Whole-Cell Modeling Summer School

Jonathan Karr

October 15, 2015

Timeline

- Application
  - Deadline: Nov 30
  - Decisions: Dec 15
- Oral abstract
  - Submission: Winter
  - Decisions: Winter
- School: Apr 3-8

Whole-Cell Modeling Summer School

Whole-cell models are promising tools for predicting phenotype from genotype by accounting for every individual gene and cell function. Whole-cell modeling has the potential to enable rational bioengineering and precision medicine. However, significant work remains to develop fully complete and accurate whole-cell models. The goal of the 2016 Whole-Cell Modeling Summer School is to provide young investigators cutting-edge training in large-scale dynamical modeling and model integration.

Why participate?

The course will be the first course focused on multi-algorithm whole-cell modeling. It will teach strategies for building and managing large models which aren't covered by any other course including multi-algorithm modeling, model organism database curation, surrogate modeling, and software development. The five-day course will feature didactic lectures, interactive hands-on tutorials, and student research talks. The mornings will feature lectures on modeling individual pathways. The afternoons will feature interactive hands-on tutorials on building and analyzing multi-algorithm models to generate and evaluate hypotheses. Throughout the course, students will work toward building a small whole-cell model. In addition, the course will include student talks to enable students to share their own research.

Who is the course for?

The course is designed for PhD students and postdoctoral scholars who wish to gain training in large-scale dynamical modeling. Students should already have a strong foundation in computational systems biology including dynamical modeling and scientific programming. See the pre-requesites section below for more information.

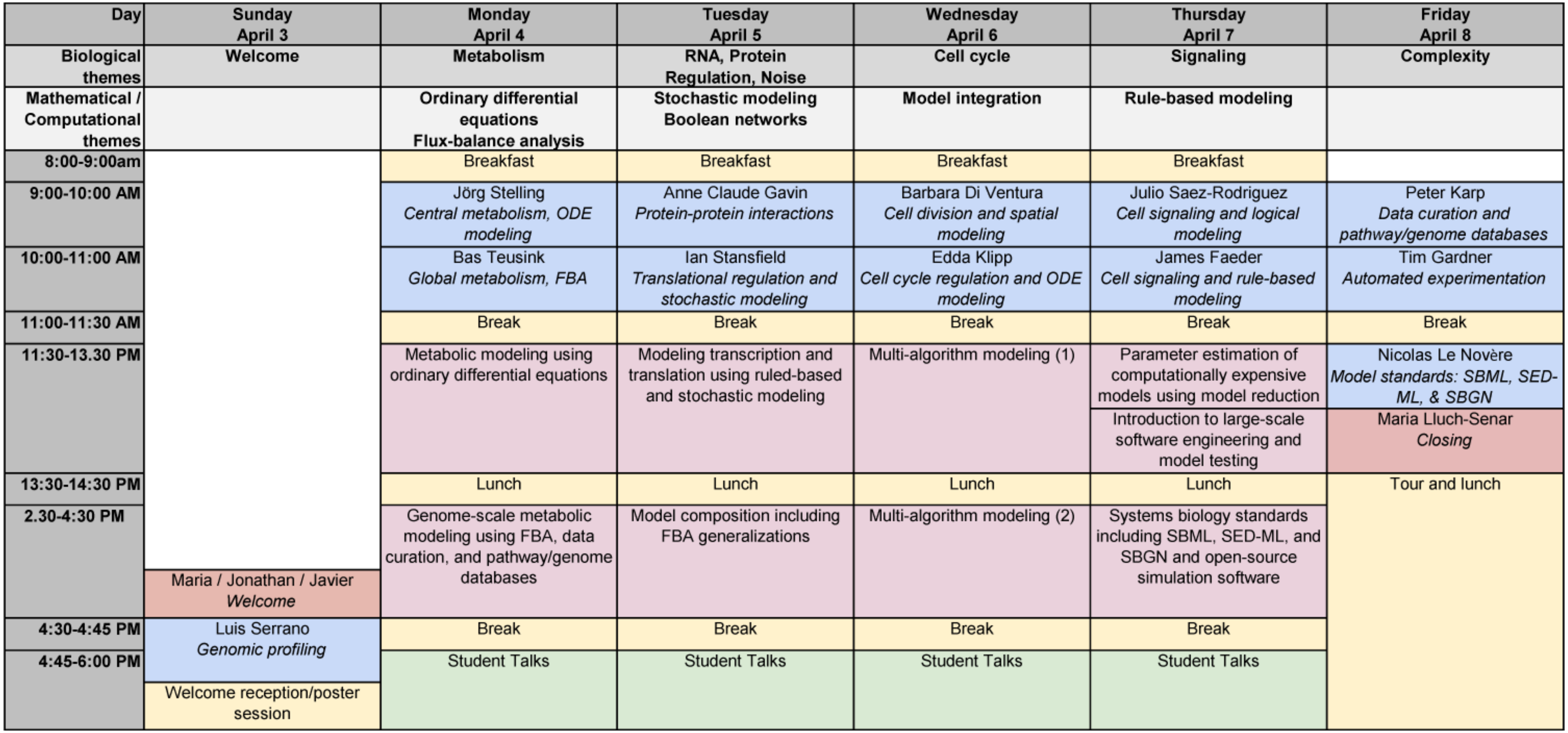
Prerequisites

The course will focus on teaching students theory and techniques for large-scale dynamical modeling. Participation in the course requires prior knowledge of dynamical modeling and computer programming including:

- Flux balance analysis
- Ordinary differential equations
- Stochastic modeling
- Object-oriented programming in MATLAB or Python

To enable non-biologists and non-quantitative scientists to participate in the course, we will pair students together for the tutorials such that every pair includes at least one biologist and one quantitative scientist/ engineer. The lectures will also provide all needed biology knowledge.

Schedule



Day	Sunday April 3	Monday April 4	Tuesday April 5	Wednesday April 6	Thursday April 7	Friday April 8
Biological themes	Welcome	Metabolism	RNA, Protein Regulation, Noise	Cell cycle	Signaling	Complexity
Mathematical / Computational themes		Ordinary differential equations Flux-balance analysis	Stochastic modeling Boolean networks	Model integration	Rule-based modeling	
8:00-9:00am		Breakfast	Breakfast	Breakfast	Breakfast	
9:00-10:00 AM		Jörg Stelling *Central metabolism, ODE modeling*	Anne Claude Gavin *Protein-protein interactions*	Barbara Di Ventura *Cell division and spatial modeling*	Julio Saez-Rodriguez *Cell signaling and logical modeling*	Peter Karp *Data curation and pathway/genome databases*
10:00-11:00 AM		Bas Teusink *Global metabolism, FBA*	Ian Stansfield *Translational regulation and stochastic modeling*	Edda Klipp *Cell cycle regulation and ODE modeling*	James Faeder *Cell signaling and rule-based modeling*	Tim Gardner *Automated experimentation*
11:00-11:30 AM		Break	Break	Break	Break	Break
11:30-13.30 PM		Metabolic modeling using ordinary differential equations	Modeling transcription and translation using ruled-based and stochastic modeling	Multi-algorithm modeling (1)	Parameter estimation of computationally expensive models using model reduction Introduction to large-scale software engineering and model testing	Nicolas Le Novère *Model standards: SBML, SED-ML, & SBGN* Maria Lluch-Senar *Closing*
13:30-14:30 PM		Lunch	Lunch	Lunch	Lunch	Tour and lunch
2.30-4:30 PM	Maria / Jonathan / Javier *Welcome*	Genome-scale metabolic modeling using FBA, data curation, and pathway/genome databases	Model composition including FBA generalizations	Multi-algorithm modeling (2)	Systems biology standards including SBML, SED-ML, and SBGN and open-source simulation software	
4:30-4:45 PM	Luis Serrano *Genomic profiling*	Break	Break	Break	Break	
4:45-6:00 PM	Welcome reception/poster session	Student Talks	Student Talks	Student Talks	Student Talks	

- 4 full days
- 12 1-h morning lectures
- 8 2-h afternoon tutorials
- 4 1.25-h evening student talks
- Poster session

- Tour
- Lunches
- Dinner on own

Lectures

No.	Topic	Instructor
1.	Genomic profiling	Luis Serrano
2.	Central metabolism dynamics and ODE modeling	Jorg Stelling
3.	Global metabolism and FBA	Bas Teusink
4.	Protein-protein interactions	Anne-Claude Gavin
5.	Translational regulation and stochastic modeling	Ian Stansfield
6.	Cell division and spatial modeling	Barbara Di Ventura
7.	Cell cycle regulation and ODE modeling	Edda Klipp
8.	Cell signaling and logical modeling	Julio Saez-Rodriguez
9.	Cell signaling and rule-based modeling	James Faeder
10.	Data curation and pathway/genome databases	Peter Karp
11.	Automated experimentation	Tim Gardner
12.	Systems biology standards including SBML	Nicolas Le Novere

Lectures

- Cover range of biological pathways
  - Metabolism
  - Translation
  - Signaling
  - Cell cycle
  - Cell division
- Cover range of experiments
  - Genomics
  - Mass-spectrometry
  - Automated experimentation
- Cover range of modeling formalisms
  - ODEs
  - FBA
  - Spatial
  - Logical
  - Rule-based
- Cover range of modeling methods
  - Data curation
  - Representation formats
- Lectures do not cover whole-cell modeling specifically
  - Data integration
  - Model composition
  - Multi-algorithm modeling
  - Parameter estimation
  - Modeling testing & validation

Tutorials

- Focus on whole-cell modeling
  - Data curation
  - Modeling formalisms
  - Model composition
  - Parameter identification
  - Modeling testing
- Interactive
  - Most of the time should be spent at the computer coding
- Mini lectures
  - Where necessary, tutorials need to start with brief lectures to introduce students to concepts in tutorials not covered by lectures

Tutorial workload

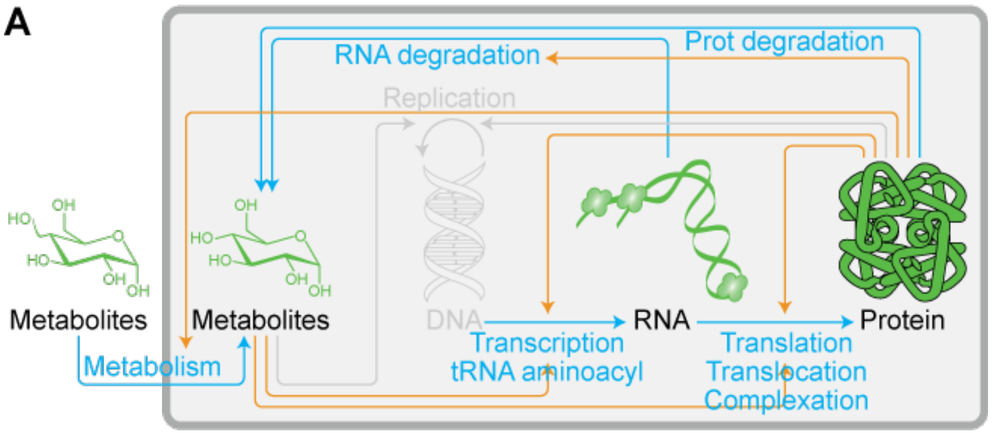
- 8 tutorials, 7 instructors → 1 tutorial per instructor
- Each instructor creates 1 tutorial
- Each instructor reviews 1-2 tutorials
- Everyone helps students work through tutorials

Draft tutorial

- Introductory presentation slides (20 min max)
- Hands-on exercise
  - Prompts questions
  - Code fragments
  - Data
- Exercise answers
  - Code
  - Figures
  - Writeup
- Brief supplemental reading list

Tutorials

- Build up to constructing a (small) whole-cell model
  - Composed of simple submodels
  - Submodels implemented using multiple formalisms
  - State uses multiple representations (population, particle)
- Start with 1-2 submodels
- Teach methods for combining models
- Teach methods for identifying models, including reducing models
- At every point, ask students to make model predictions
- Track how predictions change as additional pathways incorporated into model



Tutorials

No.	Topic	Instructor
1.	Pathway modeling using ordinary differential equations	Veronica
2.	Qualitative genome-scale modeling using and public data	Maria
3.	Ruled-based approaches to quantitative genome-scale modeling	Samuel
4.	Model composition including FBA generalizations	Marc
5.	Multi-algorithm modeling I	Jonathan
6.	Multi-algorithm modeling I	Javier Carrera
7.	Parameter estimation of computationally expensive models using model reduction	Jonathan/Javier
8.	Introduction to large-scale software engineering and model testing	Marie
9.	Systems biology standards and open-source software	Yin Hoon

Tutorials

- **Tutorial 1:** Pathway modeling using ordinary differential equations – Review of pathway modeling. Students will develop an ODE model of central carbon metabolism based on a mock journal article. Students will develop mathematical equations, write Python scripts, and conduct in silico experiments to understand the factors which control the reaction rates.
- **Tutorial 2:** Qualitative genome-scale modeling using and public data – Students will learn how to construct qualitative genome-scale models based on public data. Students will retrieve chemical reactions from EcoCyc and use these reactions to develop an FBA model of Escherichia coli. Students will conduct in silico experiments to understand how the growth rate is controlled.
- **Tutorial 3:** Ruled-based approaches to quantitative genome-scale modeling – Students will learn how to construct quantitative genome-scale models using rule-based modeling and high-throughput data. Students will construct reaction patterns that describe transcription and RNA degradation, expand their reaction patterns using transcriptomic data, and implement their models in Python.
- **Tutorial 4:** Model composition including FBA generalizations – Students will learn about model composition by using reaction kinetic data to directly combine the models they created in Tutorials 2 and 3. Students will use the combined model to explore how gene expression controls growth.
- **Tutorials 5/6:** Multi-algorithm modeling – Students will learn multi-algorithm modeling. Students will combine four sub-models into a single model by creating global state variables, mapping the sub-models onto the state variables, and implementing a serial multi-algorithm numerical integrator. Students will also be given Python scripts for two additional multi-algorithm integrators, and will compare the integrators.
- **Tutorial 7:** Parameter estimation of computationally expensive models using model reduction – Students will learn how to identify the parameters of large models using maximum likelihood estimation, model reduction, and numerical optimization. Students will reduce the integrated model that they developed in Tutorial 6 and numerically optimize their reduced model to identify its parameter values.
- **Tutorial 8:** Introduction to large-scale software engineering and model testing – Students will learn practical skills for building large models including revisioning, unit testing, and continuous integration. Students will setup a repository and encode their experiments from Tutorial 6 into unit tests.
- **Tutorial 9:** Systems biology standards and open-source software – Students will learn the SBML and SED-ML standards and related open-source software. Students will encode the models that they developed in Tutorials 2 and 3 into SBML and simulate them using SED-ML, COPASI, and COBRApy

Tutorials timeline

- MATLAB
- April 3-8
- Drafts: Dec 30
- Refine: January
- Answers: Early March

- Collect lecture abstracts