CAPITOLUL II

COMPACTAREA DATELOR

II.1 Surse discrete de informație

O sursă de semnal se numește **discretă** dacă generează simboluri dintr-un alfabet finit (de exemplu: binar, hexazecimal, alfanumeric etc), la momente discrete de timp.

Modelarea matematică a surselor discrete de informație se face pe baza teoriei probabilităților [Mun97], [Pap65], [Zie97].

MODELUL I: Sursa discretă, completă, fără memorie (SDCFM) se definește printr-un set finit de **caractere (mesaje)** (s_1, s_2,..., s_n) independente, asociat cu un **câmp de probabilități** (p_1,p_2,...p_n):

$$S:\begin{pmatrix} s_1 & s_2 & \dots & s_n \\ p_1 & p_2 & \dots & p_n \end{pmatrix} \quad \text{(II.1)}$$

satisfăcând relația:

$$\sum_{k=1}^{n} p_k = 1 \quad \text{(II.2)}$$

Informația furnizată de mesajul s_k se exprimă cu relația:

$$i_k = -\log_2 p_k \text{ (biți)} \quad \text{(II.3)}$$

Valoarea medie a informațiilor furnizate de mesajele sursei se numește **entropie** și se determină cu relația:

$$H(S) = \overline{i_k} \overset{\Delta}{=} \sum_{k=1}^{n} p_k i_k = -\sum_{k=1}^{n} p_k \log_2 p_k \quad (biti/mesaj) \quad \text{(II.4)}$$

Valoarea maximă a entropiei se obține în cazul generării echiprobabile a mesajelor:

$$\max[H(S)] = \log_2 n \quad \text{(II.5)}$$

(**n** - numărul mesajelor sursei.)

Redundanța semnalului generat de sursă este definită prin relația:

$$R = \max [H(S)] - H(S) \quad \text{(II.6)}$$

Eficiența sursei se definește ca raportul dintre entropia reală și valoarea sa maximă.

$$\eta = \frac{H(S)}{\max[H(S)]} \quad \text{(II.7)}$$

MODELUL al II-lea: Extensia de ordin m a sursei DCFM generează grupe de m mesaje ale sursei inițiale.

Entropia sursei extinse S^m se calculează cu relația:

$$H(S^m) = mH(S) \tag{II.8}$$

MODELUL al III-lea: **Sursa discretă cu memorie** (SDM) sau **sursa Markov** se caracterizează prin faptul că fiecare mesaj furnizat depinde de unul sau mai multe mesaje anterioare.

Ordinul sursei este dat de numărul momentelor de timp la care un mesaj influențează ieșirea sursei cu excepția momentului propriu.

Starea SDM de ordin m se definește ca succesiunea ultimelor m mesaje furnizate anterior momentului considerat.

Descrierea SDM se face fie prin metoda grafurilor orientate cu n^m noduri (stări), fie matricial cu matrici de probabilități staționare P și de tranziție T stochastice.

Dacă elementele matricii de tranziție sunt invariante în timp, sursa se numește **staționară**.

Prin definiție, SDM staționară este **ergodică** dacă este satisfăcută relația:

$$\lim_{t\to\infty} T^t = K \tag{II.9}$$

unde K este o matrice cu elemente constante finite.

Entropia sursei DM ergodice se determină conform relației:

$$H(S) = -\sum_{k=1}^{n^m}\sum_{j=1}^{n} p_k p(s_j|S_k)\log_2 p(s_j|S_k) \tag{II.10}$$

Notații:

m - ordinul sursei;

n - numărul mesajelor sursei;

p_k - probabilitatea stării S_k (k = 1, 2.. n^m);

$p(s_j \mid S_k)$-probabilitatea de generare a mesajului s_j din starea S_k (j = 1, 2.. n).

Observație: În cazul transmisiei digitale a textelor este suficientă modelarea sursei ca sursă Markov de ordinul 1 sau 2. Modelele de ordin superior necesită circuite de codare/decodare pentru compactarea datelor de mare complexitate, cu câștiguri nesemnificative în ceea ce privește reducerea redundanței semnalului transmis.

Optimizarea codării textelor poate fi realizată prin folosirea unui dicționar de cuvinte uzuale în care sunt incluse și literele alfabetului respectiv, pentru formarea cuvintelor rare, neintroduse în dicționar.

Mai nou, managementul comunicațiilor utilizează **modele preferențiale** ale surselor informaționale [Tut88],[Mun92], prin care se stabilesc ierarhiile priorităților mesajelor.

Modelul preferențial al unei surse discrete S cu N mesaje $\{s_1, s_2....s_N\}$ constă într-un **set de ponderi calitative** $\{K_1, K_2....K_N\}$ asociate mesajelor, referitor la importanța, semnificația sau relevanța acestora:

$$S:\begin{pmatrix} s_1 & s_2 & \dots & s_N \\ k_1 & k_2 & \dots & k_N \end{pmatrix} \qquad \text{(II.11)}$$

Pe baza modelului preferențial al sursei, se poate construi **modelul probabilistic cu preferințe**:

$$S:\begin{pmatrix} s_1 & s_2 & \dots & s_N \\ p_1 & p_2 & \dots & p_N \\ k_1 & k_2 & \dots & k_N \end{pmatrix} \qquad \text{(II.12)}$$

unde $\bar{p} = (p_1, p_2.....p_N)$ este vectorul probabilităților mesajelor iar $\bar{k} = (k_1, k_2.....k_N)$ este vectorul ponderilor.

În cazul necunoașterii probabilităților mesajelor, se poate deduce vectorul de probabilități pe baza vectorului **k,** pe criteriul *maxentropiei*.

Topologia mulțimii preferințelor

Coeficienții k_1, k_2 ... k_N iau valori dintr-un set discret $\{v_1,...,v_L\}$, $L \leq N$, asemănător modelelor fuzzy, care corespund gradelor calitative: {foarte mic; mic; tipic; mare; foarte mare}.Valorile asociate acestor criterii calitative pot respecta o scală simetrică sau asimetrică, liniară sau neliniară (compresată sau expandată). (Fig. II.1; II.2)

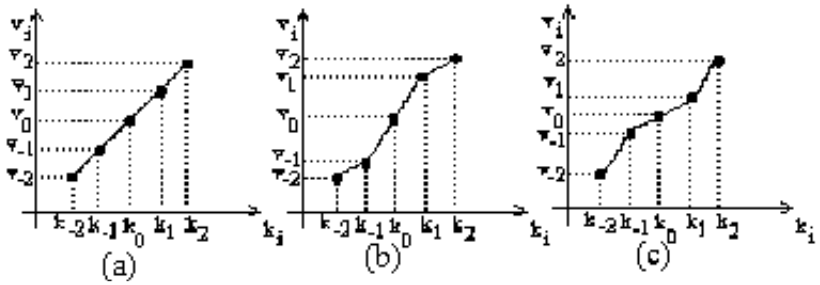

Fig.II.1: Caracteristica simetrică (a) liniară; (b) compresată; (c) expandată.

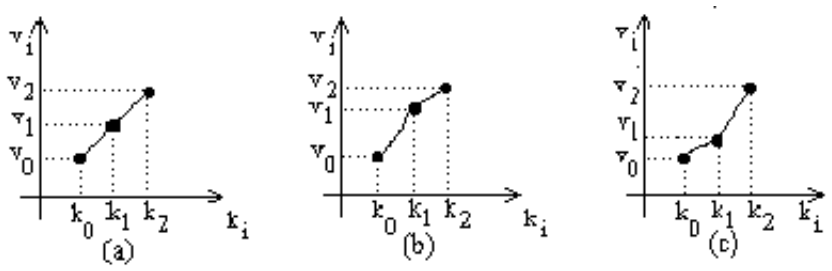

Fig.II.2: Caracteristica asimetrică (a) liniară; (b) compresată; (c) expandată.

În general se lucrează cu 2; 3; 4 sau 5 valori de pondere. Deci vor exista mai multe mesaje cu aceeași pondere k, prioritar echivalente.

Dacă se notează $d_i = v_{i+1} - v_i$ atunci:

Cazul I: pentru $d_1 = d_2 = d_3 = ...$ se obține caracteristica liniară a preferințelor.

Cazul II: pentru $d_1 < d_2 < d_3$ .... se obține caracteristica preferențială expandată.

Cazul III: pentru $d_1 > d_2 > d_3$ .... se obține caracteristica preferențială compresată.

Valorile ponderilor calitative pot fi alese, în funcție de aplicație:

◆ în *progresie aritmetică* (exemple: {0.5 ; **1**; 1.5} ; {**1**; 2; 3; ...}).

◆ în *progresie geometrică* (exemple: {0.5; **1**; 2} ; {0.25; 0.5; **1**; 2; 4}).

(**1** reprezintă valoarea ponderii mesajelor standard - *default value*).

Entropia surselor de informație fără memorie modelate probabilistic și preferențial [Mun92] este dată de relația:

$$H(s) = -\sum_i p_i k_i \log_2 p_i \cong -\sum_i p_i \log_2 p_i \qquad \text{(II.13)}$$

Valoarea maximă a entropiei se obține pentru setul de probabilități:

$$p_i = 2^{k_i} / \sum_i 2^{k_i} \ , \ i = \overline{1,N} \qquad \text{(II.14)}$$

Aceste modele combinate sunt utile pentru realizarea unor coduri mai eficiente, neuniforme de compresie, compactare sau de corecție a erorilor (**UEP** - *UnEqual Protectted Codes)* [Aks95].

Exemple:

◆ În cazul imaginilor discrete alb-negru, se pot utiliza pentru proiectarea unor coduri de compactare, modele probabilistice cu caracteristică expandată.

◆ În sistemul de comunicații GSM (*Global System for Mobile Communications)* [Kuc91], se utilizează un cod de compresie cu procesare diferențiată a vocalelor față de consoane, care permite reducerea ratei de transmisie de la 104 kbps la 13 kbps.

◆ În același sistem, se utilizează un cod corector de erori de tip UEP care asigură corecția erorilor numai pentru 182 de biți dintr-un cadru de 260, clasificarea respectivă a biților realizându-se pe criteriul importanței acestora în asigurarea calității semnalului vocal transmis.

II.2 Coduri Huffman pentru compactarea datelor

Compactarea datelor este procedeul de reducere a redundanței semnalului transmis prin păstrarea constantă a entropiei acestuia astfel încât datele pot fi extrase fără erori prin decodarea secvenței codate.

Compactarea este utilă în cazul semnalelor furnizate de sursele de informație cu redundanță și debit binar generat de valori foarte mari. Sursele discrete de informație (facsimil; voce; imagini mobile alb-negru sau color; înregistrări digitale etc) produc debite binare relativ mari (de ordinul sutelor de Gbps). Semnalele digitale furnizate de aceste surse prezintă în general o redundanță ridicată. Reducerea numărului de biți care trebuie transmiși se realizează prin aplicarea unui cod adecvat de compactare. În prezent se utilizează mai multe tipuri de coduri pentru compactare. Unele presupun cunoașterea apriorică a sursei și folosesc modelul probabilistic deja construit al acesteia. Alte metode de compactare modelează sursa din chiar semnalul digital transmis care constituie semnal de intrare în codor. Cele mai avansate tehnici de compactare a datelor nu necesită un model al sursei și pot prelucra surse nemodelabile sau nestaționare prin folosirea unor așa-numite "dicționare" de simboluri, în continuu proces de adaptare la sursa de date pe durata transmisiei.

Codurile-prefix sunt coduri-bloc cu lungime variabilă a cuvântului de cod și au avantajul că sunt autosincronizante. Prin definiție, în cazul codurilor-prefix nici un cuvint de cod nu poate fi secvență de început (*prefix*) pentru alt cuvânt de cod.

Un **cod-prefix complet** are proprietatea că orice șir semiinfinit de simboluri de cod poate fi împărțit în mod unic în cuvinte de cod.

Se cunosc din literatura de specialitate următoarele teoreme privind codurile de tip prefix.

Teorema I (de existență): Într-un alfabet care conține K simboluri există un cod-prefix cu M cuvinte de cod cu lungimile l_m dacă și numai dacă este verificată *inegalitatea lui Kraft*:

$$\sum_{m=0}^{M-1} K^{-l_m} \le 1 \qquad \text{(II.15)}$$

Codul-prefix este complet în caz de egalitate.

Demonstrație: Fie N lungimea maximă a cuvintelor de cod. Pe nivelul al N-lea al diagramei de tip "arbore" care descrie sursa de semnal există k^N stări (ramuri). Prin alegerea unei secvențe de l simboluri drept cuvânt de cod, se elimină din diagramă k^{N-l} ramuri care derivă din starea finală corespunzătoare acestei secvențe. Numărul total al ramurilor eliminate din diagramă, prin alegerea tuturor cuvintelor de cod cu lungimile l_m impuse, nu poate depăși numărul ramurilor de pe ultimul nivel. Deci,

$$\sum_{m=0}^{M-1} K^{N-l_m} \le K^N \;\Rightarrow\; \sum_{m=0}^{M-1} K^{-l_m} \le 1 \quad \text{q.e.d.} \qquad \text{(II.16)}$$

Teorema a II-a (a lungimii medii a cuvintelor codului-prefix): Pentru o sursă discretă staționară, cu distribuția probabilistică $\bar{p}$, lungimea medie $\bar{l}$ a blocului codat la ieșirea codorului-prefix, satisface inegalitatea dublă:

$$H(\bar{p}) \le \bar{l} < H(\bar{p}) + 1 \qquad \text{(II.17)}$$

unde $H(\bar{p})$ reprezintă entropia sursei.

Demonstrație: Fără a restrânge generalitatea, vom considera o sursă binară staționară. Pentru a determina lungimea medie a cuvintelor codului-prefix optim, pe baza inegalității lui Kraft, după efectuarea a câtorva calcule algebrice simple se obține:

$$\bar{l} - H(\bar{p}) = \sum_{m=0}^{M-1} l_m p_m - \sum_{m=0}^{M-1} p_m \log_2 \frac{1}{p_m} = -\sum_{m=0}^{M-1} p_m \log_2 \frac{2^{-p_m}}{l_m} =$$

$$= -\sum_{m=0}^{M-1} p_m \log_2 e \ln \frac{2^{-p_m}}{lm} \ge -\log_2 e \sum_{m=0}^{M-1} p_m \left(\frac{2^{-p_m}}{lm} - 1 \right) = \log_2 e \left(\sum_{m=0}^{M-1} 2^{-p_m} - 1 \right) \ge 0 \quad \text{q.e.d.}$$

Limita maximă a lungimii medii a cuvintelor de cod, prevăzută de această teoremă, se obține în cazul codurilor de compactare propuse de Shannon, respectiv Fano.

Codul Huffman necesită utilizarea unor tabele de codare și de decodare construite pe baza diagramei în formă de arbore. Aceasta implică folosirea unei capacități de memorie relativ extinse, pentru aplicarea acestui cod-bloc cu lungime variabilă a cuvintelor de cod.

Spre deosebire de codurile Huffman, Shannon propune un algoritm de compactare care permite determinarea iterativă, cu un minimum de operații algebrice, a cuvintelor de cod, la cerere, fără o memorare prealabilă a lor.

Lungimea cuvântului de cod de ordin m se determină conform relației:

$$l_m = -[\log_2 p_m], \forall m = \overline{0, M-1} \qquad \text{(II.18)}$$

(*Notație:* [x] reprezintă în acest caz partea întreagă a numărului x.)

Pe baza proprietăților funcției "parte întreagă", lungimea medie a cuvintelor de cod satisface următoarele:

$$\bar{l} = -\sum_{m=0}^{M-1} p_m[\log_2 p_m] < -\sum_{m=0}^{M-1} p_m(\log_2 p_m - 1) = H(\bar{p}) + 1 \quad \text{q.e.d.}$$

În cazul în care probabilitățile elementelor sursei informaționale sunt numere de forma unor puteri negative ale lui 2, codul Shannon obținut este optim și coincide, ca set de cuvinte de cod, cu cel Huffman. Deci, în general, pentru un vector oarecare de probabilități, codul Shannon este suboptim. În mod asemănător se poate demonstra că și codul Fano este un cod suboptim.

Observație: Cuvintele de cod Shannon se obțin din dezvoltarea în binar a elementelor vectorului de probabilități cumulate, calculat după ordonarea în șir descrescător a vectorului probabilistic al sursei. Se reține primii l_m biți de după virgulă, conform relației (II.18).

Proprietățile codului de tip "prefix" optim:

❶ Dacă probabilitățile a două mesaje ale sursei sunt diferite și $p_i \geq p_k$, atunci cuvintele de cod asociate lor au lungimile: $l_i \leq l_k$ astfel încât să se respecte principiul compactării datelor.

❷ Cuvintele de cod cu aceeași lungime diferă numai pe ultimele poziții.

Teorema a III-a (Algoritmul de construcție a codului-prefix optim Huffman)

Dacă cele două simboluri ale sursei de probabilități minime se combină într-un simbol artificial și noul cod este optim pentru sursa artificială, atunci codul inițial este optim și se obține din celălalt adăugând câte un bit, 0 și 1, la sfârșitul cuvântului de cod artificial format (demonstrația acestei afirmații se găsește în [Bla90]). Operația de reunire a simbolurilor de probabilități minime se repetă pentru sursa nou-formată până la obținerea unei surse cu doar două elemente, care se codează binar: 0, respectiv 1.

Observații:

❶ Probabilitatea simbolului nou-format prin reunirea a două simboluri ale sursei se calculează ca sumă a probabilităților simbolurilor reunite.

❷ Dacă lungimea medie a cuvintelor de cod este egală cu entropia sursei de semnal, atunci codul construit este considerat *optim*.

❸ Micșorarea lungimii medii a cuvintelor de cod se poate face prin codarea pe blocuri de simboluri. Probabilitatea unui simbol compus se va calcula, în cazul surselor fără memorie, ca produs al probabilităților elementelor componente .

❹ În cazul compactării textelor (fără elemente grafice) este mai utilă structura pe cuvinte și nu pe litere. Se construiesc dicționare de cuvinte uzuale (de exemplu, de circa 4000 de cuvinte în cazul folosirii limbii engleze) la care se adaugă toate literele (pentru codarea cuvintelor neincluse în dicționar), cifrele și semnele de punctuație. Se obțin astfel coduri-bloc cu lungime variabilă de dimensiuni foarte mari dar cu performanțe superioare codării pe structura de caractere alfanumerice. Un avantaj al acestei metode îl constituie faptul că rata de compactare nu depinde de mărimea sau de tipul caracterelor folosite pentru redactare.

❺ Pentru codarea Huffman a surselor de dimensiuni mari, este necesară construirea unei diagrame de tip arbore cu număr foarte mare de stări. Se preferă folosirea în acest caz a codurilor Shannon-Fano care permit implementarea sub formă de algoritmi cu procesoare digitale de semnal.

❻ Spre deosebire de codurile-prefix Huffman care utilizează modele prestabilite ale sursei de informație, codurile universale de compactare (Ziv-Lempel) permit codarea surselor nemodelate sau nestaționare, fie prin construirea unui model al sursei adaptabil în timp din chiar semnalul transmis, fie prin realizarea unor *dicționare* de simboluri cu probabilități mari de apariție, de asemenea readaptabile pe durata transmisiei la semnalul transmis.

II.3 Aplicații

P1. Să se construiască și să se analizeze codul Huffman pentru o sursă informațională discretă, fără memorie, care generează 5 simboluri (A; B; C; D; E) cu vec-

torul de probabilități **p** = (1/2; 1/4; 1/8; 1/16; 1/16).

***Soluție*:** Se construiește diagrama 'arbore' pe setul de simboluri, în ordine descrescătoare a probabilităților de apariție. Se aplică algoritmul de construcție a codului Huffman (fig.II.3).

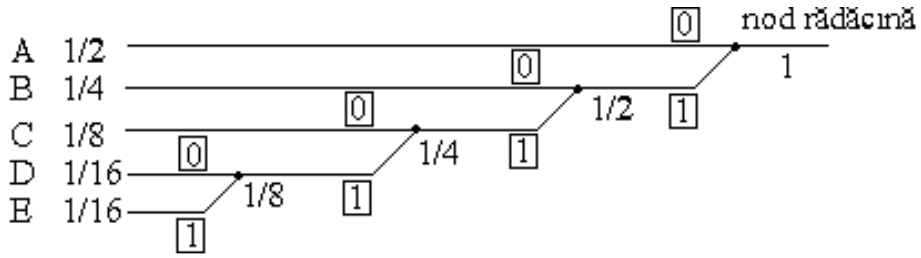

Fig.II.3 Diagrama-arbore a codului Huffman construit pentru sursa cu 5 simboluri

Cuvintele de cod asociate simbolurilor sursei se citesc în diagramă, de la dreapta la stânga, plecând din nodul rădăcină. Codul Huffman obținut este descris în Tabelul II.1 (cuvintele de cod și lungimile acestora).

Tabelul II.1

Cod Huffman pentru SDCFM cu 5 simboluri

Simbol	Probabilitate de apariție	Cuvânt de cod	Lungimea cuvântului de cod (biți)
A	1/2	0	1
B	1/4	10	2
C	1/8	110	3
D	1/16	1110	4
E	1/16	1111	4

Entropia sursei:

$$H(\bar{p}) = -\sum_i p_i \log_2 p_i = \frac{1}{2}\cdot 1 + \frac{1}{4}\cdot 2 + \frac{1}{8}\cdot 3 + \frac{1}{16}\cdot 4 \cdot 2 = \frac{15}{8} = 1.875 biti/simbol$$

Lungimea medie a cuvintelor de cod:

$$\bar{l} = \sum_i p_i l_i = \frac{1}{2}\cdot 1 + \frac{1}{4}\cdot 2 + \frac{1}{8}\cdot 3 + \frac{1}{16}\cdot 4 \cdot 2 = \frac{15}{8} = 1,875 biti/simbol = H(\bar{p})$$

Codul Huffman construit este optim, cu eficiență de 100% și redundanță nulă a semnalului codat. Dacă simbolurile sursei ar fi fost codate prin secvențe binare cu lungime fixă (3 biți/simbol: 000; 001; 010; 011; 100) ar fi rezultat o valoare a redundan-

ței semnalului codat de: $R = \frac{3-1,875}{1,875} = 60\%$.

P2. a) Construiți și analizați codul Huffman pe un simbol pentru o SDCFM cu 3 caractere (A; B; C) caracterizată de vectorul probabilistic (3/4; 3/16; 1/16).

b) Studiați eficiența codului Huffman proiectat pe structura de caractere compuse din 2 simboluri ale sursei.

c) Pentru a nu putea fi dedusă natura sursei informaționale, se poate determina un nou set de simboluri ale sursei, obținute prin descompunerea fiecărei valori de probabilitate în sumă de puteri negative ale lui 2. Reproiectați codul Huffman pentru sursa dată, în acest caz. Ce observați?

Soluție: a) Se reprezintă diagrama-arbore a codului Huffman:

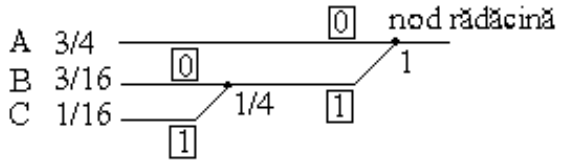

Fig.II.4 Diagrama-arbore a codului Huffman construit pentru sursa cu 3 simboluri

Se calculează entropia sursei, lungimea medie a cuvintelor de cod și eficiența codului:

$H(\bar{p}) = 1,012 biti/simbol; \bar{l} = 1,25 biti/simbol; \eta = 76,5\%$

b) Creșterea eficienței codului este posibilă prin utilizarea de caractere compuse. Pentru simplitatea calculului se lucrează cu ponderile caracterelor obținute prin multiplicare cu 256 a valorilor probabilităților. Diagrama-arbore este reprezentată în figura II.5. Codul obținut este descris în tabelul II.2.

Lungimea medie a cuintelor de cod este:

$\bar{l}_2 = 2,07 biti/caracter compus = 1,035 biti/simbol$

ceea ce corespunde unei eficiențe de codare de 97,73%.

c) Descompunem simbolurile astfel încât să obținem doar valori de probabilitate de forma unor puteri negative ale lui 2: (A_1;A_2; B_1; B_2; C) --> (1/2; 1/4; 1/8;1/16; 1/16).

Construiți codul Huffman folosind noul set de simboluri.

Se observă creșterea lungimii medii a cuvintelor de cod. Optimizarea codului este posibilă prin considerarea caracterelor compuse din minimum 2 simboluri.

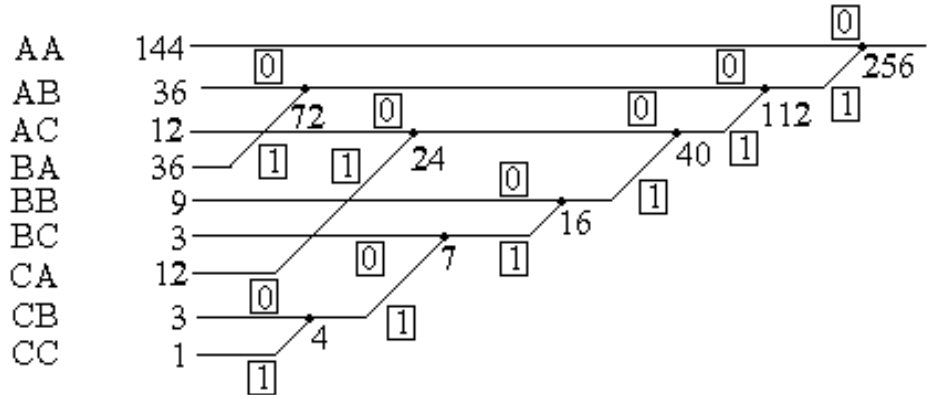

Fig.II.5 Diagrama codului Huffman construit pe simboluri compuse

Tabelul II.2

Codul Huffman 2 construit pe setul de caractere compuse

Simbol	Probabilitate de apariție	Cuvânt de cod	Lungimea cuvântului de cod (biți)
AA	144/256	0	1
AB	36/256	100	3
AC	12/256	1100	4
BA	36/256	101	3
BB	9/256	1110	4
BC	3/256	11110	5
CA	12/256	1101	4
CB	3/256	111110	6
CC	1/256	111111	6

P3. Proiectați și analizați codul Huffman pentru o SDCFM cu 7 simboluri și setul probabilistic:(3/8; 3/16; 3/16; 1/8; 1/16; 1/32; 1/32).

P4. Construiți și analizați codul Huffman pe structura de caractere compuse din 2 simboluri pentru o SDCFM cu 4 simboluri A;B;C;D cu probabilitățile:1/2; 1/4; 1/8; 1/8.

P5. O sursă generează două simboluri A și B cu probabilitățile 5/8 și 3/8. Construiți și analizați codul Huffman pe subsimboluri având probabilitățile ca puteri negative ale lui 2. Reluați algoritmul lucrând cu caractere compuse din 2 subsimboluri.