

# Syllabus for CS 5984 (Spring 2006)

## Computational Systems Biology

<http://www.cs.vt.edu/~murali/teaching/csb/csb.html>

**Instructor** T. M. Murali  
**Office** 2160B Torgerson Hall  
**Email** murali@cs.vt.edu

## Description

This new graduate course will introduce students to Computational Systems Biology (CSB), a rapidly-growing field that tries to understand the behaviour and function of a living cell as a system of interconnected components and modules. This field studies fundamental high-level biological questions such as:

- What are the properties of the biological networks in a cell?
- How do these networks interact with each other?
- How does a biological system behave over time under various conditions?
- How does a cell maintain its robustness and stability?
- How can we modify or construct biological systems with desired properties?

We will use a selection of papers from the literature to discuss these topics. Lectures by me, student presentations, and one or two invited lectures at the end of the semester will drive the course.

## Pre-requisites

You must have graduate student standing in the Department of Computer Science, a Life Science department, the Department of Mathematics, or the Department of Statistics.

### Computer Science students

- Required: Data and Algorithm Analysis (CS 4104) or equivalent.
- Preferred: Algorithms in Bioinformatics (CS 5124)
- Useful: Biological Paradigms for Bioinformatics (PPWS 5984).
- Useful: Applied Combinatorics (MATH 3134) or a similar course on combinatorics and graph theory.

### Life Science students

- Required: Undergraduate or graduate training in biology, genetics, biochemistry, or a similar subject.
- Preferred: Computation for Life Sciences (CS 5045).

### Other students

- Please meet me to discuss your qualifications.
- Preferred: Applied Combinatorics (MATH 3134) or a similar course on combinatorics and graph theory.
- Preferred: Computation for Life Sciences (CS 5045).

I will accept coursework at the undergraduate or graduate level that is equivalent to these pre-requisites. In addition to these pre-requisites, computational students should have prior programming experience and should be able to create a reasonable software package from a given project specification.

## Course Format and Grading Policy

Lectures, presentations of papers from the literature by groups of students, and invited lectures will drive the course. Class participation is extremely important. All students will complete a term paper or a software project and submit it at the end of the semester. Your group presentation will make up 20%, your class participation 30%, and your final project 50% of your grade. This ratio is commensurate with the amount of effort I expect you to put in each of these items.

I encourage students to form groups of 2–3 people before the second class of the semester, go over the list of papers on the course website, and carefully consider what topics and the papers you want to present. Choose at least two or three topics in case your choice conflicts with somebody else's. We will assign papers and topics to students on August 28 (in the second class). Student presentations will start the week of September 1 or the week of September 8. Depending on the number of students in the class and the number of groups, each group will make a presentation either once or twice during the semester.

**Presentation** The group of students assigned to present the papers should prepare a formal presentation and lead the discussion. Prepare your presentation like how you would for a conference or symposium. Use a set of prepared slides to drive your presentation but use the blackboard if necessary. I would like a copy of each group's slides. Please provide them to me in PDF, PostScript, or DVI format. If you use L<sup>A</sup>T<sub>E</sub>X, I would also like a copy of your L<sup>A</sup>T<sub>E</sub>X sources.

**Participation** I also expect every student to carefully read each paper assigned to each class and participate in the discussion. Given the interdisciplinary nature of CSB, it is critical that both life and computational scientists contribute to our understanding of the research we will all learn about.

**Final Project** The final project can either be a term paper or a group software project.

**Term paper** You can write the term paper on a superset of the papers you present in class or on a different set of papers, if you so choose. Write the paper as if you were submitting a review paper to a forum like *ACM Computing Surveys* or one of the *Nature Reviews* journals. If your paper is well-written, thoughtful, and broad in scope, you may indeed be able to submit it to such a journal.

**Software project** The group and I will decide on the specifications of the project. A significant component of your project must involve performing biological analysis of your results; if your group contains a life-student student, this analysis will primarily be his or her responsibility. A high-quality software project can result in a publication in a premier conference like ISMB, RECOMB, PSB, and/or an appropriate journal.

## Announcement

If any student needs special accommodations because of a disability, please contact me during the first week of classes.

## Intended Topics

We will discuss research papers in the following topics:

- the analysis of DNA microarray data: basic clustering algorithms; defining and computing biologically-interesting patterns of co-expression; computing condition-specific patterns of co-expression; identifying sets of co-regulated genes.
- high-throughput biological data: new experimental techniques for detecting protein-protein interactions, protein-DNA binding, low gene expression levels; assessing the quality of this data; improving the reliability of the data.

- regulatory networks: determining their basic building blocks and structural properties; extracting them from gene expression data;
- metabolic pathways: high-level structural properties; gleaning evidence of robustness in these pathways; reconstructing metabolic pathways from data; modelling metabolic networks.
- obtaining biological insights by combining diverse sources of data: functional annotation of proteins; improving the reliability of the data; refining our current knowledge about biochemical pathways.

The course web-site contains a detailed list of papers on each of these topics that we will select from for discussing in the course. Students can suggest other topics and papers to me. If you have any such ideas, talk to me as soon as possible so that we can include them in the schedule before it is settled.