Beyond Numbers: Physical Simulation with Complex States

Harold Fellermann
Senior Research Associate

Newcastle University
School of Computing Science
Nat's Group

Algorithmic Nature of Biology

"Computers are to Biology as Mathematics is to Physics."

Harold Morowitz

- Biological systems are highly organized
- We recognize data structures and programs
- Nature Computes!
- Especially true for Synthetic Biology

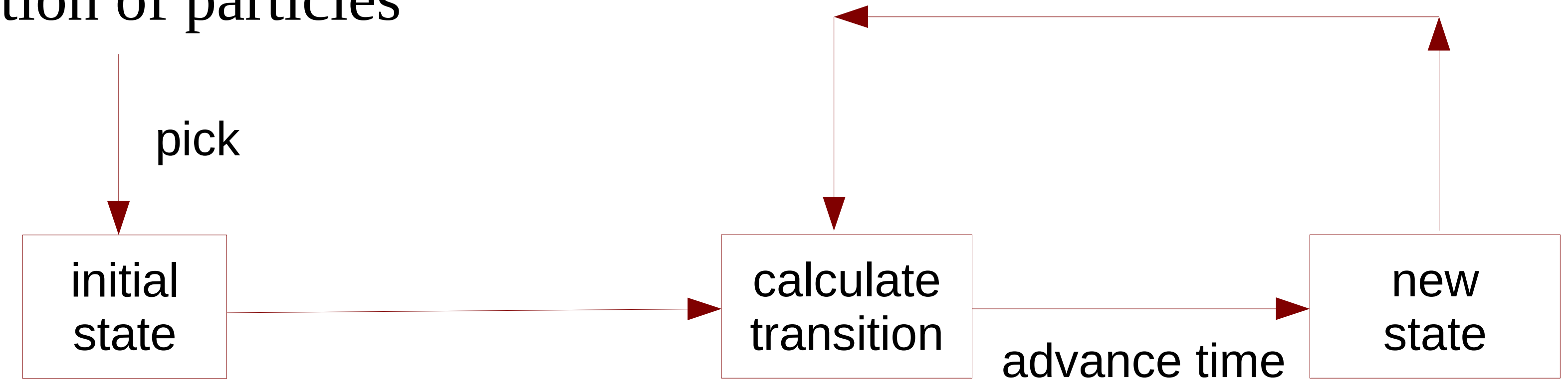
Outline

- **Physical simulation for Synthetic Biology**
  - Self-replicating emulsion compartments
  - Molecular DNA/RNA replicators
  - DNA assembly and computing
- **Formal calculi for Synthetic Biology**
  - Molecular DNA/RNA replicators
  - Compartmented reaction systems
  - Reconfiguring biological DNA
- **Conclusion**

Simulation – The Mathematics of Time

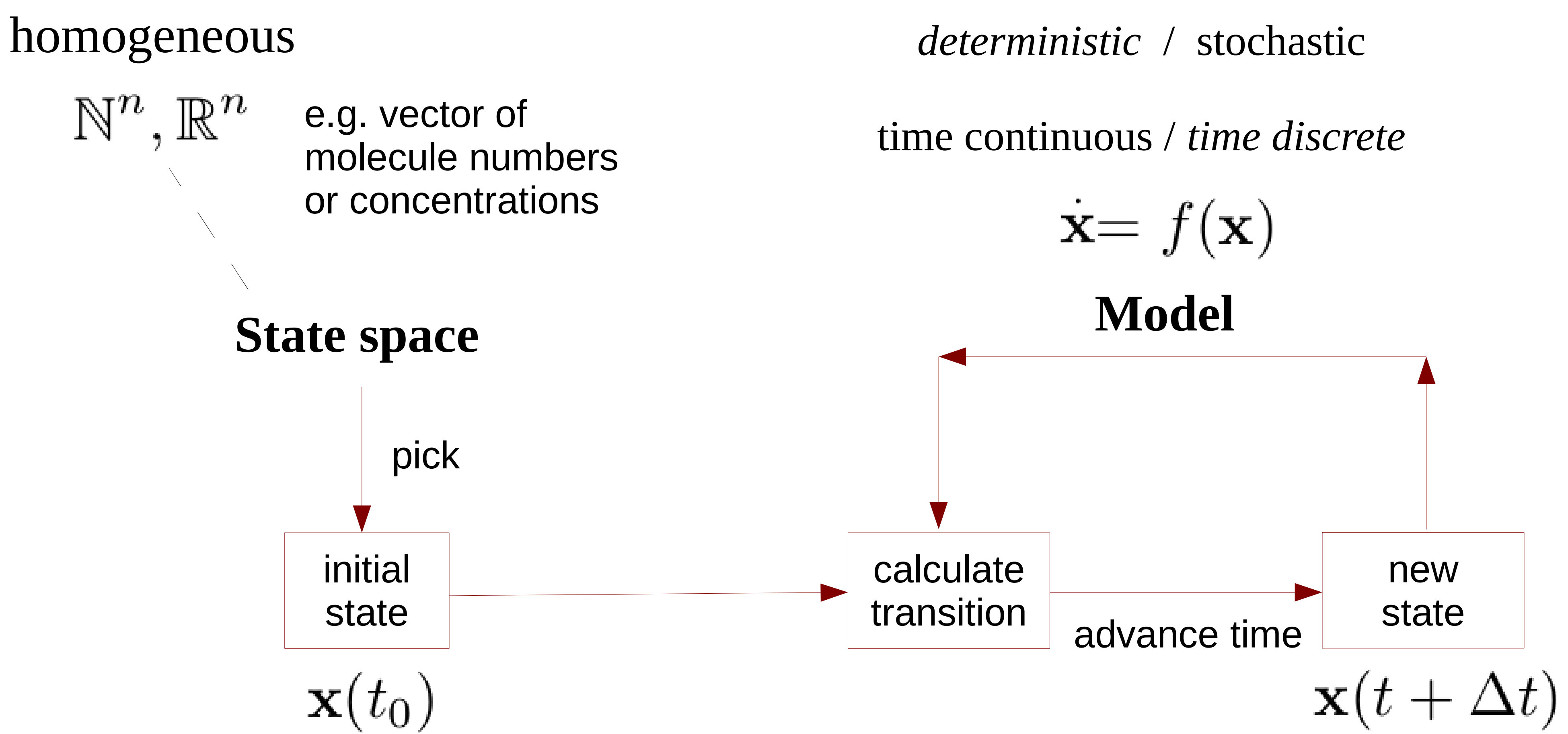
State space,e.g.
- mol. concentrations
- position of particles

Model
defines interactions
(domain expertise enter here!)

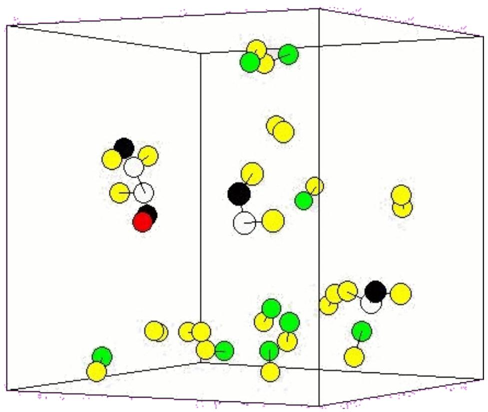


→ predict, explain, guide experiments, illuminate uncertainties, ...

Simulation – The Mathematics of Time



Simulation – The Mathematics of Time



particle based

$\mathbb{R}^{3N}$

State space

deterministic / stochastic

equation of motion:

$$\dot{\mathbf{x}}_i\ (t) = \mathbf{v}_i(t)$$

$$\dot{\mathbf{v}}_i\ (t) = -\nabla F_i(\mathbf{x}(t))$$

F defines interactions

Model

pick

initial state

$\mathbf{x}(t_0)$

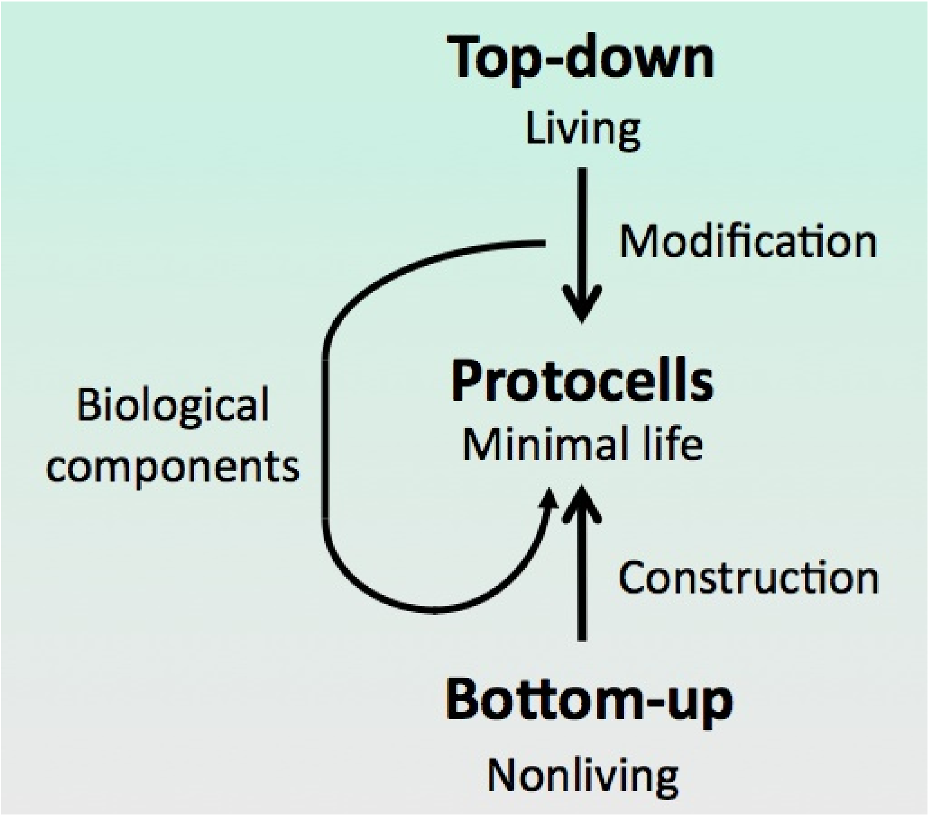
calculate transition

advance time

new state

$\mathbf{x}(t + \Delta t)$

Top-Down & Bottom-Up Synthetic Biology



This talk is about bottom-up approaches:

Biomolecules are used to assemble biomimetic system:
Amphiphiles (lipids), DNA/RNA, proteins

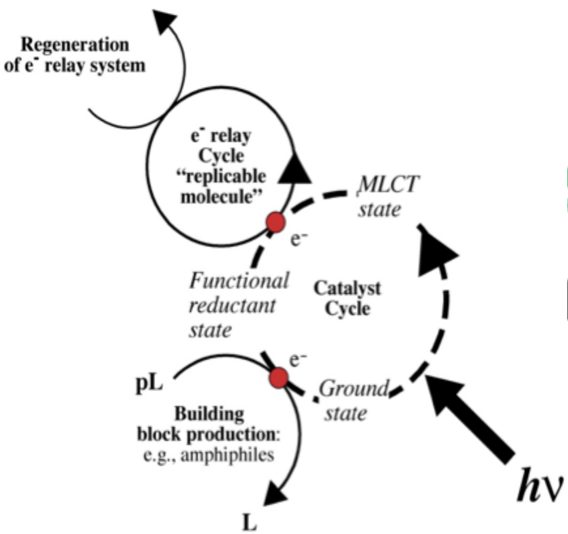
Protocells: Bottom-Up Synthetic Biology

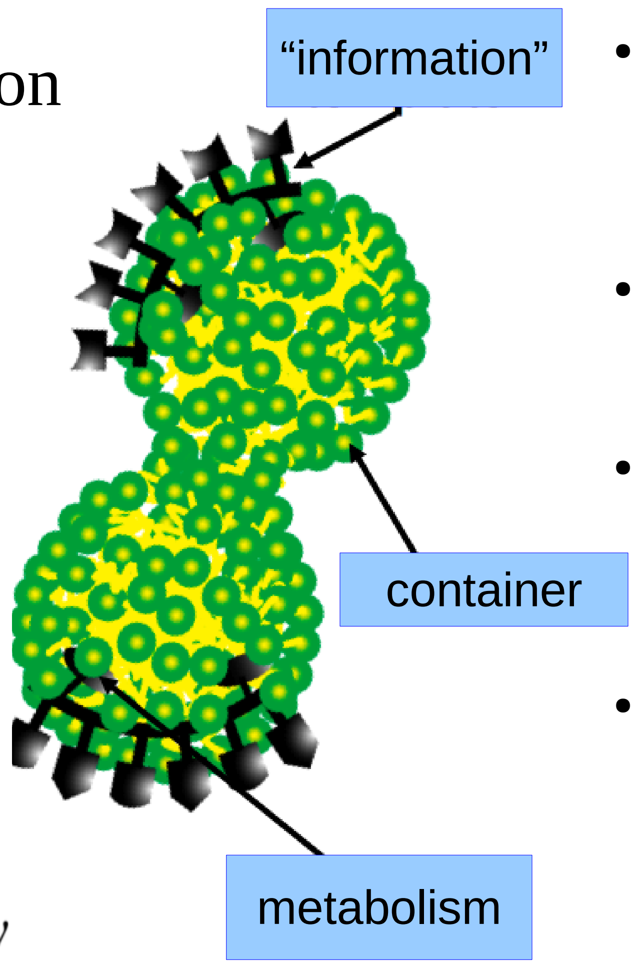
Danmarks Grundforskningsfond
Danish National Research Foundation

Aim:

De novo creation of chemical aggregates able to

- metabolize
- replicate
- undergo evolution



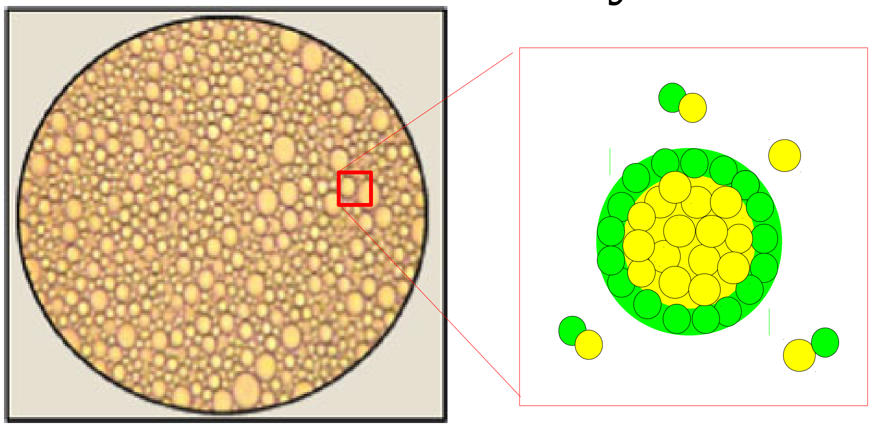


- Everything attached to the **external interface** of a lipid aggregate
- A **single reaction mechanism** for all metabolic reactions
- **direct participation of information** molecules in the metabolic reaction (no proteins!)
- Information is sequence-dependent but **not encoding**

Rasmussen et al., *Artif. Life*, 2003

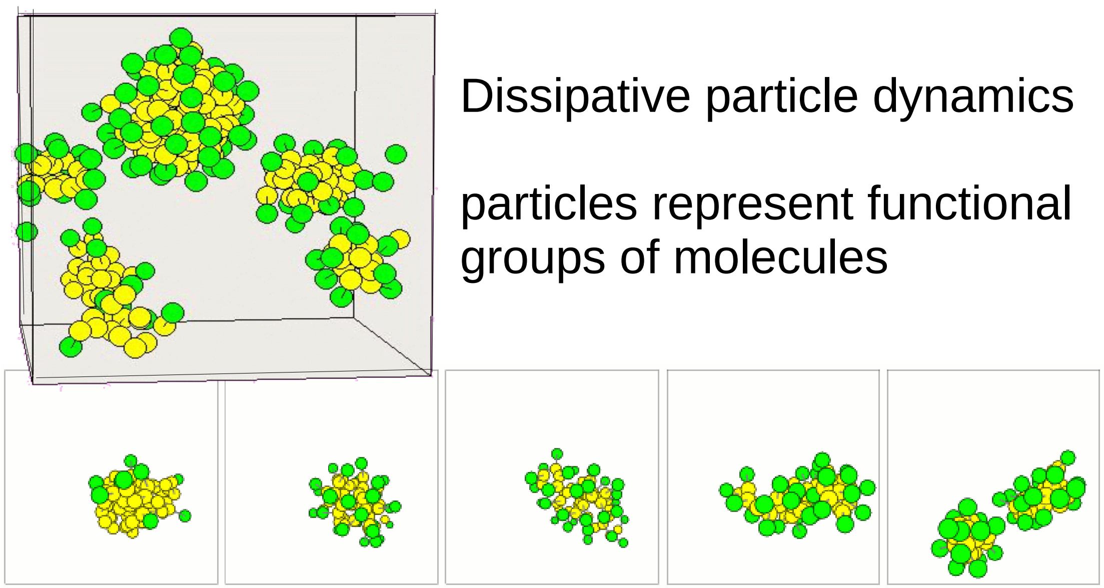
Protocells: Replicating Compartments

Oil-water-surfactant systems form emulsion compartments



Metabolism that can transform
oily precursor into functional surfactant:

“Coarse-grained” simulation



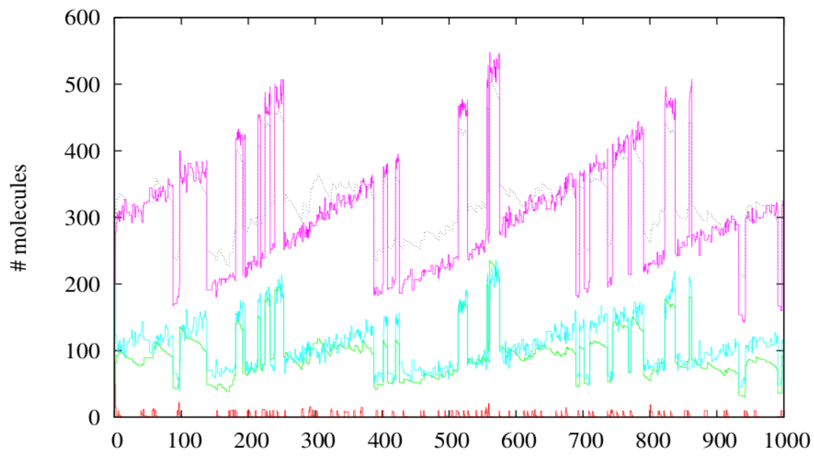
Dissipative particle dynamics

particles represent functional groups of molecules

Fellermann & Sole, *Phil. Trans. R. Soc., B*, 2007

Non-Spatial simulation

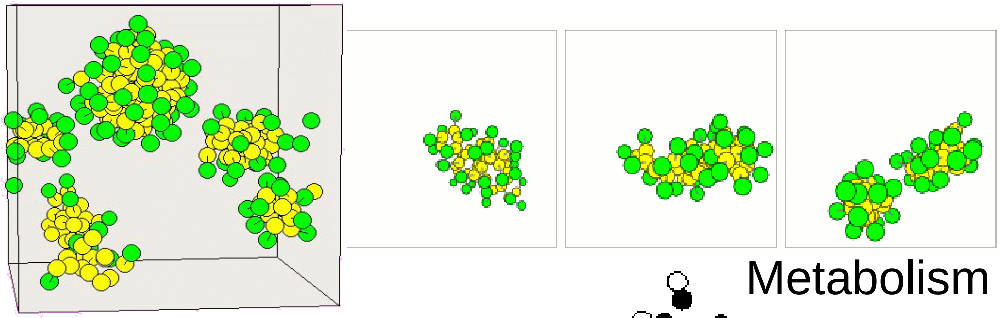
$$(L, L_{\mathrm{P}})_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})} \in \mathbb{N}^4$$



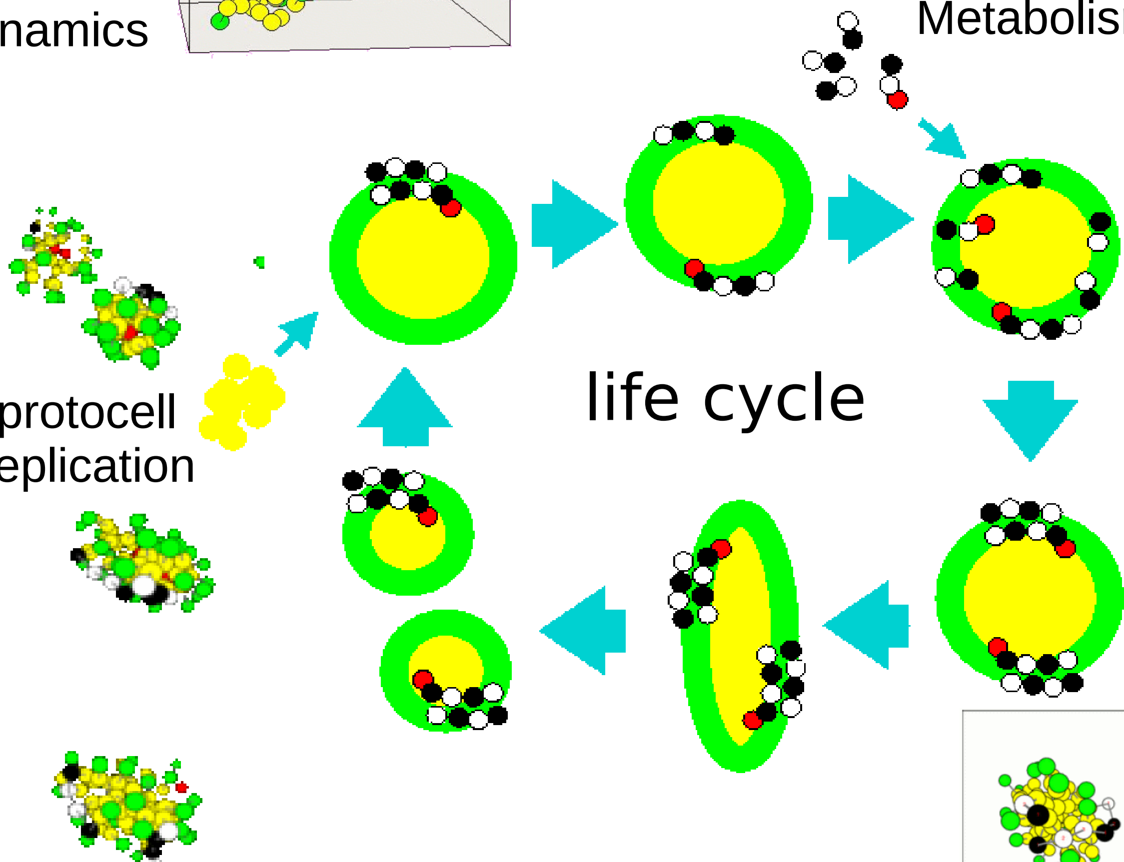
stochastic (Gillespie) simulation

Protocells: Physical Simulation

Container dynamics

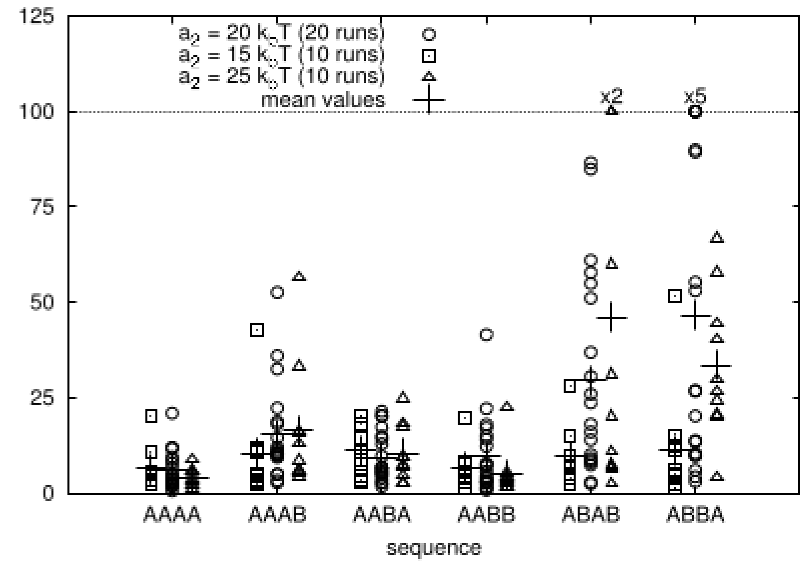


Metabolism

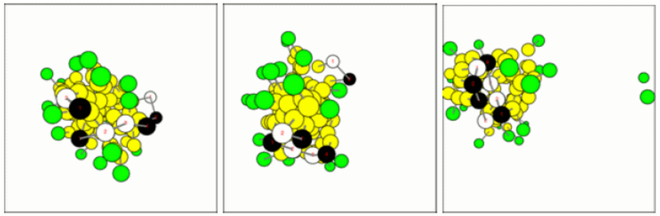


protocell replication

Emergent fitness function



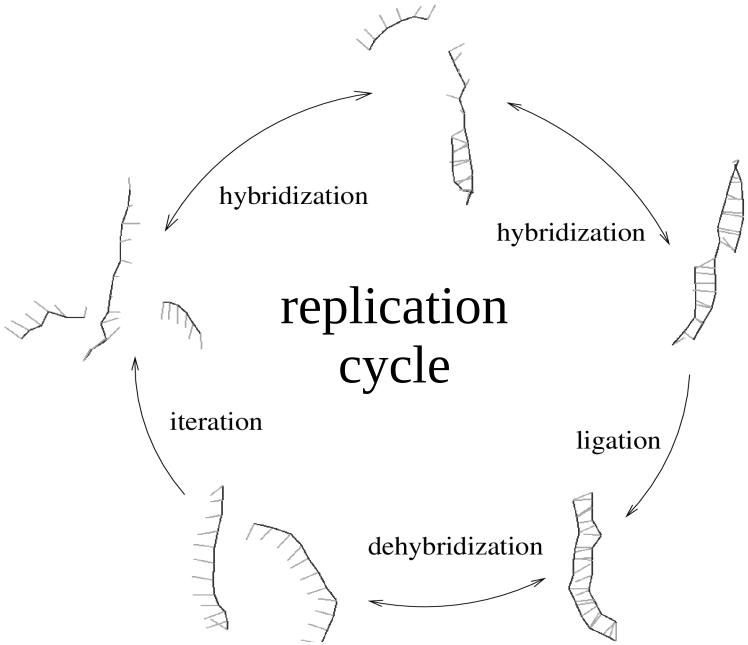
DNA replication

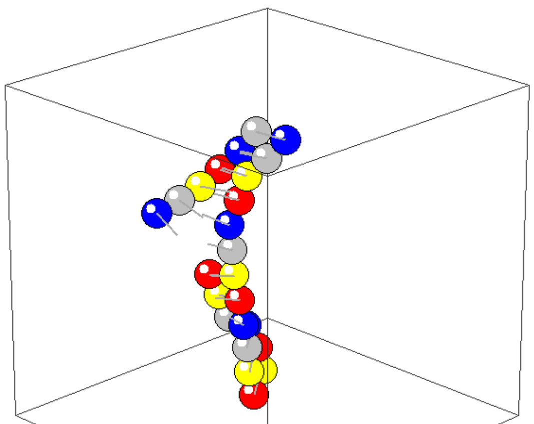


Fellermann & Sole, *Phil. Trans. R. Soc., B*, (2007) **362:** 1803

Non-enzymatic DNA/RNA replication

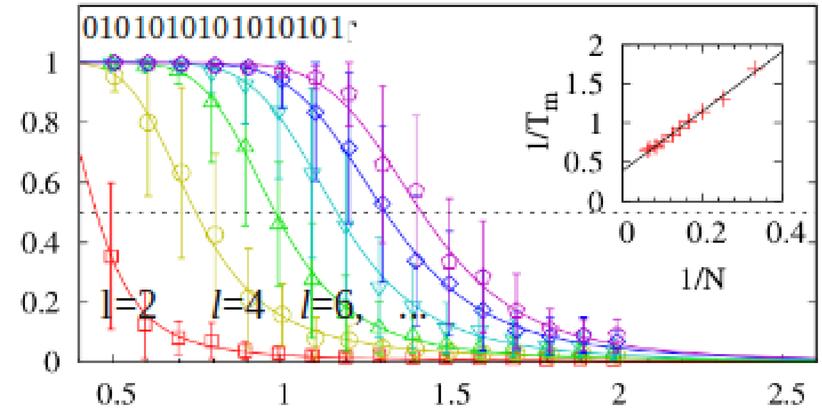
Template directed replication reaction



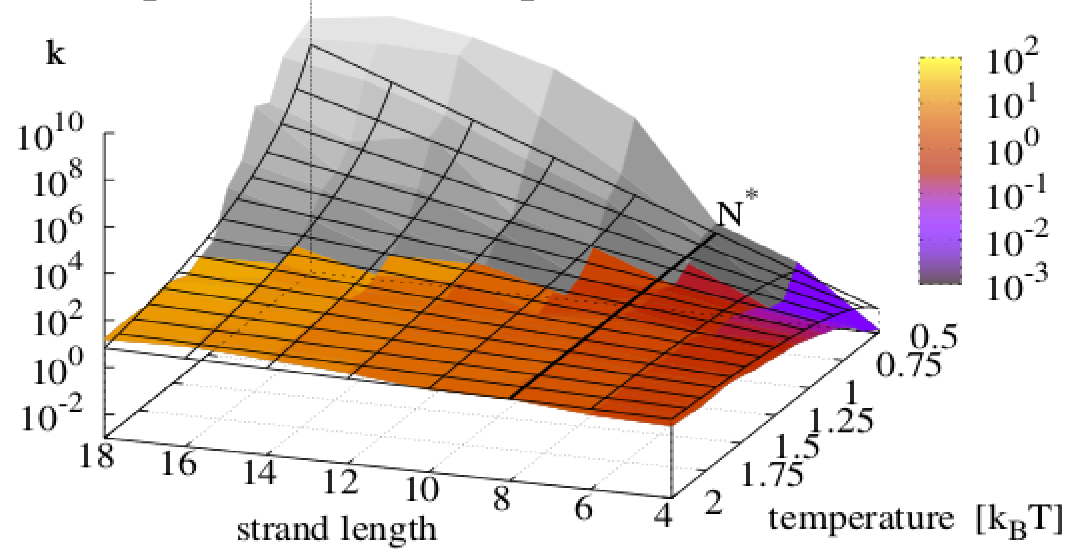


3D constrained Langevin dynamics simulation

melting curve measurements



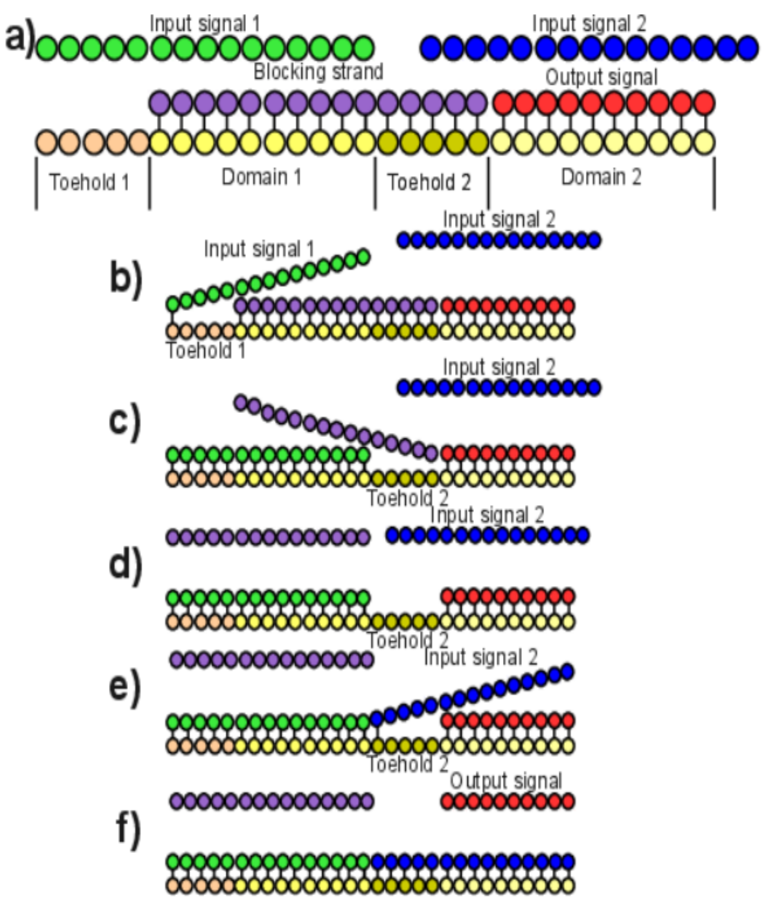
replication rate dependence

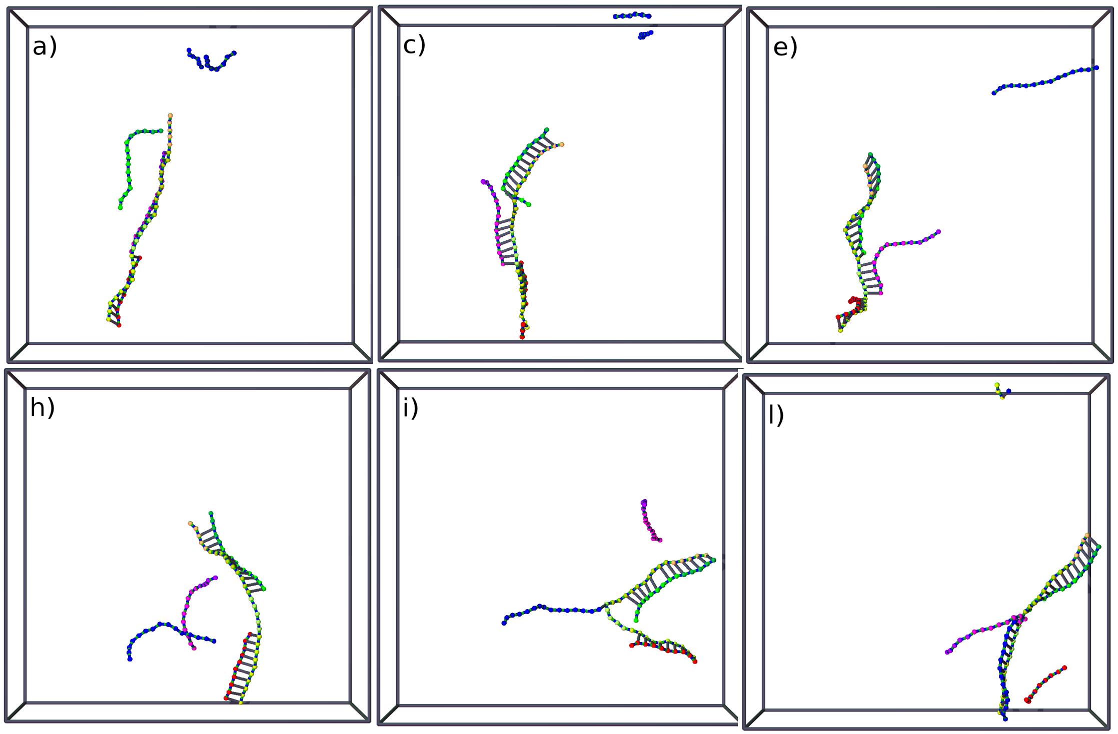


Fellermann, Rasmussen, *Entropy* 2011

DNA Strand Displacement Computing

DNA strand displacement *join* gate (Cardelli, 2010)





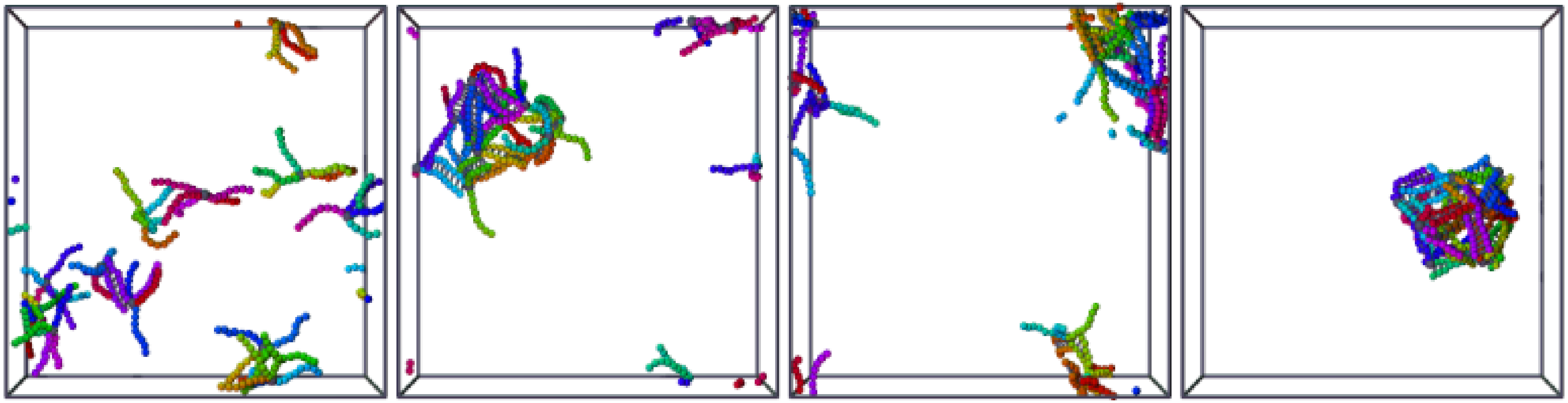
Fidelity of the gate:

11	0.95 ± 0.05
10	0.02 ± 0.01
01	$\boxed{0.58 \pm 0.15}$
00	0.00 ± 0.00

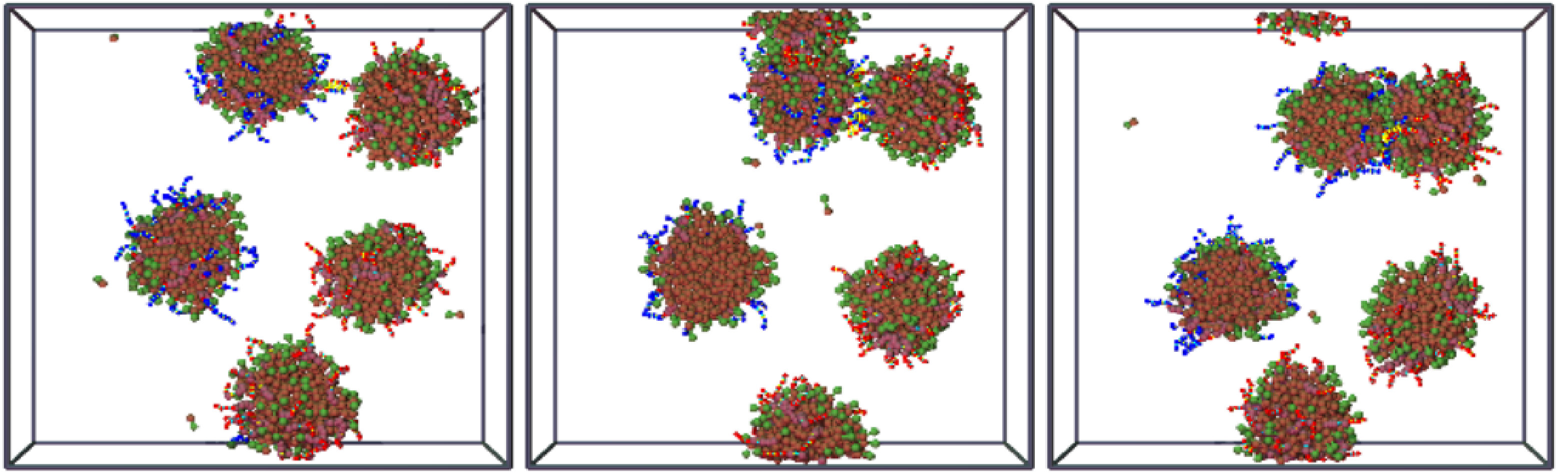
Svaneborg, Fellermann, Rasmussen *Lect. Notes Comput. Sc.* 2012

Physical Simulation of DNA Assembly

DNA assembly of an icosahedron from trisoligomers



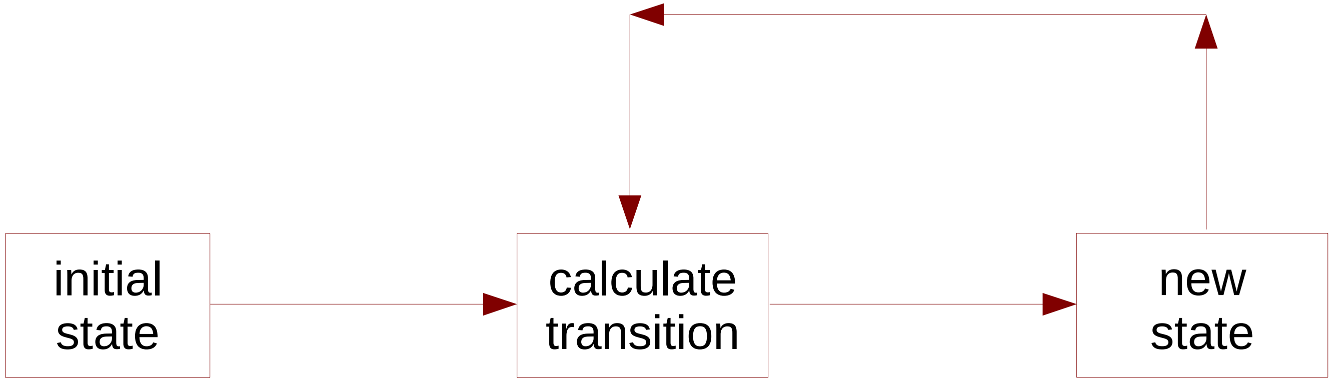
DNA induced association and fusion of oil-in-water compartments



Svaneborg, Fellermann, Rasmussen *Lect. Notes Comput. Sc.* 2012

Simulation Beyond Numbers

- Most simulation frameworks operate over a **simple, static** and relatively **small** state space.



$$(L, L_{\mathrm{P}})_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})} \in \mathbb{N}^4$$

$$(L, L_{\mathrm{P}})_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})} \xrightarrow{k_{\mathrm{L}}^{+} L^{\mathrm{b}}} (L+1, L_{\mathrm{P}})_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})}$$

$$(L, L_{\mathrm{P}})_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})} \xrightarrow{k_{\mathrm{L}}^{-} L} (L-1, L_{\mathrm{P}})_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})}$$

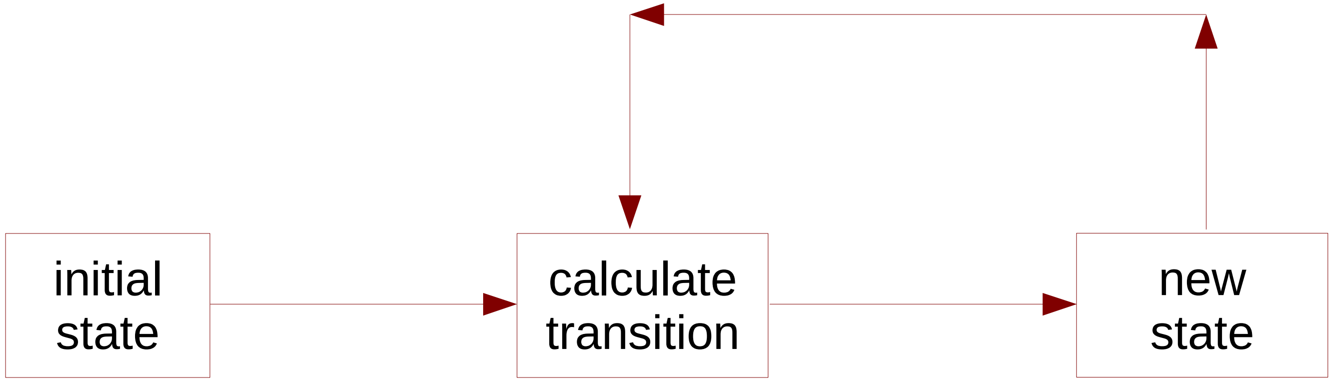
$$(L, L_{\mathrm{P}})_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})} \xrightarrow{k_{\mathrm{L_P}}^{+} L_{\mathrm{P}}^{\mathrm{b}}} (L, L_{\mathrm{P}}+1)_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})}$$

$$(L, L_{\mathrm{P}})_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})} \xrightarrow{k_{\mathrm{L_P}}^{-} L_{\mathrm{P}}} (L, L_{\mathrm{P}}-1)_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})}$$

The state is dynamic, but it is embedded in a static state space.

Simulation Beyond Numbers

- Most simulation frameworks operate over a **simple, static** and relatively **small** state space.



$$(L, L_{\mathrm{P}})_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})} \in \mathbb{N}^4$$

$$
\begin{aligned}
(L, L_{\mathrm{P}})_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})} &\xrightarrow{k_{\mathrm{L}}^{+} L^{\mathrm{b}}} (L+1, L_{\mathrm{P}})_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})} \\
(L, L_{\mathrm{P}})_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})} &\xrightarrow{k_{\mathrm{L}}^{-} L} (L-1, L_{\mathrm{P}})_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})} \\
(L, L_{\mathrm{P}})_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})} &\xrightarrow{k_{\mathrm{L_P}}^{+} L_{\mathrm{P}}^{\mathrm{b}}} (L, L_{\mathrm{P}}+1)_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})} \\
(L, L_{\mathrm{P}})_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})} &\xrightarrow{k_{\mathrm{L_P}}^{-} L_{\mathrm{P}}} (L, L_{\mathrm{P}}-1)_{(L^{\mathrm{tot}}, L_{\mathrm{P}}^{\mathrm{tot}})}
\end{aligned}
$$

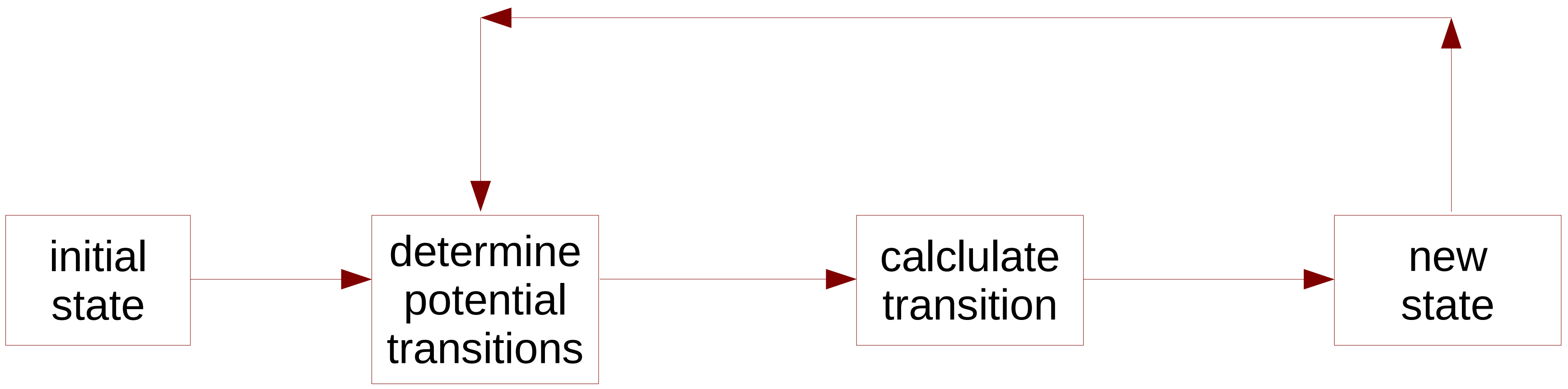
The state is dynamic, but it is embedded in a static state space.

Simulation Beyond Numbers

- Most simulation frameworks operate over a **simple, static** and relatively **small** state space.
- Biological state spaces are often **dynamic, complex**, and can be **arbitrarily large**.
- Examples:
  - Reconfiguring polymers (RNA, DNA, oligosaccharides)
  - Protein complexes
  - Compartment structures

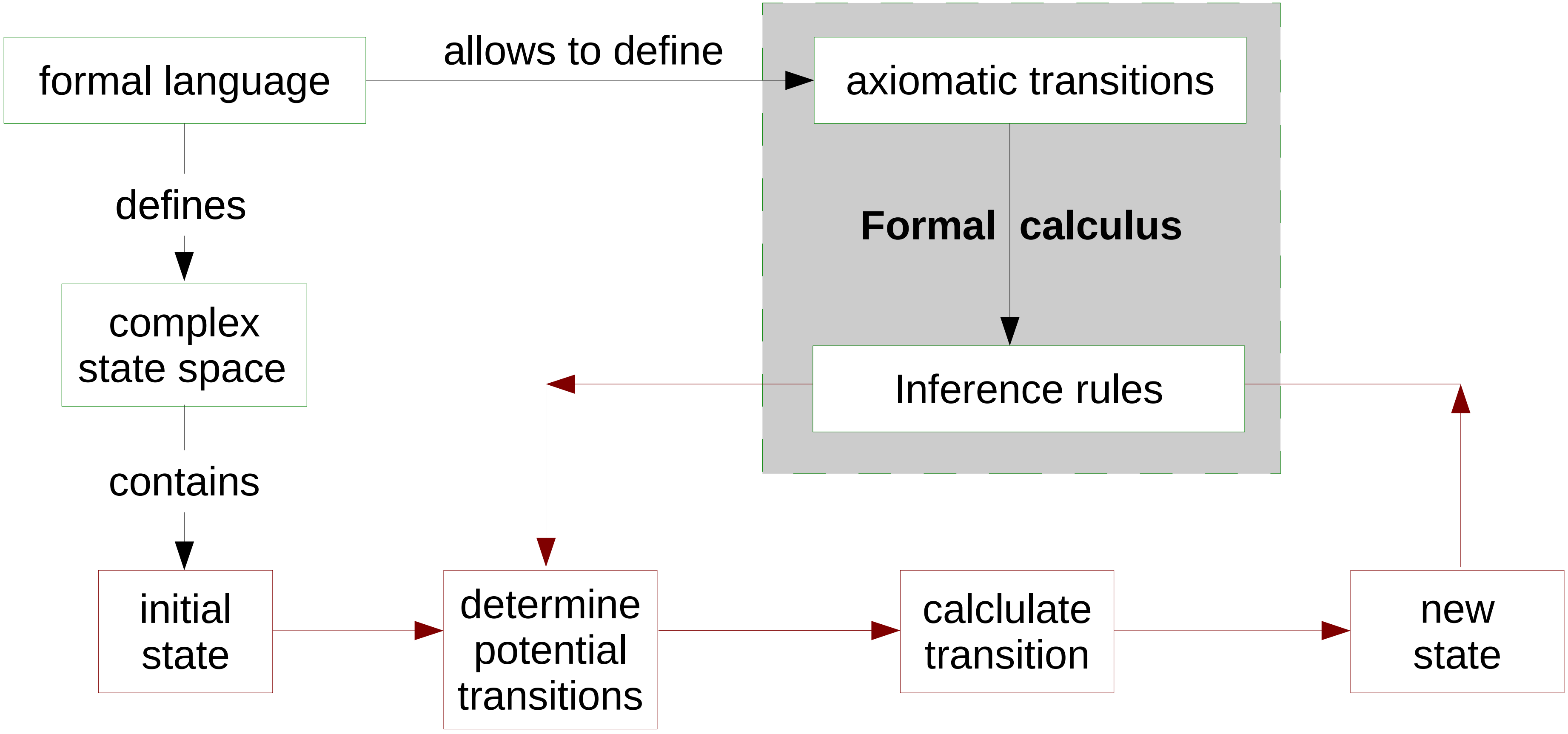
Simulation Beyond Numbers

- Most simulation frameworks operate over a **simple, static** and relatively **small** state space.
- Biological state spaces are often **dynamic, complex**, and can be **arbitrarily large**.
- CS offers tools to operate over “dynamic” state spaces.



Simulation Beyond Numbers

- CS offers tools to operate over dynamic state spaces.



Example: Self-Replicating Polymers

Alphabet of monomers: $A = \{0,1\} = \{$○, ●$\}$

Polymers are strings over A^*

We assume the following processes:

1. Degradation: $l.m \longrightarrow l + m$ 

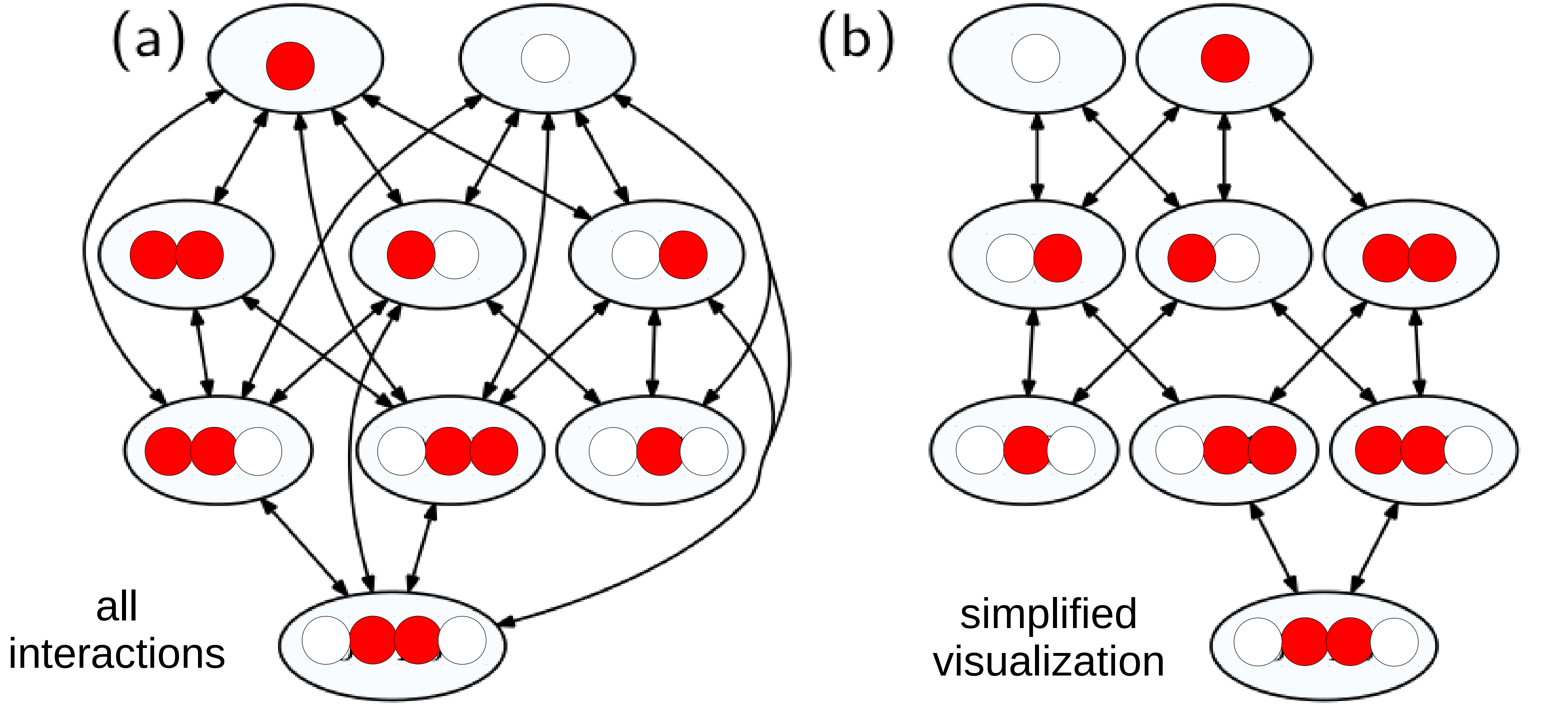
2. Random ligation: $l + m \longrightarrow l.m$ 

3. Autocatalysis: $l + m + l.m \longrightarrow 2\, l.m$ 

The number of possible species A^* is infinite and scales exponential with strand length.

Tanaka, Fellermann, Rasmussen. *Euro Phys Lett*, (submitted)

Example: Self-Replicating Polymers



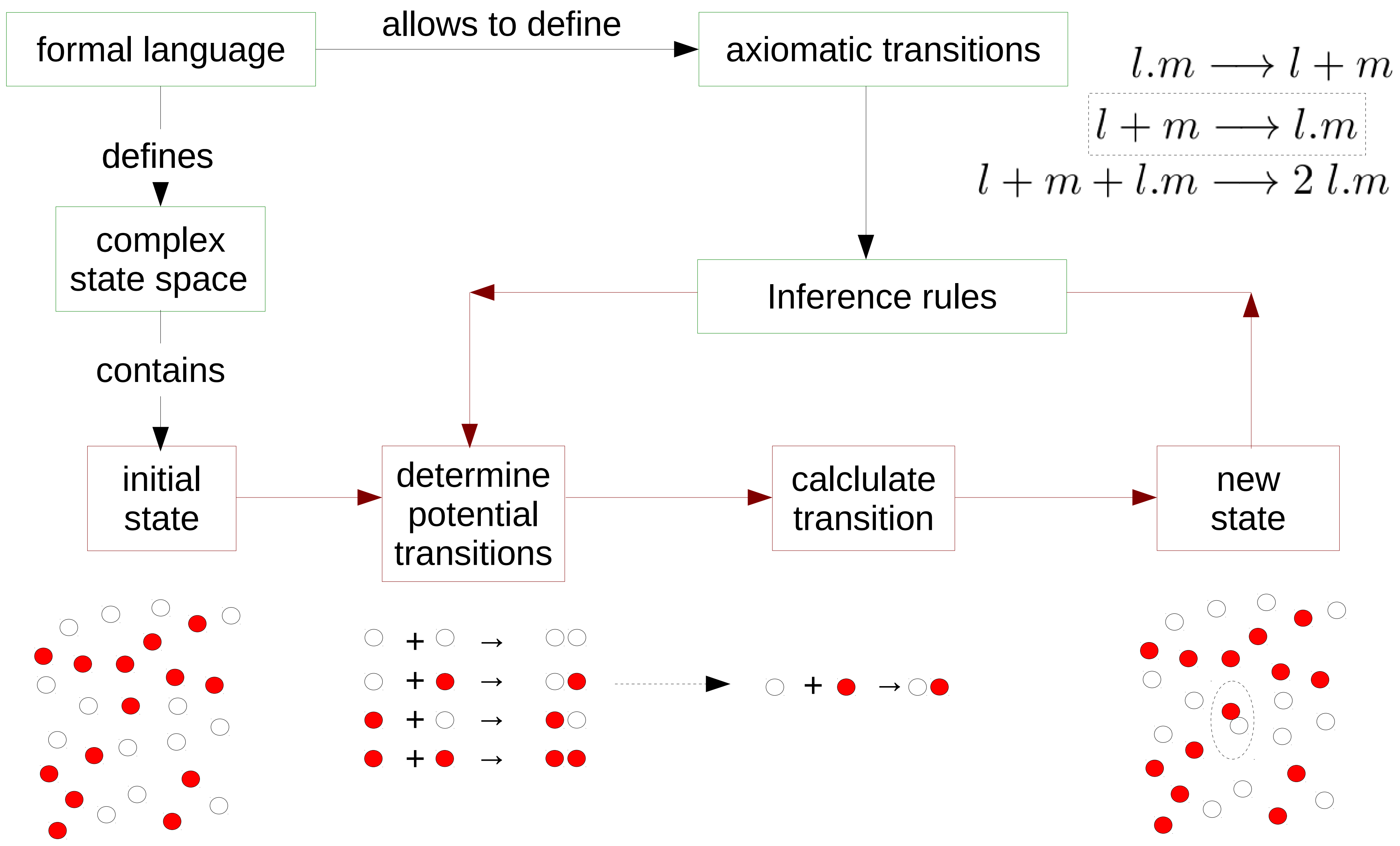
Replicators compete for resources.

They are each other's reaction and degradation products.

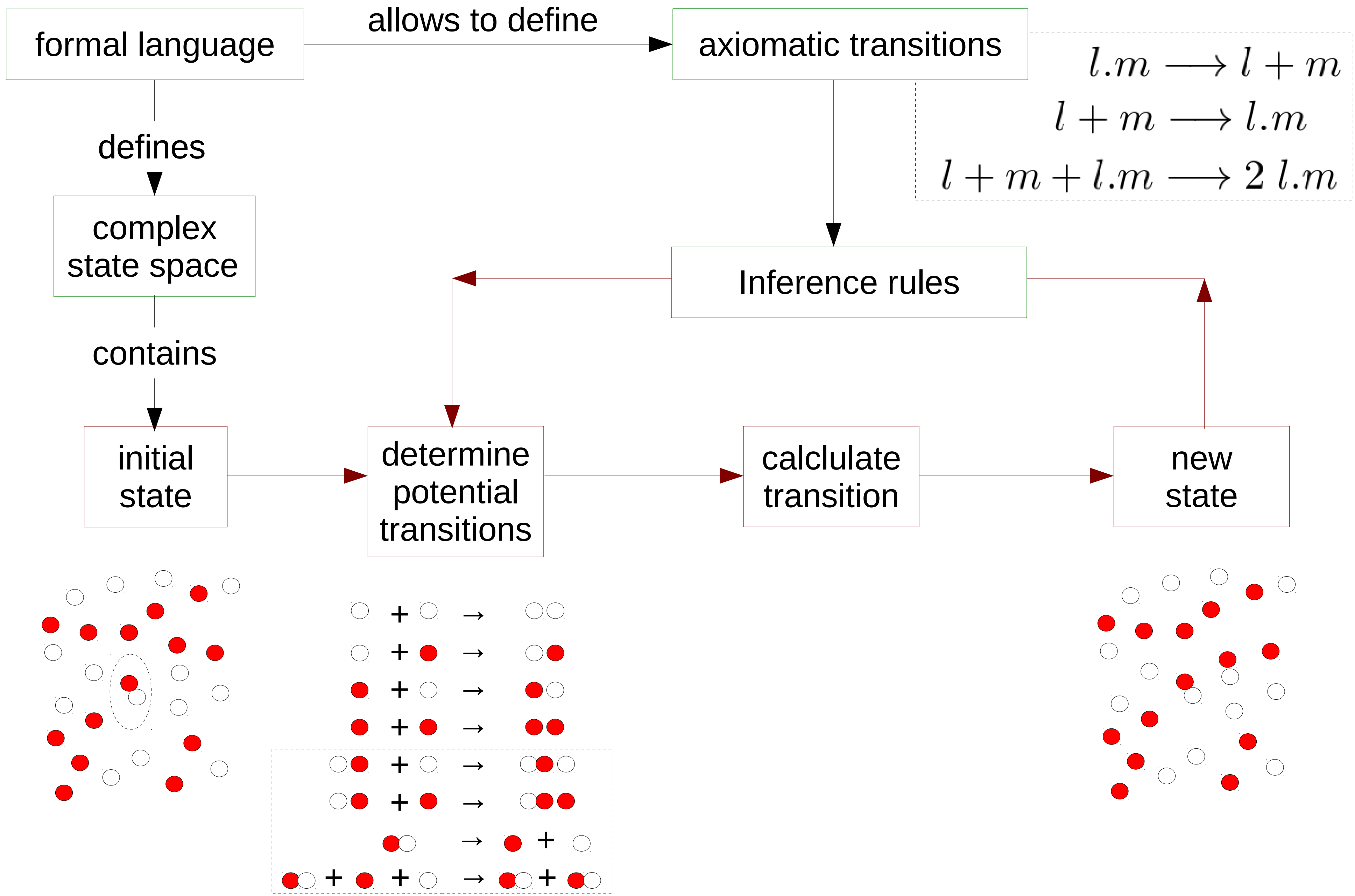
System is closed but energy flow is assumed.

What happens in a pool of monomers?

Example: Self-Replicating Polymers

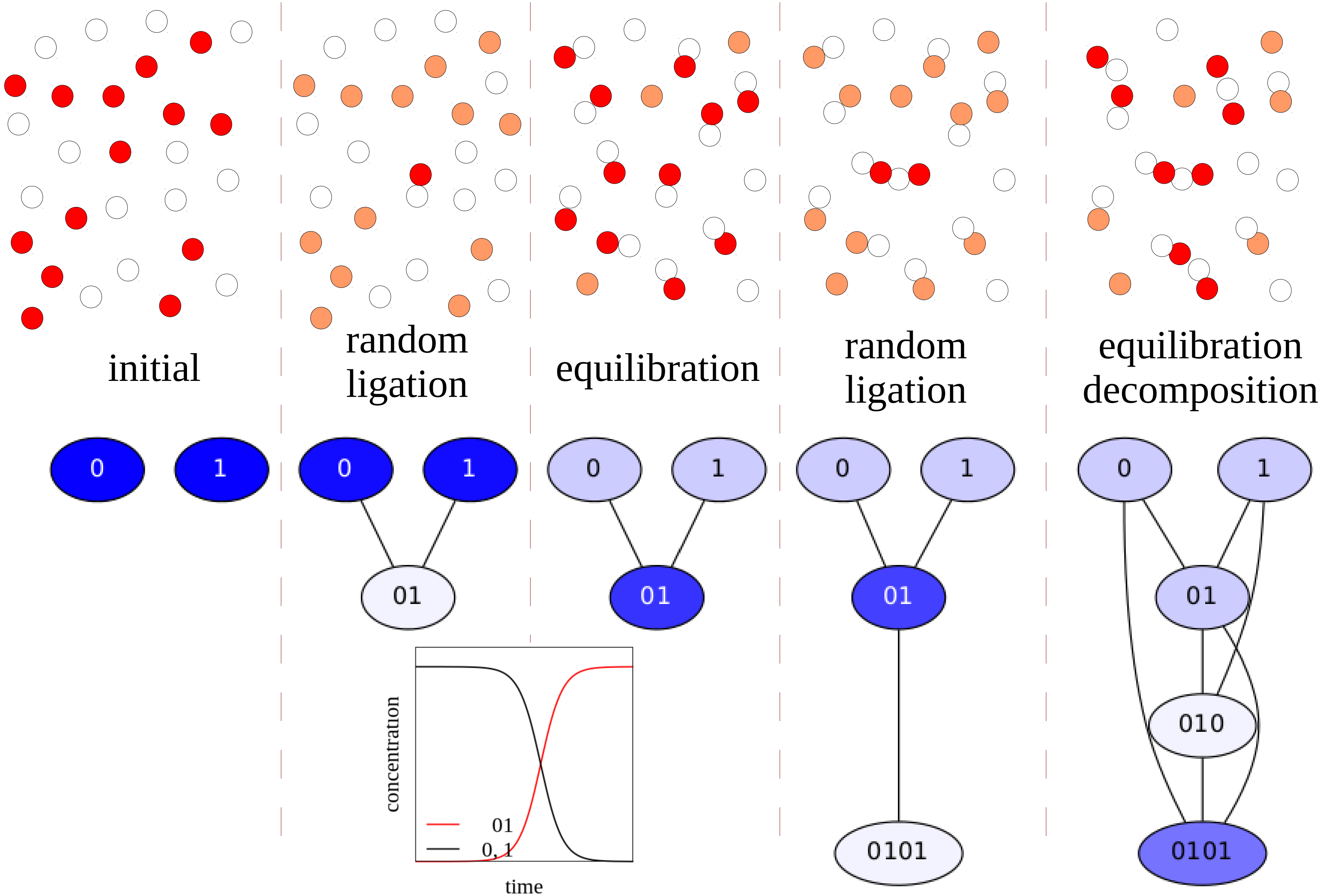


Example: Self-Replicating Polymers



Example: Self-Replicating Polymers

If random ligation is rare

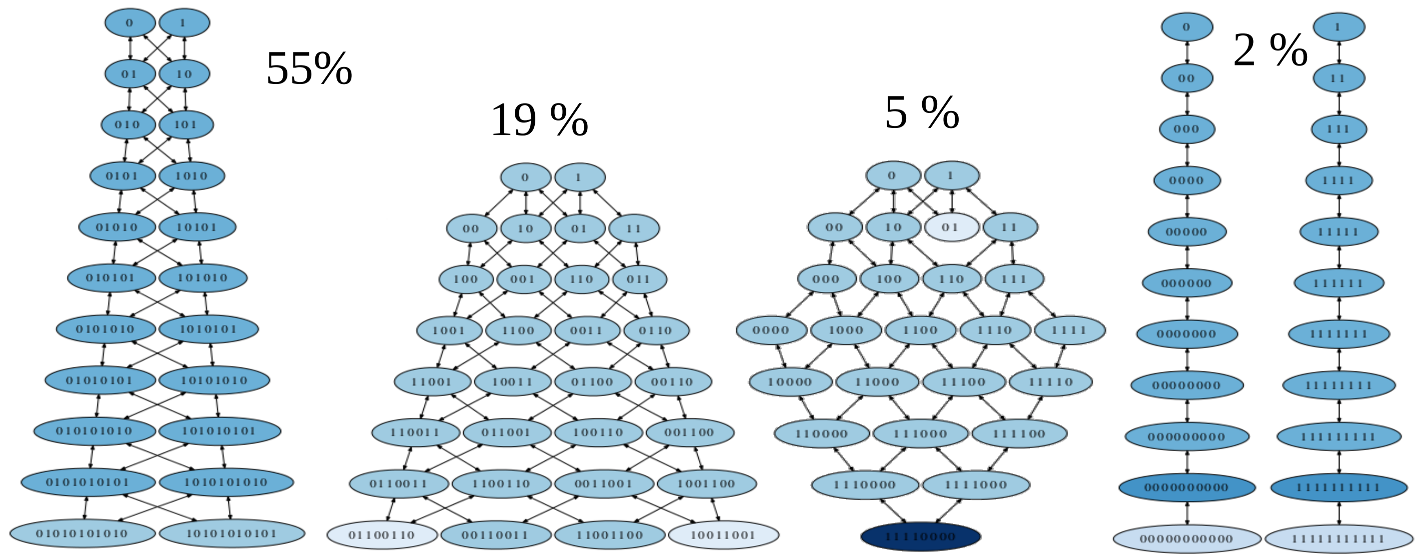


Example: Self-Replicating Polymers

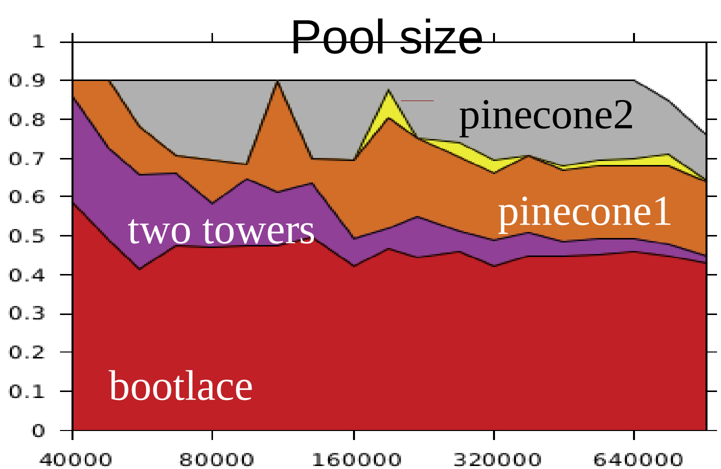
If random ligation is rare, highly ordered sequence patterns emerge:

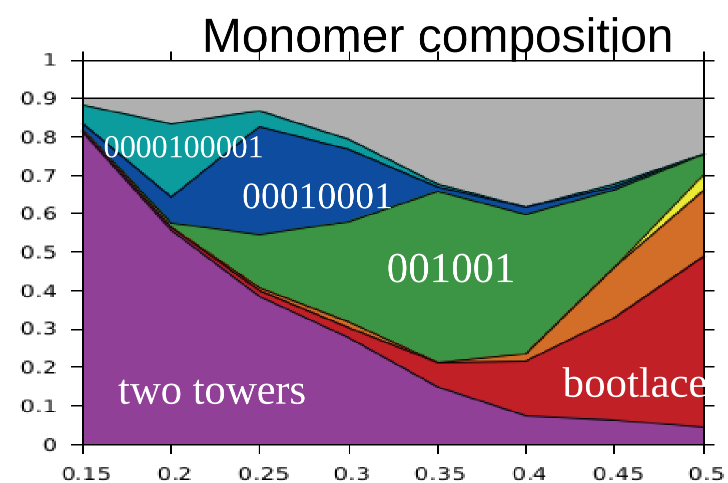
Gillespie simulation over
infintie dimensional
state space

HCA of final states



10101010 11001100 11110000 111111 00 0000

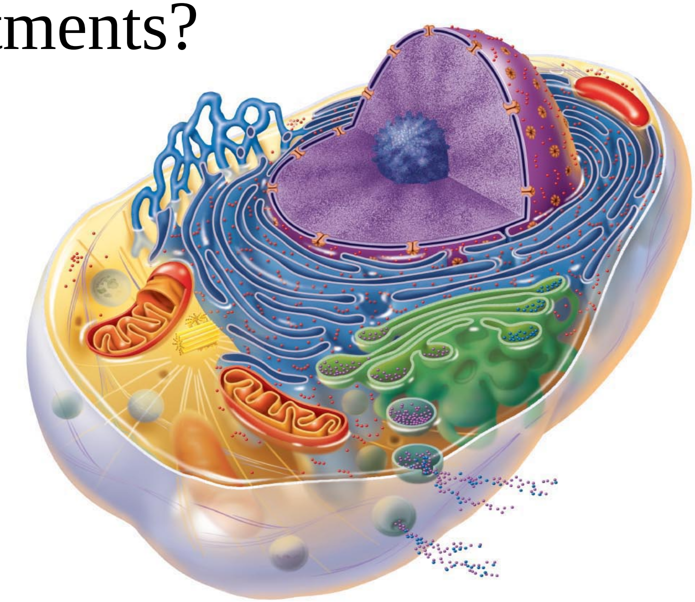




Tanaka, Fellermann, Rasmussen. *Euro Phys Lett*, (submitted)

Compartment Dynamics

Biological systems are commonly compartmentalized.
How to capture dynamics in <u>and of</u> compartments?



Language of nested parentheses:

Recursive grammar:

$$\mathsf{P} := \emptyset \;\mid\; \mathsf{P} + \mathsf{P} \;\mid\; \llparenthesis\, \mathsf{P}\, \rrparenthesis \;\mid\; m_j$$

empty state   composition   compartment   mol. cargo

Transitions among compartments:

$$\llparenthesis\, P \,\rrparenthesis + \llparenthesis\, Q \,\rrparenthesis \longrightarrow \llparenthesis\, P + Q \,\rrparenthesis$$

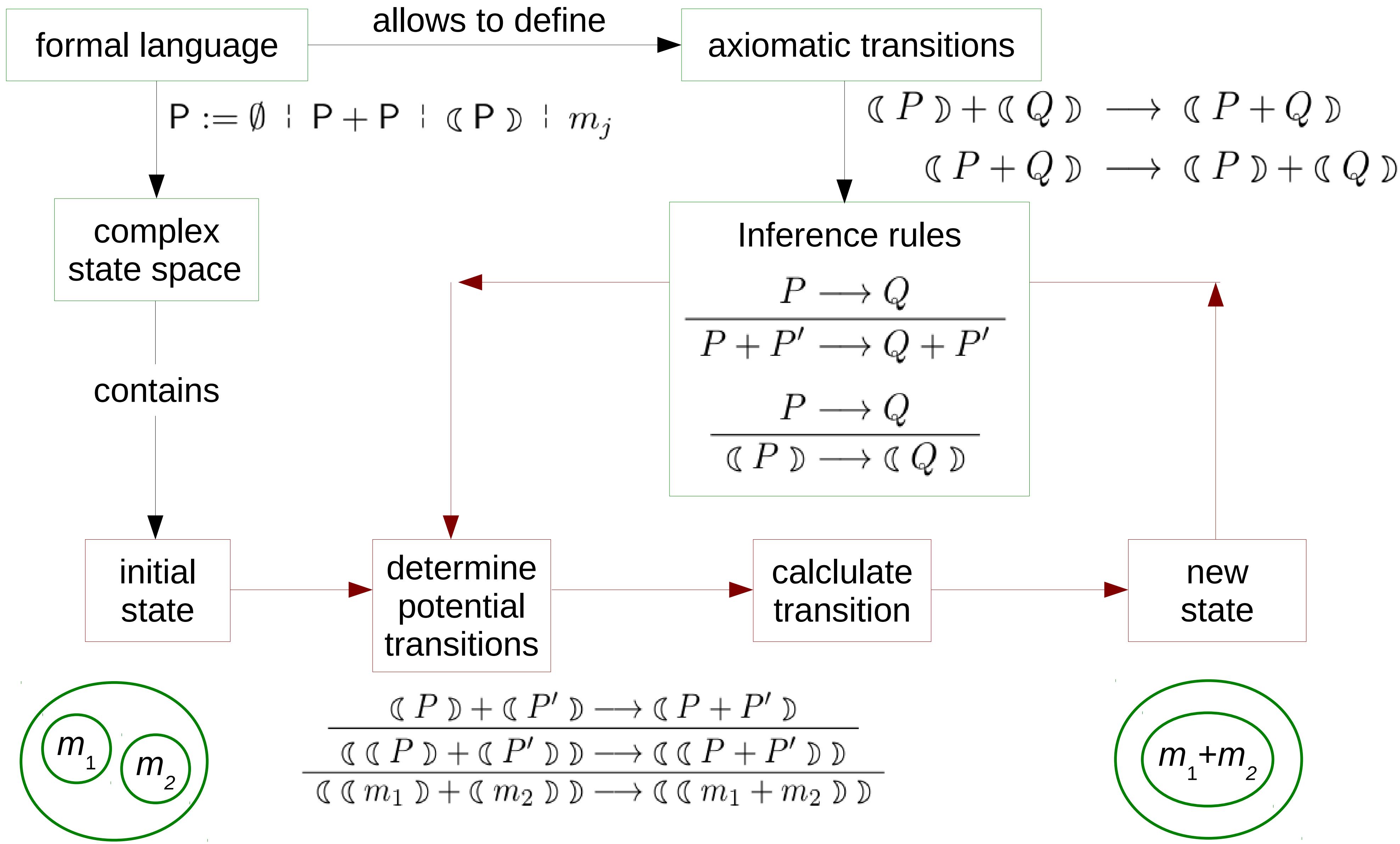
$$\llparenthesis\, P + Q \,\rrparenthesis \longrightarrow \llparenthesis\, P \,\rrparenthesis + \llparenthesis\, Q \,\rrparenthesis$$

e.g. *brane calculus*:

$$\mathrm{mate}_i \llparenthesis\, P \,\rrparenthesis + \mathrm{mate}_i^{\top} \llparenthesis\, Q \,\rrparenthesis \longrightarrow \llparenthesis\, P + Q \,\rrparenthesis$$

Cardelli, 2005

Example: Compartment Dynamics



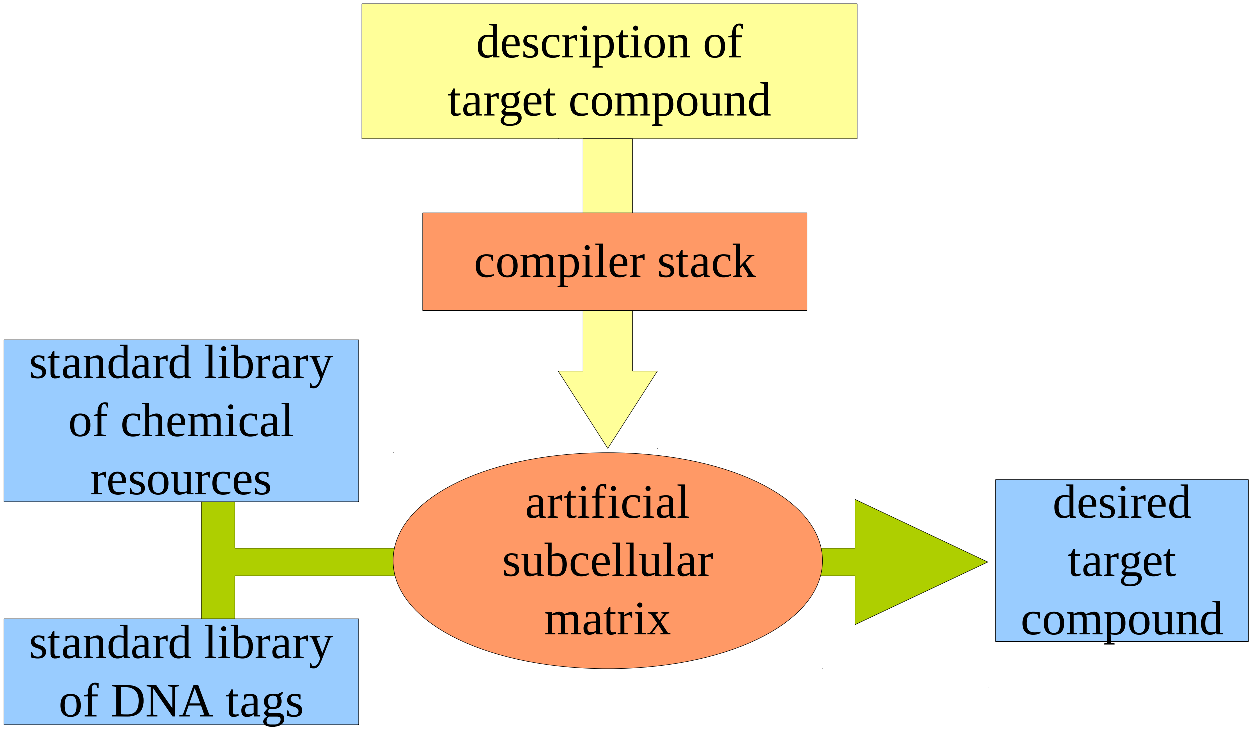
MATCHIT – Matrix for Chemical IT

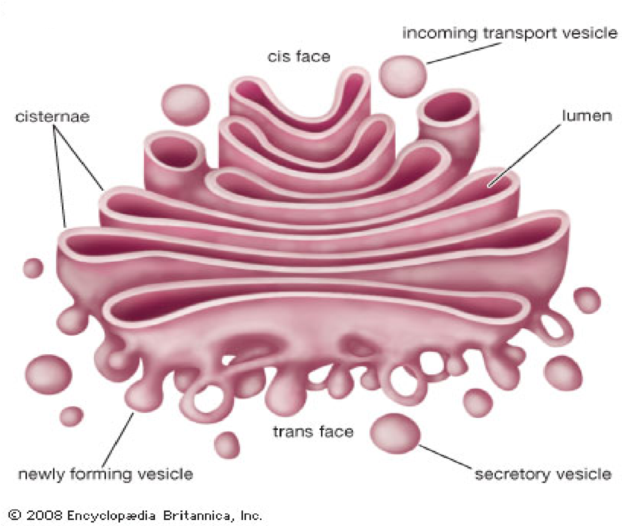
Aim:

Toward personal chemical manufactoring in an “artificial subcellular matrix”.

Methodology:

Integrating molecular computing and chemical production in microfluidic environments.

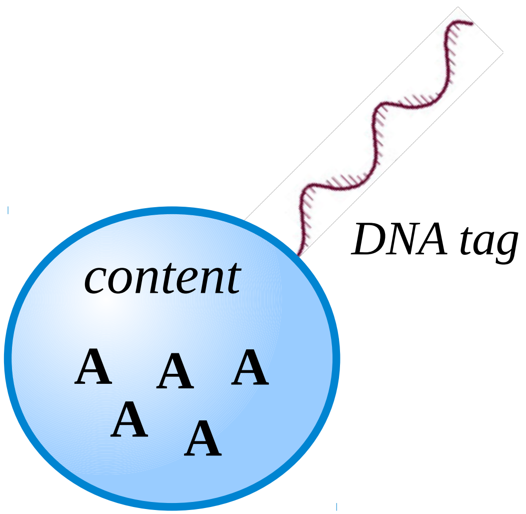




Golgi apparatus

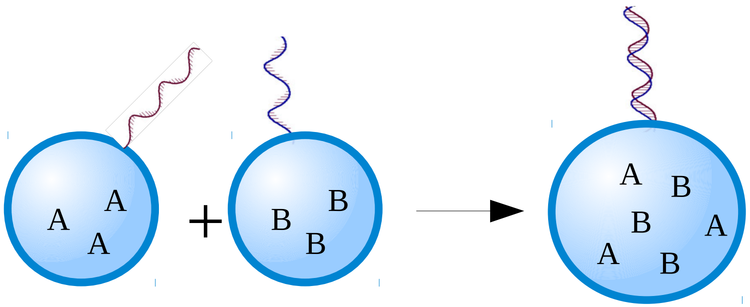
Programming Chemistry in Addressable Microcompartments

"Chemtainer" approach

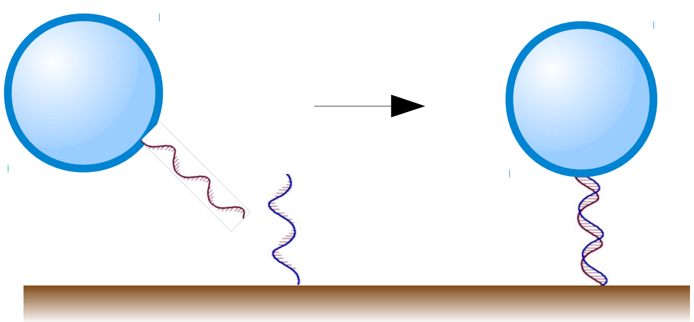


e.g. lipid vesicles,
oil droplets,
DNA nanocages...

DNA programmable chemtainer interactions

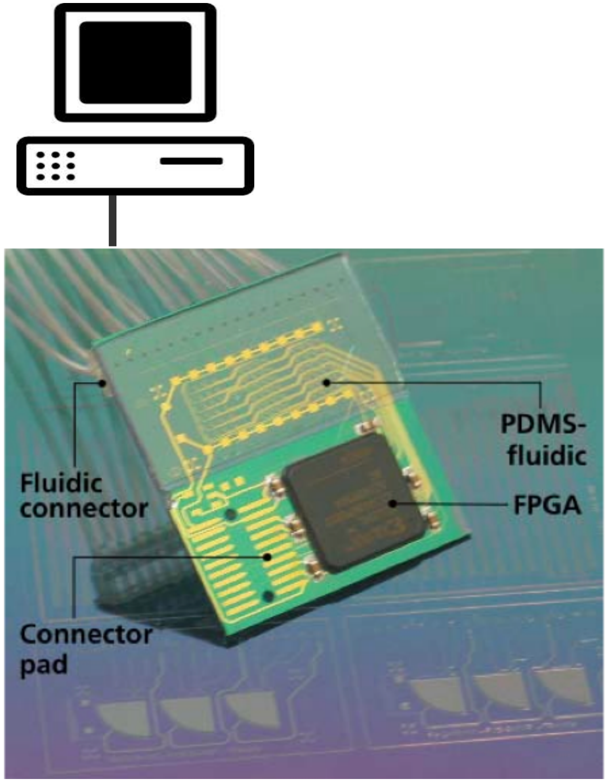


Chemtainer-chemtainer interaction



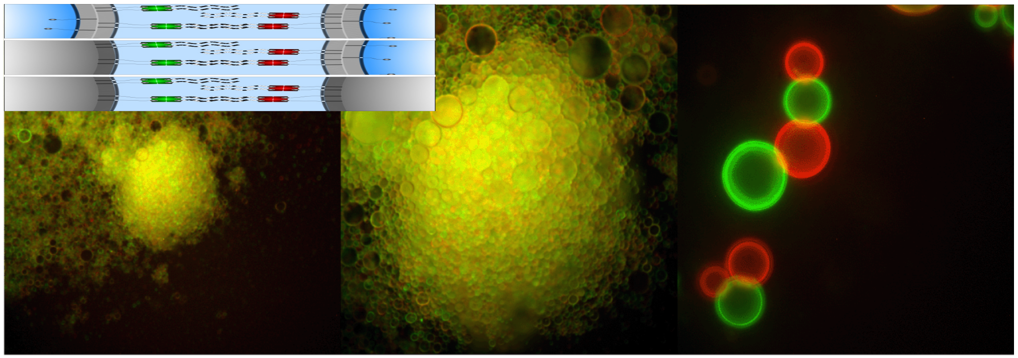
Chemtainer-matrix interaction

microfluidic embedding and computer control

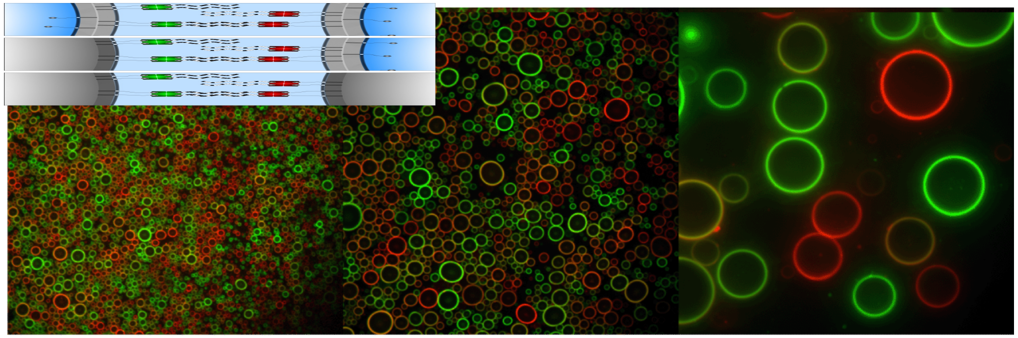


Experimental Chemtainer Interactions

Complementary tags

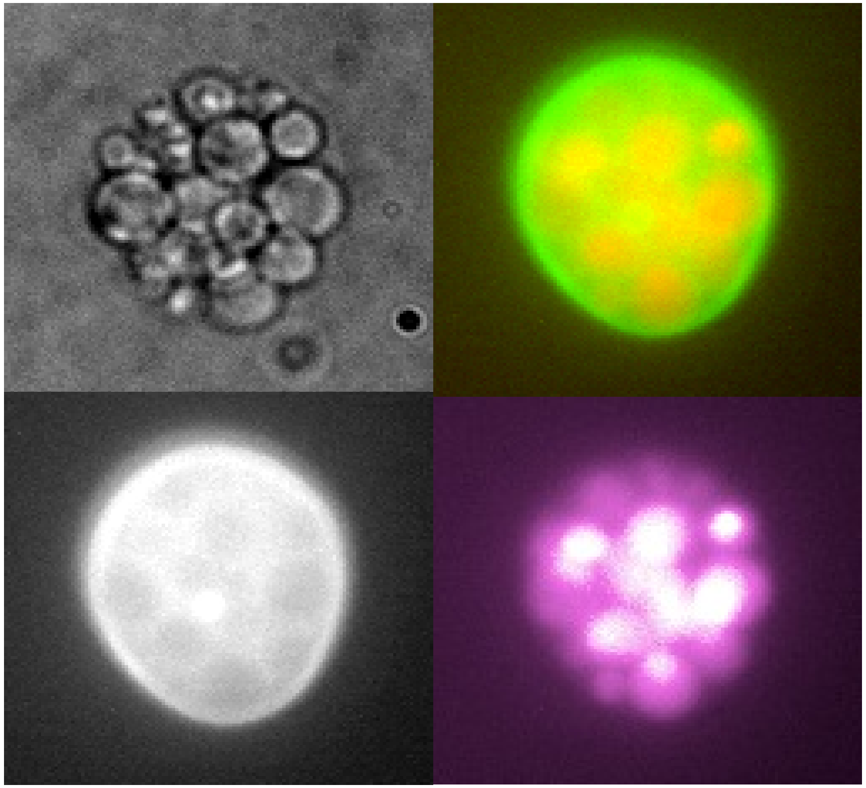


Non-complementary tags



4x 10x 100x

Hierarchical encapsulation

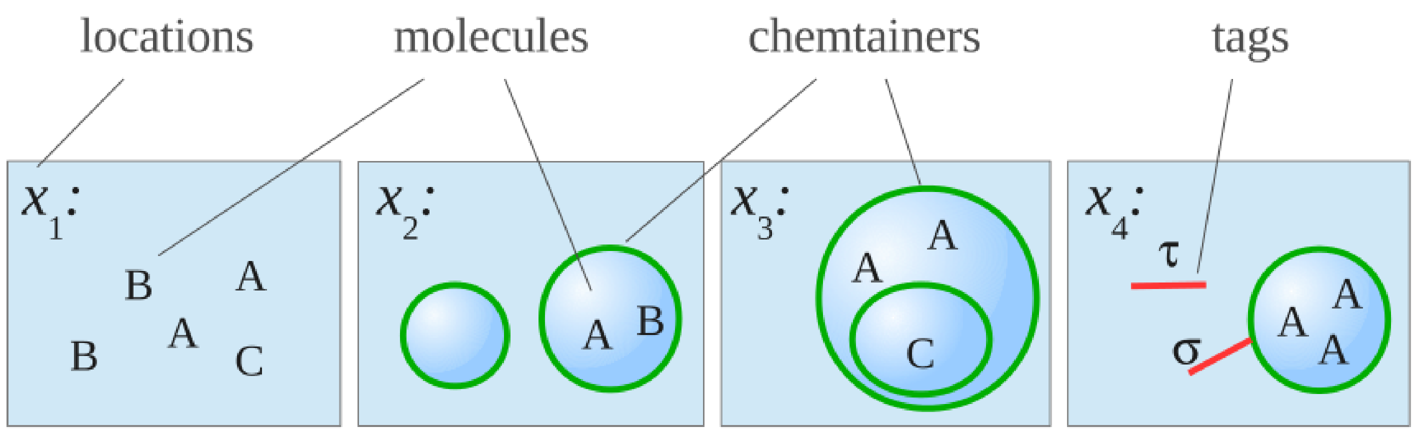


Hadorn, Bönzli, Sørensen, Fellermann, Eggenberger Hotz, Hanczyc; *PNAS* 109(47) 2012
Hadorn, Bönzli, Hanczyc, Eggenberger-Hotz; *PLOS One* 2012

Chemtainer Calculus: Grammar

System states are arrangements of localized molecules, chemtainers, and DNA tags

Example:



$$x_1 : 2A + 2B + C \circ x_2 : \llparenthesis\ \rrparenthesis + \llparenthesis A + B \rrparenthesis \circ x_3 : \llparenthesis 2A + \llparenthesis C \rrparenthesis \rrparenthesis \circ x_4 : \tau + \sigma \llparenthesis 3A \rrparenthesis$$

Grammar:

global state	$\mathsf{S} := \emptyset \mid \mathsf{S} \circ \mathsf{S} \mid x_i : \mathsf{P}$
local state	$\mathsf{P} := 0 \mid \mathsf{P} + \mathsf{P} \mid \mathsf{q}^* \llparenthesis \mathsf{P} \rrparenthesis \mid \mathsf{q} \mid m_j$
tag	$\mathsf{q} := \mathsf{s} \mid \mathsf{s}^* \triangleright \mathsf{s}^*$

Fellermann & Cardelli, *Interface* 2014 (in press)

Chemtainer Calculus: Transitions

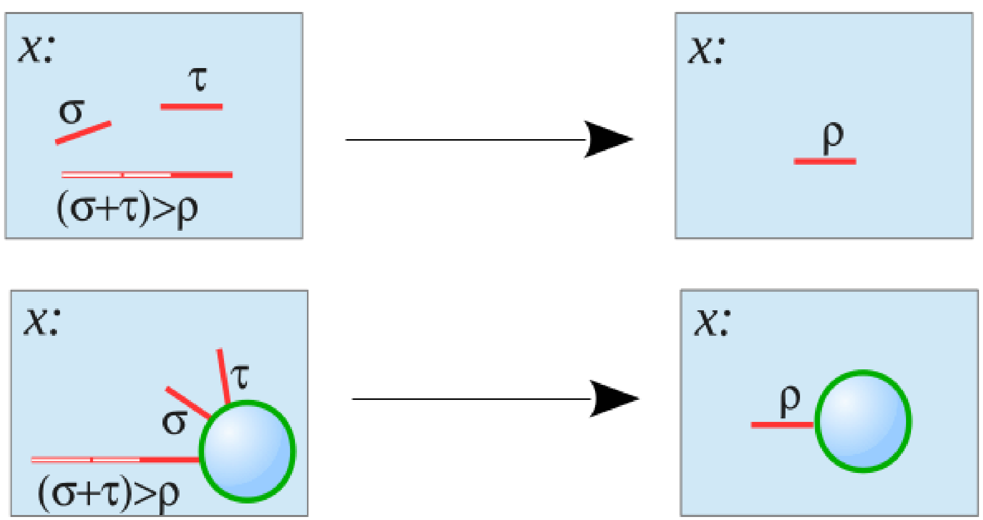
Autonomous transitions:

1. Application chemistry

$$P_i + M_i E^*_{x_i} \longrightarrow P_{i+1} + E_{x_i}$$

2. DNA *join & fork* gates $s_1^* \triangleright s_2^*$

$$s_1^* \triangleright s_2^* \; + \; s_1^* \longrightarrow s_2^*$$



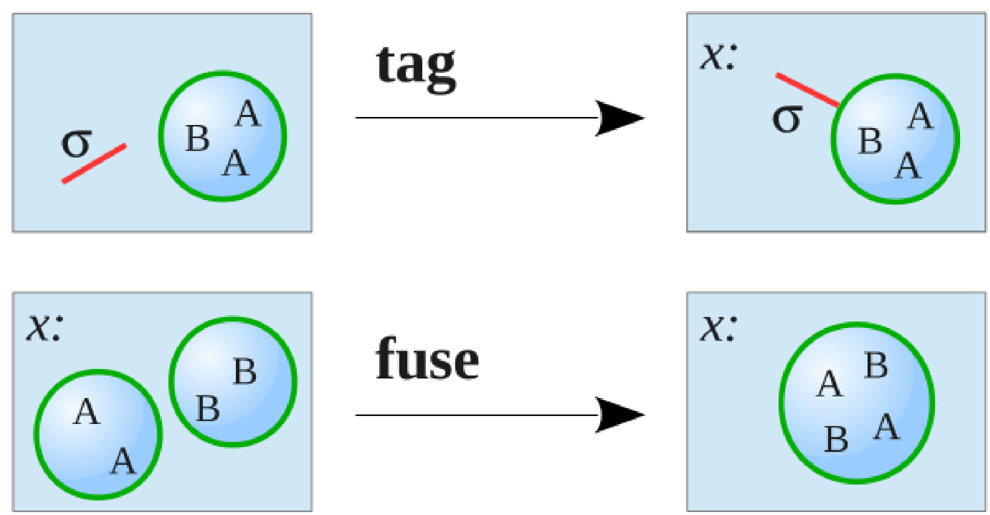
Induced transitions:

Triggered by microfluidic control

6 - 8 operations, e.g.

tag:

$$x : s + q^* \llparenthesis\, P \,\rrparenthesis \longrightarrow x : (s + q^*) \llparenthesis\, P \,\rrparenthesis$$



Fellermann & Cardelli, *Interface* 2014 (in press)

Chemtainer Calculus: Language

- Domain specific language
  specified in non-deterministic structural operational semantics

- Sequential imperative language

$$\frac{\langle \pi, S'' \rangle \longrightarrow S \qquad I : S' \longrightarrow S''}{\langle I; \pi, S' \rangle \longrightarrow S}$$

- Parallel composition

$$\frac{\langle \pi', S' \rangle \longrightarrow \bar{S}' \qquad \langle \pi'', S'' \rangle \longrightarrow \bar{S}''}{\langle \pi' | \pi'', S' \circ S'' \rangle \longrightarrow \bar{S}' \circ \bar{S}''}$$

- Spontaneous transitions may occurr any time during execution

$$\frac{\langle \pi, S \rangle \longrightarrow S' \qquad S' \longrightarrow S''}{\langle \pi, S \rangle \longrightarrow S''}$$

Fellermann & Cardelli, *Interface* 2014 (in press)

Chemtainer Calculus: Programming

Program set $\ni$ **Program**

Π_{min} Π_{wrap} Π_{burst}

Instruction
Instruction
Instruction
Instruction

I_{min} I_{wrap} I_{burst}

Instruction sets $\ni$ $\boxed{\text{Instruction}}$

Instruction
Instruction

State n $\xrightarrow{\text{Transitions}}$ States n+1

Initial state $\xrightarrow{\text{Program Set}}$ $\omega^+(\Pi_X)$ Constructive closure

$$\omega^+(\Pi_{\mathbf{min}}) \subset \omega^+(\Pi_{\mathbf{wrap}}) \subset \omega^+(\Pi_{\mathbf{burst}}) = L(G_S)$$

Constructive proof.

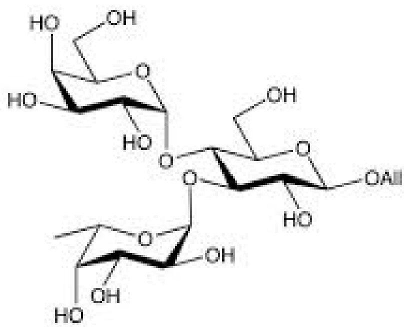
We can *automatically derive programs* to build any given target state.

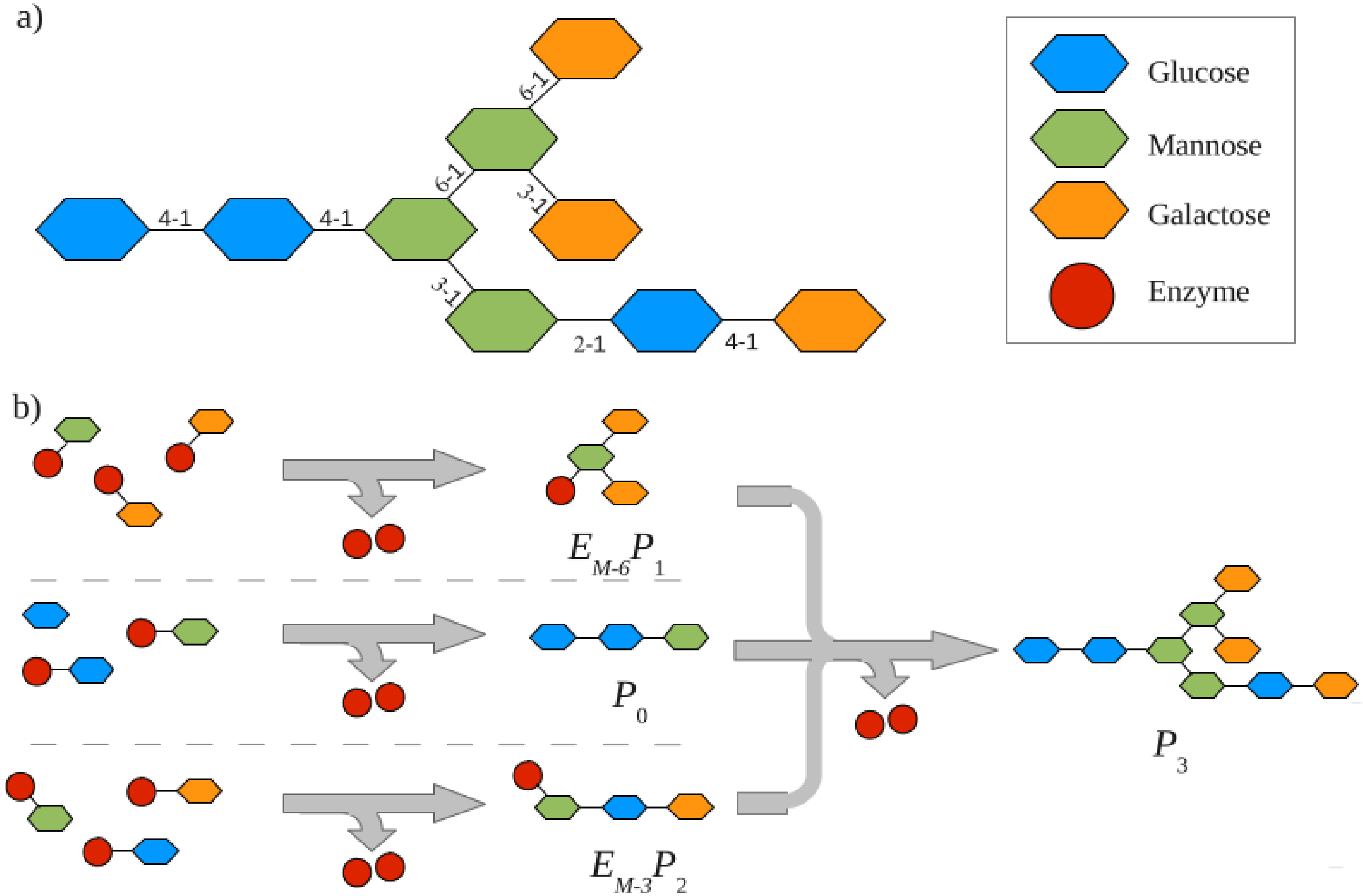
Fellermann & Cardelli, *Interface* 2014 (in press)

Branches oligo-saccharides (e.g. antibodies)

Limited number of monomers and linking sites

Combinatorial explosion of targets – and side reactions :-(





Weyland, Füchslin, Sorek, Lancet, Fellermann, Rasmussen, *Comp. Math. Meht. Med.* 2013

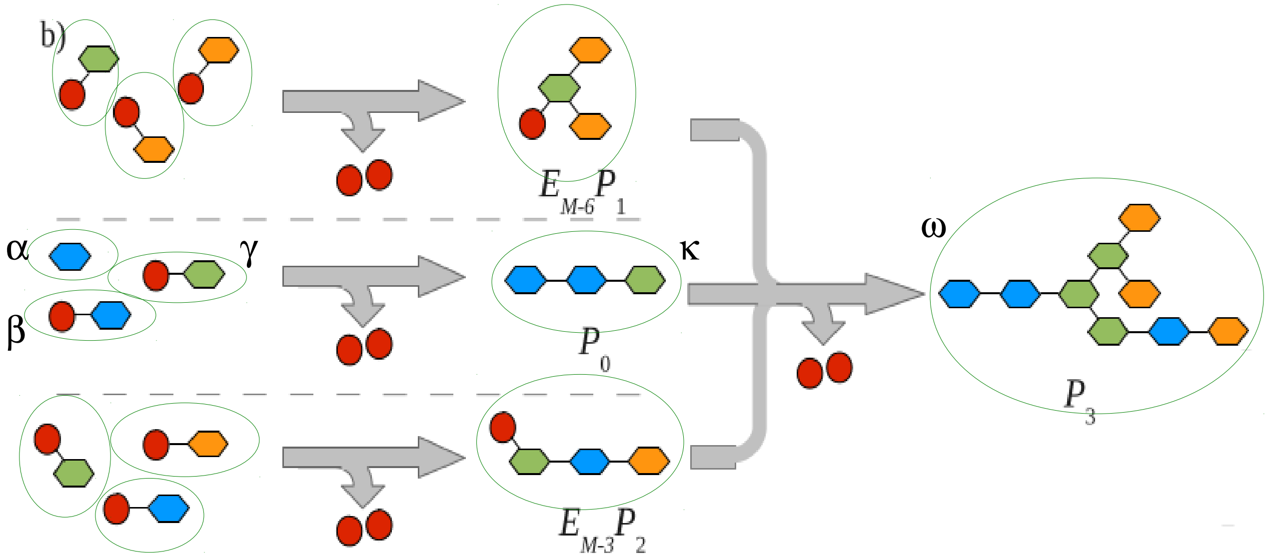
Chemtainer Calculus: Programmable Synthesis

Reaction cascades

$$P_i + M_i E^*_{x_i} \longrightarrow P_{i+1} + E_{x_i}$$

are encoded in DNA gates:

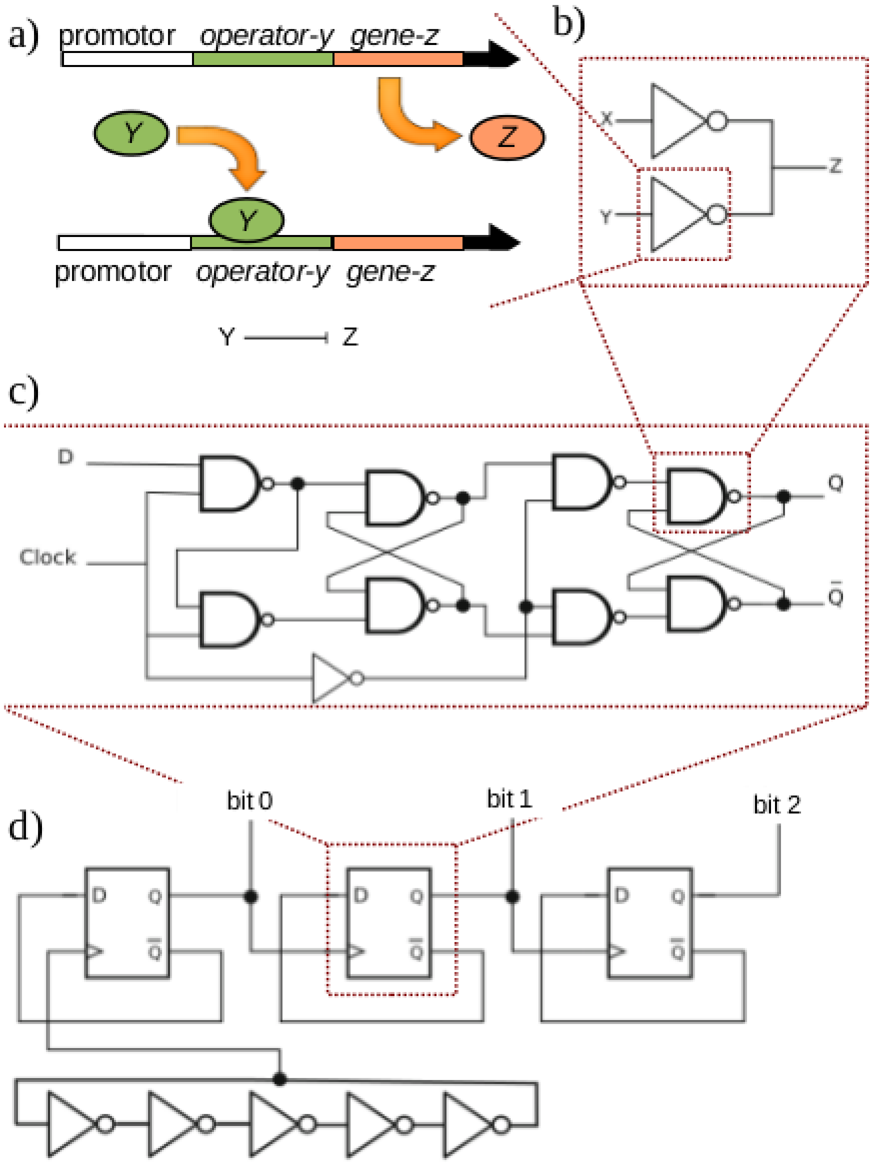
$$\begin{aligned}
(\alpha + \beta + \gamma) &\rhd \kappa \\
(\delta + \epsilon + \zeta) &\rhd \lambda \\
(\eta + \theta + \iota) &\rhd \mu \\
(\kappa + \lambda + \mu) &\rhd \omega
\end{aligned}$$



DNA computing reports fusion-induced chemical reactions on the chemtainer surface:

$$\begin{array}{ll}
 & x_0 : (\alpha+\beta+\gamma) \rhd \kappa \llparenthesis\ \rrparenthesis \circ\ x_S : \alpha \llparenthesis \mathrm{Gal} \rrparenthesis + \beta \llparenthesis \mathrm{E}^*_{\text{Gal-4}}\mathrm{Gal} \rrparenthesis + \gamma \llparenthesis \mathrm{E}^*_{\text{Gal-4}}\mathrm{Man} \rrparenthesis \\
\mathbf{move}(\alpha, x_S, x_0) & x_0 : (\alpha+\beta+\gamma) \rhd \kappa \llparenthesis\ \rrparenthesis + \alpha \llparenthesis \mathrm{Gal} \rrparenthesis \circ\ x_S : \beta \llparenthesis \mathrm{E}^*_{\text{Gal-4}}\mathrm{Gal} \rrparenthesis + \gamma \llparenthesis \mathrm{E}^*_{\text{Gal-4}}\mathrm{Man} \rrparenthesis \\
\mathbf{move}(\beta, x_S, x_0) & x_0 : (\alpha+\beta+\gamma) \rhd \kappa \llparenthesis\ \rrparenthesis + \alpha \llparenthesis \mathrm{Gal} \rrparenthesis + \beta \llparenthesis \mathrm{E}^*_{\text{Gal-4}}\mathrm{Gal} \rrparenthesis \circ\ x_S : \gamma \llparenthesis \mathrm{E}^*_{\text{Gal-4}}\mathrm{Man} \rrparenthesis \\
\mathbf{move}(\gamma, x_S, x_0) & x_0 : (\alpha+\beta+\gamma) \rhd \kappa \llparenthesis\ \rrparenthesis + \alpha \llparenthesis \mathrm{Gal} \rrparenthesis + \beta \llparenthesis \mathrm{E}^*_{\text{Gal-4}}\mathrm{Gal} \rrparenthesis + \gamma \llparenthesis \mathrm{E}^*_{\text{Gal-4}}\mathrm{Man} \rrparenthesis \\
\mathbf{fuse}(x_0) & x_0 : (\alpha + (\alpha+\beta+\gamma) \rhd \kappa) \llparenthesis \mathrm{Gal} \rrparenthesis + \beta \llparenthesis \mathrm{E}^*_{\text{Gal-4}}\mathrm{Gal} \rrparenthesis + \gamma \llparenthesis \mathrm{E}^*_{\text{Gal-4}}\mathrm{Man} \rrparenthesis \\
\mathbf{fuse}(x_0) & x_0 : (\alpha + \beta + (\alpha+\beta+\gamma) \rhd \kappa) \llparenthesis \mathrm{Gal} + \mathrm{E}^*_{\text{Gal-4}}\mathrm{Gal} \rrparenthesis + \gamma \llparenthesis \mathrm{E}^*_{\text{Gal-4}}\mathrm{Man} \rrparenthesis \\
\mathbf{fuse}(x_0) & x_0 : (\alpha + \beta + \gamma + (\alpha+\beta+\gamma) \rhd \kappa) \llparenthesis \mathrm{Gal} + \mathrm{E}^*_{\text{Gal-4}}\mathrm{Gal} + \mathrm{E}^*_{\text{Gal-4}}\mathrm{Man} \rrparenthesis \\
 & x_0 : \kappa \llparenthesis P_0 + 2\,\mathrm{E}_{\text{Gal-4}} \rrparenthesis \\
\mathbf{move}(\kappa, x_0, x_S) & x_S : \kappa \llparenthesis P_0 + 2\,\mathrm{E}_{\text{Gal-4}} \rrparenthesis
\end{array}$$

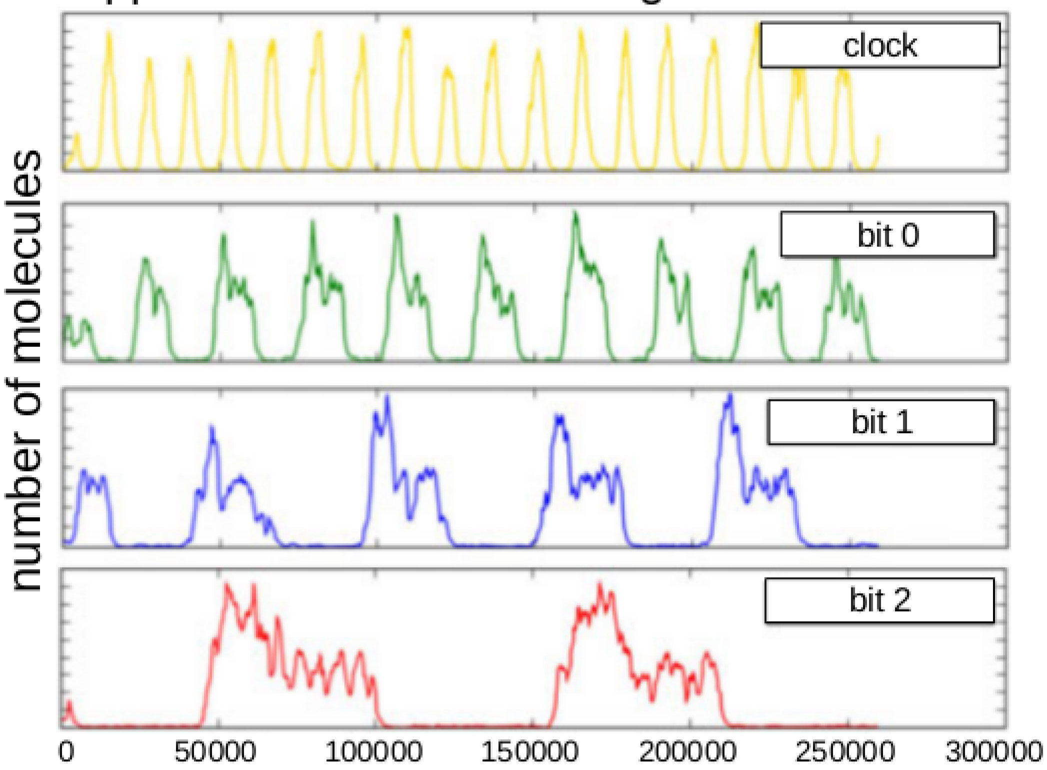
Distributed Molecular Computing



Complex circuits builts from gene regulatory networks

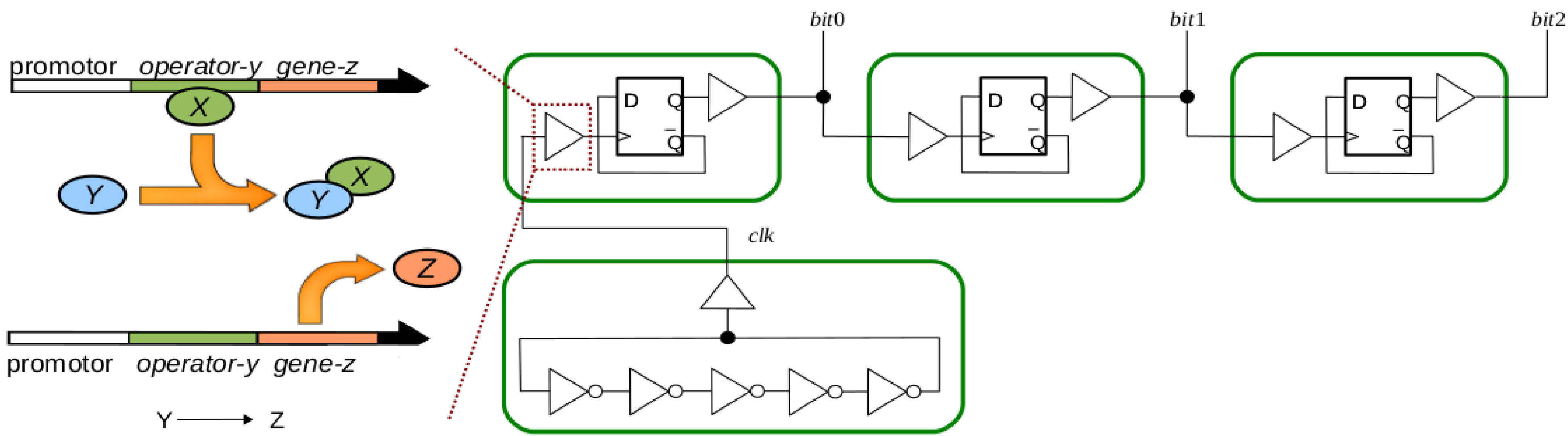
Smaldon et al. *Syst. Synth. Biol.* **4**(3), 2010

Ripple counter with 5-bit ring oscillator clock



Distributed Molecular Computing

Using chemtainer calculus, we compile and wire a distributed implementation from few standard parts.



Example wiring of a flipflop by fusing transducers:

$$ {}^{\alpha^\top+\beta^\top+\gamma^\top}\llparenthesis\ \rrparenthesis + {}^{\alpha}\llparenthesis\, F(x,y)\,\rrparenthesis + {}^{\beta}\llparenthesis\, T(clk,x)\,\rrparenthesis + {}^{\gamma}\llparenthesis\, T(y,bit0)\,\rrparenthesis $$

$$ \longrightarrow {}^{\alpha^{||}+\beta^{||}+\gamma^{||}}\llparenthesis\, F(clk,bit0)\,\rrparenthesis $$

Fellermann, Krasnogor, CiE proceedings 2014 (accepted)
Fellermann, Hadorn, Füchslin, Krasnogor, *JETC* 2014 (submitted)

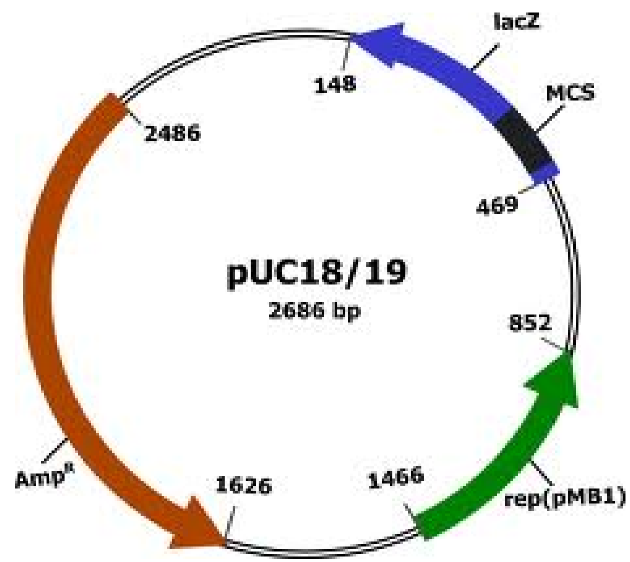
Calculus for DNA manipulation

Formal language to denote domains on plasmids/chromosome:

$$
\begin{aligned}
\mathsf{STATE} &:= 0 \mid \mathsf{STATE} + \mathsf{STATE} \mid \mathsf{DNA} \mid \mathsf{RNA} \mid \mathsf{PROT} \\
\mathsf{DNA} &:= [\mathsf{DSEQ}] \mid < \mathsf{DSEQ} > \\
\mathsf{DSEQ} &:= \epsilon \mid \mathsf{DSEQ.DSEQ} \mid \mathsf{DSEQ\hat{}} \mid \{\mathsf{RNA}\}\mathsf{DSEQ} \mid \mathsf{DOM} \\
\mathsf{RNA} &:= [\mathsf{SEQ}] \mid < \mathsf{SEQ} > \\
\mathsf{SEQ} &:= \epsilon \mid \mathsf{SEQ.SEQ} \mid \{\}\mathsf{SEQ} \mid \mathsf{DOM} \\
\mathsf{DOM} &:= \mathsf{TYPE} : \mathsf{IDENT}
\end{aligned}
$$

Domain can be promoters, operators, terminators, introns, restriction sites, etc.

```
pUC19 = <P:x.G:lacZ.G:AmpR.T:y.P:z.pMB1.T:y>
```



Calculus for DNA manipulation

Axiomatic transitions for

- Translation

$$\texttt{P:}x.s \longrightarrow \texttt{P:}x.\{\epsilon\}s$$
$$\{r\}\texttt{O:}x.s \longrightarrow \texttt{O:}x.\{r\}s$$
$$\{r\}\texttt{T:}x \longrightarrow \texttt{T:}x + [r]$$

- Transcription

$$\texttt{B:}x.r \longrightarrow \texttt{B:}x.\{\}r$$
$$\{\}\texttt{G:}x.r \longrightarrow \texttt{G:}x.\{\}r + x$$

- Splicing

$$r.\texttt{I:}x.r' \longrightarrow r.r'$$

- Operon regulation

$$x + [s.\texttt{O:}x.s'] \longrightarrow [s.\emptyset\texttt{:}x.s']$$

- Restriction
- Recombination

- Transposons, etc.

Conclusion

- Physical simulations give detailed insight into biological systems
  - prediction
  - verification
  - explaining
  - design of experiments
  - design of systems

- Formal calculi allow to apply physical simulations to complex states
  - applicable to unbounded state spaces
  - capture logical organization
  - allow for analytic treatment (proofs!)

Thanks for you attention!

Questions?