Polynomial -time Reduction

We formalise reduction by defining p-time reduction in terms of Turing Machines.

Fast non-deterministic solutions to old yes/no problems

Definition of p-time reduction ‘ ≤ ’

Let A, B be any two yes/no problems

X a deterministic Turing Machine

X reduces A to B if: for every yes-instance w of A, \( f_X(w) \) is defined and is a yes-instance of B

for every no-instance w of A, \( f_X(w) \) is defined and is a no-instance of B

A reduces to B in p-time if \[ \exists \] a det TM X running in p-time that reduces A to B ( \( A \leq B \) if A reduces to B in polynomial time).

If \( A \leq B \) and \( B \leq A \) \( \square \) A \( \sim \) B.
If $A \leq B$ and we have a solution to $B$ we can build a solution to $A$:

Example: we can reduce HCP to TSP in p-time, so $HCP \leq TSP$.

This does not mean that either HCP or TSP can be solved in p-time in the general case: but it does mean that a fast (p-time) solution to TSP would facilitate a fast solution to HCP too.

TSP and HCP are in NP-complete:
 each NP-complete problem reduces to all the others:
 - a p-time solution to any NP-complete problem would give a p-time solution to all the others (so putting them all into P)
\( \leq \) is a pre-order

\( A \leq B \) \( \square \) we can use a fast solution to B to solve A fast..

\( \leq \) can be interpreted as “is no harder than”

\( \leq \) is a pre-order: a reflexive, transitive, binary relation.

To prove that \( \leq \) is a pre-order on the class of yes/no problems:

Show:

1. \( \leq \) is reflexive: ..show that for any yes/no problem \( A \), \( A \leq A \)
   …that there is a deterministic p-time TM which reduces \( A \) to \( A \).
2. \( \leq \) is transitive: ..show that if \( A \leq B \) and \( B \leq C \) then \( A \leq C \).
   …that there is a deterministic p-time TM which reduces \( A \) to \( C \).
1. \( \leq \) is reflexive: \( A \leq A \)

let \( I \) : alphabet used to write all instances of \( A \)
let \( X = (\{q0\}, I \sqcup \{^\}\), I, q0, \( \emptyset \), \{q0\}) 
be an ordinary deterministic TM

\( X \) Halts and Succeeds with no actions  
output = input (unchanged)

so if \( w \) is a yes-instance of \( A \)  
\( \square \ f_X(w) \) is a yes-instance of \( A \)  
if \( w \) is a no-instance of \( A \)  
\( \square \ f_X(w) \) is a no-instance of \( A \).

\( \square \ X \) reduces \( A \) to \( A \).

\( \text{time}_X(n) = 0. \)  
So \( \leq \) is reflexive
2. \(\leq\) is transitive
suppose \(A \leq B\) and \(B \leq C\),
\(A, B, C\) all yes-no problems
Prove \(A \leq C\).

\(X*Y\) is a deterministic TM which reduces \(A\) to \(C\)
Running time of $X*Y$ reducing $A$ to $C$:

$X$ reduces $A$ to $B$ in $p$-time…suppose $\text{time}_X(n) \leq p(n)$

$Y$ reduces $B$ to $C$ in $p$-time….suppose $\text{time}_Y(n) \leq q(n)$

where $p$ and $q$ are polynomials in $n$

The running time of $X*Y$ on input $w$ of length $n$:

1. Mark sq. 0 1
2. Run $X$ on $w$ $\leq p(n)$
3. Return to sq. 0 and unmark it $\leq p(n)$
4. Run $Y$ on $f_X(w) \leq q(p(n))$

Total run time $\leq 1 + p(n) + p(n) + q(p(n))$

..a polynomial in $n$.

So $X*Y$ reduces $A$ to $C$ in $p$-time $\leq$ is transitive
NP is closed under p-time reduction

if $A \leq B$ we can use a fast solution to $B$ to solve $A$ fast. If $B$ is solved a by a p-time NDTM, that is $B \in NP$.. then is $A \in NP$?

Prove: if $A \leq B$ and $B \in NP$ then $A \in NP$.

$A \leq B$ if a det. TM, $X$ which reduces $A$ to $B$.
$B \in NP$ if a NDTM, $N$ which solves $B$.

We construct a new TM, $X*N$ which solves $A$.
we need to show that:

$X*N$ is non-deterministic and
$X*N$ runs in p-time $A \in NP$. 
suppose $A \leq B$ and $B \in NP$, prove $A \in NP$.

$X^N$ is a non-deterministic TM which solves $A$.
running time of $X^N \leq 1+p(n) + p(n)+q(p(n))$..a polynomial

$A \in NP$

$X^N$ accepts a yes-instance of $A$
$X^N$ rejects a no-instance of $A$

Computability and Complexity

Lecture 18
P is closed downwards under reduction

if $A \leq B$ and $B \in P$ then $A \in P$.

time function of $X^*M$: $\text{time}_{X^*M}(n) \leq 1 + p(n) + p(n) + q(p(n))$

A polynomial

$X^*M$ is a deterministic p-time TM which solves $A$

**Diagram:**

- **X**
  - Mark sq. 0
  - Time function: $p(n)$
  - Return to sq. 0
  - Output of X on $w$ is an instance of $B$

- **M**
  - Time function: $q(n)$
  - Deterministic TM solving yes/no problem $B$ in p-time

- **X^*M**
  - Deterministic TM reducing $A$ to $B$ in p-time
  - Input word: instance of $A$
  - $w$
The P-problems are the \( \leq \)-easiest.
(The problems in P are the easiest with respect to our difficulty ordering)

Prove that if A is any problem in P and B is any yes/no problem \( A \leq B \)
let M be a det. p-time TM which solves A: modify H&S and H&F to give M'

- **Input w:** an instance of A
- **M' solves A in p-time**
- **w1:** a yes instance of B
- **w2:** a no-instance of B

\[ X \]

1. **Mark sq 0**
2. **M'**
3. If M accepts w:
   - return to sq 0
   - output w1
4. If M rejects w:
   - return to sq 0
   - output w2
Summary
We have defined:

**polynomial-time reduction**: if A, B are yes/no problems:

A reduces to B in p-time if there exists a deterministic TM X running in p-time that reduces A to B (A ≤ B if A reduces to B in polynomial time).

Properties of ≤: ≤ is a pre-order.

..a reflexive, transitive, binary relation

..permitting an ordering in difficulty or complexity of problems.

**Properties of P and NP with respect to reduction:**

P is closed downwards under ≤: for yes/no problems A, B

if A ≤ B and B ∈ P ..then A ∈ P

NP is closed downwards under ≤: for yes/no problems A, B

if A ≤ B and B ∈ NP ..then A ∈ NP

The class P is the ≤-easiest class of yes/no problems.