The Simplified Master Method for Solving Recurrences

Consider recurrences of the form

\[ T(n) = aT(n/b) + n^c \]

for constants \( a \geq 1, b > 1, \) and \( c \geq 0. \) Recurrences of this form include

- **mergesort**, \( T(n) = 2T(n/2) + n, \)
- **Strassen’s algorithm** for matrix multiplication, \( T(n) = 7T(n/2) + n^2, \)
- **binary search**, \( T(n) = T(n/2) + 1, \)

and so on.

We can solve the general form of this recurrence via iteration. Rewriting the recurrence with the recursive component last and using a generic parameter not to be confused with \( n \), we obtain:

\[ T(2) = 2c + aT(2/2) \]

Since our pattern (Equation 1) is valid for any value of \( n \), we may use it to “iterate” the recurrence as follows.

\[
T(n) = n^c + aT(n/b) + n^c(a/b)^c + a^2T(n/b^2) + n^c(a/b)^c + a^2(n/b^2)^c + a^3T(n/b^3) + n^c(a/b)^c + a^2(n/b^2)^c + a^3(n/b^3)^c + a^4T(n/b^4) + n^c(a/b)^c + a^2(n/b^2)^c + a^3(n/b^3)^c + a^4T(n/b^4)
\]

Pulling out the \( n^c \) term common in each of the first four factors, we may simplify this expression and obtain a pattern as follows.

\[
T(n) = n^c + n^c(a/b^c) + n^c(a/b^c)^2 + n^c(a/b^c)^3 + a^4T(n/b^4)
\]

We will next show that the pattern we have established is correct, by induction.

**Claim 1** For all \( k \geq 1, \) \( T(n) = n^c \sum_{j=0}^{k-1} \left( \frac{a}{b^c} \right)^j + a^kT(n/b^k). \)
Proof: The proof is by induction on \( k \). The base case, \( k = 1 \), is trivially true since the resulting equation matches the original recurrence. For the inductive step, assume that the statement is true for \( k = i - 1 \); i.e.,

\[
T(n) = n^c \sum_{j=0}^{i-2} \left( \frac{a}{b^c} \right)^j + a^{i-1} T(n/b^{i-1}).
\]

Our task is then to show that the statement is true for \( k = i \); i.e.,

\[
T(n) = n^c \sum_{j=0}^{i-1} \left( \frac{a}{b^c} \right)^j + a^i T(n/b^i).
\]

This may be accomplished by starting with the inductive hypothesis and applying the definition of the recurrence, as follows.

\[
T(n) = n^c \sum_{j=0}^{i-2} \left( \frac{a}{b^c} \right)^j + a^{i-1} \left[ (n/b^{i-1})^c + a T(n/b^i) \right]
\]

\[
= n^c \sum_{j=0}^{i-2} \left( \frac{a}{b^c} \right)^j + n^c \left( \frac{a}{b^c} \right)^{i-1} + a^i T(n/b^i)
\]

\[
= n^c \sum_{j=0}^{i-1} \left( \frac{a}{b^c} \right)^j + a^i T(n/b^i)
\]

We thus have that \( T(n) = n^c \sum_{j=0}^{k-1} \left( \frac{a}{b^c} \right)^j + a^k T(n/b^k) \) for all \( k \geq 1 \). We next choose a value of \( k \) which causes our recurrence to reach a known base case. Since \( n/b^k = 1 \) when \( k = \log_b n \), and \( T(1) = \Theta(1) \), we have

\[
T(n) = n^c \sum_{j=0}^{\log_b n - 1} \left( \frac{a}{b^c} \right)^j + a^{\log_b n} T(1)
\]

\[
= n^c \sum_{j=0}^{\log_b n - 1} \left( \frac{a}{b^c} \right)^j + n^{\log_b a} \Theta(1)
\]

\[
= n^c \sum_{j=0}^{\log_b n - 1} \left( \frac{a}{b^c} \right)^j + \Theta(n^{\log_b a})
\]

The solution to our recurrence involves the geometric series \( \sum_{j=0}^{\log_b n - 1} \left( \frac{a}{b^c} \right)^j \). In order to bound this series, we must consider three cases: \( a/b^c > 1 \), \( a/b^c = 1 \), and \( a/b^c < 1 \). This is equivalent to considering the cases \( c < \log_b a \), \( c = \log_b a \), and \( c > \log_b a \).

**Case 1:** \( c < \log_b a \) \( \iff \) \( a/b^c > 1 \).

If \( a/b^c > 1 \), we then have

\[
\sum_{j=0}^{\log_b n - 1} \left( \frac{a}{b^c} \right)^j = \frac{(a/b^c)^{\log_b n} - 1}{(a/b^c) - 1} = \Theta((a/b^c)^{\log_b n}) = \Theta \left( \frac{n^{\log_b a}}{n^c} \right).
\]
From this we may conclude that

\[ T(n) = n^c \sum_{j=0}^{\log_b n - 1} \left( \frac{a}{b^c} \right)^j + \Theta(n^{\log_b a}) \]

\[ = n^c \cdot \Theta\left( \frac{n^{\log_b a}}{n^c} \right) + \Theta(n^{\log_b a}) \]

\[ = \Theta(n^{\log_b a}). \]

**Case 2:** \( c = \log_b a \leftrightarrow a/b^c = 1. \)

If \( a/b^c = 1 \), we then have

\[ \log_b n - 1 \sum_{j=0}^{\log_b n - 1} \left( \frac{a}{b^c} \right)^j = \sum_{j=0}^{\log_b n - 1} 1^j = \log_b n. \]

Noting that \( c = \log_b a \), we may then conclude that

\[ T(n) = n^c \sum_{j=0}^{\log_b n - 1} \left( \frac{a}{b^c} \right)^j + \Theta(n^{\log_b a}) \]

\[ = n^c \cdot \log_b n + \Theta(n^{\log_b a}) \]

\[ = \Theta(n^c \log n) = \Theta(n^{\log_b a \log n}) \]

**Case 3:** \( c > \log_b a \leftrightarrow a/b^c < 1. \)

If \( a/b^c < 1 \), we then have

\[ \sum_{j=0}^{\log_b n - 1} \left( \frac{a}{b^c} \right)^j \geq \left( a/b^c \right)^0 = 1 = \Omega(1) \]

and

\[ \sum_{j=0}^{\log_b n - 1} \left( \frac{a}{b^c} \right)^j \leq \sum_{j=0}^{\infty} \left( \frac{a}{b^c} \right)^j = \frac{1}{1 - a/b^c} = O(1) \]

which yields

\[ \sum_{j=0}^{\log_b n - 1} \left( \frac{a}{b^c} \right)^j = \Theta(1). \]

Noting that \( c > \log_b a \), we may then conclude that

\[ T(n) = n^c \sum_{j=0}^{\log_b n - 1} \left( \frac{a}{b^c} \right)^j + \Theta(n^{\log_b a}) \]

\[ = n^c \cdot \Theta(1) + \Theta(n^{\log_b a}) \]

\[ = \Theta(n^c) \]

Note that in each case, either \( c \) or \( \log_b a \) appears in the exponent of the solution, and it is the larger of these two values which appears. If these terms are equal, then an extra log factor appears as well. In summary, we have

| Case 1: \( c < \log_b a \) | \( T(n) = \Theta(n^{\log_b a}) \) |
| Case 2: \( c = \log_b a \) | \( T(n) = \Theta(n^c \log n) = \Theta(n^{\log_b a \log n}) \) |
| Case 3: \( c > \log_b a \) | \( T(n) = \Theta(n^c) \) |