TESTAREA EXPERIMENTALĂ A UNOR ALGORITMI DE COMPRESIE A DATELOR

drd. ing. Cristian-Valentin Eremia cristian_valentin2003@yahoo.com

prof. dr. ing. Mihai Tertişco mihai_tertisco@yahoo.com

Universitatea Politehnica Bucureşti

Rezumat: Lucrarea prezintă rezultatele testării experimentale a 5 algoritmi de compresie a datelor (Shannon-Fano, LZW, Huffman Standard, Huffman Dinamic, Compresie Aritmetică). Testarea a avut scop evidenţierea performanţelor acestor algoritmi de compresie în funcţie de caracterul datelor (text, imagine, sunet, executabil, librării dinamice, C++, Pascal).

Cuvinte cheie: Shannon-Fano, LZW, Huffman Standard, Huffman Dinamic, Compresie Aritmetica

Abstract: The paper presents results of experimental testing of 5 data compression algorithms (Shannon-Fano, LZW, Standard Huffman, Dynamic Huffman, Arithmetic Compression). Testing was intended to highlight the performance of compression algorithms depending on the nature of the data (text, image, sound, executable, dynamic libraries, C++, Pascal).

Keywords: Shannon-Fano, LZW, Standard Huffman, Dynamic Huffman, Arithmetic Compression

1. Introducere

Scopul compresiei de date este acela de a îmbunătăţi performanţele operaţiilor ulterioare efectuate asupra acestor date: stocare, transmisie etc. Toate sistemele de compresie necesită doi algoritmi: unul pentru comprimarea datelor la sursă şi altul pentru decomprimarea datelor la destinatar. Prin compresie de date se urmăreşte ca, folosind anumite procedee, să se realizeze o trecere (transformare) a unui fişier F, de lungime L biţi, într-un fişier Fc, de lungime $Lc<L$. Deci decompresia transformă un fişier comprimat Fc într-unul identic cu cel original F sau într-unul apropiat de F, din punct de vedere al conţinutului. Algoritmii de compresie şi respectiv decompresie prezintă unele asimetrii privind cerinţele impuse performanţelor acestora în funcţie de domeniul de aplicaţie al datelor din fişiere. Spre exemplu, în multe aplicaţii, un document multimedia va fi codificat o singură dată (la sursă) şi decodificat de mii de ori (la destinaţie). Deci, această asimetrie oferă posibilitatea ca algoritmul de codificare să fie lent, în timp ce algoritmul pentru decodificare trebuie să fie rapid. În multe sisteme de compresie s-au făcut eforturi pentru ca decodificarea să se facă cât mai simplu şi să fie cât mai rapidă chiar şi cu preţul încetinirii şi complicării codificării. Acest lucru nu este valabil şi pentru aplicaţiile în timp real (spre exemplu la o video conferinţă). În timp real codificarea se face cu algoritmi diferiţi de cei folosiţi în cazul sistemelor de memorare (stocare) a unor fişiere cu date video, caz în care algoritmii pot fi mai lenţi decât în sistemul de timp real [1].

O altă asimetrie constă în faptul că procesul codificare/decodificare nu trebuie totdeauna să fie strict inversabil. Adică, atunci când se comprimă un fişier – program sau fişier text, dorim ca după decomprimare să-l obţinem exact pe cel original până la ultimul bit. În cazul fişierelor multimedia această cerinţă nu se pune. De obicei se acceptă să avem după decodificare un semnal puţin diferit de original. Atunci când ieşirea decodificată este identică cu intrarea originală sistemul este cu pierderi. Sistemele cu pierderi sunt importante deoarece acceptarea unui număr mic de informaţii pierdute poate oferi un avantaj imens în termenii de rată de comprimare. Indiferent dacă fişierele conţin texte, imagini, sau reprezentări ale sunetelor, la baza lor stă un alfabet.

Un alfabet se defineşte ca fiind totalitatea simbolurilor elementare folosite într-un fişier. Alfabetul se caracterizează prin lungime exprimată ca număr de simboluri.

Un cuvânt este o succesiune de simboluri elementare din alfabet grupate astfel încât să aibă o anumită semnificaţie. Cuvântul se caracterizează prin: **număr de simboluri; delimitator de început şi delimitator de sfârşit; elementul corespondent dintr-o mulţime.**

Un limbaj este alcătuit din mulţimea de cuvinte şi regulile de folosire a acestora.

Compresia datelor este folosită şi în cadrul sistemelor de stocare ori de transmisie la distanţă, a datelor de tip text sau a datelor de tip sunet ori de tip imagine reprezentate în format digital. Explozia dimensională a fişierelor de date are loc în primul rând în cazul sistemelor multimedia care comportă transmisia şi prelucrarea unui volum mare de informaţii în reţelele de calculatoare.

Toate sistemele de compresie conţin un codor şi un decodor. Codorul realizează conversia datelor provenite de la sursă, în date comprimate, iar decodorul încearcă, la recepţie, să reconstituiască datele iniţiale pe baza datelor comprimate. Datele reconstituite prin decodare fie coincid cu datele emise de sursă, fie diferă într-o măsură nesemnificativă pentru scopul propus. Această diferenţă între lungimea în biţi a fişierului compresat şi a celui necompresat trebuie evaluată şi sistemul de compresie se ajustează funcţie de scopul propus. Pentru a evalua calităţile unui sistem de compresie se defineşte o mărime numită raport de compresie:

$$\frac{\text{Dimensiunea datelor emise de sursă în biţi}}{\text{Dimensiunea datelor comprimate în biţi}} = r \qquad (1.1)$$

Un raport de compresie "2 la 1", semnifică faptul că datele comprimate au jumătate din dimensiunea datelor emise de sursă. Un raport de compresie mare va desemna un sistem de compresie mai bun.

Procedurile de compresie de date se utilizează de mai multă vreme, fiind la început legate de stocarea pe un suport magnetic a textelor redactate în format electronic. Chiar şi în acest domeniu evoluţia a fost foarte spectaculoasă, performanţele algoritmului de compresie îmbunătăţindu-se în ritm accelerat. Dar aceste tehnici care permit prin operaţia inversă de decompresie, refacerea exactă (fără pierderi) a ansamblului original oferă un grad redus de compresie, deoarece se bazează doar pe eliminarea redundanţei naturale a sursei de informaţie, care chiar dacă există, are o valoare limitată. Rapoartele mult mai mari de compresie se pot obţine dacă se renunţă la refacerea exactă a semnalului informaţional, dar atunci trebuie acceptat un compromis între factorul de compresie şi precizia cu care se realizează refacerea.

Scopul acestei lucrări este să se realizeze:

- Studiul teoriei frecvenţiale Shannon de compresie a fişierelor de tip TXT;
- Implementarea unui sistem ilustrativ bazat pe algoritmul Shannon–Fano, Huffman etc. de compresie;
- Testarea experimentală a principalelor metode de compresie a fişierelor de date şi evaluarea comportamentului acestora în cazul diverselor tipuri de fişiere: **TXT, EXE, C++, Pascal, sunet, video**. Evaluarea metodelor se face pe baza ratei de compresie.

1.1. Teorema lui Shannon privind fişierul compresat

Conceptele de entropie, cantitate de informaţii şi redundanţă au fost definite la începutul anilor '40, adică odată cu primele noţiuni de Teoria Informaţiei. Ideea de bază care a declanşat numeroase cercetări în domeniul Compresiei Datelor a fost următoarea: *dacă s-ar cunoaşte probabilitatea ce apare într-un set de date, atunci aceste simboluri ar putea fi recodificate în aşa fel încât lungimea totală a noilor coduri să fie inferioară lungimii originale a setului de date.* În termeni de compresie, acest efect se numeşte *minimizarea redundanţei setului de date*.

Redundanţa empirică se evaluează prin diferenţa între numărul de biţi iniţial ***No*** şi numărul de biţi din fişierul compresat ***Nc.***

ENTROPIA unui simbol exprimă cantitatea de "nedeterminare" pe care o elimină "prezenţa" simbolului respectiv în fişier. Spre exemplu, dacă din alfabet este "şters" unul din simbolurile de frecvenţă maximă, practic nu se mai înţelege ce vrea să exprime conţinutul informaţional al fişierului. Deci, cantitatea de informaţie conţinută în simbol, respectiv aportul informaţional al simbolului la cantitatea de informaţie a întregului fişier, este mare, dacă

simbolul are frecvenţa mare şi deci ***entropia aceluiaşi simbol este mică.***

Relaţia de definiţie, propusă de SHANNON pentru ENTROPIA ***H(sk)*** a unui simbol ***sk*** din alfabet este[2]:

$$H(sk)=-f(sk)\log_2 f(sk), \qquad (1.2)$$

în care ***log2*** este logaritmul în baza 2.

De remarcat că, întrucât frecvenţele sunt aceleaşi atât pentru simbolurile din fişierul compresat cât şi necompresat, entropiile simbolurilor sunt aceleaşi dar lungimile codurilor compresate sunt diferite în fişierul compresat şi necompresat.

1.2. Teorema Shannon privind lungimea şi entropia fişierului compresat:

Observaţie:

Nici o metodă de compresie nu poate furniza o lungime Lc, a fişierului compresat, mai mică decât H – entropia fişierului.

Compresia de date este operaţia prin care se reprezintă compact datele furnizate de o anumită sursă. În funcţie de sursa care le generează, forma şi tipul de date variază:

- text,
- imagine,
- semnal vocal,
- semnal video.

Când ne referim la multimedia ne referim la combinarea dintre două sau mai multe media continue, adică media care trebuie să se desfăşoare într-un interval bine definit, de obicei folosind interacţiunea cu utilizatorul. În practică cele două media sunt ***audio*** şi ***video***. O undă audio este o undă cu specific acustic.

1.3. Standardul *JPEG* pentru compresia imaginilor în tonuri continue

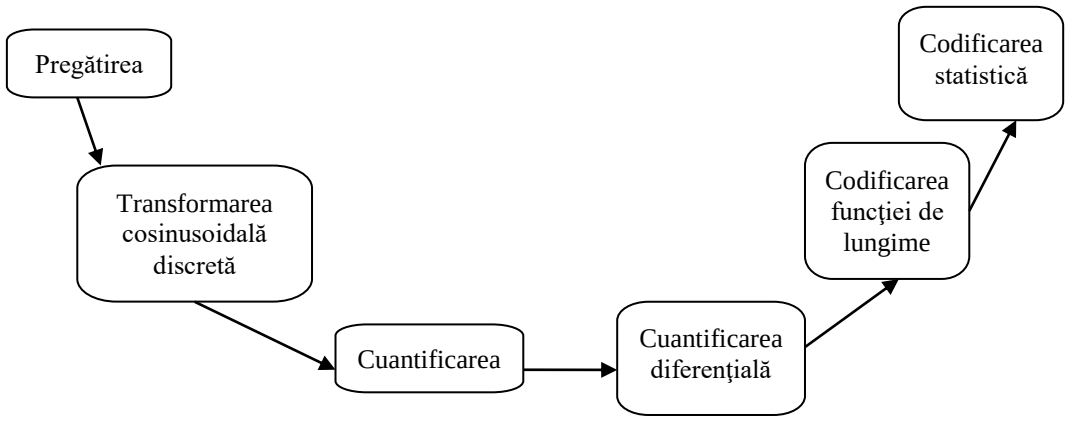

Figura 1. Structura şi conţinutul procedurii, în 6 paşi de codificare a imaginii RGB cu JPEG

Standardul JPEG (Joint Photographic Experts Group) pentru comprimarea imaginilor în tonuri continue, a fost dezvoltat de experţii în fotografii. Este important pentru multimedia deoarece la o primă aproximare, standardul multimedia pentru filme, MPEG este codificarea JPEG a fiecărui cadru separat, plus câteva caracteristici pentru comprimarea între cadre şi detectarea mişcării. JPEG este definit în Standardul Internaţional 10918. JPEG are patru moduri şi multe opţiuni, fiind folosit în mod normal pentru codificarea imaginilor video de 24 biţi RGB. Ne vom ocupa mai mult de modul secvenţial cu pierderi. În figura 1.1 este prezentată structura şi conţinutul procedurii, în 6 paşi de codificare, cu JPEG, a unei imagini **RGB.**

1.4. Standard *MPEG* (Motion Picture Experts Group)

Algoritmii principali din familia MPEG sunt destinaţi pentru compresia video şi sunt standarde internaţionale din 1993. Deoarece filmele conţin atât imagini cât şi sunete, MPEG le poate comprima pe amândouă. Primul standard finalizat a fost MPEG-1. Scopul lui a fost de a produce ieşiri video de calitatea video-recorder-elor (352x240) folosind o rată de biţi de l,2 Mbps. Video necomprimat poate ajunge la 472 Mbps, reducerea lui la 1,2 Mbps este însemnată, chiar şi la această rezoluţie scăzută. MPEG-1 poate fi transmis pe linii torsadate la distante modeste.Următorul standard din familia MPEG a fost MPEG-2 care a fost proiectat iniţial pentru comprimarea video-ului de calitate de difuzare între 4 şi 6 Mbps. Mai târziu MPEG a fost extins pentru a suporta rezoluţii înalte. MPEG-4 este pentru video-conferinţe de rezoluţie medie cu cadre de viteză scăzută, (10 cadre/sec) şi la lărgimi scăzute de banda (64 Kbps). Principiile de bază ale lui MPEG-1 şi MPEG-2 sunt similare dar detaliile sunt diferite.

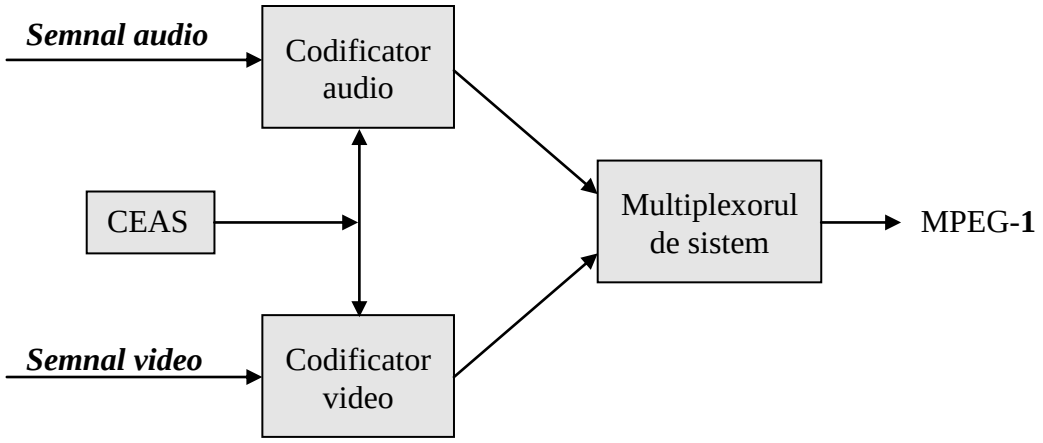

Figura 2. Structura standardului MPEG-1

Compresia audio MPEG este făcută prin eşantionarea formei de undă la 33 kHz, 44,1 kHz sau 48 kHz. Ea poate să gestioneze mono, stereo disjunct (fiecare canal separat) sau stereo reunit (este exploatată redundanţa între canale). Este organizată pe trei nivele, fiecare dintre ele aplicând optimizări suplimentare pentru a obţine o compresie mai mare. Nivelul 1 este schema de bază. Acest nivel este folosit, de exemplu, în sistemul de bandă DCC. Nivelul 2 adaugă schemei de bază alocarea avansată de biţi. Este folosit pentru CD-ROM-uri audio şi de pistele sonore ale filmelor. Nivelul 3 adaugă filtre hibride, cuantificare neuniformă, codificare Huffman şi alte tehnici avansate.

Audio MPEG poate comprima un CD de muzică până la 96Kbps fără o pierdere perceptibilă a calităţii. Numărul diferă de la o muzică la alta pentru că rata de semnal-zgomot diferă. Comprimarea audio este realizată prin transformata Fourier rapidă a semnalului audio pentru a-l converti din domeniul timpului în cel al frecvenţei. Spectrul realizat este divizat în 32 benzi de frecvenţă fiecare procesată separat. Atunci când există două canale stereo, redundanţa inerentă în a avea două surse audio suprapuse este de asemenea exploatată. Fluxul audio MPEG-1 rezultat este ajustabil de la 32Kbps la 448Kbps.

Obiectivul principal al cercetării în faza definitivării tezei de doctorat este testarea unor algoritmi de compresie pe diverse tipuri de fişiere de date cu scopul de a verifica experimental pentru care tipuri de fişiere se pretează, cel mai bine, diverşii algoritmi pentru compresii de date. Rezultatele experimentale ale acestei testări sunt conţinute de acest capitol.

2. Testarea experimentală

În ceea ce priveşte programele de implementare a diverşilor algoritmi testaţi, le-am folosit pe cele care se găsesc prezentate în lucrarea recentă, intitulată „***Compresia datelor***”, publicată în anul 2003, de către domnul profesor universitar Dr. ing. Dan STEFĂNOIU.

Există o diversitate foarte mare de algoritmi de compresie şi variaţii ale acestora. Vom face o analiză comparată pe mai multe tipuri de fişiere, de dimensiuni diferite şi cu diferiţi algoritmi.

În analiză vom folosi următoarele 8 tipuri de fişiere: **dll**: biblioteci dinamice; **pas:** fişiere Pascal; **txt**: fişiere text; **exe:** fişiere executabile; **wav:** fişiere sunet; **bmp:** fişiere BitMap; **cpp**: fişiere C++; **img**: fişiere imagine.

Analiza o vom face folosind următorii algoritmi:

- A: compresia Huffman standard;
- B: compresia Shannon-Fano;
- C: compresie Huffman dinamic;
- D: compresie aritmetică de ordin n;
- E: compresie LZW

2.1. Calculul gradului de compresie

Relaţia de definiţie a gradului de compresie g, utilizată pentru prelucrarea datelor experimentale este următoarea:

$$g = \left(1 - \frac{L'}{L}\right) * 100 \qquad (1.3)$$

Unde:

- L' - lungimea fişierului compresat printr-un algoritm de compresie din mulţimea celor selectate pentru testare;
- L - lungimea fişierului iniţial necompresat.

2.2. Rezultatele testării celor 5 algoritmi pe 10 diverse tipuri de fişiere de lungimi diferite

Tabelul 1 - Rezultate pentru fişiere de tip dll (librării dinamice):

Fişier iniţial (dim în bytes)	Huffman Standard	Shannon – Fano	Huffman Dinamic	Compresie Aritmetică	LZW
37440	33%	31%	34%	26%	49%
119056	26%	18%	19%	31%	52%
161552	33%	18%	19%	31%	48%
59904	32%	22%	23%	17%	54%
109424	30%	16%	17%	14%	47%
11776	31%	29%	34%	26%	39%
217088	43%	17%	18%	20%	50%
470528	27%	21%	16%	34%	51%
469504	33%	22%	16%	24%	52%
5632	33%	28%	38%	19%	49%
Media	33%	23%	24%	25%	50%

Tabelul 2 - Rezultate pentru fişiere de tip PAS (surse Pascal)

Fişier iniţial (dim în bytes)	Huffman Standard	Shannon – Fano	Huffman Dinamic	Compresie Aritmetică	LZW
25840	38%	37%	38%	38%	55%
49249	43%	43%	44%	44%	59%
18845	41%	41%	42%	42%	61%
27712	35%	35%	36%	36%	57%
33336	36%	36%	37%	37%	55%
30238	37%	37%	38%	38%	58%
10901	35%	34%	36%	36%	53%
7494	33%	33%	35%	35%	50%
16738	36%	36%	38%	37%	55%
9432	33%	32%	34%	34%	51%
Media	37%	36%	38%	38%	56%

Tabelul 3 - Rezultate pentru fişiere de tip TXT(fişiere text)

Fişier iniţial (dim în bytes)	Huffman Standard	Shannon – Fano	Huffman Dinamic	Compresie Aritmetică	LZW
19411	37%	36%	38%	38%	53%
36232	38%	37%	38%	39%	50%
42862	41%	41%	42%	42%	53%
4549	35%	35%	38%	39%	55%
36459	41%	41%	42%	42%	57%
20309	38%	38%	39%	42%	53%
23294	37%	37%	38%	38%	54%
10884	40%	40%	42%	42%	54%
60646	35%	37%	38%	38%	56%
90108	7%	34%	35%	35%	35%
Media	35%	37%	40%	41%	54%

Tabelul 4 - Rezultate pentru fişiere de tip EXE (executabile)

Fişier iniţial (dim în bytes)	Huffman Standard	Shannon – Fano	Huffman Dinamic	Compresie Aritmetică	LZW
74192	37%	25%	29%	26%	22%
39280	30%	30%	31%	31%	29%
19456	29%	27%	32%	31%	29%
2708	9%	3%	19%	17%	13%
10774	9%	9%	14%	14%	11%
44032	25%	25%	26%	26%	22%
15656	17%	15%	20%	20%	*16%*
77312	49%	33%	36%	34%	33%
111376	36%	23%	25%	24%	19%
19729	17%	15%	19%	19%	16%
Media	26%	20%	25%	25%	22%

Tabelul 5 - Rezultate pentru fişiere de tip WAV (sunet)

Fişier iniţial (dim în bytes)	Huffman Standard	Shannon – Fano	Huffman Dinamic	Compresie Aritmetică	LZW
12479	37%	15%	26%	26%	18%
55490	17%	6%	50%	45%	51%
1226	11%	18%	47%	30%	49%
7754	16%	16%	46%	50%	50%
4296	14%	12%	42%	42%	74%
12106	20%	13%	53%	22%	39%
25704	20%	20%	51%	51%	9%
22570	12%	12%	49%	14%	54%
33848	24%	2%	54%	51%	48%
28282	25%	12%	60%	47%	65%
Media	19%	14%	48%	38%	48%

Tabelul 6 - Rezultate pentru fişiere de tip bmp (BitMap)

Fişier iniţial (dim în bytes)	Huffman Standard	Shannon – Fano	Huffman Dinamic	Compresie Aritmetică	LZW
2118	75%	75%	77%	79%	74%
582	56%	55%	61%	58%	57%
470	46%	46%	58%	52%	35%
2102	31%	30%	39%	35%	37%
578	60%	60%	67%	62%	67%
38462	20%	20%	22%	22%	22%
590	50%	50%	59%	55%	55%
339178	66%	54%	59%	54%	64%
578	53%	53%	59%	56%	33%
198	29%	29%	47%	34%	27%
Media	49%	47%	55%	51%	52%

Tabelul 7 - Rezultate pentru fişierele de tip CPP (surse C++)

Fişier iniţial (dim în bytes)	Huffman Standard	Shannon – Fano	Huffman Dinamic	Compresie Aritmetică	LZW
15375	37%	37%	38%	38%	55%
40382	34%	34%	34%	35%	51%
23162	32%	32%	33%	33%	51%
478	14%	14%	34%	26%	25%
12501	32%	32%	33%	34%	47%
1354	25%	25%	36%	34%	35%
990	24%	24%	37%	34%	34%
35746	44%	43%	44%	45%	62%
405	16%	15%	40%	32%	21%
12234	35%	34%	36%	36%	59%
Media	30%	29%	37%	35%	44%

Tabelul 8 - Rezultate pentru fişiere de tip IMG (imagine)

Fişier iniţial (dim în bytes)	Huffman Standard	Shannon – Fano	Huffman Dinamic	Compresie Aritmetică	LZW
40410	26%	26%	27%	27%	27%
54696	29%	29%	31%	30%	65%
33822	34%	34%	35%	35%	38%
40028	17%	17%	19%	29%	51%
16900	26%	26%	29%	49%	48%
35640	18%	17%	19%	39%	36%
33622	17%	16%	18%	38%	41%
33230	19%	18%	20%	40%	44%
20894	19%	19%	21%	32%	70%
31652	18%	18%	20%	40%	39%
Media	23%	22%	25%	33%	46%

2.3. Rezultate ale analizei comparative a testului

În cadrul etapei de analiză comparativă a rezultatelor experimentale ale testului s-au realizat reprezentari grafice ,prin bargrafuri, pentru toate rezultatele comparaţiei ratelor medii calculate de compresie. Aceste reprezentari grafice ale rezultatelor analizei comparative si concluziile desprinse din acestea sunt prezentate in continuare.

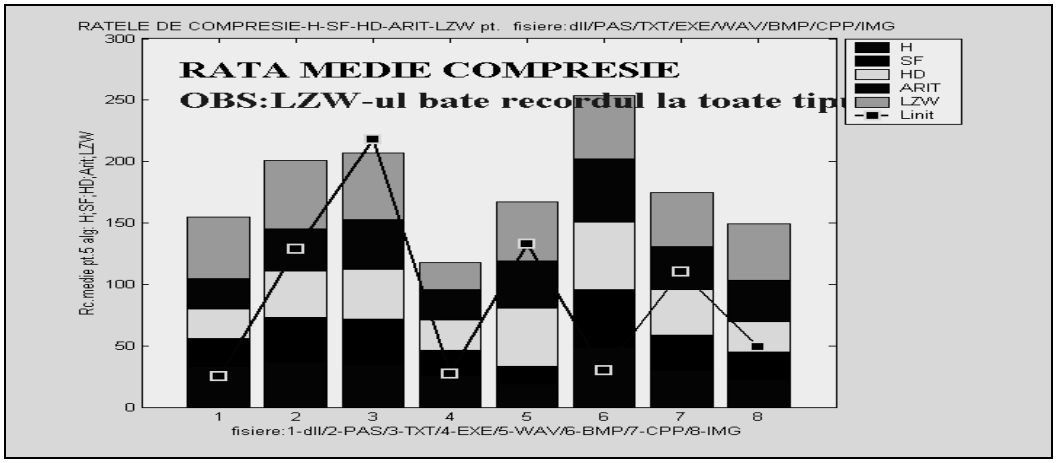

Figura 3. Valorile medii ale ratelor de compresie a 5 algoritmi şi 8 tipuri de fişiere

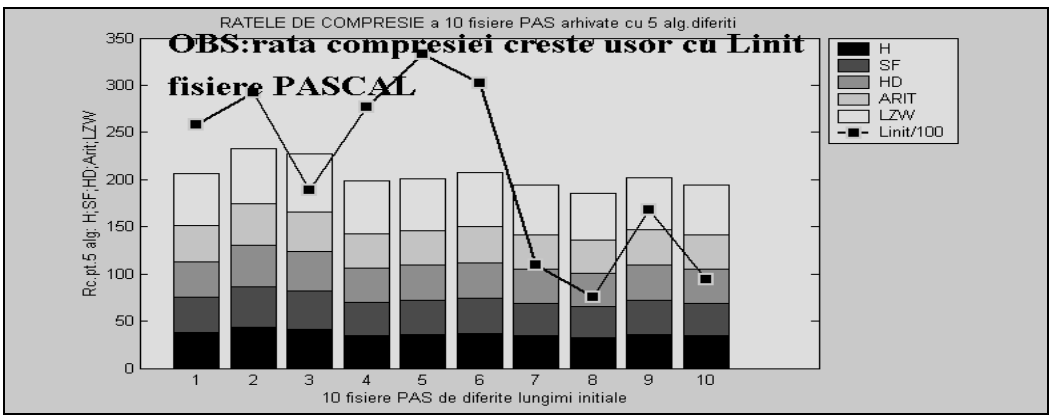

Figura 4. Cei 5 algoritmi testaţi pe 10 fişiere de acelaşi tip PASCAL dar de lungimi diferite prezentate în tabelele de mai sus.

În figura 3, cele 8 fişiere sunt ordonate crescător ca lungime iniţială. Concluzia care se desprinde este că, cea mai bună comportare o are algoritmul LZW care practic prezintă cea mai mare rată de compresie pentru toate fişierele, indiferent de lungimeaacestora.În figurile 4 şi 5 sunt prezentate rezultatele testării celor 5 algoritmi pe 10 fişiere de lungimi diferite dar de acelaşi tip, **PASCAL** respectiv **WAV**. Din figura 5 rezultă că la toţi algoritmii, rata de compresie creşte cu lungimea fişierelor şi mai rezultă că algoritmii **LZW** şi **HD** *furnizează cele mai bune rezultate la compresia fişierelor de sunet.*

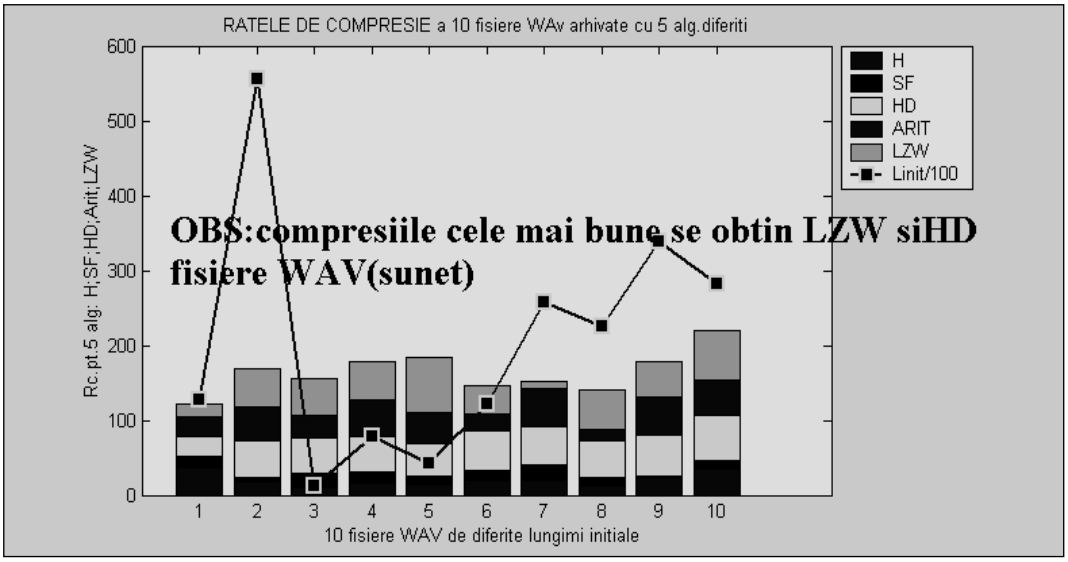

Figura 5. Cei 5 algoritmi testaţi pe 10 fişiere de acelaşi tip WAV dar de lungimi diferite

3. Concluzii

- Compresia datelor este folosită atât în cadrul sistemelor de stocare ori de transmisie la distanţă, a datelor de tip text sau a datelor de tip sunet ori de tip imagine reprezentate în format digital. Aplicaţiile dezvoltate în domeniul conducerii centralizate a unor roboţi cu vedere artificială, precum şi în cazul avioanelor de spionaj fără pilot este necesară transmisia datelor video într-o formă compresată pentru scurtarea duratei achiziţiei şi transmisiei datelor la sol. Aceasta explică importanţa acordată compresiei datelor de către cercetători.
- Tehnicile pentru compresia imaginilor sunt dominate de prelucrările de semnale în domeniul tehnologiei informaţiei. Creşterea vertiginoasă a aplicaţiilor care folosesc reprezentări 2D şi 3D şi în special transferul de fişiere grafice pe Internet au impus dezvoltarea de tehnici speciale şi mai ales de standarde specializate. Procedurile de compresie de imagini sunt complexe, asociind mai mulţi algoritmi. De aceea, rata distorsiunii este decisivă în alegerea nivelului de compresie, pentru că în funcţie de aplicaţie se poate merge până la rate foarte înalte de compresie, dacă detaliile imaginii nu sunt importante sau se lucrează cu imagini binarizate. Un alt aspect important este acela a timpilor de rulare a algoritmilor de compresie/decompresie, anumite proceduri nefiind indicate pentru transmitere la distanţă, ci doar pentru stocare, datorită timpului prea lung în care se face compresia. În fine, un aspect care nu poate fi neglijat este acela că există aplicaţii în care compresia de imagini trebuie asociată cu cea a semnalului audio şi eventual a unor dale de proces sau fişiere text, ceea ce ridică probleme de împachetare a cadrelor şi mai ales de armonizare a debitelor informaţionale.
- Fişierele de sunet având dimensiuni relativ mici, comprimarea lor se realizează în bune condiţii şi în timp scurt folosind algoritmul Huffman Standard chiar dacă rata de compresie pentru aceste fişiere este destul de scăzuta 35%.

BIBLIOGRAFIE

1. **DIATCU, E., A. TERTIŞCO, F. IACOB, M. TACHE, Z. RACOVIŢĂ**: Elemente fundamentale ale teoriei sistemelor şi calculatoarelor, editura HYPERION XXI, 1997;
2. **STEFĂNOIU, D.**: Compresia datelor, editura Printech, 2003;