In Vitro Molecular Evolution

GECCO-2005 Tutorial
June 26, 2005, Washington, D.C.

Tak Zhang

Biointelligence Laboratory
School of Computer Science and Engineering
Seoul National University
Seoul 151-742, Korea

btzhang@cse.snu.ac.kr
http://bi.snu.ac.kr/

Natural Computation

- Neural Computation
  - ♦ A network of neurons
- Evolutionary Computation
  - ♦ A population of chromosomes
- Molecular Computation
  - ♦ A test tube of molecules
- Molecular Evolutionary Computation ← This tutorial
  - ♦ A test tube of "evolving" molecules
  - ♦ "In vitro molecular evolution"

Scope of This Tutorial

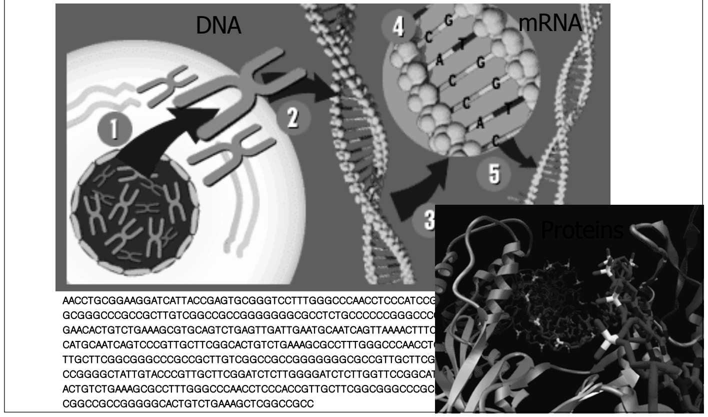
- What is "in vitro evolution"?
- How do we exploit this as EC technology, i.e. for molecular evolutionary computation (MEC)?
- What new opportunities this offers to EC research ers?
  - ♦ In theory and in applications
  - ♦ In science and in technology
- What challenges the new applications face?
- Where can I find more materials?

Overview

- Molecular Computation (without Evolution) ……………………………….. 5
  - ♦ Benett, Conrad, Adleman
  - ♦ Molecular Computing Examples
- In Vitro Evolution (without Computation) …………………………………33
  - ♦ Eigen's Molecular Theory of Evolution
  - ♦ In Vitro Molecular Evolution
- Molecular Evolutionary Computation (MEC) in Vitro: Theory …………… 50
  - ♦ The Theory of Bayesian Evolution
  - ♦ The Probabilistic Library Model (PLM)
- Molecular Programming (MP): In Vitro Evolution of Genetic Programs … 68
  - ♦ Evolving Genetic Programs in a Test Tube
  - ♦ Empirical Results
- New Issues for EC Researchers ……………………………………….……96
  - ♦ Research Issues
  - ♦ New Applications
- Books and Web Sites …………………………………………………...... 108

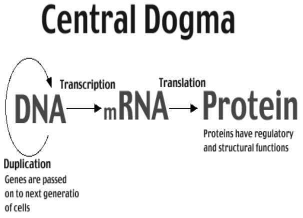
Molecular Computation (without Evolution)

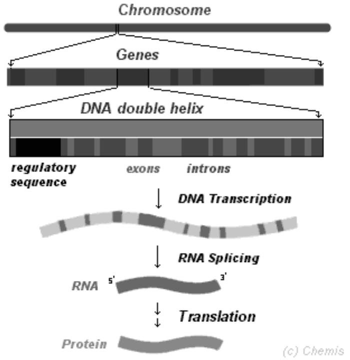
Biomolecular Information Processing



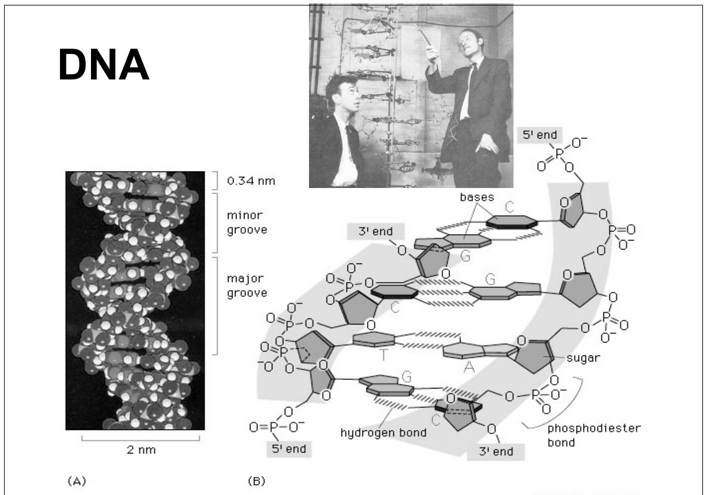
AACCTGCGGAAGGATCATTACCGAGTGCGGGTCCTTTGGGCCCAACCTCCCATCCG
GCGGGCCCGCCGCTTGTCGGCCGCCGGGGGGGCGCCTCTGCCCCCCGGGCCC
GAACACTGTCTGAAAGCGTGCAGTCTGAGTTGATTGAATGCAATCAGTTAAAACTTTC
CATGCAATCAGTCCCGTTGCTTCGGCACTGTCTGAAAGCGCCTTTGGGCCCAACCT
TTGCTTCGGCGGGCCCGCCGCTTGTCGGCCGCCGGGGGGGCGCCGTTGCTTCG
CCGGGGCTATTGTACCCGTTGCTTCGGATCTCTTGGGGATCTCTTGGTTCCGGCAT
ACTGTCTGAAAGCGCCTTTGGGCCCAACCTCCCACCGTTGCTTCGGCGGGCCCGC
CGGCCGCCGGGGGCACTGTCTGAAAGCTCGGCCGCC

Chromosomes, Genes, DNA, RNA, Proteins, and the Central Dogma


Central Dogma
Transcription
Translation
DNA
mRNA
Protein
Proteins have regulatory and structural functions
Duplication
Genes are passed on to next generatio of cells



DNA

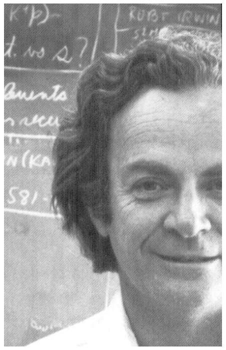


©1998 GARLAND PUBLISHING

Molecular Computing: A Brief History

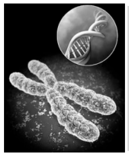
- Feynman (1959)
  - ♦ Potential of molecules
- Benett (1982)
  - ♦ DNA and thermodynamic computation
- Seeman (1991)
  - ♦ Self-assembly of a DNA cube
- Conrad (1992)
  - ♦ Lock-and-key paradigm for molecular computing
- Adleman (1994)
  - ♦ Experimental demonstration of DNA computing

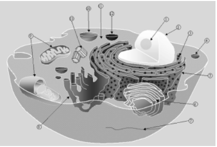
Feynman (1959)



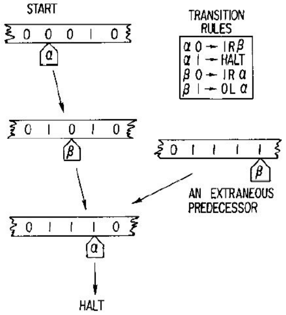
- *“There’s Plenty of Room at the Bott om”*
- Biological molecules can carry enor mous amounts of information in an e xceedingly small space.

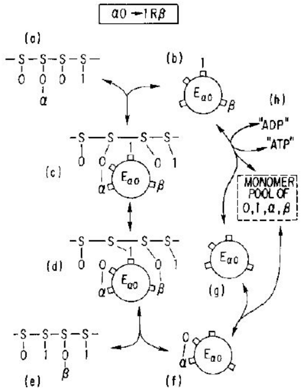
→ Inborn computing power!



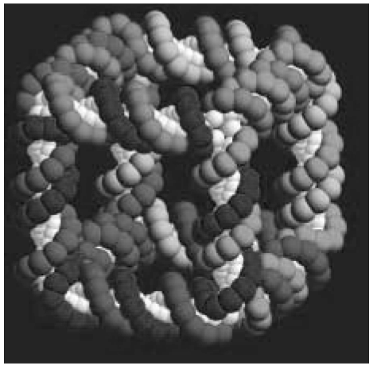


Benett (1982)



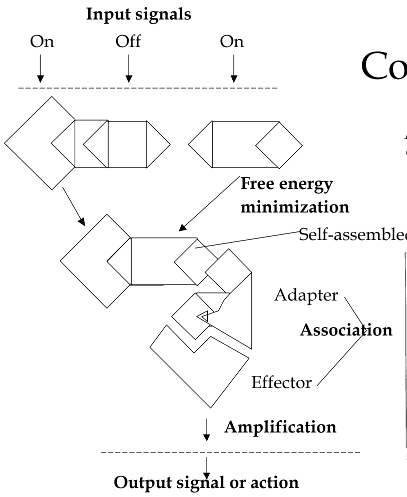


Seeman (1991)



Conrad (1992)

Molecular Computing: The Lock-Key Paradigm



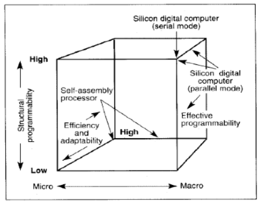
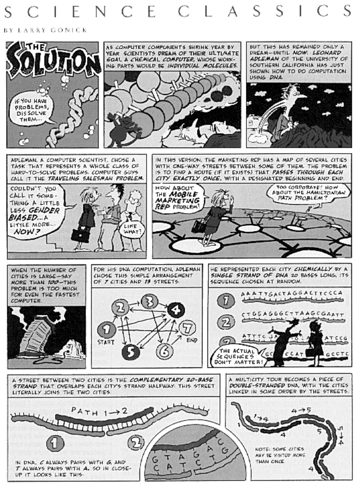
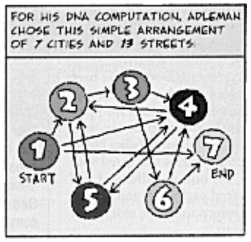

Silicon digital computer (serial mode)
High
Structural programmability
Self-assembly processor
Silicon digital computer (parallel mode)
Effective programmability
Efficiency and adaptability
High
Low
Micro
Macro

Figure 5. Trade-off cube.

Adleman (1994)


SCIENCE CLASSICS
BY LARRY GONICK
THE SOLUTION
IF YOU HAVE PROBLEMS, DISSOLVE THEM...
AS COMPUTER COMPONENTS SHRINK YEAR BY YEAR SCIENTISTS DREAM OF THEIR ULTIMATE GOAL A CHEMICAL COMPUTER, WHOSE WORKING PARTS WOULD BE INDIVIDUAL MOLECULES.
BUT THIS HAS REMAINED ONLY A DREAM—UNTIL NOW. LEONARD ADLEMAN OF THE UNIVERSITY OF SOUTHERN CALIFORNIA HAS JUST SHOWN HOW TO DO COMPUTATION USING DNA.
ADLEMAN, A COMPUTER SCIENTIST, CHOSE A TASK THAT REPRESENTS A WHOLE CLASS OF HARD-TO-SOLVE PROBLEMS. COMPUTER GUYS CALL IT THE TRAVELING SALESMAN PROBLEM.
COULDN'T YOU CALL IT SOMETHING A LITTLE LESS GENDER BIASED... A LITTLE MORE... NOW?
LIKE WHAT?
IN THIS VERSION, THE MARKETING REP HAS A MAP OF SEVERAL CITIES WITH ONE-WAY STREETS BETWEEN SOME OF THEM. THE PROBLEM IS TO FIND A ROUTE (IF IT EXISTS) THAT PASSES THROUGH EACH CITY EXACTLY ONCE, WITH A DESIGNATED BEGINNING AND END.
HOW ABOUT THE MOBILE MARKETING REP PROBLEM!
TOO CORPORATE! HOW ABOUT THE HAMILTONIAN PATH PROBLEM?
WHEN THE NUMBER OF CITIES IS LARGE—SAY MORE THAN 100—THIS PROBLEM IS TOO MUCH FOR EVEN THE FASTEST COMPUTER.
FOR HIS DNA COMPUTATION, ADLEMAN CHOSE THIS SIMPLE ARRANGEMENT OF 7 CITIES AND 13 STREETS.
START
END
HE REPRESENTED EACH CITY CHEMICALLY BY A SINGLE STRAND OF DNA 20 BASES LONG, ITS SEQUENCE CHOSEN AT RANDOM.
AAATTGACTAGGACTTCCCA
CTGGAGGGCTTAAGCGAATT
THE ACTUAL SEQUENCES DON'T MATTER!
A STREET BETWEEN TWO CITIES IS THE COMPLEMENTARY 20-BASE STRAND THAT OVERLAPS EACH CITY'S STRAND HALFWAY. THIS STREET LITERALLY JOINS THE TWO CITIES.
PATH 1→2
IN DNA, C ALWAYS PAIRS WITH G, AND T ALWAYS PAIRS WITH A, SO IN CLOSE-UP IT LOOKS LIKE THIS:
A MULTICITY TOUR BECOMES A PIECE OF DOUBLE-STRANDED DNA, WITH THE CITIES LINKED IN SOME ORDER BY THE STREETS.
NOTE: SOME CITIES MAY BE VISITED MORE THAN ONCE.


FOR HIS DNA COMPUTATION, ADLEMAN CHOSE THIS SIMPLE ARRANGEMENT OF 7 CITIES AND 13 STREETS.
1
2
3
4
5
6
7
START
END

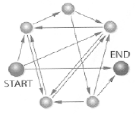
Discover magazine published an art icle in comic strip format about Leo nard Adleman's discovery of DNA computation. Not only entertainin g, but also the most understandable explanation of molecular computat ion I have ever seen.

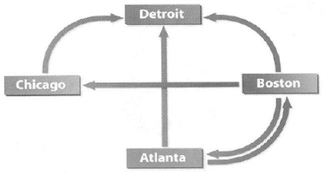
An Example Problem Illustrated

Hamiltonian Path Problem

Consider a map of cities connected by certain nonstop flights (*top right*). For instance, in the example shown here, it is possible to travel directly from Boston to Detroit but not vice versa. The goal is to determine whether a path exists that will commence at the start city (Atlanta), finish at the end city (Detroit) and pass through each of the remaining cities exactly once. In DNA computation, each city is assigned a DNA sequence (*ACTTGCAG* for Atlanta) that can be thought of as a first name (*ACTT*) followed by a last name (*GCAG*). DNA flight numbers can then be defined by concatenating the last name of the city of origin with the first name of the city of destination (*bottom right*).

The complementary DNA city names are the Watson-Crick complements of the DNA city names in which every *C* is replaced by a *G*, every *G* by a *C*, every *A* by a *T*, and every *T* by an *A*. (To simplify the discussion here, details of the 3′ versus 5′ ends of the DNA molecules have been omitted.) For this particular problem, only one Hamiltonian path exists, and it passes through Atlanta, Boston, Chicago and Detroit in that order. In the computation, this path is represented by *GCAGTCGGACTGGGCTATGTCCGA*, a DNA sequence of length 24. Shown at the left is the map with seven cities and 14 nonstop flights used in the actual experiment. —*L.M.A.*



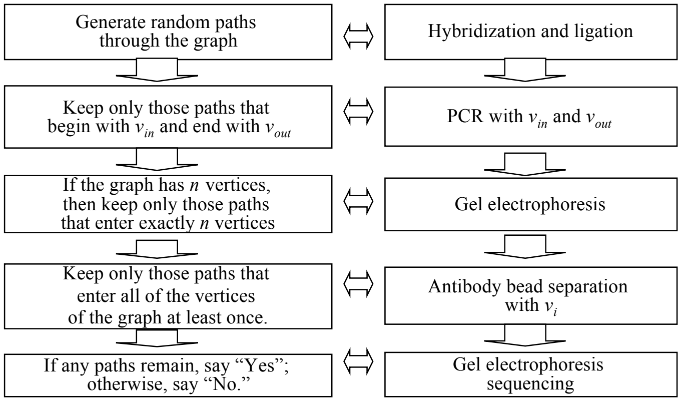


CITY	DNA NAME	COMPLEMENT
ATLANTA	ACTTGCAG	TGAACGTC
BOSTON	TCGGACTG	AGCCTGAC
CHICAGO	GGCTATGT	CCGATACA
DETROIT	CCGAGCAA	GGCTCGTT

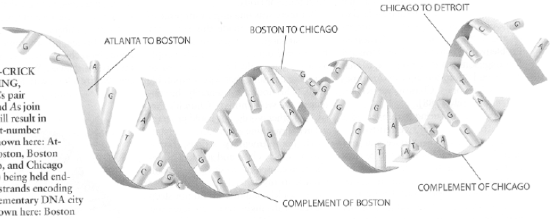
FLIGHT	DNA FLIGHT NUMBER
ATLANTA - BOSTON	GCAGTCGG
ATLANTA - DETROIT	GCAGCCGA
BOSTON - CHICAGO	ACTGGGCT
BOSTON - DETROIT	ACTGCCGA
BOSTON - ATLANTA	ACTGACTT
CHICAGO - DETROIT	ATGTCCGA

[Adleman, *Scientific American* 1998]

Bio-Lab Procedure

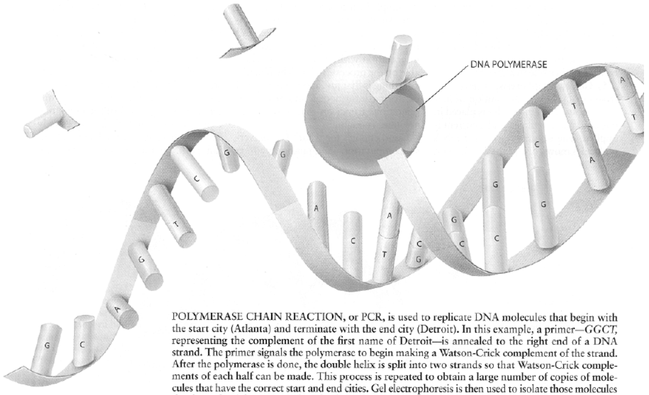


Generate random paths through the graph	⇔ Hybridization and ligation
Keep only those paths that begin with v_{in} and end with v_{out}	⇔ PCR with v_{in} and v_{out}
If the graph has n vertices, then keep only those paths that enter exactly n vertices	⇔ Gel electrophoresis
Keep only those paths that enter all of the vertices of the graph at least once.	⇔ Antibody bead separation with v_i
If any paths remain, say “Yes”; otherwise, say “No.”	⇔ Gel electrophoresis sequencing

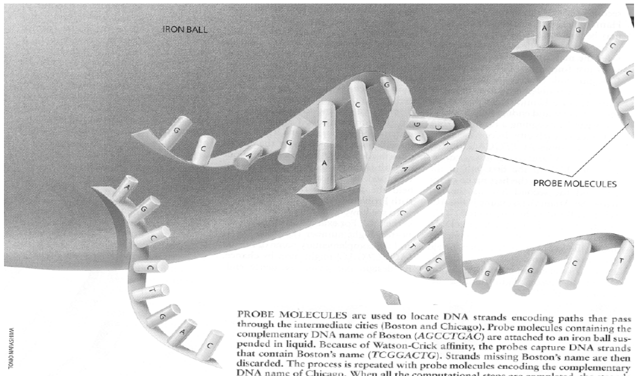


WATSON-CRICK ANNEALING, in which *C*s pair with *G*s and *A*s join with *T*s, will result in DNA flight-number strands (shown here: Atlanta to Boston, Boston to Chicago, and Chicago to Detroit) being held end-to-end by strands encoding the complementary DNA city names (shown here: Boston and Chicago).

LIGASES connect the splinted molecules. Wherever the protein finds two strands of DNA in proximity, it will covalently bond them into a single strand.



POLYMERASE CHAIN REACTION, or PCR, is used to replicate DNA molecules that begin with the start city (Atlanta) and terminate with the end city (Detroit). In this example, a primer—*GGCT,* representing the complement of the first name of Detroit—is annealed to the right end of a DNA strand. The primer signals the polymerase to begin making a Watson-Crick complement of the strand. After the polymerase is done, the double helix is split into two strands so that Watson-Crick complements of each half can be made. This process is repeated to obtain a large number of copies of molecules that have the correct start and end cities. Gel electrophoresis is then used to isolate those molecules that have the right sequence length of 24.



PROBE MOLECULES are used to locate DNA strands encoding paths that pass through the intermediate cities (Boston and Chicago). Probe molecules containing the complementary DNA name of Boston (*AGCCTGAC*) are attached to an iron ball suspended in liquid. Because of Watson-Crick affinity, the probes capture DNA strands that contain Boston's name (*TCGGACTG*). Strands missing Boston's name are then discarded. The process is repeated with probe molecules encoding the complementary DNA name of Chicago. When all the computational steps are completed, the strands left will be those that encode the solution *GCAGTCGGACTGGGCTATGTCCGA*.

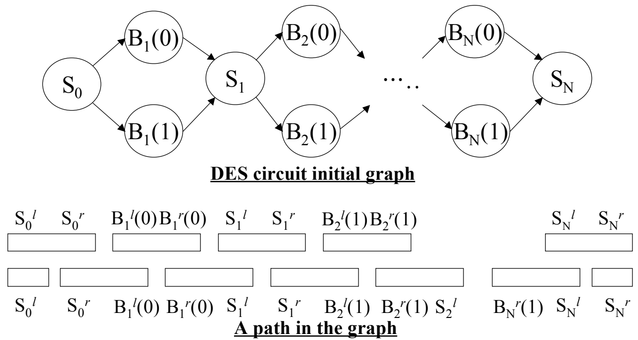
Basic Ideas in DNA Computing

- Exhaustive search
- Parallelism
- Density
- Miniaturization
- Energy efficiency

Recent Applications

- Computational
  - ♦ Cryptography (Boneh et al., 1995)
  - ♦ Chess (Landweber et al., *PNAS* 2000)
  - ♦ 20-var 3-SAT (Adleman, *Science* 2002)
  - ♦ Tic-Tac-Toe (Stojanovic, *Nature Biotech* 2004)
- Biology and Medicine
  - ♦ Genetic switch (Weiss et al., *PNAS* 2002)
  - ♦ Gene control (Benenson et al., *Nature* 2004)
- Nanotechnology
  - ♦ DNA crystals (Winfree & Seeman et al., *Nature* 1998)
  - ♦ Molecular tweezer (Yurke & Turberfield et al., *Nature* 2000)
  - ♦ TX complexes (Reif & Seeman et al, *Nature* 2000)
  - ♦ Tiles (LaBean & Reif, 2003)

Breaking DES


S_0
$B_1(0)$
$B_1(1)$
S_1
$B_2(0)$
$B_2(1)$
……
$B_N(0)$
$B_N(1)$
S_N
S_0^l S_0^r $B_1^l(0)$ $B_1^r(0)$ S_1^l S_1^r $B_2^l(1)$ $B_2^r(1)$ S_N^l S_N^r
S_0^l S_0^r $B_1^l(0)$ $B_1^r(0)$ S_1^l S_1^r $B_2^l(1)$ $B_2^r(1)$ S_2^l $B_N^r(1)$ S_N^l S_N^r

DES circuit initial graph

A path in the graph

[Boneh et al., 1995]

Self-Assembly of DNA Crystals

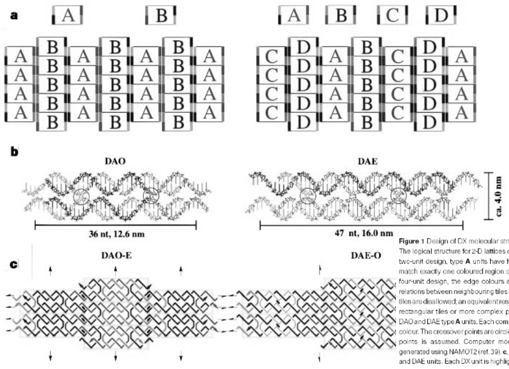


Figure 1 Design of DX molecular structure and arrangement into 2-D lattices. **a**, The logical structure for 2-D lattices consisting of two units and four units. In the two-unit design, type **A** units have four coloured edge regions, each of which match exactly one coloured region of the adjacent type **B** units. Similarly, in the four-unit design, the edge colours are chosen uniquely to define the desired relations between neighbouring tiles. Note that rotations and reflections of Wang tiles are disallowed; an equivalent restriction could also be obtained by using non-rectangular tiles or more complex patterns of colours. **b**, Model structures for DAO and DAE type **A** units. Each component oligonucleotide is shown in a unique colour. The crossover points are circled. Complete base stacking at the crossover points is assumed. Computer models showing every nucleotide (nt) were generated using NAMOT2 (ref. 39). **c**, The lattice topologies produced by the DAO and DAE units. Each DX unit is highlighted by a grey rectangle. A unique colour is

[Winfree et al., *Nature* 1998]

RNA Solution to a Chess Problem

$((\neg h \wedge \neg f) \vee \neg a) \wedge (\neg g \wedge \neg i) \vee \neg b) \wedge ((\neg d \wedge \neg h) \vee \neg c) \wedge ((\neg c \wedge \neg i) \vee \neg d) \wedge (\neg a \wedge \neg g) \vee \neg f) \wedge$
$(\neg b \wedge \neg f) \vee \neg g) \wedge ((\neg a \wedge \neg c) \vee \neg h) \wedge (\neg d \neg b) \vee \neg i)$

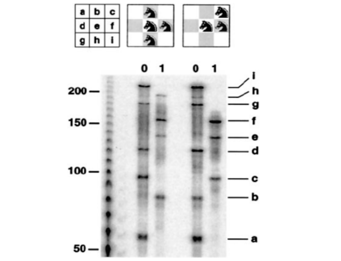
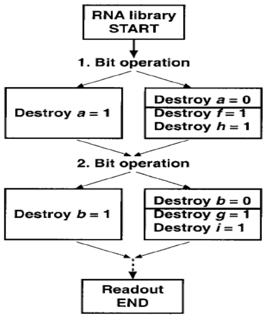


Fig. 3. Multiplex colony PCR readout. Colony PCR (10) allowed rapid screening and recovery of individual library strands. The boards represented by two clones are shown above their respective readouts on a 6% polyacrylamide gel. These clones represent the strings 010 011 010 and 001 011 000. Left lane, 10-bp ladder.



RNA library START

1. Bit operation

Destroy $a = 1$

Destroy $a = 0$
Destroy $f = 1$
Destroy $h = 1$

2. Bit operation

Destroy $b = 1$

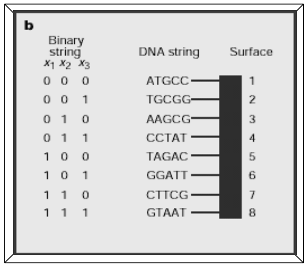
Destroy $b = 0$
Destroy $g = 1$
Destroy $i = 1$

Readout END

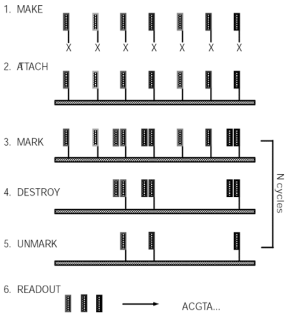
[Faulhammer et al., *PNAS* 2000]

Solving a 3-SAT Problem on Chip

[redacted]

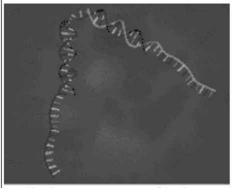


Binary string x_1	x_2	x_3	DNA string	Surface
0	0	0	ATGCC	1
0	0	1	TGCGG	2
0	1	0	AAGCG	3
0	1	1	CCTAT	4
1	0	0	TAGAC	5
1	0	1	GGATT	6
1	1	0	CTTCG	7
1	1	1	GTAAT	8

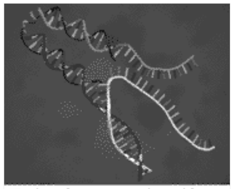


[Liu et al., *Nature 2000*]

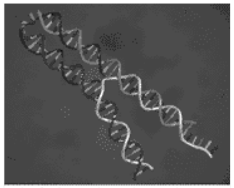
Making a Molecular Tweezer



"Bell Labs scientists assemble their tweezer motor from DNA with two double-stranded arms connected by a singled-stranded hinge, and two single-stranded "handles" at the ends of the arms."



"To close the tweezers, they add a special "fuel" strand of DNA"



"The fuel strand attaches to the handles and draws the two arms of the tweezers together."

* Motion affected via white fuel strand via strand displacement.

[Yurke et al., *Nature 2000*]

Solving a 20-var 3-CNF Problem

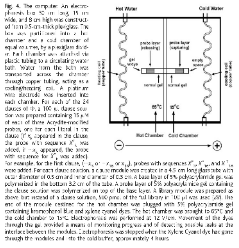
A

Φ =($\sim x_3$ *or* $\sim x_{16}$ *or* x_{18}) *and* (x_5 *or* x_{12} *or* $\sim x_9$) *and* ($\sim x_{13}$ *or* $\sim x_2$ *or* x_{20}) *and* (x_{12} *or* $\sim x_9$ *or* $\sim x_5$) *and* (x_{19} *or* $\sim x_4$ *or* x_6) *and* (x_9 *or* x_{12} *or* $\sim x_5$) *and* ($\sim x_1$ *or* x_4 *or* $\sim x_{11}$) *and* (x_{13} *or* $\sim x_2$ *or* $\sim x_{19}$) *and* (x_5 *or* x_{17} *or* x_9) *and* (x_{15} *or* x_9 *or* $\sim x_{17}$) *and* ($\sim x_5$ *or* $\sim x_9$ *or* $\sim x_{12}$) *and* (x_6 *or* x_{11} *or* x_4) *and* ($\sim x_{15}$ *or* $\sim x_{17}$ *or* x_7) *and* ($\sim x_6$ *or* x_{19} *or* x_{13}) *and* ($\sim x_{12}$ *or* $\sim x_9$ *or* x_5) *and* (x_{12} *or* x_1 *or* x_{14}) *and* (x_{20} *or* x_3 *or* x_2) *and* (x_{10} *or* $\sim x_7$ *or* $\sim x_8$) *and* ($\sim x_5$ *or* x_9 *or* $\sim x_{12}$) *and* (x_{18} *or* $\sim x_{20}$ *or* x_3) *and* ($\sim x_{10}$ *or* $\sim x_{18}$ *or* $\sim x_{16}$) *and* (x_1 *or* $\sim x_{11}$ *or* $\sim x_{14}$) *and* (x_8 *or* $\sim x_7$ *or* $\sim x_{15}$) *and* ($\sim x_8$ *or* x_{16} *or* $\sim x_{10}$)

B

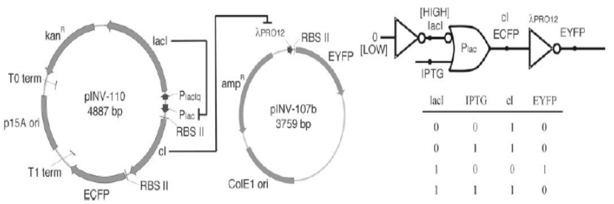
x_1=F, x_2=T, x_3=F, x_4=F, x_5=F, x_6=F, x_7=T, x_8=T, x_9=F, x_{10}=T, x_{11}=T, x_{12}=T, x_{13}=F, x_{14}=F, x_{15}=T, x_{16}=T, x_{17}=T, x_{18}=F, x_{19}=F, x_{20}=F

Fig. 1. The computational problem. **(A)** 20-variable 3-CNF Boolean formula Φ. The symbol "$\sim$" indicates "not." **(B)** The unique truth assignment satisfying Φ.



[Braich et al., *Science* 2002]

Directed Evolution of a Genetic Circuit

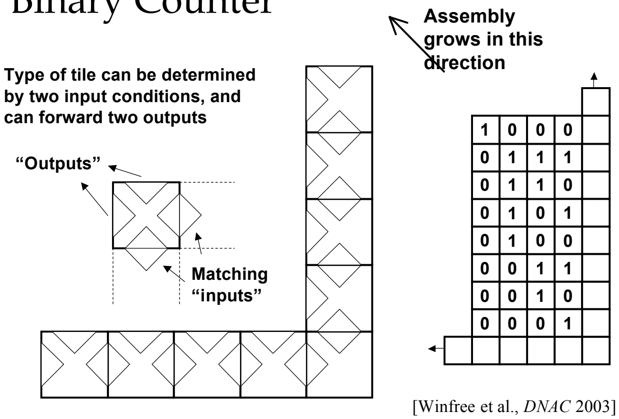


lacI	IPTG	cI	EYFP
0	0	1	0
0	1	1	0
1	0	0	1
1	1	1	0

Fig. 1. The plasmid diagram shows the implementation of the present circuit. Plasmid pINV-110 constitutively expresses the LacI repressor, which inhibits transcription from the P_{lac} promoter in the absence of IPTG. The expression of the CI repressor and ECFP fluorescent marker is controlled by P_{lac}, which is inducible by externally added IPTG. Repressor CI acts on λP_{RO12} on pINV-107b to repress the transcription of the EYFP gene, the output fluorescence indicator. The two plasmids contain different origins of replication as well as different antibiotic resistance genes, which allow them to be maintained stably within a single cell. The logic diagram (*Upper Right*) represents the logical representation of the same biochemical circuit. The P_{lac} promoter comprises an IMPLIES logic gate with respect to the two inputs LacI and IPTG and the output CI, whose truth table is shown below the diagram. The output of the IMPLIES gate, CI, is the input to the inverter based on the λP_{RO12} promoter, ultimately controlling expression of the fluorescent output, EYFP. Note that the levels of EYFP output are the inverse of the input CI in the truth table. In this study, we targeted mutations to the CI protein.

[Weiss et al., *PNAS* 2002]

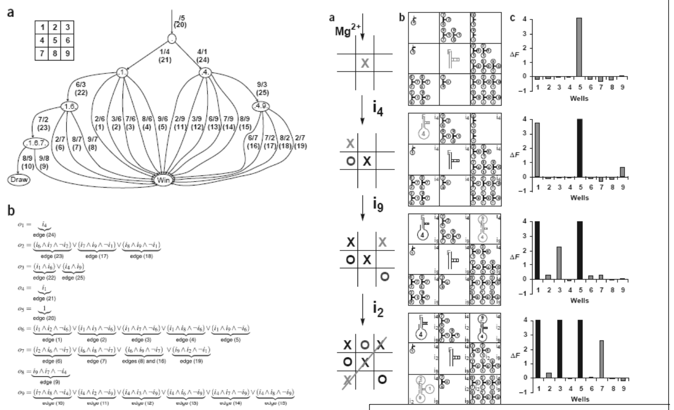
Binary Counter


Assembly grows in this direction
Type of tile can be determined by two input conditions, and can forward two outputs
"Outputs"
Matching "inputs"

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

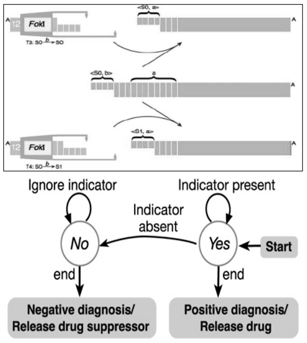
[Winfree et al., *DNAC* 2003]

Playing a Tic-Tac-Toe Game



Stojanovic et al., *Nature Biotech.* 2004]

DNA as Smart Drugs


FokI
Ignore indicator
Indicator present
Indicator absent
No
Yes
Start
end
end
Negative diagnosis/ Release drug suppressor
Positive diagnosis/ Release drug

[Benenson *et al., Nature* 2001 & *Nature,* 2004]

PPAP2B↓ & GSTP1↓ & PIM1↑ & HEPSIN↑ → Administer GTTGGTATTGCACAT

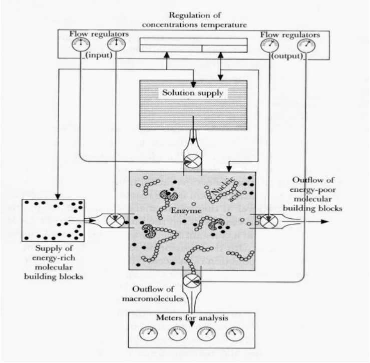
Difficulties in Current Molecular Computing Paradigms

- Scalability
  - ♦ For big problems, exhaustive search is not effective.
- Reliability
  - ♦ DNA reaction is error-prone.
- Fault tolerance
  - ♦ What if a single molecule malfunctions?
- Design
  - ♦ How to design the decision (or diagnosis) rules?

In Vitro Evolution
(without Computation)

Eigen's *Theory of Molecular Evoluti on (1979)*

Manfred Eigen
(1927 -)


Regulation of
concentrations temperature
Flow regulators
(input)
Flow regulators
(output)
Solution supply
Nucleic acid
Enzyme
Supply of energy-rich molecular building blocks
Outflow of energy-poor molecular building blocks
Outflow of macromolecules
Meters for analysis

In Vitro Evolution Experiments

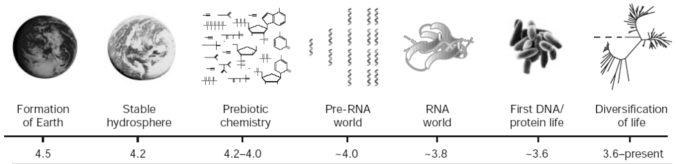


Figure 1 Timeline of events pertaining to the early history of life on Earth, with approximate dates in billions of years before the present.

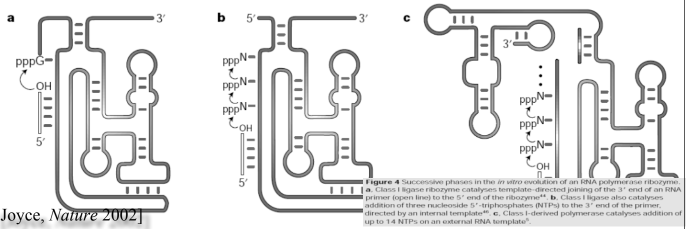
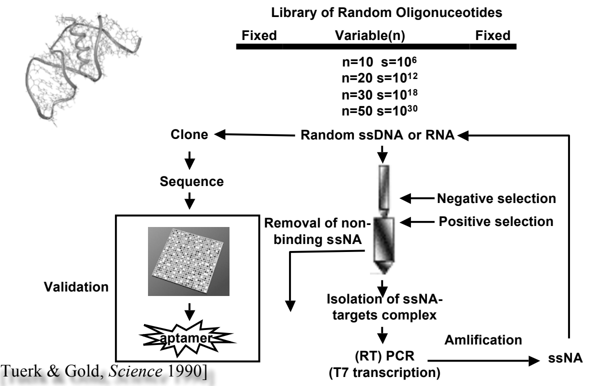


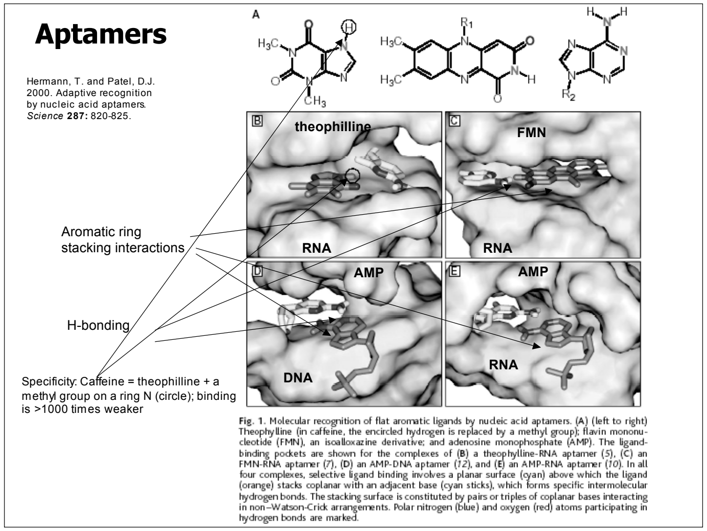
Figure 4 Successive phases in the *in vitro* evolution of an RNA polymerase ribozyme. **a**, Class I ligase ribozyme catalyses template-directed joining of the 3′ end of an RNA primer (open line) to the 5′ end of the ribozyme[44]. **b**, Class I ligase also catalyses addition of three nucleoside 5′-triphosphates (NTPs) to the 3′ end of the primer, directed by an internal template[46]. **c**, Class I-derived polymerase catalyses addition of up to 14 NTPs on an external RNA template[5].

[Joyce, *Nature* 2002]

SELEX (Systematic Evolution of Ligands by EXponential Enrichment)



[Tuerk & Gold, *Science* 1990]



Aptamers

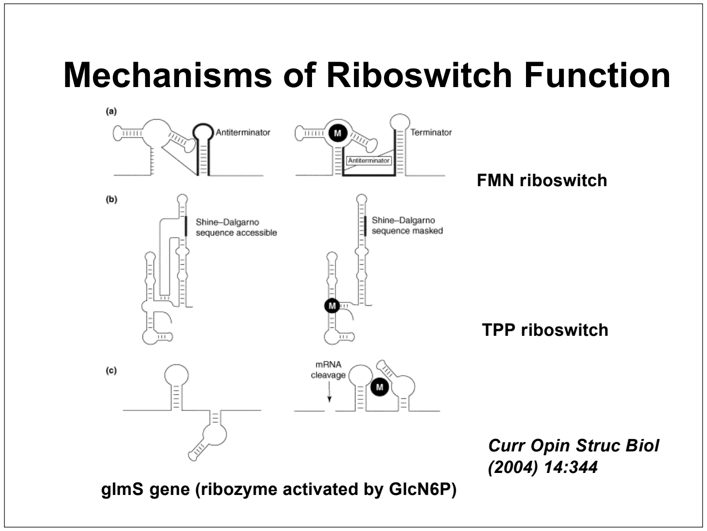
Hermann, T. and Patel, D.J. 2000. Adaptive recognition by nucleic acid aptamers. *Science* **287:** 820-825.

Aromatic ring stacking interactions

H-bonding

Specificity: Caffeine = theophilline + a methyl group on a ring N (circle); binding is >1000 times weaker

Fig. 1. Molecular recognition of flat aromatic ligands by nucleic acid aptamers. (**A**) (left to right) Theophylline (in caffeine, the encircled hydrogen is replaced by a methyl group); flavin mononucleotide (FMN), an isoalloxazine derivative; and adenosine monophosphate (AMP). The ligand-binding pockets are shown for the complexes of (**B**) a theophylline-RNA aptamer (*5*), (**C**) an FMN-RNA aptamer (*7*), (**D**) an AMP-DNA aptamer (*12*), and (**E**) an AMP-RNA aptamer (*10*). In all four complexes, selective ligand binding involves a planar surface (cyan) above which the ligand (orange) stacks coplanar with an adjacent base (cyan sticks), which forms specific intermolecular hydrogen bonds. The stacking surface is constituted by pairs or triples of coplanar bases interacting in non–Watson-Crick arrangements. Polar nitrogen (blue) and oxygen (red) atoms participating in hydrogen bonds are marked.



Mechanisms of Riboswitch Function

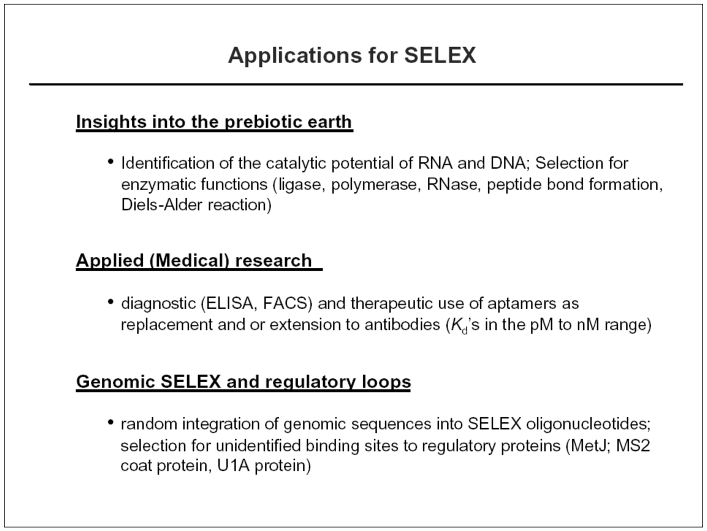
(a)
Antiterminator
M
Terminator
Antiterminator
(b)
Shine–Dalgarno sequence accessible
Shine–Dalgarno sequence masked
M
(c)
mRNA cleavage
M

FMN riboswitch

TPP riboswitch

glmS gene (ribozyme activated by GlcN6P)

Curr Opin Struc Biol (2004) 14:344



Applications for SELEX

Insights into the prebiotic earth

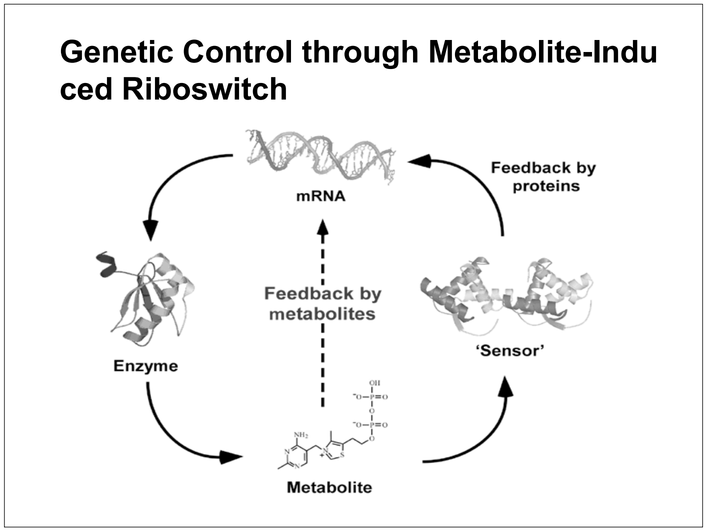
- Identification of the catalytic potential of RNA and DNA; Selection for enzymatic functions (ligase, polymerase, RNase, peptide bond formation, Diels-Alder reaction)

Applied (Medical) research

- diagnostic (ELISA, FACS) and therapeutic use of aptamers as replacement and or extension to antibodies (K_d's in the pM to nM range)

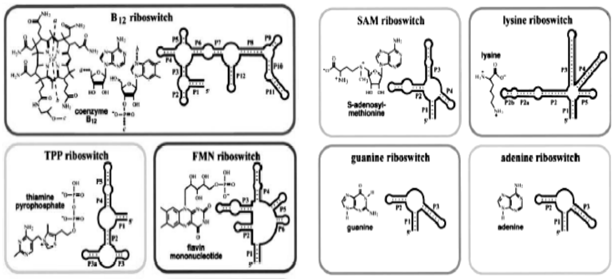
Genomic SELEX and regulatory loops

- random integration of genomic sequences into SELEX oligonucleotides; selection for unidentified binding sites to regulatory proteins (MetJ; MS2 coat protein, U1A protein)

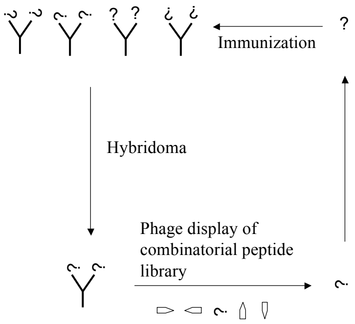


Genetic Control through Metabolite-Indu ced Riboswitch

Metabolite-Induced Riboswitch



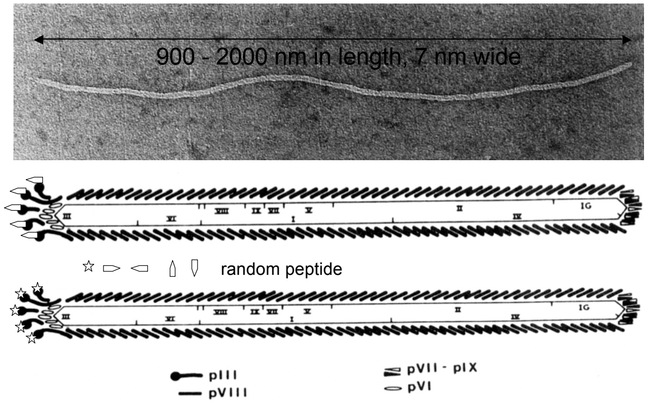
Peptide Library



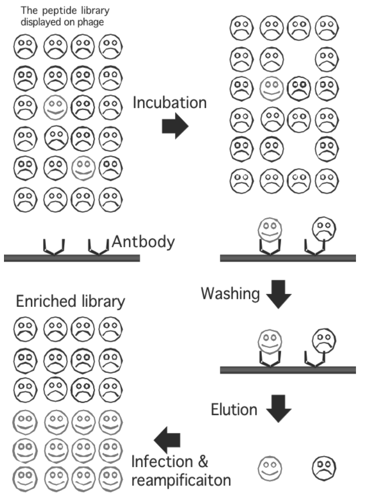
CD99

MARGAALALLLFGLLGVLV
AAPDGGFDLSDALPDNEN
KKPTAIPKKPSAGDDFDLG
DAVVDGENDDPRPPNPPK
PMPNPNPNHPSSSGSFSD
ADLADGVSGGEGKGGSDG
GGSHRKEGEEADAPGVIP
GIVGAVVVAVAGAISSFIAY
QKKKLCFKENDG

Phage Display of Peptide Library

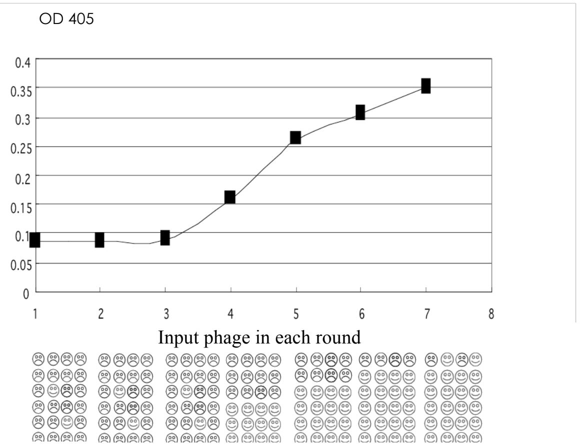


Bio-Panning

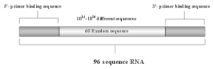


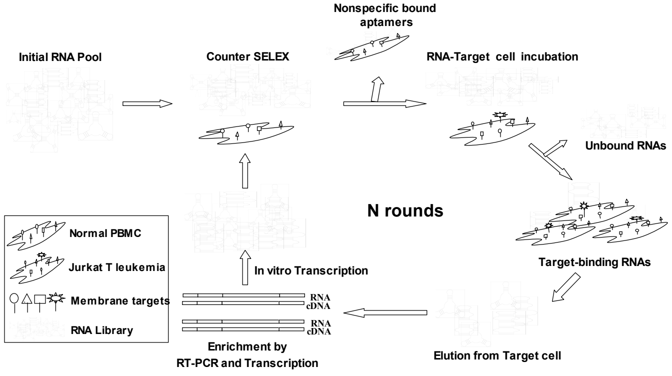
The peptide library displayed on phage
Incubation
Antbody
Washing
Enriched library
Elution
Infection & reampificaiton

Enriching the binding clones through bio-panning



Cell SELEX Procedure

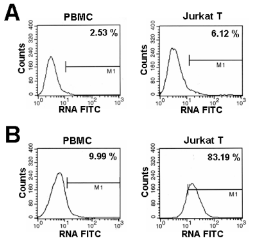

5'- primer binding sequence
3'- primer binding sequence
10^{14}–10^{16} different sequences
60 Random sequence
96 sequence RNA



Cancer-Specific Aptamers

Sequence (Occurrence)

```
I-1     (3)   5′ GAACGCGAAUUUGACACUGAUCAGGCGUGUUGAUCGCCAACUGUAAUGUCCUCUGUCUCG 3′
I-2     (1)   5′ ---------------------------------------------------C-------- 3′

II-1    (3)   5′ GUCGAUGCUUAAGUGAUUUAUCCUCAAUACAACGAGUGGAUCAAUGUUUGUACGCGUGG 3′
II-2    (1)   5′ ---------------------------G------------------------------- 3′

III     (2)   5′ GGUUGAGGACUAUGAAUUAUUAUCCCAGACUGGUUCCCGAACACUCACAUGCUGCGUGG 3′

IV-1    (1)   5′ GAACGAGCAUUAAGUCAUAGCAGCAUUUUUAGAAGUGGGAAUUGUUGUCGCUGUUUGUGUUG 3′
IV-2    (1)   5′ ---------------- -------------------------------------------C- 3′

V       (2)   5′ CGCCAUUGCUCAAGAGUAGCAUUCGCUGUGACUAAUUCCCAUGCUGGUUGGUGUCGGUGG 3′

VI      (1)   5′ GGCGAUGAAUAGUUUUCCACGAGUGUAAUGAGUUUUGTGAAUGCUCAGUACCUGUCCCG 3′

VII     (1)   5′ UAGUUGCGCAAACGUAGUAAGACAUGAUUGCUGUUGGUAGUAUUAUUUGUGUGGCUGGUG 3′

VIII    (1)   5′ AUCGACAAUUAAACAUUUUCUCUGGCAGUGAUUCCAUAGCGUGUACCCGUUGGCUGGUCG 3′

IX      (1)   5′ AACCCGUGGAACGUAGAAGUACAAUUGACUAGUAGUCCUUUUGACCCGUGTACCGCGCG 3′

X       (1)   5′ AGCCGUCUGACAUUUACCAUAUAUUAGCGUAAAACGACCUUUACACUGUGUAUCCGGGUGG 3′

XI      (1)   5′ GAUCAAUGGCACAAUACAACAGUUCGCUGAAUAUAGCUGUCGUCCUUUGGUAUCCGUGUGG 3′
```



- ❖ A specific disease could be targeted without prior knowledge of any molecular changes associated with the disease state
- ❖ As capture probes to identify cancer-specific molecular markers
- ❖ As specific cell sorter & identifier

Kauffman's *Theory of Collectively Autocatalytic Sets (2001)*

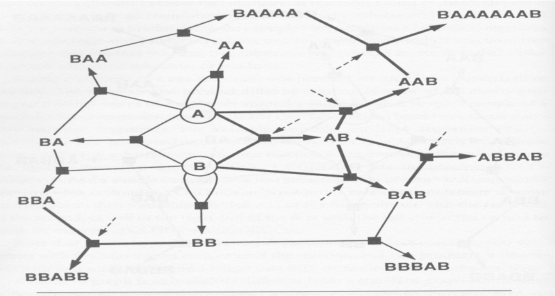


FIGURE 2.7b Molecules catalyzing reactions. In Figure 2.7a, all reactions were assumed to be spontaneous. What happens when we add catalysts to speed some of the reactions? Here the reaction squares indicated by dashed line arrows are catalyzed, and the heavy darker lines connect substrates and products whose reactions are catalyzed. The result is a pattern of heavy lines indicating the catalyzed subgraph of the reaction graph.

In Vitro (Molecular) Evolution: Synonyms

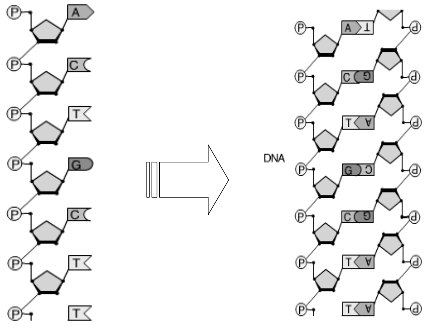
- In vitro selection
- Directed evolution
- In vitro evolution
- Molecular evolution
- SELEX
- Bio-panning
- “In vitro molecular evolution”

Molecular Evolutionary Computation (MEC) in Vitro: Theory

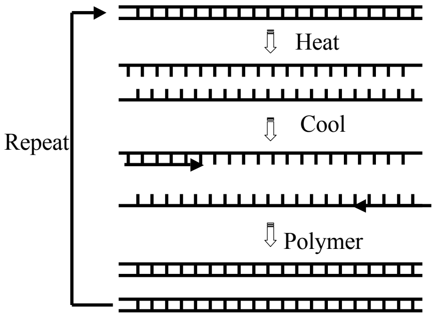
Motivation: In Vitro Evolution as EC Technology

- Each DNA molecule represents an individual at nanoscale
- A huge population of up to Avogadro number (6 x 10^{23}) molecules
- Molecular recognition by chemistry
- Exponential self-replication by PCR
- Massively parallel variation-selection operators
- Ultra-low energy consumption
- Evolvable “wet” “molecular” hardware

Molecular Recognition



Self-Replication



In Vitro Evolutionary Computation

- Problems in Existing DNA Computing Paradigms: Revisited
  - Scalability
    - For big problems, exhaustive search does not work.
  - Reliability
    - DNA reaction is error-prone.
  - Fault tolerance
    - What if a single molecule malfunctions?
  - Design
    - How to design the decision (or diagnosis) rules?
- In Vitro Evolution + Molecular Computation

  = Molecular Evolutionary Computation (MEC)

  = Bayesian Evolution + Probabilistic Library Model

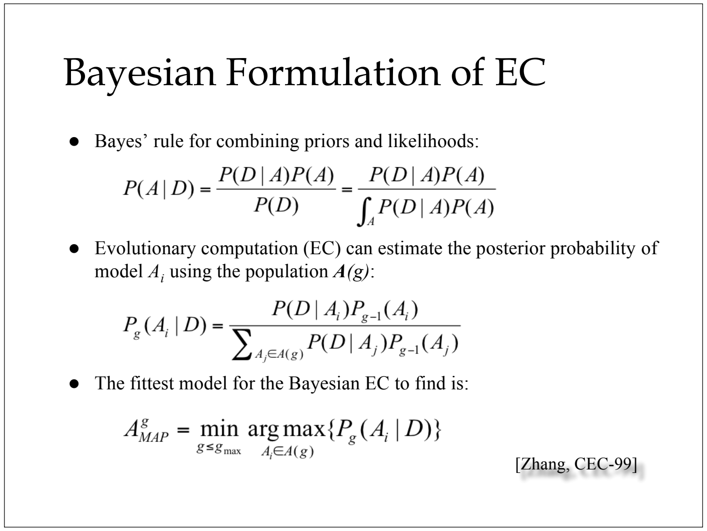
Why Try Molecular EC?

- 6.022×10^{23} molecules / mole
- Massively Parallel Search
  - Desktop: 10^9 operations / sec
  - Supercomputer: 10^{12} operations / sec
  - 1 μmol of DNA: 10^{26} reactions
- Favorable Energetics: Gibbs Free Energy

  [redacted]
  - 1 *J* for 2×10^{19} operations
- Storage Capacity: 1 bit per cubic nanometer
- The fastest supercomputer vs. DNA computer
  - 10^6 op/sec vs. 10^{14} op/sec
  - 10^9 op/J vs. 10^{19} op/J (in ligation step)
  - 1bit per 10^{12} nm^3 vs. 1 bit per 1 nm^3 (video tape vs. molecules)

The Theory of Bayesian Evolution

- Evolution as a Bayesian inference process
- Evolutionary computation (EC) is viewed as an ite rative process of *generating the individuals of ever higher posterior probabilities* from the priors and t he observed data.



Bayesian Formulation of EC

- Bayes' rule for combining priors and likelihoods:

$$P(A\,|\,D) = \frac{P(D\,|\,A)P(A)}{P(D)} = \frac{P(D\,|\,A)P(A)}{\int_A P(D\,|\,A)P(A)}$$

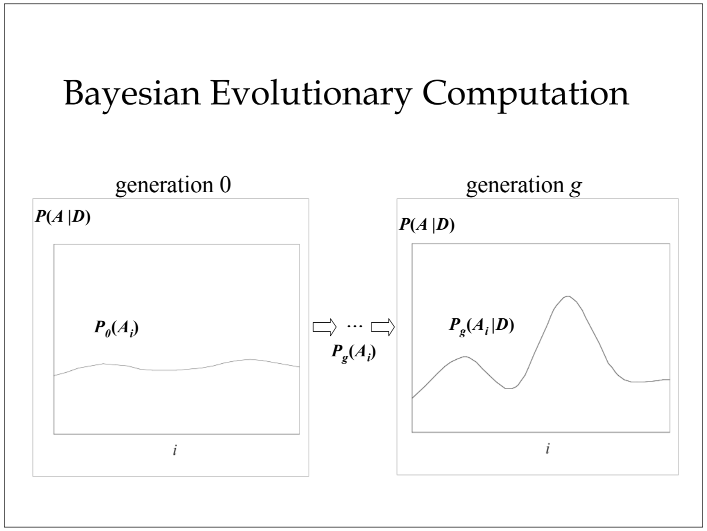
- Evolutionary computation (EC) can estimate the posterior probability of model A_i using the population ***A(g)***:

$$P_g(A_i\,|\,D) = \frac{P(D\,|\,A_i)P_{g-1}(A_i)}{\sum_{A_j \in A(g)} P(D\,|\,A_j)P_{g-1}(A_j)}$$

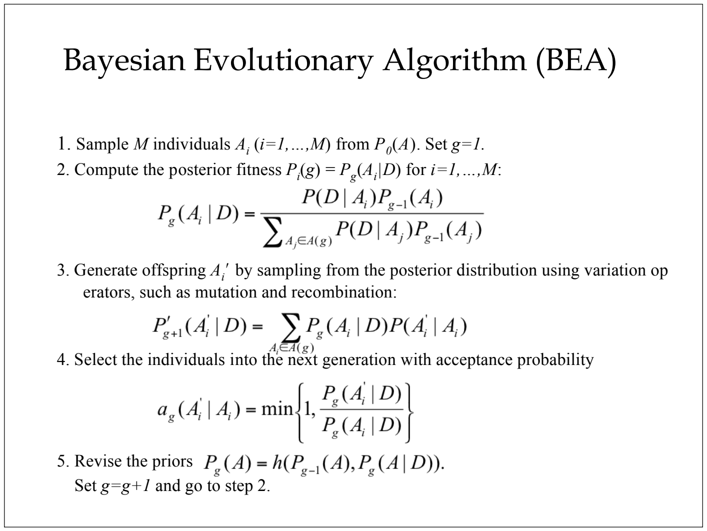
- The fittest model for the Bayesian EC to find is:

$$A^g_{MAP} = \min_{g \le g_{\max}} \arg\max_{A_i \in A(g)} \{P_g(A_i\,|\,D)\}$$

[Zhang, CEC-99]



Bayesian Evolutionary Computation



Bayesian Evolutionary Algorithm (BEA)

1. Sample *M* individuals A_i ($i=1,\ldots,M$) from $P_0(A)$. Set $g=1$.
2. Compute the posterior fitness $P_i(g) = P_g(A_i|D)$ for $i=1,\ldots,M$:

$$P_g(A_i\,|\,D) = \frac{P(D\,|\,A_i)P_{g-1}(A_i)}{\sum_{A_j \in A(g)} P(D\,|\,A_j)P_{g-1}(A_j)}$$

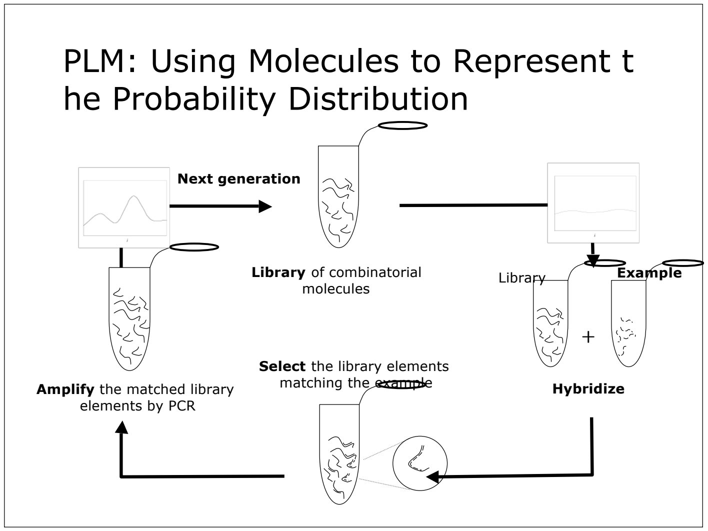
3. Generate offspring A_i' by sampling from the posterior distribution using variation op erators, such as mutation and recombination:

$$P'_{g+1}(A_i'\,|\,D) = \sum_{A_i \in A(g)} P_g(A_i\,|\,D)P(A_i'\,|\,A_i)$$

4. Select the individuals into the next generation with acceptance probability

$$a_g(A_i'\,|\,A_i) = \min\left\{1, \frac{P_g(A_i'\,|\,D)}{P_g(A_i\,|\,D)}\right\}$$

5. Revise the priors $P_g(A) = h(P_{g-1}(A), P_g(A\,|\,D))$.
   Set $g=g+1$ and go to step 2.



PLM: Using Molecules to Represent t he Probability Distribution

Next generation
Library of combinatorial molecules
Library
Example
+
Hybridize
Select the library elements matching the example
Amplify the matched library elements by PCR

The Probabilistic Library Model (PLM)

- A library of DNA molecules represents the empirical distribution of data variables.
- Each library element consists of *n* variables, $X_1, \ldots, X_n$.
- A big number of molecules are maintained in the library.
  - ♦ $L = \{ x_i \mid i = 1, \ldots, N \}$
  - ♦ *N*: typically 10^{15} with 10 nM
- Duplications of elements are allowed. And the number of duplications is proportional to the strength of the element.
- The library is so maintained that it represents the joint probability distribution of the data variables.
  - ♦ $P(X) = P(X_1, \ldots, X_n)$

[Zhang, DNAC-2004]

The PLM (cont'd)

- The probability of variable X_k having value x_k is computed chemically by putting in the library the complementary sequence $-x_k$ of the query sequence x_k and extracting the hybridized sequences followed by normalization.
  - ♦ $P(X_k=x_k) \sim c(x_k)/|L|$
- Conditional probabilities are computed by the relative frequencies of the molecules.
  - ♦ $P(X_i|X_k) = P(X_i,X_k) / P(X_k)$
  - ♦ Here $P(X_i=x_i,X_k=x_k) \sim c(x_{i,}x_k)/|L|$ and $P(X_k=x_k) \sim c(x_k)/|L|$
- The library as an ensemble
- Probabilistic computation
- Massively parallel computation of probabilities

The PLM as a Pattern Classifier

- Assume *L* contains sequence patterns x_i with known labels y_i (training set)
  - ♦ $L = \{ (x_i, y_i) \mid i = 1, \ldots, N \}$
  - ♦ $x_i = \{A,T,G,C\}^n$: observable input, e.g. DNA sequence
  - ♦ $y_i = \{0, 1\}$, observable output, e.g. cancer or normal

[redacted]

- Given a query sequence x_q
  - ♦ Put $-x_q$ into the test tube   -x: complementary to x

[redacted]

- Find the correct class y_q for x_q (classification)
  - ♦ $y_q = 1$: cancer
  - ♦ $y_q = 0$: normal

Classification Decision: Probabilistic Formulation

- $P(X)$: Probability of observing protein sequence X
- $P(X,Y)$: Probability of sequence X being in class Y
- $P(X,Y,Z)$: Probability of sequence X being in class Y with some parameter Z
- $P(Y|X)$: Conditional probability of class Y given X

[redacted]

Classification Learning: In Vitro Evolution

1. Start with library L of random samples (molecules)
2. Given a training sample $s = (x, y)$
3. Classify s using L

- ♦ Extract x → $N(x) := P(x)$
- ♦ Extract Y → $N(x,Y) := P(x,Y)$
- ♦ $y^* = \text{argmax}_Y N(x,Y)$

4. Update L

- ♦ If $y^*=y$, $P(y^*|x)$ ← $d\ N(y^*|x)$ with $d>1$
- ♦ Otherwise, $P(y^*|x)$ ← $d\ N(y^*|x)$ with $d<1$

[redacted]

[redacted]

The Learning Rule Leads to Bayesian Update

[redacted]

[redacted]

Update of $N(y^*|x)$ leads to update of the posterior probability distribution $P(z|y,x)$, resulting in a Bayesian learning rule for classification learning with DNA computing

[redacted]

[Zhang, DNA10]

PLM vs. Probabilistic Model-Building GAs (or EDAs)

- Some recent genetic and evolutionary algorithms build explicit probabilistic models for the population.
- These distribution-estimation algorithms (EDAs) generate offspring by sampling from the probabilistic model rather than using crossover and mutation.
- Like EDAs, the probabilistic library model (PLM) generates the offspring by sampling from a probability distribution.
- Unlike in EDAs, in PLM no extra probabilistic model is built. The PLM itself represents a probability distribution.
- The use of a huge number of molecules (10^{15} or more) enables the test tube to represent the empirical probability distribution.

Molecular Programming (MP): In Vitro Evolution of Genetic Programs

Molecular Programming (MP):
Evolving Genetic Programs in a Test Tube

- Theory
  - Bayesian evolution [Zhang, CEC-99; Zhang, Handbook-2003]
- Model
  - Probabilistic library model [Zhang, DNA10 & DNA11]
- Algorithm
  - Molecular algorithms [Zhang, GP-98]
- Representation
  - Decision lists [Zhang, GECCO-2005]
- Operators
  - Molecular operators for variation and selection [Zhang, GECCO-2005]

Molecular Programming of the PLM

1. Let the library *L* represent the current distribution *P*(*X,Y*).
2. Get a training example (**x**,*y*).
3. Classify **x** using *L* as follows
   - 3.1 Extract all molecules matching **x** into *M*.
   - 3.2 From *M* separate the molecules into classes:
     - Extract the molecules with label *Y*=0 into M^0
     - Extract the molecules with label *Y*=1 into M^1
   - 3.3 Compute $y^* = \mathrm{argmax}_{Y \in \{0,1\}} | M^Y | / |M|$
4. Update *L*
   - If y^*=y, then $L_n \leftarrow L_{n-1} + \{\Delta c(\mathbf{u}, v)\}$ for **u**=**x** and v=y for $(\mathbf{u}, v) \in L_{n-1}$,
   - If $y^* \neq$ y, then $L_n \leftarrow L_{n-1} - \{\Delta c(\mathbf{u}, v)\}$ for **u**=**x** and v ≠ y for $(\mathbf{u}, v) \in L_{n-1}$

5.Goto step 2 if not terminated.

[Zhang, GECCO-2005]

Step 1: Probability Distribution in the Library

$$D = \{(\mathbf{x}_i, y_i)\}_{i=1}^{K} \qquad \mathbf{x}_i = (x_{i_1}, x_{i_2}, \cdots, x_{i_n}) \in \{0,1\}^n$$

$$y_i \in \{0,1\}$$

$$P(X,Y) \approx \frac{1}{|L|} \sum_{i=1}^{|L|} f_i^{(n)}(X_1, X_2, \ldots, X_n, Y)$$

Step 2: Presentation of an Example (or Query)

$$P(\mathbf{x}_i, y_i \mid \mathbf{x}_q, y_q) = \frac{\exp(-\Delta G(\mathbf{x}_i, y_i \mid \mathbf{x}_q, y_q))}{\sum_j \exp(-\Delta G(\mathbf{x}_i, y_i \mid \mathbf{x}_q, y_q))}$$

Step 3: Classify the Example (Decision Making)

$$\begin{aligned} y^* &= \arg\max_{Y \in \{0,1\}} P(Y \mid \mathbf{x}) \\ &= \arg\max_{Y \in \{0,1\}} \frac{P(Y, \mathbf{x})}{P(\mathbf{x})} \end{aligned} \qquad c(\mathbf{x}) / |L| = |M| / |L| \approx P(\mathbf{x})$$

$$\begin{aligned} y^* &= \arg\max_{Y \in \{0,1\}} c(Y \mid \mathbf{x}) / |M| \\ &= \arg\max_{Y \in \{0,1\}} c(Y \mid \mathbf{x}) \\ &\approx \arg\max_{Y \in \{0,1\}} P(Y \mid \mathbf{x}) \end{aligned} \qquad c(Y \mid \mathbf{x}) / |M| = |M^Y| / |M| \approx P(Y \mid \mathbf{x})$$

Step 4: Update the Library (Learning)

$L \leftarrow L + \{(\mathbf{u}, v)\}$ $L \leftarrow L - \{(\mathbf{u}, v)\}$

$$P_n(X, Y \mid \mathbf{x}, y) = (1+\delta) P_{n-1}(X, Y \mid \mathbf{x}, y)$$

$$\delta = \frac{P(\mathbf{x}, y \mid X, Y) - P(\mathbf{x}, y)}{P(\mathbf{x}, y)}$$

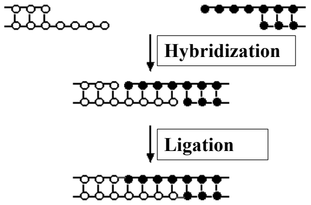
$$\delta = \frac{\Delta c(\mathbf{x}, y)}{c_{n-1}(\mathbf{x}, y)}$$

Molecular Operators

- Variation
  - Ligation
  - Restriction
  - Mutation (PCR)
- Selection
  - Gel electrophoresis
  - Affinity separation (beads)
  - Capillary electrophoresis
- Amplification
  - Polymerase chain reaction (PCR)
  - Rolling circle amplification (RCA)

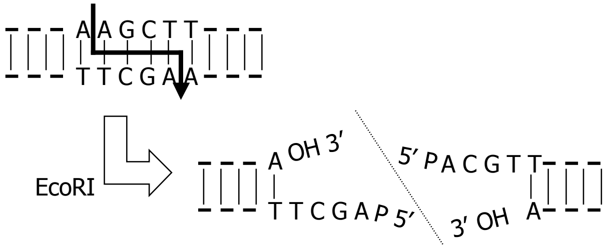
Variation: Hybridization & Ligation

- Hybridization
  - base-pairing between two c omplementary single-strand molecules to form a double stranded DNA molecule
- Ligation
  - Joining DNA molecules tog ether
- Usually used for candidate solution generation.



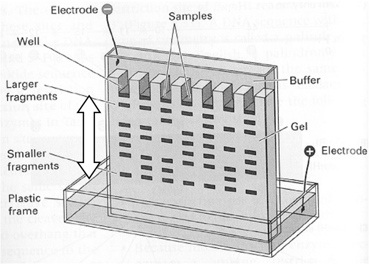
Variation: Restriction

- Cut the specific DNA site.
- Solution detection or filtering step

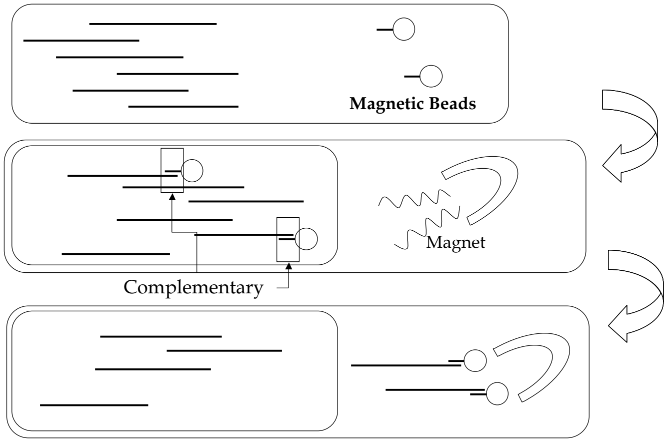


Selection: Gel Electrophoresis

- Detection desired solutions.
- Separate solution molecules by length



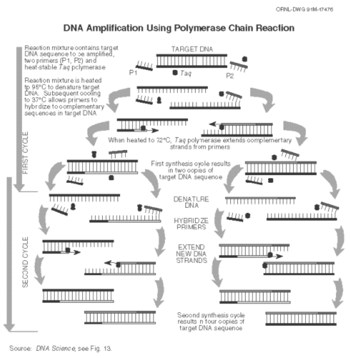
Selection: Bead Separation



Detect & separate the specific DNA

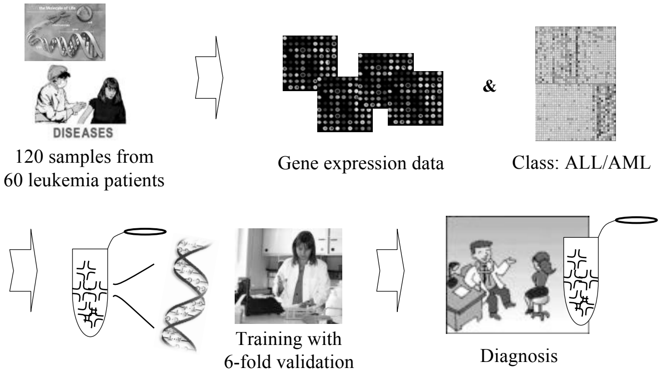
Amplification: PCR

- Polymerase chain reaction
- Amplifies (produces identical copies of) selected dsDNA molecules.
- Make 2^n copies (n : number of iteration)
- Used to filter solutions or detection.


ORNL-DWG 91M-17476
DNA Amplification Using Polymerase Chain Reaction
Reaction mixture contains target DNA sequence to be amplified, two primers (P1, P2) and heat-stable Taq polymerase
TARGET DNA
P1
Taq
P2
Reaction mixture is heated to 95°C to denature target DNA. Subsequent cooling to 37°C allows primers to hybridize to complementary sequences in target DNA
When heated to 72°C, Taq polymerase extends complementary strands from primers
FIRST CYCLE
First synthesis cycle results in two copies of target DNA sequence
DENATURE DNA
HYBRIDIZE PRIMERS
EXTEND NEW DNA STRANDS
SECOND CYCLE
Second synthesis cycle results in four copies of target DNA sequence

Source: *DNA Science*, see Fig. 13.

Application to Leukemia Diagnosis



120 samples from
60 leukemia patients

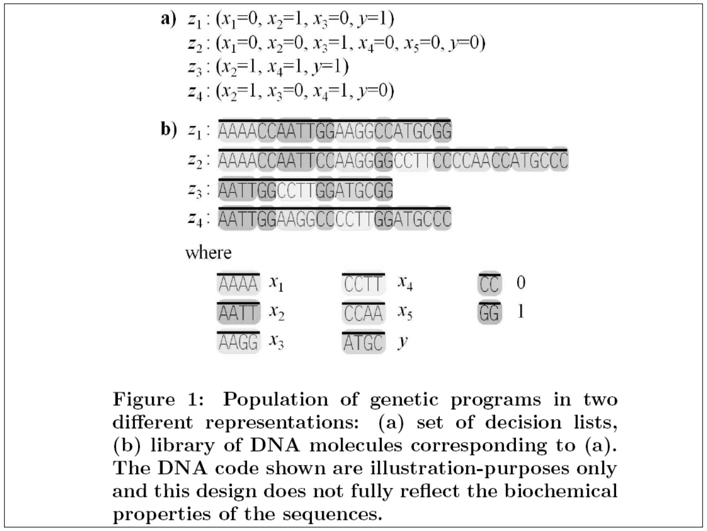
Gene expression data

Class: ALL/AML

Training with
6-fold validation

Diagnosis

[Cheok *et al.*, *Nature Genetics*, 2003]



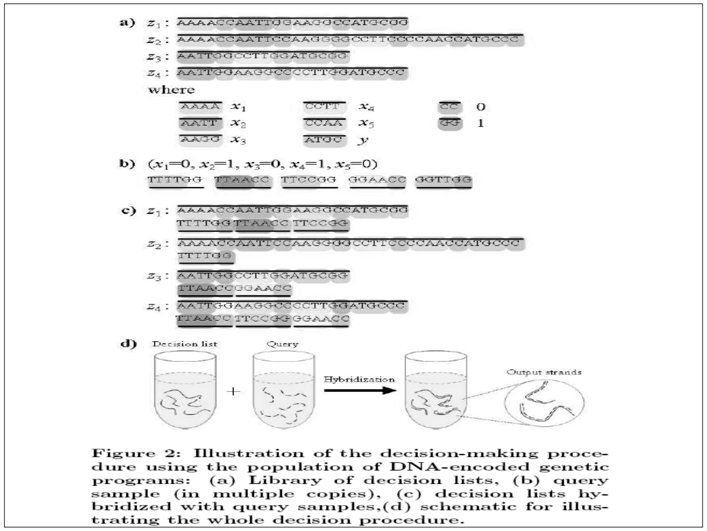
a) z_1 : (x_1=0, x_2=1, x_3=0, y=1)
z_2 : (x_1=0, x_2=0, x_3=1, x_4=0, x_5=0, y=0)
z_3 : (x_2=1, x_4=1, y=1)
z_4 : (x_2=1, x_3=0, x_4=1, y=0)

b) z_1 : AAAACCAATTGGAAGGCCATGCGG
z_2 : AAAACCAATTCCAAGGGGCCTTCCCCAACCATGCCC
z_3 : AATTGGCCTTGGATGCGG
z_4 : AATTGGAAGGCCCCTTGGATGCCC

where

AAAA	x_1	CCTT	x_4	CC	0
AATT	x_2	CCAA	x_5	GG	1
AAGG	x_3	ATGC	y		

Figure 1: Population of genetic programs in two different representations: (a) set of decision lists, (b) library of DNA molecules corresponding to (a). The DNA code shown are illustration-purposes only and this design does not fully reflect the biochemical properties of the sequences.



a) z_1 : AAAACCAATTGGAAGGCCATGCGG
z_2 : AAAACCAATTCCAAGGGGCCTTCCCCAACCATGCCC
z_3 : AATTGGCCTTGGATGCGG
z_4 : AATTGGAAGGCCCCTTGGATGCCC
where

AAAA	x_1	CCTT	x_4	CC	0
AATT	x_2	CCAA	x_5	GG	1
AAGG	x_3	ATGC	y		

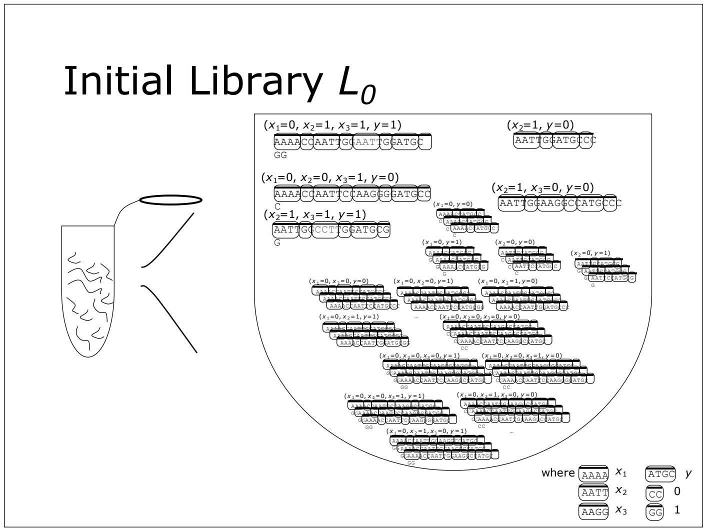
b) (x_1=0, x_2=1, x_3=0, x_4=1, x_5=0)
TTTTGG TTAACC TTCCGG GGAACC GGTTGG

c) z_1 : AAAACCAATTGGAAGGCCATGCGG
TTTTGGTTAACCTTCCGG
z_2 : AAAACCAATTCCAAGGGGCCTTCCCCAACCATGCCC
TTTTGG
z_3 : AATTGGCCTTGGATGCGG
TTAACCGGAACC
z_4 : AATTGGAAGGCCCCTTGGATGCCC
TTAACCTTCCGGGGAACC

d)

Figure 2: Illustration of the decision-making procedure using the population of DNA-encoded genetic programs: (a) Library of decision lists, (b) query sample (in multiple copies), (c) decision lists hybridized with query samples,(d) schematic for illustrating the whole decision procedure.

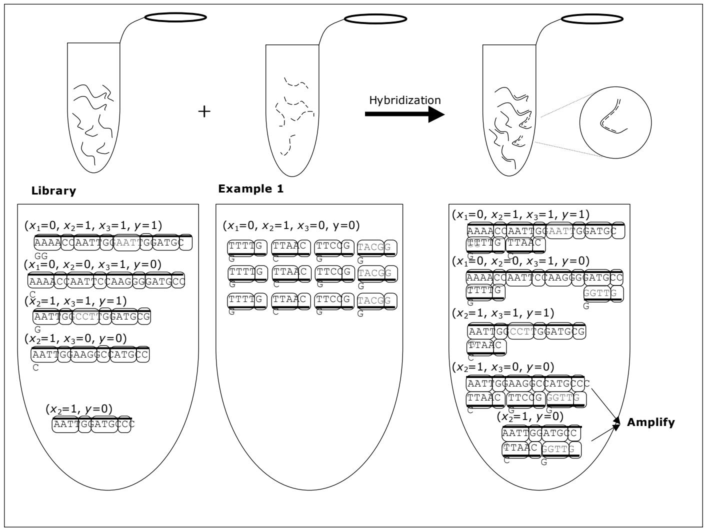
Initial Library L_0



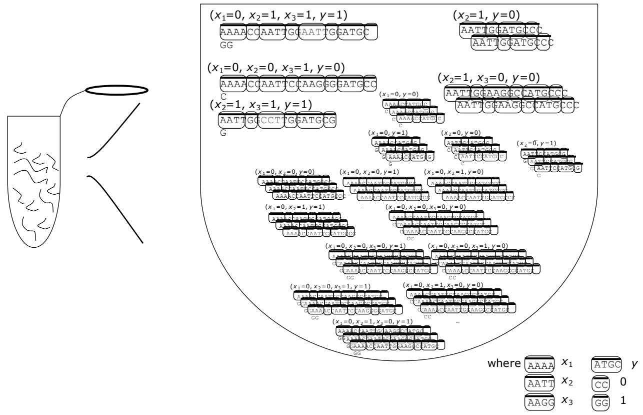
(x_1=0, x_2=1, x_3=1, y=1)
AAAACCAATTGGAATTGGATGC
GG
(x_2=1, y=0)
AATTGGATGCCC
(x_1=0, x_2=0, x_3=1, y=0)
AAAACCAATTCCAAGGGGATGCC
C
(x_2=1, x_3=0, y=0)
AATTGGAAGGCCATGCCC
(x_2=1, x_3=1, y=1)
AATTGGCCTTGGATGCG
G
(x_1=0, y=0)
(x_1=0, y=1)
(x_2=0, y=0)
(x_2=0, y=1)
(x_1=0, x_2=0, y=0)
(x_1=0, x_2=0, y=1)
(x_1=0, x_2=1, y=0)
(x_1=0, x_2=1, y=1)
...
(x_1=0, x_2=0, x_3=0, y=0)

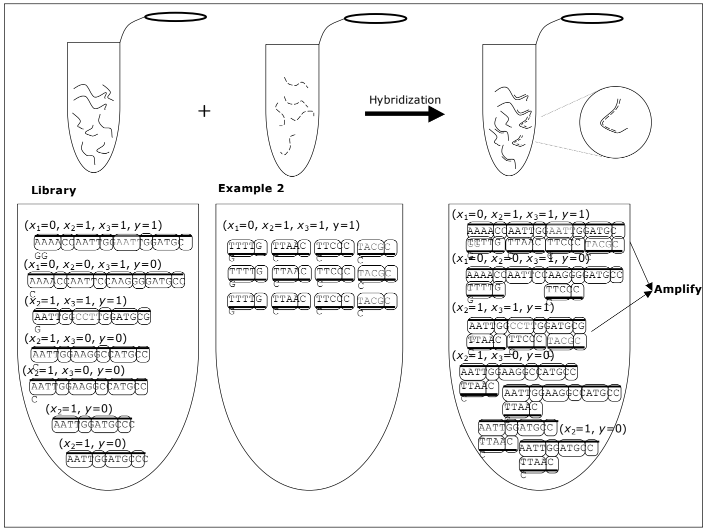
where

AAAA	x_1	ATGC	y		
AATT	x_2	CC	0		
AAGG	x_3	GG	1		

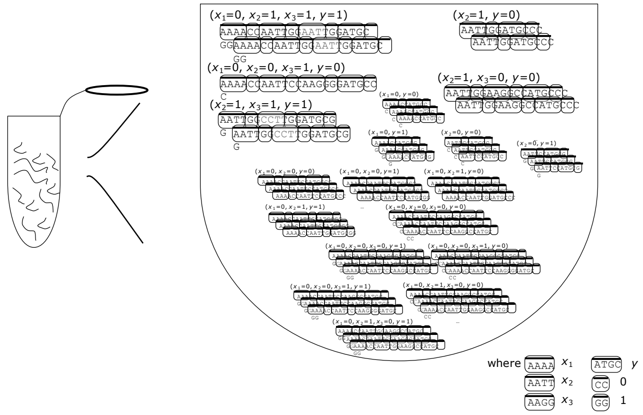


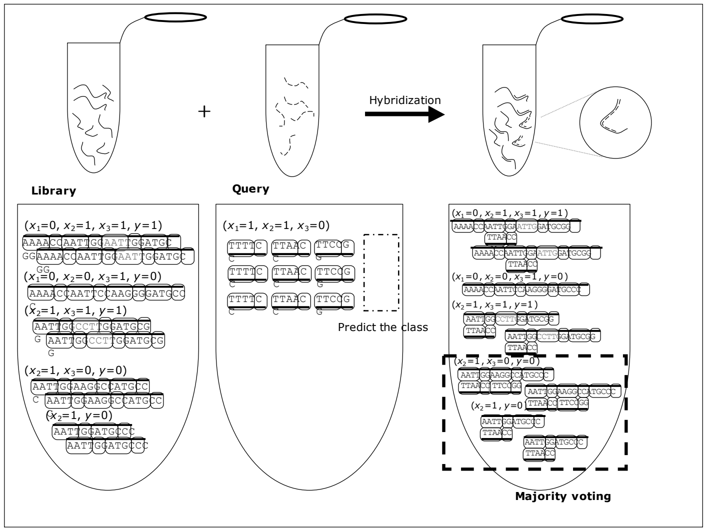
Updated Library L_1





Updated Library L_2





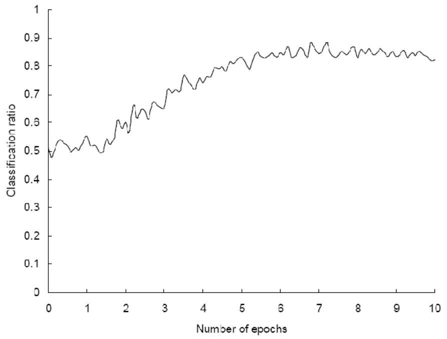


Figure 5: Fitness evolution of the population of molecular genetic programs. Though there are fluctuations the fitness values tend to converge 90 % accuracy. The reproduction rate was 0.01.

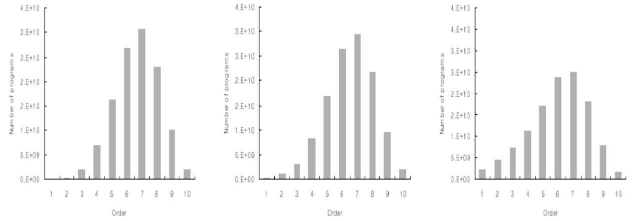


Figure 6: Distribution of the size of genetic programs. Shown are the number of programs of each size in the final population in a run. It shows the tendency that, as generation goes on, smaller programs are used more frequently than larger ones. The reproduction rate was 0.01.

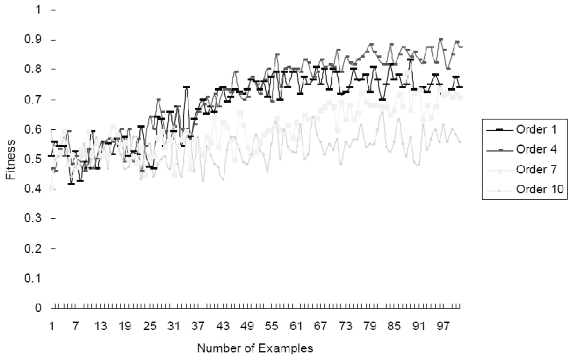


Figure 7: Fitness curve for runs with fixed-size programs. Shown are average fitness values for runs with programs of fixed-order 1, 4, 7, and 10.

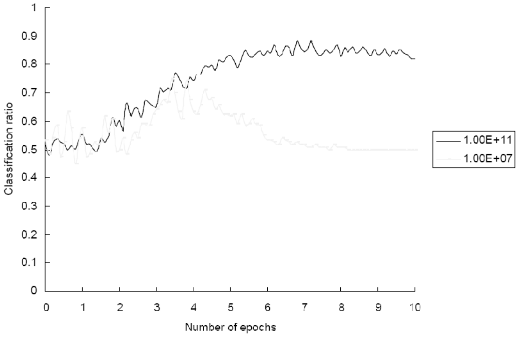
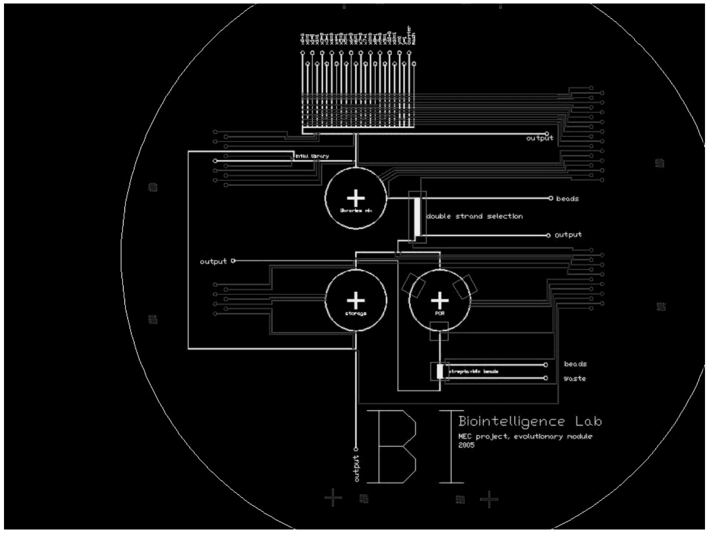


Figure 8: Effect of population size on ensemble performance. Shown are the best-fitness curves for population sizes of 10^{11} (in our experiments) 10^{7} and (subsampling case for testing). The results show that too much subsampling degrades the performance.

MP vs. GP

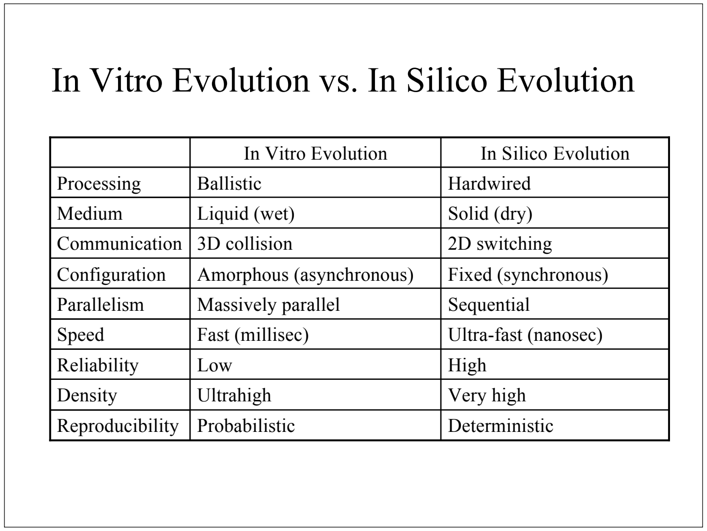
	Genetic Programming (G P)	Molecular Programming (MP)
Representation	Variable-size trees	Variable-length lists
Variation	Random xover, mutation	Combinatorial sampling
Selection	Proportional selection	Amplification (PCR)
Population size	~ O(10^4)	~ O(10^{15})
Parallelism	Can be parallelized	Inherently parallel
Solution	Single individual	Ensemble of individuals
Interaction	2D matrix	3D collision
Material	Silicon (dry, hard)	Carbon (wet, soft)



Molecular Programming (MP) as a New P aradigm for Molecular Computing

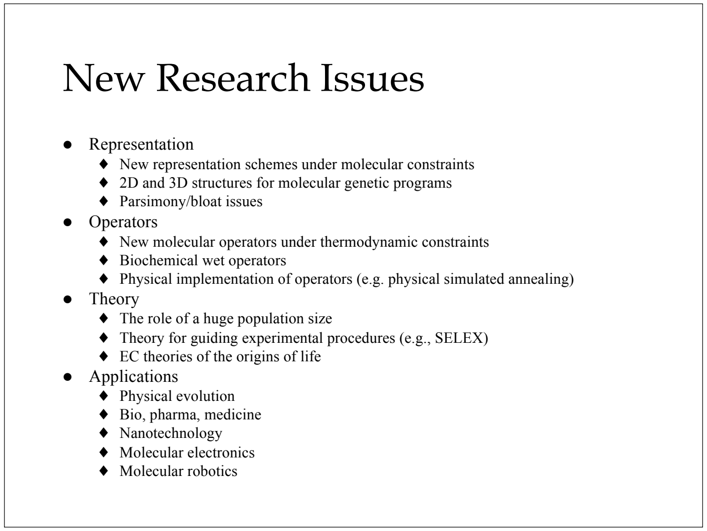
- Scalability
  - ♦ *Problem:* For big problems, exhaustive search does not work.
  - ♦ *Solution:* Evolutionary search
- Reliability
  - ♦ *Problem:* DNA reaction is error-prone.
  - ♦ *Solution:* Probabilistic formulation
- Fault tolerance
  - ♦ *Problem:* What if a single molecule malfunctions?
  - ♦ *Solution:* Ensemble machine approach
- Design
  - ♦ *Problem:* How to design the decision (or diagnosis) rules?
  - ♦ *Solution:* Evolutionary learning from examples

New Issues for the EC Researchers



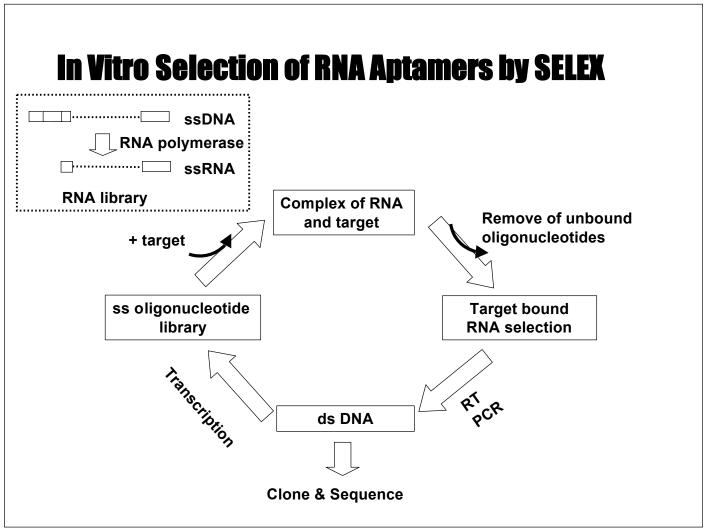
In Vitro Evolution vs. In Silico Evolution

	In Vitro Evolution	In Silico Evolution
Processing	Ballistic	Hardwired
Medium	Liquid (wet)	Solid (dry)
Communication	3D collision	2D switching
Configuration	Amorphous (asynchronous)	Fixed (synchronous)
Parallelism	Massively parallel	Sequential
Speed	Fast (millisec)	Ultra-fast (nanosec)
Reliability	Low	High
Density	Ultrahigh	Very high
Reproducibility	Probabilistic	Deterministic



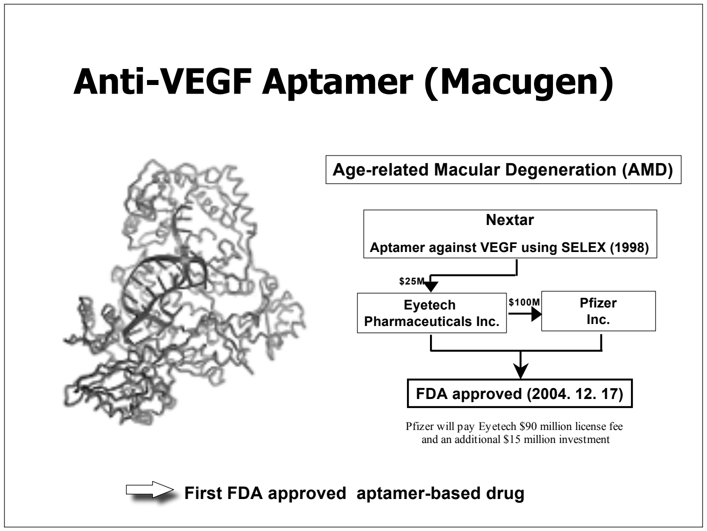
New Research Issues

- Representation
  - New representation schemes under molecular constraints
  - 2D and 3D structures for molecular genetic programs
  - Parsimony/bloat issues
- Operators
  - New molecular operators under thermodynamic constraints
  - Biochemical wet operators
  - Physical implementation of operators (e.g. physical simulated annealing)
- Theory
  - The role of a huge population size
  - Theory for guiding experimental procedures (e.g., SELEX)
  - EC theories of the origins of life
- Applications
  - Physical evolution
  - Bio, pharma, medicine
  - Nanotechnology
  - Molecular electronics
  - Molecular robotics



In Vitro Selection of RNA Aptamers by SELEX

ssDNA
RNA polymerase
ssRNA
RNA library
Complex of RNA and target
+ target
Remove of unbound oligonucleotides
ss oligonucleotide library
Target bound RNA selection
Transcription
RT PCR
ds DNA
Clone & Sequence



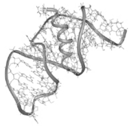
Anti-VEGF Aptamer (Macugen)

Age-related Macular Degeneration (AMD)
Nextar
Aptamer against VEGF using SELEX (1998)
$25M
Eyetech Pharmaceuticals Inc.
$100M
Pfizer Inc.
FDA approved (2004. 12. 17)

Pfizer will pay Eyetech $90 million license fee and an additional $15 million investment

First FDA approved aptamer-based drug

RNA Aptamers into Therapeutics



- Easily screened
- High affinity and specificity
- Reduced in size and chemically synthesized
- Easily modified for bioavailability or delivery
- Reversible antagonist; regulatable
- Highly expressed
- May not induce immune response

1) *As drug leads – lessen form VEGF$_{165}$*
2) *As gene therapy*

RNA Aptamers into Diagnostics

- Molecular ligand to variable molecules including carbohydrates and lipids
- High affinity and specificity
- Spot onto solid surface with high density
- Mass production with low cost, rapidity and high purity
- Reversible denaturation: stable and long storage
- Fixed with variable reporter

Rivalry to Antibody
Nanochip/biosensor

Applications for SELEX

Insights into the prebiotic earth

- Identification of the catalytic potential of RNA and DNA; Selection for enzymatic functions (ligase, polymerase, RNase, peptide bond formation, Diels-Alder reaction)

Applied (Medical) research

- diagnostic (ELISA, FACS) and therapeutic use of aptamers as replacement and or extension to antibodies (K_d's in the pM to nM range)

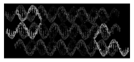
Genomic SELEX and regulatory loops

- random integration of genomic sequences into SELEX oligonucleotides; selection for unidentified binding sites to regulatory proteins (MetJ; MS2 coat protein, U1A protein)

Programmable Patterning of DNA Lattices

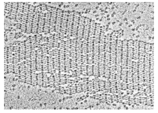
(John Reif, Duke)

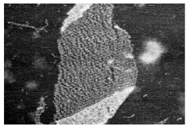
A New, Powerful Technology
- for the construction of molecular scale structures
- for Rendering Patterns at the Molecular Level.



A 2D DNA lattice is constructed by a self-assembly process:
--Begins with the assembly of DNA tile nanostructures:
- DNA tiles of size 14 x 7 nanometers
- Composed of short DNA strands with Holliday junctions

- These DNA tiles self-assemble to form a 2D lattice:



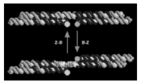


-The Assembly is Programmable:
-Tiles have sticky ends that provide programming for the patterns to be formed.
-Alternatively, tiles self-assemble around segments of a DNA strand encoding a 2D pattern.
- Patterning: Each of these tiles has a surface perturbation depending on the pixel intensity.
-pixel distances 7 to 14 nanometers
-not diffraction limited

Key Applications: Assembly of molecular electronic components & circuits, molecular robotic c omponents, image rendering, cryptography, mutation detection.

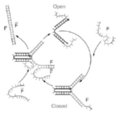
DNA-Based Nanorobotics Devices

Rotation



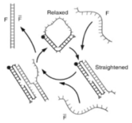
(Mao *et al* 99)

Open/close



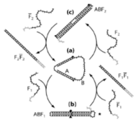
(Yurke *et al* 00)

Open/close



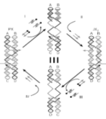
(Simmel *et al* 01)

Open/close



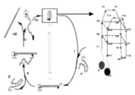
(Simmel *et al* 02)

Rotation



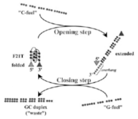
(Yan *et al* 02)

Extension/contraction



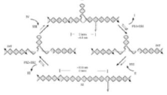
(Li *et al* 02)

Extension/contraction



(Alberti *et al* 03)

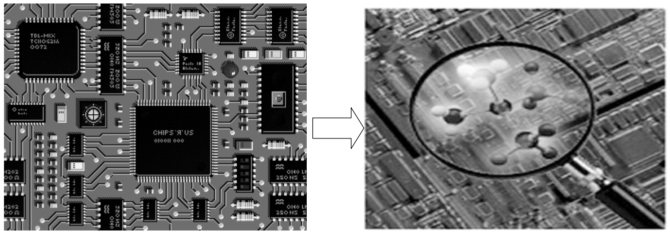
Extension/contraction



(Feng *et al* 03)

Evolvable Biomolecular Hardware

- Sequence programmable and evolvable molecular systems c an be constructed as cell-free chemical systems using biom olecules such as DNA and proteins.



Acknowledgements

Collaborating Labs

- Biointelligence Laboratory, Seoul National University
- Biochemistry Lab, Seoul National Univ. Medical School
- Cell and Microbiology Lab, Seoul National University
- Advanced Proteomics Lab, Hanyang University
- DigitalGenomics, Inc.
- GenoProt, Inc.

Supported by

- National Research Lab Program of Min. of Sci. & Tech.
- Next Generation Tech. Program of Min. of Ind. & Comm.

More Information at

- http://bi.snu.ac.kr/MEC/
- http://cbit.snu.ac.kr/

Books and Web Sites

Books (General)

- Calude, C.S., Casti, J. and Dinneen, M.J. (Eds.) *Unconventional Models of Computation*, Springer, 1998.
- Eigen, M. and Winkler, R., *Laws of the Game: How the Principles of Nature Govern Chance*, Princeton University Press, 1993 (English translation).
- Kauffman, S.A., *The Origins of Order: Self-Organization and Selection in Evolution*, Oxford University Press, 1993.
- Kueppers, B.-O.. *Molecular Theory of Evolution: Outline of a Physico-Chemical Theory of the Origin of Life*, Springer, 1983.
- Landweber, L.F., Winfree, E. (Eds.) *Evolution as Computation*, Springer, 2003,
- Page, R.D.M and Holmes, E.C., *Molecular Evolution: A Phylogenetic Approach*, Blackwell Science, 1998.
- Scheutz, M. (Ed.) *Computationalism: New Directions*, MIT Press, 2000.
- Sienko, T., Adamatzky, A., Rambidi, N.G., and Conrad, M. (Eds.) *Molecular Computing*, MIT Press, 2003.

Some References

- Adleman, L., “Computing with DNA,” *Scientific American*, 34-41, 1998.
- Conrad, M., “Molecular computing: The lock-key paradigm,” *IEEE Computer*, 25(1): 11-20, 1992.
- Joyce, G. F. “The antiquity of RNA-based evolution,” *Nature*, 418: 214-221, 2002.
- Seeman, N. C., “Biochemistry and structural DNA nanotechnology: An evolving symbiotic relationship,” *Biochemistry*, 42(24): 7259-7269, 2003.
- Tuerk, C. and Gold, L., “Systematic evolution of ligands by exponential enrichment: RNA ligands to bacteriophage T4 DNA polymerase,” *Science*, 249(4968): 505-510, 1990.
- Wright, M. C. and Joyce, G. F., “Continuous in vitro evolution of catalytic function,” *Science*, 276: 614-617, 1997.
- Zhang, B.-T. and Jang, H.-Y., Molecular programming: Evolving genetic programs in a test tube, *Proc. Genetic and Evolutionary Computation Conf. (GECCO-2005)*, Washington, D.C., 2005 (to appear)
- Zhang, B.-T. and Jang, H.-Y., A Bayesian algorithm for in vitro molecular evolution of pattern classifiers, *Proc. 10th Int. Conf. on DNA Computing*, DNA10, LNCS 3384: pp. 458-467, 2005.

Web Sites

- California Inst. of Tech. http://www.dna.caltech.edu/ (Erik Winfree)
- Duke Univ. http://www.cs.duke.edu/~reif/ (John Reif)
- Harvard Univ. http://genetics.mgh.harvard.edu/szostakweb/ (Jack Szostak)
- New York Univ. http://seemanlab4.chem.nyu.edu/ (Ned Seeman)
- Scripps Res. Inst. http://exobio.ucsd.edu/joyce.htm/ (Gerald Joyce)
- Seoul National Univ. http://bi.snu.ac.kr/ (Tak Zhang)
- Univ. of Bonn http://famulok.chemie.uni-bonn.de/ (Michael Famulok)
- Univ. of Tokyo http://nicosia.is.s.u-tokyo.ac.jp/ (Masami Hagiya)
- Univ. of Southern California http://www.usc.edu/dept/molecular-science/ (Leon Adleman)
- Univ. of Vienna http://www.tbi.univie.ac.at/ (Peter Schuster)
- Weizmann Inst. of Tech. http://www.weizmann.ac.il/mathusers/lbn/ (Ehud Shapiro)