CS 598 TMC
Algorithms from the Fine-Grained Perspective

Courses.grainger.illinois.edu/cs598-tmc

Course Work:
- 4 HWs 45%
- presentation 15%
- project 40%
  (may work in groups of ≤ 3)

Prereq:
- undergrad algms (CS374)

Theme:
- understand complexity of
  basic algorithmic problems
  beyond polynomial vs. NP-hard
  e.g. $n^3$ vs. $n^2$ etc.

Ex1
All-Pairs Shortest Paths (APSP)
for dense weighted graphs

Floyd-Warshall (by DP) $O(n^3)$ time
Dijkstra $n$ times $O(n^3)$
better? Fredman '75 $\sim O\left(\frac{n^3}{\log^{1/3} n}\right)$
: C'07 $\sim O\left(\frac{n^3}{\log^2 n}\right)$
Williams '14 $O\left(\frac{n^3}{\sqrt{\log n}}\right)$
**Conjecture**

no truly subcubic algo\(n^{2.9999}\)

---

**Ex2**

*Longest Common Subsequence (LCS)*

of 2 strings \(a_1, \ldots, a_n\)
\(b_1, \ldots, b_n\)

\[
DP \Rightarrow O(n^2) \text{ time}
\]

\[
L(i,j) = \max \left\{ \begin{array}{ll}
L(i-1,j) & \text{if } a_i \neq b_j \\
L(i,j-1) + 1 & \text{if } a_i = b_j
\end{array} \right.
\]

better? current record \(\sim O\left(\frac{n^2}{\log n}\right)\)

related Problems: edit dist, Frechet dist, …

---

**Ex3**

3SUM

Given set \(S\) of \(n\) numbers \(\tau\) target \(\tau\),

\(\exists a, b, c \in S\) s.t. \(a + b + c = \tau\)?

- trivial: \(O(n^3)\)
- standard HW prob: \(O(n^2)\)

better? Grönland-Pettie ’14: \(\sim O\left(\frac{n^2}{\log n}\right)\)

C’18: \(\sim O\left(\frac{n^2}{\log^2 n}\right)\)

**Conjecture**

no truly subquad algo

---

k-SUM:

- trivial \(O(n^k)\)
- "meet-in-middle": \(O\left(n^{\frac{k+1}{2}} \log n\right)\) for \(k\) even
\(O\left(n^{(k+1)/2}\right)\) for \(k\) odd
better?

**Subset-Sam:** given $S$, & $t$, 
exists subset summing to $t$

- trivial $\sim O(2^n)$
- meet-in-middle: $\sim O(2^{n/2})$

better?

**DP:** $O(nt)$ time assuming positive integers

current record: $O((n+t) \log n)$

better?

**Ext**

closest pair in $d$ dims

- trivial: $O(dn^2)$ time
- known: $O(d^{O(d)} \log n)$

bad when $d \sim \log n$

(orthogonal vector)

---

Proving lower bds is very difficult in general comp. model

**Idea:** Prove conditional lower bds under conjectures that certain basic probs are hard via fine-grained reductions
are hard via fine-grained reductions

Similar to NP-hardness (\text{Conj } P \neq NP)

Exs of Cond. Lower Bd. Results:

1. Gajentaan-Overmars '93:
   if 3-point-collinearity could be solved in $O(n^{1.99})$ time,
   then 3SUM $\ldots \ldots \ldots \ldots O(n^{1.99})$ time

2. Abboud et al. '14:
   if graph radius could be computed in $O(n^{2.99})$ time,
   APSP $\cdots \cdots \cdots \cdots O(n^{2.9999})$ time

3. Patrasca '10:
   if triangle listing could be solved in $O(n^{4/3-\varepsilon})$ time,
   then 3SUM $\cdots \cdots \cdots \cdots O(n^{2-\varepsilon})$ time
   \[\text{for integers}\]

4.Bringmann '14:
   \[\text{edit dist, LCS}\]
   if Frechet dist could be solved in $O(n^{1.99})$ time,
   then SAT $\ldots \ldots \ldots \ldots O(1.99999^n)$ time
   \[\text{which contradicts}\]
   \[\text{Strong Exponential Time Hypothesis (SETH)}\]

\[\text{etc.}\]
Course Outline

I. Basic Algorithmic Tools (Upper Bds)
   Convolution/FFT, matrix mult.

II. Conditional Lower Bds using conjs on APSP/3SUM
    lots of reductions ...

III. Adv. Algorithmic Techniques (Back to UBs)