Master Parisien de Recherche en Informatique

Static Analysis of Kappa Models

VMCAI 2008

Jérôme Feret
Harvard Medical School
École Normale Supérieure
http://www.di.ens.fr/~feret

Joint work with
Vincent Danos, Walter Fontana, and Jean Krivine

December, 2007
The Team

Walter Fontana
Harvard Medical School

Vincent Danos
Paris VII

Jean Krivine
École polytechnique

Jérôme Feret
École normale supérieure
Overview

1. Introduction
2. Kappa language
3. Local views
4. Local set of complexes
5. Local rule systems
6. Decontextualization
7. Conclusion
Modeling signaling pathway

- A cell measures (i.e. checks thresholds, integrates, compares) the concentration of some proteins in order to make decisions.
- Many proteins (enzymes, receptors, transport molecules) are involved. They interact by binding with each other and activating each other.
- We want to track the evolution of some species:
  - There is a combinatorial blow-up.
Why using modelling tools?

- Use a concise high-level description of what happens;
- Share parts of models;
- Derive quantitative models automatically:
  - run benches of simulations,
  - modify initial conditions,
  - update/modify the model at no cost;
- Use static analysis tools in order to check the consistency of a model:
  - dead rules detection,
  - control detection (which site controls which binding),
  - wrong species detection.
Overview

1. Introduction
2. Kappa language
3. Local views
4. Local set of complexes
5. Local rule systems
6. Decontextualization
7. Conclusion
A complex

E(r!1), R(l!1,r!2), R(r!2,l!3), E(r!3)
A Unbinding/Binding Rule

\[ E(r), R(l,r) \leftrightarrow E(r!1), R(l!1,r) \]
Internal state

\[ R(Y_1 \sim u, l!1), E(r!1) \leftrightarrow R(Y_1 \sim p, l!1), E(r!1) \]
Don’t care, Don’t write

R

Y₁

u

r

≠

R

Y₁

p

r

R

Y₁

u

r

p
A contextual rule

\[ R(Y_1 \sim u, r!_) \rightarrow R(Y_1 \sim p, r) \]
Creation/Suppression

\[ R(r) \rightarrow R(r!1), R(r!1,l,Y1) \]

\[ R(r!1), R(r!1) \rightarrow R(r) \]
Early EGF example

egf rules 1

- **Ligand-receptor binding, receptor dimerisation, rtk x-phosph, & de-phosph**
  - 01: $R(l,r), E(r) \leftrightarrow R(l_1,r), E(r_1)$
  - 02: $R(l_1,r), R(l^2,r) \leftrightarrow R(l_1,r_3), R(l^2,r_3)$
  - 03: $R(r^1,Y68) \rightarrow R(r_1^1,Y68^p)$
    - $R(Y68^p) \rightarrow R(Y68)$
  - 04: $R(r_1^1,Y48) \rightarrow R(r_1,Y48^p)$
    - $R(Y48^p) \rightarrow R(Y48)$

- **Sh x-phosph & de-phosph**
  - 14: $R(r^2,Y48^{pl}), Sh(p^1,Y7) \rightarrow R(r^2,Y48^{pl}), Sh(p^1,Y7^p)$
  - ??: $Sh(p^1,Y7^p) \rightarrow Sh(p^1,Y7)$
  - 16: $Sh(p,Y7^p) \rightarrow Sh(p,Y7)$

- **Y68-G binding**
  - 09: $R(Y68^p), G(a,b) \leftrightarrow R(Y68^{pl})+G(a^1,b)$
  - 11: $R(Y68^p), G(a,b^2) \leftrightarrow R(Y68^{pl})+G(a^1,b^2)$

(protein shorthands: $E:=egf$, $R:=egfr$, $So:=Sos$, $Sh:=Sh$, $G:=grb2$)

(site abbreviations & fusions: $Y68:=Y1068$, $Y48:=Y1148/73$, $Y7:=Y317$, $\pi:=PTB/SH2$)

receptor type: $R(l,r,Y68,Y48)$

refined from
$R(Y68^p) \leftrightarrow R(Y68^{pl})+G(a^1)$

refined from
$Sh(Y7^p) \rightarrow Sh(Y7)$
Early EGF example

egf rules 2

refined from
$\text{So}(d)+\text{G}(b)<->\text{So}(d')+\text{G}(b')$

interface note: highlight the interacting parts

G-So binding
- 10: $R(Y_{68}p), G(a_1,b), \text{So}(d) <-> R(Y_{68}p), G(a_1,b'), \text{So}(d')$
- 12: $G(a,b), \text{So}(d) <-> G(a,b'), \text{So}(d')$
- 22: $\text{Sh}(\pi,Y_{7}p), G(a_2,b), \text{So}(d) <-> \text{Sh}(\pi,Y_{7}p), G(a_2,b'), S(d')$
- 19: $\text{Sh}(\pi,Y_{7}p), G(a_2,b), \text{So}(d) <-> \text{Sh}(\pi,Y_{7}p), G(a_2,b'), S(d')$

Y48-Sh binding
- 13: $R(Y_{48}p), \text{Sh}(\pi,Y_{7}), <-> R(Y_{48}p), \text{Sh}(\pi,Y_{7})$
- 15: $R(Y_{48}p), \text{Sh}(\pi,Y_{7}) <-> R(Y_{48}p), \text{Sh}(\pi,Y_{7})$
- 18: $R(Y_{48}p), \text{Sh}(\pi,Y_{7}), G(a_1,b) <-> R(Y_{48}p), \text{Sh}(\pi,Y_{7}), G(a_1,b')$
- 20: $R(Y_{48}p), \text{Sh}(\pi,Y_{7}), G(a_1,b'), S(d') <-> R(Y_{48}p), \text{Sh}(\pi,Y_{7}), G(a_1,b'), S(d')$

Sh-G binding
- 17: $R(Y_{48}p), \text{Sh}(\pi,Y_{7}), G(a,b) <-> R(Y_{48}p), \text{Sh}(\pi,Y_{7}), G(a,b)$
- 21: $\text{Sh}(\pi,Y_{7}), G(a,b) <-> \text{Sh}(\pi,Y_{7}), G(a,b)$
- 23: $\text{Sh}(\pi,Y_{7}), G(a,b') <-> \text{Sh}(\pi,Y_{7}), G(a,b')$
- 24: $R(Y_{48}p), \text{Sh}(\pi,Y_{7}), G(a,b'), S(d') <-> R(Y_{48}p), \text{Sh}(\pi,Y_{7}), G(a,b'), S(d')$

refined from
$\text{Sh}(\pi), G(a)<->\text{Sh}(\pi), G(a')$

refined from
$R(Y_{48}p)+\text{Sh}(\pi)<->R(Y_{48}p)+\text{Sh}(\pi)$

refined from
$\text{So}(d)+\text{G}(b)<->\text{So}(d)+\text{G}(b)$

Why not simply $G(b')$??
Overview

1. Introduction
2. Kappa language
3. Local views
4. Local set of complexes
5. Local rule systems
6. Decontextualization
7. Conclusion
Set of reachable complexes

Let $\mathcal{R} = \{R_i\}$ be a set of rules.
Let $\text{Complex}$ be the set of all complexes $(C, c_1, c'_1, \ldots, c_k, c'_k, \ldots \in \text{Complex})$.
Let $\text{Complex}_0$ be the set of initial complexes.

We write:

\[ c_1, \ldots, c_m \rightarrow_{R_k} c'_1, \ldots, c'_n \]

whenever:

1. there is an injection of the lhs of $R_k$ in the solution $c_1, \ldots, c_m$;
2. the (injection/rule) produces the solution $c'_1, \ldots, c'_n$.

We are interested in $\text{Complex}_\omega$ the set of all complexes that can be constructed in one or several applications of rules in $\mathcal{R}$ starting from the set $\text{Complex}_0$ of initial complexes.

(We do not care about the number of occurrences of each complex).
Inductive definition

We define the mapping $F$ as follows:

$$F : \begin{cases} \wp(\text{Complex}) & \rightarrow \wp(\text{Complex}) \\ X & \mapsto X \cup \left\{ c_j' \mid \exists R_k \in \mathcal{R}, c_1, \ldots, c_m \in X, \\
& \quad c_1, \ldots, c_m \rightarrow_{R_k} c'_1, \ldots, c'_n \right\}. \end{cases}$$

The set $\wp(\text{Complex})$ is a complete lattice. The mapping $F$ is an extensive $\cup$-complete morphism.

We have:

$$\text{Complex}_\omega = \bigcup \{ F^n(\text{Complex}_0) \mid n \in \mathbb{N} \}.$$
Local views

\[ \alpha(\{R(Y1 \sim u, l!!1), E(r!1)\}) = \{R(Y1 \sim u, l!!r.E); E(r!!R)\}. \]
Galois connexion

Let \textit{local\_view} be the set of all local views.

Let \( \alpha \in \wp(\text{Complex}) \rightarrow \wp(\text{local\_view}) \) be the function that maps any set of complexes into the set of their local views.

The set \( \wp(\text{local\_view}) \) is a complete lattice. The function \( \alpha \) is a \( \cup \)-complete morphism.

Thus, it defines a Galois connexion:

\[
\wp(\text{Complex}) \xleftrightarrow{\gamma} \wp(\text{local\_view}).
\]

(The function \( \gamma \) maps a set of local views into the set of complexes that can be built with these local views).
\( \gamma \circ \alpha \) is an upper closure operator: it abstracts away some information.

Guess the image of the following set of complexes?
\( \alpha \circ \gamma \)

\( \alpha \circ \gamma \) is a lower closure operator: it simplifies (or reduces) constraints.

Guess the image of the following set of local views?

\( \{ R \} \cdot \{ S \} \)
One more question

\(\alpha \circ \gamma\) is a lower closure operator: it simplifies (or reduces) constraints.

Guess the image of the following set of local views?

\[
\{ R_{\text{l}} ; R_{\text{r}} \}
\]
Abstract rules
Abstract counterpart to $F$

We define $F^\#$ as:

$$F^\# : \begin{cases} \phi(\text{local\_view}) & \rightarrow \phi(\text{local\_view}) \\ X & \mapsto X \cup \left\{ \mathcal{lv}' \mid \exists R_k \in R, \mathcal{lv}_1, \ldots, \mathcal{lv}_m \in X, \mathcal{lv}_1, \ldots, \mathcal{lv}_m \xrightarrow{\#} R_k \mathcal{lv}'_1, \ldots, \mathcal{lv}'_n \right\}. \end{cases}$$

We have:

- $F^\#$ is extensive;
- $F^\#$ is monotonic;
- $F \circ \gamma \subseteq \gamma \circ F^\#$;
- $F^\# \circ \alpha = \alpha \circ F \circ \gamma \circ \alpha$. 
Soundness

Theorem 1  Let:

1. \((D, \subseteq, \cup)\) and \((D^\#, \subseteq, \cup)\) be chain-complete partial orders;
2. \(D \xleftrightarrow{\gamma}{\alpha} D^\#\) be a Galois connexion;
3. \(F \in D \rightarrow D\) and \(F^\# \in D^\# \rightarrow D^\#\) be monotonic mappings such that:
   \[ F \circ \gamma \subseteq \gamma \circ F^\#; \]
4. \(x_0 \in D\) be an element such that: \(x_0 \subseteq F(x_0)\);

Then:

1. both \(lfp_{x_0} F\) and \(lfp_{\alpha(x_0)} F^\#\) exist,
2. \(lfp_{x_0} F \subseteq \gamma(lfp_{\alpha(x_0)} F^\#).\)
Combinatorial blow up
Avoiding combinatorial blow up
Packing

• packing strategy: a covering $P$ of the set of sites.
• template: a class $p \in P$ in the covering.
• projection: for any $p \in P$, the function $\Pi_p$ restricts a local view to $p$.
• sub-local-view: $\text{sub\_local\_view} = \{\Pi_p(lv) | lv \in \text{local\_view}, \ p \in P\}$.
• abstraction: the following function

$$\alpha_{\text{PACKS}} : \left\{ \begin{array}{l}
\wp(\text{local\_view}) \mapsto \wp(\text{sub\_local\_view}) \\
X \mapsto \{\Pi_p(lv) | lv \in X, \ p \in P\}.
\end{array} \right.$$  

is a $\bigcup$-complete morphism.

It defines a Galois connexion $\wp(\text{local\_view}) \xleftarrow{\gamma_{\text{PACK}}} \xrightarrow{\alpha_{\text{PACK}}} \wp(\text{sub\_local\_view})$.

• counterpart: $\mathbb{F}^d_{\text{PACK}} = \alpha_{\text{PACK}} \circ \mathbb{F}^d \circ \gamma_{\text{PACK}}$.  

Jérôme Feret, LIENS 28 December, 2007
Automatic packing

Let $P$ be a covering such that:

1. each site is at least in one class of $P$;
2. whenever a rule tests two sites $a$ and $b$, then, for any $p \in P$, $a \in p \iff b \in p$;
3. whenever a rule modifies two sites $a$ and $b$, then, for any $p \in P$, $a \in p \iff b \in p$;
4. whenever a rule modifies a site $a$ and tests a site $b$, then for any $p \in P$, $a \in p \implies b \in p$.

Then, for any $X \in \mathcal{P}_{\text{local_view}}, n \in \mathbb{N}$,

$$\mathcal{P}_{\text{PACKS}}\left(\mathcal{P}_{\text{PACKS}}^n \left( \alpha_{\text{PACKS}}(X) \right) \right) = \mathcal{P}^n(X).$$

So, automatic packing never changes analysis results !!!
Overview

1. Introduction
2. Kappa language
3. Local views
4. Local set of complexes
5. Local rule systems
6. Decontextualization
7. Conclusion
Concretization

For any $X \in \wp(\text{local_view})$, $\gamma(X)$ is given by a rewrite system:

For any $lv \in X$, we add the following rules:

$I$ and semi-links are non-terminal.
Pumping lemma

• We use this rewrite system to enumerate the complexes of $\gamma(X)$.

• There are two cases:
  1. either there is a finite number of partial rewrite sequences;
  2. or we encounter cyclic derivations
     i.e. an open complex with a cycle of the following form:

$$R.l-r.E \ldots R.l-r.E$$

     can be built.

• We only enumerate complexes that are reached through an acyclic rewriting computation.

• We prove later that: if $X \in \alpha(\varphi(\text{Complex}))$ then each partial rewrite sequence is the prefix of a terminating rewrite sequence.
Examples

It is time to make some demos.
Which information is abstracted away?

Our analysis is exact (no false positive):
- for EGF cascade (356 complexes);
- for FGF cascade (18 051 984 143 555 729 567 complexes);

We know how to build systems with false positives... but they seem to be biologically meaningless.

This raises the following issues:
- Can we characterize which information is abstracted away?
- Which is the form of the systems, for which we have no false positive?
- Do we learn something about the biological systems that we describe?
Which information is abstracted away?

**Theorem 2** We suppose that:

1. \((D, \subseteq)\) be a partial order;
2. \((D^\#, \subseteq, \sqcup)\) be chain-complete partial order;
3. \(D \xrightleftharpoons[\gamma]{\alpha} D^\#\) be a Galois connexion;
4. \(F \in D \to D\) and \(F^\# \in D^\# \to D^\#\) are monotonic;
5. \(F \circ \gamma \subseteq \gamma \circ F^\#\);
6. \(x_0, inv \in D\) such that:
   - \(x_0 \subseteq F(x_0) \subseteq F(inv) \subseteq inv\),
   - \(inv = \gamma(\alpha(inv))\),
   - and \(\alpha(F(\gamma(\alpha(inv)))) = F^\#(\alpha(inv))\);

Then, \(lfp_{\alpha(x_0)}F^\#\) exists and \(\gamma(lfp_{\alpha(x_0)}F^\#) \subseteq inv\).
When is there no false positive?

**Theorem 3** We suppose that:

1. \((D, \subseteq, \cup)\) and \((D^\#, \subseteq, \cup)\) are chain-complete partial orders;
2. \((D, \subseteq) \xleftarrow{\gamma} (D^\#, \subseteq)\) is a Galois connexion;
3. \(F : D \rightarrow D\) is a monotonic map;
4. \(x_0\) is a concrete element such that \(x_0 \subseteq F(x_0)\);
5. \(F^\# \circ \alpha = \alpha \circ F \circ \gamma \circ \alpha\)

Then:

\[
\text{lfp}_{x_0} F \in \gamma(D) \iff \text{lfp}_{x_0} F = \gamma(\text{lfp}_{\alpha(x_0)} F^\#).
\]
Local set of complexes

Definition 1  We say that a set $X \in \wp(\text{Complex})$ of complexes is local if and only if it satisfies $X = \gamma(\alpha(X))$.

Remark 1  A set $X \in \wp(\text{Complex})$ is local if and only if $X \in \gamma(\wp(\text{local\_view}))$. 
Swapping relation

We define the binary relation $\sim_{\text{SWAP}}$ among tuples $\text{Complex}^*$ of complexes. We say that $(C_1, \ldots, C_m) \sim_{\text{SWAP}} (D_1, \ldots, D_n)$ if and only if:

$(C_1, \ldots, C_m)$ matches with

while $(D_1, \ldots, D_n)$ matches with
Theorem 4  Let $X \in \wp(\text{Complex})$ be a set of complexes. The two following assertions are equivalent:

1. the set $X \in \wp(\text{Complex})$ is local (i.e. $\gamma(\alpha(X)) = X$);

2. for any tuples $(C_i), (D_j) \in \text{Complex}^*$ such that:
   - $(C_i) \in X^*$,
   - $(C_i)^{\text{SWAP}} \sim (D_j)$;
we have $(D_j) \in X^*$. 
Proof (easier implication way)

If:
- \( X = \gamma(\alpha(X)) \),
- \((C_i) \in X^*\),
- and \((C_i) \xleftrightarrow{SWAP} (D_j)\);

Then:
we have \( \alpha(\{C_i\}) = \alpha(\{D_j\}) \) (because \((C_i) \xleftrightarrow{SWAP} (D_j)\))
and \( \alpha(\{C_i\}) \subseteq \alpha(X) \) (because \((C_i) \in X^* \) and by monotonicity);
so \( \alpha(\{D_j\}) \subseteq \alpha(X) \);
so \( \{D_j\} \subseteq \gamma(\alpha(X)) \) (by def. of Galois connexions);
so \( \{D_j\} \subseteq X \) (since \( X = \gamma(\alpha(X)) \));
so \((D_j) \in X^*\).
Proof (more difficult implication way)

We suppose that $X$ is close with respect to $\sim^{\text{SWAP}}$. We want to prove that $\gamma(\alpha(X)) \subseteq X$.

We prove, by induction, that any open complex that can be built using the rewrite system can be embedded in a complex in $X$:

- By def. of $\alpha$, this is satisfied for any local view in $\alpha(X)$;
- This remains satisfied after unfolding a semi-link with a local view;
- This remains satisfied after binding two semi-links.
Initialization

\[ C \in X \quad \text{(since} \quad lv \in \alpha(X)) \]
Unfolding a semi-link

C ∈ X

open complex

C' ∈ X
Unfolding a semi-link

C' ∈ X

Jérôme Feret, LIENS
December, 2007
Binding two semi-links

C ∈ X

open complex

C'' ∈ X

(open complex)

SWAP

(Jérôme Feret, LIENS, December, 2007)
Consequences

Let $X \in \alpha(\wp(C))$ be the set of local views that occurs in a set of complexes.

1. Each open complex $C$ built with the local views in $X$ is a sub-complex of a close complex $C'$ in $\gamma(X)$.

2. When considering the rewrite system that computes $\gamma(X)$, any partial rewriting sequence can be completed in a successful one.

3. We have $F^\# \circ \alpha = \alpha \circ F \circ \gamma \circ \alpha$.

4. We have $\text{lfp}_{\alpha(x_0)} F^\# \in \alpha(\wp(\text{Complex}))$. 
Overview

1. Introduction
2. Kappa language
3. Local views
4. Local set of complexes
5. Local rule systems
6. Decontextualization
7. Conclusion
We have proved that:

- if the set $\text{Complex}_\omega$ of reachable complexes is close with respect swapping $\sim_{\text{SWAP}}$,
- then the reachability analysis is exact (i.e. $\text{Complex}_\omega = \gamma(lfp_{\alpha}(\text{Complex}_0)_F^\#))$.

We sketch a proof in order to discover sufficient conditions that ensure this property:

- We consider tuples of complexes in which the same kind of links occur twice.
- We want to swap these links.
- We introduce the history of their computation.
- There are several cases...
First case (I/V)

complex ∈ Complex_ω

complex ∈ Complex_ω
First case (II/V)
just before the links are made

complex ∈ Complex^*_ω

complex ∈ Complex^*_ω
First case (III/V)

we suppose we can swap the links

complex ∈ Complex_ω^∗
First case (IV/V)

Then, we ensure that further computation steps:

- are always possible;
- have the same effect on local views;
- commute with the swapping relation $\mathbin{\sim}^{\text{SWAP}}$.

\[
\begin{array}{c}
C_n \\ \sim^\sigma \\
\downarrow R,\phi \\
C_n + 1 \sim^\sigma \\
\end{array} \quad \mathbin{\sim}^{\text{SWAP}} \quad \\
\begin{array}{c}
C'_n \\ \sim^\sigma \\
\downarrow R,\phi \\
C'_n + 1 \sim^\sigma \\
\end{array}
\]
First case (V/V)

$\text{complex } \in \text{Complex}_{\omega}^*$
Second case (I/II)

\[ \text{complex} \in \text{Complex}_\omega \]

we assume that the complex \textit{complex} is acyclic
Second case (II/II)
Sufficient conditions

Whenever the following assumptions:

1. initial agents are not bound;
2. rules are atomic;
3. rules are local:
   - only agents that interact are tested,
   - no cyclic patterns (neither in lhs, nor in rhs);
4. binding rules do not interfere i.e. if both:
   - $A(a\sim m,S),B(b\sim n,T) \rightarrow A(a\sim m!1,S),B(b\sim n!1,T)$
   - and $A(a\sim m',S'),B(b\sim n',T') \rightarrow A(a\sim m'!1,S'),B(b\sim n'!1,T')$,
   then:
   - $A(a\sim m,S),B(b\sim n',T') \rightarrow A(a\sim m!1,S),B(b\sim n'!1,T')$;
5. . . .

are satisfied, the set of reachable complexes is local.
Non local systems

\( \text{Complex}_0 \triangleq R(a \sim u) \)

Rules

\( \triangleq \begin{cases} 
R(a \sim u) & \leftrightarrow R(a \sim p) \\
R(a \sim u), R(a \sim u) & \rightarrow R(a \sim u!1), R(a \sim u!1) \\
R(a \sim p), R(a \sim u) & \rightarrow R(a \sim p!1), R(a \sim p!1) \\
R(a \sim p), R(a \sim p) & \rightarrow R(a \sim p!1), R(a \sim p!1) 
\end{cases} \)

\( R(a \sim u!1), R(a \sim u!1) \in \text{Complex}_\omega \)

\( R(a \sim p!1), R(a \sim p!1) \in \text{Complex}_\omega \)

But \( R(a \sim u!1), R(a \sim p!1) \notin \text{Complex}_\omega \).
Non local systems

\[ \begin{align*}
\text{Complex}_0 \triangleq & \quad A(a \sim u), B(a \sim u) \\
\text{Rules} \triangleq & \quad \left\{ \\
& \quad \begin{array}{l}
A(a \sim u), B(a \sim u) \rightarrow A(a \sim u!1), B(a \sim u!1) \\
A(a \sim u!1), B(a \sim u!1) \rightarrow A(a \sim p!1), B(a \sim u!1) \\
A(a \sim u!1), B(a \sim u!1) \rightarrow A(a \sim u!1), B(a \sim p!1)
\end{array}
\right. \\
\end{align*} \]

A(a \sim u!1), B(a \sim p!1) \in \text{Complex}_\omega \\
A(a \sim p!1), B(a \sim u!1) \in \text{Complex}_\omega \\
\text{But } A(a \sim p!1), B(a \sim p!1) \not\in \text{Complex}_\omega.

Jérôme Feret, LIENS

December, 2007
Non local systems

\[ \text{Complex}_0 \triangleq A(a \sim u) \]

\[ \text{Rules} \triangleq \left\{ \begin{array}{l}
A(a \sim u) \leftrightarrow A(a \sim p) \\
A(a \sim u), A(a \sim p) \rightarrow A(a \sim u!1), A(a \sim p!1)
\end{array} \right\} \]

\[ A(a \sim u!1), A(a \sim p!1) \in \text{Complex}_\omega \]

But \[ A(a \sim p!1), A(a \sim p!1) \notin \text{Complex}_\omega. \]
Non local systems

\[ \text{Complex}_0 \triangleq R(a,b) \]

\[ \text{Rules} \triangleq \{ R(a,b), R(a) \rightarrow R(a,b!1), R(a!1) \} \]

\[ R(a,b!2), R(a!2,b!1), R(a!1,b) \in \text{Complex}_\omega \]

But \[ R(a!1,b!1) \notin \text{Complex}_\omega. \]
Third case (I/III)

complex ∈ Complex_ω
Third case (II/III)

$\text{complex} \in \text{Complex}_\omega^*$
Third case (II/III)

$\text{complex } \in \text{Complex}_{\omega}^*$
Dangerous sites

A site is dangerous if it may occur in a cycle within a complex ($\in \gamma(\alpha(Complex_\omega))$).

To ensure that the set of reachable complexes is local, we also require that:

- The binding state of a dangerous site is never tested, unless for binding or unbinding this site.
- When we bind dangerous sites, we only test that these sites are free.

Then, we prove that:

1. we can build any complex with free dangerous sites,
2. then, we can bind them as much as we like.
Overview

1. Introduction
2. Kappa language
3. Local views
4. Local set of complexes
5. Local rule systems
6. Decontextualization
7. Conclusion
Outline

• we have a syntactic criterion in order to ensure that the set of reachable complexes of a kappa system is local;

• we now design program transformations to help systems satisfying this criterion;

1. decontextualization
   - is fully automatic;
   - preserves the transition system;
   - simplifies rules thanks to reachability analysis.

2. conjugation
   - manual;
   - preserves the set of reachable complexes;
   - uses backtrack to add new rules.
Example

Initial rule:

\[ R_2(l!2, r), R_1(l!1, r), E_2(r!1), E_1(r!2) \rightarrow R_2(l!3, r!1), R_1(l!2, r!1), E_2(r!2), E_1(r!3) \]

Decontextualized rule:

\[ R_2(l!_, r), R_1(l!_, r) \rightarrow R_2(l!_, r!1), R_1(l!_, r!1) \]

We can remove redundant tests.
Example

Initial rules:

\[
\begin{align*}
\text{Sh}(Y7 \sim p!2, pi!1), G(a!2, b), R(Y48 \sim p!1) & \rightarrow \text{Sh}(Y7 \sim p, pi!1), G(a, b), R(Y48 \sim p!1) \\
\text{Sh}(Y7 \sim p!3, pi!1), G(a!3, b!2), So(d!2), R(Y48 \sim p!1) & \rightarrow \text{Sh}(Y7 \sim p, pi!1), G(a, b!2), So(d!2), R(Y48 \sim p!1) \\
\text{Sh}(Y7 \sim p!1, pi), G(a!1, b) & \rightarrow \text{Sh}(Y7 \sim p, pi), G(a, b) \\
\text{Sh}(Y7 \sim p!1, pi), G(a!1, b\_0) & \rightarrow \text{Sh}(Y7 \sim p, pi), G(a, b\_0)
\end{align*}
\]

Decontextualized rule:

\[
\text{Sh}(Y7!1), G(a!1) \rightarrow \text{Sh}(Y7), G(a)
\]

We can remove exhaustive enumerations.
How does it work?

To remove a test, we prove that:

- this test is satisfied whenever the other tests are satisfied;
- or each complex that passes all tests but this one also matches with the left hand side of another rule that performs the same action.
More formally:

- Each rule \( R \) is associated with the set \( S(R) \) of open complexes that can match its lhs;
- Rules are gathered in equivalence classes according to the actions they perform;
- For each class \([R]\), we compute:

\[
G([R]) = \bigcup \{S(R') \mid R' \in [R]\}.
\]

- For each class \([R]\), \( Reach([R]) \) is an over approximation of the set of open complexes that may match the lhs of a rule \( R' \in [R] \).

A rule \( R \) may be decontextualized in a rule \( R' \) if:

\[
S(R') \subseteq G([R]) \setminus Reach([R]).
\]

Decontextualization is more efficient, if the reachability analysis is accurate.
An undecontextualizable rule

Initial rule:

Sh(Y7⁻, pi!1), R(Y48⁻, p!1, r!_) -> Sh(Y7⁻, p, pi!1), R(Y48⁻, p!1, r!_)

Decontextualized rule:

Sh(Y7⁻, u, pi!1), R(Y48, p!1, r!_) -> Sh(Y7⁻, p, pi!1), R(Y48, p!1, r!_)
Conjugation

If a rule $R'$ is equivalent to a rule in the transitive closure of the system. Then it may be included in the system without modifying reachable states.

To remove the context $C$ of a rule, we try to apply it for another context $C'$ by:

1. removing the context $C'$ (backtrack) ;
2. building the context $C$ ;
3. applying the initial rule ;
4. removing the context $C$ (backtrack) ;
5. building the context $C'$.

This is proved manually.
Overview

1. Introduction
2. Kappa language
3. Local views
4. Local set of complexes
5. Local rule systems
6. Decontextualization
7. Conclusion
Conclusion

- A scalable static analysis to abstract the reachable complexes.
- A class of models for which the abstraction is complete.
- Many applications:
  - idiomatic description of reachable complexes;
  - dead rule detection;
  - rule decontextualization;
  - computer-driven kinetic refinement.
- It can also help simulation algorithms:
  - wake up/inhibition map (agent-based simulation);
  - flat rule system generation (for bounded set of complexes);
  - on the fly flat rule generation (for large/unbounded set)
Future works

• Reachability analysis.
  – Shape analysis: three valued logic, Mauborgne’s graph abstract domains.

• Quantitative analysis.
  – Can we lift the local-view abstraction to stochastic/differential semantics?
  – Which information do we obtain?

• Semi-quantitative abstractions.
  – Can we design abstract domains to discover semi-quantitative properties (i.e. overshoot detection)?