# 動態規劃演算法簡介

盧奕璋／臺大電機系教授、電子所教授

相信大家應該都學過線性規劃（Linear Programming）這個方法。在可以利用線性規劃求解的題目中，通常符合一個標準的形式，其中包含有希望最大化或最小化的目標函數（Objective Function），以及幾種常見的限制條件（Constraints）。以下是一個簡單的例子：

Maximize $Z = 5x_1 + 7x_2$

subject to

$x_1 \le 5$
$2x_2 \le 9$
$3x_1 + 2x_2 \le 12$
$x_1 \ge 0$
$x_2 \ge 0$

維度低的線性規劃題目，可以用圖解法求得。對於較高維度的題目，也已經有寫好商用軟體（例如 Excel、Matlab 等）可以提供答案。

今天要討論的動態規劃（Dynamic Programming）演算法，其實也是用來計算最大值或最小值。只不過與線性規劃演算法最大的不同點，是在於利用動態演算法來求解的題目，不同問題之間的形式變化可以很大，因此要決定一個題目是否適合利用動態演算法求解，需要稍微思考一下。通常以動態演算法求解問題的流程，可歸納出以下幾個特點：

(1) 問題可以分解成前後幾個步驟（Stages），每個步驟有數個可能的開始狀態（States）。

(2) 每個步驟會進行一次決策過程，把目前步驟的開始狀態，經過計算得到對應下一個步驟的各種可能開始狀態。

(3) 對於一個給定的狀態，進行之後步驟的決策過程時不會再考慮先前步驟的決策過程。

(4) 決定如何得到完整的最佳解，可從最後一個步驟往回推出每個步驟的決策過程。

如果把上面的句子以數學的語言來描述如下：

假設總共分成有 N 個步驟；

把現今步驟標記為 n，n 為 1 到 N 的整數；

對於現今步驟 n，存在有狀態變數 $s_n$ 對應到可能的各種狀態；

對於現今步驟 n，存在有決策變數 $x_n$ 對應到可能的各種決策；

給定 $s_n$ 的情況下，最佳的決策為 $x_n^*$；

$f_n(s_n, x_n)$ 為步驟 n … N 對目標函數值的貢獻；

$f_n^*(s_n, x_n^*)$ 為最佳決策下，步驟 n … N 對目標函數值的貢獻；

有了上面的定義之後，我們就可以利用動態規劃法來找到最佳解（或者是說做出最佳的決定）。讓我們來試一個簡單的題目：假設某研究機構想要設計一部導航電腦供飛往土星的探測器使用，目前機構裡有三個研究團隊同時用不同的方法進行設計，經過專家的評估，三個團隊目前的失敗率分別是 0.4、0.5、0.8。假設三個團隊同時都失敗的機率是 (0.4)*(0.5)*(0.8) = 0.16，為了降低失敗率，高層決定從彗星小組借調兩位有經驗的工程師，但得決定把他們放入哪一個團隊。專家分析後發現，增加人手對於降低三個團隊的失敗率有不同的影響（如下表），那應該要怎麼分配會比較好？因為團隊和工程師的數目都不大，我們當然可以用窮舉法找出最佳解，但是讓我們試試動態規劃法的步驟。

| 新加入的工程師人數 | 失敗率 | | |
|---|---|---|---|
| | 團隊 | | |
| | 1 | 2 | 3 |
| 0 | 0.4 | 0.5 | 0.8 |
| 1 | 0.25 | 0.3 | 0.6 |
| 2 | 0.2 | 0.25 | 0.4 |

如果把分到三個團隊的決策過程看成是三個步驟，整個題目可表示成：

Minimize $\prod_{i=1}^{3} p_i(x_i) = p_1(x_1)\ p_2(x_2)\ p_3(x_3)$

subject to $\sum_{i=1}^{3} x_i = 2$

$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0$

從定義可以得到：$f_n(s_n, x_n) = p_n(x_n) \cdot \min \prod_{i=n+1}^{3} p_i(x_i)$ 與 $\sum_{i=n}^{3} x_i = s_n$。

其中：

$s_n$ 為分到第 n 到第 3 個團隊的工程師數目；

$x_n$ 為分到第 n 個團隊的工程師數目。

同時，如果定義 $f_n^*(s_n) = \min_{x_n = 0, 1, \cdots s_n} f_n(s_n, x_n)$，可以推得：

$$f_n(s_n, x_n) = p_n(x_n) \cdot f_{n+1}^*(s_n - x_n) \ 。$$

如此一來，整個遞迴關係就可表示成：

$$f_n^*(s_n) = \min_{x_n = 0, 1, \cdots s_n} \{ p_n(x_n) \cdot f_{n+1}^*(s_n - x_n) \}, n = 1, 2,$$

$$f_3^*(s_3) = \min_{x_s = 0, 1, \cdots s_n} p_3(x_3).$$

所以整個的求解過程，就可以整理如下：

n = 3:

| $s_3$ \ $x_3$ | $f_3(s_3, x_3) = p_3(x_3)$ | | | $f_3^*(s_3)$ | $x_3^*$ |
|---|---|---|---|---|---|
| | 0 | 1 | 2 | | |
| 0 | 0.8 | X | X | 0.8 | 0 |
| 1 | X | 0.6 | X | 0.6 | 1 |
| 2 | X | X | 0.4 | 0.4 | 2 |

n = 2:

| $s_2$ \ $x_2$ | $f_2(s_2, x_2) = p_2(x_2) \cdot f_3^*(s_2 - x_2)$ | | | $f_2^*(s_2)$ | $x_2^*$ |
|---|---|---|---|---|---|
| | 0 | 1 | 2 | | |
| 0 | 0.4 | X | X | 0.4 | 0 |
| 1 | 0.3 | 0.24 | X | 0.24 | 1 |
| 2 | 0.2 | 0.18 | 0.2 | 0.18 | 1 |

n = 1:

| $s_1$ \ $x_1$ | $f_1(s_1, x_1) = p_1(x_1) \cdot f_2^*(s_1 - x_1)$ | | | $f_1^*(s_1)$ | $x_1^*$ |
|---|---|---|---|---|---|
| | 0 | 1 | 2 | | |
| 0 | X | X | X | X | X |
| 1 | X | X | X | X | X |
| 2 | 0.072 | 0.06 | 0.08 | 0.06 | 1 |

由表中可推知，失敗率最低的最佳解應為 $x_1 = 1, x_2 = 1, x_3 = 0, f^* = 0.06$。因此，兩位工程師應該分別分配至第一及第二團隊。在這邊我們是使用製表的方式來進行整理，在實際應用上則會利用編寫程式的方法來進行。

電機資訊領域使用動態規劃法對問題求最佳解，基本上就是類似上述的方法，只不過問題的規模更大，而形式也不一定是單純相乘[1]（如上面的例題）或是單純相加[2]。這些應用包含通訊編碼的解碼計算、字串的比對搜尋程序等等，將來有機會再跟大家做深入的介紹。

---

1. $f_n^*(s_n) = \min\limits_{x_n = 0, 1, \cdots s_n} \{p_n(x_n) \cdot f_{n+1}^*(s_n - x_n)\}$ 或是 $f_n^*(s_n) = \min\limits_{x_n = 0, 1, \cdots s_n} \{p_n(x_n) \cdot f_{n+1}^*(s_n - x_n)\}$

2. $f_n^*(s_n) = \min\limits_{x_n = 0, 1, \cdots s_n} \{p_n(x_n) + f_{n+1}^*(s_n - x_n)\}$ 或是 $f_n^*(s_n) = \min\limits_{x_n = 0, 1, \cdots s_n} \{p_n(x_n) + f_{n+1}^*(s_n - x_n)\}$