ADAM: Analysis of Discrete Models of Biological Systems Using Computer Algebra

Bonny Guang, Madison Brandon, Rustin McNeill

August 6th, 2010

Why use models in Biology?

- In biological systems, concerned with how different elements in the system interact with one another.
- One way to describe such interactions: create a model which describes the system.
- Can be either quantitative or qualitative descriptions, or both.
- One can obtain relevant information about system from models without having to perform costly experiments.

Discrete vs. Continuous Models

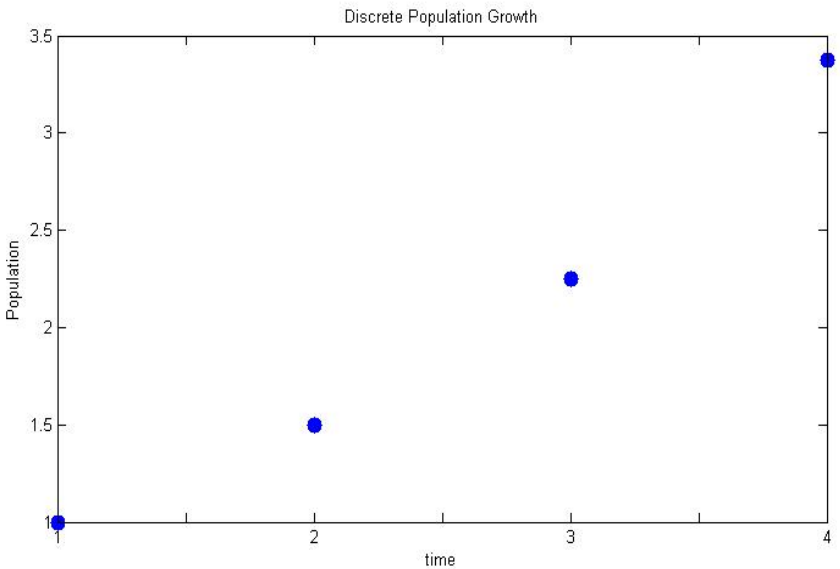
- Simple Model of Population Growth
- Given $P_0 = 1$, $r = 0.5$, and $K = 100$.

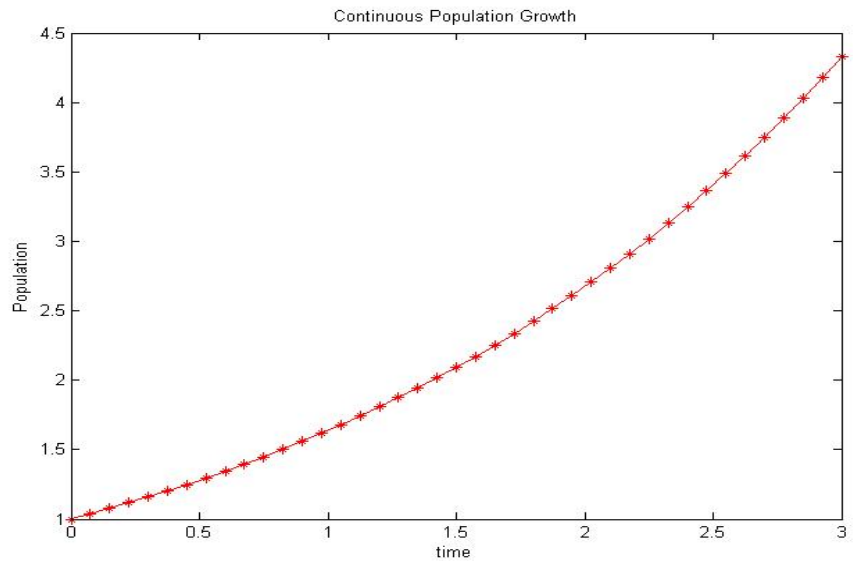
Discrete

$P_t = P_t + rP_t,\ t = 0, 1, 2, 3$

Continuous

$\frac{dP}{dt} = rP(1 - \frac{P}{K}),\ t \in [0, 3]$





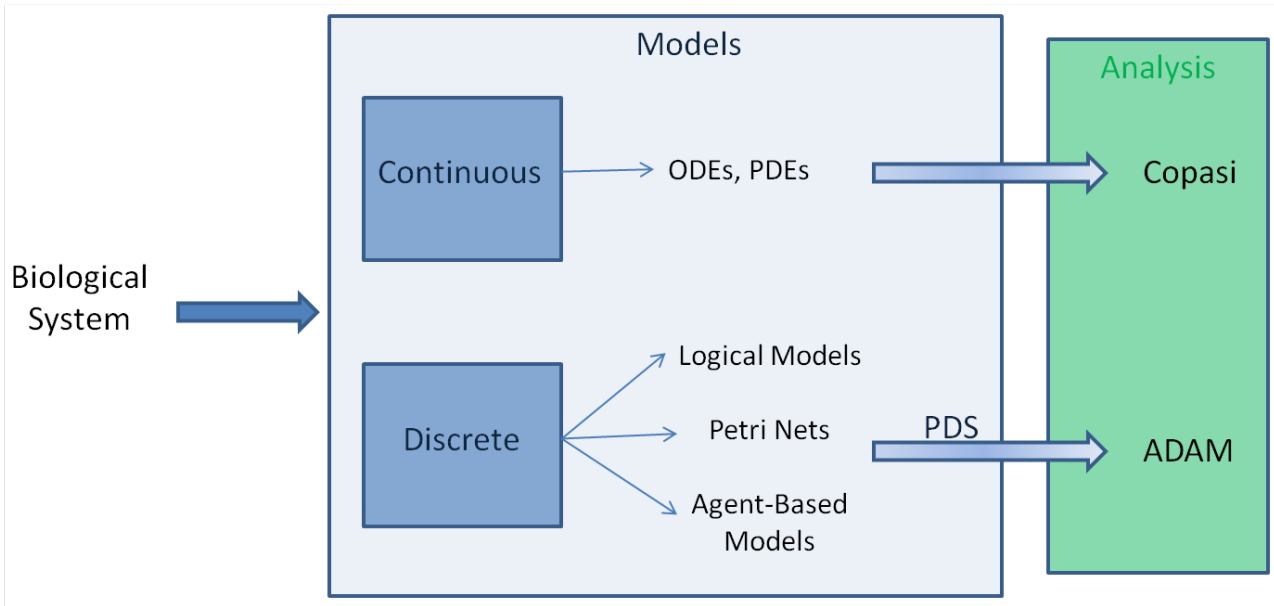
Models in Biology

Continuous Models:

- Rely on exact parameter rates
- Often not intuitive
- Many tools available for analysis

Discrete Models:

- Finite number of states
- Intuitive
- Few mathematical tools available for analysis



Applications of Discrete Models

Molecular Systems Biology 5; Article number 331; doi:10.1038/msb.2009.87
Citation: *Molecular Systems Biology* 5:331

www.molecularsystemsbiology.com

molecular systems biology

Discrete logic modelling as a means to link protein signalling networks with functional analysis of mammalian signal transduction

Julio Saez-Rodriguez[1,2,3,5], Leonidas G Alexopoulos[1,2,3,5,6], Jonathan Epperlein[1,2], Regina Samaga[4], Douglas A Lauffenburger[1,3], Steffen Klamt[4] and Peter K Sorger[1,2,3,*]

BMC Systems Biology

BioMed Central

Research article

Open Access

Reconstruction and logical modeling of glucose repression signaling pathways in *Saccharomyces cerevisiae*

Tobias S Christensen[†1,2], Ana Paula Oliveira[†1,3] and Jens Nielsen[*1,4]

Address: [1]Center for Microbial Biotechnology, Department of Systems Biology, Technical University of Denmark, Building 223, DK-2800 Kgs. Lyngby, Denmark, [2]Current address: Department of Chemical Engineering, Massachusetts Institute of Technology, Building 66, 25 Ames Street, Cambridge, MA 02139, USA, [3]Current address: Institute for Molecular Systems Biology, ETH Zurich, CH-8093, Zurich, Switzerland and [4]Current address: Department of Chemical and Biological Engineering, Chalmers University of Technology, SE-412 96, Gothenburg, Sweden

Computational Biology and Chemistry 33 (2009) 46–61

Contents lists available at ScienceDirect

Computational Biology and Chemistry

journal homepage: www.elsevier.com/locate/compbiolchem

Modeling of the U1 snRNP assembly pathway in alternative splicing in human cells using Petri nets

J. Kielbassa[a,1], R. Bortfeldt[a], S. Schuster[a], I. Koch[b,*]

[a] *Friedrich-Schiller-University of Jena, Department of Bioinformatics, Ernst-Abbe-Pl. 2, 07743 Jena, Germany*
[b] *Technical University of Applied Sciences Berlin, FB VI, Bioinformatics, Seestr. 64, 13347 Berlin, Germany*

Key Dynamical Features

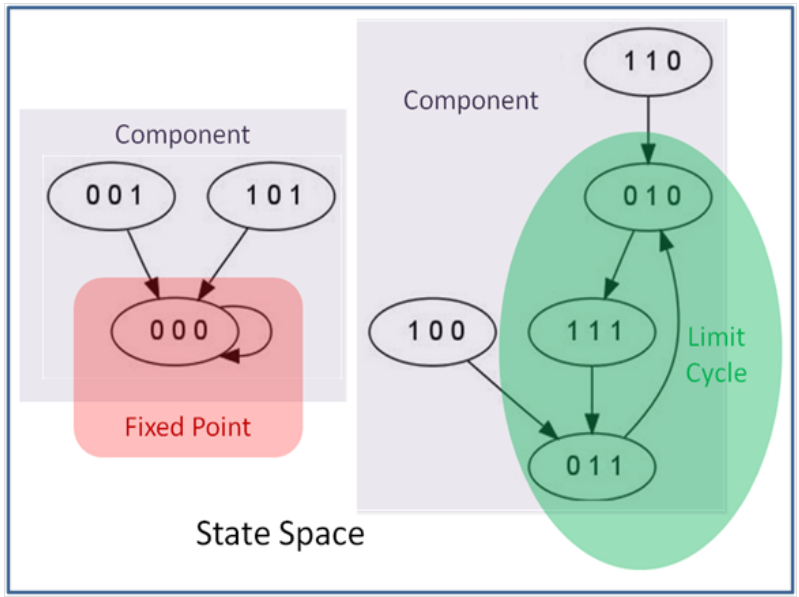


Figure: Example State Space

- Fixed Point - may give researcher info about what combination of gene states leads to permanently fixed state.
- Limit Cycle - limit cycles and their length can indicate recurring processes in the cell cycle.
- Component - typically modeler expects small limit cycles with large component sizes.

Simple Example

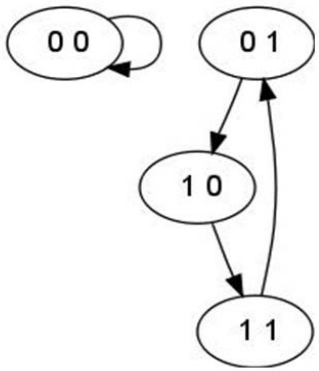
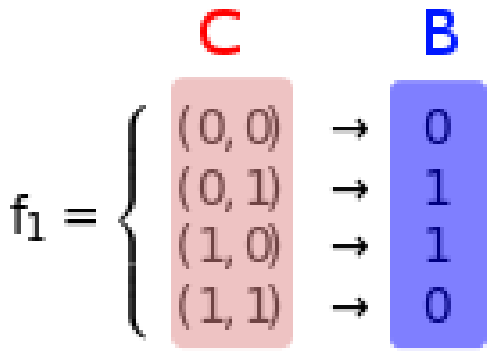


Figure: State space for a 2 by 2 system

x1	x2	f1	f2
0	0	0	0
0	1	1	0
1	0	1	1
1	1	0	1

Figure: Truth table for 2 by 2 system

Simple Example



Can write any boolean function defined this way as polynomial $f : \mathbb{F}_2^2 \to \mathbb{F}_2^2$ where

$$f_1(x1, x2) = \sum_{i=1}^{4} b_i \prod_{j=1}^{2} (1 - (c_{i,j} - x_i))$$

where $c_{i,j}$ are the i, jth entries into the matrix C and b_i are the ith entries into the vector B.

Simple Example

Then

$$\begin{aligned} f_1 &= 0 * (1 - (0 - x_1)(1 - (0 - x_2)) \\ &+ 1 * (1 - (0 - x_1))(1 - (1 - x_2)) \\ &+ 1 * (1 - (1 - x_1))(1 - (0 - x_2)) \\ &+ 0 * (1 - (1 - x_1))(1 - (1 - x_2)) \end{aligned}$$

$$= (1 - x_1)x_2 + x_1(1 - x_2)$$

$$= x1 + x2$$

This is a polynomial in $\mathbb{F}_2^2$.

Lambda Phage Example

- Lambda phage is virus that hijacks host cell
- Viral reproduction process shown below is called the Lysogenic Cycle.

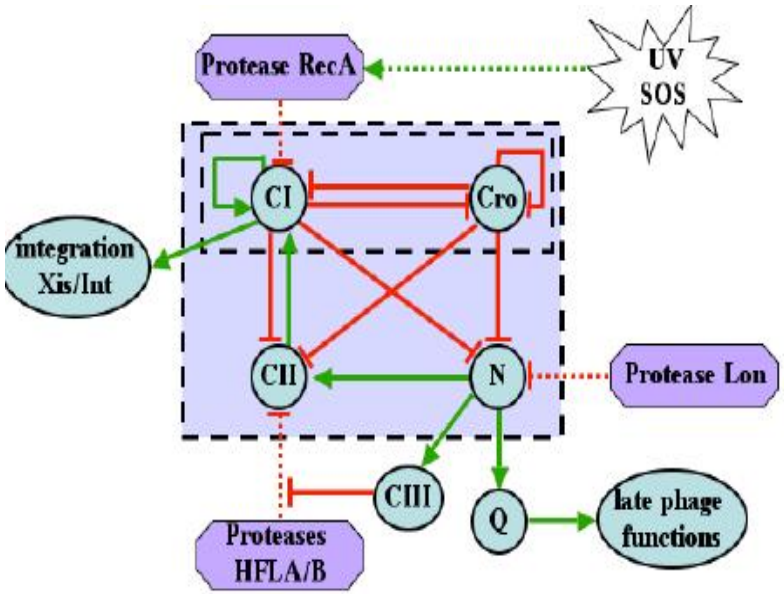
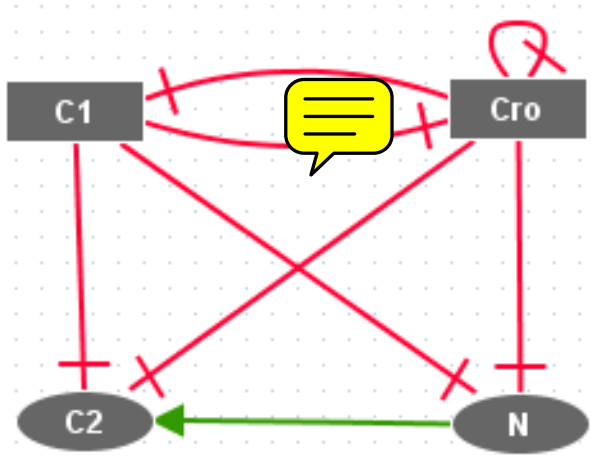


Figure: Logical model of lysogenic cycle by Lambda Phage virus

- Red arrows signify inhibition.
- Green arrows signify activation.
- Ex: CII activates CI; CI maintains its own expression while repressing CII, Cro, and N.

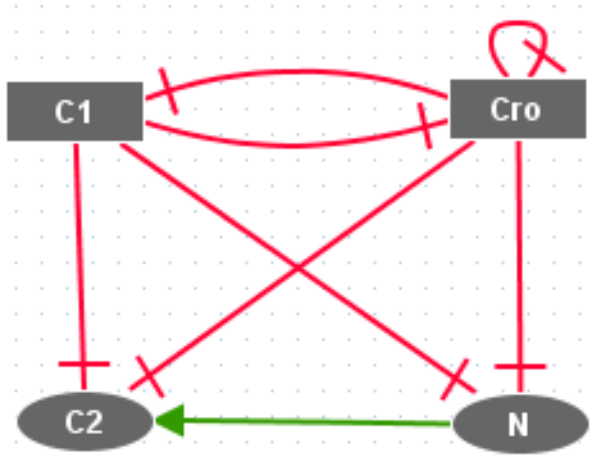
Lambda Phage Example

- State written as vector with four entries: (CI,CII,Cro,N)
- Genes have 5 possible values; hence $5^4 = 625$ states.
- State (1,0,0,3) means concentration of CI is low, genes CII and Cro are off, and N is medium.
- f_i determines state of the gene represented by x_i.
- Ex: $f_{C1} = -2x_{Cro}^4 + 2$ in means x_{C1} is 0 or 2 based on x_{Cro}.



Lambda Phage Example

- State written as vector with four entries: (CI,CII,Cro,N)
- Genes have 5 possible values; hence $5^4 = 625$ states.
- State (1,0,0,3) means concentration of CI is low, genes CII and Cro are off, and N is medium.
- f_i determines state of the gene represented by x_i.
- Ex: $f_{C1} = -2x_{Cro}^4 + 2$ in means x_{C1} is 0 or 2 based on x_{Cro}.



x_{Cro}	f_{C1}
0	2
1	0
2	0
3	0
4	0

Lambda Phage Example

- Lambda phage model can be uploaded to ADAM for analysis.
- ADAM will output PDS from truth table with corresponding variable descriptions.
- ADAM will also output analysis of dynamics.

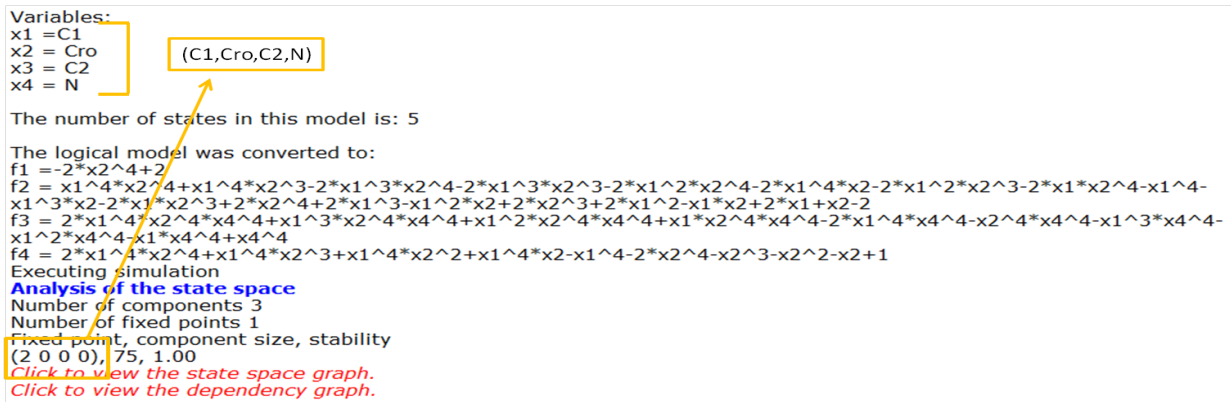
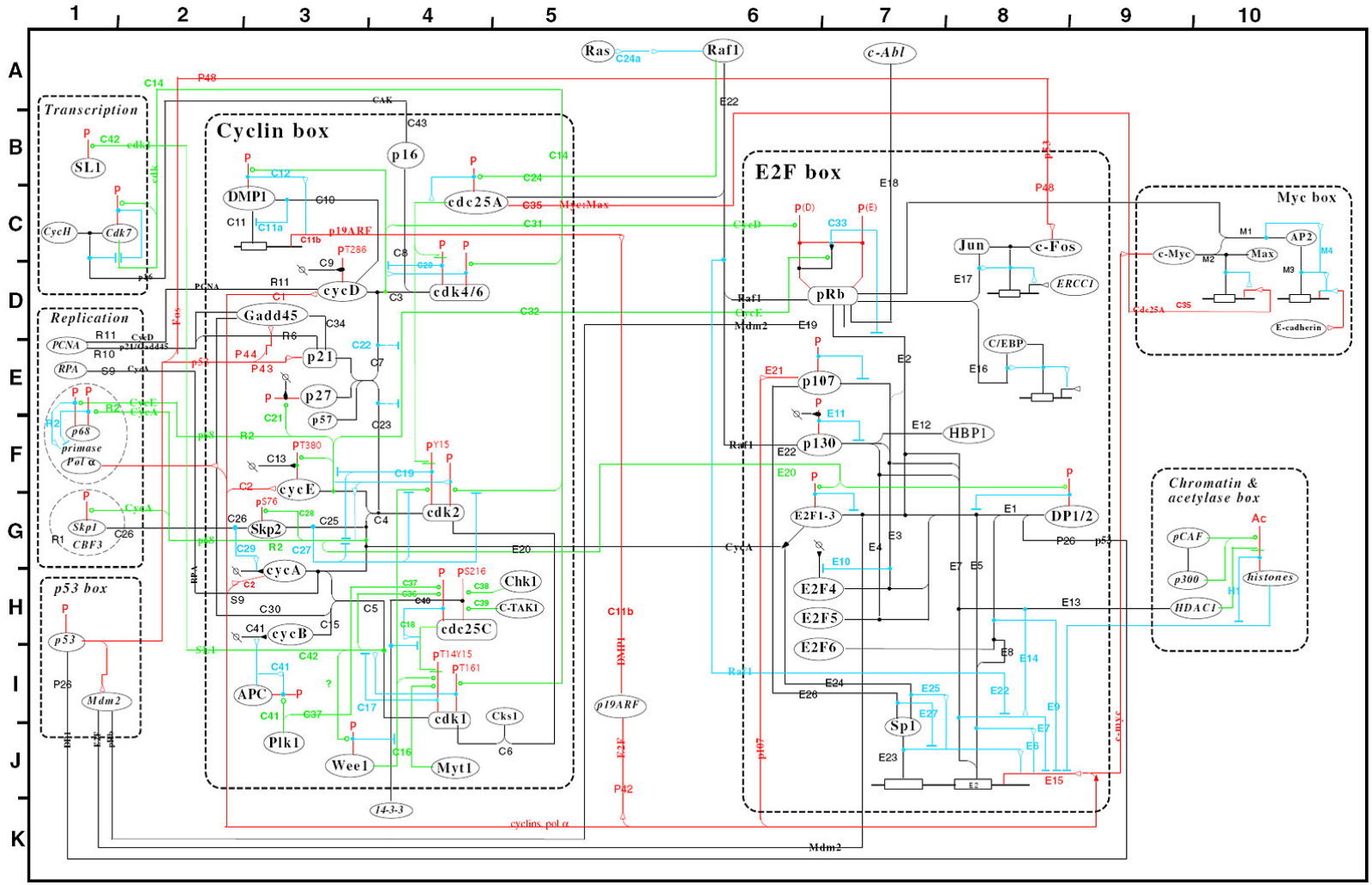


Figure: Output from ADAM for analysis of lambda phage

- One fixed point at (2,0,0,0)
- Two limit cycles

What Happens as the System Gets Larger?

Figure: Mammalian cell cycle with at least 60 nodes.



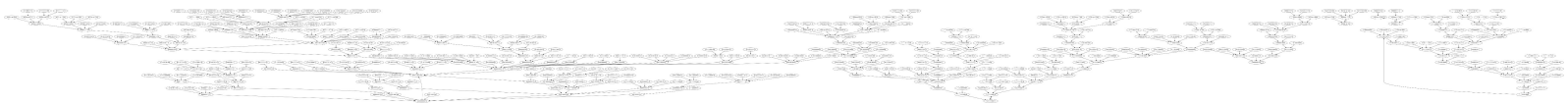
State Space Diagrams

- $2^{60} \approx 1,200,000,000,000,000$ states. Let's scale down to $n = 9$ nodes. (ADAM only allows simulation up to 11 nodes)
- Even for $n = 9$, when we try to see the state space graph...

State Space Diagrams

- $2^{60} \approx 1,200,000,000,000,000$ states. Let's scale down to $n = 9$ nodes. (ADAM only allows simulation up to 11 nodes)
- Even for $n = 9$, when we try to see the state space graph...

Figure: Boolean state space graph, 9 nodes.

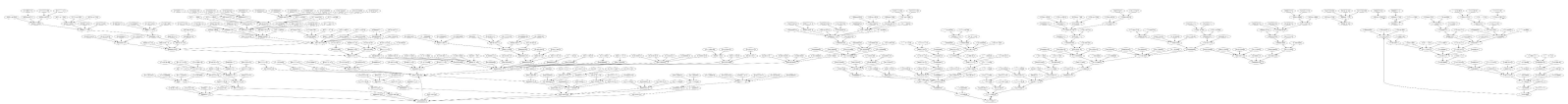


Impossible to even see the states on one screen!

State Space Diagrams

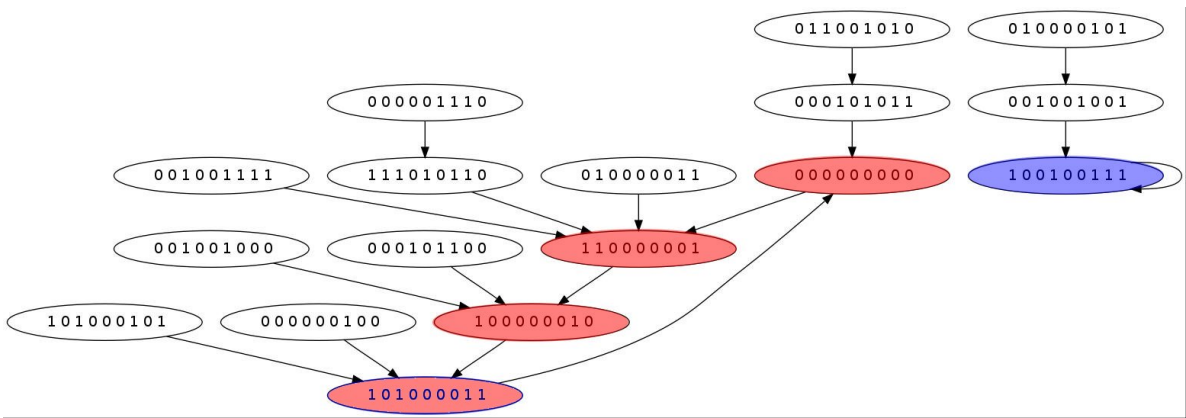
- $2^{60} \approx 1,200,000,000,000,000$ states. Let's scale down to $n = 9$ nodes. (ADAM only allows simulation up to 11 nodes)
- Even for $n = 9$, when we try to see the state space graph...

Figure: Boolean state space graph, 9 nodes.



Impossible to even see the states on one screen!

Figure: Cropping of 3% of graph



Why simulation isn't enough

How many states can a biological system have?

Network controlling ErBb2 regulation: 17 nodes, off, low or high

- $3^{17} = 129,140,163$

Mamillian Cell Cycle: 60 nodes, on or off

- $2^{60} = 1,152,921,504,606,846,976$

Budding Yeast Cell Cycle: 27 nodes, off, low or high

- $3^{27} = 7,625,597,484,987$

Searching for an Alternative to Simulation

- In biological systems, input generally only comes from a few nodes.
- In gene regulatory networks genes are regulated by only a handful of regulators.
- Hence PDSs representing such biological networks are sparse.
- We compute the Gröbner basis of the PDS to simplify analysis.
- Computations are fast because sparse structure is preserved by Gröbner basis.

Benchmark Test Results

Based on benchmark tests on algorithms which compute fixed points, showed that algorithms particularly fast on large sparse systems. All benchmark tests run on a 2.27 GHz processor.

Time(min)	# of tests	Nodes	Nodes per term	# of terms	*testing*
Under 0.05	44	10-20	10-15	5-9	*High # of nodes per term*
Under 1	5	"	"	"	
Under 7	1	"	"	"	
Under 0.5	8	20-30	10-15	5-9	
Under 0.5	38	10-20	2-6	100-1,000,000	*High # of terms*
Under 6	5	"	"	"	
Under 10	1	"	"	"	
Over 20	3	"	"	"	
Under 0.05	50	10-20	2-6	1	*Monomial (Conjunctive) Networks*
"	10	20-30	"	"	
"	5	30-40	"	"	
Under 0.5	5	40-50	"	"	
Under 1	3	50-100	2-6	1-5	*High # of nodes*
Under 0.5	50	50-100	2	6-10	
Under 0.5	7	40-50	5	1-8	
Under 1	1	"	"	"	
10	1	"	"	"	
Over 10	2	"	"	"	

Figure: Benchmark tests run on randomly generated Boolean functions with different ranges of variables, terms per function, and maximum of variables per term.

91 % of computations completed in less than 30 seconds.

GINsim Benchmark Tests

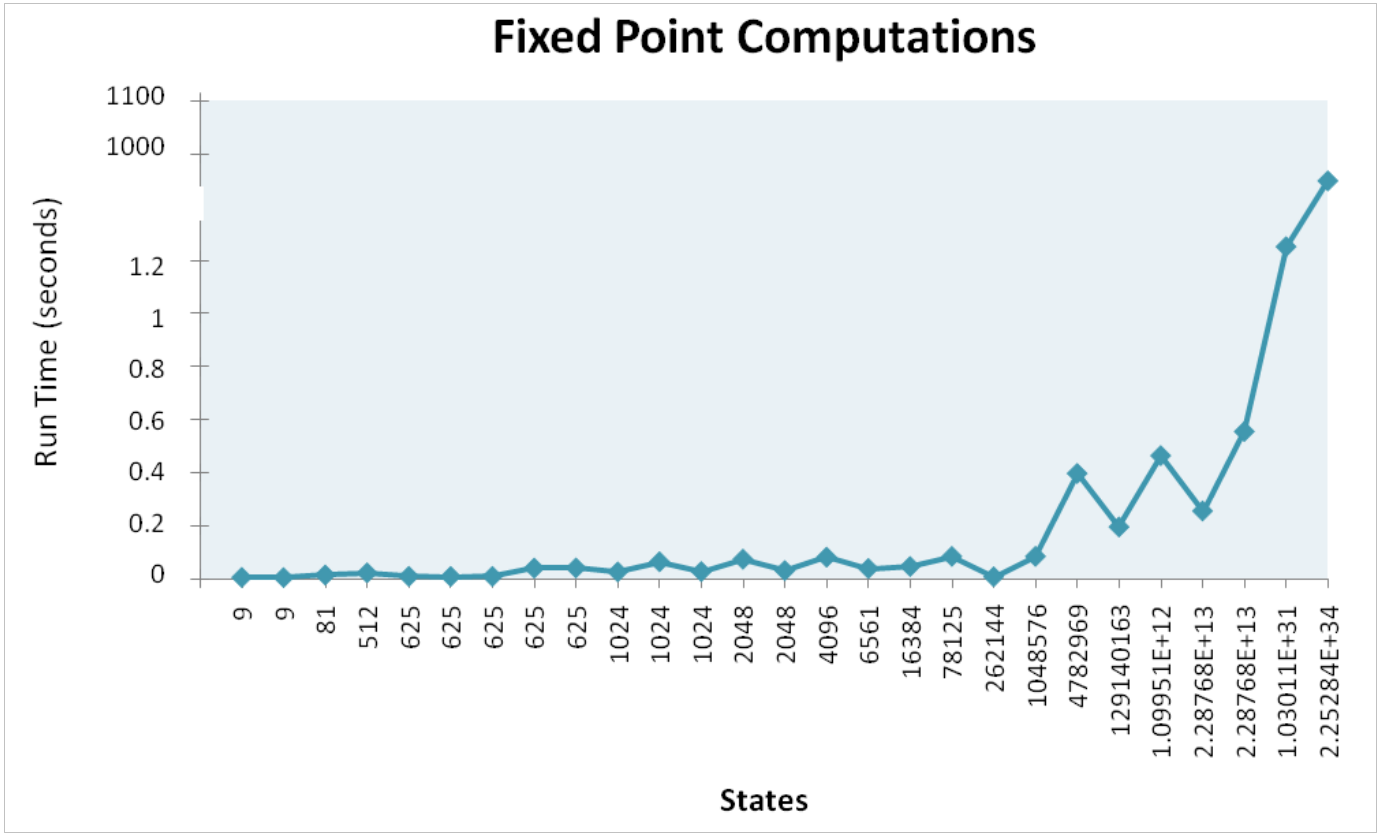
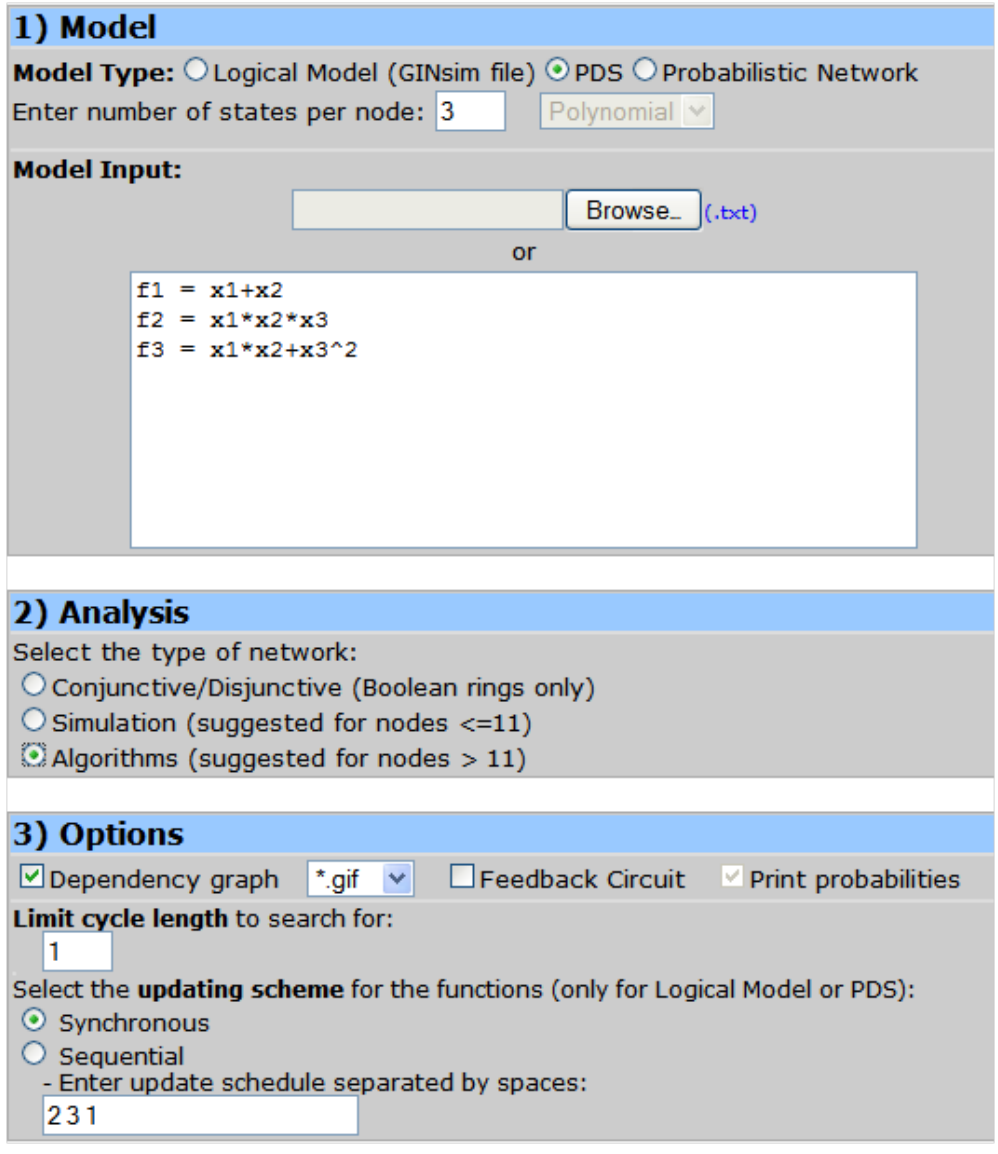


Figure: Plot representing run times for fixed point calculations done on 25 logical models. The number of nodes in each file ranges from 2 to 72.

ADAM: Analysis of Discrete Algebraic Models

- Analogy: MatLAB - solves continuous models such as ODEs - does not require understanding of ODE solvers.
- ADAM analyzes discrete models by a combination of simulation and algorithms - does not require understanding of underlying mathematics.
- ADAM can analyze:
  - **Logical Models**(in GINSim format)
  - **Polynomial Dynamical Systems (PDS)**
  - **Probabilistic Boolean (or multistate) Networks**
- <http://dvd.vbi.vt.edu/cgi-bin/git/adam.pl>

Meet ADAM: <http://dvd.vbi.vt.edu/cgi-bin/git/adam.pl>



TCR Signalization Pathway

- Boolean logical model with 40 nodes
  $\rightarrow 2^{40} = 1,099,511,627,776$ states
- For a state space this large simulation is inefficient

Using ADAM:

- fixed point analysis $<$ .5 seconds
- analyzed limit cycles up to length 20 with a total runtime $<$ 1 minute
- finds a limit cycle which was not found in the published analysis

Running analysis now ...
There are 1 limit cycles of length 7 and they are:

1110011010001000000000000000010100001000,	1110011110011000000000000000010100001000,
1110111011001000000000000000011100001000,	1111001110010100000000000000000000001000,
1111001111111010000000000000000000001000,	1111101011100000000000000000001000001000,
1111111011100000000000000000001000001000	

Conclusions

- Discrete Modeling techniques useful tool for **analyzing biological systems**
- Can analyze discrete models, i.e. logical networks, petri-nets, or agent-based models, by **converting into polynomial dynamical systems (PDS)**
- Once these models have an algebraic structure, use **tools from computational algebra to compute key dynamics**.
- Algorithms we developed are **fast for sparse systems**, a structure maintained by most biological systems
- All algorithms included in **user-friendly and free software package ADAM**

Future Developments

ADAM can be extended to...

- Include better visualization for larger networks
- Incorporate into other software packages like Polynome or GINsim
- Automatic conversion for other model types such as petri nets
- Implement analytic methods/algorithms to reduce computational complexity and improve efficiency

Acknowledgements

This research was conducted during the Research Experience for Undergraduates (REU), Modeling and Simulation in Systems Biology (MSSB) at Virginia Bioinformatics Institute (VBI), Virginia Tech University. This research was funded by the National Science Foundation, fund number 0755322.