CS 598 TMC Algorithms from the Fine-Grained Perspective

http://courses.engr.illinois.edu/cs598tmc

Timothy Chan  (tmc@illinois.edu)
off hr on zoom: Thu 3p-4p
or by appointment
online, synchronous, lectures recorded (mediaspace)
Scribbles on website

Course Work
4 Hwls 40%
Presentation 20%
Project 40%
\{ may work in groups of \leq 3 \}

Prerequisite: undergrad algms (CS374)
(CS473 not required but helpful ...)

---

Theme - understand the complexity of fine-grained
basic algorithmic problems

but go beyond polynomial vs. NP-hard

**Ex1**
all-pairs shortest paths (APSP)
for \( n \)-weighted graph with \( n \) vertices
Floyd-Warshall (DP) \( O(n^3) \) time
Dijkstra \( n \) times \( O(n^3) \) time
better? \( \frac{1}{n^3} \) \( \Theta(n^3) \)
better?

$\sim O\left(\frac{n^3}{\log^3 n}\right) \ldots O\left(\frac{n^3}{\log^2 n}\right) \ldots O\left(\frac{n^3}{c\log n}\right)$

Fredman '75

C'07

Williams '14

Conj: no truly subcubic alg'm e.g. $O(n^{2.9999})$?

$\frac{\sqrt{\log n}}{c} > \log n$

$\frac{\sqrt{\log n}}{c} < n$

(many related graph problems: diameter, radius, shortest cycles, ...)

Ex 2

longest common subseq (LCS)

of 2 strings $a_1 \ldots a_n$

$\ldots b_n$

$DP \Rightarrow O(n^2)$ time

$L(i,j) = \max \left\{ \begin{array}{l} L(i-1,j) \\ L(i,j-1) \\ L(i-1,j-1) + 1 \quad \text{if } a_i = b_j \end{array} \right.$

better?

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

(many similar problems: edit dist, Fréchet dist, ...)

Ex 3

3SUM

given $n$ numbers $S$ & target $t$. 
3SUM

Given \( n \) numbers \( S \) & target \( t \),
\( \exists a,b,c \in S \) s.t. \( a+b+c=t \)?

trivial: \( O(n^3) \)

\( \text{std HW prob: } O(n^2) \)

(one way: sort all \( a+b \) ... )

(another way: \( n \) instances of 2SUM)

\( O(n) \) time after sorting

better?

Gao-Hedet-Pettie '14
\( \sim O\left( \frac{n^2}{\log^2 n} \right) \)
\( \sim O\left( \frac{n^2}{\log^2 n} \right) \)

 Conj: no truly subquad alg'm

( many related problems from geometry ... )

\( k \) SUM:

trivial \( O(n^k) \)

"meet-in-middle" \( O(n^{k/2} \log n) \) if \( k \) even
\( O\left( n^{k/2} \right) \) if \( k \) odd

better?

Subset-Sum:
\( \exists \) subset of \( S \) summing to \( t \)?

trivial: \( O(2^n) \) time

"meet-in-middle" \( \sim O(2^{n/2}) \) time
"meet-in-middle" \( \sim O(2^{n/2}) \) time \( \sim 2^{n/4} \) space

better?

for positive integers:

DP \( O( n t ) \) time (can be improved to \( O(t) \))

better?

Ex4

closest pair in moderate dims \( d \sim \log n \)

& other geom problems

\( \sim O(dn^2) \)

(trivial:

\( \sim O(dn^2) \)

(related prob: "orthogonal vectors" (OV))

Proving lower bds in general models is very difficult

Idea - prove conditional lower bd via reductions (under conjectures that certain basic probs are hard)

Similar to NP-completeness pf but fine-grained

E.g. Abbad - Grandoni - Vassiloukti W. '14:

if we could compute graph radius in \( O(n^{2.99}) \) time,

then \( \sim O(n^{2.99}) \) time
if we could compute your... in $O(n^{2.99})$ time, we could solve APSP in $O(n^{2.99})$ time

Bringmann '14: or edit dist, LCS
if we could compute Frechet dist in $O(n^{1.99})$ time, we could solve CNF-SAT in $O(1.99999^n)$ time
which refutes
Strong Exponential Time Hypothesis (SETH)

Course Outline

I. Basic Algmic Tools:
   - Convolution/FFT, matrix mult,

II. Conditional LBs
   - reductions... APSP/3SUM/SETH...

III. Advanced Algmic Techniques
   - log shaving
   - polynomial method
   - additive combinatorics