Capitolul 4

Codarea surselor

Codarea este operaţia prin care simbolurilor unei surse (denumite sursă primară) i se ataşeaza cuvinte de cod formând o nouă sursa de numită sursă secundară. Ideea de bază a codării este ca entropia sursei secundare va fi mai mare decât cea a sursei primare sau altfel spus se reduc, chiar anulează, redundanţa sursei primare.

Un cod bun va încerca să atribuie unui simbol cu probabilitatea o lungime a cuvântului de cod mică (se repetă des deci aici se poate face economie), iar celor cu probabilitate mare cuvinte mai lungi (sunt rare deci pierderile sunt mai mici).

Un exemplu foate des folosit este codul Lempel-Ziv-Welch care stă la baza programelor de tip WinRar, WinZip; rezultatul unei asemenea codării duce la compresia fişierelor si deci la economisirea spatiului pe un disc de stocare a informaţiei.

Mai departe vom formaliza cele spuse în alineatul precedent. Ne vom restrânge considera numai cazul surselor fără memorie.

Sursa primară are alfabetul de lungime N :

$$[S] = [s_1, s_2, \ldots s_N]$$

cu setul de probabilităţi:

$$[P_S] = [p(s_1), p(s_2), \ldots p(s_N)]$$

Sursa secundară este caracterizată de alfabetul (de lungime $D, D \leq N$):

$$[X] = [x_1, x_2, \ldots x_D]$$

Cu aceste simboluri se formează cuvinte de cod. Fiecărui simbol al sursei i se ataşează un cuvânt de cod. Totalitatea cuvintelor formează **codul**.

Ne interesează numai **codurile unic decodabile** adică acele coduri unde unei secvenţe a cuvintelor de cod îi corespunde o unică succesiune de simboluri ale sursei.

Exemple:

1. Fie o sursă cu 4 simboluri: s_1, s_2, s_3, s_4. Acestora le corespund cuvintelor de cod provenite dintr-un alfabet binar : $s_1 \to 00, s_2 \to 01, s_3 \to 10, s_4 \to 11$ Acesta este un cod unic decodabil, pentru ca orice secventă de cuvinte de cod s-ar considera decodarea este simplă: se grupează doi câte biţii; fiecare grup are o singura corespondentă în simbolurile sursei.

2. Fie o sursă cu 4 simboluri: s_1, s_2, s_3, s_4. Corespondenţa este: $s_1 \to 0, s_2 \to 1, s_3 \to 00, s_4 \to 11$. Acesta nu este un cod unic decodabil pentru că unei secvenţe de tipul : 0011 îi poate corespunde secvenţa $s_1 s_1 s_2 s_2$ sau secvenţa $s_3 s_4$. Deci nu avem informaţie certă la ieşirea decodorului.

Mai mult decât atât, în practică se folosesc coduri instantanee sau **ireductibile**. Acestea sunt acele coduri care respectă proprietatea de prefix (nici un cuvânt de cod nu este prefixul altui cuvânt). Un exemplu de cod reductibil este:

$$s_1 \to 0, s_2 \to 01, s_3 \to 011, s_4 \to 0111$$

Acesta este un cod unic decodabil pentru ca orice cuvânt de cod începe cu 0. Nu este un cod instantaneu pentru că la o secventa de tipul 01 nu se poate decide ce simbol a fost transmis şi trebuie să se aştepte bitul următor. Dacă e 0 atunci se decide că anterior a fost transmis s_1, dar cu întârziere. Dacă e 1, secvenţa devine 011 şi mai trebuie aşteptat. Se folosesc coduri instantanee din motive de viteză.

Se defineşte lungimea medie a unui cuvânt de cod:

$$\bar{l} = \sum_{i=1}^{N} p(s_i) l_i \tag{4.1}$$

unde l_i reprezintă lungimea (numărul de simboluri) cuvântului de cod corespunzator simbolului s_i. Se poate demonstra că lungimea uni cuvânt de cod este:

$$\bar{l} = \frac{H(S)}{H(X)} \tag{4.2}$$

După cum am precizat, scopul codării este de a reduce redundanţa, sau echivalent, de a maximiza entropia sursei secundare. Aceasta îşi atinge valoarea maximă pentru probabilităţi egale şi este:

$$H_{max}(X) = \log D$$

În acest caz lungimea medie devine minimă:

$$\bar{l}_{min} = \frac{H(S)}{H_{max}(X)} = \frac{H(S)}{\log D} \tag{4.3}$$

Altfel spus un cod bun este acela care minimizeaza lungimea medie.

Se defineşte **eficienţa** codului ca fiind raportul între lungimea medie minimă a codului şi lungimea medie a unui cuvânt de cod.

$$\eta = \frac{\bar{l}_{min}}{\bar{l}} \tag{4.4}$$

Un cod se numeşte **absolut optimal** dacă are eficienţa egală cu unitatea sau, altfel spus, un cod pentru care lungimea medie este minimă şi deci entrpoia sursei a devenit maximă.

Un cod **optimal** este cel care pentru care se obţine cea mai mică lungime medie posibilă. Trebuie precizat că absolut optimalitatea este data de setul de probabilităti ale sursei si nu de cod. Adică există seturi de probabilităţi pentru care nu exista cod absolut optimal.

În practică se caută coduri optimale; acestea, în anumite situaţii particulare, devin absolut optimale. Un procedeu de codare optimală este data de Huffman în codul care-i poartă numele.

Codul Huffman devine absolut optimal dacă setul de probabilităţi al sursei primare conţine numai valori de tipul D^{-k}, cu $k \in \mathbf{N}$

Se poate demonstra că un cod ireductibil îndeplineşte inegalitatea Kraft-McMillan:

$$\sum_{i=1}^{N} D^{-l_i} \leqslant 1 \tag{4.5}$$

Se poate arăta (vezi problema rezolvată 3) că un cod optimal trebuie să indeplinească relaţia:

$$\frac{N-D}{D-1} \in \mathbf{N} \tag{4.6}$$

4.1 Probleme rezolvate

1. [3] *O sursă S generează 6 simboluri, de probabilităţi descrise de vectorul $P = [0.3; 0.25; 0.2; 0.1; 0.1; 0.05]$. Sursa este codată optimal, simbol cu simbol, cu cuvinte de cod generate cu simboluri dintr-un alfabet ternar. Să se calculeze cuvintele de cod, arborele de codare şi eficienţa codării.*

   Rezolvare: Codarea optimală a unei surse se realizează conform algoritmului Huffman. Se ştie că numărul de simboluri ale sursei ce se codează trebuie să îndeplinească relaţia (4.6); în acest caz, $N = 6$, $D = 3$ şi deci

   $$\frac{N-D}{D-1} = \frac{6-3}{3-1} = \frac{3}{2} \notin \mathbf{N}$$

   Completarea se face adăugând sursei simboluri de probabilitate nulă; în acest caz, cu un singur astfel de simbol adăugat obţinem $N = 7$ şi

   $$\frac{N-D}{D-1} = \frac{7-3}{3-1} = \frac{4}{2} \in \mathbf{N}$$

   Entropia sursei extinse S este dată de:

   $$\begin{aligned} H(S) &= -\sum_{i=1}^{7} p(s_i) \log p(s_i) \\ &= -\left(0.3 \log 0.3 + 0.25 \log 0.25 + 0.2 \log 0.2 + 2 \cdot 0.1 \log 0.1\right) - \\ &\quad -0.05 \log 0.05 - 0 \log 0 = 2.366 \text{ bit/simbol} \end{aligned}$$

   Atunci, conform (4.3), lungimea medie minimă a unui cuvânt de cod este:

   $$\bar{l}_{\min} = \frac{H(S)}{\log D} = \frac{2.366}{\log 3} = 1.493$$

   Codarea Huffman este realizată conform tabelului 4.1.

Sursă primară	$p(s_i)$	Cod	Sursă redusă	$p(r_{ji})$	Cod	Sursă redusă	$p(r_{ji})$	Cod
						r_{21}	0.45	0
s_1	0.3		r_{11}	0.3		r_{22}	0.3	1
s_2	0.25		r_{12}	0.25		r_{23}	0.25	2
s_3	0.2		r_{13}	0.2	00			
			r_{14}	0.15	01			
s_4	0.1		r_{15}	0.1	02			
s_5	0.1	010						
s_6	0.05	011						
s_7	0	012						

Tabela 4.1: Codare Huffman

Cuvintele de cod sunt grupate în tabelul 4.2; pe baza lor putem calcula lungimea medie a cuvintelor de cod:

$$\bar{l} = \sum_{i=1}^{7} l_i p(s_i) = 1 \cdot 0.3 + 1 \cdot 0.25 + 2 \cdot 0.2 + 2 \cdot 0.1 + 3 \cdot 0.1 + 3 \cdot 0.05 + 3 \cdot 0 = 1.6$$

Eficienţa codării este atunci:

$$\eta = \frac{\bar{l}_{\min}}{\bar{l}} = \frac{1.493}{1.6} = 0.933$$

Sursă primară	$p(s_i)$	Cod	Lungime
s_1	0.3	1	1
s_2	0.25	2	1
s_3	0.2	00	2
s_4	0.1	02	2
s_5	0.1	010	3
s_6	0.05	011	3
s_7	0	012	3

Tabela 4.2: Coduri asociate simbolurilor sursei şi lungimile lor

2. [4] *Un cod are 2 cuvinte de lungime 1, 2 cuvinte de lungime 2, 2 cuvinte de lungime 3, 3 cuvinte de lungime 4 şi un cuvânt de lungime 5 şi este construit cu simboluri dintr-un alfabet cu* $D = 3$. *Să se studieze proprietăţile de ireductibilitate ale codului.*
Rezolvare: Din definiţiile noţiunilor fundamentale de ireductibilitate şi prefix, putem sintetiza următoarele: un cod este ireductibil dacă are proprietatea de prefix, adică nici un cuvânt al codului nu este prefix pentru un alt cuvânt al codului. Acesta înseamnă, pe de o parte satisfacerea inegalităţii McMillan (4.5), iar din punctul de vedere a construirii arborelui de codare, fiecare cuvânt de cod trebuie să fie nod terminal (frunză).

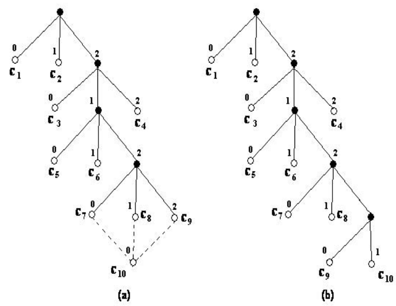

Figura 4.1: Arbore de codare. COd reductibil (a) si cod ireductibil (b)

Verificarea inegalităţii (4.5) în condiţiile date ($D = 3$, $N = 10$, $l = [1, 1, 2, 2, 3, 3, 4, 4, 4, 5]$) conduce la:

$$2 \cdot 3^{-1} + 2 \cdot 3^{-2} + 2 \cdot 3^{-3} + 3 \cdot 3^{-4} + 1 \cdot 3^{-5} = 1 + 3^{-5} > 1$$

Inegalitatea (4.5) nu este verificată, şi deci codul nu este ireductibil. Acelaşi lucru poate fi constatat şi pe arborele de codare (prezentat în figura 4.1): arborele este ternar (fiecare nod, cu excepţia nodurilor terminale, are trei descendenţi) iar satisfacerea proprietăţii de prefix înseamnă că fiecare cuvânt (nod terminal) nu provinde din alt cuvânt. Pentru datele problemei aceasta nu este posibilă: cuvântul de lungime 5 nu poate fi construit decât dintr-un alt cuvânt de cod (altfel spus, în arborele de codare nu mai există nici o poziţie liberă de la care să se construiască cuvinte noi); corectarea acestei situaţii se poate face prin transformarea unui cuvânt de lungime 4 în lungime 5.

3. [3] *Să se demonstreze formula (4.6) prin care se verifică necesitatea extinderii sursei de codat cu simboluri de probabilitate nulă.*

   Rezolvare: Principiul codării Huffman cu D simboluri în alfabetul codului este reducerea iterativă a sursei primare, prin gruparea ultimelor D simboluri (cele mai puţin probabile). Aceste ultime D simboluri se înlocuiesc (în sursa redusă) cu un unic simbol; astfel, prima sursă redusă are deci un număr de $N - D + 1$ simboluri; a doua sursă redusă are deci $(N - D + 1) - D + 1$ simboluri, ş.a.m.d. Prin inducţie matematică se poate demonstra cu uşurinţă că după k paşi de reducere, sursa redusă de ordinmul k are un număr de $N - k(D - 1)$ simboluri.

   Ultima sursă redusă trebuie să aibă exact D simboluri (fiecărui simbol alocându-i-se un simbol din alfabetul codului); atunci, dacă numărul de paşi de reducere necesari este n, obţinem relaţia necesară

   $$N - n(D - 1) = D$$

   de unde rezultă că

   $$n = \frac{N - D}{D - 1}$$

   Dar n este un număr natural (este numărul de etape de reducere), şi deci este evident că trebuie satisfăcută condiţia (4.6), deci

   $$\frac{N - D}{D - 1} \in \mathbf{N}$$

4.2 Probleme propuse

1. [2] O sursă de informaţie $S1$ generează în mod independent 6 simboluri, ale căror probabilităţi sunt descrise de vectorul $P1$, iar o sursă de informaţie $S2$ generează în mod independent 7 simboluri, ale căror probabilităţi sunt descrise de vectorul $P2$. a) Să se codeze simbolurile surselor prin cuvinte compuse din simboluri ale alfabetului $X = \{0, 1, 2\}$, astfel încât eficienţa codării să fie maximă. b) Să se realizeze codarea Huffman binară a simbolurilor surselor $S1$ şi $S2$. c) Să se calculeze şi să se compare lungimile medii, eficienţa şi arborii de codare pentru codările realizate la a) şi b).

   (a) $P1 = [0.2; 0.05; 0.25; 0.21; 0.25; 0.04]$; $P2 = [0.03; 0.32; 0.26; 0.1; 0.1; 0.03; 0.16]$

   (b) $P1 = [0.1; 0.01; 0.26; 0.14; 0.13; 0.27]$; $P2 = [0.01; 0.2; 0.2; 0.2; 0.01; 0.01; 0.37]$

   (c) $P1 = [0.01; 0.3; 0.3; 0.32; 0.04; 0.03]$; $P2 = [0.22; 0.22; 0.1; 0.06; 0.16; 0.06; 0.18]$

   (d) $P1 = [0.17; 0.18; 0.13; 0.15; 0.15; 0.22]$; $P2 = [0.05; 0.05; 0.05; 0.05; 0.5; 0.15; 0.15]$

   (e) $P1 = [0.3; 0.1; 0.1; 0.13; 0.17; 0.2]$; $P2 = [0.07; 0.25; 0.08; 0.12; 0.18; 0.05; 0.25]$

   (f) $P1 = [0.25; 0.15; 0.25; 0.07; 0.12; 0.16]$; $P2 = [0.1; 0.1; 0.1; 0.1; 0.1; 0.25; 0.25]$

   (g) $P1 = [0.18; 0.11; 0.1; 0.06; 0.23; 0.32]$; $P2 = [0.05; 0.4; 0.1; 0.1; 0.03; 0.08; 0.24]$

   (h) $P1 = [0.04; 0.16; 0.14; 0.28; 0.22; 0.16]$; $P2 = [0.16; 0.18; 0.16; 0.17; 0.15; 0.14; 0.04]$

   (i) $P1 = [0.26; 0.18; 0.24; 0.04; 0.08; 0.2]$; $P2 = [0.2; 0.2; 0.16; 0.14; 0.03; 0.16; 0.11]$

   (j) $P1 = [0.14; 0.1; 0.3; 0.01; 0.43; 0.02]$; $P2 = [0.15; 0.15; 0.1; 0.15; 0.15; 0.2; 0.1]$

   (k) $P1 = [0.22; 0.29; 0.07; 0.05; 0.1; 0.27]$; $P2 = [0.25; 0.1; 0.11; 0.13; 0.16; 0.1; 0.15]$

   (l) $P1 = [0.23; 0.05; 0.25; 0.15; 0.15; 0.17]$; $P2 = [0.03; 0.07; 0.17; 0.15; 0.25; 0.07; 0.26]$

   (m) $P1 = [0.05; 0.15; 0.23; 0.16; 0.26; 0.15]$; $P2 = [0.15; 0.15; 0.08; 0.11; 0.5; 0.2; 0.16]$

   (n) $P1 = [0.05; 0.4; 0.16; 0.05; 0.24; 0.1]$; $P2 = [0.14; 0.39; 0.1; 0.01; 0.25; 0.03; 0.08]$

   (o) $P1 = [0.2; 0.18; 0.38; 0.05; 0.07; 0.12]$; $P2 = [0.2; 0.15; 0.07; 0.22; 0.1; 0.18; 0.08]$

   (p) $P1 = [0.2; 0.03; 0.21; 0.2; 0.26; 0.1]$; $P2 = [0.15; 0.22; 0.08; 0.2; 0.23; 0.06; 0.06]$

   (q) $P1 = [0.1; 0.05; 0.2; 0.14; 0.07; 0.44]$; $P2 = [0.14; 0.14; 0.14; 0.14; 0.14; 0.14; 0.02]$

   (r) $P1 = [0.15; 0.09; 0.12; 0.22; 0.25; 0.17]$; $P2 = [0.1; 0.1; 0.2; 0.2; 0.05; 0.05; 0.3]$

   (s) $P1 = [0.15; 0.1; 0.2; 0.16; 0.16; 0.23]$; $P2 = [0.01; 0.01; 0.01; 0.07; 0.3; 0.3; 0.3]$

   (t) $P1 = [0.3; 0.26; 0.08; 0.16; 0.15; 0.05]$; $P2 = [0.25; 0.2; 0.05; 0.125; 0.125; 0.125; 0.125]$

   (u) $P1 = [0.05; 0.38; 0.05; 0.25; 0.24; 0.03]$; $P2 = [0.1; 0.3; 0.02; 0.08; 0.05; 0.05; 0.4]$

2. O sursă generează în mod independent simbolurile s_1 şi s_2. Să se construiască sursele extinse de ordinele 2 şi 3 şi să se codeze optim acestea cu un alfabet binar; să se calculeze creşterea de eficacitate obţinută prin extensia sursei primare, dacă: 1) $p(s_1) = 0.01$ 2) $p(s_1) = 0.02$ 3) $p(s_1) = 0.03$ 4) $p(s_1) = 0.04$ 5) $p(s_1) = 0.05$ 6) $p(s_1) = 0.06$ 7) $p(s_1) = 0.07$ 8) $p(s_1) = 0.08$ 9) $p(s_1) = 0.1$ 10) $p(s_1) = 0.12$ 11) $p(s_1) = 0.15$ 12) $p(s_1) = 0.2$ 13) $p(s_1) = 0.22$ 14) $p(s_1) = 0.25$ 15) $p(s_1) = 0.27$ 16) $p(s_1) = 0.3$ 17) $p(s_1) = 0.33$ 18) $p(s_1) = 0.35$ 19) $p(s_1) = 0.37$ 20) $p(s_1) = 0.40$ 21) $p(s_1) = 0.42$ 22) $p(s_1) = 0.45$ 23) $p(s_1) = 0.47$ 24) $p(s_1) = 0.48$ 25) $p(s_1) = 0.49$.

3. O sursă generează în mod independent simbolurile s_1, s_2 şi s_3. Să se construiască sursa extinsă de ordinul 2 şi să se codeze optim cu un alfabet binar, respectiv ternar; să se calculeze creşterea de eficacitate obţinută prin extensia sursei primare, dacă:

   (a) $p(s_1) = 0.1p(s_2) = 0.1$
   (b) $p(s_1) = 0.1, p(s_2) = 0.2$
   (c) $p(s_1) = 0.1, p(s_2) = 0.3$
   (d) $p(s_1) = 0.1, p(s_2) = 0.5$
   (e) $p(s_1) = 0.1, p(s_2) = 0.6$
   (f) $p(s_1) = 0.2, p(s_2) = 0.2$
   (g) $p(s_1) = 0.2, p(s_2) = 0.3$
   (h) $p(s_1) = 0.2, p(s_2) = 0.5$
   (i) $p(s_1) = 0.2, p(s_2) = 0.7$
   (j) $p(s_1) = 0.3, p(s_2) = 0.1$
   (k) $p(s_1) = 0.3, p(s_2) = 0.25$
   (l) $p(s_1) = 0.3, p(s_2) = 0.4$
   (m) $p(s_1) = 0.3, p(s_2) = 0.5$
   (n) $p(s_1) = 0.5, p(s_2) = 0.1$
   (o) $p(s_1) = 0.5, p(s_2) = 0.3$
   (p) $p(s_1) = 0.5, p(s_2) = 0.4$
   (q) $p(s_1) = 0.8, p(s_2) = 0.1$
   (r) $p(s_1) = 0.7, p(s_2) = 0.2$
   (s) $)p(s_1) = 0.6, p(s_2) = 0.3$
   (t) $p(s_1) = 0.25, p(s_2) = 0.25$
   (u) $p(s_1) = 0.33, p(s_2) = 0.2$
   (v) $p(s_1) = 0.4, p(s_2) = 0.4$
   (w) $p(s_1) = 0.33, p(s_2) = 0.33$
   (x) $p(s_1) = 0.4, p(s_2) = 0.59$
   (y) $p(s_1) = 0.05, p(s_2) = 0.05$.

   Indicaţie: Sursa extinsă de ordinul 2 presupune crearea simbolurilor extinse: $se_1 = s_1s_1$, $se_2 = s_1s_2$, $se_3 = s_1s_3$, $se_4 = s_2s_1$, $se_5 = s_2s_2$, $se_6 = s_2s_3$, $se_7 = s_3s_1$, $se_8 = s_3s_2$, $se_9 = s_3s_3$. Probabilităţile simbolurilor sursei extinse se obţin prin înmulţirea probabilităţilor sursei primare (este o sursă fără memorie). Această sursa se codează ca orice sursa cu 9 simboluri. Eficientă codării unei surse extinse va fi întotdeauna mai mare (sau măcar egală) decât codarea sursei primare. Egalitatea se obţine, de pildă, pentru o eficienta egală 1 a sursei primare.

4. Să se estimeze lungimea maximă a cuvintelor de cod generate printr-o codare optimală (după un alfabet cu D litere) a unei surse de informaţie cu N simboluri. Să se verifice dacă este posibilă o codare (neoptimală), dar care să aibă o lungime maximă a cuvintelor de cod mai mică decât cea determinată anterior.

5. [8] Tastatura unui terminal telefonic fără fir are 12 taste: cifrele şi tastele de control # şi *. Probabilitatea de acţionare a tastelor este aceeaşi, cu excepţia tastelor 0, 6, şi 7, care se folosesc de două ori mai des, şi a tastelor de control, care se folosesc de două ori mai rar. Dacă transmisia codurilor asociate tastelor se face pe un canal binar, să se codeze acţiunile de apăsare a tastelor optimal şi după un cod ZCB (Zecimal Codificat Binar) şi să se calculeze parametrii codurilor obţinute.

6. [8] Tastatura unui terminal telefonic fără fir are 12 taste: cifrele şi tastele de control # şi *. Probabilitatea de acţionare a tastelor este aceeaşi, cu excepţia tastelor 0, şi 5, care se folosesc de trei ori mai des, a tastelor 9, 6 şi 7 care se folosesc de două ori mai des, şi a tastelor de control, care se folosesc de două ori mai rar. Transmisia codurilor asociate tastelor se face prin asocierea de frecvenţe. Să se codeze optimal acţiunile de apăsare a tastelor şi să se calculeze parametrii codurilor pentru folosirea a trei, şi respectiv patru frecvenţe.

7. [8] Un program scris pentru un micro-controler este compus din cuvinte de 4 biţi (*nibble*). Pentru reducerea spaţiului de stocare (şi pentru secretizare), programul este codat optimal cu două simboluri. Să se genereze cuvintele de cod şi să se calculeze parametrii codului, ştiind că probabilităţile de apariţie a cuvintelor din program sunt egale, cu excepţia cuvintelor 0H şi FH ce au o probabilitate de patru ori mai mare, şi a cuvintelor 1H, 2H şi 3H, ce au o probabilitate de trei ori mai mică.

8. [2] Se consideră sursa de informaţie text scris cu litere mari într-o limbă latină, fără semne diacritice, fără punctuaţie, incluzând spaţiul. Dându-se probabilităţile de apariţie a diferitelor litere, să se genereze codul optimal corespunzător şi să se calculeze parametrii acestuia, pentru alfabete de cod cu 2, respectiv 4 simboluri. Probabilitătile de apariţie a diferitelor litere sunt, pentru limba română: $p(A) =$, $p(B) =$, $p(C) =$, $p(D) =$, $p(E) =$, $p(F) =$, $p(G) =$, $p(H) =$, $p(I) =$, $p(J) =$, $p(K) =$, $p(L) =$, $p(M) =$, $p(N) =$, $p(O) =$, $p(P) =$, $p(Q) =$, $p(R) =$, $p(S) =$, $p(T) =$, $p(U) =$, $p(V) =$, $p(W) =$, $p(X) =$, $p(Y) =$, $p(Z) =$, $p(\text{spaţiu}) =$.

9. [8] Tastatura terminalului unui programator de microcontroler are 10 taste: cifrele octale (în baza opt) şi tastele de control OK şi C. Probabilitatea de acţionare a tastelor este aceeaşi, cu excepţia tastelor 0, 1, şi 7, care se folosesc de două ori mai des, şi a tastelor de control (OK şi C), care se folosesc de două ori mai rar. Dacă transmisia codurilor asociate tastelor se face pe un canal binar, să se codeze acţiunile de apăsare a tastelor optimal şi după un cod ZCB (Zecimal Codificat Binar) şi să se calculeze parametrii codurilor obţinute.

10. [8] O sursă ce emite N simboluri este codată absolut optimal, astfel încât lungimea maximă a cuvintelor de cod obţinute este $l_{\max}$. Să se deducă distribuţia lungimilor cuvintelor de cod (deci câte cuvinte au lungimea 1, 2, ..., $l_{\max}$). Particularizaţi pentru $N = 8$, $D = 2$ şi $l_{\max} = 4$.

Bibliografie

[1] Mihai Ciuc. "Note de seminar".

[2] A. T. Murgan, I. Spânu, I. Gavăt, I. Sztojanov, V. E. Neagoe, şi A. Vlad. *Teoria Transmisiunii Informatîei - probleme*. Editura Didactică şi Pedagogică, Bucureşti, România, 1983.

[3] Alexandru Spătaru. *Teoria Transmisiunii Informatţiei*. Editura Didactică şi Pedagogică, Bucureşti, România, 1983.

[4] Alexandru Spătaru. *Fondements de la theorie de la transmisssion de línformation*. Presses polytechniques romandes, Lausanne, Elveţia, 1987.

[5] Rodica Stoian. "Note de seminar".

[6] Dan Alexandru Stoichescu. "Note de seminar".

[7] Eugen Vasile. "Note de seminar".

[8] Constantin Vertan. "Note de seminar".