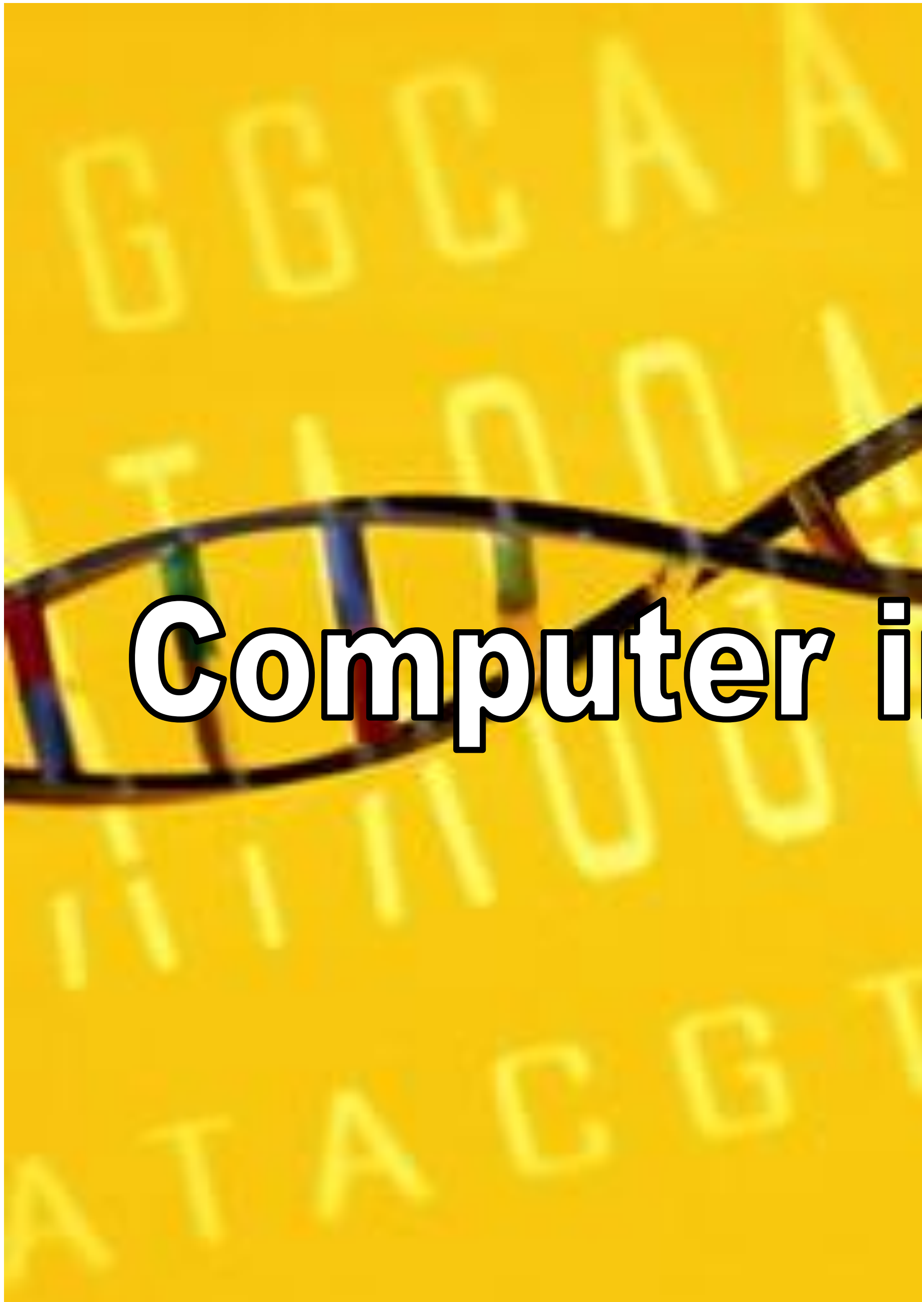
Eindhoven

Honours Class

Foundations of Informatics
Algorithmic Adventures

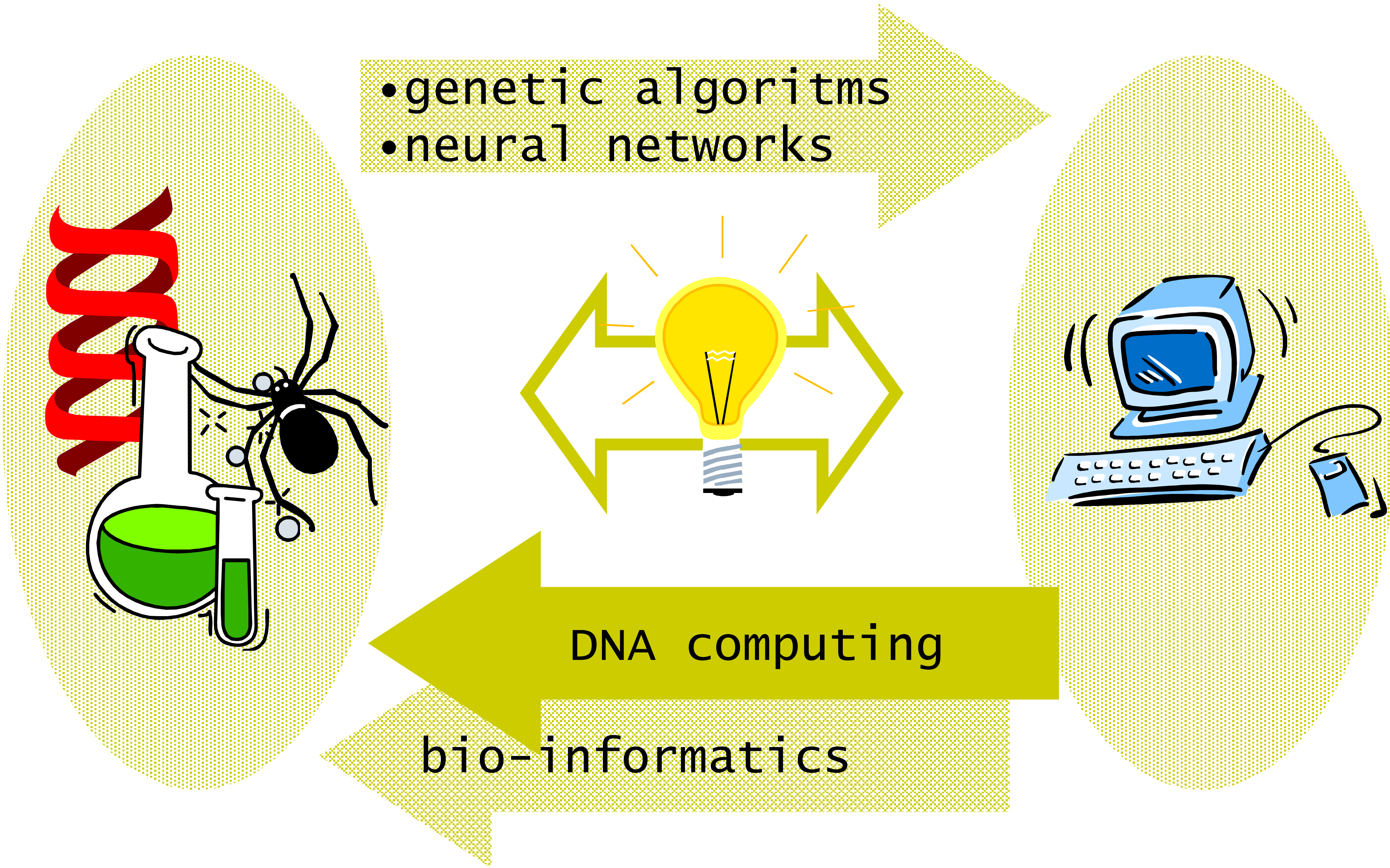


Computer in a TestTube

DNA computing

Hendrik Jan Hoogeboom
Computer Science Leiden
13 december 2010

natural computation

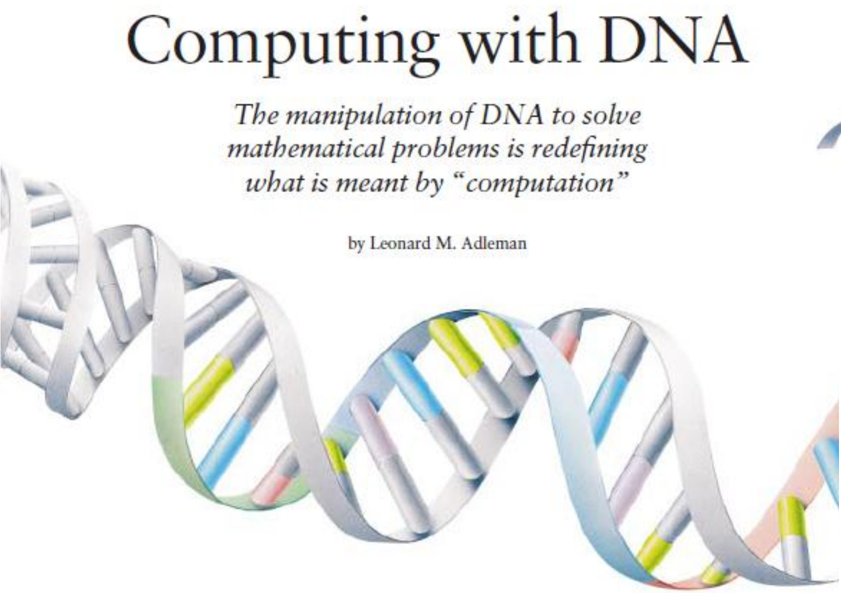


Len Adleman

Molecular Computation of Solutions to Combinatorial Problem, Science, 266: 1021-1024, (Nov. 11) 1994.

http://www.usc.edu/dept/molecular-science/fm-adleman.htm

Scientific American


Computing with DNA

The manipulation of DNA to solve mathematical problems is redefining what is meant by "computation"

by Leonard M. Adleman

"In other words, one could program a Turing machine to produce Watson-Crick complementary strings, factor numbers, play chess and so on.

This realization caused me to sit up in bed and remark to my wife, Lori, 'Jeez, these things could compute.' I did not sleep the rest of the night, trying to figure out a way to get DNA to solve problems."

Leonard M. Adleman - Computing with DNA
Scientific American August 1998

Physicists plunder life's tool chest

If we look inside the cell, we see extraordinary machines that we couldn't make ourselves, says Len Adleman. "It's a great tool chest - and we want to see what can we build with it."

Adleman created the first computer to use DNA to solve a problem. He was struck by the parallels between DNA, with its long ribbon of information, and the theoretical computer known as the Turing Machine.

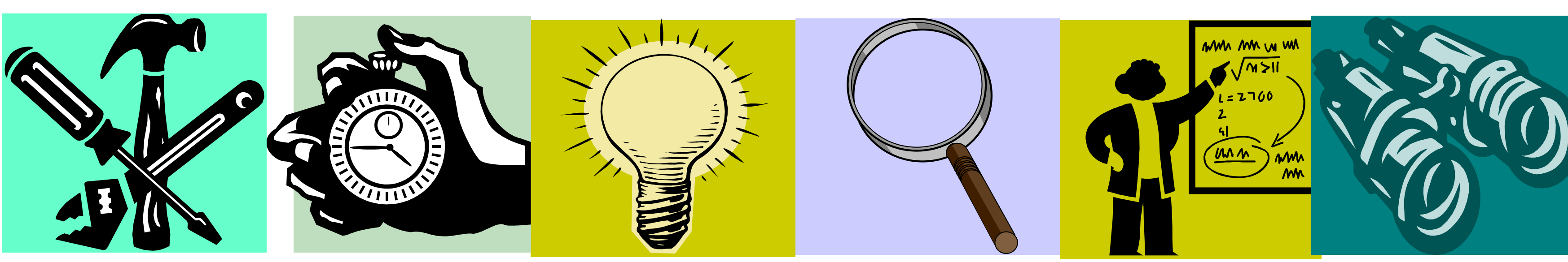
Nature News Service april 2003

Adleman tackled the famous ‘travelling salesman’ problem - finding the shortest route between cities. Such problems rapidly become mind-boggling. The only way is to examine every possible option. With many cities, this number is astronomical.

DNA excels at getting an astronomical amount of data into a tiny space. “One gram of DNA can store as much information as a trillion compact discs,” says Adleman. Myriad DNA molecules can examine every possible route at once, rather than one at a time, as in a conventional computer.

contents

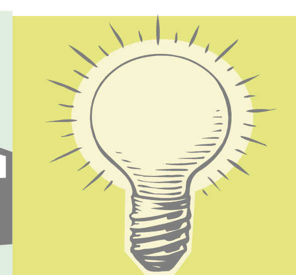
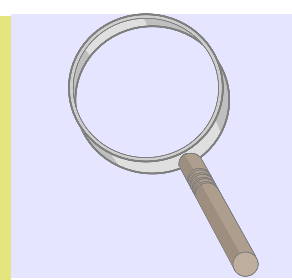
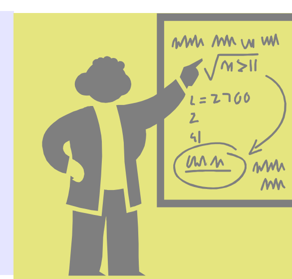
- DNA … the tool chest
- problem complexity … P & NP
  Hamilton Path Problem
- Adleman's algorithm
- comments
- theory … Turing machine
- recent work + future



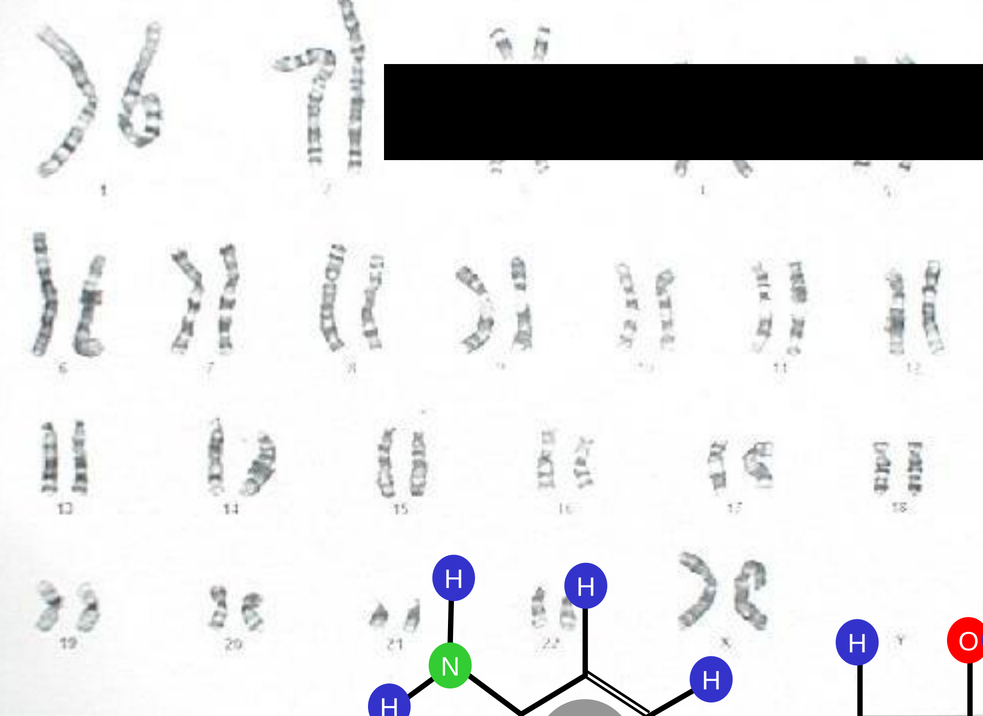
contents

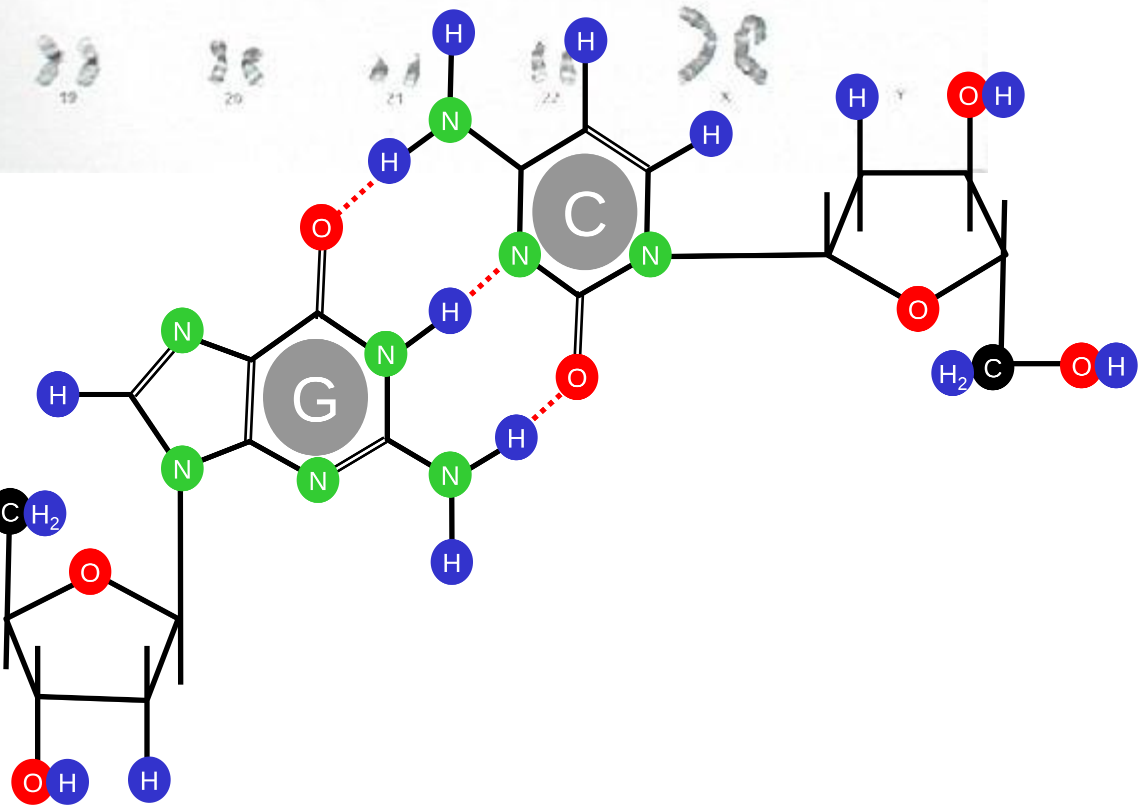
- DNA … the tool chest
- problem complexity … P & NP
  - Hamilton Path Problem
- Adleman’s algorithm
- comments
- theory … Turing machine
- recent work + future

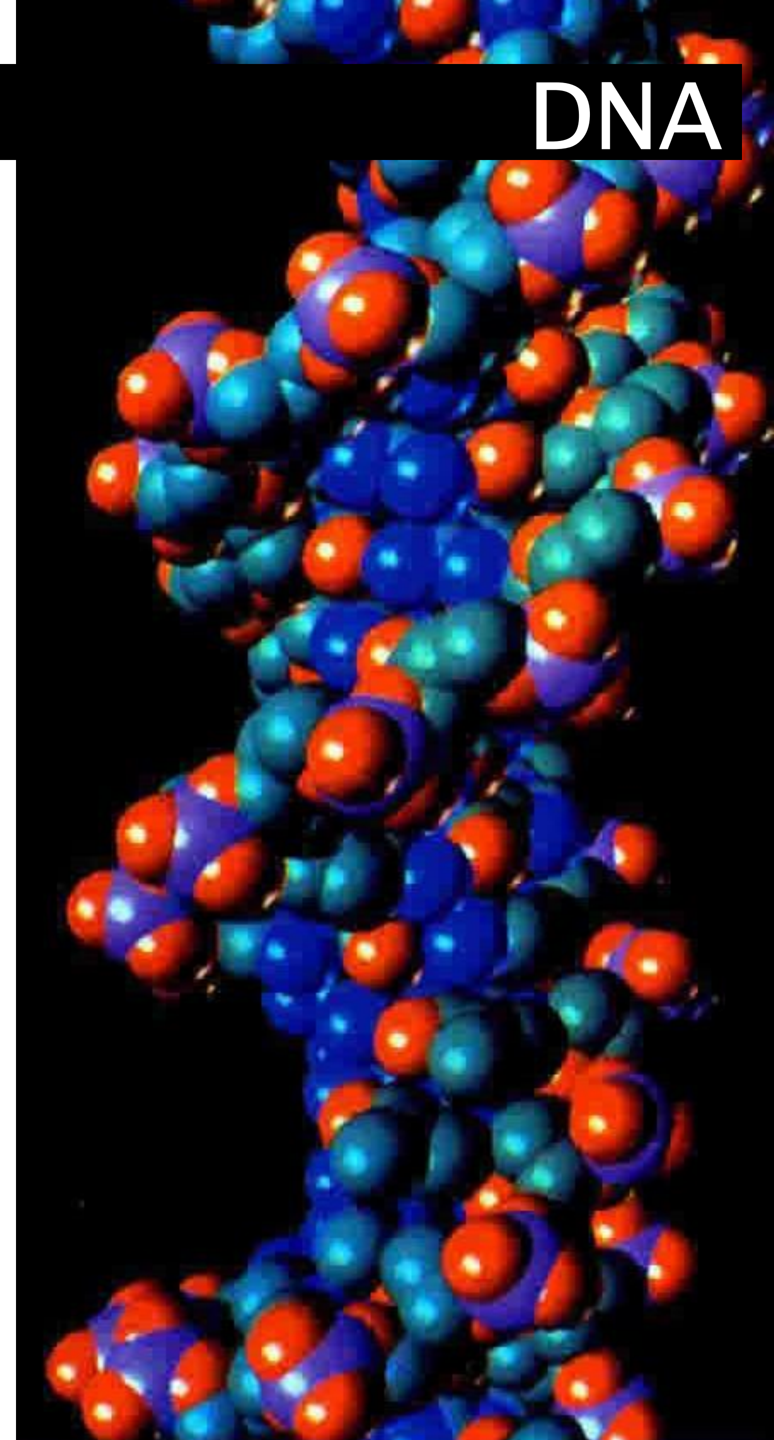


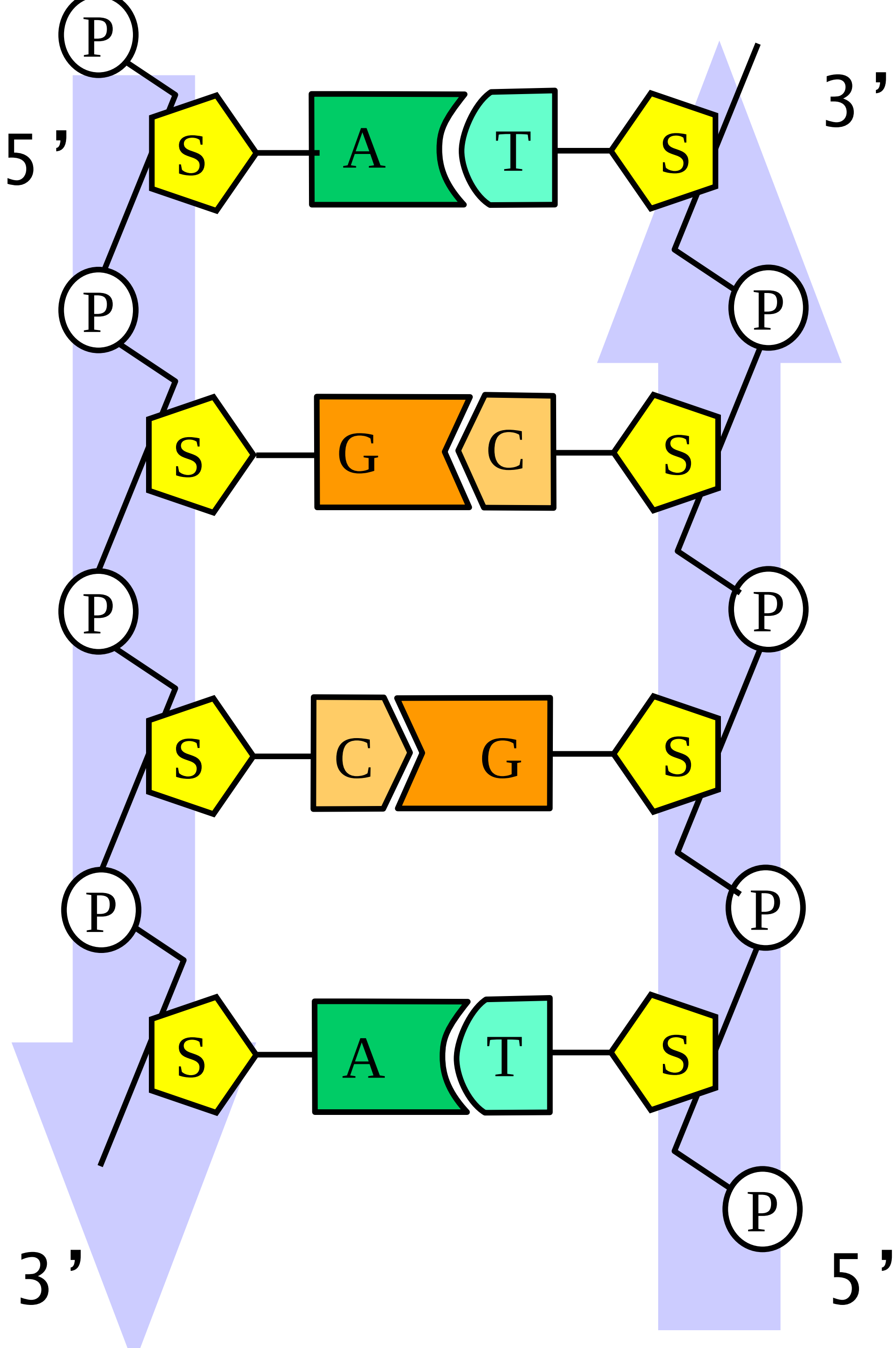





DNA







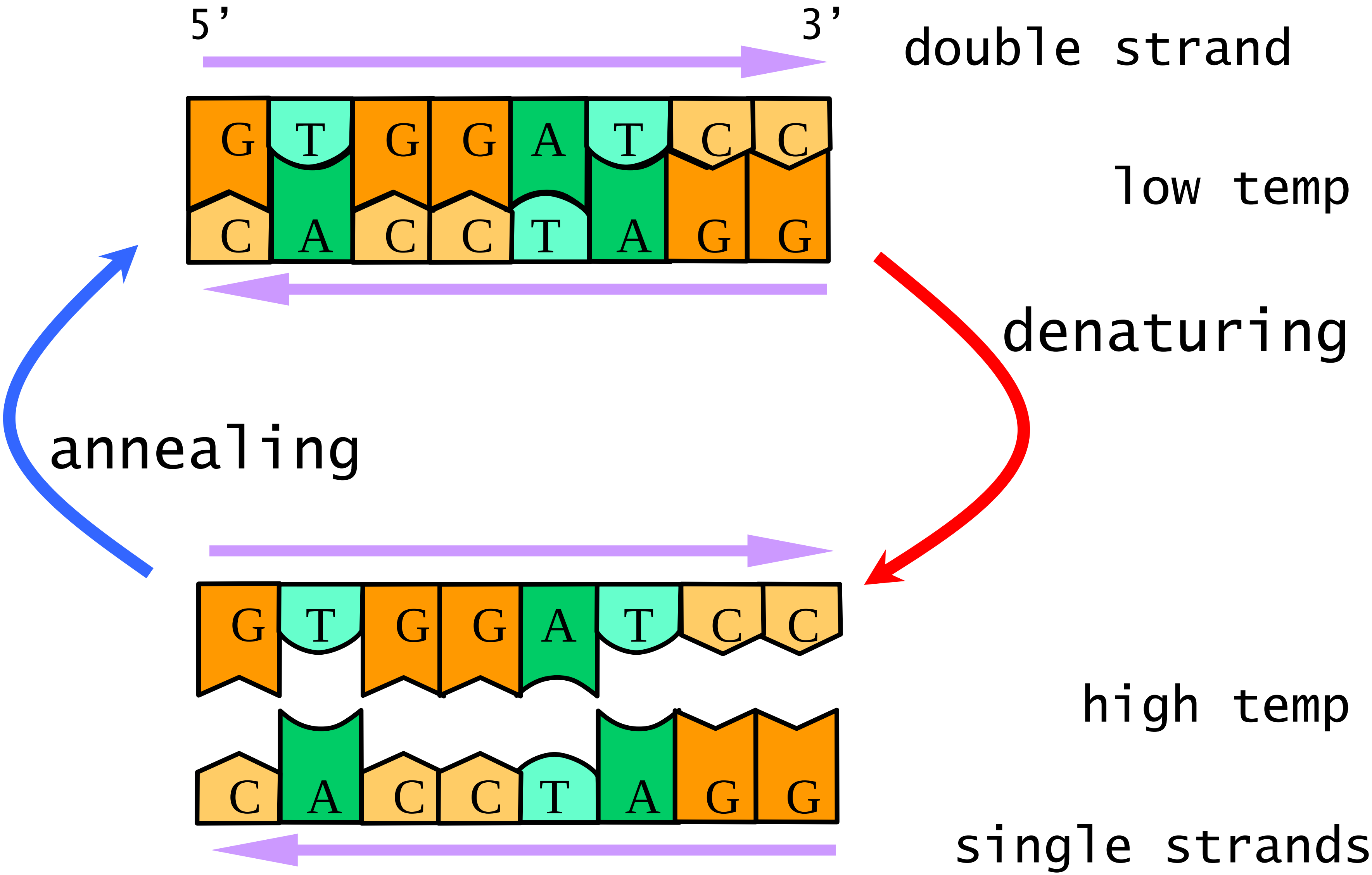


DNA

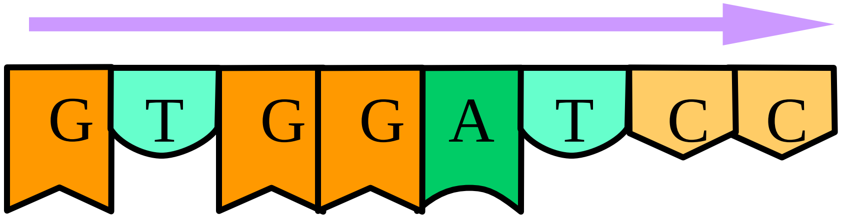
Base pairs
Watson & Crick
[& Rosalind Franklin]

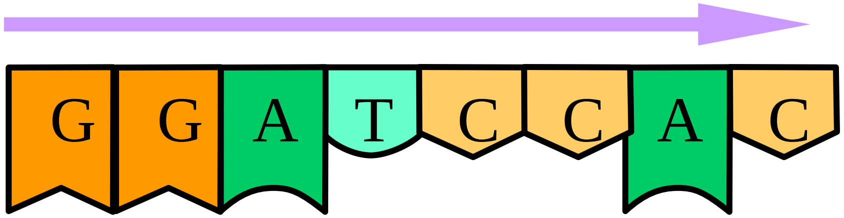
A=T
adenine – thymine
C≡G
guanine – cytosine

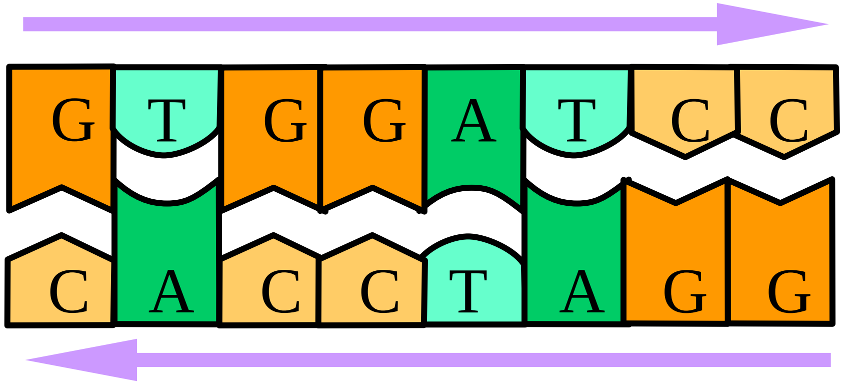
single - double strand



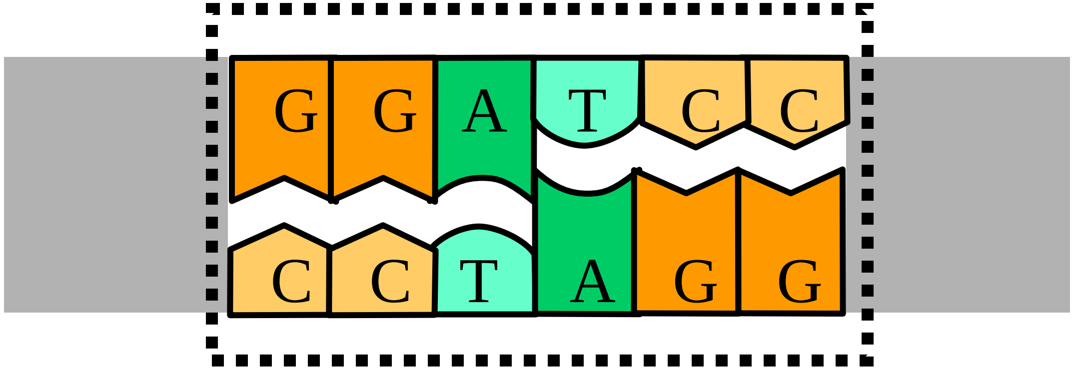
complementarity



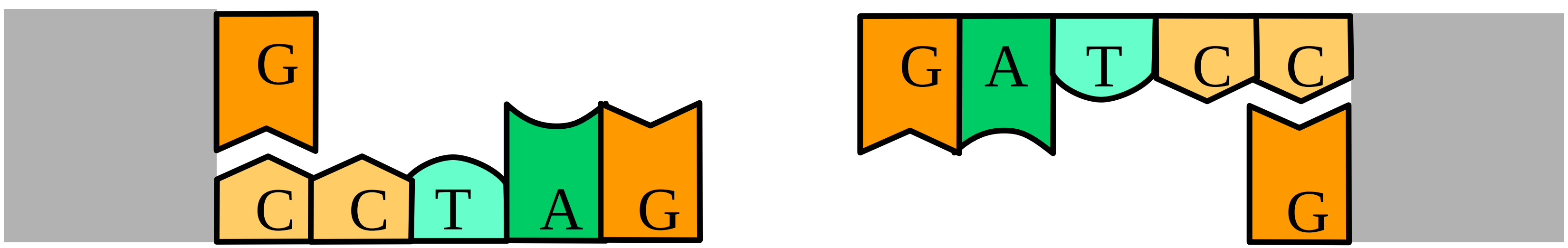




restriction enzymes

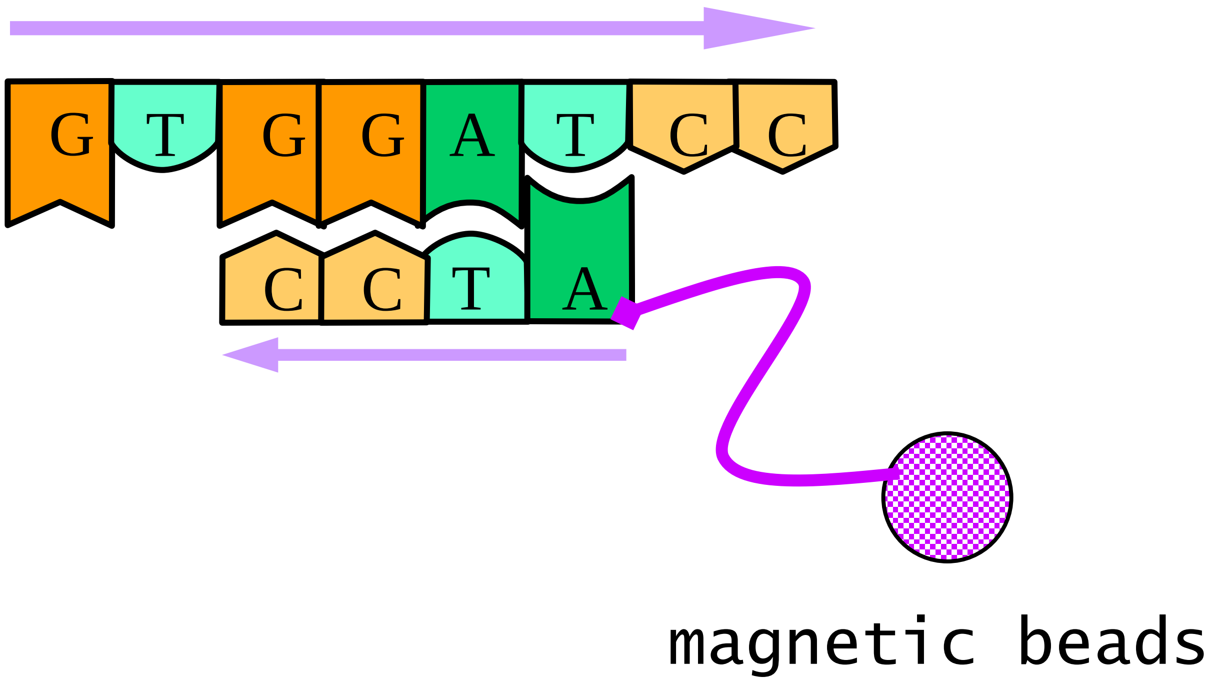


BamHI



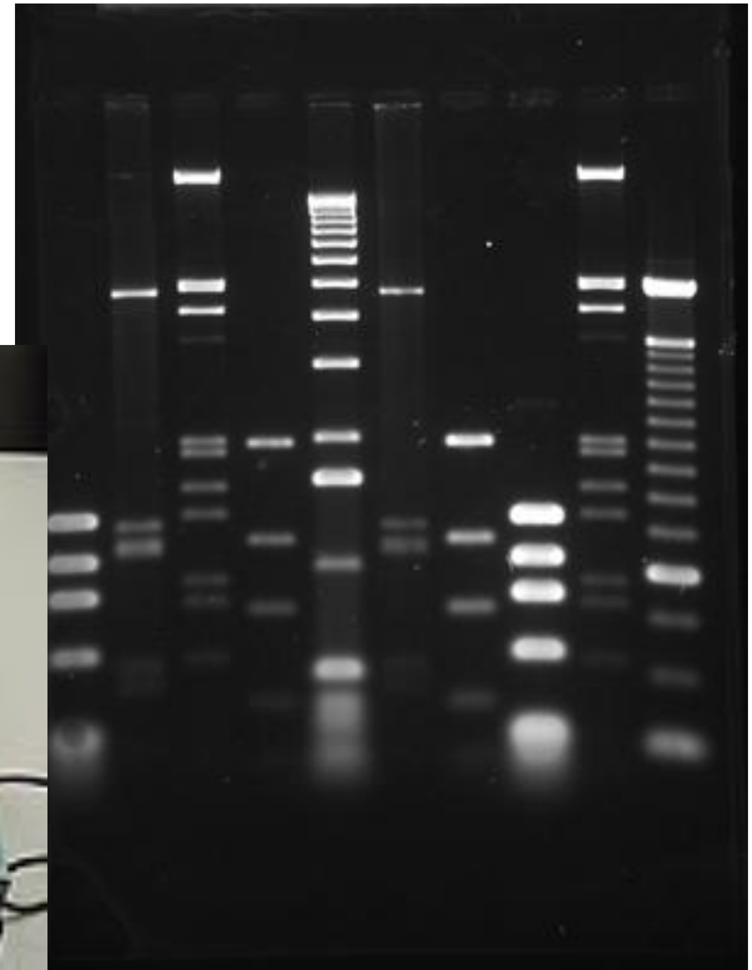
sticky ends

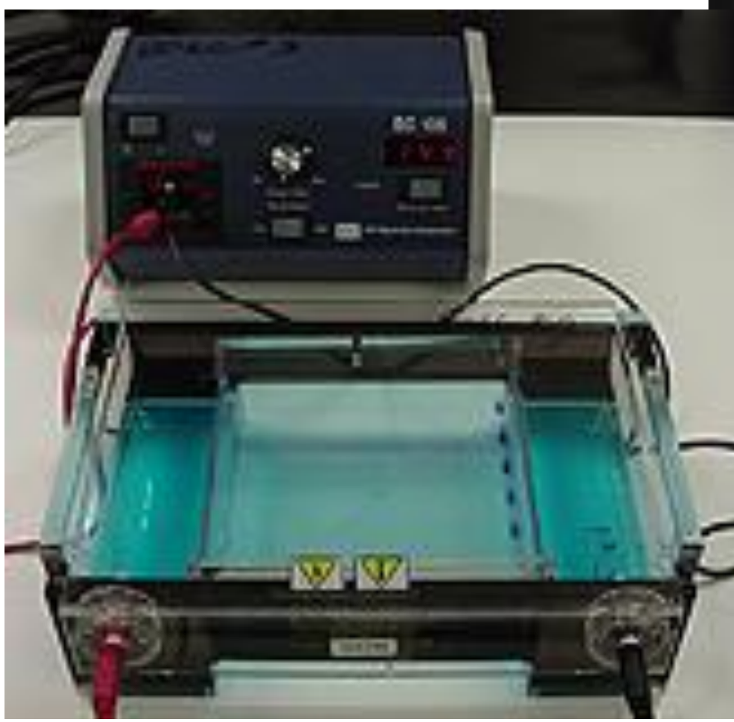
subsequence selection



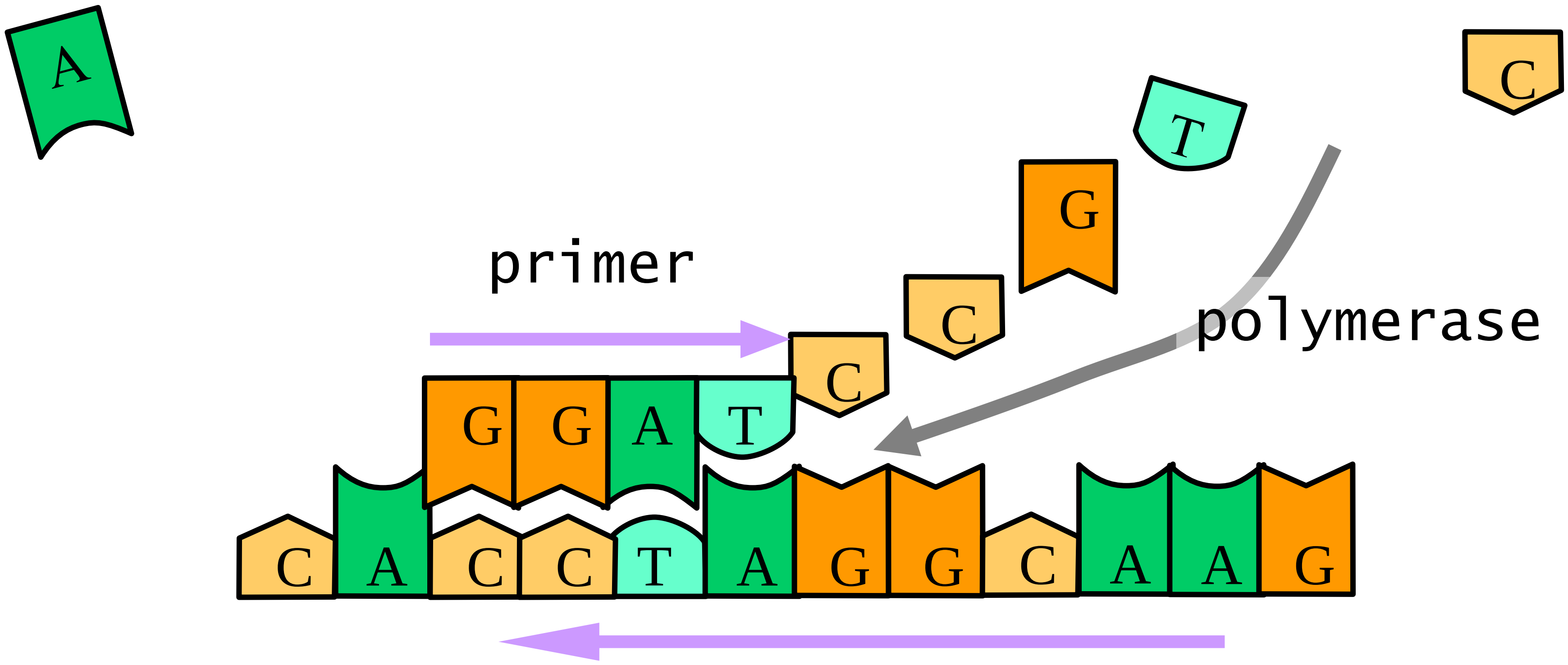
separation on length

DNA gel electrophoresis





multiplication / amplification

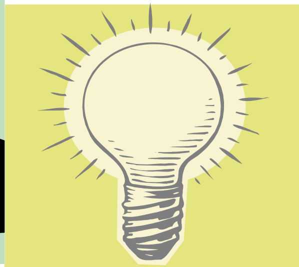
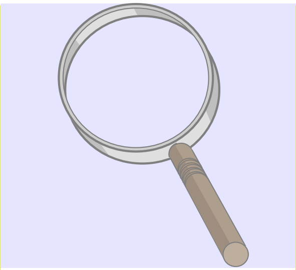
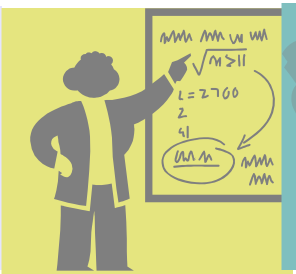
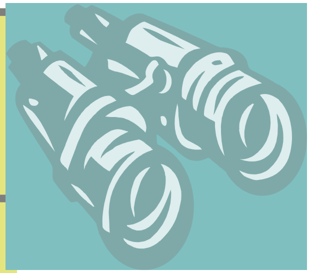


PCR – polymerase chain reaction

contents

- DNA … the tool chest
- problem complexity … P & NP
  - Hamilton Path Problem
- Adleman's algorithm
- comments
- theory … Turing machine
- recent work + future



complexity

second
minute
day
year
century

	n=10	30	50	60
n	10^{-5}s	3×10^{-5}s	5×10^{-5}s	6×10^{-5}s
n^2	10^{-4}s	9×10^{-4}s	2×10^{-3}s	4×10^{-3}s
n^5	10^{-1}s	24 s	1.7 m	13 m
2^n	10^{-3}s	18 m	13 d	366 c
3^n	6×10^{-2}s	6.5 y	3855 c	10^{13} c

polynomial vs.
exponential

	now	100x	1000x
n	N	100N	1000N
n^2	N	10N	32N
n^5	N	2.5N	4N
2^n	N	N+6.6	N+10
3^n	N	N+4.2	N+6.3

The general idea

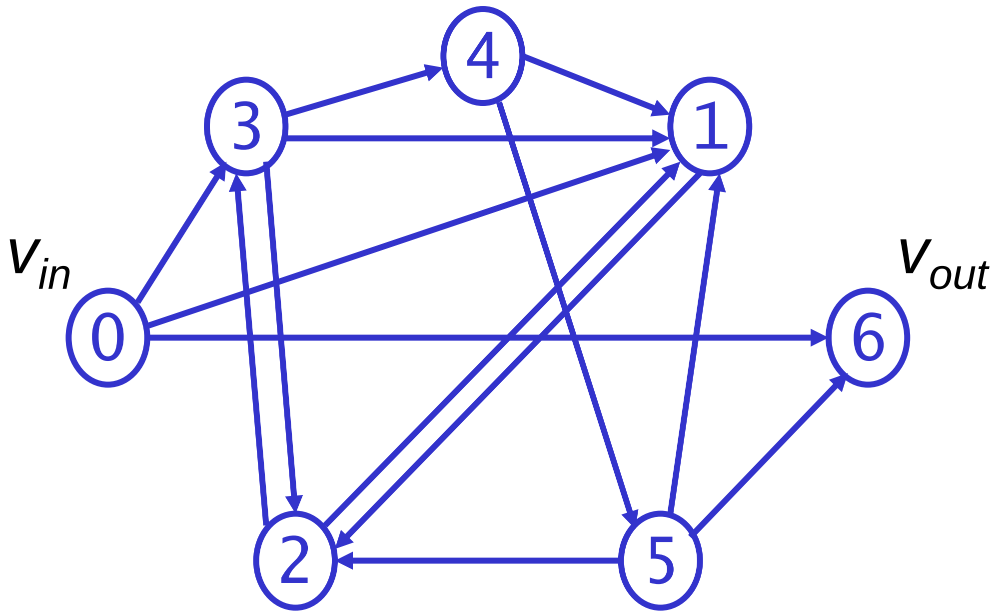
custom made
single strands of DNA
(many copies)

is there a
double strand
with my desired
properties?

properties:
- length,
- subsequence.

if we can do this, then we can solve
certain problems (efficiently)!

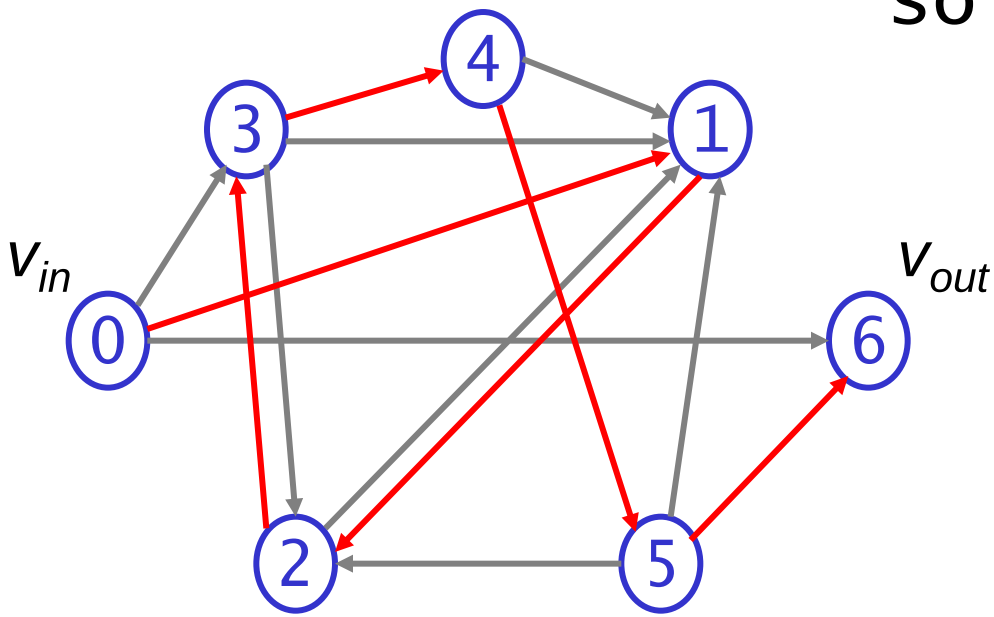
HPP: Hamilton Path Problem



'travelling salesman'

given: directed graph (points & connections)
question: is there a path that visits each point exactly once ?

HPP: Hamilton Path Problem

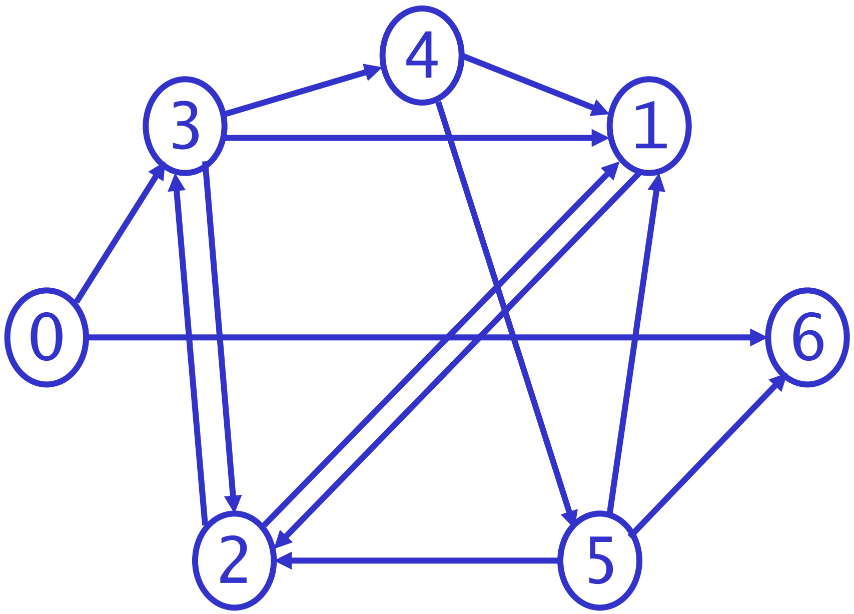


solution

'travelling salesman'

given: directed graph (points & connections)
question: is there a path that visits each point exactly once ?

HPP: Hamilton Path Problem

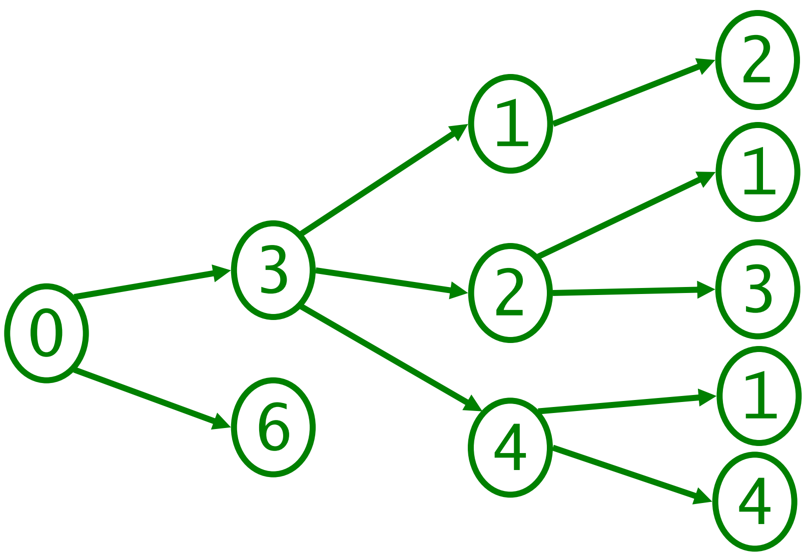


no solution?

exponential time:
try all possibilities

representative class
'**NP complete**'

heuristics



complexity (theory) - P vs. NP

P

polynomial algorithm
to **find** a solution

NP

polynomial algorithm
to **verify** a solution

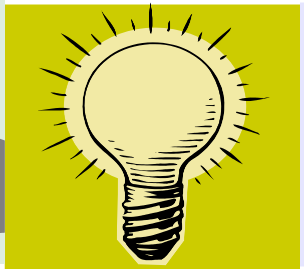
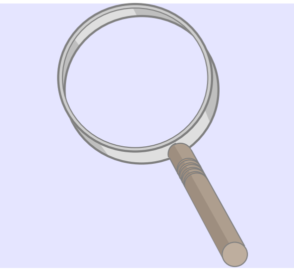
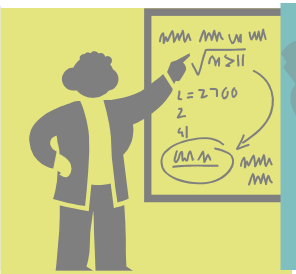
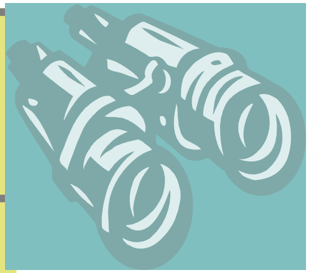
NP-complete

millenium prize problem $P \stackrel{?}{=} NP$

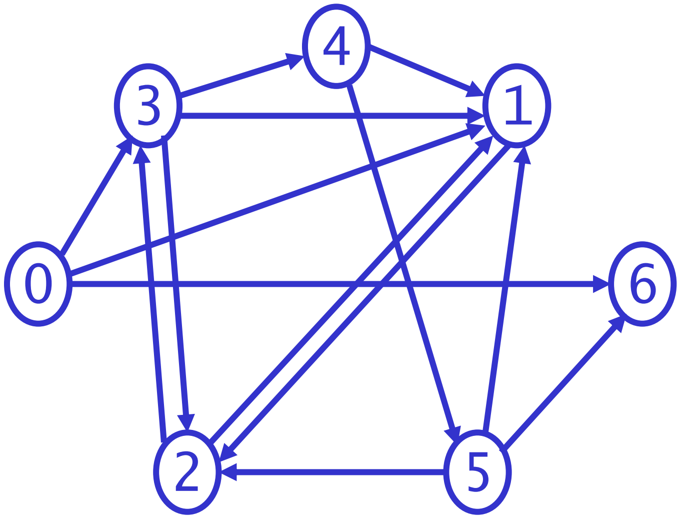
www.claymath.org/Millennium_Prize_Problems/

contents

- DNA … the tool chest
- problem complexity … P & NP
  - Hamilton Path Problem
- Adleman's algorithm
- comments
- theory … Turing machine
- recent work + future

Adleman's algorithm

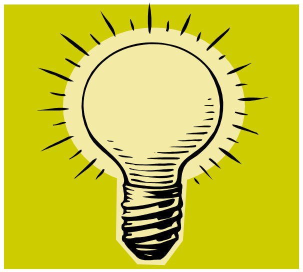


1. generate 'all' paths

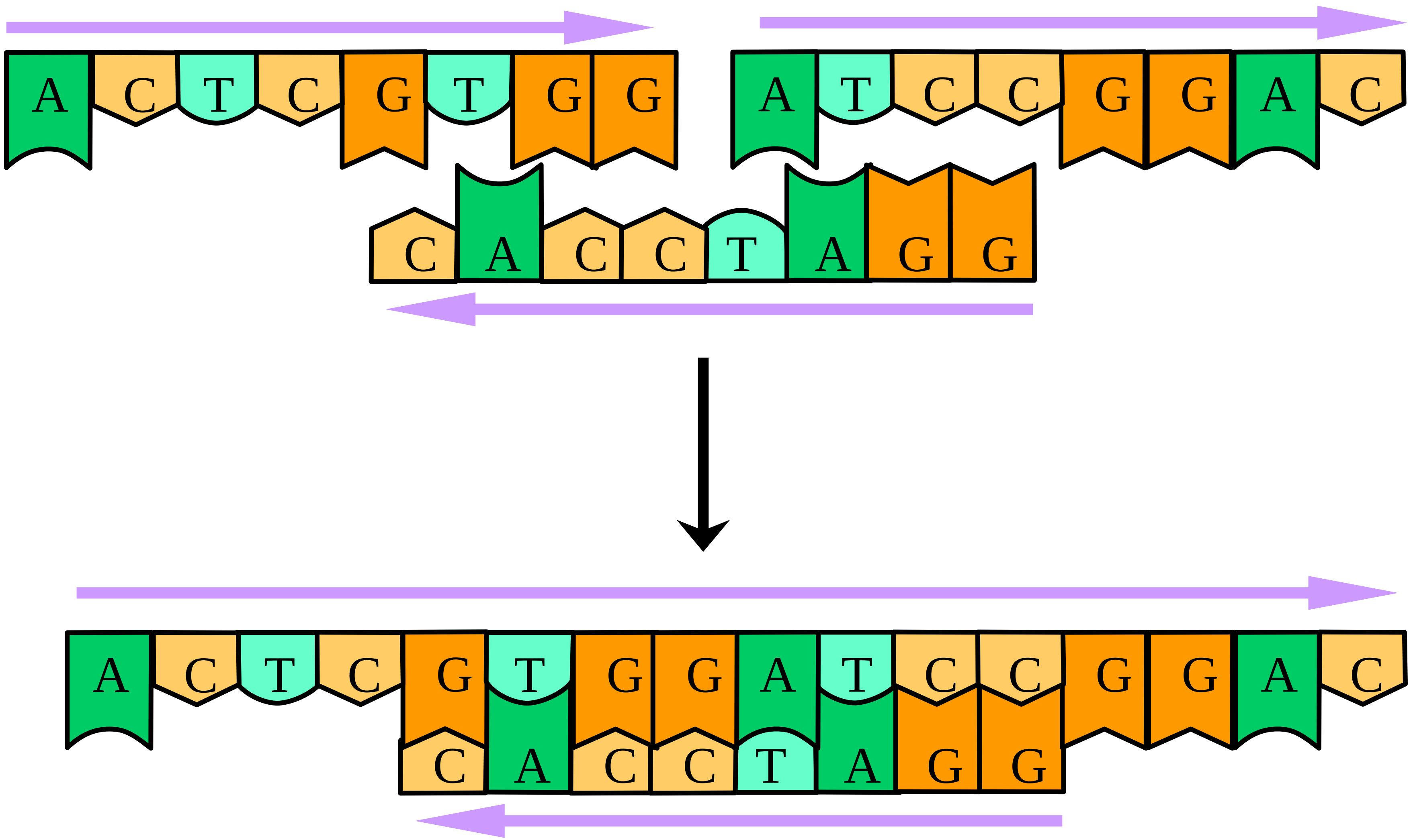
   keep only paths
2. … from v_{in} to v_{out}
3. … that enter n vertices
4. … that enter all vertices

5. if any path remains OK

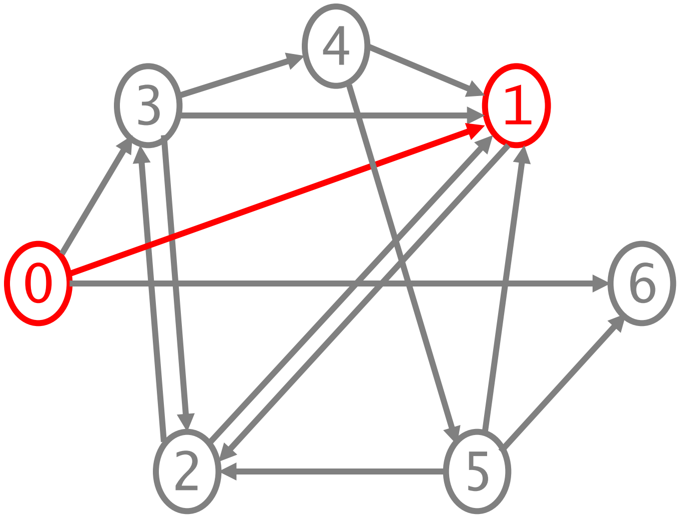
'*massive parallellism*'



building blocks



Adleman's algorithm

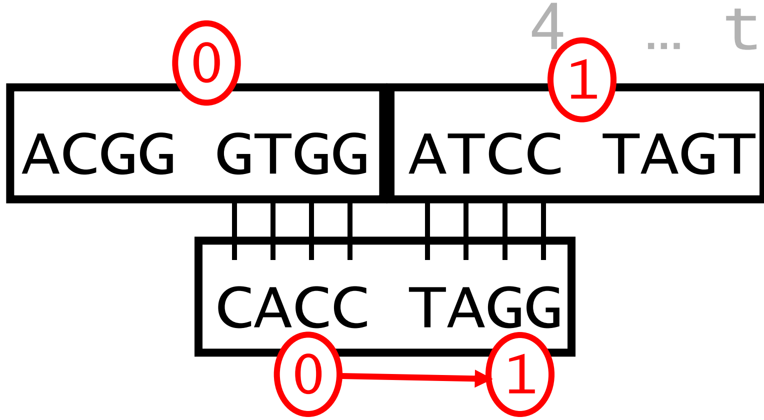


0. coding the graph

1. generate ‘all’ paths

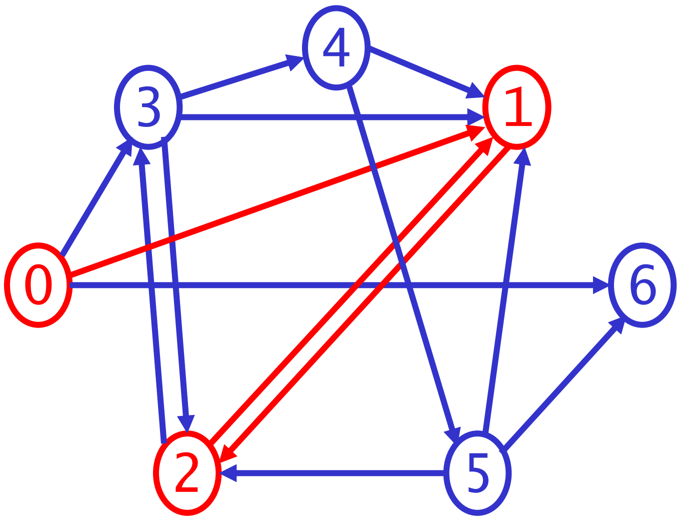
   keep only paths

2. … from v_{in} to v_{out}
3. … that enter n vertices
4. … that enter all vertices



any path remains OK

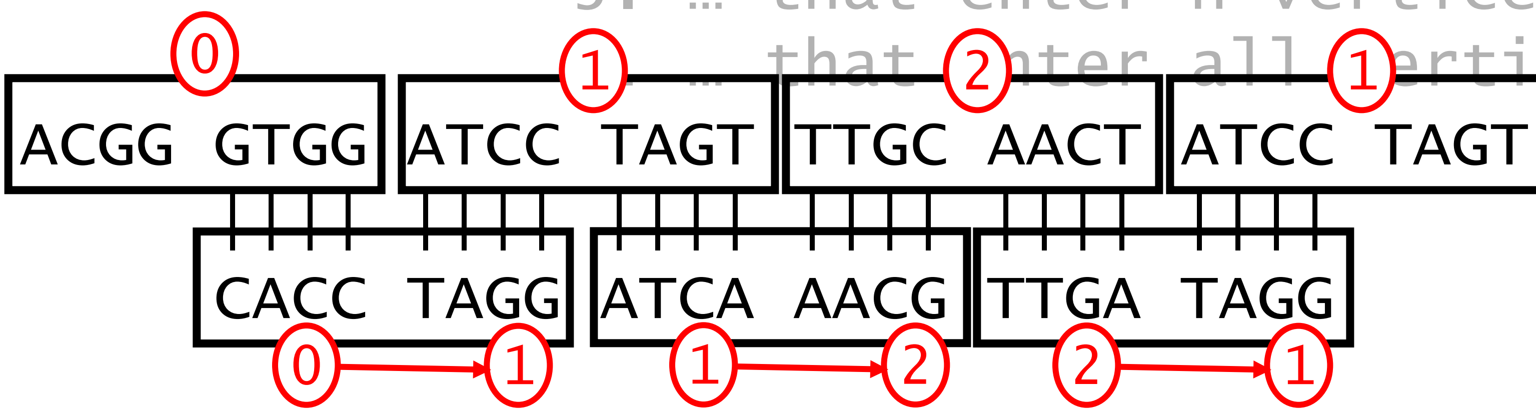
Adleman's algorithm



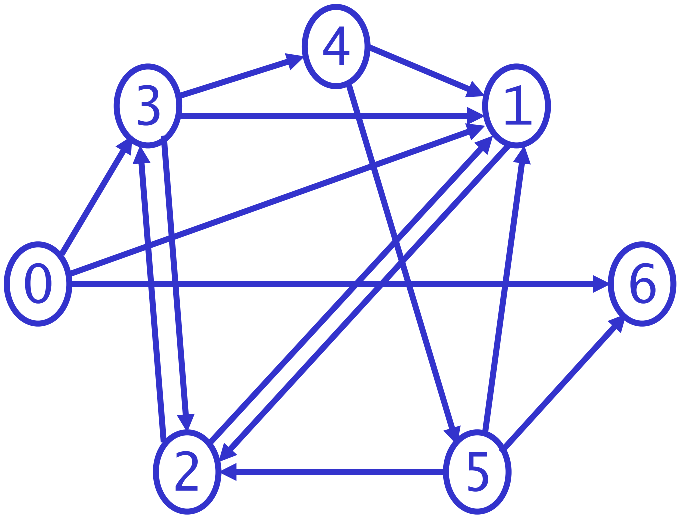
0. coding the graph
1. **generate 'all' paths**

   keep only paths

2. … from v_{in} to v_{out}
3. … that enter n vertices
4. … that enter all vertices



Adleman's algorithm



0. coding the graph
1. generate 'all' paths

   keep only paths

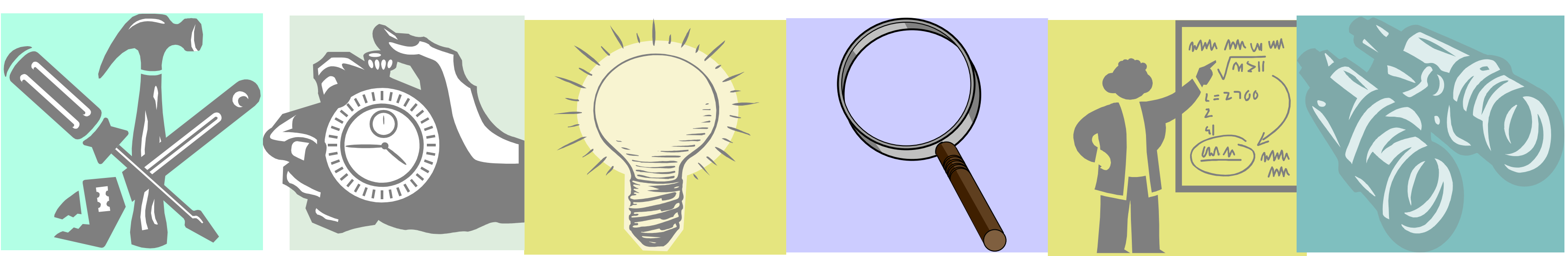
2. **… from v_{in} to v_{out}**
3. **… that enter n vertices**
4. **… that enter all vertices**

5. **if any path remains OK**

- PCR with v_{in} and v_{out} primers
- gel: separate on length, amplify & purify
- magnetic beads: select strands
- PCR amplification & gel

contents

- DNA … the tool chest
- problem complexity … P & NP
  Hamilton Path Problem
- Adleman's algorithm
- comments
- theory … Turing machine
- recent work + future



comments …

- “clear that the methods could be scaled up to … larger graphs”

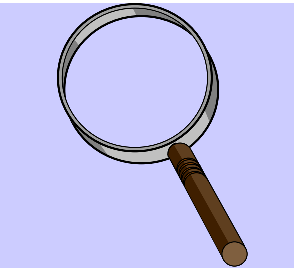
  + bath tub of DNA ?
  + suitable algorithms

- approximately 7 days of lab work

  + automation
  + alternative molecular algorithms

- possibility of errors

  + pseudopaths: accidental ligation
  + PCR, separation procedures
  + hairpin loops
  + stability when scaled



comments …

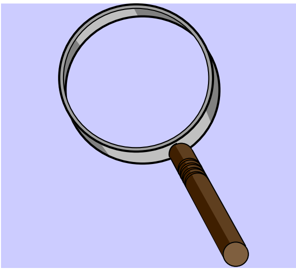
- “power of this method of computation”
  - 10^{14} operations 10^{20} plausable
  - exceed supercomputers by thousandfold

:)

- “not clear whether … used to solve real computational problems”

. multiplying 100 digit numbers

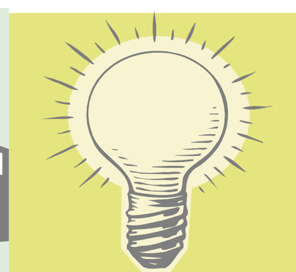
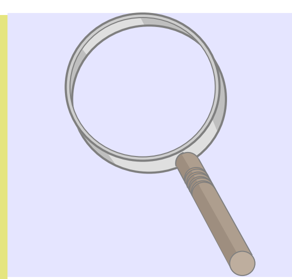
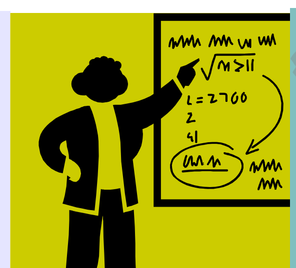
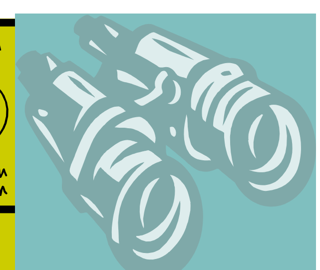
- potential: massively parallel searches



contents

- DNA … the tool chest
- problem complexity … P & NP
  Hamilton Path Problem
- Adleman’s algorithm
- comments
- theory … Turing machine
- recent work + future



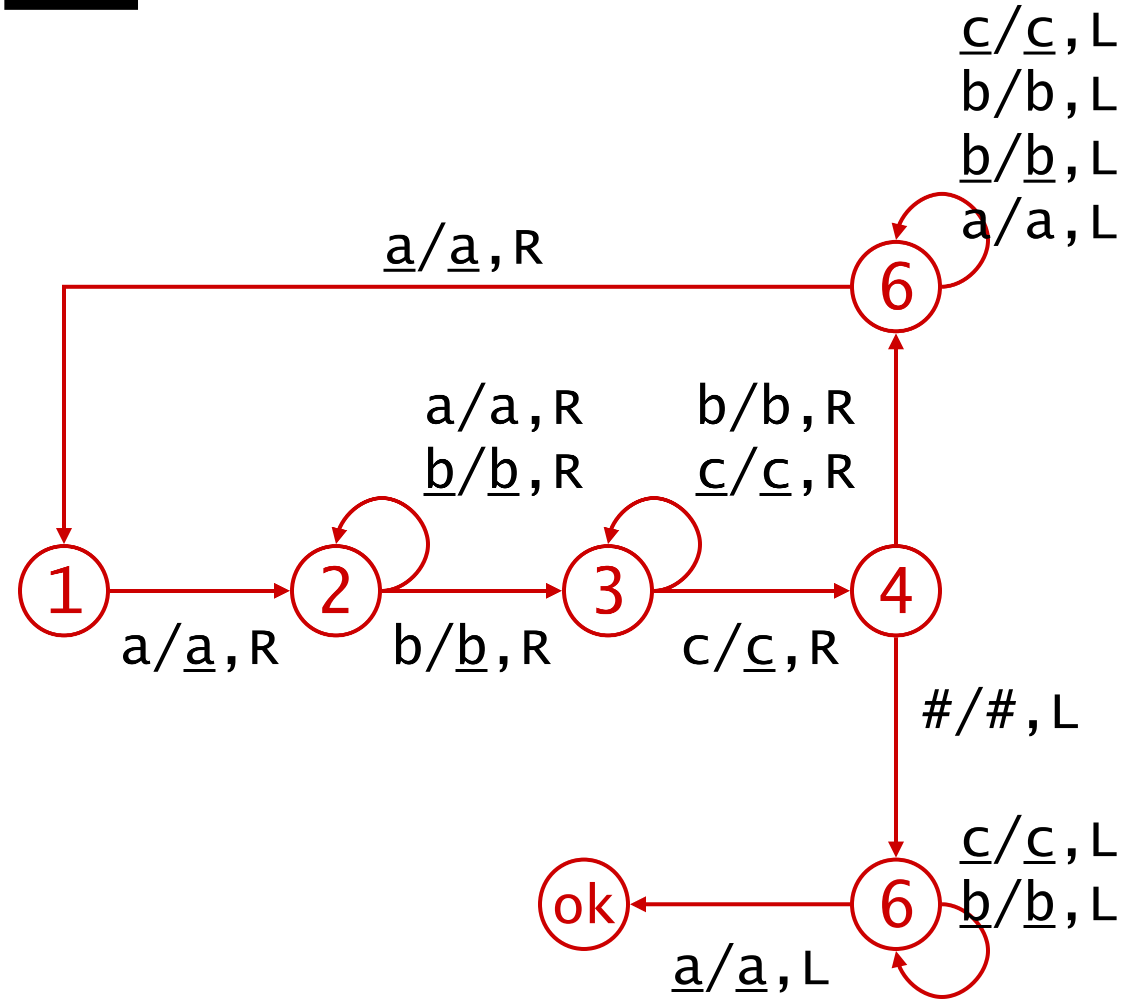





Turing machine

#	a	a	a	a	a	b	b	b	b	b	c	c	c	c	c	#

tape

1. mark a
2. move to b's
   mark b
3. move to c's
   mark c
4. if another c
5. then back to a's
   goto 1.
   else back to a's
6. check marks
   stop



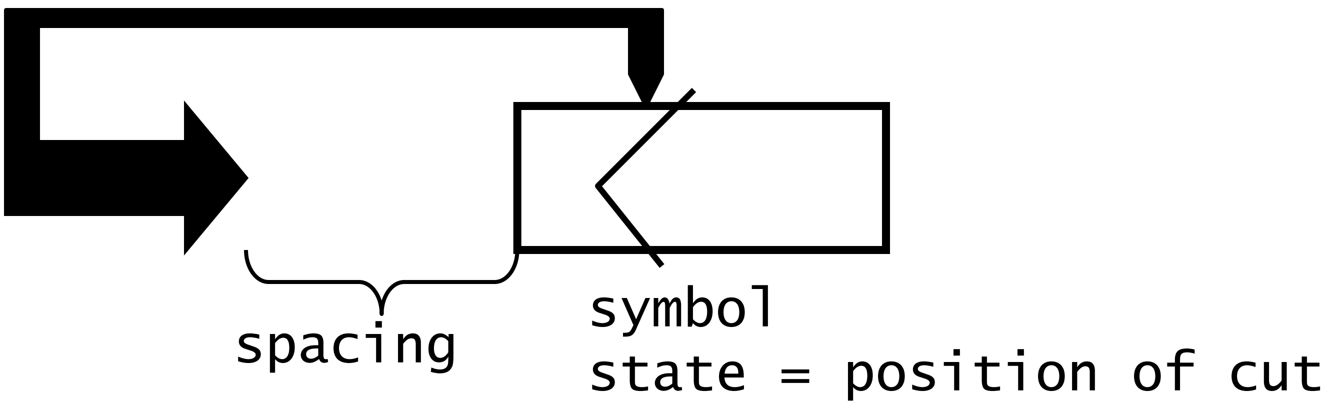
‘universal’ Turing machine

GGATGnnnnnnnnn
CCTACnnnnnnnnnnnnn

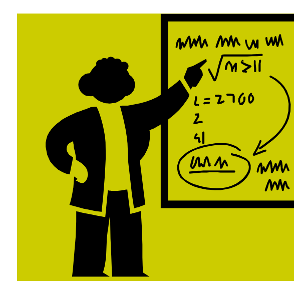
Rothemund

*Fok*I

circular DNA



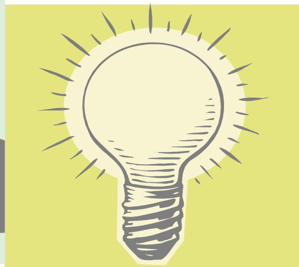
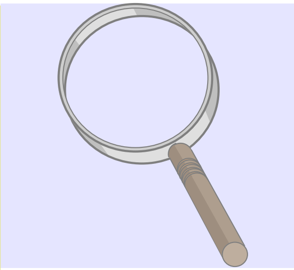
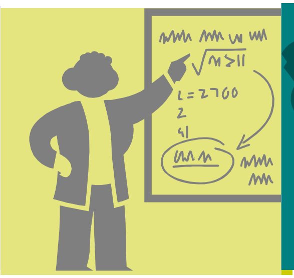
- cut states with restriction enzyme
- mix ‘instructions’ with ‘tape’
- ‘activate’ instructions (cut protected end)
- ligate to form circles
- cut old symbol
- recircularize



contents

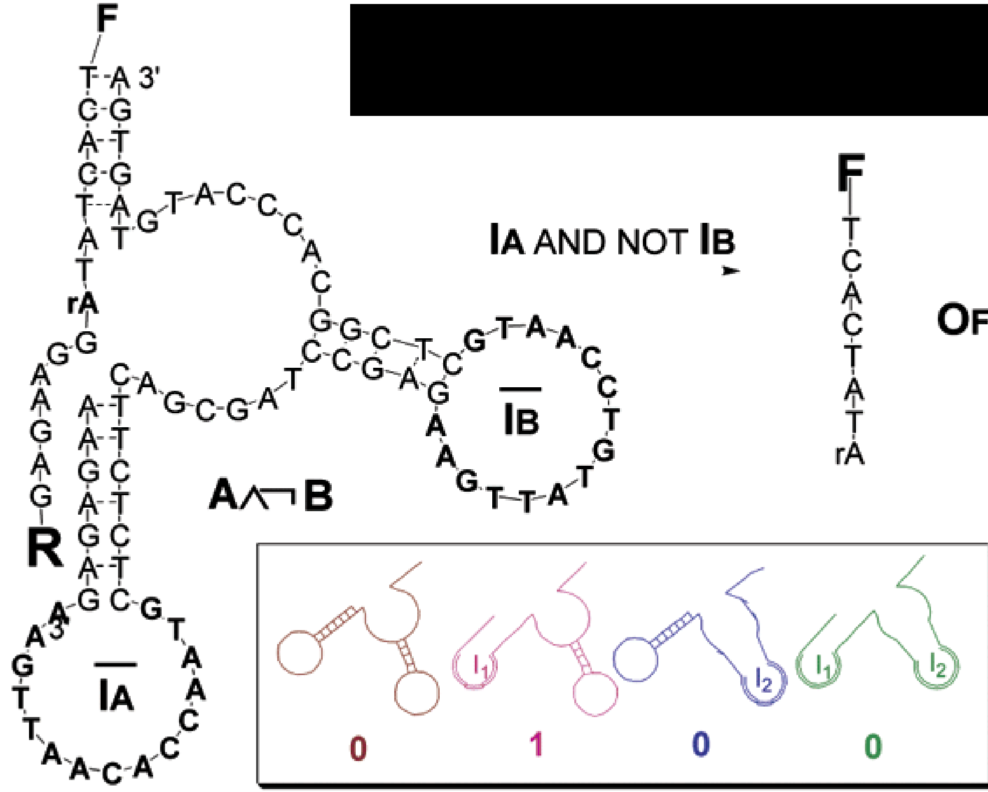
- DNA … the tool chest
- problem complexity … P & NP
  - Hamilton Path Problem
- Adleman's algorithm
- comments
- theory … Turing machine
- recent work + future

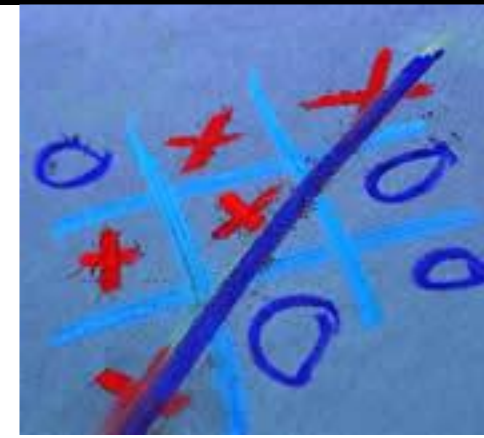


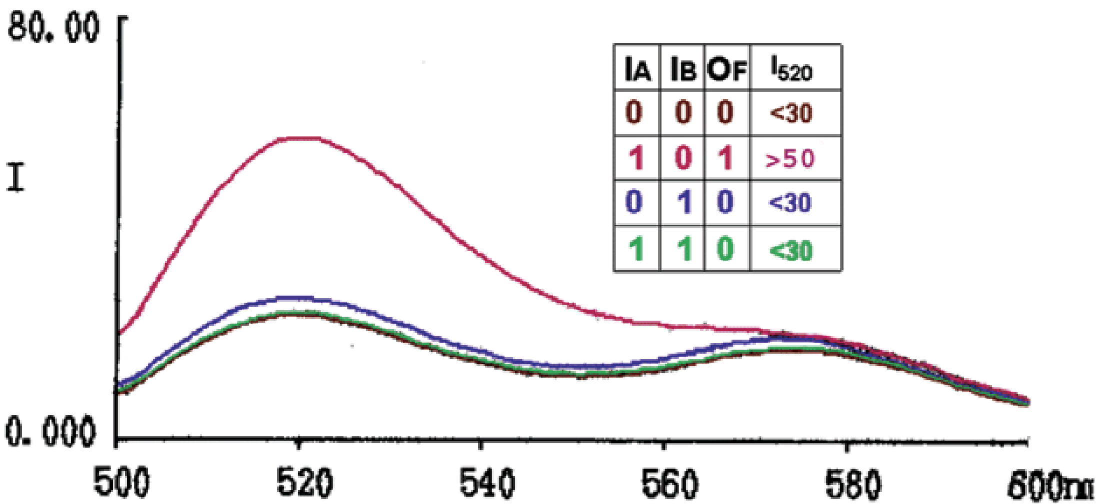

tic-tac-toe

(a)





(b)

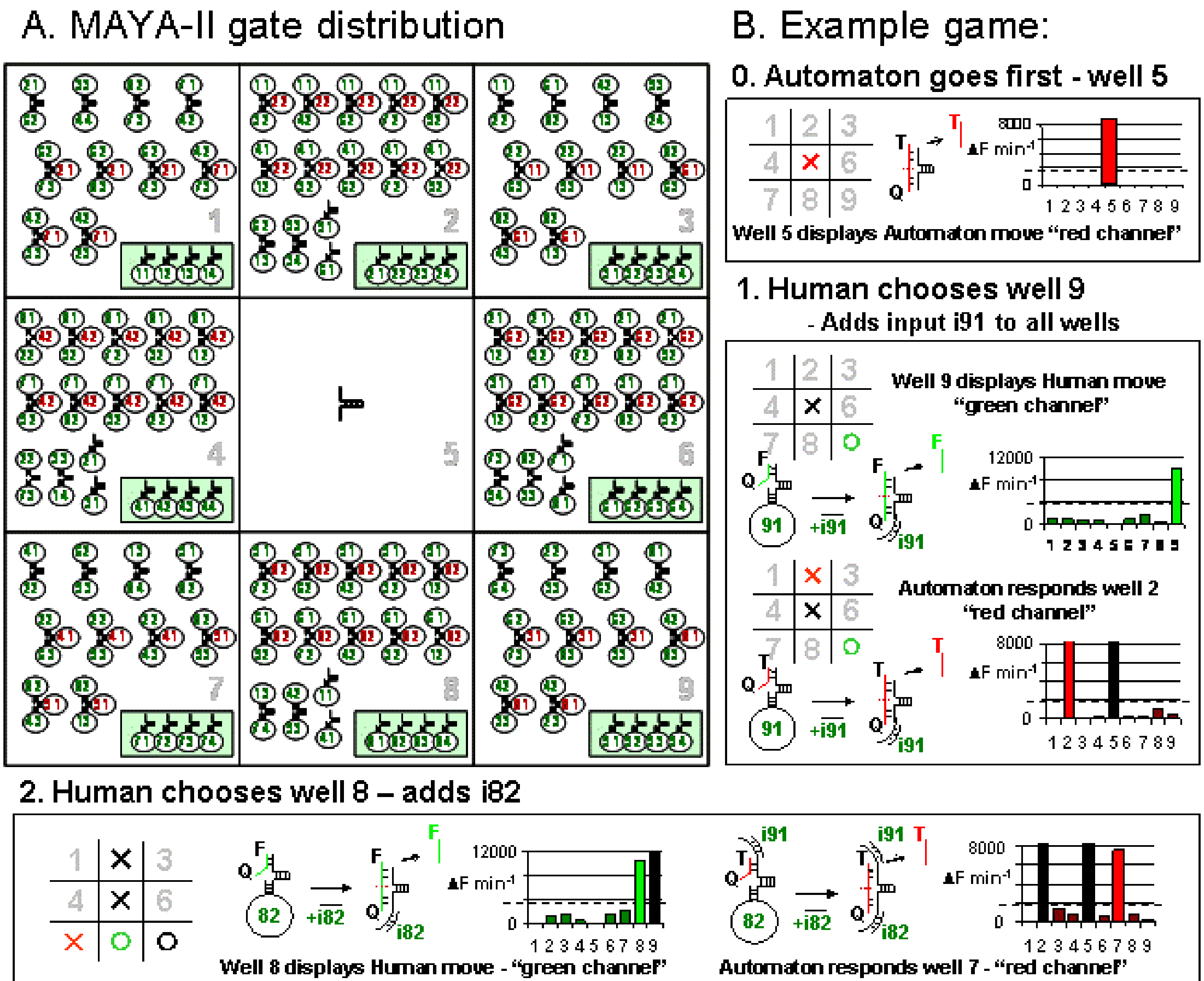


IA	IB	OF	I_{520}
0	0	0	<30
1	0	1	>50
0	1	0	<30
1	1	0	<30

logic gates
fluorescence

Stojanovic & Stefanovic, **Deoxyribozyme-Based Molecular Automaton**. *Nature Biotechn.* 2003. **Deoxyribozyme-Based Logic Gates** *J. Am. Chem. Soc*. 2002. **Medium Scale Integration of Molecular Logic Gates in an Automaton** *Nano Letters* 2006.

MAYA-II

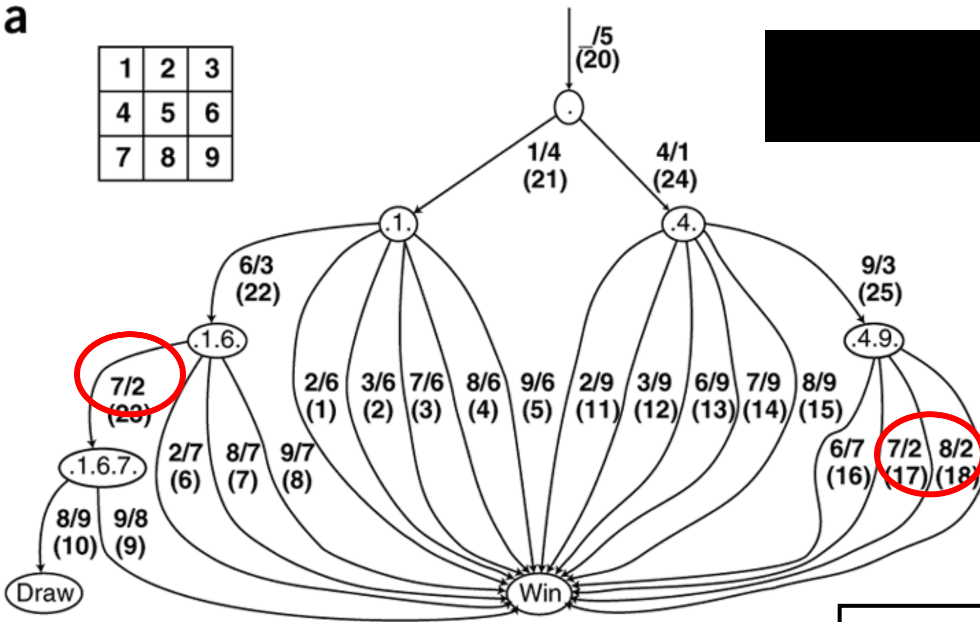


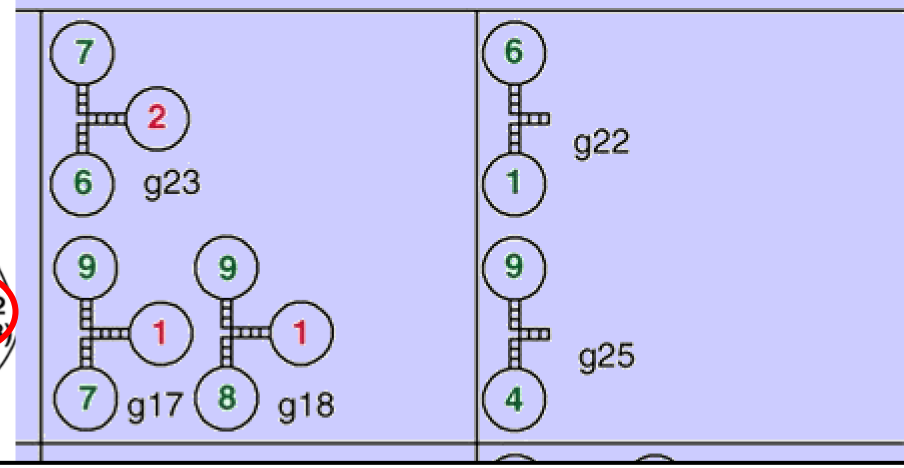
first "medium-scale integrated molecular circuit", integrating
128 deoxyribozyme-based logic gates, 32 input DNA molecules, and 8 two-channel fluorescent outputs across 8 wells

digamma.cs.unm.edu/wiki/bin/view/McogPublicWeb/MolecularAutomataMAYAII

tic-tac-toe

a





$$o_2 = (i_6 \wedge i_7 \wedge \neg i_2) \vee (i_7 \wedge i_9 \wedge \neg i_1) \vee (i_8 \wedge i_9 \wedge \neg i_1)$$

b

$$o_1 = \underbrace{i_4}_{\text{edge (24)}}$$

$$o_2 = \underbrace{(i_6 \wedge i_7 \wedge \neg i_2)}_{\text{edge (23)}} \vee \underbrace{(i_7 \wedge i_9 \wedge \neg i_1)}_{\text{edge (17)}} \vee \underbrace{(i_8 \wedge i_9 \wedge \neg i_1)}_{\text{edge (18)}}$$

$$o_3 = \underbrace{(i_1 \wedge i_6)}_{\text{edge (22)}} \vee \underbrace{(i_4 \wedge i_9)}_{\text{edge (25)}}$$

$$o_4 = \underbrace{i_1}_{\text{edge (21)}}$$

$$o_5 = \underbrace{1}_{\text{edge (20)}}$$

$$o_6 = \underbrace{(i_1 \wedge i_2 \wedge \neg i_6)}_{\text{edge (1)}} \vee \underbrace{(i_1 \wedge i_3 \wedge \neg i_6)}_{\text{edge (2)}} \vee \underbrace{(i_1 \wedge i_7 \wedge \neg i_6)}_{\text{edge (3)}} \vee \underbrace{(i_1 \wedge i_8 \wedge \neg i_6)}_{\text{edge (4)}} \vee \underbrace{(i_1 \wedge i_9 \wedge \neg i_6)}_{\text{edge (5)}}$$

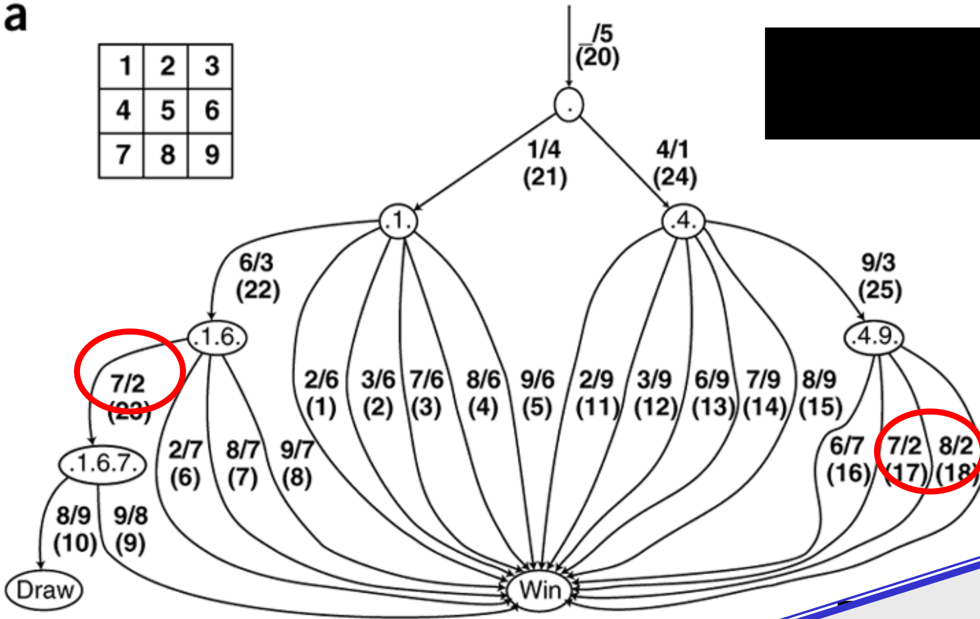
$$o_7 = \underbrace{(i_2 \wedge i_6 \wedge \neg i_7)}_{\text{edge (6)}} \vee \underbrace{(i_6 \wedge i_8 \wedge \neg i_7)}_{\text{edge (7)}} \vee \underbrace{(i_6 \wedge i_9 \wedge \neg i_7)}_{\text{edges (8) and (16)}} \vee \underbrace{(i_9 \wedge i_2 \wedge \neg i_1)}_{\text{edge (19)}}$$

$$o_8 = \underbrace{i_9 \wedge i_7 \wedge \neg i_4}_{\text{edge (9)}}$$

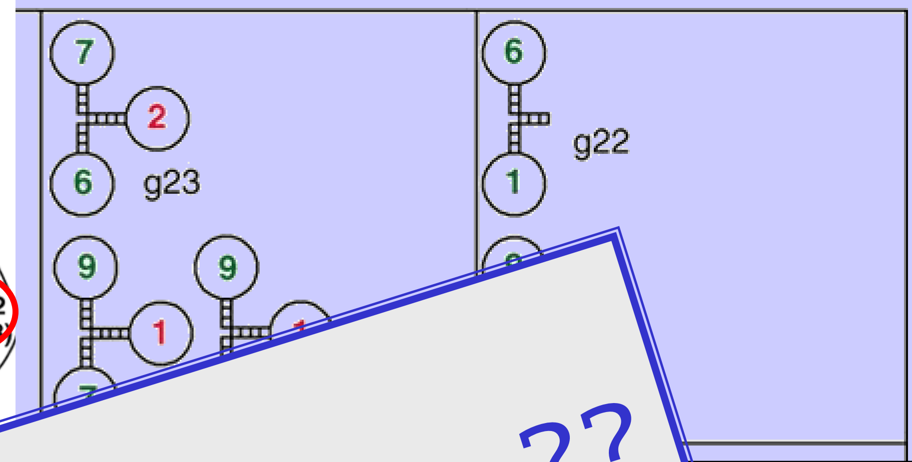
$$o_9 = \underbrace{(i_7 \wedge i_8 \wedge \neg i_4)}_{\text{edge (10)}} \vee \underbrace{(i_4 \wedge i_2 \wedge \neg i_9)}_{\text{edge (11)}} \vee \underbrace{(i_4 \wedge i_3 \wedge \neg i_9)}_{\text{edge (12)}} \vee \underbrace{(i_4 \wedge i_6 \wedge \neg i_9)}_{\text{edge (13)}} \vee \underbrace{(i_4 \wedge i_7 \wedge \neg i_9)}_{\text{edge (14)}} \vee \underbrace{(i_4 \wedge i_8 \wedge \neg i_9)}_{\text{edge (15)}}$$

i_1	TCT GCG TCT ATA AAT
i_2	ATC GTA TGT TGT TCA
i_3	GTA TAG TCT GTT TGT
i_4	G TAA GTG CTC AAA TGT C
i_5	G TCT AAT TCT CAC GGT C

a



tic-tac-toe



$_1)\vee(i_8\wedge i_9\wedge\neg i_1)$

eeuh …
is this a computer ??

b

$o_1 = \underbrace{i_4}_{\text{edge (24)}}$

$o_2 = \underbrace{(i_6 \wedge i_7 \wedge \neg i_2)}_{\text{edge (23)}} \vee$

$o_3 = \underbrace{(i_1 \wedge i_6)}_{\text{edge (22)}} \vee \underbrace{(i_4 \wedge i_9)}_{\text{edge (25)}}$

$o_4 = \underbrace{i_1}_{\text{edge (21)}}$

$o_5 = \underbrace{1}_{\text{edge (20)}}$

$o_6 = \underbrace{(i_1 \wedge i_2 \wedge \neg i_6)}_{\text{edge (1)}} \vee \underbrace{(i_1 \wedge i_3 \wedge \neg i_6)}_{\text{edge (2)}} \vee \underbrace{(i_1 \wedge i_7 \wedge \neg i_6)}_{\text{edge (3)}} \vee \underbrace{(i_1 \wedge i_8 \wedge \neg i_6)}_{\text{edge (4)}} \vee \underbrace{(i_1 \wedge i_9 \wedge \neg i_6)}_{\text{edge (5)}}$

$o_7 = \underbrace{(i_2 \wedge i_6 \wedge \neg i_7)}_{\text{edge (6)}} \vee \underbrace{(i_6 \wedge i_8 \wedge \neg i_7)}_{\text{edge (7)}} \vee \underbrace{(i_6 \wedge i_9 \wedge \neg i_7)}_{\text{edges (8) and (16)}} \vee \underbrace{(i_9 \wedge i_2 \wedge \neg i_1)}_{\text{edge (19)}}$

$o_8 = \underbrace{i_9 \wedge i_7 \wedge \neg i_4}_{\text{edge (9)}}$

$o_9 = \underbrace{(i_7 \wedge i_8 \wedge \neg i_4)}_{\text{edge (10)}} \vee \underbrace{(i_4 \wedge i_2 \wedge \neg i_9)}_{\text{edge (11)}} \vee \underbrace{(i_4 \wedge i_3 \wedge \neg i_9)}_{\text{edge (12)}} \vee \underbrace{(i_4 \wedge i_6 \wedge \neg i_9)}_{\text{edge (13)}} \vee \underbrace{(i_4 \wedge i_7 \wedge \neg i_9)}_{\text{edge (14)}} \vee \underbrace{(i_4 \wedge i_8 \wedge \neg i_9)}_{\text{edge (15)}}$

i_1 TCT GCG TCT ATA AAT
i_2 ATC GTA TGT TGT TCA
i_3 GTA TAG TCT GTT TGT
i_4 G TAA GTG CTC AAA TGT C
i_5 G TCT AAT TCT CAC GGT C

DNA computing after 10 years

"There are many practical hurdles. Even with the best techniques of today, DNA still lags behind silicon computers," says Ehud Shapiro. Instead, he advocates creating DNA devices that can do things, and go to places, that silicon can't - such as inside our cells, to make and control drugs.

...

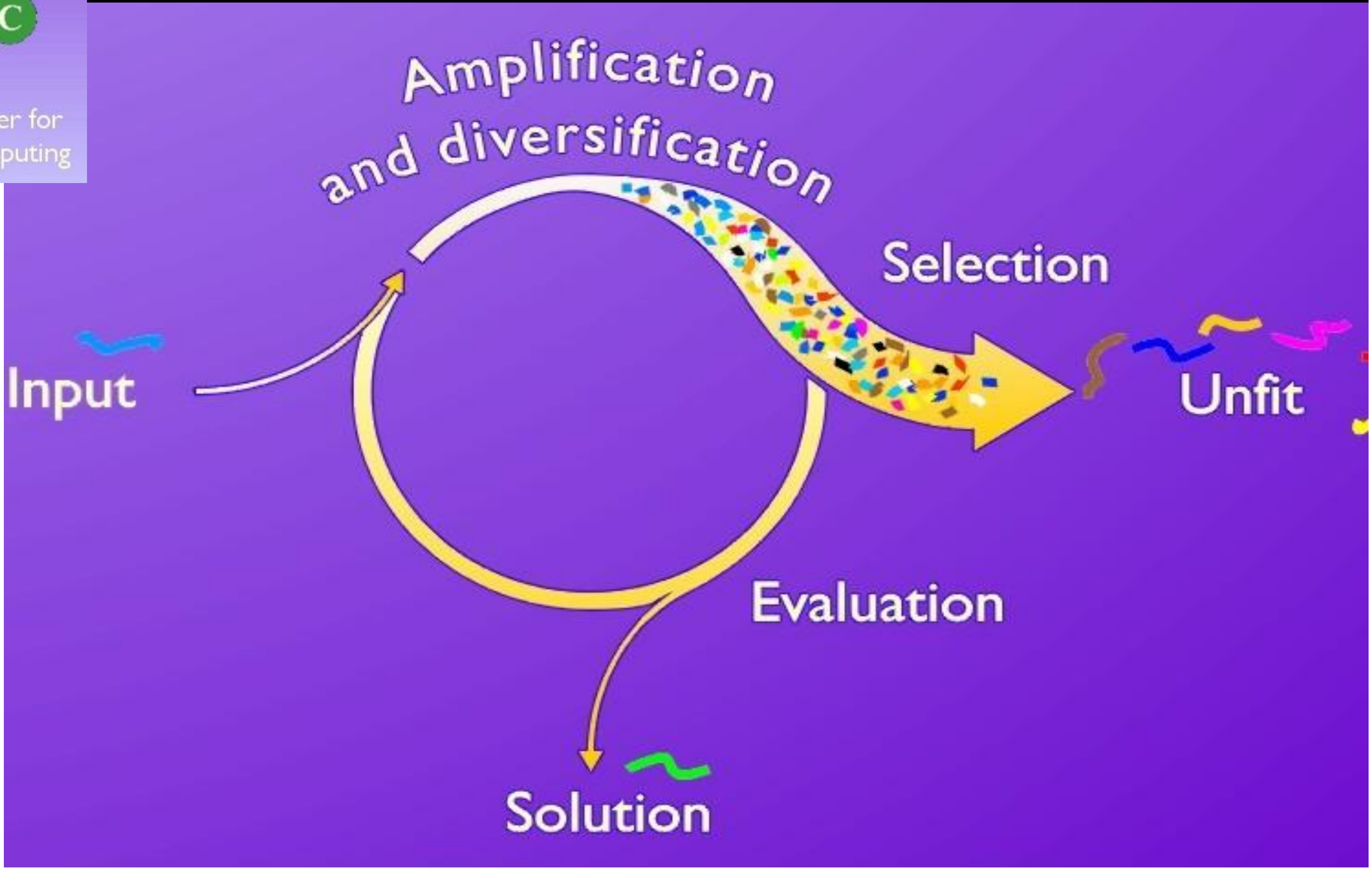
Ultimately, Seeman hopes to build DNA scaffolding for electrical circuits, or for other molecular machines.

...

Yurke is focusing on DNA machines with moving parts. In 2000, he and his colleagues devised a set of DNA tweezers ....

Nature News Service april 2003

evolutionary DNA computing



Cross-fertilization between evolutionary computation and DNA-based computing T.Back; J.N. Kok; G. Rozenberg
Proceedings 1999 Evolutionary Computation.

links van Tom

Researchers make significant advances in molecular computing, *University of Kent*, 10-Dec-2009
http://www.kent.ac.uk/news/stories/dchu/2009

Dr Chu explained: 'Our research demonstrates that the speed of bio-molecular computers is fundamentally limited by their metabolic rate or their ability to process energy. One of our main findings is that a molecular computer has to balance a trade-off between the speed with which a computation is performed and the accuracy of the result. However, a molecular computer can increase both the speed and reliability of a computation by increasing the energy it invests in the computation. With molecular computers this energy may be derived from food sources.'

links van Tom

DNA computer 'ansers questions', *BBC News*, 05-Aug-2009
http://news.bbc.co.uk/2/hi/technology/8184033.stm

... they tried the system with simple "if… then…" propositions. One of these went as follows: "All men are mortal. Socrates is a man. Therefore, Socrates is mortal."

The answer was encoded in a flash of green light. Some of the DNA strands were equipped with a naturally glowing fluorescent molecule bound to a second molecule which keeps the light covered.

The system can take in facts and rules as a computer file of simple text. The robotic "compiler" can then turn those facts and rules into the DNA starting products of a logical query.

In other words, computers that go to work inside a cell.

links van Tom

Future directions in computing: DNA Computing, *BBC News*, 13 Nov 2007
http://news.bbc.co.uk/2/hi/technology/7085154.stm

"This soup of DNA and enzymes implements a well know mathematical model of computation known as finite automaton," he explained.
"This finite automaton knows how to do very simple computation such as recognising whether a list of zeros and ones has an even number of ones."

In the case of his 2004 computer this method of computation was used to analyze ratios of specific molecules related to prostate cancer and a specific type of lung cancer.

The "computer" consisted of a chain of three segments of DNA and an enzyme which could cut the strands.

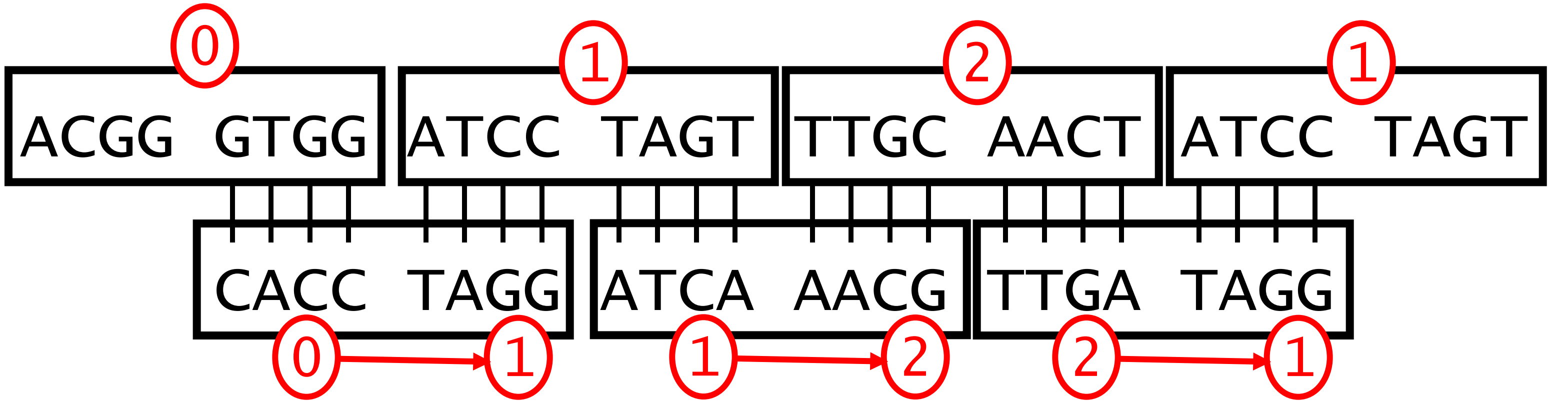
DNA logic gates herald injectable computers, *New Scientist*, 02 June 2010
http://www.newscientist.com/article/dn18989-dna-logic-gates-herald-injectable-computers.html

"The biocomputer would sense biomarkers and immediately react by releasing counter-agents for the disease," says Itamar Willner, who led the work.

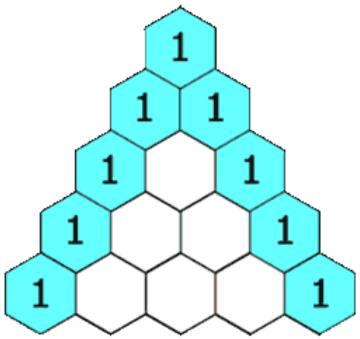
The new logic gates are formed from short strands of DNA and their complementary strands, …. Two strands act as the input: each represents a 1 when present or a 0 when absent. … Take the "exclusive OR" or XOR logic gate. It produces an output when either of the two inputs is present but not when both are present or both are absent.

Willner and his team added molecules to both the complementary strands that caused them to fluoresce when each was present in isolation, representing a logical 1 as the output. But when both were present, the complementary strands combined and quenched the fluorescence, representing a 0 output.

self assembly



Sierpinski triangle



1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
1 6 15 20 15 6 1
1 7 21 35 35 21 7 1
1 8 28 56 70 56 28 8 1

Pascal's triangle

addition

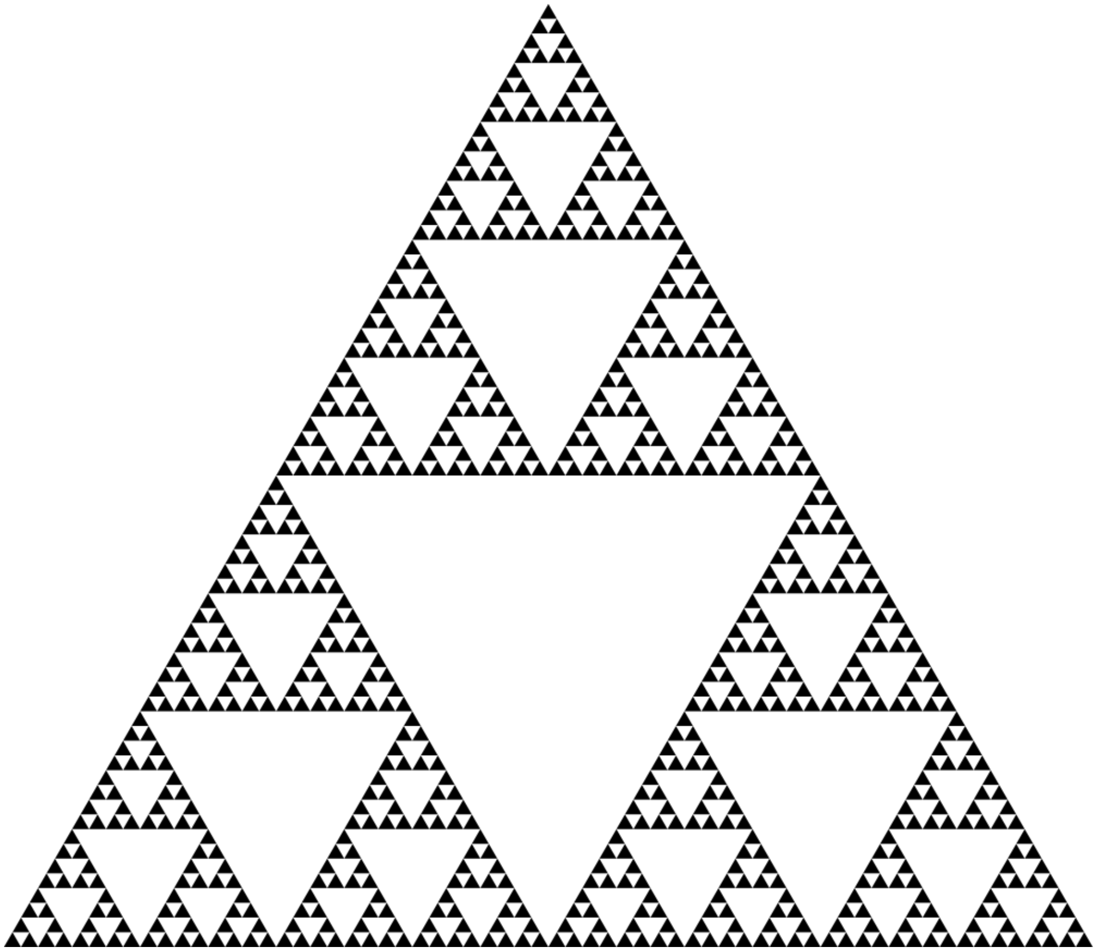
1
1 1
1 0 1
1 1 1 1
1 0 0 0 1
1 1 0 0 1 1
1 0 1 0 1 0 1
1 1 1 1 1 1 1 1
1 0 0 0 0 0 0 0 1

Sierpinski triangle

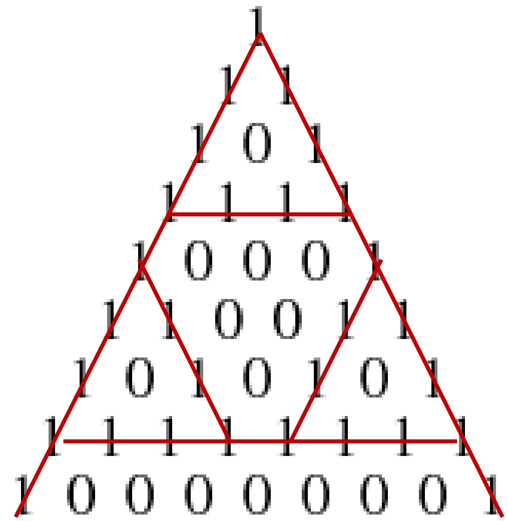
$\oplus$ XOR
even / odd

$\oplus$	0	1
0	0	1
0	1	0

Sierpinski triangle



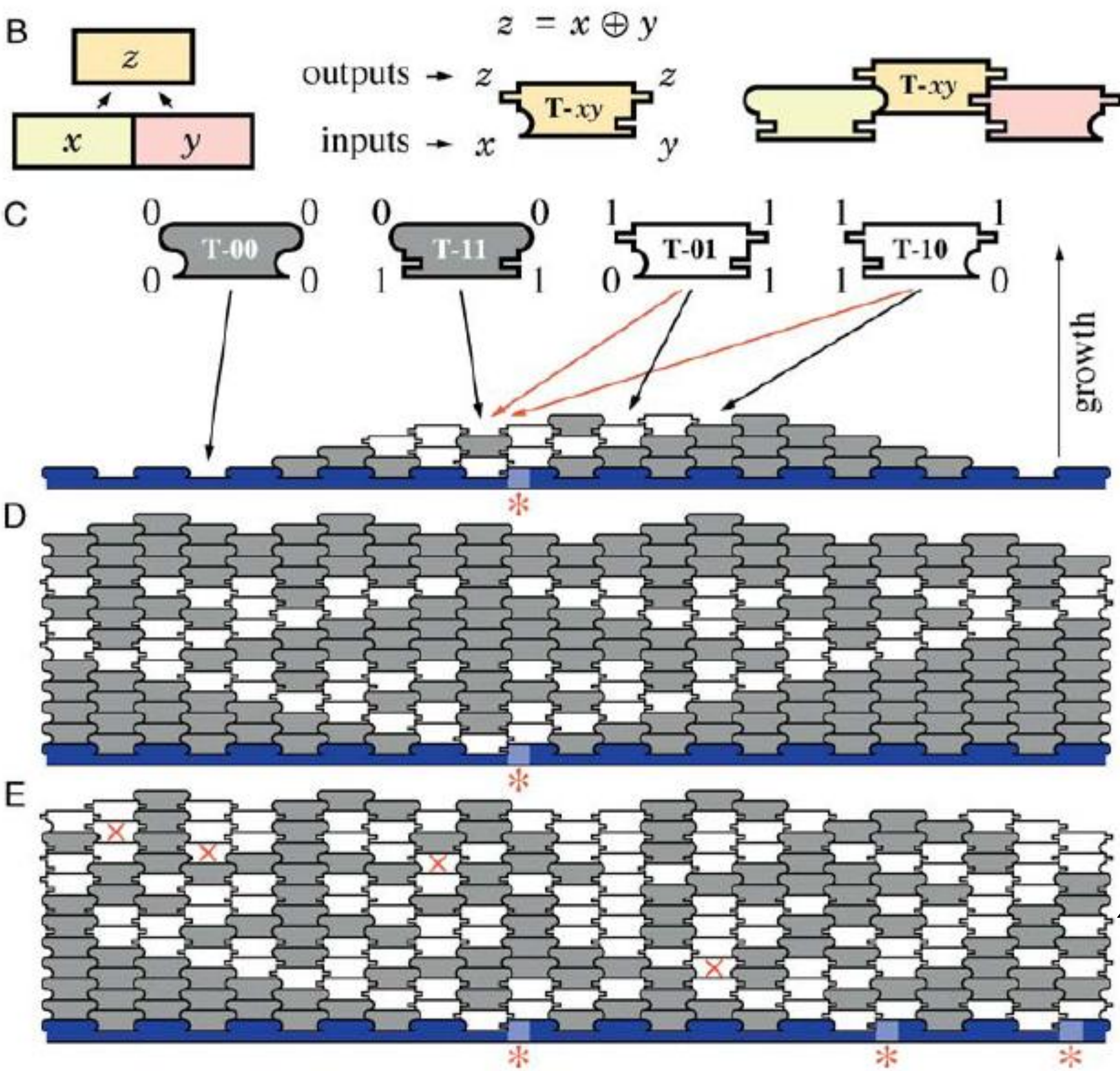
‘fractal’

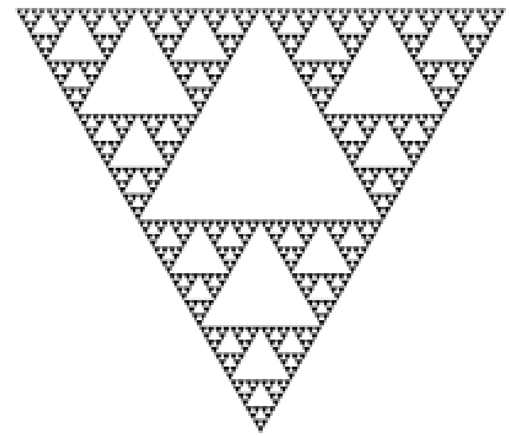


Sierpinski triangle

⊕ XOR
even / odd

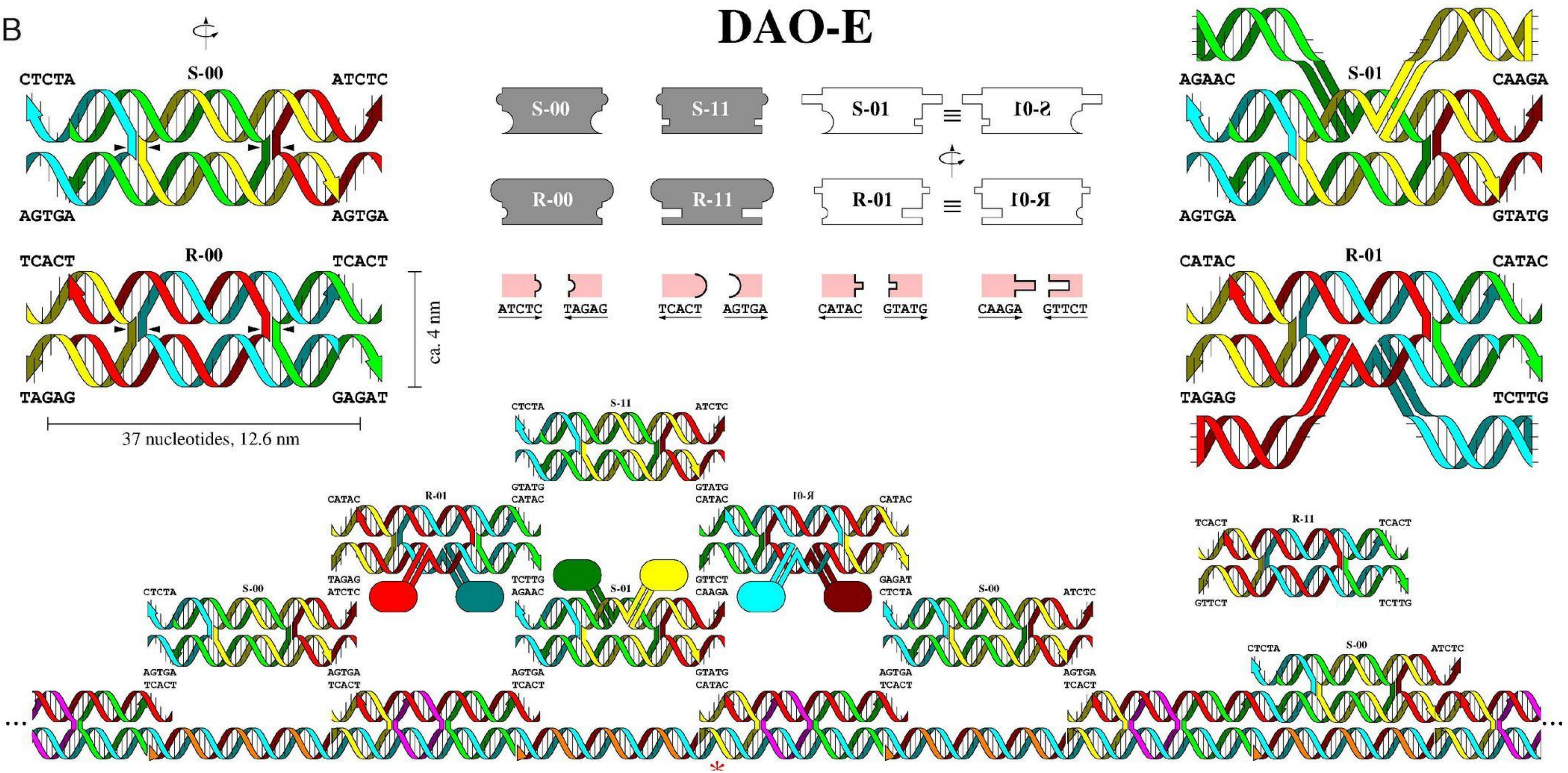
self assembly: Sierpinski





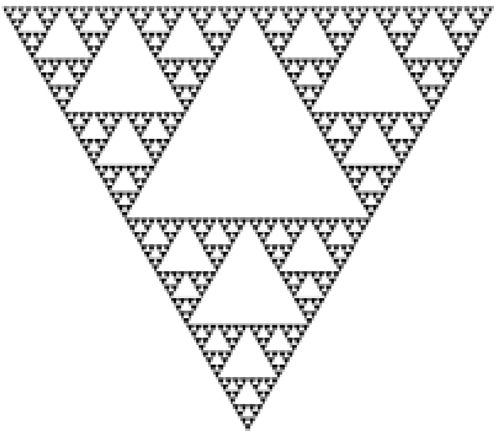
Algorithmic Self-Assembly of DNA Sierpinski Triangles, Rothemund, Papadakis, Winfree; PLoS Biology (2004)

self assembly

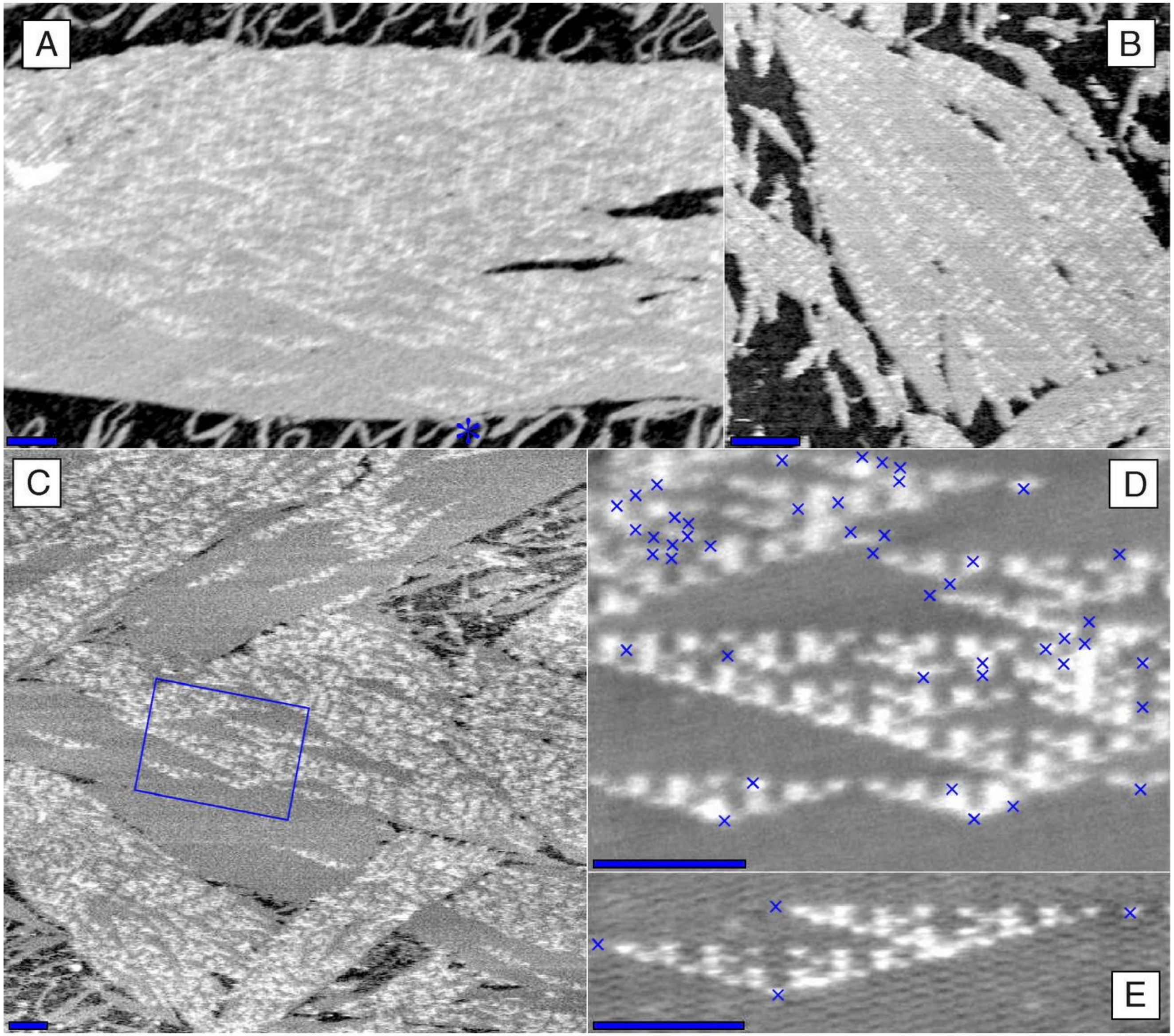


http://dx.doi.org/10.1371/journal.pbio.0020424
Algorithmic Self-Assembly of DNA Sierpinski Triangles
Rothemund, Papadakis, Winfree; PLoS Biology (2004)

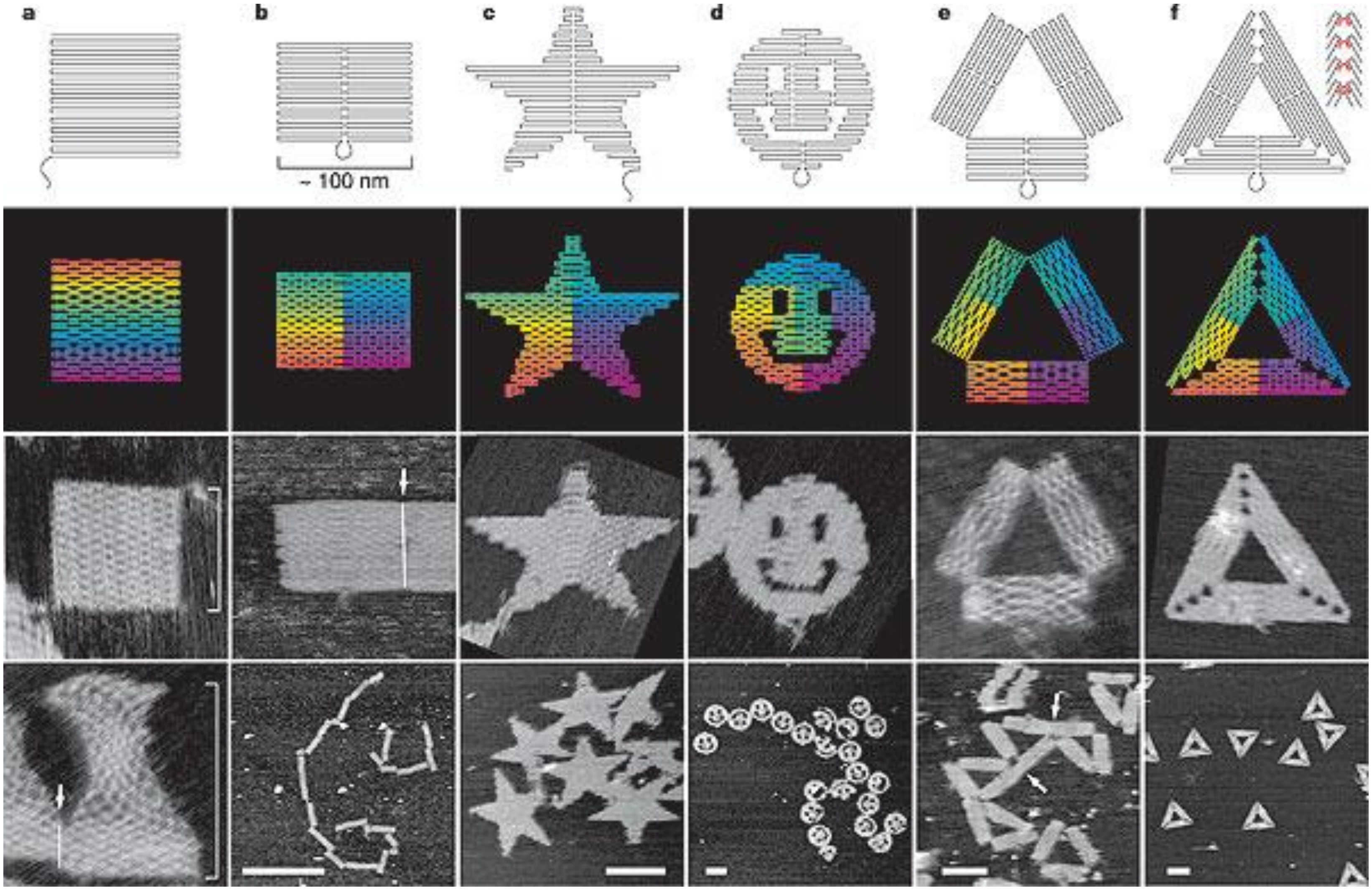
self assembly



Sierpinski

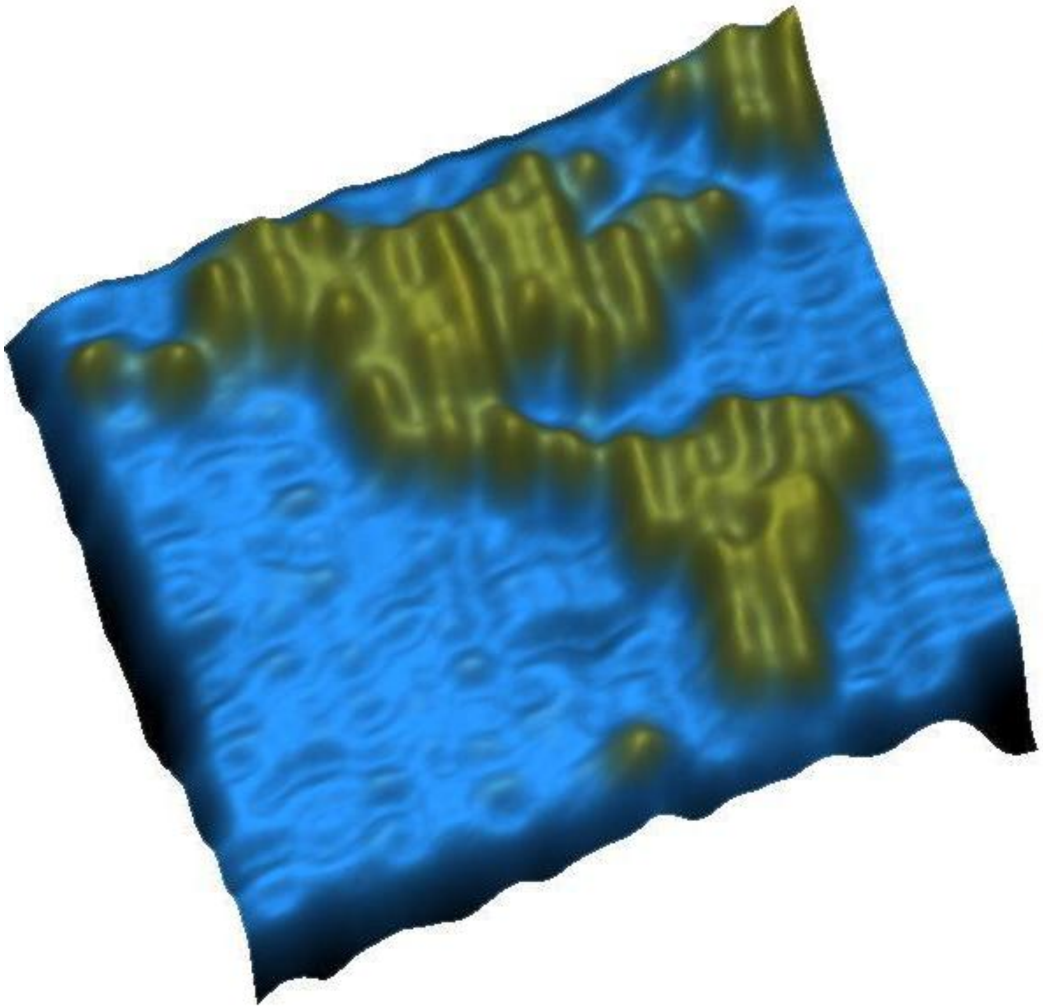


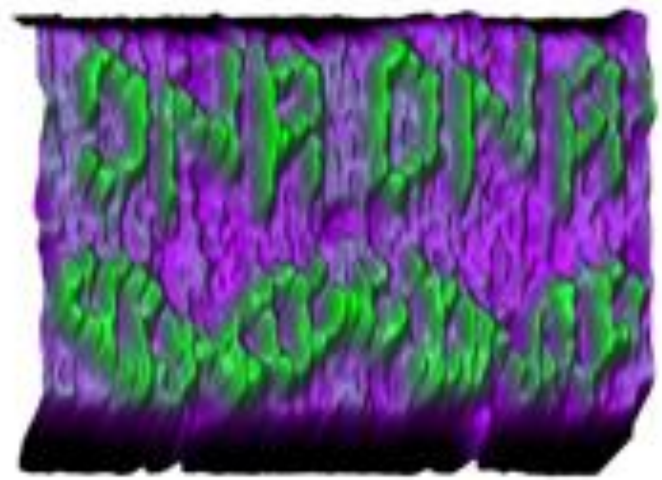
self assembly: DNA origami



Folding DNA to create nanoscale shapes and patterns
Paul W. K. Rothemund, Nature 440, 297-302 (16 March 2006)

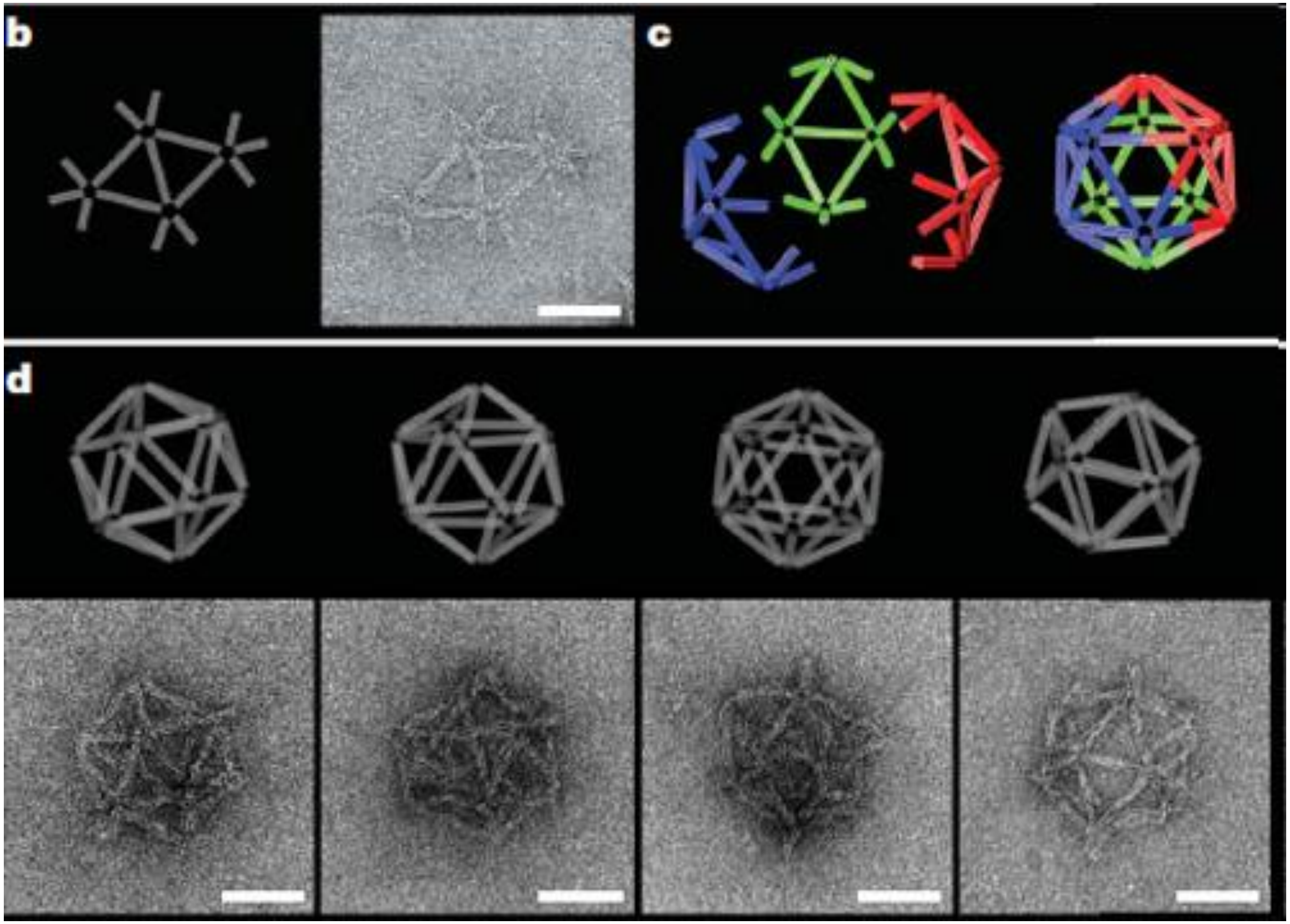
Self Assembly: DNA origami





Paul W. K. Rothemund, http://www.dna.caltech.edu/~pwkr/

3D DNA origami



Self-assembly of DNA into nanoscale three-dimensional shapes
S.M. Douglas, H. Dietz, T. Liedl, B. Hogberg, F. Graf, W.M. Shih,
Nature 459, 414-418 (21 May 2009)

conclusion

take home message

DNA can be used for applications it was not "intended" for

computing
a very interesting proof of concept

find niche

[redacted]