# Semantics and Verification of Software

## Lecture 17: Dataflow Analysis IV (Equation Solving)

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

# Complete Lattices

## Definition (Complete lattice)

A complete lattice is a partial order $(D, \sqsubseteq)$ such that all subsets of $D$ have least upper as well as greatest lower bounds. In this case,

$$\bot := \bigsqcup \emptyset = \bigsqcap D \text{ and}$$
$$\top := \bigsqcap \emptyset = \bigsqcup D$$

denote the least and the greatest element of $D$, respectively.

## Example

1. (Available Expressions) $(D, \sqsubseteq) = (2^{AExp_c}, \supseteq)$ is a complete lattice with $\bot = AExp_c$ and $\top = \emptyset$
2. (Live Variables) $(D, \sqsubseteq) = (2^{Var_c}, \subseteq)$ is a complete lattice with $\bot = \emptyset$ and $\top = Var_c$

# Chains

Chains represent the approximation of the analysis information.

## Definition (Chain; repetition of Def. 6.4 and 6.6)

Let $(D, \sqsubseteq)$ be a partial order.

1. A subset $S \subseteq D$ is called a chain in $D$ if, for every $s_1, s_2 \in S$,
$$s_1 \sqsubseteq s_2 \text{ or } s_2 \sqsubseteq s_1$$
(that is, $S$ is a totally ordered subset of $D$).
2. $(D, \sqsubseteq)$ is called chain complete (CCPO) if each of its chains has a least upper bound.
3. $(D, \sqsubseteq)$ satisfies the Ascending Chain Condition (ACC) if each ascending chain $d_1 \sqsubseteq d_2 \sqsubseteq \ldots$ eventually stabilizes, i.e., there exists $n \in \mathbb{N}$ such that $d_n = d_{n+1} = \ldots$

## Corollary

*Complete lattices are CCPOs.*

# Monotonicity of Functions

Transfer functions formalize the impact of a block in the program on the analysis information.

## Definition (Monotonicity; repetition of Def. 7.1)

Let $(D, \sqsubseteq)$ and $(D', \sqsubseteq')$ be partial orders, and let $F : D \to D'$. $F$ is called monotonic (w.r.t. $(D, \sqsubseteq)$ and $(D', \sqsubseteq')$) if, for every $d_1, d_2 \in D$,

$$d_1 \sqsubseteq d_2 \implies F(d_1) \sqsubseteq' F(d_2).$$

## Example

1. (Available Expressions) $(D, \sqsubseteq) = (2^{AExp_c}, \supseteq)$
   Each transfer function $\varphi_{l'}(A) := (A \setminus \mathsf{kill}_{\mathsf{AE}}(B^{l'})) \cup \mathsf{gen}_{\mathsf{AE}}(B^{l'})$ is monotonic
2. (Live Variables) $(D, \sqsubseteq) = (2^{Var_c}, \subseteq)$
   Each transfer function $\varphi_{l'}(V) := (V \setminus \mathsf{kill}_{\mathsf{LV}}(B^{l'})) \cup \mathsf{gen}_{\mathsf{LV}}(B^{l'})$ is monotonic

# Fixpoints

## Theorem (Fixpoint Theorem; repetition of Thm. 7.7)

*Let $(D, \sqsubseteq)$ be a CCPO and $F : D \to D$ continuous. Then*

$$\mathsf{fix}(F) := \bigsqcup \{F^n (\bigsqcup \emptyset) \mid n \in \mathbb{N}\}$$

*is the least fixpoint of $F$.*

## Definition (Continuity; repetition of Def. 7.5)

Let $(D, \sqsubseteq)$ and $(D', \sqsubseteq')$ be CCPOs and $F : D \to D'$ monotonic. Then $F$ is called continuous (w.r.t. $(D, \sqsubseteq)$ and $(D', \sqsubseteq')$) if, for every non-empty chain $S \subseteq D$,

$$F(\bigsqcup S) = \bigsqcup F(S).$$

## Corollary

*Monotonic functions on partial orders that satisfy ACC are continuous.*

# Dataflow Systems I

## Definition (Dataflow system)

A dataflow system $S = (L, E, F, (D, \sqsubseteq), \iota, \varphi)$ consists of

- a finite set of (program) labels $L$ (here: $L_c$),
- a set of extremal labels $E \subseteq L$ (here: $\{\mathsf{init}(c)\}$ or $\mathsf{final}(c)$),
- a flow relation $F \subseteq L \times L$ (here: $\mathsf{flow}(c)$ or $\mathsf{flow}^R(c)$),
- a complete lattice $(D, \sqsubseteq)$ that satisfies ACC (with LUB operator $\bigsqcup$ and least element $\bot$),
- an extremal value $\iota \in D$ (for the extremal labels), and
- a collection of monotonic transfer functions $\{\varphi_l \mid l \in L\}$ of type $\varphi_l : D \to D$.

## Example

| Problem | **Available Expressions** | **Live Variables** |
|---|---|---|
| $E$ | $\{\mathsf{init}(c)\}$ | $\mathsf{final}(c)$ |
| $F$ | $\mathsf{flow}(c)$ | $\mathsf{flow}^R(c)$ |
| $D$ | $2^{AExp_c}$ | $2^{Var_c}$ |
| $\sqsubseteq$ | $\supseteq$ | $\subseteq$ |
| $\bigsqcup$ | $\bigcap$ | $\bigcup$ |
| $\bot$ | $AExp_c$ | $\emptyset$ |
| $\iota$ | $\emptyset$ | $Var_c$ |
| $\varphi_l$ | $\varphi_l(d) = (d \setminus \mathsf{kill}(B^l)) \cup \mathsf{gen}(B^l)$ | |

# Outline

# The Equation System

**Definition 17.1 (Dataflow equation system)**

Let $S = (L, E, F, (D, \sqsubseteq), \iota, \varphi)$ be a dataflow system. $S$ defines the following equation system over the set of variables $\{\mathsf{AI}_l \mid l \in L\}$:

$$\mathsf{AI}_l = \begin{cases} \iota & \text{if } l \in E \\ \bigsqcup\{\varphi_{l'}(\mathsf{AI}_{l'}) \mid (l', l) \in F\} & \text{otherwise} \end{cases}$$

# The Functional

Just as in the denotational semantics of `while` loops, the equation system determines a functional whose fixpoints are exactly the solutions of the equation system.

## Definition 17.2 (Dataflow functional)

The equation system of a dataflow system $S = (L, E, F, (D, \sqsubseteq), \iota, \varphi)$ induces a functional

$$\Phi_S : D^n \to D^n : (d_{l_1}, \ldots, d_{l_n}) \mapsto (d'_{l_1}, \ldots, d'_{l_n})$$

where $L = \{l_1, \ldots, l_n\}$ and, for each $1 \leq i \leq n$,

$$d'_{l_i} := \begin{cases} \iota & \text{if } l_i \in E \\ \bigsqcup\{\varphi_{l'}(d_{l'}) \mid (l', l_i) \in F\} & \text{otherwise} \end{cases}$$

# Fixpoint Iteration I

**Remarks:**

- $(D, \sqsubseteq)$ being a complete lattice ensures that $\Phi_S$ is well defined
- $(d_1, \ldots, d_n)$ is a solution of the equation system iff it is a fixpoint of $\Phi_S$
- If $(D, \sqsubseteq)$ is a complete lattice satisfying ACC, then so is $(D^n, \sqsubseteq^n)$ (where $(d_1, \ldots, d_n) \sqsubseteq^n (d'_1, \ldots, d'_n)$ iff $d_i \sqsubseteq d'_i$ for every $1 \leq i \leq n$)
- Every transfer function $\varphi_l$ monotonic in $D$
  $\Longrightarrow$ $\Phi_S$ monotonic in $D^n$
- Thus the (least) fixpoint is effectively computable by iteration:

$$\mathsf{fix}(\Phi_S) = \bigsqcup \{\Phi_S^i(\bot_{D^n}) \mid i \in \mathbb{N}\}$$

  where $\bot_{D^n} = (\underbrace{\bot_D, \ldots, \bot_D}_{n \text{ times}})$

- If maximal length of chains in $D$ is $m$
  $\Longrightarrow$ maximal length of chains in $D^n$ is $m \cdot n$
  $\Longrightarrow$ fixpoint iteration requires at most $m \cdot n$ steps

# Fixpoint Iteration II

## Example 17.3 (Available Expressions; cf. Example 14.9)

Program:

$c =$ `[x := a+b]`$^1$`;`
`[y := a*b]`$^2$`;`
`while [y > a+b]`$^3$ `do`
`[a := a+1]`$^4$`;`
`[x := a+b]`$^5$

Equation system:

$$\begin{aligned}
\mathsf{AE}_1 &= \emptyset \\
\mathsf{AE}_2 &= \mathsf{AE}_1 \cup \{\texttt{a+b}\} \\
\mathsf{AE}_3 &= (\mathsf{AE}_2 \cup \{\texttt{a*b}\}) \cap (\mathsf{AE}_5 \cup \{\texttt{a+b}\}) \\
\mathsf{AE}_4 &= \mathsf{AE}_3 \cup \{\texttt{a+b}\} \\
\mathsf{AE}_5 &= \mathsf{AE}_4 \setminus \{\texttt{a+b}, \texttt{a*b}, \texttt{a+1}\}
\end{aligned}$$

Fixpoint iteration:

| $i$ | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 0 | $AExp_c$ | $AExp_c$ | $AExp_c$ | $AExp_c$ | $AExp_c$ |
| 1 | $\emptyset$ | $AExp_c$ | $AExp_c$ | $AExp_c$ | $\emptyset$ |
| 2 | $\emptyset$ | {a+b} | {a+b} | $AExp_c$ | $\emptyset$ |
| 3 | $\emptyset$ | {a+b} | {a+b} | {a+b} | $\emptyset$ |
| 4 | $\emptyset$ | {a+b} | {a+b} | {a+b} | $\emptyset$ |

# Fixpoint Iteration III

## Example 17.4 (Live Variables; cf. Example 15.3)

Program:

$$
\begin{array}{l}
[\texttt{x := 2}]^1;[\texttt{y := 4}]^2; \\
[\texttt{x := 1}]^3; \\
\texttt{if } [\texttt{y > 0}]^4 \texttt{ then} \\
\quad [\texttt{z := x}]^5 \\
\texttt{else} \\
\quad [\texttt{z := y*y}]^6; \\
[\texttt{x := z}]^7
\end{array}
$$

Equation system:

$$
\begin{aligned}
\mathsf{LV}_1 &= \mathsf{LV}_2 \setminus \{\texttt{y}\} \\
\mathsf{LV}_2 &= \mathsf{LV}_3 \setminus \{\texttt{x}\} \\
\mathsf{LV}_3 &= \mathsf{LV}_4 \cup \{\texttt{y}\} \\
\mathsf{LV}_4 &= ((\mathsf{LV}_5 \setminus \{\texttt{z}\}) \cup \{\texttt{x}\}) \cup ((\mathsf{LV}_6 \setminus \{\texttt{z}\}) \cup \{\texttt{y}\}) \\
\mathsf{LV}_5 &= (\mathsf{LV}_7 \setminus \{\texttt{x}\}) \cup \{\texttt{z}\} \\
\mathsf{LV}_6 &= (\mathsf{LV}_7 \setminus \{\texttt{x}\}) \cup \{\texttt{z}\} \\
\mathsf{LV}_7 &= \{\texttt{x}, \texttt{y}, \texttt{z}\}
\end{aligned}
$$

Fixpoint iteration:

| $i$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 0 | $\emptyset$ | $\emptyset$ | $\emptyset$ | $\emptyset$ | $\emptyset$ | $\emptyset$ | $\emptyset$ |
| 1 | $\emptyset$ | $\emptyset$ | $\{\texttt{y}\}$ | $\{\texttt{x}, \texttt{y}\}$ | $\{\texttt{z}\}$ | $\{\texttt{z}\}$ | $\{\texttt{x}, \texttt{y}, \texttt{z}\}$ |
| 2 | $\emptyset$ | $\{\texttt{y}\}$ | $\{\texttt{x}, \texttt{y}\}$ | $\{\texttt{x}, \texttt{y}\}$ | $\{\texttt{y}, \texttt{z}\}$ | $\{\texttt{y}, \texttt{z}\}$ | $\{\texttt{x}, \texttt{y}, \texttt{z}\}$ |
| 3 | $\emptyset$ | $\{\texttt{y}\}$ | $\{\texttt{x}, \texttt{y}\}$ | $\{\texttt{x}, \texttt{y}\}$ | $\{\texttt{y}, \texttt{z}\}$ | $\{\texttt{y}, \texttt{z}\}$ | $\{\texttt{x}, \texttt{y}, \texttt{z}\}$ |

# Outline

# Uniqueness of Solutions

Just as in the denotational semantics of `while` loops, solutions of dataflow equation systems are not unique.

## Example 17.5

1. Available Expressions: consider

$$
\begin{array}{ll}
[\texttt{z := x+y}]^1; & \\
\texttt{while } [\texttt{true}]^2 \texttt{ do} & \Longrightarrow \quad \mathsf{AE}_1 = \emptyset \\
\quad [\texttt{skip}]^3; & \qquad\quad \mathsf{AE}_2 = (\mathsf{AE}_1 \cup \{\texttt{x+y}\}) \cap \mathsf{AE}_3 \\
& \qquad\quad \mathsf{AE}_3 = \mathsf{AE}_2 \\
& \Longrightarrow \quad \mathsf{AE}_1 = \emptyset \\
& \qquad\quad \mathsf{AE}_2 = \{\texttt{x+y}\} \cap \mathsf{AE}_3 \\
& \qquad\quad \mathsf{AE}_3 = \mathsf{AE}_2
\end{array}
$$

$\Longrightarrow$ Solutions: $\mathsf{AE}_1 = \mathsf{AE}_2 = \mathsf{AE}_3 = \emptyset$ or
$\mathsf{AE}_1 = \emptyset, \mathsf{AE}_2 = \mathsf{AE}_3 = \{\texttt{x+y}\}$

Here: greatest solution $\{\texttt{x+y}\}$ (maximal potential for optimization)

2. Live Variables: see Exercise 9.3