遞迴函數的求解技巧

許介圭

大葉大學　通識與計算機工程學系

引言

假定我們問張三他最大的女兒今年幾歲，他告訴我們：「我最大的女兒比我的第二個女兒大五歲，我的第二個女兒又比我唯一的兒子大三歲，而我兒子今年 11 歲。」雖然張三沒有直接回答我們的問題，卻給了我們足夠的資訊來推算出他最大的女兒今年是 19 歲。

也許讀者覺得上面的例子不大自然，那麼再看下面這個可能發生在日常生活中的例子：假定我們想要去動物園，不曉得它的位置而求助於路人李四，他告訴我們：「從這裡往前走約一百公尺會在路口看到一間警察局，在那個路口左轉，往前走到第三個紅綠燈會在路口看到一間郵局，在那個路口右轉再走約兩百公尺就到了。」雖然李四沒有直接告訴我們動物園的地址，卻提供了足夠的資訊讓我們可以到達目的地。

讀者不難從上面兩個例子看出筆者想要表達的概念；雖然張三和李四沒有直接回答我們的問題，從他們的回答中我們還是可以獲取想要的資訊。第一個例子中，我們透過算出張三其他子女的年齡來求得他的大女兒的年齡；第二個例子中，我們先到達其他的地點（警察局和郵局）以便到達真正想去的地點；這兩個例子中，張三和李四回答問題的方式都用到了數學上一個稱做遞迴的概念。

什麼是「遞迴」？

「遞迴」（recurrence），或稱「遞迴關係」（recurrence relation），是指將一個函數在某個點（通常只討論整數點）的函數值以此函數在其他點的函數值來表示的方式，例如：$f(n) = f(n/2)+1$，$f(n) = f(n-1)+f(n-2)$，$f(n) = 3f(\sqrt{n})+2^{n-1}$，……等。

我們以上面的第一個式子為例說明。為了討論方便，假設我們限制第一個式子中的n是 2 的正整數次方（以便讓 2 可整除 n），則根據此式，函數 f 在 n = 8 時的函數值等於 n = 4 時的函數值再加 1 ，而 n = 4 時的函數值又等於 n = 2 時的函數值再加 1 。當然，上面的函數定義並不完整，因為式子中只表明了 f (n) 與 f (n / 2) 兩個函數值之間的關係，我們無法得知函數 在任何一點的函數值是多少，因此如果要將某個函數以遞迴的方式明確地定義，定義中必須包含該函數在至少一個點的函數值，以便讓我們可以由此已知的函數值出發，推出該函數在其他點的值；此已知的函數值通常稱做「邊界條件」（boundary condition）。上面的例子中，如果我們加上一個邊界條件：

$$\begin{cases} f(n) = f(n/2)+1 \\ f(1) = 1 \end{cases}$$

則

$$f(1) \quad = 1$$

$$f(2) = f(1)+1 = 2$$
$$f(4) = f(2)+1 = 3$$
$$f(8) = f(4)+1 = 4$$
$$\cdots$$

只要 n 是 2 的某個非負整數次方，f (n) 即可求得。

由上面的幾個不同的 n 及對應的函數值，我們不難「猜」出似乎有 $f(2^k)=k+1$ 的

，

Basis step:

$f(2^0)=0+1=1$ 與已知的邊界條件相符。

Inductive step:

假設對某個非負整數 k ，$f(2^k)=k+1$ 成立，

$$f(2^{k+1}) = f(2^k)+1 \text{〈根據遞迴關係〉}$$
$$= (k+1)+1 \text{〈根據假設〉}$$

因此根據數學歸納法得證：對所有 $k \geq 0$ ，$f(2^k)=k+1$ ，也就是當 n 為 2 的任意非負整數次方，函數 f(n) 的一般式為 $f(n)=\log_2 n+1$ 。

一 的

的一般式為 的「解」(solution)。

更多的遞迴求解技巧

以下，我們再透過例題介紹幾個由遞迴關係求出函數一般式的技巧。

例題一：假設 n 是 4 的任意非負整數次方，而且

$$f(n)=\begin{cases}27 & n=1\\ 2f(n/4)+n & n>1\end{cases}$$

f (n) 的一般式。

解：在前面的例子中，我們是由 n = 1 時的函數值 f (n) 開始，由小而大地推導出 n = 2, 4, 8 時的函數值並設法由其中觀察出 n 與 f (n) 的關係；在處理許多遞迴問題時，以由大而小的相反方向來推導也常常是一個不錯的方法，也就是由 f (n) 出發：

$$f(n) = 2f(n/4)+n$$
$$= 2(2f(n/4^2)+n/4)+n$$

整理後可得

$$f(n)=2^2 f(n/4^2)+n/2+n$$

$f(n/4^2)$ ：

$$f(n)=2^2(2f(n/4^3)+n/4^2)+n/2+n$$

$$f(n)=2^3 f(n/4^3)+n/2^2+n/2^1+n/2^0$$

「猜」出以下的規則：

$$f(n)=2^k f(n/4^k)+n/2^{k-1}+n/2^{k-2}+\cdots+n/2^1+n/2^0$$

由於 n 是 4 的非負整數次方，因此當推導至 $n/4^k=1$（即 $n=4^k$ ）時，$f(n/4^k)=27$ 為已

$$f(n) = 2^k f(1)+n(1/2^{k-1}+1/2^{k-2}+\cdots+1/2^0)$$
$$= 27\times 2^k + n(2-1/2^{k-1})$$
$$= 27\sqrt{n}+n(2-2/\sqrt{n})$$
$$= 2n+25\sqrt{n}\ .$$

，

証實。

例題二：假設 n 是任意非負整數且

$$f(n)=\begin{cases}5 & n=0\\ f(n-1)+n & n>0\end{cases}$$

求 f (n)的一般式。

解：仿例題一將 n 值由大而小推導的做法：

$$f(n) = f(n-1)+n$$
$$= f(n-2)+(n-1)+n$$
$$= f(n-3)+(n-2)+(n-1)+n$$
$$= f(n-k)+(n-k+1)+(n-k+2)+\cdots+n$$
$$= f(n-n)+1+2+3+\cdots+n$$

$= 5 + n(n+1)/2$

f (n) 的一般式為 f (n) = 5 + n (n + 1) / 2；同樣的，這可由數學歸納法予以証實。

例題三：假設 $n = 2^{2^k}$（　是任意非負整數）且

$$f(n) = \begin{cases} 1 & n = 2 \\ f(\sqrt{n}) + 3 & n > 2 \end{cases}$$

求 f (n)的一般式。

解：許多看似複雜的遞迴問題常可透過變數的代換或是定義新的函數而被轉換成較容易處理的問題。以這個題目來說，之前的做法還是可行，另一個做法則是令 $m = \log_2 n$ ，則

$$f(2^m) = \begin{cases} 1 & m = 1 \\ f(2^{m/2}) + 3 & m > 1 \end{cases}$$

一個　　$g(m) = f(2^m)$ ，則

$$g(m) = \begin{cases} 1 & m = 1 \\ g(m/2) + 3 & m > 1 \end{cases}$$

這個函數與原來的函數 f (n)相比很明顯容易處理多了，仿照前面的例題，很容易可解得 $g(m) = 3\log_2 m + 1$ ，因此

$$\begin{aligned} & f(2^m) = 3\log_2 m + 1 \\ \Rightarrow\ & f(n) = 3\log_2 \log_2 n + 1 \end{aligned}$$

此即為 f (n)的一般式（可由數學歸納法予以証實）。

另解：定義一個新的函數 $g(n) = f(2^{2^n})$ ，則

$$f(n) = \begin{cases} 1 & n = 2 \\ f(\sqrt{n}) + 3 & n > 2 \end{cases}$$

$$\Rightarrow\ g(n) = \begin{cases} 1 & n = 0 \\ g(n-1) + 3 & n > 0 \end{cases}$$

的

$$\begin{aligned} g(n) &= g(n-1) + 3 \\ &= g(n-2) + 2 \times 3 \\ &= g(n-3) + 3 \times 3 \\ &= g(n-k) + k \times 3 \\ &= g(n-n) + n \times 3 \end{aligned}$$

因此

$$\begin{aligned} & g(n) = 3n + 1 \\ \Rightarrow\ & f(2^{2^n}) = 3n + 1 \\ \Rightarrow\ & f(n) = 3\log_2 \log_2 n + 1 \end{aligned}$$

得到與前一個做法相同的結果。

例題四：假設 $n = 2^{2^k}$（　是任意非負整數）且

$$f(n) = \begin{cases} 2 & n = 2 \\ \sqrt{n} f(\sqrt{n}) + 3n & n > 2 \end{cases}$$

求 f (n)的一般式。

解：當 n > 2 時， $f(n) = \sqrt{n} f(\sqrt{n}) + 3n$ ，若等　　時　　n ，得

$$\frac{f(n)}{n} = \frac{f(\sqrt{n})}{\sqrt{n}} + 3$$

如果我們定義一個新的函數 $g(n) = f(n)/n$ ，

$$g(n) = \begin{cases} 1 & n = 2 \\ g(\sqrt{n}) + 3 & n > 2 \end{cases}$$

這個函數我們在例題三已經處理過，因此，

$$\begin{aligned} & g(n) = 3\log_2 \log_2 n + 1 \\ \Rightarrow\ & f(n) = 3n\log_2 \log_2 n + n \end{aligned}$$

此即為 f (n)一般式（可由數學歸納法予以証實）。

接下來我們將嘗試用較一般性的原則來處理某一類型的遞迴問題。假設某遞迴函數以如下的方式定義：

$$f(n) = \begin{cases} f(1) & n = 1 \\ g(n) f(n-1) + h(n) & n > 1 \end{cases}$$

其中， f (1)為常數， g 和 h 都是 n 的函數，且 g (n) 在 $n > 1$ 時不為 0 。首先，我們試著將

(n) 仿照前面的做法推導幾步看看：

$$\begin{aligned} f(n) &= g(n)f(n-1)+h(n) \\ &= g(n)g(n-1)f(n-2)+g(n)h(n-1)+h(n) \\ &= g(n)g(n-1)g(n-2)f(n-3) \\ &\quad + g(n)g(n-1)h(n-2)+g(n)h(n-1)+h(n) \end{aligned}$$

項數越來越多且越來越複雜，因為牽涉到越來越多的函數 g 的連乘，也許讀者會想：如果能想個辦法將所有的 g 消除該有多好！這其實正是解題的契機，因為函數 g 真的可以透過定義新的函數而完全被消除！

我們注意到如果上面的推導持續下去，最後會因為碰到邊界條件而中止，此時 $f(n)=g(n)g(n-1)g(n-2)\cdots g(3)g(2)f(1)+\cdots$

因此如果我們定義一個新的函數 $f_1(n)$ ：

$$f_1(n)=\frac{f(n)}{g(n)g(n-1)g(n-2)\cdots g(3)g(2)}$$

n > 1 ，

$$\begin{aligned} & f(n)=g(n)f(n-1)+h(n) \\ \Rightarrow\quad & g(n)g(n-1)g(n-2)\cdots g(2)f_1(n) \\ & = g(n)g(n-1)g(n-2)\cdots g(2)f_1(n-1)+h(n) \\ \Rightarrow\quad & f_1(n)=f_1(n-1)+\frac{h(n)}{g(n)g(n-1)\cdots g(2)} \end{aligned}$$

上式中，除了等號右邊的最後一項較複雜外，函數 f_1 的推導過程將因為它在等號右邊的 數為 1 而被大大地簡化；事實上，不難看出，對所有 n ≧ 1 ，

$f_1(n)=f_1(1)+\sum_{i=2}^{n}\frac{h(i)}{g(i)g(i-1)\cdots g(2)}$（當 n = 1 時，$\sum_{i=2}^{1}(\ldots)$ 為 0）

，

法。

例題五：假設 n 是任意正整數且

$$f(n)=\begin{cases}0 & n=1\\ \frac{n+1}{n}f(n-1)+\frac{2(n-1)}{n} & n>1\end{cases}$$

求 f (n)的一般式。

解：對照以上說明，得知

$$g(n)=\frac{n+1}{n}\text{且}\quad h(n)=\frac{2(n-1)}{n}$$

定義新函數 $f_1(n)$ 使得

$$\begin{aligned} f(n) &= g(n)g(n-1)g(n-2)\cdots g(3)g(2)f_1(n) \\ &= \frac{n+1}{n}\times\frac{n}{n-1}\times\frac{n-1}{n-2}\times\cdots\times\frac{4}{3}\times\frac{3}{2}\times f_1(n) \\ &= \frac{n+1}{2}f_1(n) \end{aligned}$$

$$\frac{h(n)}{g(n)g(n-1)\cdots g(2)}=\frac{2(n-1)}{n}\times\frac{2}{n+1}=\frac{4(n-1)}{n(n+1)}$$

因此

$$f_1(n)=\begin{cases}0 & n=1\\ f_1(n-1)+\frac{4(n-1)}{n(n+1)} & n>1\end{cases}$$

不難看出

$$f_1(n)=4\sum_{i=2}^{n}\frac{i-1}{i(i+1)}$$

因此

$$\begin{aligned} f(n)=\frac{n+1}{2}f_1(n) &= \frac{n+1}{2}\times 4\sum_{i=2}^{n}\frac{i-1}{i(i+1)} \\ &= 2(n+1)\sum_{i=2}^{n}\frac{i-1}{i(i+1)} \end{aligned}$$

f (n)的一般式，如果還想再簡化，可利用部分分式的技巧：

$$f(n)=2(n+1)\sum_{i=2}^{n}\frac{i-1}{i(i+1)}=2(n+1)\sum_{i=2}^{n}\left(\frac{2}{i+1}-\frac{1}{i}\right)$$

經過一番整理與化簡，可得

$$f(n)=2(n+1)\left(1+\frac{1}{2}+\frac{1}{3}+\cdots+\frac{1}{n}\right)-4n$$

同樣的，這可由數學歸納法予以証實。

結語

用遞迴的方式定義的函數在計算機演算法的設計與分析上扮演了重要的角色，尤其與遞迴程式（recursive programs）有著密切的關係；另外，在某些計數（counting）問題中，遞迴關係也是一項有力的工具，可大大簡化思考的過程。本文限於篇幅，僅就少數幾個類型的遞迴關係作了簡單的介紹，將來有機會將為讀者介紹其他類型的遞迴關係求解的技巧。

參考資料

Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest, Introduction to Algorithms, McGraw-Hill, 1990.

C. L. Liu, Elements of Discrete Mathematics, 2nd edition, McGraw-Hill, 1985.

Gregory J. E. Rawlins, Compared to What? W.H. Freeman and Company, 1992.

（上承第 35 頁）

(1991): Principles of Neural Science 3rd ed., Learning and Memory. p. 997. Elsevier Science Publishing Co., Inc.

Richter D. (1970): Aspects of Learning and Memory. William Heinemann Medical Books Limited .

Young, J.Z. (1996): The Memory System of the Brain. Uni. California Press, Berkeley, Los Angeles

伊勢村壽三等 12 人(1967) : 腦與神經系. 學習、記憶的機制, p.165 岩波書店 東京

科學朝日雜誌(1994) : 腦研究最前線－1. 情緒、記憶與局部性神經細胞 p.120 東京

科學朝日雜誌(1994): 腦研究最前線－ 2. 記憶之機制 p.120 東京

島津 浩等 12 人 (1976) : 高次腦機能與中樞規劃. 腦聯合區之機能., 產業圖書 東京 p. 145

時實利彥(1991): 人類. 記憶., p.100 岩波書店 東京

時實利彥(1998): 腦. 大腦邊緣系., p.127 岩波書店 東京