Executable Biology

Tom Henzinger

IST Austria

Executable Biology

Tom Henzinger

IST Austria

Modeling of biological objects (e.g. cells) and processes (e.g. pathways) as interacting computer programs.

Biology	Computer Science
-complex natural systems	-complex engineered systems

Biology	Computer Science
-complex natural systems	-complex engineered systems
-primacy of system	-primacy of method (“algorithm”)

Biology	Computer Science
-complex natural systems	-complex engineered systems
-primacy of system	-primacy of method ("algorithm")
-*numeric; stochastic:* continuous variables	-*symbolic; causalistic:* discrete events
Science of *approximation*: bounds errors	Science of *abstraction*: preserves properties

Computational Science

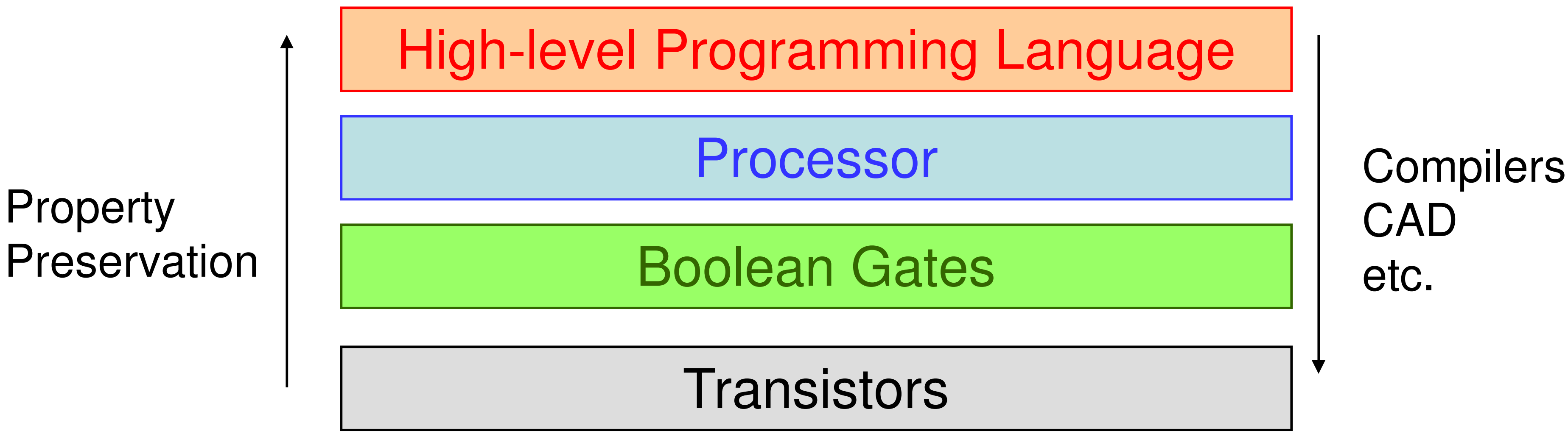
Biology	Computer Science
-complex natural systems	-complex engineered systems
-primacy of system	-primacy of method ("algorithm")
-*numeric; stochastic:* continuous variables	-*symbolic; causalistic:* discrete events
Science of *approximation*: bounds errors	Science of *abstraction*: preserves properties

The Essence of Computer Science

1. Algorithms

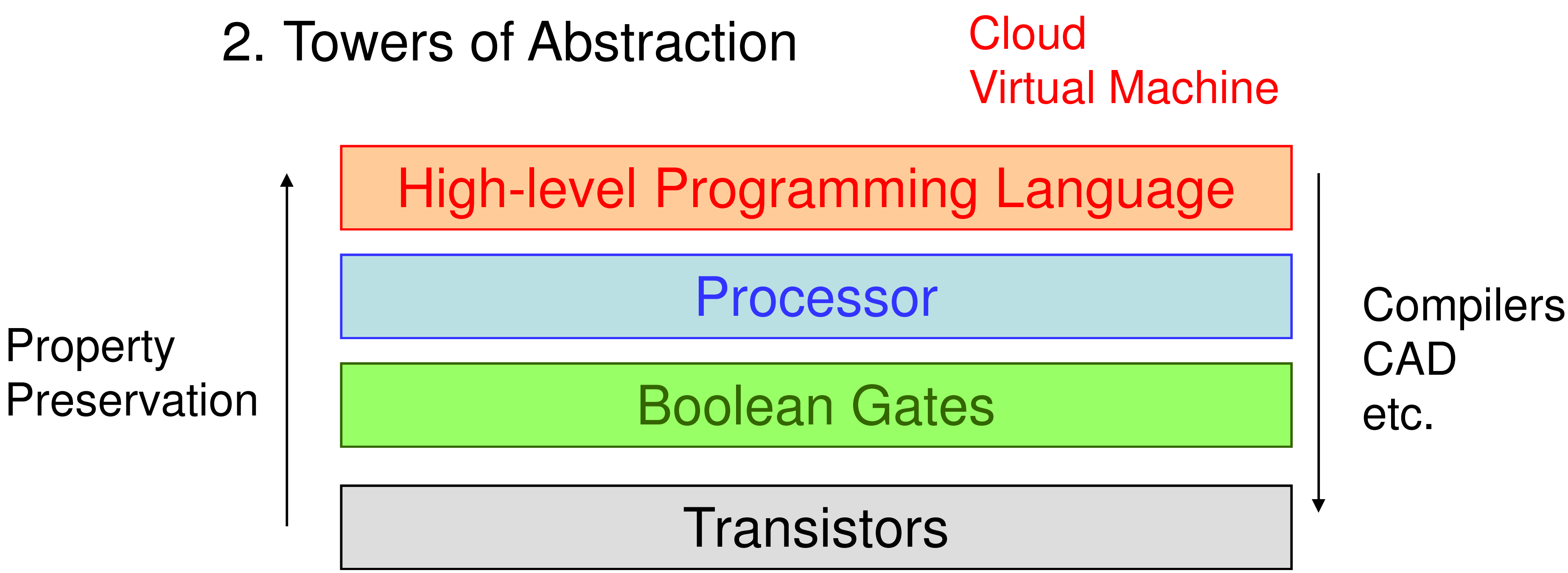
The Essence of Computer Science

1. Algorithms
2. Towers of Abstraction

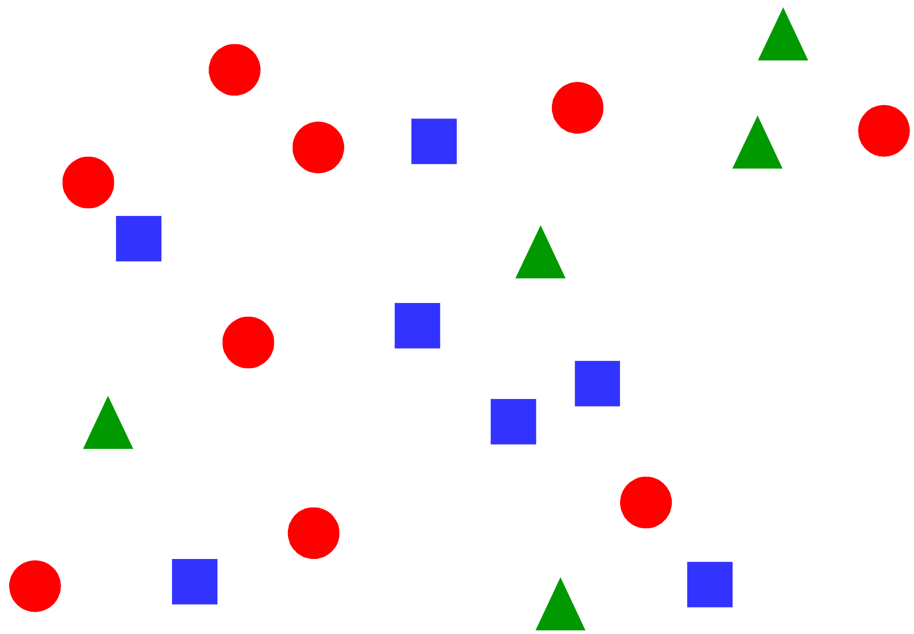


The Essence of Computer Science

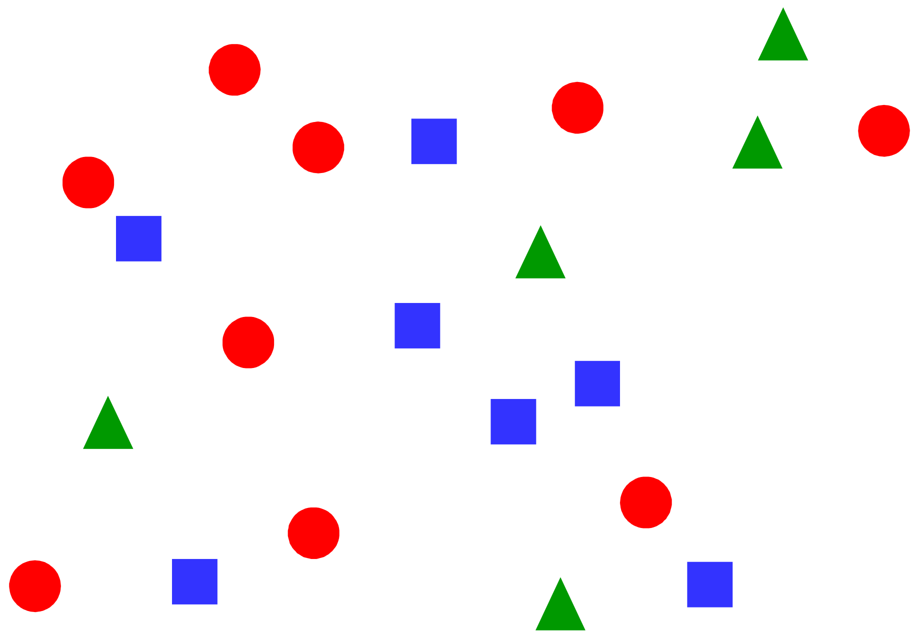
1. Algorithms
2. Towers of Abstraction

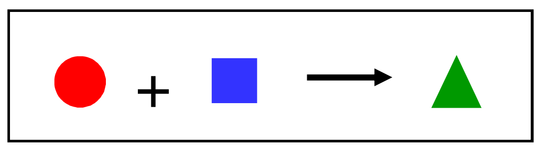


Markovian Population Models

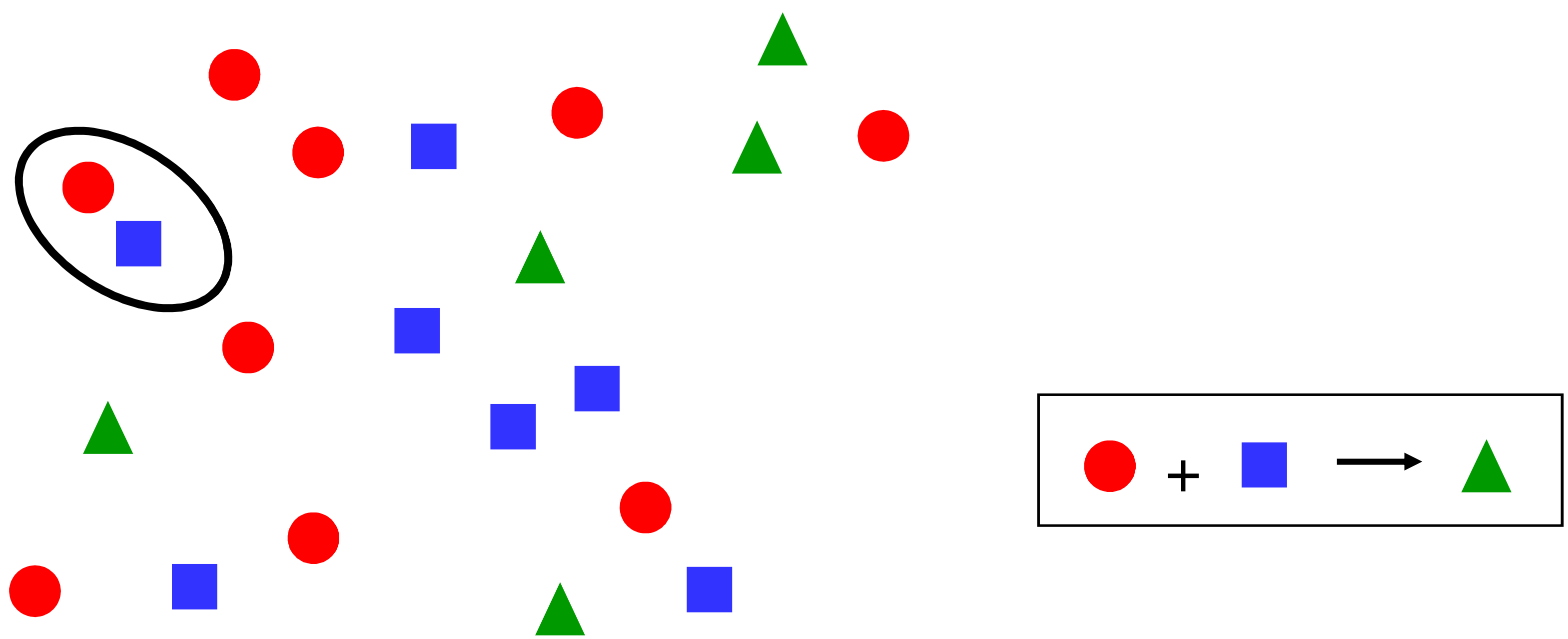


Markovian Population Models

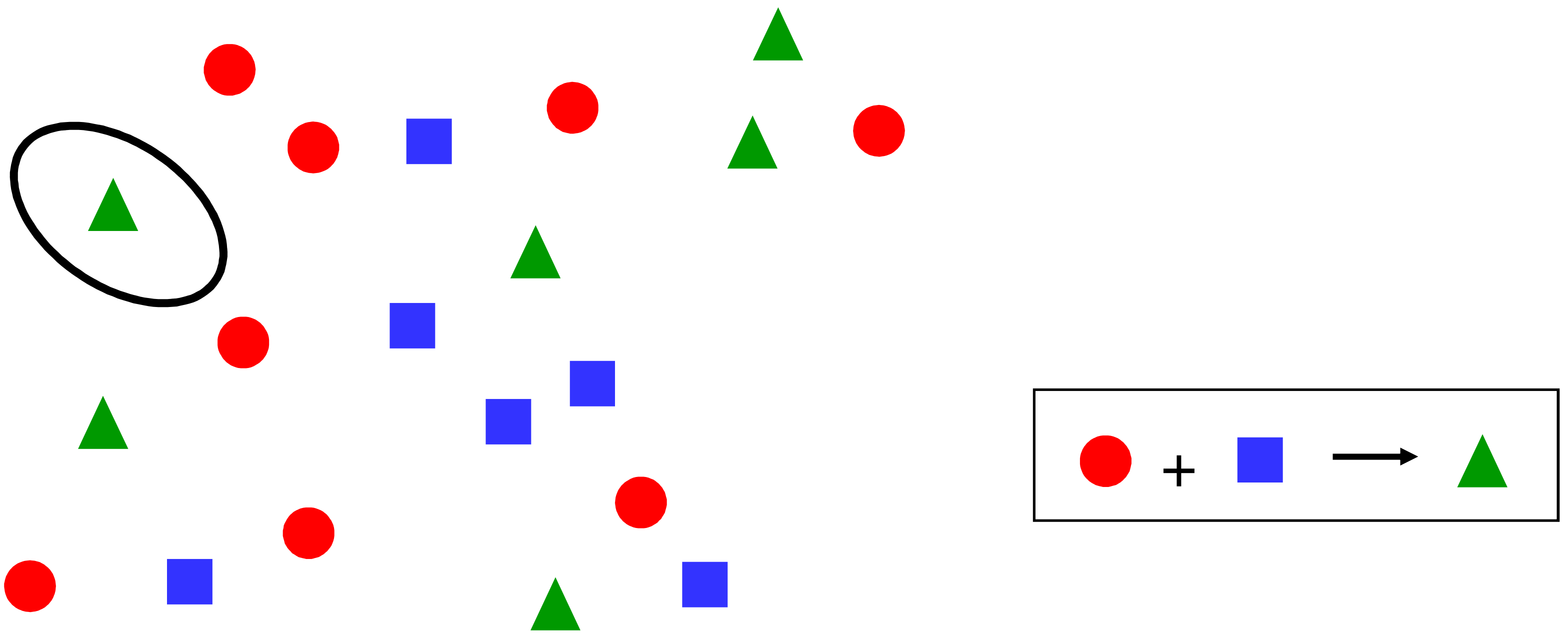




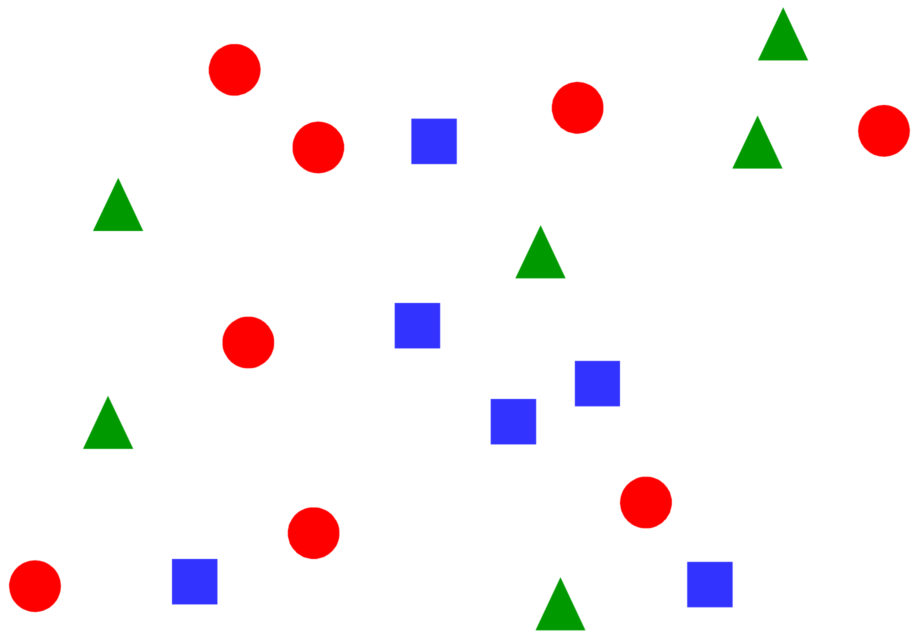
Markovian Population Models



Markovian Population Models

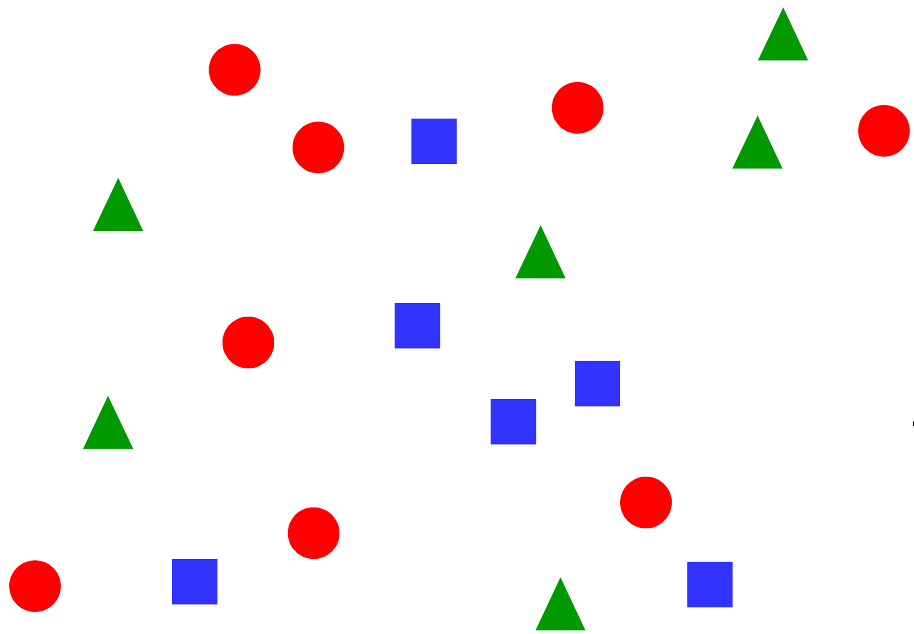


Markovian Population Models



State: (8 , 6 , 6)

Markovian Population Models

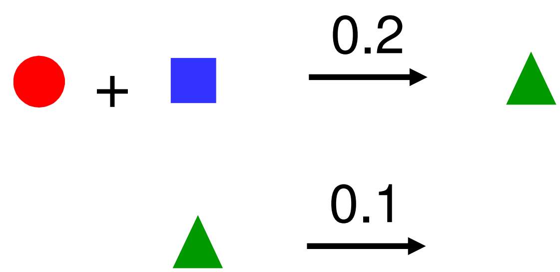


(9 , 7 , 5)

Transition: $\downarrow$

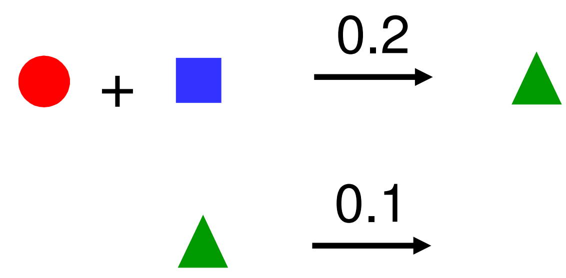
(8 , 6 , 6)

Markovian Population Models

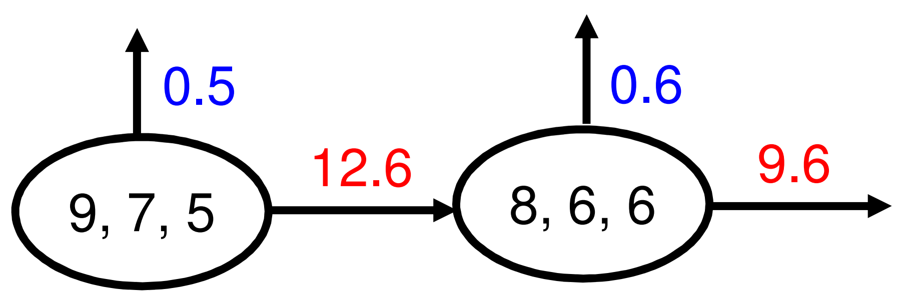


Syntax: stoichiometric equations (finite object)

Markovian Population Models



Syntax: stoichiometric equations (finite object)



Semantics: CTMC (infinite object)

Syntax: Guarded Commands

Dimension n: state $(x_1, ..., x_n)$ 2 S
state space $S = N^n$

$n = 3$

Finite set of guarded commands:

each consists of
1. guard G µ S
2. update function u: G ! S
3. rate function α: G ! R^+

G_1: x_1 ¸ 1 Æ x_2 ¸ 1
$u_1(x_1,x_2,x_3) = (x_1-1,\ x_2-1,\ x_3+1)$
$\alpha_1(x_1,x_2,x_3) = 0.2$ ¢ x_1 ¢ x_2

G_3: x_3 ¸ 1
$u_1(x_1,x_2,x_3) = (x_1,\ x_2,\ x_3-1)$
$\alpha_1(x_1,x_2,x_3) = 0.1$ ¢ x_3

Syntax matters!

1. Composition and compositionality
2. Expressiveness and succinctness
3. Executability
4. Encapsulation and abstraction

e.g. Petri nets: not compositional
Stoichometric equations: not expressive
Rate matrices: not succinct
Differential equations: not executable

Guarded Commands

-compositional

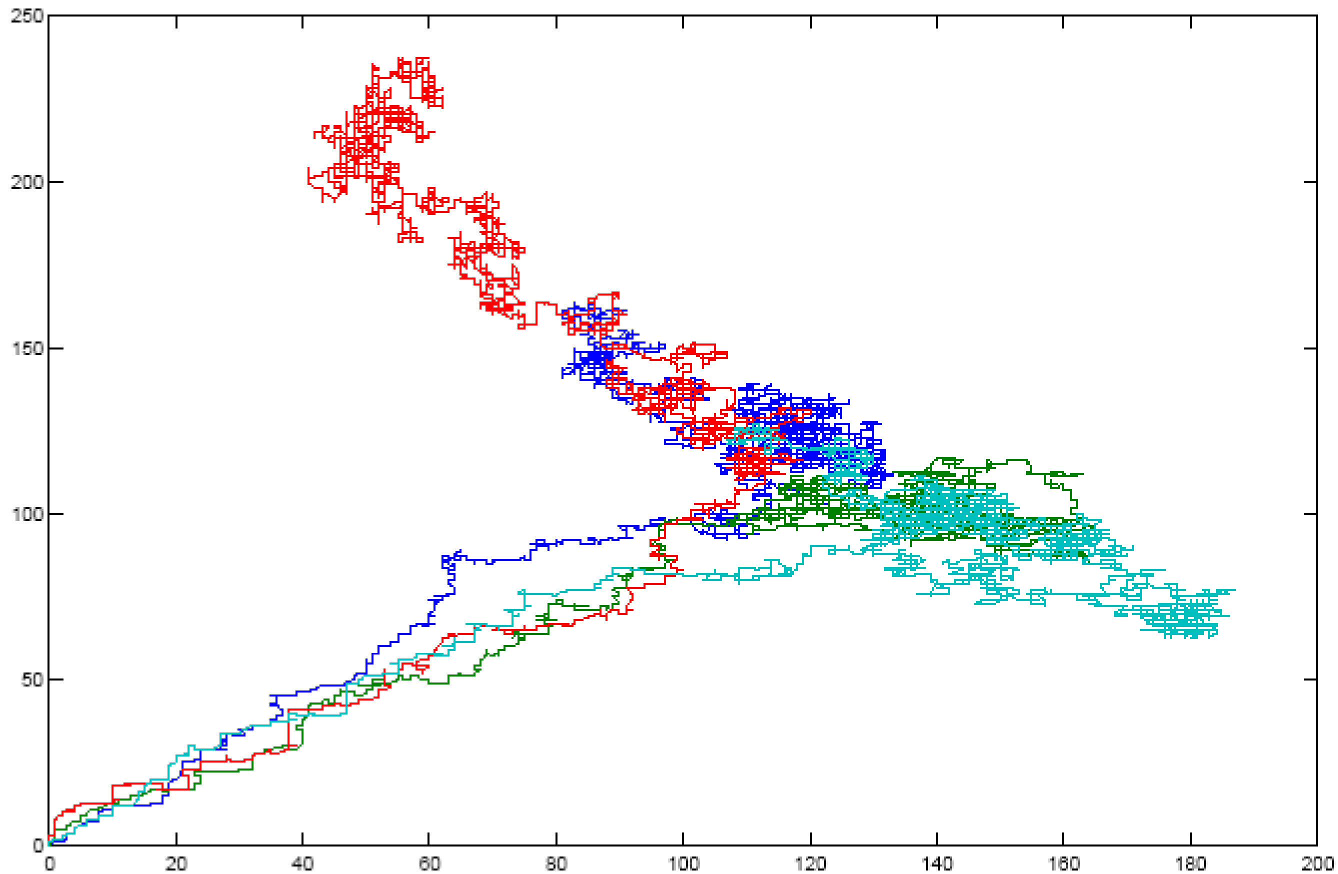
-expressive and succinct

-executable

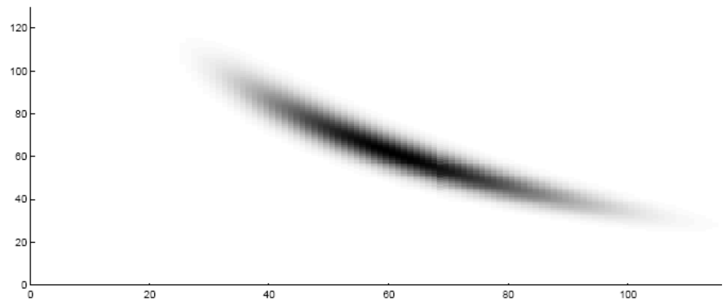
-counterexample-guided abstraction refinement

-model checking

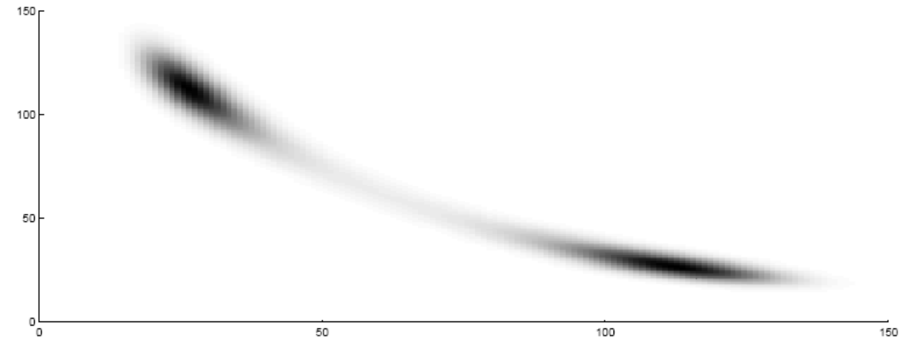
Genetic Toggle Switch: Four Simulation Runs



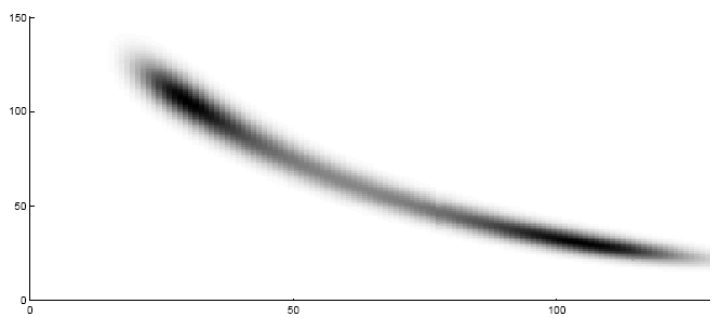
Genetic Toggle Switch: Model Checking [DHMW09]



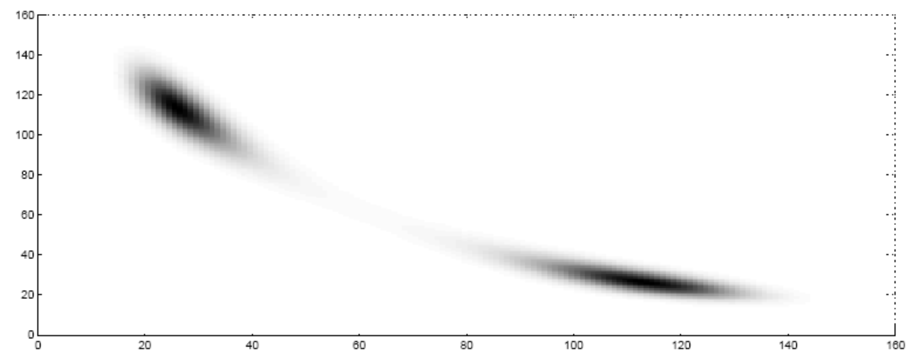
t = 5000



t = 30000



t = 15000



t = 50000

Bacteriophage λ Model [DHMW09]

Desired precision:	3×10^{-6}
Model checking:	55 min runtime
Gillespie simulation ($\beta = 0.95$):	67 h runtime (3×10^{8} runs)

Model Checking CTMCs [DHMW09]

Transient distributions can be efficiently approximated by combining adaptive uniformization with ideas from Computer Science:

1. on-the-fly state space exploration
2. removal of low-probability states

Model Checking CTMCs [DHMW09]

Transient distributions can be efficiently approximated by combining adaptive uniformization with ideas from Computer Science:

1. on-the-fly state space exploration
2. removal of low-probability states

Executable Syntax

Sliding Window Abstraction

Opportunity and Challenges

1. Computer Science can contribute to biology far more than *algorithms* and *tools*, e.g. modeling principles such as

   reactivity, *compositionality*, *executability*, *hybrid systems*, *property preserving abstraction*, and *model checking*.

Example: bounded asynchrony [FHMP08]

Opportunity and Challenges

1. Computer Science can contribute to biology far more than *algorithms* and *tools*, e.g. modeling principles such as

   reactivity, *compositionality*, *executability*, *hybrid systems*, *property preserving abstraction*, and *model checking*.

2. Computer Science needs to develop more *quantitative* models.

B versus R

Opportunity and Challenges

1. Computer Science can contribute to biology far more than *algorithms* and *tools*, e.g. modeling principles such as

   reactivity, *compositionality*, *executability*, *hybrid systems*, *property preserving abstraction*, and *model checking*.

2. Computer Science needs to develop more *quantitative* models.

3. Training of scientists!!!

IST Austria

8 biologists, 4 computer scientists, 1 PhD program