On The Computational Power of Biochemistry

Luca Cardelli

Microsoft Research

with

Gianluigi Zavattaro

University of Bologna

2008-08-01, Algebraic Biology '08

http://LucaCardelli.name

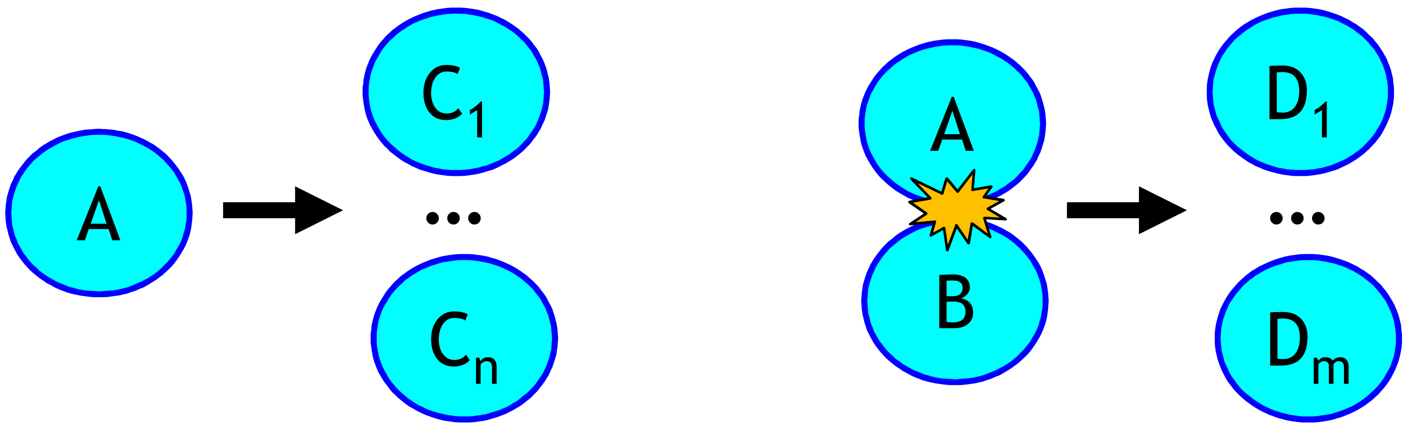
Talk Outline

- Basic Chemistry and Basic Biochemistry
  - Biochemistry = Chemistry + complexation
- Chemical Ground Form (CGF) [TCS08]
  - A process algebra for basic chemistry
  - Connections with basic chemistry (FSRN)
  - Basic chemistry can’t compute! [Sol08]
- Biochemical Ground Form (BGF) [AB08]
  - A process algebra for basic biochemistry
  - Basic biochemistry is Turing-complete.
- Conclusions
  - Basic biochemistry > Basic chemistry

Basic Chemistry

- Molecules belong to Species

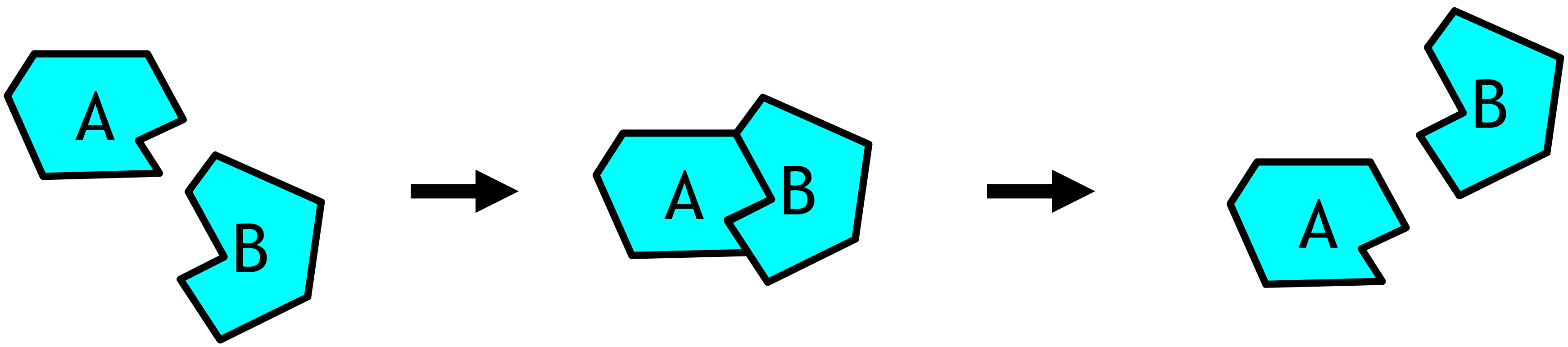
- Behavior is described by reactions between species:
  - Monomolecular: $A \rightarrow C_1+\ldots+C_n$
  - Bimolecular: $A+B \rightarrow D_1+\ldots+D_m$



- A.k.a. FSRN (Finite Stochastic Reaction Networks [Sol'08])

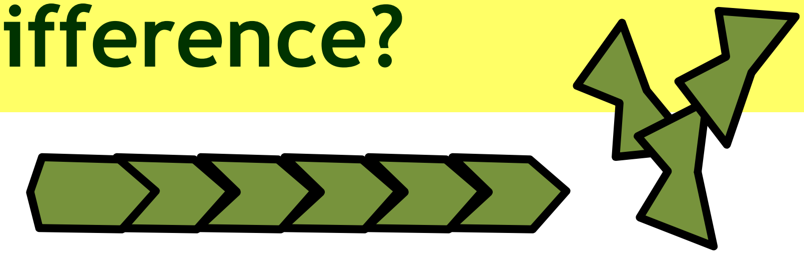
Basic Biochemistry

- Molecules may also form reversible complexes
  - Association: **$A + B \rightarrow A{:}B$**
  - Dissociation: **$A{:}B \rightarrow A + B$**



What's the Difference?

Consider linear polymerization:



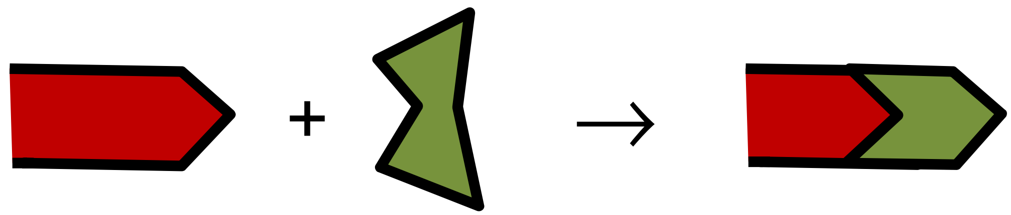
The "chemical program" for polymerization:

$P_0 + M \rightarrow P_1$
$P_1 + M \rightarrow P_2$
$P_2 + M \rightarrow P_3$
$P_3 + M \rightarrow P_4$
....

- an infinite (non-)program
- an infinite set of species
- an infinite set of ODEs

$P_{10757} + M \rightarrow P_{10758}$
Such specificity is unreal.

But "nature's program" for polymerization has to fit e.g. in the genome, so it cannot be infinite! Clearly, nature must be using a different "language" than basic chemistry:



molecule with convex patch +
molecule with concave patch →
molecule with convex patch

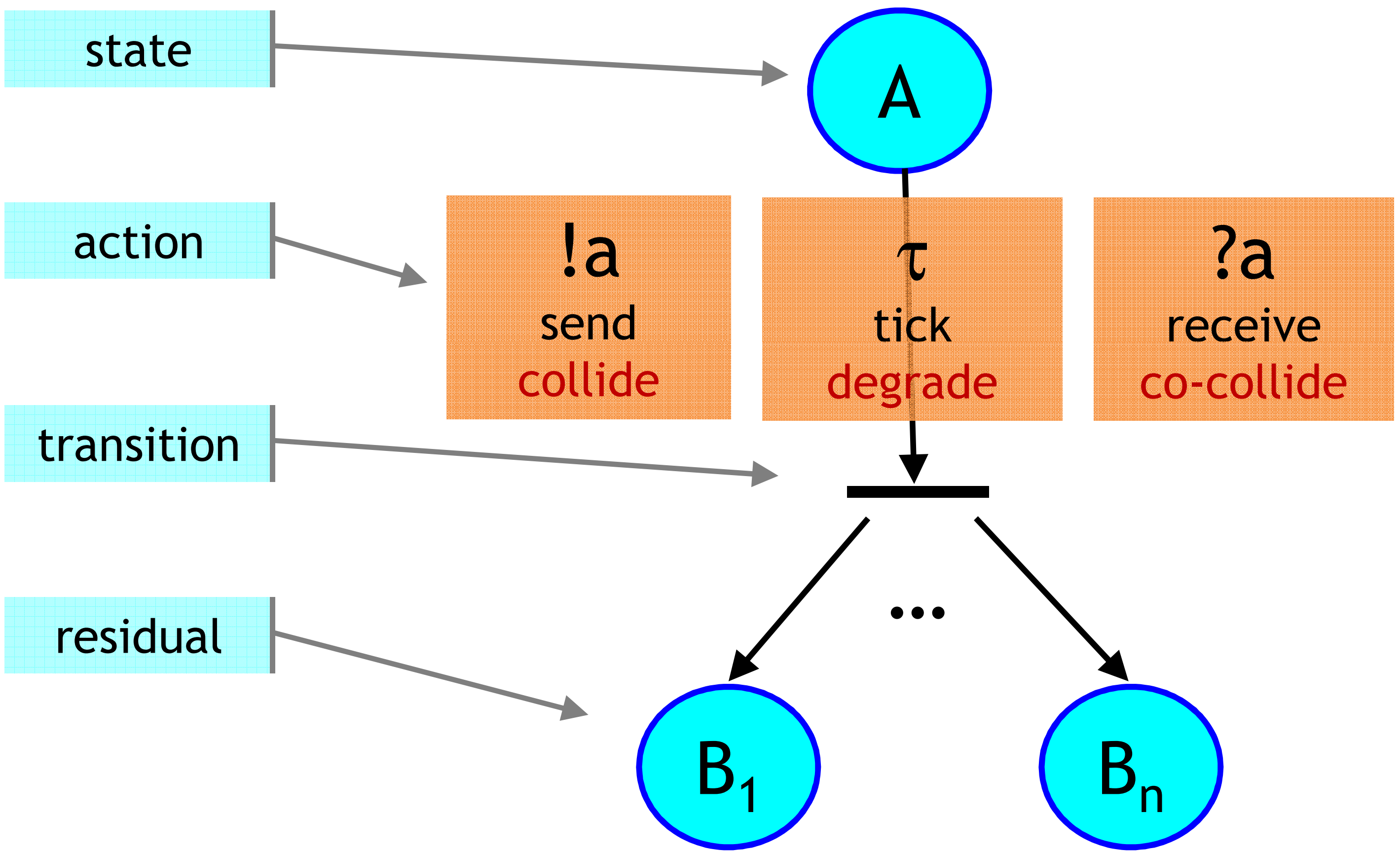
- a finite program
- a local rule

Talk Outline

- Basic Chemistry and Basic Biochemistry
  - Biochemistry = Chemistry + complexation
- Chemical Ground Form (CGF) [TCS08]
  - A process algebra for basic chemistry
  - Connections with basic chemistry (FSRN)
  - Basic chemistry can't compute! [Sol08]
- Biochemical Ground Form (BGF) [AB08]
  - A process algebra for basic biochemistry
  - Basic biochemistry is Turing-complete.
- Conclusions
  - Basic biochemistry > Basic chemistry

Chemical Ground Form (CGF)

- Stochastic variant of Milner's CCS, with an equivalent graphical notation (Stochastic Interacting Automata)

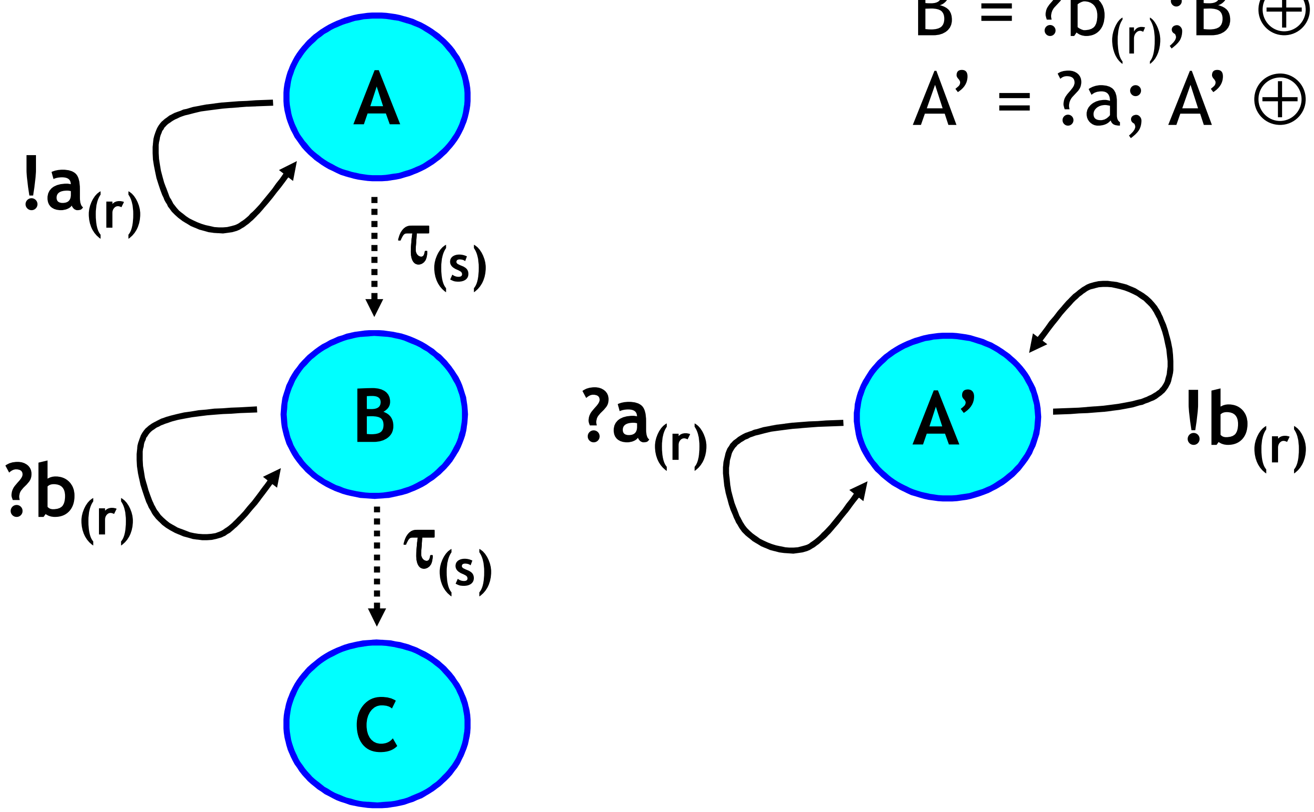


Stochastic Dynamics

- Transitions take (a variable amount of) time
- Each action has an associated rate **r**
  - Internal delay: $\tau_{(r)}$
    - **Pr(internal delay < t) = $1\text{-}e^{-rt}$**
  - Synchronization of complementary actions: $?a_{(r)}$, $!a_{(r)}$
    - **Pr(synchronization time < t) = $1\text{-}e^{-rt}$**

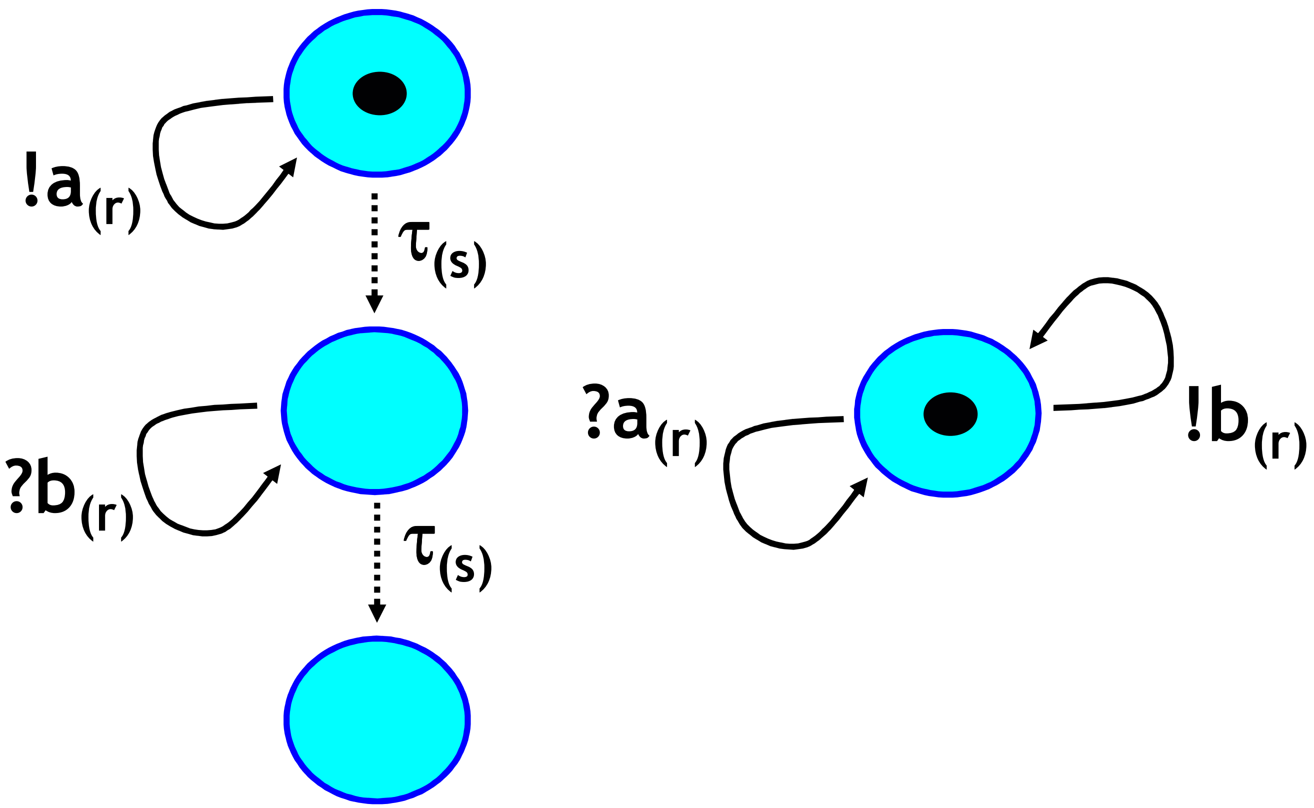
Example 1

$A = !a_{(r)};A \oplus \tau_{(s)};B$
$B = ?b_{(r)};B \oplus \tau_{(s)};C$
$A' = ?a; A' \oplus ?b_{(r)};A'$



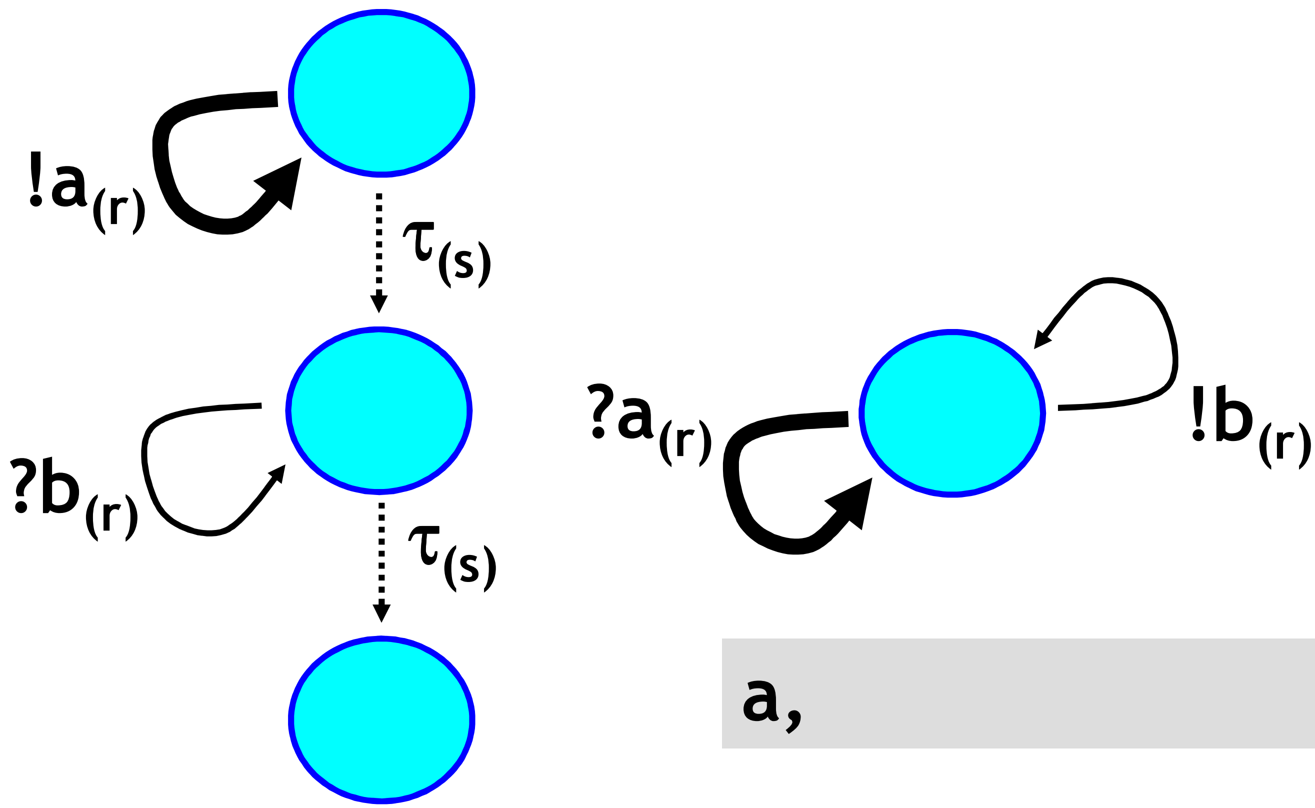
- Starting population: **A|A'**

Example 1



- Starting population: **A|A'**

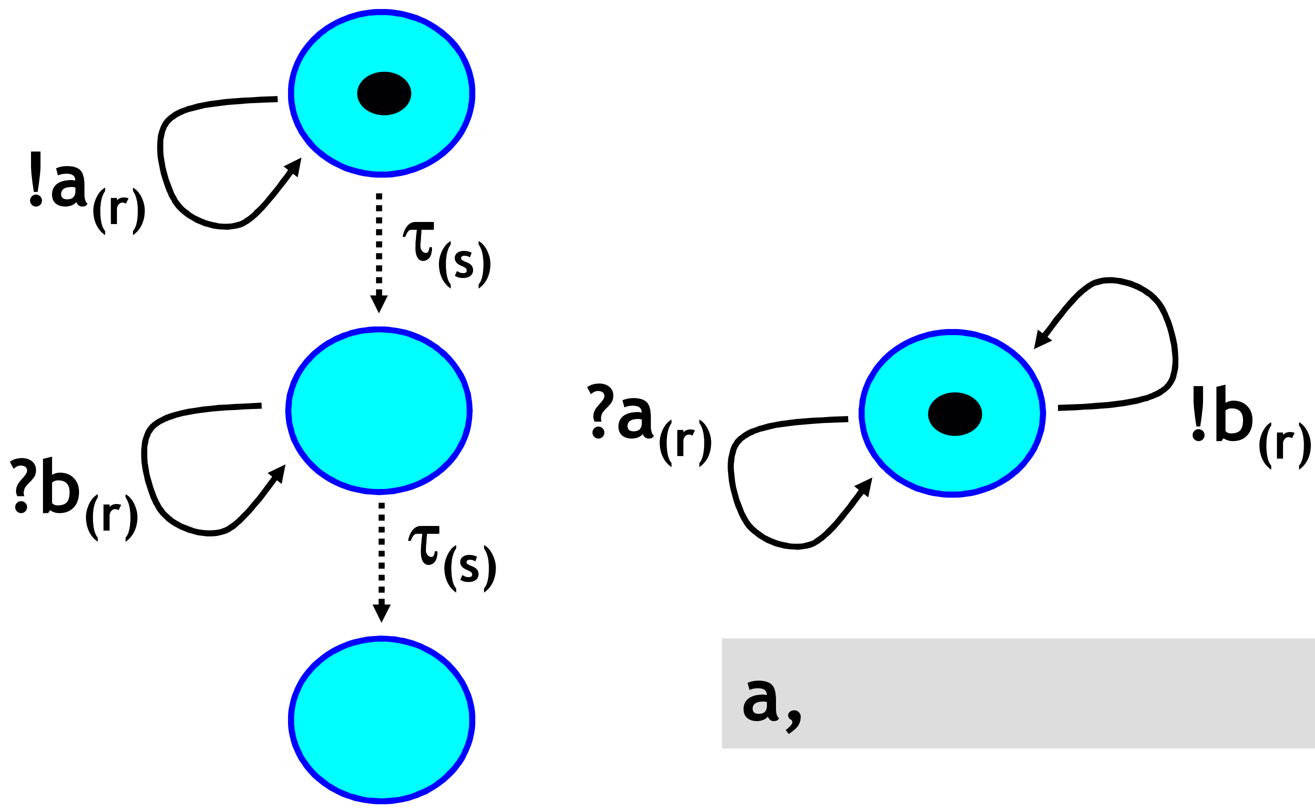
Example 1



a,

- Starting population: **A|A'**

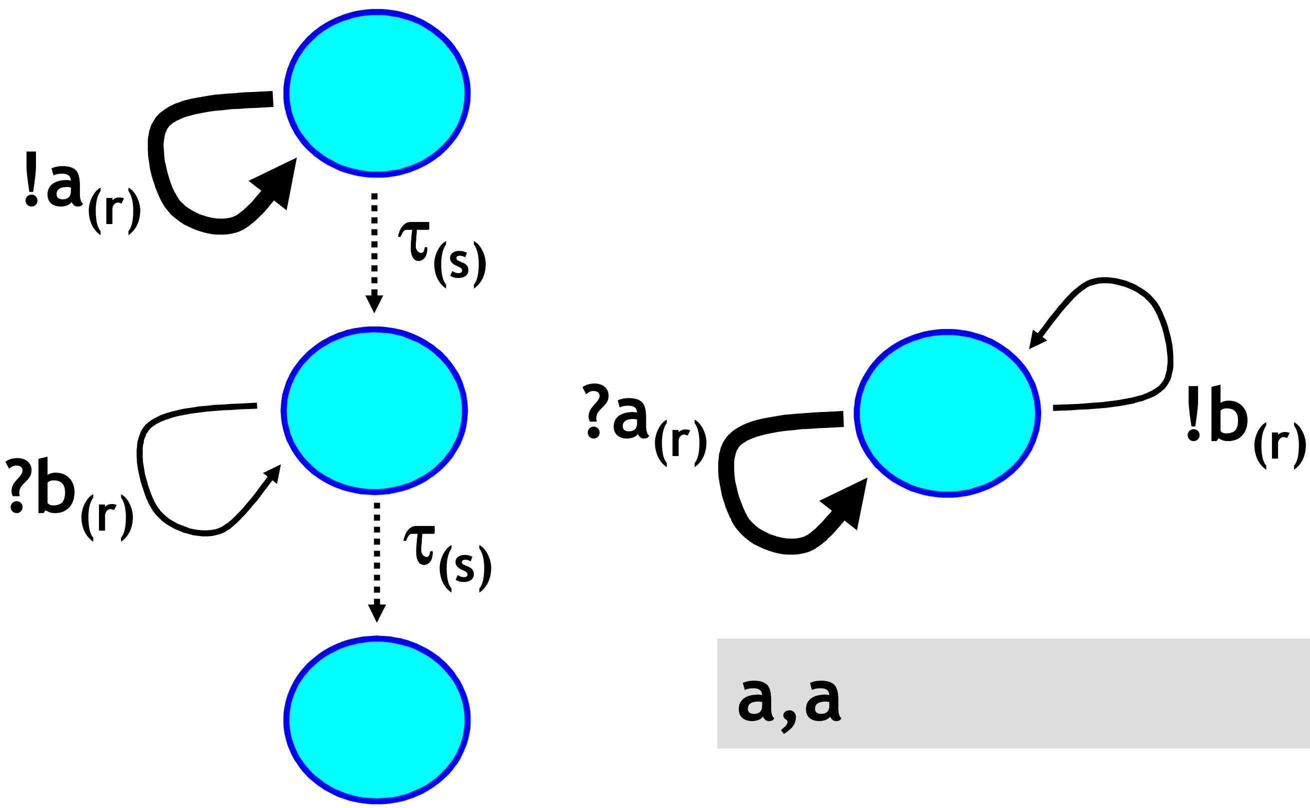
Example 1



a,

- Starting population: **A|A'**

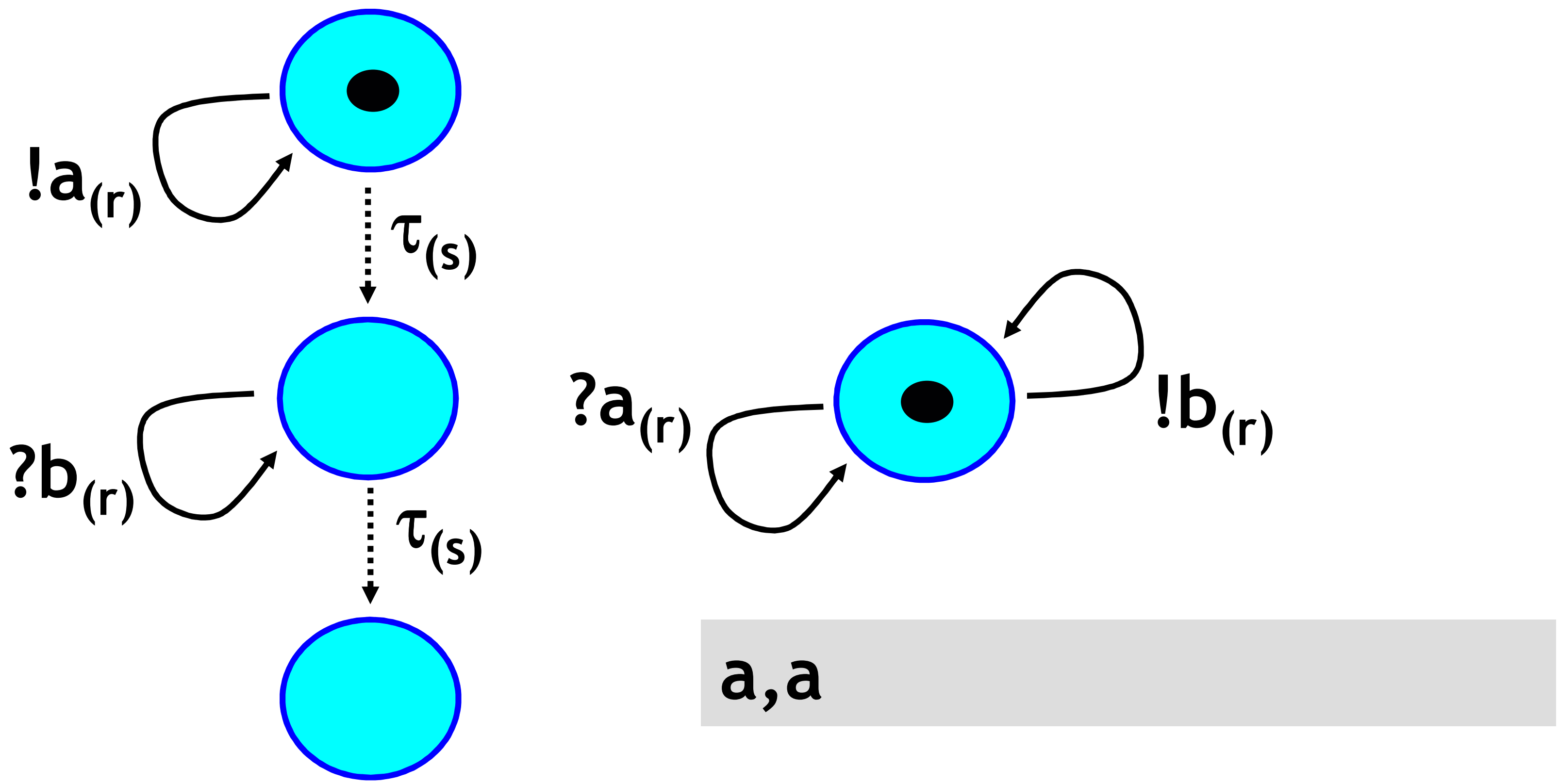
Example 1



a,a

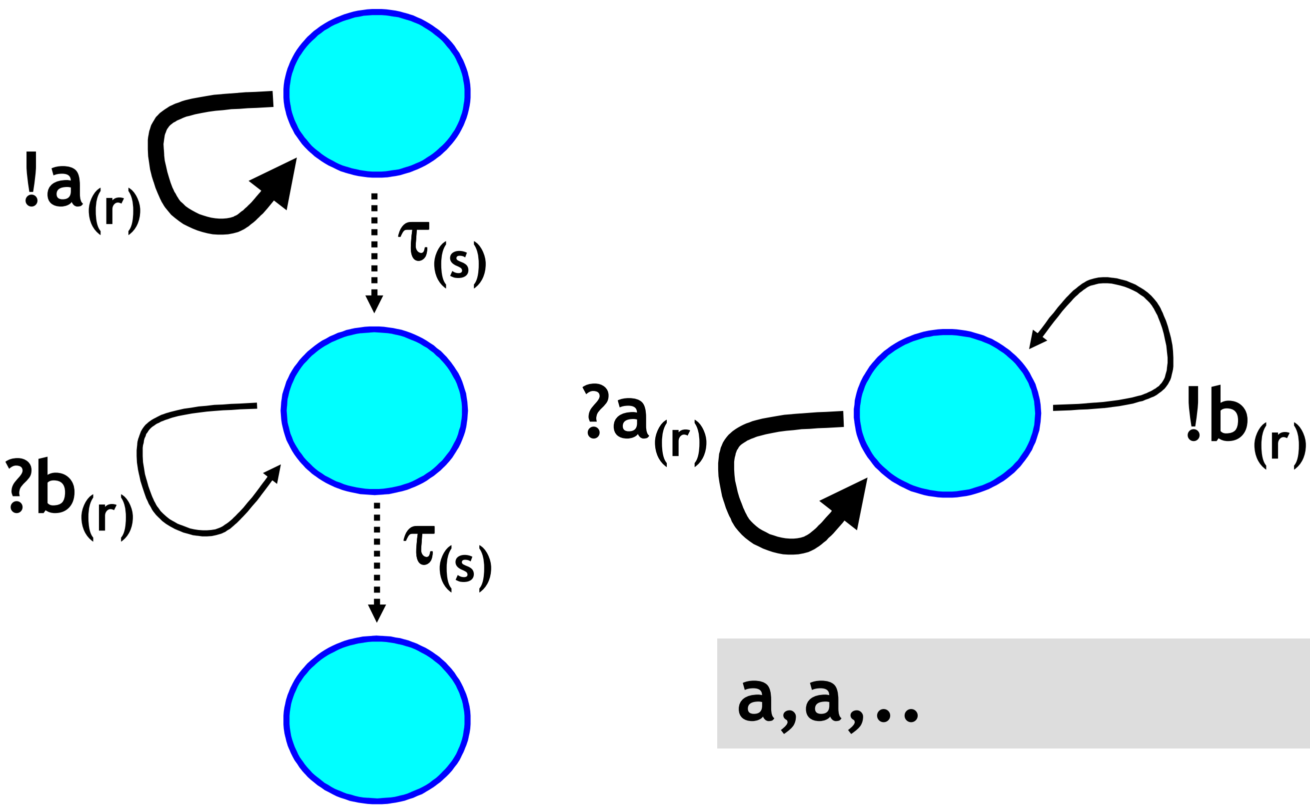
- Starting population: **A|A'**

Example 1



- Starting population: **A|A'**

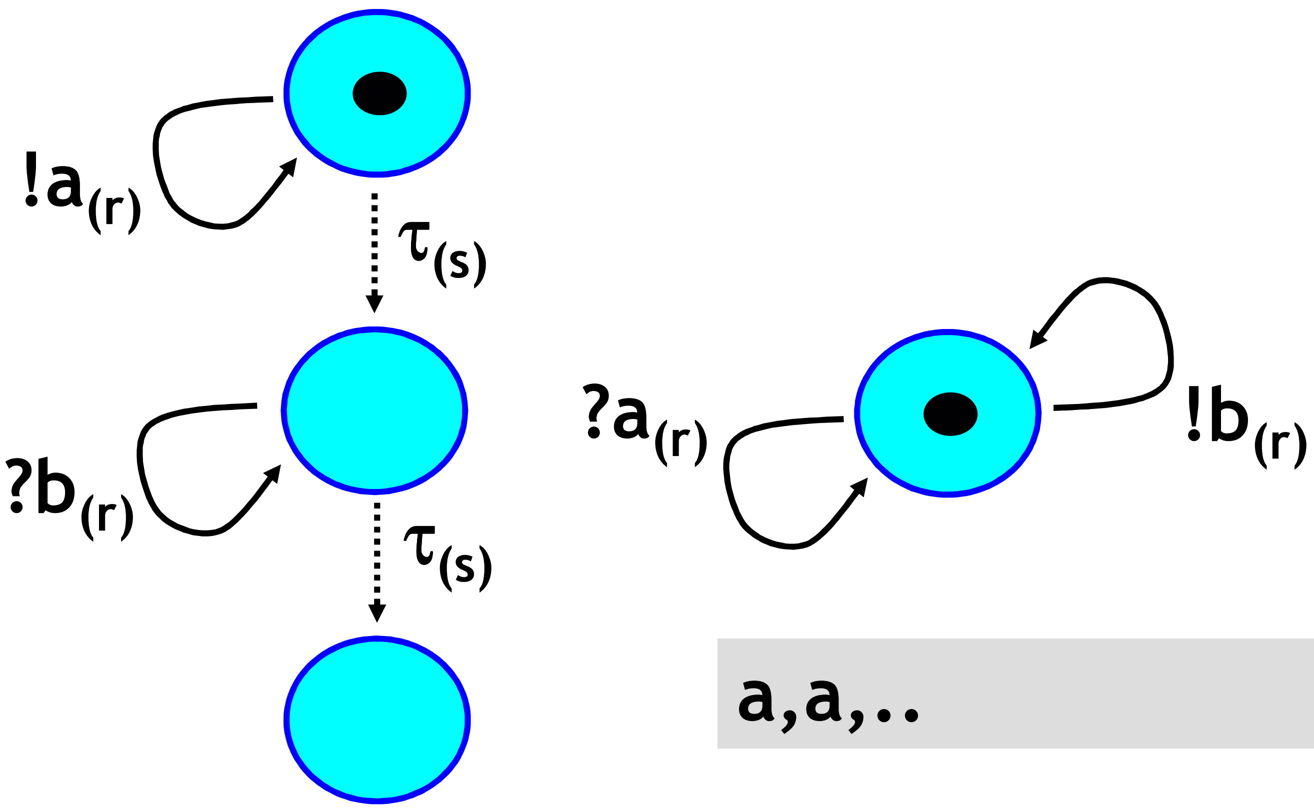
Example 1



a,a,..

- Starting population: **A|A'**

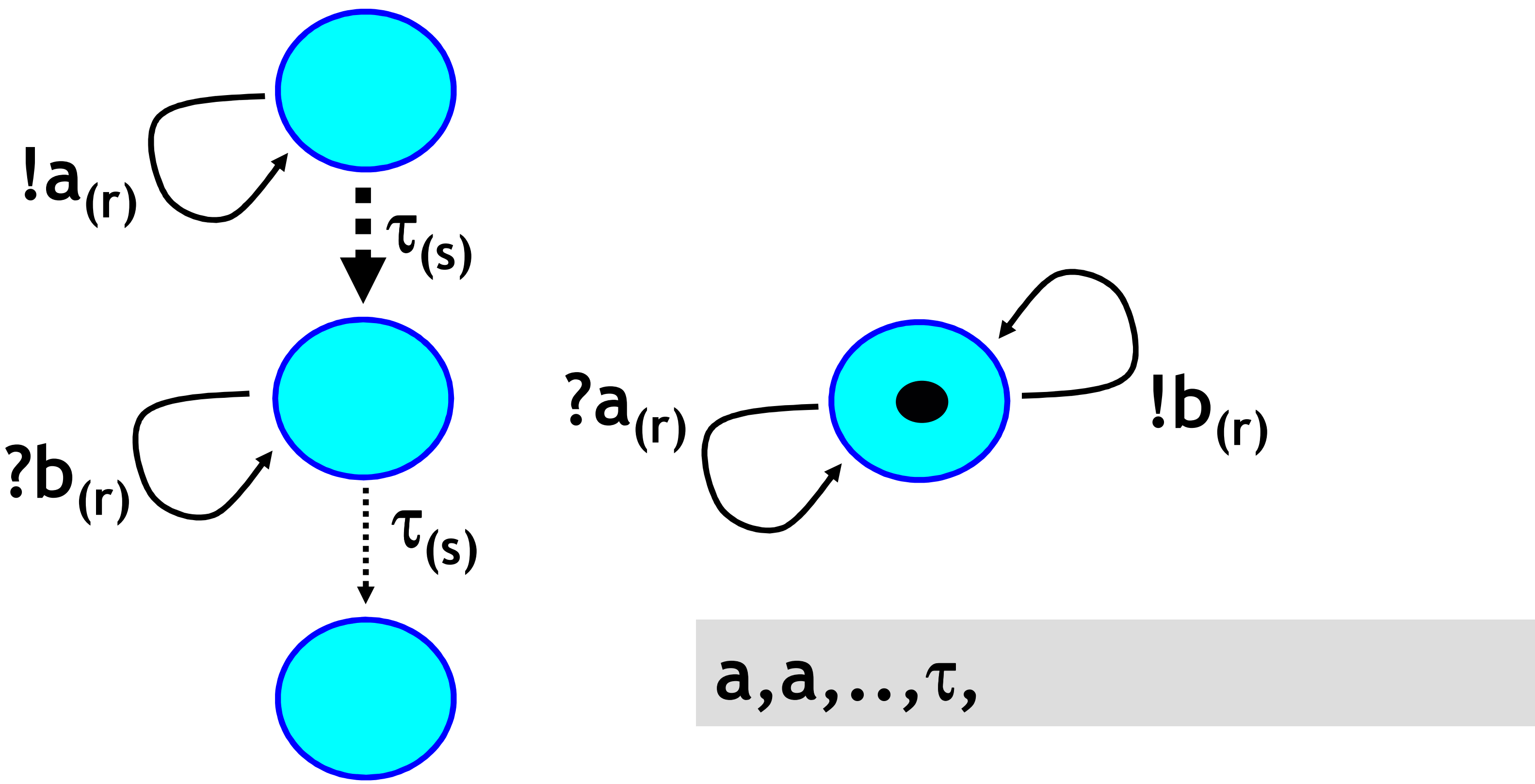
Example 1



a,a,..

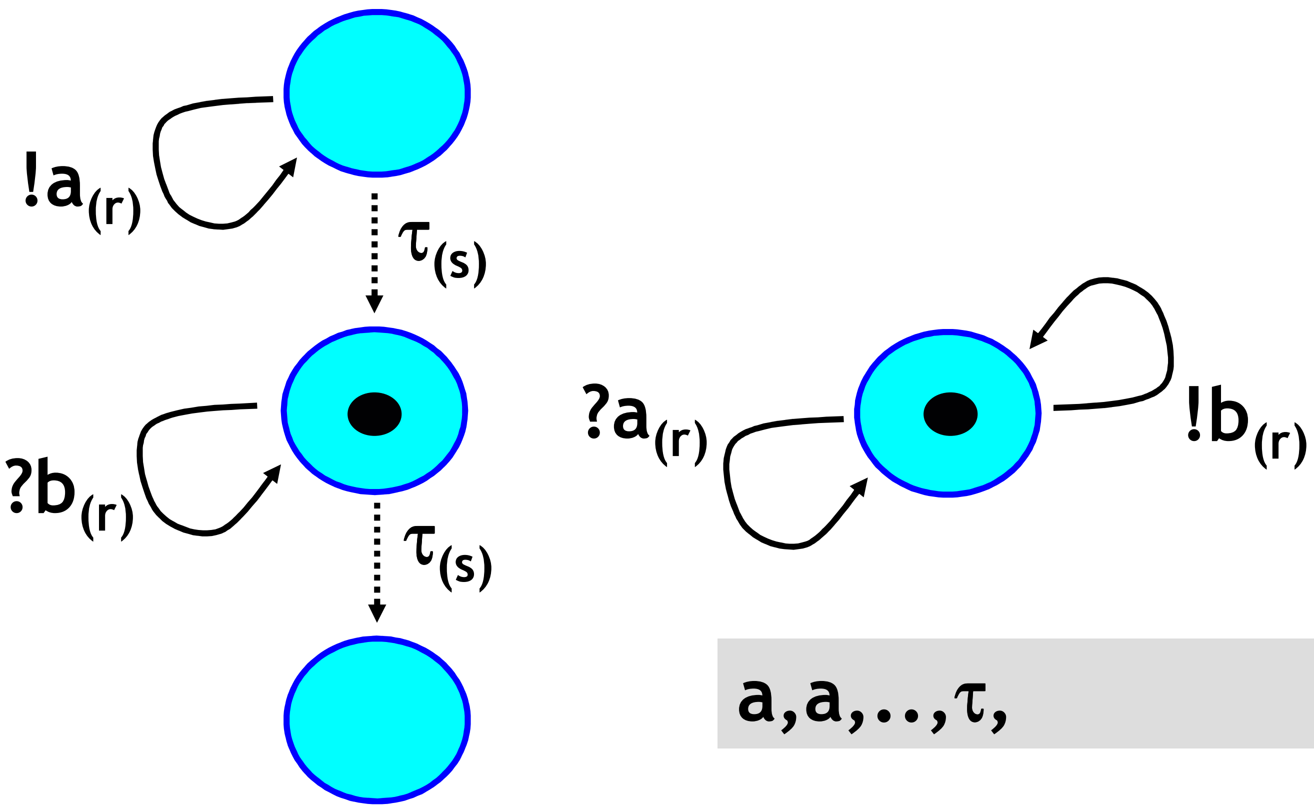
- Starting population: **A|A'**

Example 1



- Starting population: **A|A’**

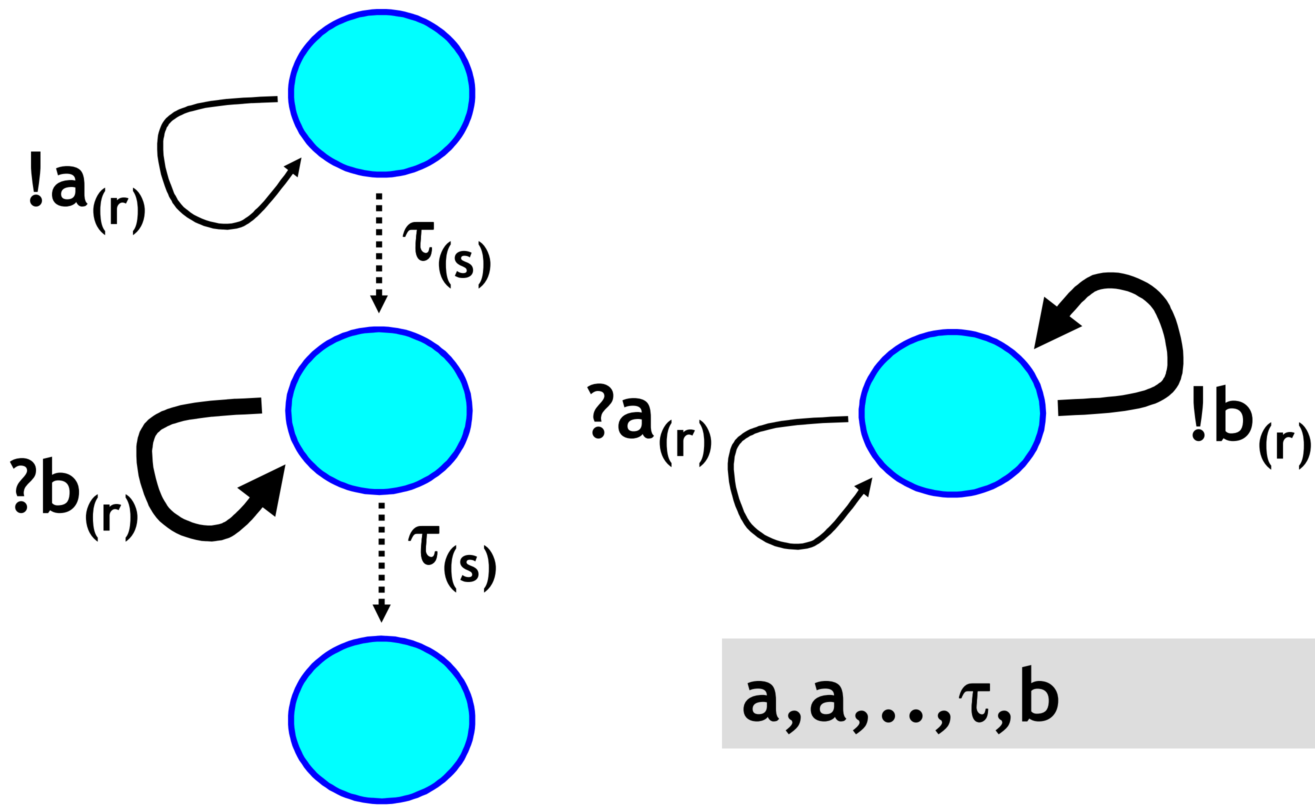
Example 1



a,a,..,τ,

- Starting population: **A|A'**

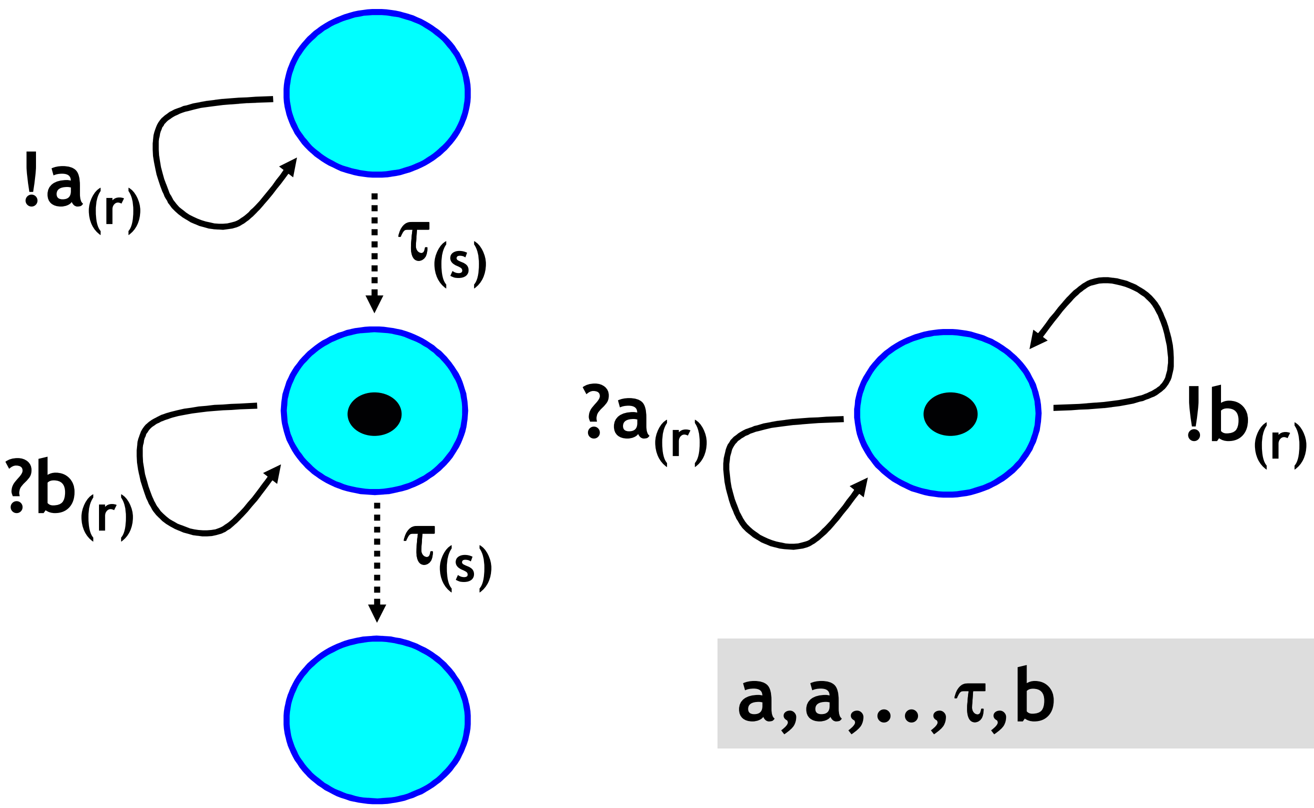
Example 1



a,a,..,τ,b

- Starting population: **A|A'**

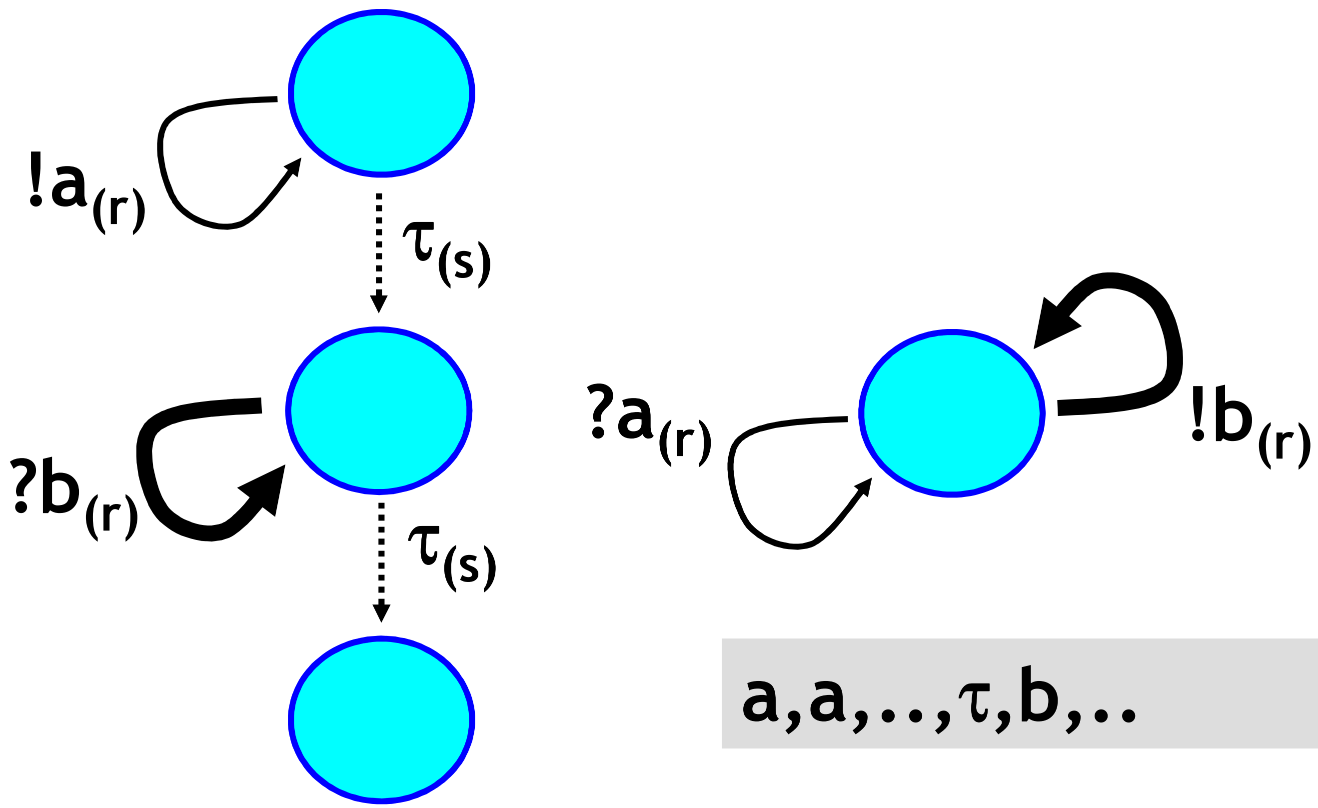
Example 1



a,a,..,τ,b

- Starting population: **A|A'**

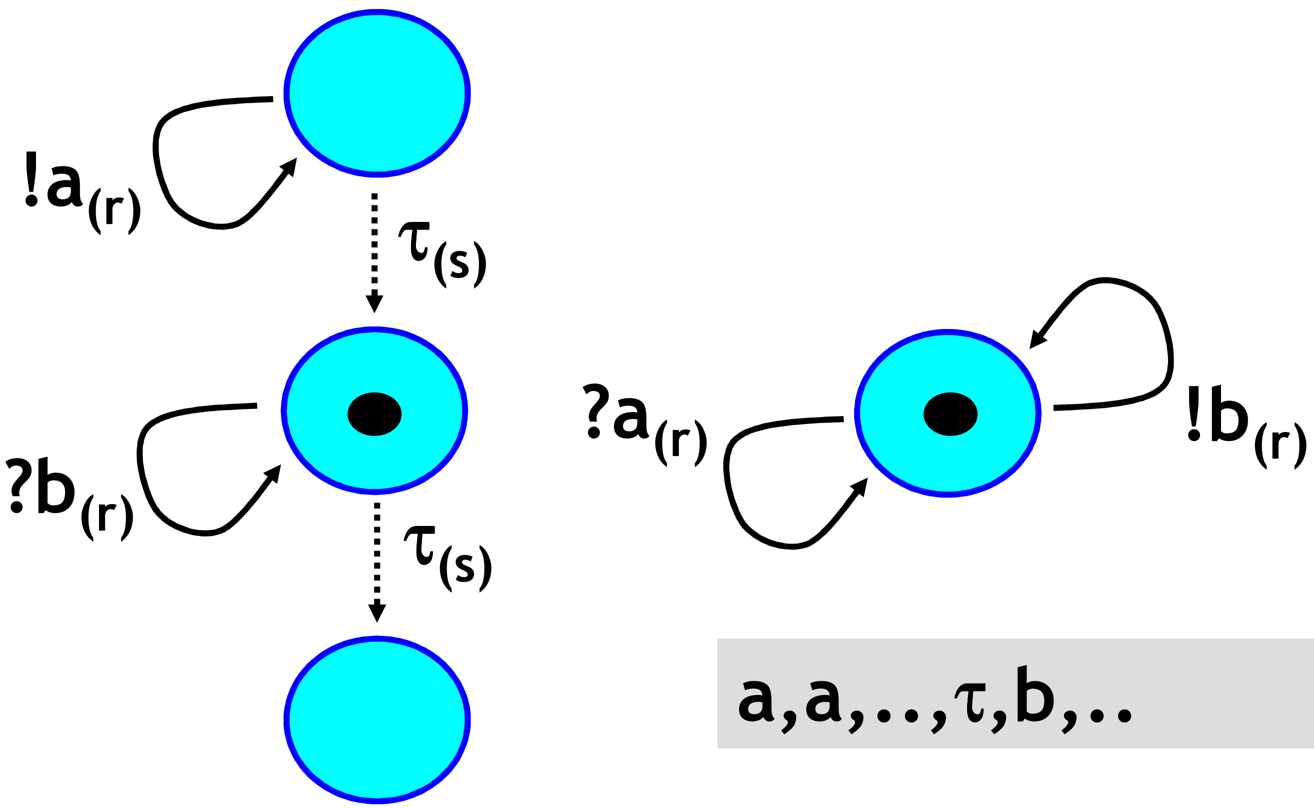
Example 1



a,a,..,τ,b,..

- Starting population: **A|A'**

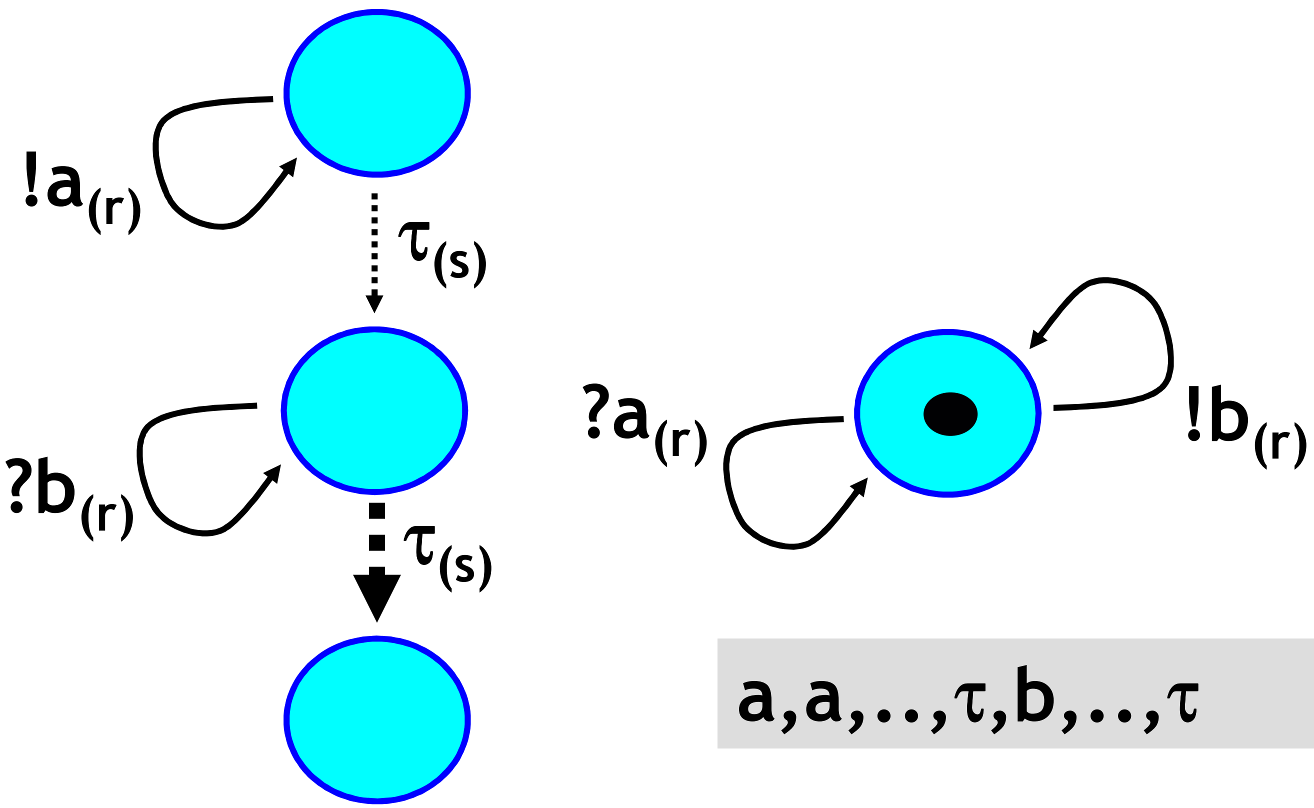
Example 1



a,a,..,τ,b,..

- Starting population: **A|A'**

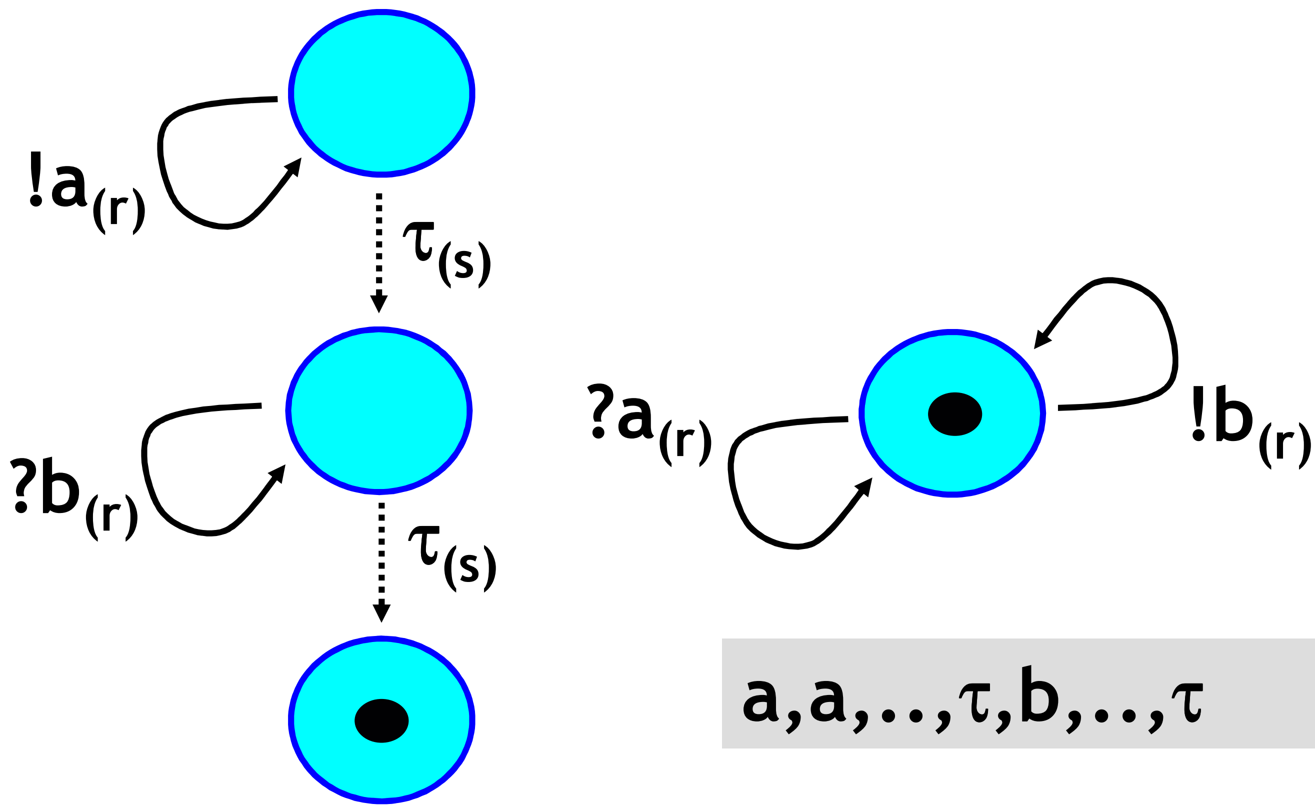
Example 1



a,a,..,τ,b,..,τ

- Starting population: **A|A'**

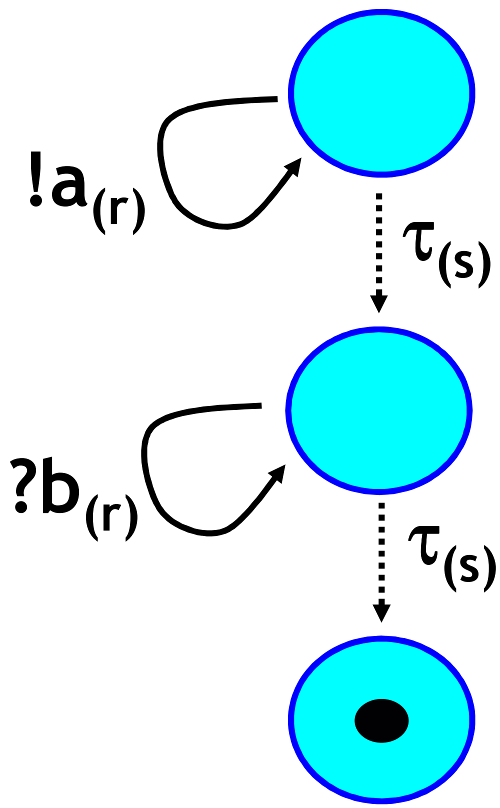
Example 1



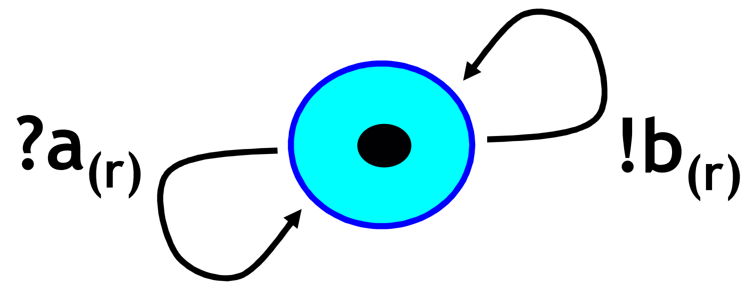
a,a,..,τ,b,..,τ

- Starting population: **A|A'**

Example 1



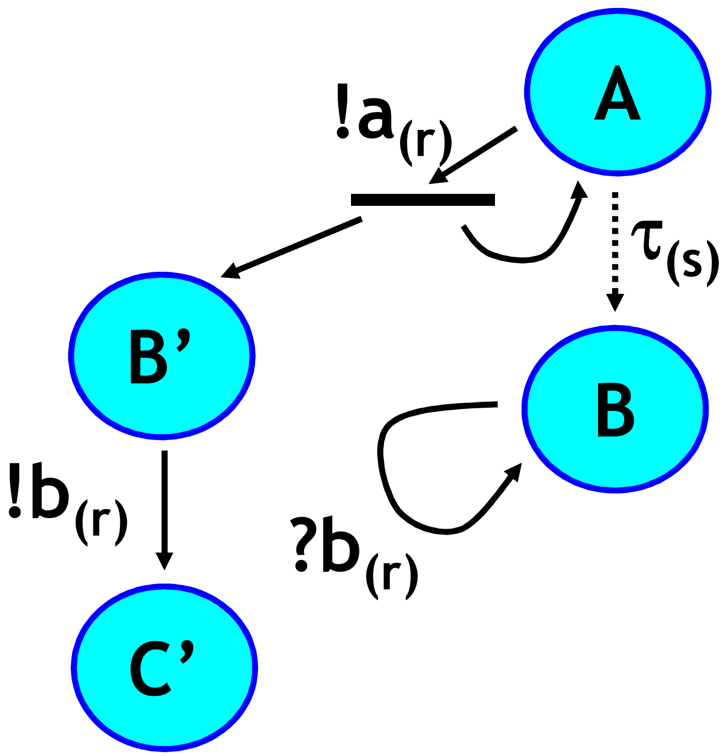
Stop



a*τb*τ

- Starting population: **A|A’**

Example 2

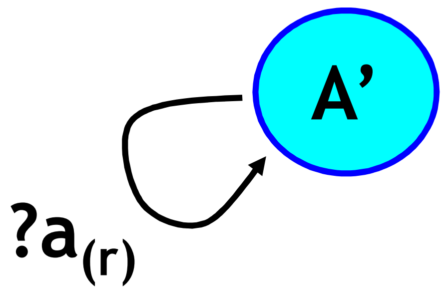


$A = !a_{(r)};(A|B') \oplus \tau_{(s)};B$

$B = ?b_{(r)};B$

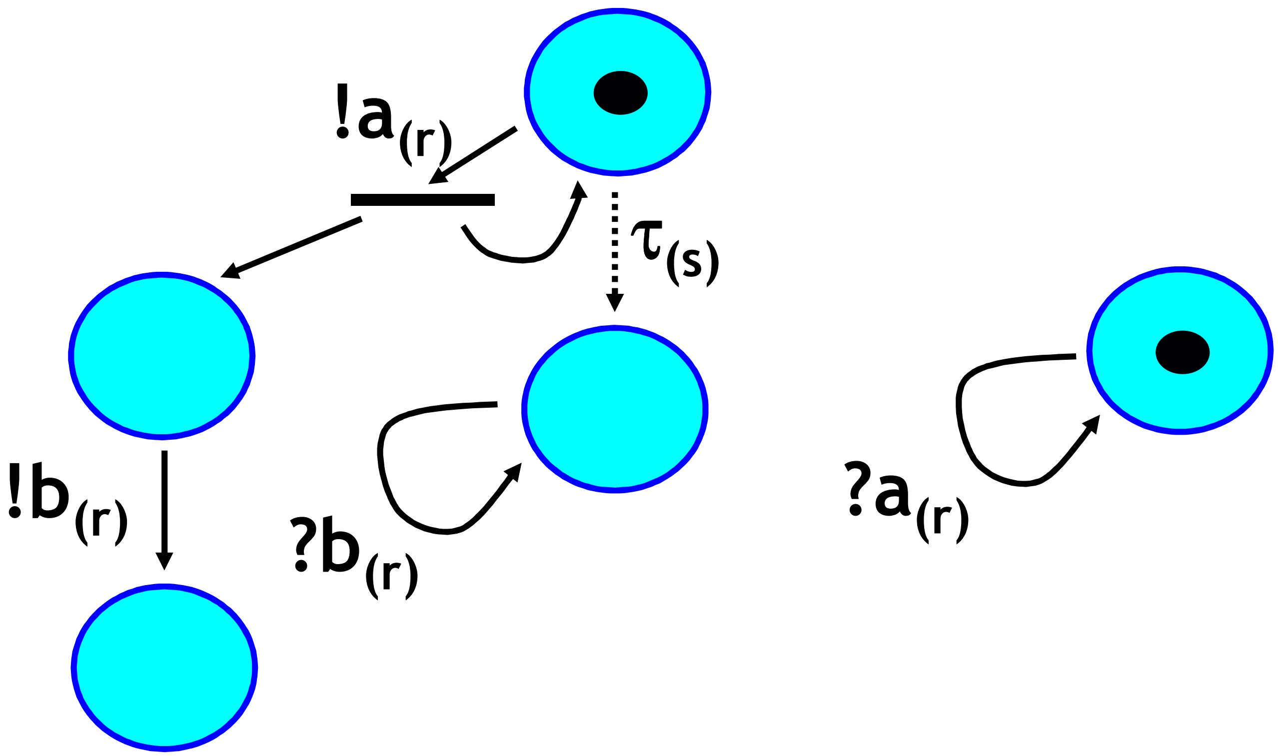
$B' = !b_{(r)};C'$

$A' = ?a_{(r)};A'$



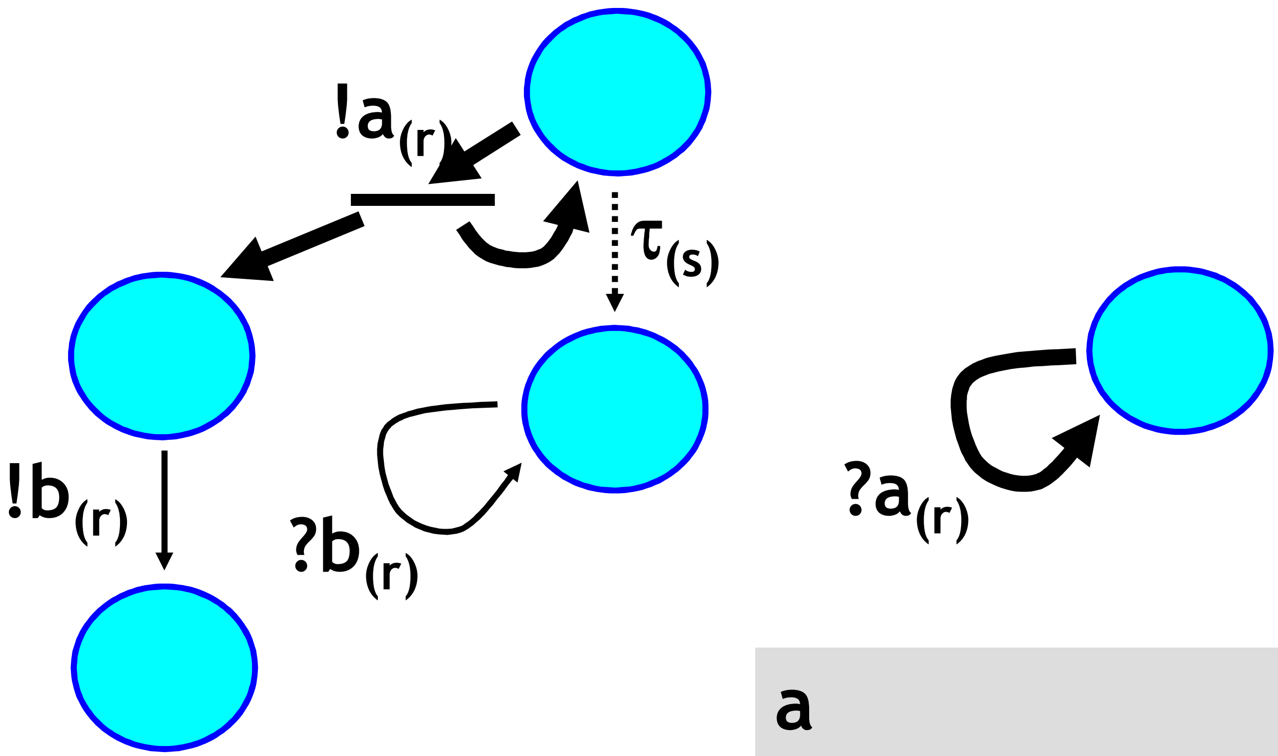
- Starting population: **A|A'**

Example 2



- Starting population: **A|A’**

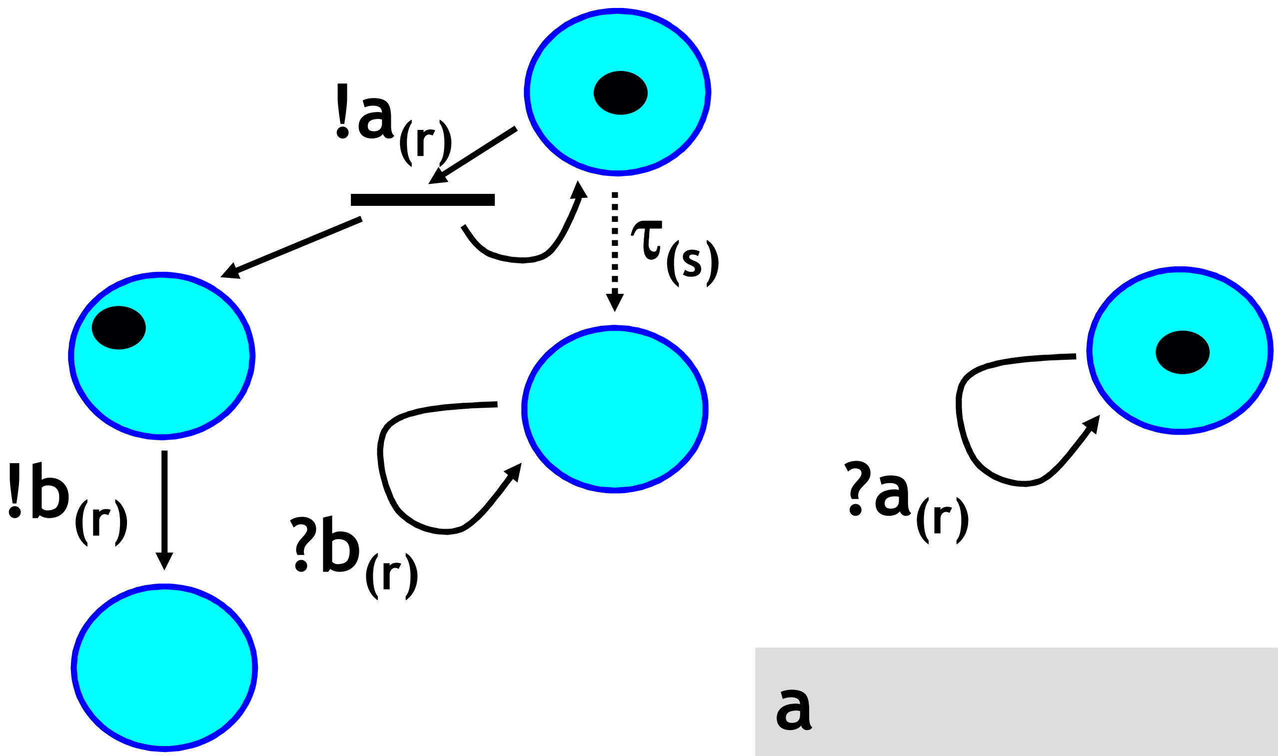
Example 2



a

- Starting population: **A|A'**

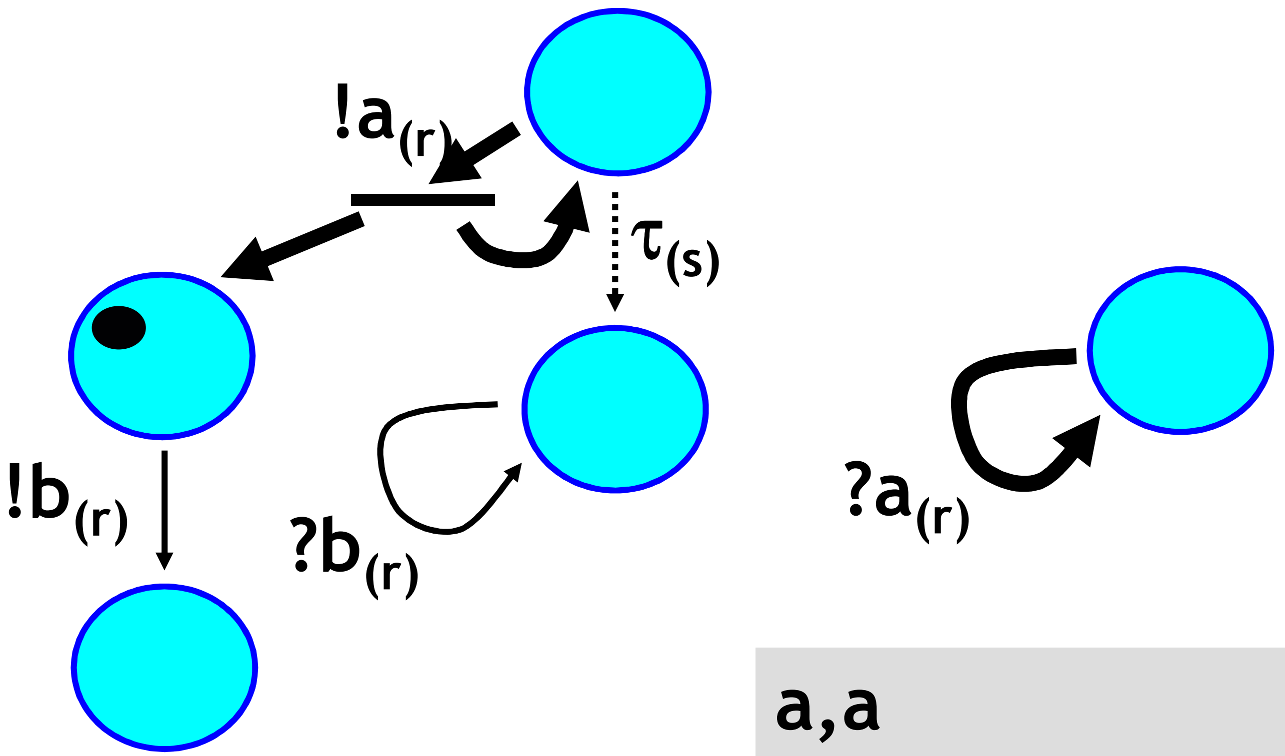
Example 2



a

- Starting population: **A|A'**

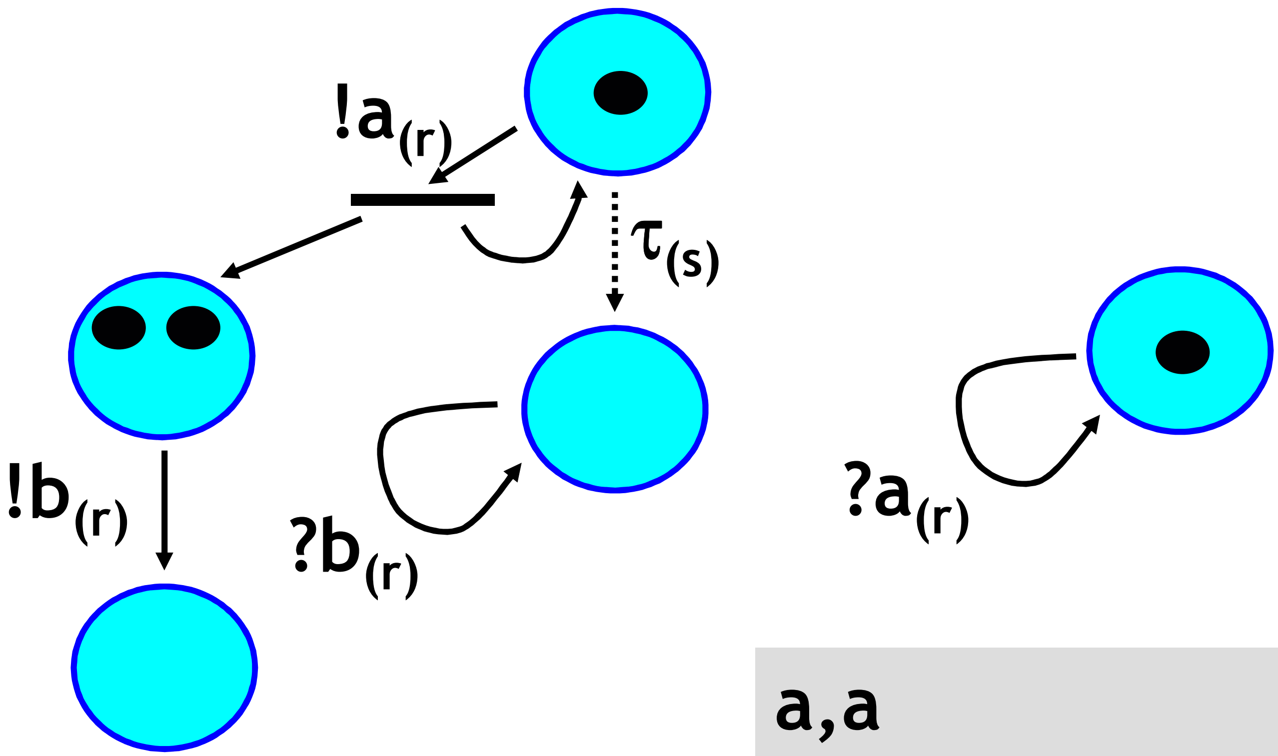
Example 2



a,a

- Starting population: **A|A'**

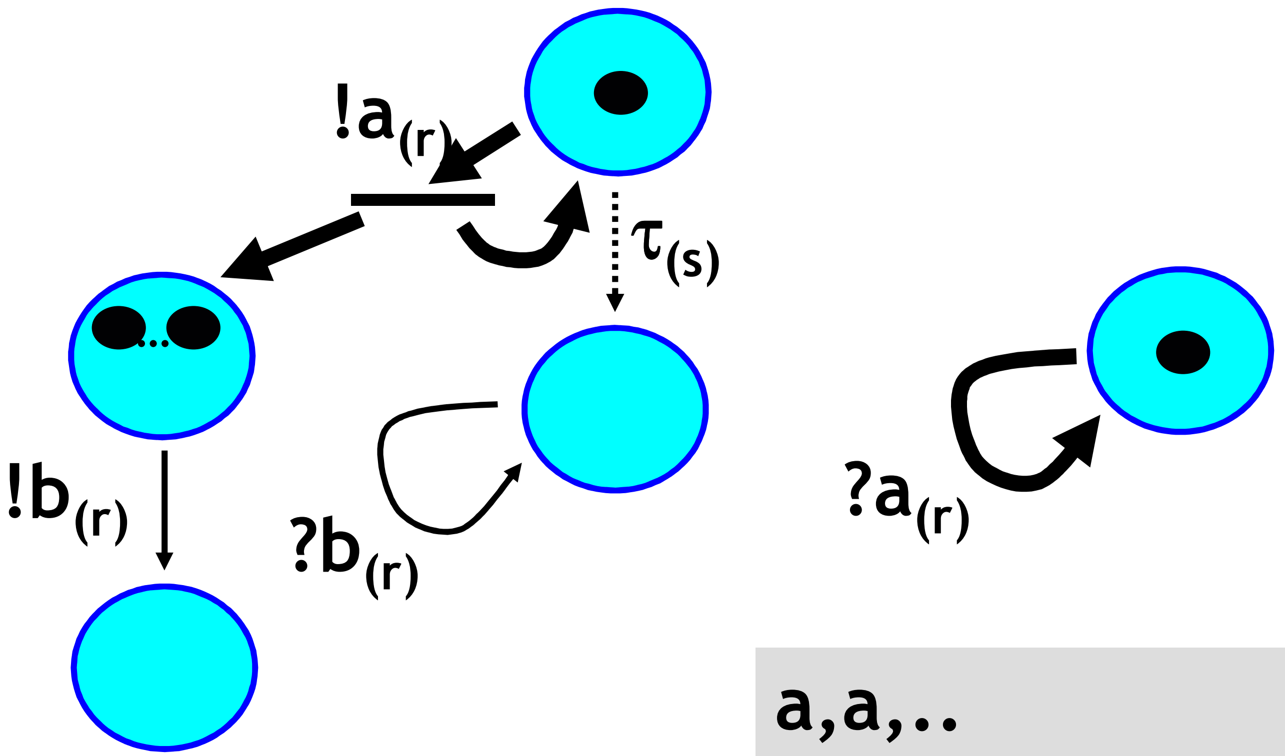
Example 2



a,a

- Starting population: **A|A'**

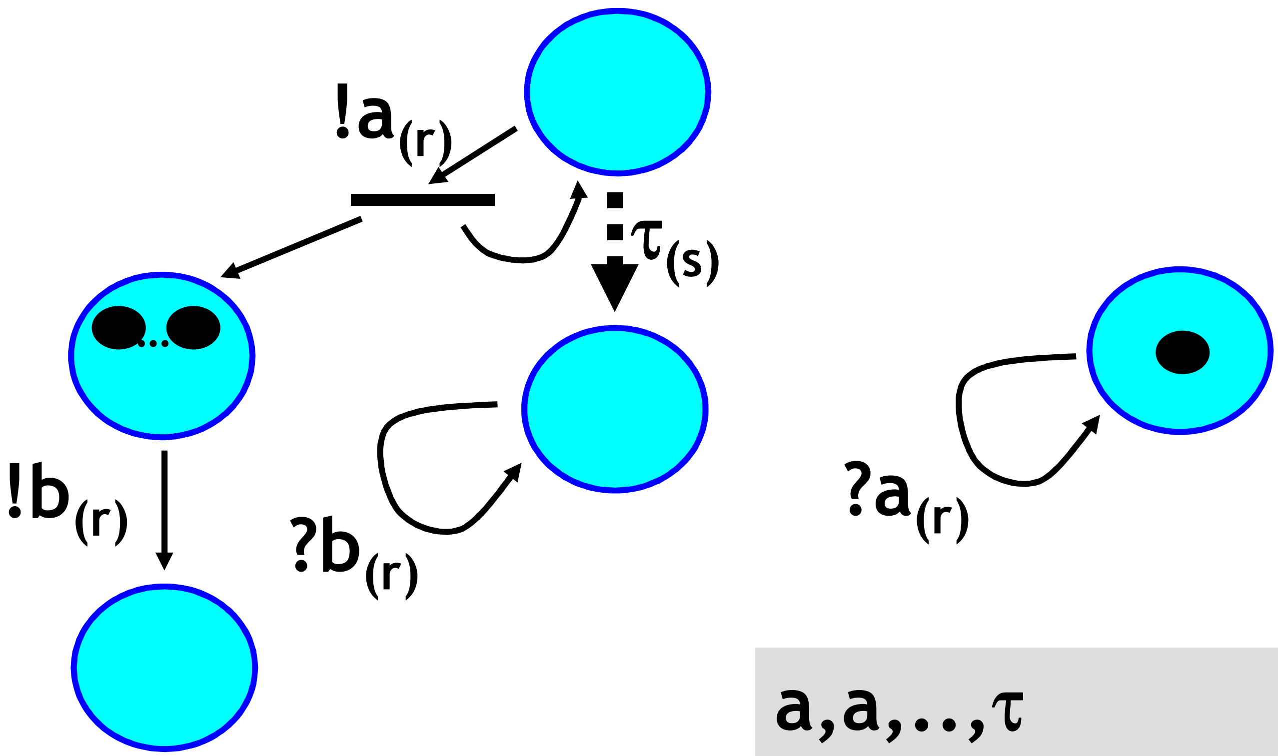
Example 2



a,a,..

- Starting population: **A|A’**

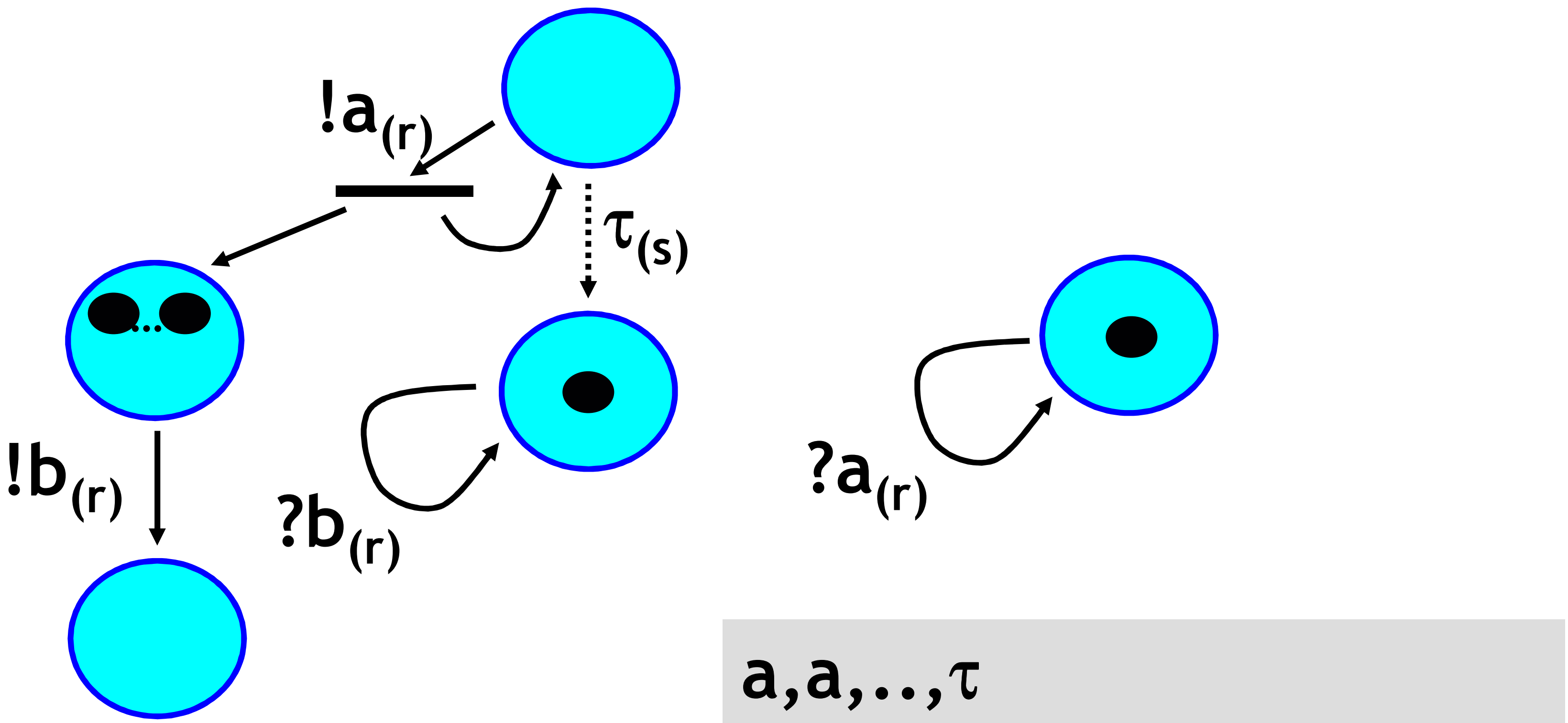
Example 2



a,a,..,τ

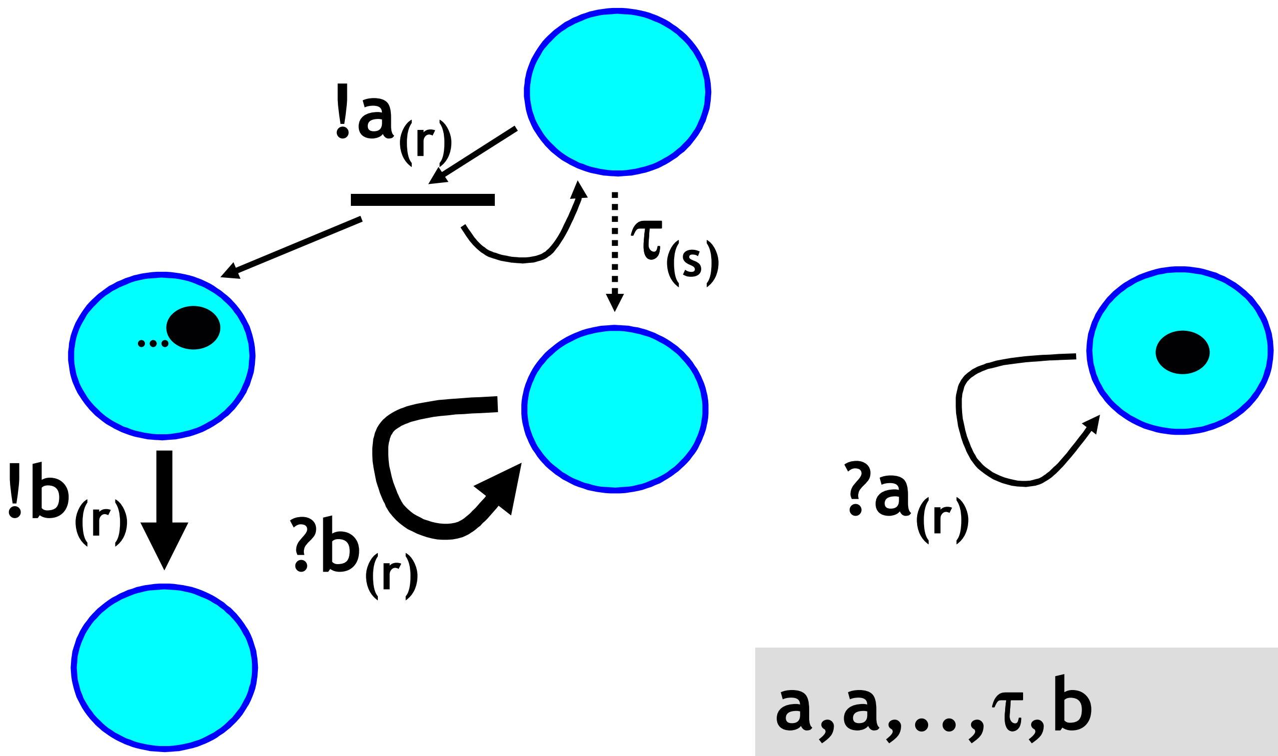
- Starting population: **A|A'**

Example 2



- Starting population: **A|A’**

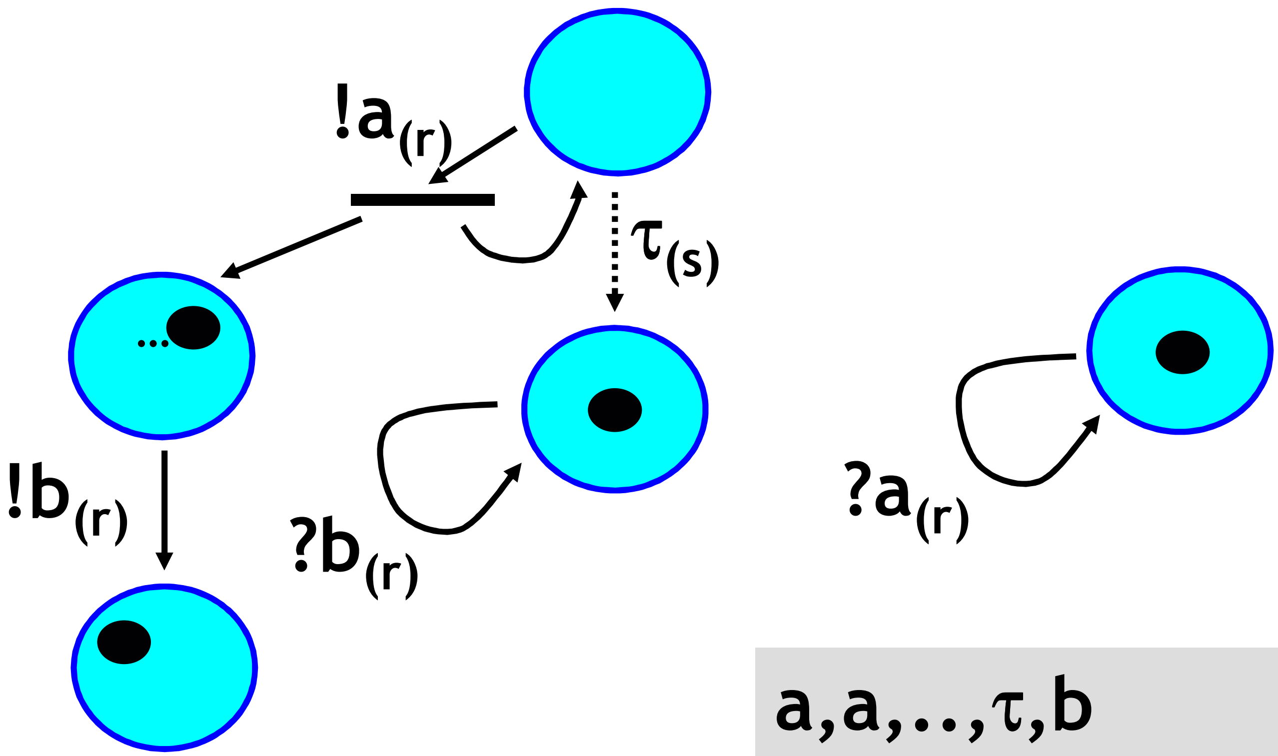
Example 2



a,a,..,τ,b

- Starting population: **A|A'**

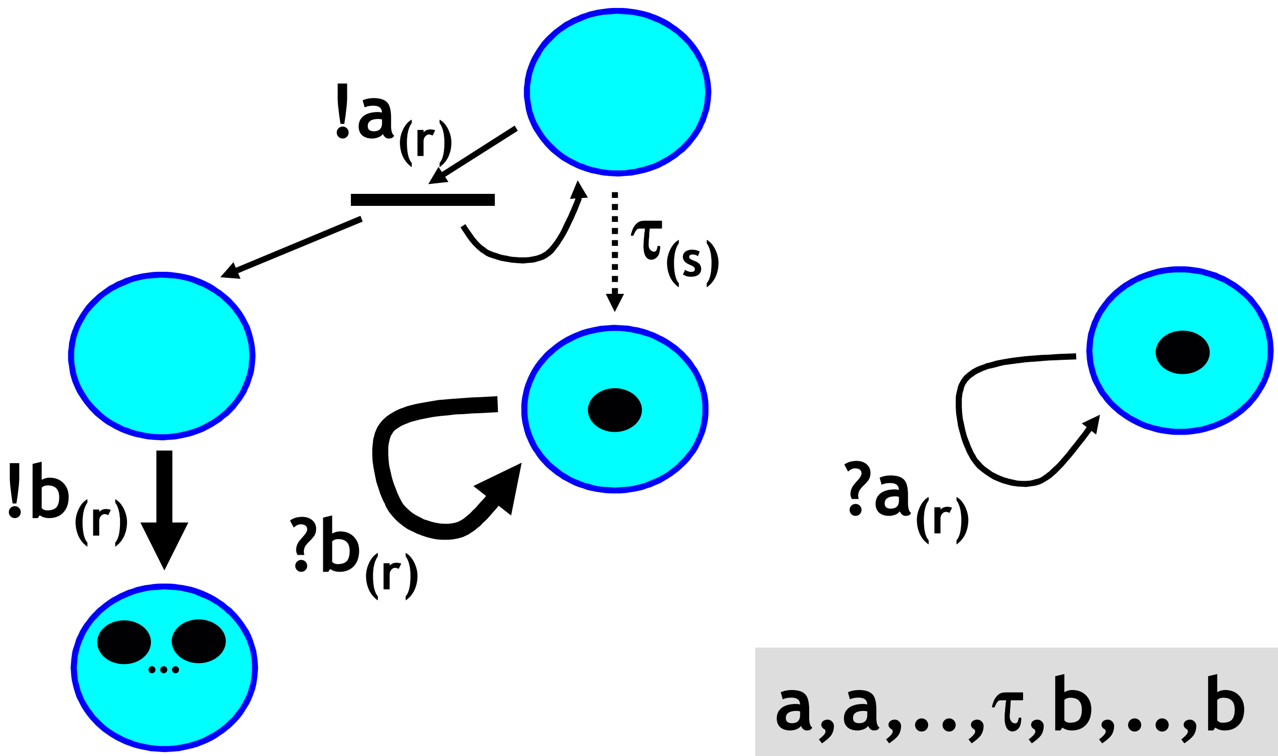
Example 2



a,a,..,τ,b

- Starting population: **A|A'**

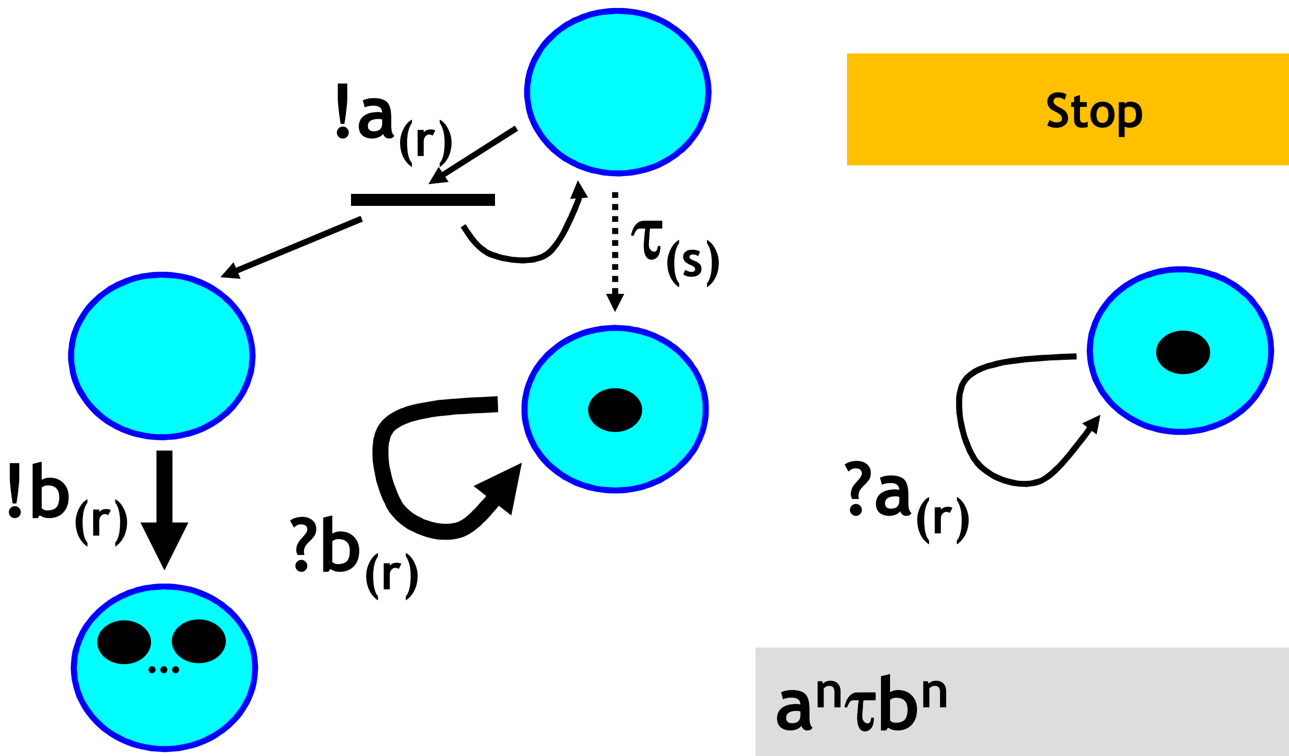
Example 2



a,a,..,τ,b,..,b

- Starting population: **A|A'**

Example 2



Stop

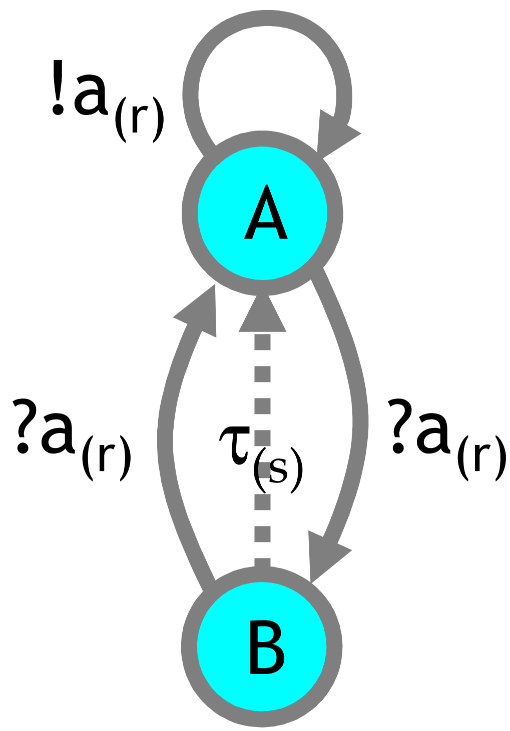
$a^n \tau b^n$

- Starting population: **A|A'**

Talk Outline

- Basic Chemistry and Basic Biochemistry
  - Biochemistry = Chemistry + complexation
- Chemical Ground Form (CGF) [TCS08]
  - A process algebra for basic chemistry
  - Connections with basic chemistry (FSRN)
  - Basic chemistry can't compute! [Sol08]
- Biochemical Ground Form (BGF) [AB08]
  - A process algebra for basic biochemistry
  - Basic biochemistry is Turing-complete.
- Conclusions
  - Basic biochemistry > Basic chemistry

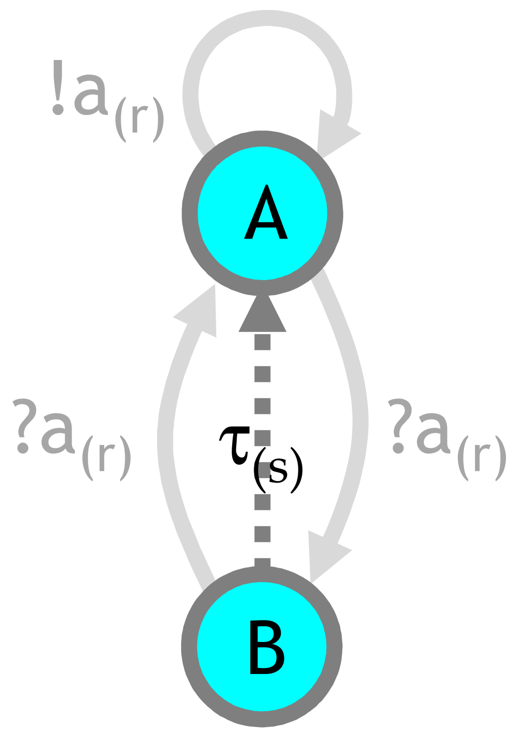
From CGF to FSRN (by example)

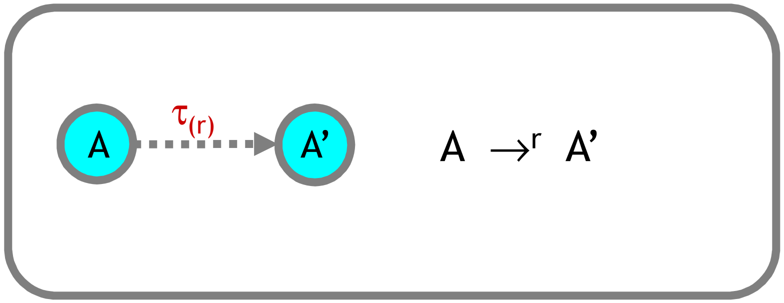


$A = !a_{(r)};A \oplus ?a_{(r)};B$

$B = ?a_{(r)};A \oplus \tau_{(s)};A$

From CGF to FSRN (by example)



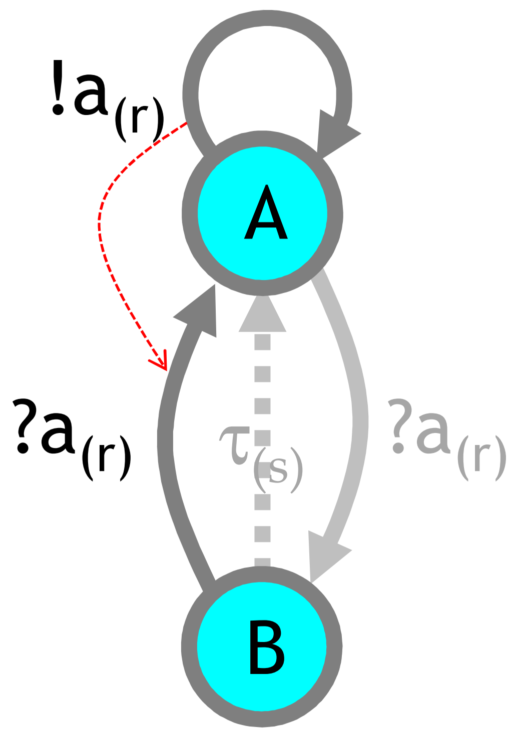


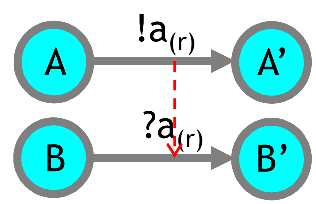
$B \rightarrow^{s} A$

$A = !a;A \oplus ?a;B$

$B = ?a;A \oplus \tau_{(s)};A$

From CGF to FSRN (by example)





$A+B \rightarrow^{r} A'+B'$

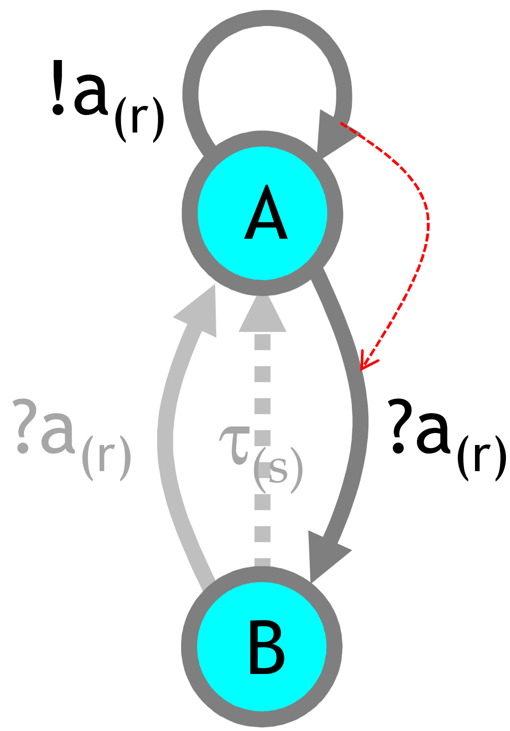
$B \rightarrow^{s} A$

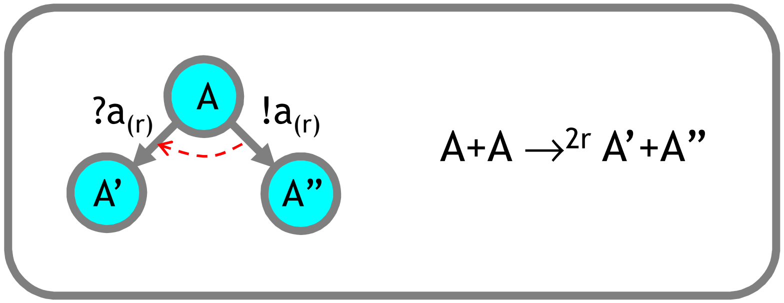
$A+B \rightarrow^{r} A+A$

$A = !a;A \oplus ?a;B$

$B = ?a;A \oplus \tau_{(s)};A$

From CGF to FSRN (by example)





$B \rightarrow^{s} A$

$A+B \rightarrow^{r} A+A$

$A+A \rightarrow^{2r} A+B$

$A = \,!a;A \oplus ?a;B$

$B = ?a;A \oplus \tau_{(s)};A$

From FSRN to CGF (by example)

$x{:}\quad B \rightarrow^{s} A$

$b{:}\quad A+B \rightarrow^{r} A+A$

$c{:}\quad A+A \rightarrow^{2r} A+B$

Unique reaction names

	$x_{(s)}$	$b_{(r)}$	$c_{(r)}$
A			
B			

Reactions names

Half-rate for homeo reactions

Species

From FSRN to CGF (by example)

$x\colon\ B \rightarrow^{s} A$

$b\colon\ A+B \rightarrow^{r} A+A$

$c\colon\ A+A \rightarrow^{2r} A+B$

	$x_{(s)}$	$b_{(r)}$	$c_{(r)}$
A			
B	$\tau;A$		

1: Fill the matrix by columns:

Degradation reaction $v_i\colon X \rightarrow_{k_i} P_i$

add $\tau;P_i$ to $\langle X,v_{ij}\rangle$.

From FSRN to CGF (by example)

x: $B \rightarrow^{s} A$

b: $A+B \rightarrow^{r} A+A$

c: $A+A \rightarrow^{2r} A+B$

	$x_{(s)}$	$b_{(r)}$	$c_{(r)}$
A		?;A|A	
B	τ;A	!;0	

1: Fill the matrix by columns:

Degradation reaction v_i: $X \rightarrow^{k_i} P_i$

add $\tau;P_i$ to $<X,v_{ij}>$.

Hetero reaction v_i: $X+Y \rightarrow^{k_i} P_i$

add $?;P_i$ to $<X,v_i>$ and $!;0$ to $<Y,v_i>$

From FSRN to CGF (by example)

x: $B \rightarrow^{s} A$

b: $A+B \rightarrow^{r} A+A$

c: $A+A \rightarrow^{2r} A+B$

	$x_{(s)}$	$b_{(r)}$	$c_{(r)}$
A		?;A|A	?;A|B !;0
B	τ;A	!;0	

1: Fill the matrix by columns:

Degradation reaction v_i: $X \rightarrow k_i\ P_i$

add $\tau;P_i$ to $\langle X,v_{ij}\rangle$.

Hetero reaction v_i: $X+Y \rightarrow k_i\ P_i$

add $?;P_i$ to $\langle X,v_i\rangle$ and $!;0$ to $\langle Y,v_i\rangle$

Homeo reaction v_i: $X+X \rightarrow k_i\ P_i$

add $?;P_i$ and $!;0$ to $\langle X,v_i\rangle$

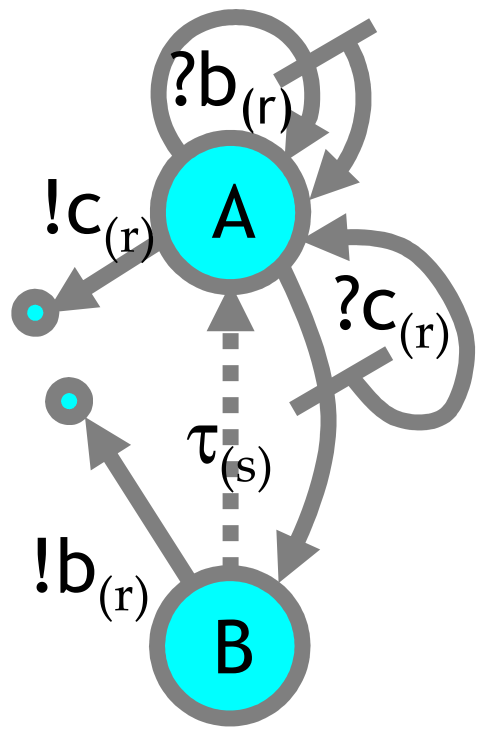
From FSRN to CGF (by example)

x: $B \rightarrow^{s} A$

b: $A+B \rightarrow^{r} A+A$

c: $A+A \rightarrow^{2r} A+B$

	$x_{(s)}$	$b_{(r)}$	$c_{(r)}$
A		?;A|A	?;A|B !;0
B	τ;A	!;0	



1: Fill the matrix by columns:

Degradation reaction v_i: $X \rightarrow k_i\ P_i$

add $\tau;P_i$ to $<X,v_{ij}>$.

Hetero reaction v_i: $X+Y \rightarrow k_i\ P_i$

add $?;P_i$ to $<X,v_i>$ and $!;0$ to $<Y,v_i>$

Homeo reaction v_i: $X+X \rightarrow k_i\ P_i$

add $?;P_i$ and $!;0$ to $<X,v_i>$

2: Read the result by rows:

$A = ?b_{(r)};(A|A) \oplus ?c_{(r)};(A|B) \oplus !c_{(r)};0$

$B = \tau_{(s)};A \oplus !b_{(r)};0$

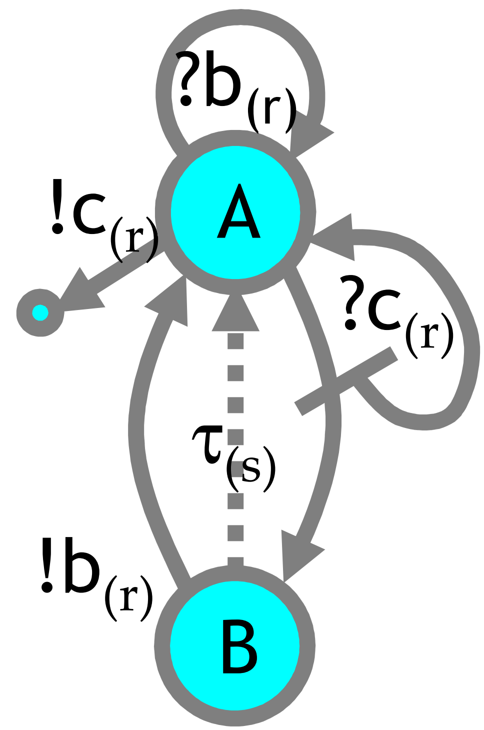
From FSRN to CGF (by example)

x: $B \rightarrow^{s} A$

b: $A+B \rightarrow^{r} A+A$

c: $A+A \rightarrow^{2r} A+B$

	$x_{(s)}$	$b_{(r)}$	$c_{(r)}$
A		?;A	?;A|B !;0
B	τ;A	!;A	



1: Fill the matrix by columns:

Degradation reaction v_i: $X \rightarrow^{k_i} P_i$

add $\tau;P_i$ to $<X,v_{ij}>$.

Hetero reaction v_i: $X+Y \rightarrow^{k_i} P_i$

add $?;P_i$ to $<X,v_i>$ and $!;0$ to $<Y,v_i>$

Homeo reaction v_i: $X+X \rightarrow^{k_i} P_i$

add $?;P_i$ and $!;0$ to $<X,v_i>$

2: Read the result by rows:

$A = ?b_{(r)};A \oplus ?c_{(r)};(A|B) \oplus !c_{(r)};0$

$B = \tau_{(s)};A \oplus !b_{(r)};A$

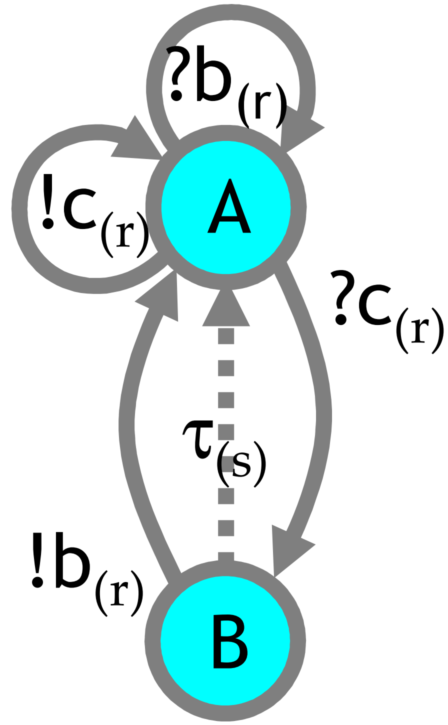
From FSRN to CGF (by example)

x: $B \rightarrow^{s} A$

b: $A+B \rightarrow^{r} A+A$

c: $A+A \rightarrow^{2r} A+B$

	$x_{(s)}$	$b_{(r)}$	$c_{(r)}$
A		?;A	?;B !;A
B	τ;A	!;A	



1: Fill the matrix by columns:

Degradation reaction v_i: $X \rightarrow_{k_i} P_i$

add $\tau;P_i$ to $<X,v_{ij}>$.

Hetero reaction v_i: $X+Y \rightarrow_{k_i} P_i$

add $?;P_i$ to $<X,v_i>$ and $!;0$ to $<Y,v_i>$

Homeo reaction v_i: $X+X \rightarrow_{k_i} P_i$

add $?;P_i$ and $!;0$ to $<X,v_i>$

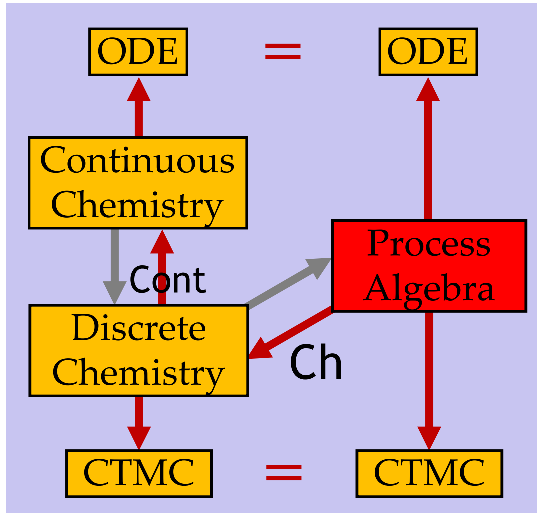
2: Read the result by rows:

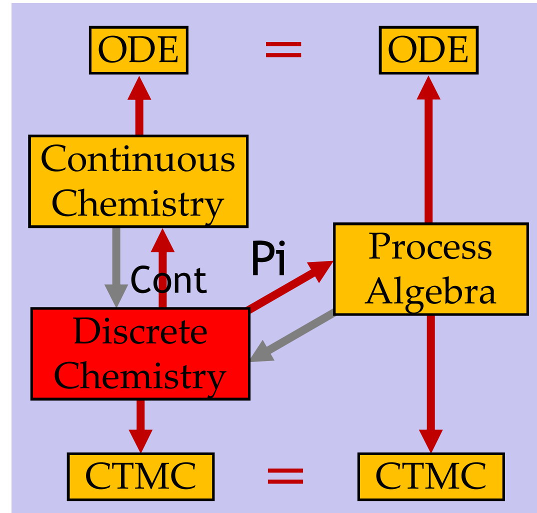
$A = ?b_{(r)};A \oplus ?c_{(r)};B \oplus !c_{(r)};A$

$B = \tau_{(s)};A \oplus !b_{(r)};A$

CGF is Equivalent to FSNR [TCS08]

- Def: ≋ is equivalent CTMC's (isomorphic graphs with same rates).
- Thm: E ≋ Ch(E)
- Thm: C ≋ Pi(C)





- Def: $\approx$ is equivalence of polynomials over the field of reals.
- Thm: E $\approx$ Cont(Ch(E))
- Thm: Cont(C) $\approx$ Pi(C)

Talk Outline

- Basic Chemistry and Basic Biochemistry
  - Biochemistry = Chemistry + complexation
- Chemical Ground Form (CGF) [TCS08]
  - A process algebra for basic chemistry
  - Connections with basic chemistry (FSRN)
  - Basic chemistry can’t compute! [Sol08]
- Biochemical Ground Form (BGF) [AB08]
  - A process algebra for basic biochemistry
  - Basic biochemistry is Turing-complete.
- Conclusions
  - Basic biochemistry > Basic chemistry

But it's all just Petri Nets!

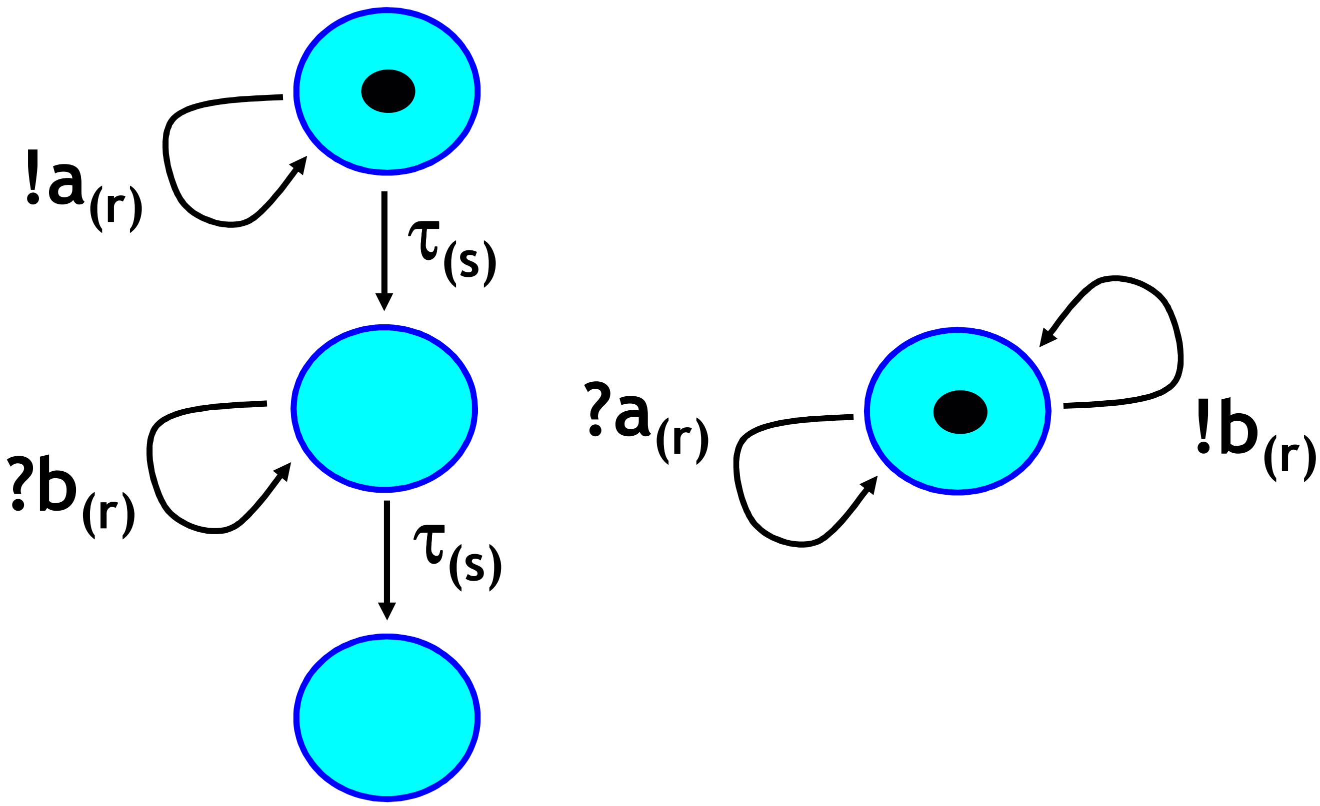
- It is possible to translate an arbitrary CGF (or FSRN) into a Place/Transition Petri Net.
  - Ignoring rates, and of course losing compositionality.

- Pretty much everything is decidable in P/T Nets.
  - In particular, reachability of a dead ("halting") state.

- Hence both CGF and FSRN are not Turing-complete!
  - Basic chemistry can't compute!
    (Soloveichik et. al., Natural Computing 2008)
  - Even though stochastic chemistry is extremely rich,
    e.g. it includes chaotic systems.

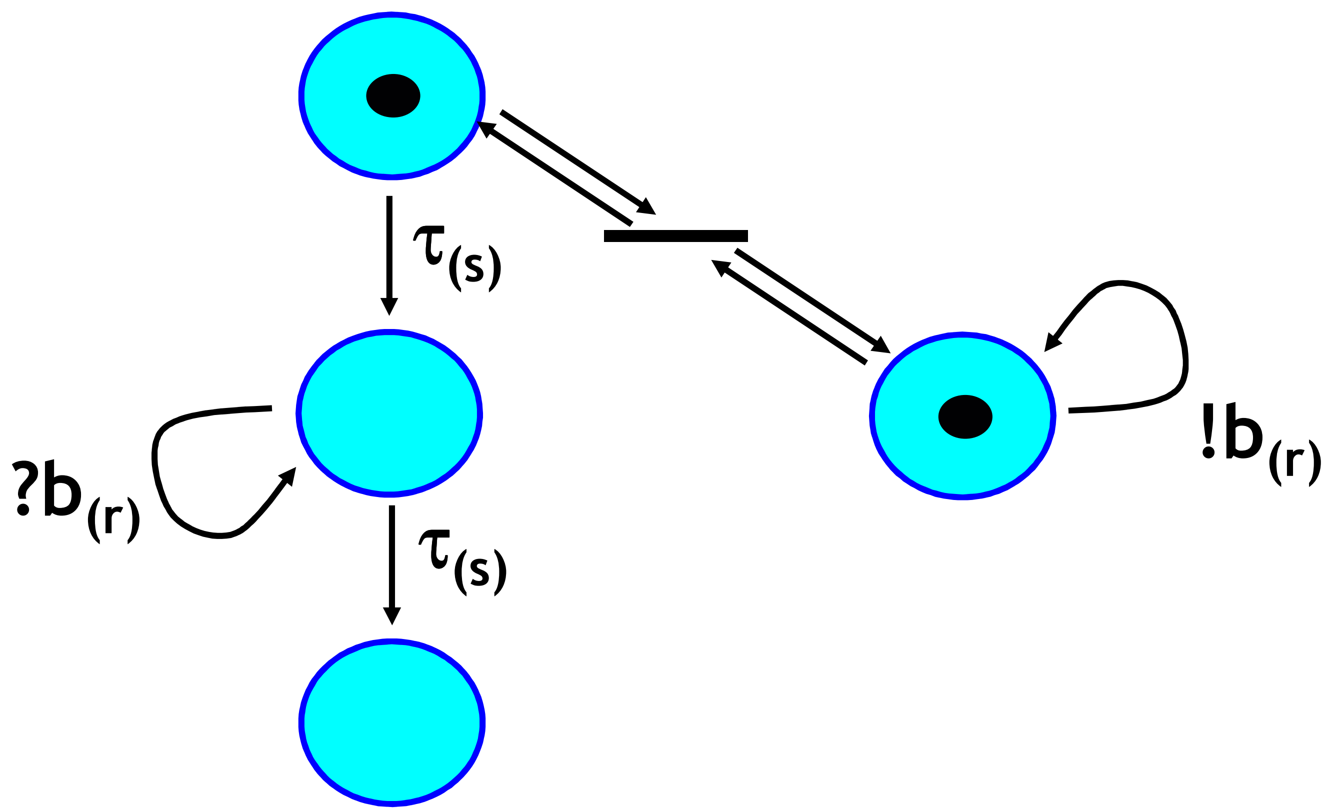
A Petri net semantics for CGF

- One place for each Species
- One transition for each reaction



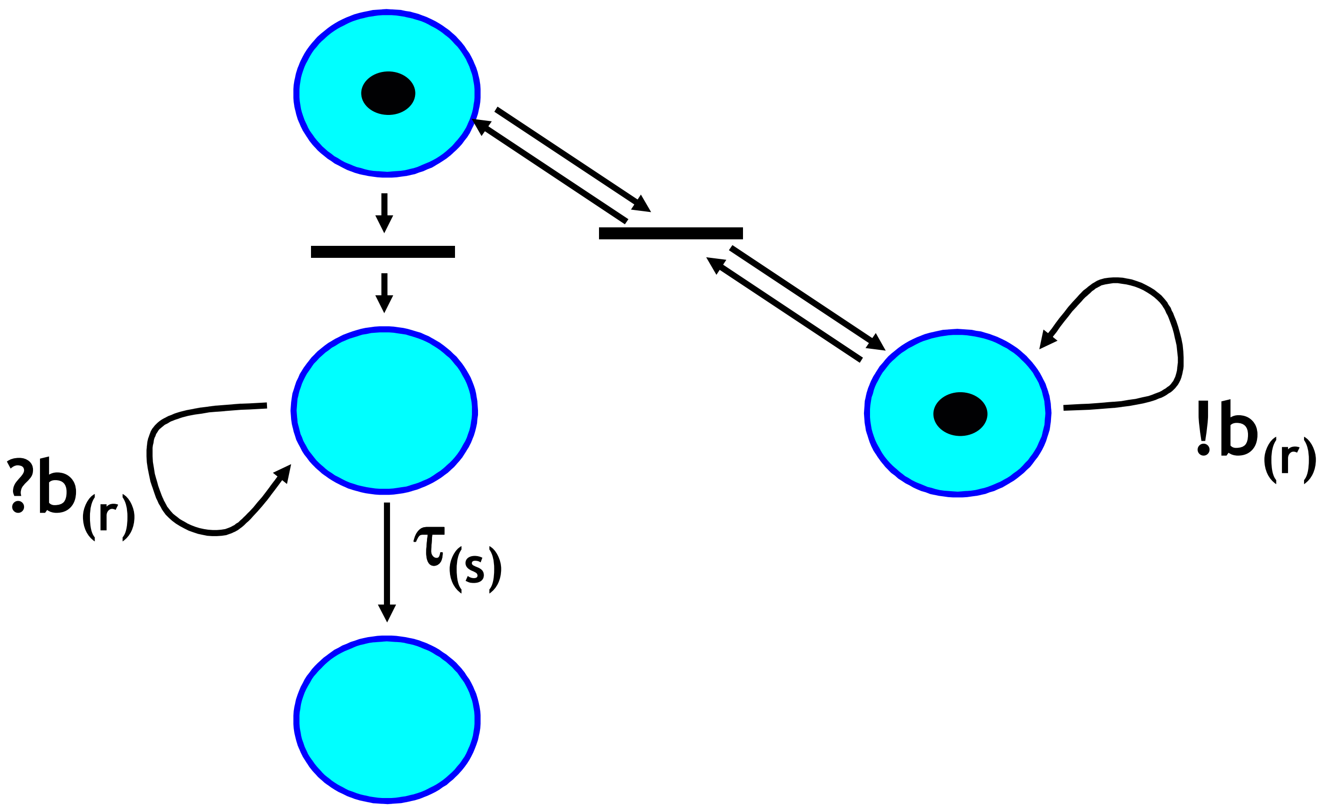
A Petri net semantics for CGF

- One place for each Species
- One transition for each reaction



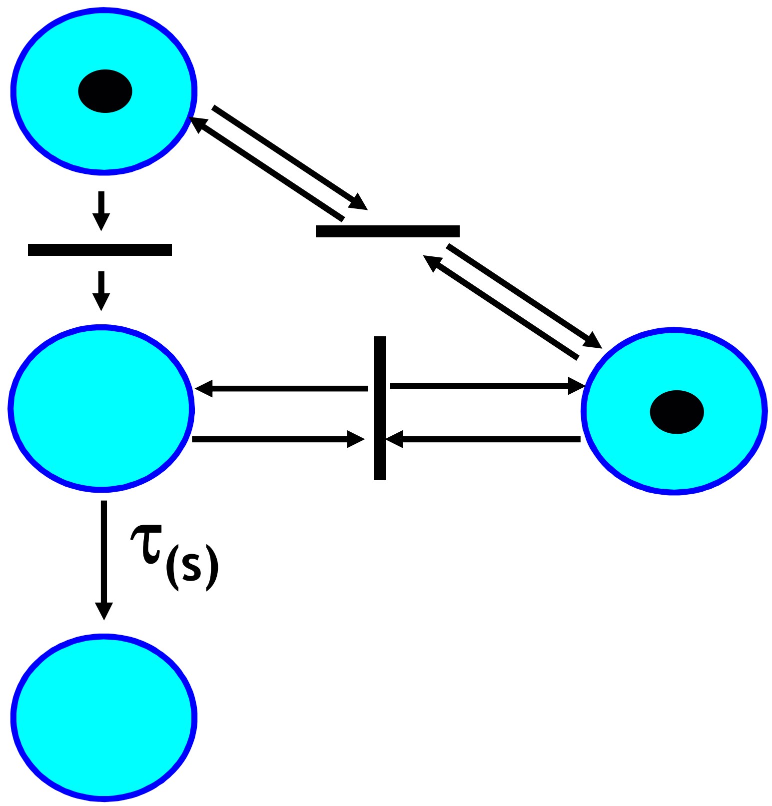
A Petri net semantics for CGF

- One place for each Species
- One transition for each reaction



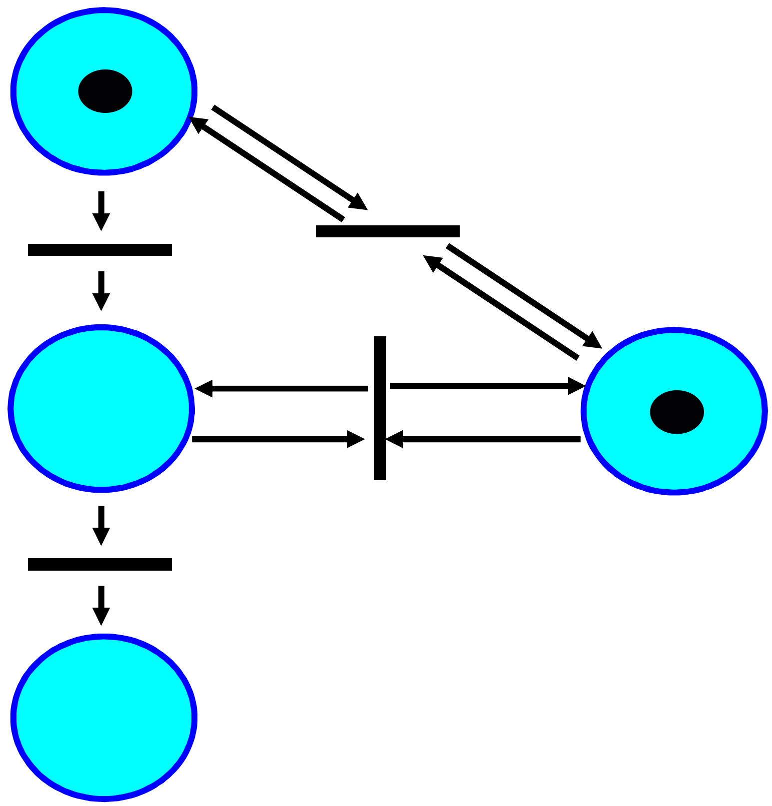
A Petri net semantics for CGF

- One place for each Species
- One transition for each reaction



A Petri net semantics for CGF

- One place for each Species
- One transition for each reaction



Talk Outline

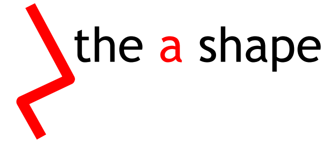
- Basic Chemistry and Basic Biochemistry
  - Biochemistry = Chemistry + complexation
- Chemical Ground Form (CGF) [TCS08]
  - A process algebra for basic chemistry
  - Connections with basic chemistry (FSRN)
  - Basic chemistry can't compute! [Sol08]
- Biochemical Ground Form (BGF) [AB08]
  - A process algebra for basic biochemistry
  - Basic biochemistry is Turing-complete.
- Conclusions
  - Basic biochemistry > Basic chemistry

"Turifying" Chemistry

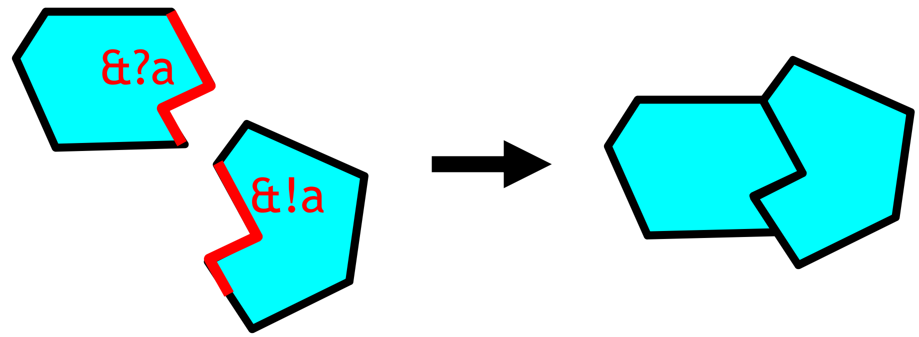
- What can we add to basic chemistry to make it Turing-complete?
- Lots of stuff
  - E.g. we can go from CGF to full π-calculus
- But is there…
  - A *basic* mechanism
  - which is also biologically *realistic*?

Association and Dissociation in BGF

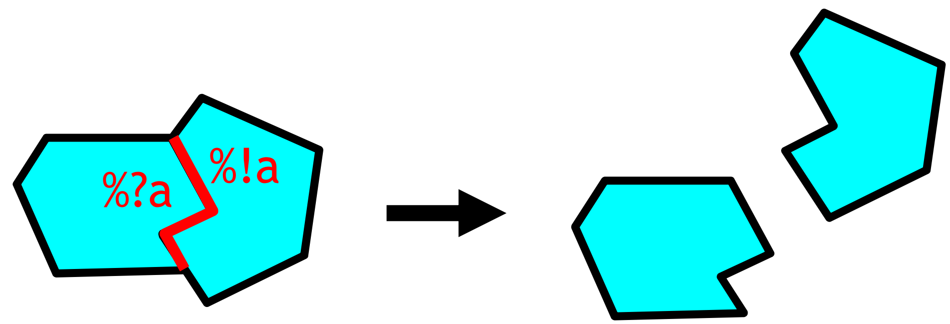
- Association patches are named



- & – association
  - &?a associate
  - &!a co-associate



- % – dissociation
  - %?a dissociate
  - %!a co-dossociate



- A given patch can *hold* only one association at a time
- Two molecules can dissociate only if *they* are associated

Example: Linear Polymerization

$SF = \tau_{(rs)}; S|SF$
$MF = \tau_{(rm)}; M|MF$

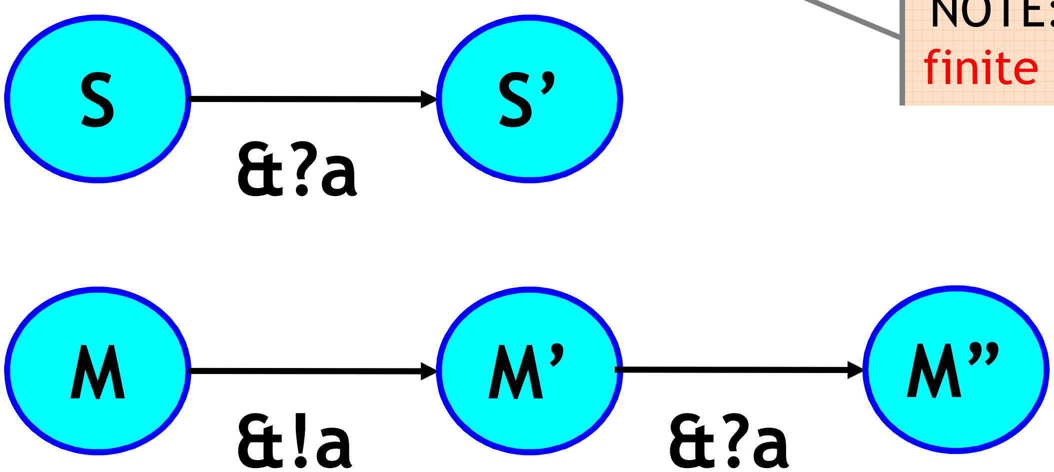
Seed factory and Monomer factory

S = &?a; S'
M = &!a; M'
M' = &?a; M"

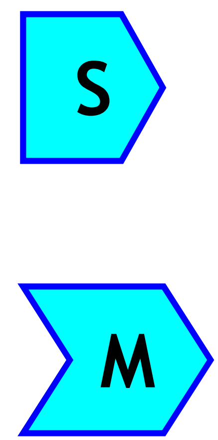
S' = ...
M" = ...

Any further behavior

NOTE: this is a finite program!

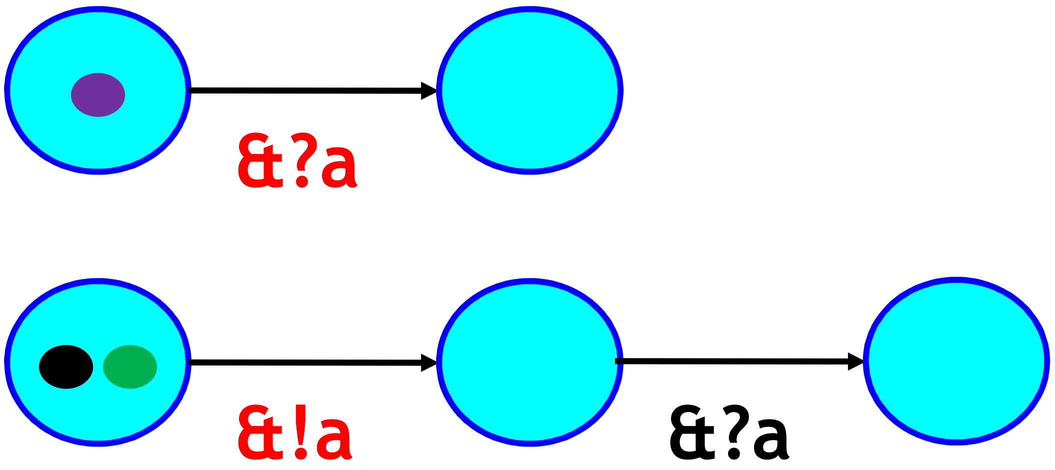


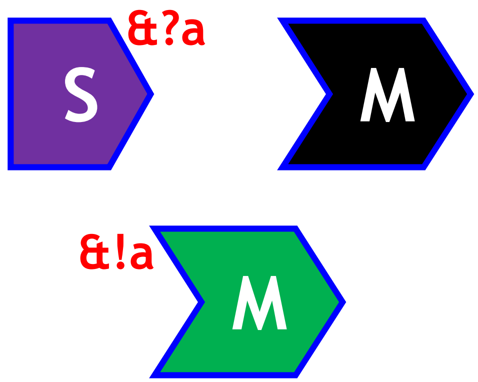
Seed



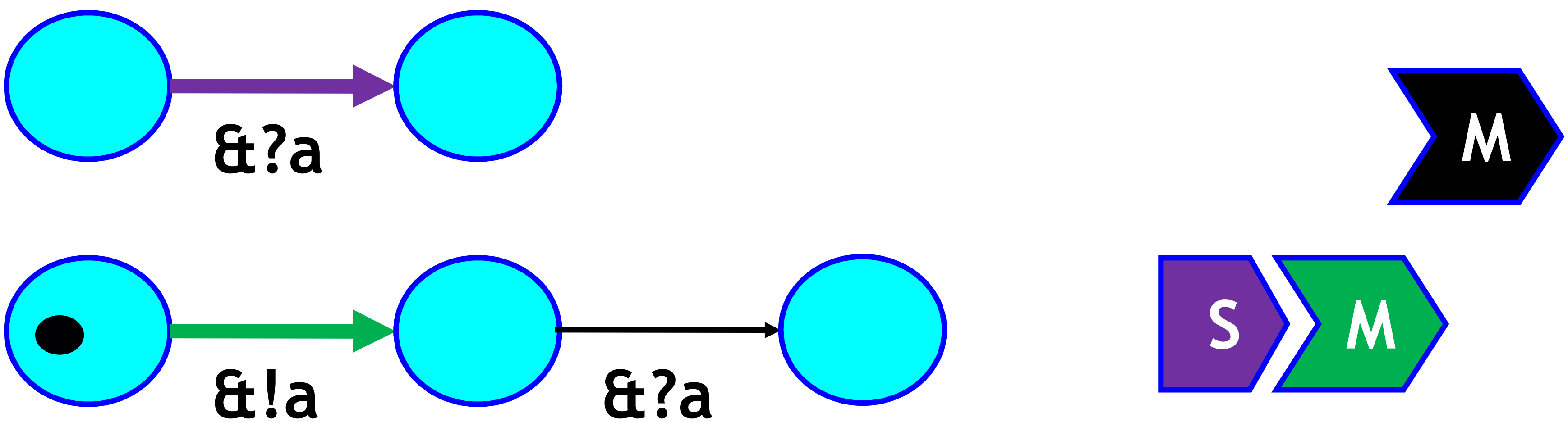
Monomer

Example: Linear Polymerization



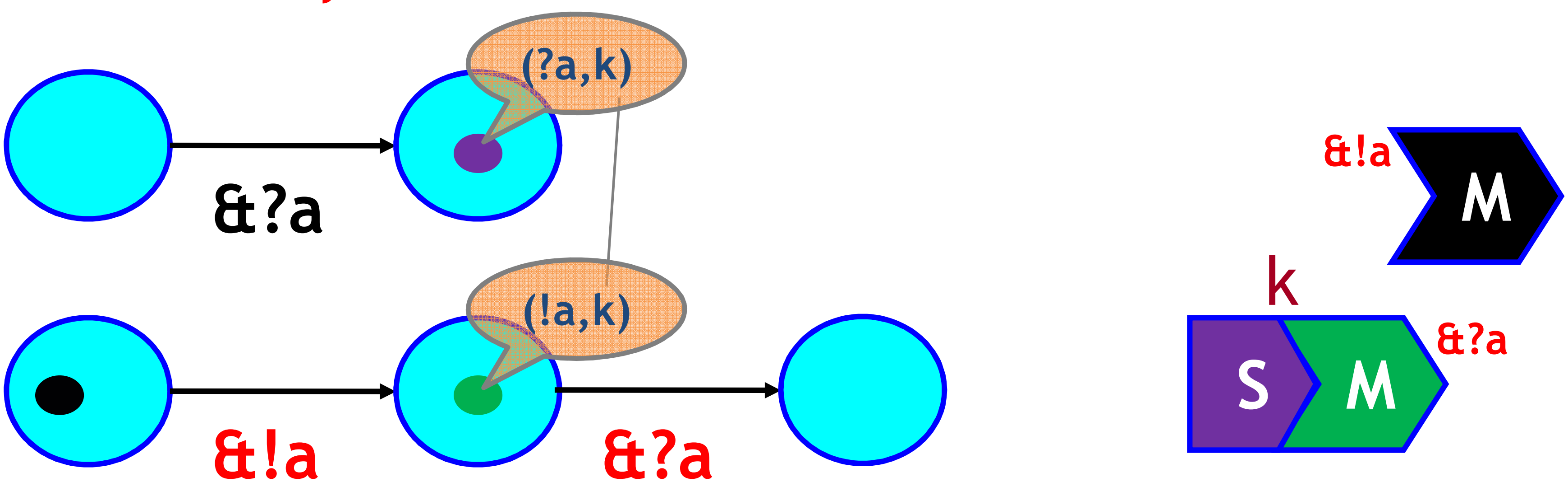


Example: Linear Polymerization

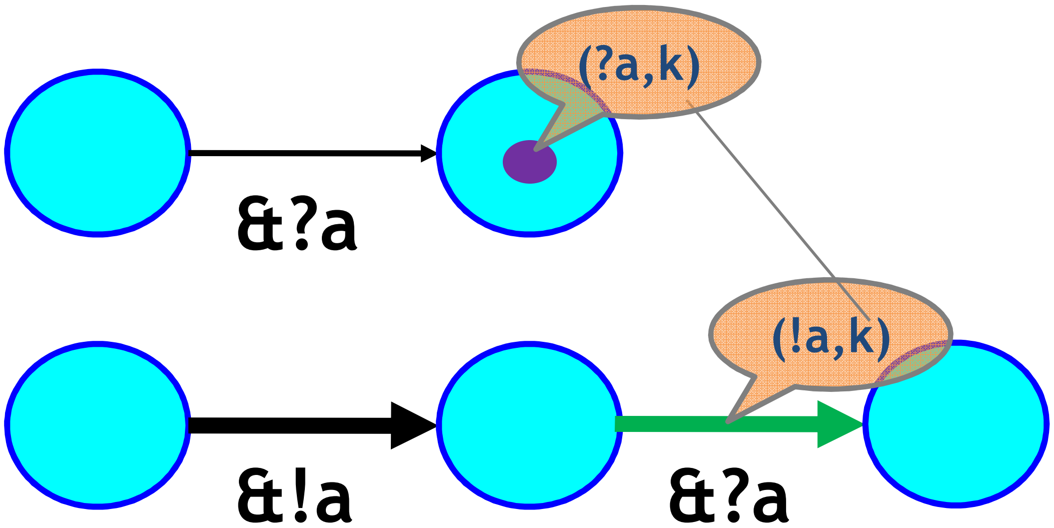


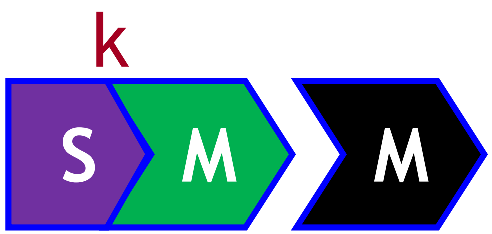
Example: Linear Polymerization

- Each association has a unique key
- Keys are stored in the molecule's association history

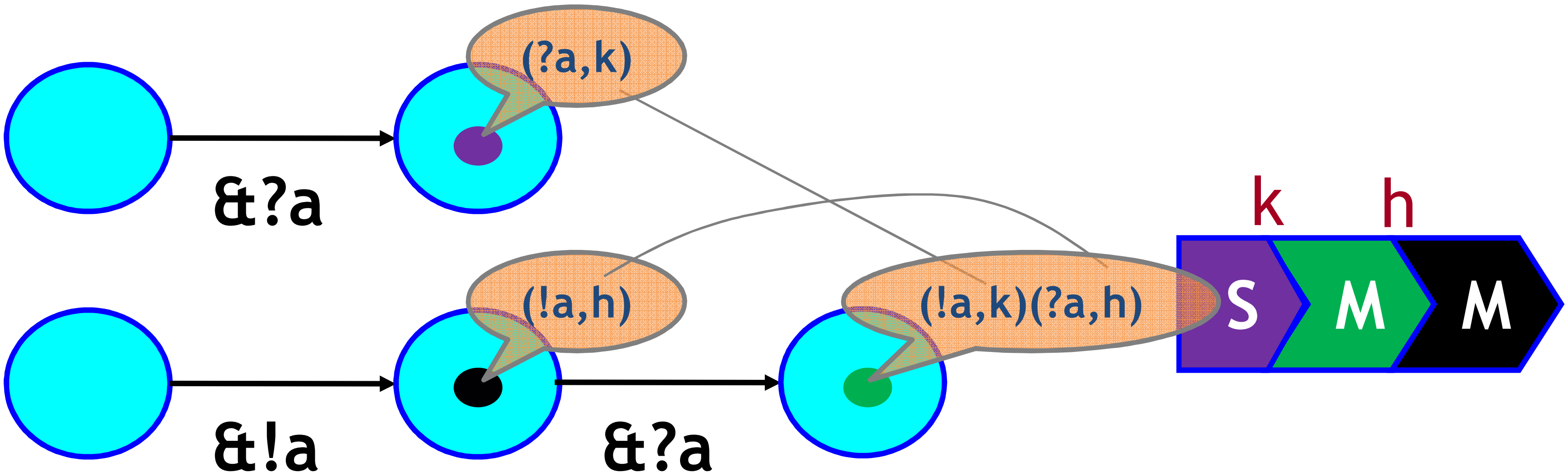


Example: Linear Polymerization



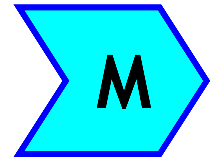


Example: Linear Polymerization

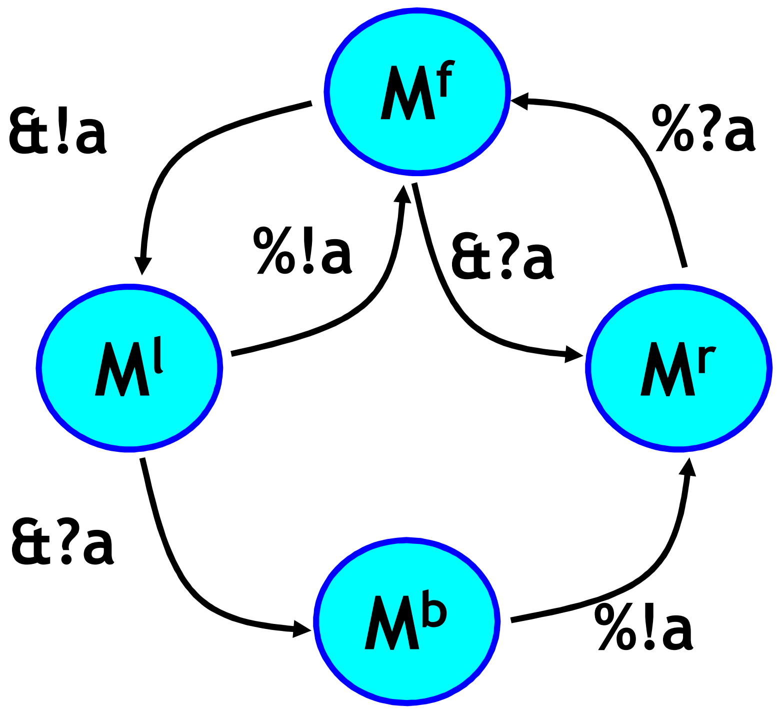


Example: Actin Polymerization

Grows only to the right, shrinks only from the left



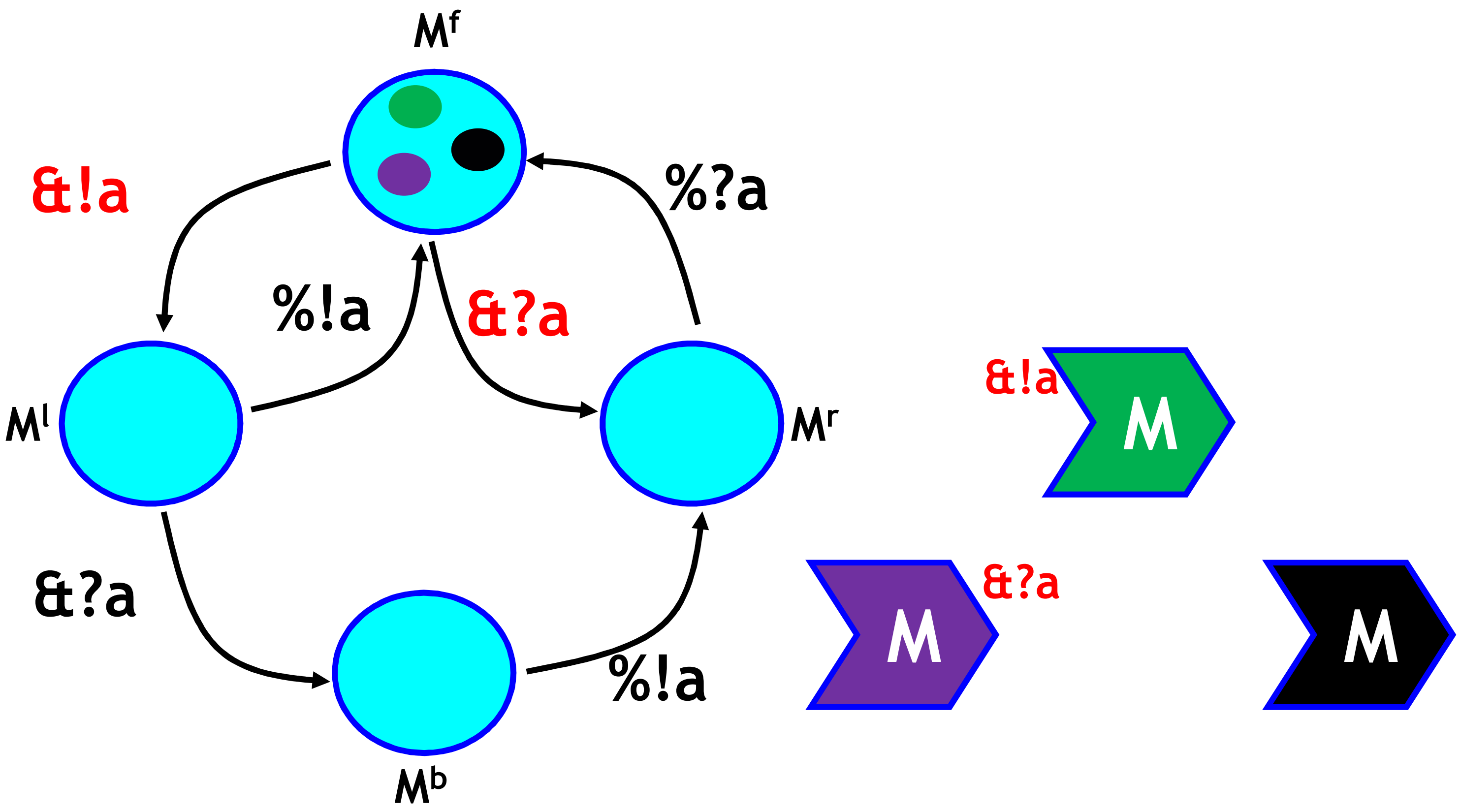
M^f = free on both sides
M^l = bound on the left
M^r = bound on the right
M^b = bound on both sides



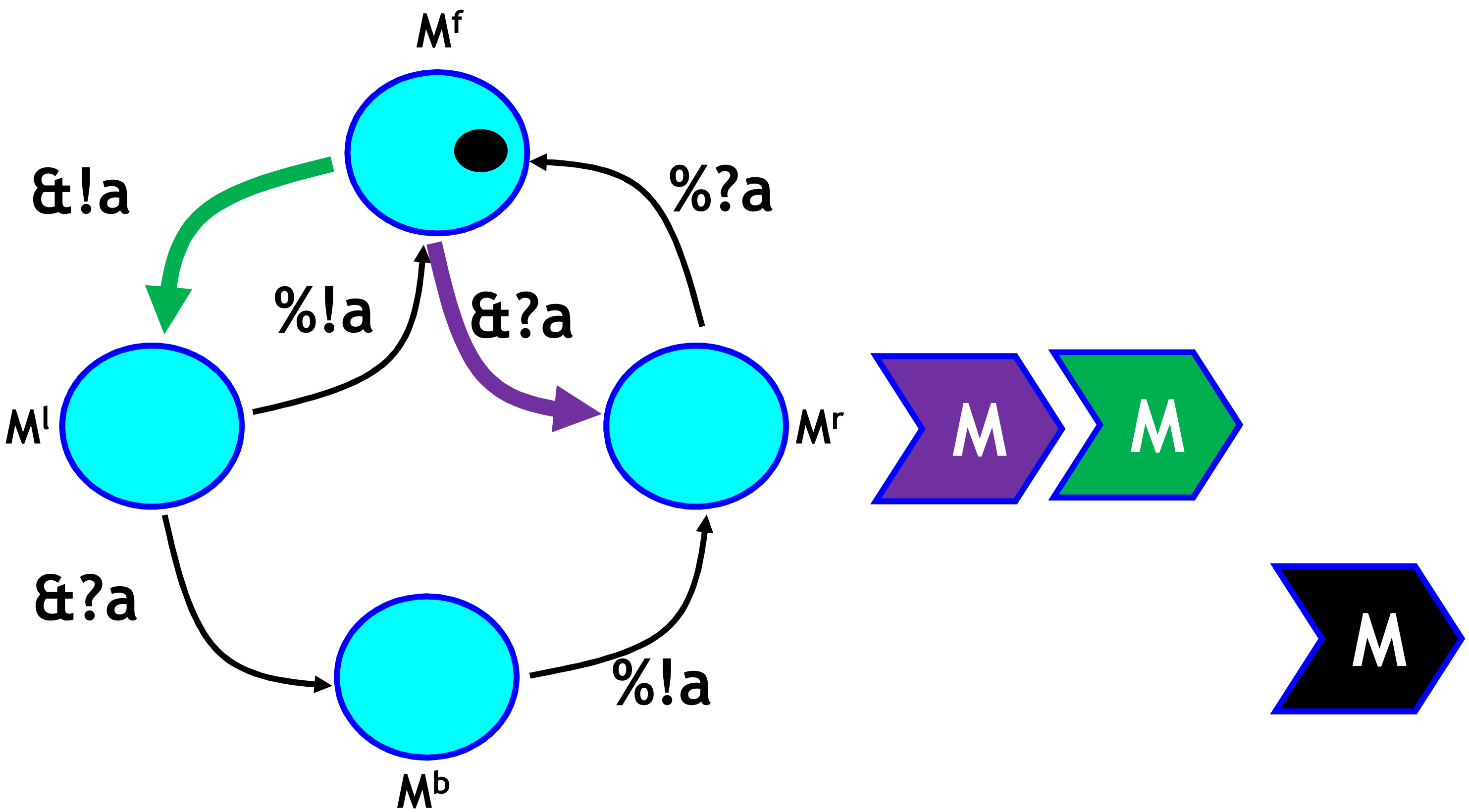
$$M^f = \&!a;\ M^l \oplus \&?a;\ M^r$$
$$M^l = \%!a;\ M^f \oplus \&?a;\ M^b$$
$$M^r = \%?a;\ M^f$$
$$M^b = \%!a;\ M^r$$

Example: Actin Polymerization

- Purple associates with green

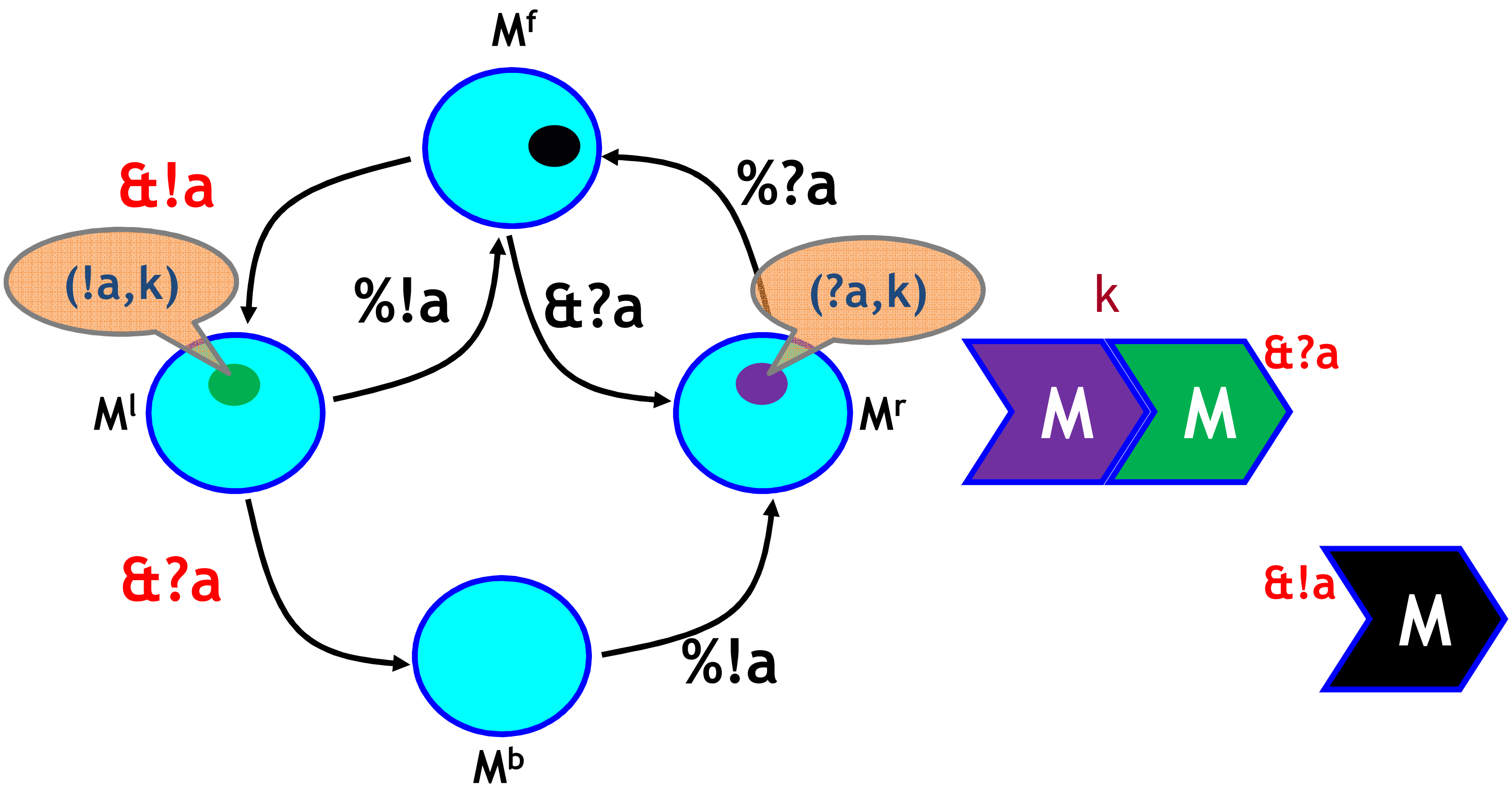


Example: Actin Polymerization



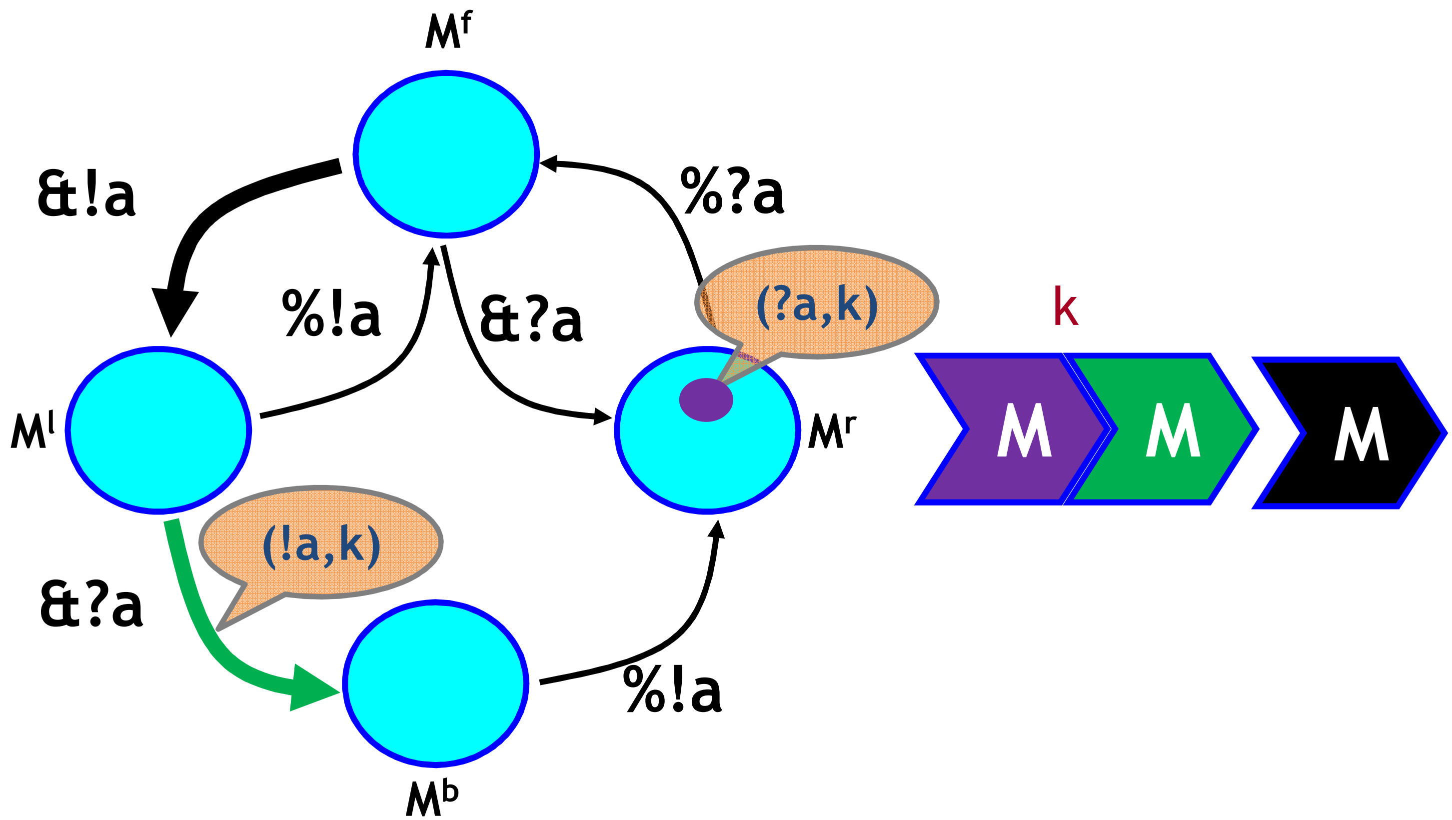
Example: Actin Polymerization

- Each association has a unique key
  Keys are stored in the molecule's history
- Black cannot associate with purple
  No complementary actions available,
  enforcing the "grow only to the right" constraint



Example: Actin Polymerization

- Green associates with black

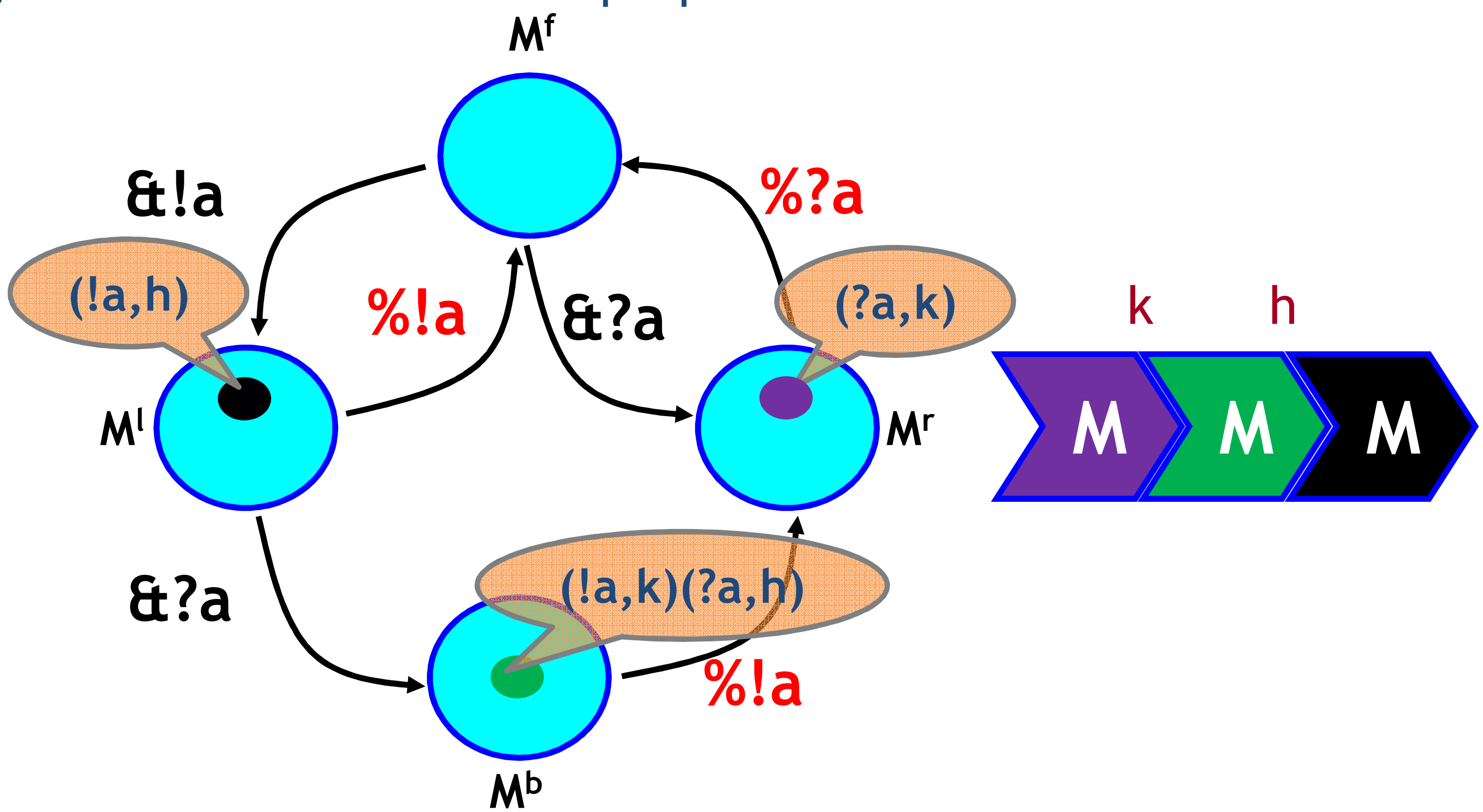


Example: Actin Polymerization

- Black cannot dissociate from green

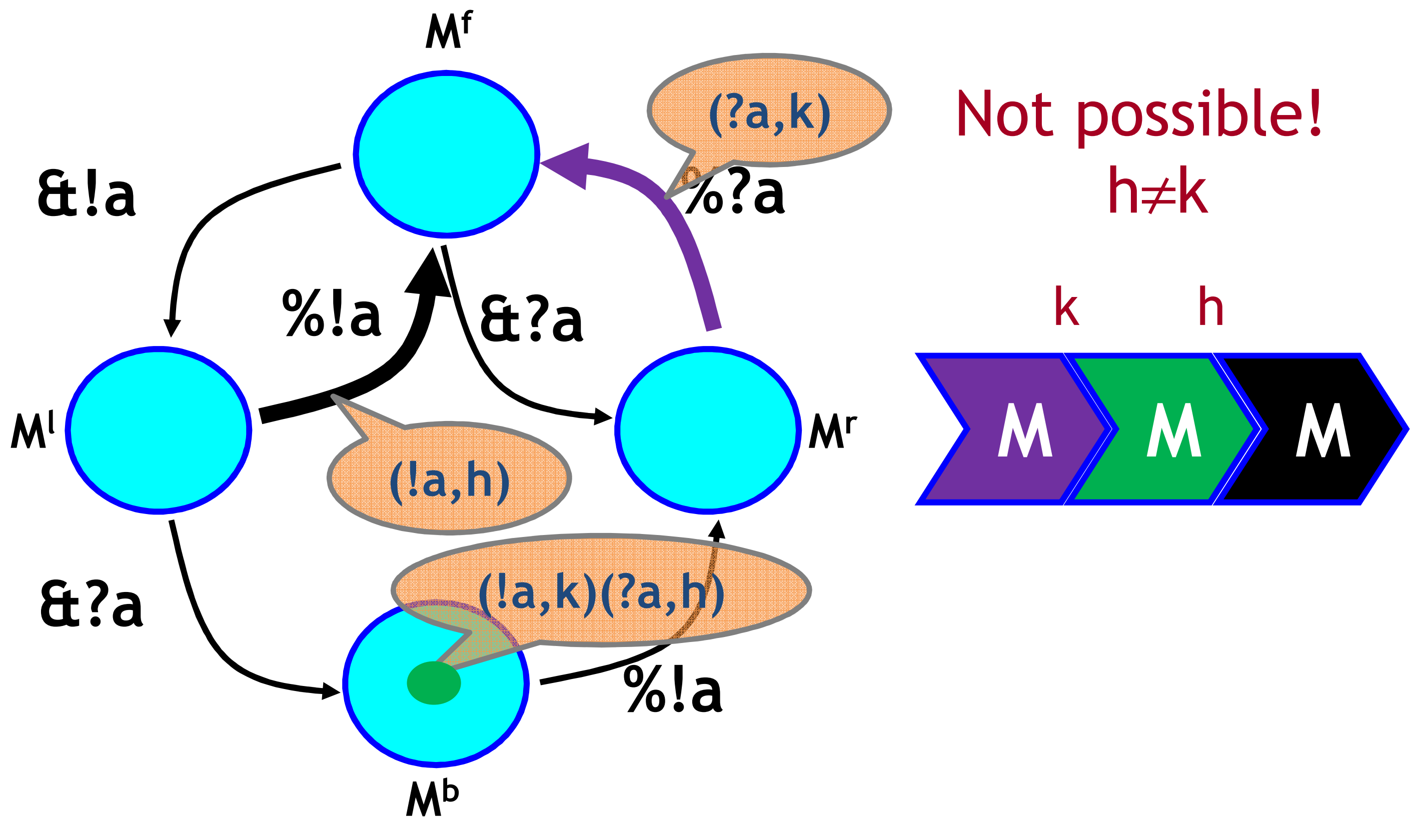
  No complementary actions available,

  enforcing the “shrink only from left” constraint
- But black can dissociate from purple (really?)
- And green can dissociate from purple



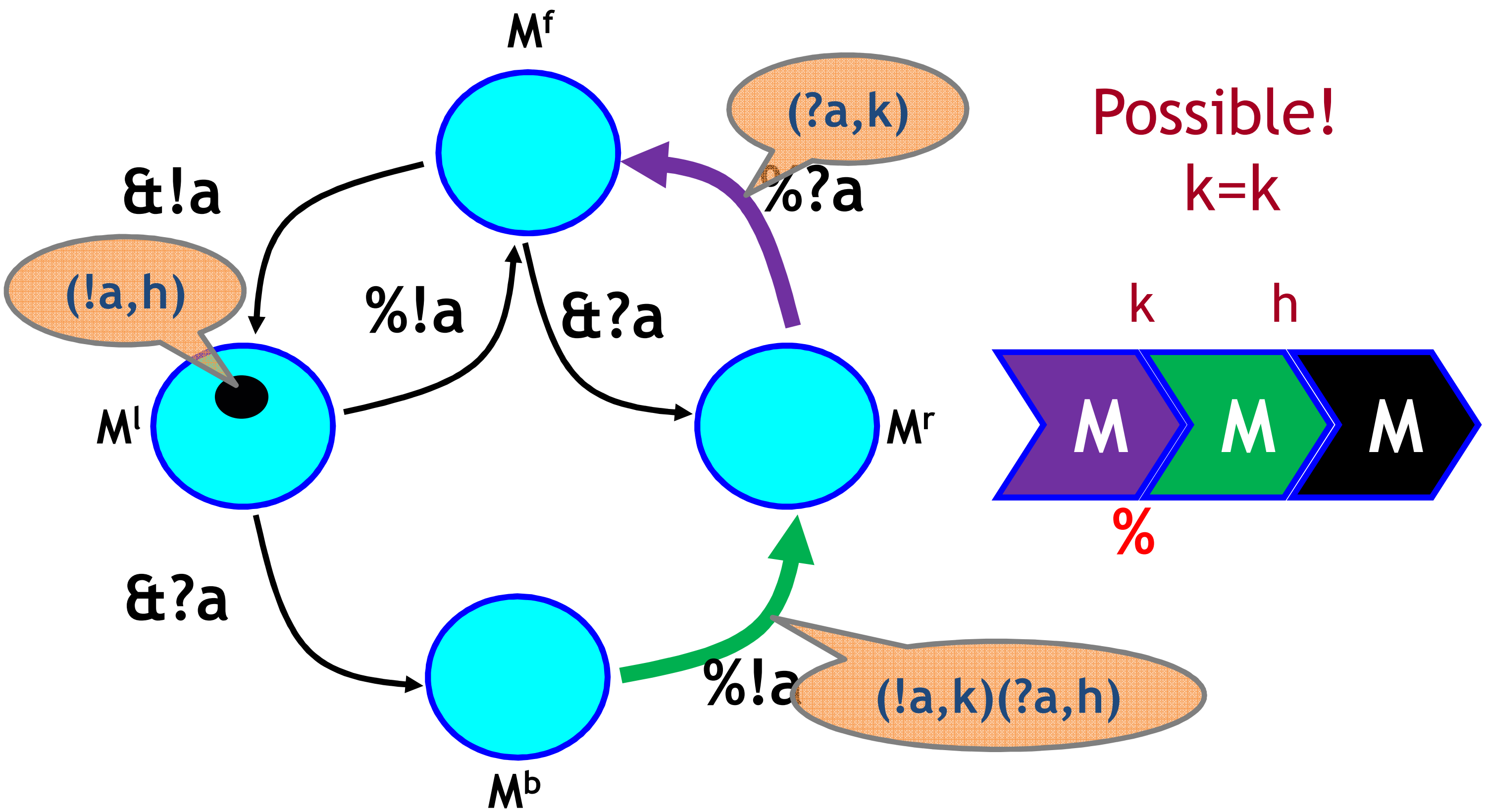
Example: Actin Polymerization

- No, black cannot dissociate from purple
  - The association history prevents it



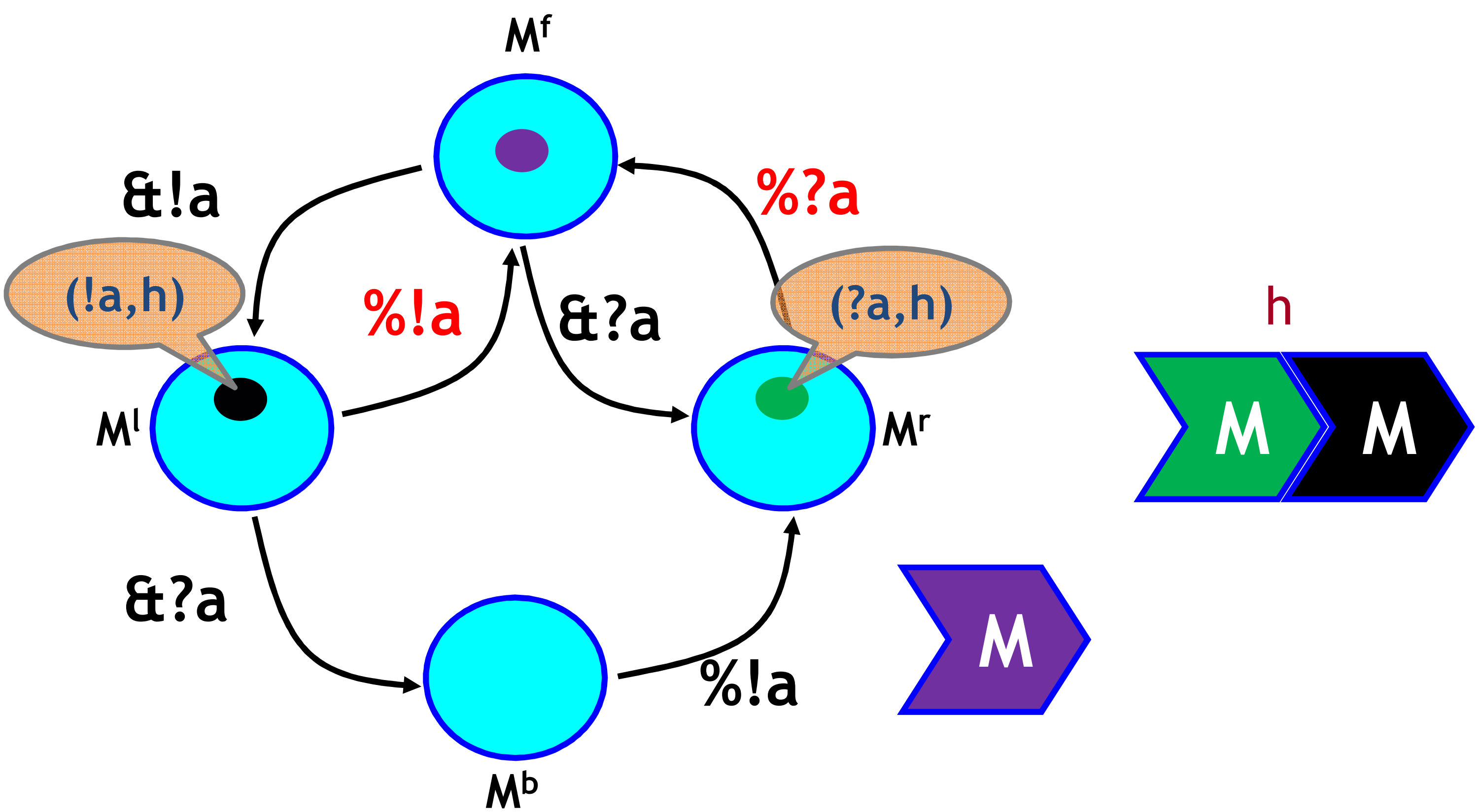
Example: Actin Polymerization

- Purple dissociates from green



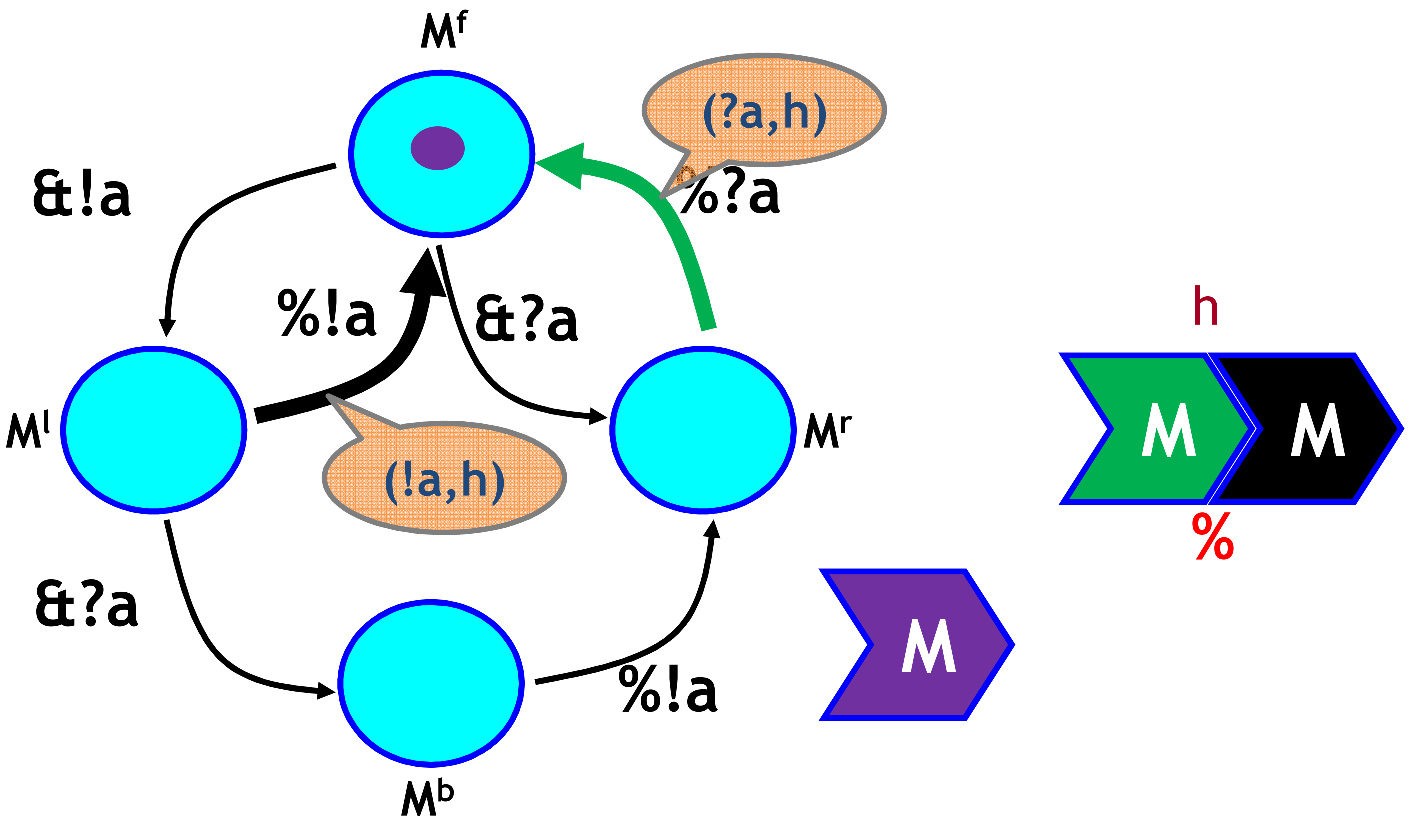
Example: Actin Polymerization

- Now purple could reassociate to black
  on the other side, but we are not going to do that



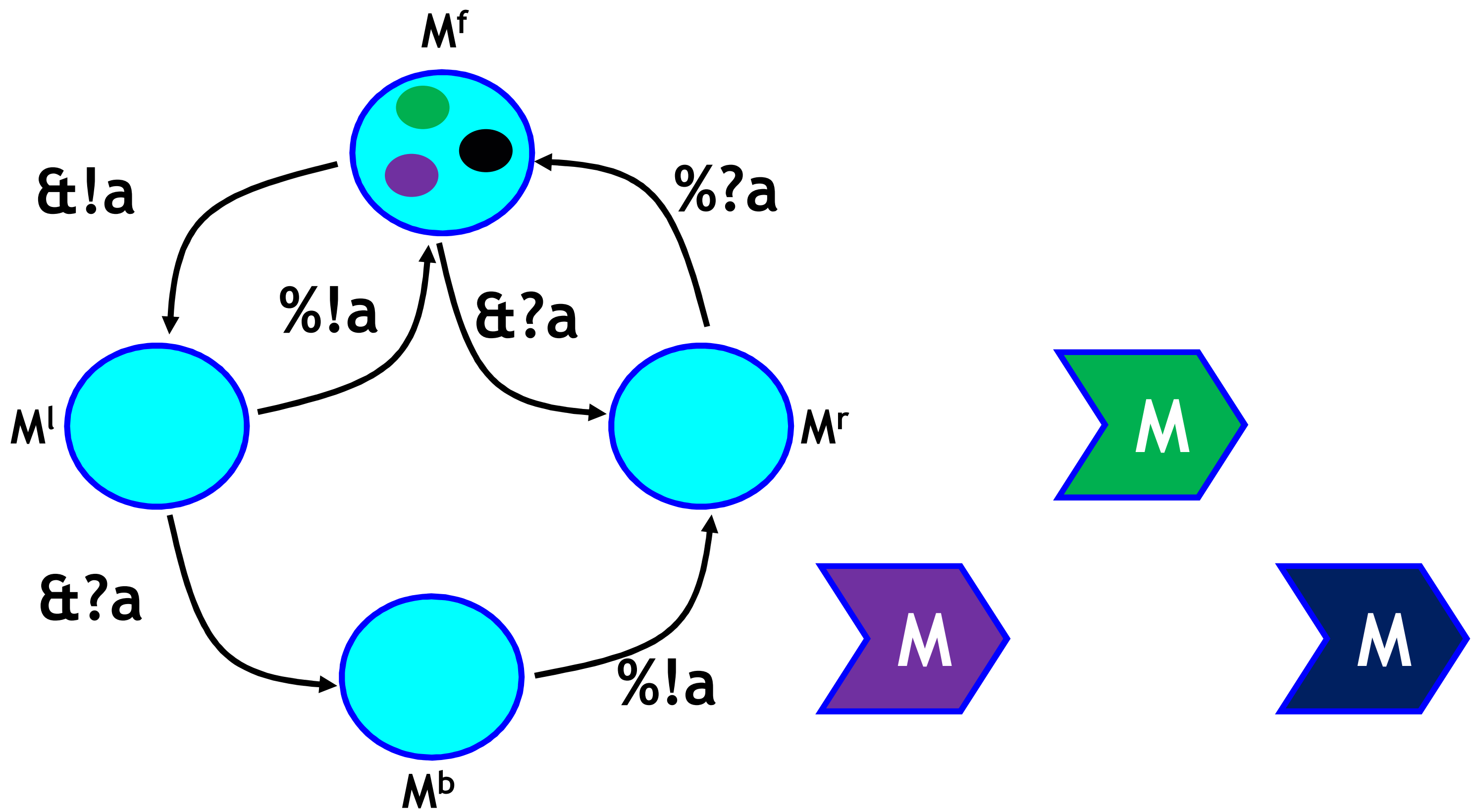
Example: Actin Polymerization

- Green dissociates from black



Example: Actin Polymerization

- Ready to start again



Talk Outline

- Basic Chemistry and Basic Biochemistry
  - Biochemistry = Chemistry + complexation
- Chemical Ground Form (CGF) [TCS08]
  - A process algebra for basic chemistry
  - Connections with basic chemistry (FSRN)
  - Basic chemistry can't compute! [Sol08]
- Biochemical Ground Form (BGF) [AB08]
  - A process algebra for basic biochemistry
  - Basic biochemistry is Turing-complete.
- Conclusions
  - Basic biochemistry > Basic chemistry

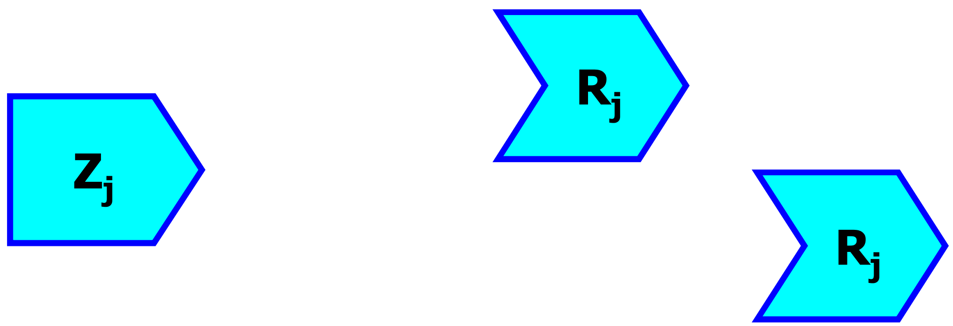
Turing completeness of BGF

- Random Access Machines: [Min67]
  - **Registers:** $r_1 \dots r_n$ hold natural numbers (unbounded)
  - **Program**: finite sequence of numbered instructions
    - **i: Inc(r_j)**: add 1 to the content of r_j and go to the next instruction
    - **i: DecJump(r_j,s)**: if the content of r_j is not 0 then decrease by 1 and go to the next instruction; otherwise jump to instruction s

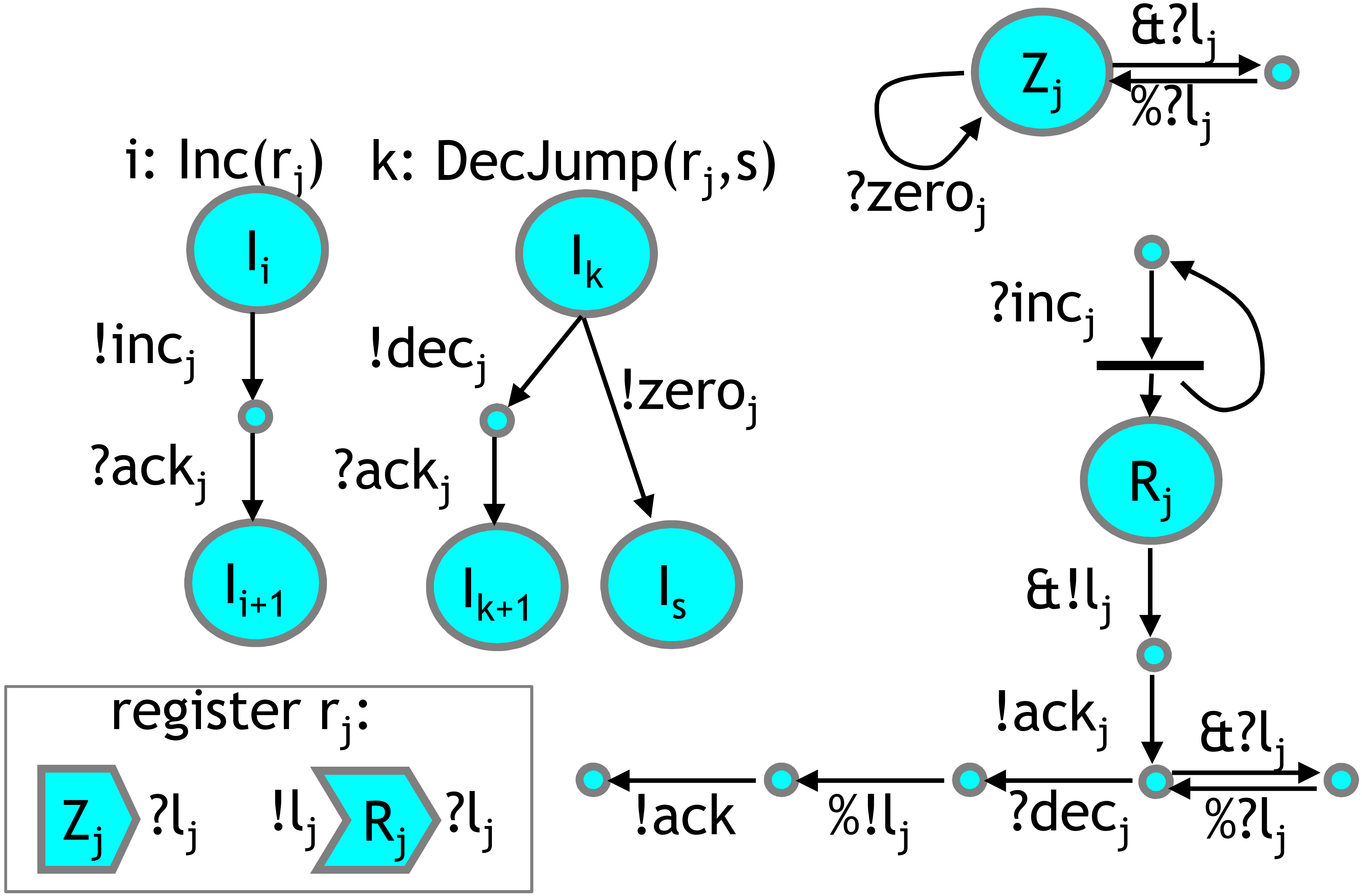
- There is a RAM encoding in BGF
  - But not, as we already showed, in CGF.
  - (Hence it is not possible to compile BGF to CGF.)

Registers as Polymers

- Initially empty register r_j: a seed $\mathbf{Z_j}$
- Increment on r_j: produce a new monomer and associate it to the polymer
- Decrement on r_j: remove last monomer

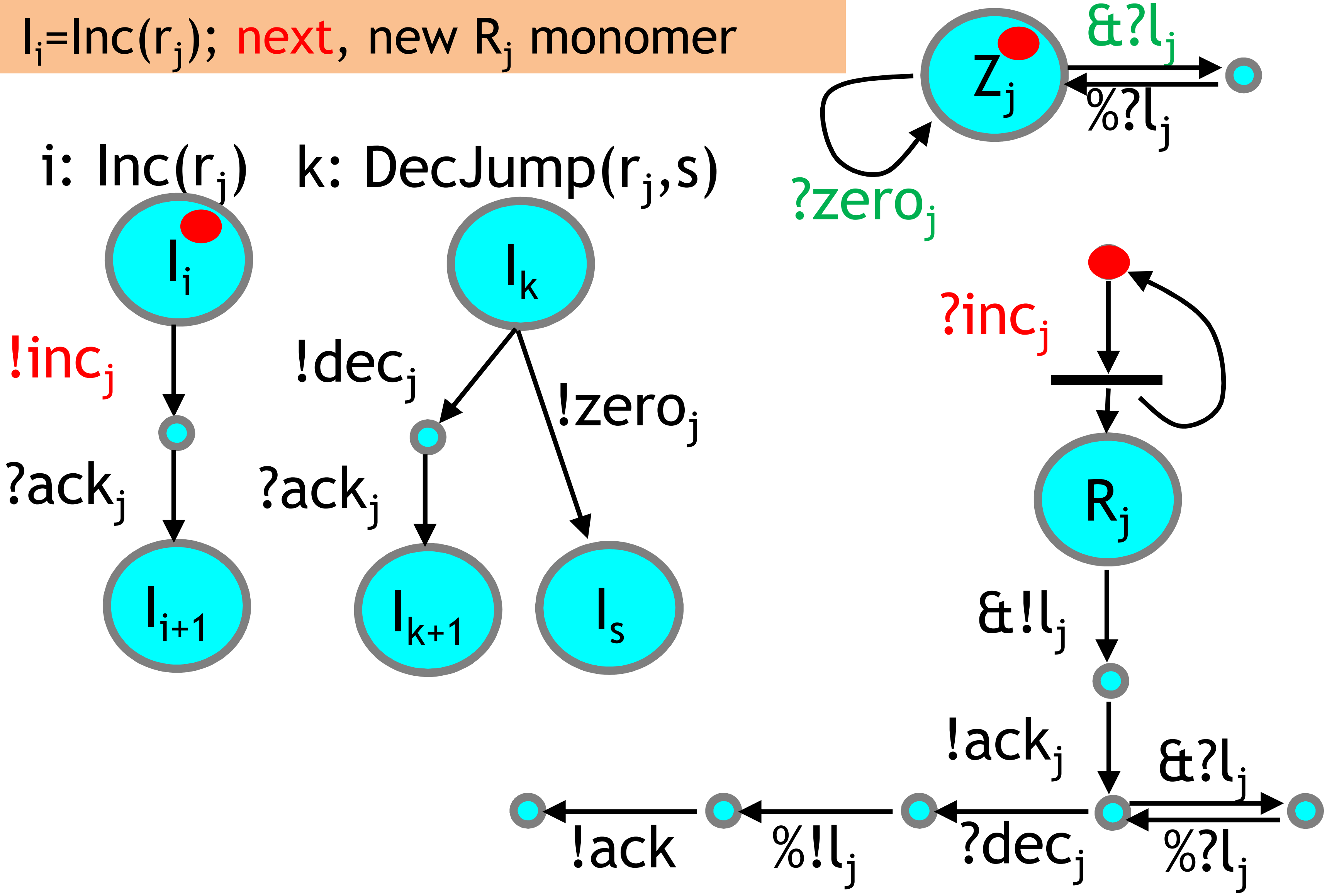


RAM encoding in BGF



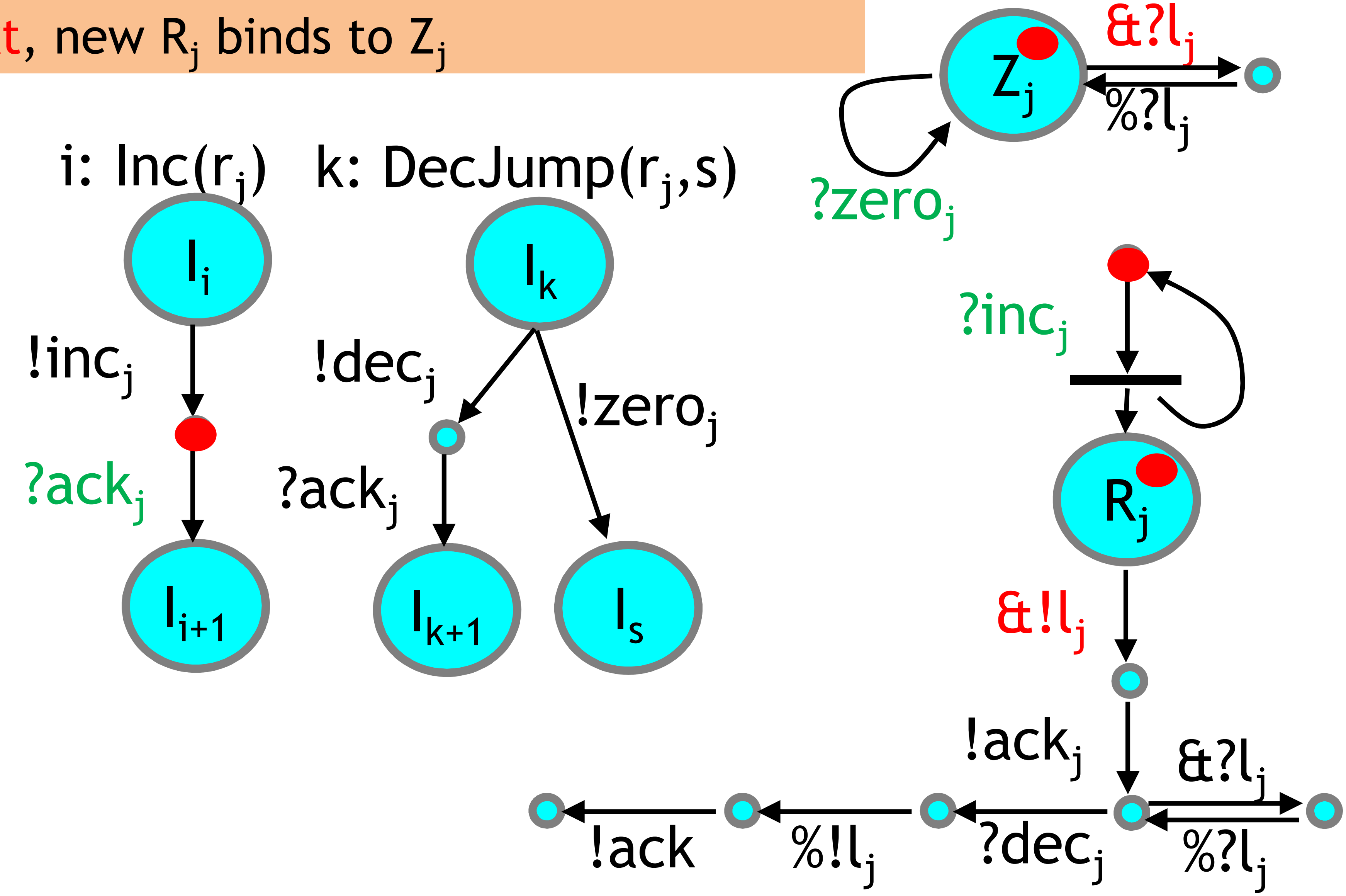
RAM encoding in BGF

r_j=0; I_i=Inc(r_j); next, new R_j monomer



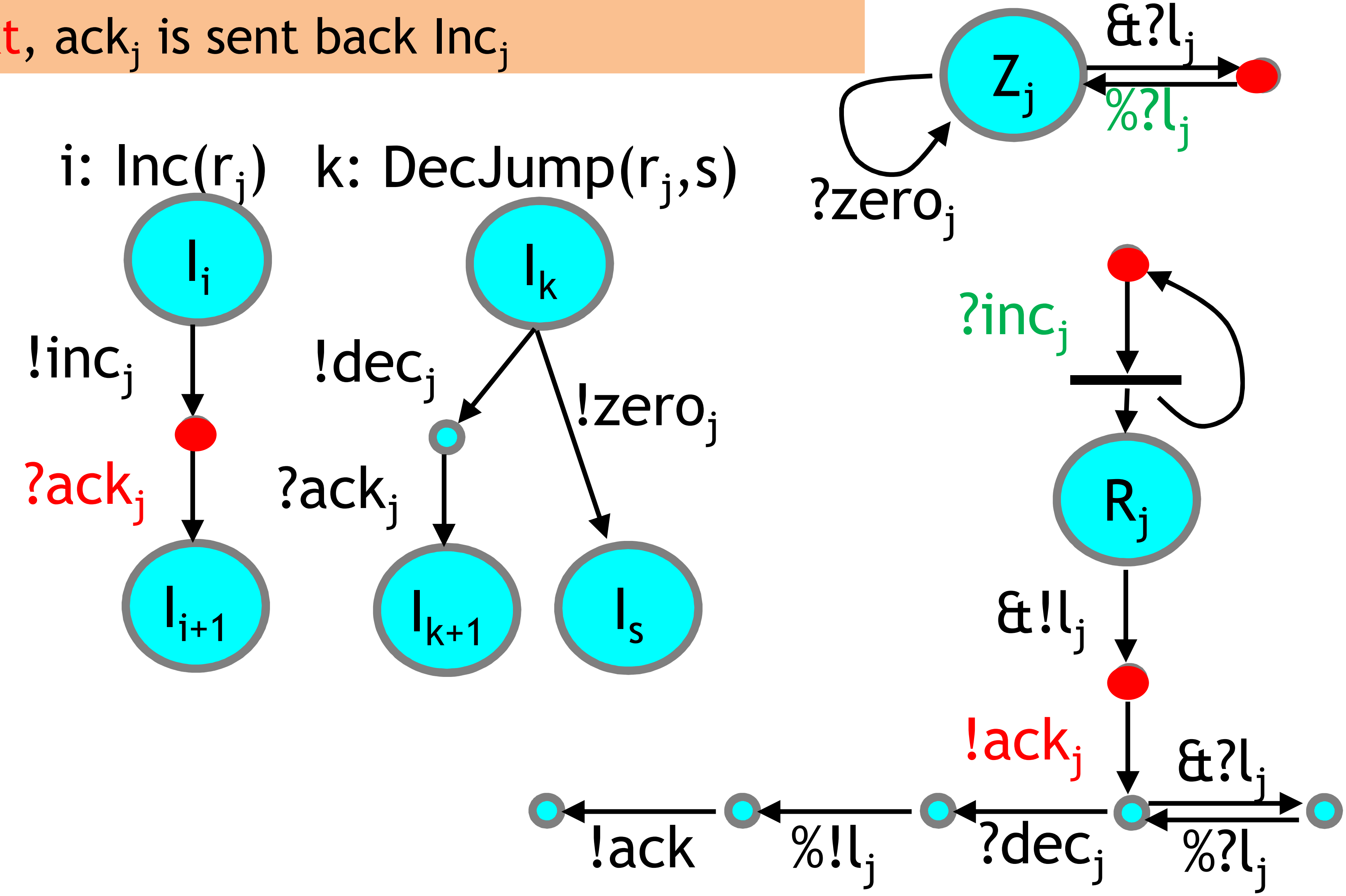
RAM encoding in BGF

next, new R_j binds to Z_j



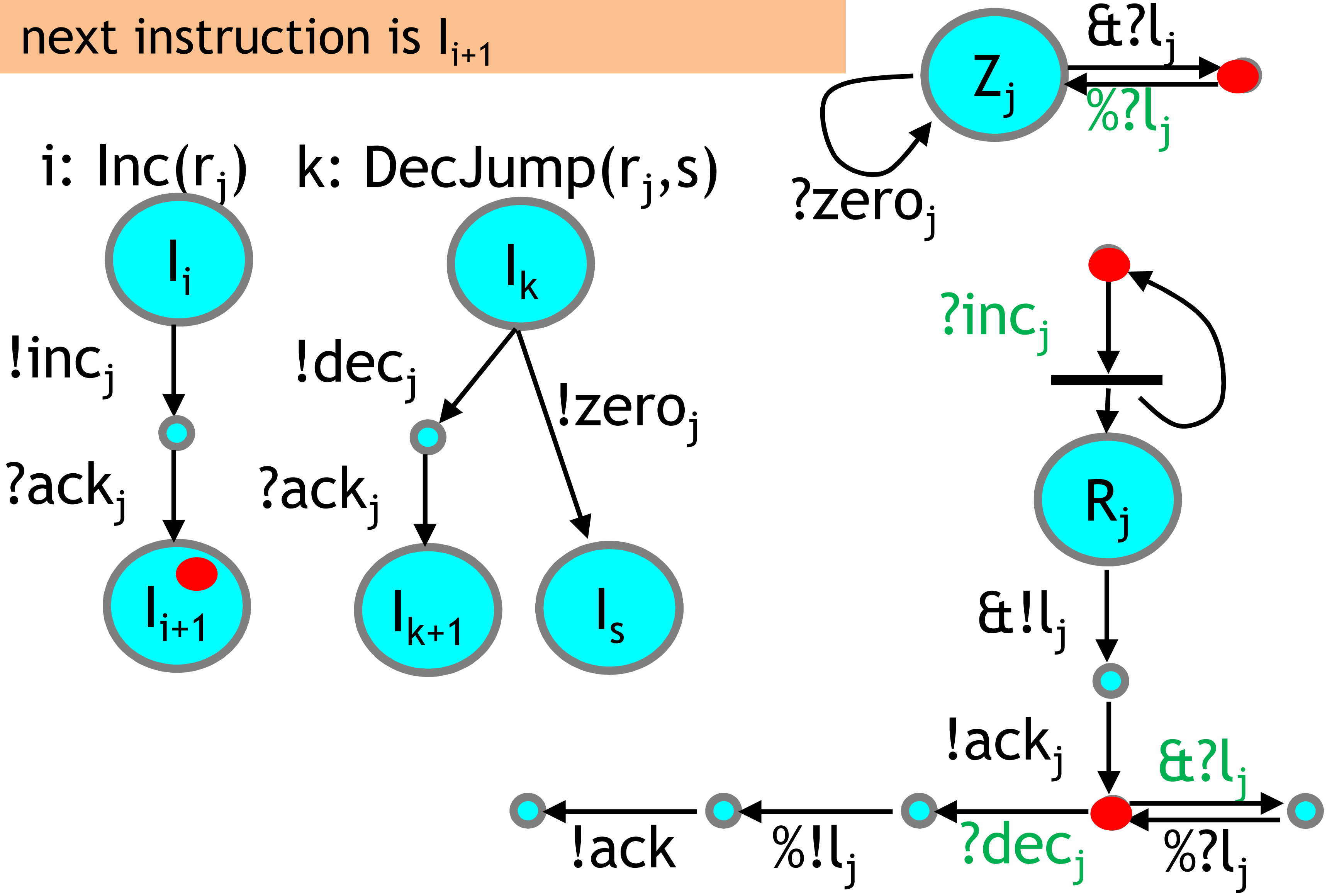
RAM encoding in BGF

next, ack_j is sent back Inc_j



RAM encoding in BGF

r_j=1; next instruction is I_{i+1}



Talk Outline

- Basic Chemistry and Basic Biochemistry
  - Biochemistry = Chemistry + complexation
- Chemical Ground Form (CGF) [TCS08]
  - A process algebra for basic chemistry
  - Basic chemistry can't compute! [Sol08]
- Biochemical Ground Form (BGF) [AB08]
  - A process algebra for basic biochemistry
  - Basic biochemistry is Turing-complete.
- Conclusions
  - Basic biochemistry > Basic chemistry

Conclusions

- A theoretical result
  - Basic Biochemistry > Basic Chemistry
    (should please the biologists…)

- Some practical modeling implications:
  - A finite model in BGF (e.g. of polymerization)
    may correspond to an infinite model in FSRN
  - A model in BGF (e.g. of multiple protein phosphorylation states)
    may correspond to an $O(2^n)$ bigger model in FSRN
  - Even a model in CGF
    may correspond to an $O(n^2)$ bigger model in FSRN

- Process algebra modeling leads to:
  - Compact model presentation
  - Component-based modeling
  - Compositional (separate-subsystems) modeling