Fehlerkorrigierende Codes

Kurzer Rückblick:

Berechnen Sie den Huffman-Code für

$p(m1)=0.5$
$p(m2)=0.25$
$p(m3)=0.13$
$p(m4)=0.12$

Geben Sie die mittlere Codewortlänge an:

Geben Sie die Entropie an:

Geben Sie die Effizienz an:

Geben Sie die mittlere Wahrscheinlichkeit für den Empfang eines „1“ Bits an:

Welchen Wert hat die mittlere Wahrscheinlichkeit für den Empfang eines „1“ Bits im Allgemeinen? Argumentieren Sie mit der Entropie und der Verkettung einer Stufe von Entropiecodierung mit einer weiteren Stufe von Entropiecodierung.

Die Quelle sendet nun (m3, m2, m4, m1, m1, m4). Geben Sie die Bitfolge an.

Wir verwenden zum Übertragen einen Linearen Block Code in systematischer Darstellung mit der Generatormatrix

$$G = [I_4 \mid P] = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

Geben Sie die (n,k) Notation des Codes an.

Im Folgenden gehen wir davon aus, dass wir das Signal aus dem ersten Aufgabenabschnitt mit diesem Code übertragen wollen.

Wie groß ist die Entropie des übertragenen Signals?

Wie groß ist die Effizienz der beiden hintereinandergeschalteten Codes?

Encodieren Sie die ersten Bits der obigen Nachricht (m3, m2, m4, m1, m1, m4).

Geben Sie die Parity Check-Matrix **$H = [P^T \mid I_{n-k}]$** an und dekodieren Sie das empfangene Signal (Nutzdaten und Parity-Check).

Invertieren Sie das erste Bit Ihrer Nachricht und geben Sie erneut die Nutzdaten und die Parity-Check Bits nach der Decodierung an.

Stellen Sie das Standard-Array auf und berechnen Sie das Syndrom pro Zeile.

Wie viele Bitfehler können Sie mit diesem Code detektieren, wie viele im Minimalfall und im Maximalfall korrigieren?