Continuous Semantics of Proto

Jake Beal
January, 2006
(Joint work w. Jonathan Bachrach)
Outline

- Amorphous Medium
- Primitives, Abstraction, & Composition
- Managing space and time
  - Operations with spatial extent
  - Conditionals
  - State
  - Error handling
- Putting it all together
Many network problems are spatial

- Exa: sensor/actuator networks, smart materials, cooperative robotics, biofilms ...
Many network problems are spatial

Approx. space w. network like this...
Many network problems are spatial

Approx. space w. network like this...this...
Many network problems are spatial

Approx. space w. network like this...this...or this?

*We shouldn't have to care!*
Amorphous Medium

- Compact manifold with a device at every point
  - Lagged internal state visible to neighbors

*Ph34r th3 Unc0nt4bility!*
Fields

F: X → value
Primitives & Evaluation

2
Primitives & Evaluation
Operators

- implicit input
- outputs (ordered)

- inputs (ordered)
- implicit output

```
op
```
Operators

1.2

2.2

floor

1

2

0.2
Operators

:red

0.9

act

environment

:light

0.9

sense

0

1
Composition

\[(\text{act} : \text{red} (\text{sense} : \text{light}))\]
Composition

$$(\text{act : red} \ (\text{sense : light}))$$
Abstraction: $\lambda$

$\lambda: (\lambda (x \ y) (\sqrt{+(*x\ x)(*y\ y)}))$
Abstraction: $\lambda$

\[
(\lambda (x \ y) \ (\text{sqrt} \ (+ \ (* \ x \ x) \ (* \ y \ y))))
\]
Abstraction: def

(\texttt{def} foo (x y)
 (sqrt (+ (* x x) (* y y)))))
Abstraction: let

(let ((x (/ 4 f)) (y (+ f 1))) (* x y))
Abstraction: let

(let ((x (/ 4 f)) (y (+ f 1))) (* x y))
Operations with Spatial Extent

Implicit communication in reductions over nbr vals

*Ph34r th3 Unc0nt4bility!*
`nbrval` gathers neighbor values

`(nbrval f) → field of fields of nbr values`
Quantifiers summarize nbr values

Available Quantifiers: \texttt{limsup, liminf, integral, forall, exists}
reduce-nbrs encapsulates both

(reduce-nbrs (+ f 3) (liminf nbrval))
Other spatial operations

- nbr-dist
- nbr-lag
- random
Simple Conditional: \texttt{if}

$$(\text{if } f \, 2 \, 7)$$
Simple Conditional: if

\((\text{if } f \ 2 \ 7)\)
The problem with \texttt{if}

\[(\text{if } f \ (\text{reduce-nbrs } f \ \text{forall}) \ #T)\]
The problem with if

\[(\text{if } f (\text{reduce-nbrs } f \text{ forall}) \#T)\]
restrict

#T

#F

6
Complex Conditional: \texttt{where}

(\texttt{where } f \texttt{ (reduce-nbrs } f \texttt{ forall) } \#T)
Complex Conditional: where

(\text{where } f \ (\text{reduce-nbrs } f \ \text{forall}) \ #T)
State: delay

delay:

Time finally appears!
State: **delay**

delay:

Time finally appears!
State: delay

Time finally appears!
State: delay

delay:

Time finally appears!
State: \texttt{letfed}

\begin{equation}
\texttt{letfed:} \begin{array}{c}
f \\
0 \\
+ \\
delay \\
\lambda:
\end{array}
\end{equation}

(\texttt{letfed ((i 0 (+ i f)) i))}
State: letf fed

(letf fed ((i 0 (+ i f))) i)
What happens when an error is localized?

*Conditions are values, not flow control.*
Putting it all together: \texttt{gradient}

\begin{verbatim}
(def gradient (src)
  (letfed ((n ∞ (if src 0
                 (+ (reduce-nbrs n liminf)
                    (reduce-nbrs nbr-dist limsup))))
             n)))
\end{verbatim}