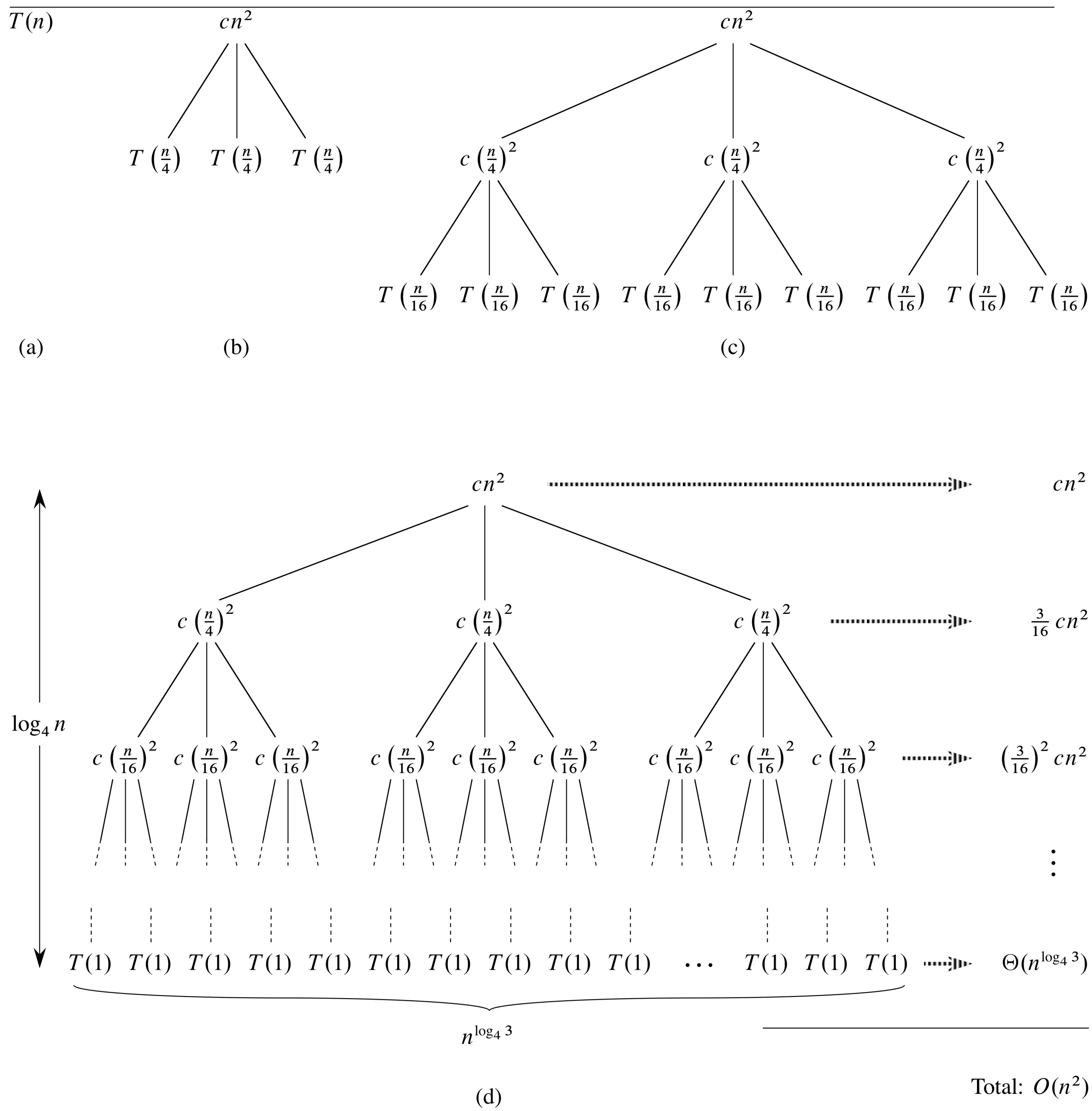
Proof of Master Theorem

Geoffrey Matthews

April 12, 2016

$$T(n) = 3T(n/4) + cn^2$$



1 Divide and Conquer Recurrence

Suppose that $a \geq 1$ and $b \geq 1$ are constants, let $f(n)$ be a function, and let $T(n)$ be defined on natural numbers with the recurrence:

$$T(n) = aT\left(\frac{n}{b}\right) + f(n) \tag{1}$$

Let's assume n is a power of b, so that

$$n = b^k \tag{2}$$
$$k = \log_b(n) \tag{3}$$

Now let's see what we get by trying to spell out terms and then cancel:

$$T(b^k) = aT(b^{k-1}) + f(b^k) \tag{4}$$
$$aT(b^{k-1}) = a^2T(b^{k-2}) + af(b^{k-1}) \tag{5}$$
$$a^2T(b^{k-2}) = a^3T(b^{k-3}) + a^2f(b^{k-2}) \tag{6}$$
$$\ldots \tag{7}$$
$$a^{k-1}T(b^1) = a^kT(1) + a^{k-1}f(b) \tag{8}$$

Adding up both sides while cancelling terms, we get

$$T(b^k) = a^kT(1) + \sum_{i=1}^{k} a^{k-i}f(b^i) \tag{9}$$
$$= a^{\log_b(n)}T(1) + \sum_{i=1}^{\log_b(n)} a^{k-i}f(b^i) \tag{10}$$
$$= n^{\log_b(a)}T(1) + \sum_{i=1}^{\log_b(n)} a^{k-i}f(b^i) \tag{11}$$

Since $T(1)$ will be a constant, the complexity of the whole thing depends on whether

$$n^{\log_b(a)} \tag{12}$$

grows faster or slower than

$$\sum_{i=1}^{\log_b(n)} a^{k-i} f(b^i) \tag{13}$$

Note that expression 13 can be rewritten as

$$\begin{align}
\sum_{i=1}^{\log_b(n)} a^{k-i} f(b^i) &= \sum_{i=0}^{\log_b(n)-1} a^i f(b^{k-i}) \tag{14}\\
&= \sum_{i=0}^{\log_b(n)-1} a^i f\left(\frac{b^k}{b^i}\right) \tag{15}\\
&= \sum_{i=0}^{\log_b(n)-1} a^i f\left(\frac{n}{b^i}\right) \tag{16}
\end{align}$$

Let's see what we can find out about this expression, using either 13 or 16, and how it compares to 12.

2 $f(n) = \Theta\left(n^{\log_b(a)}\right)$

Let's first suppose

$$f(n) = \Theta\left(n^{\log_b(a)}\right) \tag{17}$$

then

$$\begin{align}
f(b^i) &= \Theta\left((b^i)^{\log_b(a)}\right) \tag{18}\\
&= \Theta\left(b^{i\log_b(a)}\right) \tag{19}\\
&= \Theta\left(\left(b^{\log_b(a)}\right)^i\right) \tag{20}\\
&= \Theta(a^i) \tag{21}
\end{align}$$

Plugging this back in our expression 13, gives

$$\begin{align}
\sum_{i=1}^{\log_b(n)} a^{k-i} f(b^i) &= \sum_{i=1}^{\log_b(n)} a^{k-i}\Theta(a^i) \tag{22}\\
&= \sum_{i=1}^{\log_b(n)} \Theta(a^k) \tag{23}\\
&= \sum_{i=1}^{\log_b(n)} \Theta(a^{\log_b(n)}) \tag{24}\\
&= \sum_{i=1}^{\log_b(n)} \Theta(n^{\log_b(a)}) \tag{25}\\
&= (\log_b(n))\Theta(n^{\log_b(a)}) \tag{26}\\
&= \Theta(\lg(n) n^{\log_b(a)}) \tag{27}
\end{align}$$

Recall the expression for the complexity of $T(n) = T(b^k)$ in equation 11:

$$T(b^k) = n^{\log_b(a)} T(1) + \sum_{i=1}^{\log_b(n)} a^{k-i} f(b^i) \tag{11, revisited}$$

Putting 27 into 11, the complexity of $T(n)$, shows that if

$$f(n) = \Theta\left(n^{\log_b(a)}\right) \tag{17, revisited}$$

then

$$T(n) = \Theta\left(\lg(n) n^{\log_b(a)}\right) \tag{28}$$

3 $f(n) = O\left(n^{\log_b(a)-\epsilon}\right)$

Now let's suppose that $f(n)$ has some slightly lower complexity. Suppose $0 < \epsilon < \log_b(a)$ is some (small) positive real number, and

$$f(n) = O\left(n^{\log_b(a)-\epsilon}\right) \tag{29}$$

then

$$\begin{align}
f\left(\frac{n}{b^i}\right) &= O\left(\left(\frac{n}{b^i}\right)^{\log_b(a)-\epsilon}\right) \tag{30}\\
&= O\left(\frac{n^{\log_b(a)-\epsilon}}{b^{i\log_b(a)-i\epsilon}}\right) \tag{31}\\
&= O\left(\left(\frac{n^{\log_b(a)}}{n^{\epsilon}}\right)\left(\frac{b^{i\epsilon}}{b^{i\log_b(a)}}\right)\right) \tag{32}\\
&= O\left(\frac{b^{\epsilon}}{b^{\log_b(a)}}\right)^i \tag{33}\\
&= O\left(\frac{b^{\epsilon}}{a}\right)^i \tag{34}
\end{align}$$

Plugging this back into expression 16 gives a geometric series which can be solved:

$$\sum_{i=0}^{\log_b(n)-1} a^i f\left(\frac{n}{b^i}\right) = O\left(\sum_{i=0}^{\log_b(n)-1} a^i \left(\frac{b^\epsilon}{a}\right)^i\right) \tag{35}$$

$$= O\left(\sum_{i=0}^{\log_b(n)-1} (b^\epsilon)^i\right) \tag{36}$$

$$= O\left(\frac{(b^\epsilon)^{\log_b(n)} - 1}{b^\epsilon - 1}\right) \tag{37}$$

$$= O\left(\frac{(n)^{\log_b(b^\epsilon)} - 1}{b^\epsilon - 1}\right) \tag{38}$$

$$= O\left(\frac{n^\epsilon - 1}{b^\epsilon - 1}\right) \tag{39}$$

$$= O\left(n^\epsilon\right) \tag{40}$$

$$= O\left(n^{\log_b(a)}\right) \tag{41}$$

Since b and ϵ are constants, and $0 < \epsilon < \log_b(a)$.

Recall again the expression for the complexity of $T(n) = T(b^k)$ in equation 11:

$$T(b^k) = n^{\log_b(a)} T(1) + \sum_{i=1}^{\log_b(n)} a^{k-i} f(b^i) \tag{11, revisited}$$

Plugging 41 into 11 shows that if

$$f(n) = O\left(n^{\log_b(a)-\epsilon}\right) \tag{29, revisited}$$

then

$$T(n) = \Theta\left(n^{\log_b(a)}\right) \tag{42}$$

4 $f(n) = \Omega\left(n^{\log_b(a)+\epsilon}\right)$

We leave it as an exercise to show that if f has some slightly higher complexity,

$$f(n) = \Omega\left(n^{\log_b(a)+\epsilon}\right) \tag{43}$$

and $af\left(\frac{n}{b}\right) \leq cf(n)$ for some c and large enough n, then

$$T(n) = \Theta(f(n)) \tag{44}$$