Solve The Following Recurrence Using Substitution Method: T(n) =T(n/3) + T(n/5) + 90n, T(1) = 45

Solve The Following Recurrence Using Substitution Method: T(n) = T(n/3) + T(n/5) + 90n, T(1) = 45

Introduction

Recurrence relations are fundamental in computer science, especially in analyzing the time complexity of recursive algorithms. They describe how the runtime or space requirements of an algorithm grow concerning the input size. Solving recurrence relations provides insight into the efficiency of algorithms, guiding optimization and implementation choices.

One common method for solving recurrence relations is the substitution method—also known as the guess-and-verify method. This technique involves hypothesizing a solution based on the recurrence's structure and then mathematically proving its correctness through induction.

In this article, we focus on a specific recurrence relation:

$$T(n) = T\left(\frac{n}{3}\right) + T\left(\frac{n}{5}\right) + 90n, \quad T(1) = 45$$

Our goal is to solve this recurrence relation using the substitution method, derive an asymptotic bound, and understand the behavior of the function as n grows large.

Understanding the Recurrence Relation

The Recurrence Components

The recurrence relation:

$$T(n) = T\left(\frac{n}{3}\right) + T\left(\frac{n}{5}\right) + 90n$$

contains three parts:

1. Recursive Calls:
- $T(n/3)$
- $T(n/5)$

2. Combining Cost:
- $90n$, which is the cost of work outside the recursive calls (e.g., dividing, merging).

3. Base Case:
- $T(1) = 45$

Challenges in Solving

Unlike standard recurrence relations with a single recursive term, this recurrence involves two recursive calls with different input sizes ($n/3$ and $n/5$). This complicates solving using common methods like the Master theorem directly. Therefore, the substitution method becomes a valuable approach here.

Approach to Solving Using the Substitution Method

The substitution method involves:

1. Making an educated guess about the form of the solution $T(n)$.
2. Using mathematical induction to verify that the guess holds true.
3. Adjusting the guess if necessary.

Given the additive nature of the recurrence, and the linear cost $90n$, a common initial hypothesis is that $T(n)$ is proportional to n times a logarithmic factor or a polynomial. We will test these possibilities.

Step-by-Step Solution

Step 1: Making an Initial Guess

Because the recurrence involves dividing n by constants and adding a linear term, a reasonable first guess is:

$$T(n) = O(n \log n)$$

or perhaps $T(n) = O(n)$.

However, to be precise, given the recurrence's structure, let's hypothesize:

$$T(n) \leq c \cdot n \log n$$

for some constant $c > 0$.

Step 2: Expressing the Recurrence with the Guess

Suppose:

$$T(n) \leq c \cdot n \log n$$

Our goal is to verify if this inequality satisfies the recurrence:

\[

$$
T(n) \leq T(n/3) + T(n/5) + 90n
$$

Substituting the inductive hypotheses:

$$
T(n/3) \leq c \cdot \frac{n}{3} \log \left(\frac{n}{3}\right)
$$

$$
T(n/5) \leq c \cdot \frac{n}{5} \log \left(\frac{n}{5}\right)
$$

Therefore,

$$
T(n) \leq c \cdot \frac{n}{3} \log \left(\frac{n}{3}\right) + c \cdot \frac{n}{5} \log \left(\frac{n}{5}\right) + 90 n
$$

Our task reduces to checking if:

$$
c \cdot n \log n \geq c \cdot \frac{n}{3} \left(\log n - \log 3 \right) + c \cdot \frac{n}{5} \left(\log n - \log 5 \right) + 90 n
$$

which simplifies to:

$$
c n \log n \geq c n \left(\frac{1}{3} \log n - \frac{1}{3} \log 3 + \frac{1}{5} \log n - \frac{1}{5} \log 5 \right) + 90 n
$$

Combine like terms:

$$
c n \log n \geq c n \left(\left(\frac{1}{3} + \frac{1}{5} \right) \log n - \frac{1}{3} \log 3 - \frac{1}{5} \log 5 \right) + 90 n
$$

Calculate the sum:

$$
\frac{1}{3} + \frac{1}{5} = \frac{5}{15} + \frac{3}{15} = \frac{8}{15}
$$

Thus,

$$
c n \log n \geq c n \left(\frac{8}{15} \log n - \frac{1}{3} \log 3 - \frac{1}{5} \log 5 \right) + 90 n
$$

Dividing both sides by n:

$$
c \log n \geq c \left(\frac{8}{15} \log n - \frac{1}{3} \log 3 - \frac{1}{5} \log 5 \right) + 90
$$

Bring all to one side:

$$
c \log n - c \frac{8}{15} \log n \geq - c \frac{1}{3} \log 3 - c \frac{1}{5} \log 5 + 90
$$

Simplify the left:

$$
c \left(1 - \frac{8}{15} \right) \log n \geq - c \frac{1}{3} \log 3 - c \frac{1}{5} \log 5 + 90
$$

Calculate $1 - \frac{8}{15} = \frac{15}{15} - \frac{8}{15} = \frac{7}{15}$:

$$
c \cdot \frac{7}{15} \log n \geq - c \left(\frac{1}{3} \log 3 + \frac{1}{5} \log 5 \right) + 90
$$

Note that for sufficiently large n, $\log n$ dominates the constants. Therefore, for large n, the inequality will hold if:

$$
c \cdot \frac{7}{15} \log n \geq \text{some constant} + 90
$$

which is true for sufficiently large n.

Conclusion: The hypothesis $T(n) = O(n \log n)$ is plausible.

Step 3: Formalizing the Inductive Proof

To confirm, we proceed with an induction proof.

Claim: For some constants $c \geq 1$, $T(n) \leq c n \log n$ for all $n \geq 1$.

Base Case: $n=1$:

$$
T(1) = 45
$$

Check whether:

$$
45 \leq c \cdot 1 \cdot \log 1
$$

Since $\log 1 = 0$, this inequality holds only if c is large enough. Alternatively, for small n, we can define a constant c to cover the base case.

Inductive Step:

Assume $T(k) \leq c k \log k$ for all $k < n$.

Then,

$$
T(n) = T(n/3) + T(n/5) + 90 n
$$

Applying the inductive hypothesis:

$$
T(n) \leq c \frac{n}{3} \log \left(\frac{n}{3} \right) + c \frac{n}{5} \log \left(\frac{n}{5} \right) + 90 n
$$

As shown earlier, this simplifies to:

$$
T(n) \leq c n \left(\frac{1}{3} \log n - \frac{1}{3} \log 3 + \frac{1}{5} \log n - \frac{1}{5} \log 5 \right) + 90 n
$$

which is less than or equal to:

$$
c n \left(\frac{8}{15} \log n \right) + \text{constant terms} + 90 n
$$

Choosing c sufficiently large to absorb the constants, and

Frequently Asked Questions

How do you approach solving the recurrence T(n) = T(n/3) + T(n/5) + 90n using the substitution method?

You assume a form for T(n), such as T(n) = O(n), and then verify if this assumption holds by substituting it back into the recurrence to find a constant that satisfies the inequality. This involves expressing T(n) in terms of T(n/3) and T(n/5), and simplifying to establish an upper bound.

What is the initial step in applying the substitution method to the recurrence T(n) = T(n/3) + T(n/5) + 90n?

The initial step is to hypothesize a solution of the form T(n) = C n for some constant C, and then substitute this into the recurrence to check whether the assumption leads to a valid bound.

How do the terms T(n/3) and T(n/5) influence the overall solution of the recurrence?

Since T(n/3) and T(n/5) are smaller subproblems, their contributions can be bounded if the assumed solution is proportional to n. The key is to show that the sum of these recursive calls does not exceed a constant multiple of n, ensuring a linear solution.

What is the significance of the base case T(1) = 45 in solving this recurrence?

The base case provides the initial condition needed to verify the correctness of the solution and to determine the constant factors in the asymptotic bound when using the substitution method.

Can the substitution method be used to find an exact solution for T(n) in this recurrence?

No, the substitution method typically helps establish an asymptotic bound (like Big O notation) rather than an exact closed-form solution. Exact solutions often require solving the recurrence directly or using generating functions.

What is the expected order of growth for T(n) in this recurrence?

Given the recurrence structure and the linear term 90n, the expected order of growth for T(n) is linear, i.e., T(n) = O(n).

How do you determine the constant C when assuming T(n) = C n in the substitution method?

You substitute T(n) = C n into the recurrence, resulting in an inequality involving C. Solving this inequality for C ensures that the assumed form is a valid upper bound, often leading to choosing C sufficiently large.

What is the final step after assuming T(n) = C n in the substitution method?

The final step is to verify that the chosen C satisfies the inequality derived from substituting into the recurrence. If it does, then T(n) = O(n) is established as an upper bound for the recurrence.

Additional Resources

Solving recurrence relations is a fundamental aspect of analyzing the efficiency of algorithms, especially those that divide problems into subproblems of varying sizes. Among the various methods for solving recurrences, the substitution method stands out due to its intuitive approach of guessing the form of the solution and then proving it via induction. In this article, we delve into a specific recurrence relation: T(n) = T(n/3) + T(n/5) + 90n, with the base case T(1) = 45, and explore how to solve it using the substitution method. Our goal is to not only derive the asymptotic behavior of T(n) but also to understand the techniques and reasoning involved in tackling such complex recurrences.

Understanding the Recurrence Relation

Problem Statement and Context

The recurrence relation under consideration is:

$$
T(n) = T\left(\frac{n}{3}\right) + T\left(\frac{n}{5}\right) + 90n, \quad T(1) = 45
$$

This relation models algorithms or divide-and-conquer processes where the problem splits into two subproblems of sizes n/3 and n/5, respectively, with an additional linear cost of 90n at each step.

Such a recurrence is more complex than the classic divide-and-conquer recurrence (like the Master Theorem typically handles), because it involves two different recursive calls of different sizes, making it an interesting candidate for the substitution method.

Key Challenges

- Handling multiple recursive calls with different reduction factors.
- Determining the dominant term in the solution (whether the recursive calls or the linear term 90n dominates).
- Guessing an appropriate form for the solution, considering the non-uniform division.

Analyzing the Recurrence: Approach and Strategy

Intuition and Initial Observations

When solving recurrences, a common starting point is to analyze the recurrence tree or attempt to find a pattern in the total work done at each level. For the recurrence:

$$
T(n) = T(n/3) + T(n/5) + 90n
$$

each recursive call reduces the problem size, but by different factors, leading to a complex tree structure. The total work at each level combines the contributions from subproblems of different sizes.

The key insight is that, as n decreases, the recursive calls continue until reaching the base case T(1). The sum of the costs across levels resembles a geometric series, but because the reductions differ, the analysis is less straightforward.

Goal of the Substitution Method

The substitution method involves:
1. Making an educated guess (hypothesis) about the form of T(n).
2. Proving that the guess is valid via mathematical induction.
3. Refining the guess if necessary.

Given that the cost at each level involves a linear term (90n), it’s reasonable to suspect that T(n) is at least linear or slightly super-linear. As the recursive calls are of fractional sizes, the overall complexity might be dominated by the linear term, but the recursive structure could introduce additional overhead.

Step-by-Step Solution Using the Substitution Method

Step 1: Hypothesize a Solution Form

Based on the recurrence's structure, a plausible guess is that T(n) grows linearly with n, i.e.,

$$
T(n) = O(n)
$$

or possibly $T(n) = c \cdot n \log n$ if the recursion tree's depth introduces a logarithmic factor.

Given the linear cost 90n at each level and the recursive splits, a natural initial hypothesis is:

$$
T(n) \leq a n
$$

for some constant $a > 0$. Our goal is to verify whether this bound holds for some a.

Step 2: Assume an Inductive Hypothesis

Suppose for all $k < n$,

$$
T(k) \leq a k
$$

where a is a constant to be determined.

We aim to show:

$$
T(n) \leq a n
$$

by substitution into the recurrence.

Step 3: Substituting into the Recurrence

Applying the inductive hypothesis:

$$
T(n) = T(n/3) + T(n/5) + 90n \leq a \frac{n}{3} + a \frac{n}{5} + 90n
$$

Simplify the sum:

$$
T(n) \leq a n \left(\frac{1}{3} + \frac{1}{5}\right) + 90n
$$

Calculate the sum inside parentheses:

$$
\frac{1}{3} + \frac{1}{5} = \frac{5}{15} + \frac{3}{15} = \frac{8}{15}
$$

Thus:

$$
T(n) \leq a n \frac{8}{15} + 90n
$$

To ensure $T(n) \leq a n$, we require:

$$
a n \geq a n \frac{8}{15} + 90n
$$

Divide both sides by n (assuming $n > 0$):

$$
a \geq a \frac{8}{15} + 90
$$

Rearranged:

$$
a - a \frac{8}{15} \geq 90
$$

Factor out a:

$$
a\left(1 - \frac{8}{15}\right) \geq 90
$$

Compute the parenthesis:

$$
1 - \frac{8}{15} = \frac{15}{15} - \frac{8}{15} = \frac{7}{15}
$$

So:

$$
a \cdot \frac{7}{15} \geq 90 \implies a \geq 90 \times \frac{15}{7}
$$

Calculate:

$$
a \geq 90 \times \frac{15}{7} = \frac{90 \times 15}{7} = \frac{1350}{7} \approx 192.86
$$

Conclusion:

Choosing $a \geq 193$ ensures the inductive hypothesis holds, i.e.,
$$
T(n) \leq 193 n
$$
for sufficiently large n. This suggests that $T(n)$ is $O(n)$.

Refining the Solution and Confirming the Asymptotic Behavior

Step 4: Formalizing the Bound

From the derivation, the substitution indicates that $T(n)$ is bounded above by a linear function. To confirm this rigorously, we can establish the following:

- For the base case $n=1$, $T(1) = 45$, which is less than $193 \times 1 = 193$.
- Assume $T(k) \leq 193k$ for all $k < n$.
- Show that $T(n) \leq 193 n$.

Substituting back:
$$
T(n) \leq T(n/3) + T(n/5) + 90n \leq 193 \frac{n}{3} + 193 \frac{n}{5} + 90n
$$
which simplifies to:
$$
T(n) \leq 193 n \left(\frac{1}{3} + \frac{1}{5}\right) + 90n = 193 n \frac{8}{15} + 90n
$$
As previously calculated, this is less than or equal to:
$$
193 n
$$
for the chosen a. This completes the induction, confirming the linear asymptotic bound.

Step 5: Asymptotic Complexity

Since the recurrence is dominated by the linear term $90n$, and recursive calls reduce problem size by factors of 3 and 5, which are constants, the overall time complexity is linear:
$$
T(n) = \Theta(n)
$$

Key insight:
The recursive calls' contribution diminishes proportionally to n, and the linear cost at each step accumulates in a manner that remains proportional to n.

Conclusion: Final Solution and Implications

The comprehensive analysis reveals that the recurrence relation:

$$
T(n) = T(n/3) + T(n/5) + 90n, \quad T(1) = 45
$$

solves to:

$$
T(n) = \Theta(n)
$$

This indicates that the total computational effort grows linearly with the input size n. The key steps involved hypothesizing a linear bound, verifying it through substitution, and confirming its validity via induction.

Implications for Algorithm Design:
Understanding such recurrences helps algorithm designers predict performance, especially for divide-and-conquer algorithms with multiple recursive branches. Recognizing that the total work remains linear despite multiple recursive calls provides confidence in the scalability of the algorithm.

Broader Context:
The substitution method exemplifies a powerful technique in algorithm analysis, allowing precise bounds even for recurrences involving

Solve The Following Recurrence Using Substitution Method T N T N 3 T N 5 90n T 1 45

Solve The Following Recurrence Using Substitution Method T N T N 3 T N 5 90n T 1 45

Related Articles

- In His History Of The Peloponnesian War, The Greek Historian Thucydides Broke With Tradition By
- Using Your Eyes, How Does The Double Slit Pattern Change As You Increase The Slit Separation?
- A Curve Has Equation Y = X -kx +1.When X = 2, The Gradient Of The Curve Is 6.(a) Show That K = 1.5.

Back to Home