Syllabus for ECE 598 -- Molecular Storage and Computing: Practice and Theory

Summary

The course will introduce a number of DNA- and polymer-based data storage platforms and the relevant mathematical and biological concepts needed to understand their implementation. In the first part of the course we will describe modern synthesis and sequencing platforms and the problem of reconstructing sequences based on evidences sets of the form of substrings, subsequences or substring/subsequence weight. Topics of relevance on the biological side include reviews of Sanger and shotgun sequencing and nanopore sequencing. Topics on the mathematical side include sequence alignment, deBruijn graphs, deep learning methods for base calling, clustering methods for efficient synthesis, k-decks and trace reconstruction. In the second part of the course we will discuss random access and error-correction problems, with a special emphasis on PCR-based random access techniques and primer design, coding for shotgun sequencing, coded trace reconstruction and Catalan sequence based encoding methods. In the third part of the course we will discuss DNA editing mechanisms and topological storage, along with molecular computing methods based on strand displacement. Special emphasis will be placed on emerging in-memory computational paradigms such as SIM||DNA.

Prerequisites: ECE 313 (Probability with Engineering Applications). Not needed but desirable is basic knowledge of certain concepts from ECE 563 (Information Theory) and discrete mathematics. No background knowledge of genomics and computational biology is assumed.

Coursework: The course will combine standard lectures, several seminar-style presentations of research papers and project presentation. The grades will be based on:

1. **Five homework assignments (30% total).** One or more assignments will involve accessing data from a repository, processing it via an existing computational biology software tool and then performing some inference or learning task.
2. **Two paper reading/presentation assignments for paired students (30%).** Papers will be assigned based on student's research field and interests. Pairs of students will be asked to read two papers and each paper will be presented by exactly one student, chosen by the instructor.
3. **Project presentation (40%). Team member of the best ranked project will be given the option to implement their design experimentally and test its performance through the Center for High-Throughput Sequencing.**

Textbook: No textbook will be required. Instead, the instructor's tutorial papers on the subject and multiple seminar and keynote presentations will be used instead.

Subject areas:

Part 1:

a. A gentle introduction to molecular biology: DNA, RNA and the structure/organization of cells.
b. The Central dogma: DNA replication, transcription and translation.
c. Enzymes, DNA editing, and basic concepts in synthetic biology.

1-1.5 weeks, based on the composition of the class.

Part 2:

a. DNA synthesis, DNA amplification (PCR), DNA sequencing (Sanger sequencing, shotgun sequencing, nanopore sequencing).
b. Sequencing file formats and the process of base calling.
c. Deep learning architectures for base calling (CNNs, RNNs etc).
d. Edit distance, dynamic programs for computing the edit distance and sequence alignment (e.g., MUSCLE). Running MUSCLE on real genomic data
e. Sequence clustering with edit distances.
f. Synthesis protocols based on sequence clustering.
g. Sequence alignment. deBruijn graphs and the EULER software. Running EULER on real sequencing data.

3 weeks.

Part 3:

a. Limits of unique sequence reconstruction via Ukkonnen's and Skienna's analysis.
b. Reconstructing strings based on substrings.
c. Reconstructing strings based on traces and k-decks.
d. Reconstructing strings based on multiset compositions and the turnpike problem. Turnpikes in multiple dimensions.
e. Coded string reconstruction (coding for repeat removals, coded trace reconstruction, coded k-decks etc).

3 weeks.

Part 4:

a. Principles of DNA-based data storage in synthetic oligos.
b. Random access and the problem of combinatorial primer design. Guibas and Odlyzko's constrained coding for pattern avoidance.
c. Homopolymer codes and Asymmetric Lee distance codes.
d. Topological DNA-based data storage.
e. 2DDNA storage systems for images: Image compression, smoothing, inpainting and unequal error-control coding.
f. Polymer based data storage, mass spectrometry readouts and interleaved Catalan codes.

4-5 weeks.

Part 5:

a. DNA computing via hybridization – Adleman's Travelling salesmen problem.
b. DNA origami.
c. Code design for computing via hybridization.
d. Nussinov's algorithm and coding to mitigate secondary structures. Running Vienna and Rfold on real biological data.
e. DNA strand displacement.

f. Implementing sorting, counting and Rule 110 in memory – SIM||DNA.
3-3.5 weeks.