CIS 575. Introduction to Algorithm Analysis
Material for February 7, 2024

The Substitution Method: Case 2

The topic of this note is covered in *Cormen*'s Section 4.3.

1 Applying the Substitution Method, Example 2

Recall the recurrence for MERGESORT

$$T(n) \in 2T(\frac{n}{2}) + \Theta(n)$$

which in a previous note we claimed has the solution $T(n) \in \Theta(n \lg(n))$. In this note, we shall verify one half of that claim: that $T(n) \in O(n \lg(n))$; for that purpose, we shall consider a slightly modified recurrence

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

We would like to find c such that we can prove $T(n) \leq cn \lg(n)$ for all $n \geq 1$. But we face an obstacle: when $n = 1$, we have $\lg(n) = 0$ and hence the claim becomes $T(1) \leq 0$ which does not necessarily hold.

We can get around this obstacle in two ways, as described in the rest of this note.

Approach 1: Consider only $n \geq 2$: To show $T(n) \in O(n \lg(n))$ we shall prove, for suitable $c > 0$ to be found along the way, that for all $\mathbf{n \geq 2}$ we have

$$T(n) \leq cn \lg(n). \tag{1}$$

The proof shall be by induction in n, and the inductive step will rely on $\lfloor \frac{n}{2} \rfloor$ also satisfying (1) which we can assume to be the case if $2 \leq \lfloor \frac{n}{2} \rfloor < n$, which will hold for $\mathbf{n \geq 4}$. And for $n \geq 4$, we have the calculation

$$\begin{array}{lll} & & T(n) \\ & = & 2T(\lfloor \frac{n}{2} \rfloor) + n \\ \textbf{IH} & \leq & 2c\lfloor \frac{n}{2} \rfloor \lg(\lfloor \frac{n}{2} \rfloor) + n \\ & \leq & 2c\frac{n}{2} \lg(\frac{n}{2}) + n \\ & = & cn \lg(n) - cn + n \\ & \leq & cn \lg(n) \end{array}$$

which shows the desired $T(n) \leq cn\lg(n)$ provided the last inequality holds, but that will be the case if $c \geq 1$.

We have completed the inductive step, for which $c \geq 1$ will suffice, but we need to also consider the base cases. Since we must prove (1) for all $n \geq 2$, and the inductive step covers when $n \geq 4$, our task is to examine $n = 2$ and $n = 3$. Our obligations are to ensure that

$$\begin{array}{rcl} T(2) & \leq & 2c \\ T(3) & \leq & 3c\lg(3) \end{array}$$

which will obviously hold for big enough c.

Approach 2: Prove it for $\lg(2n)$**:** Alternatively, we can prove (for suitable $c > 0$ to be found along the way) that for all $\mathbf{n \geq 1}$ we have

$$T(n) \leq cn\lg(2n). \tag{2}$$

The proof shall be by induction in n, and the inductive step will rely on $\lfloor \frac{n}{2} \rfloor$ also satisfying (2) which we can assume to be the case if $1 \leq \lfloor \frac{n}{2} \rfloor < n$, which will hold for $\mathbf{n \geq 2}$. And for $n \geq 2$, we have the calculation

$$\begin{array}{rcl} & & T(n) \\ & = & 2T(\lfloor \frac{n}{2} \rfloor) + n \\ \textbf{IH} & \leq & 2c\lfloor \frac{n}{2} \rfloor \lg(2\lfloor \frac{n}{2} \rfloor) + n \\ & \leq & cn\lg(n) + n \\ & = & cn(\lg(n) + 1) - cn + n \\ & = & cn\lg(2n) - cn + n \\ & \leq & cn\lg(2n) \end{array}$$

which shows the desired $T(n) \leq cn\lg(2n)$ provided the last inequality holds, but that will be the case if $c \geq 1$.

We have completed the inductive step, for which $c \geq 1$ will suffice, but we need to also consider the base case(s). Since we must prove (2) for all $n \geq 1$, and the inductive step covers when $n \geq 2$, our task is to examine $n = 1$. Our obligation is to ensure that

$$T(1) \quad \leq \quad c\lg(2) = c$$

and we can conclude that with $c \geq \max(1, T(1))$ the inductive proof will indeed go through.

We have proved that (2) holds for all $n \geq 1$, from which it is easy to see that $T(n) \in O(n\lg(n))$: for $n \geq 2$ we have $\lg(n) \geq 1$ and thus

$$T(n) \leq cn\lg(2n) = cn(\lg(n) + 1) \leq 2cn\lg(n).$$