

Metoda HUFFMAN neadaptiva de compresie/decompresie

- Fazele de compresie si decompresie trec prin aceleasi etape ca in cazul alg. Shannon-Fano.
- Constructia arborelui pleaca de la frunze spre radacina => codurile simbolilor sunt construite de la bitul cel mai putin semnificativ la bitul cel mai semnificativ.

Pas 1: Se parcurge secvential D si se construiesc $A^0 \Rightarrow N(s)$ pentru fiecare $s \in A^0$

Pas 2: Rearanjarea simbolilor alfabetului: $A^0 = \{s_1, s_2, \dots, s_N\}$ cu $N(s_1) \geq N(s_2) \geq \dots \geq N(s_N)$

Se reoteaza si reindexeaza elementele alfabetului astfel: $A^0 = \{t_{N+k-1} = s_k\}_{k \in \overline{1, N}}$

Simbolii $t_N, t_{N+1}, \dots, t_{2N-1}$ sunt utilizati pentru etichetarea frunzelor arborelui binar, iar alfabetul A^0 devine *primul alfabet curent* al algoritmului si va fi renotat prin $A^{0,N}$. Fiecare alfabet curent al algoritmului se va nota prin $A^{0,(N-k)}$, $k=0 \dots N-1$

$k=0 \Rightarrow$ frunzele arborelui sunt etichetate cu simbolii primului alfabet curent

Pas 3: Constructia arborelui binar se realizeaza simultan cu formarea noului alfabet curent.

Noul alfabet curent, $A^{0,(N-k-1)}$ cu $k=0 \dots N-2$, se construiesc din vechiul alfabet $A^{0,(N-k)}$, aplicandu-se urmatoarea regula: se elimina 2 dintre simbolii cu cele mai mici contoare din vechiul alfabet curent, inlocuindu-le cu un simbol virtual, notat prin „ \square ” si care are contorul egal cu suma contoarelor celor doi simbolii. Daca exista mai mult de doi simbolii slabi avand acelasi contor, se aleg ultimii doi din alfabet => cardinalul noului alfabet scade cu o unitate fata de cardinalul alfabetului precedent.

$$A^{0,(N-k-1)} = [A^{0,(N-k)} \setminus \{t_p, t_q\}] \cup \{t_{N-k-1}\}$$

$$t_{N-k-1} = \text{simbol virtual}$$

$$N(t_{N-k-1}) = N(t_p) + N(t_q)$$

$$\# A^{0,(N-k-1)} = \# A^{0,(N-k)} - 1 = N - k - 1$$

- Simbolul virtual eticheteaza un nod intermediar al arborelui, avand ca descendenti cele doua noduri etichetate de simbolii eliminati.
- Fiecare nod intermediar din arbore are cate 2 descendenti, contorul fiului din stanga fiind cel putin egal cu cel al fiului din dreapta.

Pas 4: Se repeta Pas 3 de $N-1$ ori pentru fiecare alfabet curent $A^{0,(N-k)}$ pana cand se eticheteaza si radacina arborelui (t_1).

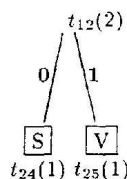
Pas 5: Se reindexeaza nodurile arborelui astfel incat fiii oricarui nod sa fie etichetati cu indici consecutivi, de la stanga la dreapta, incepand de la radacina.

Constructia arborelui binar

Setul de date D : **IT IS BETTER LATER THAN NEVER.** => $N=13$ elem., nr. noduri=25, $k=12$

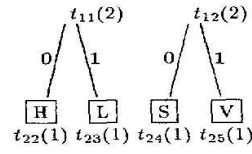
Nod	t_{13}	t_{14}	t_{15}	t_{16}	t_{17}	t_{18}	t_{19}	t_{20}	t_{21}	t_{22}	t_{23}	t_{24}	t_{25}
A^0		E	T	R	A	I	N	.	B	H	L	S	V
$N(s)$	5	5	5	3	2	2	2	1	1	1	1	1	1

$$k = 1$$



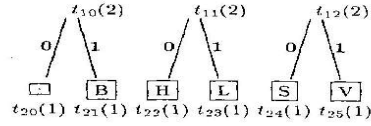
Nod	t_{13}	t_{14}	t_{15}	t_{16}	t_{17}	t_{18}	t_{19}	t_{20}	t_{21}	t_{22}	t_{23}	t_{12}
$A^{0,12}$	\square	E	T	R	A	I	N	.	B	H	L	\square
$N(s)$	5	5	5	3	2	2	2	1	1	1	1	2

$$k = 2$$



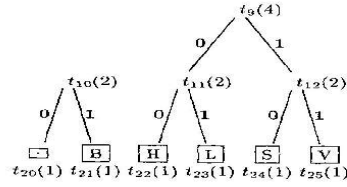
Nod	t_{13}	t_{14}	t_{15}	t_{16}	t_{17}	t_{18}	t_{19}	t_{20}	t_{21}	t_{11}	t_{12}
$\mathcal{A}^{0,11}$	\sqcup	E	T	R	A	I	N		B	\square	\square
$\mathcal{N}(s)$	5	5	5	3	2	2	2	1	1	2	2

$$k = 3$$



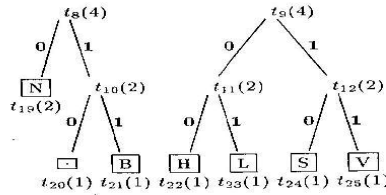
Nod	t_{13}	t_{14}	t_{15}	t_{16}	t_{17}	t_{18}	t_{19}	t_{10}	t_{11}	t_{12}
$\mathcal{A}^{0,10}$	\sqcup	E	T	R	A	I	N	\square	\square	\square
$\mathcal{N}(s)$	5	5	5	3	2	2	2	2	2	2

$$k = 4$$



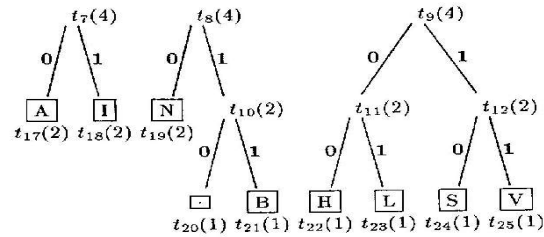
Nod	t_{13}	t_{14}	t_{15}	t_{16}	t_{17}	t_{18}	t_{19}	t_{10}	t_9
$\mathcal{A}^{0,9}$	\sqcup	E	T	R	A	I	N	\square	\square
$\mathcal{N}(s)$	5	5	5	3	2	2	2	2	4

$$k = 5$$



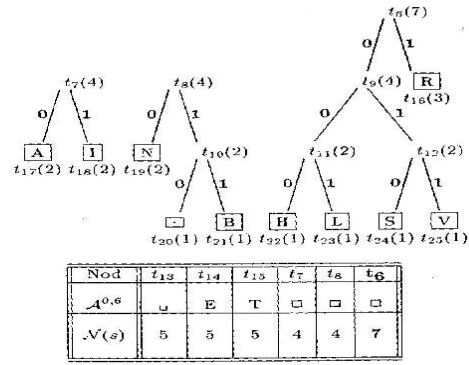
Nod	t_{13}	t_{14}	t_{15}	t_{16}	t_{17}	t_{18}	t_8	t_9
$\mathcal{A}^{0,8}$	\sqcup	E	T	R	A	I	\square	\square
$\mathcal{N}(s)$	5	5	5	3	2	2	4	4

$$k = 6$$

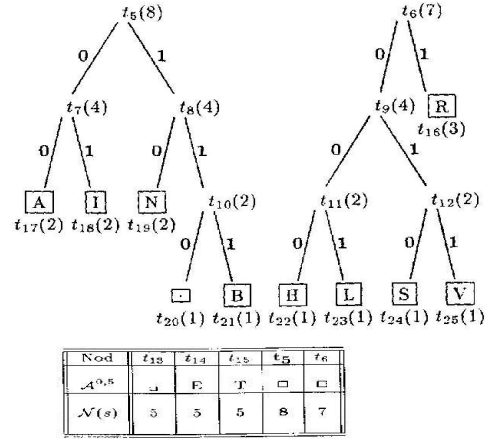


Nod	t_{13}	t_{14}	t_{15}	t_{16}	t_7	t_8	t_9
$\mathcal{A}^{0,7}$	\sqcup	E	T	R	\square	\square	\square
$\mathcal{N}(s)$	5	5	5	3	4	4	4

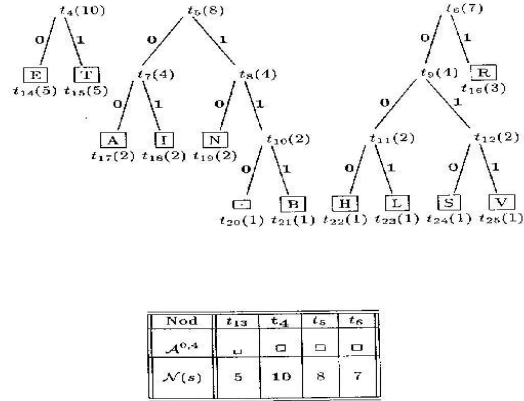
$$k = 7$$



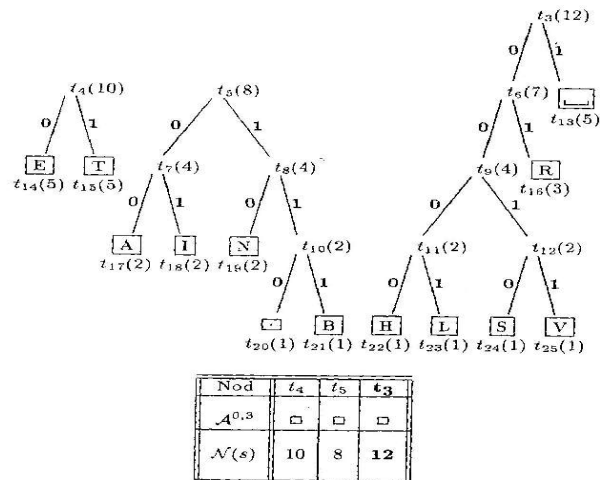
$$k = 8$$



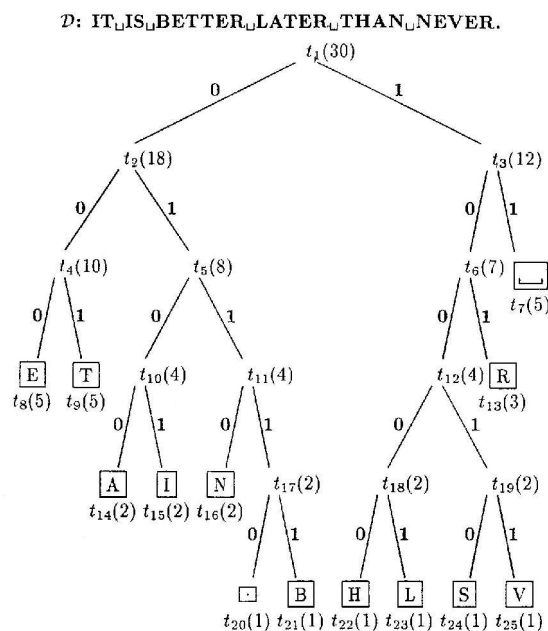
$$k = 9$$



$$k = 10$$



Structura setului de date comprimate



Un arbore binar de tip HUFFMAN (alfabet static de ordin 0).

Informatia obtinuta pe fluxul de iesire este:

- Informatia auxiliara (identica cu cea de la alg. Shannon-Fano)
- Informatia utila:

Cod	0101	001	11	0101	10010	11	01111	000	001	001
Info	I	T		I	S		B	E	T	T
Nr. biți	4	3	2	4	5	2	5	3	3	3

Cod	000	101	11	10001	0100	001	000	101	11	001
Info	E	R		L	A	T	E	R		T
Nr. biți	3	3	2	5	4	3	3	3	2	3

Cod	10000	0100	0110	11	0110	000	10011	000	101	01110
Info	H	A	N		N	E	V	E	R	.
Nr. biți	5	4	4	2	4	3	5	3	3	5

Obs:

- Noile coduri ale simbolilor nu depasesc 5 biti in lungime
- Codurile simbolilor difera de cele obtinute prin metoda Shannon-Fano
- Analiza performantelor este similara cu cea de la algoritmul Shannon-Fano
- Pentru seturi mai lungi de date, metoda Huffman ofera o compresie mai mare decat metoda Shannon-Fano