Symbolic Systems Biology and Pathway Logic

Carolyn Talcott
SRI International
September 2008

PLAN

- Symbolic systems biology
- Executable Specification in RWL
- Pathway Logic
- Computing with PL models
  - Egf Stimulation
  - A sleep model

Symbolic Systems Biology

Biological Systems

- Biological processes are complex
  - genes, proteins, metabolites
  - cells, organs, organisms
- Dynamics that range over huge timescales
  - microseconds to years
- Spatial scales over 12 orders of magnitude
  - single protein to cell, cell to whole organism
- Oceans of experimental biological data generated
- Important intuitions captured in mental models that biologists build of biological processes

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
- Which biology? Causal networks of biomolecular interactions and reactions
- Goals:
  - Develop formal models that are as close as possible to domain expert's mental models
  - Compute with, analyze and reason about these complex networks
  - New insights into / understanding of biological mechanisms

Logical Framework

- Making description and reasoning precise
- Language
  - for describing things and/or properties
  - given by a signature -- symbols to be used, and rules for generating expressions (terms, formulas)
- Semantic model -- mathematical structure (meaning)
  - interpretation of terms
  - satisfaction of formulas: M |= wff
- Reasoning rules
  - for inferring valid formulae
  - for calculating, simulating
- Symbolic model -- theory (axioms) used to answer questions

Executable Symbolic Models

- Describe system states and rules for change
- From an initial state, derive a transition graph
  - nodes -- reachable states
  - edges -- rules connecting states
- Path -- sequence of nodes and edges in transition graph (computation / derivation)
- Execution strategy -- picks a path

A Sampling of Formalisms

- Rule-based + Temporal logics
- Petri nets + Temporal logics
- Membrane calculi -- spatial process calculi / logics
- Statecharts + Live sequence charts
- Stochastic transitions systems and logics
- Hybrid Automata + Abstraction

Executable Specification in Rewriting Logic (Maude) A Petri Net Example

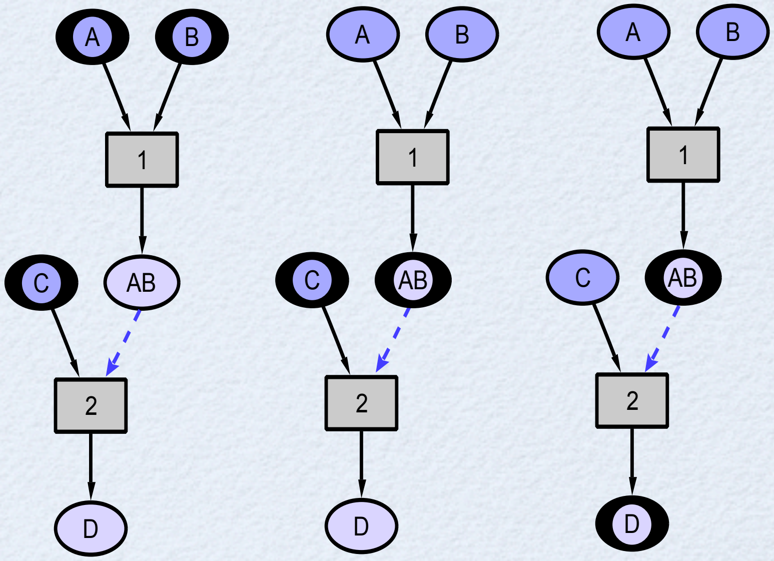
About Rewriting Logic

- 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!)

About Petri Nets

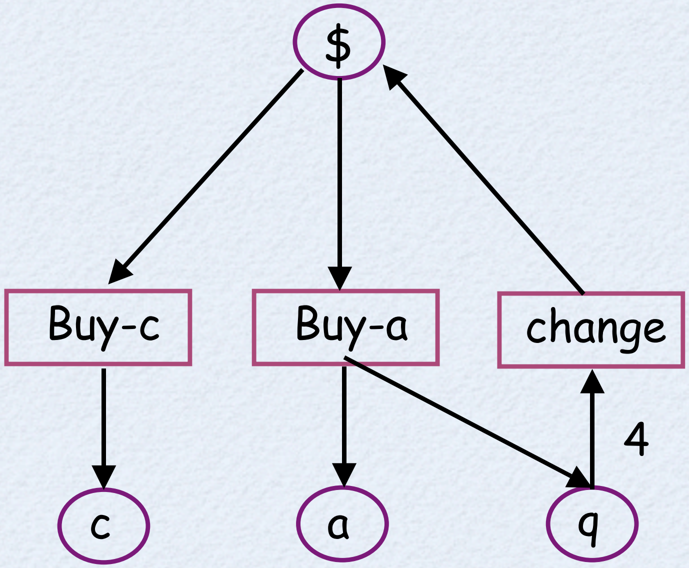
A Petri net is represented as a graph with two kinds of nodes:
 * transitions/rules (reactions--squares)
 * places/occurrences (reactants, products, modifiers--ovals)

A Petri net process has <u>tokens</u> on some of its places. A rule can fire if all of its inputs have tokens. Firing a rule moves tokens from input to output.



An execution is a sequence of rule firings.
A pathway is represented as an execution subgraph.

Petri Net Model of a VendingMachine



```
mod VENDING-MACHINE is
  sorts Coin Item Place Marking .
  subsorts Coin Item < Place < Marking .
  op null : -> Marking .
               *** empty marking
  ops $ q : -> Coin .
  ops a c : -> Item .
  op _ _ : Marking Marking -> Marking
           [assoc comm id: null] .
           *** multiset
  rl[buy-c]: $ => c .
  rl[buy-a]: $ => a q .
  rl[change]: q q q q => $ .
endm
```

Using the Vending Machine

Execution: What is one way to use 3 $s?

```
Maude> rew $ $ $ .
result Marking: q a c c
```

Search: How can I get 2 apples with 3 $s?

```
Maude> search $ $ $ =>! a a M:Marking .

Solution 1 (state 8)
M:Marking --> q q c

Solution 2 (state 9)
M:Marking --> q q q a

No more solutions.
states: 10  rewrites: 12)
```

Model Checking I

- Algorithm for determining if M |= P (M satisfies P) where M is a `model' and `P' is a property.

- In our case a model is a Maude specification of a system together with a staAlgorithm for determining if M |= P (M satisfies P) where M is a `model' and `P' is a property.

- In our case a model is a Maude specification of a system together with a state of interest.

Model Checking the Vending Machine 1

Starting with 5 $s, can we get 6 apples without accumulating more than 4 quarters? Model check the claim that we can't.

```
Maude>
   red modelCheck(vm($ $ $ $ $),[]~(lte4Q U nApples(6))) .
result ModelCheckResult: counterexample(
  {vm($ $ $ $ $),'buy-a}
  {vm($ $ $ $ q a),'buy-a}
  {vm($ $ $ q q a a),'buy-a}
  {vm($ $ q q q a a a),'buy-a}
  {vm($ q q q q a a a a),'change}
  {vm($ $ a a a a),'buy-a}
  {vm($ q a a a a a), 'buy-a},
  {vm(q q a a a a a a),deadlock})
```

A counterexample to a formula is a pair of transition lists representing an infinite compuation which fails to satisfy the formula. A transition is a state and a rule identifier. The second list (deadlock) represents a loop.

Symbolic Analysis I

- 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

Symbolic Analysis III

- 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.

Symbolic Analysis V

- Meta analysis -- reasoning about the model itself
  - find transitions producing / consuming X
  - find all phosphorylation reactions
  - check that transitions satisfy some property such as stoichiometry
  - transform a model and property to another logic (for access to tools)

Pathway Logic (PL) Representation of Signaling

http://pl.csl.sri.com/

Signaling Pathways

- 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.

About Pathway Logic

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 down stream of a given rule or component

Pathway Logic Organization

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

rasNet

Early steps in Egf signaling

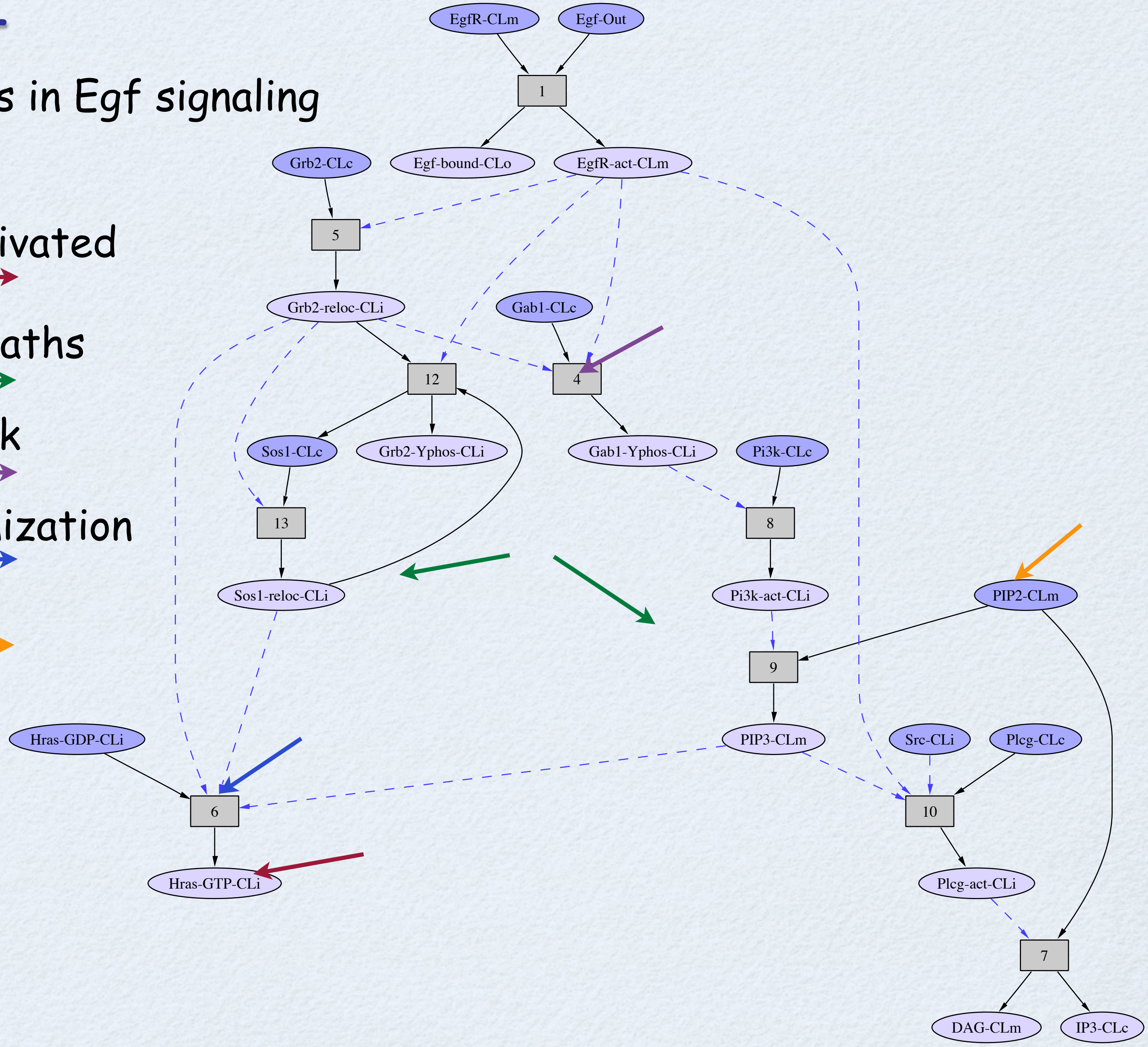
Hras activated →

Parallel paths →

Cross talk →

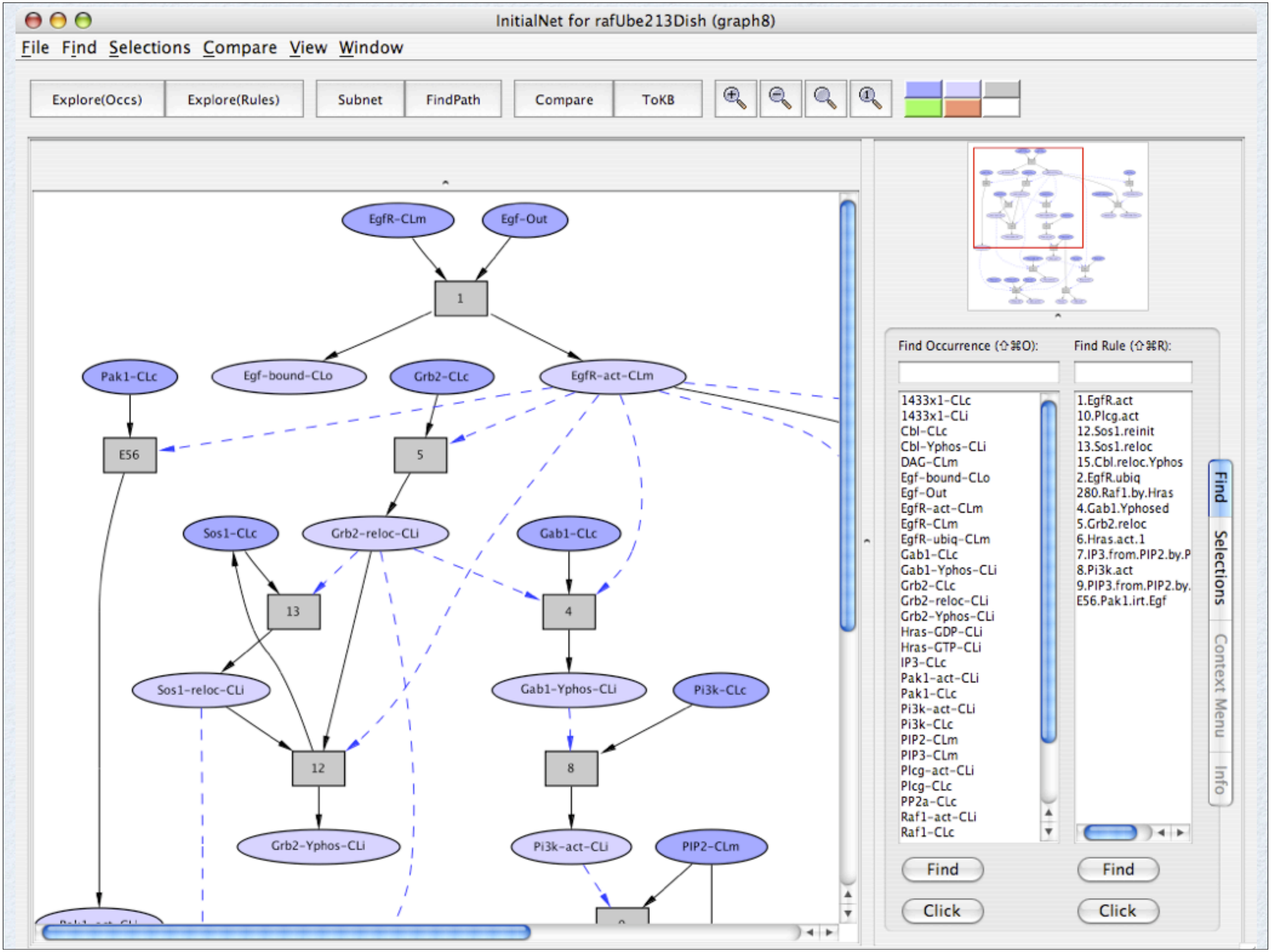
Synchronization →

Conflict →



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)
  - SBML (xml based standard for model exchange)



Find Occurrence (⇧⌘O):

1433x1-CLc
1433x1-CLi
Cbl-CLc
Cbl-Yphos-CLi
DAG-CLm
Egf-bound-CLo
Egf-Out
EgfR-act-CLm
EgfR-CLm
EgfR-ubiq-CLm
Gab1-CLc
Gab1-Yphos-CLi
Grb2-CLc
Grb2-reloc-CLi
Grb2-Yphos-CLi
Hras-GDP-CLi
Hras-GTP-CLi
IP3-CLc
Pak1-act-CLi
Pak1-CLc
Pi3k-act-CLi
Pi3k-CLc
PIP2-CLm
PIP3-CLm
Plcg-act-CLi
Plcg-CLc
PP2a-CLc
Raf1-act-CLi
Raf1-CLc

Find Rule (⇧⌘R):

1.EgfR.act
10.Plcg.act
12.Sos1.reinit
13.Sos1.reloc
15.Cbl.reloc.Yphos
2.EgfR.ubiq
280.Raf1.by.Hras
4.Gab1.Yphosed
5.Grb2.reloc
6.Hras.act.1
7.IP3.from.PIP2.by.P
8.Pi3k.act
9.PIP3.from.PIP2.by.
E56.Pak1.irt.Egf

A Simple Query Language

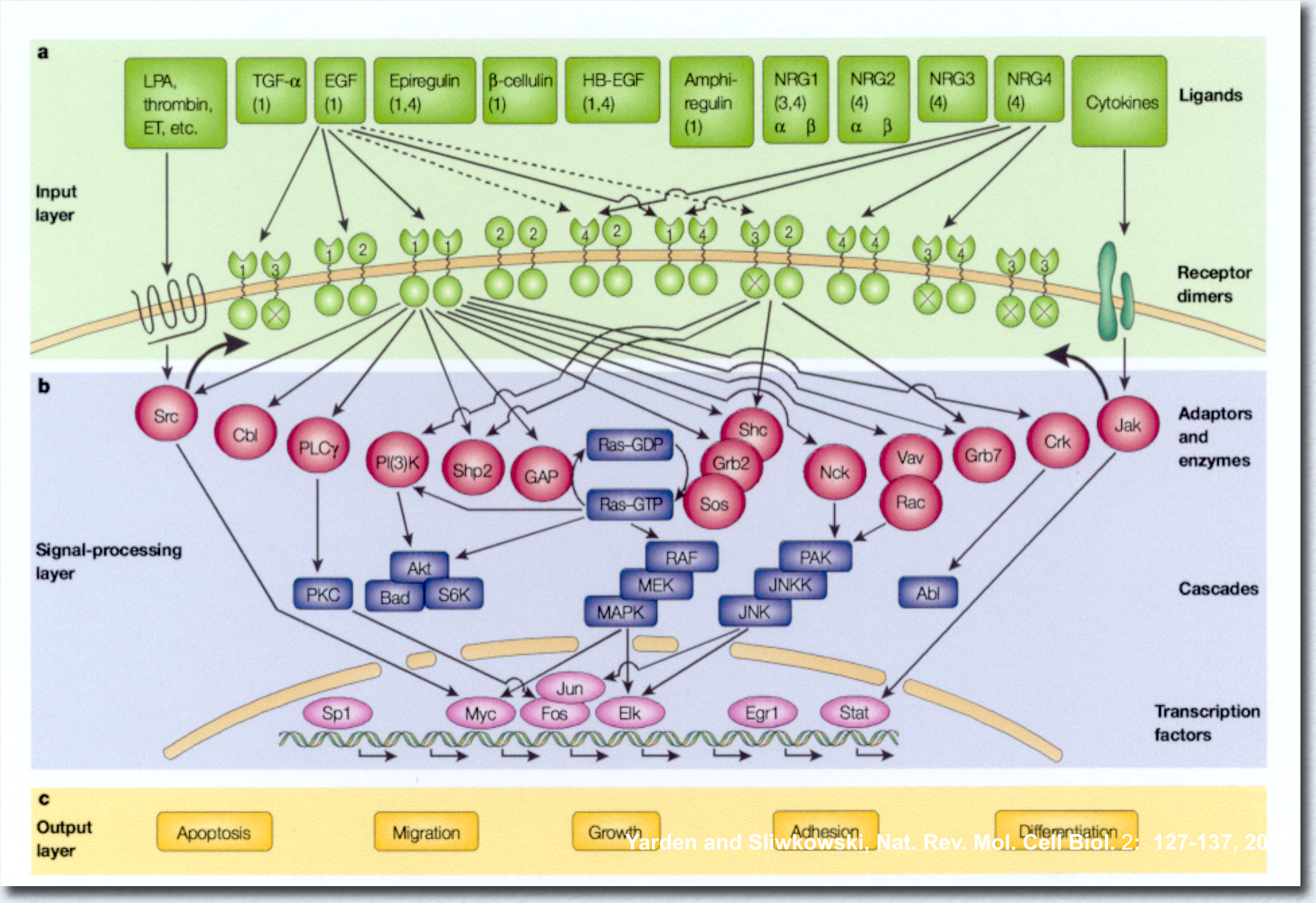
- Given a Petri net with transitions P and initial marking O (for occurrences) there are two types of query
  - subnet
  - findPath - a computation / unfolding
- For each type there are three parameters
  - G: a goal set---occurrences required to be present at the end of a path
  - A: an avoid set---occurrences that must not appear in any transition fired
  - H: as list of identifiers of transitions that must not be fired

findPath returns a pathway (transition list) generating a computation satisfying the requiremments -- uses model-checking.

subnet returns a subnet containing all (minimal) such pathways -- uses backward and forward simulation .

FULL MODEL
OF
EGF STIMULATION

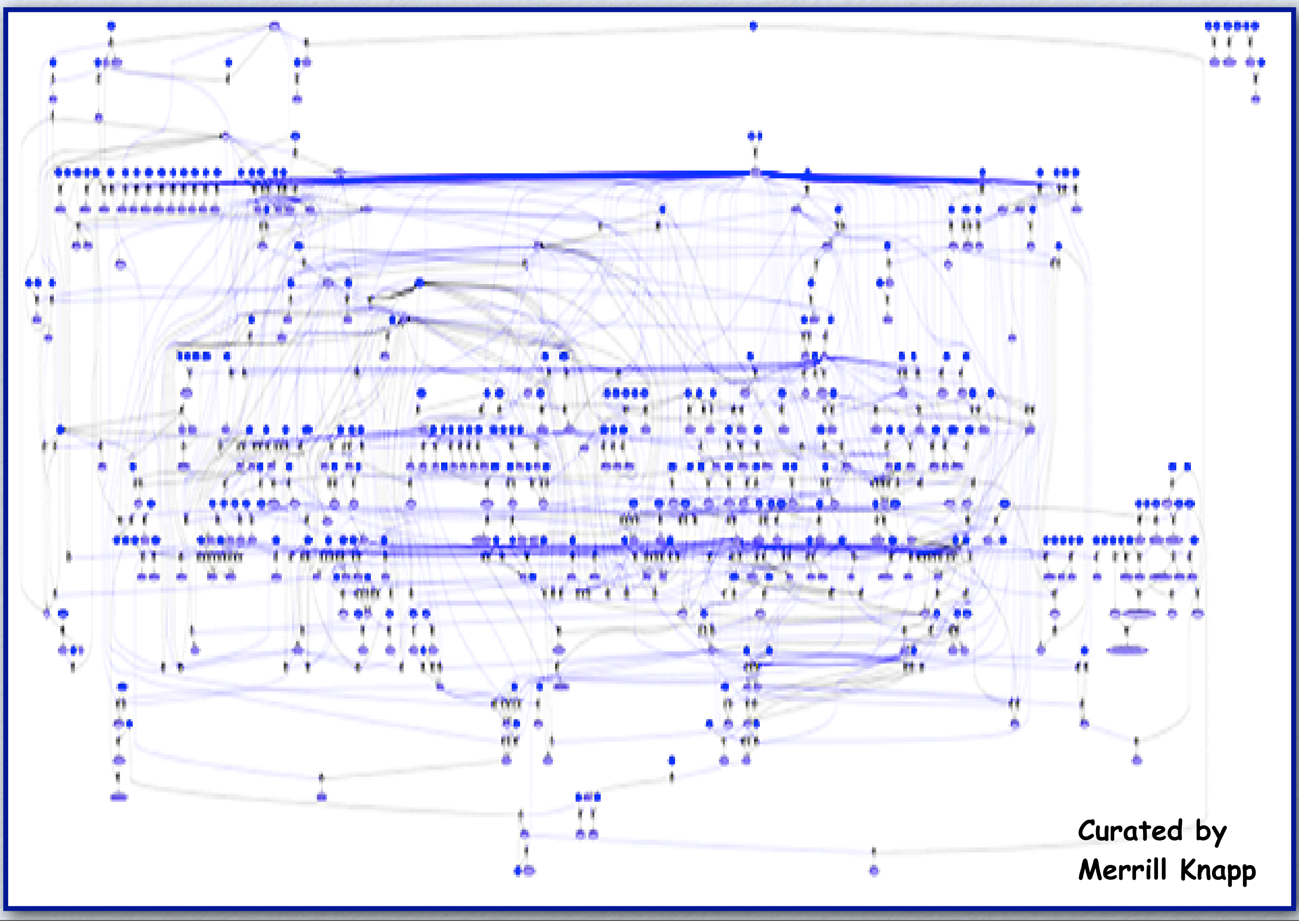
The ErbB Network (Cartoon Form)



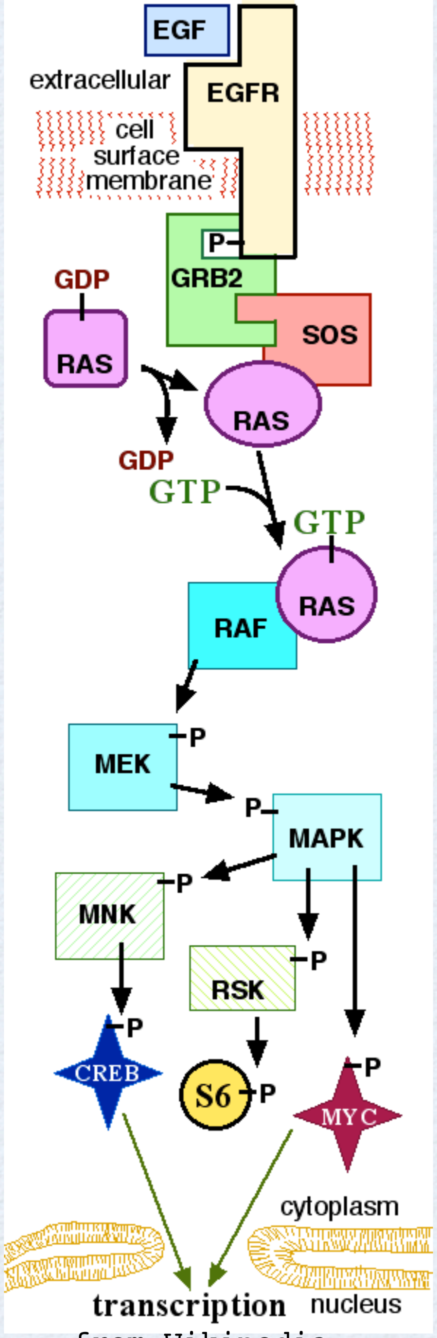
Yarden and Sliwkowski, Nat. Rev. Mol. Cell Biol. 2: 127-137, 20

PL EGF MODEL

Events that could occur in response to Egf



Curated by
Merrill Knapp



from Wikipedia

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

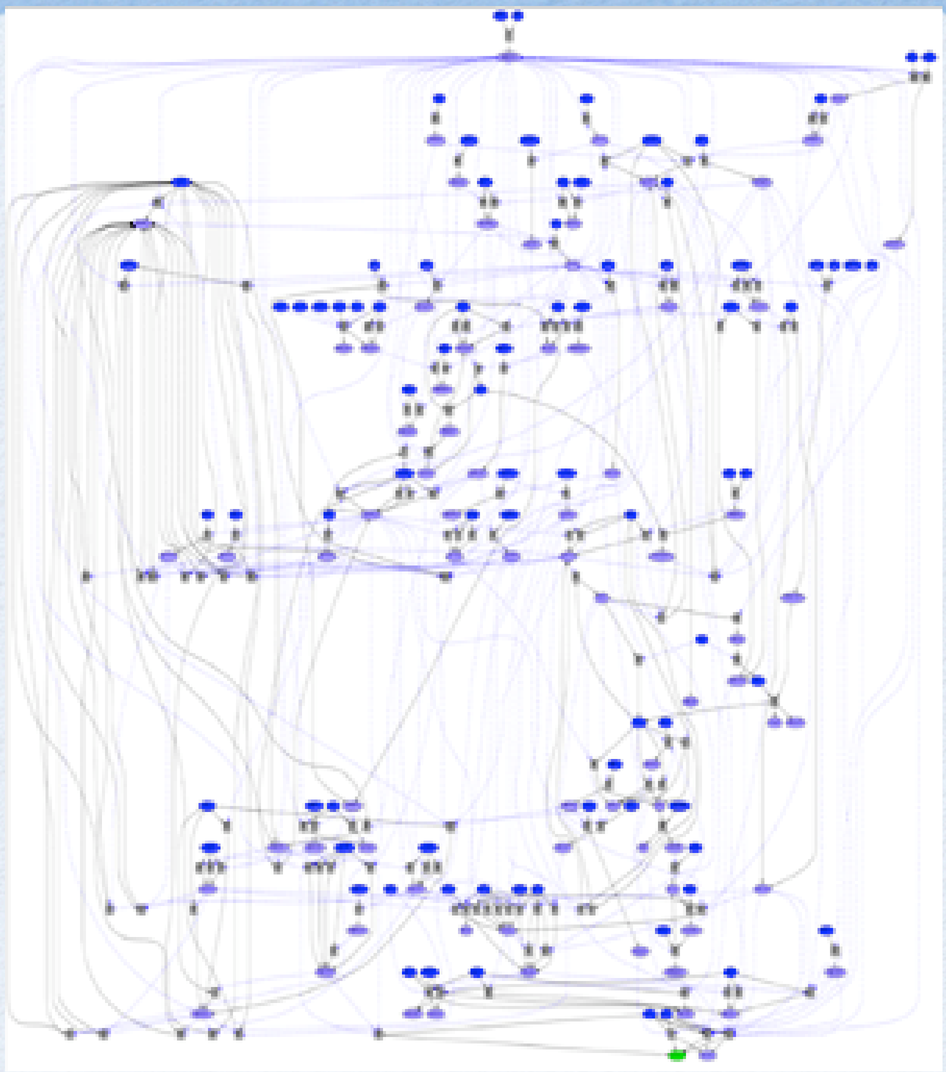
Egf → EgfR → Grb2 → Sos1 → Ras → Raf1 → Mek → Erk

- 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)
- ...

Subnet relevant to Erk activation

Subnet containing all pathways leading to activation of Erk.

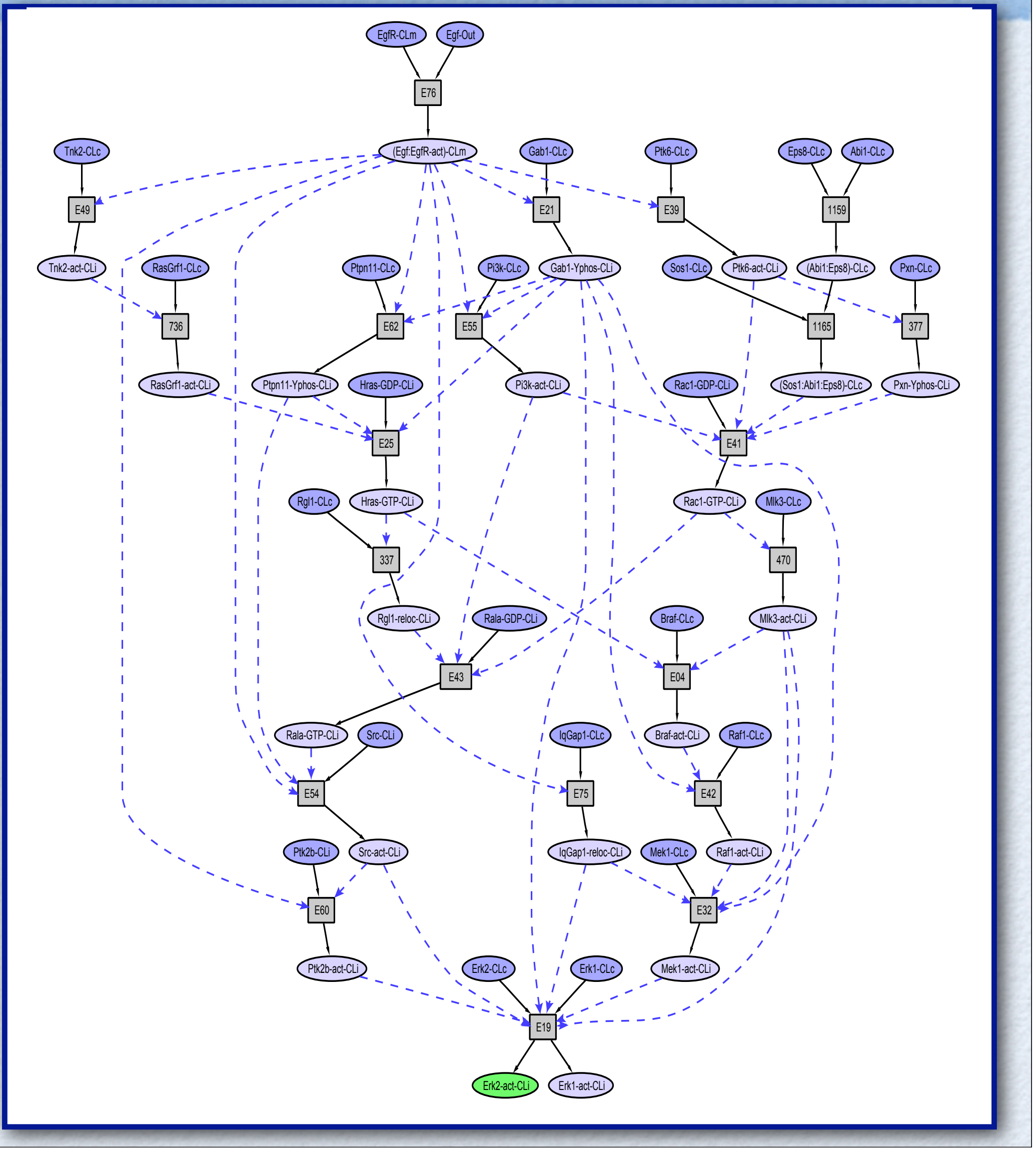
Obtained by backwards followed by forwards collection



Activation of Erk irt Egf

The path leading to activation of Erk in a network constrained to meet all the recorded events happening in response to Egf.

This path exists in the context of all the other experimental observations,



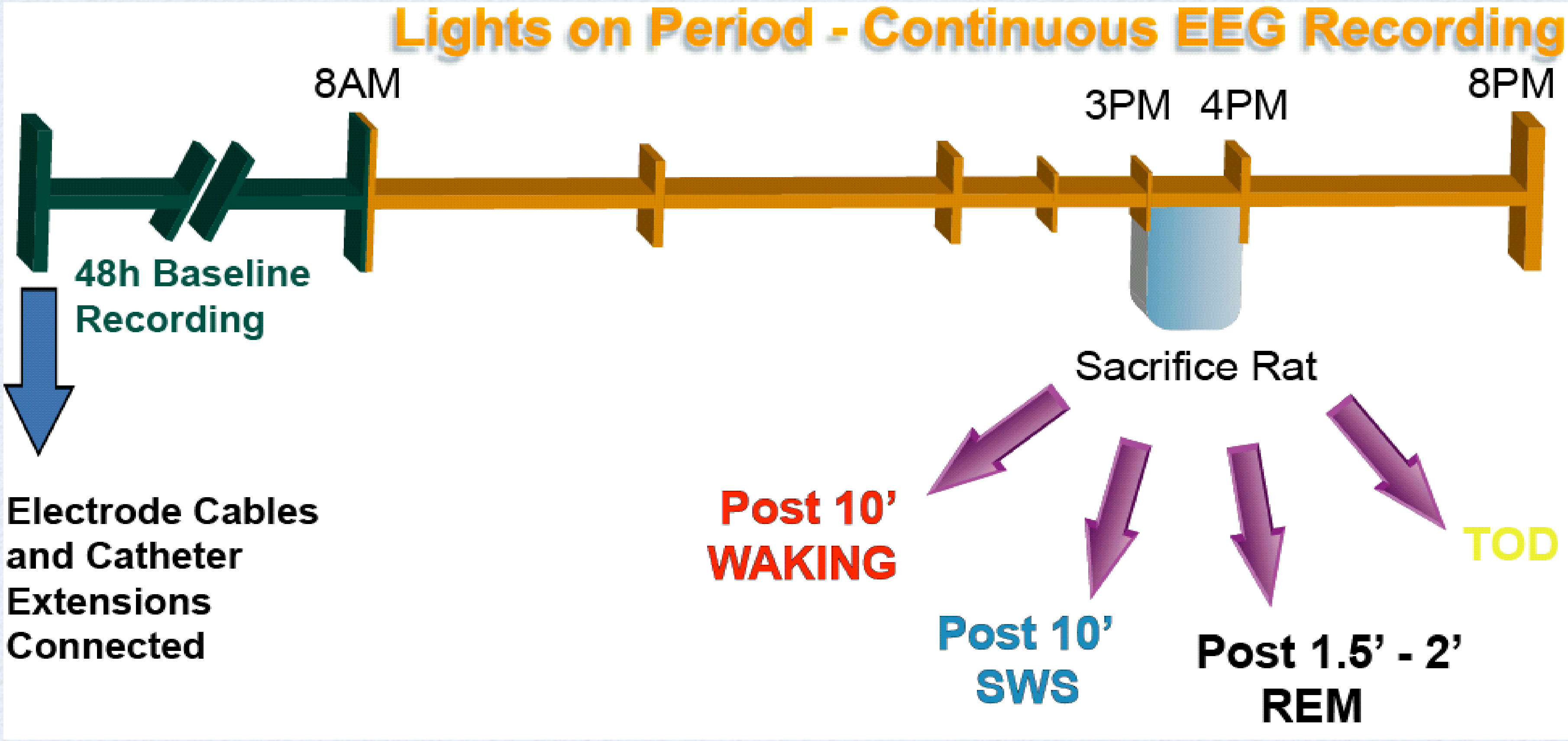
A Small Sleep Model

Biological Timing and Sleep

- Why sleep?
  - An essential behavior
  - Regulated by circadian and homeostatic influences
  - Accounts for about 1/3 of lifetime
    - the function is unknown

- Questions
  - What are your organs doing when you sleep?
  - When you are awake?
  - What is common across organs?
  - What is unique to an organ?

Natural Sleep Paradigm



Lights on Period - Continuous EEG Recording

8AM

3PM 4PM

8PM

48h Baseline Recording

Electrode Cables and Catheter Extensions Connected

Sacrifice Rat

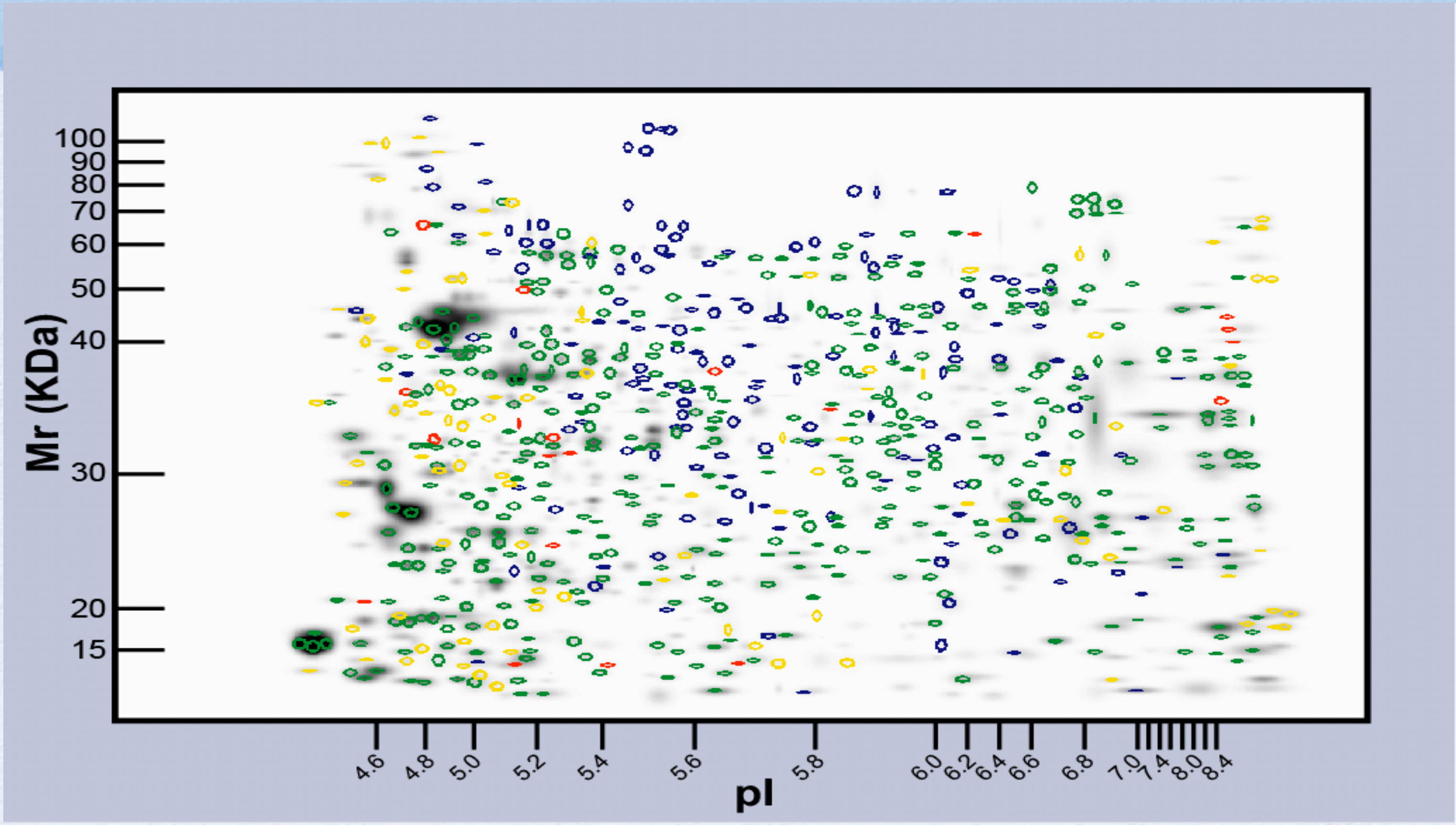
Post 10' WAKING

Post 10' SWS

Post 1.5' - 2' REM

TOD

2D Master Gel -- Frontal Cortex

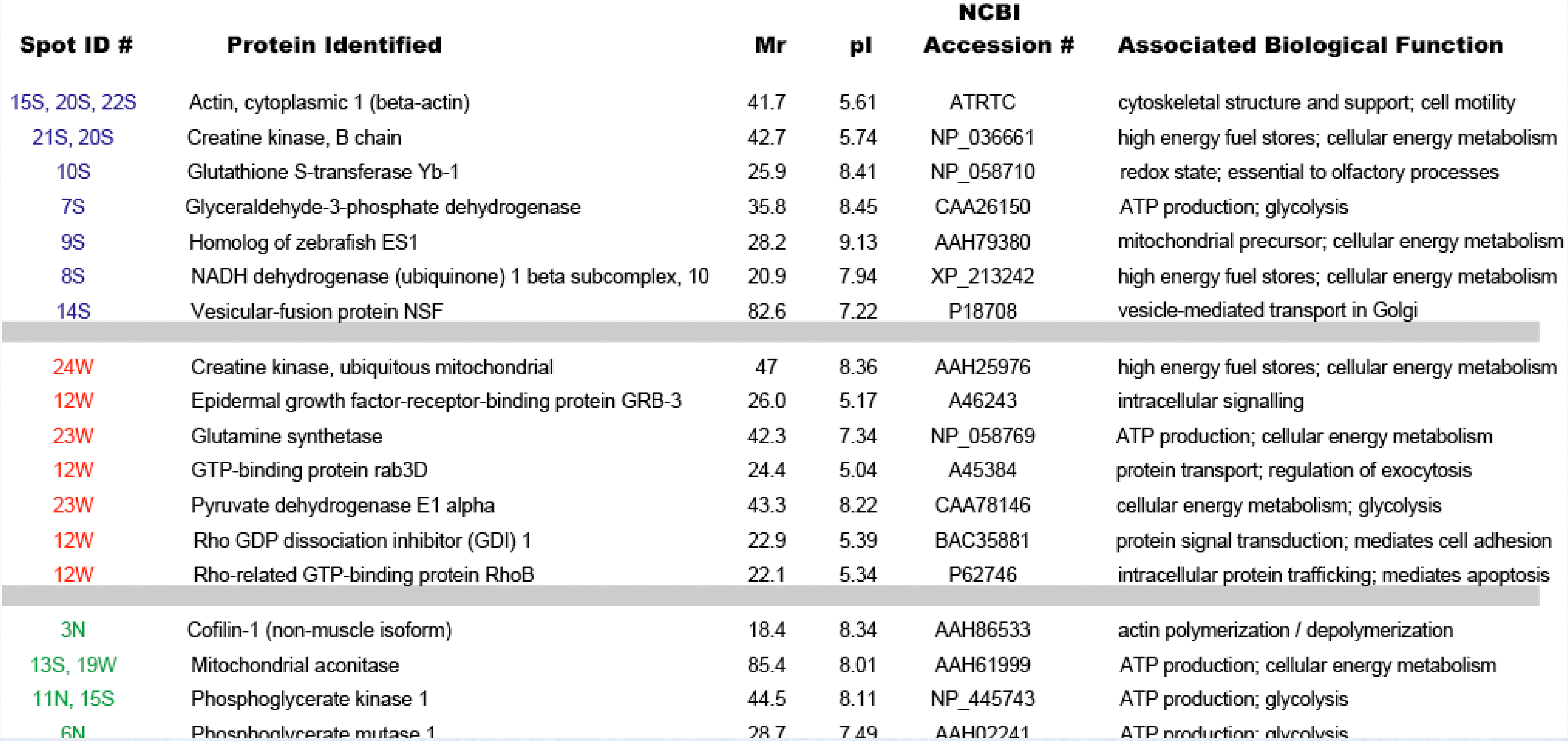


Proteins unique to different states were identified

Those modeled in PL included Actin and Rhob

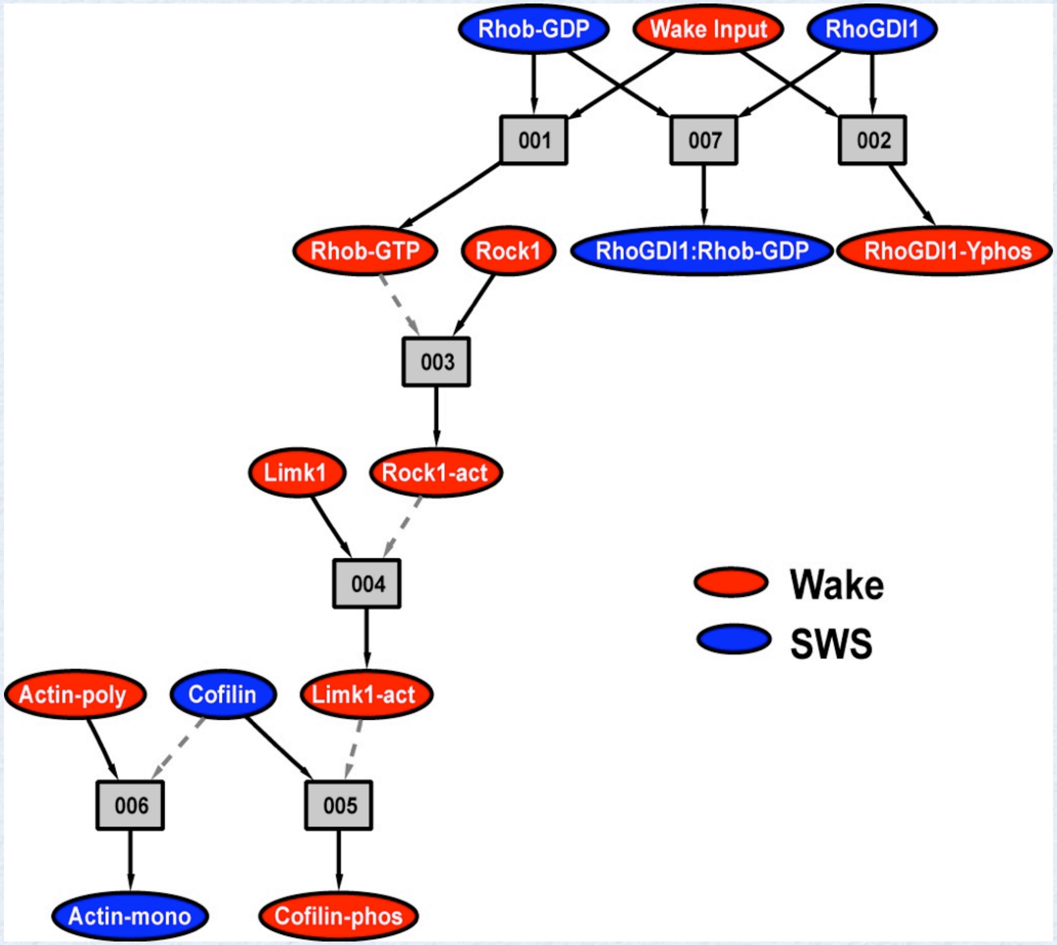
Use the PLA explorer to find signaling connections

Spots Identified by MS



Spot ID #	Protein Identified	Mr	pI	NCBI Accession #	Associated Biological Function
15S, 20S, 22S	Actin, cytoplasmic 1 (beta-actin)	41.7	5.61	ATRTC	cytoskeletal structure and support; cell motility
21S, 20S	Creatine kinase, B chain	42.7	5.74	NP_036661	high energy fuel stores; cellular energy metabolism
10S	Glutathione S-transferase Yb-1	25.9	8.41	NP_058710	redox state; essential to olfactory processes
7S	Glyceraldehyde-3-phosphate dehydrogenase	35.8	8.45	CAA26150	ATP production; glycolysis
9S	Homolog of zebrafish ES1	28.2	9.13	AAH79380	mitochondrial precursor; cellular energy metabolism
8S	NADH dehydrogenase (ubiquinone) 1 beta subcomplex, 10	20.9	7.94	XP_213242	high energy fuel stores; cellular energy metabolism
14S	Vesicular-fusion protein NSF	82.6	7.22	P18708	vesicle-mediated transport in Golgi
24W	Creatine kinase, ubiquitous mitochondrial	47	8.36	AAH25976	high energy fuel stores; cellular energy metabolism
12W	Epidermal growth factor-receptor-binding protein GRB-3	26.0	5.17	A46243	intracellular signalling
23W	Glutamine synthetase	42.3	7.34	NP_058769	ATP production; cellular energy metabolism
12W	GTP-binding protein rab3D	24.4	5.04	A45384	protein transport; regulation of exocytosis
23W	Pyruvate dehydrogenase E1 alpha	43.3	8.22	CAA78146	cellular energy metabolism; glycolysis
12W	Rho GDP dissociation inhibitor (GDI) 1	22.9	5.39	BAC35881	protein signal transduction; mediates cell adhesion
12W	Rho-related GTP-binding protein RhoB	22.1	5.34	P62746	intracellular protein trafficking; mediates apoptosis
3N	Cofilin-1 (non-muscle isoform)	18.4	8.34	AAH86533	actin polymerization / depolymerization
13S, 19W	Mitochondrial aconitase	85.4	8.01	AAH61999	ATP production; cellular energy metabolism
11N, 15S	Phosphoglycerate kinase 1	44.5	8.11	NP_445743	ATP production; glycolysis
6N	Phosphoglycerate mutase 1	28.7	7.49	AAH02241	ATP production; glycolysis

A hypothetical model pathway relating state and synaptic plasticity



Wake state:
unknown signal(s)
=> phosphorylation of Rock1
=> activation of Limk1
=> phosphorylation of cofilin
=> increase in polymerized actin
(Phosphorylated cofilin is unable
to depolymerize actin)

SWS:
RhoDG11 binds Rhob-GDP
(is not phosphorylated)
=> Rock1, Limk1, and cofillin would
not be phosphorylated and
=> actin depolymerization
=> decrease in synaptic weight

Project Ideas -- I

- Pnet query
  - find rules using/modifying X,
  - find rules of specified type
    - translocation, phosphorylation
- Pnet Structure
  - t invariants, finding modules
  - Sachman 2006 paper

Project Ideas -- II

- Pnet analysis -- given a set of rules
    - find (minimal) initial state(s) reaching a specified goal
        - (proof will be a pathway)
    - find a minimal initial state so all rules are reachable
    - Find all pathways for a given goal (probably quite difficult)
    - Extend PLA to do probabilistic simulation and statistical model checking
- The idea is to start with a very simple (inefficient) function and then optimize.

Project Ideas --III

- Metabolic network analysis
  - Find connected components in a metabolic network (modules).
  - Connecting molecules in a reaction network
    - find a local part of network containing the components
    - look for causal connections, 2's 3's ...
  - Simplify, for example suppressing connections that are tooo common