Joint Conference of 1996 International Computer Symposium
December 19~21, Kaohsiung, Taiwan, R.O.C.

Multiple-Size Divide-and-Conquer Recurrences*

Ming-Yang Kao
Department of Computer Science
Duke University
Durham, NC 27708
U.S.A.

kao@cs.duke.edu

1 Introduction

This note reports a tight asymptotic solution to the following recurrence on all positive integers n:

$$T(n) = c{\cdot}n^{\alpha}{\cdot}\log^{\beta} n + \sum_{i=1}^{k} a_i{\cdot}T(\lceil b_i{\cdot}n \rceil) \quad \text{for } n \geq n_0, \tag{1}$$

$$0 < T(n) \leq d \quad \text{for } n < n_0, \tag{2}$$

where

- $\alpha \geq 0, \beta \geq 0, c > 0, d > 0$,
- k is a positive integer,
- $a_i > 0$ and $1 > b_i > 0$ for $i = 1, \ldots, k$,
- $n_0 \geq \max_{i=1}^{k} \frac{1}{1-b_i}$.

Since $n_0 \geq \max_{i=1}^{k} \frac{1}{1-b_i}$, $\lceil b_i{\cdot}n \rceil \leq n - 1$ for all b_i and $n \geq n_0$. Thus, the $T(n)$ term on the left-hand side of (1) is defined on T-terms with smaller n, and (2) properly specifies the initial values of T.

A special case of this recurrence, namely, $k = 1$, is discussed in [2, 5] and standard textbooks on algorithms and is used extensively to analyze divide-and-conquer strategies [1, 4]. A specific recurrence with $k = 2$ is used to analyze a divide-and-conquer algorithm for selecting a key with a given rank [1, 3, 4].

Let $g(x) = \sum_{i=1}^{k} a_i{\cdot}b_i^x$. The *characteristic equation* of the general recurrence is the equation $g(x) = 1$. Our solution to the general recurrence is summarized in the following theorem.

Theorem 1 *If r is the solution to the characteristic equation of the general recurrence, then*

$$T(n) = \begin{cases} \Theta(n^r) & \text{if } r > \alpha; \\ \Theta(n^{\alpha} \log^{1+\beta} n) & \text{if } r = \alpha; \\ \Theta(n^{\alpha} \log^{\beta} n) & \text{if } r < \alpha. \end{cases}$$

The key ingredient of our proof for this theorem is the notion of a characteristic equation. With this new notion, our proof is essentially the same as that of the case with $k = 1$ [1, 2, 4, 5]. This note concentrates on elaborating the characteristic equation's role in our proof by detailing an upper bound proof for a certain case. Once this example is understood, it is straightforward to reconstruct a general proof for Theorem 1. Consequently, we omit the general proof for the sake of brevity and clarity.

*Research supported in part by NSF Grant CCR-9101385.

2 An Example

This section discusses the general recurrence with $k = 3$. To further focus our attention on the characteristic equation's role, we assume that $\beta = 0$, r is a positive integer, and $r > \alpha$. Then, according to Theorem 1, $T(n) = \Theta(n^r)$. We will only prove $T(n) = O(n^r)$. The lower bound proof is similar.

Let $S(n) = f_1 \cdot n^r - f_2 \cdot n^{r-\frac{1}{2}} - f_3 \cdot n^\alpha$. It suffices to show that there exist some positive constants f_1, f_2, f_3 such that $T(n) = O(S(n))$. These constants and some others are chosen as follows. Without loss of generality, we assume $b_1 < b_2 < b_3$.

$$
\begin{aligned}
f_3 &= \frac{c}{g(\alpha) - 1}; \\
f_2 &= \text{any positive constant}; \\
f_1 &= f_2 + f_3 + 1; \\
m_0 &= \max\{n_0, \frac{1}{b_1}, \left(\frac{f_1 \cdot 2^r \cdot \frac{1}{b_1}}{f_2 \cdot (g(r - \frac{1}{2}) - 1)} \right)^2 \}; \\
M &= \max_{n < m_0} \{1, T(n)\}.
\end{aligned}
$$

Note that since $0 < b_i < 1$ for all b_i, g is a decreasing function. Then since $r > \alpha$ and $r > r - \frac{1}{2}$, $g(\alpha) > 1$ and $g(r - \frac{1}{2}) > 1$. Thus, the above constants are all positive. We next consider the following new recurrence:

$$
\begin{aligned}
R(n) = c \cdot n^\alpha + a_1 \cdot R(\lceil b_1 \cdot n \rceil) + a_2 \cdot R(\lceil b_2 \cdot n \rceil) + a_3 \cdot R(\lceil b_3 \cdot n \rceil) &\quad \text{for } n \geq m_0, \\
R(n) = 1 &\quad \text{for } n < m_0,
\end{aligned} \tag{3}
$$

It can be shown by induction that $T(n) \leq M \cdot R(n)$ for all n. Thus, to prove $T(n) = O(S(n))$, it suffices to show $R(n) \leq S(n)$ for all n.

Base Case: $R(m) \leq S(m)$ for all $m < m_0$. This follows from the choice of f_1.

Given some $n \geq m_0$, we need to show $R(n) \leq S(n)$.

Induction Hypothesis: $R(m) \leq S(m)$ for all integers m where $m_0 \leq m < n$.

Induction Step:

$$
\begin{aligned}
R(n) &\leq c \cdot n^\alpha + a_1 \cdot S(\lceil b_1 \cdot n \rceil) + a_2 \cdot S(\lceil b_2 \cdot n \rceil) + a_3 \cdot S(\lceil b_3 \cdot n \rceil) && (4) \\
&\leq c \cdot n^\alpha + f_1 \cdot g(r) \cdot (n + \frac{1}{b_1})^r - f_2 \cdot g(r - \frac{1}{2}) \cdot n^{r-\frac{1}{2}} - f_3 \cdot g(\alpha) \cdot n^\alpha && (5) \\
&\leq c \cdot n^\alpha + f_1 \cdot g(r) \cdot n^r + f_1 \cdot 2^r \cdot n^{r-1} \cdot \frac{1}{b_1} - f_2 \cdot g(r - \frac{1}{2}) \cdot n^{r-\frac{1}{2}} - f_3 \cdot g(\alpha) \cdot n^\alpha && (6)
\end{aligned}
$$

In this above derivation,

- (4) follows from (3), the inequality $m_0 \geq n_0$, the base step and the induction hypothesis;
- (5) follows from the fact that $\lceil b_i \cdot n \rceil \leq b_i \cdot (n + \frac{1}{b_1})$;
- (6) follows from the fact that $(n + \frac{1}{b_1})^r \leq n^r + 2^r \cdot n^{r-1} \cdot \frac{1}{b_1}$ because r is a positive integer and $m_0 \geq \frac{1}{b_1}$.

To finish the induction step, note that the right-hand side of (6) is at most $S(n)$ as desired for the following reasons.

- By the choice of f_3, $c \cdot n^\alpha + f_3 \cdot g(\alpha) \cdot n^\alpha \leq -f_3 \cdot n^\alpha$.
- Since $m_0 \geq \left(\frac{f_1 \cdot 2^r \cdot \frac{1}{b_1}}{f_2 \cdot (g(r - \frac{1}{2}) - 1)} \right)^2$, $f_1 \cdot 2^r \cdot n^{r-1} \cdot \frac{1}{b_1} - f_2 \cdot g(r - \frac{1}{2}) \cdot n^{r-\frac{1}{2}} \leq -f_2 \cdot n^{r-\frac{1}{2}}$.

Acknowledgments

The author wishes to thank Don Rose for helpful discussions.

References

[1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman. *The Design and Analysis of Computer Algorithms*. Addison-Wesley, Reading, MA, 1974.

[2] J. L. Bentley, D. Haken, and J. B. Saxe. A general method for solving divide-and-conquer recurrences. *SIGACT News*, 12(3):36–44, 1980.

[3] M. Blum, R. W. Floyd, V. Pratt, R. L. Rivest, and R. E. Tarjan. Time bounds for selection. *Journal of Computer and System Sciences*, 7(4):448–461, 1973.

[4] T. H. Cormen, C. L. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. MIT Press, Cambridge, MA, 1991.

[5] G. S. Lueker. Some techniques for solving recurrences. *ACM Computing Surveys*, 12(4):419–436, 1980.