5. Rekursionen

- Laufzeiten insbesondere von Divide&Conquer Algorithmen werden häufig durch Rekursions-gleichungen beschrieben.

- Werden eine allgemeine Methode kennenlernen, um solche Gleichungen zu lösen, die Rekursionsbaum-Methode.

- Diese Methode kann verfeinert werden, um das Master Theorem für Rekursionsgleichungen zu beweisen.

Master-Theorem (Einfache Version)

Satz 5.1: Für positive Konstanten a,b,c mit $n=b^k$ für eine natürliche Zahl k sei

$$T(n) \leq c \qquad \text{falls } n=1$$

$$T(n) \leq a \cdot T(n/b) + c \cdot n \qquad \text{falls } n>1$$

Dann gilt

$$T(n) = \Theta(n) \qquad \text{falls } a<b$$

$$T(n) = \Theta(n \log_b n) \qquad \text{falls } a=b$$

$$T(n) = \Theta(n^{\log_b a}) \qquad \text{falls } a>b$$

Master-Theorem

Beweis:

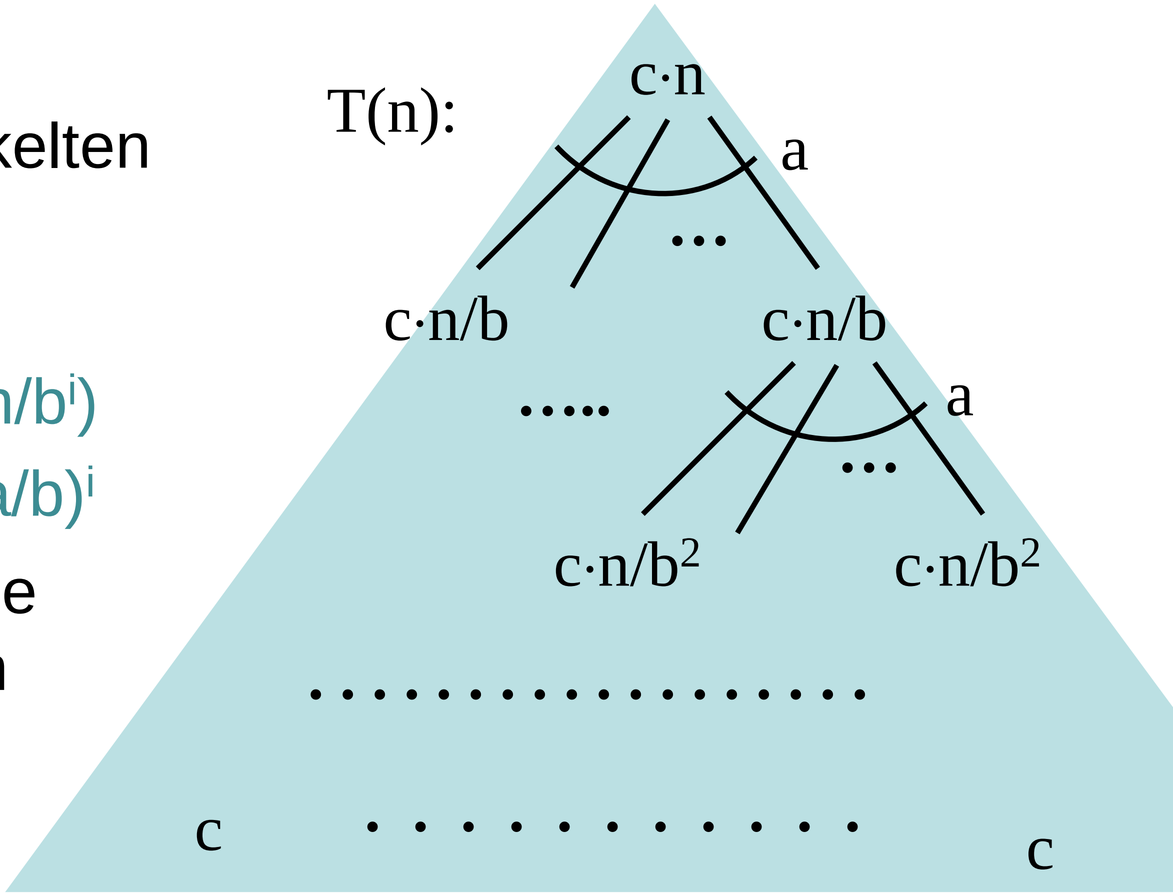
- Betrachte abgewickelten Laufzeitbaum.
- Daraus folgt:

  $T(n) \leq \Sigma_{i=0}^{k} a^i \, (c \cdot n/b^i)$

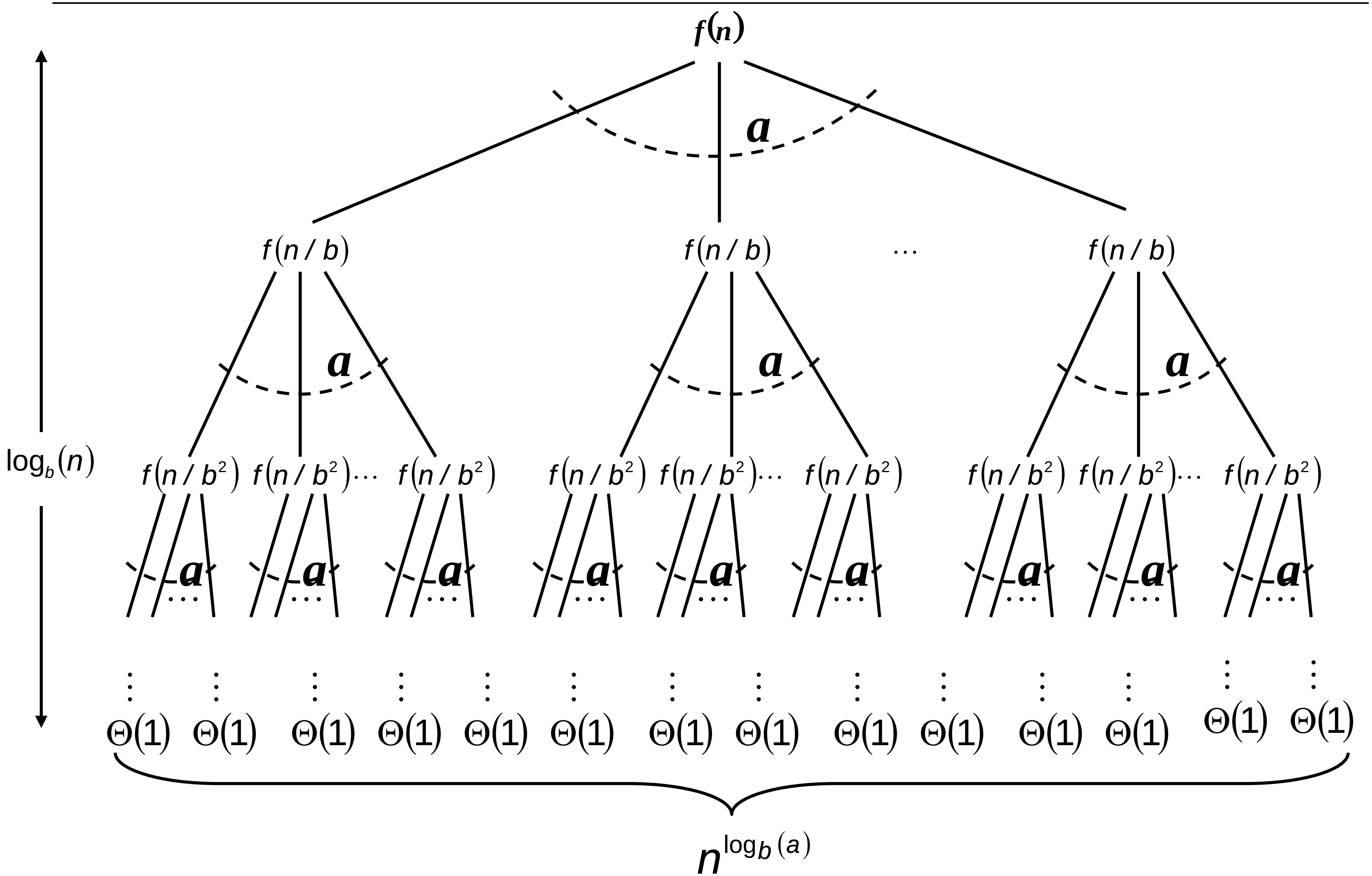
  $\leq c \cdot n \, \Sigma_{i=0}^{k} (a/b)^i$

- Einsetzung der Fälle ergibt das Theorem (verwende für $a \neq b$

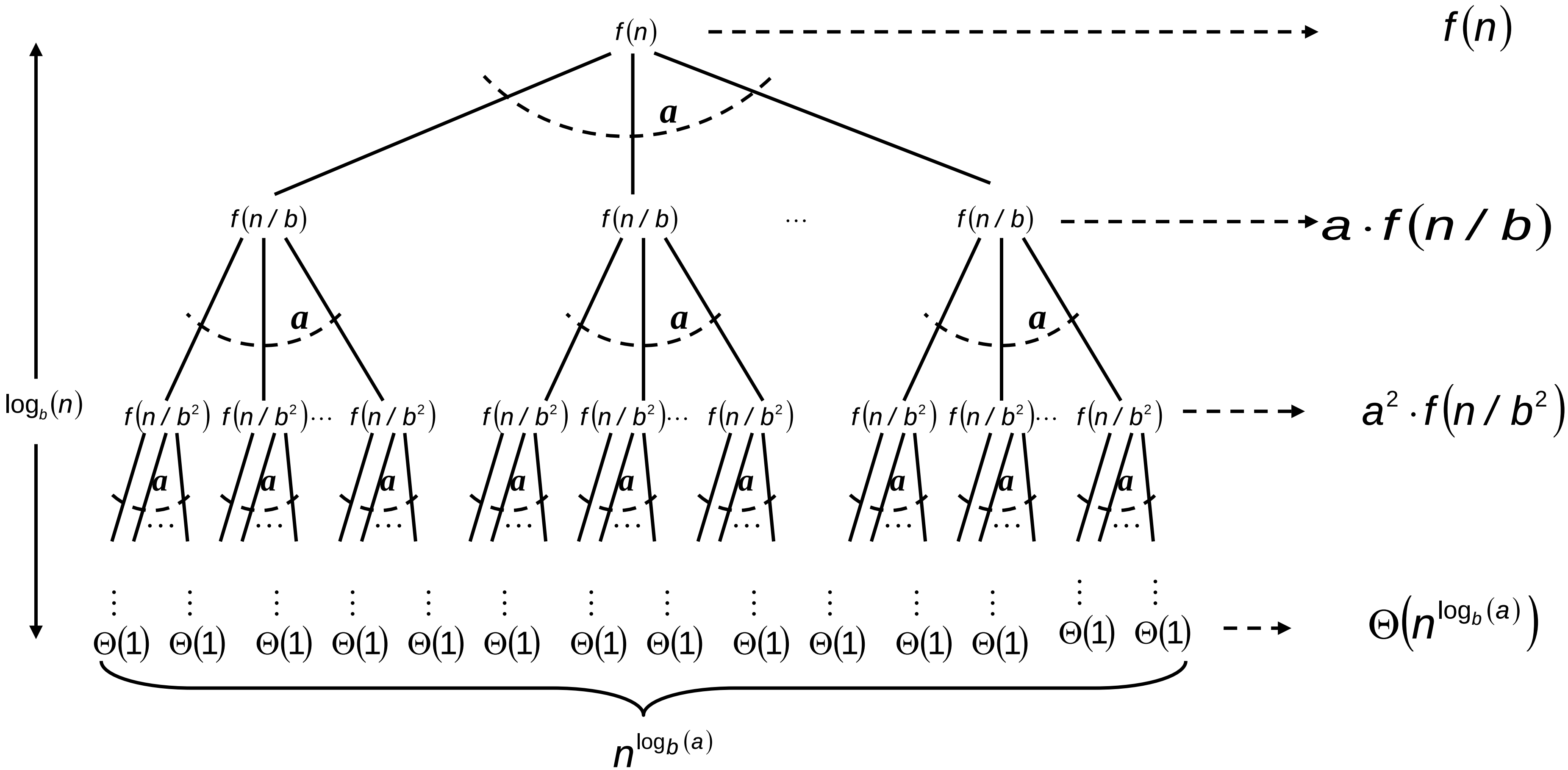
  $\Sigma_{i=0}^{k} z^i = \frac{z^{k+1}-1}{z-1}$)



Rekursionsbaum für $T(n)=aT(n/b)+f(n)$



Rekursionsbaum und Summation



$$\text{Insgesamt}: \Theta\left(n^{\log_b(a)}\right) + \sum_{j=0}^{\log_b(n)-1} a^j f\left(n / b^j\right)$$

Lösen durch Rekursionsbäume - Beispiel

- Betrachten $T(n) = \begin{cases} 3T(\lfloor n/4 \rfloor) + cn^2, & n > 1 \\ c, & n = 1 \end{cases}, c > 0.$

- Erhalten durch Ignorieren der Abrundung und mit Rekursionsbaum

$$T(n) = cn^2 + \frac{3}{16}cn^2 + \cdots + \left(\frac{3}{16}\right)^{\log_4(n)-1} cn^2 + cn^{\log_4(3)}$$

$$= \sum_{i=0}^{\log_4(n)-1} \left(\frac{3}{16}\right)^i cn^2 + cn^{\log_4(3)} \qquad .$$

$$= \frac{(3/16)^{\log_4(n)} - 1}{(3/16) - 1} cn^2 + cn^{\log_4(3)} = \mathrm{O}\left(n^2\right)$$

Das (allgemeine) Master-Theorem

Satz 5.2 (Master Theorem):

Seien $a,b \geq 1$ Konstanten, sei $f(n)$ eine Funktion und sei $T(n)$ definiert durch die Rekursionsgleichung

$$T(n) = aT(n/b) + f(n).$$

Hierbei kann n/b auch durch $\lfloor n/b \rfloor$ oder $\lceil n/b \rceil$ ersetzt werden. Dann kann $T(n)$ folgendermaßen abgeschätzt werden.

1. Ist $f(n) = O\left(n^{\log_b(a)-\varepsilon}\right)$ für $\varepsilon > 0$, dann gilt $T(n) = O\left(n^{\log_b(a)}\right)$.
2. Ist $f(n) = \Theta\left(n^{\log_b(a)}\right)$, dann gilt $T(n) = O\left(n^{\log_b(a)} \cdot \log(n)\right)$.
3. Ist $f(n) = \Omega\left(n^{\log_b(a)+\varepsilon}\right)$ für $\varepsilon > 0$, und ist $af(n/b) \leq cf(n)$ für eine Konstante $c < 1$ und $n \to \infty$, dann gilt $T(n) = \Theta(f(n))$.