

COMP170

Discrete Mathematical Tools for Computer Science

Advanced Induction

Version 1: Last updated, Jan 1, 2006

*Discrete Math for Computer Science
K. Bogart, C. Stein and R.L. Drysdale
Section 4.5, pp. 189-193*

Slides © 2005 by M. J. Golin and G. Trippen

*Note: We have skipped section 4.4 of the textbook because the material it contains, especially the **Master Theorem**, will be taught in later classes, e.g., COMP271*

More Advanced Induction

- Induction, as we've seen it so far, was about defining a statement $p(n)$, and then proving $p(n-1) \Rightarrow p(n)$ or $(p(1) \wedge p(2) \wedge \cdots \wedge p(n-1)) \Rightarrow p(n)$
- In “practice”, in some real induction proofs, $p(n)$ might not be fully defined *before* we start the proof and will only be fully described *during* the description of the proof
- In some cases it also helps to use a **stronger** induction hypothesis than the “natural” one.

We will illustrate these concepts with three example proofs:

Example 1 If $T(n) \leq 2T(n/2) + cn$ for some constant c ,
then $T(n) = O(n \log n)$.

Example 2 If $T(n) \leq T(n/3) + cn$ for some constant c ,
then $T(n) = O(n)$.

Examples 1 & 2 will illustrate how to derive the induction
statement $p(n)$ while proving $p(n)$

Example 3 If $T(n) \leq 4T(n/2) + cn$ for some constant c ,
then $T(n) = O(n^2)$.

Example 3 will illustrate what is meant by using a stronger
induction hypothesis.

Example 1:

if $T(n) \leq 2T(n/2) + cn$ for some constant c ,
then $T(n) = O(n \log n)$.

From definition of big O we need to show that

$\exists n_0, k$ such that $\forall n > n_0, T(n) \leq kn \log n$

As before we will assume that n is a power of 2

A naive induction proof would assume that

(*) $T(n) \leq kn \log n$ was true for $n = 2^{i-1}$
and then prove that (*) was also true for $n = 2^i$

Our problem is that we do not know what k is
so we can't prove (*)

We want to prove that if, for all $n = 2^i$,
 $T(n) \leq 2T(n/2) + cn$ for some constant c ,
 $\Rightarrow \forall n > n_0, T(n) \leq kn \log n \quad (*)$

Our proof will be by induction, but with a twist.

We will assume that we have a k for which $(*)$ holds in the inductive hypothesis and then continue on to prove the inductive step.

$$\left((*) \text{ True for } n = 2^{i-1} \right) \Rightarrow \left((*) \text{ True for } n = 2^i \right)$$

While we are doing this, we will discover sufficient assumptions on k (and n_0) to ensure that k exists.

We want to prove that if, for all $n = 2^i$,

$$T(n) \leq 2T(n/2) + cn \text{ for some constant } c,$$

$$\Rightarrow \exists n_0, \exists k \text{ s.t. } \forall n > n_0, T(n) \leq kn \log n$$

Assumptions

1) $T(n) \leq kn \log n$ does **not** hold for $n = 1$, because $\log 1 = 0$.

$$n_0 \geq 1$$

2) Want $T(n) \leq kn \log n$ to be true for $n = 2$. Requiring $k \geq T(2)/2$ guarantees this, since it gives

$$k \geq T(2)/2$$

$$T(2) \leq k \cdot 2 \log 2 = k \cdot 2.$$

Our inductive hypothesis:

if $m = 2^j$ with $1 \leq j < i$ then $T(m) \leq km \log m$.

Now suppose $n = 2^i$.

By the i.h. $T(n/2) \leq k(n/2) \log n/2$, so

$$\begin{aligned} T(n) &\leq 2T(n/2) + cn \\ &\leq 2k(n/2) \log(n/2) + cn \\ &= kn \log(n/2) + cn \\ &= kn \log n - kn \log 2 + cn \\ &= kn \log n - kn + cn. \end{aligned}$$

In order to guarantee

$$T(n) \leq kn \log n$$

we must have

$$-kn + cn \leq 0.$$

We therefore make the
final assumption:

$$k \geq c.$$

We have just shown that if, for all $n = 2^i$,

$$T(n) \leq 2T(n/2) + cn \text{ for some constant } c,$$

and (assumption 1) $n > n_0 = 1$ then

(i) If $n = 2$ then $T(n) \leq kn \log n$

as long as (assumption 2) $k \geq T(2)/2$

(ii) If $T(m) \leq km \log m$ for $m = 2^j$ with $1 \leq j < i$ then

$$T(n) \leq kn \log n \text{ for } n = 2^i$$

as long as (assumption 3) $k \geq c$

We therefore conclude, *by the principle of mathematical induction*,
that as long as all three of our assumptions are satisfied,

$$\forall n > n_0 = 1 \quad T(n) \leq kn \log n.$$

We have therefore **proved** that $T(n) = O(n \log n)$.

Our inductive hypothesis:

if $m = 2^j$ with $1 \leq j < i$ then $T(m) \leq km \log m$.

- Note that the inductive hypothesis (and associated inductive step) was **not** fully defined when we started the proof, since we didn't say what the value of k was.
- It was only while in the middle of the proof that we specified the value of k (by discovering the conditions on k that would allow the inductive step to work)
- After the fact, it is possible to write a more traditional inductive proof, in which the value of k is given, but this can be even more confusing.

A More Traditional Induction Proof

We want to prove that if, for all $n = 2^i$,

$$T(n) \leq 2T(n/2) + cn \text{ for some constant } c,$$

$$\Rightarrow \forall n > 1, \quad T(n) \leq kn \log n$$

$$\text{where } k \geq \max \{c, T(2)/2\}$$

(i) Since $\log 2 = 1$, $T(2) = \frac{T(2)}{2} 2 \leq k 2 \log 2$

(ii) Let $n = 2^i$. Suppose $T(m) \leq km \log m$ for all $m = 2^j$ with $1 \leq j < i$.

$$T(n) \leq 2T(n/2) + cn$$

$$\leq 2k(n/2) \log(n/2) + cn$$

$$= kn \log n - kn \log 2 + cn$$

$$= kn \log n - kn + cn.$$

$$\leq kn \log n$$

And we are done!

Two things to note about “Traditional” proof:

1) Choice of k seems very arbitrary.

Why did we define $k = \max \{c, T(2)/2\}$?

2) Implicit choice of $n_0 = 1$ in big O statement also seems arbitrary.

Because the discussion in the first proof explained **why** we were making the choices we did, many people prefer the structure of the first proof to that of the second.

This type of inductive proof – in which conditions on the parameters are developed **during** the proof – is therefore used quite often in books and articles.

Example 2: We now prove by induction that, for T defined on $n = 3^i$, $i = 0, 1, 2, \dots$

if $T(n) \leq T(n/3) + cn$ for some constant c ,
then $T(n) = O(n)$.

From definition of big O we need to show that

$\exists n_0, k$ such that $\forall n > n_0, T(n) \leq kn$

As before, we will start with k undefined, and then derive assumptions under which the inductive proof will work.

Let $n_0 = 0$. In order for the inequality $T(n) \leq kn$ to hold when $n = 1$ our first assumption must be $k \geq T(1)$

Assume *inductively* that for $m = 3^j$, $0 \leq j < i$,

$$T(m) \leq km$$

Then, for $n = 3^i$,

$$\begin{aligned} T(n) &\leq T(n/3) + cn \\ &\leq k(n/3) + cn \\ &= kn + (c - 2k/3)n. \end{aligned}$$

So, if $c - 2k/3 \leq 0$, that is, if we assume $k \geq 3c/2$,
we conclude that $T(n) \leq kn$

Thus, if we choose $k \geq \max\{3c/2, T(1)\}$
we prove by mathematical induction that

if $T(n) \leq T(n/3) + cn$ for some constant c ,
then $T(n) = O(n)$.

The Corresponding “Traditional” Proof

We want to prove that if, for all $n = 3^i$,

$$T(n) \leq T(n/3) + cn \text{ for some constant } c,$$

$$\Rightarrow \forall n > 0, \quad T(n) \leq kn$$

$$\text{where } k = \max \{3c/2, T(1)\}$$

(i) When $n = 1$, $T(1) \leq kn$ by definition

(ii) Let $n = 3^i$. Suppose $T(m) \leq km$ for all $m = 3^j$ with $0 \leq j < i$.

$$T(n) \leq T(n/3) + cn$$

$$\leq k(n/3) + cn$$

$$= kn + (c - 2k/3)n.$$

$$\leq kn$$

And we are done!

Example 3: We now prove by induction that, for T defined on $n = 2^i$, $i = 0, 1, 2, \dots$

if $T(n) \leq 4T(n/2) + cn$ for some constant c ,
then $T(n) = O(n^2)$.

From definition of big O we need to show that

$\exists n_0, k$ such that $\forall n > n_0, T(n) \leq kn^2$

As before, we will start with k undefined, and then derive assumptions under which the inductive proof will work.

Let $n_0 = 0$. In order for the inequality $T(n) \leq kn^2$ to hold when $n = 1$ our first assumption must be $k \geq T(1)$

Assume *inductively* that
for $m = 2^j$, $0 \leq j < i$,
 $T(m) \leq km^2$

Then

$$\begin{aligned} T(n) &\leq 4T(n/2) + cn \\ &\leq 4(k(n/2)^2) + cn \\ &= 4\left(\frac{kn^2}{4}\right) + cn \\ &= kn^2 + cn. \end{aligned}$$

To proceed, would like to choose a k so that $cn \leq 0$.

Problem: **Impossible**. Both c and n are always positive!

What went wrong?

Statement is **too weak** to be proved by induction.

To fix this, let's see if we can prove something that is actually *stronger* than we were originally trying to prove — namely,
 $T(n) \leq k_1 n^2 - k_2 n$ for some positive constants k_1 and k_2 .

We get $T(n) \leq 4T(n/2) + cn$

$$\begin{aligned} &\leq 4(k_1(n/2)^2 - k_2(n/2)) + cn \\ &= 4(k_1n^2/4 - k_2(n/2)) + cn \\ &= k_1n^2 - 2k_2n + cn \\ &= k_1n^2 - k_2n + (c - k_2)n. \end{aligned}$$

To ensure that last line is $\leq k_1n^2 - k_2n$,
 suffices to have $(c - k_2)n \leq 0$

So assume $k_2 = c$.

Once we choose $k_2 = c$, we can then choose k_1 large enough to
 ensure correctness of base case $T(1) \leq k_1 \cdot 1^2 - k_2 \cdot 1 = k_1 - k_2$.

assume $k_1 = T(1) + c$

With these 2 assumptions we have proved *inductively* that
 $T(n) \leq k_1n^2 - k_2n$ so $T(n) = O(n^2)$.

Why was it easier to prove

stronger statement $T(n) \leq k_1 n^2 - k_2 n$

than to prove

weaker statement $T(n) \leq kn^2$?

- Proving something about $p(n)$ uses $p(1) \wedge \dots \wedge p(n-1)$.
- The stronger that $p(1) \wedge \dots \wedge p(n-1)$ are,
the greater help they provide in proving $p(n)$.
- Our problem was that $p(1), \dots, p(n-1)$ were **too weak**,
and thus we were not able to use them to prove $p(n)$.
- By using **stronger** $p(1), \dots, p(n-1)$, we were able to
prove a **stronger** $p(n)$, one that implied the original $p(n)$
we wanted.
- When we give an induction proof in this way, we are
using a **stronger inductive hypothesis**.