Towards an atomic $\lambda\mu$-calculus
YR-ICALP 2015

Fanny He
f.he@bath.ac.uk

5 July 2015
Sharing, laziness and atomicity

The $\lambda\mu$-calculus: classical logic and continuations

An atomic $\Lambda\mu S$-calculus
Sharing subexpressions

\[
\text{fibonacci } n \mid (n == 0) = 0 \\
\mid (n == 1) = 1 \\
\mid (n > 1) = \text{fibonacci} (n-1) + \text{fibonacci} (n-2)
\]

\[
\text{fibonacci2 } n = \text{fib } 1 0 n \\
\text{where} \\
\text{fib } n1 n2 n \mid (n == 0) = n2 \\
\mid (n == 1) = n1 \\
\mid (n > 1) = \text{fib} (n1+n2) n1 (n-1)
\]
Sharing subexpressions

Exponential:

\[
\text{fibonacci } n \mid (n == 0) = 0 \\
\mid (n == 1) = 1 \\
\mid (n > 1) = \text{fibonacci } (n-1) + \text{fibonacci } (n-2)
\]
Sharing subexpressions

Linear:

\[
\text{fibonacci2 } n = \text{fib } 1 \ 0 \ n
\]

\[
\begin{aligned}
&\text{where} \\
&\text{fib } n1 \ n2 \ n \ | \ (n == 0) = n2 \\
&\quad | \ (n == 1) = n1 \\
&\quad | \ (n > 1) = \text{fib} (n1+n2) \ n1 \ (n-1)
\end{aligned}
\]

\[
\begin{array}{c}
\downarrow \\
(n-1) \\
\downarrow \\
(n-2)
\end{array}
\]

\[
\begin{array}{c}
\downarrow \\
n1 \\
\downarrow \\
n2
\end{array}
\]

\[
\begin{array}{c}
\downarrow \\
n1 \\
\downarrow \\
n2
\end{array}
\]

\[
\begin{array}{c}
\downarrow \\
n1 \\
\downarrow \\
n2
\end{array}
\]

\[
\begin{array}{c}
\downarrow \\
n1 \\
\downarrow \\
n2
\end{array}
\]

\[
\begin{array}{c}
\downarrow \\
n1 \\
\downarrow \\
n2
\end{array}
\]

\[
\begin{array}{c}
\downarrow \\
n1 \\
\downarrow \\
n2
\end{array}
\]

\[
\begin{array}{c}
\downarrow \\
n1 \\
\downarrow \\
n2
\end{array}
\]

\[
\begin{array}{c}
\downarrow \\
n1 \\
\downarrow \\
n2
\end{array}
\]

\[
\begin{array}{c}
\downarrow \\
n1 \\
\downarrow \\
n2
\end{array}
\]
Lazy evaluation

\[\text{fib} = 0:1:\text{zipWith} \ (\+) \ \text{fib} \ (\text{tail} \ \text{fib})\]
\[\text{fibo} \ n = \text{fib} \ !! \ n\]

- \[\text{fib=} [0, 1, 1, ...]\]
- \[\text{tail fib} = [1, 1, 2, ...]\]
- \[0:1:\text{zipWith} \ (\+) \ \text{fib} \ (\text{tail} \ \text{fib})\]
  \[= [0, 1, 0 + 1, 1 + 1, 1 + 2, ...]\]
  \[= [0, 1, 1, 2, 3, ...]\]
### A $\lambda$-calculus with atomicity

The $\lambda$-calculus
[Church]

\[ \Lambda : \quad t, u ::= x \mid \lambda x.t \mid (t)u \]

- $$(\lambda x.t)(\lambda y.u) \xrightarrow{\beta} t\{(\lambda y.u)/x\}$$

The atomic $\lambda$-calculus
[Gundersen, Heijltjes, Parigot]

\[ \Lambda_a : t, u ::= x \mid \lambda x.t \mid (t)u \mid u[c] \]

- \( [c] ::= [x_1, \ldots, x_p \leftarrow t] \mid \)
  \[ [x_p \leftarrow \lambda y.(t_p)[c_1] \ldots [c_r]] \]

- Independent duplication of $\lambda y$ and $u$
- Naturally retrieves sharing and laziness
Extend these properties to other calculi?
Continuations: the sandwich approach

1. In front of the refrigerator, thinking about a sandwich,
2. Stick a continuation in your pocket,
3. Use ingredients and make a sandwich (sitting on the counter),
4. Invoke the continuation in your pocket,
5. Back to 1, but there is a sandwich on the counter, and all ingredients are gone: eat the sandwich.
Continuations: the sandwich approach

1. In front of the refrigerator, thinking about a sandwich,
2. Stick a continuation in your pocket,
3. Use ingredients and make a sandwich (sitting on the counter),
4. Invoke the continuation in your pocket,
5. Back to 1, but there is a sandwich on the counter, and all ingredients are gone: eat the sandwich.
Continuations: the sandwich approach

1. In front of the refrigerator, thinking about a sandwich,
2. Stick a continuation in your pocket,
3. Use ingredients and make a sandwich (sitting on the counter),
4. Invoke the continuation in your pocket,
5. Back to 1, but there is a sandwich on the counter, and all ingredients are gone: eat the sandwich.
Continuations: the sandwich approach

1. In front of the refrigerator, thinking about a sandwich,
2. Stick a continuation in your pocket,
3. Use ingredients and make a sandwich (sitting on the counter),
4. Invoke the continuation in your pocket,
5. Back to 1, but there is a sandwich on the counter, and all ingredients are gone: eat the sandwich.
Continuations: the sandwich approach

1. In front of the refrigerator, thinking about a sandwich,
2. Stick a continuation in your pocket,
3. Use ingredients and make a sandwich (sitting on the counter),
4. Invoke the continuation in your pocket,
5. Back to 1, but there is a sandwich on the counter, and all ingredients are gone: eat the sandwich.
Continuations: the sandwich approach

1. In front of the refrigerator, thinking about a sandwich,
2. Stick a continuation in your pocket,
3. Use ingredients and make a sandwich (sitting on the counter),
4. Invoke the continuation in your pocket,
5. Back to 1, but there is a sandwich on the counter, and all ingredients are gone: eat the sandwich.
\(\lambda\mu\)-calculi [Parigot, Saurin]

- \(\lambda \cong_{CH} \text{Intuitionistic Logic}\)
- \(\lambda\mu \cong_{CH} \text{Classical Logic } (A \lor \neg A)\)
- Classical operators \(\leftrightarrow\) Continuations [Griffin]

The \(\Lambda\mu S\)-calculus:

\[
\text{Streams } S, T ::= \alpha \mid t \circ S
\]
\[
\text{Terms } t, u ::= x \mid \lambda x.t \mid (t)u \mid (t)S \mid \mu \alpha. t
\]
\[
(\mu \alpha. t)u \rightarrow_{\mu} \mu \alpha. t\{u \circ \alpha/\alpha\}
\]
Explicit sharings and atomicity in $\lambda\mu$-calculi?
A $\Lambda\mu S$-calculus with explicit sharings

Closures $[\phi], [\psi] ::= [x_1, \ldots, x_p \leftarrow t] \mid [\gamma_1, \ldots, \gamma_p \leftarrow S]$

Streams $S, T ::= \alpha \mid t \circ S \mid S[\phi]$

Terms $t, u ::= x \mid \lambda x.t \mid (t)u \mid (t)S \mid \mu \alpha.t \mid u[\phi]$
The atomic $\Lambda\mu S$-calculus

Closures $[\phi], [\psi] ::= [\vec{x}_q ← t] | [\vec{γ}_q ← S] | [\vec{x}_q ← \lambda y.t^q] | [\vec{x}_q ← \mu β.t^q]$

Streams $S, T ::= \alpha | t \circ S | S[\phi]$

Terms $t, u ::= x | \lambda x.t | (t)u | (t)S | \mu α.t | u[\phi]$

$\lambda$-tuples $t^p ::= \langle t_1, \ldots, t_p \rangle | t^p[\phi]$
Future work

- Check the properties of this calculus:
  - Termination in a typed setting ✓
  - Type preservation under reduction
  - Confluence
  - Preserving termination w.r.t. λ

- Extend to calculi with more general effects