دانشکده علوم ریاضی و آمار

مدرس: دکتر مجتبی رفیعی نیمسال اول ۱۴۰۰-۱۴۰۱

ساختمان داده‌ها و الگوریتم‌ها

جلسه ۲۰

نگارنده: زهرا فریدونی

۲۰ آبان ۱۴۰۰

فهرست مطالب

۱ یادآوری

در جلسات قبل درمورد حل روابط بازگشتی صحبت شد و همچنین قضیه‌ی زیر برای پیدا کردن پیچیدگی زمانی این روابط مطرح شد:

قضیه: اگر رابطه‌ی بازگشتی به فرم زیر داشته باشیم:

$T(n) = aT(\frac{n}{b}) + f(n)$, where $a \geq 1$ *and* $b > 1$

آنگاه:

- حالت اول: اگر $f(n) = O(n^{c-\epsilon})$ باشد، جاییکه $\epsilon > 0$ آنگاه $T(n) = \theta(n^c)$.
- حالت دوم: اگر $f(n) = \theta(n^c \log^k n)$ باشد، جاییکه k یک عدد صحیح نامنفی باشد، آنگاه $T(n) = \theta(n^c \log^{k+1} n)$.
- حالت سوم: اگر $f(n) = \Omega(n^{c+\epsilon})$ باشد، جاییکه $\epsilon > 0$ است و همچنین $af(\frac{n}{b}) \leq hf(n)$ باشد برای یک $0 < h < 1$ و هر n به اندازه‌ی کافی بزرگ، آنگاه $T(n) = \theta(f(n))$.

در هر سه حالت ذکر شده در بالا $c = \log_b a$ می‌باشد (با توجه به روش‌های قبلی ذکر شده برای حل روابط بازگشتی، علت این مقدار برای c واضح است).
نکته: برای مشخص کردن حالت‌های ۱، ۲ و ۳ به نحوی بحث روی n^c است:

- یک ϵ کمتر،
- یک $\log^k n$ ضربدر n^c (تقریباً در مرتبه‌ی n^c)
- یک ϵ بیشتر.

سوال

آیا حالت‌های بیان شده در قضیه، همه‌ی توابع f(n) صعودی را پوشش می‌دهد؟

پاسخ منفی است، در ادامه با ذکر مثال‌هایی به این موضوع می‌پردازیم:

۲ مثال‌هایی از حل روابط بازگشتی با استفاده از قضیه

۱.۲ مثال اول

$T(n) = 4T(\frac{n}{2}) + n^2 \ \rightarrow \ f(n) = n^2 \qquad\qquad a = 4,\ b = 2,\ c = \log_b a = \log_2 4 = 2$

- case 1: $n^2 \leq n^{2-\epsilon}$ ✕
- case 2: $n^2 = \theta(n^2 \log^k n),\ k = 0 \ \Rightarrow \ n^2 = \theta(n^2)$ ✓ $\rightarrow \ T(n) = \theta(n^2 \log n)$
- case 3: $n^2 \geq n^{2+\epsilon}$ ✕

۲.۲ مثال دوم

$T(n) = 4T(\frac{n}{2}) + n^3 \ \rightarrow \ f(n) = n^3 \qquad\qquad a = 4,\ b = 2,\ c = \log_b a = \log_2 4 = 2$

- case 1: $n^3 \leq n^{2-\epsilon}$ ✕
- case 2: $n^3 = \theta(n^2 \log^k n)$ ✕
- case 3: $n^3 \geq n^{2+\epsilon}$ and $af(\frac{n}{b}) \leq hf(n)$

$$\Rightarrow 4\frac{n^3}{8} \leq hn^3$$
$$\Rightarrow \frac{n^3}{2} \leq hn^3$$
$$\Rightarrow \text{for } h = \frac{2}{3} \Rightarrow \ \frac{n^3}{2} \leq \frac{2n^3}{3} \ ✓ \rightarrow \ T(n) = \theta(n^3)$$

۳.۲ مثال سوم

$T(n) = 4T(\frac{n}{2}) + n\sqrt{n} \ \rightarrow \ f(n) = n^{\frac{3}{2}}$ $\qquad a = 4,\ b = 2,\ c = \log_b a = \log_2 4 = 2$

- case 1: $n^{\frac{3}{2}} \leq n^{2-\epsilon}$ ✓ $\rightarrow \ T(n) = \theta(n^2)$
- case 2: $n^{\frac{3}{2}} = \theta(n^2 \log^k n)$ ✗
- case 3: $n^{\frac{3}{2}} \geq n^{2+\epsilon}$ ✗

۴.۲ مثال چهارم

$T(n) = 2T(\frac{n}{2}) + n\log n \ \rightarrow \ f(n) = n\log n$ $\qquad a = 2,\ b = 2,\ c = \log_b a = \log_2 2 = 1$

- case 1: $n\log n \leq n^{1-\epsilon}$ ✗
- case 2: $n\log n = \theta(n \log^k n),\ k = 1 \ \Rightarrow \ n\log n = \theta(n\log n)$ ✓ $\rightarrow \ T(n) = \theta(n\log^2 n)$
- case 3: $n\log n \geq n^{1+\epsilon}$ and $af(\frac{n}{b}) \leq hf(n)$

$$\Rightarrow 2\frac{n}{2}\log\frac{n}{2} \leq hn\log n$$
$$\Rightarrow n\log n - n \leq hn\log n \ (*)$$

برای $h < 1$ رابطه $(*)$ برقرار نیست.

۵.۲ مثال پنجم

$T(n) = 2T(\frac{n}{2}) + n\log(\log n) \ \rightarrow \ f(n) = n\log(\log n)$ $\qquad a = 2,\ b = 2,\ c = \log_b a = \log_2 2 = 1$

- case 1: $n\log(\log n) \leq n^{1-\epsilon}$ ✗
- case 2: $n\log(\log n) = \theta(n \log^k n)$ ✗
- case 3: $n\log(\log n) \geq n^{1+\epsilon}$ and $af(\frac{n}{b}) \leq hf(n)$

بررسی قسمت اول:

$$\Rightarrow n\log(\log n) \geq n^{1+\epsilon}$$

از دو طرف log بگیریم:

$$\Rightarrow \log n + \log(\log(\log n)) \geq (1+\epsilon)\log n$$
$$\Rightarrow \log(\log(\log n)) \ngeq \epsilon \log n$$

پس شرایط case 3 نیز برقرار نیست.

۳ اثبات قضیه اصلی

- حالت اول:

$T(n) = aT(\frac{n}{b}) + f(n),\ a \geq 1,\ b > 1,\ c = \log_b a$

if $f(n) = O(n^{c-\epsilon})$ then $T(n) = \theta(n^c)$

طبق رابطه‌ای که برای $T(n)$ از روش‌های جایگذاری و درخت بازگشت به‌دست آوردیم، داریم:

$$T(n) = n^{\log_b a} T(1) + \sum_{i=0}^{(\log_b n)-1} a^i f(\frac{n}{b^i})$$

قرار می‌دهیم $\sum_{i=0}^{(\log_b n)-1} a^i f(\frac{n}{b^i}) = A^*$ و رابطه را ساده‌تر می‌کنیم:

$$T(n) = n^{\log_b a}T(1) + A^* = n^cT(1) + A^* = \theta(n^c) + A^*$$

طبق $f(n) = O(n^{c-\epsilon})$، داریم:

$$A^* \le \sum_{i=0}^{(\log_b n)-1} a^i h(\frac{n}{b^i})^{c-\epsilon}$$

$$= hn^{c-\epsilon} \sum_{i=0}^{(\log_b n)-1} (\frac{a}{b^{c-\epsilon}})^i$$

$$= hn^{c-\epsilon}(\frac{(\frac{a}{b^{c-\epsilon}})^{\log_b n} - 1}{\frac{a}{b^{c-\epsilon}} - 1})$$

عبارت $(\frac{a}{b^{c-\epsilon}})^{\log_b n}$ را ساده‌تر می‌کنیم:

$$(\frac{a}{b^{c-\epsilon}})^{\log_b n} = n^{\log_b \frac{a}{b^{c-\epsilon}}} = n^{\log_b a-(c-\epsilon)} = n^{c-c+\epsilon} = n^\epsilon$$

در نتیجه داریم:

$$A^* \le hn^{c-\epsilon}n^\epsilon = hn^c = \theta(n^c)$$

و در نهایت داریم $T(n) = \theta(n^c)$.

- حالت دوم:

$T(n) = aT(\frac{n}{b}) + f(n),\ a \ge 1,\ \ b > 1,\ \ c = \log_b a$

if $\ f(n) = \theta(n^c\log^k n)\ $ then $\ T(n) = \theta(n^c\log^{k+1} n)$

طبق رابطه‌ی به‌دست آمده برای $T(n)$ با استفاده از درخت بازگشت داریم:

$$T(n) = n^cT(1) + \sum_{i=0}^{(\log_b n)-1} a^i f(\frac{n}{b^i})$$

قرار می‌دهیم $\sum_{i=0}^{(\log_b n)-1} a^i f(\frac{n}{b^i}) = A^*$ و رابطه را ساده‌تر می‌کنیم:

$$T(n) = n^cT(1) + A^* = \theta(n^c) + A^*$$

طبق $f(n) = \theta(n^c\log^k n)$، داریم:

$$A^* \le \sum_{i=0}^{(\log_b n)-1} a^i h(\frac{n}{b^i})^c\log^k(\frac{n}{b^i})$$

از آنجایی که می‌دانیم $\log^k(\frac{n}{b^i}) \le \log^k n$ داریم:

$$A^* \le \sum_{i=0}^{(\log_b n)-1} a^i h(\frac{n}{b^i})^c\log^k n$$

$$= hn^c\log^k n \sum_{i=0}^{(\log_b n)-1} (\frac{a}{b^c})^i$$

از آنجایی‌که $c = \log_b a$، برای b^c داریم $b^{\log_b a} = a$ و در نهایت عبارت بالا برابر است با:

$$A^* \leq hn^c(\log^k n) \log n = \theta(n^c \log^{k+1} n)$$

بنابراین داریم:

$$T(n) = \theta(n^c) + \theta(n^c \log^{k+1} n) = \theta(n^c \log^{k+1} n)$$