

計算機圖學與應用 6650

General N Dimension Arnold Transform Matrix.

授課教師：王宗銘

2022/12/14

1. 請以 python 程式語言撰寫 N 維 Arnold Transform Matrix, M_N 。
2. N 維 Arnold Transform Matrix 可由 N=2 擴展到 N=3, 接著擴展到 N=4 維, ... 如此類推到 N 維。擴展時需要讓使用者輸入 2 維 Matrix, 然後根據以下兩個步驟, 即可完成擴展。茲以 N=2 維擴展到 N=4 維為例。

1. 使用者輸入 2 個整數值, 例如 $a=1, b=1$, 程式據此構建 $M_2 = \begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}$ 。

2. 建構 3 個 matrices, $A_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 2 \end{bmatrix}$, $A_2 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 2 \end{bmatrix}$, $A_3 = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ 部分內容由 M_2 , 部分是對角線為 1 的其餘為 0 的行、列, 如紅色所示。

3. 使用矩陣相乘, 產出 $M_3 = A_1 \times A_2 \times A_3$

$$M_3 = A_1 \times A_2 \times A_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 2 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 3 & 2 \\ 3 & 4 & 4 \end{bmatrix}$$

4. 相同方式, 建構 4 個矩陣 $A_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 2 & 3 & 2 \\ 0 & 3 & 4 & 4 \end{bmatrix}$, $A_2 = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 2 & 0 & 3 & 2 \\ 3 & 0 & 4 & 4 \end{bmatrix}$,

$A_3 = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 2 & 3 & 0 & 2 \\ 0 & 0 & 1 & 0 \\ 3 & 4 & 0 & 4 \end{bmatrix}$, $A_4 = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 2 & 3 & 2 & 0 \\ 3 & 4 & 4 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$. 部分內容由 M_3 , 部分是對角線為 1 的其餘為 0 的行、列, 如紅色所示。

5. 使用矩陣相乘, 產出 $M_4 = A_1 \times A_2 \times A_3 \times A_4 = \begin{bmatrix} 4 & 5 & 5 & 1 \\ 28 & 36 & 35 & 6 \\ 67 & 86 & 84 & 14 \\ 110 & 141 & 138 & 24 \end{bmatrix}$, 其行

列式值 $\|M_4\| = 1$.

6. 請特別注意; 每個維度的行列值均維 1: $\|M_i\| = 1$ for $i \geq 2$

2. 繳交檔案

(1) python 程式，程式名稱：學號-07-GeneralARD.py。

(2) 請輸出 M_8 ，並比較論文所提之錯誤矩陣，如下圖所示。

K. Ganesan and K. Murali, “Image encryption using eight dimensional chaotic cat map,” *The European Physical Journal*, Special Topics, 223, pp. 1611-1322, 2014.

$$A = \begin{bmatrix} 1 & 7 & 33 & 125 & 403 & 1119 & 2591 & 4279 \\ 1 & 8 & 39 & 150 & 487 & 1356 & 3141 & 5182 \\ 1 & 7 & 34 & 130 & 421 & 1171 & 2712 & 4476 \\ 1 & 6 & 26 & 96 & 305 & 842 & 1948 & 3224 \\ 1 & 5 & 19 & 63 & 192 & 520 & 1200 & 2000 \\ 1 & 4 & 13 & 38 & 104 & 272 & 644 & 1056 \\ 1 & 3 & 8 & 20 & 48 & 112 & 256 & 448 \\ 1 & 2 & 4 & 8 & 16 & 32 & 64 & 128 \end{bmatrix}$$