

a basic calculus for molecular biology

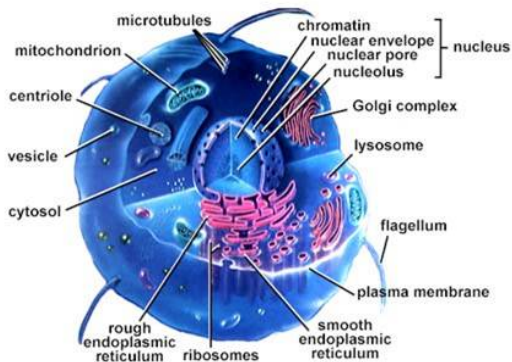
Cosimo Laneve

Department of Computer Science
University of Bologna

joint work with Fabien Tarissan (Paris VII)

motivations

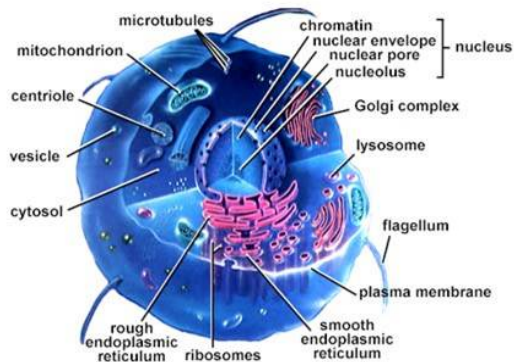
molecular biology is complex – example: the organization of an eucaryotic cell



. . . a lot of different agents that cooperate for the life and a lot of different interaction mechanisms

motivations

molecular biology is complex – example: the organization of an eucaryotic cell



. . . a lot of different agents that cooperate for the life and a lot of different interaction mechanisms

biological cooperation is complex as well

- ▶ several agents interact at the same time by means of several domains (sites)
 - parallelism
 - competition
 - nondeterminism
- ▶ interactions may involve simple agents – **proteins** – or complex ones – **cells** – and may cause small local changes (phosphorylations) or huge structural changes (fusions, translocations, etc.)
- ▶ the interaction is **stochastic**, cf. Gillespies semantics
- ▶ the overall behaviour is **deterministic**: processes are highly regulated by means of controls, feedbacks, etc.

biological cooperation is complex as well

- ▶ several agents interact at the same time by means of several domains (sites)
 - parallelism
 - competition
 - nondeterminism
- ▶ interactions may involve simple agents – **proteins** – or complex ones – **cells** – and may cause small local changes (phosphorylations) or huge structural changes (fusions, translocations, etc.)
- ▶ the interaction is **stochastic**, cf. Gillespies semantics
- ▶ the overall behaviour is **deterministic**: processes are highly regulated by means of controls, feedbacks, etc.

biological cooperation is complex as well

- ▶ several agents interact at the same time by means of several domains (sites)
 - parallelism
 - competition
 - nondeterminism
- ▶ interactions may involve simple agents – **proteins** – or complex ones – **cells** – and may cause small local changes (phosphorylations) or huge structural changes (fusions, translocations, etc.)
- ▶ the interaction is **stochastic**, cf. Gillespies semantics
- ▶ the overall behaviour is **deterministic**: processes are highly regulated by means of controls, feedbacks, etc.

biological cooperation is complex as well

- ▶ several agents interact at the same time by means of several domains (sites)
 - parallelism
 - competition
 - nondeterminism
- ▶ interactions may involve simple agents – **proteins** – or complex ones – **cells** – and may cause small local changes (phosphorylations) or huge structural changes (fusions, translocations, etc.)
- ▶ the interaction is **stochastic**, cf. Gillespies semantics
- ▶ the overall behaviour is **deterministic**: processes are highly regulated by means of controls, feedbacks, etc.

biological cooperation is complex as well

- ▶ several agents interact at the same time by means of several domains (sites)
 - parallelism
 - competition
 - nondeterminism
- ▶ interactions may involve simple agents – **proteins** – or complex ones – **cells** – and may cause small local changes (phosphorylations) or huge structural changes (fusions, translocations, etc.)
- ▶ the interaction is **stochastic**, cf. Gillespies semantics
- ▶ the overall behaviour is **deterministic**: processes are highly regulated by means of controls, feedbacks, etc.

biological cooperation is ... too complex

we intend to pursue an **algebraic approach** à la pi-calculus

- ▶ few (biological) constructs
- ▶ a “faithful” rendering of biological interactions – not *via* an encoding
- ▶ a **compositional** semantics based on the notion of **interaction**

but simplicity has a cost! we are loosing:

- expressiveness
- stochastic behaviours
- transactional mechanisms

biological cooperation is ... too complex

we intend to pursue an **algebraic approach** à la pi-calculus

- ▶ few (biological) constructs
- ▶ a “faithful” rendering of biological interactions – not *via* an encoding
- ▶ a **compositional** semantics based on the notion of **interaction**

but simplicity has a cost! we are loosing:

- expressiveness
- stochastic behaviours
- transactional mechanisms

biological cooperation is ... too complex

we intend to pursue an **algebraic approach** à la pi-calculus

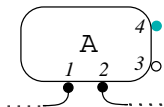
- ▶ few (biological) constructs
- ▶ a “faithful” rendering of biological interactions – not *via* an encoding
- ▶ a **compositional** semantics based on the notion of **interaction**

but simplicity has a cost! we are loosing:

- expressiveness
- stochastic behaviours
- transactional mechanisms

the bioK -calculus – a calculus of proteins and cells

proteins



- *visible site*
- *hidden site*
- *bound site*

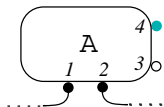
is described in bioK by $A(1^x + 2^z + 3^v + 4^h)$ – actually we write $A(1^x + 2^z + 3 + \overline{4})$

syntactically, a protein is $A(\sigma)$

- A belongs to a countable set of *protein names*
- for every A , $s(a)$ gives an integer – the number of *sites*
- there is a set \mathcal{E} of *edge names* that are ranged over by x, y, z , etc. –
 $v, h \notin \mathcal{E}$
- σ is a total function from $1..s(a)$ to $\{v, h\} \cup \mathcal{E}$ such that σ is injective on \mathcal{E}

the bioK-calculus – a calculus of proteins and cells

proteins



- *visible site*
- *hidden site*
- *bound site*

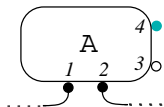
is described in bioK by $A(1^x + 2^z + 3^v + 4^h)$ – actually we write $A(1^x + 2^z + 3 + \overline{4})$

syntactically, a protein is $A(\sigma)$

- A belongs to a countable set of *protein names*
- for every A, $s(a)$ gives an integer – the number of *sites*
- there is a set \mathcal{E} of *edge names* that are ranged over by x, y, z , etc. –
 $v, h \notin \mathcal{E}$
- σ is a total function from $1..s(a)$ to $\{v, h\} \cup \mathcal{E}$ such that σ is injective on \mathcal{E}

the bioK-calculus – a calculus of proteins and cells

proteins



- *visible site*
- *hidden site*
- *bound site*

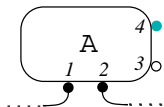
is described in bioK by $A(1^x + 2^z + 3^v + 4^h)$ – actually we write $A(1^x + 2^z + 3 + \overline{4})$

syntactically, a protein is $A(\sigma)$

- A belongs to a countable set of *protein names*
- for every A , $s(a)$ gives an integer – the number of *sites*
- there is a set \mathcal{E} of *edge names* that are ranged over by x, y, z , etc. –
 $v, h \notin \mathcal{E}$
- σ is a total function from $1..s(a)$ to $\{v, h\} \cup \mathcal{E}$ such that σ is injective on \mathcal{E}

the bioK-calculus – a calculus of proteins and cells

proteins



- *visible site*
- *hidden site*
- *bound site*

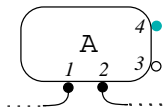
is described in bioK by $A(1^x + 2^z + 3^v + 4^h)$ – actually we write $A(1^x + 2^z + 3 + \overline{4})$

syntactically, a protein is $A(\sigma)$

- A belongs to a countable set of *protein names*
- for every A, $s(a)$ gives an integer – the number of *sites*
- there is a set \mathcal{E} of *edge names* that are ranged over by x, y, z , etc. –
 $v, h \notin \mathcal{E}$
- σ is a total function from $1..s(a)$ to $\{v, h\} \cup \mathcal{E}$ such that σ is injective on \mathcal{E}

the `bioK`-calculus – a calculus of proteins and cells

proteins



- *visible site*
- *hidden site*
- *bound site*

is described in `bioK` by $A(1^x + 2^z + 3^v + 4^h)$ – actually we write $A(1^x + 2^z + 3 + \bar{4})$

syntactically, a protein is $A(\sigma)$

- A belongs to a countable set of *protein names*
- for every A , $s(a)$ gives an integer – the number of *sites*
- there is a set \mathcal{E} of *edge names* that are ranged over by x, y, z , etc. –
 $v, h \notin \mathcal{E}$
- σ is a total function from $1..s(a)$ to $\{v, h\} \cup \mathcal{E}$ such that σ is injective on \mathcal{E}

the bioK-calculus – a calculus of proteins and cells

cells $m(\langle M \rangle)[S]$

- m belongs to a countable set of *membrane types*
- M is the *membrane*
- S is a biological solution (that may contain cells)
- **well-formedness constraints:**
 - (edge-condition) every solution is such that edge names occur at most twice;
 - (membrane-condition) every membrane is a multiset of proteins, that is cells do not occur in membranes – *we are abstracting out the bilipidic layer*
 - (nucleus-condition) the dangling edges of nuclei of cells are connected to the corresponding membrane, that is, for every $m(\langle M \rangle)[S]$, $\text{de}(S) \subseteq \text{de}(M)$.

the bioK-calculus – a calculus of proteins and cells

cells $m(\langle M \rangle)[S]$

- m belongs to a countable set of *membrane types*
- M is the *membrane*
- S is a biological solution (that may contain cells)
- **well-formedness constraints:**
 - (edge-condition) every solution is such that edge names occur at most twice;
 - (membrane-condition) every membrane is a multiset of proteins, that is cells do not occur in membranes – *we are abstracting out the bilipidic layer*
 - (nucleus-condition) the dangling edges of nuclei of cells are connected to the corresponding membrane, that is, for every $m(\langle M \rangle)[S]$, $\text{de}(S) \subseteq \text{de}(M)$.

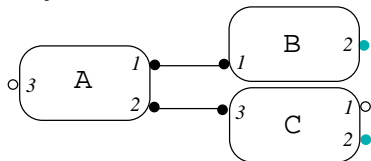
the bioK-calculus – a calculus of proteins and cells

cells $m(\langle M \rangle)[S]$

- m belongs to a countable set of *membrane types*
- M is the *membrane*
- S is a biological solution (that may contain cells)
- **well-formedness constraints:**
 - (edge-condition) every solution is such that edge names occur at most twice;
 - (membrane-condition) every membrane is a multiset of proteins, that is cells do not occur in membranes – *we are abstracting out the bilipidic layer*
 - (nucleus-condition) the dangling edges of nuclei of cells are connected to the corresponding membrane, that is, for every $m(\langle M \rangle)[S]$, $\text{de}(S) \subseteq \text{de}(M)$.

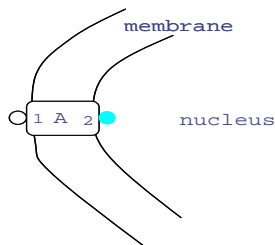
examples in bioK

complexes:



$$A(1^x + 2^y + 3), B(1^x + \bar{2}), C(1 + \bar{2} + 3^y)$$

activators:



$$m(A(1 + \bar{2})) [S]$$

remark: we do not specify whether a site of a protein is outside or inside a membrane

bioK-calculus: the syntax

biological solutions S:

S ::=	solution
0	(empty solution)
A(σ)	(protein)
m(S)[S]	(cell)
S, S	(group)

auxiliary functions:

- $en(\cdot)$ returns the set of *edge names*
- $de(\cdot)$ returns the set of *dangling edge names*

remark: with “cell” we address everything that is wrapped by a membrane – such as *nuclei, biological cells*

bioK-calculus: the syntax

biological solutions S:

S ::=	solution
$\mathbf{0}$	(empty solution)
$A(\sigma)$	(protein)
$m(S)[S]$	(cell)
S, S	(group)

auxiliary functions:

- $\text{en}(\cdot)$ returns the set of *edge names*
- $\text{de}(\cdot)$ returns the set of *dangling edge names*

remark: with “cell” we address everything that is wrapped by a membrane – such as *nuclei, biological cells*

bioK-calculus: the syntax

biological solutions S:

S ::=	solution
$\mathbf{0}$	(empty solution)
$A(\sigma)$	(protein)
$m(S)[S]$	(cell)
S, S	(group)

auxiliary functions:

- $\text{en}(\cdot)$ returns the set of *edge names*
- $\text{de}(\cdot)$ returns the set of *dangling edge names*

remark: with “cell” we address everything that is wrapped by a membrane – such as *nuclei, biological cells*

bioK-calculus: the transition relation – preliminaries

- v - h -maps, ranged over by ϕ, ψ, \dots , are partial maps from naturals to $\{v, h\}$

– $\bar{\phi}$ is the v - h -map:

$$\bar{\phi}(i) = \begin{cases} h & \text{if } \phi(i) = v \\ v & \text{if } \phi(i) = h \\ \text{undefined} & \text{otherwise} \end{cases}$$

- abuse of notation: σ, σ' range over partial maps from naturals to $\{v, h\} \cup \mathcal{E}$
- α, β , etc. range over (A, i, ϕ) , such that $\{i\} \uplus \text{dom}(\phi) \subseteq 1..s(A)$
- **complexations** $\overset{\bullet}{\mathcal{R}}$ and **decomplexations** $\overset{\circ}{\mathcal{R}}$ are symmetric relations (α, β)
- $\overset{\circ\circ}{x}$ range over $\overset{\circ}{x}$ or $\overset{\bullet}{x}$; $\overset{\circ\circ}{\mathcal{R}}$ ranges over $\overset{\circ}{\mathcal{R}}$ or $\overset{\bullet}{\mathcal{R}}$
- μ ranges over τ or $\overset{\circ}{\alpha}, \overset{\bullet}{x}$ or $\overset{\circ}{\alpha}, \overset{\bullet}{x}$

bioK-calculus: the transition relation – preliminaries

- v - h -maps, ranged over by ϕ, ψ, \dots , are partial maps from naturals to $\{v, h\}$

– $\bar{\phi}$ is the v - h -map:

$$\bar{\phi}(i) = \begin{cases} h & \text{if } \phi(i) = v \\ v & \text{if } \phi(i) = h \\ \text{undefined} & \text{otherwise} \end{cases}$$

- abuse of notation: σ, σ' range over partial maps from naturals to $\{v, h\} \cup \mathcal{E}$
- α, β , etc. range over (A, i, ϕ) , such that $\{i\} \uplus \text{dom}(\phi) \subseteq 1..s(A)$
- **complexations** $\overset{\bullet}{\mathcal{R}}$ and **decomplexations** $\overset{\circ}{\mathcal{R}}$ are symmetric relations (α, β)
- $\overset{\circ\circ}{x}$ range over $\overset{\circ}{x}$ or $\overset{\bullet}{x}$; $\overset{\circ\circ}{\mathcal{R}}$ ranges over $\overset{\circ}{\mathcal{R}}$ or $\overset{\bullet}{\mathcal{R}}$
- μ ranges over τ or $\overset{\circ}{\alpha}, \overset{\bullet}{x}$ or $\overset{\circ}{\alpha}, \overset{\bullet}{x}$

bioK-calculus: the transition relation – preliminaries

- v - h -maps, ranged over by ϕ, ψ, \dots , are partial maps from naturals to $\{v, h\}$

– $\bar{\phi}$ is the v - h -map:

$$\bar{\phi}(i) = \begin{cases} h & \text{if } \phi(i) = v \\ v & \text{if } \phi(i) = h \\ \text{undefined} & \text{otherwise} \end{cases}$$

- abuse of notation: σ, σ' range over partial maps from naturals to $\{v, h\} \cup \mathcal{E}$
- α, β , etc. range over (A, i, ϕ) , such that $\{i\} \uplus \text{dom}(\phi) \subseteq 1..s(A)$
- **complexations** $\overset{\bullet}{\mathcal{R}}$ and **decomplexations** $\overset{\circ}{\mathcal{R}}$ are symmetric relations (α, β)
- $\overset{\circ\circ}{x}$ range over $\overset{\circ}{x}$ or $\overset{\bullet}{x}$; $\overset{\circ}{\mathcal{R}}$ ranges over $\overset{\circ}{\mathcal{R}}$ or $\overset{\bullet}{\mathcal{R}}$
- μ ranges over τ or $\overset{\circ}{\alpha}, \overset{\bullet}{x}$ or $\overset{\circ}{\alpha}, \overset{\bullet}{x}$

bioK-calculus: the transition relation – preliminaries

- v - h -maps, ranged over by ϕ, ψ, \dots , are partial maps from naturals to $\{v, h\}$

– $\bar{\phi}$ is the v - h -map:

$$\bar{\phi}(i) = \begin{cases} h & \text{if } \phi(i) = v \\ v & \text{if } \phi(i) = h \\ \text{undefined} & \text{otherwise} \end{cases}$$

- abuse of notation: σ, σ' range over partial maps from naturals to $\{v, h\} \cup \mathcal{E}$
- α, β , etc. range over (A, i, ϕ) , such that $\{i\} \uplus \text{dom}(\phi) \subseteq 1..s(A)$

- **complexations** $\overset{\circ}{\mathcal{R}}$ and **decomplexations** $\overset{\circ}{\mathcal{R}}$ are symmetric relations (α, β)

- $\overset{\circ\circ}{x}$ range over $\overset{\circ}{x}$ or $\overset{\bullet}{x}$; $\overset{\circ\circ}{\mathcal{R}}$ ranges over $\overset{\circ}{\mathcal{R}}$ or $\overset{\bullet}{\mathcal{R}}$

- μ ranges over τ or $\overset{\circ}{\alpha}, \overset{\bullet}{x}$ or $\overset{\circ}{\alpha}, \overset{\bullet}{x}$

bioK-calculus: the transition relation – preliminaries

- v - h -maps, ranged over by ϕ, ψ, \dots , are partial maps from naturals to $\{v, h\}$

– $\bar{\phi}$ is the v - h -map:

$$\bar{\phi}(i) = \begin{cases} h & \text{if } \phi(i) = v \\ v & \text{if } \phi(i) = h \\ \text{undefined} & \text{otherwise} \end{cases}$$

- abuse of notation: σ, σ' range over partial maps from naturals to $\{v, h\} \cup \mathcal{E}$
- α, β , etc. range over (A, i, ϕ) , such that $\{i\} \uplus \text{dom}(\phi) \subseteq 1..s(A)$
- **complexations** $\overset{\bullet}{\mathcal{R}}$ and **decomplexations** $\overset{\circ}{\mathcal{R}}$ are symmetric relations (α, β)

- $\overset{\circ\circ}{x}$ range over $\overset{\circ}{x}$ or $\overset{\bullet}{x}$; $\overset{\circ\circ}{\mathcal{R}}$ ranges over $\overset{\circ}{\mathcal{R}}$ or $\overset{\bullet}{\mathcal{R}}$

- μ ranges over τ or $\overset{\circ}{\alpha}, \overset{\bullet}{x}$ or $\overset{\circ}{\alpha}, \overset{\bullet}{x}$

bioK-calculus: the transition relation – preliminaries

- v - h -maps, ranged over by ϕ, ψ, \dots , are partial maps from naturals to $\{v, h\}$

– $\bar{\phi}$ is the v - h -map:

$$\bar{\phi}(i) = \begin{cases} h & \text{if } \phi(i) = v \\ v & \text{if } \phi(i) = h \\ \text{undefined} & \text{otherwise} \end{cases}$$

- abuse of notation: σ, σ' range over partial maps from naturals to $\{v, h\} \cup \mathcal{E}$
- α, β , etc. range over (A, i, ϕ) , such that $\{i\} \uplus \text{dom}(\phi) \subseteq 1..s(A)$
- **complexations** $\overset{\bullet}{\mathcal{R}}$ and **decomplexations** $\overset{\circ}{\mathcal{R}}$ are symmetric relations (α, β)
- $\overset{\circ\circ}{x}$ range over $\overset{\circ}{x}$ or $\overset{\bullet}{x}$; $\overset{\circ\circ}{\mathcal{R}}$ ranges over $\overset{\circ}{\mathcal{R}}$ or $\overset{\bullet}{\mathcal{R}}$
- μ ranges over τ or $\overset{\circ}{\alpha}, \overset{\bullet}{x}$ or $\overset{\circ}{\alpha}, \overset{\bullet}{x}$

bioK-calculus: the transition relation – preliminaries

- v - h -maps, ranged over by ϕ, ψ, \dots , are partial maps from naturals to $\{v, h\}$

– $\bar{\phi}$ is the v - h -map:

$$\bar{\phi}(i) = \begin{cases} h & \text{if } \phi(i) = v \\ v & \text{if } \phi(i) = h \\ \text{undefined} & \text{otherwise} \end{cases}$$

- abuse of notation: σ, σ' range over partial maps from naturals to $\{v, h\} \cup \mathcal{E}$
- α, β , etc. range over (A, i, ϕ) , such that $\{i\} \uplus \text{dom}(\phi) \subseteq 1..s(A)$
- **complexations** $\overset{\bullet}{\mathcal{R}}$ and **decomplexations** $\overset{\circ}{\mathcal{R}}$ are symmetric relations (α, β)
- $\overset{\circ\circ}{x}$ range over $\overset{\circ}{x}$ or $\overset{\bullet}{x}$; $\overset{\circ\circ}{\mathcal{R}}$ ranges over $\overset{\circ}{\mathcal{R}}$ or $\overset{\bullet}{\mathcal{R}}$
- μ ranges over τ or $\alpha, \overset{\circ}{x}$ or $\alpha, \overset{\bullet}{x}$

bioK-calculus: the transition relation

notation: $\text{diff}(S, S') \stackrel{\text{def}}{=} (\text{en}(S') \setminus \text{en}(S)) \cup (\text{en}(S) \setminus \text{en}(S'))$

$\alpha \in \mathcal{R}$ if there is β such that $(\alpha, \beta) \in \mathcal{R}$ (similarly for \mathcal{R})

the transition relation $\xrightarrow{\mu}$ is the least one satisfying the reductions:

- protein-protein reductions

$$\frac{(A, i, \phi) \in \mathcal{R} \quad x \notin \text{en}(\sigma)}{A(i + \phi + \sigma) \xrightarrow{A, i, \phi, \dot{x}} A(i^x + \bar{\phi} + \sigma)} \quad \frac{(A, i, \phi) \in \mathcal{R}}{A(i^x + \phi + \sigma) \xrightarrow{A, i, \phi, \dot{x}} A(i + \bar{\phi} + \sigma)}$$

$$\frac{S \xrightarrow{\alpha, \ddot{x}} S' \quad T \xrightarrow{\beta, \ddot{x}} T' \quad (\alpha, \beta) \in \mathcal{R}}{S, T \xrightarrow{\tau} S', T'} \quad \frac{S \xrightarrow{\mu} S' \quad \text{diff}(S, S') \cap \text{en}(T) = \emptyset}{S, T \xrightarrow{\mu} S', T}$$

plus the symmetric rule for groups

bioK-calculus: the transition relation

notation: $\text{diff}(S, S') \stackrel{\text{def}}{=} (\text{en}(S') \setminus \text{en}(S)) \cup (\text{en}(S) \setminus \text{en}(S'))$

$\alpha \in \mathcal{R}^\bullet$ if there is β such that $(\alpha, \beta) \in \mathcal{R}$ (similarly for \mathcal{R}°)

the **transition relation** $\xrightarrow{\mu}$ is the least one satisfying the reductions:

- protein-protein reductions**

$$\frac{(A, i, \phi) \in \mathcal{R}^\bullet \quad x \notin \text{en}(\sigma)}{A(i + \phi + \sigma) \xrightarrow{A, i, \phi, \dot{x}} A(i^x + \bar{\phi} + \sigma)} \quad \frac{(A, i, \phi) \in \mathcal{R}^\circ}{A(i^x + \phi + \sigma) \xrightarrow{A, i, \phi, \ddot{x}} A(i + \bar{\phi} + \sigma)}$$

$$\frac{S \xrightarrow{\alpha, \ddot{x}} S' \quad T \xrightarrow{\beta, \ddot{x}} T' \quad (\alpha, \beta) \in \mathcal{R}^{\circ\bullet}}{S, T \xrightarrow{\tau} S', T'} \quad \frac{S \xrightarrow{\mu} S' \quad \text{diff}(S, S') \cap \text{en}(T) = \emptyset}{S, T \xrightarrow{\mu} S', T}$$

plus the symmetric rule for groups

bioK-calculus: the transition relation

- protein-membrane reductions

$$\frac{M \xrightarrow{\alpha, \overset{\circ}{x}} M' \quad S \xrightarrow{\beta, \overset{\circ}{x}} S' \quad (\alpha, \beta) \in \mathcal{R}}{m(M)[S] \xrightarrow{\tau} m(M')[S']}$$

- cellular reductions

$$\frac{S \xrightarrow{\tau} S' \quad \text{diff}(S, S') \cap \text{en}(M) = \emptyset}{m(M)[S] \xrightarrow{\tau} m(M)[S']} \quad \frac{M \xrightarrow{\mu} M' \quad \text{diff}(M, M') \cap \text{en}(S) = \emptyset}{m(M)[S] \xrightarrow{\mu} m(M')[S]}$$

remarks 1. every edge name created in a complexation is fresh

- the interaction between a protein outside a cell and the membrane of the cell is modelled by the last rule and the reaction rule
- the nucleus of a cell cannot interact with agents external to the cell
- the reductions do not change the cellular structure – core bioK

bioK-calculus: the transition relation

- protein-membrane reductions

$$\frac{M \xrightarrow{\alpha, \overset{\circ}{x}} M' \quad S \xrightarrow{\beta, \overset{\circ}{x}} S' \quad (\alpha, \beta) \in \mathcal{R}}{m(M)[S] \xrightarrow{\tau} m(M')[S']}$$

- cellular reductions

$$\frac{S \xrightarrow{\tau} S' \quad \text{diff}(S, S') \cap \text{en}(M) = \emptyset}{m(M)[S] \xrightarrow{\tau} m(M)[S']} \quad \frac{M \xrightarrow{\mu} M' \quad \text{diff}(M, M') \cap \text{en}(S) = \emptyset}{m(M)[S] \xrightarrow{\mu} m(M')[S]}$$

remarks 1. every edge name created in a complexation is fresh

- the interaction between a protein outside a cell and the membrane of the cell is modelled by the last rule and the reaction rule
- the nucleus of a cell cannot interact with agents external to the cell
- the reductions do not change the cellular structure – core bioK

bioK-calculus: the transition relation

- protein-membrane reductions

$$\frac{M \xrightarrow{\alpha, \overset{\circ}{x}} M' \quad S \xrightarrow{\beta, \overset{\circ}{x}} S' \quad (\alpha, \beta) \in \mathcal{R}}{m(M)[S] \xrightarrow{\tau} m(M')[S']}$$

- cellular reductions

$$\frac{S \xrightarrow{\tau} S' \quad \text{diff}(S, S') \cap \text{en}(M) = \emptyset}{m(M)[S] \xrightarrow{\tau} m(M)[S']} \quad \frac{M \xrightarrow{\mu} M' \quad \text{diff}(M, M') \cap \text{en}(S) = \emptyset}{m(M)[S] \xrightarrow{\mu} m(M')[S]}$$

remarks 1. every edge name created in a complexation is fresh

- the interaction between a protein outside a cell and the membrane of the cell is modelled by the last rule and the reaction rule
- the nucleus of a cell cannot interact with agents external to the cell
- the reductions do not change the cellular structure – core bioK

bioK-calculus: the transition relation

- protein-membrane reductions

$$\frac{M \xrightarrow{\alpha, \overset{\circ}{x}} M' \quad S \xrightarrow{\beta, \overset{\circ}{x}} S' \quad (\alpha, \beta) \in \mathcal{R}}{m(M)[S] \xrightarrow{\tau} m(M')[S']}$$

- cellular reductions

$$\frac{S \xrightarrow{\tau} S' \quad \text{diff}(S, S') \cap \text{en}(M) = \emptyset}{m(M)[S] \xrightarrow{\tau} m(M)[S']} \quad \frac{M \xrightarrow{\mu} M' \quad \text{diff}(M, M') \cap \text{en}(S) = \emptyset}{m(M)[S] \xrightarrow{\mu} m(M')[S]}$$

remarks 1. every edge name created in a complexation is fresh

- the interaction between a protein outside a cell and the membrane of the cell is modelled by the last rule and the reaction rule
- the nucleus of a cell cannot interact with agents external to the cell
- the reductions do not change the cellular structure – core bioK

bioK-calculus: the transition relation

- protein-membrane reductions

$$\frac{M \xrightarrow{\alpha, \overset{\circ}{x}} M' \quad S \xrightarrow{\beta, \overset{\circ}{x}} S' \quad (\alpha, \beta) \in \mathcal{R}}{m(M)[S] \xrightarrow{\tau} m(M')[S']}$$

- cellular reductions

$$\frac{S \xrightarrow{\tau} S' \quad \text{diff}(S, S') \cap \text{en}(M) = \emptyset}{m(M)[S] \xrightarrow{\tau} m(M)[S']} \quad \frac{M \xrightarrow{\mu} M' \quad \text{diff}(M, M') \cap \text{en}(S) = \emptyset}{m(M)[S] \xrightarrow{\mu} m(M')[S]}$$

remarks 1. every edge name created in a complexation is fresh

- the interaction between a protein outside a cell and the membrane of the cell is modelled by the last rule and the reaction rule
- the nucleus of a cell cannot interact with agents external to the cell
- the reductions do not change the cellular structure – core bioK

bioK-calculus: the RTK-MAPK pathway

1. a dimeric form of the epidermal growth factor EGF binds two receptors EGFR located on some plasmic membrane
2. the receptors EGFR cross-phosphorylate each other through their tyrosine kinase sites
3. then the EGFR activate another binding site that binds an adapter protein SHC and activate it
4. ... the signal goes further till reaching the nucleus

the formal rendering

$$\begin{aligned} ((\text{EGF}, 1, \bar{2}), (\text{EGF}, 1, \bar{2})) &\in \mathcal{R} & (1) \\ ((\text{EGF}, 2, \emptyset), (\text{EGFR}, 1, \bar{4})) &\in \mathcal{R} & (2) \\ ((\text{EGFR}, 2, \bar{3} + 4), (\text{EGFR}, 2, \bar{3} + 4)) &\in \mathcal{R} & (3) \\ ((\text{EGFR}, 2, \emptyset), (\text{EGFR}, 2, \emptyset)) &\in \mathcal{R} & (3') \\ ((\text{EGFR}, 3, \emptyset), (\text{SHC}, 1, \bar{2})) &\in \mathcal{R} & (4) \end{aligned}$$

bioK-calculus: the RTK-MAPK pathway

1. a dimeric form of the epidermal growth factor EGF binds two receptors EGFR located on some plasmic membrane
2. the receptors EGFR cross-phosphorylate each other through their tyrosine kinase sites
3. then the EGFR activate another binding site that binds an adapter protein SHC and activate it
4. ... the signal goes further till reaching the nucleus

the formal rendering

$$\begin{aligned} ((\text{EGF}, 1, \bar{2}), (\text{EGF}, 1, \bar{2})) &\in \overset{\bullet}{\mathcal{R}} & (1) \\ ((\text{EGF}, 2, \emptyset), (\text{EGFR}, 1, \bar{4})) &\in \overset{\bullet}{\mathcal{R}} & (2) \\ ((\text{EGFR}, 2, \bar{3} + 4), (\text{EGFR}, 2, \bar{3} + 4)) &\in \overset{\bullet}{\mathcal{R}} & (3) \\ ((\text{EGFR}, 2, \emptyset), (\text{EGFR}, 2, \emptyset)) &\in \overset{\circ}{\mathcal{R}} & (3') \\ ((\text{EGFR}, 3, \emptyset), (\text{SHC}, 1, \bar{2})) &\in \overset{\bullet}{\mathcal{R}} & (4) \end{aligned}$$

bioK-calculus: the RTK-MAPK pathway

$\text{EGF}(1 + \bar{2}), \text{EGF}(1 + \bar{2}),$

$m(\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S]$

$\xrightarrow{\tau} \text{EGF}(1^z + 2), \text{EGF}(1^z + 2),$

$m(\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S]$ *by (1)*

$\xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2),$

$m(\text{EGFR}(1^y + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S]$ *by (2)*

$\xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u),$

$m(\text{EGFR}(1^y + 2 + \bar{3} + \bar{4}), \text{EGFR}(1^u + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S]$ *by (2)*

$\xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u),$

$m(\text{EGFR}(1^y + 2^x + 3 + \bar{4}), \text{EGFR}(1^u + 2^x + 3 + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S]$ *by (3)*

$\xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u),$

$m(\text{EGFR}(1^y + 2 + 3 + \bar{4}), \text{EGFR}(1^u + 2 + 3 + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S]$ *by (3')*

$\xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u),$

$m(\text{EGFR}(1^y + 2 + 3^x + \bar{4}), \text{EGFR}(1^u + 2 + 3 + \bar{4}), M)[\text{SHC}(1^x + 2), S]$ *by (4)*

bioK-calculus: the RTK-MAPK pathway

$$\begin{aligned} & \text{EGF}(1 + \bar{2}), \text{EGF}(1 + \bar{2}), \\ & m(\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2), \text{EGF}(1^z + 2), \\ & \quad m(\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (1)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2), \\ & \quad m(\text{EGFR}(1^y + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (2)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2 + \bar{3} + \bar{4}), \text{EGFR}(1^u + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (2)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2^x + 3 + \bar{4}), \text{EGFR}(1^u + 2^x + 3 + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (3)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2 + 3 + \bar{4}), \text{EGFR}(1^u + 2 + 3 + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (3')} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2 + 3^x + \bar{4}), \text{EGFR}(1^u + 2 + 3 + \bar{4}), M)[\text{SHC}(1^x + 2), S] \quad \text{by (4)} \end{aligned}$$

bioK-calculus: the RTK-MAPK pathway

$\text{EGF}(1 + \bar{2}), \text{EGF}(1 + \bar{2}),$

$m(\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S]$

$\xrightarrow{\tau} \text{EGF}(1^z + 2), \text{EGF}(1^z + 2),$

$m(\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (1)}$

$\xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2),$

$m(\text{EGFR}(1^y + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (2)}$

$\xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u),$

$m(\text{EGFR}(1^y + 2 + \bar{3} + \bar{4}), \text{EGFR}(1^u + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (2)}$

$\xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u),$

$m(\text{EGFR}(1^y + 2^x + 3 + \bar{4}), \text{EGFR}(1^u + 2^x + 3 + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (3)}$

$\xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u),$

$m(\text{EGFR}(1^y + 2 + 3 + \bar{4}), \text{EGFR}(1^u + 2 + 3 + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (3')}$

$\xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u),$

$m(\text{EGFR}(1^y + 2 + 3^x + \bar{4}), \text{EGFR}(1^u + 2 + 3 + \bar{4}), M)[\text{SHC}(1^x + 2), S] \quad \text{by (4)}$

bioK-calculus: the RTK-MAPK pathway

$$\begin{aligned} & \text{EGF}(1 + \bar{2}), \text{EGF}(1 + \bar{2}), \\ & m(\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2), \text{EGF}(1^z + 2), \\ & \quad m(\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (1)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2), \\ & \quad m(\text{EGFR}(1^y + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (2)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2 + \bar{3} + \bar{4}), \text{EGFR}(1^u + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (2)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2^x + 3 + \bar{4}), \text{EGFR}(1^u + 2^x + 3 + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (3)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2 + 3 + \bar{4}), \text{EGFR}(1^u + 2 + 3 + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (3')} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2 + 3^x + \bar{4}), \text{EGFR}(1^u + 2 + 3 + \bar{4}), M)[\text{SHC}(1^x + 2), S] \quad \text{by (4)} \end{aligned}$$

bioK-calculus: the RTK-MAPK pathway

$$\begin{aligned} & \text{EGF}(1 + \bar{2}), \text{EGF}(1 + \bar{2}), \\ & m(\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2), \text{EGF}(1^z + 2), \\ & \quad m(\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (1)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2), \\ & \quad m(\text{EGFR}(1^y + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (2)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2 + \bar{3} + \bar{4}), \text{EGFR}(1^u + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (2)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2^x + 3 + \bar{4}), \text{EGFR}(1^u + 2^x + 3 + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (3)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2 + 3 + \bar{4}), \text{EGFR}(1^u + 2 + 3 + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (3')} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2 + 3^x + \bar{4}), \text{EGFR}(1^u + 2 + 3 + \bar{4}), M)[\text{SHC}(1^x + 2), S] \quad \text{by (4)} \end{aligned}$$

bioK-calculus: the RTK-MAPK pathway

$$\begin{aligned} & \text{EGF}(1 + \bar{2}), \text{EGF}(1 + \bar{2}), \\ & m(\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2), \text{EGF}(1^z + 2), \\ & \quad m(\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (1)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2), \\ & \quad m(\text{EGFR}(1^y + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (2)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2 + \bar{3} + \bar{4}), \text{EGFR}(1^u + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (2)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2^x + 3 + \bar{4}), \text{EGFR}(1^u + 2^x + 3 + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (3)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2 + 3 + \bar{4}), \text{EGFR}(1^u + 2 + 3 + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (3')} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2 + 3^x + \bar{4}), \text{EGFR}(1^u + 2 + 3 + \bar{4}), M)[\text{SHC}(1^x + 2), S] \quad \text{by (4)} \end{aligned}$$

bioK-calculus: the RTK-MAPK pathway

$$\begin{aligned} & \text{EGF}(1 + \bar{2}), \text{EGF}(1 + \bar{2}), \\ & m(\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2), \text{EGF}(1^z + 2), \\ & \quad m(\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (1)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2), \\ & \quad m(\text{EGFR}(1^y + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (2)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2 + \bar{3} + \bar{4}), \text{EGFR}(1^u + 2 + \bar{3} + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (2)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2^x + 3 + \bar{4}), \text{EGFR}(1^u + 2^x + 3 + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (3)} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2 + 3 + \bar{4}), \text{EGFR}(1^u + 2 + 3 + \bar{4}), M)[\text{SHC}(1 + \bar{2}), S] \quad \text{by (3')} \\ & \xrightarrow{\tau} \text{EGF}(1^z + 2^y), \text{EGF}(1^z + 2^u), \\ & \quad m(\text{EGFR}(1^y + 2 + 3^x + \bar{4}), \text{EGFR}(1^u + 2 + 3 + \bar{4}), M)[\text{SHC}(1^x + 2), S] \quad \text{by (4)} \end{aligned}$$

bioK-calculus: expressivity problems

1. a EGFR pair connects to two different EGF dimers

$$\begin{aligned} & \text{EGF}(1 + \bar{2}), \text{EGF}(1 + \bar{2}), \text{EGF}(1 + \bar{2}), \text{EGF}(1 + \bar{2}), \\ & m(|\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M|)[\text{SHC}(1 + \bar{2}), S] \\ & \xrightarrow{\tau^2} \text{EGF}(1^x + 2), \text{EGF}(1^x + 2), \text{EGF}(1^y + 2), \text{EGF}(1^y + 2), \\ & \quad m(|\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M|)[\text{SHC}(1 + \bar{2}), S] \\ & \xrightarrow{\tau^2} \text{EGF}(1^x + 2), \text{EGF}(1^x + 2^u), \text{EGF}(1^y + 2^v), \text{EGF}(1^y + 2), \\ & \quad m(|\text{EGFR}(1^u + 2 + \bar{3} + \bar{4}), \text{EGFR}(1^v + 2 + \bar{3} + \bar{4}), M|)[\text{SHC}(1 + \bar{2}), S] \end{aligned}$$

a way out: in mobile-k we used session identifiers and pattern matching over them

2. *translocation*: at the end of the RTK-MAPK pathway there is a phosphorylation of an *extra-cellular receptor kinase* – the protein ERK1 – that enters in the nucleus

bioK-calculus: expressivity problems

1. a EGFR pair connects to two different EGF dimers

$$\begin{aligned} & \text{EGF}(1 + \bar{2}), \text{EGF}(1 + \bar{2}), \text{EGF}(1 + \bar{2}), \text{EGF}(1 + \bar{2}), \\ & m(|\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M|)[\text{SHC}(1 + \bar{2}), S] \\ & \xrightarrow{\tau^2} \text{EGF}(1^x + 2), \text{EGF}(1^x + 2), \text{EGF}(1^y + 2), \text{EGF}(1^y + 2), \\ & \quad m(|\text{EGFR}(1 + 2 + \bar{3} + \bar{4}), \text{EGFR}(1 + 2 + \bar{3} + \bar{4}), M|)[\text{SHC}(1 + \bar{2}), S] \\ & \xrightarrow{\tau^2} \text{EGF}(1^x + 2), \text{EGF}(1^x + 2^u), \text{EGF}(1^y + 2^v), \text{EGF}(1^y + 2), \\ & \quad m(|\text{EGFR}(1^u + 2 + \bar{3} + \bar{4}), \text{EGFR}(1^v + 2 + \bar{3} + \bar{4}), M|)[\text{SHC}(1 + \bar{2}), S] \end{aligned}$$

a way out: in mobile-k we used session identifiers and pattern matching over them

2. *translocation*: at the end of the RTK-MAPK pathway there is a phosphorylation of an *extra-cellular receptor kinase* – the protein ERK1 – that enters in the nucleus

bioK-calculus: the bisimilarity

- $S \xRightarrow{\tau} S'$ means $S \xrightarrow{\tau}^* S'$
- $S \xRightarrow{\mu} S'$, with $\mu \neq \tau$, means $S \xrightarrow{\tau}^* \xrightarrow{\mu} \xrightarrow{\tau}^* S'$
- $\alpha \in \mathcal{R}^{\circ\bullet}$ means there is β such that $(\alpha, \beta) \in \mathcal{R}^{\circ\bullet}$

A (weak) bisimulation is a symmetric binary relation τ between solutions such that $S \tau T$ implies:

1. if $S \xrightarrow{\tau} S'$ then $T \xRightarrow{\tau} T'$ and $S' \tau T'$;
2. if $\alpha \in \mathcal{R}^{\bullet}$ and $S \xrightarrow{\alpha, \dot{x}} S'$ then $T \xRightarrow{\alpha, \dot{y}} T'$ and $S' \tau T'$;
3. if $\alpha \in \mathcal{R}^{\circ}$ and $S \xrightarrow{\alpha, \dot{x}} S'$ then $T \xRightarrow{\alpha, \dot{y}} T'$ and $S' \tau T'$.

$S \approx T$ if $S \tau T$ for some bisimulation τ .

bioK-calculus: the bisimilarity

- $S \xRightarrow{\tau} S'$ means $S \xrightarrow{\tau}^* S'$
- $S \xRightarrow{\mu} S'$, with $\mu \neq \tau$, means $S \xrightarrow{\tau}^* \xrightarrow{\mu} \xrightarrow{\tau}^* S'$
- $\alpha \in \mathcal{R}^{\circ\bullet}$ means there is β such that $(\alpha, \beta) \in \mathcal{R}^{\circ\bullet}$

A (weak) bisimulation is a symmetric binary relation τ between solutions such that $S \tau T$ implies:

1. if $S \xrightarrow{\tau} S'$ then $T \xRightarrow{\tau} T'$ and $S' \tau T'$;
2. if $\alpha \in \mathcal{R}^{\bullet}$ and $S \xrightarrow{\alpha, \dot{x}} S'$ then $T \xRightarrow{\alpha, \dot{y}} T'$ and $S' \tau T'$;
3. if $\alpha \in \mathcal{R}^{\circ}$ and $S \xrightarrow{\alpha, \overset{\circ}{x}} S'$ then $T \xRightarrow{\alpha, \overset{\circ}{y}} T'$ and $S' \tau T'$.

$S \approx T$ if $S \tau T$ for some bisimulation τ .

bioK-calculus: properties

1. “ , ” is an abelian monoidal operator with identity $\mathbf{0}$:

$$S, T \approx T, S \quad (S, T), R \approx S, (T, R) \quad S, \mathbf{0} \approx S$$

2. \approx is preserved by injective renamings: let ι be an injective renaming on \mathcal{E} , then $S \approx \iota(S)$.
3. \approx is a congruence
4. the semantics of proteins and cells is defined by observing their own ongoing reduction capabilities, **without analyzing possible contexts**

bioK-calculus: properties

1. “ , ” is an abelian monoidal operator with identity **0**:

$$S, T \approx T, S \quad (S, T), R \approx S, (T, R) \quad S, \mathbf{0} \approx S$$

2. \approx is preserved by injective renamings: let ι be an injective renaming on \mathcal{E} , then $S \approx \iota(S)$.
3. \approx is a congruence
4. the semantics of proteins and cells is defined by observing their own ongoing reduction capabilities, **without analyzing possible contexts**

bioK-calculus: properties

1. “ , ” is an abelian monoidal operator with identity $\mathbf{0}$:

$$S, T \approx T, S \quad (S, T), R \approx S, (T, R) \quad S, \mathbf{0} \approx S$$

2. \approx is preserved by injective renamings: let ι be an injective renaming on \mathcal{E} , then $S \approx \iota(S)$.
3. \approx is a congruence
4. the semantics of proteins and cells is defined by observing their own ongoing reduction capabilities, **without analyzing possible contexts**

bioK-calculus: properties

1. “ , ” is an abelian monoidal operator with identity $\mathbf{0}$:

$$S, T \approx T, S \quad (S, T), R \approx S, (T, R) \quad S, \mathbf{0} \approx S$$

2. \approx is preserved by injective renamings: let ι be an injective renaming on \mathcal{E} , then $S \approx \iota(S)$.
3. \approx is a congruence
4. the semantics of proteins and cells is defined by observing their own ongoing reduction capabilities, **without analyzing possible contexts**

bioK-calculus: properties

1. “ , ” is an abelian monoidal operator with identity $\mathbf{0}$:

$$S, T \approx T, S \quad (S, T), R \approx S, (T, R) \quad S, \mathbf{0} \approx S$$

2. \approx is preserved by injective renamings: let ι be an injective renaming on \mathcal{E} , then $S \approx \iota(S)$.
3. \approx is a congruence
4. the semantics of proteins and cells is defined by observing their own ongoing reduction capabilities, **without analyzing possible contexts**

merging and splitting membranes

- core `bioK` does preserve the cellular structure of the solution
- it is not possible to describe phenomena such as *endosome fusions* :

$$esm(M)[S] , esm(N)[T] \longrightarrow esm(M, N)[S, T]$$

question: how to define the semantics of an endosome, regardless of endosomes in the context?

- **answer:** by means of higher-order mechanisms

merging and splitting membranes

- core `bioK` does preserve the cellular structure of the solution
- it is not possible to describe phenomena such as *endosome fusions* :

$$esm(M)[S] , esm(N)[T] \longrightarrow esm(M, N)[S, T]$$

question: how to define the semantics of an endosome, regardless of endosomes in the context?

– answer: by means of higher-order mechanisms

merging and splitting membranes

- core `bioK` does preserve the cellular structure of the solution
- it is not possible to describe phenomena such as *endosome fusions* :

$$esm(M)[S] , esm(N)[T] \longrightarrow esm(M, N)[S, T]$$

question: how to define the semantics of an endosome, regardless of endosomes in the context?

- **answer:** by means of higher-order mechanisms

the core `bioK`-calculus with mreagents

the syntax of `bioK`-calculus:

$S ::=$	solution
0	(empty solution)
$A(\sigma)$	(protein)
$m(\langle M \rangle)[S]$	(cell)
S, S	(group)
$\langle M; S \rangle \cdot S$	(mreagents)

constraint: in $\langle M; S \rangle \cdot T$

- S and T do not contain mreagents
- M is a multiset of proteins
- $\text{de}(S) \subseteq \text{de}(M)$

extending the transition relation

- the **fusion function** \mathcal{F} is a symmetric binary function on membrane types – $\mathcal{F}(m, n) = \mathcal{F}(n, m)$
- the **m-activations** $\overset{\bullet}{\mathcal{A}}$ and $\overset{\circ}{\mathcal{A}}$ are functions $(\alpha, m) \mapsto n$
- write $m \in \mathcal{F}$ if there is n such that $(m, n) \in \text{dom}(\mathcal{F})$

the **transition relation** $\xrightarrow{\mu}$ is the least one that also contains (μ also ranges over m):

$$\frac{m \in \mathcal{F}}{m(M) [S] \xrightarrow{m} \langle M; S \rangle \cdot \emptyset}$$

$$\frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S'}{S, T \xrightarrow{m} \langle M; S'' \rangle \cdot (S', T)}$$

$$\frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S' \quad T \xrightarrow{n} \langle N; T'' \rangle \cdot T' \quad \mathcal{F}(m, n) = m'}{S, T \xrightarrow{\tau} S', T', m'(M, N) [S'', T'']}$$

$$\frac{S \xrightarrow{m} \langle N; T \rangle \cdot S' \quad \mathcal{F}(m, n) = m'}{m(M) [S] \xrightarrow{\tau} m'(M, N) [S'], T}$$

$$\frac{\begin{array}{c} M \xrightarrow{\tau} M' \\ \text{diff}(M, M') \cap \text{en}(S) = \emptyset \end{array}}{m(M) [S] \xrightarrow{\tau} m(M') [S]}$$

$$\frac{\begin{array}{c} M \xrightarrow{\alpha, \overset{\bullet}{x}} M' \quad \overset{\circ}{\mathcal{A}}(\alpha, m) = n \\ \text{diff}(M, M') \cap \text{en}(S) = \emptyset \end{array}}{m(M) [S] \xrightarrow{\alpha, \overset{\bullet}{x}} n(M') [S]}$$

extending the transition relation

- the **fusion function** \mathcal{F} is a symmetric binary function on membrane types – $\mathcal{F}(m, n) = \mathcal{F}(n, m)$
- the **m-activations** $\overset{\bullet}{\mathcal{A}}$ and $\overset{\circ}{\mathcal{A}}$ are functions $(\alpha, m) \mapsto n$
- write $m \in \mathcal{F}$ if there is n such that $(m, n) \in \text{dom}(\mathcal{F})$

the **transition relation** $\xrightarrow{\mu}$ is the least one that also contains (μ also ranges over m):

$$\frac{m \in \mathcal{F}}{m(M) [S] \xrightarrow{m} \langle M; S \rangle \cdot \emptyset}$$

$$\frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S'}{S, T \xrightarrow{m} \langle M; S'' \rangle \cdot (S', T)}$$

$$\frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S' \quad T \xrightarrow{n} \langle N; T'' \rangle \cdot T' \quad \mathcal{F}(m, n) = m'}{S, T \xrightarrow{\tau} S', T', m'(M, N) [S'', T'']}$$

$$\frac{S \xrightarrow{m} \langle N; T \rangle \cdot S' \quad \mathcal{F}(m, n) = m'}{m(M) [S] \xrightarrow{\tau} m'(M, N) [S'], T}$$

$$S, T \xrightarrow{\tau} S', T', m'(M, N) [S'', T'']$$

$$m(M) [S] \xrightarrow{\tau} m'(M, N) [S'], T$$

$$M \xrightarrow{\tau} M'$$

$$M \xrightarrow{\alpha, \overset{\circ}{x}} M' \quad \overset{\circ}{\mathcal{A}}(\alpha, m) = n$$

$$\text{diff}(M, M') \cap \text{en}(S) = \emptyset$$

$$\text{diff}(M, M') \cap \text{en}(S) = \emptyset$$

$$\frac{M \xrightarrow{\tau} M' \quad \text{diff}(M, M') \cap \text{en}(S) = \emptyset}{m(M) [S] \xrightarrow{\tau} m(M') [S]}$$

$$\frac{M \xrightarrow{\alpha, \overset{\circ}{x}} M' \quad \overset{\circ}{\mathcal{A}}(\alpha, m) = n \quad \text{diff}(M, M') \cap \text{en}(S) = \emptyset}{m(M) [S] \xrightarrow{\alpha, \overset{\circ}{x}} n(M') [S]}$$

extending the transition relation

- the **fusion function** \mathcal{F} is a symmetric binary function on membrane types – $\mathcal{F}(m, n) = \mathcal{F}(n, m)$
- the **m-activations** $\overset{\bullet}{\mathcal{A}}$ and $\overset{\circ}{\mathcal{A}}$ are functions $(\alpha, m) \mapsto n$
- write $m \in \mathcal{F}$ if there is n such that $(m, n) \in \text{dom}(\mathcal{F})$

the **transition relation** $\xrightarrow{\mu}$ is the least one that also contains (μ also ranges over m):

$$\frac{m \in \mathcal{F}}{m(M) [S] \xrightarrow{m} \langle M; S \rangle \cdot \emptyset}$$

$$\frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S'}{S, T \xrightarrow{m} \langle M; S'' \rangle \cdot (S', T)}$$

$$\frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S' \quad T \xrightarrow{n} \langle N; T'' \rangle \cdot T' \quad \mathcal{F}(m, n) = m'}{S, T \xrightarrow{\tau} S', T', m'(M, N) [S'', T'']}$$

$$\frac{S \xrightarrow{m} \langle N; T \rangle \cdot S' \quad \mathcal{F}(m, n) = m'}{m(M) [S] \xrightarrow{\tau} m'(M, N) [S'], T}$$

$$S, T \xrightarrow{\tau} S', T', m'(M, N) [S'', T'']$$

$$m(M) [S] \xrightarrow{\tau} m'(M, N) [S'], T$$

$$M \xrightarrow{\tau} M'$$

$$M \xrightarrow{\alpha, \overset{\circ}{x}} M' \quad \overset{\circ}{\mathcal{A}}(\alpha, m) = n$$

$$\text{diff}(M, M') \cap \text{en}(S) = \emptyset$$

$$\text{diff}(M, M') \cap \text{en}(S) = \emptyset$$

$$\frac{M \xrightarrow{\tau} M' \quad \text{diff}(M, M') \cap \text{en}(S) = \emptyset}{m(M) [S] \xrightarrow{\tau} m(M') [S]}$$

$$m(M) [S] \xrightarrow{\alpha, \overset{\circ}{x}} n(M') [S]$$

extending the transition relation

- the **fusion function** \mathcal{F} is a symmetric binary function on membrane types – $\mathcal{F}(m, n) = \mathcal{F}(n, m)$
- the **m-activations** $\overset{\bullet}{\mathcal{A}}$ and $\overset{\circ}{\mathcal{A}}$ are functions $(\alpha, m) \mapsto n$
- write $m \in \mathcal{F}$ if there is n such that $(m, n) \in \text{dom}(\mathcal{F})$

the **transition relation** $\xrightarrow{\mu}$ is the least one that also contains (μ also ranges over m):

$$\frac{m \in \mathcal{F}}{m(M) [S] \xrightarrow{m} \langle M; S \rangle \cdot \emptyset}$$

$$\frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S'}{S, T \xrightarrow{m} \langle M; S'' \rangle \cdot (S', T)}$$

$$\frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S' \quad T \xrightarrow{n} \langle N; T'' \rangle \cdot T' \quad \mathcal{F}(m, n) = m'}{S, T \xrightarrow{\tau} S', T', m'(M, N) [S'', T'']}$$

$$\frac{S \xrightarrow{m} \langle N; T \rangle \cdot S' \quad \mathcal{F}(m, n) = m'}{m(M) [S] \xrightarrow{\tau} m'(M, N) [S'], T}$$

$$\frac{\begin{array}{c} M \xrightarrow{\tau} M' \\ \text{diff}(M, M') \cap \text{en}(S) = \emptyset \end{array}}{m(M) [S] \xrightarrow{\tau} m(M') [S]}$$

$$\frac{\begin{array}{c} M \xrightarrow{\alpha, \overset{\circ}{x}} M' \quad \overset{\circ}{\mathcal{A}}(\alpha, m) = n \\ \text{diff}(M, M') \cap \text{en}(S) = \emptyset \end{array}}{m(M) [S] \xrightarrow{\alpha, \overset{\circ}{x}} n(M') [S]}$$

extending the transition relation

- the **fusion function** \mathcal{F} is a symmetric binary function on membrane types – $\mathcal{F}(m, n) = \mathcal{F}(n, m)$
- the **m-activations** $\overset{\bullet}{\mathcal{A}}$ and $\overset{\circ}{\mathcal{A}}$ are functions $(\alpha, m) \mapsto n$
- write $m \in \mathcal{F}$ if there is n such that $(m, n) \in \text{dom}(\mathcal{F})$

the **transition relation** $\xrightarrow{\mu}$ is the least one that also contains (μ also ranges over m):

$$\frac{m \in \mathcal{F}}{m(\langle M \rangle) [S] \xrightarrow{m} \langle M; S \rangle \cdot \mathbf{0}} \quad \frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S'}{S, T \xrightarrow{m} \langle M; S'' \rangle \cdot (S', T)}$$

$$\frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S' \quad T \xrightarrow{n} \langle N; T'' \rangle \cdot T' \quad \mathcal{F}(m, n) = m'}{S, T \xrightarrow{\tau} S', T', m'(\langle M, N \rangle) [S'', T'']}$$

$$\frac{\begin{array}{c} M \xrightarrow{\tau} M' \\ \text{diff}(M, M') \cap \text{en}(S) = \emptyset \end{array}}{m(\langle M \rangle) [S] \xrightarrow{\tau} m(\langle M' \rangle) [S]}$$

$$\frac{S \xrightarrow{m} \langle N; T \rangle \cdot S' \quad \mathcal{F}(m, n) = m'}{m(\langle M \rangle) [S] \xrightarrow{\tau} m'(\langle M, N \rangle) [S'], T}$$

$$\frac{M \xrightarrow{\alpha, \overset{\circ}{x}} M' \quad \overset{\circ}{\mathcal{A}}(\alpha, m) = n \quad \text{diff}(M, M') \cap \text{en}(S) = \emptyset}{m(\langle M \rangle) [S] \xrightarrow{\alpha, \overset{\circ}{x}} n(\langle M' \rangle) [S]}$$

$$m(\langle M \rangle) [S] \xrightarrow{\alpha, \overset{\circ}{x}} n(\langle M' \rangle) [S]$$

extending the transition relation

- the **fusion function** \mathcal{F} is a symmetric binary function on membrane types – $\mathcal{F}(m, n) = \mathcal{F}(n, m)$
- the **m-activations** $\overset{\bullet}{\mathcal{A}}$ and $\overset{\circ}{\mathcal{A}}$ are functions $(\alpha, m) \mapsto n$
- write $m \in \mathcal{F}$ if there is n such that $(m, n) \in \text{dom}(\mathcal{F})$

the **transition relation** $\xrightarrow{\mu}$ is the least one that also contains (μ also ranges over m):

$$\frac{m \in \mathcal{F}}{m(\langle M \rangle) [S] \xrightarrow{m} \langle M; S \rangle \cdot \mathbf{0}} \quad \frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S'}{S, T \xrightarrow{m} \langle M; S'' \rangle \cdot (S', T)}$$

$$\frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S' \quad T \xrightarrow{n} \langle N; T'' \rangle \cdot T' \quad \mathcal{F}(m, n) = m'}{S, T \xrightarrow{\tau} S', T', m'(\langle M, N \rangle) [S'', T'']}$$

$$\frac{S \xrightarrow{m} \langle N; T \rangle \cdot S' \quad \mathcal{F}(m, n) = m'}{m(\langle M \rangle) [S] \xrightarrow{\tau} m'(\langle M, N \rangle) [S'], T}$$

$$\frac{M \xrightarrow{\tau} M' \quad \text{diff}(M, M') \cap \text{en}(S) = \emptyset}{m(\langle M \rangle) [S] \xrightarrow{\tau} m(\langle M' \rangle) [S]}$$

$$\frac{M \xrightarrow{\alpha, \overset{\bullet}{x}} M' \quad \overset{\circ}{\mathcal{A}}(\alpha, m) = n \quad \text{diff}(M, M') \cap \text{en}(S) = \emptyset}{m(\langle M \rangle) [S] \xrightarrow{\alpha, \overset{\bullet}{x}} n(\langle M' \rangle) [S]}$$

$$m(\langle M \rangle) [S] \xrightarrow{\alpha, \overset{\bullet}{x}} n(\langle M' \rangle) [S]$$

extending the transition relation

- the **fusion function** \mathcal{F} is a symmetric binary function on membrane types – $\mathcal{F}(m, n) = \mathcal{F}(n, m)$
- the **m-activations** $\overset{\bullet}{\mathcal{A}}$ and $\overset{\circ}{\mathcal{A}}$ are functions $(\alpha, m) \mapsto n$
- write $m \in \mathcal{F}$ if there is n such that $(m, n) \in \text{dom}(\mathcal{F})$

the **transition relation** $\xrightarrow{\mu}$ is the least one that also contains (μ also ranges over m):

$$\frac{m \in \mathcal{F}}{m \langle M \rangle [S] \xrightarrow{m} \langle M; S \rangle \cdot \emptyset}$$

$$\frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S'}{S, T \xrightarrow{m} \langle M; S'' \rangle \cdot (S', T)}$$

$$\frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S' \quad T \xrightarrow{n} \langle N; T'' \rangle \cdot T' \quad \mathcal{F}(m, n) = m'}{S, T \xrightarrow{\tau} S', T', m' \langle M, N \rangle [S'', T'']}$$

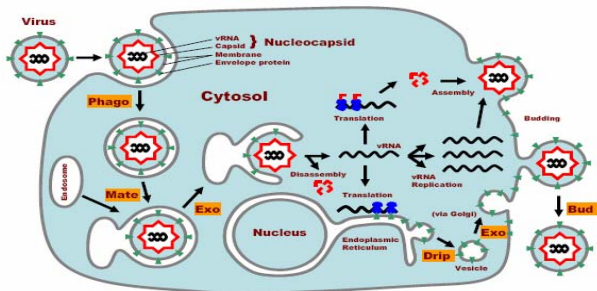
$$\frac{S \xrightarrow{m} \langle N; T \rangle \cdot S' \quad \mathcal{F}(m, n) = m'}{m \langle M \rangle [S] \xrightarrow{\tau} m' \langle M, N \rangle [S'], T}$$

$$\frac{\begin{array}{c} M \xrightarrow{\tau} M' \\ \text{diff}(M, M') \cap \text{en}(S) = \emptyset \end{array}}{m \langle M \rangle [S] \xrightarrow{\tau} m \langle M' \rangle [S]}$$

$$\frac{\begin{array}{c} M \xrightarrow{\alpha, \overset{\bullet}{x}} M' \quad \overset{\circ}{\mathcal{A}}(\alpha, m) = n \\ \text{diff}(M, M') \cap \text{en}(S) = \emptyset \end{array}}{m \langle M \rangle [S] \xrightarrow{\alpha, \overset{\bullet}{x}} n \langle M' \rangle [S]}$$

example: a virus infection

the virus is a parasite which needs a host to develop: it uses the infected cell replication machinery in order to duplicate its own genetic material



– the virus is modelled by: $Virus = vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]$

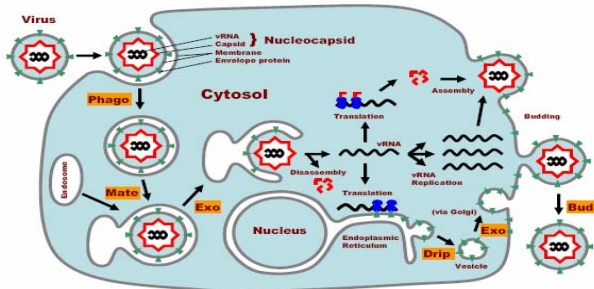
– the cell is modelled by

$Cell = cll(M) [edsm(E(1), E_m) [E_s], Cytosol, icll(v(1), N) [Virus]]$

assumption: the virus has already infected the cell – we are missing the *phagocytosis*

example: a virus infection

the virus is a parasite which needs a host to develop: it uses the infected cell replication machinery in order to duplicate its own genetic material



– the virus is modelled by: $\text{Virus} = \text{vs}(\text{V}_M) [\text{cpsd}(\mathbf{0}) [\text{V}_{\text{RNA}}]]$

– the cell is modelled by

$\text{Cell} = \text{cII}(\text{M}) [\text{edsm}(\text{E}(1), \text{E}_m) [\text{E}_s], \text{Cytosol}, \text{icII}(\text{V}(1), \text{N}) [\text{Virus}]]$

assumption: the virus has already infected the cell – we are missing the *phagocytosis*

example: a virus infection

let:

$$\begin{aligned}(E, 1, \emptyset, edsm) &\mapsto iedsm && \in \mathcal{A} \\ ((v, 1, \emptyset), (E, 1, \emptyset)) &&& \in \mathcal{R} \\ (icll, iedsm) &\mapsto iedsmv && \in \mathcal{F} \\ (iedsmv, vs) &\mapsto edsm && \in \mathcal{F}\end{aligned}$$

a run:

$$cll(M) \left[\begin{array}{l} edsm(E(1), E_m) [E_s], \text{Cytosol}, \\ icll(v(1), N) [vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]] \end{array} \right]$$

$$\xrightarrow{\tau} cll(M) \left[\begin{array}{l} iedsm(E(1^x), E_m) [E_s], \text{Cytosol}, \\ icll(v(1^x), N) [vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]] \end{array} \right]$$

$$\xrightarrow{\tau} cll(M) \left[\begin{array}{l} iedsmv(E(1^x), E_m, v(1^x), N) [E_s, vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]], \\ \text{Cytosol} \end{array} \right]$$

$$\xrightarrow{\tau} cll(M) \left[\begin{array}{l} edsm(E(1^x), E_m, v(1^x), N, V_M) [E_s,], \\ cpsd(\mathbf{0}) [V_{RNA}], \text{Cytosol} \end{array} \right]$$

example: a virus infection

let:

$$\begin{aligned}(E, 1, \emptyset, edsm) &\mapsto iedsm && \in \mathcal{A} \\ ((V, 1, \emptyset), (E, 1, \emptyset)) &&& \in \mathcal{R} \\ (icll, iedsm) &\mapsto iedsmv && \in \mathcal{F} \\ (iedsmv, vs) &\mapsto edsm && \in \mathcal{F}\end{aligned}$$

a run:

$$cll(M) \left[\begin{array}{l} edsm(E(1), E_m) [E_s], \text{Cytosol}, \\ icll(V(1), N) [vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]] \end{array} \right]$$

$$\xrightarrow{\tau} cll(M) \left[\begin{array}{l} iedsm(E(1^x), E_m) [E_s], \text{Cytosol}, \\ icll(V(1^x), N) [vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]] \end{array} \right]$$

$$\xrightarrow{\tau} cll(M) \left[\begin{array}{l} iedsmv(E(1^x), E_m, V(1^x), N) [E_s, vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]], \\ \text{Cytosol} \end{array} \right]$$

$$\xrightarrow{\tau} cll(M) \left[\begin{array}{l} edsm(E(1^x), E_m, V(1^x), N, V_M) [E_s,], \\ cpsd(\mathbf{0}) [V_{RNA}], \text{Cytosol} \end{array} \right]$$

example: a virus infection

let:

$$\begin{aligned}(E, 1, \emptyset, edsm) &\mapsto iedsm && \in \mathcal{A} \\ ((V, 1, \emptyset), (E, 1, \emptyset)) &&& \in \mathcal{R} \\ (icll, iedsm) &\mapsto iedsmv && \in \mathcal{F} \\ (iedsmv, vs) &\mapsto edsm && \in \mathcal{F}\end{aligned}$$

a run:

$$cll(M) \left[\begin{array}{l} edsm(E(1), E_m) [E_s], \text{Cytosol}, \\ icll(V(1), N) [vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]] \end{array} \right]$$

$$\xrightarrow{\tau} cll(M) \left[\begin{array}{l} iedsm(E(1^x), E_m) [E_s], \text{Cytosol}, \\ icll(V(1^x), N) [vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]] \end{array} \right]$$

$$\xrightarrow{\tau} cll(M) \left[\begin{array}{l} iedsmv(E(1^x), E_m, V(1^x), N) [E_s, vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]], \\ \text{Cytosol} \end{array} \right],$$

$$\xrightarrow{\tau} cll(M) \left[\begin{array}{l} edsm(E(1^x), E_m, V(1^x), N, V_M) [E_s,], \\ cpsd(\mathbf{0}) [V_{RNA}], \text{Cytosol} \end{array} \right]$$

example: a virus infection

let:

$$\begin{aligned}(E, 1, \emptyset, edsm) &\mapsto iedsm && \in \mathcal{A} \\ ((V, 1, \emptyset), (E, 1, \emptyset)) &&& \in \mathcal{R} \\ (icll, iedsm) &\mapsto iedsmv && \in \mathcal{F} \\ (iedsmv, vs) &\mapsto edsm && \in \mathcal{F}\end{aligned}$$

a run:

$$cll(M) \left[\begin{array}{l} edsm(E(1), E_m) [E_s], \text{Cytosol}, \\ icll(V(1), N) [vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]] \end{array} \right]$$

$$\xrightarrow{\tau} cll(M) \left[\begin{array}{l} iedsm(E(1^x), E_m) [E_s], \text{Cytosol}, \\ icll(V(1^x), N) [vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]] \end{array} \right]$$

$$\xrightarrow{\tau} cll(M) \left[\begin{array}{l} iedsmv(E(1^x), E_m, V(1^x), N) [E_s, vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]], \\ \text{Cytosol} \end{array} \right],$$

$$\xrightarrow{\tau} cll(M) \left[\begin{array}{l} edsm(E(1^x), E_m, V(1^x), N, V_M) [E_s,], \\ cpsd(\mathbf{0}) [V_{RNA}], \text{Cytosol} \end{array} \right]$$

example: a virus infection

let:

$$\begin{aligned}(E, 1, \emptyset, edsm) &\mapsto iedsm && \in \mathcal{A} \\ ((V, 1, \emptyset), (E, 1, \emptyset)) &&& \in \mathcal{R} \\ (icll, iedsm) &\mapsto iedsmv && \in \mathcal{F} \\ (iedsmv, vs) &\mapsto edsm && \in \mathcal{F}\end{aligned}$$

a run:

$$cll(M) \left[\begin{array}{l} edsm(E(1), E_m) [E_s], \text{Cytosol}, \\ icll(V(1), N) [vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]] \end{array} \right]$$

$$\xrightarrow{\tau} cll(M) \left[\begin{array}{l} iedsm(E(1^x), E_m) [E_s], \text{Cytosol}, \\ icll(V(1^x), N) [vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]] \end{array} \right]$$

$$\xrightarrow{\tau} cll(M) \left[\begin{array}{l} iedsmv(E(1^x), E_m, V(1^x), N) [E_s, vs(V_M) [cpsd(\mathbf{0}) [V_{RNA}]]], \\ \text{Cytosol} \end{array} \right],$$

$$\xrightarrow{\tau} cll(M) \left[\begin{array}{l} edsm(E(1^x), E_m, V(1^x), N, V_M) [E_s,], \\ cpsd(\mathbf{0}) [V_{RNA}], \text{Cytosol} \end{array} \right]$$

context bisimilarity

a **context bisimulation** is a symmetric binary relation τ between solutions such that $S \tau T$ implies:

1. if $S \xrightarrow{\tau} S'$ then $T \xrightarrow{\tau} T'$ and $S' \tau T'$
2. if $\alpha \in \mathcal{R}$ and $S \xrightarrow{\alpha, \dot{x}} S'$ then $T \xrightarrow{\alpha, \dot{y}} T'$ and $S' \tau T'$
3. if $\alpha \in \mathcal{R}$ and $S \xrightarrow{\alpha, \circ x} S'$ then $T \xrightarrow{\alpha, \circ y} T'$ and $S' \tau T'$
4. if $m \in \mathcal{F}$ and $S \xrightarrow{m} \langle M; S'' \rangle \cdot S'$ then $T \xrightarrow{m} \langle N; T'' \rangle \cdot T'$ and, for every n, M', R :
 - if $\mathcal{F}(m, n) = m'$ then
 $(S', m'(\langle M, M' \rangle [S'', R])) \tau (T', m'(\langle N, M' \rangle [T'', R]))$ and
 $(S', m'(\langle M, M' \rangle [S''])) \tau (T', m'(\langle N, M' \rangle [T'']))$

$S \approx T$ if $S \tau T$ for some context bisimulation τ .

remark: activation is not mentioned!

context bisimilarity

a **context bisimulation** is a symmetric binary relation τ between solutions such that $S \tau T$ implies:

1. if $S \xrightarrow{\tau} S'$ then $T \xRightarrow{\tau} T'$ and $S' \tau T'$
2. if $\alpha \in \mathcal{R}$ and $S \xrightarrow{\alpha, \dot{x}} S'$ then $T \xRightarrow{\alpha, \dot{y}} T'$ and $S' \tau T'$
3. if $\alpha \in \mathcal{R}$ and $S \xrightarrow{\alpha, \circ x} S'$ then $T \xRightarrow{\alpha, \circ y} T'$ and $S' \tau T'$
4. if $m \in \mathcal{F}$ and $S \xrightarrow{m} \langle M; S'' \rangle \cdot S'$ then $T \xRightarrow{m} \langle N; T'' \rangle \cdot T'$ and, for every n, M', R :
 - if $\mathcal{F}(m, n) = m'$ then
 $(S', m'(\langle M, M' \rangle [S'', R])) \tau (T', m'(\langle N, M' \rangle [T'', R]))$ and
 $(S', m'(\langle M, M' \rangle [S''])) \tau (T', m'(\langle N, M' \rangle [T'']))$

$S \approx T$ if $S \tau T$ for some context bisimulation τ .

remark: activation is not mentioned!

context bisimilarity

a **context bisimulation** is a symmetric binary relation τ between solutions such that $S \tau T$ implies:

1. if $S \xrightarrow{\tau} S'$ then $T \xrightarrow{\tau} T'$ and $S' \tau T'$
2. if $\alpha \in \mathcal{R}$ and $S \xrightarrow{\alpha, \dot{x}} S'$ then $T \xrightarrow{\alpha, \dot{y}} T'$ and $S' \tau T'$
3. if $\alpha \in \mathcal{R}$ and $S \xrightarrow{\alpha, \circ x} S'$ then $T \xrightarrow{\alpha, \circ y} T'$ and $S' \tau T'$
4. if $m \in \mathcal{F}$ and $S \xrightarrow{m} \langle M; S'' \rangle \cdot S'$ then $T \xrightarrow{m} \langle N; T'' \rangle \cdot T'$ and, for every n, M', R :
 - if $\mathcal{F}(m, n) = m'$ then
 $(S', m'(\langle M, M' \rangle [S'', R])) \tau (T', m'(\langle N, M' \rangle [T'', R]))$ and
 $(S', m'(\langle M, M' \rangle [S''])) \tau (T', m'(\langle N, M' \rangle [T'']))$

$S \approx T$ if $S \tau T$ for some context bisimulation τ .

remark: activation is not mentioned!

context bisimilarity

a **context bisimulation** is a symmetric binary relation τ between solutions such that $S \tau T$ implies:

1. if $S \xrightarrow{\tau} S'$ then $T \xrightarrow{\tau} T'$ and $S' \tau T'$
2. if $\alpha \in \mathcal{R}$ and $S \xrightarrow{\alpha, \dot{x}} S'$ then $T \xrightarrow{\alpha, \dot{y}} T'$ and $S' \tau T'$
3. if $\alpha \in \mathcal{R}$ and $S \xrightarrow{\alpha, \circ x} S'$ then $T \xrightarrow{\alpha, \circ y} T'$ and $S' \tau T'$
4. if $m \in \mathcal{F}$ and $S \xrightarrow{m} \langle M; S'' \rangle \cdot S'$ then $T \xrightarrow{m} \langle N; T'' \rangle \cdot T'$ and, for every n, M', R :
 - if $\mathcal{F}(m, n) = m'$ then
 $(S', m'(\langle M, M' \rangle [S'', R])) \tau (T', m'(\langle N, M' \rangle [T'', R]))$ and
 $(S', m'(\langle M, M' \rangle [S''])) \tau (T', m'(\langle N, M' \rangle [T'']))$

$S \approx T$ if $S \tau T$ for some context bisimulation τ .

remark: activation is not mentioned!

properties of context bisimilarity

1. “ $,$ ” is an abelian monoidal operator with identity $\mathbf{0}$:
2. \approx is a congruence
3. context bisimilarity retains a universal quantification that is hard to check

still to be done:

– properties that reduce the quantifications:

– instead of \forall we may test:

$\forall x \in \mathcal{C}, \forall \sigma \in \Sigma^{\mathcal{C}} \text{ and } \sigma \dashv \{x, \sigma'\} \text{ if } \sigma \dashv \sigma' \text{ then } \sigma \dashv \sigma' \text{ if } \sigma \dashv \sigma'$

$\forall x \in \mathcal{C}, \forall \sigma \in \Sigma^{\mathcal{C}}$

$\sigma \dashv \sigma \text{ if } \sigma \dashv \sigma$

$\sigma \dashv \sigma \text{ if } \sigma \dashv \sigma$

$\sigma \dashv \sigma$

$\sigma \dashv \sigma$

properties of context bisimilarity

1. “ , ” is an abelian monoidal operator with identity **0**:
2. \approx is a congruence
3. context bisimilarity retains a universal quantification that is hard to check

still to be done:

– properties that reduce the quantifications:

– instead of \forall we may use:

$\exists x \in \mathcal{C} \text{ and } \exists \tau \in \mathcal{T} \text{ (where } \mathcal{C} \text{ is a context and } \mathcal{T} \text{ is a term)}$

$\exists x \in \mathcal{C} \text{ and } \exists \tau \in \mathcal{T} \text{ (where } \mathcal{C} \text{ is a context and } \mathcal{T} \text{ is a term)}$

$\exists x \in \mathcal{C} \text{ and } \exists \tau \in \mathcal{T} \text{ (where } \mathcal{C} \text{ is a context and } \mathcal{T} \text{ is a term)}$

$\exists x \in \mathcal{C} \text{ and } \exists \tau \in \mathcal{T} \text{ (where } \mathcal{C} \text{ is a context and } \mathcal{T} \text{ is a term)}$

$\exists x \in \mathcal{C} \text{ and } \exists \tau \in \mathcal{T} \text{ (where } \mathcal{C} \text{ is a context and } \mathcal{T} \text{ is a term)}$

$\exists x \in \mathcal{C} \text{ and } \exists \tau \in \mathcal{T} \text{ (where } \mathcal{C} \text{ is a context and } \mathcal{T} \text{ is a term)}$

properties of context bisimilarity

1. “ , ” is an abelian monoidal operator with identity **0**:
2. \approx is a congruence
3. context bisimilarity retains a universal quantification that is hard to check

still to be done:

– properties that reduce the quantifications:

– instead of \forall we may say:

$\forall x \in \mathcal{C} \text{ and } \forall \sigma \in \Sigma \{ \sigma(x) \in \mathcal{C} \text{ and } \sigma(0) = 0 \}$ and

$\forall \sigma \in \Sigma \{ \sigma(0) = 0 \}$

– $\forall x \in \mathcal{C} \{ \sigma(x) \in \mathcal{C} \}$

– $\forall \sigma \in \Sigma \{ \sigma(x) \in \mathcal{C} \text{ and } \sigma(0) = 0 \}$

– $\forall \sigma \in \Sigma$

– $\forall \sigma \in \Sigma \{ \sigma(0) = 0 \}$

properties of context bisimilarity

1. “ , ” is an abelian monoidal operator with identity **0**:
2. \approx is a congruence
3. context bisimilarity retains a universal quantification that is hard to check

still to be done:

– properties that reduce the quantifications:

– instead of \forall we may say:

$\forall x \in \mathcal{C} \text{ and } \forall \sigma \in \Sigma \{ \sigma(x) \in \mathcal{C} \text{ and } \sigma(0) = 0 \}$ and

$\forall \sigma \in \Sigma \{ \sigma(0) = 0 \}$

– $\forall \sigma \in \Sigma \{ \sigma(0) = 0 \}$

– $\forall \sigma \in \Sigma \{ \sigma(0) = 0 \}$

– $\forall \sigma \in \Sigma$

– $\forall \sigma \in \Sigma$

properties of context bisimilarity

1. “ , ” is an abelian monoidal operator with identity **0**:
2. \approx is a congruence
3. context bisimilarity retains a universal quantification that is hard to check

still to be done:

– properties that reduce the quantifications:

- instead of 4 we may use:

if $m \in \mathcal{F}$ and $S \xrightarrow{m} \{M; S''\} \cdot S'$ then $T \xrightarrow{m} \{N; T''\} \cdot T'$ and
 $M \approx N, S'' \approx T'', S' \approx T'$

too much discriminating?

- constraining fusions when interacting cells have particular structures

too little discriminating?

properties of context bisimilarity

1. “, ” is an abelian monoidal operator with identity $\mathbf{0}$:
2. \approx is a congruence
3. context bisimilarity retains a universal quantification that is hard to check

still to be done:

– properties that reduce the quantifications:

- instead of 4 we may use:

if $m \in \mathcal{F}$ and $S \xrightarrow{m} \{M; S''\} \cdot S'$ then $T \xrightarrow{m} \{N; T''\} \cdot T'$ and
 $M \tau N, S'' \tau T'', S' \tau T'$

too much discriminating?

- constraining fusions when interacting cells have particular structures

too little discriminating?

properties of context bisimilarity

1. “, ” is an abelian monoidal operator with identity $\mathbf{0}$:
2. \approx is a congruence
3. context bisimilarity retains a universal quantification that is hard to check

still to be done:

– properties that reduce the quantifications:

- instead of 4 we may use:

if $m \in \mathcal{F}$ and $S \xrightarrow{m} \langle M; S'' \rangle \cdot S'$ then $T \xrightarrow{m} \langle N; T'' \rangle \cdot T'$ and

$M \tau N, S'' \tau T'', S' \tau T'$

too much discriminating?

- constraining fusions when interacting cells have particular structures

too little discriminating?

properties of context bisimilarity

1. “, ” is an abelian monoidal operator with identity $\mathbf{0}$:
2. \approx is a congruence
3. context bisimilarity retains a universal quantification that is hard to check

still to be done:

- properties that reduce the quantifications:
 - instead of 4 we may use:
if $m \in \mathcal{F}$ and $S \xrightarrow{m} \langle M; S'' \rangle \cdot S'$ then $T \xrightarrow{m} \langle N; T'' \rangle \cdot T'$ and
 $M \tau N, S'' \tau T'', S' \tau T'$
too much discriminating?
 - constraining fusions when interacting cells have particular structures
too little discriminating?

conclusions/1

- our approach is **algebraic**
 - reuse tools already developed for the pi-calculus
 - problems in the semantics of cells because of higher order / compare with alternative algebraic approaches such as *ambients*
- it is questionable how to extend the theory with biological reactions such as *phagocytosis*

$$\frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S' \quad T \xrightarrow{n} \langle N; T'' \rangle \cdot T' \quad \mathcal{P}(m, n) = m'}{S, T \xrightarrow{\tau} S', T', m(M) [m'(\mathbf{0}) [n(N) [T''] , S'']]}$$

- *translocations* are easier to define

conclusions/1

- our approach is **algebraic**
 - reuse tools already developed for the pi-calculus
 - problems in the semantics of cells because of higher order / compare with alternative algebraic approaches such as *ambients*
- it is questionable how to extend the theory with biological reactions such as *phagocytosis*

$$\frac{S \xrightarrow{m} \langle M; S'' \rangle \cdot S' \quad T \xrightarrow{n} \langle N; T'' \rangle \cdot T' \quad \mathcal{P}(m, n) = m'}{S, T \xrightarrow{\tau} S', T', m(\langle M \rangle) [m'(\mathbf{0}) [n(\langle N \rangle) [T''] , S'']]}$$

- *translocations* are easier to define

conclusions/2

- our modelling of membranes requires further analysis: we miss comparisons with other models (ambients)
- our semantics is parametrized by the database of protein/membrane reactions
 - is it possible to extrapolate properties out of such database?
 - perhaps, in most of the case, we do need a congruence wrt certain contexts
- (of course) we should also study extensions