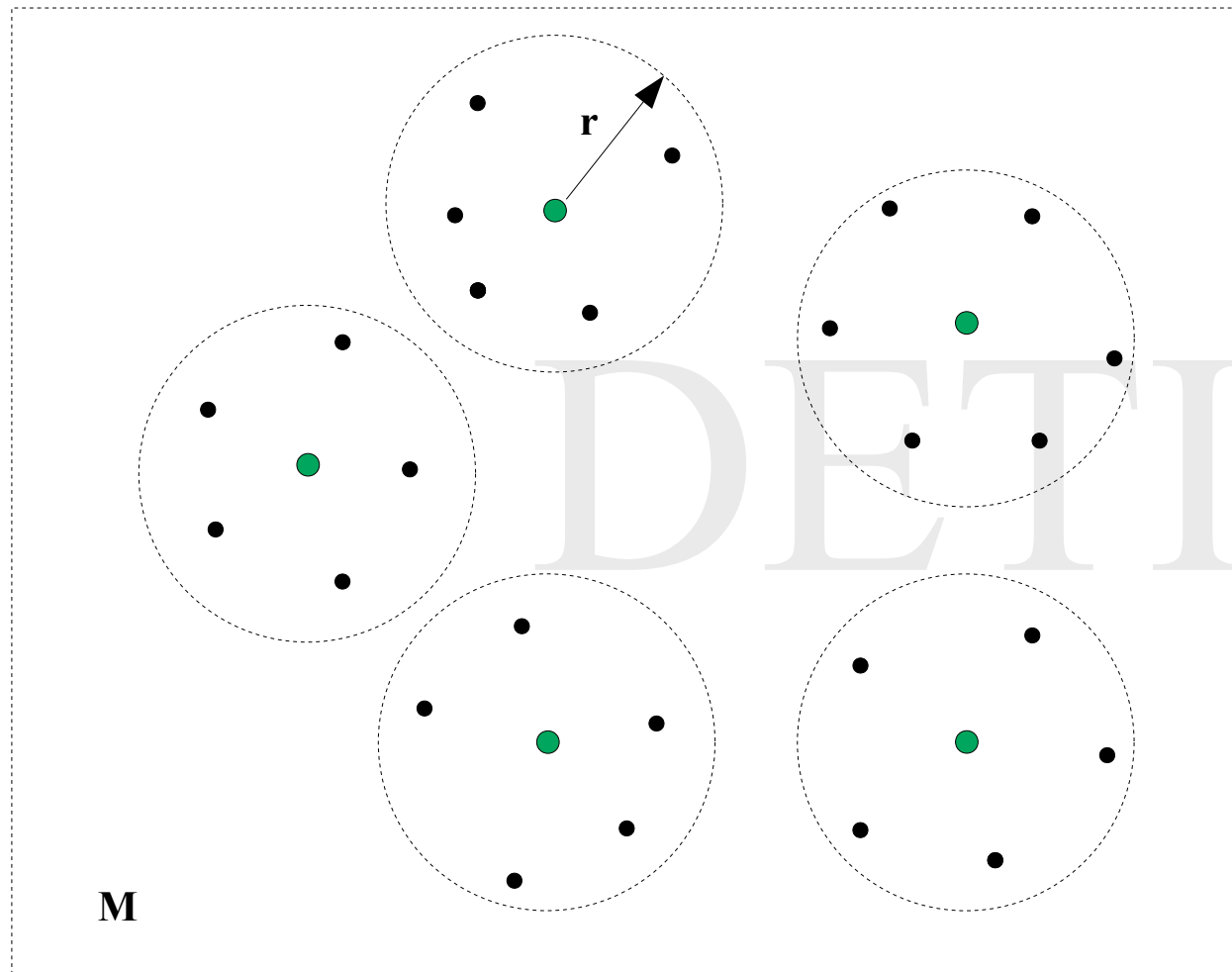
$$\mathbf{G} = \begin{vmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{vmatrix}$$

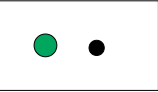
Hadamard code $[8,4,4]_2$

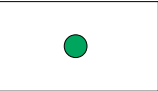
$$\mathbf{x} = \parallel x_0 \ x_1 \ \cdots \ x_7 \parallel = \mathbf{m} \times \mathbf{G} = \parallel m_1 \ m_2 \ \cdots \ m_4 \parallel \times \mathbf{G}$$

$$\begin{aligned} \mathbf{x}^0 &= \parallel 0 \ 0 \ 0 \ 0 \parallel \times \mathbf{G} = \parallel 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \parallel \\ \mathbf{x}^1 &= \parallel 1 \ 0 \ 0 \ 0 \parallel \times \mathbf{G} = \parallel 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 \parallel \\ \mathbf{x}^2 &= \parallel 0 \ 1 \ 0 \ 0 \parallel \times \mathbf{G} = \parallel 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \parallel \\ \mathbf{x}^3 &= \parallel 1 \ 1 \ 0 \ 0 \parallel \times \mathbf{G} = \parallel 0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 0 \parallel \\ \mathbf{x}^4 &= \parallel 0 \ 0 \ 1 \ 0 \parallel \times \mathbf{G} = \parallel 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \parallel \\ \mathbf{x}^5 &= \parallel 1 \ 0 \ 1 \ 0 \parallel \times \mathbf{G} = \parallel 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 0 \parallel \\ \mathbf{x}^6 &= \parallel 0 \ 1 \ 1 \ 0 \parallel \times \mathbf{G} = \parallel 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0 \parallel \\ \mathbf{x}^7 &= \parallel 1 \ 1 \ 1 \ 0 \parallel \times \mathbf{G} = \parallel 0 \ 1 \ 1 \ 0 \ 1 \ 0 \ 0 \ 1 \parallel \\ \mathbf{x}^8 &= \parallel 0 \ 0 \ 0 \ 1 \parallel \times \mathbf{G} = \parallel 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \parallel \\ \mathbf{x}^9 &= \parallel 1 \ 0 \ 0 \ 1 \parallel \times \mathbf{G} = \parallel 1 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0 \parallel \\ \mathbf{x}^{10} &= \parallel 0 \ 1 \ 0 \ 1 \parallel \times \mathbf{G} = \parallel 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \parallel \\ \mathbf{x}^{11} &= \parallel 1 \ 1 \ 0 \ 1 \parallel \times \mathbf{G} = \parallel 1 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \parallel \\ \mathbf{x}^{12} &= \parallel 0 \ 0 \ 1 \ 1 \parallel \times \mathbf{G} = \parallel 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \parallel \\ \mathbf{x}^{13} &= \parallel 1 \ 0 \ 1 \ 1 \parallel \times \mathbf{G} = \parallel 1 \ 0 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \parallel \\ \mathbf{x}^{14} &= \parallel 0 \ 1 \ 1 \ 1 \parallel \times \mathbf{G} = \parallel 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \parallel \\ \mathbf{x}^{15} &= \parallel 1 \ 1 \ 1 \ 1 \parallel \times \mathbf{G} = \parallel 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \parallel \end{aligned}$$

Robustness of Hadamard code $[8,4,4]_2$



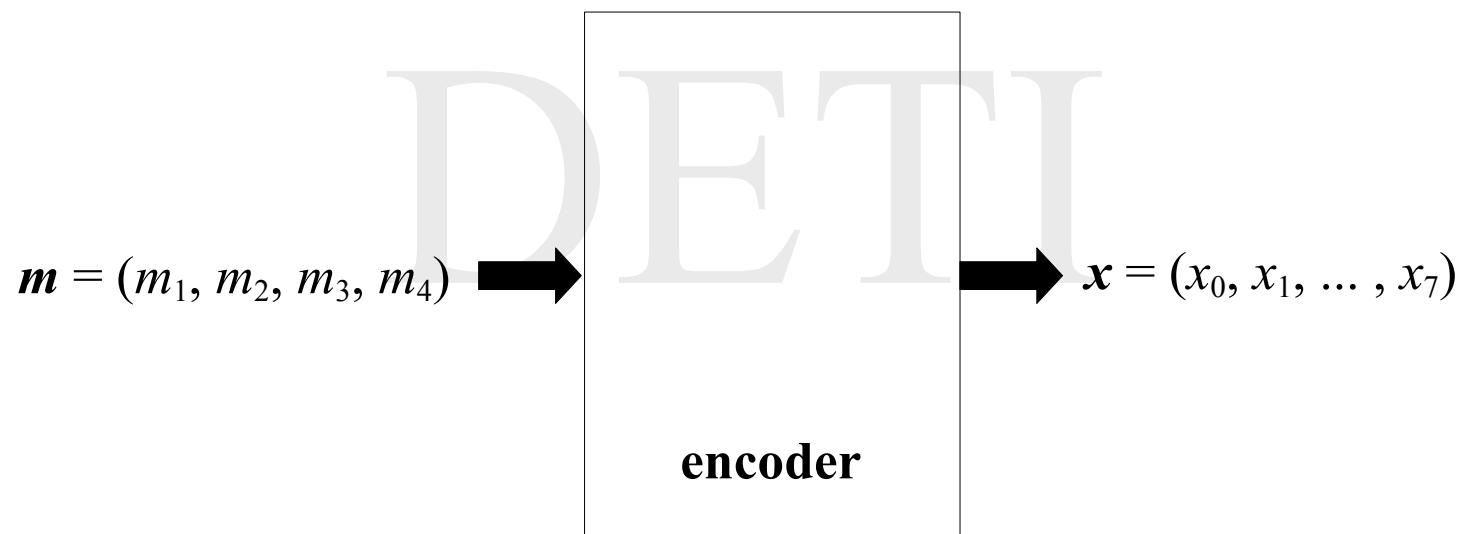
 metric space **M**
where codewords
are immersed
256 elements

 subspace of **M**
consisting solely
of codewords
16 elements

$$r = \frac{d-1}{2} = 1,5$$

maximum radius of each codeword
which allows recovery through
nearest-neighbor approximation
error in 1 bit is corrected
error in 2 bits is detected

Parallel implementation of the encoder for $[8,4,4]_2 - 1$



Parallel implementation of the encoder for $[8,4,4]_2 - 2$

Straightforward implementation

$$x_0 = m_4$$

$$x_1 = m_1 \oplus m_4$$

$$x_2 = m_2 \oplus m_4$$

$$x_3 = m_1 \oplus m_2 \oplus m_4$$

$$x_4 = m_3 \oplus m_4$$

$$x_5 = m_1 \oplus m_3 \oplus m_4$$

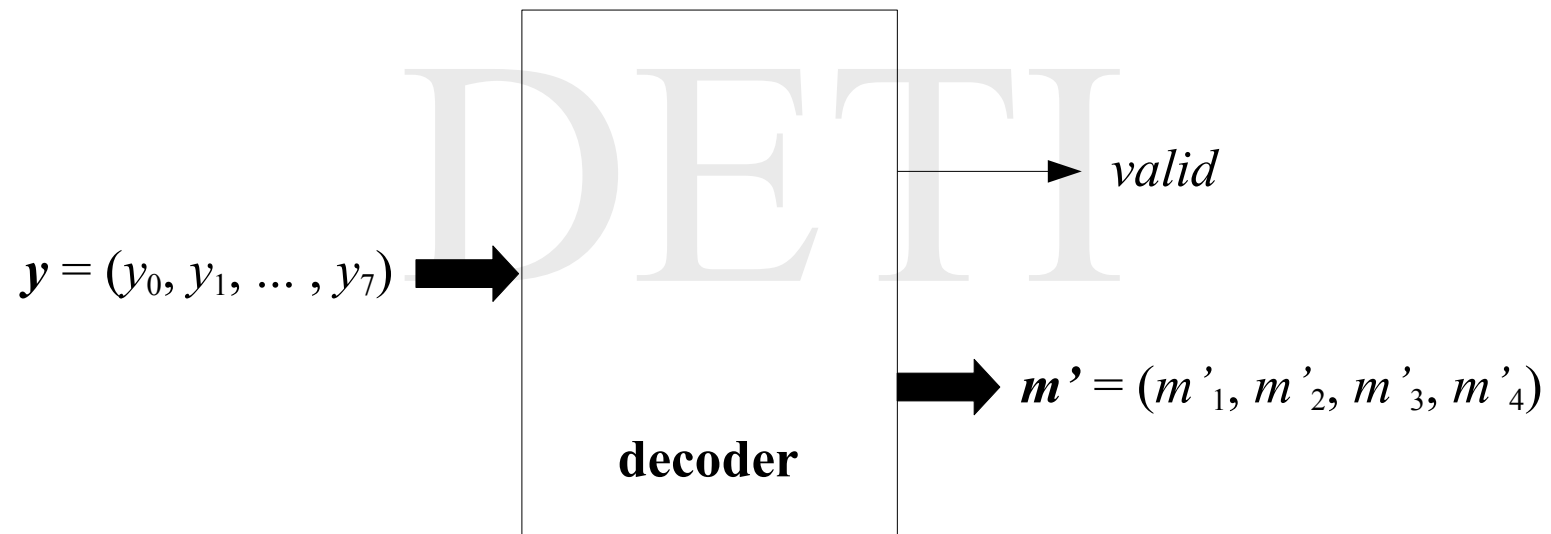
$$x_6 = m_2 \oplus m_3 \oplus m_4$$

$$x_7 = m_1 \oplus m_2 \oplus m_3 \oplus m_4$$



- 12 x-ors are needed
- 3 x-or propagation time delays in the worst case.

Parallel implementation of the decoder for $[8,4,4]_2 - 1$



Parallel implementation of the decoder for $[8,4,4]_2 - 2$

Application of the property of local decodability

$$c_{11} = y_0 \oplus y_1$$

$$c_{12} = y_2 \oplus y_3$$

$$c_{13} = y_4 \oplus y_5$$

$$c_{14} = y_6 \oplus y_7$$

$$c_{21} = y_0 \oplus y_2$$

$$c_{22} = y_1 \oplus y_3$$

$$c_{23} = y_4 \oplus y_6$$

$$c_{24} = y_5 \oplus y_7$$

$$c_{31} = y_0 \oplus y_4$$

$$c_{32} = y_1 \oplus y_5$$

$$c_{33} = y_2 \oplus y_6$$

$$c_{34} = y_3 \oplus y_7$$

		$c_{\#1} c_{\#0}$			
		00	01	11	10
$c_{\#3} c_{\#2}$	00	0	0	E	0
	01	0	E	1	E
	11	E	1	1	1
	10	0	E	1	E