

```
1
2 // *** 時頻數值計算(Precisely Time-Frequency Numerical Computations) ***
3
4 // 微分方程式： $M(t) * y''(t) + C(t) * y'(t) + K(t) * y(t) = 0$ 
5 // 空間維度有m個自由度、r階(Order)之齊次微分方程式，求得系統或狀態矩陣 A，
6 // 本求解法可對應於 Laplace、Fourier、Z Transform 或是 捲積積分法 等等。
7
8 // 系統矩陣 A 的特徵值與特徵向量，為系統的潛在特性，並在系統受到
9 // 外力時，才會顯現出來。
10
11 // 若要求得系統的訊號響應值[Signal Response]，應由實際量測的初始值或
12 // 是邊界值，求得複數係數向量 d。
13 // 再依據如下推導的公式求得。(初始值和邊界值分別參見App_6J和App_6M儲存庫)
14 //  $[y''(t) \mid y'(t)] = A * [y'(t) \mid y(t)]$ 。"|"是垂直合併運算子。
15 //  $[y'(t) \mid y(t)] = \text{Hexp}(D, Q, t) * d$ 。
16 //  $A * Q = Q * D \Rightarrow A = Q * D * Q_i$ 。
17 // D為複數特徵值矩陣，Q為複數特徵向量矩陣，Qi為Q之逆矩陣，Hexp(D, Q, t)和d分別
18 // 為複數矩陣和向量。
19 //  $[y'(t) \mid y(t)] = [y_h'(t) \mid y_h(t)] + [y_p'(t) \mid y_p(t)]$ 
20
21 using System;
22 using Matrix_0;
23
24 namespace ConsoleApp48
25 {
26     internal class Program
27     {
28         static void Main(string[] args)
29         {
30
31
32 // 空間維度有m個自由度。
33 int m = 3;
34 // 微分方程式有r個階度(Order)。
35 int r = 2;
36
37 // 建構初始矩陣 M、C、K、O、I
38 ReMatrix M = (new Zero(m)).GetMatrix;
39 ReMatrix C = (new Zero(m)).GetMatrix;
40 ReMatrix K = (new Zero(m)).GetMatrix;
41 ReMatrix O = (new Zero(m)).GetMatrix;
```

```
42 ReMatrix I = (new Iden(m)).GetMatrix;
43
44 // 建構初始系統矩陣 A、特徵值矩陣 D、特徵向量矩陣 Q。
45 ReMatrix A = (new Zero(m * r)).GetMatrix;
46 CxMatrix D = (new Zero(m * r)).GetMatrix;
47 CxMatrix Q = (new Zero(m * r)).GetMatrix;
48
49 // 狀態響應：加速度、速度，和變位。(Step = 1.0秒，共計 t = 20秒)
50 double step = 1.0;
51 int iRow = (int)(20 / step + 1);
52
53 // 建構時間軸上的儲存矩陣(時間t壹行和儲存矩陣m * r行。
54 int iColD = m * r + 1;
55 // 儲存矩陣為 iRow X iColD
56 CxMatrix CxVal = new CxMatrix(iRow, iColD);
57 ReMatrix ReVal = new ReMatrix(iRow, iColD);
58
59 for (int i = 0; i != iRow; i++)
60 {
61     double t = step * i;
62
63     // 建構實際的M、C、K矩陣。
64     M.Matrix[0, 0] = 19;
65     M.Matrix[0, 1] = -1.5;
66     M.Matrix[0, 2] = -2 + 13.3 * Math.Sin(0.85 * t);
67     M.Matrix[1, 0] = -1;
68     M.Matrix[1, 1] = 15;
69     M.Matrix[1, 2] = 0;
70     M.Matrix[2, 0] = -10 - 2.7 * Math.Cos(1.3 * t);
71     M.Matrix[2, 1] = -3;
72     M.Matrix[2, 2] = 27;
73     // End of M
74
75     C.Matrix[0, 0] = 35;
76     C.Matrix[0, 1] = -1 - 13.2 * Math.Sin(0.35 * t);
77     C.Matrix[0, 2] = -0.5;
78     C.Matrix[1, 0] = -1.5;
79     C.Matrix[1, 1] = 40;
80     C.Matrix[1, 2] = -1.5;
81     C.Matrix[2, 0] = -1.2 + 22.5 * Math.Cos(1.95 * t);
82     C.Matrix[2, 1] = -1.5;
83     C.Matrix[2, 2] = 75;
84     // End of Matrix C
85
86     K.Matrix[0, 0] = 60;
87     K.Matrix[0, 1] = -8;
88     K.Matrix[0, 2] = -2 - 332 * Math.Sin(1.37 * t);
```

```

89     K.Matrix[1, 0] = -16;
90     K.Matrix[1, 1] = 180;
91     K.Matrix[1, 2] = -120;
92     K.Matrix[2, 0] = -20;
93     K.Matrix[2, 1] = -100 + 579 * Math.Cos(0.24 * t);
94     K.Matrix[2, 2] = 300;
95     // End of Matrix K
96
97     // 隨時間變化的系統(狀態)矩陣 A , A 矩陣為6X6的實數矩陣( m = 3, r = 2)。
98     ReMatrix Mi = ~M;
99     A = (-1.0 * Mi * C) & (-1.0 * Mi * K);
100    A = A | (I & 0);
101
102    Console.WriteLine(" i = {0} t = {1} ", i, t);
103    Console.WriteLine("計算特徵值和特徵向量矩陣之前:");
104    // 隨時間變化的系統特徵值矩陣 D , 特徵向量 Q 。
105    D = (new EIG(A)).CxMatrixD;
106    Q = (new EIG(A)).CxMatrixQ;
107    Console.WriteLine(" ** 計算特徵值和特徵向量矩陣之後 : **");
108
109    // 將時間轉為複數值。
110    CxScalar cxScalar = new CxScalar(t, 0);
111    // 隨時間變化的特徵值矩陣(複數)。
112    CxVal[i, 0] = new CxMatrix(cxScalar);
113    CxVal[i, 1] = D[0, 0];
114    CxVal[i, 2] = D[1, 1];
115    CxVal[i, 3] = D[2, 2];
116    CxVal[i, 4] = D[3, 3];
117    CxVal[i, 5] = D[4, 4];
118    CxVal[i, 6] = D[5, 5];
119
120    // 隨時間變化的角頻率(實數值轉為矩陣)。
121    double[,] tMatrix = { { t } };
122    ReVal[i, 0] = (ReMatrix)tMatrix;
123    ReVal[i, 1] = D[0, 0].Im;
124    ReVal[i, 2] = D[1, 1].Im;
125    ReVal[i, 3] = D[2, 2].Im;
126    ReVal[i, 4] = D[3, 3].Im;
127    ReVal[i, 5] = D[4, 4].Im;
128    ReVal[i, 6] = D[5, 5].Im;
129
130 }
131
132 Console.WriteLine("\n*** 時間和特徵值(有六組), 共有七組複數值 ***");
133 Console.WriteLine("\n{0}\n\n", new PR(CxVal));

```

```
134
135 Console.WriteLine("\n *** 特徵值矩陣的虛數值即角頻率 ***\n");
136 Console.WriteLine("        時間 t        .... 六個角頻率 ");
137 Console.WriteLine("\n{0}\n", new PR(ReVal));
138
139 // 轉為序列方式, 使用python程式繪圖。
140 Console.WriteLine("\n時間序列:  t\n{0}\n", new PR4(ReVal, 0));
141 Console.WriteLine("\n角頻率序列:w0\n{0}\n", new PR4(ReVal, 1));
142 Console.WriteLine("\n角頻率序列:w1\n{0}\n", new PR4(ReVal, 2));
143 Console.WriteLine("\n角頻率序列:w2\n{0}\n", new PR4(ReVal, 3));
144 Console.WriteLine("\n角頻率序列:w3\n{0}\n", new PR4(ReVal, 4));
145 Console.WriteLine("\n角頻率序列:w4\n{0}\n", new PR4(ReVal, 5));
146 Console.WriteLine("\n角頻率序列:w5\n{0}\n", new PR4(ReVal, 6));
147
148
149     }
150 }
151 }
152
153
```