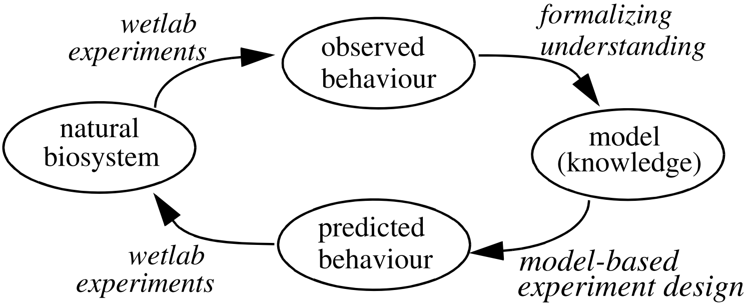
Part IV
PN-Based Model Checking of Biochemical Networks

Monika Heiner

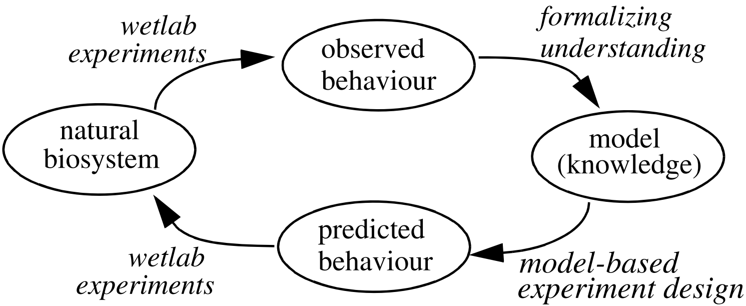
joint work with David Gilbert (London/UK), Robin Donaldson (Glasgow/UK)

University of Zaragoza, February 23, 2010

- *modelling as formal knowledge representation*

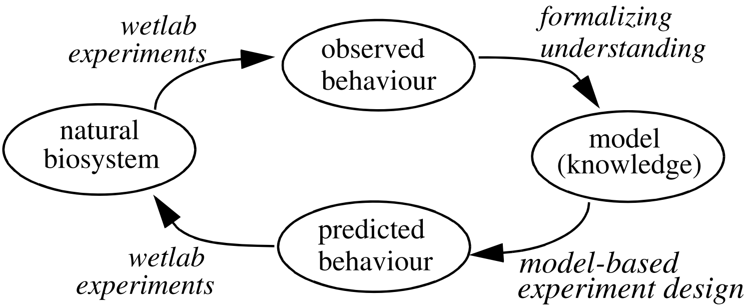


- *modelling as formal knowledge representation*



many assumptions, fuzzy/changing/growing knowledge

- *modelling as formal knowledge representation*

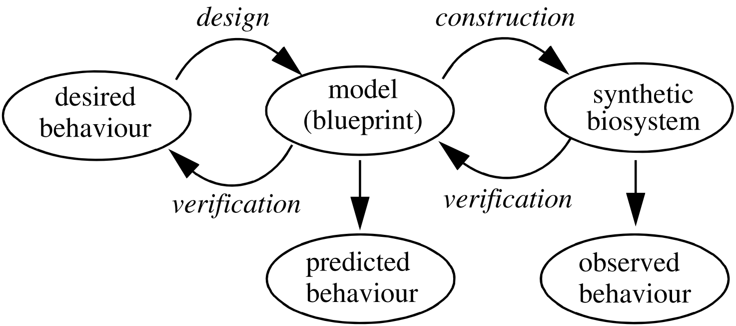


many assumptions, fuzzy/changing/growing knowledge

model needs to be validated

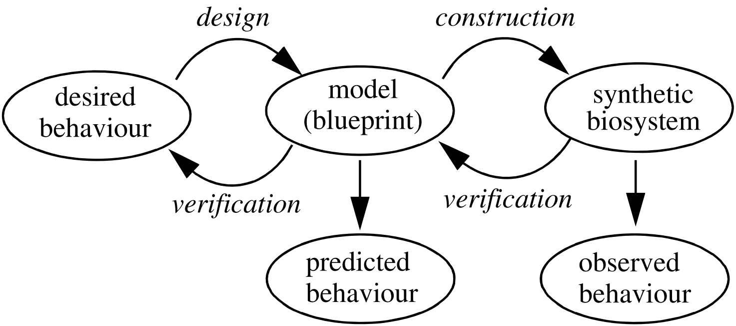
Models in Synthetic Biology

- *modelling for system construction*



Models in Synthetic Biology

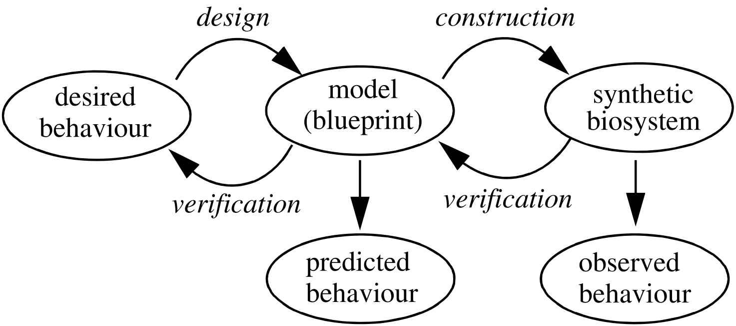
- *modelling for system construction*



reliable and robust engineering

Models in Synthetic Biology

- *modelling for system construction*



reliable and robust engineering

models serve as blueprint, need to be verified

Model Checking

In a sentence :

- "Formally check whether a model of a biochemical system does what we want"

Components :

- a model
  - the current description of a biochemical system of interest
- a property
  - a property which we think the system should have
- a model checker
  - a program to test whether the model has the property

What can we do with Model Checking ?

- *model validation*

  - Show that our model of the pathway matches the (stochastic) lab data.
- *model analysis*

  - In a collection of variants of a model (e.g., in silico gene knock-outs), which models show a certain behaviour (loss of oscillations . . .) ?
- *model development*

  - If the model doesn't do what we want, change the model automatically until it does ! (parameters, structures, . . .)
- *model finding*

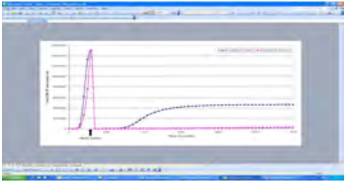
  - Many models in a database ; can use model checking to query the database :
  "Give me all the models in the database which oscillate."
- *biosystem verification - synthetic biology*

  - Does the constructed system do what we intended ?

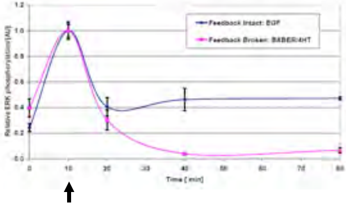
Biologists will often talk in qualitative or semi- quantitative language (trends).

- "This protein peaks after 5 minutes, then falls to half concentration."
- Often quite certain about time,
- but not about concentrations.

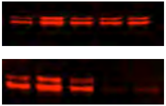
Simulation



Experiment



U0126 added



pERK1/2, +EGF

pERK1/2, + BXBER/4HT

0 10 20 40 80 min stimulation

Examples :

- after 100 seconds the concentration of Protein1 is stable
- protein1 peaks and falls
- protein1 peaks and stays constant
- protein1 peaks before Protein2
- protein1 oscillates 4 times in 5,000 seconds
- molecules of protein2 are required for molecules of protein1 to be created

Temporal logics to formally express properties

Various logics each with different expressivity :

- *Branching-time logics* consider all branching time lines
  - Computational Tree Logic (CTL)
  - Continuous Stochastic Logic (CSL)

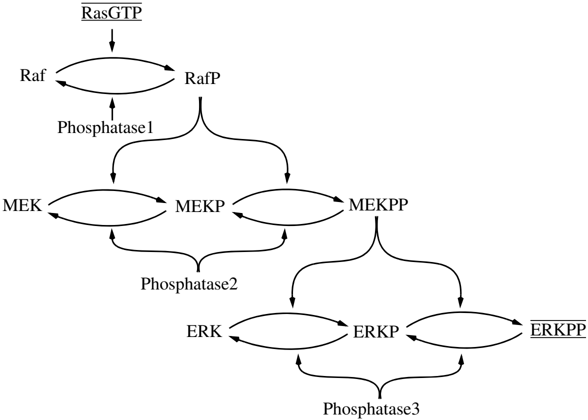
  "There is a possibility that I will stay hungry forever."
  "There is a possibility that eventually I am no longer hungry."

- *Linear-time logics* consider separately all single time lines
  - Linear-time Temporal Logics (LTL, LTLc, PLTLc)

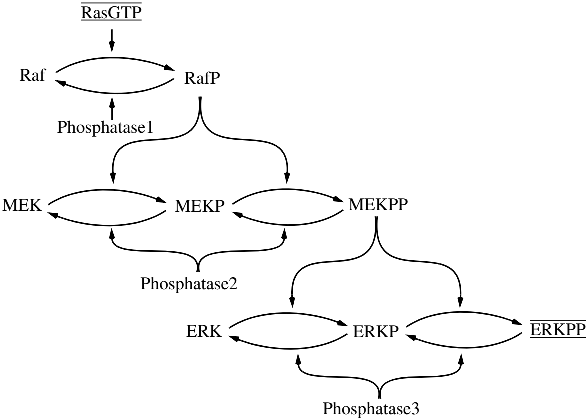
  "I am hungry."
  "I am always hungry."
  "I will eventually be hungry."
  "I will be hungry until I eat something."

Running Case Study

- *. . . a typical signalling cascade*



- ... *a typical signalling cascade*

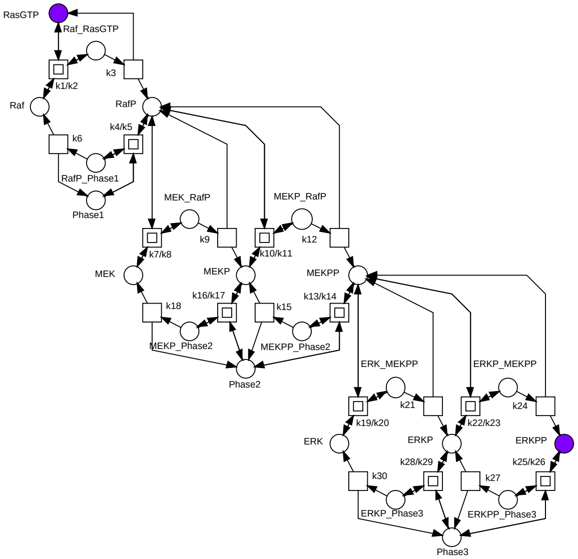


modelled in [Levchenko et al. 2000] like this ...

[Levchenko et al. 2000], *Supplemental Material : ODEs*

$$\begin{aligned}
d\mathrm{Raf}/dt &= k_2 * \mathrm{Raf_RasGTP} + k_6 * \mathrm{RafP_Phase1} - k_1 * \mathrm{Raf} * \mathrm{RasGTP} \\
d\mathrm{RasGTP}/dt &= k_2 * \mathrm{Raf_RasGTP} + k_3 * \mathrm{Raf_RasGTP} - k_1 * \mathrm{Raf} * \mathrm{RasGTP} \\
d\mathrm{Raf_RasGTP}/dt &= k_1 * \mathrm{Raf} * \mathrm{RasGTP} - k_2 * \mathrm{Raf_RasGTP} - k_3 * \mathrm{Raf_RasGTP} \\
d\mathrm{RafP}/dt &= k_3 * \mathrm{Raf_RasGTP} + k_{12} * \mathrm{MEKP_RafP} + k_9 * \mathrm{MEK_RafP} + \\
&\quad k_5 * \mathrm{RafP_Phase1} + k_8 * \mathrm{MEK_RafP} + k_{11} * \mathrm{MEKP_RafP} - \\
&\quad k_7 * \mathrm{RafP} * \mathrm{MEK} - k_{10} * \mathrm{MEKP} * \mathrm{RafP} - k_4 * \mathrm{Phase1} * \mathrm{RafP} \\
d\mathrm{RafP_Phase1}/dt &= k_4 * \mathrm{Phase1} * \mathrm{RafP} - k_5 * \mathrm{RafP_Phase1} - k_6 * \mathrm{RafP_Phase1} \\
d\mathrm{MEK_RafP}/dt &= k_7 * \mathrm{RafP} * \mathrm{MEK} - k_8 * \mathrm{MEK_RafP} - k_9 * \mathrm{MEK_RafP} \\
d\mathrm{MEKP_RafP}/dt &= k_{10} * \mathrm{MEKP} * \mathrm{RafP} - k_{11} * \mathrm{MEKP_RafP} - k_{12} * \mathrm{MEKP_RafP} \\
d\mathrm{MEKP_Phase2}/dt &= k_{16} * \mathrm{Phase2} * \mathrm{MEKP} - k_{18} * \mathrm{MEKP_Phase2} - k_{17} * \mathrm{MEKP_Phase2} \\
d\mathrm{MEKPP_Phase2}/dt &= k_{13} * \mathrm{MEKPP} * \mathrm{Phase2} - k_{15} * \mathrm{MEKPP_Phase2} - k_{14} * \mathrm{MEKPP_Phase} \\
d\mathrm{ERK}/dt &= k_{20} * \mathrm{ERK_MEKPP} + k_{30} * \mathrm{ERKP_Phase3} - k_{19} * \mathrm{MEKPP} * \mathrm{ERK} \\
d\mathrm{ERK_MEKPP}/dt &= k_{19} * \mathrm{MEKPP} * \mathrm{ERK} - k_{20} * \mathrm{ERK_MEKPP} - k_{21} * \mathrm{ERK_MEKPP} \\
d\mathrm{ERKP_MEKPP}/dt &= k_{22} * \mathrm{MEKPP} * \mathrm{ERKP} - k_{24} * \mathrm{ERKP_MEKPP} - k_{23} * \mathrm{ERKP_MEKPP} \\
\mathit{etcetera} &= \ldots
\end{aligned}$$

Running Case Study



- **initial marking construction**
  P-invariants

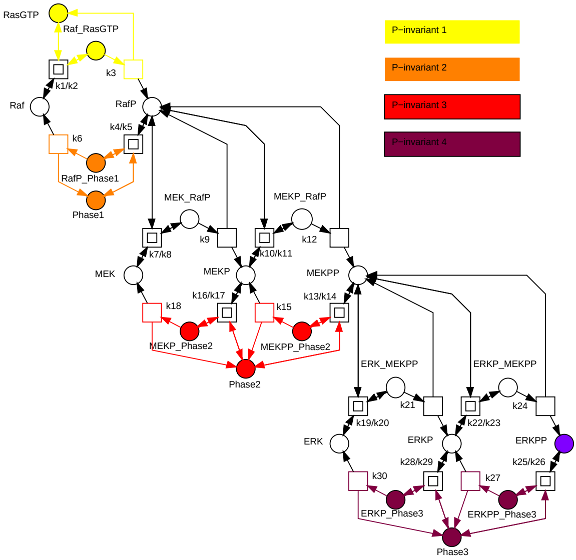
- **initial marking construction**
  P-invariants
- **subnetwork identification**
  - P-invariants : token preserving modules (*mass conservation*)
  - T-invariants : state repeating modules (*elementary modes*)

- **initial marking construction**
  P-invariants
- **subnetwork identification**
  - P-invariants : token preserving modules (*mass conservation*)
  - T-invariants : state repeating modules (*elementary modes*)
- **general behavioural properties**
  - *boundedness :* every place gets finite token number only
  - *liveness :* every transition may happen forever
  - *reversibility :* every state may be reached forever

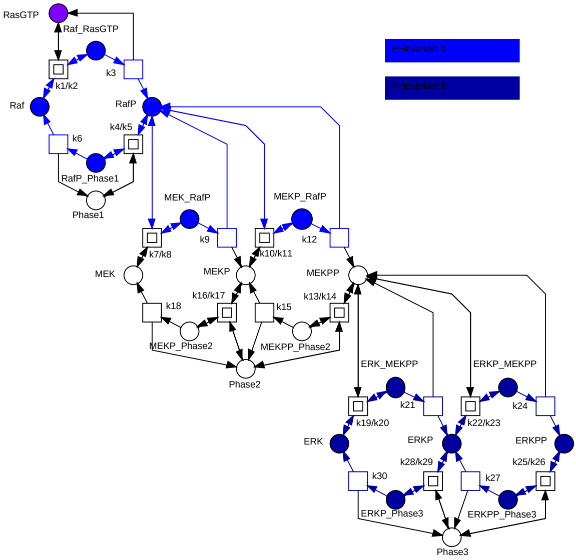
Qualitative Analysis

- **initial marking construction**
  P-invariants
- **subnetwork identification**
  - P-invariants : token preserving modules (*mass conservation*)
  - T-invariants : state repeating modules (*elementary modes*)
- **general behavioural properties**
  - *boundedness :* every place gets finite token number only
  - *liveness :* every transition may happen forever
  - *reversibility :* every state may be reached forever
- **special behavioural properties**
  CTL / LTL model checking

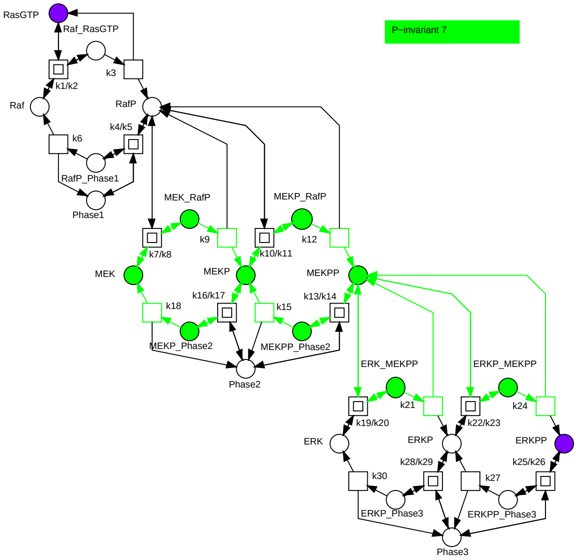
Running Case Study - P-invariants



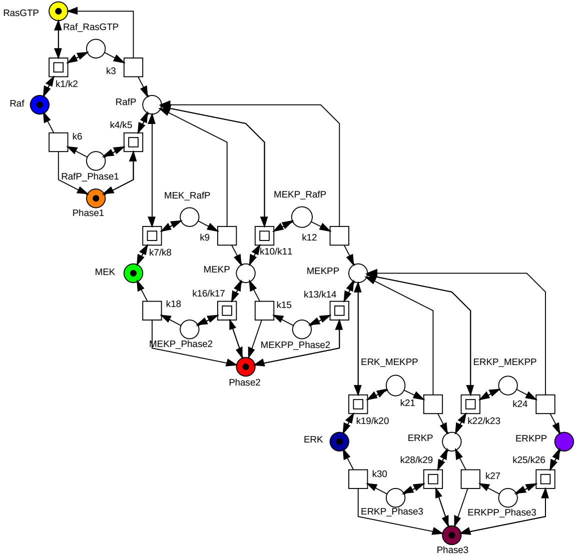
Running Case Study - P-invariants



Running Case Study - P-invariants



Running Case Study - initial marking



Running Case Study - general properties

- Covered by P-invariants (CPI) $\Rightarrow$ **bounded**

Running Case Study - general properties

- Covered by P-invariants (CPI) ⇒ **bounded**
- Deadlock-Trap Property (DTP) holds ⇒ **no dead states**

Running Case Study - general properties

- Covered by P-invariants (CPI) $\Rightarrow$ **bounded**
- Deadlock-Trap Property (DTP) holds $\Rightarrow$ **no dead states**
- *state space*

levels	reachability graph number of states	IDD data structure number of nodes
1	118	52
4	$2.4 \cdot 10^4$	115
8	$6.1 \cdot 10^6$	269
80	$5.6 \cdot 10^{18}$	13,472
120	$1.7 \cdot 10^{21}$	29,347

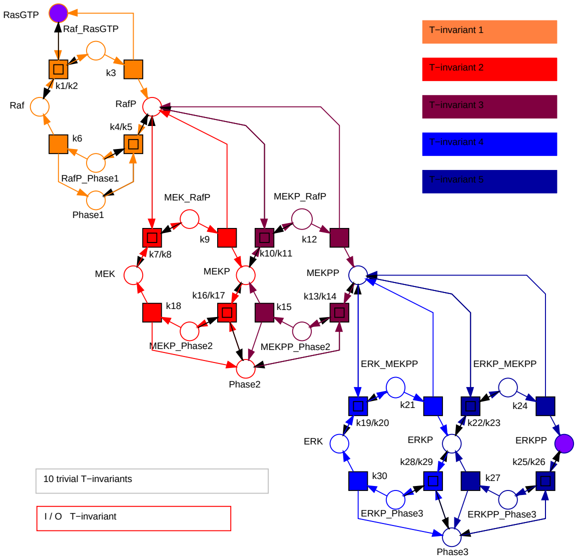
Running Case Study - general properties

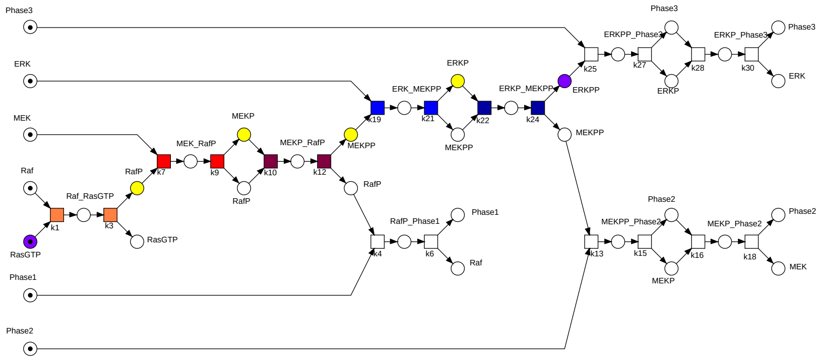
- Covered by P-invariants (CPI) ⇒ **bounded**
- Deadlock-Trap Property (DTP) holds ⇒ **no dead states**
- *state space*

levels	reachability graph number of states	IDD data structure number of nodes
1	118	52
4	$2.4 \cdot 10^4$	115
8	$6.1 \cdot 10^6$	269
80	$5.6 \cdot 10^{18}$	13,472
120	$1.7 \cdot 10^{21}$	29,347

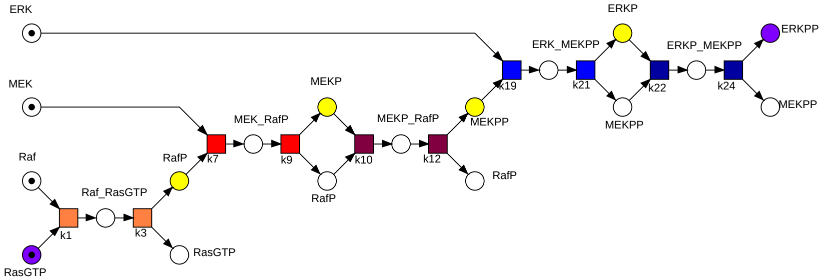
- reachability graph
  - strongly connected ⇒ **reversible**
  - contains every transition (reaction) ⇒ **live**

Running Case Study - T-invariants




Phase3
ERK
MEK
Raf
RasGTP
Phase1
Phase2
k1
Raf_RasGTP
k3
RasGTP
RafP
k7
MEK_RafP
k9
MEKP
RafP
k10
MEKP_RafP
k12
MEKPP
RafP
k19
ERK_MEKPP
k21
ERKP
MEKPP
k22
ERKP_MEKPP
k24
ERKPP
MEKPP
k4
RafP_Phase1
k6
Phase1
Raf
k25
ERKPP_Phase3
k27
Phase3
ERKP
k28
ERKP_Phase3
k30
Phase3
ERK
k13
MEKPP_Phase2
k15
Phase2
MEKP
k16
MEKP_Phase2
k18
Phase2
MEK

Running Case Study - partial order run of I/O T-invariant

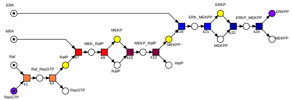


There is a path ...

- $EX\phi$
  - if there is a state reachable by one step where ϕ holds.
- $EF\phi$
  - if there is a path where ϕ holds finally, i.e., at some point.
- $EG\phi$
  - if there is a path where ϕ holds globally, i.e., forever.
- $E(\phi_1 U \phi_2)$
  - if there is a path where ϕ_1 holds until ϕ_2 holds.

For all path . . .

- $AX\phi$
  - if ϕ holds for all states which are reachable by one step.
- $AF\phi$
  - if ϕ holds finally (at some point) for all paths.
- $AG\phi$
  - if ϕ holds globally (i.e. for ever) for all paths.
- $A(\phi_1 U \phi_2)$
  - if ϕ_1 holds until ϕ_2 holds for all paths.



property Q1 :

The signal sequence predicted by the partial order run of the I/O T-invariant is the only possible one ;
i.e., starting at the initial state, it is necessary to pass through RafP, MEKP, MEKPP and ERKP in order to reach ERKPP.

$$\neg\, [\, \mathbf{E}\,(\, \neg\, \text{RafP}\ \mathbf{U}\ \text{MEKP}\,) \vee \mathbf{E}\,(\, \neg\, \text{MEKP}\ \mathbf{U}\ \text{MEKPP}\,) \vee \mathbf{E}\,(\, \neg\, \text{MEKPP}\ \mathbf{U}\ \text{ERKP}\,) \vee \mathbf{E}\,(\, \neg\, \text{ERKP}\ \mathbf{U}\ \text{ERKPP}\,)\ ]$$

Stochastic Model Checking - Preparation

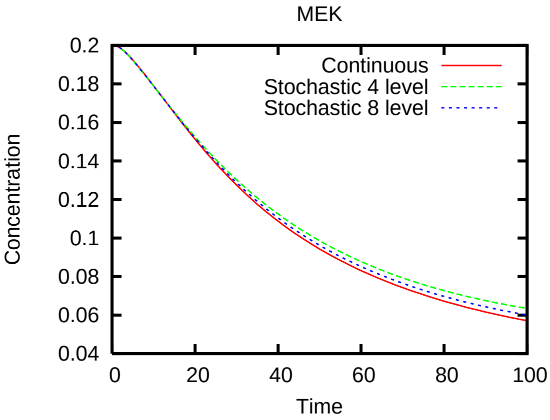
- *isomorphy of reachability graph and CTMC,*
  thus all qualitative properties still valid

- *How many levels needed for quantitative evaluation ?*
  - state space(1 levels) = 118 (Boolean interpretation)
  - state space(4 levels) = 24,065
  - state space(8 levels) = 6,110,643

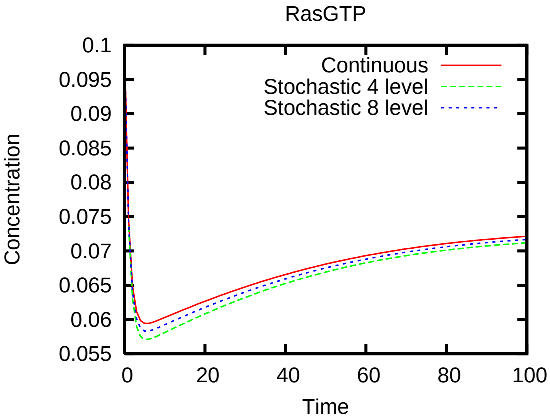
- *equivalence check*

$$C_{RafP}(t) \;=\; \frac{0.1}{s} \cdot \underbrace{\sum_{i=1}^{4s} (i \cdot P(L_{RafP}(t) = i))}_{expected\;\; value\;\; of\;\; L_{RafP}(t)}$$

- *equivalence check, results, e.g. for MEK :*



- *equivalence check, results, e.g. for RasGTP :*



Replaces the path quantifiers (E, A) in CTL by the probability operator $P_{\trianglelefteq x}$, where $\trianglelefteq x$ specifies the probability x of the formula.

- $P_{=?}[X\phi]$
  - prob there is a state reachable by one step where ϕ holds.
- $P_{=?}[F\phi]$
  - prob there is a path where ϕ holds finally, i.e., at some point.
- $P_{=?}[G\phi]$
  - prob there is a path where ϕ holds globally, i.e., forever.
- $P_{=?}[\phi_1 U \phi_2]$
  - prob there is a path where ϕ_1 holds until ϕ_2 holds.

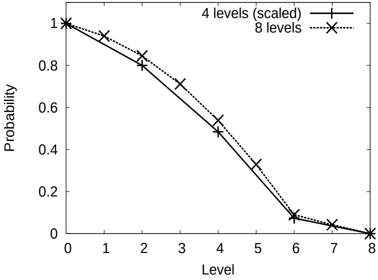
Syntactic sugar

- $\phi_1\{\phi_2\}$ - ϕ_1 happens from the first time ϕ_2 happens, where no temporal operators in ϕ_2.

property S1 :

What is the probability of the concentration of RafP increasing, when starting in a state where the level is already at L ?

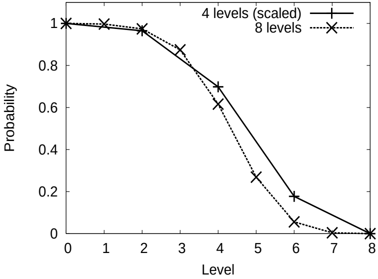
$$P_{=?}\ [\ (\ \mathrm{RafP}\ =\ \mathrm{L}\ )\ \mathbf{U}^{<=100}\ (\ \mathrm{RafP} > \mathrm{L}\ )\ \{\ \mathrm{RafP} = \mathrm{L}\ \}\ ]$$



property S2 :

What is the probability that RafP is the first species to react ?

$$P_{=?}\,[\,(\,(\,MEKPP\ =\ 0\,)\wedge(\,ERKPP\ =\ 0\,)\,)\,\mathbf{U}^{<=100}\,(\,RafP > \ L\,)\ \{\,(\,MEKPP\ =\ 0\,)\ \wedge\ (\,ERKPP\ =\ 0\,)\ \wedge\ (\,RafP = \ 0\,)\,\}\ ]$$

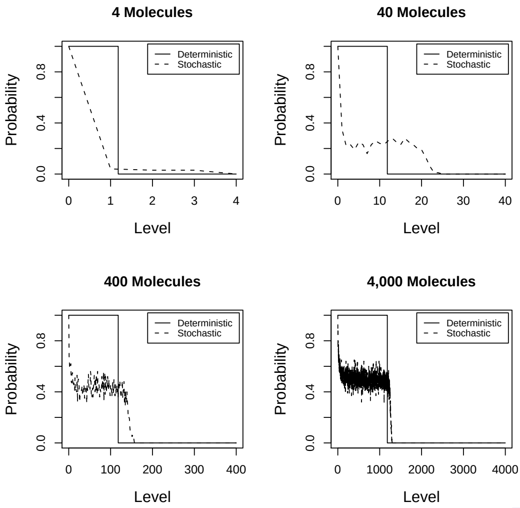


Example figures for MC2 model checking of property S1 at varying number of levels/molecules.

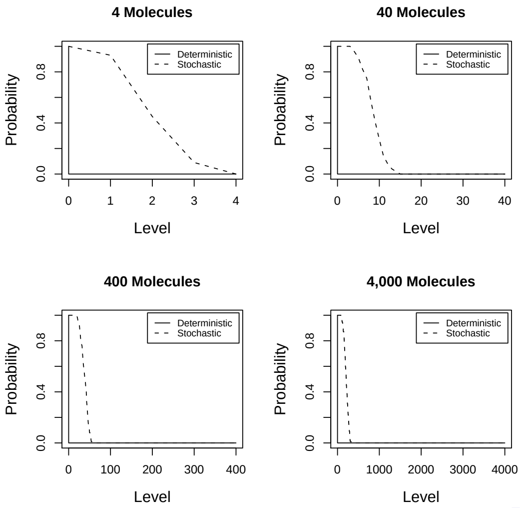
Levels	MC Time[a]	Simulation Output Size
4	10 s [b]	750 KB
8	15 s [b]	1.5 MB
40	1.5 minutes [b]	7.5 MB
400	1 minute [c]	80 MB
4,000	30 minutes [c]	900 MB

[a] Both Gillespie simulation and MC2 checking.
[b] Computation on a standard workstation.
[c] Distributed computation on a computer cluster comprising 45 Sun X2200 servers each with 2 dual core processors (180 CPU cores).

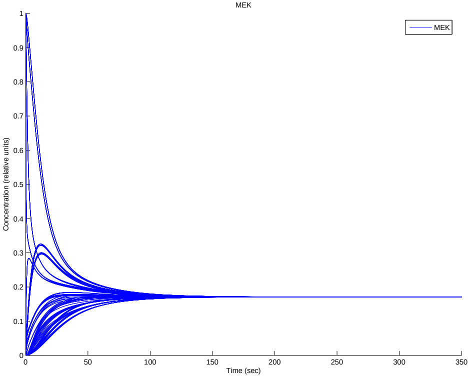
- *S1 at varying number of molecules shows progression towards deterministic behaviour as number of molecules increases.*



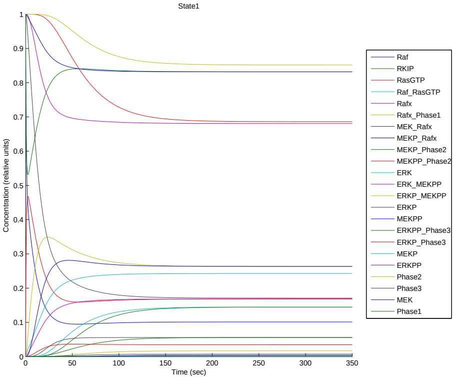
- *S2 at varying number of molecules shows progression towards deterministic behaviour as number of molecules increases.*



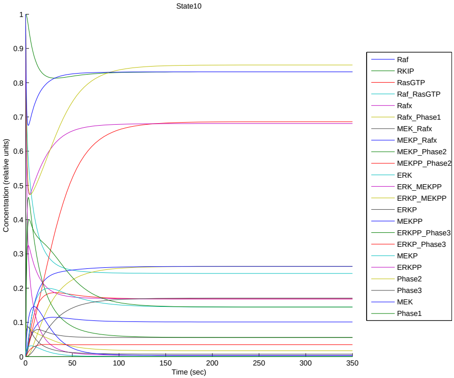
- *steady state analysis, results for all 118 'good' states, e.g. for MEK :*



- *steady state analysis for state 1 :*



- *steady state analysis for state 10 :*



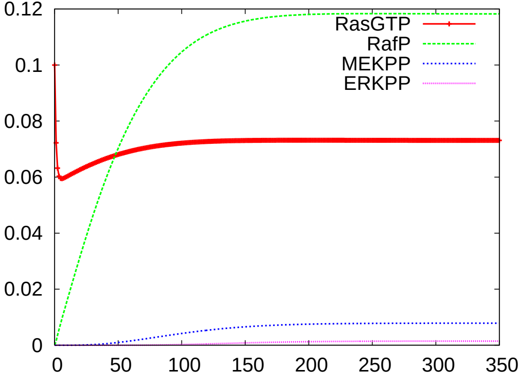
For all single path ...

- $X\phi$
  - ϕ happens in the next time point.
- $F\phi$
  - ϕ happens at some time.
- $G\phi$
  - ϕ always happens.
- $A(\phi_1 U \phi_2)$
  - ϕ_1 happens until ϕ_2 happens.

Syntactic sugar

- $\phi_1\{\phi_2\}$ - ϕ_1 happens from the first time ϕ_2 happens, where no temporal operators in ϕ_2.

- *transient analysis for RasGTP, RafP, MEKPP, ERKPP :*



property C1 :

The concentration of RafP rises to a significant level, while the concentrations of MEKPP and ERKPP remain close to zero ; i.e. *RafP is really the first species to react.*

$$((\text{MEKPP} < 0.001) \wedge (\text{ERKPP} < 0.0002))\ \mathbf{U}\ (\text{RafP} > 0.06)$$

property C2 :

if the concentration of RafP is at a significant concentration level and that of ERKPP is close to zero, then both species remain in these states until the concentration of MEKPP becomes significant ; i.e. *MEKPP is the second species to react.*

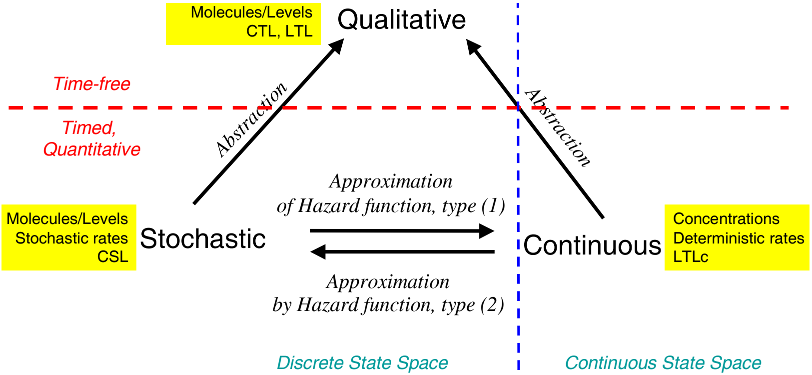
$$((\text{RafP} > 0.06) \wedge (\text{ERKPP} < 0.0002)) \Rightarrow$$
$$((\text{RafP} > 0.06) \wedge (\text{ERKPP} < 0.0002)) \ \mathbf{U} \ (\text{MEKPP} > 0.004)$$

property C3 :

if the concentrations of RafP and MEKPP are significant, they remain so, until the concentration of ERKPP becomes significant ; i.e. *ERKPP is the third species to react.*

$$((\text{RafP} > 0.06) \wedge (\text{MEKPP} > 0.004)) \Rightarrow$$
$$((\text{RafP} > 0.06) \wedge (\text{MEKPP} > 0.004)) \ \mathbf{U} \ (\text{ERKPP} > 0.0005)$$

Framework


Molecules/Levels
CTL, LTL
Qualitative
Time-free
Timed,
Quantitative
Abstraction
Abstraction
Molecules/Levels
Stochastic rates
CSL
Stochastic
Approximation
of Hazard function, type (1)
Continuous
Concentrations
Deterministic rates
LTLc
Approximation
by Hazard function, type (2)
Discrete State Space
Continuous State Space

- *validation criterion 1*
  - all expected structural properties hold
  - all expected general behavioural properties hold

- *validation criterion 1*
  - all expected structural properties hold
  - all expected general behavioural properties hold
- *validation criterion 2*
  - CPI
  - no minimal P-invariant without biological interpretation (?)

- *validation criterion 1*
  - all expected structural properties hold
  - all expected general behavioural properties hold
- *validation criterion 2*
  - CPI
  - no minimal P-invariant without biological interpretation (?)
- *validation criterion 3*
  - CTI
  - no minimal T-invariant without biological interpretation
  - no known biological behaviour without corresponding T-invariant

Bionetworks, Validation

- *validation criterion 1*
  - all expected structural properties hold
  - all expected general behavioural properties hold
- *validation criterion 2*
  - CPI
  - no minimal P-invariant without biological interpretation (?)
- *validation criterion 3*
  - CTI
  - no minimal T-invariant without biological interpretation
  - no known biological behaviour without corresponding T-invariant
- *validation criterion 4*
  - all expected special behavioural properties hold
  - temporal-logic properties yield TRUE

Summary

- *unifying framework*
  qualitative & stochastic & continuous paradigms

Summary

- *unifying framework*
  qualitative & stochastic & continuous paradigms
- *three models sharing structure*
  - qualitative Petri nets → time-free analyses
  - stochastic Petri nets → CTMC
  - continuous Petri nets → ODEs

Summary

- *unifying framework*
  qualitative & stochastic & continuous paradigms
- *three models sharing structure*
  - qualitative Petri nets $\longrightarrow$ time-free analyses
  - stochastic Petri nets $\longrightarrow$ CTMC
  - continuous Petri nets $\longrightarrow$ ODEs
- *running case study*
  ERK signalling pathway

Summary

- *unifying framework*
  qualitative & stochastic & continuous paradigms
- *three models sharing structure*
  - qualitative Petri nets → time-free analyses
  - stochastic Petri nets → CTMC
  - continuous Petri nets → ODEs
- *running case study*
  ERK signalling pathway
- *focus - transient analysis*, esp. by
  - transition invariants & partial order run
  - qualitative & stochastic & continuous model checking

Summary

- *unifying framework*
  qualitative & stochastic & continuous paradigms
- *three models sharing structure*
  - qualitative Petri nets → time-free analyses
  - stochastic Petri nets → CTMC
  - continuous Petri nets → ODEs
- *running case study*
  ERK signalling pathway
- *focus - transient analysis*, esp. by
  - transition invariants & partial order run
  - qualitative & stochastic & continuous model checking
- **not bound to the Petri net perspective**

Some Open Problems

- *increasing level number = increasing accuracy*
  BUT, monotonous liveness holds for substructures (EFC) only !

Some Open Problems

- *increasing level number = increasing accuracy*
  BUT, monotonous liveness holds for substructures (EFC) only !
- *unbounded qualitative model + time = bounded model*
  BUT, that's not always the case !
  (structural) criteria for time-dependent boundedness ?

Some Open Problems

- *increasing level number = increasing accuracy*
  BUT, monotonous liveness holds for substructures (EFC) only !
- *unbounded qualitative model + time = bounded model*
  BUT, that's not always the case !
  (structural) criteria for time-dependent boundedness ?
- *continuous behaviour = averaged stochastic behaviour*
  BUT, that's not always the case !
  stochastic and continuous behaviour may differ ; why ? when ?

Some Open Problems

- *increasing level number = increasing accuracy*
  BUT, monotonous liveness holds for substructures (EFC) only !
- *unbounded qualitative model + time = bounded model*
  BUT, that's not always the case !
  (structural) criteria for time-dependent boundedness ?
- *continuous behaviour = averaged stochastic behaviour*
  BUT, that's not always the case !
  stochastic and continuous behaviour may differ ; why ? when ?
- *sharing structure = sharing properties*
  BUT, to which extend ?
  relation : qualitative properties & continuous behaviour ?

Finally

- all data files and analysis results available at www-dssz.informatik.tu-cottbus.de/examples/levchenko
- M Heiner, D Gilbert, R Donaldson :
  Petri Nets for Systems and Synthetic Biology ;
  SFM 2008, Springer LNCS 5016, pp. 215-264, 2008.