Design of Digital Logic by Genetic Regulatory Networks

Ron Weiss
Department of Electrical Engineering
Princeton University

Computing Beyond Silicon Summer School, Caltech, Summer 2002

Programming Cell Communities

Program cells to perform various tasks using:
• Intra-cellular circuits
  – Digital & analog components
• Inter-cellular communication
  – Control outgoing signals, process incoming signals
Programmed Cell Applications

- Biomedical
  - combinatorial gene regulation with few inputs; tissue engineering
- Environmental sensing and effecting
  - recognize and respond to complex environmental conditions
- Engineered crops
  - toggle switches control expression of growth hormones, pesticides
- Cellular-scale fabrication
  - cellular robots that manufacture complex scaffolds

Outline

- *In-vivo* logic circuits
- Intercellular communications
- Signal processing and analog circuits
- Programming cell aggregates
A Genetic Circuit Building Block

Logic Circuits based on Inverters

- Proteins are the wires/signals
- Promoter + decay implement the gates
- NAND gate is a universal logic element:
  - any (finite) digital circuit can be built!
Why Digital?

• We know how to program with it
  – Signal restoration + modularity = robust complex
    circuits

• Cells do it
  – Phage ? cI repressor: Lysis or Lysogeny?
    [Ptashne, A Genetic Switch, 1992]
  – Circuit simulation of phage ?
    [McAdams & Shapiro, Science, 1995]

• Also working on combining analog &
  digital circuitry

BioCircuit Computer-Aided Design

• BioSPICE: a prototype biocircuit CAD tool
  – simulates protein and chemical concentrations
  – intracellular circuits, intercellular communication
  – single cells, small cell aggregates
Genetic Circuit Elements

Modeling a Biochemical Inverter

\[
\begin{align*}
mRNA_A + rRNA & \xrightarrow{k_{translate}} mRNA_A + rRNA + A \quad (1) \\
mRNA_A & \xrightarrow{k_{decay}} {mRNA_A} \quad (2) \\
A + A & \xrightarrow{k_{dimerization}} A_2 \quad (3) \\
A_2 & \xrightarrow{k_{single}} A + A \quad (4) \\
A & \xrightarrow{k_{decay}} A_2 \quad (5) \\
A_2 & \xrightarrow{k_{decay}} A_2 \quad (6) \\
P_Z + A_2 & \xrightarrow{k_{rep}]A_2} P_Z A_2 \quad (7) \\
P_Z A_2 & \xrightarrow{k_{dissolution}} P_Z + A_2 \quad (8) \\
P_Z A_2 & \xrightarrow{k_{dissolution}} P_Z A_2 \quad (9) \\
P_Z + RNA & \xrightarrow{k_{transcribe}} P_Z + RNA + mRNA_2 \quad (13) \\
mRNA_2 & \xrightarrow{k_{decay}} {mRNA_2} \quad (14) \\
\end{align*}
\]
A BioSPICE Inverter Simulation

![Graph showing dynamic behavior of inverter](image)

- Input
- Repressor
- Promoter
- Output

“Proof of Concept” Circuits

- Work in BioSPICE simulations [Weiss, Homsy, Nagpal, 1998]

  **RS-Latch (“flip-flop”)**

  ![RS-Latch diagram](image)

  - They work in vivo
    - Flip-flop [Gardner & Collins, 2000]
    - Ring oscillator [Elowitz & Leibler, 2000]

- Ring oscillator

  ![Ring oscillator diagram](image)

  - However, cells are very complex environments
    - Current modeling techniques poorly predict behavior
The IMPLIES Gate

- Inducers that inactivate repressors:
  - IPTG (Isopropylthio-β-galactoside) $\rightarrow$ Lac repressor
  - aTc (Anhydrotetracycline) $\rightarrow$ Tet repressor
- Use as a logical Implies gate: \[ (\text{NOT } R) \text{ OR } I \]

<table>
<thead>
<tr>
<th>Repressor</th>
<th>Inducer</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

The Toggle Switch

[Gardner & Collins, 2000]

pIKE = lac/tet
pTAK = lac/cIIts
Actual Behavior of Toggle Switch

[Gardner & Collins, 2000]

The Ring Oscillator

[Elowitz, Leibler 2000]
Example of Oscillation

Evaluation of the Ring Oscillator

Reliable long-term oscillation doesn’t work yet:
- Will matching gates help?
- Need to better understand noise
- Need better models for circuit design

[Elowitz & Leibler, 2000]
A Ring Oscillator with Mismatched Inverters

A = original cl/?PRO
B = repressor binding 3X weaker
C = transcription 2X stronger

Device Physics in Steady State

“Ideal” inverter

- Curve can be achieved with certain dna-binding proteins
- Inverters with these properties can be used to build complex circuits

Transfer curve:
- gain (flat, steep, flat)
- adequate noise margins
Measuring a Transfer Curve

• Construct a circuit that allows:
  – Control and observation of input protein levels
  – Simultaneous observation of resulting output levels

• Also, need to normalize CFP vs YFP

Flow Cytometry (FACS)
Drive Input Levels by Varying Inducer

Controlling Input Levels

Also use for CFP/YFP calibration
Cell Population Behavior

Red = pPROLAR
Rest = pINV-102 with IPTG (0.1 to 1000 uM)

CFP: a Weak Fluorescent Protein

Induction of CFP expression
Measuring a Transfer Curve
for lacI/p(lac)

Transfer Curve Data Points

0¢1
undefined
1¢0

1 ng/ml aTc
10 ng/ml aTc
100 ng/ml aTc
**lacI/p(lac) Transfer Curve**

![Diagram of the lacI/p(lac) system with aTc and YFP regulation](image)

- **Input (Normalized CFP):**
- **Output (YFP):**

![Graph of the lacI/p(lac) transfer curve](image)

- **Evaluating the Transfer Curve**

  - **Gain / Signal restoration:**
  - **Noise margins:**

![Histograms illustrating gain/signal restoration](image)

*note: graphing vs. aTc (i.e. transfer curve of 2 gates)*
Transfer Curve of *Implies*

The Cellular Gate Library

Add the cI/λ<sub>P(R)</sub> Inverter

- *cI* is a highly efficient repressor

- Use lacI/p(lac) as driver
Initial Transfer Curve for $cI/\lambda_{P(R)}$

Recall Inverter Components
Functional Composition of an Inverter

\[ \psi \circ \phi = \text{digital inversion} \]

\[ \psi = \text{output mRNA} \]
\[ \phi = \text{input protein} \]
\[ \rho = \text{bound operators} \]
\[ \psi_i = \text{input mRNA} \]

Genetic Process Engineering I: Reducing Ribosome Binding Site Efficiency

RBS

Orig: ATTAAAGAGGAGAAATTAAGCATG
RBS-1: TCACACAGGAAACCGGTTCGATG
RBS-2: TCACACAGGAAAGGCCTCGATG
RBS-3: TCACACAGGACGGCCGGATG
Experimental Results for cI/\lambda_p(R) Inverter with Modified RBS

Genetic Process Engineering II:
Mutating the \lambda_p(R) operator

\begin{align*}
\text{Orig:} & \quad \text{TACCTGTGCCGTTGATA} \\
\text{mut4:} & \quad \text{TACATGTGGCGGTGATA} \\
\text{mut5:} & \quad \text{TACATGTGGCGGTGATA} \\
\text{mut6:} & \quad \text{TACAGATGGCGGTGATA}
\end{align*}
Experimental Results for Mutating $\lambda_{P(R)}$

<table>
<thead>
<tr>
<th>IPTG (uM)</th>
<th>Output (YFP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>1,000.00</td>
</tr>
<tr>
<td>1.0</td>
<td>100.00</td>
</tr>
<tr>
<td>10.0</td>
<td>10.00</td>
</tr>
<tr>
<td>100.0</td>
<td>1.00</td>
</tr>
<tr>
<td>1,000.0</td>
<td>0.10</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IPTG (uM)</th>
<th>Output (YFP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>1,000.00</td>
</tr>
<tr>
<td>1.0</td>
<td>100.00</td>
</tr>
<tr>
<td>10.0</td>
<td>10.00</td>
</tr>
<tr>
<td>100.0</td>
<td>1.00</td>
</tr>
<tr>
<td>1,000.0</td>
<td>0.10</td>
</tr>
</tbody>
</table>

Genetic Process Engineering

- Genetic modifications required to make circuit work
- Need to understand “device physics” of gates
  - enables construction of complex circuits
Self-perfecting Genetic Circuits
[Arnold, Yokobayashi, Weiss]

- Use directed evolution to optimize circuits
- Screening criteria based on transfer curve
- Initial results are promising

Lab-on-a-chip: µFACS [Quake]

Molecular Evolution of the Circuit

![Graphs showing YFP fluorescence vs. IPTG concentration for different sequences.](image)
Prediction of Circuit Behavior

Can the behavior of a complex circuit be predicted using only the behavior of its parts?

Input signal
Output signal

Prediction of Circuit Behavior
preliminary results