Towards a High-Level Programming Language for Standardizing and Automating Biology Protocols

Vaishnavi Ananthanarayanan and William Thies
Microsoft Research India

First International Workshop on Bio-Design Automation
San Francisco, CA
July 27, 2009
Genetic Control of Surface Curvature

Utpal Nath, Brian C. W. Crawford, Rosemary Carpenter, Enrico Coen*
Genetic Control of Surface Curvature

Utpal Nath, Brian C. W. Crawford, Rosemary Carpenter, Enrico Coen*

Material and Methods

In situ Hybridization. The methods used for tissue preparation, digoxigenin-labelling of RNA probes, and in situ hybridisation were as described previously \((S13)\). The probe used

**floricaula**: A Homeotic Gene Required for Flower Development in *Antirrhinum majus*

Enrico S. Coen, José M. Romero,* Sandra Doyle, Robert Elliott, George Murphy, and Rosemary Carpenter

“Immunological detection ... was carried out as described in the Boehringer digoxigenin-nucleic acid detection kit with some modifications.”
Genetic Control of Surface Curvature

Utpal Nath, Brian C. W. Crawford, Rosemary Carpenter, Enrico Coen*

RNA probes, and in situ hybridisation were as described previously (S13). The probe used to detect the CIN transcript was a 1048 bp fragment from the cDNA clone, covering the entire ORF. For H4, the probe consisted of the entire cDNA (S14). For CYCLIN D3b, a 3'-
Immunological detection... was carried out as described in the Boehringer digoxigenin-nucleic acid detection kit with some modifications.

to detect the CIN transcript was a 1048 bp fragment from the cDNA clone, covering the entire ORF. For H4, the probe consisted of the entire cDNA (S14). For CYCLIN D3b, a 3'-terminal fragment of the cDNA lacking the poly-A tail was used (S15).
Problems with Existing Descriptions of Protocols

- **Incomplete**
  - Cascading references several levels deep
  - Some information missing completely

- **Ambiguous**
  - One word can refer to many things
  - E.g., “inoculate” a culture

- **Non-uniform**
  - Different words can refer to the same thing
  - E.g., “harvest”, “pellet down”, “centrifuge” are equivalent

- **Not suitable for automation or for programming standard biological parts**
Towards a High-Level Programming Language for Biology Protocols

Goal: in scientific publications, replace textual description of methods used with code

1. Enable automation via microfluidic chips
2. Improve reproducibility of manual experiments
Contributions to Date

• **Microfluidics:** first manipulation of discrete samples using soft-lithography [LabChip’06]

• **Programming:** first mapping of single ISA across different chips [DNA’06, NatCo’07]

• **Optimization:** first efficient algorithm for complex mixing on chip [DNA’06, NatCo’07]

• **Computer Aided Design:** first tool that routes channels, generates GUI [MIT’09]

• **Work in Progress:** programming language for expressing and automating broad class of experiments
The BioStream Language

- **BioStream is a protocol language for reuse & automation**
  - Portable
  - Volume-independent

- **Initial focus: molecular biology**
  - Mixing
  - Cell culture
  - Electrophoresis
  - Heating / cooling
  - Centrifugation
  - Timing constraints

- **Implemented as a C library**
  - Used to express 15 protocols
  - Initial backend: emit readable instructions for human

- **Validation in progress**
  - Intern at Indian Institute of Science
  - Would represent first biology experiment grounded in architecture-independent programmed description
Language Primitives

- **Declaration / measurement / disposal**
  - declare_fluid
  - declare_column
  - measure_sample
  - measure_fluid
  - volume
  - discard
  - transfer
  - transfer_column
  - declare_tissue

- **Combination / mixing**
  - combine
  - mix
  - combine_and_mix
  - addto_column
  - mixing_table

- **Centrifugation**
  - centrifuge_pellet
  - centrifuge_phases
  - centrifuge_column

- **Temperature**
  - set_temp
  - use_or_store
  - autoclave

- **Timing**
  - wait
  - time_constraint
  - store_until
  - inoculation
  - invert_dry

- **Detection**
  - ce_detect
  - gas_chromatography
  - nanodrop
  - electrophoresis
  - mount_observe_slide
  - sequencing
Example: Plasmid DNA Extraction

I. Original protocol (Source: Klavins Lab)

Add 100 ul of 7X Lysis Buffer (Blue) and mix by inverting the tube 4-6 times. Proceed to step 3 within 2 minutes.

II. BioStream code

FluidSample f1 = measure_and_add(&f0, &lysis_buffer, 100*uL);
FluidSample f2 = mix(&f1, INVERT, 4, 6);
time_constraint(&f1, 2*MINUTES, next_step);

III. Auto-generated text output

Add 100 ul of 7X Lysis Buffer (Blue).
Invert the tube 4-6 times.
NOTE: Proceed to the next step within 2 mins.
Example: Plasmid DNA Extraction

DNA Miniprep Protocol

Solutions/reagents:
- bacterial culture grown in LB medium
- 7X Lysis Buffer (Blue)
- Neutralization Buffer (Yellow)
- Endo-Wash Buffer
- Zippy™ Wash Buffer
- Zippy™ Elution Buffer
- Zymo-Spin™ II Column

Equipment:
- Centrifuge
- Microfuge

Steps:
1. Measure out 600 µl of bacterial culture grown in LB medium into a 1.5ml reaction tube.

2. Add 100 µl of 7X Lysis Buffer (Blue).
   Invert the tube 4-6 times.
   NOTE: Proceed to the next step within 2 mins.

3. Add 350 µl of Neutralization Buffer (Yellow).
   Vortex the mixture for a few secs.
1. Standardizing Ad-Hoc Language

- Need to convert qualitative words to quantitative scale

- **Example: a common scale for mixing**
  - When a protocol says “mix”, it could mean many things
  - Level 1: tap
  - Level 2: stir
  - Level 3: invert
  - Level 4: vortex / resuspend / dissolve
2. Separating Instructions from Hints

• How to translate abstract directions?
  – “Remove the medium by aspiration, *leaving the bacterial pellet as dry as possible.*”

Centrifuge(&medium, ...); Aspirate and remove medium. hint(pellet_dry) \[→\] Leave the pellet as dry as possible.

• Separating instructions and hints keeps language tractable
  – Small number of precise instructions
  – Extensible set of hints
3. Generating Readable Instructions

• In typical programming languages- minimal set of orthogonal primitives

• But can detract from readability

  Original: “Mix the sample with 1uL restriction enzyme.”

  BioStream with orthogonal primitives:

  ```
  FluidSample s1 = measure(&restriction_enzyme, 1*uL);
  FluidSample s2 = combine(&sample, &s1);
  mix(s2, tap);
  ```

  Measure out 1ul of restriction enzyme.
  Combine the sample with the restriction enzyme.
  Mix the combined sample by tapping the tube.
3. Generating Readable Instructions

• In typical programming languages- minimal set of orthogonal primitives

• But can detract from readability
  
  **Original:** “Mix the sample with 1uL restriction enzyme.”
  
  **BioStream with compound primitives:**

  ```
  combine_and_mix(&restriction_enzyme, 1*uL, 
  &sample, tap);
  ```

  Add 1uL restriction enzyme and mix by tapping the tube.

✔ Define a standard library that combines primitive operations
3. Generating Readable Instructions

mixing_table_pcr(7,20, array_pcr, initial_conc, final_conc, vol);

1. Set up a reaction as follows on ice:

<table>
<thead>
<tr>
<th></th>
<th>Initial concentration</th>
<th>Final concentration</th>
<th>Final volume per 20 µl reaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Taq buffer</td>
<td>10X</td>
<td>1X</td>
<td>2 µl</td>
</tr>
<tr>
<td>dNTPs</td>
<td>10 mM</td>
<td>0.5 mM</td>
<td>1 µl</td>
</tr>
<tr>
<td>Primers</td>
<td>10 µM</td>
<td>1 µM</td>
<td>2 µl</td>
</tr>
<tr>
<td>Taq polymerase</td>
<td>5 U µl⁻¹</td>
<td>2 U</td>
<td>8 µl</td>
</tr>
<tr>
<td>Genomic DNA</td>
<td>--</td>
<td>100 ng</td>
<td>X</td>
</tr>
<tr>
<td>sterile distilled water</td>
<td>--</td>
<td>--</td>
<td>Make up volume to 20 µl</td>
</tr>
</tbody>
</table>
# Benchmark Suite

<table>
<thead>
<tr>
<th>Name</th>
<th>Source</th>
<th>Lines of Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alkaline DNA Miniprep (Animal)</td>
<td>Textbook</td>
<td>114</td>
</tr>
<tr>
<td>AllPrep RNA/Protein (Animal)</td>
<td>Qiagen kit</td>
<td>180</td>
</tr>
<tr>
<td>Immunolocalization</td>
<td>Lab notes</td>
<td>127</td>
</tr>
<tr>
<td>DNA Sequencing</td>
<td>Published paper</td>
<td>162</td>
</tr>
<tr>
<td>Molecular barcodes methods</td>
<td>Published paper</td>
<td>267</td>
</tr>
<tr>
<td>SIRT1 Redistribution</td>
<td>Published paper</td>
<td>220</td>
</tr>
<tr>
<td>Splinkerette PCR</td>
<td>Published paper</td>
<td>248</td>
</tr>
<tr>
<td>Touchdown PCR</td>
<td>Published paper</td>
<td>65</td>
</tr>
<tr>
<td>Transcriptional instability</td>
<td>Published paper</td>
<td>187</td>
</tr>
<tr>
<td>DNA Miniprep (Bacterial)</td>
<td>Class notes</td>
<td>102</td>
</tr>
<tr>
<td>Restriction enzyme digestion</td>
<td>Class notes</td>
<td>55</td>
</tr>
<tr>
<td>Restriction enzyme ligation</td>
<td>Class notes</td>
<td>67</td>
</tr>
<tr>
<td>DNA Extraction (Plant)</td>
<td>Lab notes</td>
<td>481</td>
</tr>
<tr>
<td>Plant RNA isolation</td>
<td>Lab notes</td>
<td>137</td>
</tr>
<tr>
<td>Plasmid purification</td>
<td>Qiagen kit</td>
<td>158</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td><strong>2570</strong></td>
</tr>
</tbody>
</table>
Example: PCR

repeat thermocycling
Example: Molecular Barcodes

Preparation

+ PCR (2)
Example: DNA Sequencing

Preparation

PCR
PCR
PCR
PCR

Analysis
3. Add 1.5 vol. CTAB to each MCT and vortex. Incubate at 65°C for 10-30 mins


5. Centrifuge at 13000g at room temperature for 5 mins

6. Transfer aqueous (upper) layer to clean MCT and repeat the extraction using chloroform: Isoamylalcohol: 96:4
3. Add 1.5 vol. CTAB to each MCT and vortex. Incubate at 65° C for 10-30 mins
5. Centrifuge at 13000g at room temperature for 5 mins
6. Transfer aqueous (upper) layer to clean MCT and repeat the extraction using chloroform: Isoamylalcohol: 96:4

Coding protocols in precise language removes ambiguity and enables consistency checking
Validating the Language

- **Eventual validation: automatic execution**
  - But BioStream more capable than most chips today
  - Need to decouple language research from microfluidics research
  - Also validate in a synthetic biology context

- **Initial validation: human execution**
  - In collaboration with Prof. Utpal Nath’s lab at IISc
  - Target Plant DNA Isolation, common task for summer intern

Biologist is never exposed to original lab notes

- To the best of our knowledge, first execution of a real biology protocol from a portable programming language
Future Work

• Adapt the language to biologists
  – *Currently looking for collaborators to use the language!*
  – Focus on ‘natural language’ authoring rather than programming
  – Share language and protocols on a public wiki

• Backends for BioStream
  – Generate graphical protocol
  – Program a part of/ complete synthetic biological system to perform a given protocol/function

• Automatic scheduling
  – Schedule separate protocols onto shared hardware, maximizing utilization of shared resource (e.g., thermocycler)
Related Work

• **EXACT**: EXperimental ACTions ontology as a formal representation for biology protocols [Soldatova et al., 2009]

• **Aquacore**: ISA and architecture for programmable microfluidics, builds on our prior work [Amin et al., 2007]

• **Robot Scientist**: functional genomics driven by macroscopic laboratory automation [King et al., 2004]

• **PoBol**: RDF-based data exchange standard for BioBricks
Conclusions

• A high-level programming language for biology protocols is tractable and useful
  – Improves readability
  – Enables automation

• Vision: a defacto language for experimental science
  – Replace ad-hoc language with precise, reusable description
  – Download a colleague’s code, automatically map to your microfluidic chip or lab setup

• Seeking users and collaborators!
  1. Send us your protocols
  2. We code them in BioStream
  3. You inspect standardized protocol, optionally validate it in lab
Acknowledgements

• Dr. Utpal Nath, Indian Institute of Science
• Mansi Gupta, Subhashini Muralidharan, Sushmita Swaminathan, Indian Institute of Science
• Dr. Eric Klavins, University of Washington