

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

$S ::=$

- $x = y$
- $x = \&y$
- $x = *y$
- $x = \text{allocate}(y)$
- $*x = y$
- $x = \text{fun}(f_1, \dots, f_n) \text{ returns } r \text{ in } S^*$
- $x = p(y_1, \dots, y_n)$

Steensgaard's Analysis

- ▶ Non standard Types

$s \in \text{Symbols}$

$\tau \in \text{Locations} ::= (\varphi, \alpha)$

$\varphi \in \text{Ids} ::= \{s_1, \dots, s_n\}$

$\alpha \in \text{Values} ::= \perp \mid \text{ptr}(\tau)$

A denotes type environment.

Steensgaard's Analysis

► Partial Order

$$\alpha_1 \sqsubseteq \alpha_2 \Leftrightarrow (\alpha_1 = \perp) \vee (\alpha_1 = \alpha_2)$$

Steensgaard's Analysis: Typing Rules

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

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

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

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

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

Steensgaard's Analysis

► Function Definitions

► Need a new type value: $(\tau_1 \dots \tau_n) \rightarrow \tau$

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

Steensgaard's Analysis

► Function Calls

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

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!