Abstract Interpretation

Yu Zhang

Most content comes from http://cs.au.dk/~amoeller/spa/

Agenda

- **Collecting semantics**
- Abstraction and concretization
- Soundness
- Optimality

Program Semantics as Constraint Systems

- Concrete state: program variables to integers

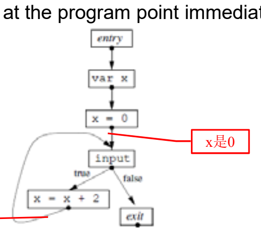
  $ConcreteStates = Vars \to \mathbb{Z}$

- Constraint variable for each CFG node v

  $[\![v]\!] \subseteq ConcreteStates$

  - Denote the state at the program point immediately after v

```
var x;
x = 0;
while (input) {
  x = x + 2;
}
```



The Semantics of Expressions

- Concrete execution ➔ Abstract execution

  $ceval : ConcreteStates \times E \to 2^{\mathbb{Z}}$

  - A concrete state ρ results in a set of possible integer values

  $ceval(\rho, X) = \{\rho(X)\}$
  $ceval(\rho, I) = \{I\}$
  $ceval(\rho, \mathbf{input}) = \mathbb{Z}$
  $ceval(\rho, E_1 \ \mathbf{op}\ E_2) = \{v_1 \ \mathbf{op}\ v_2 \mid v_1 \in ceval(\rho, E_1) \ \wedge \ v_2 \in ceval(\rho, E_2)\}$

- Overload *ceval* to work on sets of concrete states

  $ceval(R, E) = \bigcup_{\rho \in R} ceval(\rho, E) \qquad ceval : 2^{ConcreteStates} \times E \to 2^{\mathbb{Z}}$

Successors and Joins

- Possible successors of a CFG node relative to a concrete state $csucc: ConcreteStates \times Nodes \to 2^{Nodes}$

  ➔ work on a set of concrete states

  $csucc: 2^{ConcreteStates} \times Nodes \to 2^{Nodes}$

  $csucc(R, v) = \bigcup_{\rho \in R} csucc(\rho, v)$

$CJOIN(v) =$

$\{\rho \in ConcreteStates \mid \exists w \in Nodes \colon \rho \in [\![w]\!] \wedge v \in csucc(\rho, w)\}$

Semantics of Statements

A flow-insensitive analysis that tracks function values:

$[\![X{=}E]\!] = \{\rho[X \mapsto z] \mid \rho \in CJOIN(v) \ \wedge \ z \in ceval(\rho, E)\}$

$[\![\mathbf{var}\ X_1, \ldots, X_n]\!] =$
$\{\rho[X_1 \mapsto z_1, \ldots, X_n \mapsto z_n] \mid \rho \in CJOIN(v) \wedge z_1 \in \mathbb{Z} \wedge \cdots \wedge z_n \in \mathbb{Z}$

$[\![entry]\!] = \{[\,]\}$

$[\![v]\!] = CJOIN(v)$

The Resulting Constraint System

- A program with n CFG nodes, $v_1, \cdots, v_n$

$$[\![v_1]\!] = cf_1([\![v_1]\!], \ldots, [\![v_n]\!])$$
$$[\![v_2]\!] = cf_2([\![v_1]\!], \ldots, [\![v_n]\!])$$
$$\vdots$$
$$[\![v_n]\!] = cf_n([\![v_1]\!], \ldots, [\![v_n]\!])$$

- Combine n functions into one

$$cf(x_1, \ldots, x_n) = (cf_1(x_1, \ldots, x_n), \ldots, cf_n(x_1, \ldots, x_n))$$
$$cf : (2^{ConcreteStates})^n \to (2^{ConcreteStates})^n$$

$$x = cf(x)$$

7

Example

```
var x;
x = 0;
while (input) {
  x = x + 2;
}
```

	solution 1	solution 2
$[\![entry]\!]$	$\{[]\}$	$\{[]\}$
$[\![$var x$]\!]$	$\{[x \mapsto z] \mid z \in \mathbb{Z}\}$	$\{[x \mapsto z] \mid z \in \mathbb{Z}\}$
$[\![$x = 0$]\!]$	$\{[x \mapsto 0]\}$	$\{[x \mapsto 0]\}$
$[\![$input$]\!]$	$\{[x \mapsto z] \mid z \in \{0, 2, 4, \ldots\}\}$	$\{[x \mapsto z] \mid z \in \mathbb{Z}\}$
$[\![$x = x + 2$]\!]$	$\{[x \mapsto z] \mid z \in \{2, 4, \ldots\}\}$	$\{[x \mapsto z] \mid z \in \mathbb{Z}\}$
$[\![exit]\!]$	$\{[x \mapsto z] \mid z \in \{0, 2, 4, \ldots\}\}$	$\{[x \mapsto z] \mid z \in \mathbb{Z}\}$

the least solution

8

A Fixed Point Theorem for Continuous Functions

- $f: L_1 \to L_2$ is continuous if

$$f(\bigsqcup A) = \bigsqcup_{a \in A} f(a) \qquad \text{for every } A \subseteq L$$

- If f is continuous

$$fix(f) = \bigsqcup_{i \geq 0} f^i(\bot)$$

(even when L has infinite height!)

- cf is continuous

9

Semantics vs. Analysis

```
var a,b,c;
a = 42;
b = 87;
if (input) {
  c = a + b;
} else {
  c = a - b;
}
```

$$\{\![\text{b = 87}]\!\} = \{[a \mapsto 42, b \mapsto 87, c \mapsto z] \mid z \in \mathbb{Z}\}$$
$$\{\![\text{c = a - b}]\!\} = \{[a \mapsto 42, b \mapsto 87, c \mapsto -45]\}$$
$$\{\![exit]\!\} = \{[a \mapsto 42, b \mapsto 87, c \mapsto 129], [a \mapsto 42, b \mapsto 87, c \mapsto -45]\}$$

$$[\![\text{b = 87}]\!] = [a \mapsto +, b \mapsto +, c \mapsto \top]$$
$$[\![\text{c = a - b}]\!] = [a \mapsto +, b \mapsto +, c \mapsto \top]$$
$$[\![exit]\!] = [a \mapsto +, b \mapsto +, c \mapsto \top]$$

10

Agenda

- Collecting semantics
- **Abstraction and concretization**
- Soundness
- Optimality

11

Abstract Functions for Sign Analysis

- Abstract functions

$$\alpha_a : 2^{\mathbb{Z}} \to Sign$$
$$\alpha_b : 2^{ConcreteStates} \to States \qquad ConcreteStates = Vars \to \mathbb{Z}$$
$$\alpha_c : (2^{ConcreteStates})^n \to States^n \qquad State = Vars \to Sign$$

$$\alpha_a(D) = \begin{cases} \bot & \text{if } D \text{ is empty} \\ + & \text{if } D \text{ is nonempty and contains only positive integers} \\ - & \text{if } D \text{ is nonempty and contains only negative integers} \\ 0 & \text{if } D \text{ is nonempty and contains only the integer } 0 \\ \top & \text{otherwise} \end{cases}$$
for any $D \in 2^{\mathbb{Z}}$

$\alpha_b(R) = \sigma$ where $\sigma(X) = \alpha_a(\{\rho(X) \mid \rho \in R\})$
for any $R \subseteq ConcreteStates$ and $X \in Vars$

$\alpha_c(R_1, \ldots, R_n) = (\alpha_b(R_1), \ldots, \alpha_b(R_n))$
for any $R_1, \ldots, R_n \subseteq ConcreteStates$

12

Concretization Functions for Sign Analysis

- Concretization functions

$$\gamma_a : Sign \rightarrow 2^{\mathbb{Z}}$$
$$\gamma_b : States \rightarrow 2^{ConcreteStates}$$
$$\gamma_c : States^n \rightarrow (2^{ConcreteStates})^n$$

$$\gamma_a(s) = \begin{cases} \emptyset & \text{if } s = \bot \\ \{1, 2, 3, \dots\} & \text{if } s = + \\ \{-1, -2, -3, \dots\} & \text{if } s = - \\ \{0\} & \text{if } s = 0 \\ \mathbb{Z} & \text{if } s = \top \end{cases}$$
for any $s \in Sign$

$$\gamma_b(\sigma) = \{\rho \in ConcreteStates \mid \rho(X) \in \gamma_a(\sigma(X)) \text{ for all } X \in Vars\}$$
for any $\sigma \in States$

$$\gamma_c(\sigma_1, \dots, \sigma_n) = (\gamma_b(\sigma_1), \dots, \gamma_b(\sigma_n))$$
for any $(\sigma_1, \dots, \sigma_n) \in States^n$

13

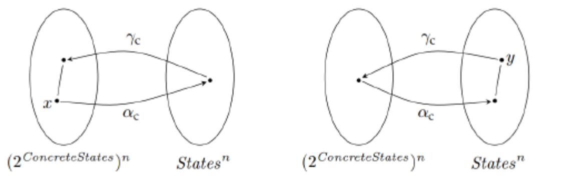
Galois Connections

- Galois Theory 伽罗瓦理论
  - 建立域论和群论之间的联系

The pair of monotone functions, α and γ, is called a *Galois connection* if

$\gamma \circ \alpha$ is extensive $x \sqsubseteq \gamma(\alpha(x))$ for all $x \in L_1$

$\alpha \circ \gamma$ is reductive $\alpha(\gamma(y)) \sqsubseteq y$ for all $y \in L_2$



14

Galois Connections

- The concretization function uniquely determines the abstraction function

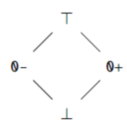
$$\gamma(y) = \bigsqcup_{x \in L_1 \text{ where } \alpha(x) \sqsubseteq y} x$$

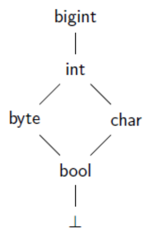
$$\alpha(x) = \bigsqcap_{y \in L_2 \text{ where } x \sqsubseteq \gamma(y)} y$$

15

Galois Connections

- For each of these two lattices, given the “obvious” concretization function, is there an abstraction function such that the concretization function and the abstraction function form a Galois connection?





16

Agenda

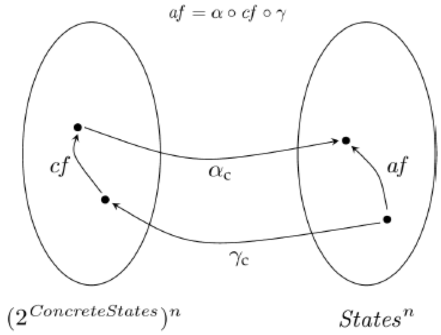
- Collecting semantics
- Abstraction and concretization
- Soundness
- **Optimality**

17

Optimal Approximations

af is an *optimal* approximation of cf if

$$af = \alpha \circ cf \circ \gamma$$



18

Optimal Approximations in Sign Analysis?

$\widehat{*}$ is optimal:

$$s_1 \widehat{*} s_2 = \alpha_a(\gamma_a(s_1) \cdot \gamma_a(s_2))$$

$eval$ is *not* optimal:

$$\sigma(\mathtt{x}) = \top$$

$$eval(\sigma, \mathtt{x\text{-}x}) = \top$$

$$\alpha_b(ceval(\gamma_b(\sigma), \mathtt{x\text{-}x})) = \mathbf{0}$$

Even if we could make $eval$ optimal, the analysis result is not always optimal:

```
x = input;
y = x;
z = x - y;
```

19