Introduction

• Recording events in DNA


• High throughput sequencing and synthesis

http://www2.technologyreview.com/news/427677/nanopore-sequencing/

https://www.twistbioscience.com/technology
Preorder Recorder

• Detecting molecular events is very difficult and very important
• In science we want to know “what’s going on?”
• In bioengineering we want to know “what when wrong?”
• We often want to know the order of events to help determine causation

• We discuss a “preorder recorder” algorithm that reads out the preorder of first-occurrence of a set of events in a chemical soup, where an event is the appearance of a DNA/RNA strand in the soup
• These events could be DNA circuit signals, or naturally transcribed RNA, or DNA/RNA transduced in response to e.g. presence of certain proteins
How to Read DNA (Output)

• Fluorescence Readout
  • Limited readout capability: 3/4 "colors" of output.
  • Output can be read continuously over time

• Atomic Force Microscope Readout
  • Detecting shapes and patterns
  • Comprehensive view of the results

• Sequencing Readout
  • At the end of a computation, sequence the strand types left in the soup
  • Output is a multiset of strand types (each with a real-valued concentration)
High Throughput Sequencing

- **Sequencing by Synthesis**
  - Like Sanger sequencing, but done in parallel on a "lawn" of single strands, removing the fluorophores at each step to carry on.

- **Nanopore Sequencing**
  - ~ 200 single different DNA molecules sequenced in parallel


http://www2.technologyreview.com/news/427677/nanopore-sequencing/
How to Write DNA (Gates + Input)

- Synthesizing DNA using silicon microfabrication technology

Twist Bioscience developed a proprietary semiconductor-based synthetic DNA manufacturing process featuring a high-throughput silicon platform that allows us to miniaturize the chemistry necessary for DNA synthesis. This miniaturization allows us to reduce the reaction volumes by a factor of 1,000,000 while increasing throughput by a factor of 1,000, enabling the synthesis of 9,600 genes on a single silicon chip at full scale. Traditional synthesis methods produce a single gene in the same physical space using a 96-well plate.

=> DNA Storage
Cloning

- Higher quality than ordinary synthesis, and supporting very long strands (very complex "gates").
The Pace of Biotechnology

Sequencing costs (USD / Mb)

How can we take full advantage of all this, for DNA-based algorithms?

Wetterstrand KA. DNA Sequencing Costs: Data from the NHGRI Genome Sequencing Program (GSP) Available at: www.genome.gov/sequencingcostsdata.
Many DNA strand displacement computational schemes are "Universal"

- 4-domain, 3-domain, 2-domain, split-domain ...

- Can be used to systematically compile arbitrary finite chemical reaction networks to DNA molecules that exhibit (approximately) the same kinetics.

- But not all can be written by cloning and read by sequencing.
A Typical 3-domain Scheme

2-input "join"  \( x + y \rightarrow z \)

Non-clonable, non-sequenceable because of the secondary structures
A 2-domain Scheme

2-input "join" \[ x + y \rightarrow \emptyset \]
Clonable but not Sequenceable

Sequencing (of double strands) must be preceded by *polymerase extension* (to remove single-stranded gaps) and *ligation* (to remove nicks)

Then input and output look the same under sequencing
Sequenceable Join gate

A 2-input join gate, $\text{join}(a,b)$:

if $a$, $b$ are present together, then after full activation:

an “$abqr+q$” read (after ligation) reveals there was activation of $\text{join}(a,b)$, hence both $a$ and $b$ occurred. Otherwise, we would read “$abq+qr$”.

Two-domain gate architecture [L.Cardelli 2013] based on double stranded DNA (no secondary structure) hence gates can be sequenced by standard means
Join Gate activation steps

Sequence the soup: an "abqr" read indicates that both "a" and "b" were present.
What we can use

- Technologies to write (synthesize) whole sets of DNA strands in parallel

- Technologies to read (sequence) whole sets of DNA strands in parallel

- An architecture to do computation on DNA strands and produce sequenceable results

- Hence ... highly concurrent algorithms!
Coincidence Recorder

Goal: determine which pairs of a set of events were present together in the pot.

Algorithm:
At the beginning, add all the pairs join(x,y) for x,y in Events.
At the end, sequence the whole pot.
End.

N.B. join(x,x) tells us if x was ever present.

N^2 algorithm: great, we make “good use” of high-throughput synthesis and sequencing!
It uses no power when events are not present (it does not record timing, only coincidence).
Choice gate Specification

A choice gate is a two-input gate denoted $a?b$ between input events $a$ and $b$. As an abstract operator it is symmetric: $a?b = b?a$. Its desired behavior is as follows:

- If $a$ arrives no later than $b$, then $a?b$ produces a distinct result that we indicate $a \leq b$ or equivalently $b \geq a$.
- If $b$ arrives no later than $a$, then $a?b$ produces a distinct result that we indicate $b \leq a$ or equivalently $a \geq b$.
- If $a$ and $b$ arrive together, then $a?b$ produces a result that we indicate $a \sim b$ or equivalently $b \sim a$. (This is in practice an equal mixture of $a \leq b$ and $b \leq a$, or an unequal mixture if they arrive slightly offset.)
- As a special case, if $a$ ever arrives, then $a?a$ produces a result $a \sim a$.

That is, we want to implement the CRN:

"$a?b$" + a -> a + "a≤b"
"a?b" + b -> b + "b≤a"  

with "$a?b$" = "b?a" (identical DNA structures)

And we want the output to be readable by sequencing
Sequenceable Choice gate


(also clonable)
Sequenceable Choice gate outcomes

Sequence pattern:

If $b$ arrives first:
- If $a \geq b$:
  - $b \leq a$: $pabqr + spbaq$
- If $b \geq a$:
  - $a \leq b$: $pbaqr + spabq$

If $a$ arrives first:
- $a \geq b$:
  - $b \leq a$: $pabqr + spbaq$
- $b \geq a$:
  - $a \leq b$: $pbaqr + spabq$
Preorder Recorder

**Goal:** Record the preorder of first arrivals of a set of events that occur in a pot.

**Algorithm:**
At the beginning, add all the pairs \(x?y\), for \(x,y\) in Events.
At the end, sequence the whole pot and reconstruct the preorder by transitive reduction.

End.

E.g.: Events = \{a,b,c\}

<table>
<thead>
<tr>
<th>gates</th>
<th>structures</th>
<th>after ‘(-c)’</th>
<th>after ‘(-b)’</th>
</tr>
</thead>
<tbody>
<tr>
<td>a?a</td>
<td>[a?a]</td>
<td>[a?a]</td>
<td>[a?a]</td>
</tr>
<tr>
<td>b?b</td>
<td>[b?b]</td>
<td>[b?b]</td>
<td>(b \geq b)</td>
</tr>
<tr>
<td>c?c</td>
<td>[c?c]</td>
<td>(c \geq c)</td>
<td>(c \geq c)</td>
</tr>
<tr>
<td>a?b</td>
<td>[a?b]</td>
<td>[a?b]</td>
<td>(a \geq b)</td>
</tr>
<tr>
<td>a?c</td>
<td>[a?c]</td>
<td>(a \geq c)</td>
<td>(a \geq c)</td>
</tr>
<tr>
<td>b?c</td>
<td>[b?c]</td>
<td>(b \geq c)</td>
<td>(b \geq c)</td>
</tr>
</tbody>
</table>

That’s a definite \(c < b\), because we observe \(c \leq b\) but not \(b \leq c\). Moreover, we do not observe \(a \leq a\) which means that \(a\) never arrived. If we were to observe \(c \leq b\) and \(b \leq c\), then we would deduce that \(c,b\) arrived together, up to our time resolution.
Cas9 Join gate

- The target DNA is (completely) cut only if both gRNAs are present at the same time (DNA repair will otherwise fix single nicks):

- This scheme is sufficient to run a coincidence recorder, assuming single nicks are being repaired.
Cas9 Choice gate

The two cuts are distinguishable by sequencing.

N.B. the events must be in the form of gRNA, so they can be picked up by Cas9.
Conclusions

- Technological advances
  - High-throughput synthesis and sequencing

- Provide new readout opportunities
  - Reading and writing $n^2$ elements feasibly

- Which can inspire a new class of parallel algorithms
  - Coincidence Recorder, Preorder Recorder, ...