Collapsing Closures

Xuan Li, Andy King and Lunjin Lu

Funded by NSF INT-0327760 and EPSRC EP/C015517

Familiar program

append(Xs,Ys,Zs) :- Xs = [], Ys = Zs.
append(Xs,Ys,Zs) :- Xs = [X|Vs], Zs = [X|Ws], append(Vs,Ys,Ws).
The behaviour can be captured with a T-operator [TCS, 1989]:

$$
\begin{array}{ll}
I_0 = \emptyset & \\
I_1 = \{\text{append}(Xs, Ys, Zs) :- \theta_1\} & \text{where } \theta_1 = \{Xs \mapsto [], Ys \mapsto Zs\} \\
I_2 = \{\text{append}(Xs, Ys, Zs) :- \theta_2\} \cup I_1 & \text{where } \theta_2 = \{Xs \mapsto [X], Zs \mapsto [X|Ys]\} \\
I_3 = \{\text{append}(Xs, Ys, Zs) :- \theta_3\} \cup I_2 & \text{where} \\
 & \theta_3 = \{Xs \mapsto [X, Y], Zs \mapsto [X, Y|Ys]\} \\
I_i = \{\text{append}(Xs, Ys, Zs) :- \theta_i\} \cup I_{i-1} & \text{where} \\
 & \theta_i = \{Xs \mapsto [X_1, \ldots, X_{i-1}], Zs \mapsto [X_1, \ldots, X_{i-1}|Ys]\}
\end{array}
$$

The limit of the sequence (lfp of T operator) is the interpretation $I = \{\text{append}(Xs, Ys, Zs) :- \theta_i \mid i \in \mathbb{N}\}$ which is an *infinite* set.

To *finitely* compute a set-sharing description of the limit I

- A set-sharing description for a substitution θ is constructed from a *set* of sharing groups
- The sharing group $occ(\theta, y)$ for a substitution θ and a variable y is defined as $occ(\theta, y) = \{x \in \mathcal{V} \mid y \in var(\theta(x))\}$
- For $\theta_3 = \{Xs \mapsto [X, Y], Zs \mapsto [X, Y|Ys]\}$ it follows that:

$$\begin{aligned}
occ(\theta_3, X\ ) &= \{X, Xs, Zs\} \\
occ(\theta_3, Y\ ) &= \{Y, Xs, Zs\} \\
occ(\theta_3, Xs) &= \emptyset \\
occ(\theta_3, Ys) &= \{Ys, Zs\} \\
occ(\theta_3, Zs) &= \emptyset \\
occ(\theta_3, y\ \ ) &= \{y\} \quad \text{where} \quad y \notin \{Xs, Ys, Zs, X, Y\}
\end{aligned}$$

- The set $\{occ(\theta_3, y) \mid y \in \mathcal{V}\}$ records the *absence* of sharing within θ_3 but, alas, is intractable

Inducing finiteness

- The abstraction map $\alpha_{\mathcal{X}}(\theta)$ for *single* substitutions parameterised by a *finite* set of program variables $\mathcal{X}$ and defined so that $\alpha_{\mathcal{X}}(\theta) = \{occ(\theta, y) \cap \mathcal{X} \mid y \in \mathcal{V}\}$.
- If $\mathcal{X} = \{Xs, Ys, Zs\}$ then $\alpha_{\mathcal{X}}(\theta_3) = \{\emptyset, \{Xs, Zs\}, \{Ys, Zs\}\}$.
- The *description* $\alpha_{\mathcal{X}}(\theta_3)$ still shows that *Xs* and *Ys* do not share
- A note for Galois geeks: Galois connection (actually insertion) obtained by:
  - Abstraction map $\alpha_{\mathcal{X}} : \wp(Sub) \to \wp(\wp(\mathcal{X}))$ defined by: $\alpha_{\mathcal{X}}(\Theta) = \cup_{\theta \in \Theta} \alpha_{\mathcal{X}}(\theta)$.
  - Concretisation map $\gamma_{\mathcal{X}} : \wp(\wp(\mathcal{X})) \to \wp(Sub)$ defined by: $\gamma_{\mathcal{X}}(S) = \{\theta \in Sub \mid \alpha_{\mathcal{X}}(\theta) \subseteq S\}$.

J_i as a description of I_i

- $\gamma_{\mathcal{X}}(S)$ pins down the meaning of a set-sharing abstraction S
- $\gamma_{\mathcal{X}}$ provides a criteria for judging the correctness of an abstract version of the T operator
- Successive interpretations J_i are deemed to be correct iff for each constrained atom $\mathrm{append}(Xs, Ys, Zs) :\!-\, \theta \in I_i$ there exists $\mathrm{append}(Xs, Ys, Zs) :\!-\, S \in J_i$ such that $\theta \in \gamma_{\mathcal{X}}(S)$.

$$
\begin{array}{ll}
J_0 = \emptyset & \\
J_1 = \{\mathrm{append}(Xs, Ys, Zs) :\!-\, S_1\} & \text{where } S_1 = \{\emptyset, \{Ys, Zs\}\} \\
J_2 = \{\mathrm{append}(Xs, Ys, Zs) :\!-\, S_2\} \cup J_1 & \text{where} \\
\qquad S_2 = \{\emptyset, \{Xs, Zs\}, \{Ys, Zs\}, \{Xs, Ys, Zs\}\} & \\
J_3 = J_2 &
\end{array}
$$

Consider computing J_3 (fixpoint)

- A single application of the abstract T operator takes, as input, an interpretation J_i and produces, as output, an interpretation J_{i+1}.
- J_{i+1} is obtained as the union of the two interpretations: one generated by each clause in the program acting on J_i.
- Applying the first and second clauses to J_2 yield $\{\text{append}(Xs, Ys, Zs) :- S_1\}$ and $\{\text{append}(Xs, Ys, Zs) :- S_2\}$ respectively which, when combined, give $J_3 = J_2$.
- A *computational* problem arises in computing $\{\text{append}(Xs, Ys, Zs) :- S_2\}$.

Overview of analysis of second clause in 5 steps

append(Xs,Ys,Zs) :- Xs = [X|Vs], Zs = [X|Ws], append(Vs,Ys,Ws)

.

1. Set initial description to $S_0 = \alpha_{\mathcal{X}}(id) = \{\emptyset, \{Vs\}, \{Ws\}, \{X\}, \{Xs\}, \{Ys\}, \{Zs\}\}$ where $\mathcal{X} = \{Vs, Ws, X, Xs, Ys, Zs\}$
2. Simulate Xs = [X|Vs] with input S_0 to obtain output S_1:
   - apply $S_1 = amgu(Xs, [X|Vs], S_0)$
   - comes with the correctness guarantee that if $\theta \in \gamma_{\mathcal{X}}(S_0)$ and $\delta \in mgu(\theta(Xs), \theta([X|Vs]))$ then $\delta \circ \theta \in \gamma_{\mathcal{X}}(S_1)$
3. Apply $S_2 = amgu(Zs, [X|Ws], S_1)$
4. Add the bindings imposed append(Vs, Ys, Ws) by S_2 to obtain S_3 (amounts to 3 more *amgu* applications)
5. Restrict S_3 to variable set $\{Xs, Ys, Zs\}$ to obtain finally obtain the description $S_4 = \{\emptyset, \{Xs, Zs\}, \{Ys, Zs\}, \{Xs, Ys, Zs\}\}$

The problem at the heart of the *amgu*

- Put $S_0 = \{\emptyset, \{Vs\}, \{Ws\}, \{X\}, \{Xs\}, \{Ys\}, \{Zs\}\}$ (step 1)
- Compute $S_1 = amgu(X, [X|Vs], S_0)$ where:

$$amgu(X, [X|Vs], S_0) = S_0 \setminus (T_1 \cup T_2) \cup cl(T_1 \uplus T_2)$$

$$\begin{aligned} T_1 = rel(Xs, S_0) &= \{\{Xs\}\} \\ T_2 = rel([X|Vs], S_0) &= \{\{Vs\}, \{X\}\} \\ T_1 \uplus T_2 &= \{\{Vs, Xs\}, \{X, Xs\}\} \\ cl(T_1 \uplus T_2) &= \{\{Vs, Xs\}, \{Vs, X, Xs\}, \{X, Xs\}\} \end{aligned}$$

  to obtain $S_1 = \{\emptyset, \{Vs, Xs\}, \{Vs, X, Xs\}, \{Ws\}, \{X, Xs\}, \{Ys\}, \{Zs\}\}$ (step 2)
- Computing $S_2 = amgu(Zs, [X|Ws], S_1)$ is analogous (step 3)

The problem with closure made worse (step 4)

- The bindings imposed by the body atom append(Vs, Ys, Ws) need to be added to S_2.
- These bindings are recorded in J_2, not in terms of append(Vs, Ys, Ws), but in terms of append(Xs, Ys, Zs).
- Extend S_2 to give $S_3 = S_2 \cup \{\{\underline{Xs}, \underline{Zs}\}, \{\underline{Ys}, \underline{Zs}\}, \{\underline{Xs}, \underline{Ys}, \underline{Zs}\}\}$ where $\underline{Xs}$, $\underline{Ys}$ and $\underline{Zs}$ are fresh variables.
- Then interleave *amgu* and projection like so:
  - $S_4 = amgu(Vs, \underline{Xs}, S_3)$ and $S_5 = S_4 \restriction (\mathcal{X} \setminus \{\underline{Xs}\})$
  - $S_6 = amgu(Ys, \underline{Ys}, S_5)$ and $S_7 = S_6 \restriction (\mathcal{X} \setminus \{\underline{Ys}\})$
  - $S_8 = amgu(Ws, \underline{Zs}, S_7)$ and $S_9 = S_8 \restriction (\mathcal{X} \setminus \{\underline{Zs}\})$

Intermediate variables aggravate closure calculation

$$
\begin{aligned}
S_4 \;=\;& \{\emptyset, \{\underline{Xs}, \underline{Ys}, \underline{Zs}, Vs, Ws, X, Xs, Zs\}, \{\underline{Xs}, \underline{Ys}, \underline{Zs}, Vs, X, Xs, Zs\}, \\
& \{\underline{Xs}, \underline{Ys}, \underline{Zs}, Vs, Xs\}, \{\underline{Xs}, \underline{Zs}, Vs, Ws, X, Xs, Zs\}, \\
& \{\underline{Xs}, \underline{Zs}, Vs, X, Xs, Zs\}, \{\underline{Xs}, \underline{Zs}, Vs, Xs\}, \{\underline{Ys}, \underline{Zs}\}, \\
& \{Ws, X, Xs, Zs\}, \{Ws, Zs\}, \{X, Xs, Zs\}, \{Ys\}\} \\
S_5 \;=\;& \{\emptyset, \{\underline{Ys}, \underline{Zs}, Vs, Ws, X, Xs, Zs\}, \{\underline{Ys}, \underline{Zs}, Vs, X, Xs, Zs\}, \\
& \{\underline{Ys}, \underline{Zs}, Vs, xs\}, \{\underline{Zs}, Vs, Ws, X, Xs, Zs\} \\
& \{\underline{Zs}, Vs, X, Xs, Zs\}, \{\underline{Zs}, Vs, Xs\}, \{\underline{Ys}, \underline{Zs}\}, \\
& \{Ws, X, Xs, Zs\}, \{Ws, Zs\}, \{X, Xs, Zs\}, \{Ys\}\} \\
\vdots \\
S_9 \;=\;& \{\emptyset, \{Vs, Ws, X, Xs, Ys, Zs\}, \{Vs, Ws, Xs, Ys, Zs\}, \\
& \{Vs, Ws, X, Xs, Zs\}, \{Vs, Ws, Xs, Zs\}, \{Ws, X, Xs, Ys, Zs\}, \\
& \{Ws, Ys, Zs\}, \{X, Xs, Zs\}\}
\end{aligned}
$$

Recovering tractability with projection (step 5)

- The description $S_9 = \{\emptyset, \{Vs, Ws, X, Xs, Ys, Zs\}, \{Vs, Ws, Xs, Ys, Zs\}, \{Vs, Ws, X, Xs, Zs\}, \{Vs, Ws, Xs, Zs\}, \{Ws, X, Xs, Ys, Zs\}, \{Ws, Ys, Zs\}, \{X, Xs, Zs\}\}$ expresses the bindings imposed on the variables of the *whole clause*.
- The restriction $S_{10} = S_9 \upharpoonright \{Xs, Ys, Zs\} = \{\emptyset, \{Xs, Zs\}, \{Ys, Zs\}, \{Xs, Ys, Zs\}\}$ then describes bindings on the head.
- Since $S_{10} = S_2$ (original) it follows that a fixpoint has been reached and therefore J_2 describes the limit I.

Observation: this application of the abstract T requires 5 closures that all compute complicated descriptions over ≥ 6 variables.
Irony: S_{10} is simple since it is defined over 3 variables.

Idea: defer evaluation until a propitious moment

- Consider again the definition

$$amgu(t_1, t_2, S) = (S \setminus (T_1 \cup T_2)) \cup cl(T_1 \uplus T_2)$$

- Instead of computing $cl(T_1 \uplus T_2)$, the tactic is to tag all the groups within $T_1 \uplus T_2$ with an identifier — a unique number — that identifies those groups that participate in the closure.
- The tags are retained until head projection whereupon they are used to activate closure calculation.
- Then the tags are discarded.

Tagging with an identifier

Put $S_0 = \{\emptyset, \{Vs\}, \{Ws\}, \{X\}, \{Xs\}, \{Ys\}, \{Zs\}\}$ (step 1)
Compute $S_1 = amgu'(X, [X|Vs], 1, S_0)$ where:

$$amgu'(t_1, t_2, n, S) = (S \setminus (T_1 \cup T_2)) \cup tag(T_1 \uplus' T_2, n)$$

$$\begin{array}{rcl} T_1 = rel(Xs, S_0) & = & \{\{Xs\}\} \\ T_2 = rel([X|Vs], S_0) & = & \{\{Vs\}, \{X\}\} \\ T_1 \uplus' T_2 & = & \{\{Vs, Xs\}, \{X, Xs\}\} \\ tag(T_1 \uplus' T_2, 1) & = & \{\langle\{Vs, Xs\}, \{1\}\rangle, \langle\{X, Xs\}, \{1\}\rangle\} \end{array}$$

hence $S_1 = \{\emptyset, \langle\{Vs, Xs\}, \{1\}\rangle, \{Ws\}, \langle\{X, Xs\}, \{1\}\rangle, \{Ys\}, \{Zs\}\}$ (step 2)

Tagging with two identifiers

Repeating this tactic for $S_2 = amgu'(Zs, [X|Ws], 2, S_1)$ yields:

$$\begin{array}{rcl} T_1 = rel(Zs, S_1) & = & \{\{Zs\}\} \\ T_2 = rel([X|Ws], S_1) & = & \{\{Ws\}, \langle\{X, Xs\}, \{1\}\rangle\} \\ T_1 \uplus' T_2 & = & \{\{Ws, Zs\}, \langle\{X, Xs, Zs\}, \{1\}\rangle\} \\ tag(T_1 \uplus' T_2, 2) & = & \{\langle\{Ws, Zs\}, \{2\}\rangle, \langle\{X, Xs, Zs\}, \{1, 2\}\rangle\} \end{array}$$

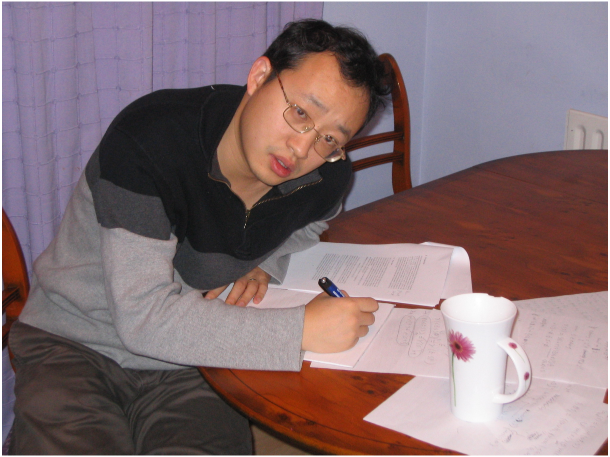
and $S_2 = \{\emptyset, \langle\{Vs, Xs\}, \{1\}\rangle, \langle\{Ws, Zs\}, \{2\}\rangle, \langle\{X, Xs, Zs\}, \{1, 2\}\rangle, \{Ys\}\}$.
Identifiers 1 and 2 show which groups participate in which closures.
Note that, unlike before, $|S_2| < |S_1| < |S_0|$.

Rules for collapsing closures (after head projection)

Suppose n is maximal identifier of S and $S_1 = cl(S, 1)$ and $S_i = cl(S_{i-1}, i)$. Denote $cl'(S) = untag(S_n)$.

- duplicated group rule:
  - $S_{10} = \{\emptyset, \langle\{Xs, Zs\}, \{1,2,3,5\}\rangle, \langle\{Xs, Zs\}, \{1,2,3,4,5\}\rangle, \langle\{Ys, Zs\}, \{2,4,5\}\rangle, \langle\{Xs, Zs\}, \{1,2\}\rangle\}$
  - $S_{11} = \{\emptyset, \langle\{Xs, Zs\}, \{1,2,3,4,5\}\rangle, \langle\{Ys, Zs\}, \{2,4,5\}\rangle\}$
- uniqueness rule:
  - $S_{12} = \{\emptyset, \langle\{Xs, Zs\}, \{2,4,5\}\rangle, \langle\{Ys, Zs\}, \{2,4,5\}\rangle\}$
- covering rule:
  - $S_{13} = \{\emptyset, \langle\{Xs, Zs\}, \{4,5\}\rangle, \langle\{Ys, Zs\}, \{4,5\}\rangle\}$
  - $S_{14} = \{\emptyset, \langle\{Xs, Zs\}, \{5\}\rangle, \langle\{Ys, Zs\}, \{5\}\rangle\}$

Proof of correctness



Tables of numbers

file	*classic*	*collapsed*
append.pl	9	16
qsort.pl	40	47
treeorder.pl	3360	797
peep.pl	3787	2330
boyer.pl	1016	343

Conclusions

- Collapsing closures is simpler than embedding set-sharing into another computational domain
- Set-sharing usually applied with linearity and freeness to:
  - identify where unification does not require closures;
  - improve the precision of set-sharing analysis itself
- Suggests the general technique of procrastination