From qualitative to quantitative formal methods for biochemical signalling pathways

Oana Andrei

University of Glasgow

LIFL, February 25, 2010

joint work with Hélène Kirchner (INRIA Bordeaux) and
Muffy Calder (University of Glasgow)

Outline

- Motivation
- Rule-based modelling
- Abstractions for CTMCs
- Conclusion and perspectives

Formal methods for modelling biological systems

- lab experiments
- computational model
- results/analysis

Goals: to understand, to predict, to control

Cell signalling

- communication between cells
- cellular processes: cell growth, proliferation, apoptosis...
- malfunctions may lead to diseases

Challenges

- suitable formalisms
- abstraction techniques
- analysis
- scalability

Our approaches

- qualitative: rule-based, higher-order calculus, runtime-verification
- quantitative: abstraction for CTMCs - CTMCs with levels, stochastic model checking

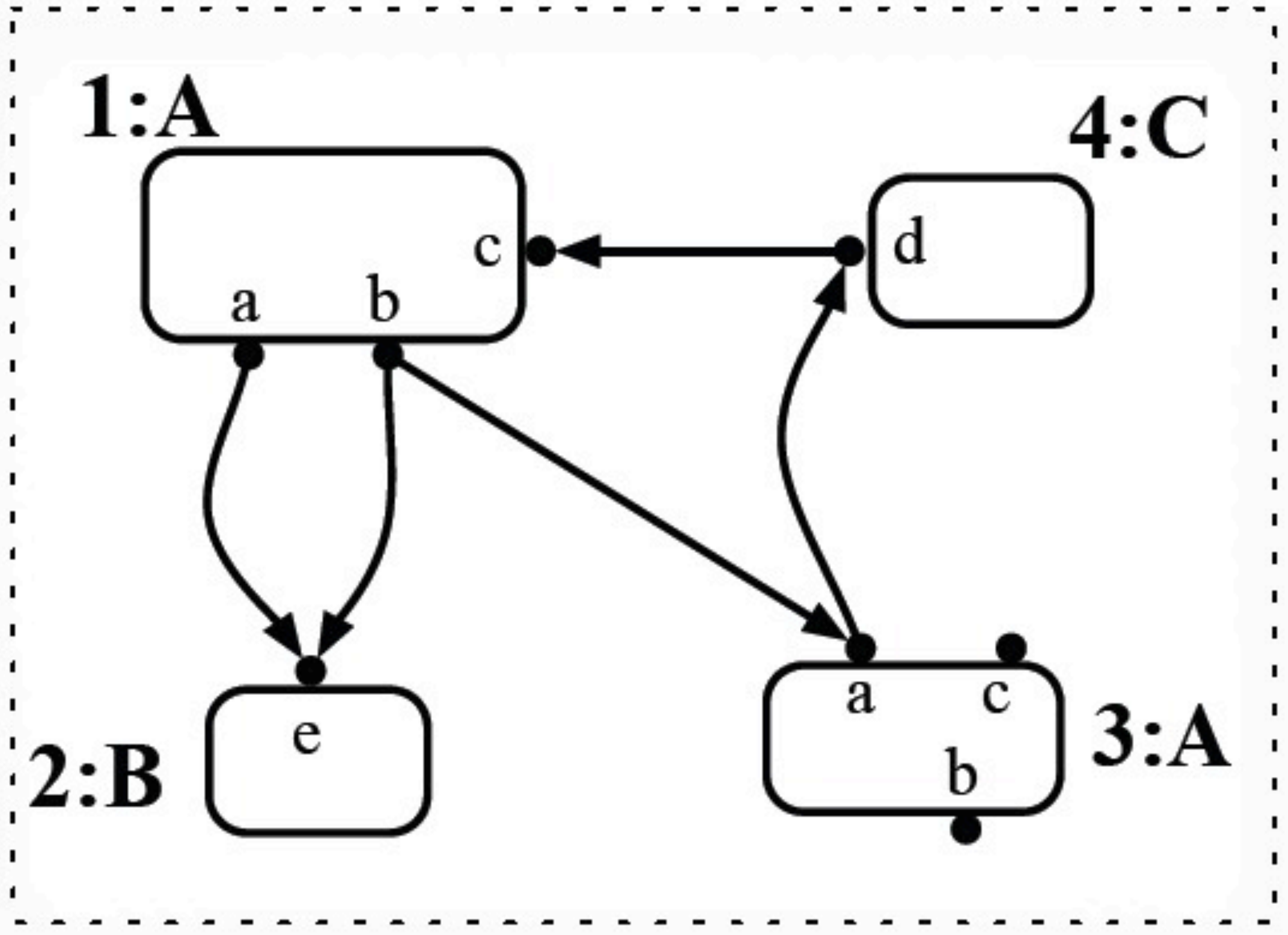
Higher-order rule-based modelling

Port graphs

- graphs with multiple edges and loops
- edges connect to ports of nodes
- defined over a signature (N, P)

[AndreiKirchner07-RULE]

A port graph



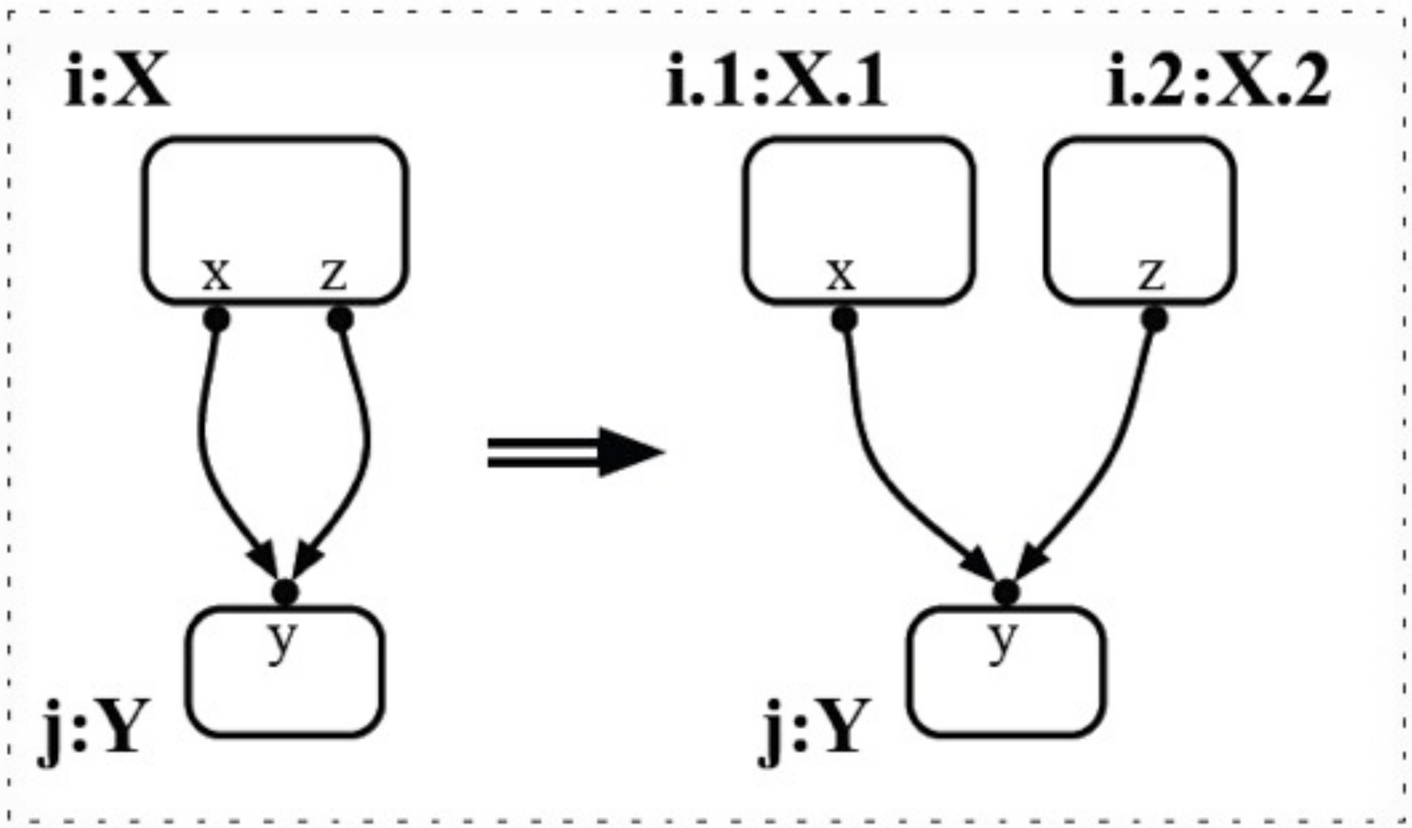
Molecular graphs as port graphs

Molecular complex	Port graph
protein	node
site	port
bond	edge
interaction	rewrite rule

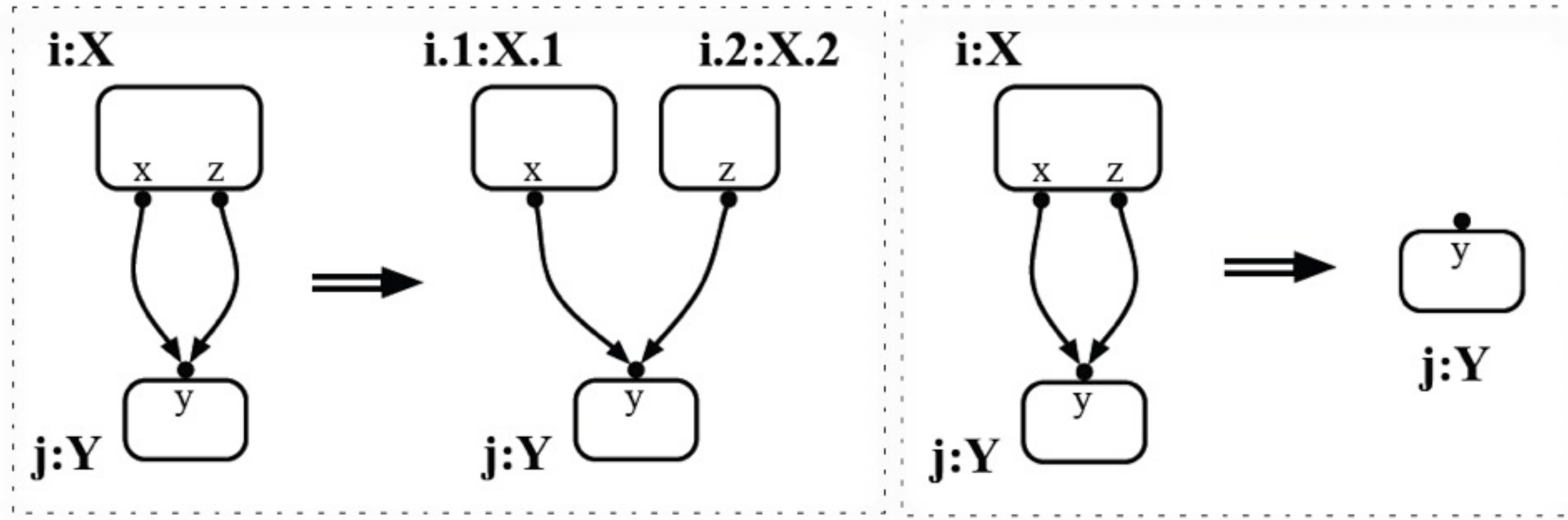
Rewrite rules and strategies

- well-suited for modelling bio-molecular interactions
- a rule $L \rightarrow R$ defines a class of reactions
- rewrite strategies control the rule application (*Identity, Failure, Sequence, Not, First, ...*)

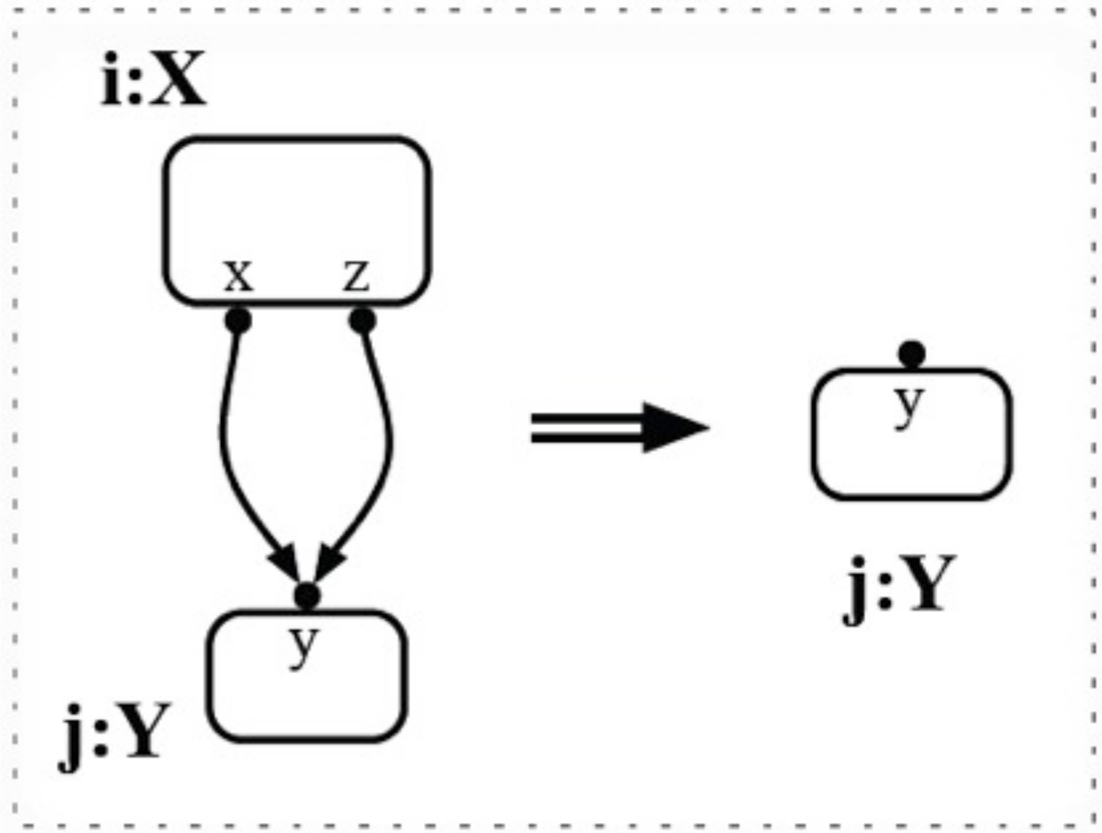
Port Graph Rewrite Rules



Port Graph Rewrite Rules

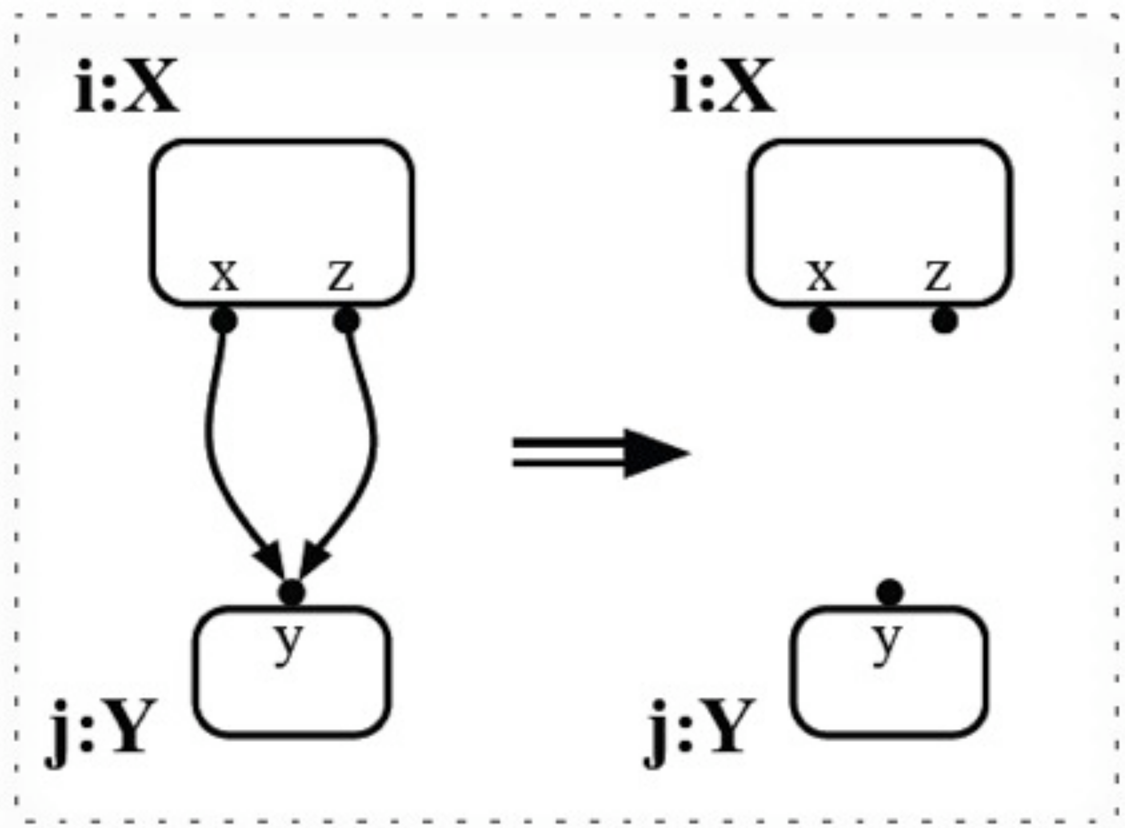


Port Graph Rewrite Rules

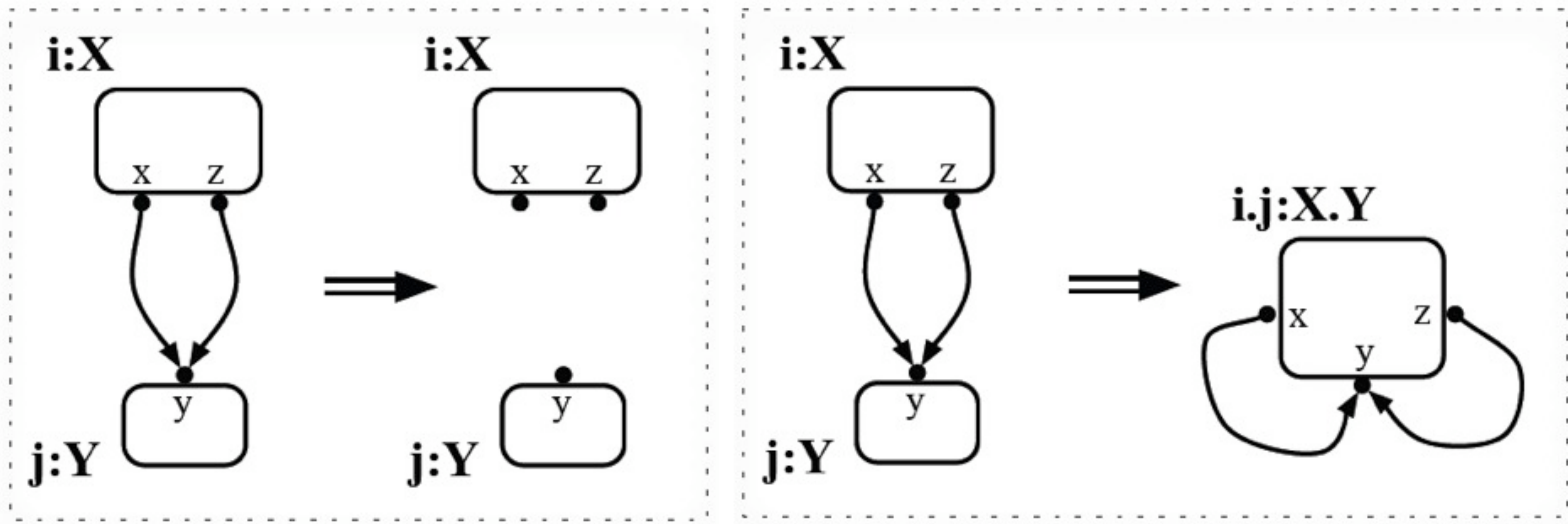


Port Graph Rewrite Rules

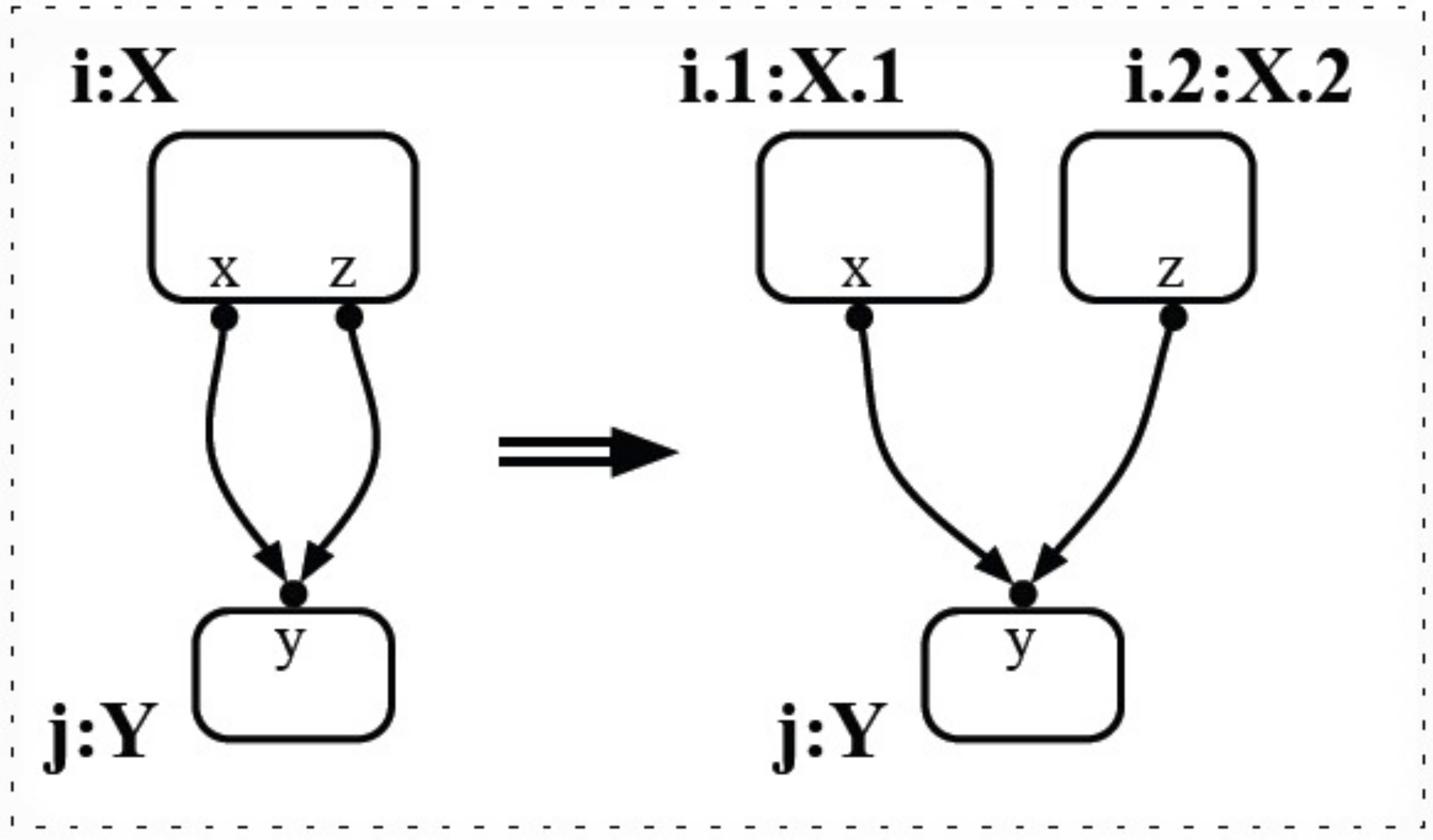
Port Graph Rewrite Rules



Port Graph Rewrite Rules

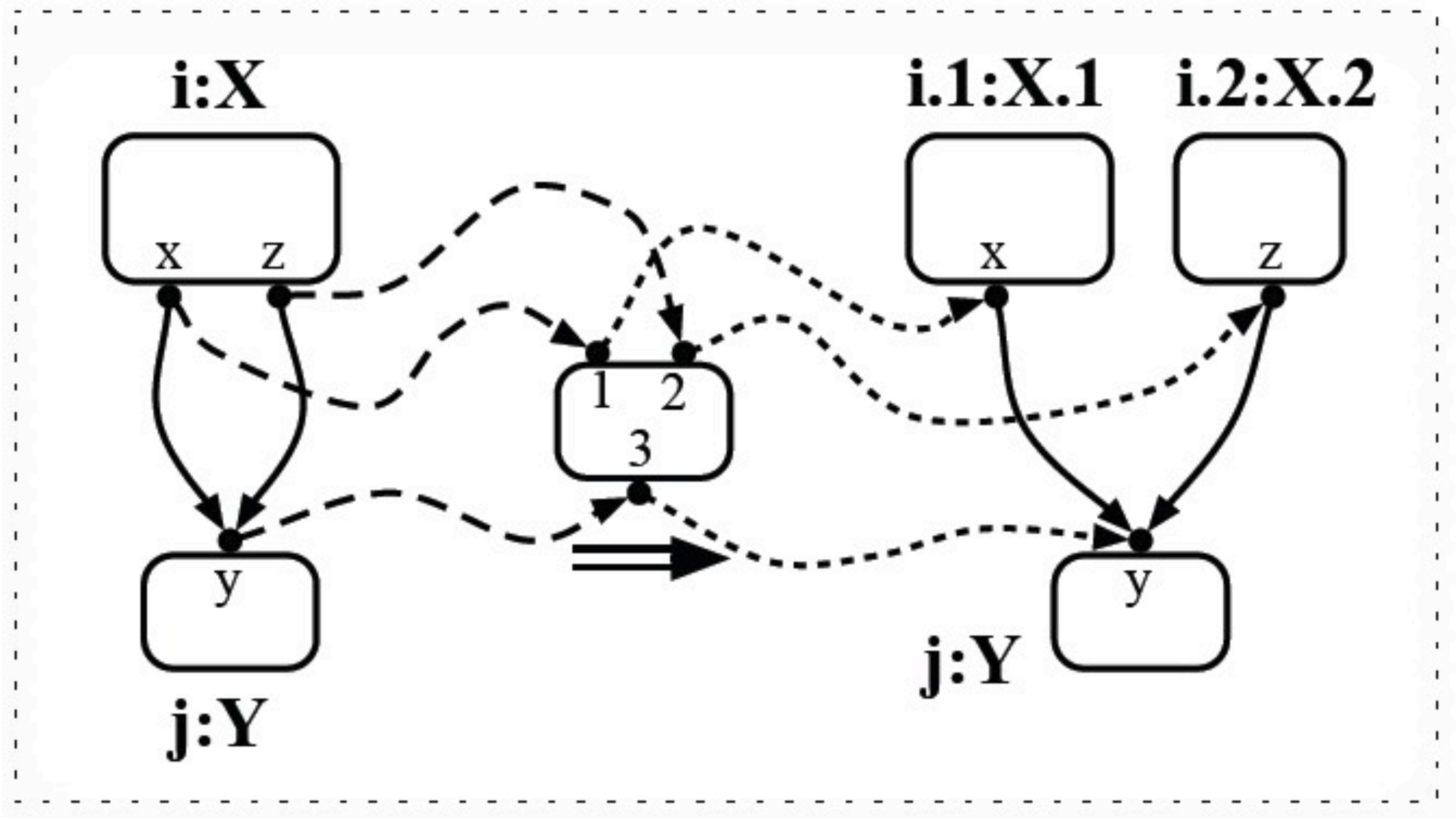


A Port Graph Rewrite Rule is a Port Graph



A Port Graph Rewrite Rule is a Port Graph

A Port Graph Rewrite Rule is a Port Graph



Port Graph Rewriting Relation

$$G \Rightarrow_{L \Rightarrow R} G' \qquad \text{if} \qquad \exists (g, G^{-}, \mathcal{B}) \in \mathcal{S}ol(L \lll G)$$

such that

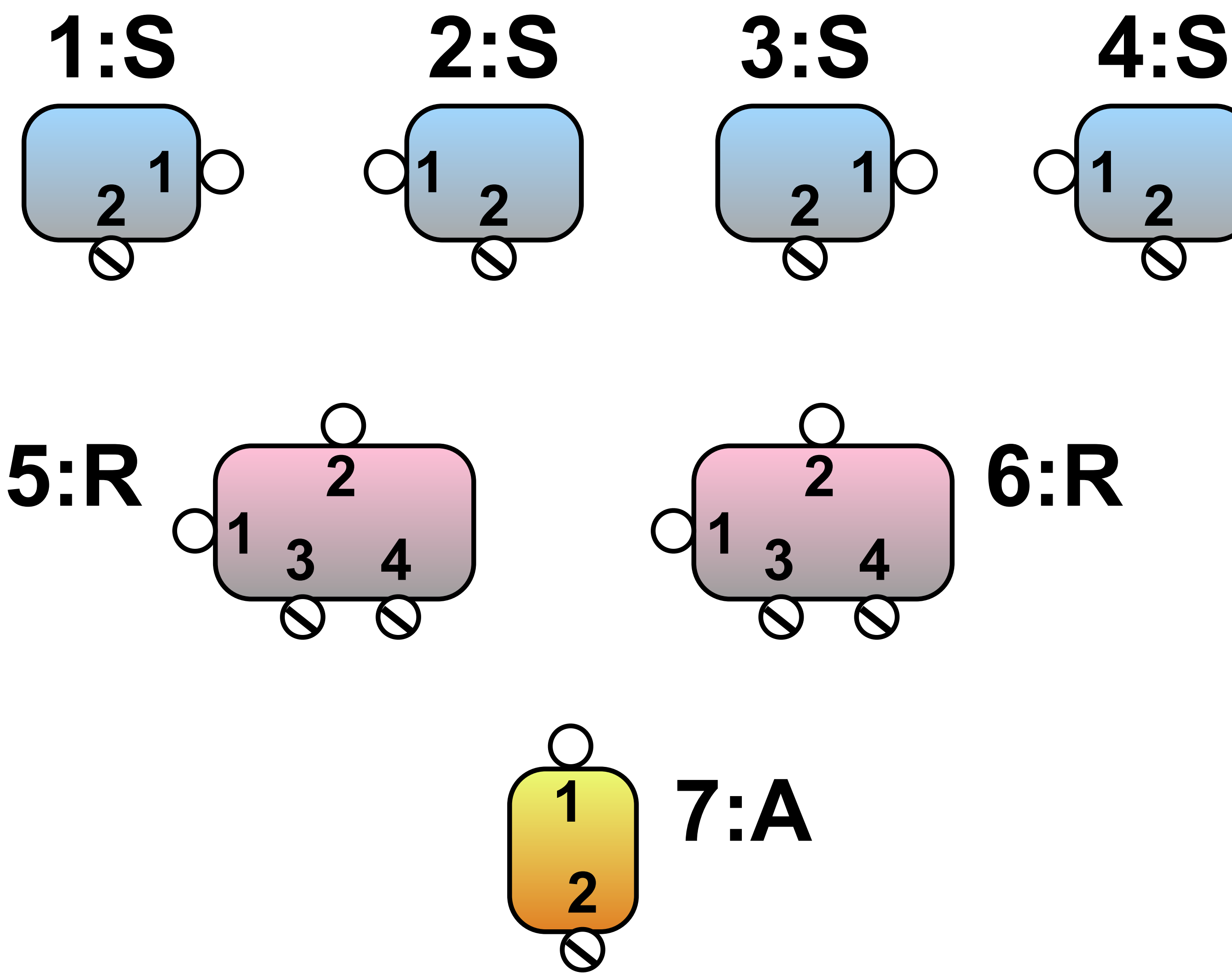
$$G = G^{-} \lfloor g(L) \rfloor_{\mathcal{B}}$$

and

$$G' = G^{-} \lfloor g(R) \rfloor_{\Downarrow_g \mathcal{B}}$$

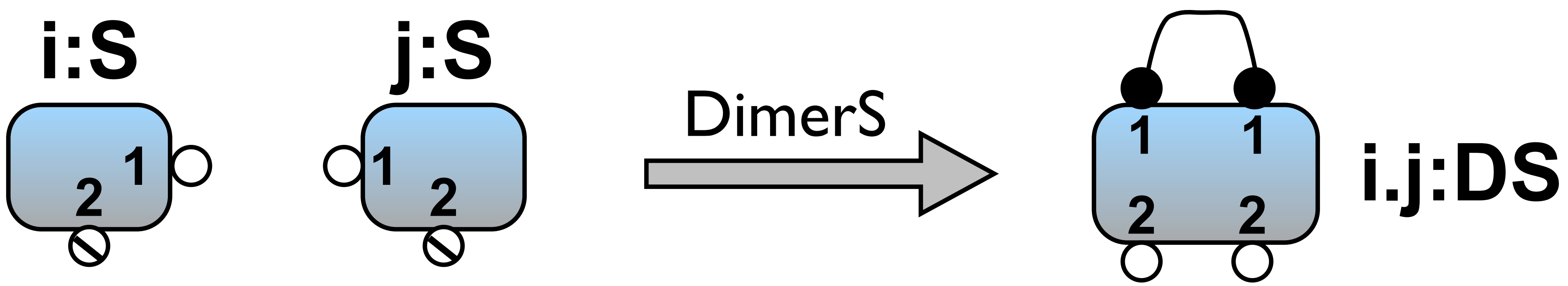
Example: a fragment of the EGFR signaling pathway

Initial state:

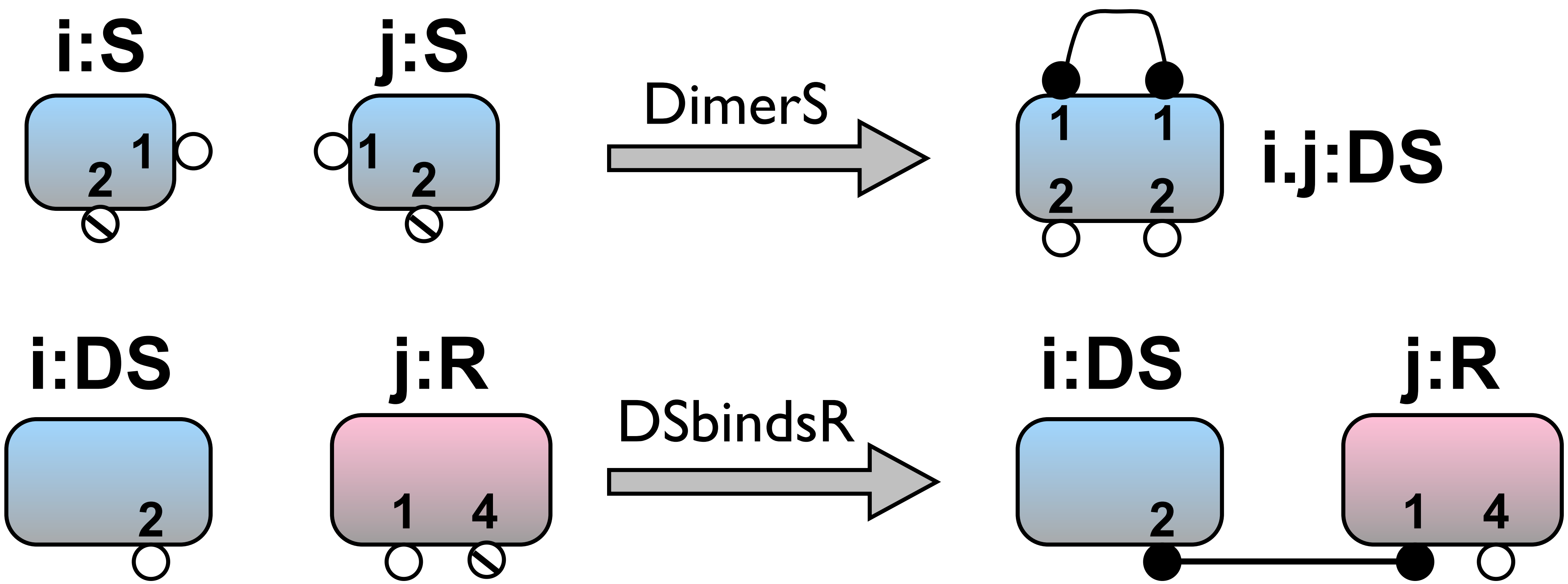


[AndreiKirchner07-NCA]

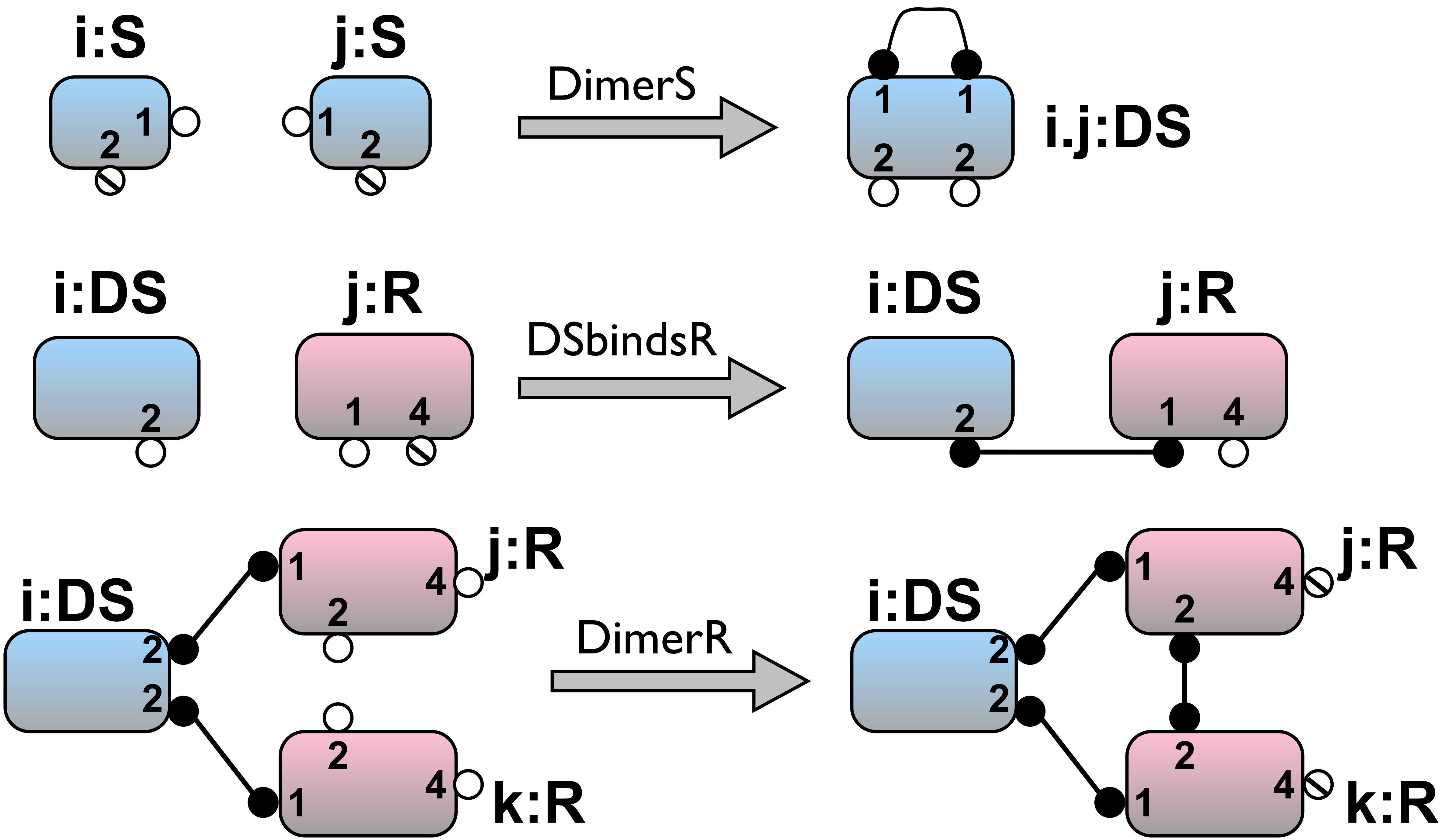
Example: a fragment of the EGFR signaling pathway



Example: a fragment of the EGFR signaling pathway

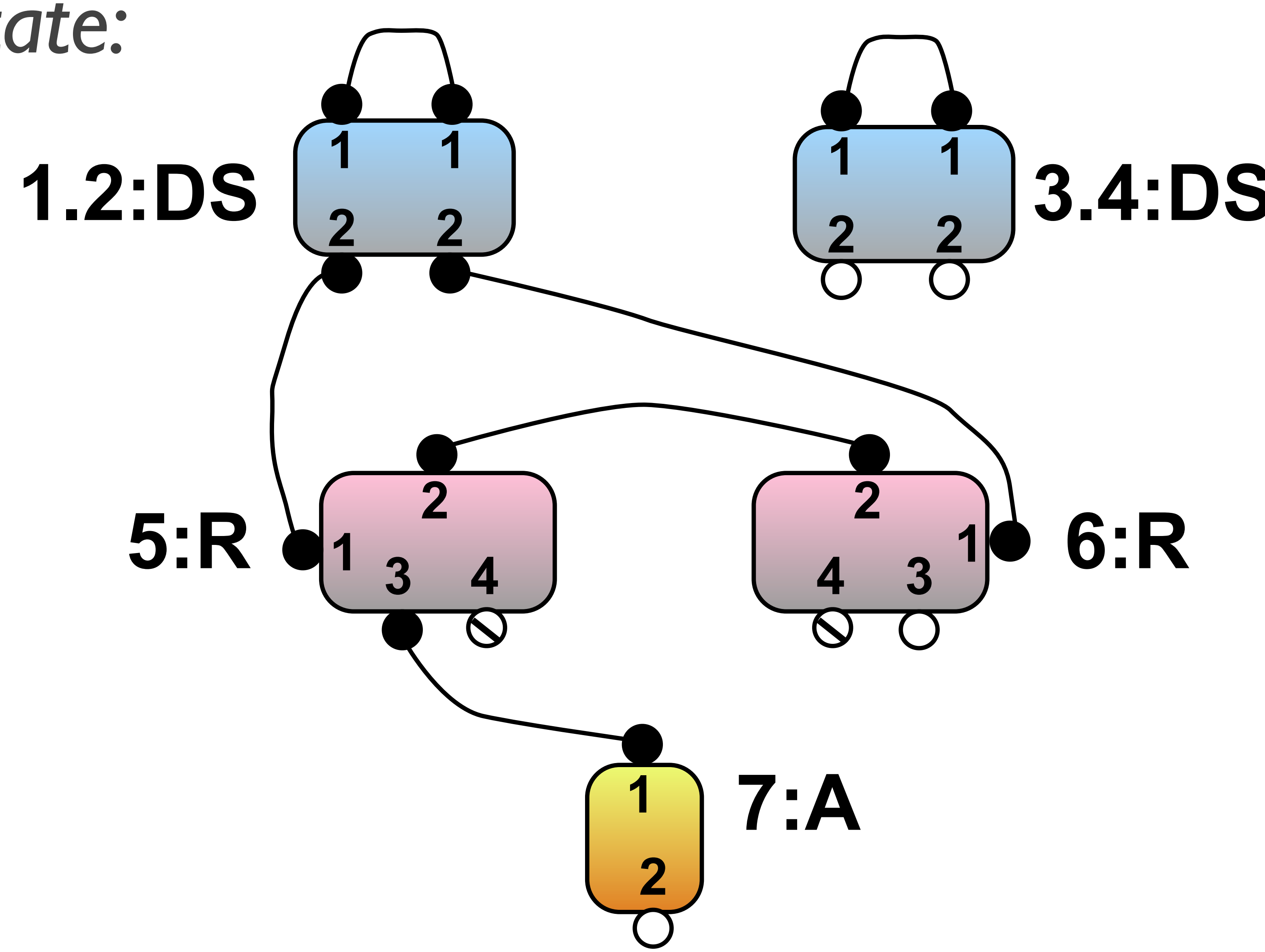


Example: a fragment of the EGFR signaling pathway



Example: a fragment of the EGFR signaling pathway

A stable state:



- 2 x DimerS
- 2 x DSbindsR
- 1 x DimerR
- 2 x ActivateDR
- 1 x DRbindsA

Graph-base approaches

- κ-calculus, Kappa factory [Danos et al.]
- BioNetGen [Hlavacek et al.]
- Pathway Logic [Talcott et al.]

Chemical programming

- γ-calculus = λ-calculus + chemical paradigm [BanatreFR04-07]
- a chemical solution where molecules interact freely according to reaction rules
- everything is a molecule

prod = **replace** *X*, *Y* **by** *X*×*Y*

$\langle prod, 3, 1, 4, 5, 2 \rangle \;\rightarrow\; \langle prod, 1, 4, 15, 2 \rangle \;\rightarrow^{*}\; \langle prod, 120 \rangle$

Rewriting calculus

- extends first-order term rewriting and the λ-calculus [CirsteaK01]
- terms, rules, rule application are explicit objects of the calculus

$$(s(x)+y \rightarrow s(x+y))\ (s(5)+s(2)) \rightarrow_{\rho} s(5+s(2))$$

Biochemical calculus

- add biochemical flavour to the chemical calculus - structures (like port graphs)
- rewrite rules and strategies
- verification

Syntax

- objects: port graphs
- rewrite rules
- abstractions
- application

$$
\begin{array}{rl}
\textbf{(Objects)} & \mathcal{O} ::= \mathcal{OBJ} \mid \mathcal{X} \mid \mathcal{O}{\bullet}\mathcal{O} \\
\textit{(Rule)} & \mathcal{R} ::= \mathcal{O} \Rightarrow \mathcal{O} \\
\textbf{(Molecule)} & \mathcal{M} ::= \mathcal{O} \mid \mathcal{R} \mid \mathcal{M}{\bullet}\mathcal{M} \\
\textit{(Abstraction)} & \mathcal{A} ::= \mathcal{M} \Rrightarrow \mathcal{M} \\
\textbf{(Configuration)} & \mathcal{K} ::= \mathcal{M} \mid \mathcal{A} \mid \mathcal{K}{\bullet}\mathcal{K} \\
\textbf{(System)} & \mathcal{S} ::= [\mathcal{K}]
\end{array}
$$

Semantics

$$\boxed{\textbf{(Interaction)} \;\; [K \bullet (M \Rrightarrow N) \bullet M'] \longrightarrow_i [K \bullet \varsigma(N)] \quad \textbf{if } \varsigma \in \mathcal{S}ol(M \ll M')}$$

More control?

Strategies:

- enforce confluence and termination
- provide control over the composition or choice of the abstraction to apply

★ *Identity, Failure, Sequence, Not, First, Repeat...*

$$\mathit{First}(S_1,S_2)(G) = \begin{cases} S_1(G) & \text{if } S_1 \text{ does not fail,} \\ S_2(G) & \text{otherwise} \end{cases}$$

Strategies-based extensions

- tackling application failure

$$\boxed{(\textbf{InteractionR})\quad [K \bullet T \bullet M] \longrightarrow_{ir} [K \bullet \mathtt{seq}(T, \mathtt{try}(\mathsf{stk} \Rrightarrow T \bullet M))@M]}$$

✦ persistent strategies *S!*

Invariant verification

- invariant:
  - rule $G \Rightarrow G$
  - strategy `first`($G \Rightarrow G$, $X \Rightarrow$ *"Failure"*)!
- remove ($G \Rightarrow$ *"Failure"*)! or "repair" ($G \Rightarrow H$)!
- but we can do more...

Structural Formulas

Structural Formulas

Structural formulas:

$$\varphi ::= \top \mid \bot \mid \gamma \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \rightarrow \varphi_2 \mid \lozenge\varphi$$

Structural Formulas

Structural formulas:

$$\varphi ::= \top \mid \bot \mid \gamma \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \rightarrow \varphi_2 \mid \diamondsuit\varphi$$

Satisfaction relation:

$$\begin{array}{lcl} G \models \gamma & \Leftrightarrow & \exists\sigma \text{ such that } G = \sigma(\gamma) \\ G \models \diamondsuit\varphi & \Leftrightarrow & \exists G' \sqsubseteq G \text{ such that } G' \models \varphi \end{array}$$

Mapping Structural Formulas to Strategies

$$\begin{array}{lcl}
\tau(\top) & = & \mathsf{id} \\
\tau(\bot) & = & \mathtt{fail} \\
\tau(\lozenge\gamma) & = & \gamma \Rightarrow \gamma \\
\tau(\neg\varphi) & = & \mathtt{not}(\tau(\varphi)) \\
\tau(\varphi_1 \wedge \varphi_2) & = & \mathtt{seq}(\tau(\varphi_1), \tau(\varphi_2)) \\
\tau(\varphi_1 \vee \varphi_2) & = & \mathtt{first}(\tau(\varphi_1), \tau(\varphi_2)) \\
\tau(\varphi_1 \rightarrow \varphi_2) & = & X \Rightarrow \mathtt{seq}(\tau(\varphi_1), \mathtt{first}(\mathsf{stk} \Rightarrow X, \tau(\varphi_2)))\mathtt{@}X
\end{array}$$

$$G \models \varphi \textit{ if and only if } \tau(\varphi)@G \longrightarrow^* G$$

$$G \not\models \varphi \textit{ if and only if } \tau(\varphi)@G \longrightarrow^* \mathsf{stk}$$

Guarded systems

- define a new reduction relation

$$[K]_\varphi \Longmapsto [K']_\varphi \text{ if } [K] \Mapsto [K'] \text{ and } K' \models \varphi$$

- use strategies

$$[K]_\varphi \Longmapsto \texttt{ifThenElse}(\tau(\varphi), X_1 \Rrightarrow [K']_\varphi, X_2 \Rrightarrow \text{error\_message})@K'$$

$$\text{if } [K] \Mapsto [K']$$

Example: $\varphi = \neg \mathit{Virus} \vee (\mathit{Virus} \wedge \mathit{Antiviral})$

Conclusions (first part)

- port graphs: a biologically-inspired graphical structure
- biochemical calculus: a higher-order rule-based formalism
- verification of invariant properties
- applications to protein-protein interactions and autonomic systems

Future work

- embed runtime verification
  - diagnose faults at execution
  - repair faults (adaptive behaviour)
  - identify properties to monitor
  - choose temporal logic: LTL_3 ($\top$, $\bot$, ?)
- add a stochastic semantics
- robustness analysis

Abstractions for continuous-time Markov chains

CTMCs

- state-based formalisms for describing dynamic systems: $C = (S, s_0, R, L)$
- discrete steps, continuous time-steps
- suitable for modelling signalling pathways: stochastic, computational, concurrent

CTMCs with levels

- population (species) based modelling
- discrete levels of concentrations
  - maximum molar concentration M
  - choose N as granularity for the abstraction, concentration step size H = M/N
  - 0,1, ...,N levels of concentrations correspond to 0, $(0,H]$, $(H,2*H]$,..., $((N-1)*H, N*H]$

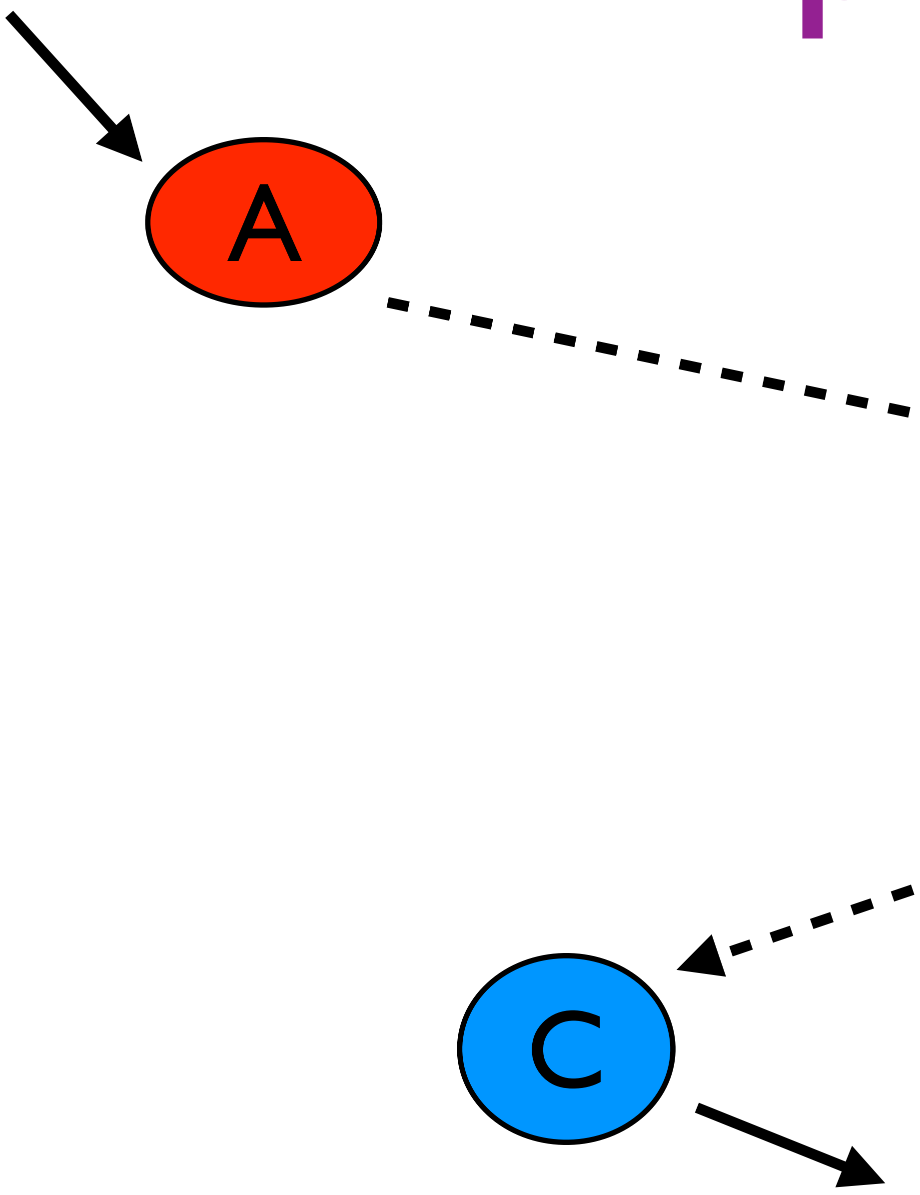
CTMCs with levels

- mass-action kinetics
- reaction $A + B \rightarrow C$ with k constant rate
- transition rate: $k * (L_A * H) * (L_B * H)/H$

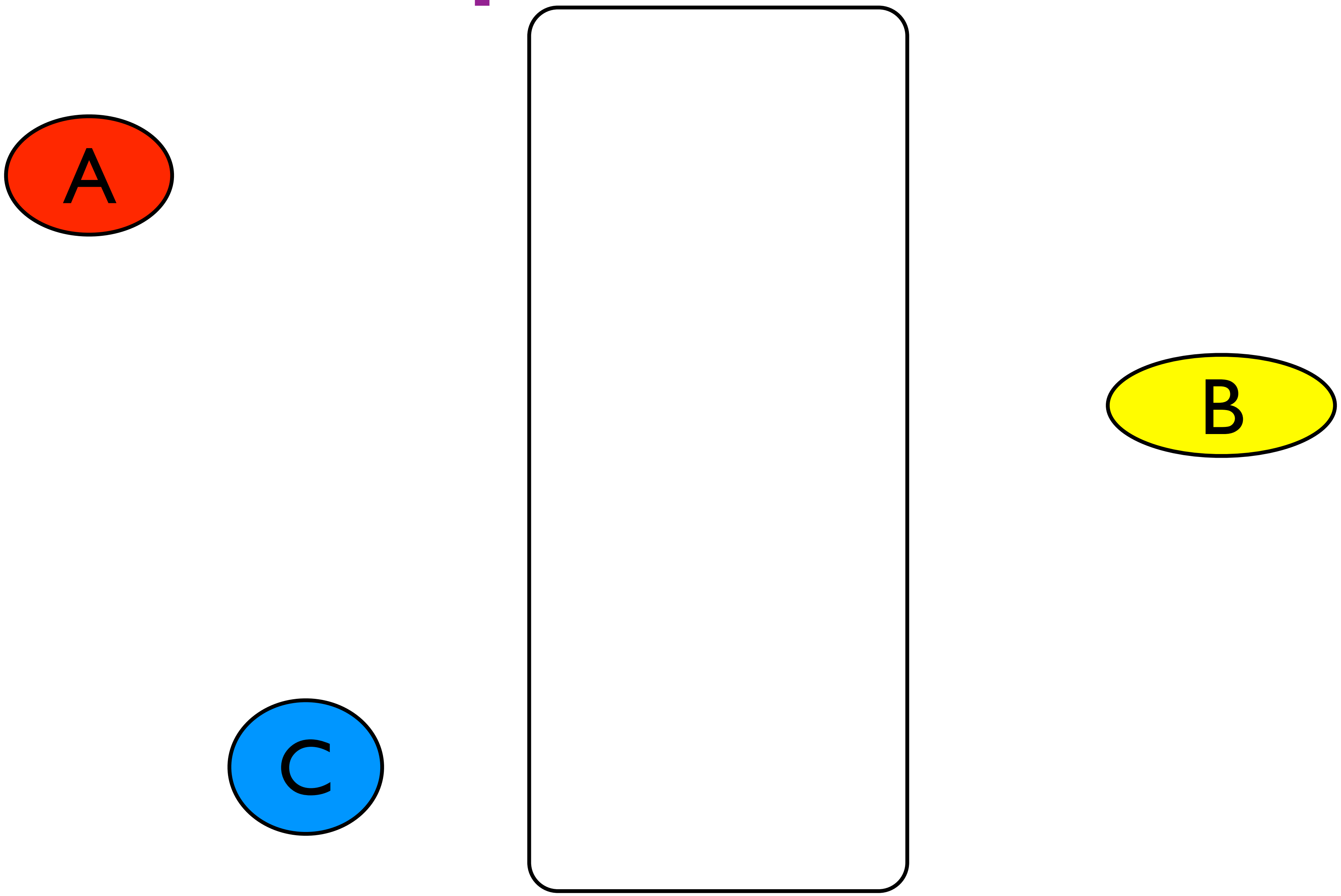
CTMCs with levels

- mass-action kinetics
- reaction $A + B \rightarrow C$ with k constant rate
- transition rate: $k * (L_A * H) * (L_B * H) / H$
- let's see a real example ...

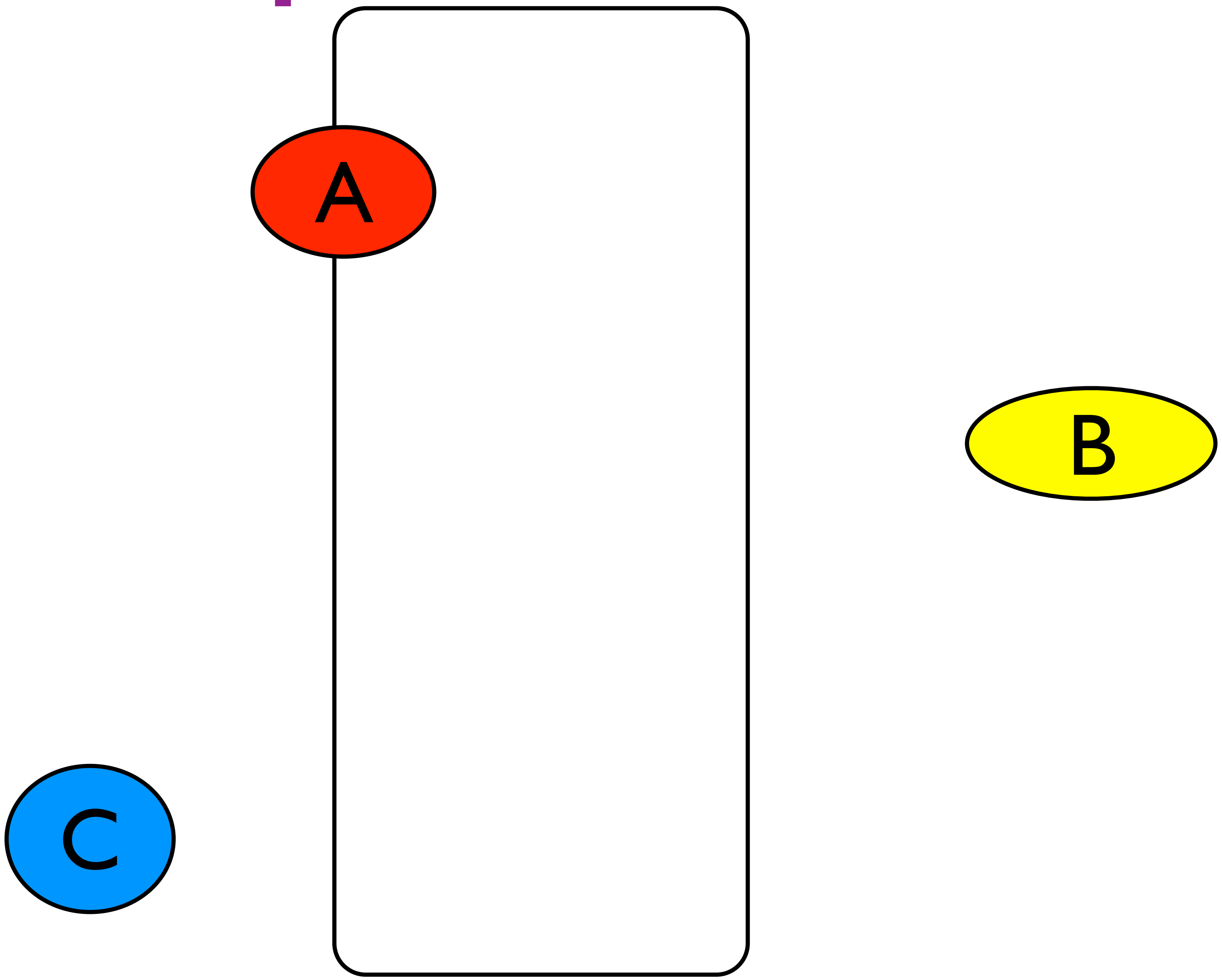
Signalling and scaffold proteins



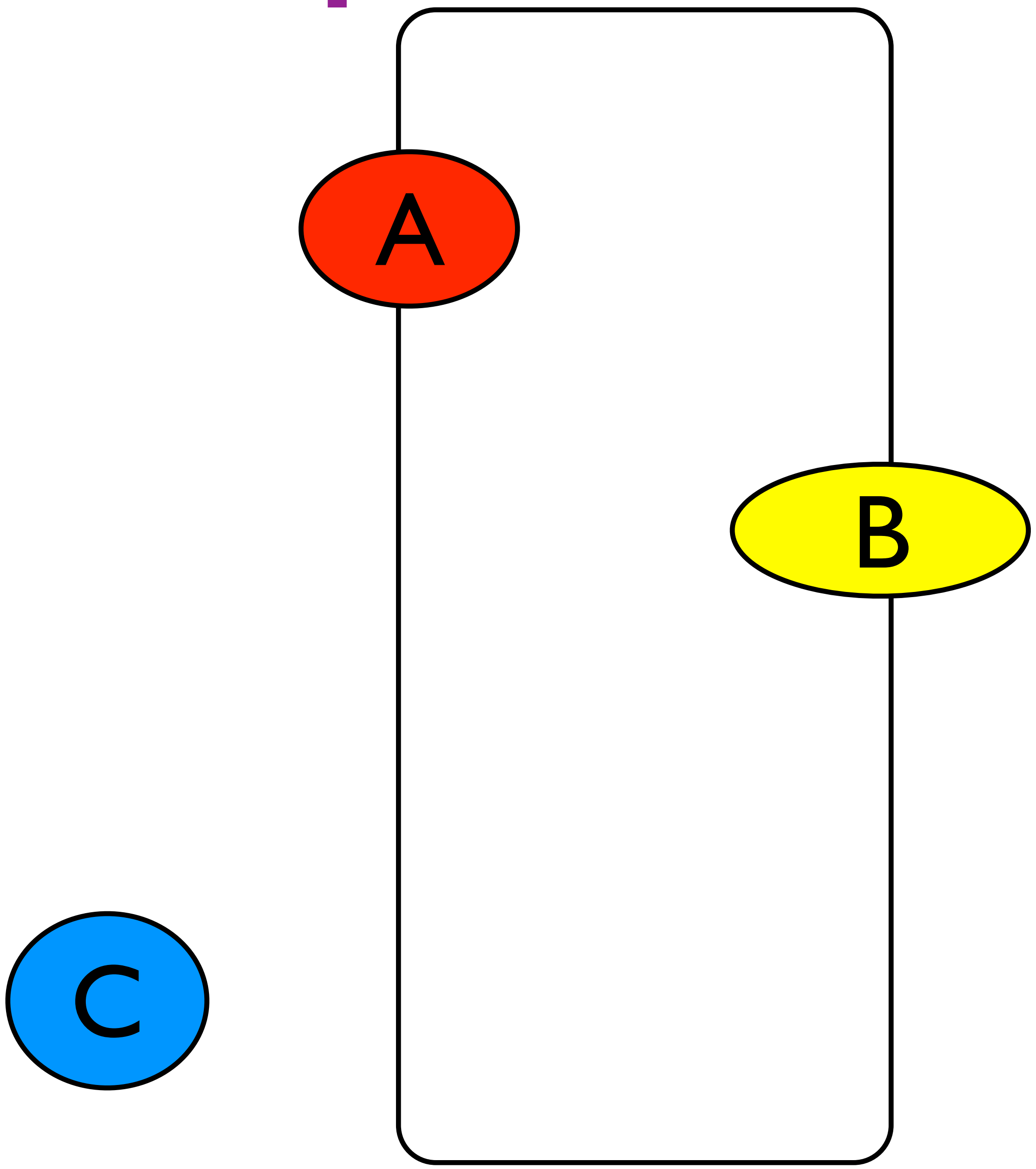
Signalling and scaffold proteins



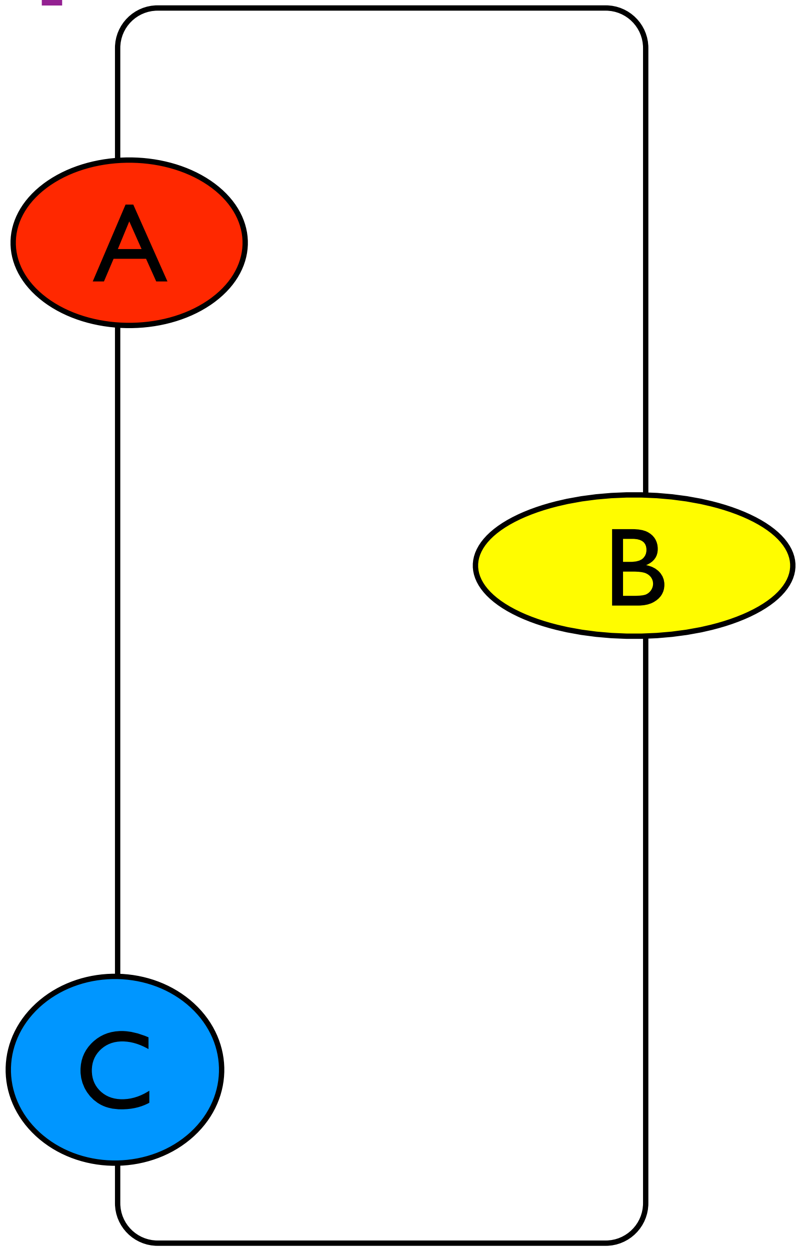
Signalling and scaffold proteins



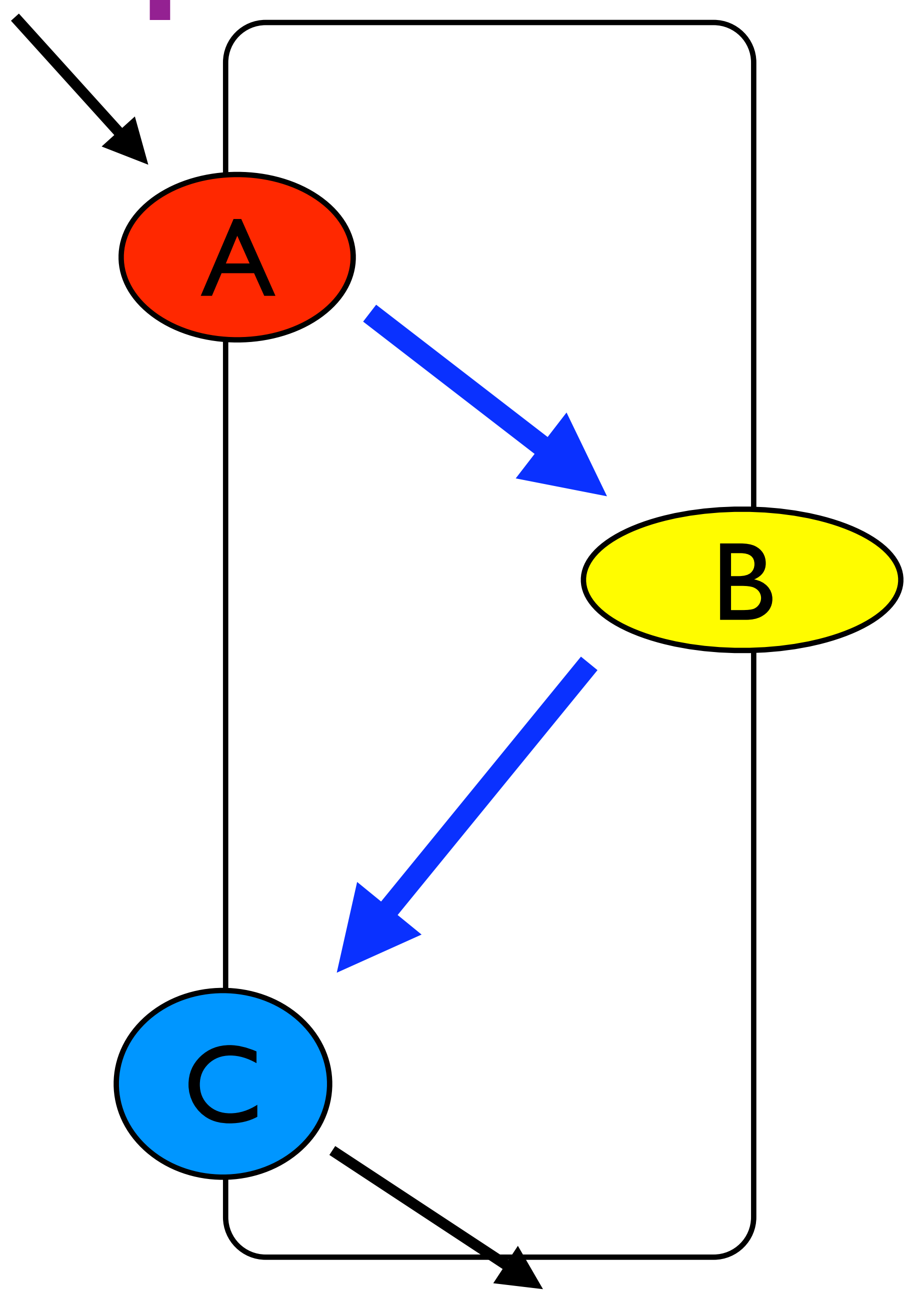
Signalling and scaffold proteins



Signalling and scaffold proteins

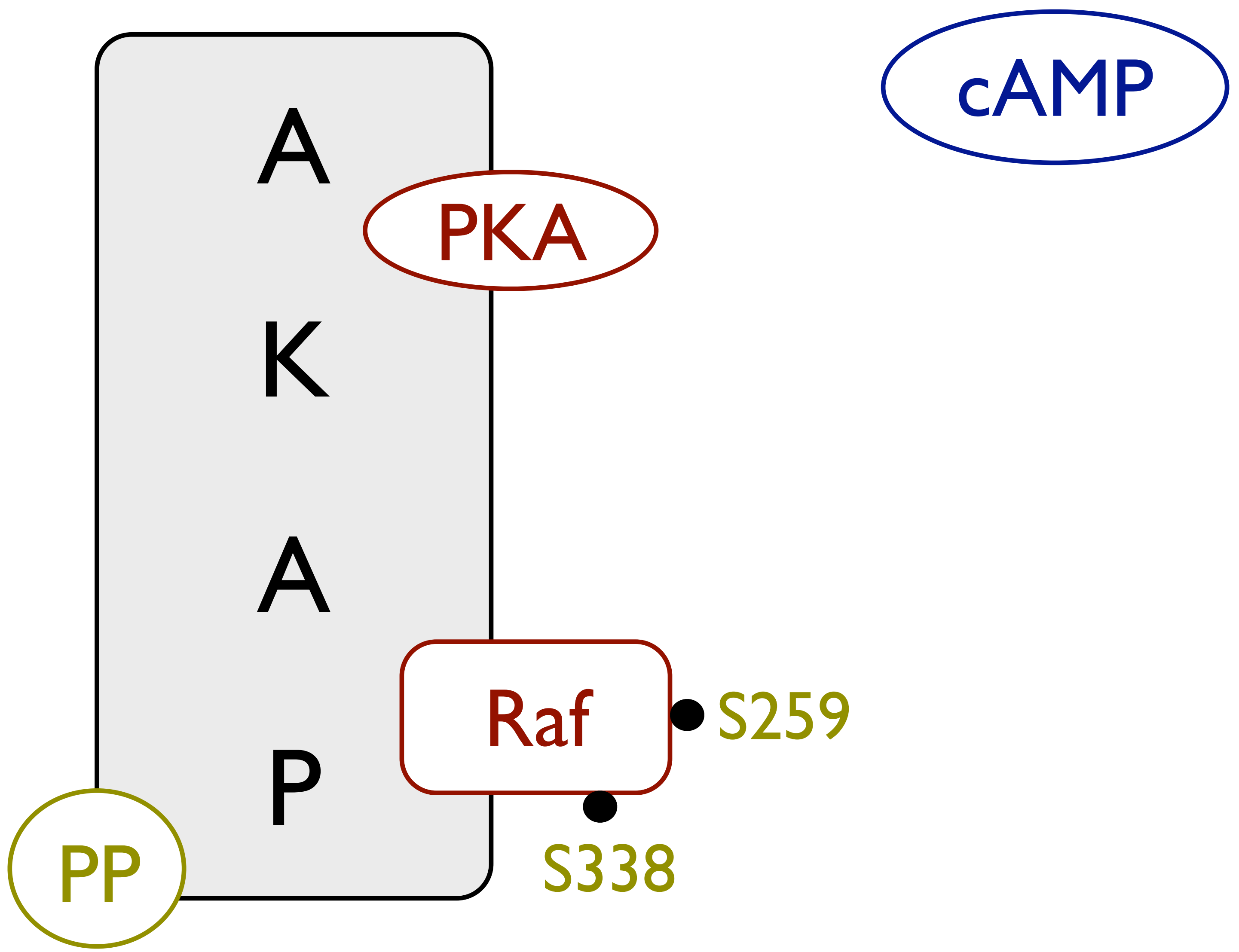


Signalling and scaffold proteins



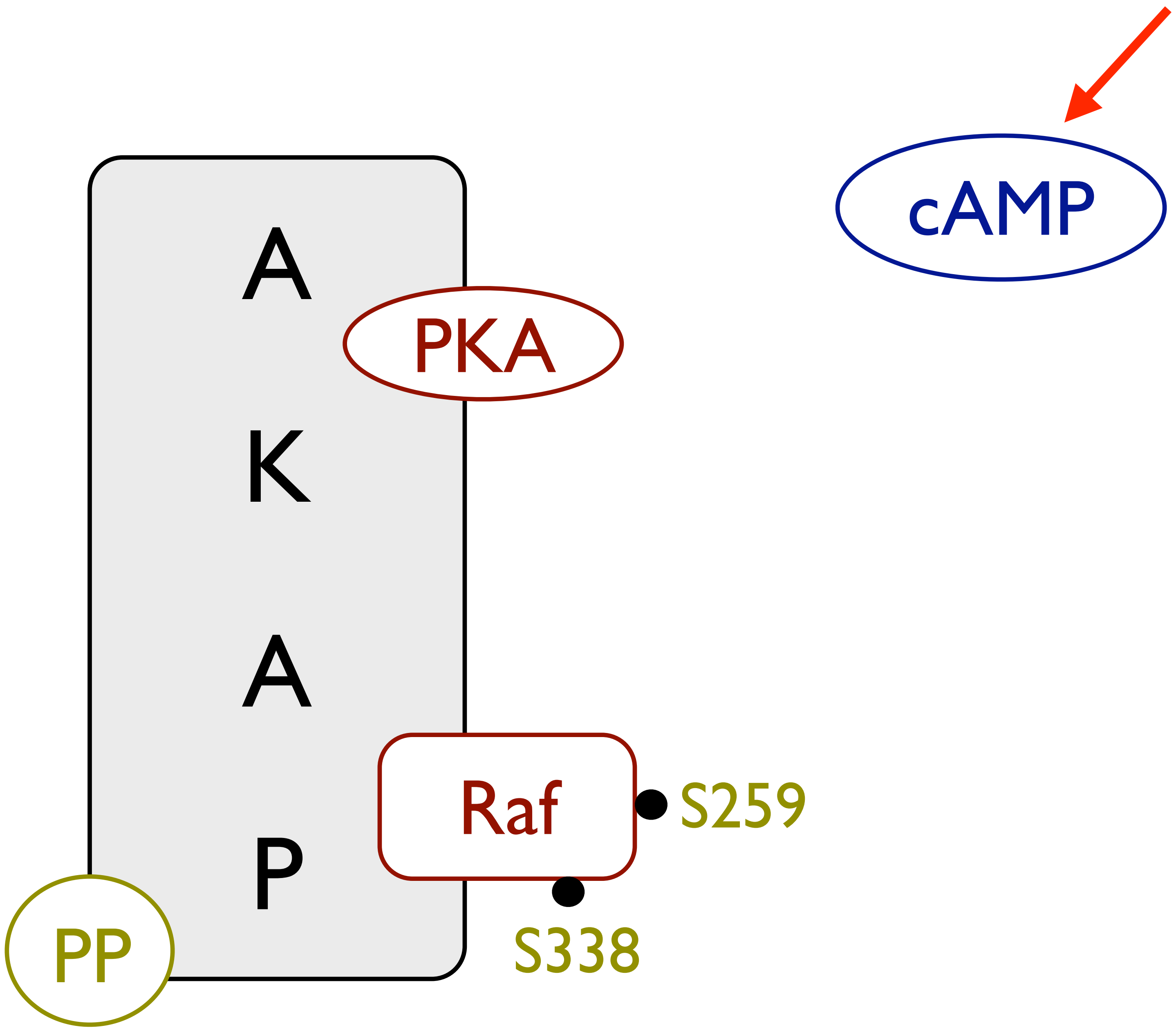
AKAP

(A-kinase anchoring protein)



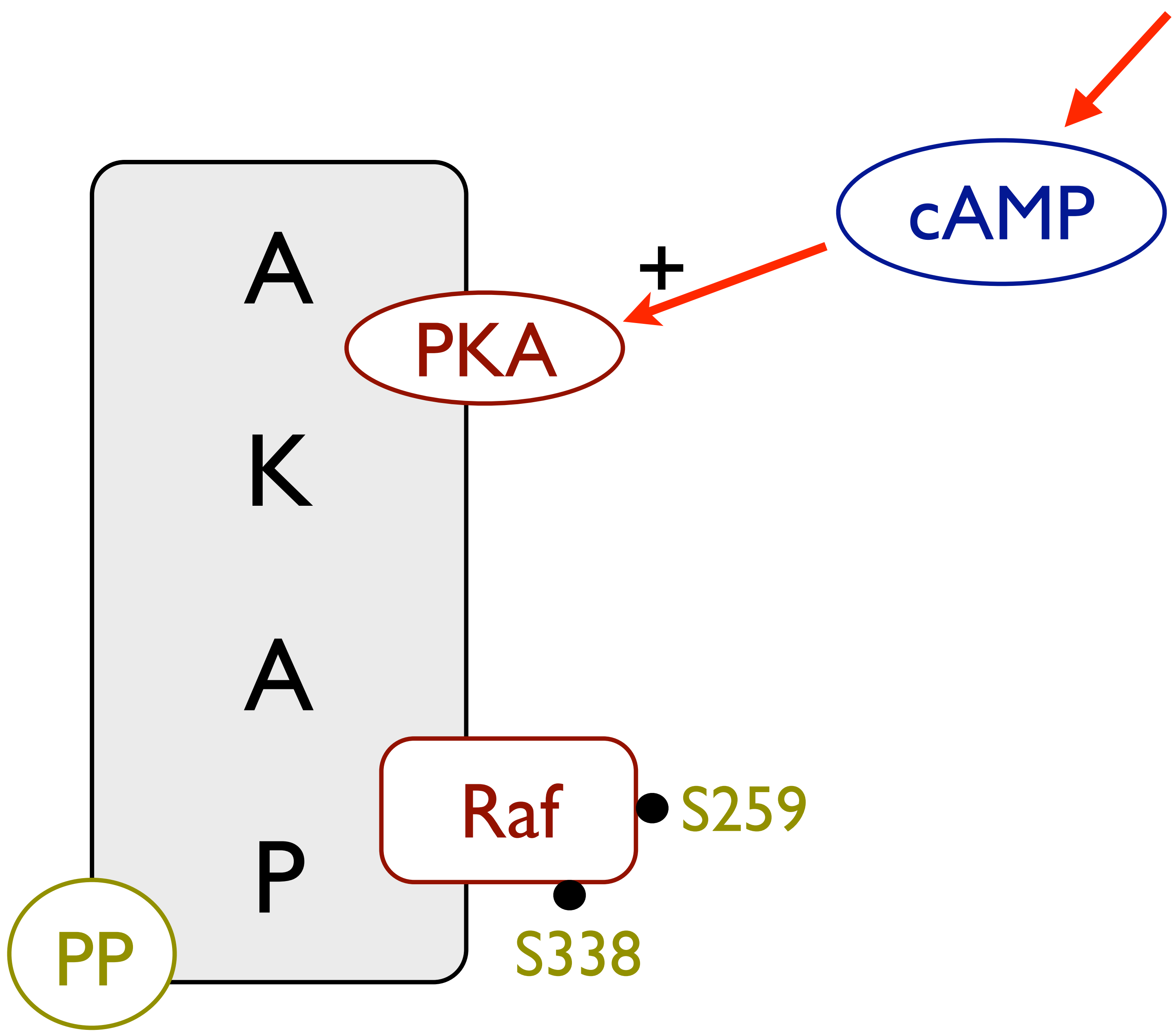
AKAP

(A-kinase anchoring protein)



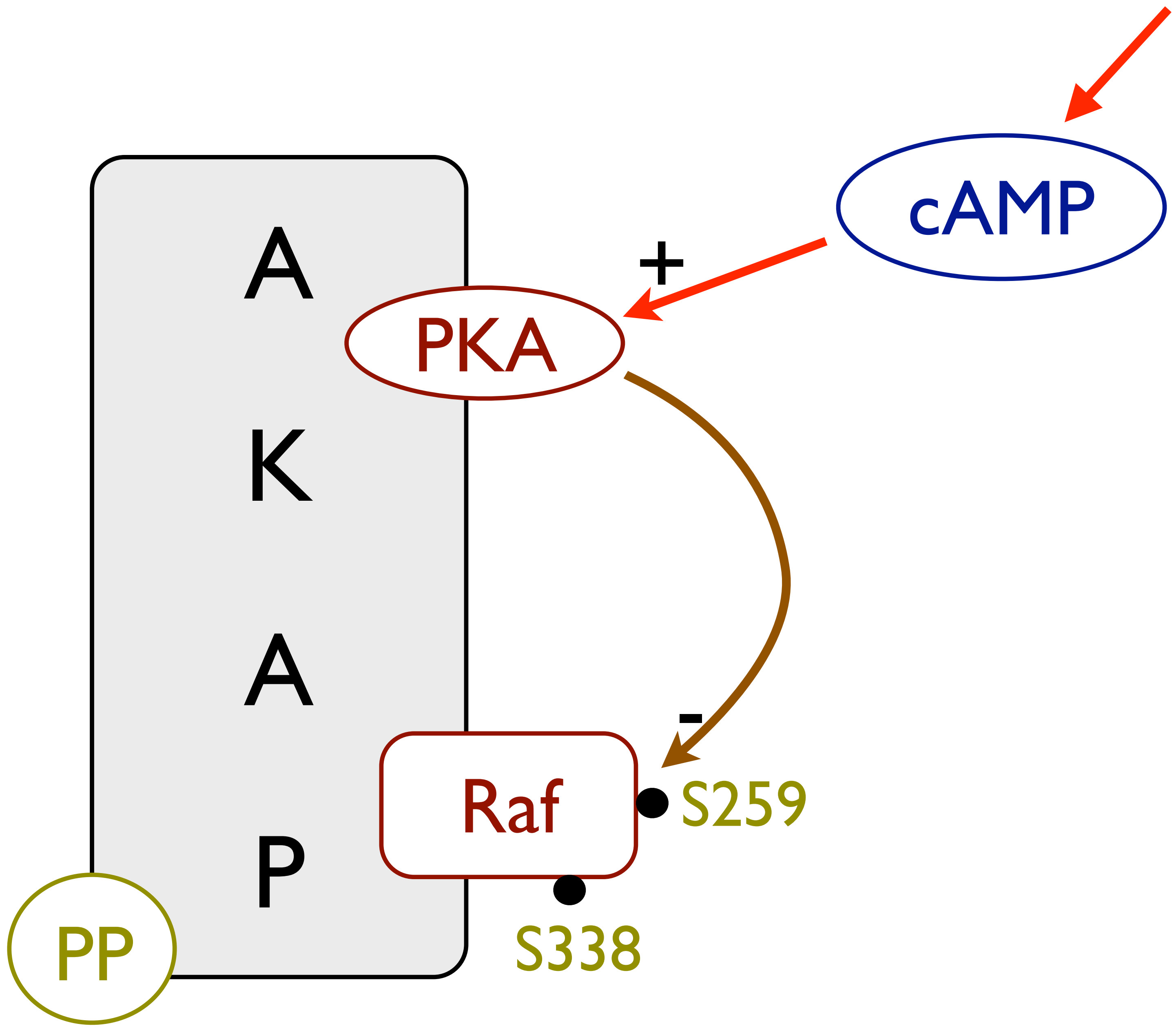
AKAP

(A-kinase anchoring protein)

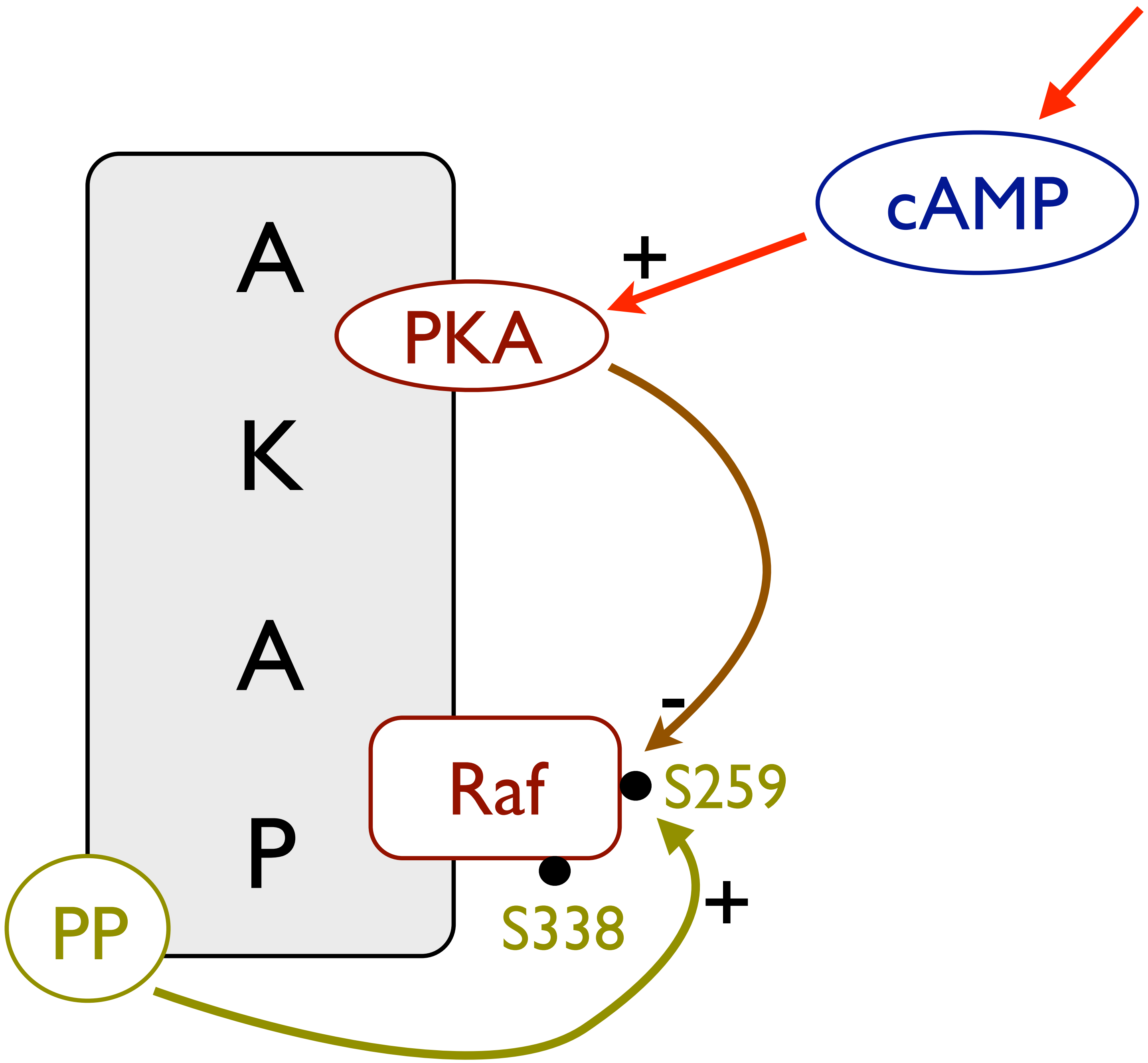


AKAP

(A-kinase anchoring protein)

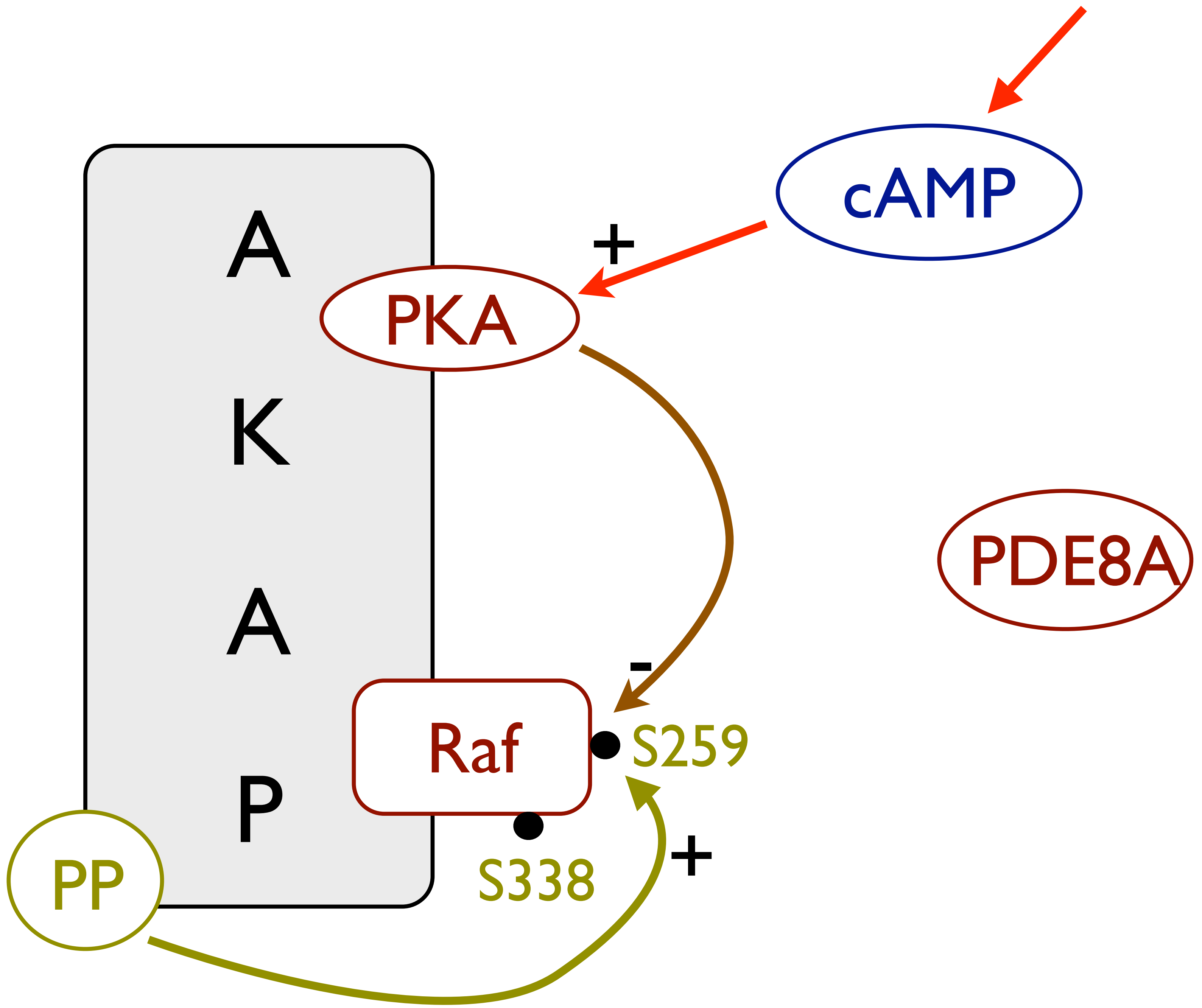


AKAP
(A-kinase anchoring protein)



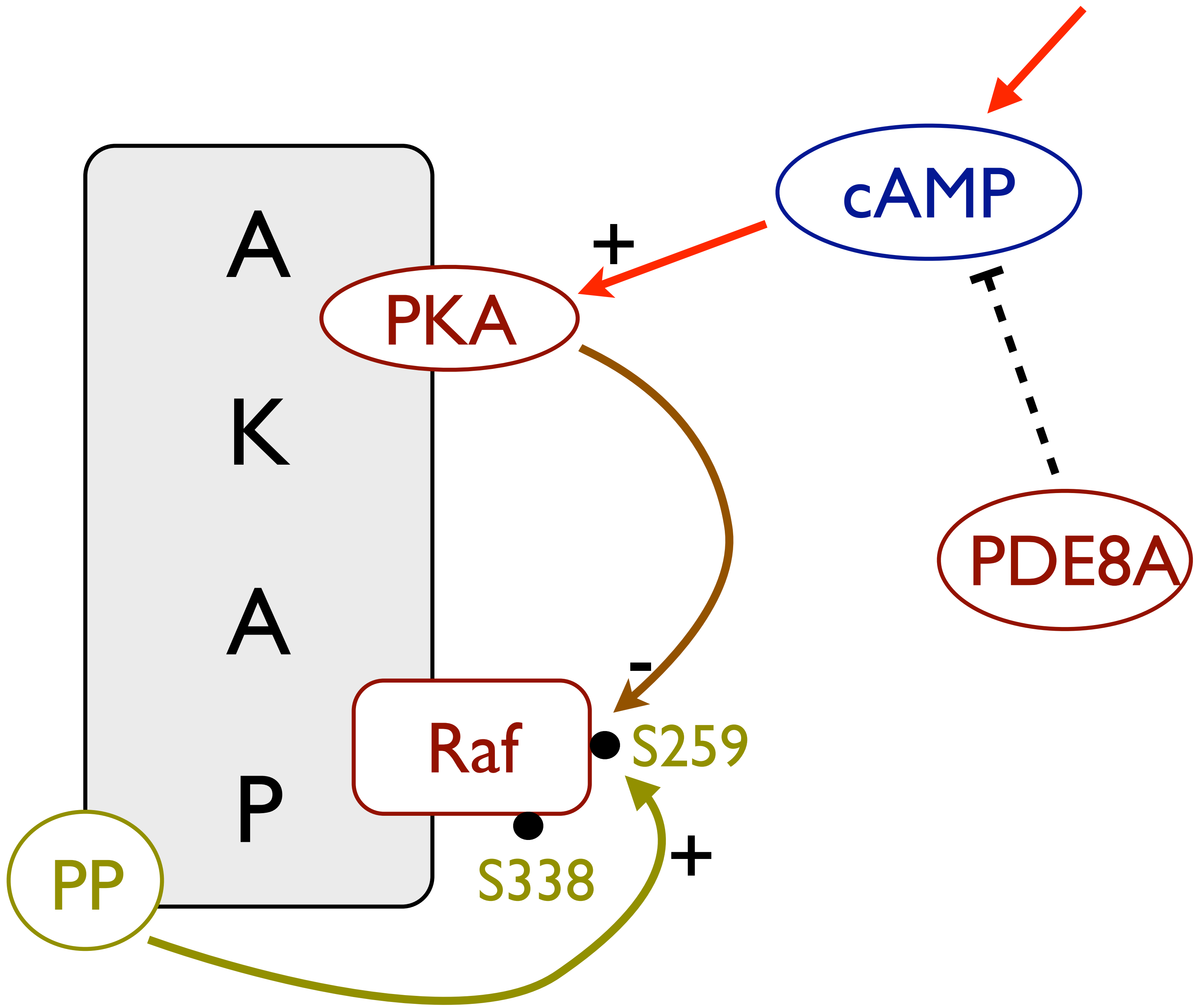
AKAP

(A-kinase anchoring protein)



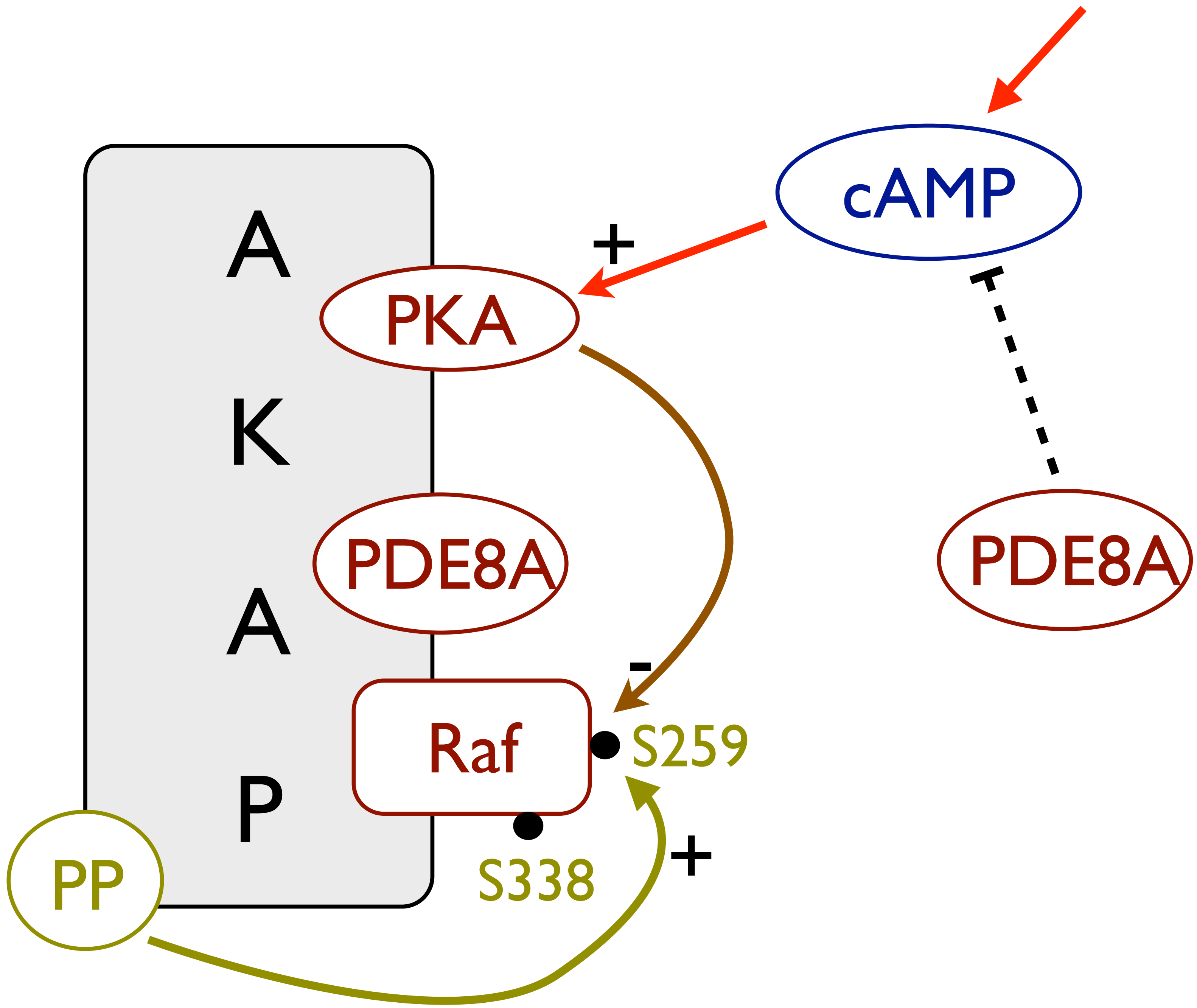
AKAP

(A-kinase anchoring protein)



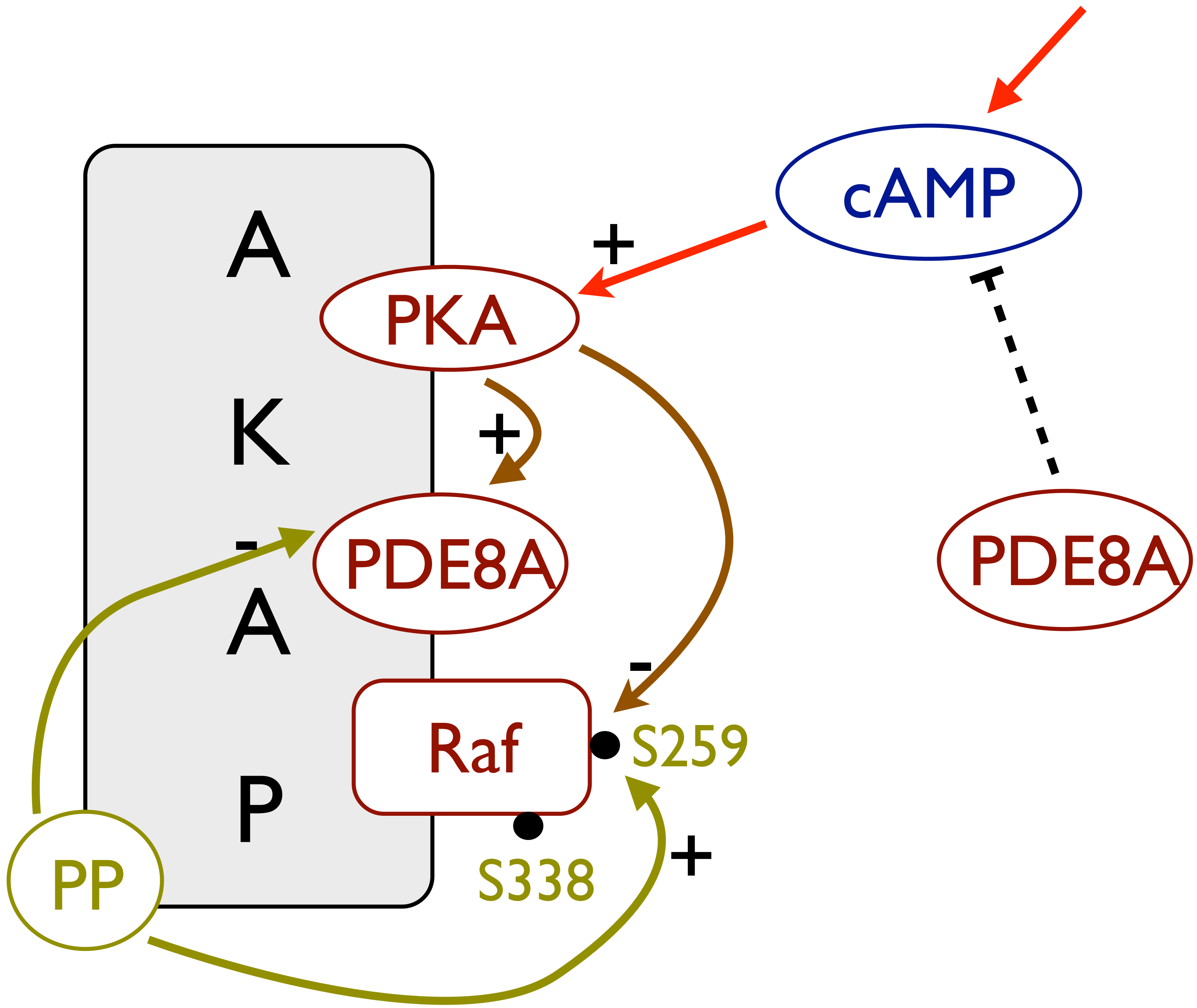
AKAP

(A-kinase anchoring protein)



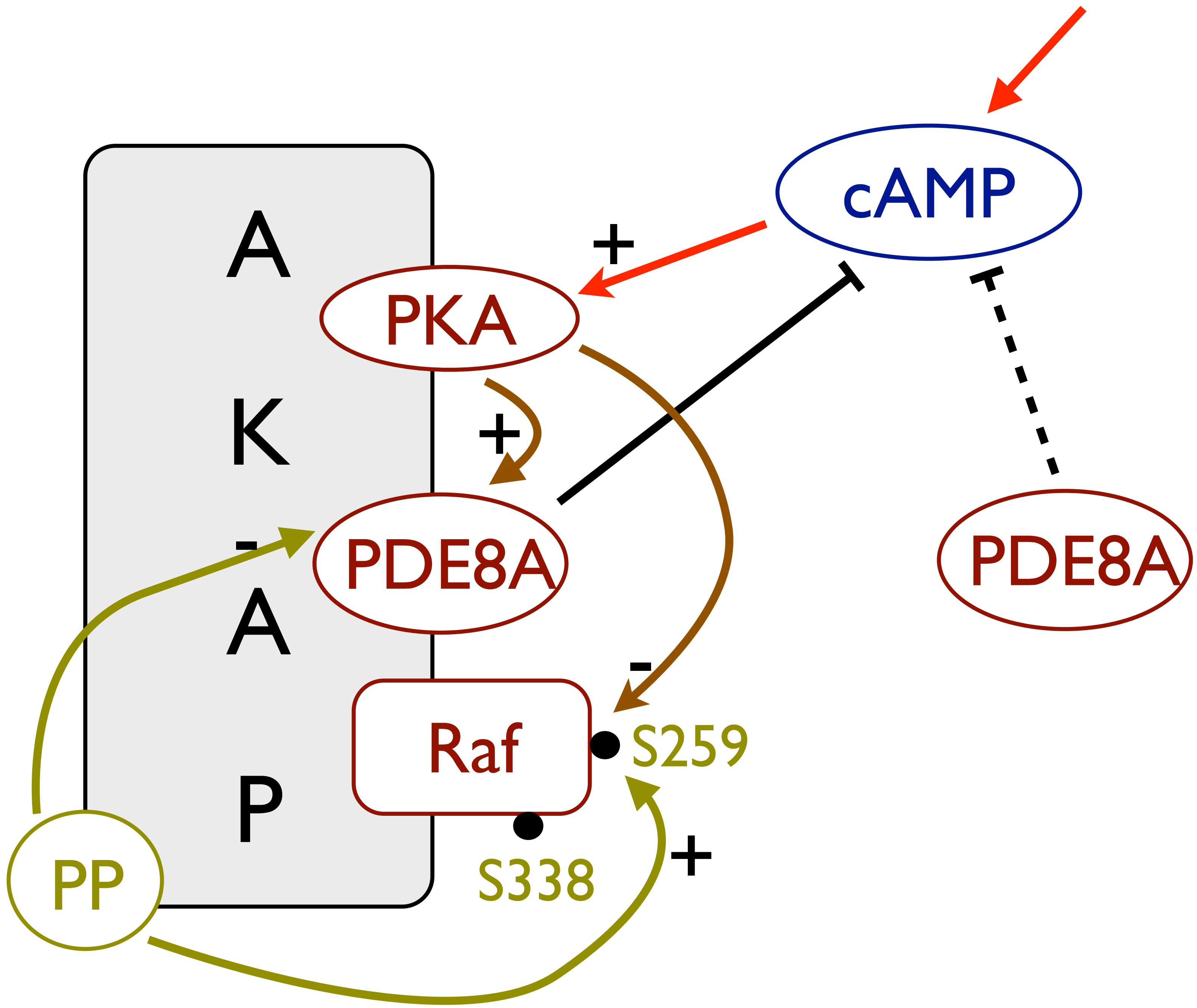
AKAP

(A-kinase anchoring protein)



AKAP

(A-kinase anchoring protein)



Expected behaviour

Q1: ↑pPDE8A1 ➡ ↓cAMP ➡ ↓PKA^{+} ➡ ↑ Raf activity
➡ ↓ $pRaf_{S259}$

What is the time relation or causality between events?

Expected behaviour

Q_1: ↑pPDE8A1 ➡ ↓cAMP ➡ ↓PKA^+ ➡ ↑ Raf activity
➡ ↓ $pRaf_{S259}$

What is the time relation or causality between events?

Q_2: Pulsating behaviour

Formal model

- continuous time Markov chains with levels
- properties expressed as formulas in Continuous Stochastic Logic (CSL)
- symbolic probabilistic model checker PRISM

PRISM model

- modules for cAMP, scaffold, free PDE8A1, PP
- mass action kinetics
- information on constant rates ratios

PRISM model

The PKA activation reaction S000 + cAMP $\rightarrow_{r2}$ S100 is modeled as follows:

- in the module for cAMP:

```
[activate_PKA] (cAMP > basal_camp) -> (cAMP) : (cAMP' = cAMP-1);
```

- in the module for the scaffold:

```
[activate_PKA] (S000 > 0) & (S100 < scaffold_max) ->
               (r2*S000) : (S100' = S100+1) & (S000' = S000-1);
```

Continuous Stochastic Logic

- extension of non-probabilistic CTL
- probability operator P
- steady-state operator S

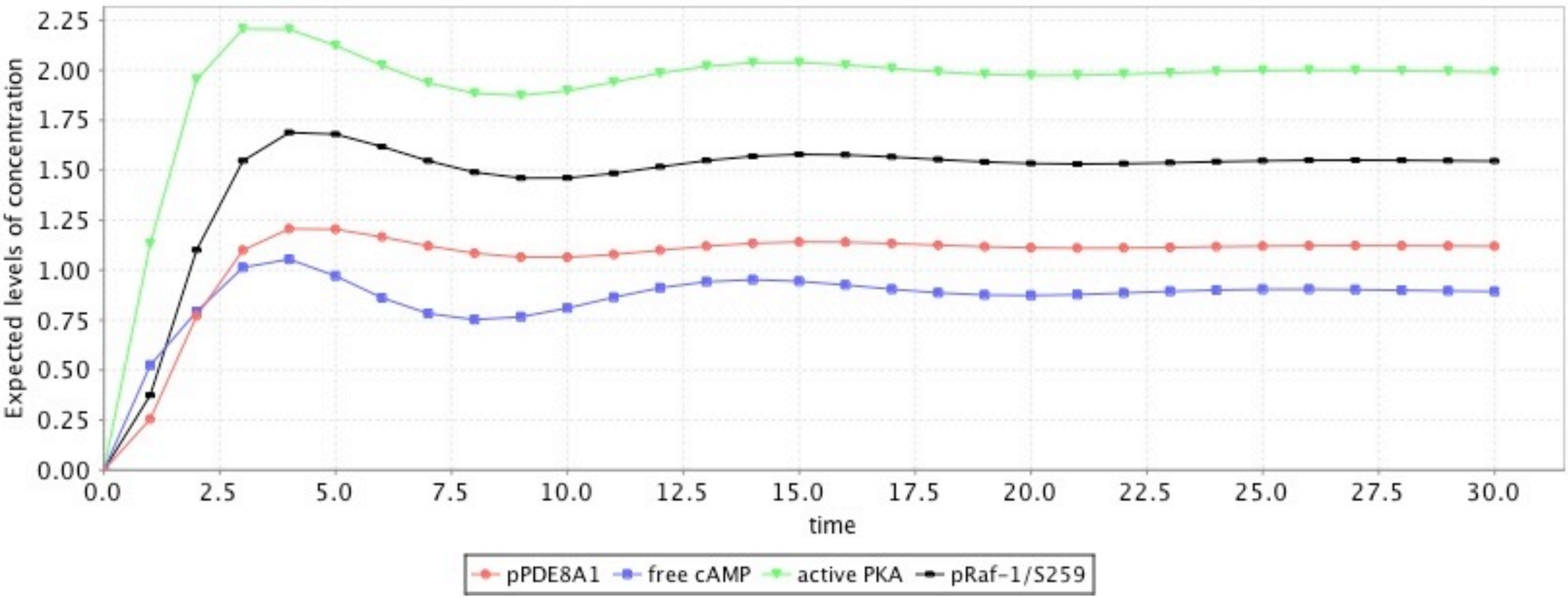
State formulae	$\Phi ::= \top \mid a \mid \neg\Phi \mid \Phi \wedge \Phi \mid \mathsf{P}_{\bowtie p}[\phi] \mid \mathsf{S}_{\bowtie p}[\Phi]$
Path formulae	$\phi ::= \mathsf{X}\,\Phi \mid \Phi\,\mathsf{U}^I\,\Phi$

Reward-based properties

- use of rewards (or costs) in CSL
  - real values assigned to states or transitions
  - to track variable values in states
  - to compute the expected value of a variable at a given time

Reward-based properties

- state rewards for computing the expected levels for cAMP, pPDE8A1, PKA$^+$, pS259



Trend Variables

- keep track of decreasing or increasing variable values
- define new variables in the PRISM modules for cAMP, PKA^+ and pS259

  cAMP' = cAMP-1 & trend_cAMP' = -1

- ↓x (↑x) ascending (descending) trend for variable x

Necessarily Preceded

[Monteiro et al. 08]

For $\varphi = \downarrow cAMP \wedge \downarrow PKA^{+}$ and $\psi = \uparrow pPDE8A1$

CTL: $(EF\ \varphi) \wedge AG((\neg\psi) \Rightarrow AG(\neg\ \varphi))$

CSL: $P_{>0}[F\ \varphi] \wedge P_{\leq 0}[F(\neg((\neg\ \psi) \Rightarrow P_{\geq 1}[F(\neg\ \varphi)]))]$

Pulsations

Show that the levels of pPDE8A1 fluctuate:

- $\varphi = \uparrow$pPDE8A1 and $\psi = \downarrow$pPDE8A1
- pulsation in CTL [Fages05,Ballarini et al. 09]:

$$AG((\varphi \Rightarrow EF\psi) \wedge (\psi \Rightarrow EF\varphi))$$

- pulsation in CSL:

$$P_{\leq 0}[F\ (\neg(\varphi \Rightarrow P_{>0}[F\psi]) \vee \neg(\psi \Rightarrow P_{>0}[F\varphi])$$

Pulsations

- for cAMP: $\varphi = \uparrow$cAMP and $\psi = \downarrow$cAMP
- for PKA^+: $\varphi = \uparrow PKA^+$ and $\psi = \downarrow PKA^+$
- coordinated pulsations:

$\varphi = \uparrow pPDE8A1 \wedge \downarrow cAMP \wedge \downarrow PKA^+$ and

$\psi = \downarrow pPDE8A1 \wedge \uparrow cAMP \wedge \uparrow PKA^+$

Overview AKAP

Overview AKAP

- ☑ formal model of a biological process

Overview AKAP

- ☑ formal model of a biological process
- ☑ the biologists validated our results

Overview AKAP

- ☑ formal model of a biological process
- ☑ the biologists validated our results
- ☐ refine the model with more experimental data

Overview AKAP

- ☑ formal model of a biological process
- ☑ the biologists validated our results
- ☐ refine the model with more experimental data
- ☐ trend variables, amplitude of oscillations

Overview AKAP

- ☑ formal model of a biological process
- ☑ the biologists validated our results
- ☐ refine the model with more experimental data
- ☐ trend variables, amplitude of oscillations
- ☐ formulate new properties and express them using a temporal logic

Abstractions for CTMCs with levels

- relation between C^{kN} and C^{N} for $k \geq 2$, $N \geq 4$
- aim: preserve temporal properties
- if C^{N} satisfies temporal formula φ, then C^{kN} satisfies $f(\varphi)$ - who is f ?
- (weak) simulation does not work...

Temporal properties

- classification of temporal properties for signalling pathways
- BIOCHAM [Fages et al.]
- patterns [Monteiro et al.08]
- stochastic models, not only qualitative or probabilistic

Temporal properties

- is CSL expressive enough?
- what about LTL(R) ? [Fages et al.]
- linear versus branching time for biologists?
- satisfaction probabilities for biologists?

Acknowledgement

- Hélène Kirchner and Horatiu Cirstea
- Muffy Calder and SIGNAL project
- Walter Kolch, George Baillie and Kim Brown from the Faculty of Biomedical & Life Science, University of Glasgow

Thank you!
Questions?

Bibliography

- **[Monteiro et al. 08]** Pedro T. Monteiro, Delphine Ropers, Radu Mateescu, Ana T. Freitas, and Hidde de Jong. *Temporal logic patterns for querying dynamic models of cellular interaction networks.* Bioinformatics, 24(16):227--233, 2008.
- **[Ballarini et al. 09]** Paolo Ballarini, Radu Mardare, and Ivan Mura. *Analysing Biochemical Oscillation through Probabilistic Model Checking*. ENTCS, 229(1):3--19, 2009
- **[Fages05]** François Fages. *Temporal Logic Constraints in the Biochemical Abstract Machine BIOCHAM.* LOPSTR'05, volume 3901 of LNCS, pages 1--5. Springer, 2005.