Solving Recurrences: Recursion tree

Dr. Ahmed Zekri

Adapted from lectures of

Prof. Charles Leiserson , MIT &

CLRC textbook 3rd ed ch4, pages 88 ~ 97

Outline

- Solving Recurrences
  – Recursion tree method

Solving recurrences

- The analysis of merge sort *required us to solve a recurrence of the form*

  $$T(n) = 2\,T(n/2) + \Theta(n)$$

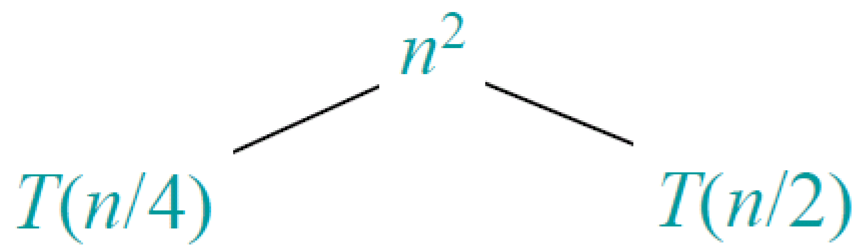
- <u>*How to determine T(n)?*</u>
- *There are three methods:*
  - *Recursion Tree*
  - *Master Theorem*
  - *Substitution method (later)*

Recursion-tree method

- A recursion tree models the costs (time) of a recursive execution of an algorithm.
- each node represents the cost of a single subproblem somewhere in the set of recursive function invocations
- We sum the costs within each level of the tree to obtain a set of per-level costs, and then we sum all the per-level costs to determine the total cost of all levels of the recursion.
- The recursion tree method is good for generating guesses for the substitution method.

Example

Solve $T(n) = T(n/4) + T(n/2) + n^2$:

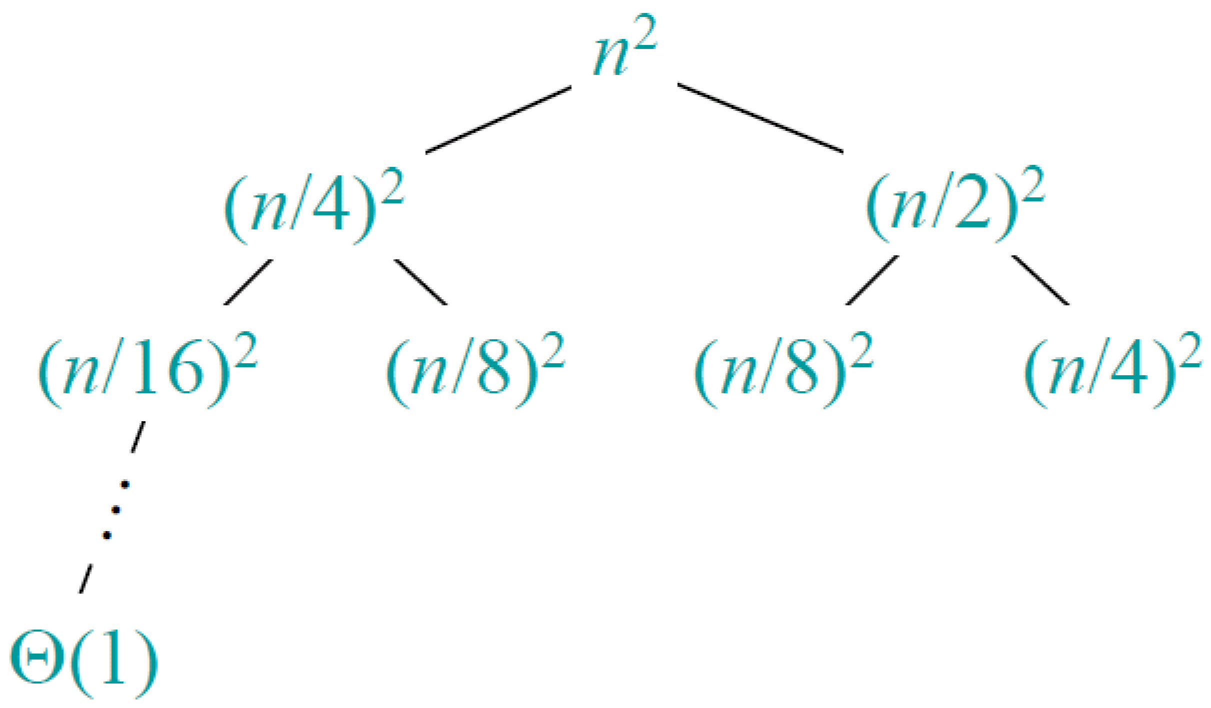


Solve $T(n) = T(n/4) + T(n/2) + n^2$:

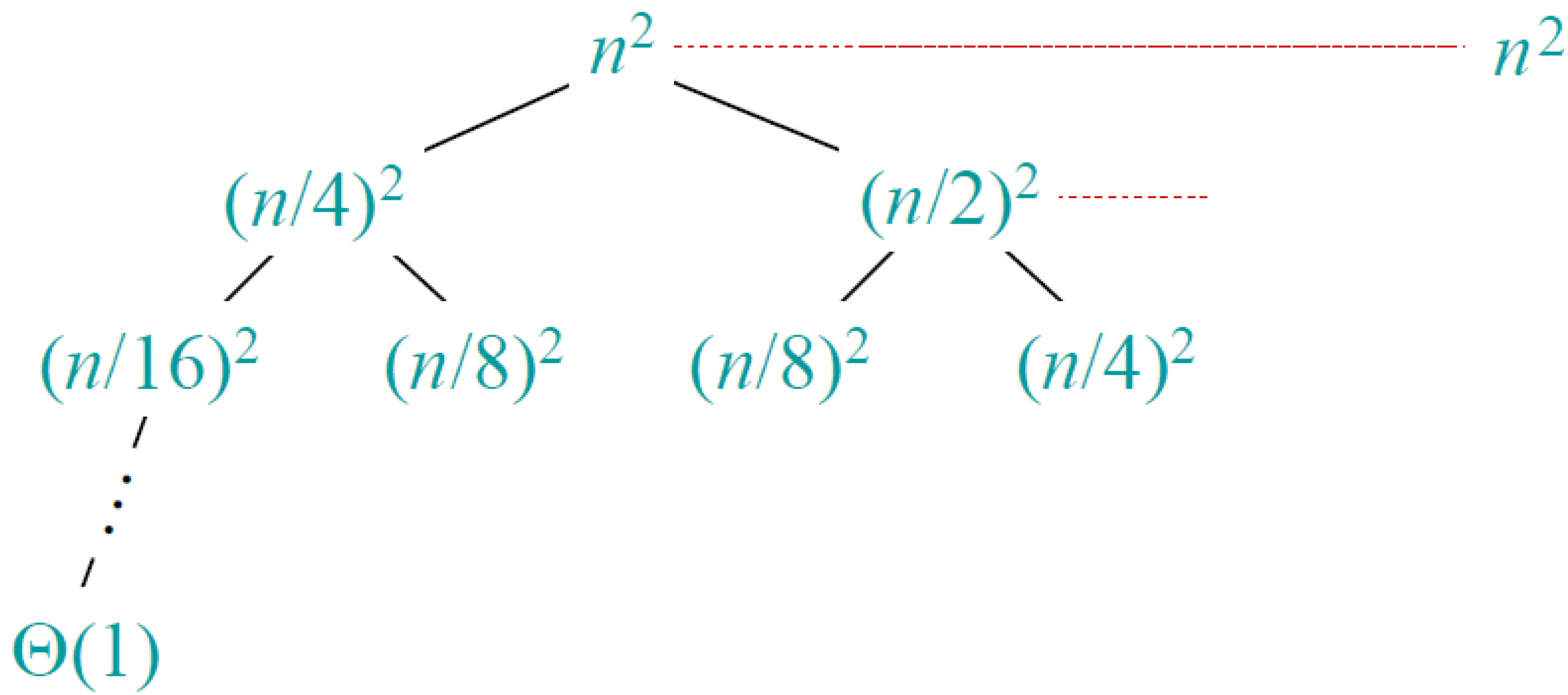
$$n^2$$

$$(n/4)^2 \qquad\qquad (n/2)^2$$

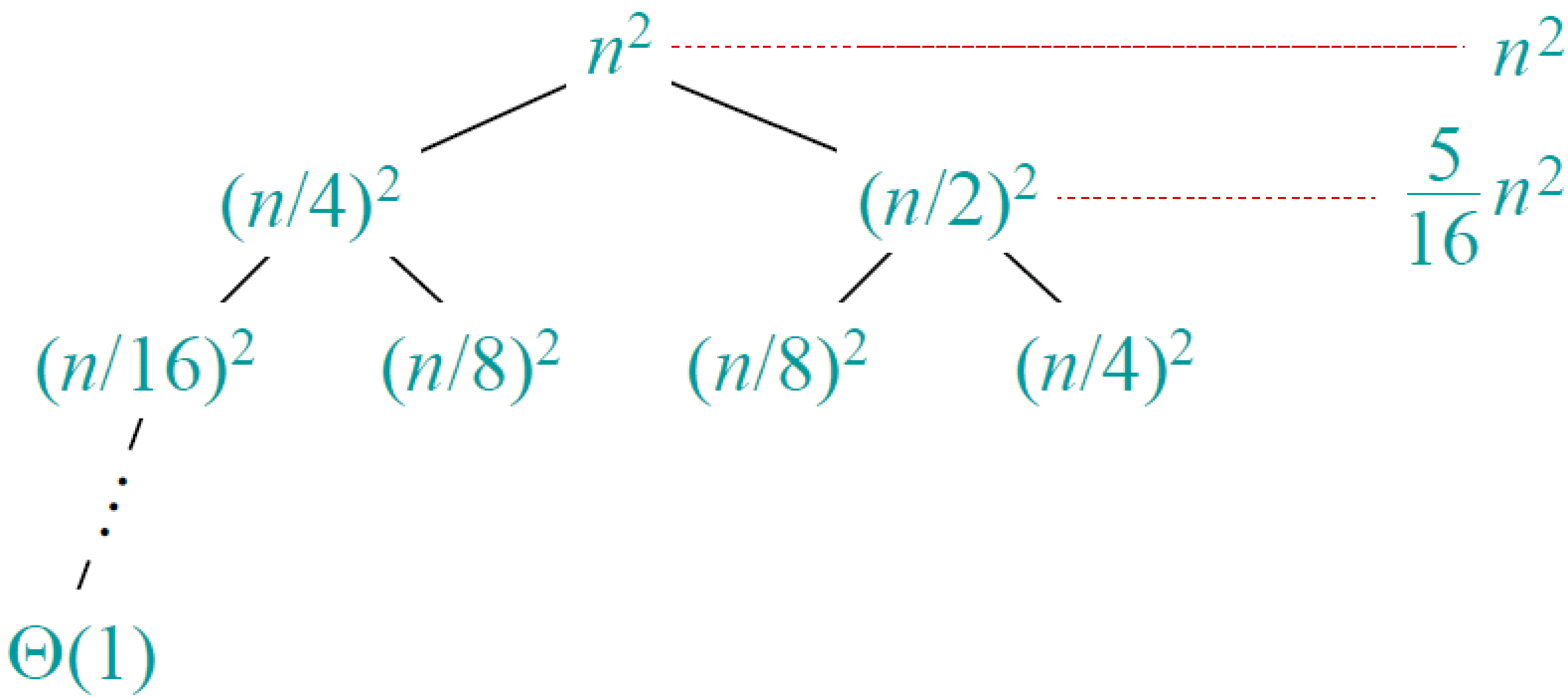
$$T(n/16) \quad T(n/8) \quad T(n/8) \quad T(n/4)$$


n^2
$(n/4)^2$
$(n/2)^2$
$(n/16)^2$
$(n/8)^2$
$(n/8)^2$
$(n/4)^2$
$\Theta(1)$

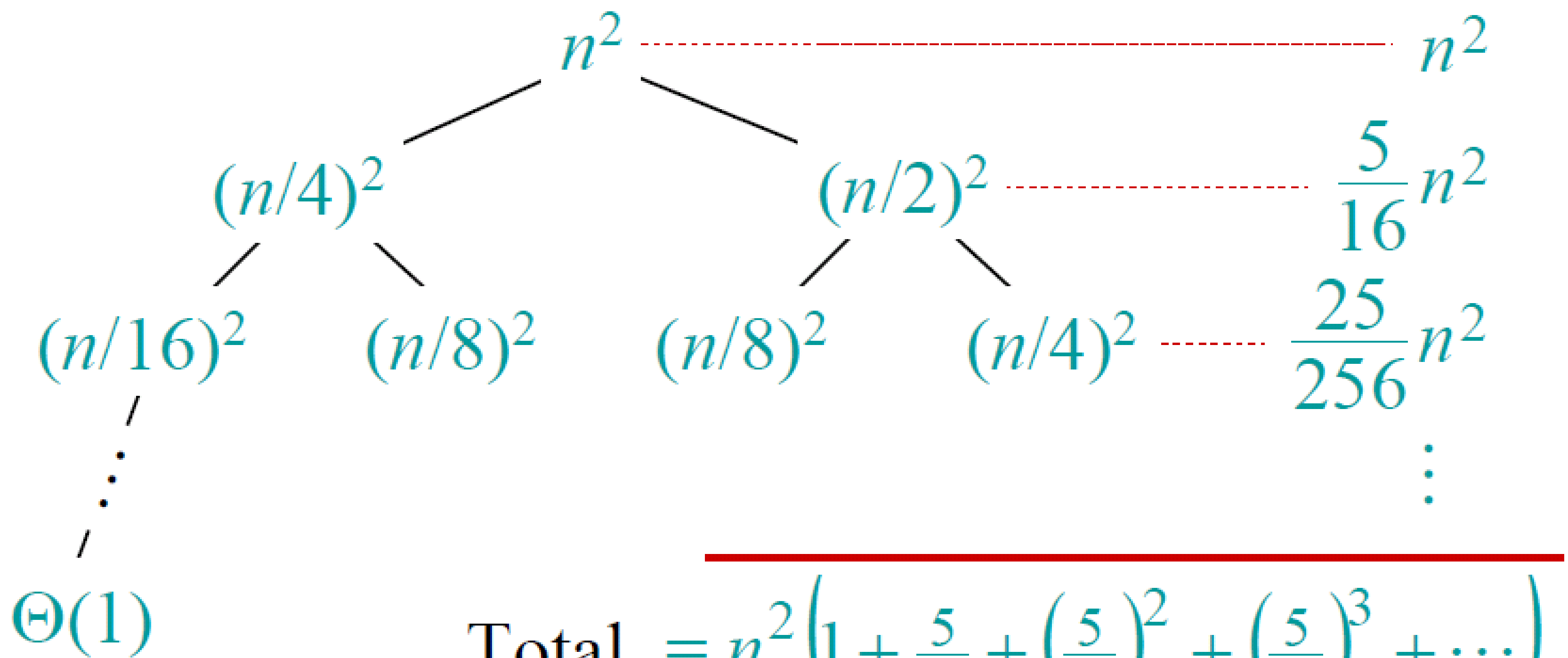
Solve $T(n) = T(n/4) + T(n/2) + n^2$:



Solve $T(n) = T(n/4) + T(n/2) + n^2$:

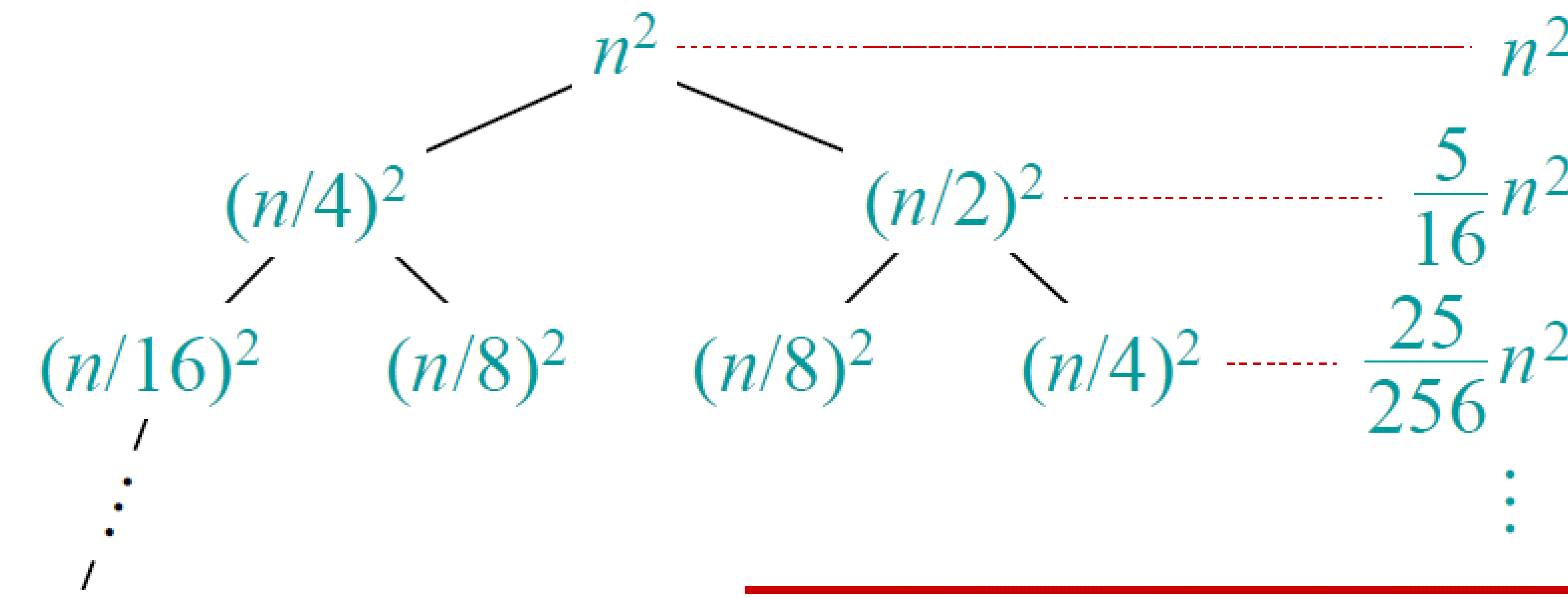


Solve $T(n) = T(n/4) + T(n/2) + n^2$:



$$\text{Total} = n^2\left(1 + \frac{5}{16} + \left(\frac{5}{16}\right)^2 + \left(\frac{5}{16}\right)^3 + \cdots\right)$$

Solve $T(n) = T(n/4) + T(n/2) + n^2$:



$\Theta(1)$

Total $= n^2\left(1 + \frac{5}{16} + \left(\frac{5}{16}\right)^2 + \left(\frac{5}{16}\right)^3 + \cdots\right)$

$= \Theta(n^2)$ *geometric series*

Geometric Series

$$1 + x + x^2 + \cdots + x^n = \frac{1 - x^{n+1}}{1 - x} \quad \text{for } x \neq 1$$

$$1 + x + x^2 + \cdots = \frac{1}{1 - x} \quad \text{for } |x| < 1$$