Ch. 7: NP & Cook-Levin Theorem & Reduction & Fine Print

Idea:

Many problems are of the following type:

(1) after ≥45 years, there is no algorithm to solve them in poly time

(2) there is a natural search, through exponentially many "possibilities" but any "possibility" is quick to check

e.g.

formula:

$f = x_1$ and $\neg x_2$ and $(x_1$ or $x_2)$

involves $x_1, \ldots, x_n$, want to know if f can be satisfied.

(1) don't know how to decide SAT in poly time

(2) for each possible $x_1 =$ T or F, $x_2 =$ T or F, ... $x_n =$ T or F (2^n possibilities) we can check if f is true or not.

$\Rightarrow$ if then $f = f(x_1, \ldots, x_n)$, $\langle f \rangle$ (longer than n symbols)

(1) we don't how to solve 3COLOR quickly (in P)

(2) there is a natural exponential search through a bunch of "possibilities" given a possible colouring it's fast to check if it works (i.e., is a proper colouring)

$\Rightarrow$

SUBSET-SUM:

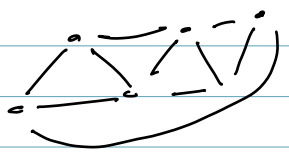
given $n_1, \ldots, n_m; S$ integers. Is there $I \subseteq \{1, \ldots, m\}$ s.t. $\sum_{i \in I} n_i = S$

f as string:

x1 and not x2 and (x1 or x2)

length of $\langle f \rangle$ here is 14 symbols

3COLOR:



$G = (V, E)$

We want to know: if we guess colouring $c: V \to \{1, 2, 3\}$ we can check if c is a proper colouring

SAT, 3COLOR, SUBSET-SUM, many others:

(1) we haven't yet, in 45 years, found a poly-time algorithm to decide

(2) there's a natural algorithm if you are allowed to "guess" from an exponentially large set of possible solutions.

NP, Cook-Levin Theorem show: if 3COLOR has poly time algorithm then so does SAT, SUBSET-SUM ...

e.g.

1, 3, 10, 16; 21

there is no ~~subsequence~~ way to choose numbers from the list 1, 3, 10, 16 to get a sum of 21

1, 3, 10, 16; 21 $\notin$ SUBSET-SUM

But 1, 2, 3, 4, 5, 6; 19

we have $1+3+4+5+6 = 19$

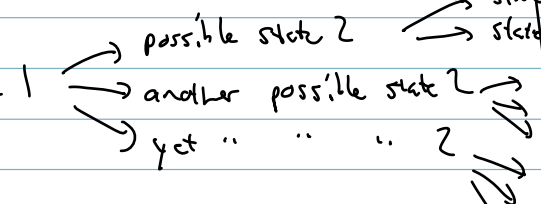
so $1, 2, 3, 4, 5, 6; 19 \in$ SUBSET-SUM

non-determinitic:

given (q, γ)

$\delta(q,\gamma)$ is not a single transition to some $(\tilde{q}, \tilde{\gamma}$, L or R)

rather $\delta(q,\gamma)$ we have a number of possible transitions



=

Formally:

$\delta: Q \times \Gamma \to$ same subset of all of $Q \times \Gamma \times \{L, R\}$

formalize equivalence "up to poly time" of SAT, 3COLOR, SUBSET-SUM, and many others

=

(SAT, 3COLOR, SUBSET-SUM, INDEPENDENT-SET, ... $\in$ NP

NP = non-deterministic poly time

helpful but not essential

non-determinism:

state 1 → state 2 → ...

in a Turing machine

3COLOR:

Either

$V_1 \leftarrow$ red, $V_2 \leftarrow$ red, ..., $V_n \leftarrow$ red is a proper colouring

OR

$V_1 \leftarrow$ blue, $V_2 \leftarrow$ red - -

OR

$V_1 \leftarrow$ green, $V_2 \leftarrow$ red, - -

OR

.

.

.

exponential # of cases

OR $\leftrightarrow$ non-determinism

Remark:

Search [document] Joel or Friedman followed by terrific or wonderful

(JOEL V FRIEDMAN) ~~test~~

(TERR V WONDERFUL)

Non-determinism $\swarrow$ OR

$J^3 J^* (JA \text{ or } RR)$

- - - J J J | JA
RR
JJA
JRR

Non-determinism with OR in regular languages

Formally:

$\delta(q, \gamma) = (q_1, \gamma_1, L)$ OR (q_2, γ_2, R) OR .

.

.

i.e.

$\delta : Q \times \Gamma \to$ Power$(Q \times \Gamma \times \{L, R\})$

we say non-det. T.m., M, accepts w if there is some possible state 1 $\to$ state 2 $\to$ ... $\to$ reaches q_{acc}

i.e. set

$V_1 \leftarrow$ red OR blue OR green

and

$V_2 \leftarrow$ .. ..

.

.

Paradoxes $\leftrightarrow$ - self-reference - negate NOT

=

Non-determinism $\leftrightarrow$ check many cases via OR

=

① Make Turing machines have -OR -non-det

If $\exists$ some computation that reaches q_{acc},

"M accepts w"

"Decide in time $3n^5$"

step 1

step $3n^5$

could be $2^{(3n^5)}$ computation poss. bilities