Semantics and Verification of Software

Lecture 20: Dataflow Analysis VII (MOP vs. Fixpoint Solution)

Thomas Noll

Lehrstuhl für Informatik 2
(Software Modeling and Verification)

RWTH Aachen University

noll@cs.rwth-aachen.de

http://www-i2.informatik.rwth-aachen.de/i2/svsw08/

Winter semester 2008/09

Outline

The MOP Solution I

- Other solution method for dataflow systems
- MOP = Meet Over all Paths
- Analysis information for block B^l = least upper bound over all paths leading to l

Definition (Paths)

Let $S = (L, E, F, (D, \sqsubseteq), \iota, \varphi)$ be a dataflow system. For every $l \in L$, the set of paths up to l is given by

$$\begin{aligned} \mathit{Path}(l) := \{[l_1, \ldots, l_{k-1}] \mid\ & k \geq 1, l_1 \in E, \\ & (l_i, l_{i+1}) \in F \text{ for every } 1 \leq i \leq k, l_k = l\}. \end{aligned}$$

For a path $p = [l_1, \ldots, l_{k-1}] \in \mathit{Path}(l)$, we define the transfer function $\varphi_p : D \to D$ by

$$\varphi_p := \varphi_{l_{k-1}} \circ \ldots \circ \varphi_{l_1} \circ \mathsf{id}_D$$

(so that $\varphi_{[]} = \mathsf{id}_D$).

The MOP Solution II

Definition (MOP solution)

Let $S = (L, E, F, (D, \sqsubseteq), \iota, \varphi)$ be a dataflow system where $L = \{l_1, \ldots, l_n\}$. The MOP solution for S is determined by

$$\mathsf{mop}(S) := (\mathsf{mop}(l_1), \ldots, \mathsf{mop}(l_n)) \in D^n$$

where, for every $l \in L$,

$$\mathsf{mop}(l) := \bigsqcup\{\varphi_p(\iota) \mid p \in Path(l)\}.$$

Remark:

- $Path(l)$ is generally infinite

$\Longrightarrow$ not clear how to compute $\mathsf{mop}(l)$

- In fact: MOP solution generally undecidable (later)

Outline

Formalizing Constant Propagation Analysis I

The dataflow system $S = (L, E, F, (D, \sqsubseteq), \iota, \varphi)$ is given by

- set of labels $L := L_c$,
- extremal labels $E := \{\mathsf{init}(c)\}$ (forward problem),
- flow relation $F := \mathsf{flow}(c)$ (forward problem),
- complete lattice $(D, \sqsubseteq)$ where
  - $D := \{\delta \mid \delta : Var_c \to \mathbb{Z} \cup \{\bot, \top\}\}$
    - $\delta(x) = z \in \mathbb{Z}$: x has constant value z
    - $\delta(x) = \bot$: x undefined
    - $\delta(x) = \top$: x overdefined (i.e., different possible values)
  - $\sqsubseteq \subseteq D \times D$ defined by pointwise extension of $\bot \sqsubseteq z \sqsubseteq \top$ (for every $z \in \mathbb{Z}$)

Example

$Var_c = \{\mathtt{w}, \mathtt{x}, \mathtt{y}, \mathtt{z}\}$,

$\delta_1 = (\underbrace{\bot}_{\mathtt{w}}, \underbrace{1}_{\mathtt{x}}, \underbrace{2}_{\mathtt{y}}, \underbrace{\top}_{\mathtt{z}}),\ \delta_2 = (\underbrace{3}_{\mathtt{w}}, \underbrace{1}_{\mathtt{x}}, \underbrace{4}_{\mathtt{y}}, \underbrace{\top}_{\mathtt{z}})$

$\Longrightarrow\ \delta_1 \sqcup \delta_2 = (\underbrace{3}_{\mathtt{w}}, \underbrace{1}_{\mathtt{x}}, \underbrace{\top}_{\mathtt{y}}, \underbrace{\top}_{\mathtt{z}})$

Dataflow system $S = (L, E, F, (D, \sqsubseteq), \iota, \varphi)$ (continued):

- extremal value $\iota := \delta_\top \in D$ where $\delta_\top(x) := \top$ for every $x \in Var_c$,
- transfer functions $\{\varphi_l \mid l \in L\}$ defined by

$$\varphi_l(\delta) := \begin{cases} \delta & \text{if } B^l = \texttt{skip} \text{ or } B^l \in BExp \\ \delta[x \mapsto \mathfrak{A}[\![a]\!]\delta] & \text{if } B^l = (x \texttt{ := } a) \end{cases}$$

where

$$\mathfrak{A}[\![x]\!]\delta := \delta(x) \qquad \mathfrak{A}[\![z]\!]\delta := z$$

$$\mathfrak{A}[\![a_1 \ op \ a_2]\!]\delta := \begin{cases} z_1 \ op \ z_2 & \text{if } z_1, z_2 \in \mathbb{Z} \\ \bot & \text{if } z_1 = \bot \text{ or } z_2 = \bot \\ \top & \text{otherwise} \end{cases}$$

if $z_1 := \mathfrak{A}[\![a_1]\!]\delta$ and $z_2 := \mathfrak{A}[\![a_2]\!]\delta$

Example

If $\delta = (\underbrace{\bot}_{\texttt{w}}, \underbrace{1}_{\texttt{x}}, \underbrace{2}_{\texttt{y}}, \underbrace{\top}_{\texttt{z}})$, then

$$\varphi_l(\delta) = \begin{cases} (\underbrace{0}_{\texttt{w}}, \underbrace{1}_{\texttt{x}}, \underbrace{2}_{\texttt{y}}, \underbrace{\top}_{\texttt{z}}) & \text{if } B^l = (\texttt{w := 0}) \\ (\underbrace{3}_{\texttt{w}}, \underbrace{1}_{\texttt{x}}, \underbrace{2}_{\texttt{y}}, \underbrace{\top}_{\texttt{z}}) & \text{if } B^l = (\texttt{w := y+1}) \\ (\underbrace{\bot}_{\texttt{w}}, \underbrace{1}_{\texttt{x}}, \underbrace{2}_{\texttt{y}}, \underbrace{\top}_{\texttt{z}}) & \text{if } B^l = (\texttt{w := w+x}) \\ (\underbrace{\top}_{\texttt{w}}, \underbrace{1}_{\texttt{x}}, \underbrace{2}_{\texttt{y}}, \underbrace{\top}_{\texttt{z}}) & \text{if } B^l = (\texttt{w := z+2}) \end{cases}$$

Outline

Undecidability of the MOP Solution

Theorem 20.1 (Undecidability of MOP solution)

The MOP solution for Constant Propagation is undecidable.

Proof.

Based on undecidability of Modified Post Correspondence Problem:
Let Γ be some alphabet, $n \in \mathbb{N}$, and $u_1, \ldots, u_n, v_1, \ldots, v_n \in \Gamma^+$.
Does there exist $i_1, \ldots, i_m \in \{1, \ldots, n\}$ with $m \geq 1$ and $i_1 = 1$ such that $u_{i_1} u_{i_2} \ldots u_{i_m} = v_{i_1} v_{i_2} \ldots v_{i_m}$?

(on the board) □

Outline

MOP vs. Fixpoint Solution I

Theorem 20.2 (MOP vs. Fixpoint Solution)

Let $S = (L, E, F, (D, \sqsubseteq), \iota, \varphi)$ *be a dataflow system. Then*

$$\mathsf{mop}(S) \sqsubseteq \mathsf{fix}(\Phi_S)$$

Proof.

on the board □

The next example shows that both solutions can indeed be different.

Example 20.3 (Constant Propagation)

```
c := if [z > 0]^1 then
       [x := 2;]^2
       [y := 3;]^3
     else
       [x := 3;]^4
       [y := 2;]^5
     [z := x+y;]^6
     [...]^7
```

Transfer functions (for $\delta = (\delta(\mathtt{x}), \delta(\mathtt{y}), \delta(\mathtt{z})) \in D$):

$$\varphi_1((a,b,c)) = (a,b,c)$$
$$\varphi_2((a,b,c)) = (2,b,c)$$
$$\varphi_3((a,b,c)) = (a,3,c)$$
$$\varphi_4((a,b,c)) = (3,b,c)$$
$$\varphi_5((a,b,c)) = (a,2,c)$$
$$\varphi_6((a,b,c)) = (a,b,a+b)$$

1. Fixpoint solution:

$$\begin{aligned}
\mathsf{CP}_1 &= \iota && = (\top,\top,\top)\\
\mathsf{CP}_2 &= \varphi_1(\mathsf{CP}_1) && = (\top,\top,\top)\\
\mathsf{CP}_3 &= \varphi_2(\mathsf{CP}_2) && = (2,\top,\top)\\
\mathsf{CP}_4 &= \varphi_1(\mathsf{CP}_1) && = (\top,\top,\top)\\
\mathsf{CP}_5 &= \varphi_2(\mathsf{CP}_2) && = (3,\top,\top)\\
\mathsf{CP}_6 &= \varphi_3(\mathsf{CP}_3) \sqcup \varphi_5(\mathsf{CP}_5) &&\\
&= (2,3,\top) \sqcup (3,2,\top) && = (\top,\top,\top)\\
\mathsf{CP}_7 &= \varphi_6(\mathsf{CP}_6) && = (\top,\top,\top)
\end{aligned}$$

2. MOP solution:

$$\begin{aligned}
\mathsf{mop}(7) &= \varphi_{[1,2,3,6]}(\top,\top,\top) \sqcup\\
&\quad\ \varphi_{[1,4,5,6]}(\top,\top,\top)\\
&= (2,3,5) \sqcup (3,2,5)\\
&= (\top,\top,5)
\end{aligned}$$

Distributive Transfer Functions I

A sufficient criterion for the coincidence of MOP and Fixpoint Solution is the distributivity of the transfer functions.

Definition 20.4 (Distributivity)

- Let $(D, \sqsubseteq)$ and $(D', \sqsubseteq')$ be complete lattices, and let $F : D \to D'$. F is called distributive (w.r.t. $(D, \sqsubseteq)$ and $(D', \sqsubseteq')$) if, for every $d_1, d_2 \in D$,
$$F(d_1 \sqcup_D d_2) = F(d_1) \sqcup_{D'} F(d_2).$$
- A dataflow system $S = (L, E, F, (D, \sqsubseteq), \iota, \varphi)$ is called distributive if every $\varphi_l : D \to D$ $(l \in L)$ is so.

Distributive Transfer Functions II

Example 20.5

1. The Available Expressions dataflow system is distributive:

$$\begin{aligned}\varphi_l(A_1 \sqcup A_2) &= ((A_1 \cap A_2) \setminus \mathsf{kill}_{\mathsf{AE}}(B^l)) \cup \mathsf{gen}_{\mathsf{AE}}(B^l) \\ &= ((A_1 \setminus \mathsf{kill}_{\mathsf{AE}}(B^l)) \cup \mathsf{gen}_{\mathsf{AE}}(B^l)) \cap \\ &\quad\, ((A_2 \setminus \mathsf{kill}_{\mathsf{AE}}(B^l)) \cup \mathsf{gen}_{\mathsf{AE}}(B^l)) \\ &= \varphi_l(A_1) \sqcup \varphi_l(A_2)\end{aligned}$$

2. The Live Variables dataflow system is distributive (similar)
3. The Constant Propagation dataflow system is not distributive:

$$\begin{aligned}(\top, \top, \top) &= \varphi_{\mathtt{z:=x+y}}((2, 3, \top) \sqcup (3, 2, \top)) \\ &\neq \varphi_{\mathtt{z:=x+y}}((2, 3, \top)) \sqcup \varphi_{\mathtt{z:=x+y}}((3, 2, \top)) \\ &= (\top, \top, 5)\end{aligned}$$

Coincidence of MOP and Fixpoint Solution

Theorem 20.6 (MOP vs. Fixpoint Solution)

Let $S = (L, E, F, (D, \sqsubseteq), \iota, \varphi)$ be a distributive dataflow system. Then

$$\mathsf{mop}(S) = \mathsf{fix}(\Phi_S)$$

Proof.

- by showing that $\Phi_S(\mathsf{mop}(S)) = \mathsf{mop}(S)$ ...
  (see [Nielson/Nielson/Hankin 2005, p. 81])
- ... and using $\mathsf{mop}(S) \sqsubseteq \mathsf{fix}(\Phi_S)$ (Theorem 20.2)

□