Tutorial 6

1 Akra-Bazzi

The Akra-Bazzi method is an easy way to solve general Divide and Conquer recurrences. Recall that Divide and Conquer recurrences have the following form:

$$T(x) = \begin{cases} \Theta(1) & \text{for } x \le x_0 \\ \sum_{i=1}^{k} a_i T(b_i x) + g(x) & \text{for } x > x_0 \end{cases}$$

The Akra-Bazzi method is as follows. Set p so that $\sum_{i=1}^{k} a_i b_i^p = 1$. If $|g'(x)| \le x^{O(1)}$, then

$$T(x) = \Theta(x^p(1 + \int_1^x \frac{g(u)}{u^{p+1}}\, du))$$

Here are two useful tricks for applying the Akra-Bazzi method.

Computing p. One difficulty is computing p. This is hard because you have to solve the funny equation $\sum_{i=1}^{k} a_i b_i^p = 1$. A good way to do this is to guess a value for p and compute $\sum_{i=1}^{k} a_i b_i^p$. If the result is greater than 1, then try a larger value of p. If the result is less than 1, then try a smaller value of p. This approach is most efficient if you pick p halfway between the best upper and lower bound you have computed so far. (This is called a binary search.) In this way, you can quicky home in on the correct value for p.

Avoiding computing p. Sometimes computing p is unnecessary! Suppose that $g(x) = x^q$ where $q > p$. Then $T(x)$ can be computed as follows:

$$\begin{aligned} T(x) &= \Theta(x^p(1 + \int_1^x \frac{u^q}{u^{p+1}}\, du)) \\ &= \Theta(x^p(1 + \int_1^x u^{q-p-1}\, du)) \\ &= \Theta(x^p(1 + \int_1^x u^{q-p-1}\, du)) \\ &= \Theta(x^p(1 + \frac{1}{q-p}(x^{q-p} - 1))) \end{aligned}$$

Since $q > p$, the term x^{q-p} dominates all constants, and so we get:

$$\begin{aligned} T(x) &= \Theta(x^p x^{q-p}) \\ &= \Theta(x^q) \end{aligned}$$

The solution does not even depend on the hard-to-compute value p! Therefore, if $g(x) = x^q$, a good trick is to compute $\sum_{i=1}^{k} a_i b_i^q$. If the result is less than 1, then $q > p$ and we know immediately that the solution is $\Theta(x^q)$.

For example, suppose we have the recurrence $T(x) = T(x/3) + T(x/4) + x$. We could compute p by binary search as described above, but this is still a lot of work. Notice that $g(x) = x^1$, so $q = 1$. Since $(1/3)^1 + (1/4)^1 < 1$, we know that $p < q$ and the solution is $\Theta(x^q)$. We never needed to compute p!

2 Master Method

The Akra-Bazzi theorem is easy to apply and quite amazing in its generality. Before it was known however, another fairly general theorem called the *Master theorem* was already quite popular. Most courses in discrete mathematics use that theorem. The Master theorem can be used to solve any recurrence of the form:

$$T(n) = aT(n/b) + n^\alpha$$

where $a > 0$ and $b > 1$.

It is a special case of the Akra-Bazzi theorem, so let us see what asymptotic bounds we can get for this type of recurrence using the Akra-Bazzi theorem. First we find p such that $a(\frac{1}{b})^p = 1$. That is $a = b^p$ or $p = \log_b a$.

Then from the theorem,

$$\begin{aligned} T(n) &= \Theta(n^p(1 + \int_1^n \frac{u^\alpha}{u^{p+1}} du)) \\ &= \Theta(n^p(1 + \int_1^n u^{\alpha-p-1} du)) \end{aligned}$$

In the case when $\alpha \neq p$,

$$= \Theta(n^p(1 + \left[\frac{u^{\alpha-p}}{\alpha - p}\right]_1^n))$$

$$= \Theta(n^p(1 + \frac{1}{\alpha - p}(n^{\alpha-p} - 1))) = \Theta((1 - \frac{1}{\alpha - p})n^p + \frac{1}{\alpha - p}n^\alpha)$$

Assuming a, b, α are fixed constants (i.e., don't change with n),

$$T(n) = \Theta(n^p + n^\alpha) = \Theta(n^{\log_b a} + n^\alpha)$$

If $\alpha = p$, then

$$\begin{aligned} T(n) &= \Theta(n^p(1 + \int_1^n \frac{1}{u} du)) \\ &= \Theta(n^p(1 + \ln n)) = \Theta(n^\alpha \log n) \end{aligned}$$

So all together we have,

$$T(n) = \begin{cases} \Theta(n^{\log_b a}) & \text{if } \log_b a > \alpha \\ \Theta(n^\alpha \log n) & \text{if } \log_b a = \alpha \\ \Theta(n^\alpha) & \text{if } \alpha > \log_b a \end{cases}$$

3 Change of variable

Another useful method to make a difficult looking recurrence easier is by a change of variable. This is similar to the change of variable technique in calculus. For example, consider the recurrence

$$T(n) = 2T(\lfloor\sqrt{n}\rfloor) + \log n$$

with base condition $T(1) = 1$. Note that this recurrence is neither linear nor divide and conquer; however, we can turn it into a divide and conquer recurrence with a change of variables. Here $T(n)$ is defined for all $n \in \mathcal{N}$. Let us first note that $T(n)$ is a non-decreasing function of n.

Claim 1 *For all $n \geq 2$, $T(n) \geq T(n-1)$.*

Proof. By strong indcution on n.
I.H.: $P(n) : T(n) \geq T(n-1)$.
Base: $n = 2$: $T(2) = 2T(\lfloor\sqrt{2}\rfloor) + 1 = 2T(1) + 1 = 3 > 1 = T(1)$.
Ind. Step: Assume $P(k)$ is true for all $2 \leq k \leq n$. We will prove that $P(n+1)$ is true.

$$\begin{aligned} T(n+1) &= 2T(\lfloor\sqrt{n+1}\rfloor) + \log(n+1) \\ &\geq 2T(\lfloor\sqrt{n}\rfloor) + \log n \end{aligned}$$

The latter is true because either $\lfloor\sqrt{n+1}\rfloor = \lfloor\sqrt{n}\rfloor$ or else $\lfloor\sqrt{n+1}\rfloor = \lfloor\sqrt{n}\rfloor + 1$. In the first case, the inequality holds because $T(\lfloor\sqrt{n+1}\rfloor) = T(\lfloor\sqrt{n}\rfloor)$. In the second case, $T(\lfloor\sqrt{n+1}\rfloor) = T(\lfloor\sqrt{n}\rfloor + 1) \geq T(\lfloor\sqrt{n}\rfloor)$ by induction.

But then the right hand side is $T(n)$. So $T(n+1) \geq T(n)$. ■

We will now solve the recurrence assuming n is a power of 2. First, make the substitution $n = 2^m$. The recurrence becomes

$$T(2^m) = 2T(\lfloor 2^{m/2}\rfloor) + m$$

Now define $S(m) = T(\lfloor 2^m\rfloor)$ for all integers $m \geq 0$. Note that since T is non-decreasing, for all $m > 1$, $S(m+1) \geq S(m)$, and hence S is also a non-decreasing function.

Therefore, (remember m is an integer),

$$\begin{aligned} S(m) &= T(\lfloor 2^m\rfloor) \\ &= T(2^m) \\ &= 2T(\lfloor 2^{m/2}\rfloor) + m \\ &\leq 2T(2^{\lceil m/2\rceil}) + m \\ &\leq 2S(\lceil m/2\rceil) + m \end{aligned}$$

That is $S(m) \leq 2S(\lceil m/2 \rceil) + m$ with $S(0) = 1$.

We can then view this as $S(m) \leq 2S(m/2 + h(m)) + m$ where $h(m) = \lceil m/2 \rceil - m/2$. Clearly $h(m) < 1$ and hence it is certainly $h(m) = O(\frac{m}{(\log m)^2})$.

Now to solve this we can apply the special case of the Akra-Bazzi theorem from the previous section to get $S(m) = O(m \log m)$.

From this we have that

$$T(n) \leq T(2^{\lceil \log n \rceil}) = S(\lceil \log n \rceil) = O(\log n \log \log n)$$

.

What about a lower bound? This can be derived in a similar fashion.

$$\begin{aligned} S(m) &= T(\lfloor 2^m \rfloor) \\ &= T(2^m) \\ &= 2T(\lfloor 2^{m/2} \rfloor) + m \\ &\geq 2T(2^{\lfloor m/2 \rfloor}) + m \\ &\geq 2S(\lfloor m/2 \rfloor) + m \end{aligned}$$

That is $S(m) \geq 2S(\lfloor m/2 \rfloor) + m$.

Define a new recurrence $S'(m) = 2S'(\lfloor m/2 \rfloor) + m$ with $S'(0) = 1$.

Claim 2 $S(m) \geq S'(m)$

Proof. By strong Induction on m.

I.H. : $P(m) : S(m) \geq S'(m)$.

Base : $m = 1$: $S(0) = S'(0) = 1$.

Ind. Step: Assume $P(k)$ is true for all $k \leq m$. We will prove $P(m+1)$.

$$\begin{aligned} S(m) &\geq 2S(\lfloor m/2 \rfloor) + m \\ &\geq 2S'(\lfloor m/2 \rfloor) + m \\ &\geq S'(m) \end{aligned}$$

■

Now we can apply Akra-Bazzi to solve the recurrence for S' in a similar way as for the upper bound to get $S'(m) = \Theta(m \log m)$. From the claim above this implies that $S(m) = \Omega(m \log m)$.

Hence $T(n) = \Omega(\log n \log \log n)$. From the upper and lower bounds we conclude that $T(n) = \Theta(\log n \log \log n)$.

The hard part in changing variables to solve recurrences is guessing what substitution to make, but this is a skill that comes out of practice, just like in calculus.

4 Team Problems

1. *Life and Death.* In ancient Mars, cannibalism was an essential way of life. The Marons and the Batons were two species that lived for several thousand years in close contact. Every year, all the Marons sustained themselves by first consuming Batons: Each Maron ate one Baton. Then they reproduced: The Marons tripled in number. Then it was the Batons' turn. The surviving Batons consumed one Maron each and then the Batons reproduced to quadruple in number. This cycle repeated every year. After 1 year there was 1 Maron, and there were two Batons. How many of each will there be of each species at the end of n years?

2. *Stepping up.* Alice the "elegant" climbs up a flight of stairs one or two stairs in one stride. Bob the "elongated beast" on the other hand clambers up 2, 4, 6 or any even number of stairs in one stride. Who can climb a flight of n stairs in more ways? (Assume n is even. A *way* of climbing a flight of n stairs is the sequence of strides taken to get to the top. E.g. to climb up 4 stairs, Alice can go 1,1,1,1 or 1,1,2, or 1,2,1, or 2,1,1 or 2,2 i.e., in 5 different ways in all.)