

# The Calculus of Looping Sequences

Paolo Milazzo

Dipartimento di Informatica, Università di Pisa, Italy

Bertinoro – September, 2008

## Our aim...

At the beginning of our work our aim was to try to apply formal methods to models of biological systems

We were looking for a formalism

- based on term rewriting
- with a simple semantics
- very general

As a consequence, we defined the Calculus of Looping Sequences (CLS)...

# Outline of the talk

- 1 Introduction
- 2 The Calculus of Looping Sequences (CLS)
  - Definition of CLS
  - The *lac* operon in CLS
- 3 Bisimulations in CLS
  - A labeled semantics for CLS
  - Bisimulations in CLS
  - Bisimulations applied to the CLS model of the *lac* operon
- 4 Stochastic CLS
- 5 Ongoing work
  - Spatial CLS
  - Translation of Stochastic CLS into Maude
- 6 References

# The Calculus of Looping Sequences (CLS)

We assume an alphabet  $\mathcal{E}$ . **Terms**  $T$  and **Sequences**  $S$  of CLS are given by the following grammar:

$$\begin{aligned} T &::= S \mid (S)^L \mid T \mid T \\ S &::= \epsilon \mid a \mid S \cdot S \end{aligned}$$

where  $a$  is a generic element of  $\mathcal{E}$ , and  $\epsilon$  is the empty sequence.

The operators are:

$S \cdot S$  : Sequencing

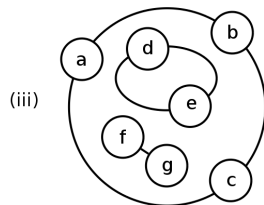
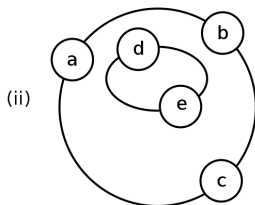
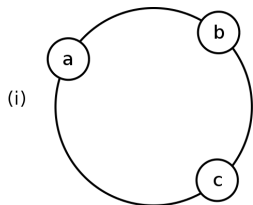
$(S)^L$  : Looping ( $S$  is closed and it can rotate)

$T_1 \mid T_2$  : Containment ( $T_1$  contains  $T_2$ )

$T \mid T$  : Parallel composition (juxtaposition)

Actually, looping and containment form a single binary operator  $(S)^L \mid T$ .

# Examples of Terms



$$(i) \quad (a \cdot b \cdot c)^L \mid \epsilon$$

$$(ii) \quad (a \cdot b \cdot c)^L \mid (d \cdot e)^L \mid \epsilon$$

$$(iii) \quad (a \cdot b \cdot c)^L \mid (f \cdot g \mid (d \cdot e)^L \mid \epsilon)$$

# Structural Congruence

The **Structural Congruence** relations  $\equiv_S$  and  $\equiv_T$  are the least congruence relations on sequences and on terms, respectively, satisfying the following rules:

$$S_1 \cdot (S_2 \cdot S_3) \equiv_S (S_1 \cdot S_2) \cdot S_3 \quad S \cdot \epsilon \equiv_S \epsilon \cdot S \equiv_S S$$

$$T_1 \mid T_2 \equiv_T T_2 \mid T_1 \quad T_1 \mid (T_2 \mid T_3) \equiv_T (T_1 \mid T_2) \mid T_3$$

$$T \mid \epsilon \equiv_T T \quad (S_1 \cdot S_2)^L \rfloor T \equiv_T (S_2 \cdot S_1)^L \rfloor T$$

We write  $\equiv$  for  $\equiv_T$ .

# CLS Patterns

Let us consider variables of three kinds:

- term variables ( $X, Y, Z, \dots$ )
- sequence variables ( $\tilde{x}, \tilde{y}, \tilde{z}, \dots$ )
- element variables ( $x, y, z, \dots$ )

**Patterns**  $P$  and **Sequence Patterns**  $SP$  of CLS extend CLS terms and sequences with variables:

$$\begin{aligned} P & ::= SP \mid (SP)^L \mid P \mid P \mid X \\ SP & ::= \epsilon \mid a \mid SP \cdot SP \mid x \mid \tilde{x} \end{aligned}$$

where  $a$  is a generic element of  $\mathcal{E}$ ,  $\epsilon$  is the empty sequence, and  $x, \tilde{x}$  and  $X$  are generic element, sequence and term variables

The structural congruence relation  $\equiv$  extends trivially to patterns

# Rewrite Rules

A **Rewrite Rule** is a pair  $(P, P')$ , denoted  $P \mapsto P'$ , where:

- $P, P'$  are patterns
- variables in  $P'$  are a subset of those in  $P$

A rule  $P \mapsto P'$  can be applied to all terms that are instantiations of  $P$ .

**Example:**  $a \cdot x \cdot a \mapsto b \cdot x \cdot b$

- can be applied to  $a \cdot c \cdot a$  (producing  $b \cdot c \cdot b$ )
- cannot be applied to  $a \cdot c \cdot c \cdot a$

**Example:**  $(a \cdot \tilde{x})^L \rfloor (b \mid X) \mapsto (c \cdot \tilde{x})^L \rfloor X$

- can be applied to  $(a \cdot a \cdot a)^L \rfloor (b \mid b \mid (a)^L \rfloor b)$
- the result is either  $(c \cdot a \cdot a)^L \rfloor (b \mid (a)^L \rfloor b)$  or  $(a \cdot a \cdot a)^L \rfloor (b \mid b \mid (c)^L \rfloor \epsilon)$



# Formal Semantics

$P\sigma$  denotes the term obtained by replacing any variable in  $T$  with the corresponding term, sequence or element.

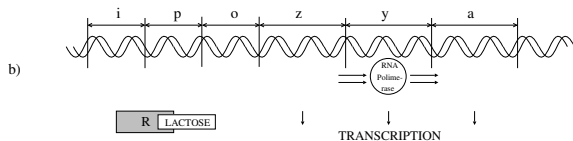
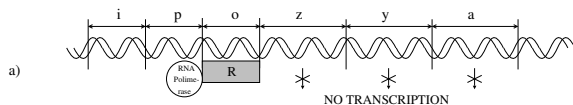
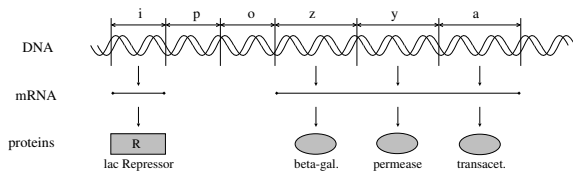
$\Sigma$  is the set of all possible instantiations  $\sigma$

Given a set of rewrite rules  $\mathcal{R}$ , evolution of terms is described by the transition system given by the least relation  $\rightarrow$  satisfying

$$\frac{P \mapsto P' \in \mathcal{R} \quad P\sigma \neq \epsilon \quad \sigma \in \Sigma}{P\sigma \rightarrow P'\sigma}$$
$$\frac{T \rightarrow T'}{T \mid T'' \rightarrow T' \mid T''} \quad \frac{T \rightarrow T'}{(S)^L \mid T \rightarrow (S)^L \mid T'}$$

and closed under structural congruence  $\equiv$ .

# CLS modeling examples: the *lac* operon (1)



## CLS modeling examples: the *lac* operon (2)

$$Ecoli ::= (m)^L \mid (lacI \cdot lacP \cdot lacO \cdot lacZ \cdot lacY \cdot lacA \mid polym)$$

Rules for DNA transcription/translation:

$$lacI \cdot \tilde{x} \mapsto lacI' \cdot \tilde{x} \mid repr \quad (R1)$$

$$polym \mid \tilde{x} \cdot lacP \cdot \tilde{y} \mapsto \tilde{x} \cdot PP \cdot \tilde{y} \quad (R2)$$

$$\tilde{x} \cdot PP \cdot lacO \cdot \tilde{y} \mapsto \tilde{x} \cdot lacP \cdot PO \cdot \tilde{y} \quad (R3)$$

$$\tilde{x} \cdot PO \cdot lacZ \cdot \tilde{y} \mapsto \tilde{x} \cdot lacO \cdot PZ \cdot \tilde{y} \quad (R4)$$

$$\tilde{x} \cdot PZ \cdot lacY \cdot \tilde{y} \mapsto \tilde{x} \cdot lacZ \cdot PY \cdot \tilde{y} \mid betagal \quad (R5)$$

$$\tilde{x} \cdot PY \cdot lacA \mapsto \tilde{x} \cdot lacY \cdot PA \mid perm \quad (R6)$$

$$\tilde{x} \cdot PA \mapsto \tilde{x} \cdot lacA \mid transac \mid polym \quad (R7)$$

## CLS modeling examples: the *lac* operon (3)

$$Ecoli ::= (m)^L \rfloor (lacI \cdot lacP \cdot lacO \cdot lacZ \cdot lacY \cdot lacA \mid polym)$$

Rules to describe the binding of the lac Repressor to gene o, and what happens when lactose is present in the environment of the bacterium:

$$repr \mid \tilde{x} \cdot lacO \cdot \tilde{y} \mapsto \tilde{x} \cdot RO \cdot \tilde{y} \quad (R8)$$

$$LACT \mid (m \cdot \tilde{x})^L \rfloor X \mapsto (m \cdot \tilde{x})^L \rfloor (X \mid LACT) \quad (R9)$$

$$\tilde{x} \cdot RO \cdot \tilde{y} \mid LACT \mapsto \tilde{x} \cdot lacO \cdot \tilde{y} \mid RLACT \quad (R10)$$

$$(\tilde{x})^L \rfloor (perm \mid X) \mapsto (perm \cdot \tilde{x})^L \rfloor X \quad (R11)$$

$$LACT \mid (perm \cdot \tilde{x})^L \rfloor X \mapsto (perm \cdot \tilde{x})^L \rfloor (LACT \mid X) \quad (R12)$$

$$betagal \mid LACT \mapsto betagal \mid GLU \mid GAL \quad (R13)$$

## CLS modeling examples: the *lac* operon (4)

$$Ecoli ::= (m)^L \rfloor (lacI \cdot lacP \cdot lacO \cdot lacZ \cdot lacY \cdot lacA \mid polym)$$

Example:

$$Ecoli \mid LACT \mid LACT$$

$$\rightarrow^* (m)^L \rfloor (lacI' \cdot lacP \cdot lacO \cdot lacZ \cdot lacY \cdot lacA \mid polym \mid repr) \mid LACT \mid LACT$$

$$\rightarrow^* (m)^L \rfloor (lacI' \cdot lacP \cdot RO \cdot lacZ \cdot lacY \cdot lacA \mid polym) \mid LACT \mid LACT$$

$$\rightarrow^* (m)^L \rfloor (lacI' \cdot lacP \cdot lacO \cdot lacZ \cdot lacY \cdot lacA \mid polym \mid RLACT) \mid LACT$$

$$\rightarrow^* (perm \cdot m)^L \rfloor (lacI' - A \mid betagal \mid transac \mid polym \mid RLACT) \mid LACT$$

$$\rightarrow^* (perm \cdot m)^L \rfloor (lacI' - A \mid betagal \mid transac \mid polym \mid RLACT \mid GLU \mid GAL)$$

# Outline of the talk

- 1 Introduction
- 2 The Calculus of Looping Sequences (CLS)
  - Definition of CLS
  - The *lac* operon in CLS
- 3 Bisimulations in CLS
  - A labeled semantics for CLS
  - Bisimulations in CLS
  - Bisimulations applied to the CLS model of the *lac* operon
- 4 Stochastic CLS
- 5 Ongoing work
  - Spatial CLS
  - Translation of Stochastic CLS into Maude
- 6 References

# Bisimulations

Bisimilarity is widely accepted as the finest extensional behavioral equivalence one may impose on systems.

- Two systems are bisimilar if they can perform step by step the same interactions with the environment.
- Properties of a system can be verified by assessing the bisimilarity with a system known to enjoy them.

Bisimilarities need semantics based on labeled transition relations capturing the potential interactions with the environment.

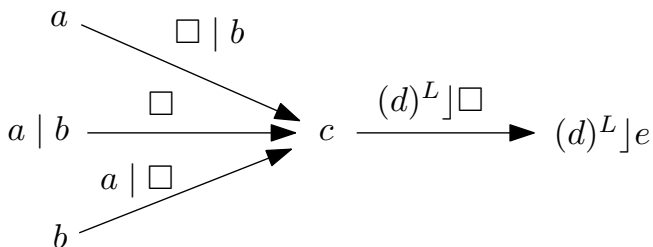
- In process calculi, transitions are usually labeled with actions.
- In CLS labels are contexts in which rules can be applied.

## Labeled semantics

**The idea:** There is a (labeled) transition between terms  $T$  and  $T'$  if there exists a context  $C$  such that a rewrite rule can be applied to  $C[T]$  with  $T'$  as result.

- $C$  is used as transition label
- $C$  must not provide the whole left hand side of the applied rewrite rule

**An example:** Let  $\mathcal{R} = \{ a \mid b \mapsto c, (d)^L \rfloor c \mapsto (d)^L \rfloor e \}$





## Labeled semantics

**Contexts**  $\mathcal{C}$  are given by the following grammar:

$$\mathcal{C} ::= \square \mid \mathcal{C} \mid T \mid T \mid \mathcal{C} \mid (S)^L \rfloor \mathcal{C}$$

where  $T \in \mathcal{T}$  and  $S \in \mathcal{S}$ . Context  $\square$  is called the *empty context*.

Given a set of rewrite rules  $\mathcal{R} \subseteq \mathfrak{R}$ , the **labeled semantics** of CLS is the labeled transition system given by the following inference rules:

$$\begin{array}{c} \text{(rule\_appl)} \quad \frac{P \mapsto P' \in \mathcal{R} \quad \mathcal{C}[T''] \equiv P\sigma \quad T'' \not\equiv \epsilon \quad \sigma \in \Sigma \quad \mathcal{C} \in \mathcal{C}}{T'' \xrightarrow{\mathcal{C}} P'\sigma} \\ \\ \text{(cont)} \quad \frac{T \xrightarrow{\square} T'}{(S)^L \rfloor T \xrightarrow{\square} (S)^L \rfloor T'} \quad \text{(par)} \quad \frac{T \xrightarrow{\mathcal{C}} T' \quad \mathcal{C} \in \mathcal{C}_P}{T \mid T'' \xrightarrow{\mathcal{C}} T' \mid T''} \end{array}$$

where  $\mathcal{C}_P$  are contexts that do not include  $(S)^L \rfloor \mathcal{C}$  and the dual version of the *(par)* rule is omitted.

## Bisimulations in CLS (1)

A binary relation  $R$  on terms is a **strong bisimulation** if, given  $T_1, T_2$  such that  $T_1RT_2$ , the two following conditions hold:

- $T_1 \xrightarrow{C} T'_1 \implies \exists T'_2$  s.t.  $T_2 \xrightarrow{C} T'_2$  and  $T'_1RT'_2$
- $T_2 \xrightarrow{C} T'_2 \implies \exists T'_1$  s.t.  $T_1 \xrightarrow{C} T'_1$  and  $T'_2RT'_1$ .

The *strong bisimilarity*  $\sim$  is the largest of such relations.

A binary relation  $R$  on terms is a **weak bisimulation** if, given  $T_1, T_2$  such that  $T_1RT_2$ , the two following conditions hold:

- $T_1 \xrightarrow{C} T'_1 \implies \exists T'_2$  s.t.  $T_2 \xRightarrow{C} T'_2$  and  $T'_1RT'_2$
- $T_2 \xrightarrow{C} T'_2 \implies \exists T'_1$  s.t.  $T_1 \xRightarrow{C} T'_1$  and  $T'_2RT'_1$ .

The *weak bisimilarity*  $\approx$  is the largest of such relations.

**Theorem:** Strong and weak bisimilarities are congruences.

## Bisimulations in CLS (2)

Consider the following set of rewrite rules:

$$\mathcal{R} = \{ a \mid b \mapsto c, \quad d \mid b \mapsto e, \quad e \mapsto e, \quad c \mapsto e, \quad f \mapsto a \}$$

We have that  $a \sim d$ , because

$$a \xrightarrow{\square \mid b} c \xrightarrow{\square} e \xrightarrow{\square} e \xrightarrow{\square} \dots$$

$$d \xrightarrow{\square \mid b} e \xrightarrow{\square} e \xrightarrow{\square} \dots$$

and  $f \approx d$ , because

$$f \xrightarrow{\square} a \xrightarrow{\square \mid b} c \xrightarrow{\square} e \xrightarrow{\square} e \xrightarrow{\square} \dots$$

On the other hand,  $f \not\sim e$  and  $f \not\approx e$ .

$$e \xrightarrow{\square} e \xrightarrow{\square} e \xrightarrow{\square} \dots$$

## Bisimulations in CLS (3)

Let us consider systems  $(T, \mathcal{R}) \dots$

A binary relation  $R$  is a **strong bisimulation on systems** if, given  $(T_1, \mathcal{R}_1)$  and  $(T_2, \mathcal{R}_2)$  such that  $(T_1, \mathcal{R}_1)R(T_2, \mathcal{R}_2)$ :

- $\mathcal{R}_1 : T_1 \xrightarrow{C} T'_1 \implies \exists T'_2$  s.t.  $\mathcal{R}_2 : T_2 \xrightarrow{C} T'_2$  and  $(T'_1, \mathcal{R}_1)R(T'_2, \mathcal{R}_2)$
- $\mathcal{R}_2 : T_2 \xrightarrow{C} T'_2 \implies \exists T'_1$  s.t.  $\mathcal{R}_1 : T_1 \xrightarrow{C} T'_1$  and  $(T_2, \mathcal{R}_2)R(T'_1, \mathcal{R}_1)$ .

The *strong bisimilarity on systems*  $\sim$  is the largest of such relations.

A binary relation  $R$  is a **weak bisimulation on systems** if, given  $(T_1, \mathcal{R}_1)$  and  $(T_2, \mathcal{R}_2)$  such that  $(T_1, \mathcal{R}_1)R(T_2, \mathcal{R}_2)$ :

- $\mathcal{R}_1 : T_1 \xrightarrow{C} T'_1 \implies \exists T'_2$  s.t.  $\mathcal{R}_2 : T_2 \xRightarrow{C} T'_2$  and  $(T'_1, \mathcal{R}_1)R(T'_2, \mathcal{R}_2)$
- $\mathcal{R}_2 : T_2 \xrightarrow{C} T'_2 \implies \exists T'_1$  s.t.  $\mathcal{R}_1 : T_1 \xRightarrow{C} T'_1$  and  $(T'_2, \mathcal{R}_2)R(T'_1, \mathcal{R}_1)$

The *weak bisimilarity on systems*  $\approx$  is the largest of such relations.

Strong and weak bisimilarities on systems are NOT congruences.

## Bisimulations in CLS (4)

Consider the following sets of rewrite rules

$$\mathcal{R}_1 = \{a \mid b \mapsto c\} \quad \mathcal{R}_2 = \{a \mid d \mapsto c, b \mid e \mapsto c\}$$

We have that  $\langle a, \mathcal{R}_1 \rangle \approx \langle e, \mathcal{R}_2 \rangle$  because

$$\mathcal{R}_1 : a \xrightarrow{\square|b} c \quad \mathcal{R}_2 : e \xrightarrow{\square|b} c$$

and  $\langle b, \mathcal{R}_1 \rangle \approx \langle d, \mathcal{R}_2 \rangle$ , because

$$\mathcal{R}_1 : b \xrightarrow{\square|a} c \quad \mathcal{R}_2 : d \xrightarrow{\square|a} c$$

but  $\langle a \mid b, \mathcal{R}_1 \rangle \not\approx \langle e \mid d, \mathcal{R}_2 \rangle$ , because

$$\mathcal{R}_1 : a \mid b \xrightarrow{\square} c \quad \mathcal{R}_2 : e \mid d \not\xrightarrow{\square}$$

## Applying bisimulations to the *lac* operon (1)

By using the weak bisimilarity on systems we can prove that from the state in which the repressor is bound to the DNA we can reach a state in which the enzymes are synthesized only if lactose appears in the environment.

We replace rule

$$\tilde{x} \cdot RO \cdot \tilde{y} \mid LACT \mapsto \tilde{x} \cdot lacO \cdot \tilde{y} \mid RLACT \quad (R10)$$

with

$$\begin{aligned} (\tilde{w})^L \mid (\tilde{x} \cdot RO \cdot \tilde{y} \mid LACT \mid X) \mid START &\mapsto \\ (\tilde{w})^L \mid (\tilde{x} \cdot lacO \cdot \tilde{y} \mid RLACT \mid X) &\quad (R10bis) \end{aligned}$$

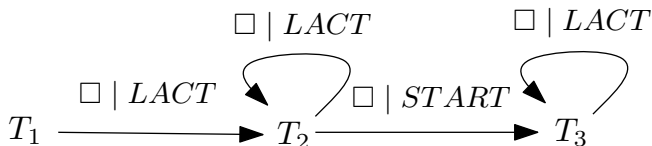
## Applying bisimulations to the *lac* operon (2)

The obtained model is weakly bisimilar to  $(T_1, \mathcal{R})$  where  $\mathcal{R}$  is

$$T_1 \mid LACT \mapsto T_2 \quad (\text{R1}') \qquad T_2 \mid START \mapsto T_3 \quad (\text{R3}')$$

$$T_2 \mid LACT \mapsto T_2 \quad (\text{R2}') \qquad T_3 \mid LACT \mapsto T_3 \quad (\text{R4}')$$

that is a system satisfying the wanted property.



# Outline of the talk

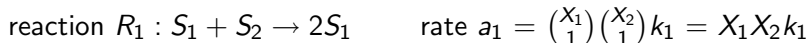
- 1 Introduction
- 2 The Calculus of Looping Sequences (CLS)
  - Definition of CLS
  - The *lac* operon in CLS
- 3 Bisimulations in CLS
  - A labeled semantics for CLS
  - Bisimulations in CLS
  - Bisimulations applied to the CLS model of the *lac* operon
- 4 Stochastic CLS
- 5 Ongoing work
  - Spatial CLS
  - Translation of Stochastic CLS into Maude
- 6 References



## Background: Gillespie's simulation algorithm

- represents a chemical solution as a multiset of molecules
- each chemical reaction is associated with a kinetic constant
- computes the reaction rate  $a_\mu$  by multiplying the kinetic constant by the number of possible combinations of reactants

Example: chemical solution with  $X_1$  molecules  $S_1$  and  $X_2$  molecules  $S_2$



Given a set of reactions  $\{R_1, \dots, R_M\}$  and a current time  $t$

- The time  $t + \tau$  at which the next reaction will occur is randomly chosen with  $\tau$  exponentially distributed with parameter  $\sum_{\nu=1}^M a_\nu$ ;
- The reaction  $R_\mu$  that has to occur at time  $t + \tau$  is randomly chosen with probability  $\frac{a_\mu}{\sum_{\nu=1}^M a_\nu}$ .

At each step  $t$  is incremented by  $\tau$  and the chemical solution is updated.

# Stochastic CLS (1)

Stochastic CLS incorporates Gillespie's stochastic framework into the semantics of CLS

- Rewrite rules are enriched with kinetic constants

What is a reactant in Stochastic CLS?

- A *reactant combination* is an occurrence (up to  $\equiv$ ) of a left hand side of a rewrite rule

**Example:** The application rate of  $a \mid b \xrightarrow{k} c$  to  $a \mid a \mid b \mid b$  is  $6k$

**Example:** The application rate of  $(a \cdot \tilde{x})^L \mid (b \mid X) \xrightarrow{k} (c \cdot \tilde{x})^L \mid X$  to  $(a \cdot a \cdot a)^L \mid (b \mid b) \mid (a \cdot a)^L \mid b$  is

- $6k$ , with  $(c \cdot a \cdot a)^L \mid b \mid (a \cdot a)^L \mid b$  as result
- $+ 2k$ , with  $(a \cdot a \cdot a)^L \mid (b \mid b) \mid (c \cdot a)^L \mid \epsilon$  as result
- $= 8k$

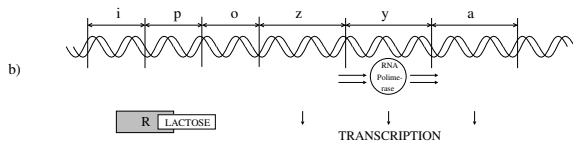
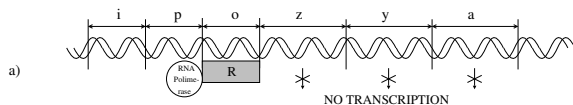
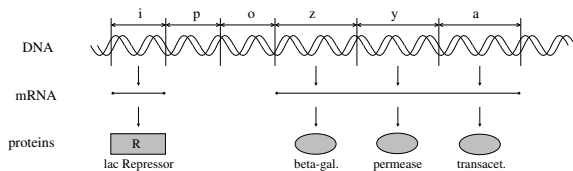
## Stochastic CLS (2)

Given a finite set of stochastic rewrite rules  $\mathcal{R}$ , the semantics of Stochastic CLS is the least transition relation  $\xrightarrow{R, T, r, b}$  closed wrt  $\equiv$  and satisfying by the following inference rules:

$$\frac{R : P_L \xrightarrow{k} P_R \in \mathcal{R} \quad \sigma \in \Sigma}{P_L \sigma \xrightarrow{R, P_L \sigma, k \cdot \text{comb}(P_L, \sigma), 1} P_R \sigma} \quad \frac{T_1 \xrightarrow{R, T, r, b} T_2}{T_1 \mid T_3 \xrightarrow{R, T, r, b \cdot \text{binom}(T, T_1, T_3)} T_2 \mid T_3}$$
$$\frac{T_1 \xrightarrow{R, T, r, b} T_2}{(T_1)^L \mid T_3 \xrightarrow{R, (T_1)^L \mid T_3, r \cdot b, 1} (T_2)^L \mid T_3} \quad \frac{T_1 \xrightarrow{R, T, r, b} T_2}{(T_3)^L \mid T_1 \xrightarrow{R, (T_3)^L \mid T_1, r \cdot b, 1} (T_3)^L \mid T_2}$$

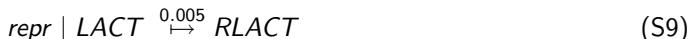
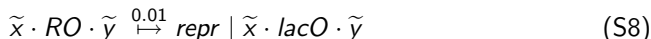
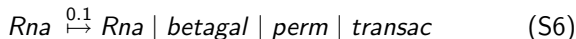
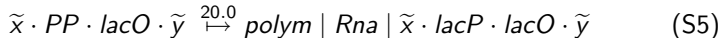
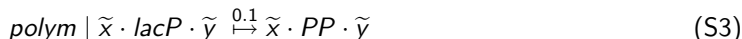
The transition system obtained can be easily transformed into a *Continuous Time Markov Chain*

# A Stochastic CLS model of the *lac* operon (1)



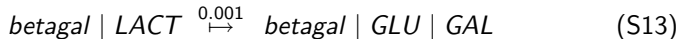
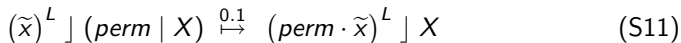
## A Stochastic CLS model of the *lac* operon (2)

Transcription of DNA, binding of lac Repressor to gene *o*, and interaction between lactose and lac Repressor:

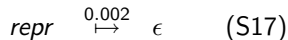
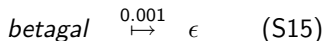


## A Stochastic CLS model of the *lac* operon (3)

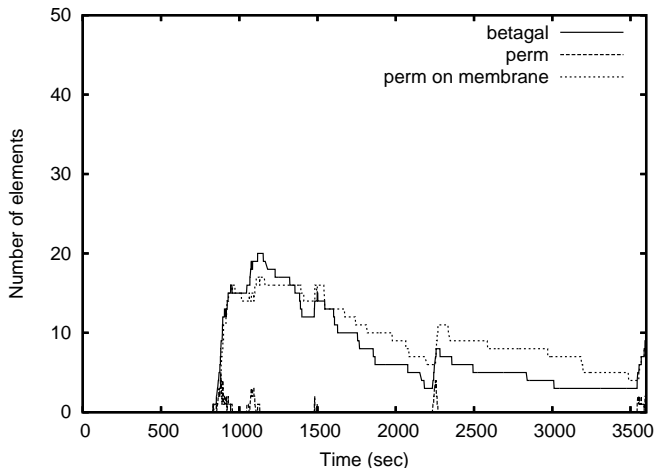
The behaviour of the three enzymes for lactose degradation:



Degradation of all the proteins and mRNA involved in the process:



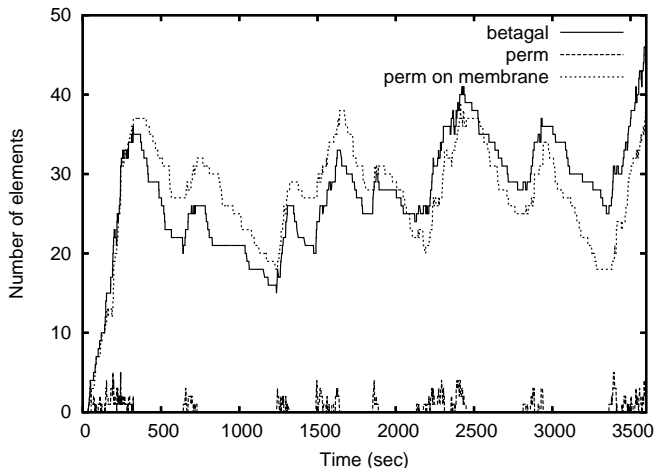
# Simulation results (1)



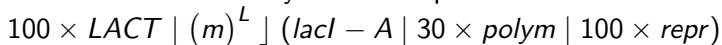
Production of enzymes in the absence of lactose

$$(m)^L \rfloor (lacl - A \mid 30 \times polym \mid 100 \times repr)$$

## Simulation results (2)

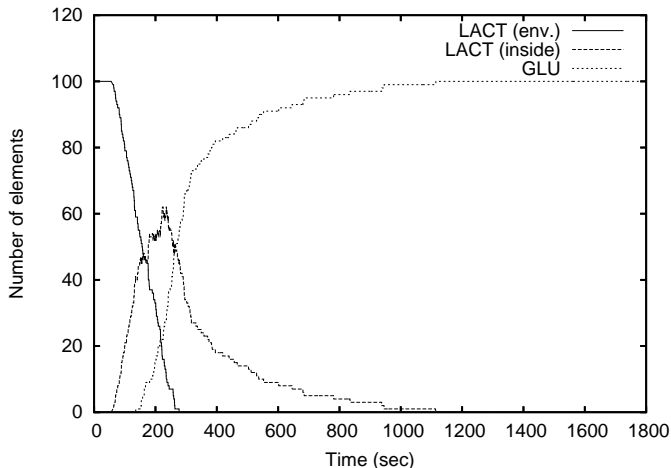


Production of enzymes in the presence of lactose





## Simulation results (3)



Degradation of lactose into glucose

$$100 \times LACT \mid (m)^L \mid (lact - A \mid 30 \times polym \mid 100 \times repr)$$

# Outline of the talk

- 1 Introduction
- 2 The Calculus of Looping Sequences (CLS)
  - Definition of CLS
  - The *lac* operon in CLS
- 3 Bisimulations in CLS
  - A labeled semantics for CLS
  - Bisimulations in CLS
  - Bisimulations applied to the CLS model of the *lac* operon
- 4 Stochastic CLS
- 5 Ongoing work
  - Spatial CLS
  - Translation of Stochastic CLS into Maude
- 6 References

# Spatial CLS

The spatial organization of elements may affect system dynamics

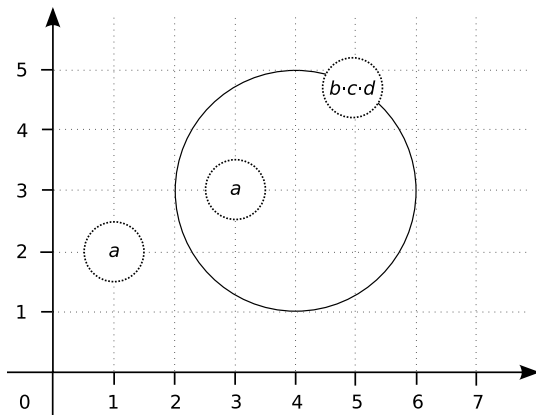
- reaction-diffusion system
- molecular crowding

We developed Spatial CLS by extending the Calculus of Looping Sequences

Elements of Spatial CLS are spheres in a continuous space

- the containment hierarchy is reflected in the spheres
- elements can move autonomously
- interactions can depend on the spatial information of elements (position, radius, ecc.)
- rewrite rules are endowed with rates

## Example of Spatial CLS term



$$T = (a)_{[(1,2),m_1],0.5} \mid ((b \cdot c \cdot d)_{.,0.5})_{[(4,3),m_2],2}^L \mid (a)_{[(-1,0),m_3],0.5}$$

# Rewrite rules

$$R : [f_c] \quad P_L \xrightarrow{k} P_R$$

$k$  : reaction rate

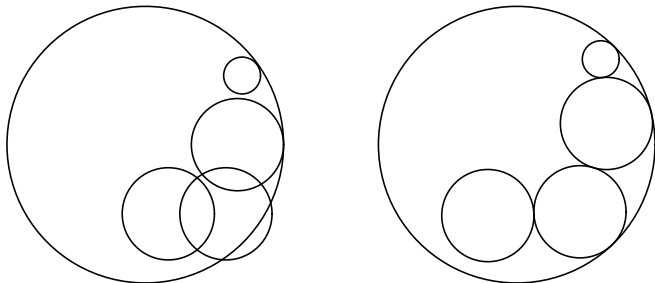
$f_c$  : application constraints

- takes into account the spatial information of involved elements (eg. position, radius, ecc.)

## Example

$$[dist(p, q) \leq 5] \quad (a)_{[p, f_1], r_1} \mid (b)_{[q, f_2], r_2} \xrightarrow{0.8} (c)_{[\frac{p+q}{2}, m], r_3}$$

## Resolving space conflicts



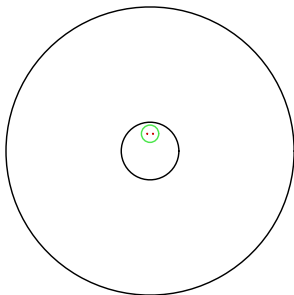
Elements push each other

- the pushing effect is modeled with a system of differential equations
- the rearranged state corresponds to its equilibrium state

# Modeling cell proliferation

Initial state of the system:

$$T = (b)_{.,50}^L \mid (m)_{[(0,0),m_1],10}^L \mid (n)^L \mid (cr \cdot g_1 \cdot g_2 \cdot g_3 \mid cr \cdot g_4 \cdot g_5)$$



- $(b)_{.,50}^L$  : the available space
- $(m)_{[(0,0),m_1],10}^L$  : the membrane of the cell
- $(n)^L$  : the nucleus
- $cr \cdot \dots$  : the chromosomes

# Rewrite rules modeling the behavior

$$R_1 : [r = 7] \quad (m)_{[p,f],r}^L \Downarrow X \xrightarrow{0.33} (m)_{[p,f],10}^L \Downarrow X$$

$$R_2 : [r = 10] \quad (m)_{[p,f],r}^L \Downarrow X \xrightarrow{0.25} (m)_{[p,f],14}^L \Downarrow X$$

$$R_3 : [r = 14] \quad (m)_{[p,f],r}^L \Downarrow \left( (n)^L \Downarrow X \right) \xrightarrow{0.5} (m)_{[p,f],r}^L \Downarrow \left( (n_{\text{dup}})^L \Downarrow X \right)$$

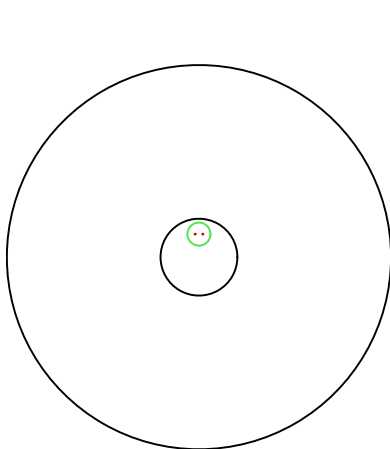
$$R_4 : (n_{\text{dup}})^L \Downarrow (cr \cdot \tilde{x} \mid X) \xrightarrow{0.125} (n_{\text{dup}})^L \Downarrow (2cr \cdot \tilde{x} \mid X)$$

$$R_5 : (n_{\text{dup}})^L \Downarrow (2cr \cdot \tilde{x} \mid 2cr \cdot \tilde{y}) \xrightarrow{0.17} \\ (n)^L \Downarrow (cr \cdot \tilde{x} \mid cr \cdot \tilde{y}) \mid (n)^L \Downarrow (cr \cdot \tilde{x} \mid cr \cdot \tilde{y})$$

$$R_6 : (m)_{[(x,y),f],r}^L \Downarrow \left( (n)^L \Downarrow X \mid (n)^L \Downarrow Y \right) \xrightarrow{1} \\ (m)_{[(x-5,y),f],7}^L \Downarrow (n)^L \Downarrow X \mid (m)_{[(x+5,y),f],7}^L \Downarrow (n)^L \Downarrow Y$$

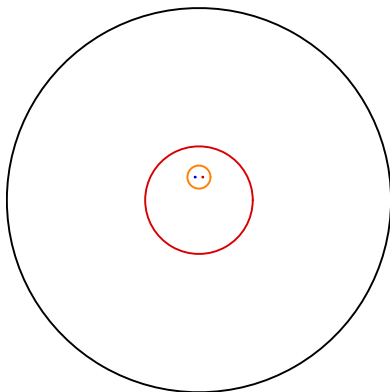


# Simulation



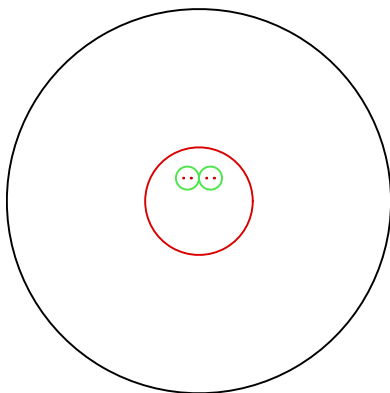
$t = 0h$

# Simulation



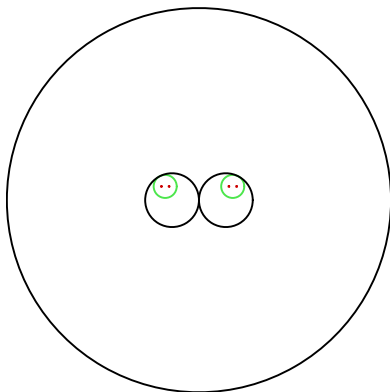
$t = 6h$

# Simulation



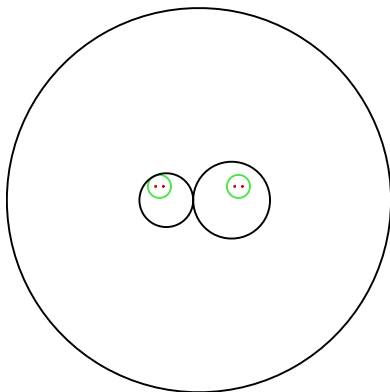
$t = 15h$

# Simulation



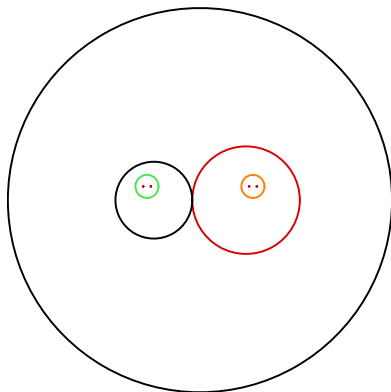
$t = 16h$

# Simulation



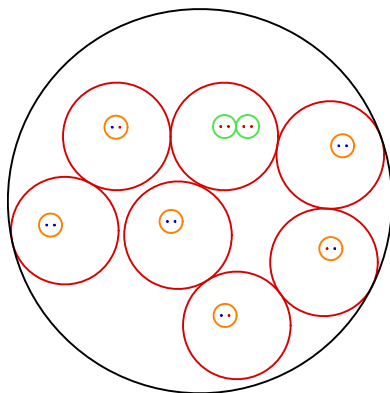
$t = 20h$

# Simulation



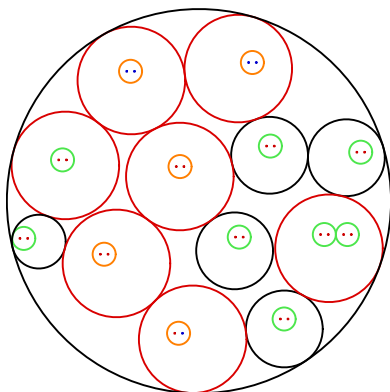
$t = 25.66h$

# Simulation



$t = 88h$

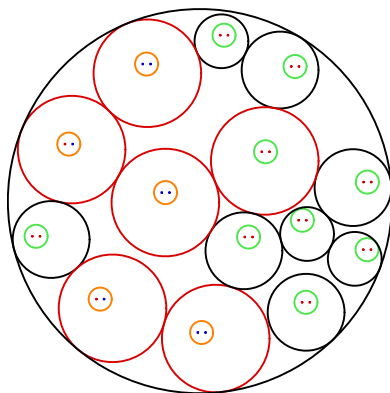
# Simulation



$t = 98h$

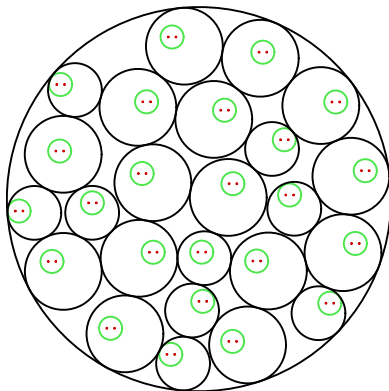


# Simulation



$t = 102h$

# Simulation



$t = 141h$

# Outline of the talk

- 1 Introduction
- 2 The Calculus of Looping Sequences (CLS)
  - Definition of CLS
  - The *lac* operon in CLS
- 3 Bisimulations in CLS
  - A labeled semantics for CLS
  - Bisimulations in CLS
  - Bisimulations applied to the CLS model of the *lac* operon
- 4 Stochastic CLS
- 5 Ongoing work
  - Spatial CLS
  - Translation of Stochastic CLS into Maude
- 6 References

# A model checker for Stochastic CLS

As candidate model checkers we have considered:

- PRISM
- Murphi
- PMaude

All of them are probabilistic/stochastic model checkers

PMaude is the most suitable

- It uses a language based on rewrite rules (rewrite logic) that eases the translation of Stochastic CLS rules

Unfortunately, the model checking module of PMaude seems not to be available

- a possible alternative: Real-Time Maude

# Real-Time Maude

Maude is a specification language equipped with efficient analysis tools, which supports three modelling paradigms:

- algebraic style (via equations)
- rewrite logic (via rewrite rules)
- object oriented (via classes and messages)

Real-Time Maude extends Maude with a notion of time

- rewrite rule applications might consume (a fixed amount of) time

Real-Time Maude has two kinds of rules

- instantaneous rules:

`cr1 [l] : t => t' if cond`

- tick rules:

`cr1 [l] : t => t' in time  $\tau$  if cond`

# Translation of Stochastic CLS into Real-Time Maude

Real-Time Maude is not stochastic

- we will include Gillespie's simulation algorithm (slightly changed) in the translation of Stochastic CLS models
- it will be used to generate single executions of the model
- Real-Time Maude analysis tools will be applied to the simulation results

This is *statistical model checking*

- we loose exhaustivity (properties are checked on a number of runs)
- huge systems could be handled

# Translation of Stochastic CLS into Real-Time Maude

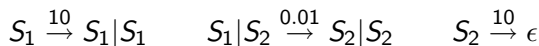
$$\begin{aligned} T &::= S \mid (S)^L \mid T \mid T \mid T \\ S &::= \epsilon \mid a \mid S \cdot S \end{aligned}$$

```
(omod CLS is
  pr NAT
  sorts Elem Seq Term Loop
  subsorts Elem < Seq < Term

  op empty : -> Seq [ctor]
  op ... : Seq Seq -> Seq
    [assoc gather (E e) id: empty ctor]
  op '['_']LContains '['_'] : Seq Term -> Term
    [prec 41 gather (& &) ctor]
  op _|_ : Term Term -> Term
    [assoc comm prec 45 gather (E e) id: empty ctor]
endom)
```

# Translation of Stochastic CLS into Real-Time Maude

Lotka reactions as Stochastic CLS rules



```
r1 [ S1 ] :  
  < 0 : CLSTerm | term : (T | S1), mu : 1, step : 4 >  
=>  
  < 0 : CLSTerm | term : (T | S1 | S1), step : 5 >  
  
r1 [ S2 ] :  
  < 0 : CLSTerm | term : (T | S1 | S2), mu : 2, step : 4 >  
=>  
  < 0 : CLSTerm | term : (T | S2 | S2), step : 5 >  
  
r1 [ S3 ] :  
  < 0 : CLSTerm | term : (T | S2), mu : 3, step : 4 >  
=>  
  < 0 : CLSTerm | term : T, step : 5 >
```



# Analysis example: statistical model checking

Initialisation of 100 stochastic simulations

```
rl [ initialise1 ] :  
  < step : 0 >  
=>  
  < seed : random(1), step : 1 >  
:  
:  
rl [ initialise100 ] :  
  < step : 0 >  
=>  
  < seed : random(100), step : 1 >
```

## Analysis example: statistical model checking

Verification of properties expressed as LTL formulas. Some state formulas:

`vanished(T)` indicates that term `T` has vanished from the system,

`IsLessThan(T,T')` indicates that the occurrences of term `T` are less than the occurrences of `T'`.

Starting with  $4 \times S_1$  and  $4 \times S_2$  we prove

- that  $S_2$  will eventually disappear (i.e.  $\diamond \text{vanished}(S_2)$ )
- that the amount of  $S_2$  will eventually become less than the amount of  $S_1$  (i.e.  $\diamond \text{IsLessThan}(S_2, S_1)$ )

```
(mc INIT({S1}4 | {S2}4) |=t <> vanished(S2) in time<=1 .)
```

Result Bool : true

```
(mc INIT({S1} 4 | {S2} 4) |=t <> IsLessThan(S2,S1) in time<=1 .)
```

Result Bool : true

# References

- P. Milazzo, *Formal Modeling in Systems Biology. A Theoretical Computer Science Approach*, VDM Verlag Dr. Muller, Saarbruecken, 2008.
- P. Milazzo. *Qualitative and Quantitative Formal Modeling of Biological Systems*, PhD Thesis, Università di Pisa, 2007.
- R. Barbuti, A. Maggiolo-Schettini, P. Milazzo and A. Troina. *A Calculus of Looping Sequences for Modelling Microbiological Systems*. *Fundamenta Informaticae* 72, 21-35, 2006.
- R. Barbuti, A. Maggiolo-Schettini and P. Milazzo. *Extending the Calculus of Looping Sequences to Model Protein Interaction at the Domain Level*. *Symp. on Bioinf. Research and Applic. (ISBRA'07)*, LNBI 4463, 638-649, Springer, 2007.
- R. Barbuti, A. Maggiolo-Schettini, P. Milazzo and A. Troina. *The Calculus of Looping Sequences for Modeling Biological Membranes*. *Invited paper at the 8th Work. on Membrane Computing (WMC8)*, LNCS 4860, 54-76, Springer, 2007.
- R. Barbuti, A. Maggiolo-Schettini, P. Milazzo and A. Troina. *Bisimulations in Calculi Modelling Membranes*. *Formal Aspects of Computing* 20, 351-377, 2008.
- T.A. Basuki, A. Cerone, P. Milazzo. *Translating Stochastic CLS into Maude*. *Work. on Membr. Computing and Biol. Inspired Calculi (MeCBIC'08)*, 2008.
- R. Barbuti, A. Maggiolo-Schettini, P. Milazzo, P. Tiberi and A. Troina. *Stochastic CLS for the Modeling and Simulation of Biological Systems*. *Transaction on Computational Systems Biology*, in press.