Formal modelling of biological systems

Vashti Galpin

Laboratory for Foundations of Computer Science
School of Informatics
University of Edinburgh

(Joint work with Jane Hillston)

5 March 2013
Outline

Process algebras

Bio-PEPA

Semantics

Enzyme example

Hybrid approach

Src trafficking

Conclusions
Process algebras – history

- developed to model concurrent computing/behaviour in mid 1980’s
Process algebras – history

- developed to model concurrent computing/behaviour in mid 1980’s
- three distinct approaches
  - Robin Milner: CCS, operational
  - Tony Hoare: CSP, denotational
  - Bergstra and Klop: ACP, equational and algebraic
Process algebras – history

- developed to model concurrent computing/behaviour in mid 1980’s
- three distinct approaches
  - Robin Milner: CCS, operational
  - Tony Hoare: CSP, denotational
  - Bergstra and Klop: ACP, equational and algebraic
- similar ideas, all compositional
Process algebras – history

► developed to model concurrent computing/behaviour in mid 1980’s
► three distinct approaches
  ► Robin Milner: CCS, operational
  ► Tony Hoare: CSP, denotational
  ► Bergstra and Klop: ACP, equational and algebraic
► similar ideas, all compositional
► compact, elegant formal language

Prefix $a.P$  Choice $P_1 + P_2$  Parallel $P_1 \parallel P_2 \ldots$
Process algebras – history (continued)

➤ operational semantics gives labelled transition system

\[ a.P \xrightarrow{a} P \]
Process algebras – history (continued)

- operational semantics gives labelled transition system

\[ a \cdot P \xrightarrow{a} P \]

\[ P_1 + P_2 \xrightarrow{a} Q \quad \text{whenever} \quad P_1 \xrightarrow{a} Q \]
Process algebras – history (continued)

- operational semantics gives labelled transition system

\[
\begin{align*}
  a.P & \xrightarrow{a} P \\
  P_1 + P_2 & \xrightarrow{a} Q \quad \text{whenever} \quad P_1 \xrightarrow{a} Q \\
  P_1 \parallel P_2 & \xrightarrow{a} Q \parallel P_2 \quad \text{whenever} \quad P_1 \xrightarrow{a} Q
\end{align*}
\]
Process algebras – history (continued)

- operational semantics gives labelled transition system

\[ a.P \xrightarrow{a} P \]

\[ P_1 + P_2 \xrightarrow{a} Q \text{ whenever } P_1 \xrightarrow{a} Q \]

\[ P_1 \parallel P_2 \xrightarrow{a} Q \parallel P_2 \text{ whenever } P_1 \xrightarrow{a} Q \]

\[ P_1 \parallel P_2 \xrightarrow{f(a_1,a_2)} Q_1 \parallel Q_2 \text{ whenever } P_1 \xrightarrow{a_1} Q_1, P_2 \xrightarrow{a_2} Q_2 \]
Process algebras – history (continued)

- operational semantics gives labelled transition system
  
  \[ a.P \overset{a}{\rightarrow} P \]

  \[ P_1 + P_2 \overset{a}{\rightarrow} Q \quad \text{whenever} \quad P_1 \overset{a}{\rightarrow} Q \]

  \[ P_1 \parallel P_2 \overset{a}{\rightarrow} Q \parallel P_2 \quad \text{whenever} \quad P_1 \overset{a}{\rightarrow} Q \]

  \[ P_1 \parallel P_2 \overset{f(a_1,a_2)}{\rightarrow} Q_1 \parallel Q_2 \quad \text{whenever} \quad P_1 \overset{a_1}{\rightarrow} Q_1, P_2 \overset{a_2}{\rightarrow} Q_2 \]

- equivalences such as trace equivalence and bisimulation
  
  \[ a.b.0 + b.a.0 \sim a.0 \parallel b.0 \]
Process algebras – history (continued)

- operational semantics gives labelled transition system
  \[ a.P \xrightarrow{a} P \]
  \[ P_1 + P_2 \xrightarrow{a} Q \quad \text{whenever} \quad P_1 \xrightarrow{a} Q \]
  \[ P_1 \parallel P_2 \xrightarrow{a} Q \parallel P_2 \quad \text{whenever} \quad P_1 \xrightarrow{a} Q \]
  \[ P_1 \parallel P_2 \xrightarrow{f(a_1,a_2)} Q_1 \parallel Q_2 \quad \text{whenever} \quad P_1 \xrightarrow{a_1} Q_1, P_2 \xrightarrow{a_2} Q_2 \]

- equivalences such as trace equivalence and bisimulation
  \[ a.b.0 + b.a.0 \sim a.0 \parallel b.0 \]

- congruence results support compositionality
  \[ P_1 \sim P_2 \quad \text{implies} \quad P_1 \parallel R \sim P_2 \parallel R \]
Process algebras – history (continued)

- Operational semantics gives labelled transition system

\[ a.P \xrightarrow{a} P \]
\[ P_1 + P_2 \xrightarrow{a} Q \quad \text{whenever} \quad P_1 \xrightarrow{a} Q \]
\[ P_1 \parallel P_2 \xrightarrow{a} Q \parallel P_2 \quad \text{whenever} \quad P_1 \xrightarrow{a} Q \]
\[ P_1 \parallel P_2 \xrightarrow{f(a_1,a_2)} Q_1 \parallel Q_2 \quad \text{whenever} \quad P_1 \xrightarrow{a_1} Q_1, P_2 \xrightarrow{a_2} Q_2 \]

- Equivalences such as trace equivalence and bisimulation

\[ a.b.0 + b.a.0 \sim a.0 \parallel b.0 \]

- Congruence results support compositionality

\[ P_1 \sim P_2 \quad \text{implies} \quad P_1 \parallel R \sim P_2 \parallel R \]

- No notion of time, only ordering, so various extensions
Process algebras – stochastic

- addition of random time
  - interleaved with actions
  - associated with actions: Prefix \((a, r).P\)
Process algebras – stochastic

- addition of random time
  - interleaved with actions
  - associated with actions: Prefix \((a, r).P\)
- PEPA: developed by Hillston (1996)
Process algebras – stochastic

- addition of random time
  - interleaved with actions
  - associated with actions: Prefix $(a, r).P$
- PEPA: developed by Hillston (1996)
- semantics given as continuous-time Markov chain
Process algebras – stochastic

- addition of random time
  - interleaved with actions
  - associated with actions: Prefix \((a, r).P\)
- PEPA: developed by Hillston (1996)
- semantics given as continuous-time Markov chain
- fluid dynamics: semantics as ordinary differential equations
Process algebras – stochastic

- addition of random time
  - interleaved with actions
  - associated with actions: Prefix \((a, r).P\)
- PEPA: developed by Hillston (1996)
- semantics given as continuous-time Markov chain
- fluid dynamics: semantics as ordinary differential equations
- applied to biological modelling
  - reagent-centric and reaction-centric styles
  - limitations: stoichiometry, functional rates
- Bio-PEPA: developed by Ciocchetta and Hillston (2009)
# Systems biology modelling

- general approach (Regev, Silverman, Shapiro)

<table>
<thead>
<tr>
<th>Concurrency</th>
<th>Molecular biology</th>
<th>Metabolism</th>
<th>Signal transduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concurrent computational processes</td>
<td>molecules</td>
<td>enzymes and metabolites</td>
<td>interacting proteins</td>
</tr>
<tr>
<td>Synchronous communication</td>
<td>molecular interaction</td>
<td>binding and catalysis</td>
<td>binding and catalysis</td>
</tr>
</tbody>
</table>
**Systems biology modelling**

- general approach (Regev, Silverman, Shapiro)

<table>
<thead>
<tr>
<th>Concurrency</th>
<th>Molecular biology</th>
<th>Metabolism</th>
<th>Signal transduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concurrent computational processes</td>
<td>molecules</td>
<td>enzymes and metabolites</td>
<td>interacting proteins</td>
</tr>
<tr>
<td>Synchronous communication</td>
<td>molecular interaction</td>
<td>binding and catalysis</td>
<td>binding and catalysis</td>
</tr>
</tbody>
</table>

- molecules as processes or species as processes?
# Systems biology modelling

- general approach (Regev, Silverman, Shapiro)

<table>
<thead>
<tr>
<th>Concurrency</th>
<th>Molecular biology</th>
<th>Metabolism</th>
<th>Signal transduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concurrent computational processes</td>
<td>molecules</td>
<td>enzymes and metabolites</td>
<td>interacting proteins</td>
</tr>
<tr>
<td>Synchronous communication</td>
<td>molecular interaction</td>
<td>binding and catalysis</td>
<td>binding and catalysis</td>
</tr>
</tbody>
</table>

- molecules as processes or species as processes?
- stochastic model or deterministic model?
Systems biology modelling

- general approach (Regev, Silverman, Shapiro)

<table>
<thead>
<tr>
<th>Concurrency</th>
<th>Molecular biology</th>
<th>Metabolism</th>
<th>Signal transduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concurrent computational processes</td>
<td>molecules</td>
<td>enzymes and metabolites</td>
<td>interacting proteins</td>
</tr>
<tr>
<td>Synchronous communication</td>
<td>molecular interaction</td>
<td>binding and catalysis</td>
<td>binding and catalysis</td>
</tr>
</tbody>
</table>

- molecules as processes or species as processes?
- stochastic model or deterministic model?
- aims of modelling: good enough and practical enough
Bio-PEPA syntax

- species: reactions, stoichiometry, locations

\[ S@L \overset{\text{def}}{=} (\alpha_1, \kappa_1) \text{op}_1 S@L + \ldots + (\alpha_n, \kappa_n) \text{op}_n S@L \]

where \( \text{op}_i \in \{\downarrow, \uparrow, \oplus, \ominus, \oslash\} \)
Bio-PEPA syntax

- species: reactions, stoichiometry, locations

\[ S \circ L \overset{def}{=} (\alpha_1, \kappa_1) \circ p_1 S \circ L + \ldots + (\alpha_n, \kappa_n) \circ p_n S \circ L \]

where \( \circ p_i \in \{ \downarrow, \uparrow, \oplus, \ominus, \odot \} \)
Bio-PEPA syntax

- species: reactions, stoichiometry, locations

\[ S@L \overset{\text{def}}{=} (\alpha_1, \kappa_1) \circ_{\text{op}_1} S@L + \ldots + (\alpha_n, \kappa_n) \circ_{\text{op}_n} S@L \]

where \( \text{op}_i \in \{ \downarrow, \uparrow, \oplus, \ominus, \odot \} \)
Bio-PEPA syntax

- species: reactions, stoichiometry, locations

\[ S@L \overset{\text{def}}{=} (\alpha_1, \kappa_1) \circ_1 S@L + \ldots + (\alpha_n, \kappa_n) \circ_n S@L \]

where \( \circ_i \in \{ \downarrow, \uparrow, \oplus, \ominus, \odot \} \)
Bio-PEPA syntax

- species: reactions, stoichiometry, locations

\[ S@L \overset{\text{def}}{=} (\alpha_1, \kappa_1) \circ_1 S@L + \ldots + (\alpha_n, \kappa_n) \circ_n S@L \]

where \( \circ_i \in \{ \downarrow, \uparrow, \oplus, \ominus, \odot \} \)

- model: quantities of species, interaction between species

\[ P \overset{\text{def}}{=} S_1@L_1(x_1) \otimes \ldots \otimes S_p@L_p(x_p) \]
Bio-PEPA syntax

- species: reactions, stoichiometry, locations
  \[ S@L \overset{\text{def}}{=} (\alpha_1, \kappa_1) \text{ op}_1 S@L + \ldots + (\alpha_n, \kappa_n) \text{ op}_n S@L \]
  where \( \text{ op}_i \in \{\downarrow, \uparrow, \oplus, \ominus, \odot\} \)

- model: quantities of species, interaction between species
  \[ P \overset{\text{def}}{=} S_1@L_1(x_1) \bowtie \ldots \bowtie S_p@L_p(x_p) \]
Bio-PEPA syntax

► species: reactions, stoichiometry, locations

\[ S@L \overset{\text{def}}{=} (\alpha_1, \kappa_1) \circ p_1 S@L + \ldots + (\alpha_n, \kappa_n) \circ p_n S@L \]

where \( \circ p_i \in \{ \downarrow, \uparrow, \oplus, \ominus, \odot \} \)

► model: quantities of species, interaction between species

\[ P \overset{\text{def}}{=} S_1@L_1(x_1) \boxdot \ldots \boxdot S_p@L_p(x_p) \]

► system: includes other information required for modelling

\[ \mathcal{L} \] compartments and locations, dimensionality, sizes

\[ \mathcal{N} \] species quantities, minimums, maximums, step size

\[ \mathcal{K} \] parameter definitions

\[ \mathcal{F} \] functional rates for reactions, definition of \( f_\alpha \)
Bio-PEPA syntax

- species: reactions, stoichiometry, locations
  \[ S@L \overset{\text{def}}{=} (\alpha_1, \kappa_1) \circ_1 S@L + \ldots + (\alpha_n, \kappa_n) \circ_n S@L \]
  where \( \circ_i \in \{\downarrow, \uparrow, \oplus, \ominus, \otimes\} \)

- model: quantities of species, interaction between species
  \[ P \overset{\text{def}}{=} S_1@L_1(x_1) \bigotimes \ldots \bigotimes S_p@L_p(x_p) \]

- system: includes other information required for modelling
  - \( L \): compartments and locations, dimensionality, sizes
  - \( N \): species quantities, minimums, maximums, step size
  - \( K \): parameter definitions
  - \( F \): functional rates for reactions, definition of \( f_\alpha \)

- process-as-species rather than process-as-molecules
Example: reaction with enzyme

\[ S + E \xleftrightarrow{} C \rightarrow P + E \]
Example: reaction with enzyme

\[ S + E \rightleftharpoons C \rightarrow P + E \]

\[ S(\ell_S) \bowtie E(\ell_E) \bowtie C(\ell_C) \bowtie P(\ell_P) \quad \text{where} \]

\[ S \overset{\text{def}}{=} (\alpha, 1) \downarrow S + (\beta, 1) \uparrow S \]
\[ E \overset{\text{def}}{=} (\alpha, 1) \downarrow E + (\beta, 1) \uparrow E + (\gamma, 1) \uparrow E \]
\[ C \overset{\text{def}}{=} (\alpha, 1) \uparrow C + (\beta, 1) \downarrow C + (\gamma, 1) \downarrow C \]
\[ P \overset{\text{def}}{=} (\gamma, 1) \uparrow P \]
Example: reaction with enzyme

\( S + E \xleftrightarrow{\text{reaction}} C \xrightarrow{} P + E \)

\( \text{where} \quad S \overset{\text{def}}{=} (\alpha, 1) \downarrow S + (\beta, 1) \uparrow S \\
E \overset{\text{def}}{=} (\alpha, 1) \downarrow E + (\beta, 1) \uparrow E + (\gamma, 1) \uparrow E \\
C \overset{\text{def}}{=} (\alpha, 1) \uparrow C + (\beta, 1) \downarrow C + (\gamma, 1) \downarrow C \\
P \overset{\text{def}}{=} (\gamma, 1) \uparrow P \)
Example: reaction with enzyme

\[ S + E \xrightarrow{\alpha} C \xrightarrow{\beta} P + E \]

\[ S(\ell_S) \boxtimes E(\ell_E) \boxtimes C(\ell_C) \boxtimes P(\ell_P) \text{ where} \]

\[ S \overset{\text{def}}{=} (\alpha, 1) \downarrow S + (\beta, 1) \uparrow S \]
\[ E \overset{\text{def}}{=} (\alpha, 1) \downarrow E + (\beta, 1) \uparrow E + (\gamma, 1) \uparrow E \]
\[ C \overset{\text{def}}{=} (\alpha, 1) \uparrow C + (\beta, 1) \downarrow C + (\gamma, 1) \downarrow C \]
\[ P \overset{\text{def}}{=} (\gamma, 1) \uparrow P \]
Example: reaction with enzyme

\[ S + E \xleftrightarrow{\gamma} C \xrightarrow{\alpha} P + E \]

\[ S(\ell_S) \otimes E(\ell_E) \otimes C(\ell_C) \otimes P(\ell_P) \]

where

\[ S \overset{\text{def}}{=} (\alpha, 1) \downarrow S + (\beta, 1) \uparrow S \]
\[ E \overset{\text{def}}{=} (\alpha, 1) \downarrow E + (\beta, 1) \uparrow E + (\gamma, 1) \uparrow E \]
\[ C \overset{\text{def}}{=} (\alpha, 1) \uparrow C + (\beta, 1) \downarrow C + (\gamma, 1) \downarrow C \]
\[ P \overset{\text{def}}{=} (\gamma, 1) \uparrow P \]
Example: reaction with enzyme

\[ S + E \rightleftharpoons C \rightarrow P + E \]

\[ S(\ell_S) \otimes E(\ell_E) \otimes C(\ell_C) \otimes P(\ell_P) \]

where

\[ S \overset{\text{def}}{=} (\alpha, 1) \downarrow S + (\beta, 1) \uparrow S \]

\[ E \overset{\text{def}}{=} (\alpha, 1) \downarrow E + (\beta, 1) \uparrow E + (\gamma, 1) \uparrow E \]

\[ C \overset{\text{def}}{=} (\alpha, 1) \uparrow C + (\beta, 1) \downarrow C + (\gamma, 1) \downarrow C \]

\[ P \overset{\text{def}}{=} (\gamma, 1) \uparrow P \]

\[ S \xrightarrow{E} P \]
Example: reaction with enzyme

\[ S + E \xleftrightarrow{\rightarrow} C \rightarrow P + E \]

\[ S(\ell_S) \bowtie \ * \ E(\ell_E) \bowtie \ * \ C(\ell_C) \bowtie \ * \ P(\ell_P) \text{ where} \]

\[ S \overset{\text{def}}{=} (\alpha, 1) \downarrow S + (\beta, 1) \uparrow S \]

\[ E \overset{\text{def}}{=} (\alpha, 1) \downarrow E + (\beta, 1) \uparrow E + (\gamma, 1) \uparrow E \]

\[ C \overset{\text{def}}{=} (\alpha, 1) \uparrow C + (\beta, 1) \downarrow C + (\gamma, 1) \downarrow C \]

\[ P \overset{\text{def}}{=} (\gamma, 1) \uparrow P \]

\[ S \xrightarrow{E} P \]

\[ S'(\ell_{S'}) \bowtie \ * \ E'(\ell_{E'}) \bowtie \ * \ P'(\ell_{P'}) \text{ where} \]
Example: reaction with enzyme

\[ S + E \xrightarrow{\leftrightarrow} C \rightarrow P + E \]

\[ S(\ell_S) \otimes E(\ell_E) \otimes C(\ell_C) \otimes P(\ell_P) \text{ where} \]

\[ S \overset{\text{def}}{=} (\alpha, 1) \downarrow S + (\beta, 1) \uparrow S \]
\[ E \overset{\text{def}}{=} (\alpha, 1) \downarrow E + (\beta, 1) \uparrow E + (\gamma, 1) \uparrow E \]
\[ C \overset{\text{def}}{=} (\alpha, 1) \uparrow C + (\beta, 1) \downarrow C + (\gamma, 1) \downarrow C \]
\[ P \overset{\text{def}}{=} (\gamma, 1) \uparrow P \]

\[ S \xrightarrow{E} P \]

\[ S'(\ell_{S'}) \otimes E'(\ell_{E'}) \otimes P'(\ell_{P'}) \text{ where} \]

\[ S' \overset{\text{def}}{=} (\gamma, 1) \downarrow S' \quad E' \overset{\text{def}}{=} (\gamma, 1) \oplus E' \quad P' \overset{\text{def}}{=} (\gamma, 1) \uparrow P' \]
Example: reaction with enzyme

\[ S + E \leftrightarrow C \rightarrow P + E \]

\[ S(\ell_S) \otimes E(\ell_E) \otimes C(\ell_C) \otimes P(\ell_P) \text{ where} \]

\[ S \overset{\text{def}}{=} (\alpha, 1) \downarrow S + (\beta, 1) \uparrow S \]
\[ E \overset{\text{def}}{=} (\alpha, 1) \downarrow E + (\beta, 1) \uparrow E + (\gamma, 1) \uparrow E \]
\[ C \overset{\text{def}}{=} (\alpha, 1) \uparrow C + (\beta, 1) \downarrow C + (\gamma, 1) \downarrow C \]
\[ P \overset{\text{def}}{=} (\gamma, 1) \uparrow P \]

\[ S \overset{E}{\rightarrow} P \]

\[ S'(\ell_{S'}) \otimes E'(\ell_{E'}) \otimes P'(\ell_{P'}) \text{ where} \]

\[ S' \overset{\text{def}}{=} (\gamma, 1) \downarrow S' \quad E' \overset{\text{def}}{=} (\gamma, 1) \oplus E' \quad P' \overset{\text{def}}{=} (\gamma, 1) \uparrow P' \]
Example: reaction with enzyme

\[ S + E \xleftrightarrow{} C \rightarrow P + E \]

\[ S(\ell_S) \boxtimes E(\ell_E) \boxtimes C(\ell_C) \boxtimes P(\ell_P) \text{ where} \]

\[ S \overset{\text{def}}{=} (\alpha,1) \downarrow S + (\beta,1) \uparrow S \]
\[ E \overset{\text{def}}{=} (\alpha,1) \downarrow E + (\beta,1) \uparrow E + (\gamma,1) \uparrow E \]
\[ C \overset{\text{def}}{=} (\alpha,1) \uparrow C + (\beta,1) \downarrow C + (\gamma,1) \downarrow C \]
\[ P \overset{\text{def}}{=} (\gamma,1) \uparrow P \]

\[ S \xrightarrow{E} P \]

\[ S'(\ell_{S'}) \boxtimes E'(\ell_{E'}) \boxtimes P'(\ell_{P'}) \text{ where} \]

\[ S' \overset{\text{def}}{=} (\gamma,1) \downarrow S' \]
\[ E' \overset{\text{def}}{=} (\gamma,1) \oplus E' \]
\[ P' \overset{\text{def}}{=} (\gamma,1) \uparrow P' \]
Bio-PEPA semantics

- operational semantics for capability relation $\rightarrow_c$
Bio-PEPA semantics

- operational semantics for capability relation $\rightarrow_c$

- Prefix rules

$((\alpha, \kappa) \downarrow S@L)(\ell) \xrightarrow{(\alpha,[S@L:\downarrow(\ell,\kappa)])} S@L(\ell - \kappa) \quad \kappa \leq \ell \leq N_{S@L}$

$((\alpha, \kappa) \uparrow S@L)(\ell) \xrightarrow{(\alpha,[S@L:\uparrow(\ell,\kappa)])} S@L(\ell + \kappa) \quad 0 \leq \ell \leq N_{S@L} - \kappa$

$((\alpha, \kappa) \oplus S@L)(\ell) \xrightarrow{(\alpha,[S@L:\oplus(\ell,\kappa)])} S@L(\ell) \quad \kappa \leq \ell \leq N_{S@L}$

$((\alpha, \kappa) \ominus S@L)(\ell) \xrightarrow{(\alpha,[S@L:\ominus(\ell,\kappa)])} S@L(\ell) \quad 0 \leq \ell \leq N_{S@L}$

$((\alpha, \kappa) \otimes S@L)(\ell) \xrightarrow{(\alpha,[S@L:\otimes(\ell,\kappa)])} S@L(\ell) \quad 0 \leq \ell \leq N_{S@L}$
Bio-PEPA semantics

- operational semantics for capability relation $\rightarrow_c$
- Prefix rules

$\left((\alpha, \kappa) \downarrow S@L\right)(\ell) \xrightarrow{(\alpha,[S@L:\downarrow(\ell,\kappa)])} c\ S@L(\ell - \kappa) \quad \kappa \leq \ell \leq N_{S@L}$

$\left((\alpha, \kappa) \uparrow S@L\right)(\ell) \xrightarrow{(\alpha,[S@L:\uparrow(\ell,\kappa)])} c\ S@L(\ell + \kappa) \quad 0 \leq \ell \leq N_{S@L} - \kappa$

$\left((\alpha, \kappa) \oplus S@L\right)(\ell) \xrightarrow{(\alpha,[S@L:\oplus(\ell,\kappa)])} c\ S@L(\ell) \quad \kappa \leq \ell \leq N_{S@L}$

$\left((\alpha, \kappa) \ominus S@L\right)(\ell) \xrightarrow{(\alpha,[S@L:\ominus(\ell,\kappa)])} c\ S@L(\ell) \quad 0 \leq \ell \leq N_{S@L}$

$\left((\alpha, \kappa) \odot S@L\right)(\ell) \xrightarrow{(\alpha,[S@L:\odot(\ell,\kappa)])} c\ S@L(\ell) \quad 0 \leq \ell \leq N_{S@L}$
Bio-PEPA semantics

- operational semantics for capability relation $\rightarrow_c$
- Prefix rules

\[
((\alpha, \kappa) \downarrow S@L)(\ell) \xrightarrow{(\alpha, [S@L: \downarrow(\ell, \kappa)])} c S@L(\ell - \kappa) \quad \kappa \leq \ell \leq N_{S@L}
\]

\[
((\alpha, \kappa) \uparrow S@L)(\ell) \xrightarrow{(\alpha, [S@L: \uparrow(\ell, \kappa)])} c S@L(\ell + \kappa) \quad 0 \leq \ell \leq N_{S@L} - \kappa
\]

\[
((\alpha, \kappa) \oplus S@L)(\ell) \xrightarrow{(\alpha, [S@L: \oplus(\ell, \kappa)])} c S@L(\ell) \quad \kappa \leq \ell \leq N_{S@L}
\]

\[
((\alpha, \kappa) \ominus S@L)(\ell) \xrightarrow{(\alpha, [S@L: \ominus(\ell, \kappa)])} c S@L(\ell) \quad 0 \leq \ell \leq N_{S@L}
\]

\[
((\alpha, \kappa) \odot S@L)(\ell) \xrightarrow{(\alpha, [S@L: \odot(\ell, \kappa)])} c S@L(\ell) \quad 0 \leq \ell \leq N_{S@L}
\]
**Bio-PEPA semantics (continued)**

- Cooperation for $\alpha \in M$

\[
\begin{align*}
\frac{P \xrightarrow{(\alpha,v)} c P'}{P \Join M Q \xrightarrow{(\alpha,v::u)} c P' \Join M Q'} \quad \frac{Q \xrightarrow{(\alpha,u)} c Q'}{\alpha \in M}
\end{align*}
\]
Bio-PEPA semantics (continued)

- Cooperation for $\alpha \in M$

$$ P \xrightarrow{(\alpha, v)}_c P' \quad Q \xrightarrow{(\alpha, u)}_c Q' \quad \alpha \in M $$

$$ P \boxdot M Q \xrightarrow{(\alpha, v::u)}_c P' \boxdot M Q' $$

- operational semantics for stochastic relation $\rightarrow_s$

$$ P \xrightarrow{(\alpha, v)}_c P' $$

$$ \langle V, N, K, F, Comp, P \rangle \xrightarrow{(\alpha, f_\alpha(v, V, N, K)/h)}_s \langle V, N, K, F, Comp, P' \rangle $$
Bio-PEPA semantics (continued)

- Cooperation for $\alpha \in M$

$$
P \xrightarrow{(\alpha,v)}_c P' \quad Q \xrightarrow{(\alpha,u)}_c Q'
$$

$$
P \Join^M Q \xrightarrow{(\alpha,v::u)}_c P' \Join^M Q'
$$

- operational semantics for stochastic relation $\rightarrow_s$

$$
P \xrightarrow{(\alpha,v)}_c P'
$$

$$
\langle V, N, K, F, Comp, P \rangle \xrightarrow{(\alpha,f_\alpha(v,V,N,K)/h)}_s \langle V, N, K, F, Comp, P' \rangle
$$

- rate function $f_\alpha$ uses information about the species and locations in the string $v$, together with the species and location information and rate parameters in calculating the actual rate of the reaction
Example: reaction with enzyme, max level 3

- state vector \((S, E, C, P)\) and \(N_S = N_E = N_C = N_P = 3\)
Example: reaction with enzyme, max level 3

- state vector \((S, E, C, P)\) and \(N_S = N_E = N_C = N_P = 3\)

\[
\begin{align*}
(3, 3, 0, 0) & \xleftrightarrow{\alpha} (2, 2, 1, 0) \xleftrightarrow{\alpha} (1, 1, 2, 0) \xleftrightarrow{\alpha} (0, 0, 3, 0) \\
& \quad \quad \quad \downarrow \gamma \quad \quad \quad \downarrow \gamma \quad \quad \quad \downarrow \gamma \\
(2, 3, 0, 1) & \xleftrightarrow{\alpha} (1, 2, 1, 1) \xleftrightarrow{\alpha} (0, 1, 2, 1) \\
& \quad \quad \quad \downarrow \gamma \quad \quad \quad \downarrow \gamma \quad \quad \quad \downarrow \gamma \\
(1, 3, 0, 2) & \xleftrightarrow{\alpha} (0, 2, 1, 2) \\
& \quad \quad \quad \downarrow \gamma \\
(0, 3, 0, 3)
\end{align*}
\]
Example: reaction with enzyme, max level 7

- state vector $S \ E \ C \ P$ and $N_S = N_E = N_C = N_P = 7$
Different types of analysis

Bio-PEPA model

CTMC with levels, semantic equivalences

ordinary differential equations (ODEs)

stochastic simulation (Gillespie’s algorithm)

PRISM (model checking)
Deterministic versus stochastic simulation

```
S(30) ⊙ E(10) ⊙ C(0) ⊙ P(0)
```

\[
k_\alpha = 10 \quad k_\beta = 100 \quad k_\gamma = 0.1
\]
Deterministic versus stochastic simulation

\[ S(30) \bowtie E(10) \bowtie C(0) \bowtie P(0) \]
\[ k_\alpha = 10 \quad k_\beta = 100 \quad k_\gamma = 0.1 \]
Deterministic versus stochastic simulation

\[ S(30) \bowtie E(10) \bowtie C(0) \bowtie P(0) \]

\[ k_\alpha = 10 \quad k_\beta = 100 \quad k_\gamma = 0.1 \]
Deterministic versus stochastic simulation

**Deterministic trace**

**Average of 1000 stochastic traces**

\[
S(30) \bowtie E(10) \bowtie C(0) \bowtie P(0)
\]

\[
k_\alpha = 10 \quad k_\beta = 100 \quad k_\gamma = 0.1
\]
Bio-PEPA Eclipse Plug-in

- software tool for Bio-PEPA modelling
Bio-PEPA Eclipse Plug-in

- software tool for Bio-PEPA modelling
- Eclipse front-end and separate back-end library

<table>
<thead>
<tr>
<th>User Interface</th>
<th>Core</th>
</tr>
</thead>
<tbody>
<tr>
<td>editor for the Bio-PEPA language</td>
<td>parser for the Bio-PEPA language</td>
</tr>
<tr>
<td>problems view</td>
<td>static analysis</td>
</tr>
<tr>
<td>outline view for the reaction-centric view</td>
<td>ISBJava time series analysis (ODE, SSA)</td>
</tr>
<tr>
<td>graphing support via common plugin</td>
<td>export facility (SBML; PRISM)</td>
</tr>
</tbody>
</table>
Bio-PEPA Eclipse Plug-in

- software tool for Bio-PEPA modelling
- Eclipse front-end and separate back-end library

User Interface:
- editor for the Bio-PEPA language
- problems view
- outline view for the reaction-centric view
- graphing support via common plugin

Core:
- parser for the Bio-PEPA language
- static analysis
- ISBJava time series analysis (ODE, SSA)
- export facility (SBML; PRISM)

- available for download at www.biopepa.org
- case studies, publications, manuals
Bio-PEPA Eclipse Plug-in (continued)
Recent research: hybrid modelling

- stochastic HYPE: stochastic hybrid process algebra
Recent research: hybrid modelling

- stochastic HYPE: stochastic hybrid process algebra
- map Bio-PEPA model to stochastic HYPE model
  - combine stochastic and deterministic elements
  - dynamic modelling of these elements
- extend to Bio-PEPA with events
- provide well-structured process algebra model
- provide framework for modelling in stochastic HYPE
Recent research: hybrid modelling

- stochastic HYPE: stochastic hybrid process algebra
- map Bio-PEPA model to stochastic HYPE model
  - combine stochastic and deterministic elements
  - dynamic modelling of these elements
- extend to Bio-PEPA with events
  - provide well-structured process algebra model
  - provide framework for modelling in stochastic HYPE
- provides benefits of both stochastic simulation and deterministic simulation
Recent research: hybrid modelling

- stochastic HYPE: stochastic hybrid process algebra
- map Bio-PEPA model to stochastic HYPE model
  - combine stochastic and deterministic elements
  - dynamic modelling of these elements
- extend to Bio-PEPA with events
  - provide well-structured process algebra model
  - provide framework for modelling in stochastic HYPE
- provides benefits of both stochastic simulation and deterministic simulation
- requires specification of thresholds to determine switching of reaction simulation
Deterministic and stochastic simulation

trace: \( \gamma \) slow

\[
S(30) 
\begin{array}{cccc}
\ast & E(10) & \ast & C(0) \\
\end{array} 
\ast P(0)
\]

\[
k_\alpha = 10 \quad k_\beta = 100 \quad k_\gamma = 0.1
\]
Src trafficking

- non-receptor protein tyrosine kinase, member of Src family
Src trafficking

- non-receptor protein tyrosine kinase, member of Src family
- in either inactive or active configuration
Src protein: inactive and active

Src trafficking

- non-receptor protein tyrosine kinase, member of Src family
- in either inactive or active configuration
Src trafficking

- non-receptor protein tyrosine kinase, member of Src family
- in either inactive or active configuration
- active Src at membrane implicated in cancer
Src trafficking

- non-receptor protein tyrosine kinase, member of Src family
- in either inactive or active configuration
- active Src at membrane implicated in cancer
- location in normal cell without growth factor (FGF) addition
  - inactive pool near nucleus, little active on membrane
Src trafficking

- non-receptor protein tyrosine kinase, member of Src family
- in either inactive or active configuration
- active Src at membrane implicated in cancer
- location in normal cell without growth factor (FGF) addition
  - inactive pool near nucleus, little active on membrane
- location in normal cell after FGF addition
  - inactive pool near nucleus, increase in active on membrane
Src trafficking

- non-receptor protein tyrosine kinase, member of Src family
- in either inactive or active configuration
- active Src at membrane implicated in cancer
- location in normal cell without growth factor (FGF) addition
  - inactive pool near nucleus, little active on membrane
- location in normal cell after FGF addition
  - inactive pool near nucleus, increase in active on membrane
- endosomes: membrane-bound compartments within cells
- move along microfilaments or microtubules in one direction
Mechanisms

- experimental research from the Frame laboratory has shown

After stimulation with FGF, Src is found in endosomes throughout the cytoplasm. There is a gradient of inactive Src to active Src from perinuclear region to membrane. Src activation takes place in endosomes. (Sandilands et al, 2004)
Mechanisms

- experimental research from the Frame laboratory has shown

After stimulation with FGF, Src is found in endosomes throughout the cytoplasm. There is a gradient of inactive Src to active Src from perinuclear region to membrane. Src activation takes place in endosomes. (Sandilands et al, 2004)

The persistence of active Src at the membrane is inversely related to the quantity of FGF added. (Sandilands et al, 2007)
Mechanisms: persistence of response to FGF

(Sandilands et al, EMBO Reports 8, 2007)
Modelling protein trafficking

- modelling aspects
  - **dynamic**: behaviour over time, addition of FGF
  - **spatial**: movement of molecules, endosomes
Modelling protein trafficking

- modelling aspects
  - **dynamic**: behaviour over time, addition of FGF
  - **spatial**: movement of molecules, endosomes

- modelling challenges
  - **concrete** enough and **abstract** enough
  - **data**: very limited
Modelling protein trafficking

- modelling aspects
  - **dynamic**: behaviour over time, addition of FGF
  - **spatial**: movement of molecules, endosomes

- modelling challenges
  - **concrete** enough and **abstract** enough
  - **data**: very limited

- data
  - **qualitative**: gradient, recycling loops
  - **quantitative**: response to addition, endosome data
membrane

**active Src**

---

perinuclear region

**inactive Src**
Vashti Galpin
Formal modelling of biological systems
Informatics/Systems Biology Colloquium, Masaryck University

**Src trafficking**
Formal modelling of biological systems

Vashti Galpin

Informatics/Systems Biology Colloquium, Masaryk University
Formal modelling of biological systems

Vashti Galpin

Informatics/Systems Biology Colloquium, Masaryck University
Vashti Galpin

Formal modelling of biological systems

Informatics/Systems Biology Colloquium, Masaryk University
membrane

**active Src**

perinuclear region

**inactive Src**
membrane

active Src

membrane

perinuclear region

inactive Src
Formal modelling of biological systems

Vashti Galpin

Informatics/Systems Biology Colloquium, Masaryck University
Vashti Galpin

Formal modelling of biological systems

Informatics/Systems Biology Colloquium, Masaryck University
Formal modelling of biological systems

Vashti Galpin

Informatics/Systems Biology Colloquium, Masaryck University
Process algebras
Bio-PEPA
Semantics
Enzyme example
Hybrid approach
Src trafficking
Conclusions

membrane

active Src

inactive Src

perinuclear region

inactive Src
Vashti Galpin

Formal modelling of biological systems

Informatics/Systems Biology Colloquium, Masaryck University
Formal modelling of biological systems

Vashti Galpin

Informatics/Systems Biology Colloquium, Masaryck University
Generic recycling loop

- modelling of endosome trafficking

\[ \begin{align*}
  xA & \xrightarrow{\text{form}} \text{Endo}_\text{in} & \text{in} & \text{Endo}_\text{dec} & \xrightarrow{\text{deg}} & \emptyset \\
  yB & \xleftarrow{\text{rel}} \text{Endo}_\text{rel} & \xleftarrow{\text{out}} \text{Endo}_\text{rec} & \xrightarrow{\text{rec}} & \\
\end{align*} \]
Two loop trafficking model – results

The diagram illustrates the effects of different concentrations of FGF addition on the quantities of active Src and FGFR. The graph shows the changes over time (in minutes) for different concentration levels:

- **Active Src (high concentration)**
- **FGFR (high concentration)**
- **Active Src (low concentration)**
- **FGFR (low concentration)**

The quantities are measured on the y-axis, ranging from 0 to 300,000, while the time in minutes is indicated on the x-axis, from 0 to 180 minutes.
Equivalences

- when do two Bio-PEPA models have the same behaviour?
- when do congruence results apply?
Equivalences

- when do two Bio-PEPA models have the same behaviour?
- when do congruence results apply?
- compression bisimulation, qualitative
  - compares behaviour of a system with different levels of discretisation
  - results and hypothesis about congruence
Equivalences

► when do two Bio-PEPA models have the same behaviour?
► when do congruence results apply?
► compression bisimulation, qualitative
  ► compares behaviour of a system with different levels of discretisation
  ► results and hypothesis about congruence
► $g$-bisimulation, can be quantitative
  ► $g$ is a function over labels of capability relation
Equivalences

- when do two Bio-PEPA models have the same behaviour?
- when do congruence results apply?
- compression bisimulation, qualitative
  - compares behaviour of a system with different levels of discretisation
  - results and hypothesis about congruence
- $g$-bisimulation, can be quantitative
  - $g$ is a function over labels of capability relation
- fast-slow bisimulation, qualitative
  - based on quasi-steady-state assumption (QSSA), identifies species to be abstracted
Conclusions

- process algebras are compact mathematical languages to describe concurrent behaviour
Conclusions

- process algebras are compact mathematical languages to describe concurrent behaviour
- Bio-PEPA: process algebra for modelling biological systems
Conclusions

- process algebras are compact mathematical languages to describe concurrent behaviour

- Bio-PEPA: process algebra for modelling biological systems

- Bio-PEPA permists multiple analysis techniques
Conclusions

- process algebras are compact mathematical languages to describe concurrent behaviour
- Bio-PEPA: process algebra for modelling biological systems
- Bio-PEPA permits multiple analysis techniques
- new mapping to stochastic HYPE adds new technique
Conclusions

- process algebras are compact mathematical languages to describe concurrent behaviour
- Bio-PEPA: process algebra for modelling biological systems
- Bio-PEPA permits multiple analysis techniques
- new mapping to stochastic HYPE adds new technique
- Bio-PEPA can be applied to complex examples with limited data
Acknowledgements

**PEPA Group**
University of Edinburgh
Jane Hillston
Stephen Gilmore
Allan Clark
Maria Luisa Guerriero
Federica Ciocchetta
Adam Duguid

**DMG**
University of Trieste
Luca Bortolussi

**Cancer Research UK**
Edinburgh
Margaret Frame
Emma Sandilands
Thank you