Master Theorem

Computer Science & Engineering 235: Discrete Mathematics

Christopher M. Bourke
cbourke@cse.unl.edu

Master Theorem I

When analyzing algorithms, recall that we only care about the *asymptotic behavior*.

Recursive algorithms are no different. Rather than *solve* exactly the recurrence relation associated with the cost of an algorithm, it is enough to give an asymptotic characterization.

The main tool for doing this is the *master theorem*.

Master Theorem II

Theorem (Master Theorem)

Let $T(n)$ be a monotonically increasing function that satisfies

$$\begin{aligned} T(n) &= aT(\tfrac{n}{b}) + f(n) \\ T(1) &= c \end{aligned}$$

where $a \geq 1, b \geq 2, c > 0$. If $f(n) \in \Theta(n^d)$ where $d \geq 0$, then

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

Master Theorem

Pitfalls

You *cannot* use the Master Theorem if

- $T(n)$ is not monotone, ex: $T(n) = \sin n$
- $f(n)$ is not a polynomial, ex: $T(n) = 2T(\frac{n}{2}) + 2^n$
- b cannot be expressed as a constant, ex: $T(n) = T(\sqrt{n})$

Note here, that the Master Theorem does *not* solve a recurrence relation.

Does the base case remain a concern?

Master Theorem

Example 1

Let $T(n) = T\left(\frac{n}{2}\right) + \frac{1}{2}n^2 + n$. What are the parameters?

$$\begin{aligned} a &= 1 \\ b &= 2 \\ d &= 2 \end{aligned}$$

Therefore which condition?

Since $1 < 2^2$, case 1 applies.

Thus we conclude that

$$T(n) \in \Theta(n^d) = \Theta(n^2)$$

Master Theorem

Example 2

Let $T(n) = 2T\left(\frac{n}{4}\right) + \sqrt{n} + 42$. What are the parameters?

$$\begin{aligned} a &= 2 \\ b &= 4 \\ d &= \tfrac{1}{2} \end{aligned}$$

Therefore which condition?

Since $2 = 4^{\frac{1}{2}}$, case 2 applies.

Thus we conclude that

$$T(n) \in \Theta(n^d \log n) = \Theta(\sqrt{n} \log n)$$

Master Theorem

Example 3

Let $T(n) = 3T\left(\frac{n}{2}\right) + \frac{3}{4}n + 1$. What are the parameters?

$$\begin{aligned} a &= 3 \\ b &= 2 \\ d &= 1 \end{aligned}$$

Therefore which condition?

Since $3 > 2^1$, case 3 applies. Thus we conclude that

$$T(n) \in \Theta(n^{\log_b a}) = \Theta(n^{\log_2 3})$$

Note that $\log_2 3 \approx 1.5849\ldots$. Can we say that $T(n) \in \Theta(n^{1.5849})$?

"Fourth" Condition

Recall that we cannot use the Master Theorem if $f(n)$ (the non-recursive cost) is not polynomial.

There is a limited 4-th condition of the Master Theorem that allows us to consider polylogarithmic functions.

Corollary

If $f(n) \in \Theta(n^{\log_b a} \log^k n)$ *for some* $k \geq 0$ *then*

$$T(n) \in \Theta(n^{\log_b a} \log^{k+1} n)$$

This final condition is fairly limited and we present it merely for completeness.

"Fourth" Condition

Example

Say that we have the following recurrence relation:

$$T(n) = 2T\left(\frac{n}{2}\right) + n \log n$$

Clearly, $a = 2, b = 2$ but $f(n)$ is not a polynomial. However,

$$f(n) \in \Theta(n \log n)$$

for $k = 1$, therefore, by the 4-th case of the Master Theorem we can say that

$$T(n) \in \Theta(n \log^2 n)$$