Streaming Algorithms

Meng-Tsung Tsai

05/29/2018

Schedule

Tuesday	Friday
Today	**Jun 1** Appl. of L_0-samplers.
Jun 5 No class. OH (EC 336)	**Jun 8** Proposal. **10:10 - 12:40**
Jun 12 Supporting Materials.	**Jun 15** Supporting Materials.
Jun 19 No class. OH (EC336)	**Jun 22** Presentation. **10:10-12:40**

Final Project Grading

The final project is a **perosnal** project and worth **30%** of your final grade.

1. Propose a plan in class on Jun 8.

2. Present your project in class on Jun 22.

* Everyone has ≤ 12 minutes to present.
* Ask your classmates ≥ 2 questions while they are presenting.
* You need to prepare slides for the presentation, and you "may" like to demo your program as well.
* Upload your slides to newe3 before your presentation.

Ex1. App of L_0-samplers

You are given a sequence of point insertions and deletions. Output a cricle to cover at least (1-ε)n points, where n is the number of points in the final set of points.

* What is the space usage?

* What is the running time?

* One query or multiple queries.

Any other ideas?

Ex2. App of L_0-samplers

You are given a sequence of edge insertions and deletions. If the final graph G has at least T triangles, output one of them.

* What is T?

* What is the space usage and the running time?

* Weighted triangle?

Any other settings?

Ex3. App of Communication Protocols

We know that EQ(n) can be decided w.h.p. by transmitting O(log n) bits.

Recall that GT(n) can be reduced to O(log n) copies of EQ(n) so that GT(n) can be decided w.h.p. by transmitting $O(\log^2 n)$ bits.

* Sorting?

* Median selection?

* Mode?

What else?

Ex4. App of Communication Protocols

Let X be a random variable. Design a protocol that can sending an outcome of X using expected k (~ Entropy(X)) bits, so that the other party can recover the outcome.

* You may use the Huffman code or some **self-defined** simple code.

* What if the sent messages pass through a tunnel that will reverse each bit with probability p?

Any other settings?

Ex5. App of Derandomization

Give a randomized algorithm, and show how to derandomize it via k-wise independence.

* A random partition gives a 2-approximation for Max-Cut, which can be derandomized by exhaustively search the k-wise independece probability space.

Any other variations?

Ex6.

Anything related to the topics that we learn from this course.

Guidelines

* Be creative.

* Make things simple.

* A successful example = a cute story + a little math + a work program.

Presentation (20%) // clarity
Creativity (25%)
Math (20%)
Program (20%)
Feedbacks (15%) // based on the quality of questions you ask when your classmates present their work