TCSS 343 - Assignment 3

Jake McKenzie

July 17, 2018

1 UNDERSTAND

In this problem use the Master Theorem to find and prove tight bounds for these recurrences (6 points each).

To solve this problem I used the master theorem taken from CLRS. I used a bit of a stronger statement than what was stated in the theorem but they are equivalent mathematically due to the nature of limits. I will include the theorem for the reader's benefit. To check my work I decided to include what I obtained using the Akra-Bazzi method.

Master Theorem

The Master Theorem applies to recurrences of the following form:

$$T(n) = aT(n/b) + f(n)$$

where $a \geq 1$ and $b > 1$ are constants and $f(n)$ is an asymptotically positive function.

There are 3 cases:

1. If $f(n) = O(n^{\log_b a - \epsilon})$ for some constant $\epsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$.

2. If $f(n) = \Theta(n^{\log_b a} \log^k n)$ with[1] $k \geq 0$, then $T(n) = \Theta(n^{\log_b a} \log^{k+1} n)$.

3. If $f(n) = \Omega(n^{\log_b a + \epsilon})$ with $\epsilon > 0$, and $f(n)$ satisfies the regularity condition, then $T(n) = \Theta(f(n))$. Regularity condition: $af(n/b) \leq cf(n)$ for some constant $c < 1$ and all sufficiently large n.

1.

$$T(n) = \begin{cases} c & \text{if } n < 8 \\ 16T(n/8) + n \log n & \text{if } n \geq 8 \end{cases}$$

By Master Method I obtain case 1:

$$\lim_{n \to \infty} \frac{n \log n}{n^{\log_8 16 + \varepsilon}} = \lim_{n \to \infty} \frac{n \log n}{n^{\frac{4}{3} + \frac{2}{3}}}$$

$$= \lim_{n \to \infty} (\frac{n}{n})(\frac{\log n}{n}) \to 0$$

$$T(n) \in \Theta(n^{\frac{4}{3}})$$

By Akra-Bazzi I obtain:

$$16(1/8)^p = 1$$
$$p = \frac{4}{3}$$

$$T(n) \in \Theta(n^p(1 + \int_1^n \frac{u \log u}{u^{p+1}} du))$$
$$\in \Theta(-9n + 10n^{\frac{4}{3}} - 3n \log n)$$
$$\in \Theta(n^{\frac{4}{3}})$$

2.
$$T(n) = \begin{cases} c & \text{if } n < 4 \\ 8T(n/4) + n \log n & \text{if } n \geq 4 \end{cases}$$

By Master Method I obtain case 2:

$$\lim_{n \to \infty} \frac{n^{\frac{1}{3}}}{n^{\log_8 2 + \varepsilon}} = \lim_{n \to \infty} \frac{n^{\frac{1}{3}}}{n^{\frac{1}{3}}}$$
$$= \lim_{n \to \infty} 1 \to 1$$
$$\in \Theta(n^{\frac{1}{3}} \log n)$$

By Akra-Bazzi I obtain:

$$2(1/8)^p = 1$$
$$p = \frac{1}{3}$$

$$T(n) \in \Theta(n^p(1 + \int_1^n \frac{u^{\frac{1}{3}}}{u^{p+1}} du))$$
$$\in \Theta(n^{\frac{1}{3}} + n^{\frac{1}{3}} \log n)$$
$$\in \Theta(n^{\frac{1}{3}} \log n)$$

3.
$$T(n) = \begin{cases} c & \text{if } n < 2 \\ 3T(n/2) + 9^n & \text{if } n \geq 2 \end{cases}$$

By Master Method I obtain case 3:

$$\lim_{n \to \infty} \frac{9^n}{n^{\log_3 2 + \varepsilon}} = \lim_{n \to \infty} \frac{9^n}{n^{\frac{\log 2}{\log 3} - (\frac{\log 2}{\log 3} + 1)}}$$
$$= \lim_{n \to \infty} n9^n \to \infty$$
$$\in \Theta(9^n)$$

This recurrence is not well suited for Akra-Bazzi.

4.
$$T(n) = \begin{cases} c & \text{if } n \leq 1 \\ 3T(3n/5) + n^2 & \text{if } n > 1 \end{cases}$$

By Master Method I obtain case 1:

$$\lim_{n \to \infty} \frac{n^2}{n^{\log_{\frac{5}{3}} 3 + \varepsilon}} = \lim_{n \to \infty} \frac{n^2}{n^{\log_{\frac{5}{3}} 3 + (3 - \log_{\frac{5}{3}} 3)}}$$

$$= \lim_{n \to \infty} \frac{n^2}{n^3} \to 0$$

$$\log_{\frac{5}{3}} 3 = 2.15066...$$

$$\in \Theta(n^{2.15066...})$$

By Akra-Bazzi I obtain:

$$3(3/5)^p = 1$$

$$p = \log_{\frac{5}{3}} 3$$

$$T(n) \in \Theta(n^p (1 + \int_1^n \frac{u^2}{u^{p+1}} du))$$

$$\in \Theta(7.63746 n^{2.15066...} - 6.63746 n^2)$$

$$\in \Theta(n^{2.15066...})$$

5.
$$T(n) = \begin{cases} c & \text{if } n \leq 1 \\ 3T(3n/5) + n^{2.5} & \text{if } n > 1 \end{cases}$$

By Master Method I obtain case 3:

$$\lim_{n \to \infty} \frac{n^{2.5}}{n^{\log_{\frac{5}{3}} 3 + \varepsilon}} = \lim_{n \to \infty} \frac{n^{2.5}}{n^{\log_{\frac{5}{3}} 3 + 0.1}}$$

$$= \lim_{n \to \infty} \frac{n^2.5}{n^{2.25}} \to \infty$$

$$= \lim_{n \to \infty} n^{0.25...} \to \infty$$

$$\in \Theta(n^{2.5})$$

2 EXPLORE

1. For the following problems stated as pseudo-code, let $A[\ell \ldots r]$ denote the sublist of the integer list A from the ℓ-th to the r-th element inclusive, let $\text{Cubic}(A[1 \ldots n])$ denote

an algorithm that runs in time $\Theta(n^3)$, and let $\text{Swift}(A[1\ldots n])$ denote an algorithm that runs in time $\Theta(n\log(\log n))$.

```
Three(A[1...n])
        If  n ≤ 1 Then Return // nothing to do
        Cubic(A[1...n])
        Three(A[1...⌊n/2⌋])
        Three(A[⌊n/4⌋ + 1...⌊3n/4⌋])
        Three(A[⌊n/2⌋ + 1...n])
End Three.
```