CS 514: Advanced Algorithms II – Sublinear Algorithms (Fall 2021)

Course Syllabus

1 Course Information

General information

Instructor: Sepehr Assadi. Email: `sepehr.assadi@rutgers.edu`. Office: CoRE 310.

Lectures: Tuesdays 3:00pm – 6:00pm at TIL-226 (Livingston campus).

Office hours: Thursdays 3:00pm – 4:00pm on Zoom (email the Instructor for the Zoom link).

Prerequisites: Mathematical maturity and basic background on algorithms, complexity theory, discrete mathematics, and probability at the introductory level (e.g., undergraduate courses). Some experience with theoretical computer science beyond the introductory level (say, Advanced Algorithms (CS 513)) will be helpful but it is not necessary.

Textbook: There is no official textbook for this course but some useful resources are listed on the course webpage.

Webpage: https://www.cs.rutgers.edu/~sa1497/courses/cs514-f21/

The webpage contains updated syllabus information as the semester progresses and a calendar. The lecture notes and problem sets will also be posted on the webpage. We will use Canvas for the announcements and releasing grades.

Students with disabilities: The students with disabilities are encouraged to discuss with me any appropriate accommodations that we might make on their behalf following the guidelines of the Office of Disability Services[1].

Statement of inclusivity: I am committed to creating a learning environment in which all of my students feel safe and included, regardless of race, ethnicity, religion, gender or sexual orientation. Because we are individuals with varying needs, I rely on your feedback to achieve this goal. I invite you to let me know about what I can stop, start, or continue doing to make sure every one of my students feels valued and can engage actively in our learning community.

Academic integrity: The students are expected to follow Rutgers and CS Department academic integrity policies[2] for all their work in this course.

Mask requirements: In accordance with Rutger's policy, masks must be worn during class meetings; any student not wearing a mask will be asked to leave.

[1] https://ods.rutgers.edu

[2] Rutgers policy: http://academicintegrity.rutgers.edu/
CS department policy: https://www.cs.rutgers.edu/academics/undergraduate/academic-integrity-policy

What should you expect to learn from this course?

The main goal of this course is to present the students with various advanced algorithmic ideas through the lens of sublinear algorithms – these are "ultra efficient" algorithms whose resource requirements (e.g. time or space) are substantially smaller than the size of the input that they operate on. Sublinear algorithms are particularly suited for processing massive datasets. Additionally, this course aims to prepare the interested students for doing research on algorithms in general and sublinear algorithms in particular.

The following is a tentative list of topics that will be covered in this course:

- **Sublinear Time Algorithms:** Which problems can be solved in time faster than even reading the entire input once? We will cover sublinear time algorithms for property testing, distribution testing, and graph problems. We will also examine query complexity as a main tool for proving lower bounds on the performance of sublinear time algorithms.
- **Streaming Algorithms:** Which problems can be solved in space smaller than what is needed to store the entire input? We will cover streaming algorithms for statistical estimation, numerical linear algebra, and graph problems. We will also examine communication complexity as a main tool for proving lower bounds on the performance of streaming algorithms.

Along the way, we will learn about various key ideas such as probabilistic analysis of algorithms, compressed sensing, dimensionality reduction, sparsification, sketching, (composable) coresets, etc., that are used extensively in algorithm design as a whole and sublinear algorithms in particular.

2 Assignments and Grading

Grading

- 40% Problem sets
- 40% Project
- 20% Scribe notes

Problem Sets

– **Timing:** There will be *three* problem sets in this course and a tentative schedule of release and due dates are available on the course calendar. Problem sets will be released on a Tuesday and are **due three weeks** later by **11:59 pm on Monday**.

– **Format:** Problem sets should be turned in on Canvas as **a single pdf file** containing the solutions in order. Moreover, **solutions must be typeset in LaTeX**. Simple instructions on using LaTeX are available on the course webpage and a template will be released with each problem set.

– **Collaboration:** Problem sets can (and probably should) be done in teams of two students. However, (1) the students should write their solutions *completely* independently (in particular, you should understand and be able to explain everything that is written in your solution); (2) you should include the name of your collaborator in your solutions.

When writing your solutions, you are *allowed* to use materials not discussed in the class (say, related research papers) **as long as you cite these references appropriately**. However, you are *not* allowed to get help from someone who is not currently enrolled in the class.

– **Difficulty of the problem sets?** The problem sets are going to be considerably challenging. You have three weeks to attempt each problem set so plan your time accordingly. However, do not get discouraged if you cannot solve some of the problems as they can be really hard – the goal of these

problems is to familiarize you with the type of technical problems that appear typically in research on algorithms, and, exactly as in research, you should not expect to be able to solve every problem.

Project

- There is a final project that will consist of exploring a topic of interest related to this course. This particularly involves reading one or two recent research papers **in complete details** to get a sense of the background on a research problem. The next step can then take one of the following two forms: either (*i*) providing further theoretical insights by proving new results on the problem, or (*ii*) providing further practical insights by implementing the current algorithms for the problem and measuring their performance.
- The students will have a presentation of the problem, its background, and their progress to the class at the end of the course. You will also be asked to present a written report of the project, focusing primarily on *your understanding* of the problem and any progress you have made by then.
- More details on the project plus the list of potential project ideas will be released later in the semester. However, feel free to discuss any project idea you may have with the Instructor before that.

Scribe notes

- For each lecture, there will be one team (of one or two students) in charge of taking detailed notes, typing them in LaTeX, preparing any needed figures, and sending them to the Instructor **by 11:59pm on Friday** after the lecture. These notes will be posted on the course website.
- If the quality of the notes are not satisfactory (e.g., having incomplete or imprecise proofs, many typos, lack of proper illustrations and examples, etc.), you will be asked to revise the notes until we converge to a version that is useful for your classmates. Keep in mind that the target audience of these notes is you – students – a couple of weeks after the lecture, when you already have forgotten what it was about. As such, every detail of the lecture should be addressed in the notes.
- The assignments of scribe notes will be fixed in a week or two – the first couple of lectures are scribed by the Instructor as an example. A LaTeX template with proper instructions for scribing the notes will be released as well.

3 Rutgers CS Diversity and Inclusion Statement

Rutgers Computer Science Department is committed to creating a consciously anti-racist, inclusive community that welcomes diversity in various dimensions (e.g., race, national origin, gender, sexuality, disability status, class, or religious beliefs). We will not tolerate micro-aggressions and discrimination that creates a hostile atmosphere in the class and/or threatens the well-being of our students. We will continuously strive to create a safe learning environment that allows for the open exchange of ideas and cherished freedom of speech, while also ensuring equitable opportunities and respect for all of us. Our goal is to maintain an environment where students, staff, and faculty can contribute without the fear of ridicule or intolerant or offensive language.

If you witness or experience racism, discrimination micro-aggressions, or other offensive behavior, you are encouraged to bring it to the attention to the undergraduate program director and/or the department chair. You can also report it to the Bias Incident Reporting System[3].

[3] http://inclusion.rutgers.edu/report-bias-incident/

4 COVID-19 Protocols

In order to protect the health and well-being of all members of the University community, masks must be worn by all persons on campus when in the presence of others (within six feet) and in buildings in non-private enclosed settings (e.g., common workspaces, workstations, meeting rooms, classrooms, etc.). Masks must be worn during class meetings; any student not wearing a mask will be asked to leave.

Masks should conform to CDC guidelines and should completely cover the nose and mouth[4].

If you are feeling sick, or suspect you may have been exposed to COVID-19, do not come to the class. Arrangements will be made for students who are not able to attend class because of an illness or quarantine.

[4] https://www.cdc.gov/coronavirus/2019-ncov/prevent-getting-sick/about-face-coverings.html