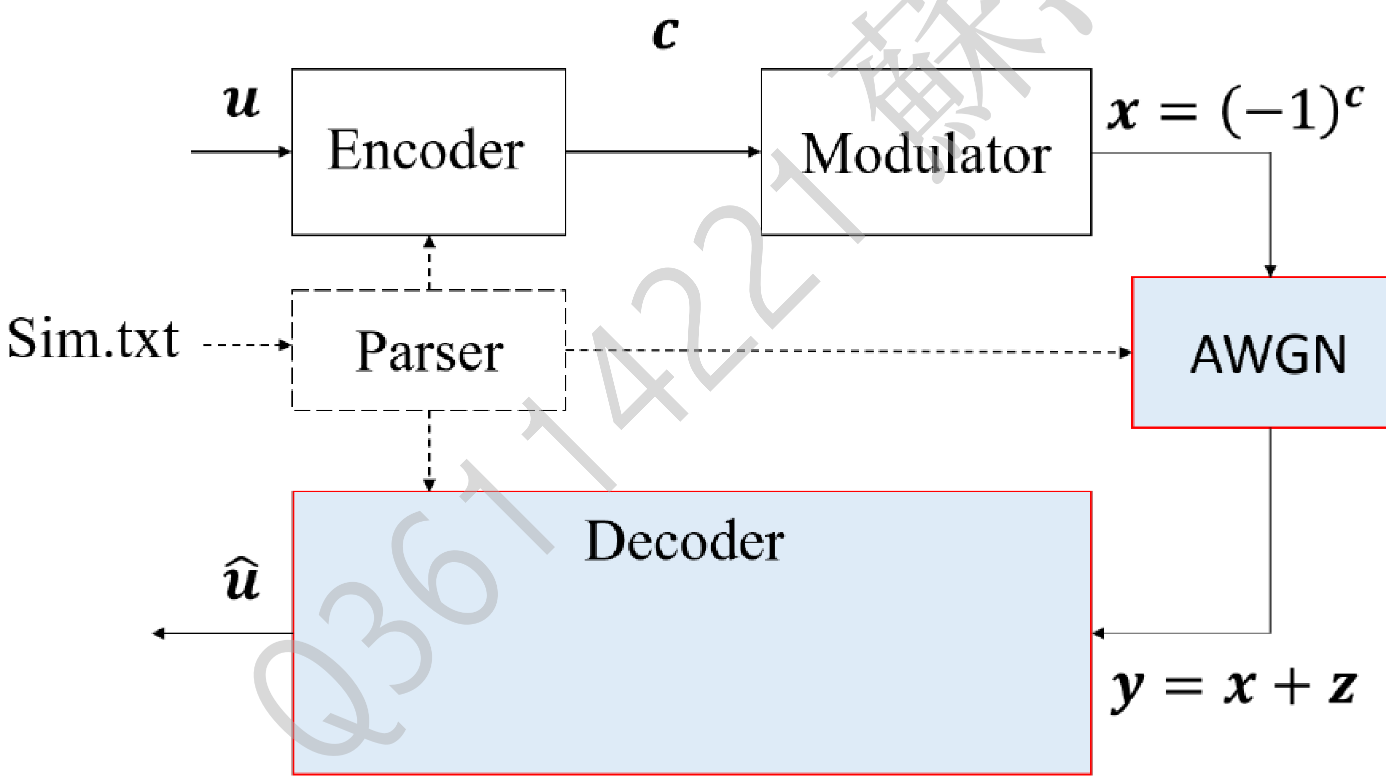
Error Control Coding - Decoding of Convolutional Codes

電通所　Q361142221　蘇沛錦

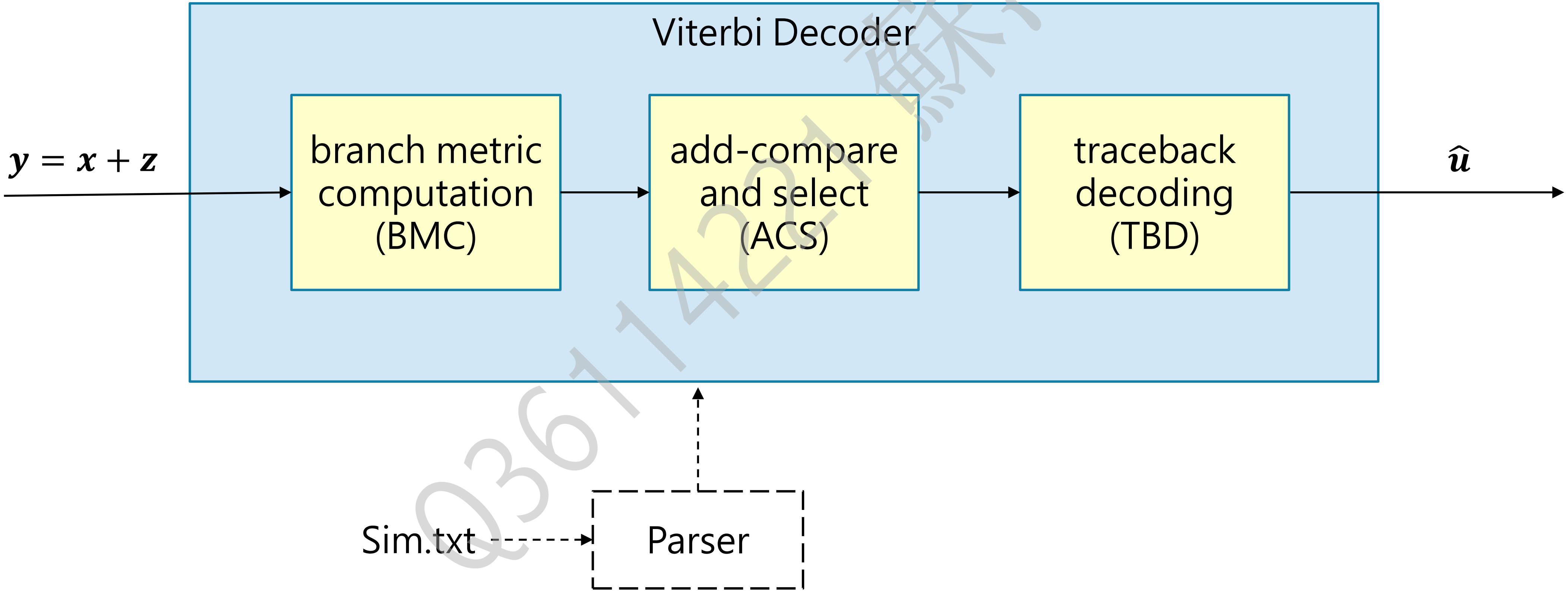
Outline

- 系統架構圖 (Block Diagram)
- 程式流程解釋
- 參數設定
- 模擬數據
- 模擬圖
- 報告加分內容(Bonus)

系統架構圖 (Block Diagram)



系統架構圖 (Block Diagram)



程式流程解釋

```
for (i = 0; i < (N + m); i++)
{
    ➢ branch metric computation (BMC)
    ➢ add-compare and select (ACS)
    ➢ traceback decoding (TBD)
    ➢ Sliding Truncation Window (→ Shift Register)
}
```

程式流程解釋

1. 首先計算 Trellis diagram 上面所有 branch 的 metric 數值並儲存起來。

```
/* branch metric computation (BMC) */
void BMC(double* bMet, int** stateSur, double y0, double y1, int TW_Idx, int choice)
{
    int i;          // for loop counter
    int bMet_Idx;   // bMet (branch metric memory) address
    int* x;         // branch codeword
    int y_est0, y_est1;

    for (i = 0; i < 128; i++) bMet[i] = -1;      // 將記憶體區塊內的值初始化

    if (TW_Idx == 0)
    {
        int2bit(0, &sReg[0], &sReg[1], &sReg[2], &sReg[3], &sReg[4], &sReg[5], &sReg[6]);
        x = BCC(sReg[0], sReg[1], sReg[2], sReg[3], sReg[4], sReg[5], sReg[6]);  // branch codeword computation (BCC)
        bMet_Idx = bit2int(sReg[0], sReg[1], sReg[2], sReg[3], sReg[4], sReg[5], sReg[6], 7);
        stateSur[bit2int(sReg[0], sReg[1], sReg[2], sReg[3], sReg[4], sReg[5], sReg[6], 6)][TW_Idx] = 65;
        switch (choice) ···

        int2bit(1, &sReg[0], &sReg[1], &sReg[2], &sReg[3], &sReg[4], &sReg[5], &sReg[6]);
        x = BCC(sReg[0], sReg[1], sReg[2], sReg[3], sReg[4], sReg[5], sReg[6]);  // branch codeword computation (BCC)
        bMet_Idx = bit2int(sReg[0], sReg[1], sReg[2], sReg[3], sReg[4], sReg[5], sReg[6], 7);
        stateSur[bit2int(sReg[0], sReg[1], sReg[2], sReg[3], sReg[4], sReg[5], sReg[6], 6)][TW_Idx] = 65;
        switch (choice) ···
    }
    else if (TW_Idx < 0) ···
    else
    {
        for (i = 0; i < 64; i++) ···
    }

    free(x);    // 釋放由指標變數所指向的記憶空間
}
```

程式流程解釋

2. 針對每個 state 選擇出 metric 總和最小的 branch，並且將該路徑儲存起來。

```c
/* add-compare and select  (ACS) */
void ACS(double* bMet, int** stateSur, double** stateMet, int TW_Idx)
{
    int i;              // for loop counter
    double bMet0, bMet1;   // branch metric
    int stateMet_Idx;   // stateMet (state metric memory) address (previous)
    double m0, m1;

    for (i = 0; i < 64; i++)
    {
        if (stateSur[i][TW_Idx] != -1)
        {
            int2bit(i, &sReg[0], &sReg[1], &sReg[2], &sReg[3], &sReg[4], &sReg[5], &sReg[6]);

            bMet0 = bMet[bit2int(sReg[0], sReg[1], sReg[2], sReg[3], sReg[4], sReg[5], 0, 7)];  // 上面 branch 的 metric
            bMet1 = bMet[bit2int(sReg[0], sReg[1], sReg[2], sReg[3], sReg[4], sReg[5], 1, 7)];  // 下面 branch 的 metric

            if (bMet0 != -1 && bMet1 == -1)         // 只有上面 branch 的 metric 有值…
            else if (bMet0 == -1 && bMet1 != -1)    // 只有下面 branch 的 metric 有值…
            else    // 兩個 branch 的 metric 皆有值 -> 選擇 metric 最小的 branch…
            stateSur[i][TW_Idx] = stateMet_Idx;      // Survivor updating
        }
    }

    for ( i = 0; i < 64; i++)
    {
        stateMet[i][0] = stateMet[i][1];    // 移動暫存器的內容
        stateMet[i][1] = -1;    // 將記憶體區塊內的值初始化
    }
}
```

程式流程解釋

3. 選擇 metric 最小的 state，並且回溯出該 state 之前儲存的路徑 (survivor)。

```c
/* traceback decoding (TBD) */
int* TBD(int** stateSur, double** stateMet, int L, int TBD_ctrl)
{
    int i;   // for loop counter
    int* u_est = (int*)calloc(L, sizeof(int));   // branch codeword
    double minMet = stateMet[0][0]; // minimum state metric
    int TB_Idx = 0; // traceback index

    switch (TBD_ctrl)
    {
        case 0:
            /* Best-state */
            for (i = 1; i < 64; i++)
            {
                if (stateMet[i][0] < minMet && stateSur[i][L-1] != -1)
                {
                    minMet = stateMet[i][0];
                    TB_Idx = i;
                }
            }
            break;
        default:    // 最終 trellis diagram 必定收斂到 00 這個 state
            break;
    }

    for (i = 0; i < L; i++)     // Perform trace-back for output
    {
        int2bit(TB_Idx, &sReg[0], &sReg[1], &sReg[2], &sReg[3], &sReg[4], &sReg[5], &sReg[6]);
        u_est[(L-1)-i] = sReg[0];
        TB_Idx = stateSur[TB_Idx][(L-1)-i];
    }

    return u_est;
}
```

程式流程解釋

4. 當 traceback 解出訊息後，滑動 Truncation window 至下一個計算時刻。

```
/* 執行 Viterbi 演算法 */
for (i = 0; i < (N + m); i++)
{
    if (i < N) ···
    else ···

    /* add-compare and select  (ACS) */
    ACS(bMet, stateSur, stateMet, TW_Idx);

    /* Sliding Window */
    if (TW_Idx == L-1)
    {
        if (i == (N + m - 1))    // 跑完整個 trellis diagram 的情況 ···
        else    // 尚未跑完整個 trellis diagram ···

        for (j = 0; j < 64; j++)
        {
            for (k = 0; k < (L-1); k++)
            {
                stateSur[j][k] = stateSur[j][k+1];
            }
            stateSur[j][L-1] = -1;
        }

        // Deallocate memory block
        free(uReg);          // 釋放由指標變數所指向的記憶空間
    }
    else
    {
        TW_Idx = TW_Idx + 1;
    }
}
```

參數設定

- 採用 Best-state 作法 (若是遇到同樣的 metric，也是選最上面的 state)。
- $L = 32$ (Truncation window length)
- ACS 遇到同樣大小的 metric 時，總是選最上面的 branch。
- $N = 500000$ (Number of decoded bits)
- 為了讓模擬出來的位元錯誤率更加準確，我使用五個不同的 seed 跑:

  Seed0 ~ Seed4 = {-1000, -1500, -2000, -2500, -3000}

  並且將這五條線分別畫出來。
- 然後把五個 seed 的資料加起來一起統計，劃出一條平均的 BER 曲線。

模擬數據

Hard-Decision

a) Seed0 = -1000

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6
No. decoded bits	499968												
No. decided bit errors	188829	162268	130758	95688	61646	35631	17447	7832	2969	967	268	63	13
bit error rate (BER)	3.78E-01	3.25E-01	2.62E-01	1.91E-01	1.23E-01	7.13E-02	3.49E-02	1.57E-02	5.94E-03	1.93E-03	5.36E-04	1.26E-04	2.60E-05

b) Seed1 = -1500

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6
No. decoded bits	499968												
No. decided bit errors	186814	162092	130579	94837	62057	36161	17652	7919	2859	1091	359	98	33
bit error rate (BER)	3.74E-01	3.24E-01	2.61E-01	1.90E-01	1.24E-01	7.23E-02	3.53E-02	1.58E-02	5.72E-03	2.18E-03	7.18E-04	1.96E-04	6.60E-05

模擬數據

Hard-Decision

c) Seed2 = -2000

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6
No. decoded bits	499968												
No. decided bit errors	188794	163007	130921	95232	62508	35837	18316	8117	3194	1244	404	113	16
bit error rate (BER)	3.78E-01	3.26E-01	2.62E-01	1.90E-01	1.25E-01	7.17E-02	3.66E-02	1.62E-02	6.39E-03	2.49E-03	8.08E-04	2.26E-04	3.20E-05

d) Seed3 = -2500

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6
No. decoded bits	499968												
No. decided bit errors	189957	163239	131612	95291	61932	36832	18348	8081	2985	1074	321	95	16
bit error rate (BER)	3.80E-01	3.26E-01	2.63E-01	1.91E-01	1.24E-01	7.37E-02	3.67E-02	1.62E-02	5.97E-03	2.15E-03	6.42E-04	1.90E-04	3.20E-05

模擬數據

參考數值：

- Hard-decision:
  - BER = 2.1×10^{-3} at $E_b/N_0 = 4.5$ dB;
  - BER = 6.4×10^{-4} at $E_b/N_0 = 5.0$ dB.
- Soft-decision:
  - BER = 2.2×10^{-3} at $E_b/N_0 = 2.5$ dB;
  - BER = 5.3×10^{-4} at $E_b/N_0 = 3.0$ dB.

Hard-Decision

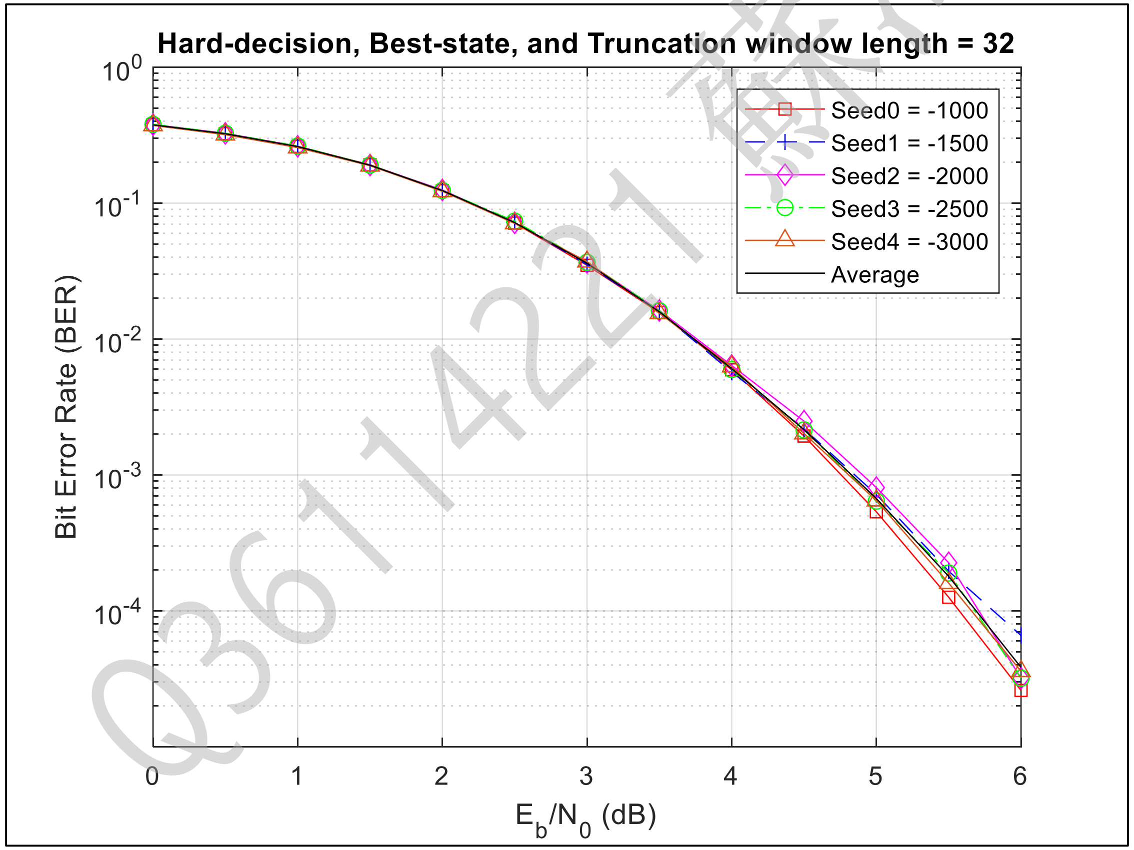
e) Seed4 = -3000

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6
No. decoded bits	499968												
No. decided bit errors	186627	159468	128100	94242	61088	35464	18557	7732	3137	1009	324	80	18
bit error rate (BER)	3.73E-01	3.19E-01	2.56E-01	1.88E-01	1.22E-01	7.09E-02	3.71E-02	1.55E-02	6.27E-03	2.02E-03	6.48E-04	1.60E-04	3.60E-05

f) Average

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6
No. decoded bits	499968												
No. decided bit errors	188204.2	162014.8	130394	95058	61846.2	35985	18064	7936.2	3028.8	1077	335.2	89.8	19.2
bit error rate (BER)	3.76E-01	3.24E-01	2.61E-01	1.90E-01	1.24E-01	7.20E-02	3.61E-02	1.59E-02	6.06E-03	2.15E-03	6.70E-04	1.80E-04	3.84E-05

模擬圖



Hard-decision, Best-state, and Truncation window length = 32
Seed0 = -1000
Seed1 = -1500
Seed2 = -2000
Seed3 = -2500
Seed4 = -3000
Average
Bit Error Rate (BER)
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
0
1
2
3
4
5
6
E_b/N_0 (dB)

模擬數據

Soft-Decision

a) Seed0 = -1000

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4
No. decoded bits	499968								
No. decided bit errors	85784	51849	26349	10860	3741	1054	251	46	21
bit error rate (BER)	1.72E-01	1.04E-01	5.27E-02	2.17E-02	7.48E-03	2.11E-03	5.02E-04	9.20E-05	4.20E-05

b) Seed1 = -1500

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4
No. decoded bits	499968								
No. decided bit errors	84547	49787	24717	10112	3418	947	261	54	10
bit error rate (BER)	1.69E-01	9.96E-02	4.94E-02	2.02E-02	6.84E-03	1.89E-03	5.22E-04	1.08E-04	2.00E-05

模擬數據

Soft-Decision

c) Seed2 = -2000

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4
No. decoded bits	499968								
No. decided bit errors	84261	51599	25956	11190	4114	1339	308	79	12
bit error rate (BER)	1.69E-01	1.03E-01	5.19E-02	2.24E-02	8.23E-03	2.68E-03	6.16E-04	1.58E-04	2.40E-05

d) Seed3 = -2500

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4
No. decoded bits	499968								
No. decided bit errors	85630	51717	26816	11525	3892	1057	225	60	16
bit error rate (BER)	1.71E-01	1.03E-01	5.36E-02	2.31E-02	7.78E-03	2.11E-03	4.50E-04	1.20E-04	3.20E-05

模擬數據

參考數值：

- Hard-decision:
  - BER = 2.1×10^{-3} at $E_b/N_0 = 4.5$ dB;
  - BER = 6.4×10^{-4} at $E_b/N_0 = 5.0$ dB.
- Soft-decision:
  - BER = 2.2×10^{-3} at $E_b/N_0 = 2.5$ dB;
  - BER = 5.3×10^{-4} at $E_b/N_0 = 3.0$ dB.

Soft-Decision

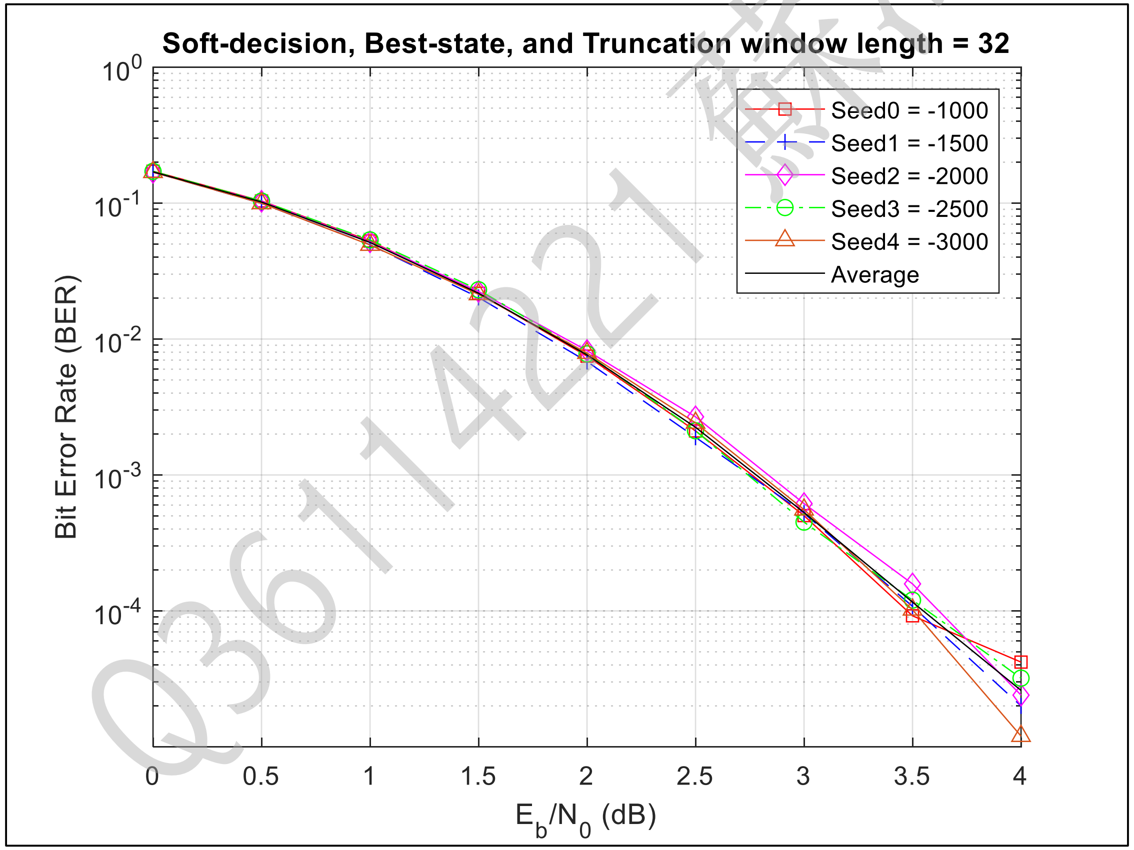
e) Seed4 = -3000

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4
No. decoded bits	499968								
No. decided bit errors	84470	49816	24613	10659	3946	1208	278	51	6
bit error rate (BER)	1.69E-01	9.96E-02	4.92E-02	2.13E-02	7.89E-03	2.42E-03	5.56E-04	1.02E-04	1.20E-05

f) Average

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4
No. decoded bits	499968								
No. decided bit errors	84938.4	50953.6	25690.2	10869.2	3822.2	1121	264.6	58	13
bit error rate (BER)	1.70E-01	1.02E-01	5.14E-02	2.17E-02	7.64E-03	2.24E-03	5.29E-04	1.16E-04	2.60E-05

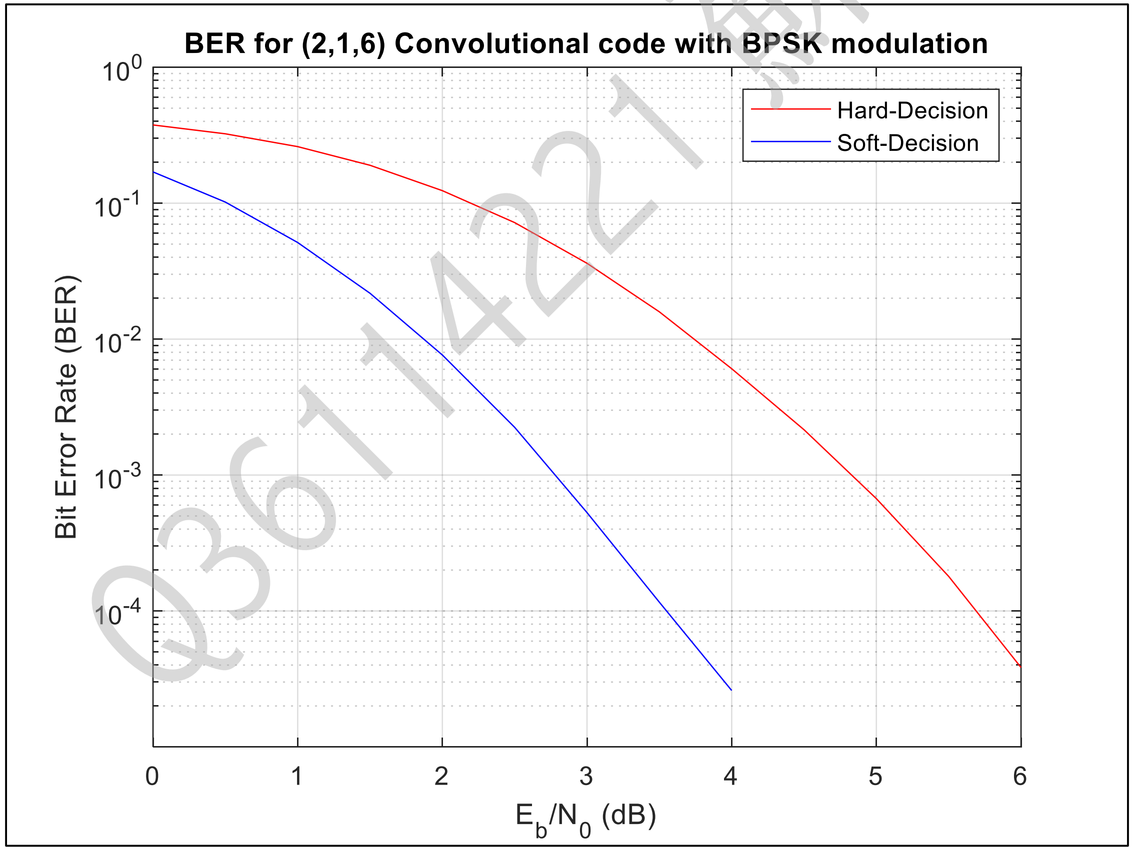
模擬圖



Soft-decision, Best-state, and Truncation window length = 32
Bit Error Rate (BER)
E_b/N_0 (dB)
10^0
10^{-1}
10^{-2}
10^{-3}
10^{-4}
0
0.5
1
1.5
2
2.5
3
3.5
4
Seed0 = -1000
Seed1 = -1500
Seed2 = -2000
Seed3 = -2500
Seed4 = -3000
Average

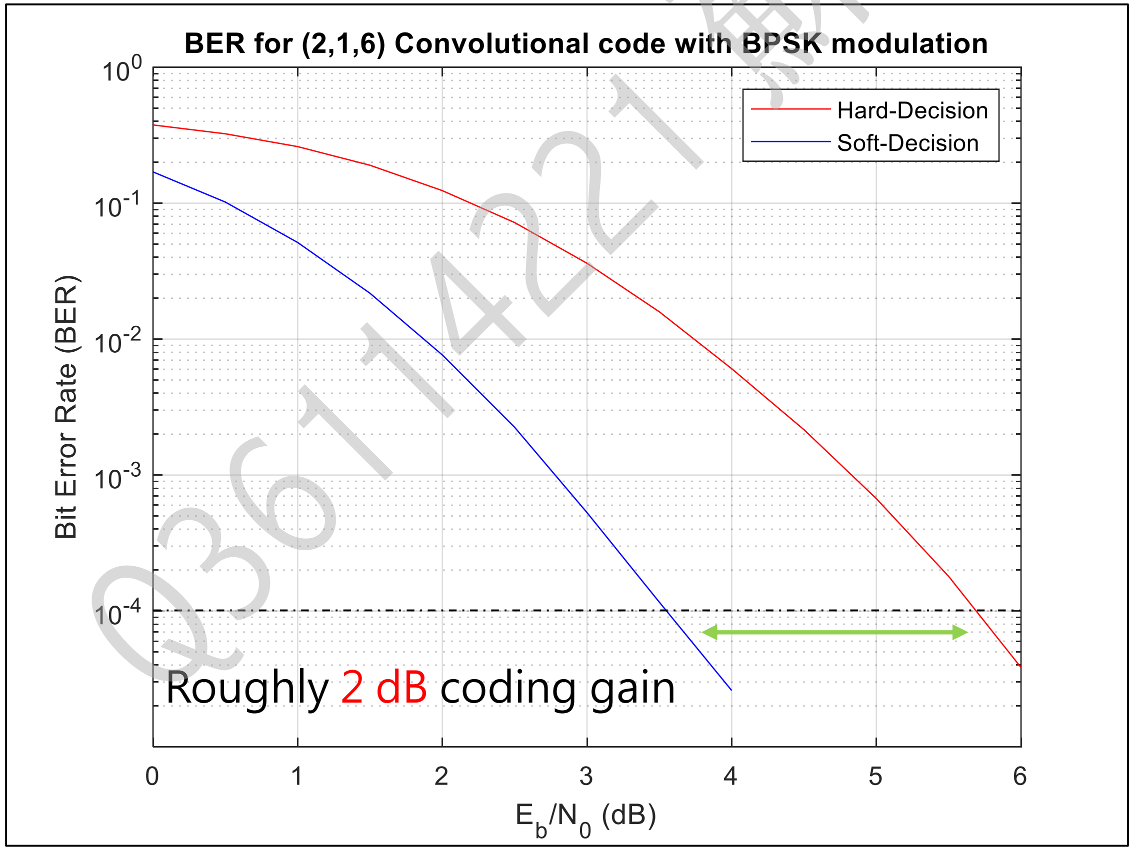
模擬圖

平均的 BER 曲線 (Hard-decision and Soft-decision)



模擬圖

Coding gain at BER = 10^{-4} (Soft-Decision over Hard-Decision)



報告加分內容(Bonus)

1. 不同 Truncation window length 的比較
2. Best-state、fixed-state的比較

不同 Truncation window length 的比較

- **參數設定**
  - 採用 Best-state 作法 (若是遇到同樣的 metric，也是選最上面的 state)。
  - $L = 32$ or 18 (Different truncation window length)
  - ACS 遇到同樣大小的 metric 時，總是選最上面的 branch。
  - $N = 500000$ (Number of decoded bits)
  - 為了讓模擬出來的位元錯誤率更加準確，我使用五個不同的 seed 跑:

    Seed0 ~ Seed4 = {-1000, -1500, -2000, -2500, -3000}

    並且將這五條線分別畫出來。
  - 然後把五個 seed 的資料加起來一起統計，劃出一條平均的 BER 曲線。

模擬數據 (把五個不同 seed 的資料加起來一起統計)

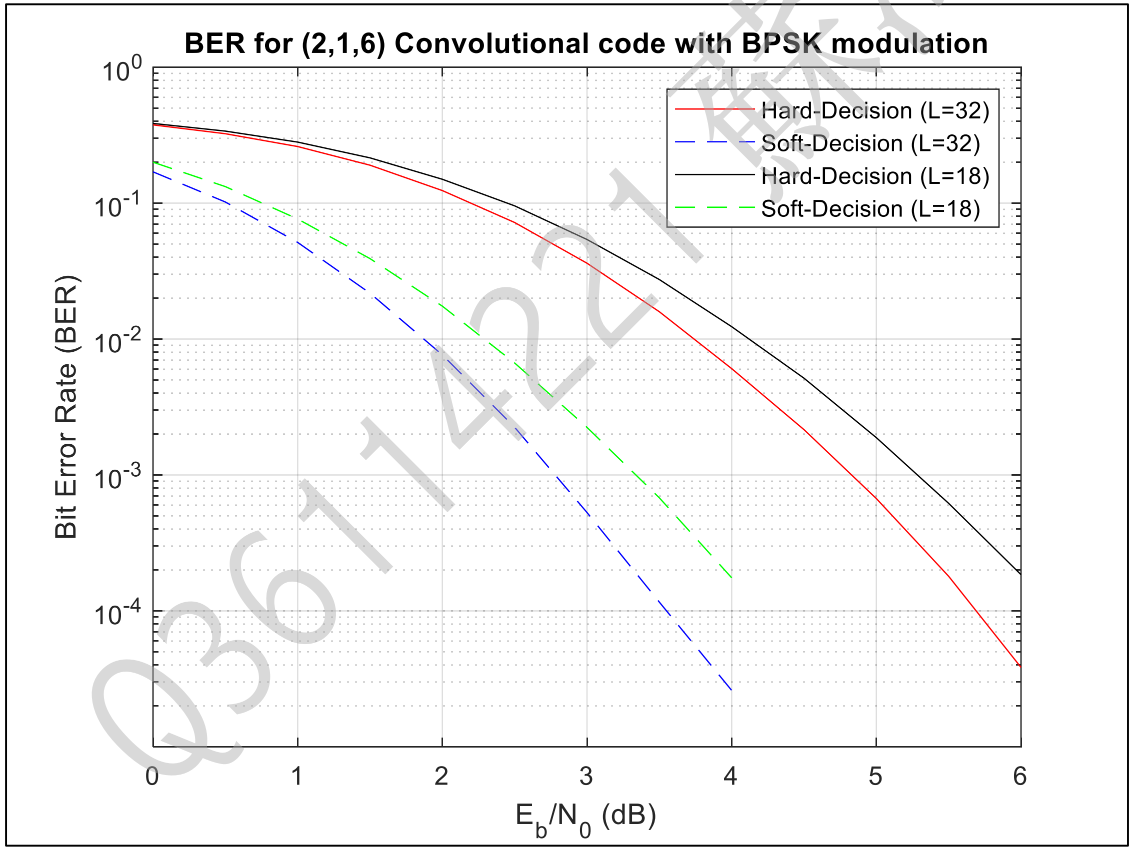
1. Hard-Decision (Truncation window length = 18)

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6
No. decoded bits	499968												
No. decided bit errors	192812.2	169356.2	140615.6	107591.8	74897.2	47717.4	26995.8	13695.4	6164	2574.6	940.6	309	92.4
bit error rate (BER)	3.86E-01	3.39E-01	2.81E-01	2.15E-01	1.50E-01	9.54E-02	5.40E-02	2.74E-02	1.23E-02	5.15E-03	1.88E-03	6.18E-04	1.85E-04

2. Soft-Decision (Truncation window length = 18)

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4
No. decoded bits	499968								
No. decided bit errors	100021.8	66135	38198.2	19532.4	8727.6	3330.8	1119.8	339.4	87.6
bit error rate (BER)	2.00E-01	1.32E-01	7.64E-02	3.91E-02	1.75E-02	6.66E-03	2.24E-03	6.79E-04	1.75E-04

模擬圖



BER for (2,1,6) Convolutional code with BPSK modulation
Hard-Decision (L=32)
Soft-Decision (L=32)
Hard-Decision (L=18)
Soft-Decision (L=18)
Bit Error Rate (BER)
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
0
1
2
3
4
5
6
E_b/N_0 (dB)

Best-state、fixed-state的比較

- **參數設定**
  - 嘗試採用 fixed-state 作法 (不管 state 的 metric 大小，固定選最上面的 state)。
  - $L = 32$ (Truncation window length)
  - ACS 遇到同樣大小的 metric 時，總是選最上面的 branch。
  - $N = 500000$ (Number of decoded bits)
  - 為了讓模擬出來的位元錯誤率更加準確，我使用五個不同的 seed 跑:

    Seed0 ~ Seed4 = {-1000, -1500, -2000, -2500, -3000}

    並且將這五條線分別畫出來。
  - 然後把五個 seed 的資料加起來一起統計，劃出一條平均的 BER 曲線。

模擬數據 (把五個不同 seed 的資料加起來一起統計)

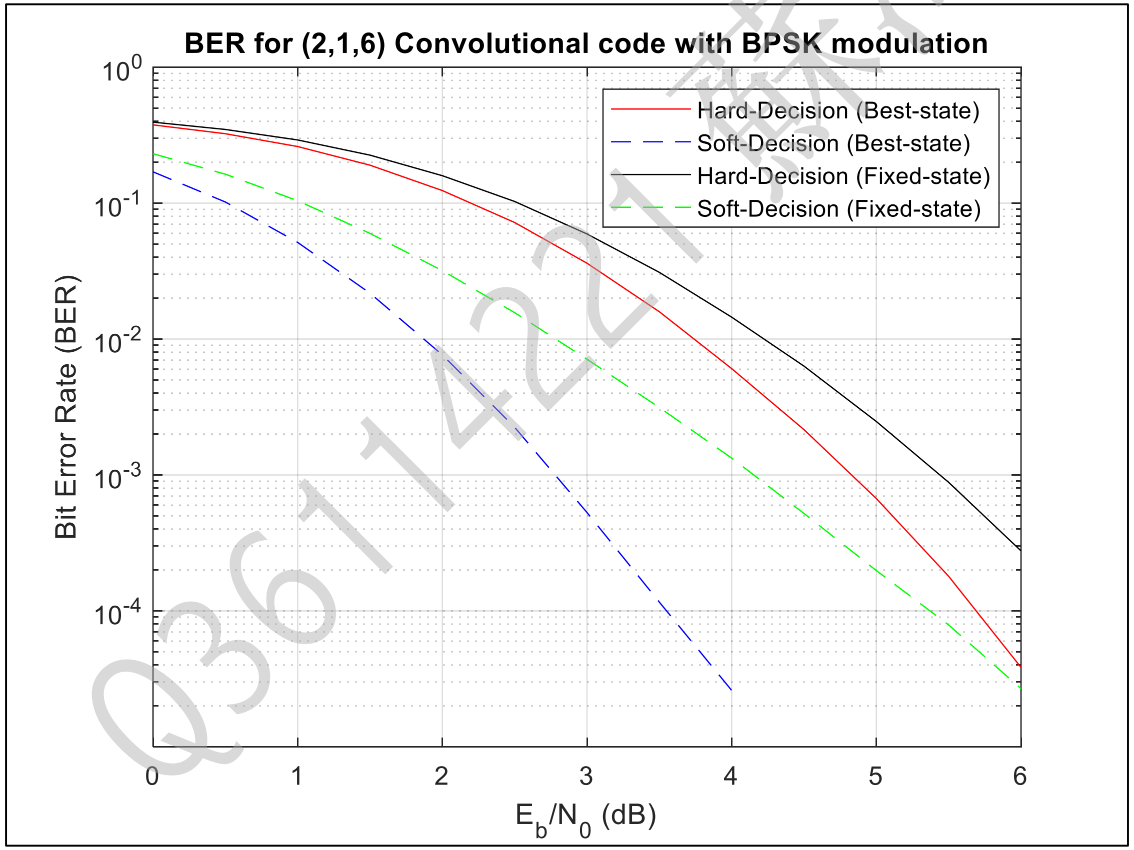
1. Hard-Decision (fixed-state)

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6
No. decoded bits	499968												
No. decided bit errors	197140.8	174134.2	145773.8	112672.8	79553.4	51382.2	29640	15468.8	7243.6	3159	1236.4	441	138.2
bit error rate (BER)	3.94E-01	3.48E-01	2.92E-01	2.25E-01	1.59E-01	1.03E-01	5.93E-02	3.09E-02	1.45E-02	6.32E-03	2.47E-03	8.82E-04	2.76E-04

2. Soft-Decision (fixed-state)

SNR (dB)	0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6
No. decoded bits	499968												
No. decided bit errors	115461.4	81602.4	52036.4	29928.4	15922.6	7817.2	3550.2	1567.2	665.2	260.4	99	39.4	13.4
bit error rate (BER)	2.31E-01	1.63E-01	1.04E-01	5.99E-02	3.18E-02	1.56E-02	7.10E-03	3.13E-03	1.33E-03	5.21E-04	1.98E-04	7.88E-05	2.68E-05

模擬圖



BER for (2,1,6) Convolutional code with BPSK modulation
Hard-Decision (Best-state)
Soft-Decision (Best-state)
Hard-Decision (Fixed-state)
Soft-Decision (Fixed-state)
Bit Error Rate (BER)
E_b/N_0 (dB)
10^0
10^{-1}
10^{-2}
10^{-3}
10^{-4}
0
1
2
3
4
5
6