Algoritmos y Estructuras de Datos Avanzadas

Análisis Matemático de Algoritmos Recursivos

Prof.: Dr. Pedro A. Rodríguez[1]

[1] Departamento de Sistemas de Información
Departamento de Ciencias de la Computación y TI
Universidad del Bio-Bio

Ecuaciones de recurrencia

Ecuaciones de recurrencia

Se pueden considerar como técnicas avanzadas de conteo

Definition

- Una sucesión es una función $f : N \to A$.
- Para indicar la imagen en A se emplea a_n.
- Una sucesión se denota por $a_0, a_1, a_2, \ldots$
- A los elementos $a_0, a_1, a_2, \ldots$ se les llama términos de la suceción.
- a_n es el término general.

Ecuaciones de recurrencia

Example

$0, 1, 1, 2, 3, 5, 8, 13, \ldots$
Ecuación de recurrencia: $a_n = a_{n-1} + a_{n-2}$, si $n > 1$.

Una expresión en la que el término general de la sucesión se escribe en función de algunos términos anteriores se llama **ecuación de recurrencia**.

Una ecuación de recurrencia no determina de manera única una sucesión. Para ello es necesario conocer algunos términos de la sucesión, los que se denominan **condiciones de borde, de frontera o condiciones iniciales**. Por ejemplo, $a_n = a_{n-1} + a_{n-2}$, si $n > 1$, $a_0 = 0$ y $a_1 = 1$.

Ecuaciones de recurrencia

Example

$a_n = 2a_{n-1}, n \geq 1,$
si $a_0 = 1$ tenemos la sucesión: $1, 2, 4, 8, 16 \ldots$
si $a_1 = 3$ tenemos la sucesión: $3, 6, 12, 24, 48 \ldots$

Para obtener un término de la sucesión en forma recurrente, es necesario obtener todos los términos anteriores. **Esto no es práctico**. Por ejemplo a_{1000} en la serio de Fibonacci.
Interesa obtener una expresión que en la que el término general de la sucesión dependa sólo de la posición que ocupa (n) y no de términos anteriores. A esta expresión se le llama **solución de la ecuación de recurrencia.**

Ecuaciones de recurrencia

Example

$a_n = 3a_{n-1}, n \geq 1, a_0 = 5$
$a_0 = 5$
$a_1 = 3a_0 = 3.5 = 15$
$a_2 = 3a_1 = 3.5 = 3.3.5 = 45$
$\ldots$
$a_n = 3^n 5$

En la solución general hay dependencia sólo de n y no de términos anteriores de la recurrencia.

Cálculo del factorial de un número entero n

El algoritmo

- Sea F(n) = n!, para $n \geq 0$. Sabemos que:

  n! = 1....(n-1)n = (n-1)!n, por definición 0! = 1.

- Podemos calcular F(n) = F(n-1)$*$n con el siguiente algoritmo:

Algorithm 1 F(int n)

```
Input: n: entero ≥ 0.
Output: factorial de n.
if ( n == 0 ) then
    return 1;
else
    return F(n-1)*n;
end if
```

Cálculo del factorial de un número entero n

Análisis previo

- Tamaño de la entrada: número de bits de n.
- ¿Necesita el algoritmo de espacio de almacenamiento adicional? ¡SI!, ¿cuál? memoria para la pila.
- Operación básica: la multiplicación.
- Para obtener la ecuación de recurrencia, denotaremos al número de multiplicaciones realizadas como T(n).

Cálculo del factorial de un número entero n

Análisis matemático

- F(n) es calculada de acuerdo a la fórmula: F(n) = F(n-1)$*$n

- El número de multiplicaciones T(n) para calcular F(n) debe satisfacer la siguiente igualdad:

  T(n) = T(n-1) + 1 ($n > 0$).

- T(n-1) representa el cálculo de F(n-1) y la constante 1 representa el producto entre F(n-1) y n.

Cálculo del factorial de un número entero n

Análisis matemático

- La ecuación de recurrencia necesita una condición inicial, la cual está dada por el caso base. Cuando n es igual 0, se realizan cero multiplicaciones ($\therefore$ T(0)=0).

- De esta forma ahora podemos plantear la ecuación de recurrencia completa:

  T(n) = T(n-1) + 1 ($n > 0$).

  T(0) = 0, condición inicial.

- T(n) nos da el número de multiplicaciones para calcular F(n).

- Para obtener la complejidad temporal del algoritmo, necesitamos resolver la ecuación de recurrencia. ¿Cómo se hace esto?

Ejemplos de ecuaciones de recurrencia

- $T(n) = T(\frac{n}{2}) + n$
- $T(n) = 2T(n-1) + 1$
- $T(n) = 2T(\frac{n}{2}) + n$
- $T(n) = T(n-1) + T(n-2), n \geq 2$
- $T(n) = T(\sqrt{n}) + 1$
- $T(n) = T(\lfloor n/2 \rfloor) + 1.$
- $T(n) = T(n-1) + 2$
- $T(n) = 64T(\frac{n}{4}) + n^3$
- $T(n) = 3T(\frac{n}{3}) + log^2 n$
- $T(n) = T(\sqrt{n}) + \log \log n$
- $T(n) = T(\frac{n}{4}) + T(\frac{3n}{4}) + 2n$
- $T(n) = 3T(\frac{n}{9}) + \sqrt{n}$
- $T(n) = 7T(\frac{n}{2}) + n^2$
- $T(n) = 64T(\frac{n}{2}) + 2^n$

Métodos para resolver ecuaciones de recurrencia

(1) Substitución forward.

(2) Substitución backward.

(3) Recurrencias telescópicas.

(4) Recurrencias lineales homogéneas.

(5) Cambio de variables.

(6) Teorema maestro.

Métodos para resolver ecuaciones de recurrencia

Substitución forward: Factorial recursivo

- Usamos la misma ecuación para generar los primeros términos para encontrar algún patrón:
- Se inicia desde la condición inicial.

T(0) = 0

T(1) = T(0) + 1 = 0 + 1 = 1

T(2) = T(1) + 1 = 1 + 1 = 2

T(3) = T(2) + 1 = 2 + 1 = 3

...

T(k) = T(k-1) + 1 = k (el patrón)

T(n) = n $\Rightarrow \Theta(n)$

Métodos para resolver ecuaciones de recurrencia

Substitución backward: Factorial recursivo

- Resolveremos esta ecuación a través del método de substitución backward.
- $T(n) = T(n\text{-}1) + 1$, substituimos $T(n\text{-}1) = T(n\text{-}2) + 1$

  $= [T(n\text{-}2) + 1] + 1 = T(n\text{-}2) + 2$, substituimos $T(n\text{-}2) = T(n\text{-}3) + 1$

  $= [T(n\text{-}3) + 1] + 2 = T(n\text{-}3) + 3$, substituimos $T(n\text{-}3) = T(n\text{-}4) + 1$

  $= [T(n\text{-}4) + 1] + 3 = T(n\text{-}4) + 4$, etc
- Hasta aquí ya hemos hallado el patrón: $T(n) = T(n\text{-}i) + i$

Métodos para resolver ecuaciones de recurrencia

Substitución backward: Factorial recursivo

- Tomamos ahora la condición inicial, T(0) = 0, debemos substituir i=n en la fórmula del patrón para obtener el último resultado de la substitución.

$$T(n) = T(n-1) + 1 = ... = T(n-i) + i = T(n-n) + n$$

$$T(n) = \cancel{T(0)}^{\,0} + n = n$$

$$T(n) = \Theta(n)$$

- Se debe notar que la versión no recursiva de este problema también es $\Theta(n)$.

Métodos para resolver ecuaciones de recurrencia

Recurrencias telescópicas

- Las recurrencias telescópicas tienen la siguiente forma general:

$$T(n) = a_{n-1} + T(n-1)$$

$$T(n) = a_{n-1} + a_{n-2} + T(n-2) \text{ (pq } T(n-1) = a_{n-2} + T(n-2))$$

$$T(n) = a_{n-1} + a_{n-2} + a_{n-3} + T(n-3)$$

...

$$T(n) = a_{n-1} + a_{n-2} + a_{n-3} + \ldots a_{n-k} + \ldots + a_0 + T(0)$$

$$T(n) = T(0) + \sum_{i=0}^{n-1} a_i = T(0) + \sum_{i=1}^{n} a_i$$

Métodos para resolver ecuaciones de recurrencia

Recurrencias telescópicas

- En general, si:

$$T(n) = f(n) + T(n-1)$$

- Tenemos:

$$T(n) = C + \sum_{i=1}^{n} f(i)$$

Donde la constante C está dada por la condición inicial de la ecuación de recurrencia.

Métodos para resolver ecuaciones de recurrencia

Recurrencias telescópicas

- En el caso del algoritmo recursivo para calcular el factorial de n:

  $T(n) = T(n-1) + 1,\ T(0) = 0$

  $T(n)$ - $T(n-1) = 1$
  $T(n-1)$ - $T(n-2) = 1$
  $T(n-2)$ - $T(n-3) = 1$
  ...
  $T(1)$ - $T(0) = 1$

- Sumando:

  $(T(n)$ - $T(n-1)) + (T(n-1)$ - $T((n-2)) + (T(n-2)$ - $T((n-3))$
  ... + $(T(2)$ - $T(1)) + (T(1)$ - $T(0)) = \sum_{i=1}^{n} 1$

Métodos para resolver ecuaciones de recurrencia

Recurrencias telescópicas

- Entonces:

$$T(n) - T(0) = \sum_{i=1}^{n} 1$$

$$T(n) = T(0) + \sum_{i=1}^{n} 1$$

$$T(n) = 0 + \sum_{i=1}^{n} 1 = n \Rightarrow T(n) = n$$

- Resolver:

  (a) $T(n) = T(n-1) + \frac{1}{n}, n > 0, T(0) = 1;$

  (b) $T(n) = T(n-1) + logn, n > 0, T(0) = 1;$

  (c) $T(n) = T(n-1) + 2^n, n > 0, T(0) = 1$

Métodos para resolver ecuaciones de recurrencia

Recurrencias telescópicas

- Sea $T(n) = T(n-1) + \frac{1}{n}$, $n > 0$, $T(0) = 1$

  $T(n)$ - $T(n-1) = \frac{1}{n}$

- Luego:

  $(T(n)$ - $T(n-1)) + (T(n-1)$ - $T((n-2)) + (T(n-2)$ - $T((n-3))$ ...

  $+ (T(2)$ - $T(1)) + (T(1)$ - $T(0)) = \sum_{i=1}^{n} \frac{1}{i}$

$$= T(n) - T(0) = \sum_{i=1}^{n} \frac{1}{i}$$

$$= T(n) = T(0) + \sum_{i=1}^{n} \frac{1}{i} = 1 + H_n$$

- Donde H_n es el n-ésimo número armónico. :

$$\sum_{i=1}^{n} \frac{1}{i} = 1 + \tfrac{1}{2} + \tfrac{1}{3} + \tfrac{1}{4} + ... + \tfrac{1}{n} = \ln n + O(1)$$

- Entonces:

$$T(n) = O(\ln n)$$

- Resolveremos ahora la ecuación del factorial a través de recurrencias lineales homogéneas.
- Dada la siguiente ecuación:

$$a_k X_{n+k} + a_{k-1} X_{n+(k-1)} + ... + a_1 X_{n+1} + a_0 X_n = 0 \; (*)$$

- Las soluciones de esta ecuación son de la forma: $X_n = \lambda^n$
- Entonces de la ecuación $(*)$ tenemos:

$$a_k \lambda^{n+k} + a_{k-1} \lambda^{n+(k-1)} + ... + a_1 \lambda^{n+1} + a_0 \lambda^n = 0$$

Métodos para resolver ecuaciones de recurrencia

Recurrencias lineales homogéneas

- Sacamos factor común λ^n:

$$\lambda^n(a_k\lambda^k + a_{k-1}\lambda^{k-1} + ... + a_1\lambda + a_0) = 0$$

- El polinomio: $a_k\lambda^k + a_{k-1}\lambda^{k-1} + ... + a_1\lambda + a_0$, se denomina polinomio característico.

- La solución de la ecuación consiste en encontrar las raíces del polinomio característico:

$$a_k\lambda^k + a_{k-1}\lambda^{k-1} + ... + a_1\lambda + a_0 = 0$$

Métodos para resolver ecuaciones de recurrencia

Recurrencias lineales homogéneas

- Sean λ_1, λ_2, ..., λ_k, las raíces del polinomio característico. La solución general es de la forma:

$$X_n = C_1\lambda_1^n + C_2\lambda_2^n + \ldots + C_k\lambda_k^n$$

- Resolvemos el sistema de ecuaciones lineales:

$$\begin{array}{llll}
C_1\lambda_1^0 & + C_2\lambda_2^0 & + \ldots & + C_k\lambda_k^0 & = X_0 \\
C_1\lambda_1^1 & + C_2\lambda_2^1 & + \ldots & + C_k\lambda_k^1 & = X_1 \\
\ldots & & \ldots & & \ldots \\
C_1\lambda_1^{k-1} & + C_2\lambda_2^{k-1} & + \ldots & + C_k\lambda_k^{k-1} & = X_{k-1}
\end{array}$$

- Los λ_i y los X_i son conocidos

Métodos para resolver ecuaciones de recurrencia

Recurrencias lineales homogéneas

- Sean λ_1 y λ_2 las raíces de la ecuación característica:

caso 1 Si $\lambda_1, \lambda_2 \in \mathbb{R}$ y $\lambda_1 \neq \lambda_2$, entonces la solución general de la ecuación de la recurrencia se obtiene con la siguiente fórmula:

$$T(n) = C_1\lambda_1^n + C_2\lambda_2^n$$

caso 2 Si $\lambda_1 = \lambda_2$, la solución general de la ecuación de la recurrencia se obtiene con la siguiente fórmula:

$$T(n) = C_1\lambda_1^n + C_2 n\lambda_2^n$$

caso 3 Si $\lambda_{1,2} = u \pm iv$

$T(n) = \gamma^n[C_1 cos(n\theta) + C_2 sin(n\theta)]$, donde $\gamma = \sqrt{u^2 + v^2}$, $\theta = arctang(\frac{v}{u})$,
C_1 y C_2 son constantes arbitrarias.

Métodos para resolver ecuaciones de recurrencia

Recurrencias lineales homogéneas

- Sea T(n) = T(n-1) + 1, T(0) = 0, también son válidas las siguientes expresiones:

  (1) T(n+1) = T(n) + 1, y

  (2) T(n+2) = T(n+1) + 1

- Restando las ecuaciones: (2) - (1), obtenemos:

$$
\begin{array}{l}
T(n+2) - T(n+1) \qquad\quad = 1 \\
\qquad\quad T(n+1) - T(n) = 1 \\
\hline
T(n+2) - 2T(n+1) + T(n) = 0
\end{array}
$$

Métodos para resolver ecuaciones de recurrencia

Recurrencias lineales homogéneas

- Obtenemos la ecuación homogénea:

  λ^{n+2} -2λ^{n+1} + λ^{n} = 0

- Sacando factor común λ^n:

  $\lambda^n(\lambda^2 - 2\lambda + 1) = 0$

  $\lambda^2 - 2\lambda + 1 = 0$ (resolver la ecuación característica)

- Corresponde a una ecuación cuadrática, donde $\lambda = \lambda_1 = \lambda_2 = 1$.

- De aquí obtenemos el siguiente sistema de ecuaciones lineales:

  T(n) = $C_1\lambda^n + C_2 n\lambda^n$, como $\lambda = 1$, reemplazamos en la ecuación:

  T(n) = $C_1 + C_2 n$ $(**)$

- Sabemos que T(0) = 0, desde aquí obtenemos el siguiente sistema de ecuaciones lineales:

  T(0) = $C_1 + C_2 0 = 0 \rightarrow C_1 = 0$ (reemplazando en la ecuación $(**)$)

  T(1) = 1 (porque T(1) = $\cancel{T(0)}^{0} + 1$)

  T(1) = $\cancel{C_1}^{0} + C_2 = 1$, entonces $C_2 = 1$.

- De la ecuación $(**)$ tenemos que: $T(n) = n$.

- En algunos casos la forma telescópica no se observa directamente. Utilizando cambio de variables podemos hacer notoria la forma anterior y facilitar la resolución de la recurrencia.

- Por ejemplo:

$$T(n) = 2T(\frac{n}{2}) + n, \; T(n) = 1, \; n \leq 1$$

- Supongamos que $n = 2^k$, entonces tenemos que:

$$T(2^k) = 2T(2^{k-1}) + 2^k$$

Métodos para resolver ecuaciones de recurrencia

Cambio de variables

- La forma telescópica aún no es notoria, pero veremos qué pasa si hacemos:

  T(2^k) = $y(k)$, tendremos:

  $y(k) = 2y(k-1) + 2^k$ / dividiendo por 2^k

  $\frac{y(k)}{2^k} = \frac{2y(k-1)}{2^k} + \frac{2^k}{2^k} = \frac{y(k-1)}{2^k 2^{-1}} + 1 = \frac{y(k-1)}{2^{k-1}} + 1$

  $\frac{y(k)}{2^k} = \frac{y(k-1)}{2^{k-1}} + 1$, haciendo $r(k) = \frac{y(k)}{2^k}$, obtenemos:

  r(k) = r(k-1) + 1, que es la ecuación telescópica buscada.

Métodos para resolver ecuaciones de recurrencia

Cambio de variables

- Pero sabemos que:

$$r(k) = r(k\text{-}1) + 1 = r(0) + \sum_{i=1}^{k} 1$$

$$r(k) = 1 + k$$

Recapitulando. Sabemos que:

$$T(n) = T(2^k) = y(k) = r(k)2^k = (1 + k)2^k = 2^k + k2^k$$

Como n = 2^k, entonces k = $log_2 n$

$$T(n) = n + nlog_2 n \Rightarrow T(n) = O(nlog_2 n)$$

Métodos para resolver ecuaciones de recurrencia

Cambio de variables

- Ejercicios:

  $$T(n) = 2T(n\text{-}1) + 1,\ T(0) = 1$$

  $$T(n) = 1 + T(\sqrt{n}),\ T(2) = 1$$

  $$T(n) = \sqrt{n}T(\sqrt{n}) + cn,\ T(2) = 1$$

Métodos para resolver ecuaciones de recurrencia

Teorema Maestro

- Sea T(n) una función eventualmente no decreciente que satisface la recurrencia:

- T(n) = aT($\frac{n}{b}$) + f(n), T(1) = c, n = b^k, k = 1,2,...

  donde a $\geq$ 1, b $\geq$ 2, c $>$ 0. Si $f(n) \in \Theta(n^d)$, d $\geq$ 0, entonces:

$$T(n) = \begin{cases} \Theta(n^d) & \text{si a} < b^d \\ \Theta(n^d logn) & \text{si a} = b^d \\ \Theta(n^{log_b a}) & \text{si a} > b^d \end{cases}$$

Métodos para resolver ecuaciones de recurrencia

Teorema Maestro: ejercicios

(1) $T(n) = 1 + T(\sqrt{n})$

(2) $T(n) = T(\lfloor \frac{n}{2} \rfloor) + 1$

(3) $T(n) = T(\frac{n}{3}) + 1$

(4) $T(n) = 2T(\frac{n}{2}) + 1$

(5) $T(n) = 64T(\frac{n}{4}) + n^3$

(6) $T(n) = 32T(\frac{n}{2}) + n^4$

(7) $T(n) = 2T(\frac{n}{2}) + cn$

(1) $T(n) = 1 + T(\sqrt{n})$

Usamos substitución: sea n = 2^k, $k = \log n$

$T(2^k) = 1 + T(2^{\frac{k}{2}})$, de aquí hacemos: $y(k) = T(2^k)$

$y(k) = 1 + y(\frac{k}{2}) = n^0 + y(\frac{k}{2})$, a = 1, b = 2, d = 0.

Ahora aplicamos el Teorema Mestro: a = $b^d = b^0 = 1$

$y(k) = 1 + O(k^0 logk) = 1 + O(logk) = 1 + O(loglogn)$

$T(n) = 1 + O(loglogn) \Rightarrow T(n) = O(loglogn)$

(2) $T(n) = T(\lfloor \frac{n}{2} \rfloor) + 1$

No es necesario usar substitución, así que aplicamos directamente el Teorema Maestro

a = 1, b = 2, d = 0 $\Rightarrow$ a = b^0

$T(n) = 1 + O(n^0 logn)$

$T(n) = O(\lfloor \log n \rfloor)$

Métodos para resolver ecuaciones de recurrencia

Teorema Maestro: ejercicios

(3) $T(n) = T(\frac{n}{3}) + 1$

a = 1, b = 3, d = 0, $\Rightarrow$ a = b^0

$T(n) = 1 + O(n^0 logn)$

$T(n) = O(\log_3 n)$

(4) $T(n) = 2T(\frac{n}{2}) + 1$

a = 2, b = 2, d = 0, $\Rightarrow$ a $> b^0$

$T(n) = O(n^{log_2 2})$

$T(n) = O(n)$

Métodos para resolver ecuaciones de recurrencia

Teorema Maestro: ejercicios

(5) $T(n) = 64T(\frac{n}{4}) + n^3$

a = 64, b = 4, d = 3, $\Rightarrow$ a = b^3

$T(n) = O(n^3 logn)$

(6) $T(n) = 32T(\frac{n}{2}) + n^4$

a = 32, b = 2, d = 4, $\Rightarrow$ a $> b^4 \Rightarrow 32 > 2^4$

$T(n) = O(n^{log_2 32}) = O(n^{log_2 2^5})$

$T(n) = O(n^5)$

(7) $T(n) = 2T(\frac{n}{2}) + cn$ (algoritmo Merge Sort)

a = 2, b = 2, d = 1, $\Rightarrow$ a = b^1

$T(n) = O(nlogn)$

(8) $T(n) = T(\frac{n}{2}) + c$ (Búsqueda binaria)

a = 1, b = 2, d = 0, $\Rightarrow$ a $=$ $b^0 \Rightarrow 1 = 2^0$

$T(n) = O(n^0 log_2 n)$

$T(n) = O(logn)$