Telling Molecules What to Do

Luca Cardelli

University of Oxford

2021-06-17, IMT Lucca
Objectives

• The promises of Molecular Programming
  • In Science & Medicine
  • In Engineering
  • In Computing

• The current practice of Molecular Programming
  • DNA technology
  • Molecular languages and tools
  • Molecular algorithms
Synthetic Biology Market

Top synthetic biology fundraisers of 2018

<table>
<thead>
<tr>
<th>Company</th>
<th>Fundraiser (Millions)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moderna Therapeutics</td>
<td>$300M</td>
</tr>
<tr>
<td>Zymeworks</td>
<td>$200M</td>
</tr>
<tr>
<td>Synthorx</td>
<td>$150M</td>
</tr>
<tr>
<td>Suva Biopharma</td>
<td>$100M</td>
</tr>
<tr>
<td>Astrella</td>
<td>$75M</td>
</tr>
<tr>
<td>Bolt Biociences</td>
<td>$50M</td>
</tr>
<tr>
<td>Impossible Foods</td>
<td>$50M</td>
</tr>
<tr>
<td>Precision BioScience</td>
<td>$50M</td>
</tr>
<tr>
<td>Synthego</td>
<td>$50M</td>
</tr>
<tr>
<td>Berkeley Lights</td>
<td>$50M</td>
</tr>
<tr>
<td>Genomatica</td>
<td>$50M</td>
</tr>
<tr>
<td>Synlogic</td>
<td>$25M</td>
</tr>
<tr>
<td>Beam Therapeutics</td>
<td>$25M</td>
</tr>
<tr>
<td>Intrexion</td>
<td>$25M</td>
</tr>
<tr>
<td>Incroce</td>
<td>$25M</td>
</tr>
<tr>
<td>Amira</td>
<td>$20M</td>
</tr>
<tr>
<td>Others</td>
<td>$20M</td>
</tr>
<tr>
<td>Total</td>
<td>$1.2B</td>
</tr>
</tbody>
</table>

SYNTHETIC BIOLOGY TECHNOLOGY INNOVATIONS LANDSCAPE

Synthetic Biology Market Size & Growth

- 2017: 5.63 billion
- 2018: 13.4 billion
- 2020: 38.7 billion

Key Enabling Technologies

- DNA Synthesis
- Sequencing Technologies
- Genome Engineering
- Microfluidics Technologies
- Biofuels Technologies
- Bioinformatics technologies
- Biological Components & integrated Systems
- Pathway engineering

Synthetic Biology: Core Market Segments

- DNA Synthesis
- Oligonucleotide Synthesis
- Pharmaceuticals
- Chemicals
- Biofuels
- Agriculture
- Synthetic DNA
- Synthetic Genes
- Synthetic Cells
- XNA
- Chassis organisms
- Enabling Products

For more information, Visit: https://www.pintels.com
For email: contactus@vajrasoftinc.com
Some (ongoing) successes stories

- ($4Bn) Reprogram a patient's own blood cells to recognise and destroy specific cancers.
- 90% remission in terminally ill leukemia patients

- ($300M) Reprogram yeast to synthesise chemicals
- Antimalarial drug in production (with Sanofi)
- Jet fuel used in commercial flights (with Total)

- Supply custom organisms for bio fabrication

- Grow meat, leather ($100Bn market) in the lab
- Proofs of concept already in production
Hacking Yoghurt

Tuur van Balen - Hacking Yoghurt
- genetically modify your yoghurt in your own kitchen

https://www.youtube.com/watch?v=Co8NOnErrPU
Live Clothing

Scientists Sew Genetically Modified E. Coli into Living Clothing

Harnessing the hygroscopic and biofluorescent behaviors of genetically tractable microbial cells to design biohybrid wearables

Wen Wang1,2, Lingh You3, Chi-Hsi Cheng1,2, Tong Zhang4, Hiroaki Aizawa6, Lida Wang5, Guanyun Wang5, Okuano Akiroytfe2

* See all authors and affiliations
Molecular Programming

A *technology* (and theory of computation) based on information-bearing molecules of historically biological origin (DNA/RNA) non necessarily involving living matter
Molecular Programming: The Hardware Aspect

Smaller and smaller things can be built
Smaller and Smaller

First working transistor
John Bardeen and Walter Brattain, Dec. 23, 1947

First integrated circuit

50+ years later

Jan 2010 25nm NAND flash
Intel & Micron, ~50 atoms

Jun 2018 7nm (54nm pitch)
TSMC, Intel, Samsung, GlobalFoundries - mass production

Single molecule transistor
Observation of molecular orbital gating
*Nature*, 2009; 462 (7276): 1039

Molecules on a chip

Very few Moore’s cycles left!

Scanning tunneling microscope image of a silicon surface: 10nm is ~20 atoms (in cubic lattice)

Race to the Bottom

Moore's Law is approaching the single-molecule limit

Carlson's Curve is the new exponential growth curve in technology

In both cases, we are now down to molecules

Waiter! There is fly DNA in my soup!

The SmidgION: A portable DNA sequencer that runs on an iPhone

Oxford Nanopore
Building the *Smallest Things*

- How do we build structures that are by definition smaller than your tools?
- Basic answer: you can’t. Structures (and tools) should build themselves!
- By *programmed self-assembly*
Molecular IKEA

• Nature can self-assemble. Can we?

• “Dear IKEA, please send me a chest of drawers that assembles itself.”

• We need a magical material where the pieces are pre-programmed to fit into each other.

• At the molecular scale many such materials exist...

Programmed Self-Assembly

Proteins

Membranes

DNA/RNA
Molecular Programming: The Software Aspect

Smaller and smaller things can be programmed
We can program...

- Information
  - Completely!

Information → Computing → Information
We can program...

- Forces
  - Completely!
    (Modulo sensors/actuators)
We can program...

- **Matter**
  - Completely and directly! By self-assembly.
  - Currently: only DNA/RNA.

- But DNA is an amazing *material*

*It's like a 3D printer without the printer!*  
[Andrew Hellington]
DNA

G-C Base Pair
Guanine-Cytosine

T-A Base Pair
Thymine-Adenine

Sequence of Base Pairs (GACT alphabet)

Interactive DNA Tutorial
(http://www.biosciences.bham.ac.uk/aba/minchin/tutorials/dna.html)
DNA Specs

- DNA in each human cell
  - 3 billion base pairs
  - 2nm thick = 4 silicon atoms (in silicon lattice)!
  - 0.34nm per basepair = 2 bits in 2/3 silicon atom!
  - 2 meters long copied in parallel at each cell division!
  - 750 megabytes
    - 80% functional, but only 1.5% protein coding
  - folded into a 6μm spherical nucleus
    \[= 140 \text{ exabytes (million terabytes)} / \text{m}^3\]
    => all the data on the internet fits in a shoebox!

- DNA in each human body
  - 10 trillion cells
  - 133 Astronomical Units long
  - 7.5 octabytes (replicated)

- DNA in human population
  - 20 million light years long

DNA wrapping into chromosomes

Andromeda Galaxy
2.5 million light years away
DNA Benchmarks

DNA replication in real time

In Humans: 50 nucleotides/second
Whole genome in a few hours (with parallel processing)

In Bacteria: 1000 nucleotides/second
(higher error rate)

DNA transcription in real time

RNA polymerase II: 15-30 base/second

Drew Berry
http://www.wehi.edu.au/wehi-tv
One molecule to rule them all

- There are many, many nanofabrication techniques and materials

- But only DNA (and RNA) can:
  - Organize ANY other matter [caveats apply]
  - Execute ANY kinetics [caveats: up to time scaling]
  - Assemble Nano-Control Devices
  - Interface to Biology

Building Nano-Control Devices

All the components of nanocontrollers can already be built entirely and solely with DNA, and interfaced to the environment.

- DNA Aptamers
- Sensing
- Computing
  - DNA Logical Gates
  - Constructing
  - Actuating
- Self-assembling DNA Tiles
- DNA Walkers & Cages
Sensing

Constructing

Actuating

Computing

Sensing
Aptamers

Artificially evolved DNA molecules that stick to (almost) anything you like highly selectively.
Pathogen Spotlights

- DNA aptamer binds to:
  - A) a pathogen
  - B) a molecule our immune system (when allergic) hates and immediately removes (eats) along with anything attached to it!

- Result: instant immunity
  - Mice poisoned with Anthrax plus aptamer (100% survival)
  - Mice poisoned with Anthrax (not so good)

Kary Mullis (incidentally, also Nobel prize for inventing the Polymerase Chain Reaction)
Constructing

Actuating

Computing

Sensing

Constructing Actuating
DNA Walkers

A cargo-sorting DNA robot

DNA Robotics
Constructing

- Sensing
- Computing
- Constructing
- Actuating

The 3D model of the computer-designed bear shape shown on top was fabricated into the nanostructures visualized with transmission electron microscopy (below).

Credit: Wyss Institute at Harvard University
Crosslinking

the dawn of structural DNA nanotechnology
Crosslinking

the dawn of structural DNA nanotechnology
Crosslinking

the dawn of structural DNA nanotechnology
Crosslinking

the dawn of structural DNA nanotechnology
Crosslinking

In nature, crosslinking is deadly (blocks DNA replication).

In engineering, crosslinking is the key to using DNA as a construction material.
DNA Tiling

4 sticky ends

crosslinking

Construction and manipulation of DNA tiles in free space
2D DNA Lattices

Chengde Mao
Purdue University, USA
3D DNA Structures

Ned Seeman
NYU

3D Crystal

Andrew Tuberfield
Oxford

Tetrahedron

Friedrich Simmel
Munich

Robotic Arm
CADnano

William Shih
Harvard

https://www.youtube.com/watch?v=Ek-FDPyGyyg

S.M. Douglas, H. Dietz, T. Liedl, B. Högb erg, F. Graf and W. M. Shih
Self-assembly of DNA into nanoscale three-dimensional shapes, Nature (2009)
DNA Origami

*Folding* a long (6407bp) naturally occurring circular ssDNA (from bacteriophage M13) via lots of short ‘staple’ strands that constrain its shape

Paul Rothemund’s
“Disc with three holes” (2006)
*Nature* 440, 297, 2006

Black/gray: 1 long viral strand (natural DNA)
Color: many short staple strands (synthetic DNA)

Lulu Qian’s
Hierarchical assembly (2017)

AFM image
DNA Circuit Boards

- DNA origami are arrays of uniquely-addressable locations
  - Each staple is different and binds to a unique location on the origami
  - It can be extended with a unique sequence so that something else will attach uniquely to it.

- More generally, we can bind “DNA gates” to specific locations
  - And so connect them into “DNA circuits” on a grid
  - Only neighboring gates will interact

Some staples are attached to “green blobs” (as part of their synthesis) Other staples aren’t
DNA Storage (Read/Write)

Information-rich physical structures can be used for storage.

DNA has a data density of $140$ exabytes ($1.4 \times 10^{20}$ bytes) per $mm^3$ compared to state-of-the-art storage media that reaches $\sim 500$ megabytes ($5 \times 10^8$ bytes) per $mm^3$.

DNA has been shown to be stable for millions of years.

We have machines that can read (sequence) and write (synthesize) DNA. The Carslon Curve of “productivity” is growing much faster than Moore’s Law.

Cost of sequencing is decreasing rapidly ($\$1000$ whole human genome), while cost of synthesis is decreasing very slowly.

[Rob Carlson, www.synthesis.cc]
Curing

Sensing
Computing
Constructing
Actuating
Interfacing to Biology

- A doctor in each cell
Programmed Drug Delivery
Molecular Programming: The Execution Aspect

How do we "run" a molecular program?
Programming Language: Chemistry

- A Lingua Franca between Biology, Dynamical Systems, and Concurrent Languages

- Chemical Reaction Networks
  - \[ A + B \rightarrow r \] C + D \] (the program)

- Ordinary Differential Equations
  - \[ \frac{d[A]}{dt} = -r[A][B] \] ... (the behavior)

- Rich analytical techniques based on Calculus and more recently on stochastic models
Chemistry as a Concurrent Language

- A connection with the theory of concurrency
  - Via Process Algebra and Petri Nets
## Chemical Programming Examples

**specification**

<table>
<thead>
<tr>
<th>Y := min(X1, X2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>X1 + X2 -&gt; Y</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Y := max(X1, X2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>X1 -&gt; L1 + Y</td>
</tr>
<tr>
<td>X2 -&gt; L2 + Y</td>
</tr>
<tr>
<td>L1 + L2 -&gt; K</td>
</tr>
<tr>
<td>Y + K -&gt; 0</td>
</tr>
</tbody>
</table>

**program**

<table>
<thead>
<tr>
<th>max(X1,X2)= (X1+X2)-min(X1,X2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(but is not computed &quot;sequentially&quot;: it is a form of concurrent computation)</td>
</tr>
</tbody>
</table>

**chemical reaction network**
Chemical Reaction Networks

- Finite list of chemical reactions over a finite set of species
  - N.B.: "abstract" species, not specific atoms/molecules that physically exist

- Computationally Powerful
  - Turing-complete up to an arbitrarily small error

- Full Turing Completeness
  - When including complexation (polymerization), which DNA enables
    (complexation encodes an actual infinity of chemical reactions by finite means)
How do we “run” Chemistry?

• Chemistry is not easily executable
  • “Please Mr Chemist, execute me this bunch of reactions that I just made up”

• Most molecular languages are not executable
  • They are descriptive (modeling) languages

• How can we execute molecular languages?
  • With real molecules?
  • That we can design ourselves?
  • And that we can buy on the web?
DNA Strand Displacement

An "unnatural" use of DNA for emulating any system of chemical reactions with real molecules
Domains

- Subsequences on a DNA strand are called **domains**
  - provided they are “independent” of each other

- Differently named domains must not **hybridize**
  - With each other, with each other’s complement, with subsequences of each other, with concatenations of other domains (or their complements), etc.
Short Domains

Reversible Hybridization

DNA double strand
Long Domains

Irreversible Hybridization
Strand Displacement
Strand Displacement

“Toehold Mediated”
Strand Displacement

Toehold Binding
Strand Displacement

Branch Migration
Strand Displacement

Displacement
Strand Displacement

Irreversible release
Bad Match
Bad Match
Bad Match
Bad Match

Cannot proceed
Hence will undo
Two-Domain Architecture

- Signals: 1 toehold + 1 recognition region
  ![Diagram of signals](image)

- Gates: “top-nicked double strands” with open toeholds
  ![Diagram of gates](image)

Garbage collection “built into” the gate operation

Two-Domain DNA Strand Displacement

Luca Cardelli

In S. B. Cooper, E. Kashefi, P. Panangaden (Eds.):
Developments in Computational Models (DCM 2010).
Transducer
Transducer $x \rightarrow y$
Transducer $x \rightarrow y$

Built by self-assembly!

$ta$ is a *private* signal (a different ‘a’ for each $xy$ pair)
Transducer $x \rightarrow y$
Transducer $x \rightarrow y$
So far, a \textbf{tx signal} has produced an \textbf{at cosignal}.
But we want signals as output, not cosignals.
Transducer $x \rightarrow y$
Transducer $x \rightarrow y$
Transducer $x \rightarrow y$
Transducer $x \rightarrow y$

Here is our output *ty signal*. But we are not done yet:
1) We need to make the output irreversible.
2) We need to remove the garbage.

We can use (2) to achieve (1).
Transducer $x \rightarrow y$
Transducer $x \rightarrow y$
Transducer $x \rightarrow y$
Transducer $x \rightarrow y$
Transducer $x \rightarrow y$
Transducer $x \rightarrow y$

Done.

N.B. the gate is consumed: it is the energy source
(no proteins, no enzymes, no heat-cycling, etc.; just DNA in salty water)
Transducer $x \rightarrow y$
Reaction: \[ x + y \rightarrow z + w \]
Reaction: \[ x + y \rightarrow z + w \]
Reaction \[ x + y \rightarrow z + w \]  

anti-garbage  \[\rightarrow\]  garbage  

harmless  \[\rightarrow\]  harmless
Join \( x + y \rightarrow z \)
Approximate Majority Algorithm

- Given two populations of agents (or molecules)
  - Randomly communicating by radio (or by collisions)
  - Reach an agreement about which population is in majority
  - By converting all the minority to the majority
    [Angluin et al., Distributed Computing, 2007]

- 3 rules of agent (or molecule) interaction
  - \( X + Y \rightarrow B + B \)
  - \( B + X \rightarrow X + X \)
  - \( B + Y \rightarrow Y + Y \)
Optimal Consensus Algorithm

- Fast: reaches agreement in $O(\log n)$ time w.h.p.
  - $O(n \log n)$ communications/collisions
  - Even when initially $|X| = |Y|$! (stochastic symmetry breaking)

- Robust: true majority wins w.h.p.
  - If initial majority exceeds minority by $\omega(\sqrt{n \log n})$
  - Hence the agreement state is stable

Stochastic simulation of worst-case scenario with initially $|X| = |Y$
DNA Implementation of the Approximate Majority algorithm
Some Large-scale Circuits (so far...)

Computing the square root of a 4-bit number

Classifying 4 distinct 4-bit patterns via 4 neurons

Classifying 9 distinct 100-bit patterns via WTA networks
Scaling up: DNA Circuit Boards

A spatially localized architecture for fast and modular DNA computing

Gourab Chatterjee, Neil Dalchau, Richard A. Muscat, Andrew Phillips and Georg Seelig

The first computational circuit boards made of DNA

Avoiding Clocks

- Muller C-Element
  - A Boolean gate
  - When $x = y$ then $z = x = y$, otherwise $z$ remembers its last state.

Core C-Element
(AM with external inputs)

Full C-Element with output rectified by another AM

Chemical Reaction Network Designs for Asynchronous Logic Circuits.
Luca Cardelli, Marta Kwiatkowska, Max Whitby.
Natural Computing Journal.
Algorithm Design

A software pipeline for Molecular Programming
A Language for DNA Structures

- Describe the initial \textit{structures} (not behavior)
Compute Species and Reactions

- Recursively computed from the initial structures
Simulation

- Deterministic
- Stochastic (Gillespie)
- Probabilistic (CME)
- Linear Noise Approximation
- "JIT"
State Space Analysis

CTMC

INITIAL STATE:
- (1)
- (1)
- (1)

TERMINAL STATE:
- (1)
- (1)
- (1)
Model checking

- Export to PRISM probabilistic model checker
Verification

• Quantitative theories of system equivalence and approximation.
Physical Execution

A wetlab pipeline for Molecular Programming
Computer Aided Design

MSRC Biological Computation Group

Visual DSD
A Development Environment for DNA Strand Displacement

A programming language for composable DNA circuits
Andrew Phillips and Luca Cardelli
Output of Design Process

- Domain structures
  - (DNA sequences to be determined)

“Ok, how do I run this for real?”

“What are the actual DNA sequences?”
From Structures to Sequences

"Ok, where do I buy these?"

www.nupack.org
“DNA Synthesis”

DNA synthesis commonly refers to DNA replication - DNA biosynthesis (in vivo DNA amplification), Polymerase chain reaction - enzymatic DNA synthesis (in vivo).
From Sequences to Molecules

- Copy&Paste from nupack
"Ok, how do I run these?"
Add Water
Execute (finally!)

- Fluorescence is your one-bit ‘print’ statement
Output
Debugging

• A core dump
Engineering Entropy-Driven Reactions and Networks Catalyzed by DNA
David Yu Zhang, et al.
Science 318, 11121 (2007);
DOI: 10.1126/science.1148532
Final Remarks
"Modern" DNA Computing

• **Non-goals**
  • Not to solve NP-complete problems with large vats of DNA
  • Not to replace silicon

• **Bootstrapping a carbon-based technology**
  • To precisely control the organization and dynamics of matter and information at the molecular level
  • DNA is our engineering material
    • Its biological origin is “accidental” (but convenient)
    • It is an information-bearing programmable material
    • Other such materials will be (are being) developed
State of the art

- Building a full software/hardware pipeline for a new fundamental technology
  - Mathematical Foundations [~ concurrency theory in the 80’s]
  - Programming Languages [~ software engineering in the 70’s]
  - Analytical Methods and Tools [~ formal methods in the 90’s]
  - Device Architecture and Manufacturing [~ electronics in the 60’s]

- To realize the potential of Molecular Programming
  - “With no alien technology” [David Soloveichik]

- We have some good strategies. Device design is now largely a ‘software problem’ but with a significant ‘engineering scaleup and integration’ problem
Ongoing Challenges

- In-vivo DNA survivability
- Complexity (and crosstalk)
- Manufacturing
- Speed
- Energy
A Brief History of DNA

Turing Machine, 1936

DNA, -3,800,000,000

Transistor, 1947

Computer programming

DNA Algorithm, 1994

Molecular programming

Systematic manipulation of information

Systematic manipulation of matter

20th century

21st century

Structural DNA Nonotech, 1982
Resources

- DNA Computing and Molecular Programming Conference - incarnations since 1995
  http://www.dna-computing.org/

- Molecular Programming Project (Caltech - U.W. - Harvard - UCSF)

- Georg Seelig’s DNA Nanotech Lab at U.W. CS&E
  http://homes.cs.washington.edu/~seelig/

- Biological Computation Group at Microsoft
  https://www.microsoft.com/en-us/research/group/biological-computation/