CSC791/495

Dr. Blair D. Sullivan

October 27, 2017

Today

1. Follow up on last week's HW (Kyle)

2. Lower bounds (ETH)

3. Logistics:
   - project setup assignment
   - weekly log requirement
   - in-class teamwork expectations

4. Research!

* New HW assignment

SAT Flashback

- SAT: given a boolean formula φ w/ n variables, m clauses
  question: is there a satisfying assignment ($\text{var} \to T / F$)
- CNF-SAT: conjunctive normal form: clauses are ORs; and they are ANDed together.
- q-SAT: CNF-SAT w/ $\leq q$ literals per clause.

—— All these are NP-complete. ——

How fast can we solve 3SAT? 2^n? $2^{\sqrt{n}}$? $2^{n \log n}$? or more exotic time?

One algorithm for 3SAT: brute force $O^*(2^n)$

Try branching. Use clauses.

c unsatisfied $\Rightarrow$ how many variable assignments can make it true? ≤ 7 (of 8)

≤ 7

3SAT

$O^*(7^{\#\text{branch points}})$

$\leq n/3$ b/c each branch point fixes assigments for 3 variables.

$O^*(7^{n/3}) \sim O^*((1.91)^n)$. How much better can we do? current <u>best</u>: $\approx (1.31)^n$

Hypothesis: must depend linearly on n in exponent.

Hypothesizing

$$\delta_q = \inf\{c \mid \exists\, O^*(2^{cn}) \text{ algorithm for } q\text{-SAT}\}$$

Exponential Time Hypothesis (ETH): $\delta_3 > 0$

$\exists c > 0$ s.t. 3SAT cannot be solved in $O^*(2^{cn})$

[there is no $2^{o(n)}$-algorithm for 3SAT]

Strong Exponential Time Hypothesis (SETH): $\lim_{q \to \infty} \delta_q = 1$

- SETH $\Rightarrow$ ETH (non-trivial)
- less people believe SETH.

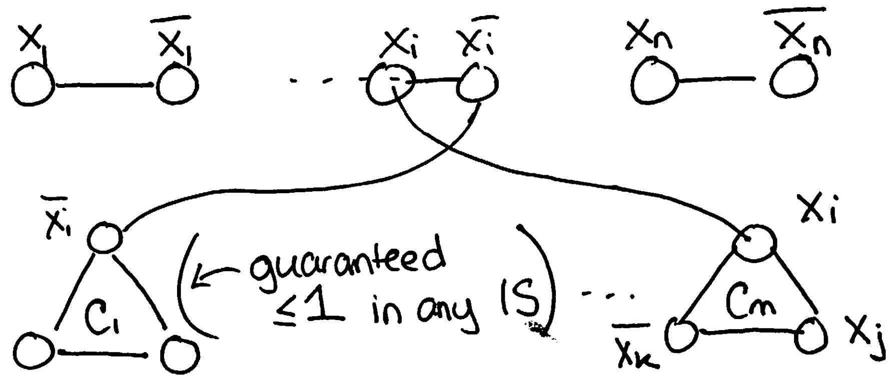
Transfer via Reductions, take 1

Vertex Cover: naive alg is $O^*(2^n)$. Is this "essentially best possible"?

Karp reduction from 3SAT. Given formula φ w/ n vars & m clauses.
Construct G s.t. G has a VC of size $f(n,m) \iff \varphi$ is satisfiable.

variables:

clauses:



φ SAT $\Rightarrow$ VC of $n+2m$

pick the satisfying assignment in the variables row
If a clause is unsatisfied $\Rightarrow$ must add all 3 vertices to our cover
(b/c of edges between vars & clauses)

$N = 2n + 3m = O(n^3)$

$O^*(2^{o(N^{1/3})})$-alg for VC $\Rightarrow$ $O^*(2^{o(n)})$-alg for 3SAT.
↑ doesn't exist by ETH.

Problem: we needed # clauses to be smaller for this to be interesting.

Sparsification

"$\exists$ subexponential-time algorithm that reduces # clauses in a q--SAT formula to $O(n)$"

<u>Thm</u> Assuming ETH, $\exists\, c>0$ s.t no algorithm solving 3SAT in $O^*(2^{c(n+m)})$.

<u>Corollary</u> no $O^*(2^{o(n)})$-alg for VC.

b/c now $N = 2n + 3m = O(n)$ (φ is sparse).

Other problems this works for: Dominating Set, Independent Set, Hamilton Cycle } standard NP-reductions give tight bounds.

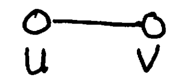
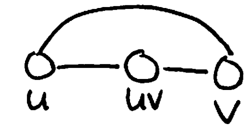
What did we need for our reduction to "work"?

linear size in #variables.

More Examples

bonus material

Dominating Set: Assuming ETH, no $O^*(2^{o(n)})$-algorithm.

Reduction: same as Vertex Cover, but replace  w/ 

Independent Set

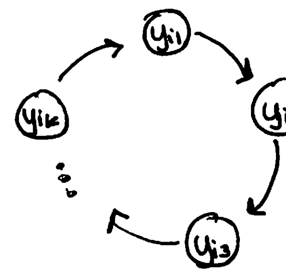
we'll need this so that $|E| \approx O(|V|)$ (ie. sparse!)

Idea Reduce from 3SAT-3 (every variable in ≤ 3 clauses).

First Show reduction from 3SAT to 3SAT-3: Let φ be a 3SAT formula w/ variables $x_1, \ldots, x_n$ and clauses $C_1, \ldots, C_m$. We will construct a formula ψ as follows:

- create variables y_{ij} for all pairs i,j so x_i is in clause C_j.
- now, "link" all y_{ij} for each i:

  say $j_1, \ldots, j_{\ell_i}$



$y_{ij_k} \Rightarrow y_{ij_{k+1}}$ $\quad k = 1, \ldots, \ell_i$

$\underbrace{(\neg y_{ij_k} \vee y_{ij_{k+1}})}$

- take ψ to be old clauses (but with x_i in C_j replaced by y_{ij}) ANDed w/ new ones

new clauses ← each y_{ij_k} is in 2 of these.

$\Rightarrow$ each y_{ij} can only appear in ≤ 3 clauses of ψ and all clauses have length 2 (new) or 3 (old).

* ψ is satisfiable $\Longleftrightarrow$ φ is. $\boxed{\text{Check}}$ size of new instance

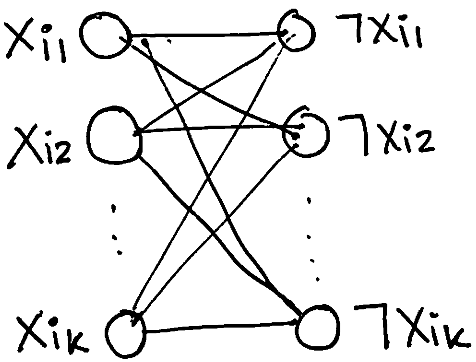
Ind Set, cont

bonus material

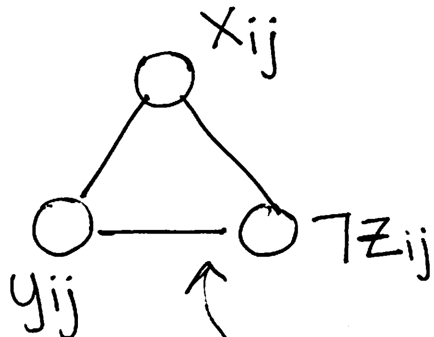
So now, we know 3SAT-3 has no $O^*(2^{o(n)})$-algorithm under ETH.

We need to design a reduction from 3SAT-3 to Independent Set with linear size blowup. Let's try the classic NP-hardness approach:

we create a component of our graph for each variable & clause.

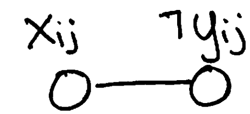


variable x_i occurring in k clauses



clause w/ 3 literals

$x_{ij} \vee y_{ij} \vee \neg z_{ij}$



clause w/ 2 literals

$x_{ij} \vee \neg y_{ij}$

Now, the fact that 3SAT-3 limits the $K_{k,k}$'s to $K_{3,3}$'s $\Rightarrow$ linear # edges! We have a satisfiable formula $\Leftrightarrow$ IS of size $3n+m$.

FPT lower bounds

① prove lower bounds on parameterized problems of the form

$O^*(2^{o(k)})$ using parameterized reductions (from 3SAT)

need to measure how k depends on n

look at k-VC:

$k = n + 2m = O(n) \Rightarrow$ no $O^*(2^{o(k)})$-alg. assuming ETH.

② ETH $\Rightarrow$ W[1] $\neq$ FPT

Show ETH $\Rightarrow$ no $f(k)\, n^{O(1)}$-alg for k-CLIQUE.

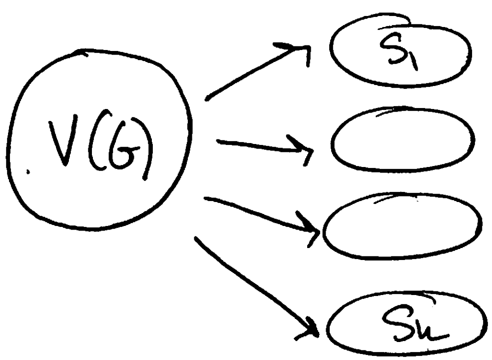
Proof sketch via reduction from 3-COLORING.

$O^*(2^{o(n)})$-lower bound under ETH

ETH and W[1]-hardness

Reduce from 3-COLORING to CLIQUE.

Given a graph G, want to construct a graph H s.t. G is 3-colorable $\iff$ H has a k-clique.



(split into) k subsets (of size n/k)

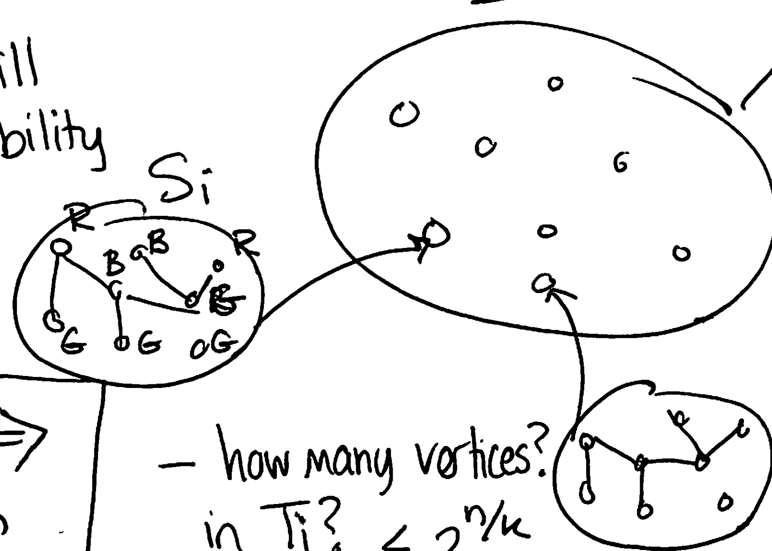
Create H:

$V(H)$ will be made of k groups of vertices

3-colorings s.t. agree on S_1 & S_2.

k

edges in H will indicate compatibility of 3-colorings.



each group will correspond to an S_i and have a vertex for every valid 3-coloring of $G[S_i]$

if max clique in H has size $k \iff$ 3-color all of G

— how many vertices in T_i? $\leq 3^{n/k}$

What are cliques in H? ≤ 1 vertex per T_i. ($\leq k$).

W[1] - continued

Suppose k-CLIQUE had an FPT algorithm $f(k)\, n^{k/s(k)}$ ← monotone, increasing unbounded

want to show a $2^{o(n)}$-alg for 3-COLORING.

let k be as large as possible s.t. $\underline{f(k) \leq n}$ and $k/s(k) \leq n$.

⇒ Then k is an unbounded function of n.

running time: (k-CLIQUE on H) $f(k)\left(k \cdot 3^{n/k}\right)^{k/s(k)} \leq n \cdot \underbrace{k^{k/s(k)}} \cdot 3^{n/s(k)}$

$|V(H)| \leq k \cdot 3^{n/k}$

$\leq \underline{n^2}\, 3^{n/s(k)} \longrightarrow O^*\left(2^{o(n)}\right)$

↑ polynomial

Contradiction to ETH.