Spatial computing: a unifying approach to computational materials

Jacob Beal

Royal Society Meeting on Heterotic Computing
November, 2013
Computing runs across physical space-time

Emerging Computational Substates

Space-Time Programming Models

device
neighborhood

Multi-Substrate Computation
Outline

- From Monolithic to Spatial Computing
- Amorphous Medium & Field Calculus
- Biological / Hybrid Computational Substrate
Traditional Monolithic Computing

The venerable von Neumann model is breaking down in several ways…
The End of Moore’s Law

High-performance computing = mesh
Everything is a wireless computer
New Computational Materials

• Synthetic Biology:

Other emerging areas too, including nanoassembly, active materials...
Fundamentally different models

How can we program aggregates adaptively & efficiently?
Can mixed systems exploit platform complementarity?
Spatial Computers

Robot Swarms

Biological Computing

Sensor Networks

Reconfigurable Computing

Cells during Morphogenesis

Modular Robotics
• A spatial computer is a collection of computational devices distributed through a physical space in which:
  – the difficulty of moving information between any two devices is strongly dependent on the distance between them, and
  – the “functional goals” of the system are generally defined in terms of the system's spatial structure
Outline

• From Monolithic to Spatial Computing
• Amorphous Medium & Field Calculus
• Biological / Hybrid Computational Substrate
Amorphous Medium

- Continuous space & time
- Infinite number of devices
- See neighbors' past state

Approximate with:
- Discrete network of devices
- Signals transmit state

[Beal, ’04]
Field Calculus:

Pointwise

Feedback: rep

Restriction: if

Neighborhood: nbr

With appropriate pointwise measurements, operations are space-time universal

[Viroli et al., ’13]
Implementation: Proto

(def gradient (src) ...)
(def distance (src dst) ...)
(def dilate (src n)
  (<= (gradient src) n))
(def channel (src dst width)
  (let* ((d (distance src dst))
    (trail (<= (+ (gradient src)
      (gradient dst))
      d)))
    (dilate trail width)))
Heterogeneous Computing Materials

Functional “streams” integrate different time scales
Outline

• From Monolithic to Spatial Computing
• Amorphous Medium & Field Calculus
• Biological / Hybrid Computational Substrate
Vision: Precision Crop Management
Application: Protective Biofilms

- Continuously grown and shed outer layer
- Secure base layer tracks state, coordinates reactive emission of defensive chemicals
- Embedded electronic monitoring interface
Synthetic Biology: Transcriptional Logic

Decay

RNA polymerase

DNA promoter

regulatory protein

RNA

Stabilizes at decay = production

Protein

Signal = Concentration

Alternatives:
PoPS
RNA concentration

ribosome
Genetic Regulatory Networks

- Parallel dataflow computation
- Continuous time evolution, feedback loops

*Example: SR-Latch*

- Spatial patterning via intercellular signaling, adhesion, cell morphology, ...
(green (not (IPTG)))

[Beal, Lu & Weiss, ’11]
Optimization of Complex Designs

```
(def sr-latch (s r)
  (letf+ ((o boolean (not (or r o-bar)))
          (o-bar boolean (not (or s o)))))
  o))

(green (sr-latch (aTc) (IPTG)))
```

Final Optimized:
5 functional units
4 transcription factors

Unoptimized:
15 functional units, 13 transcription factors
TASBE Tool-Chain

A high-level program of a system that reacts depending on sensor output

```lisp
(def simple-sensor-actuator ()
  (let ((x (test-sensor)))
    (debug-1 x)
    (debug-2 (not x))))
```

Mammalian Target  E. coli Target

[Beal et al., 2012]
TASBE Tool-Chain

Program instantiated for two target platforms

Mammalian Target

E. coli Target

[Beal et al., 2012]
TASBE Tool-Chain

Abstract genetic regulatory networks

If detect explosives: emit signal
If signal > threshold: glow red

Mammalian Target
E. coli Target

[Beal et al., 2012]
TASBE Tool-Chain

Automated part selection using database of known part behaviors

If detect explosives:
emit signal
If signal > threshold:
glow red

Mammalian Target

E. coli Target

[Beal et al., 2012]
TASBE Tool-Chain

Automated assembly step selection for two different platform-specific assembly protocols

Mammalian Target  E. coli Target

[Beal et al., 2012]
TASBE Tool-Chain

Resulting cells demonstrating expected behavior

**Uninduced**

Mammalian Target

**Induced**

E. coli Target

If detect explosives:
emit signal
If signal > threshold:
glow red

[Beal et al., 2012]
Challenge: Synthetic Device Libraries

Can use a device only once/circuit → need lots of devices!

Zinc-Finger Proteins:

TALE Proteins:

micro RNAs:

CRISPR: CAS/gRNA:
Challenge: Predictable Composition

*Improved models & metrology → high-precision circuit prediction*

---

![Graph](image.png)

**Two-Repressor Cascade**

**Three-Replicon Mixture**

[Davidsohn et al., 2013]
Biological/Hybrid Substrates: where we stand

High-level specification of complex circuits

Precise circuit prediction

Biological / Hybrid Computing Substrate

Large, well-characterized device libraries

Electronics interface: optogenetic, electrical, magnetic, ...

Dox, rtTA, EBFP2, EYFP, Feast, Golf
Summary

- Major technological trends are all driving towards a world filled with spatial computers.
- Continuous space-time models allow effective adaptive aggregate programming.
- Mixed-material computation will enable a wide range of visionary applications.
- Rapid progress towards predictable, scalable computational control of biological organisms.
Acknowledgements:

Raytheon
BBN Technologies
Aaron Adler
Brett Benyo
Taylor Campbell
Joseph Loyall
Rick Schantz
Kyle Usbeck
Fusun Yaman

MIT
Ron Weiss
Jonathan Babb
Noah Davidsohn
Tasuku Kitada
Ting Lu

Boston University
Douglas Densmore
Swapnil Bhatia
Traci Haddock
Evan Appleton
Chenkai Liu
Viktor Vasilev
Tyler Wagner

Imperial College London
Mirko Viroli
Matteo Cascadei

Ferruccio Damiani

NSF
DARPA
Tools available online!

http://proto.bbn.com/

http://synbiotools.bbn.com