Molecules as Automata

Luca Cardelli

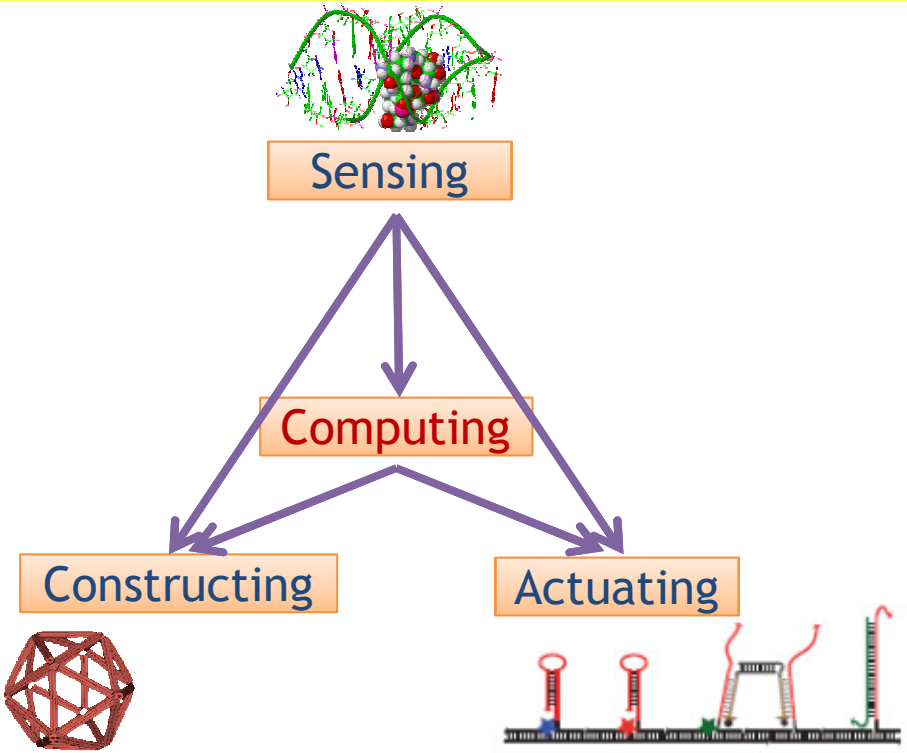
Microsoft Research

BIC-TA Beijing, 2009-10-17

http://lucacardelli.name

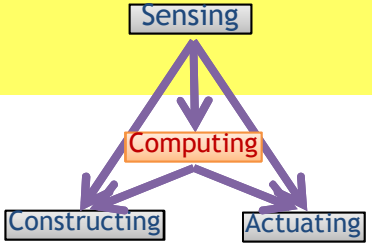
Nano Tasks

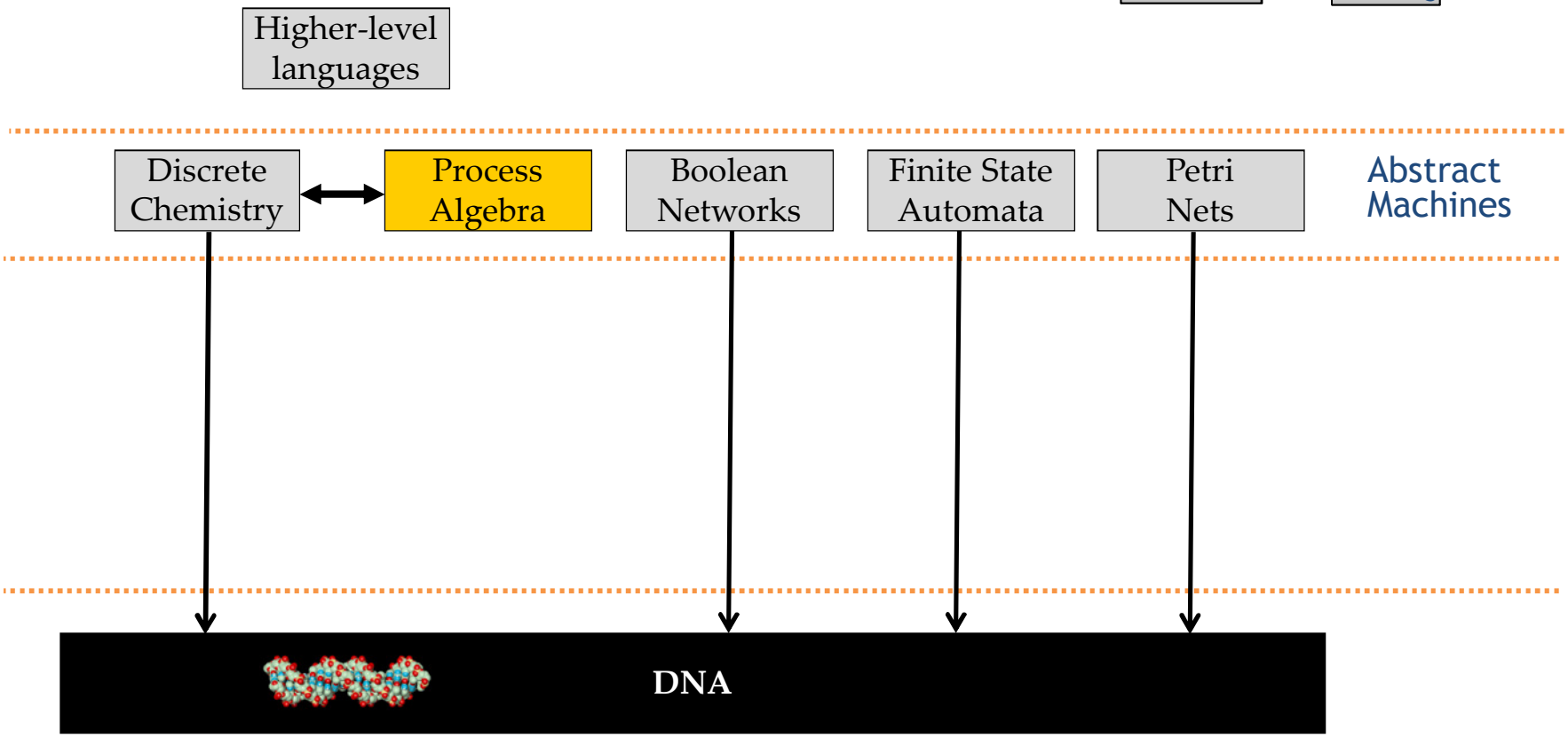
- Sensing
  - Reacting to forces
  - Binding to molecules
- Actuating
  - Releasing molecules
  - Producing forces
- Constructing
  - Spontaneous self-assembly
  - Catalyzed by stimuli
- Computing
  - All that under 'program control'
  - Analog: Signal Filtering, Amplification
  - Digital: Logical gates



- Nucleic Acids (DNA/RNA)
  - Probably the only materials that can perform all these functions.
  - Technology relatively well developed.
  - Can interface to biological entities (medical implications).

Nano Computing





What does DNA Compute?

- Electronics has *electrons*
  - All electrons are the same
  - All you can do is see if you have *few* ('False') or *lots* ('True') of electrons
  - Hence Boolean logic is at the basis of digital circuit design
  - Symbolic and numeric computation has to be encoded above that
  - But mostly we want to compute with symbols and numbers, not with Booleans

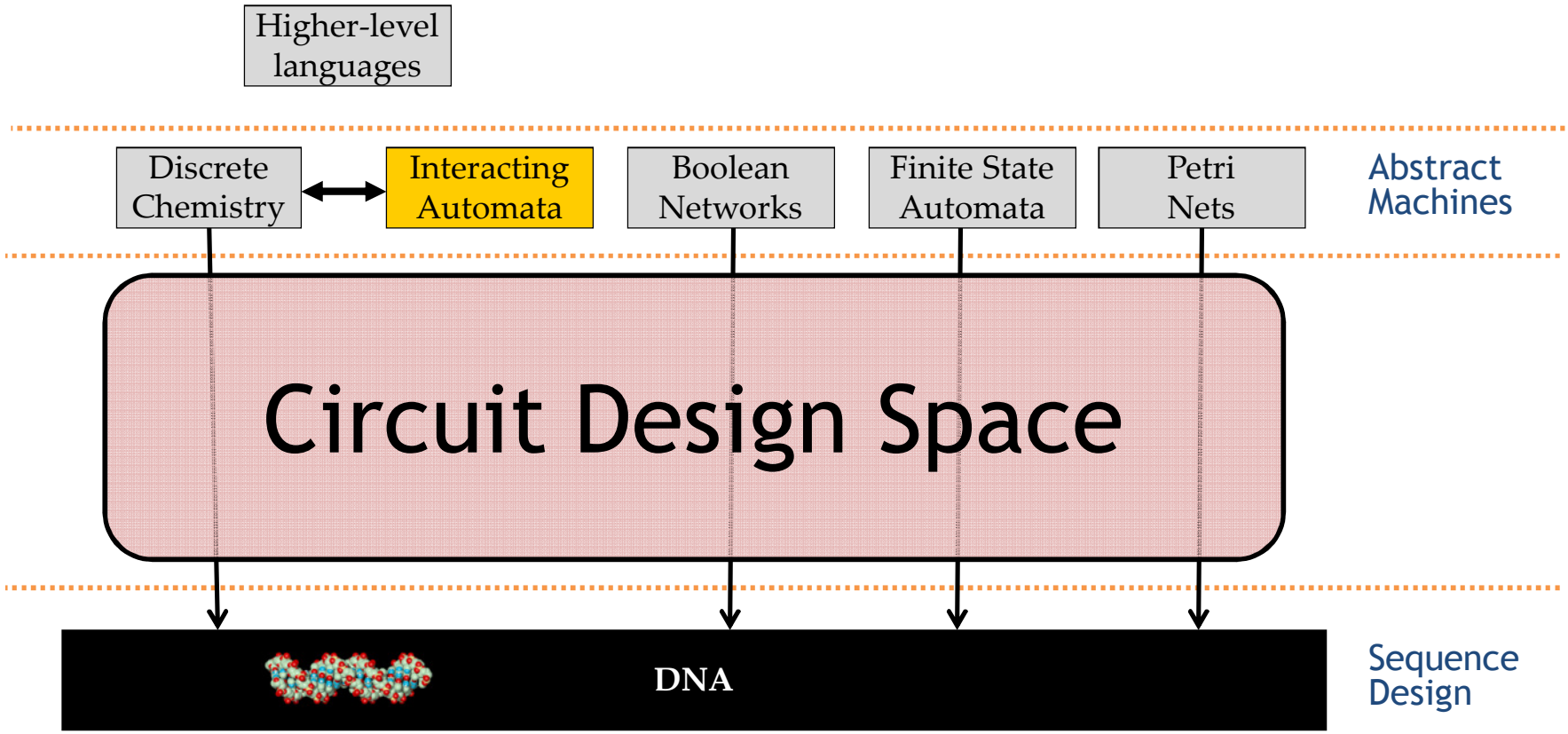
- DNA computing has *symbols* (DNA words)
  - DNA words are not all the same
  - Symbolic computation can be done *directly*
  - We can also directly use molecular concurrency

- Process Algebra as the 'Boolean Algebra' of DNA Computing
  - What are the 'gates' of symbolic concurrent computation?
  - That's what Process Algebra is about
  - (Process Algebra comes from the theory of concurrent systems)

Implementing "Arbitrary" Computing Functions

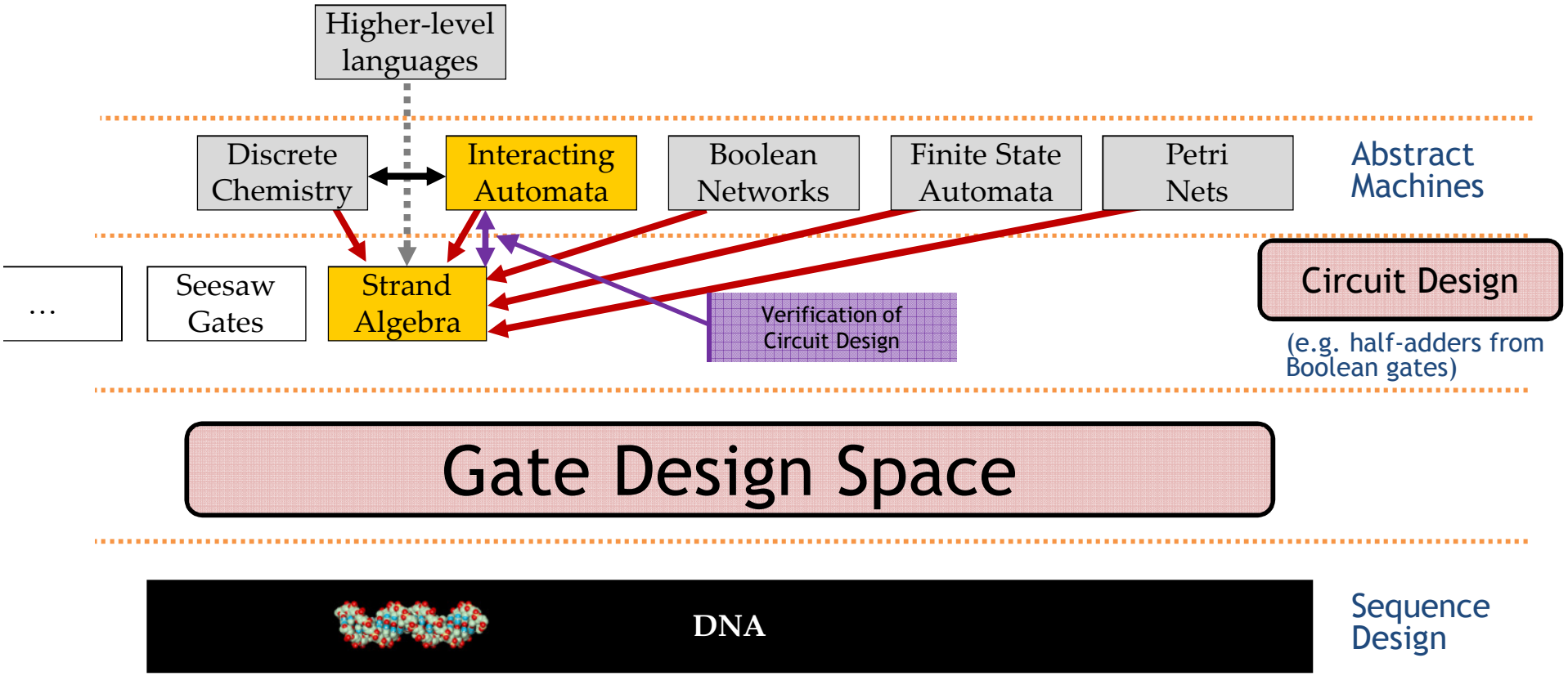
DNA Compilation

Separating Circuit Design from Gate Design



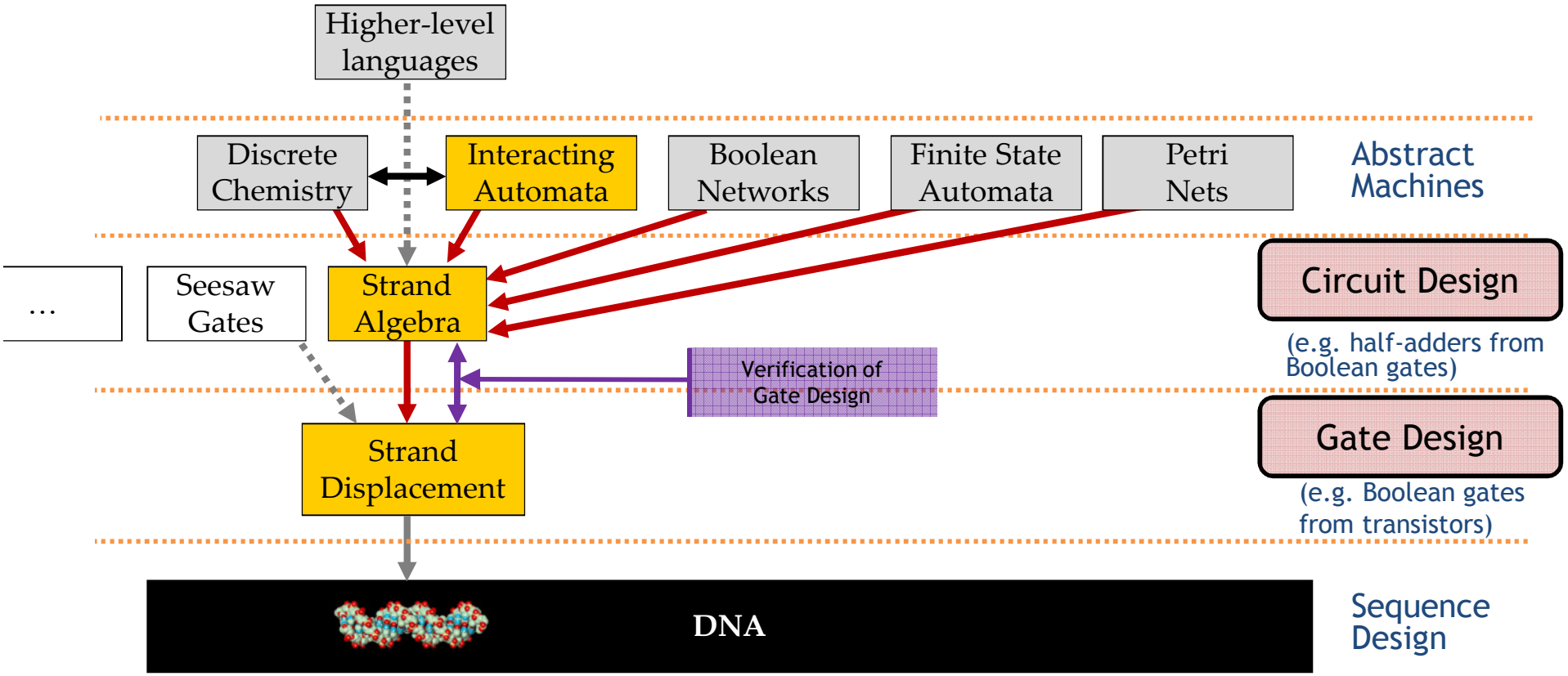
DNA Compilation

Separating Circuit Design from Gate Design



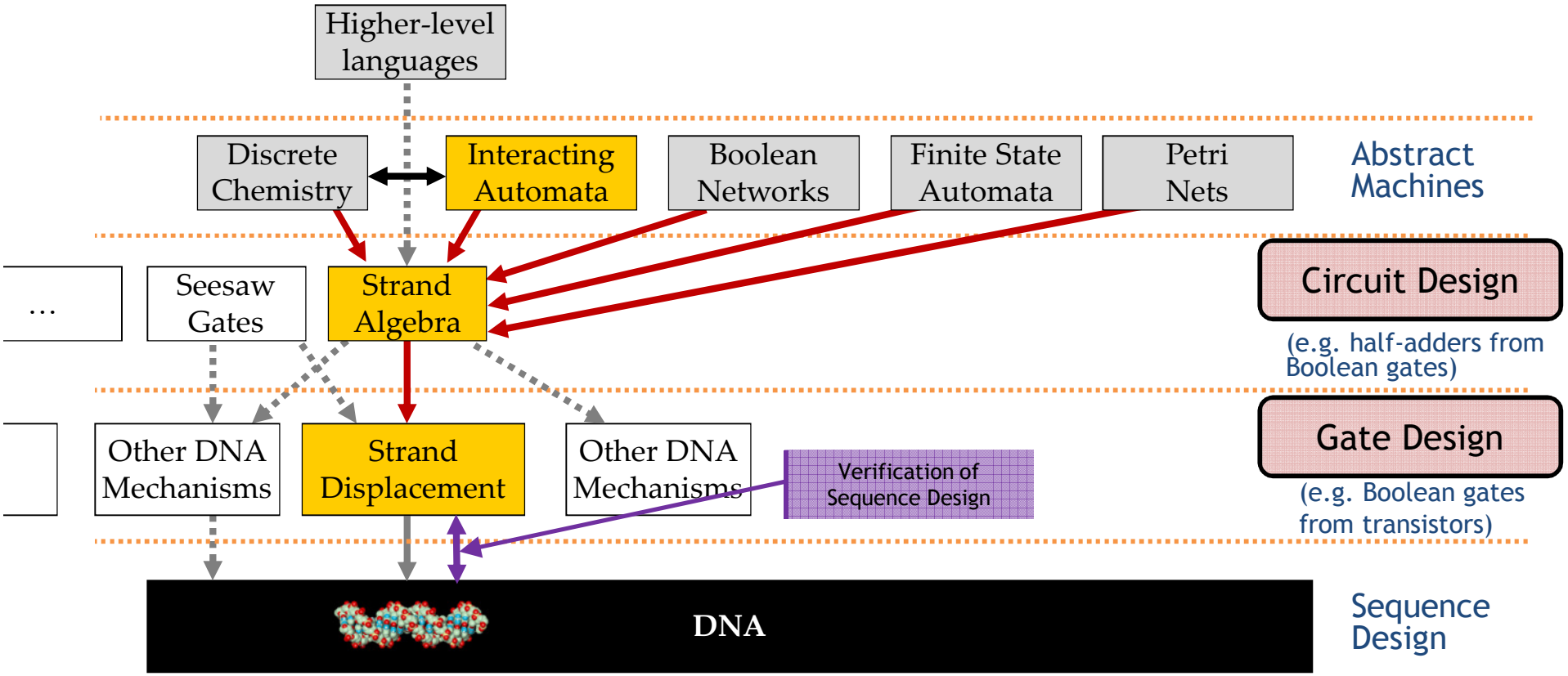
DNA Compilation

Separating Circuit Design from Gate Design



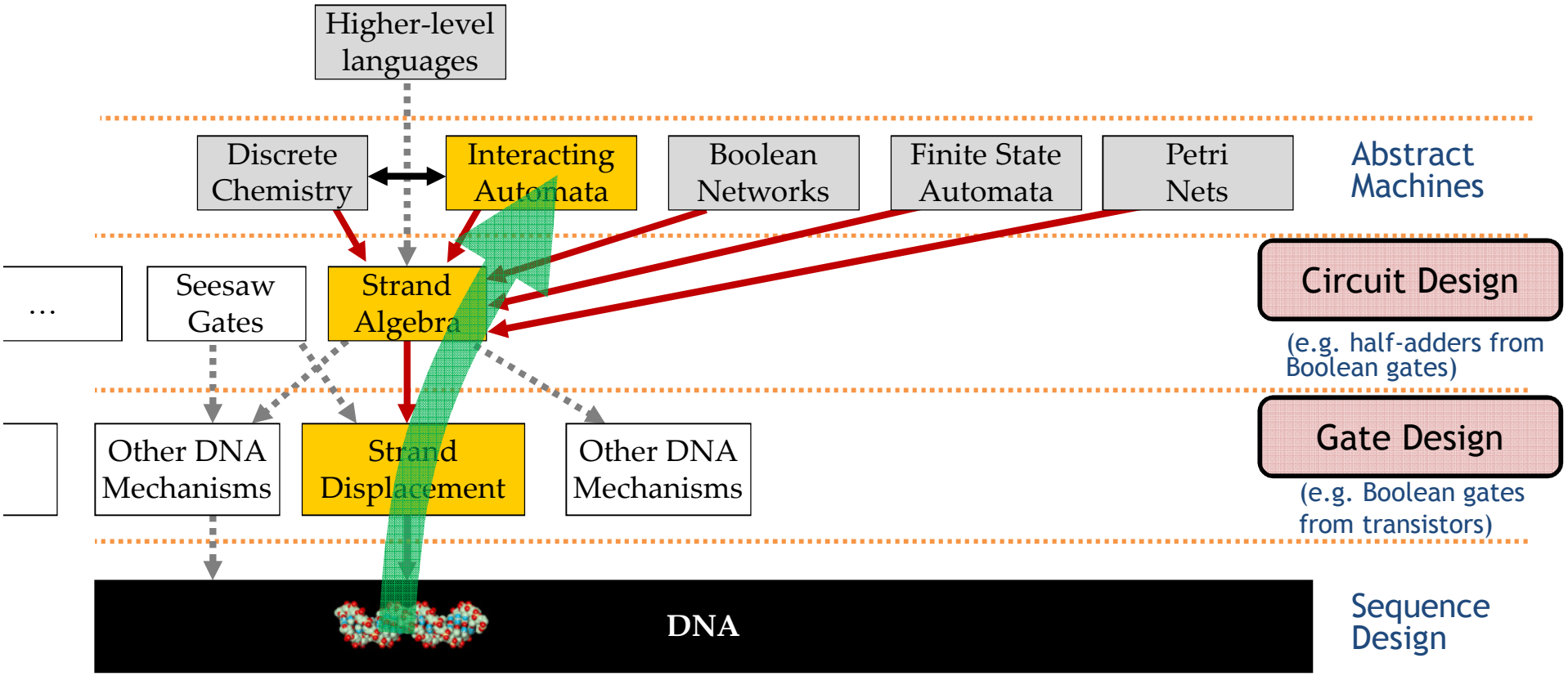
DNA Compilation

Separating Circuit Design from Gate Design



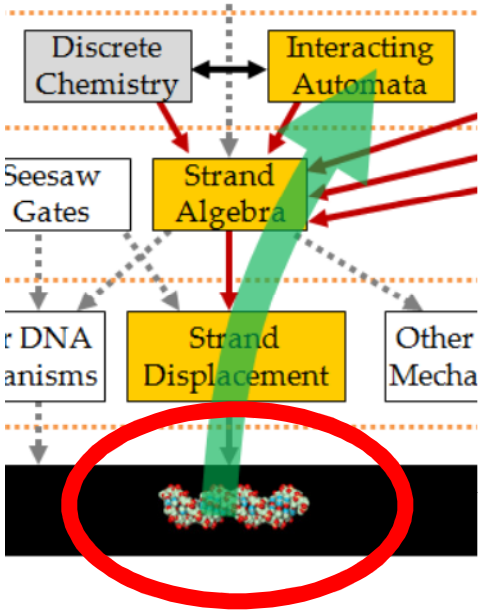
DNA Compilation

Separating Circuit Design from Gate Design

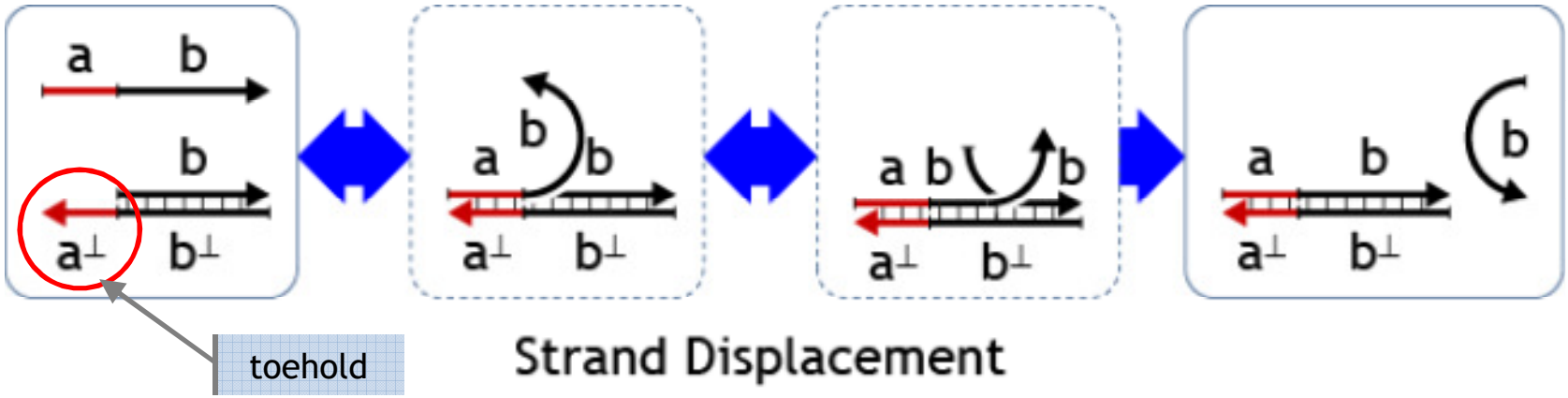


Rest of the talk: bottom up

Computational Step Design

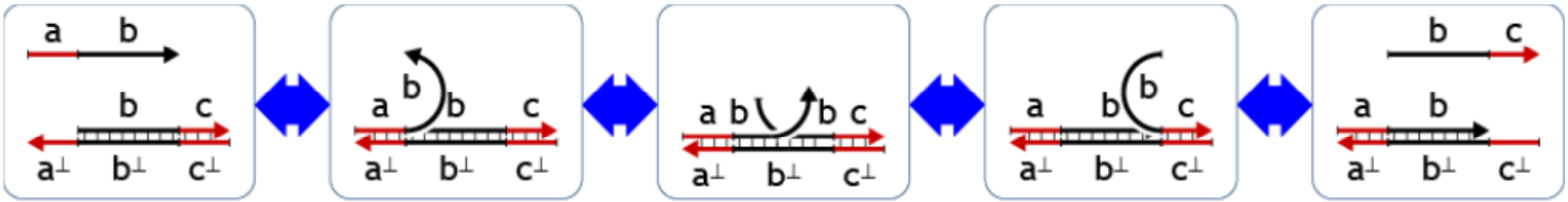


Toehold Mediated Strand Displacement



Irreversible

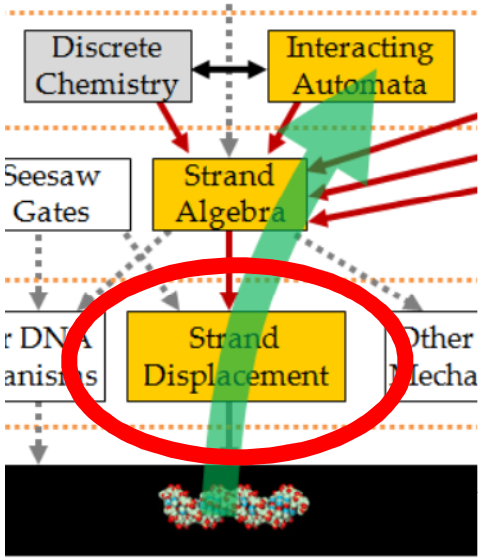
Toehold Exchange



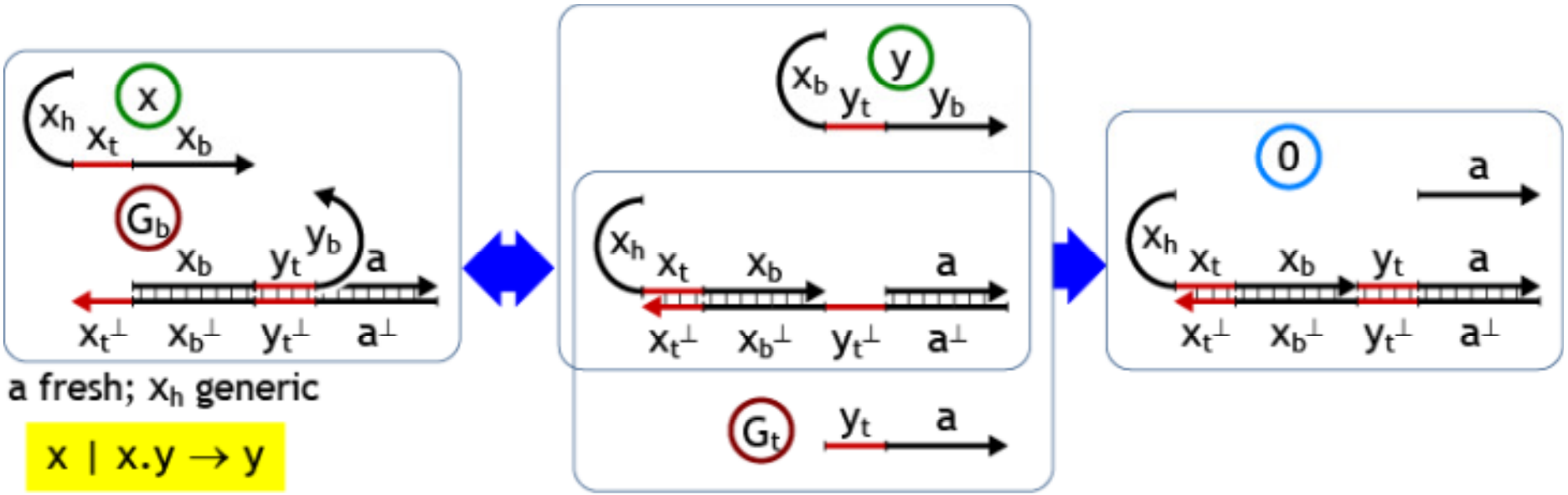
Toehold Exchange

Reversible

Gate Design


Discrete Chemistry
Interacting Automata
Seesaw Gates
Strand Algebra
Strand Displacement

x.y Transducer Gate



G_b,G_t (gate backbone and trigger) form the transducer.

Any history segment that is not determined by the gate structure is said to be ‘generic’ (can be anything).

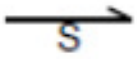
Any gate segment that is not a non-history segment of an input or output signal is taken to be ‘fresh’ (globally unique for the gate), to avoid possible interferences.

Strand Displacement Language

with Andrew Phillips

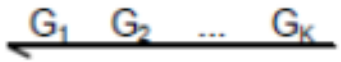
A. Syntax of DNA molecules *D*

Upper strand with sequence complementary to S




```
<S>
```

Molecule with segments $G_1,\ldots,G_K$



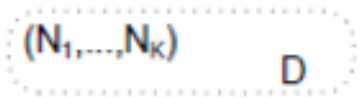

```
G1:G2:...:GK
```

Parallel molecules $D_1,\ldots,D_K$

$D_1 \quad D_2 \quad \ldots \quad D_K$

```
D1 | D2 | ... | DK
```

Molecules *D* with private domains $N_1,\ldots,N_K$




```
new (N1,...,NK) D
```

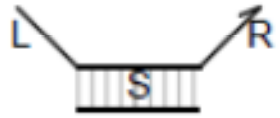
B. Syntax of DNA segments *G*

Lower strand with toehold N^c

N^c

```
N^c
```

Double strand with sequence S and overhangs L, R




```
<L>[S]<R>
```

C. Syntax of DNA sequences *S,L,R*

Sequence of domains $O_1,\ldots,O_K$

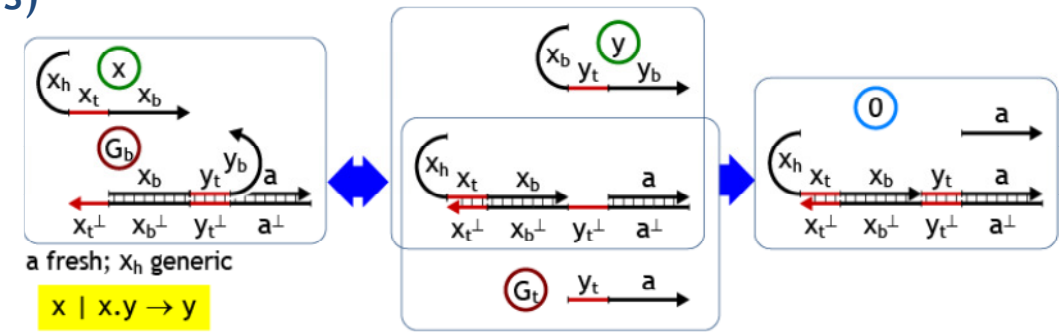
$O_1 \quad O_2 \quad \ldots \quad O_K$

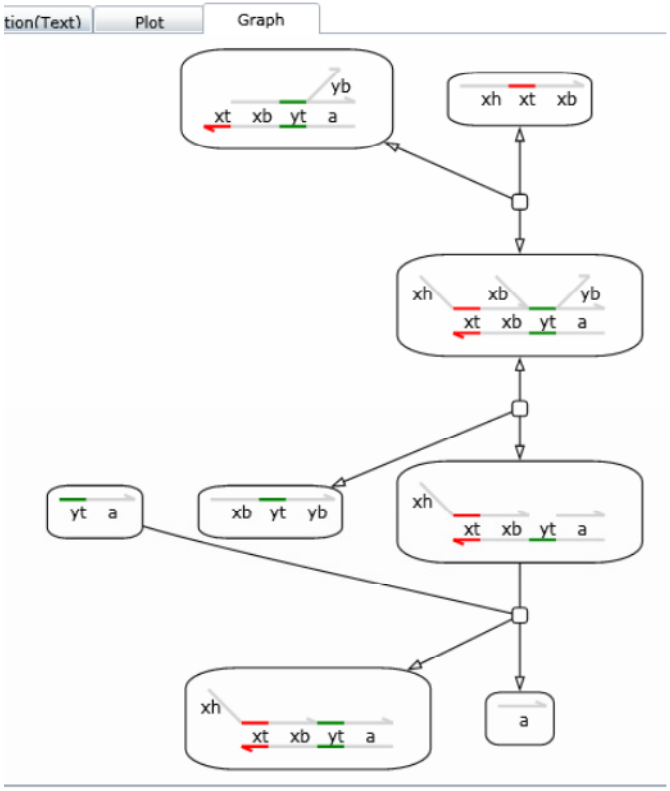
```
O1 O2 ... OK
```

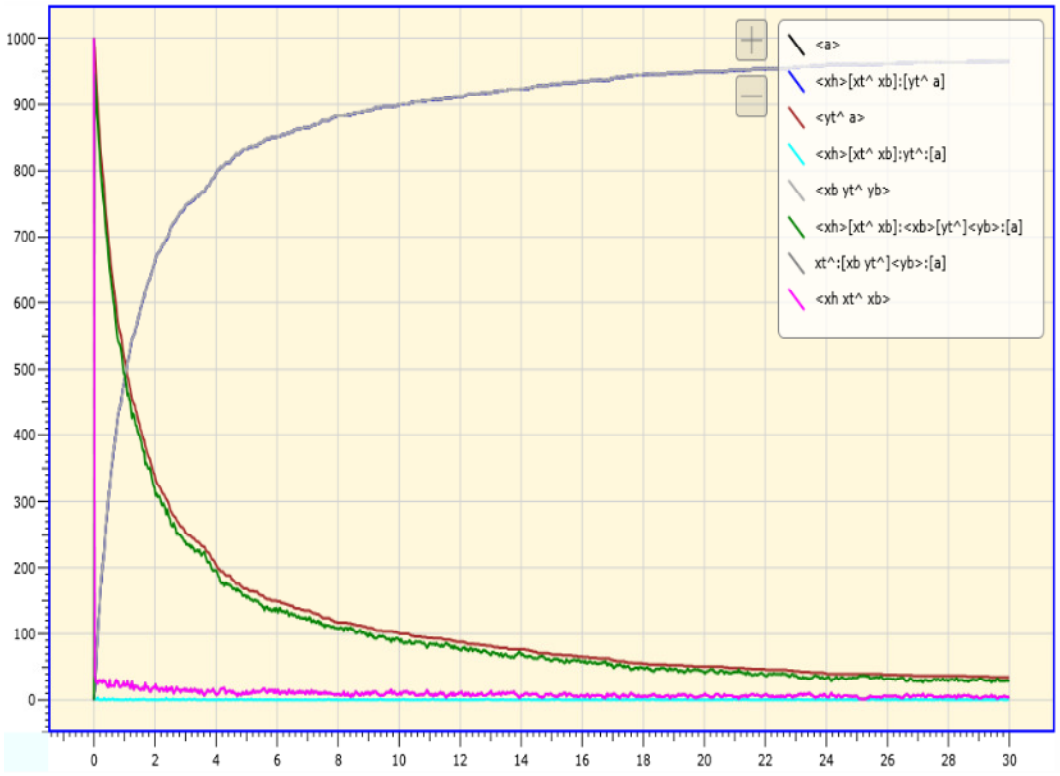
Strand Displacement Simulation Tool

Transducer gate x.y (3 initial species)

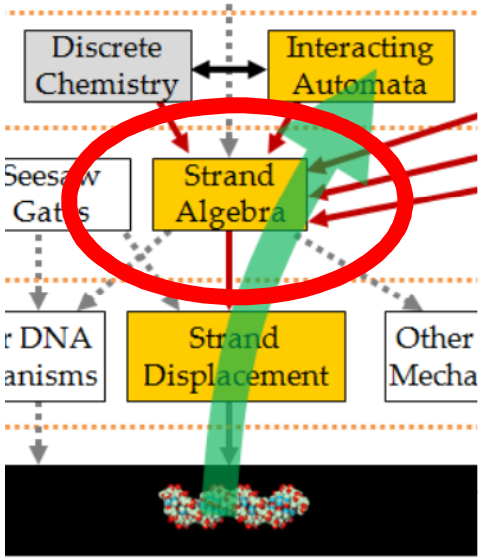
```
directive sample 30.0 1000
new xt@1.0,1.0
new yt@1.0,1.0
( 1000 * <xh xt^ xb>
| 1000 * xt^:[xb yt^]<yb>:[a]
| 1000 * <yt^ a>
)
```







Circuit Design



Strand Algebra

n x m gates

$$P ::= x \mid [x_1,..,x_n].[y_1,..,y_m] \mid 0 \mid P|P \mid P^* \qquad n\geq 1,\ m\geq 0$$

x	is a *signal*
$[x_1,..,x_n].[y_1,..,y_m]$	is a *gate*
0	is an *inert solution*
$P\|P$	is *parallel composition* of signals and gates
P^*	is a *population* (multiset) of signals and gates

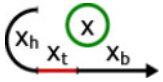
Reaction Rule

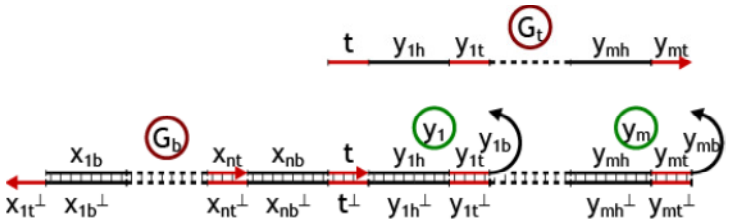
$$x_1 \mid .. \mid x_n \mid [x_1,..,x_n].[y_1,..,y_m] \rightarrow y_1 \mid .. \mid y_m$$

Equivalent to (stochastic) place-transition Petri Nets.

Compiling Strand Algebra to DNA

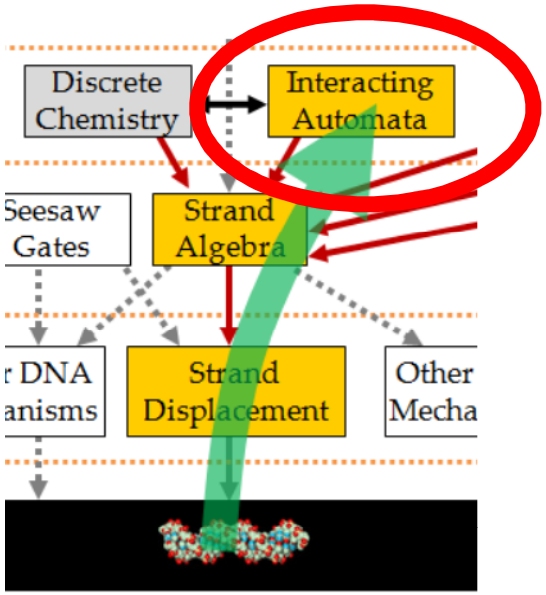
$P ::= x \mid [x_1,..,x_n].[y_1,..,y_m] \mid 0 \mid P|P \mid P^* \qquad n \geq 1,\ m \geq 0$

- compile(x) = 

- compile($[x_1,..,x_n].[y_1,..,y_m]$) = 

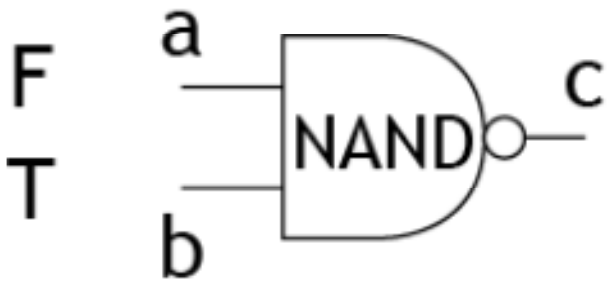
- compile(0) = empty solution
- compile(P | P') = mix(compile(P), compile(P'))
- compile(P*) = population(compile(P))

Abstract Machines



Boolean Networks

Boolean Networks to Strand Algebra



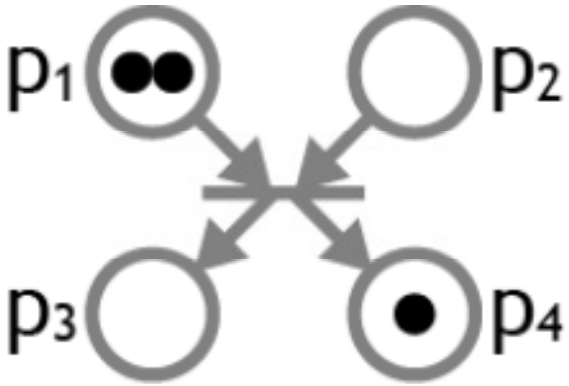
$$([a_F, b_F].c_T)^* \mid$$
$$([a_F, b_T].c_T)^* \mid$$
$$([a_T, b_F].c_T)^* \mid$$
$$([a_T, b_T].c_F)^* \mid$$
$$a_F \mid b_T$$

This encoding is *compositional*, and can encode *any* Boolean network:
- multi-stage networks can be assembled (combinatorial logic)
- network loops are allowed (sequential logic)

Petri Nets

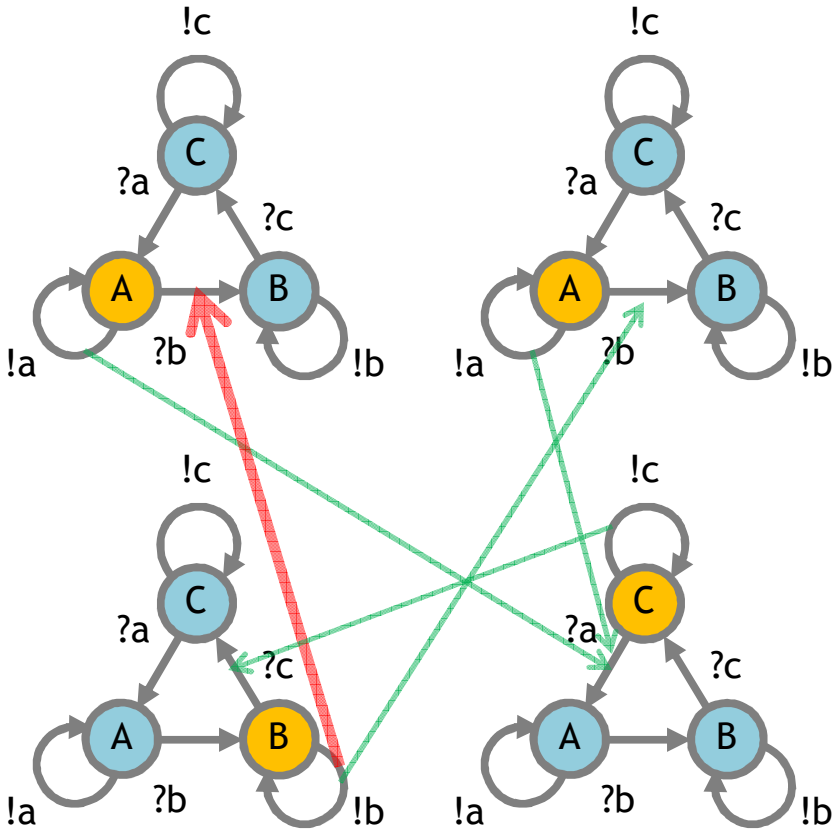
Petri Nets to Strand Algebra

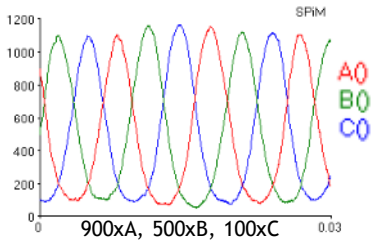
Transitions as Gates
Place markings as Signals



$$([p_1,p_2].[p_3,p_4])^* \mid p_1 \mid p_1 \mid p_4$$

Interacting Automata

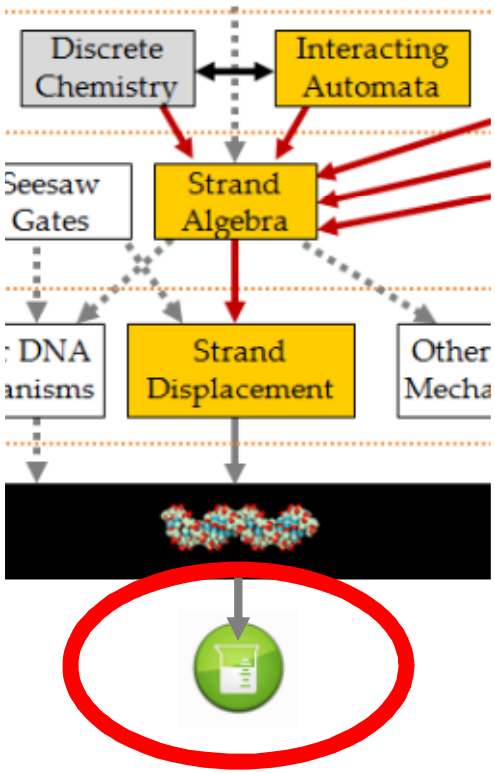




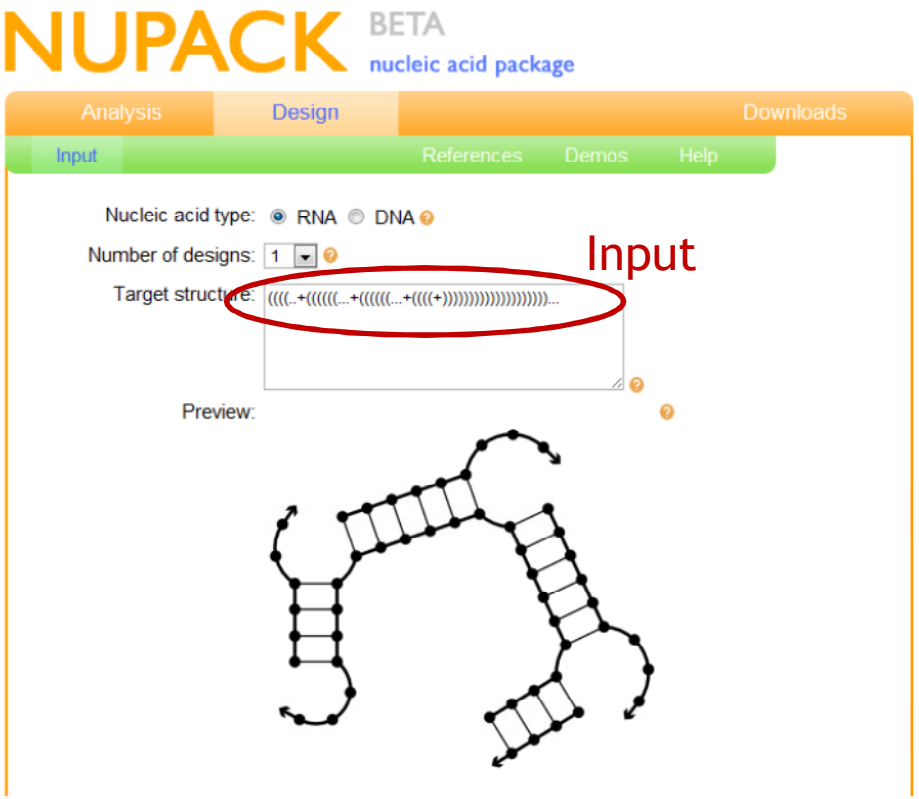
([A,B].[B,B])* |
([B,C].[C,C])* |
([C,A].[A,A])* |
A | A | B | C

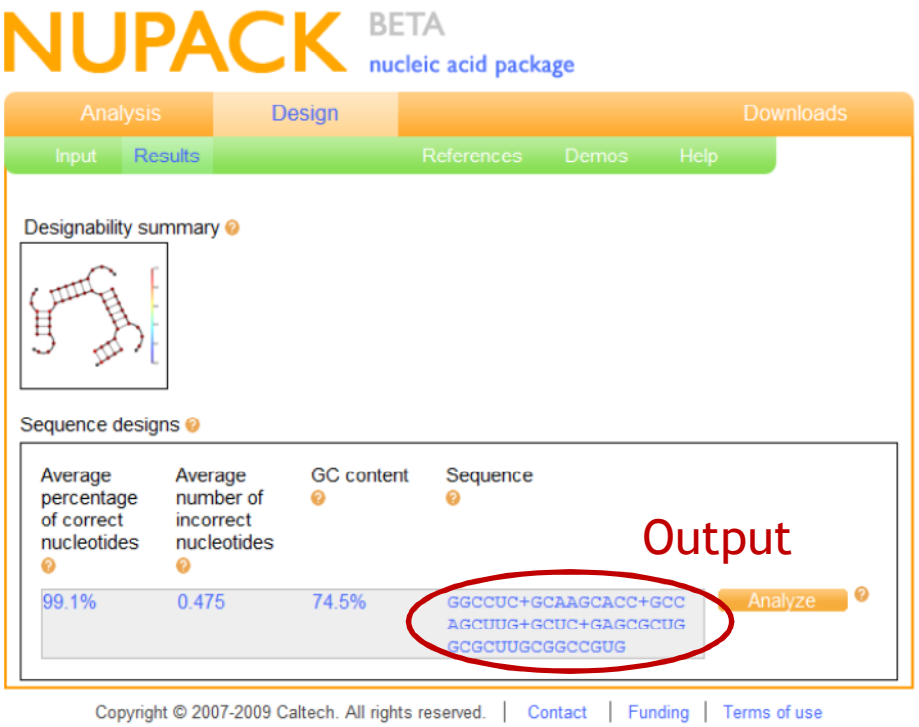
This is a uniform population of identical automata,
but heterogeneous populations of interacting automata can be similarly handled.

Sequence Design



Thermodynamic Sequence Design





http://nupack.org/

Conclusions

Conclusion

- Nucleic Acids
  - Programmable matter
- DNA Strand Displacement
  - A basic computational mechanism at the molecular level
- DNA Compilation
  - Abstract Machines (Boolean Networks, Petri Nets, Interacting Automata)
  - Via intermediate languages (Strand Algebra, Strand Displacement Language).
  - And sequence generation.
- Tools
  - Thermodynamic analysis.
  - Simulation.
  - Verification/Optimization (not yet).
- References
  - D. Soloveichik, G. Seelig, E. Winfree. DNA as a Universal Substrate for Chemical Kinetics Proc. DNA14.
  - L. Cardelli. Strand Algebras for DNA Computing. Proc. DNA15.
  - L. Cardelli, A. Phillips. A Programming Language for Composable DNA Circuits. Royal Society Interface Journal.

Q?

http://lucacardelli.name