CSL 3010: Design and Analysis of Algorithms July-November 2022

Lecture 7: Maximum Independent Set

Lecturer: Dr. Saket Saurabh *Scribe: Mukul, Sawan, Jaimin, Tanay*

1 Summary

1.1 Non-Trivial Algorithms:-

- Max/Min
- Multiplication of two-digit algorithm
- Colouring
- SAT/3-SAT

This allowed us do :-
Reduction as a methodology to design algorithms as well as "*proofs*" that same problem could be hard.

Reduction in designing algorithms $->$
Recurssion
Tower of Hanoi
mergesort
Quick Sort
Quick Select

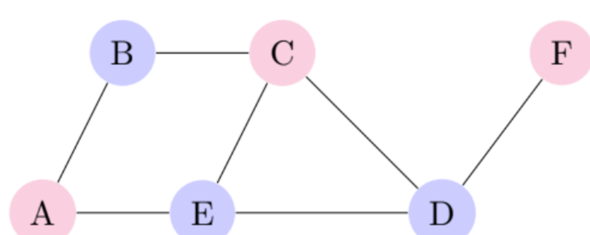
Now looking at the

"*Backtracking*" $-$ Refinement of brute-force search. Build solution increameantally by invoking recurssion to try all possiblities for the decision in each step.

2 Maximum Independent Set

Definition : A set S $\subseteq$ V(G) is an independent set if there are no edges between vertices in S.

$$\forall u, v \in S, uv \notin E.$$



Input: $G(V, E)$ **Output:** Find maximum size of independent set in G.

Algorithm 1: maxIndSet (G = (V, E))

---

max $\leftarrow$ 0
  for each subset $S \subseteq V$ do
    check is S ia an independent set
    if S is independent and $|S| > max$, then
      max = s
Output max

Runtime: $O(2^n n^{(O(1))})$

Statement: If MIS has $n^{O(1)}$ time algorithm then, so does 3-SAT ot SAT.

$$3 - SAT \to to \to MIS$$
$$\phi = (\neg x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee x_2 \vee x_3)$$

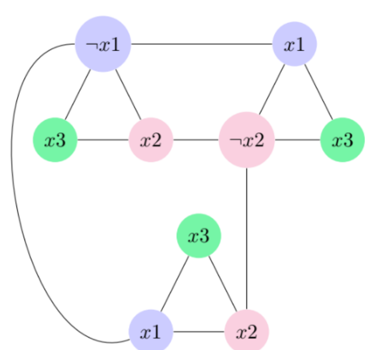
Interprating SAT:

- Way to assign 0/1 to yhe variable such that formula is satisfiable (in cavh class evaluates to true)
- Pick a literal from each class and find a truth assignment to make each of them true.

$\rightarrow$You will fail if two of the literal you pick are in conflict.

Second view of the reduction.

$$\phi = (\neg x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee x_2 \vee x_3)$$



1. G_ϕ has vertex foe each literal in clause.
2. Connect the three literals in the clause to form a triangle: the independent set will pick a at most one vertex from each clause which will corresponf ti the literal to be set of true.
3. Connect two vertices it thus label complementry literals: this ensures that the literals corresponding to independent set do not have a conflict.

Lemma:- ϕ is satisfiable $iff G_\phi$ has max independent set of size m.
Where m = number of clause

By construction, |MIS in G_ϕ| $\leq$ m

ϕ is satisfiable $\implies$
Let a be the truth assignment satisfy ϕ. Pick one of the vertices corresponding to true literals, under a from each triangle.
(this is an independent set)

Trying in reverse
$\Leftarrow$
Let S be an independent set of size m.

1. S must contain one vertex from each clause.
2. S can't contain vertices labelled by conflicting literals.
3. Thus, it is possible to obtain a truth assignment that makes the literals in S true; such an assignment satifies one literal in every clause.

for every variable $\neg x_i$ we have picked up $\rightarrow$ we choose either x_i or $\neg x_i$
So, if we have selected x_i then we make $x_i = 1$.
And, if we have selected $\neg x_i$ then we make $\neg x_i = 1$

insert figure

Theorem: Unless SAT has polynomial time, MIS dosen't hava polynomial time.

For a set S and vertex v

1. either $V \in S \rightarrow$ N(V) $\nsubseteq$ S
2. or may be V $\notin$ S
3. **insert fig**

Here we can try to reduce the size of the problem in both the cases.

1. V $\in$ S $\Rightarrow G - V - N(V)$
2. V $\notin$ S $\Rightarrow G - V$

Algorithm 2: RecursiveMIS(G = (V, E))

If G is empty return 0
    a $\leftarrow$ RecursiveMIS(G - V)
    b $\leftarrow$ RecursiveMIS(G - V - N(V))
Output max(a, b)

Runtime:

$$T(n) = T(n-1) + T(n-1-deg(V)) + O(n^2)$$
$$T(0) = T(1) = 1$$
$$deg(V) = 0 \rightarrow worstcase$$
$$WorstCase \rightarrow T(n) = 2T(n-1)$$
$$O(n^2)$$

Algorithm 3: RecursiveMIS(G = (V, E))

If $\Delta G = 0$, then return 0
Let V be a vertex of maximum degree.
a ← RecursiveMIS(G - V)
b ← RecursiveMIS(G - V - N(V))
Output max(a, b)

Runtime:

$$T(n) = T(n-1) + T(n-2)$$
$$T(0) = T(1) = 1$$
$$1.618^n$$

Correctness:
By Induction:

$$x^n$$
$$x^{n-1} + x^{n-2} \leqslant x^n$$
$$\Rightarrow x^1 + 1 \leqslant x^2$$

⟶ Can we do better?…

If $\Delta(a) \leqslant 1$
insert dot fig

$$MIS = V - E \; T(n) \leqslant T(n-1) + T(n-3) \; X^n \leqslant X^{n-1} + X^{n-3} = x^2 + 1 \quad 1.46^n$$

Algorithm 4: $N^2 RecursiveMIS$(G = (V, E))

If $\Delta G \leqslant 1$, then return $|V(G)| - |E(G)|$
V = maximum degree
a ← $N^2 RecursiveMIS(G - V)$
b ← $N^2 RecursiveMIS(G - V - N(V))$
Output max(a, b)

Runtime:

$$T(n) = T(n-1) + T(n-3) + O(n^2)$$
$$T(n) = O(n^2 * (1.46)^n)$$

$\Delta G \leqslant 2$

Algorithm 5: $N^3 RecursiveMIS$(G = (V, E))

If $\Delta G \leqslant 2$
Problem solvable in polynomial time **return** ans after solving
$LetV = maximumdegree$
$a \leftarrow N^3 RecursiveMIS(G - V)$
b $\leftarrow N^3 RecursiveMIS(G - V - N(V))$
Output max(a, b)

$$T(n) = T(n-1) + T(n-1-deg(V)) + O(n^2)$$

As the $deg(V) \geq 3$ outside the if block, this implies that T(n - 1 - deg(V)) = T(n - 1 - 3) **in the worst case**

$$T(n) = T(n-1) + T(n-4) + O(n^2)\ T(n)\ (n^2 * (1.38)^n);$$

Exercise: Show that MIS on graph with $|G| \leqslant 3$ is polynomial time.
As evident from the above reductions, it is clear that unless SAT has polynomial time MIS on graphs having maximum degree 3 does not have polynomial time algorithm.

3-SAT

Input: A 3-SAT φ **Output:** Is φ satisfiable?

1. Let C = $x_1 \cup \neg x_2 \cup x_3$
2. If φ satisfiable
   (a) Either x_1 is set to 1. Number of litersals set for assignment is n - 1.
   (b) Or $\neg x_2$ is set to 1. Number of litersals set for assignment is n - 1.
   (c) Or x_3 set to 1. Number of litersals set for assignment is n - 1.
3. Remove all clauses that are satisfied with the above partial assignment.
4. For each clause that are left, drop all the literals that are already set by partial assignment.

$$T(n) \leqslant T(n-1) + T(n-2) + T(n-3)$$
$$T(n) = O(1.83^n)$$

Thus, 3-SAT not like SAT.