

## SOFT-DECODAREA DATELOR RECEPȚIONATE DE LA NANO-SATELITUL SATUM

**Valentin Pocotilenco**  
Universitatea Tehnică a Moldovei  
[poco@mail.utm.md](mailto:poco@mail.utm.md)

**Abstract.** *Errors in communication channels are the problem that can be solved on different stages of data live, from data generation and application to the communication channel on transmitter until decoding and analyzing on receiver. This paper describes hard-decision and soft-decision decoding methods, their application, and is focused on one-sweep probabilistic algorithm of linear binary codes decoding.*

**Cuvinte-cheie:** *decodare probabilistică, trellis, hard-decision decodare, metode.*

### I. Introducere

Una din destinațiile nano-satelitului universitar SATUM este fotografierea suprafeței teritoriului R. Moldova și transmisiunea în timp real a imaginilor la stația terestră. Deoarece imaginile sunt unice și nu pot fi retransmise apare necesitatea aplicării metodelor sofisticate de corecție a erorilor. Una din aceste metode este decodarea probabilistică. Pentru codurile liniare care vor fi folosite în proiect este cunoscut algoritmul *one-sweep* de soft decodare cu decizie probabilistică [1]. În lucrare a fost elaborat instrumentarul de decodare *one-sweep* și au fost investigate caracteristicile de fiabilitate a decodării.

### II. Decodarea *soft-decision*

Ideea de bază pentru realizarea decodării *one-sweep* este executarea iterativă cu transferul probabilităților (*soft* informației) între iterații [2]. În algoritmul *one-sweep* probabilitățile condiționate se estimează iterativ conform relației:

$$P(v_n = i | \mathbf{r}, \mathbf{v} \in V) = \frac{P(\mathbf{r}_n^\perp | v_n = i, \mathbf{v} \in V) \cdot P(\mathbf{r}_n | v_n = i)}{I \cdot m(0, N)} \quad (1)$$

unde  $P(\mathbf{r}_n^\perp | v_n = i, \mathbf{v} \in V)$  sunt probabilitățile condiționate recepționării  $\mathbf{r}$  în condiția că a fost transmis  $v$ , care se calculează conform relației (2),  $P(\mathbf{r}_n | v_n = i)$  sunt valorile statistice pentru canalul de comunicație, iar  $Im(0, N)$  este metrica.

$$\begin{pmatrix} P(\mathbf{r}_n^\perp | v_n = 0, \mathbf{v} \in V) \\ P(\mathbf{r}_n^\perp | v_n = 1, \mathbf{v} \in V) \\ \dots \\ P(\mathbf{r}_n^\perp | v_n = I-1, \mathbf{v} \in V) \end{pmatrix} = M_n^{-1} \begin{pmatrix} m(0 \cdot h_n, N) \\ m(1 \cdot h_n, N) \\ \dots \\ m(I-1 \cdot h_n, N) \end{pmatrix} \quad (2)$$

unde  $m(I-1 \cdot h_n, N)$  sunt metricile drumurilor specific selectate în diagrama *trellis*.

Algoritmul *one-sweep* (fig. 1) la inițializare necesită aplicarea statisticii de canal și setarea valorilor inițiale al valorilor metricii. Statistica de canal reprezintă valoarea obținută de la ADC conec-

tat la ieșirea canalului de comunicație în dependență de simbolul aplicat la intrarea lui. În tabelul 1 sunt prezentate valorile statistice pentru ADC de 3 bit.

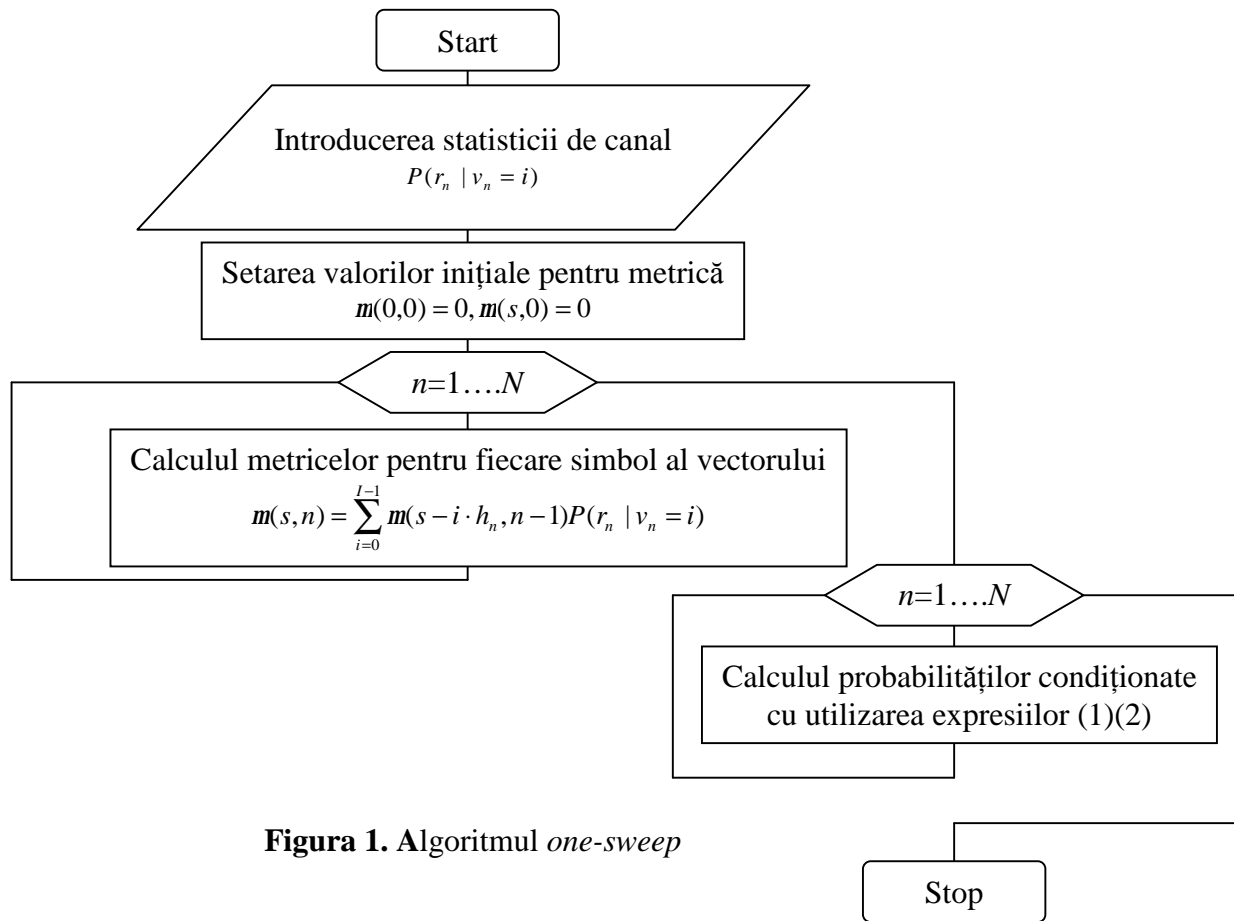


Figura 1. Algoritmul *one-sweep*

Tabelul 1. Valorile statistice în canal de comunicație

		Valori binare obținute la ADC (a) valorile statistice calculate (b)							
	a	000	001	010	011	100	101	110	111
"1"	b	0,524447	0,157586	0,128226	0,089199	0,055564	0,026364	0,011704	0,006911
"0"	b	0,006529	0,012537	0,029039	0,055676	0,091645	0,127453	0,152968	0,524153

### III. Instrumentarul

Instrumentarul elaborat permite estimarea metricelor  $m(I-1 \cdot h_n, N)$  prezentate în tabelul 2, generarea diagramei *trellis* în formă grafică, calculul numărului de căi pentru fiecare simbol al vectorului decodat, introducerea erorilor aleatorii sau în mod manual, calculul probabilităților pentru fiecare bit al vectorului recepționat. Interfața aplicației este creată utilizând mediul Matlab și constă din două pagini operaționale (fig.2).

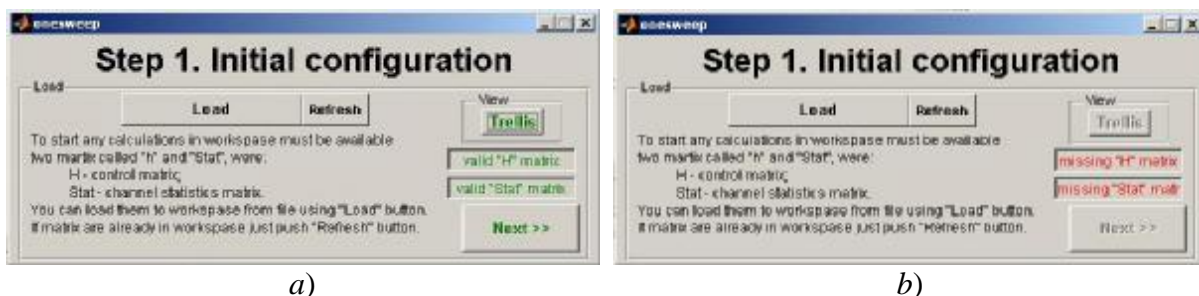
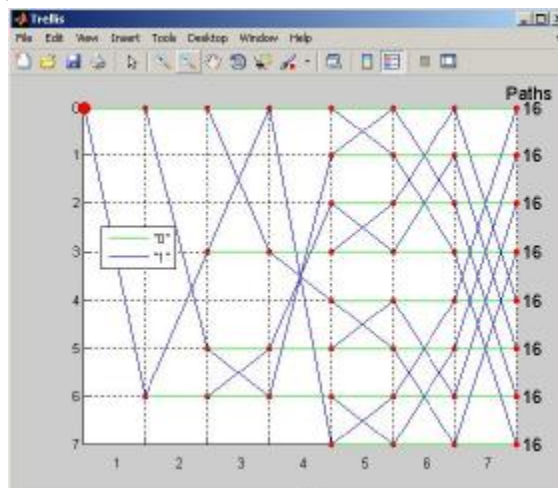


Figura 2. Interfața de lucru al instrumentarului elaborat

Pagina de bază (fig. 2,a) face posibilă încărcarea tabelului cu statistica și al matricei de control pentru codul binar linear în workspace cu ajutorul butonului “Load”. Butonul “Trellis” generează structura grafică al diagramei *trellis* (fig. 3) și calculează numărul de căi pentru fiecare punct final. La prima lansare al aplicației, la lipsa variabilelor necesare sau dacă variabilele sunt incorecte interfața va semnaliza utilizatorului prin trecerea butoanelor în formă inactivă și text de culoare roșie cu inscripțiile respective (fig. 2,b).

Figura 3. Diagrama *trellis*

Instrumentarul utilizând datele aplicate generează matricele cu valorile probabilistice de tranziție pentru calculul metricilor (tab. 3) căilor diagramei *trellis*.

Tabelul 3. Metricele codului linear

$\mu(0)$	$\mu(1)$	$\mu(2)$	$\mu(3)$	$\mu(4)$	$\mu(5)$	$\mu(6)$	$\mu(7)$
0,000149	0,000141	0,000141	0,000141	0,010906	0,000141	0,000141	0,000141

În tabelul 3 sunt prezentate metricele codului linear corector de o eroare cu parametrii  $(n,k)=(7,4)$ . Din tabel se observă că valoarea  $\mu(4)$  diferă, și această diferență este cauzată de eroare în bitul corespunzător.

La tastarea butonului “Next>>” va fi accesată a doua pagină al programului, unde propriu zis se efectuează calculele. În această pagină utilizatorul are posibilitatea de a introduce eroare singulară sau multiplă în mod manual sau automat cu ajutorul butonului “Generate” și selectând opțiunea dorită, poate calcula probabilitățile condiționate și poate reveni la pagina de inițializare (fig. 4).



Figura 4. Interfața de lucru al instrumentarului elaborat

Aplicația percepe erorile introduse manual și nu acceptă efectuarea calculelor semnalizând utilizatorului cauza (fig. 4,b).

Toate variabilele obținute în timpul calculelor sunt păstrate în spațiul de lucru Matlab, și la necesitate pot fi ușor citite și analizate (fig. 5).

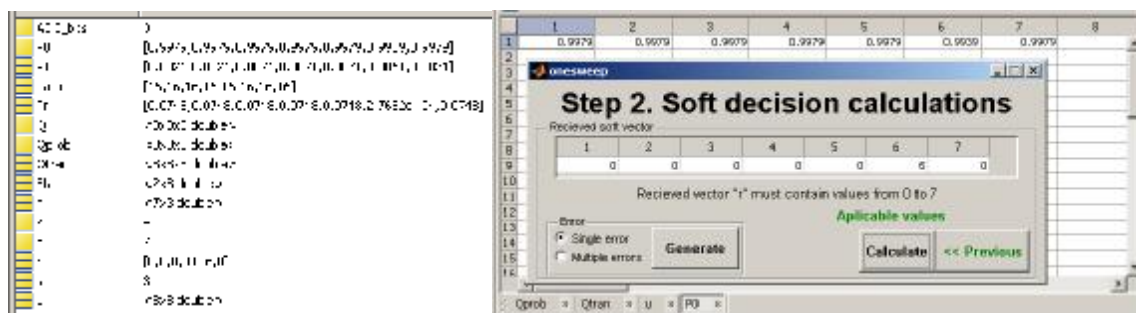


Figura 5. Prezentarea rezultatelor calculului

Aplicația permite generarea probabilităților condiționate pentru intervale de valori al erorilor singulare și multiple. În tabelul 4 sunt prezentate probabilitățile condiționale recepționării valorii “1” în condiția că a fost transmis “0”, unde  $v_1..v_7$  sunt componentele vectorului recepționat, iar  $P(r|v_1=1)..P(r|v_7=1)$  probabilitățile condiționale calculate de algoritmul *one-sweep*.

Tabelul 4. Probabilitățile valorii “1” recepționate pentru fiecare bit al cuvântului decodat

$v_1$	$v_2$	$v_3$	$v_4$	$v_5$	$v_6$	$v_7$	$P(r v_1=1)$	$P(r v_2=1)$	$P(r v_3=1)$	$P(r v_4=1)$	$P(r v_5=1)$	$P(r v_6=1)$	$P(r v_7=1)$
0	0	0	0	0	0	0	5,88E-06	5,88E-06	5,88E-06	5,88E-06	5,88E-06	5,88E-06	5,88E-06
...	...	...	...	...	...	...	...	...	...	...	...	...	...
3	0	0	0	0	0	0	0,000295	0,000103	0,000103	0,000103	0,000103	0,000103	0,000103
...	...	...	...	...	...	...	...	...	...	...	...	...	...
7	0	0	0	0	0	0	0,034609	0,011634	0,011634	0,011634	0,011634	0,011634	0,011634

În tabelul 4 se vede clar că pentru orice valoare al bitului 1 probabilitatea că este recepționat “1” este mult mai mică de cât pentru “0” logic. Prima linie conține probabilitățile condiționale ale recepționării cuvântului de cod, deoarece probabilitatea pentru “0” este cu mult mai mare de cit pentru “1” se va lua decizia ca vectorul recepționat este corect. În coloana  $v_1$  pentru următoarele linii este specificată eroarea de grad diferit. Indiferent de acesta probabilitatea condițională  $P(r|v_1=1)$  a lui “1” față de “0” rămâne mică ceea ce permite luarea deciziei că a fost transmis anume “0”.

#### IV. Concluzii

În lucrare este prezentat instrumentarul pentru decodarea *soft-decision* a codurilor binare lineare corectoare de erori în baza algoritmului *one-sweep*. În rezultatul aplicării algoritmului propus complexitatea *trellis*-ului s-a micșorat de la  $2^{2(n-k)}$  [4] până la  $2^{(n-k)}$ . A fost determinată repartiția fiabilității *soft-decodării* în dependență de numărul de erori, care permite decodarea erorilor multiple cu nivelul de încredere egal cu 0,9.

#### V. Referințe

1. Thomas Johansson, Kamil Zigangirov, “A Simple One-Sweep Algorithm for Optimal APP Symbol Decoding of Linear Block Codes”, IEEE Trans. Inform. Theory, vol. 44, Nov 1998.
2. J. Hagenauer, E. Offer, and L.Papke, “Iterative decoding of binary block and convolutional codes”, IEEE Trans. Inform. Theory, vol.42, pp. 429–445, Mar. 1996.
3. R. J. McEliece, “On the BCJR trellis for linear block codes”, IEEE Trans. Inform. Theory, vol.42, pp. 1072–1092, July1996.
4. L. R. Bahl, J. Cooke, F. Jelinek, and J. Raviv, “Optimal decoding of linear codes for minimizing symbol error rate”, IEEE Trans. Inform. Theory, vol. IT-20, pp.284–287, Mar. 1974.