Deducing Interactions in Partially Unspecified Biological Systems

P. Baldan[1] A. Bracciali[2] L. Brodo[3] R. Bruni[2]

[1]Università di Padova

[2]Università di Pisa

[3]Università di Sassari

Algebraic Biology - Hagenberg, July 2-4, 2007

Outline

1 Open Biological Systems

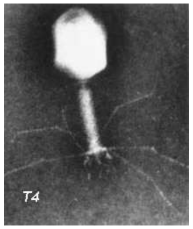
2 Open distributed and concurrent systems

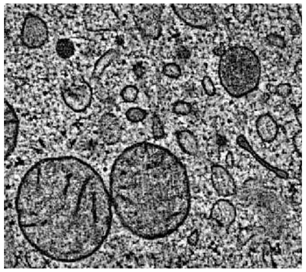
3 Symbolic Transition System

4 Biological examples

biological systems are *open* in different meanings

- **perturbable** - any living system expresses an external behaviour





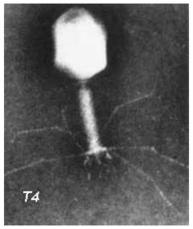
- **incomplete** - our knowledge of biological systems is incomplete

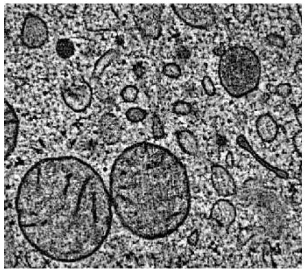
- **biotechnology** - we may always assume to add some external components to a given complex

Partial information imposes that the approach is to predict the behaviours (of missing knowledge).

biological systems are *open* in different meanings

- **perturbable** - any living system expresses an external behaviour



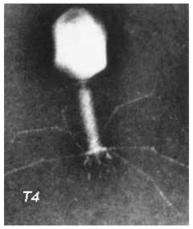


- **incomplete** - our knowledge of biological systems is incomplete

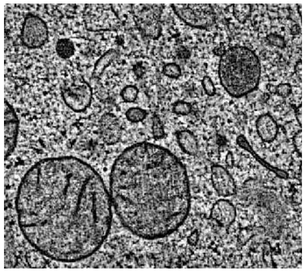
- **biotechnology** - we may always assume to add some external components to a given complex

Partial information imposes that the approach is to predict the behaviours (of missing knowledge).

biological systems are *open* in different meanings



- **perturbable** - any living system expresses an external behaviour



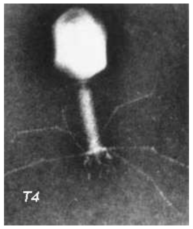
- **incomplete** - our knowledge of biological systems is incomplete

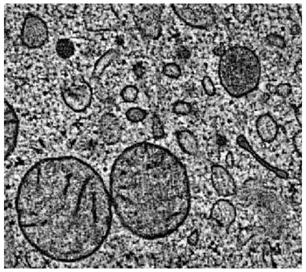
- **biotechnology** - we may always assume to add some external components to a given complex

Partial information imposes that the approach is to predict the behaviours (of missing knowledge).

biological systems are *open* in different meanings

- **perturbable** - any living system expresses an external behaviour





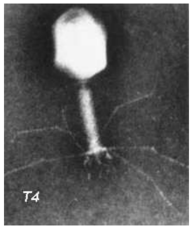
- **incomplete** - our knowledge of biological systems is incomplete

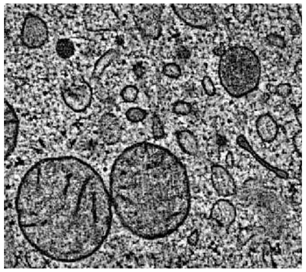
- **biotechnology** - we may always assume to add some external components to a given complex

Partial information imposes that the approach is to predict the behaviours (of missing knowledge).

biological systems are *open* in different meanings

- **perturbable** - any living system expresses an external behaviour





- **incomplete** - our knowledge of biological systems is incomplete

- **biotechnology** - we may always assume to add some external components to a given complex

Partial information imposes that the approach is to predict the behaviours (of missing knowledge).

partial biological knowledge, an abstraction

molecule - as - computation

Systems of interacting molecular entities are described and modelled by a system of interacting computational entities.
Regev, Shapiro - *Nature, September 2002*.

- **relevant**: essential properties
- **computable**: computational knowledge
- **understandable**: conceptual framework
- **extensible**: capture other real properties in the same mathematical framework

partial knowledge - as - openness

Modelling *incomplete systems* by applying a theory for *open* systems.

partial biological knowledge, an abstraction

molecule - as - computation

Systems of interacting molecular entities are described and modelled by a system of interacting computational entities.
Regev, Shapiro - *Nature, September 2002*.

- **relevant**: essential properties
- **computable**: computational knowledge
- **understandable**: conceptual framework
- **extensible**: capture other real properties in the same mathematical framework

partial knowledge - as - openness

Modelling *incomplete systems* by applying a theory for *open* systems.

partial biological knowledge, an abstraction

molecule - as - computation

Systems of interacting molecular entities are described and modelled by a system of interacting computational entities.
Regev, Shapiro - *Nature, September 2002*.

- **relevant**: essential properties
- **computable**: computational knowledge
- **understandable**: conceptual framework
- **extensible**: capture other real properties in the same mathematical framework

partial knowledge - as - openness

Modelling *incomplete systems* by applying a theory for *open* systems.

partial biological knowledge, an abstraction

molecule - as - computation

Systems of interacting molecular entities are described and modelled by a system of interacting computational entities.
Regev, Shapiro - *Nature, September 2002*.

- **relevant**: essential properties
- **computable**: computational knowledge
- **understandable**: conceptual framework
- **extensible**: capture other real properties in the same mathematical framework

partial knowledge - as - openness

Modelling *incomplete systems* by applying a theory for *open* systems.

Outline

1 Open Biological Systems

2 Open distributed and concurrent systems

3 Symbolic Transition System

4 Biological examples

Modelling open distributed and concurrent systems

Main ideas:

- open systems can evolve (*no universal closure, higher level of dynamics*);
- constructive methodology (*unification based - the most general unifier is choosed to infer the transition*);
- generality and friendly notation;
- complementary with contextualization techniques (*using contexts as labels to derive LTS, for which bisimilarity is a congruence*);

The advantage is to import:

- theories
- automatic tools

Modelling open distributed and concurrent systems

Main ideas:

- open systems can evolve (*no universal closure, higher level of dynamics*);
- constructive methodology (*unification based - the most general unifier is choosed to infer the transition*);
- generality and friendly notation;
- complementary with contextualization techniques (*using contexts as labels to derive LTS, for which bisimilarity is a congruence*);

The advantage is to import:

- theories
- automatic tools

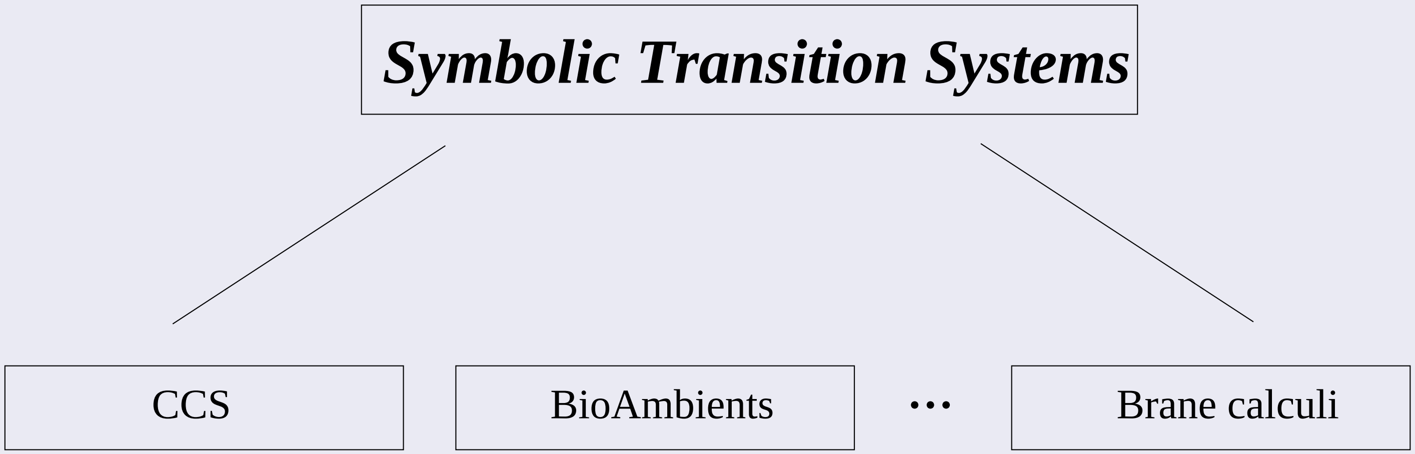
Modelling language

Process algebras are specification languages defined over a set basic actions and operators to compose them.

- general approach (basic action definition depend on the phenomenon to model)
- easy-to-use formalism

Not linked to a specific language

The theory for *Open System* we use is appliable to a language family.



Symbolic Transition Systems

CCS

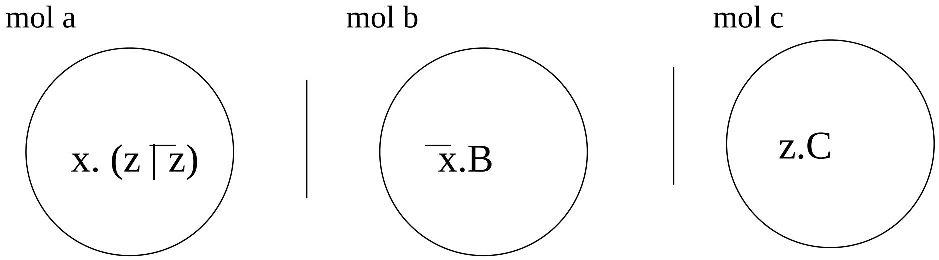
BioAmbients

···

Brane calculi

Process calculi

Process algebra terms model interacting entities



process calculi

CCS	action, co-action
BioAmbients	[. . .], *enter a*, *accept a*, . . .
Brane Calculi	. . . , phagocytosis, exocytosis, . . .
. . .	. . .

Contexts

Main ingredients: contexts and process variables.

Contexts: processes with holes

$$P[X_1]|Q[X_2]$$

The evolution of the system depends on the particular components will substitute X_1 and X_2.

Computational steps, involving external components, require some hypothesis to fix the particular behaviour needed to compute the desired action.

A simulation trace is equipped with the required hypothesis.

Outline

1. Open Biological Systems

2. Open distributed and concurrent systems

3. **Symbolic Transition System**

4. Biological examples

Transitions of open systems

Required behaviour

$$Q[X] \to Q'[Y]$$

Recording

- the hypothesis assumed at each computation step for each unspecified component;
- how the computation is modified by the new assumption taken

by using formulas as labels:

$$\varphi ::= X \mid \diamond a\, \varphi \mid f(\varphi_1, \ldots, \varphi_n)$$

Formulae as labels

General structrure of labels:

$$C[X_1, \ldots, X_n] \xrightarrow{(\varphi_1,\ldots,\varphi_n)}_a D[Y_1, \ldots, Y_m]$$

The formula labelling a transition, from one state to another, expresses the structural and behavioural assumptions on the unspecified components in order to perform the transition.

formulae

Formula definition depends on the particular process algebra adopted:

- basic actions
- structural operators
- composition operators

Simulating systems

Operational semantics

Inference rules

$$\frac{}{m[in\ n.Q \mid R] \ \mid\ n[P] \ \rightarrow_{\tau}\ n[m[Q \mid R] \ \mid P]} \quad (in\ capability)$$

$$\frac{}{n[P] \mid open\ n.Q \ \rightarrow_{\tau}\ P \mid Q} \quad (open\ capability)$$

$$\frac{P_1 \ \rightarrow_{\alpha}\ Q_1 \quad P_2 \ \rightarrow_{\alpha^{\perp}}\ Q_2}{P_1 \mid P_2 \ \rightarrow_{\tau}\ Q_1 \mid Q_2} \quad (communication)$$

specific formula set

$$\varphi ::= X \ \mid\ \diamond in\ n\ \varphi \ \mid\ \diamond open\ n\ \varphi \ \mid\ (\varphi_1 \mid \varphi_2)$$

Simulating systems

Operational semantics

Inference rules

$$\frac{}{m[in\ n.Q \mid R] \ \mid\ n[P] \ \rightarrow_{\tau}\ n[m[Q \mid R] \ \mid P]}\ (in\ capability)$$

$$\frac{}{n[P] \mid open\ n.Q \ \rightarrow_{\tau}\ P \mid Q}\ (open\ capability)$$

$$\frac{P_1 \ \rightarrow_{\alpha}\ Q_1 \quad P_2 \ \rightarrow_{\alpha^{\perp}}\ Q_2}{P_1 \mid P_2 \ \rightarrow_{\tau}\ Q_1 \mid Q_2}\ (communication)$$

specifi c formula set

$$\varphi ::= X \ \mid\ \diamond in\, n\, \varphi \ \mid\ \diamond open\, n\, \varphi \ \mid\ (\varphi_1 \ \mid\ \varphi_2)$$

Symbolic Transition

$$v[\ X\ ] \mid c[open\ v.(prot \mid rna^{\perp})]$$

initial state

If we want the *virus* $v[\dots]$ to enter the *cell* $c[\dots]$, we have to assumethat it has the capability to do that via a suitable action.

$$\xrightarrow{in\ c.Y|Z}_{\tau}$$

label

The formula captures the most general shape of the virus capable to enter the cell $c[\dots]$!

$$c[v[Y \mid Z] \mid open\ v.(prot \mid rna^{\perp})]$$

residual

In the next state (i.e. the continuation) there is a modification of the configuration of the process variables, due to the assumpions made.

Symbolic Transition

$$v[\ X\ ] \mid c[open\ v.(prot \mid rna^{\perp})]$$

initial state

If we want the *virus* $v[\ldots]$ to enter the *cell* $c[\ldots]$, we have to assumethat it has the capability to do that via a suitable action.

$$\xrightarrow{in\ c.Y|Z}_{\tau}$$

label

The formula captures the most general shape of the virus capable to enter the cell $c[\ldots]$!

$$c[v[Y \mid Z] \mid open\ v.(prot \mid rna^{\perp})]$$

residual

In the next state (i.e. the continuation) there is a modification of the configuration of the process variables, due to the assumpions made.

Symbolic Transition

$$v[\ X\ ] \mid c[open\ v.(prot \mid rna^{\perp})]$$

initial state

If we want the *virus* $v[\dots]$ to enter the *cell* $c[\dots]$, we have to assumethat it has the capability to do that via a suitable action.

$$\xrightarrow{in\ c.Y|Z}_{\tau}$$

label

The formula captures the most general shape of the virus capable to enter the cell $c[\dots]$!

$$c[v[Y \mid Z] \mid open\ v.(prot \mid rna^{\perp})]$$

residual

In the next state (i.e. the continuation) there is a modification of the configuration of the process variables, due to the assumpions made.

Symbolic Transition

$$v[\ X\ ] \mid c[open\ v.(prot \mid rna^{\perp})]$$

initial state

If we want the *virus* $v[\ldots]$ to enter the *cell* $c[\ldots]$, we have to assumethat it has the capability to do that via a suitable action.

$$\xrightarrow{in\ c.Y|Z}_{\tau}$$

label

The formula captures the most general shape of the virus capable to enter the cell $c[\ldots]$!

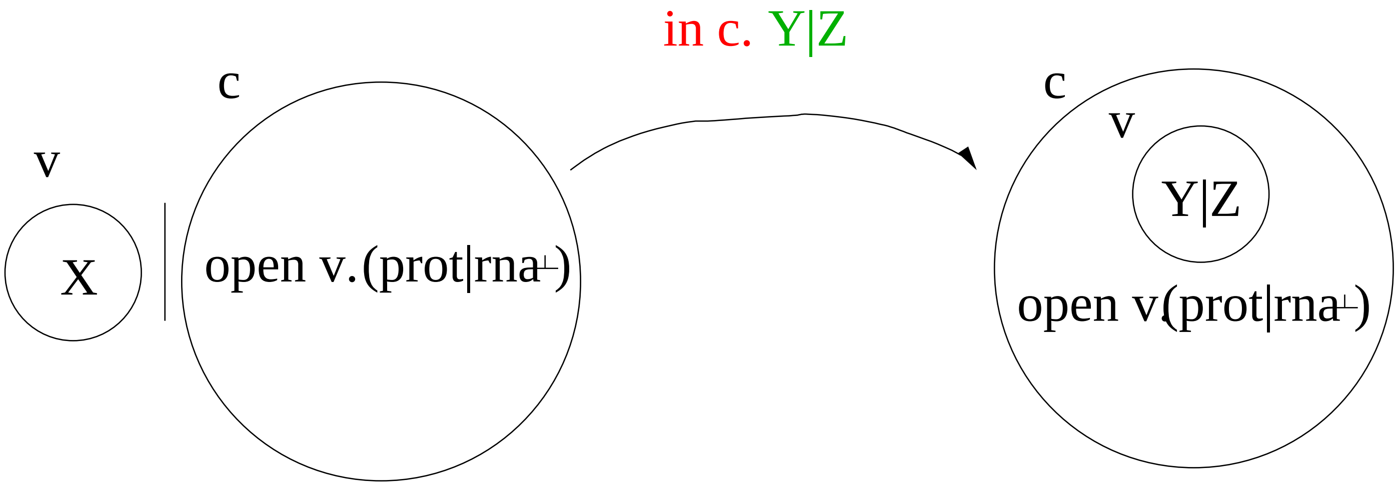
$$c[v[Y \mid Z] \mid open\ v.(prot \mid rna^{\perp})]$$

residual

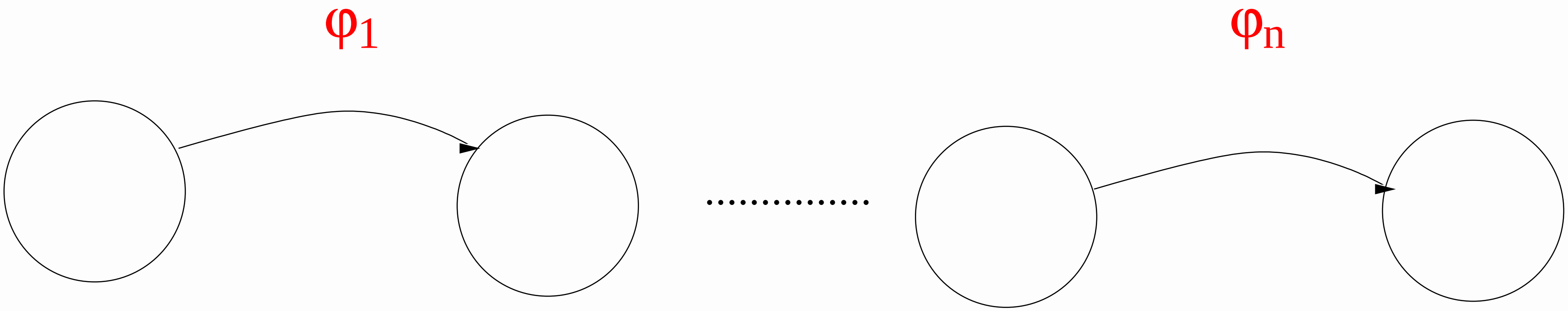
In the next state (i.e. the continuation) there is a modification of the configuration of the process variables, due to the assumpions made.

Symbolic Transition

At each step of the computation, new hypotheses are required

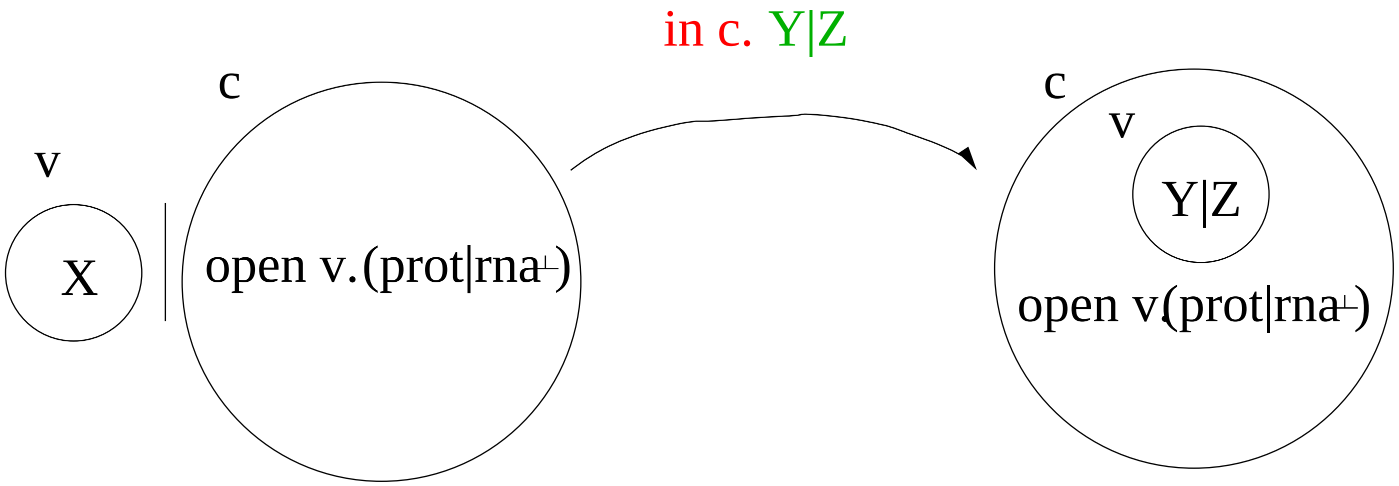


how to build a trace ?

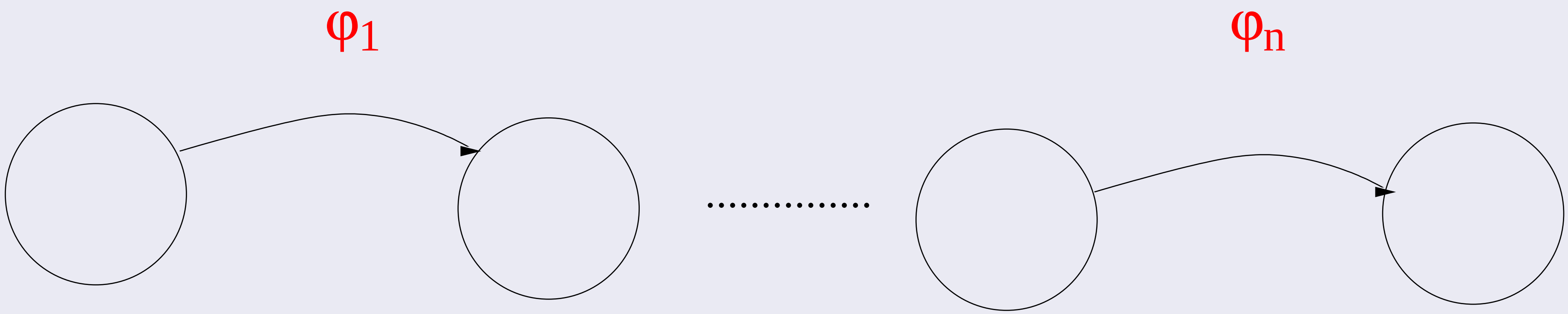


Symbolic Transition

At each step of the computation, new hypotheses are required



how to build a trace ?



Symbolic Trace

$$E[X] \; = \; v[\,X\,] \mid c[open\ v.(prot \mid rna^{\perp})]$$

first transition

$$E[X] \xrightarrow{in\ c.Y|Z}_{\tau} c[v[Y \mid Z] \mid open\ v.(prot \mid rna^{\perp})]$$

second transition

$$c[v[Y \mid Z] \mid open\ v.(prot \mid rna^{\perp})] \xrightarrow{Y,Z}_{\tau} c[Y \mid Z \mid prot \mid rna^{\perp}]$$

third transition

$$c[Y \mid Z \mid prot \mid rna^{\perp}] \xrightarrow{\diamond rna W,Z}_{c} [W \mid Z \mid prot]$$

label composition

$$in\ c.(\diamond\ rna\ W)|Y$$

Symbolic Trace

$$E[X] \;=\; v[\,X\,] \mid c[open\ v.(prot \mid rna^{\perp})]$$

first transition

$$E[X] \xrightarrow{in\ c.Y|Z}_{\tau} c[v[Y \mid Z] \mid open\ v.(prot \mid rna^{\perp})]$$

second transition

$$c[v[Y \mid Z] \mid open\ v.(prot \mid rna^{\perp})] \xrightarrow{Y,Z}_{\tau} c[Y \mid Z \mid prot \mid rna^{\perp}]$$

third transition

$$c[Y \mid Z \mid prot \mid rna^{\perp}] \xrightarrow{\diamond rnaW,Z}_{c} [W \mid Z \mid prot]$$

label composition

$$in\, c.(\diamond\, rna\ W)|Y$$

Symbolic Trace

$$E[X] \;=\; v[\,X\,] \mid c[open\ v.(prot \mid rna^{\perp})]$$

first transition

$$E[X] \xrightarrow{in\ c.Y|Z}_{\tau} c[v[Y \mid Z] \mid open\ v.(prot \mid rna^{\perp})]$$

second transition

$$c[v[Y \mid Z] \mid open\ v.(prot \mid rna^{\perp})] \xrightarrow{Y,Z}_{\tau} c[Y \mid Z \mid prot \mid rna^{\perp}]$$

third transition

$$c[Y \mid Z \mid prot \mid rna^{\perp}] \xrightarrow{\diamond rna W,Z}_{c} [W \mid Z \mid prot]$$

label composition

$$in\ c.(\diamond\ rna\ W)|Y$$

Symbolic Trace

$$E[X] \;=\; v[\,X\,] \mid c[open\ v.(prot \mid rna^{\perp})]$$

first transition

$$E[X] \xrightarrow{in\ c.Y|Z}_{\tau} c[v[Y \mid Z] \mid open\ v.(prot \mid rna^{\perp})]$$

second transition

$$c[v[Y \mid Z] \mid open\ v.(prot \mid rna^{\perp})] \xrightarrow{Y,Z}_{\tau} c[Y \mid Z \mid prot \mid rna^{\perp}]$$

third transition

$$c[Y \mid Z \mid prot \mid rna^{\perp}] \xrightarrow{\diamond rnaW,Z}_{c} [W \mid Z \mid prot]$$

label composition

$$in\ c.(\diamond\ rna\ W)|Y$$

Valid theory

- Correctness and completeness of Symbolic Transition Systems
  - **correctness** every concrete behaviour (of a full specified system) has an abstract representation (the corresponding in STS)
  - **completeness** every instance of an abstract behaviour (STS) correspond to a concrete behaviour
- symbolic bisimulation for open systems (different from universal closure)
- Simbolic Transition System is built by unification (Prolog)

bibliography

- Paolo Baldan, Andrea Bracciali, Roberto Bruni. **Bisimulation by unification**. AMAST 2002, LNCS vol.2422: 254-270.

Valid theory

- Correctness and completeness of Symbolic Transition Systems
  - **correctness** every concrete behaviour (of a full specified system) has an abstract representation (the corresponding in STS)
  - **completeness** every instance of an abstract behaviour (STS) correspond to a concrete behaviour
- symbolic bisimulation for open systems (different from universal closure)
- Simbolic Transition System is built by unification (Prolog)

bibliography

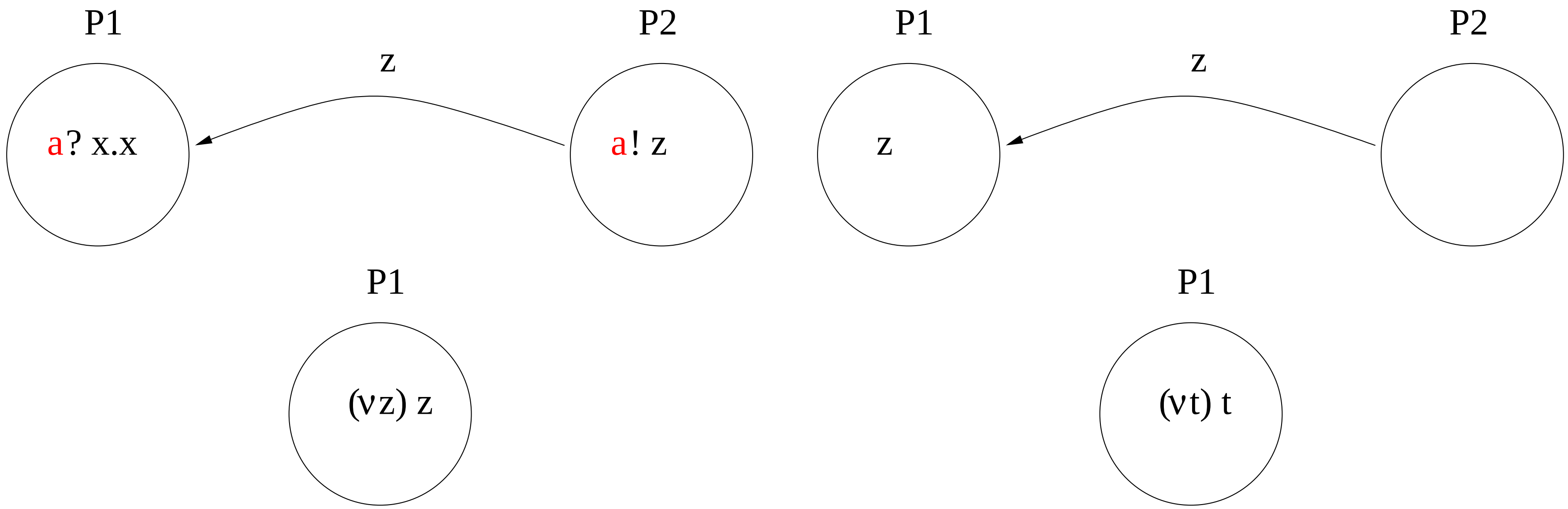
- Paolo Baldan, Andrea Bracciali, Roberto Bruni. **Bisimulation by unification**. AMAST 2002, LNCS vol.2422: 254-270.

On going work

STS works with a subset of the modeling languages: not able to handle names, **yet**!

π-calculus

$a\,!\,b$	output
$a\,?\,x$	input
$(\nu\,n)P$	private names



Outline

1 Open Biological Systems

2 Open distributed and concurrent systems

3 Symbolic Transition System

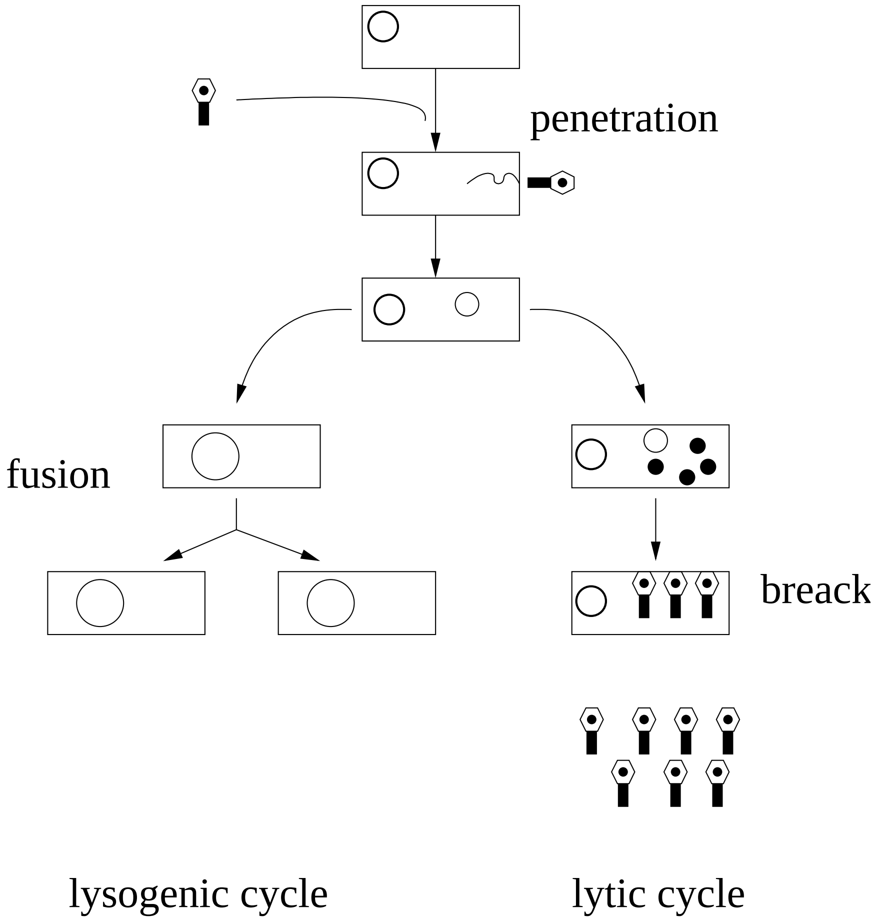
4 Biological examples

BioAmbients

Introduced for the modelling of biological membranes

$[\ldots]$	empty membrane
$[a?x \mid a!z]$	with internal behaviour
$[\cdots \mid a!z] \mid [a?x \mid \ldots]$	different membranes with internal behaviour
$[\cdots \mid \mathit{sibling}\ a!z] \mid [\mathit{sibling}\ a?x \mid \ldots]$	trespassing membranes
$[\cdots \mid \mathit{merge}^{+}\ a] \mid [\mathit{merge}^{-}\ a \mid \ldots]$	merging membranes
$[\cdots \mid \mathit{enter}\ a] \mid [\mathit{accept}\ a \mid \ldots]$	nesting membranes
$[[\cdots \mid \mathit{exit}\ a] \mid \mathit{allow}\ a \mid \ldots]$	nesting membranes

Toy example



- each protein is an ambient
- activation as communication
- inhibition as encapsulation

very simplified hyphoteses

1 - high concentration of **CI** determines lysogeny

2 - absence of **CI** determines lysis
3 - **CII** promotes the production of **CI**
4 - **CIII** can inhibit **HFL**
5 - low concentration of **CRO** stimulates **CI** production

6 - high concentration of **CRO** inhibits **CI** production

BioAmbient code

[VIRUS] = [merge^{+} virus.([C3] | [C2] | [C1] | [CRO]) | [DNAλ]]

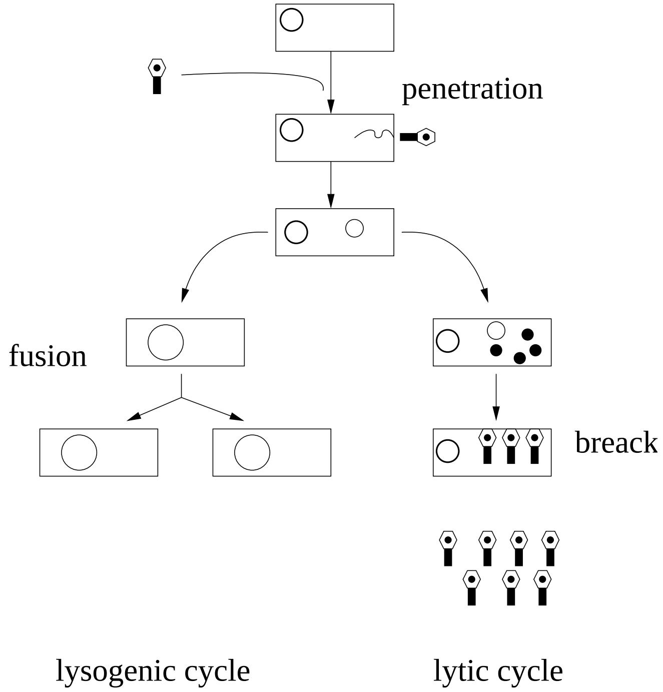
DNAλ = (lyso?.enter dnae.0)

+

lysi?.($_{\lambda}$[exit newph.VIRUS] | expel newph)

[ECOLI] = [merge^{-}virus | $_{Dna_e}$[accept dnae] | [HFL]]

HFL = enter h_c3.0 + X

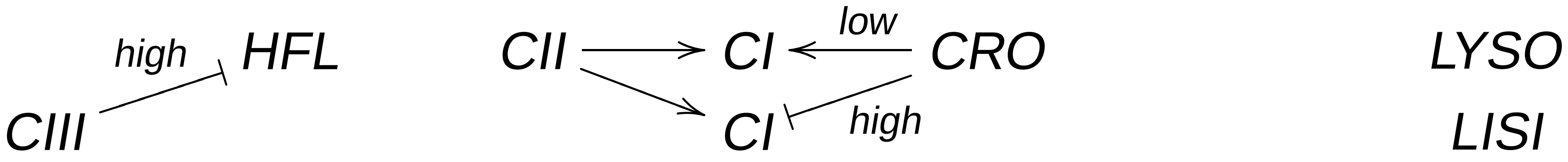


Bioambients - logical formulae

$$\varphi ::= X \mid \diamond a\, \varphi \mid \varphi_1 + \varphi_2 \mid \varphi_1 \,|\, \varphi_n \mid a.\varphi$$

$$a ::= n? \mid n! \mid enter\ n \mid accept\ n \mid \ldots$$

Assuming partial knowledge



C3 = l_c3!.0 + accept h_c3. pro_c2!.0
C2 = pro_c2?. pro_c1!.0 + enter c2.0

C1 = pro_c1?.(h_cro?. lysi!.0 + l_cro?. lyso!.0)
CRO = l_cro!.0 + h_cro.0

λ phago: lytic trace

$$_{Ecoli}[([C3] \mid [C2] \mid [C1] \mid [CRO])|[DNA\lambda] \mid_{Dna_e} [accept\ dnae] \mid [enter\ h\_c3.0 + X]]$$

$$\xrightarrow{(l\_c3?.Y_1+Y_2)|Y_3}$$

$$_{Ecoli}[_{CIII}[0] \mid [C2] \mid [C1] \mid [CRO])|[DNA\lambda] \mid \ _{Dna_e}[accept\ dnae] \mid \ _{hfl}[Y_1 \mid Y_3]]$$

$$\xrightarrow{(Y_4+accept\ c2.Y_5|Y_6),Y_3}$$

$$_{Ecoli}[(_{CIII}[0] \mid [C1] \mid [CRO])|[DNA\lambda] \mid_{Dna_e} [accept\ dnae] \mid_{hfl} [_{CII}[0]|(Y_5 \mid Y_6) \mid Y_3]]$$

$$\xrightarrow{(Y_7+lysi!.Y_8),Y_6,Y_3}$$

$$_{Ecoli}[(_{CIII}[0] \mid [C1] \mid [CRO])|[_{\lambda}[exit\ newph.VIRUS] \mid expel\ newph] \mid$$
$$_{Dna_e}[accept\ dnae] \mid_{hfl} [_{CII}[0]|(Y_8 \mid Y_6) \mid Y_3]]$$

$$\xrightarrow{Y_8,Y_6,Y_3}$$

$$_{Ecoli}[(_{CIII}[0] \mid [C1] \mid [CRO])|_{\lambda}[VIRUS] \mid_{Dna_e} [accept\ dnae] \mid_{hfl} [_{CII}[0]|(Y_8 \mid Y_6) \mid Y_3]]$$

The deduced symbolic trace

$$(l\_c3?.(Y_4 + accept\ c2(Y_7 + lysi!.Y_8) \mid Y_6) + Y_2) \mid Y_3$$

The formula encodes the minimum requirements that a system must satisfy in order to simulate the lytic path.

A valid theory

By correctness of the STS, the trace computed can be obtained in a fully specified system, satisfying the above biological hyphoteses.

The deduced symbolic trace

$$(l\_c3?.(Y_4 + accept\ c2(Y_7 + lysi!.Y_8) \mid Y_6) + Y_2) \mid Y_3$$

The formula encodes the minimum requirements that a system must satisfy in order to simulate the lytic path.

A valid theory

By correctness of the STS, the trace computed can be obtained in a fully specified system, satisfying the above biological hyphoteses.

Possible applications

How to use our approach:

Before in silico experiments

Preliminary study of the systems.

Partial data

When is not possible to design a complete model.

Extension of existing models

To build new complexes by studying the requirements the new components should have in order to interact in the desired way

Future work

- **names**: extending the approach to make the STS also works with process algebras with name restriction and name passing;
- **automatic tool** build automatic simulator working with any process algebra of interest (based on unification);
- **quantitative analysis** adding values for calculating quantitative values (probability/rate/concentrations) involved in transitions;

The end

Thanks for your attention !

Future work II

- building a *significative* trace requires to follow a criterium to discriminate between infinite moves.
- ... quantitative values may help —