第 5卷第 1期 南京师范大学学报(工程技术版) Vol 5 No 1
2005年 3月 JOURNAL OF NANJING NORMAL UNIVERSITY (ENGINEERING AND TECHNOLOGY) Mar 2005

用母函数理论分析递归算法的时间复杂度

方贤进, 潘地林, 管建军

(安徽理工大学 计算机系, 安徽 淮南 232001)

[摘要] 对算法进行时间复杂度分析是算法分析与研究的重要内容, 而对递归算法分析其时间复杂度时往往比较困难. 提出了用组合数学中的母函数与递推关系理论来分析一些特殊的递归算法的时间复杂度, 并同时得出三个推论, 在算法的分析与研究方面具有一定的参考价值.

[关键词] 时间复杂度, 递归, 母函数

[中图分类号] TP301.6, [文献标识码] A, [文章编号] 1672-1292-(2005)01-0092-03

Analyzing the Time Complexity of Recursion Algorithm by Applying Generation Function Theory

FANG Xianjin, PAN Dilin, GUAN Jianjun

(Department of Computer, Anhui University of Science and Technology, Anhui Huainan 232001, China)

Abstract Analyzing the time complexity of one algorithm is very important to the analysis of and research on the algorithm. It is usually difficult to analyze the time complexity of a recursive algorithm. In this paper, the theories of generation function and recursive relation in combinatorics is used to analyze the time complexity of some special recursive algorithm, with three conclusions reached simultaneously, which is of valuable reference to the analysis and research on algorithm.

Key words time complexity, recursion, generation function

0 引言

一个程序在计算机上运行时所耗费的时间取决于对源程序进行编译所需时间、计算机执行每条指令所需时间、程序中指令重复执行的次数.

前两条依赖于实现算法的计算机软件、硬件系统, 亦即依赖于实现算法所用语言的编译程序的性能和计算机本身的速度. 因此习惯上常常用重复执行次数最多的语句频度 (Frequency count) 来分析算法的时间复杂度, 记为 $t(n)$, 其中 n 为问题的规模或大小. 例如对 n 个存放于数组 $a[n]$ 中的数进行选择法排序的算法:

```
for( i= 0; i< n- 1; i+ + )            //共进行 n-1趟排序
  {   k= i;                            //n-1
      for( j= i+ 1; j< n; j+ + )       //n-i-1
         if( a[ j] < a[ k] ) k= j;     //第 i趟需进行 n-i-1次比较, 该语句频度最大
            if( k! = i) { t= a[ k]; a[ k] = a[ i]; a[ i] = t; }
                                       //进行 n-1次比较和交换
  }
```

该算法的时间复杂度为:

$$t(n) = \sum_{i=0}^{n-2}(n-i-1) = \frac{(n-1)n}{2} = \frac{1}{2}(n^2-n) = O(n^2).$$

对一个算法进行时间复杂度的分析方式有多种, 但有时候一个算法的时间复杂度并不象上面的例子那样可以轻易地分析出来. 也许对一个 for循环、对一个单一语句的时间复杂度可以马上分析出来, 但遇到递归算法时, 则无法轻易地分析出其时间复杂度, 而递归算法又是程序设计时经常采用的

收稿日期: 2004-01-10
作者简介: 方贤进 (1970-), 讲师, 主要从事算法分析与计算、计算机网络等方面的教学与研究. E-mail: xjfang@aust.edu.cn

有效手段. 如求解 n 阶 hanoi塔问题的递归算法:

```
void Hanoi( int n, char one, char two, char three)
//将 n个盘从 one座借助 two座, 移动到 three座
{ if( n= = 1) printf( "% c→ % c", one, three);
  else
  {   hanoi( n- 1, one, three, two);
      printf( "% c→ % c", one, three);
      hanoi( n- 1, two, one, three);
  }
}
```

类似的递归算法还有很多, 例如求 Fibonacci 数列前 n 项的递归算法等. 对于这些递归算法, 可以利用组合数学中的母函数与递推关系理论来分析其时间复杂度.

1 利用母函数与递推关系理论分析递归算法的时间复杂度

1.1 相关概念

(1) 母函数

对于序列 $a_0, a_1, a_2, \cdots$, 构造一函数 $G(x) = a_0 + a_1x + a_2x^2 + \cdots\cdots$, 则称函数 $G(x)$ 是序列 $a_0, a_1, a_2, \cdots$ 的母函数. 若求得序列的母函数, 则该序列随之确定, 反之亦然. 序列 $a_0, a_1, a_2, \cdots$ 可记为 $\{a_n\}$ 或 $a(n)$.

(2) k 阶常系数线性递推关系式与特征多项式

若母函数 $G(x) = a_0 + a_1x + a_2x^2 + \cdots\cdots$, 所对应的序列 $\{a_n\}$ 满足 $a_n + C_1a_{n-1} + C_2a_{n-2} + \cdots + C_ka_{n-k} = 0$, 并且有初始条件 $a_0 = d_0, a_1 = d_1, \cdots, a_{k-1} = d_{k-1}$, $C_1, C_2, \cdots, C_k$ 及 $d_0, d_1, \cdots, d_{k-1}$ 是常数, $C_k \neq 0$, 则称 $a_n + C_1a_{n-1} + C_2a_{n-2} + \cdots + C_ka_{n-k} = 0$ 为序列 $\{a_n\}$ 的 k 阶常系数线性递推关系.

对应于 k 阶常系数线性递推关系的多项式 $C(x) = x^k + C_1x^{k-1} + \cdots + C_{k-1}x + C_k$ 称为序列 $\{a_n\}$ 的特征多项式.

笔者根据母函数与递推关系理论 (见文献 [4]), 在分析一些递归算法的时间复杂度时, 总结出下列推论:

1.2 推论 1及其应用

设某一递归算法时间复杂度函数为 $t(n)$, 如果其 k 阶常系数递推关系式所对应的特征多项式 $C(x)$ 有 k 个不同的根 $\beta_1, \beta_2, \cdots \beta_k$, 则 $t(n)$ 的解为:

$t(n) = A_1\beta_1^n + A_2\beta_2^n \cdots + A_k\beta_k^n$, 其中 $A_1, A_2 \cdots A_k$ 为 k 个待定常数.

例如, 对上面 n 阶 hanoi塔问题的递归算法进行分析. 不妨设 $h(n)$ 表示将 n 个盘子从 one座移动到 three座所需的转移次数亦即 hanoi问题算法的时间复杂度. 根据算法先把前面 $n-1$ 个盘子转移到 two座上 (移动次数为 $h(n-1)$ 次), 然后把第 n 个盘子转移到 three座上 (移动次数为 1次); 最后再一次将 two座上的 $n-1$ 个盘子转移到 three座上 (移动次数为 $h(n-1)$ 次). 则显然有:

$$h(n) = 2h(n-1) + 1, \text{其中 } h(1) = 1 \qquad (1)$$

同样有:

$$h(n-1) = 2h(n-2) + 1 \qquad (2)$$

(1)式 − (2)式即可得到 2阶 (即 $k = 2$) 常系数线性递推关系式:

$$h(n) - 3h(n-1) + 2h(n-2) = 0$$

根据母函数与递推关系理论, 序列 $h(n)$ 所对应的特征多项式为 $C(x) = x^2 - 3x + 2$, 其有两个不同的根 $x_1 = 1, x_2 = 2$, 则 $h(n) = A_1x_1^n + A_2x_2^n = A_1 + A_2 2^n$, 其中 A_1, A_2 为待定常系数.

根据 hanoi问题, 显然有初值 $h(1) = 1, h(2) = 3$, 于是求得 $A_1 = -1, A_2 = 1$, 故 $h(n) = 2^n - 1$. 说明 n 阶 hanoi塔问题递归算法的时间复杂度为 $O(2^n)$.

按照同样的方法, 读者可自行分析用递归算法求 Fibonacci数列 n 项的时间复杂度为:

$$t(n) = \frac{\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n}{\sqrt{5}}.$$

1.3 推论 2及其应用

设某一递归算法时间复杂度为 $t(n)$, 其常系数线性递推关系式所对应的特征多项式 $C(x)$ 有 k 重根 β_1, 则递推关系的解即 $t(n)$ 对应于 β_1 的项为 $(A_0 + A_1n + \cdots A_{k-1}n^{k-1})\beta_1^n$, 其中 $A_0, A_1, \cdots A_{k-1}$ 为 k 个待定常数. 特征多项式 $C(x)$ 的非重根所对应的项按推论 1中的方法求解.

例如, 分析用递归算法求下列函数值时的时间复杂度 $t(n)$ (算法简略).

$\mathrm{func}(n) = 5\mathrm{func}(n-1) - 8\mathrm{func}(n-2) + 4\mathrm{func}(n-3)$, 并给定初值 func(1)、func(2)、func(3).

由函数的递推关系我们得到算法的时间复杂度 $t(n)$ 的线性常系数递推关系:

$$t(n) - 5t(n-1) + 8t(n-2) - 4t(n-3) = 0,$$

显然有初值 $t(0) = 0, t(1) = 1, t(2) = 1$.

则根据母函数与递推关系理论得到线性常系

数递推关系式的特征多项式: $C(x)=x^3-5x^2+8x-4$, 该特征多项式有3个根 $x_{1,2}=2$(二重根), $x_3=1$, 则递推关系式的解即 $t(n)$为:

$$t(n)=(A_0+A_1n)\cdot 2^n+A_2\cdot 1^n,$$ 其中 A_0, A_1, A_2 为待定常系数, 根据初值求得 $A_0=3$, $A_1=-1$, $A_2=-3$, 所以该递归算法的时间复杂度 $t(n)=3\cdot 2^n-n\cdot 2^n-3$.

1.4 推论3

设某一递归算法时间复杂度为 $t(n)$, 其递推关系式所对应的特征多项式 $C(x)$有不同的复根, 此时可按照"推论1"的办法处理. 不过复数有它的特点, 假如特征多项式 $C(x)$的两个共轭复根 x_1, x_2 可化成下列形式: $x_1=\rho(\cos\theta+i\sin\theta)$, $x_2=\rho(\cos\theta-i\sin\theta)$. 例如将特征多项式 $C(x)=x^2+x+1$的两个复根 $x_1=(-1+\sqrt{3}i)/2$, $x_2=(-1-\sqrt{3}i)/2$可写成:

$$x_1=\cos\frac{2}{3}\pi+i\sin\frac{2}{3}\pi,$$

$$x_2=\cos\frac{2}{3}\pi-i\sin\frac{2}{3}\pi \qquad (\rho=1).$$

此种情况下, 递推关系式的解即递归算法的时间复杂度可根据推论1有:

$$t(n)=k_1x_1^n+k_2x_2^n=k_1\rho^n(\cos n\theta+i\sin n\theta)+k_2\rho^n(\cos n\theta-i\sin n\theta)=\rho^n(k_1+k_2)\cos n\theta+i\rho^n(k_1-k_2)\sin n\theta$$

k_1+k_2和 $i(k_1-k_2)$仍然是待定常数, ρ, θ可由复根求得. 令 $A=k_1+k_2$, $B=i(k_1-k_2)$得到:

$$t(n)=A\rho^n\cos n\theta+B\rho^n\sin n\theta$$

应用实例由于篇幅所限, 作者在此不再赘述.

2 结论

在分析递归算法的时间复杂度 $t(n)$时, 如果 $t(n)$的递推关系式是一个常系数线性递推关系式, 则可以利用母函数与递推关系理论求出其时间复杂度.

[参考文献]

[1] 严蔚敏, 吴伟民. 数据结构[M]. 北京: 清华大学出版社, 1991. 303-305.

[2] 谭浩强. C程序设计[M]. 第2版. 北京: 清华大学出版社, 2003. 160-163.

[3] 黄国瑜, 叶乃箐. 数据结构(C语言版)[M]. 北京: 清华大学出版社, 2001. 13-18

[4] 卢开澄. 组合数学[M]. 北京: 清华大学出版社, 2000. 48-79.

[5] 杨骅飞, 王朝瑞. 组合数学及其应用[M]. 北京: 北京理工大学出版社, 1992. 40-48

[责任编辑: 刘健]

(上接第75页)

[参考文献]

[1] 王炳锡. 语音编码[M]. 西安: 西安电子科技大学出版社, 2002. 257-274

[2] Jamrozik M, Gowdy J. Enhanced quality modified multiband excitation model at 2400 bps. Bringing Together Education, Science and Technology[J]. Proceedings of the IEEE, 1996. 223-226.

[3] Jamrozik M, Gowdy J. Modified multiband excitation model at 2400 bps[J]. Proceedings of ICASSP-97, 2: 1603-1606.

[4] Yu W M E, Chan C F. Multiband excitation coding of speech at 2.0 kbps [J]. Proceedings of ISSIPNN-94, 2: 559-562

[5] Xu Peixia, Chen Zhou, Liu Wenfei. Wavelet analysis based multiband excited vocoder[A]. TENCON 93. Proceedings Computer, Communication, Control and Power Engineering[C]. 1993 IEEE Region 10 Conference on, 1993, (2): 349-352.

[6] Griffith D W, Lim J S. Multiband excitation vocoder. Acoustics, Speech, and Signal Processing [J]. IEEE Transactions on, 1988, 36(8): 1223-1235.

[7] 杨行峻, 迟惠生. 语音信号数字处理[M]. 北京: 电子工业出版社, 1995. 265-283.

[责任编辑: 刘健]