# Syndrome Trellis Coding

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| H(M,N)= |  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 3 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| 4 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |

M=3, N=9, , , , . We need finding such that

and the cost of modification is minimized.

Example 1. Let . Then

Function multHz(input M row number of H,

input N col number of H,

input w size of seed matrix,

input Hhat[w,w] seed matrix,

input z[N] coded vector,

output m[M} calculated syndrome){

m[1..M]=0; //output initialization

state[1..w]=0; //state initialization

Hhat\_num= int(N/w);//number of seed matrices used

for(r=0, i=1; r<=min(Hhat\_num -1,M-1); r++, i++){

for(s=1; s<=w; s++)

state=state XOR Hhat[1..w, s]\*z[r\*w+s]; // end loop on s

m[i]=state[1]; //i-th value is ready

state=shiftleft(state, 1); // shift 1 position left content of state: state[1]=state[2],

// state[2]=state[3], .., state[w-1]=state[w]

// state[w]=0; the last element after shift set zero

}// end loop on r

İf(M> Hhat\_num)//M< Hhat\_num +w;

For(i= Hhat\_num +1; i<=M; i++)

m[i]=state[i- Hhat\_num]; //take left m values from state

Example 2. Apply multHz in the conditions of Example 1.

M=3; N=9; h=w=3; Hhat=; z[1..N]=[0 1 1 0 0 1 1 0 0];

m[1..M] = [0 0 0];

State[1..3]=[0 0 0];

Hhat\_num =int(N/w)=int(9/3)=3;

İ=1;

r=0;

s=1;

state=state XOR Hhat[1..w, 1]\*={0 0 0] XOR [1 1 1]\*z[1] = [0 0 0] XOR [0 0 0] = [0 0 0];

s=2;

state=state XOR Hhat[1..w, 2]\*={0 0 0] XOR [1 0 1]\*z[2] = [0 0 0] XOR [1 0 1] = [1 0 1];

s=3;

state=state XOR Hhat[1..w, 3]\*={1 0 1] XOR [1 0 0]\*z[3] = [1 0 1] XOR [1 0 0] = [0 0 1];

m[i]=m[1]=state[1]=0;

state= shiftleft(state, 1)= [0 1 0];

i=2; r=1;

s=1;

state=state XOR Hhat[1..w, 1]\*z[1\*3+1]=state XOR Hhat[1..w,1]\*z[4]=[0 1 0] XOR [1 1 1]\*0 = [0 1 0];

s=2;

state=state XOR Hhat[1..w, 2]\*z[1\*3+2]=state XOR Hhat[1..w,2]\*z[5]=[0 1 0] XOR [1 0 1]\*0 = [0 1 0];

s=3;

state=state XOR Hhat[1..w, 3]\*z[1\*3+3]=state XOR Hhat[1..w,3]\*z[6]=[0 1 0] XOR [1 0 0]\*1 = [0 1 0] XOR [1 0 0] = [1 1 0];

m[2]=state[1]=1;

state= shiftleft(state, 1)= [1 0 0];

i=3; r=2;

s=1;

state=state XOR Hhat[1..w, 1]\*z[2\*3+1]=state XOR Hhat[1..w,1]\*z[7]=[1 0 0] XOR [1 1 1]\*1 = [1 0 0] XOR [1 1 1] = [0 1 1];

s=2;

state=state XOR Hhat[1..w, 2]\*z[2\*3+2]=state XOR Hhat[1..w,2]\*z[8]=[0 1 1] XOR [1 0 1]\*0 = [0 1 1];

s=3;

state=state XOR Hhat[1..w, 3]\*z[2\*3+3]=state XOR Hhat[1..w,3]\*z[9]=[0 1 1] XOR [1 0 0]\*0 = [0 1 1];

m[3]=state[1]=0;

state= shiftleft(state, 1)= [1 1 0];

Thus, m[1..3]=[0 1 0] that is the same as in the result of Example 1.

Consider now Syndrome trellis graph:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| R | 0 | | |  | 1 | | |  | 2 | | |  |  |
| State\s | 1 | 2 | 3 |  | 1 | 2 | 3 |  | 1 | 2 | 3 |  |  |
| 000 | 000 |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 001 |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 010 |  |  |  |  |  |  |  |  |  |  |  |  |
| 1 | 011 |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 100 |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 101 |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 110 |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 111 |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | m[1[=0 |  |  |  | m[2]=1 |  |  |  | m[3]=0 |  |

Let cover image, CI, be

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | 31 | 49 | 51 | 129 | 211 | 107 | 11 |

Then X=P(CI)=CI mod 2 = (111 111 111), Hhat=(7,5,1)

Cost matrix; shows shifts; if in the same row, 0; otherwise, 1;

When stae is not changed, resulting bit is 0, and the cost is ncreased by 1 if x=1 and by 0, if x=0, i.e, by x\*weight

When state changes, resulting bit is 1, and the cost is increased by 1 if x=0, and by 0, if x=1, i.e. by (1-x)\*weight

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Round | | 0 | | |  | 1 | | |  | 2 | | |  |
| State\s | | 1 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 3 |
| 000 | 0 | 1 | 2 | 3 | 3 | 4 | Min(5,1)  =1 | Min(2,1)  =1 | 1 | 2 | Min(3,2)  =2 | Min(3,2)  =2 | 2 |
| 001 | inf | inf | İnf | 2 | 1 | 2 | Min(3,1)  =1 | Min(2,1)  =1 | 2 | 3 | Min(4,2)  =2 | Min(2,3)  =2 | 2 |
| 010 | İnf | İnf | Min(İnf,0)  =0 | 1 | 1 | 2 | 3 | Min(4,1)  =1 | 2 | 3 | Min(4,1)  =1 | 2 | 2 |
| 011 | İnf | İnf | İnf | 0 | 1 | 2 | Min(3,1)  =1 | Min(3,2)  =2 | 2 | 3 | Min(4,2)  =2 | Min(1,3)  =1 | 2 |
| 100 | İnf | İnf | İnf | Min(İnf,1)  =1 | inf | 1 | 2 | Min(3,2)  =2 | İnf | 2 | 3 | Min(4,2)  =2 |  |
| 101 | inf | inf | Min(İnf,1)  =1 | Min(İnf,2)  =2 | İnf | 1 | Min(4,2)  =2 | Min(2,3)  =2 | İnf | 2 | Min(2,3)  =2 | 3 |  |
| 110 | İnf | İnf | İnf | Min(İnf,1)  =1 | İnf | 1 | 2 | Min(3,2)  =2 | İnf | 2 | 3 | Min(4,2)  =2 |  |
| 111 | İnf | 0 | 1 | Min(inf,2)  =2 | inf | 3 | 2 | Min(3,2)  =2 | inf | 1 | Min(3,2)  =2 | 3 |  |
|  | | | | | m[1]=0 |  | | | m[2]=1 |  | | | m[3]=0 |

The following resuslt is obtained: y=P(SI)=(011 111 011), where SI is stego-mage=

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | 3 | 31 | 49 | 51 | 129 | 210 | 107 | 11 |