A Basis Set of Operators for Space-Time Computations

Jacob Beal
3rd Spatial Computing Workshop
SASO 2010
Problem: Analysis for Spatial Computing

- Model-to-model: comparison: MGS, Proto, TOTA, LDP, etc... equivalent? complete?
- Platform-to-platform comparison: can we prove algorithms in the continuous model instead?

*Continuous model = super-Turing?*
Talk Outline

- General definition of space-time computation
- Basis set of operators
- Is Proto universal?
Amorphous Medium

<table>
<thead>
<tr>
<th>Var.</th>
<th>Definition</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M$</td>
<td>Spatial region</td>
<td>compact Reimannian manifold</td>
</tr>
<tr>
<td>$T$</td>
<td>Time interval</td>
<td>$T \subseteq (-\infty, \infty)$</td>
</tr>
<tr>
<td>$d$</td>
<td>Distance fn on M</td>
<td>$d : M \times M \to \mathbb{R}$</td>
</tr>
<tr>
<td>$c$</td>
<td>Max speed of information</td>
<td>meters per second</td>
</tr>
<tr>
<td>$N(m)$</td>
<td>Neighborhoods on M</td>
<td>$N : M \to P(M)$</td>
</tr>
</tbody>
</table>
Computation as Function

- **Computed state:**
  - Instant: $S_t : M \rightarrow V$
  - Initial: $S_0 : M \rightarrow V$
  - Interval: $S_T : M \times T \rightarrow V$

- **Sensing:**
  - $E : M \times T \rightarrow V$

- **Computation:**
  - $C : M \times T \times E \times S_0 \rightarrow S_T$

---

<table>
<thead>
<tr>
<th>Var.</th>
<th>Definition</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M$</td>
<td>Spatial region</td>
<td>compact Reimannian manifold</td>
</tr>
<tr>
<td>$T$</td>
<td>Time interval</td>
<td>$T \subseteq (-\infty, \infty)$</td>
</tr>
<tr>
<td>$d$</td>
<td>Distance fn on $M$</td>
<td>$d : M \times M \rightarrow \mathbb{R}$</td>
</tr>
<tr>
<td>$c$</td>
<td>Max speed of information</td>
<td>meters per second</td>
</tr>
<tr>
<td>$N(m)$</td>
<td>Neighborhoods on $M$</td>
<td>$N : M \rightarrow P(M)$</td>
</tr>
<tr>
<td>$V$</td>
<td>Function values</td>
<td>$\bigcup_{k \geq 0} \mathbb{R}^k$</td>
</tr>
<tr>
<td>$S_t$</td>
<td>State at time $t$</td>
<td>$S_t : M \rightarrow V$</td>
</tr>
<tr>
<td>$S_0$</td>
<td>Initial state</td>
<td>$S_0 : M \rightarrow V$</td>
</tr>
<tr>
<td>$S_T$</td>
<td>State on interval $T$</td>
<td>$S_T : M \times T \rightarrow V$</td>
</tr>
<tr>
<td>$E$</td>
<td>Environmental state</td>
<td>$E : M \times T \rightarrow V$</td>
</tr>
<tr>
<td>$C$</td>
<td>Computation</td>
<td>$C : M \times T \times E \times S_0 \rightarrow S_T$</td>
</tr>
</tbody>
</table>
Space-Time Universality

• Definition: a computation $C'$ \textbf{implements} computation $C$ if there is a restriction of $S_T$ that is equal to $S_t$ almost everywhere, and if for any non-equal point $p$, there is a sequence of points $p_i$ converging on $p$ such that 
\[
\lim_{i \to \infty} S'_T(p_i) = \lim_{i \to \infty} S_T(p_i).
\]

• A basis set of operators $B$ is \textbf{space-time universal} if, for any computation $C$ that can be specified by some basis set of operators (we need not know what or how), it is possible to implement an equivalent computation $C'$ using operators in $B$.

\textit{Note: definition of universality not dependent on a model.}
A computation $C$ is causal if at every point $(m,t)$, the value depends only on the past light cone.

A computation is finitely-approximable if all countable sequences of $\varepsilon_i$-approximations $C_i$ with decreasing $\varepsilon_i$ converge to an implementation of $C$. 
Examples of Finitely-Approximable Causal Computation

Elapsed time since environmental cue

Distance to nearest environmental cue

Is an environmental cue currently present?
Talk Outline

- General definition of space-time computation
- **Basis set of operators**
- Is Proto universal?
Basis Set of Operators

- Pointwise Turing-universal: $P$
- Metric: $n_d, g$
- Neighborhood: $n_v, n_r, n_m$

metric tensor

$g$

$n_d$

$n_v$

$n_r$

$n_m$
Universality of Basis

Theorem: any finitely-approximable causal computation $C$ can be implemented using the basis set of operators $\{g, n_d, n_v, n_r, n_m\} \cup P$.

Intuition:

- Use $n_*$ to sample past state, environment, $g$
- Use $P$ to compute approximate value
- Increasing sampling resolution converges
Talk Outline

- General definition of space-time computation
- Basis set of operators
- Is Proto universal?
Proto: Computing with fields

gradient -> source
gradient -> destination

width

+ 10

<= 37

dilate
(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)))
Application to Proto

Most operators are directly implemented:

- $P$ implemented by Proto's point-wise operators
- $n_d = \text{nbr-vec}$
- $n_v = \text{nbr}$
- $n_r = \text{if}$ applied to field types
- $n_m = \text{min-hood}$

Missing: $g$ ... but Proto has other metric ops, e.g. $\text{density, nbr-lag}$ ... partial gap cover?
Open Problems

- What are appropriate computational cost models, and what finitely-approximable operators minimize cost?
- How can we do “Nyquist rate” approximation analysis?
- Can we establish function approximability bounds?
- What families of continuous proofs can be automatically translated to discrete proofs?
- Extension of theory to dynamic manifolds?
- How can Proto be extended to cover \( g \)?
- How powerful are other spatial computing models?
Contributions

- Direct-proof motivation for super-Turing models
- Operator-free definitions for space-time computation
- Basis operators for finitely-approximable causal computations: \( \{g, n_d, n_v, n_r, n_m\} \cup P \)
- Gap analysis for universality of Proto