CS738: Advanced Compiler Optimizations

Points-to Analysis using Types

Amey Karkare
karkare@cse.iitk.ac.in

http://www.cse.iitk.ac.in/~karkare/cs738
Department of CSE, IIT Kanpur

Reference Papers

- Bjarne Steensgaard: Points-to Analysis in Almost Linear Time. POPL 1996
- Manuvir Das: Unification-based pointer analysis with directional assignments. PLDI 2000

Language

$$
\begin{array}{lll}
S & ::= & x = y \\
 & \mid & x = \&y \\
 & \mid & x = *y \\
 & \mid & x = \text{allocate}(y) \\
 & \mid & *x = y \\
 & \mid & x = \text{fun}(f_1, \ldots, f_n) \text{ returns } r \text{ in } S^* \\
 & \mid & x = \text{p}(y_1, \ldots, y_n)
\end{array}
$$

Steensgaard's Analysis

- Non standard Types

$$
\begin{array}{lll}
s & \in & \text{Symbols} \\
\tau & \in & \text{Locations} ::= (\varphi, \alpha) \\
\varphi & \in & \text{Ids} ::= \{s_1, \ldots, s_n\} \\
\alpha & \in & \text{Values} ::= \perp \mid \text{ptr}(\tau)
\end{array}
$$

A denotes type environment.

Steensgaard's Analysis

- Partial Order

$$\alpha_1 \trianglelefteq \alpha_2 \quad \Leftrightarrow \quad (\alpha_1 = \bot) \vee (\alpha_1 = \alpha_2)$$

Steensgaard's Analysis: Typing Rules

$$\frac{A \vdash x : (\varphi, \alpha) \qquad A \vdash y : (\varphi', \alpha') \qquad \alpha' \trianglelefteq \alpha}{A \vdash \text{welltyped}(x = y)}$$

$$\frac{A \vdash x : (\varphi, \alpha) \qquad A \vdash y : \tau \qquad \text{ptr}(\tau) \trianglelefteq \alpha}{A \vdash \text{welltyped}(x = \&y)}$$

$$\frac{A \vdash x : (\varphi, \alpha) \qquad A \vdash y : (\varphi', \text{ptr}(\varphi'', \alpha'')) \qquad \alpha'' \trianglelefteq \alpha}{A \vdash \text{welltyped}(x = *y)}$$

$$\frac{A \vdash x : (\varphi, \text{ptr}(\varphi', \alpha')) \qquad A \vdash y : (\varphi'', \alpha'') \qquad \alpha'' \trianglelefteq \alpha'}{A \vdash \text{welltyped}(*x = y)}$$

$$\frac{A \vdash x : \tau}{A \vdash \text{welltyped}(x = \text{allocate}(y))}$$

Steensgaard's Analysis

- Function Definitions
- Need a new type value: $(\tau_1 \dots \tau_n) \to \tau$

$$\frac{\begin{array}{c} A \vdash x : (\tau_1 \dots \tau_n) \to \tau \\ \forall i \in \{1 \dots n\}. A \vdash f_i : \tau_i \\ A \vdash r : \tau \\ \forall s \in S^*. A \vdash \text{welltyped}(s) \end{array}}{A \vdash \text{welltyped}(x = \text{fun}(f_1, \dots, f_n) \text{ returns } r \text{ in } S^*)}$$

Steensgaard's Analysis

- Function Calls

$$\frac{\begin{array}{ll} A \vdash x : \tau & \tau = (\varphi, \alpha) \\ A \vdash \text{p} : (\tau_1 \dots \tau_n) \to \tau' & \tau_i = (\varphi_i, \alpha_i) \\ \forall i \in \{1 \dots n\}. A \vdash y_i : \tau_i' & \tau_i' = (\varphi_i', \alpha_i') \\ \alpha_i' \trianglelefteq \alpha_i & \alpha' \trianglelefteq \alpha \end{array}}{A \vdash \text{welltyped}(x = \text{p}(y_1, \dots, y_n))}$$

Manuvir Das's *One-level Flow-based* Analysis

$$\begin{aligned}
\alpha_1 \leq \alpha_2 &\Leftrightarrow \text{ptr}(\tau_1) \leq \text{ptr}(\tau_2) \\
&\Leftrightarrow \text{ptr}((\varphi', \alpha')) \leq \text{ptr}((\varphi, \alpha)) \\
&\Leftrightarrow (\varphi' \subseteq \varphi) \wedge (\alpha' = \alpha)
\end{aligned}$$

One-level Flow-based Analysis

- Replace $\trianglelefteq$ by $\leq$ in Steensgaard's analysis
- Keeps "top" level pointees separate!