Symbolic systems biology
SSB @ SRI
  * BioCyc
  * Hybrid SAL
  * Neuron systems
  * Pathway Logic
SYMBOLIC SYSTEMS BIOLOGY
Biological processes are complex
- signaling, regulation, defense, ....

Huge dynamics timescales: microseconds to years

Spatial scales over 12 orders of magnitude
- single protein to cell to organ to organism ...

Oceans of experimental biological data generated

Important intuitions captured in mental models that biologists build of biological processes

How to build in silico models from all this?
SYMBOLIC SYSTEMS BIOLOGY

- Symbolic -- represented in a logical framework
- Systems -- how things interact and work together, integration of multiple parts, viewpoints and levels of abstraction
- Goals:
  - Develop formal models that are as close as possible to domain expert’s mental models
  - Compute with, analyze and reason about complex networks
  - New insights into / understanding of biological mechanisms
Executable Formal Models

- Describe system states and rules for change in a formal system
- From an initial state, derive a transition graph
  - nodes -- reachable states
  - edges -- rules connecting states
- Path in transition graph ~ computation/derivation
- Many kinds of analysis available
Static Analysis
- how are elements organized -- sort hierarchy
- control flow / dependencies
- detection of incompleteness

Forward simulation from a given state (prototyping)
- run model using a specific strategy
- fast, first exploration of a model
Symbolic Analysis II

- Forward search from a given state
  - breadth first search of transition graph
  - find ALL possible outcomes
  - find only outcomes satisfying a given property

- Backward search from a given state S
  - run a model backwards from S
  - find initial states leading to S
  - find transitions that contribute to reaching S
Model checking
determines if all pathways from a given state satisfy a given property, if not a counter example is returned

example property:
- molecule X is never produced before Y

counter example:
- pathway in which Y is produced after X
**Symbolic Analysis IV**

- **Constraint solving**
  - Find values for a set of variables satisfying given constraints.
- **MaxSat deals with conflicts**
  - weight constraints
  - find solutions that maximize the weight of satisfied constraints
- **Finding possible steady state flows of information or chemicals through a system can be formulated as a constraint problem.**
WHAT IS BIOCYC?

- A collection of Pathway/Genome Databases (PGDBs) databases
- MetaCyc: reference resource for metabolic pathways
- EcoCyc: E.Coli, HumanCyc ..
- And a tool suite
  - PathoLogic: to create PGDB from an annotated genome
  - Pathway/Genome Navigator: query, visualization, and analysis of PGDBs.
  - Pathway Editor: for pathway curation
WHAT IS A PGDB?

- A PGDB for an organism contains
  - genome and gene products
  - metabolic pathways, reactions (elements of pathways),
  - enzymes (that enable reactions), metabolites and transporter complement;
  - the genetic control network:
    - operons, transcription factors;
    - interactions between transcription factors,
    - small-molecule ligands, and DNA binding sites
Diet Planning for E. Coli

- Find all potential minimal nutrient sets for E. Coli
  - Start with the metabolic network, and a list of candidate nutrients
  - Define growth conditions as a set of products
  - Represent this information as a constraint system where the variables are reaction fluxes. (Cycles are tricky)

- Managing complexity
  - Use disjunction to avoid non-linearity
  - Eliminate impossible/useless compounds

- Transform into finding prime implicants of monotone boolean function
- Solve using BDDs and Yices solver
GOTCHA’S

- There are LOTS of solutions
  - Define a notion of equivalence on compounds and look at only canonical/reduced solutions -- hundreds rather than thousands
  - Side effect -- insights into role of equivalence classes
- Unexpected solutions: Why does the bug grow on CO2?
- This is a great way to debug a reaction network!
- We are getting close to a fixed point.
HYBRID SAL
Hybrid systems combine discrete and continuous variables

Hybrid SAL -- hybrid systems modeling language and tools
  - sound simplification and abstraction techniques
  - unknowns represented as parameters with logical constraints
  - model checking for finite state abstractions
B. SUBTILIS SPORE FORMATION

- Initiated when environment not conducive to growth
- Regulated by transcriptional network
- Hybrid model developed to study bistability

- Effect of promoters and inhibitors -- discrete transitions
- Unknown rate constants --- constrained parameters
- Determined to be bistable for range of parameters
- Bistability properties sensitive to discrete logic
- Needed to refine transcriptional logic for Spo0A
Neuron Systems:
Aplysia CPG Model
Aplysia is a sea slug often used as model to study neuron systems.

- Its neurons are large
- There are relatively few types
- It exhibits simple but interesting behaviors (biting, learning, memory) that can be correlated to neuron modules / programs
1. Traditional: Hodgkin Huxley model
   - differential equations describing different ion channels
2. Simple neuron model proposed by Iz a vich
   - based on a dynamical systems view.
   - 4 parameters -- many spiking patterns
3. Automata model -- further abstraction to capture key features of the 4 parameter systems.
   - 1-2 are analyzed by simulation
   - 3 we can analyze by model-checking!
SAL Model of Aplysia CPG

biologists cartoon model of CPG: neurons that control biting response
Spiking patterns observed during protraction (P) and retraction (R) phases

- Key features are represented as LTL formulae and model checked
- Example P1: B31 (eventually) reaches a plateau and stays there all through the protraction phase and until the start of retraction.

  \[ \text{aplysia} \models F(\text{levels}[B31]=N \land U(\text{levels}[B31] \geq N-1, \text{phase}=\text{retraction})) ; \]

- Model checking of similar properties was used to tune and validate the SAL model.
PATHWAY LOGIC (PL)
REPRESENTATION OF SIGNALING

http://pl.csl.sri.com/
Signaling pathways involve the modification and/or assembly of proteins and other molecules within cellular compartments into complexes that coordinate and regulate the flow of information.

Signaling pathways are distributed in networks having stimulatory (positive) and inhibitory (negative) feedback loops, and other concurrent interactions to ensure that signals are propagated and interpreted appropriately in a particular cell or tissue.

Signaling networks are robust and adaptive, in part because of combinatorial complex formation (several building blocks for forming the same type of complex), redundant pathways, and feedback loops.
Egf (EGF) binds to the Egf receptor (EgfR) and stimulates its protein tyrosine kinase activity to cause autophosphorylation, thus activating EgfR. The adaptor protein Grb2 (GRB2) and the guanine nucleotide exchange factor Sos1 (SOS) are recruited to the membrane, binding to EgfR. The EgfR complex activates a Ras family GTPase. Activated Ras activates Raf1, a member of the RAF serine/threonine protein kinase family. Raf1 activates the protein kinase Mek (MEK), which then activates Erk (MAPK).

Egf stimulation of the Mitogen Activated Protein Kinase (MAPK) pathway.

Egf → EgfR → Grb2 → Sos1 → Ras → Raf1 → Mek → Erk

- Egf (EGF) binds to the Egf receptor (EgfR) and stimulates its protein tyrosine kinase activity to cause autophosphorylation, thus activating EgfR.
- The adaptor protein Grb2 (GRB2) and the guanine nucleotide exchange factor Sos1 (SOS) are recruited to the membrane, binding to EgfR.
- The EgfR complex activates a Ras family GTPase.
- Activated Ras activates Raf1, a member of the RAF serine/threonine protein kinase family.
- Raf1 activates the protein kinase Mek (MEK), which then activates Erk (MAPK).
Rewriting Logic is a logical formalism that is based on two simple ideas:
- states of a system are represented as elements of an algebraic data type
- the behavior of a system is given by local transitions between states described by rewrite rules

It is a logic for executable specification and analysis of software systems, that may be concurrent, distributed, or even mobile.

It is also a (meta) logic for specifying and reasoning about formal systems, including itself (reflection!)
Pathway Logic (PL) is an approach to modeling biological processes as executable formal specifications (in Maude). The resulting models can be queried:

- Using formal methods tools: given an initial state
  - Execute --- find some pathway
  - Search --- find all reachable states satisfying a given property
  - Model-check --- find a pathway satisfying a temporal formula

- Using reflection
  - Find all rules that use / produce X (for example, activated Rac)
  - Find rules downstream of a given rule or component
  - Translate to alternative formalism and export
A Pathway Logic (PL) system has four parts
- Theops --- sorts and operations
- Components --- specific proteins, chemicals ...
- Rules --- signal transduction reactions
- Dishes --- candidate initial states

Knowledge base: Theops + Components + Rules
Equational part: Theops + Components

A PL cell signaling model is generated from
- a knowledge base
- a dish
**RULE 1: RECEPTOR BINDING**

If a dish contains an EgfR ligand (?ErbB1L:ErbB1L) outside a cell with EgfR in the cell membrane then the ligand binds to exterior part of the receptor and the receptor is activated.

\[
\text{rl}[1.\text{EgfR.act}]:
?\text{ErbB1L:ErbB1L} \quad \text{[CellType:CellType | ct \{CLm | clm EgfR\}]}
\Rightarrow
\text{[CellType:CellType | ct \{CLm | clm ([EgfR - act] : ?\text{ErbB1L:ErbB1L})\}].}
\]

Rule 1 applies to rasDish

\[
\text{PD(Egf} \quad \text{[Cell} \quad \text{\{CLm | EgfR PIP2\}} \quad \text{[CLi | [Hras - GDP] Src}}
\quad \text{[CLc | Gab1 Grb2 Pi3k Plcg Sos1}]\}
\]

with the match

?\text{ErbB1L:ErbB1L} := \text{Egf}
\text{clm} := \text{PIPI2}
\text{ct} := \{\text{CLi} | \text{[Hras - GDP] Src} \quad \text{[CLc | Gab1 Grb2 Pi3k Plcg Sos1]}\}

giving rasDish1

\[
\text{PD([Cell} \quad \text{\{CLm | ([EgfR - act] : Egf) PIP2\}}
\quad \text{[CLi | [Hras - GDP] Src}}
\quad \text{[CLc | Gab1 Grb2 Pi3k Plcg Sos1}]\}
\]
How do we Infer Rules?

Rules are inferred from evidence, called datums, curated from the literature.

Consider the rule converting Rala-GDP to Rala-GTP in response to Egf stimulation

\[
\text{rl}[1064.\text{Rala.irt.Egf}]:\\{EgfRC \mid \text{EgfR - act} : \text{Egf Pi3k RalGds clm}}\\{\text{CLi} \mid \text{cli [Rala - GDP]}}\\=\\{EgfRC \mid \{\text{EgfR - act} : \text{Egf Pi3k RalGds [Rala - GTP] clm}}\\{\text{CLi} \mid \text{cli} \}}.
\]

Evidence for this rule includes:

DID#05387: Rala[Ab] GTP[BD-PD] is increased irt Egf (tnr)
cells: Cos7 in BMLS
inhibited by: Wortmannnin [chem] --- Pi3k Inhibitor
inhibited by: LY294002 [chem] ""
source: 15034142-Fig-5c

DID#12876: xRala[xAb]IP GTP/GDP[32Pi-TLC] is increased irt Egf
cells: Cos1-xRalGds in BMS
times: 0 1+ 2++ 3++ 4+ 5 min
reqs: xRalGds [omission]
inhibited by: xRalGds-C203S "membrane-binding-mutant" [substitution]
comment: cells were pretreated with Vanadate 30 min before Egf treatment
source: 9416833-Fig-2
THE PATHWAY LOGIC ASSISTANT (PLA)
THE PATHWAY LOGIC ASSISTANT (PLA)

- Provides a means to interact with a PL model
- Manages multiple representations
  - Maude module (logical representation)
  - PetriNet (process representation for efficient query)
  - Graph (for interactive visualization)
- Exports Representations to other tools
  - Lola (and SAL model checkers)
  - Dot -- graph layout
  - JLambda (interactive visualization, Java side)
Hras activated
Parallel paths
Cross talk
Synchronization
Conflict
A SIMPLE QUERY LANGUAGE

- Given a Petri net with transitions \( P \) and initial marking \( O \) (for occurrences) there are two types of query
  - subnet
  - findPath - a computation / unfolding
- For each type there are three parameters
  - \( G \): a goal set---occurrences required to be present at the end of a path
  - \( A \): an avoid set---occurrences that must not appear in any transition fired
  - \( H \): a list of identifiers of transitions that must not be fired
- subnet returns a subnet containing all (minimal) such pathways (using backward and forward collection)
- findPath returns a pathway (transition list) generating a computation satisfying the requirements (using model checking on the negation).
FULL MODEL OF EGF STIMULATION
THE ERBB NETWORK (CARTOON FORM)
PL EGF Model

Events that could occur in response to EGF

Curated by Merrill Knapp
Subnet containing all pathways leading to activation of Erk.

Obtained by backwards followed by forwards collection
The idea is to go from all possible pathways to a plausible set, given the context.

- a list of 85 protein state changes demonstrated experimentally to occur in response to a short stimulus with Egf were set as goals and a set of concurrent paths were produced by PLA. This subnet ensures that the paths used to reach chosen goals are mutually compatible.
- (reachability of all of the goals is also a test of the model)
- Egf Rules, with requirements specific to Egf signaling, were given preference over Common Rules
THE CONSTRAINED EGF MAP
The path leading to activation of Erk1/2 in the constrained Egf network.

This path exists in the context of all the other experimental observations,
SLEEP
(with MaryAnn Greco and Merrill Knapp)
What is the function of sleep?

What are your cells doing when you sleep? vs awake?

Rat model -- proteomics from different organs at different sleep states
NATURAL SLEEP PARADIGM

Lights on Period - Continuous EEG Recording

8AM

48h Baseline Recording

Electrode Cables and Catheter Extensions Connected

3PM 4PM

8PM

Sacrifice Rat

Post 10’ WAKING

Post 10’ SWS

Post 1.5’ - 2’ REM

TOD
Proteins unique to different states were identified. Those modeled in PL included Actin and Rhob. Use the PLA explorer to find signaling connections.
EXPLORING PL KB FROM ACTIN
A HYPOTHETICAL MODEL PATHWAY RELATING STATE AND SYNAPTIC PLASTICITY

Wake state:
- unknown signal(s)
- $\Rightarrow$ phosphorylation of Rock1
- $\Rightarrow$ activation of Limk1
- $\Rightarrow$ phosphorylation of cofilin
- $\Rightarrow$ increase in polymerized actin
  (Phosphorylated cofilin is unable to depolymerize actin)

SWS:
- RhoDG11 binds Rhob-GDP
  (is not phosphorylated)
- $\Rightarrow$ Rock1, Limk1, and cofilin would not be phosphorylated and
- $\Rightarrow$ actin depolymerization
- $\Rightarrow$ decrease in synaptic weight

TODO -- test the hypothesis
CHOOSING A FORMALISM
REQUIREMENTS

- What do you want to represent?
- What questions do you want to ask?
- What material/data is available, what must be estimated/hypothesized?
- How are you going to validate?
- What are you familiar with?
- What formalism features:
  - Language primitives, semantic models, tools
  - model analysis/validation vs system analysis
What to Model?

- Species: Individual, concentration, population
- Structure: compartments, binding, location
- Process:
  - non-determinism -- computation trees
  - stochastic / probabilistic -- MC
  - deterministic -- ODE
- Quantitative: kinetics, dynamics,
- Qualitative: interactions, causal relations ...
- Effects of perturbation
  - Knockout / KnockIn / Mutations
  - Stimulus / Stress
WHAT QUESTIONS?

Examples:
- All the ways to phosphorylate Erk?
- How fast can signal get to nucleus?
- Is Pi3k required for activation of Raf?

Categories:
- Reachability
- Information/material flow
- Dynamical features
  - steady states, stable states, oscillations
  - competition/race conditions/interference
- Kinetic profiles
Some Points in Design Space

- Data source:
  - KEGG/SBML library → Model → Analysis
  - Data Curation → Datums → Rules → Model → Analysis

- Representation of components/species
  - blackbox --- constant declaration
  - explicit attributes of state --- Algebraic specification
  - as behaviors --- process calculus
MORE POINTS IN DESIGN SPACE

- Properties/questions -- tools
  - qualitative -- search/model checking/constraint solving
  - stochastic -- simulation/statistical/stochastic mc
  - continuous/kinetic -- ODE solvers ...
- View/perspective
  - physical agent point of view -- interaction centric
    - rules emerge from process descriptions
  - state/event -- centric
    - states are static structures,
    - rules specify behavior / interactions directly
FUTURE CHALLENGES

- Integration of signaling and metabolic networks
- Modeling action of transcription factors
- Modeling domain-domain interactions
- Adding semiquantitative information
- Algorithms to discover meaningful subnets
- Integrating models
SRI SSB TEAM AND COLLABORATORS

- PL Founders:
  - Patrick Lincoln
  - Keith Laderoute
- PL Core team
  - Carolyn Talcott
  - Steven Eker
  - Merrill Knapp
  - Andy Poggio
  - Malabika Sarker
  - Ashish Tiwari
  - Frederic Vigneault
- BioCyc
  - Peter Karp
  - Markus Krumenaker
  - Alex Shearer
- SRI MP Biologists
  - Anna Lisa d’Andrea
  - MaryAnn Greco
- SRI SV (CADRE)
  - Krishna Kodukula
  - Amit Galande
  - Rajeev Vaidyanathan
- LBNL -- Cancer Biology
  - Joe Gray
  - Paul Spellman
  - Laura Heiser
- Walter Reed
  - Marti Jett
- U Texas Houston/San Antonio
  - Doug Baxter
  - Riccardo Mozzachiodi