Fine-Grained Complexity and Algorithms

10/8. Algorithms for NP-Complete Problems.

Question: How much can we save over exhaustive search?

- Generic Problems:

Generic Problems	K-SAT	3-color	Ind. set
n: # of variables	# of Boolean variables	$\lvert V\rvert$	$\lvert V\rvert$
Σ: set of possible values	$2: \{0,1\}$	$3: \{R,G,B\}$	$2: \{\text{in } S, \text{not in } S\}$
m: # of constraints	# of clauses	$\lvert E\rvert$	$\lvert E\rvert$
Exhaustive search: $\lvert\Sigma\rvert^n \cdot \text{poly}(m)$	$2^n \cdot m$	$3^{\lvert V\rvert} \cdot \lvert E\rvert$	$2^{\lvert V\rvert} \cdot \lvert E\rvert$

- Supexponential: $2^{o(n)} \cdot \text{poly}(m)$

  super supexponential: $2^{n^{o(1)}}$

  moderate improvement: $\lvert\Sigma\rvert^{(1-\alpha)n} \cdot \text{poly}(m)$ $\alpha > 0$

  non-trivial: $\lvert\Sigma\rvert^n \cdot \text{poly}(m) / n^{\omega(1)}$

- Most reductions:

  $(n, m) \to (n', m')$

  $n' = O(n+m)$, $m' = O(n+m)$

  Goal: make m small $\quad$ $O(n)$

  $n_1 = O(n)$ sparsification

- Schoning's Algorithm for K-SAT

  SchA: Guess an assignment $\vec{x}$ at random (Hope $\vec{x}$ is close to a satisfying assignment $\vec{y}$)

  While $\boxed{\exists \text{ an unsatisfied clause in } \vec{x}}$:

  pick such a clause: $l_1 \vee l_2 \vee \cdots l_k$

  pick i at random & flip that bit for l_i in x $\boxed{\text{\& then mark it fixed}}$

  If we get a satisfying assignment, done!

  If fixed variables force a clause to be false, quit.

  Repeat the procedure until finding a satisfying assignment.

Analysis Fix a particular satisfying assignment $\vec{y}$.

Define $d = \text{Ham}(\vec{x}, \vec{y}) := \#$ of i s.t. $x_i \neq y_i$.

$d_t := \text{Ham}(\vec{x}^t, \vec{y})$ Hamming distance after t steps.

In each step, we have prob $\geq 1/k$ of decreasing d_t: $d_{t+1} = d_t - 1$.

$\text{Prob}[\text{reach } \vec{y} \text{ or other satisfying assignment} \mid \text{picking } \vec{x}^0] \geq (1/k)^{d_0}$

$$\sum_{d=0}^{n} \binom{n}{d} \cdot 2^{-n} \cdot (1/k)^d = 2^{-n} \sum_{d=0}^{n} \binom{n}{d} k^{-d} = 2^{-n}(1+1/k)^n = 2^{-n}\left(\frac{k+1}{k}\right)^n$$

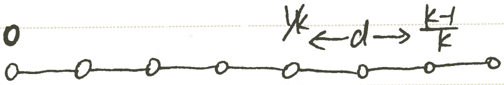
Expected time: $2^n\left(\frac{k}{k+1}\right)^n = 2^n\left(1-\frac{1}{k+1}\right)^n = 2^n \cdot 2^{-O(\frac{1}{k})n}$.

SchB: ① Remove "& then mark it fixed"

② Modify "while ($\exists$ an unsatisfied clause in $\vec{x}$, or until we repeat n steps)"

Analysis. In each step, with prob $1/k$ we decrease d_t by 1,

with prob $\frac{k-1}{k}$ we increase d_t by 1.



$$\binom{n}{d} \cdot 2^{-n} \max_{0 \leq i \leq \frac{n-d}{2}} \left\{ \binom{d+2i}{i} \left(\frac{1}{k}\right)^{d+i} \left(\frac{k-1}{k}\right)^i \right\}$$

$\Longrightarrow$ Expected time $2^n\left(1-\frac{1}{k}\right)^n$.

- **Ind. Set: Back Tracking** "enlightening case analysis".

[Tarjan & Trojanowski]

TT$(G, N_1, N_2, \cdots N_i, a_i)$:

If $V = \phi$, return ϕ

If $E = \phi$, return V.

pick $x \in V$

$S_1 = \{x\} \cup TT(G - \{x\} - N(x))$

$S_0 = TT(G - \{x\})$, note that we should pick at least 2 nodes from $N(x)$)

Return S_1 unless S_0 is larger, then return S_0.

Analysis: If pick x carefully $\rightarrow 2^{n/3}$.

Rubsun: more carefully $\rightarrow 2^{0.28n}$

[Johnson & Szegedy] Sparsification

TTS(G). If $V=\phi$, return ϕ.

If $E=\phi$, return V.

If G is d-sparse ($|E| \le d \cdot |V|$), return G.

pick $x \in V$ of deg $\ge 2d$

$\{G_1\} = \{x\} \cup TTS(G - \{x\} - N(x))$

$\{G_0\} = TTS(G - \{x\})$

Return $\{G_1\} \cup \{G_0\}$.

Analysis. $T(n) = \underset{L}{T(n-2d-1)} + \underset{R}{T(n-1)} \le \binom{n}{\frac{n}{2d+1}} \approx (2d+1)^{\frac{n}{2d+1}} = 2^{\frac{n\log(2d+1)}{2d+1}}$

$L+R=n;\ L \le \frac{n}{2d+1}$

Thm. $\forall \epsilon, \exists d$ s.t. we can reduce the general case of MIS to the d-sparse case in $2^{\epsilon \cdot d}$ time.

Cor. If $\exists \delta$, $T_{MIS}(n) \ge 2^{\delta n}$, then $\exists d, \delta'$ s.t. $T_{d\text{-sparse MIS}}(n) \ge 2^{\delta' n}$

- Sparsification Lemma for k-SAT [IPZ].

  $\forall k, \epsilon, \exists D = O((\frac{k}{\epsilon})^k)$ so that every k-CNF formula φ can be written as $\varphi = \bigvee_{i=1}^{2^{\epsilon n}} \varphi_i$ in time $2^{\epsilon n}$, where each φ_i is a k-CNF with same variables, $\le Dn$ clauses.