Abstract Interpretation

2008년 1월 30일

숙명여대

창병모

Contents

- Motivation
- Program Analysis
- Abstract Interpretation
- Applications
- References

Motivation

우리는 무엇을 해야 하는가?

❖ 컴퓨터 기술자 무엇을 해야 하는가?

The production of reliable software,
its maintenance and
safe evolution year after year (up to 20 to 30 years).

Hardware vs Software

- Hardware
  - The 25 last years, computer hardware has seen its performances multiplied by 10^4 to 10^6;
- Software
  - The size of programs executed by these computers has grown up in similar proportions;
- Example
  - Windows
    - > 30 000 000 lines
    - > 30 000 known bugs

Software development

- Software development
  - Large scale computer programming is very difficult;
  - Reasoning on large programs is very difficult;
  - Errors are quite frequent.

- Idea
  - Use the computer to find programming errors.
  - How can computers be programmed so as to analyze the behavior of software before and without executing it ?
  - This is essential for safety-critical systems
    (for example: planes, trains, launchers, nuclear plants, …)

Program Analysis

프로그램 분석Static Program Analysis

- 프로그램의 실행 성질을 실행 전에 자동으로 안전하게 어림잡는 기술이다[Yi04-05].
  - 실행 전에: 프로그램을 실행시키지 않고
  - 자동으로: 프로그램이 프로그램을 분석
  - 안전하게: 모든 가능성을 포섭하도록
  - 어림잡는: 실제 이외의 것들이 포함됨
    어림잡지 않으면 불가능

Program Analysis Techniques

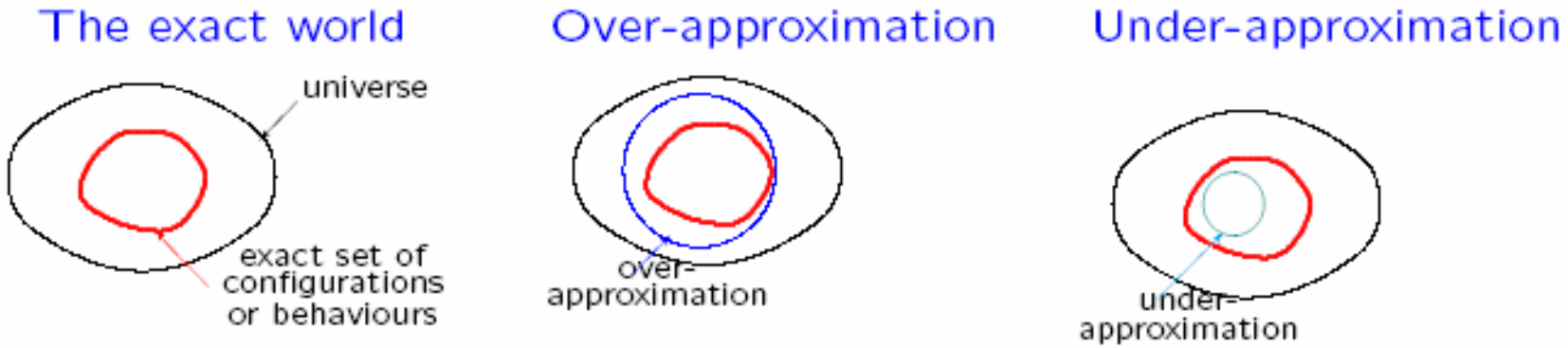
- DFA
  - Conventional dataflow analysis
- Deductive methods
  - The proof size is exponential in the program size!
- Model checking
- Constraint-based analysis
  - Constraint setup and solving
- Type-based analysis
  - Type and effect system
- What else ?

Abstract Interpretation

Abstract interpretation요약 해석

- [Cousot&Cousot77,79]
  - A theory of the approximation of the execution behavior of programs.
- Objective
  - By effective computation of the abstract semantics, it analyzes the behavior of programs before and without executing them.
- 한글로
  - 프로그램 실행 전에 실행하지 않고 그 프로그램의 실행(의미)의 요약본 계산

The Nature of Approximation



- If the approximation is rough enough,
  the abstract semantics is less precise but
  is effectively computable by a computer;

- Because of the information loss,
  not all questions can be definitely answered;

- All answers given by the abstract semantics are always
  correct(sound) with respect to the concrete semantics.

Computable approximation

- 요약(abstract or approximation)이 필요한 이유는 ?
  - 요약 없이는 분석이 끝이 없다
  - 요약 없이 실행해 보면서 모든 경우를 포섭할 수 없다.

- 요약(abstraction) α 예
  - {2,4,6,8, …} → 짝수
  - {-8, -4, -2} → 음수
  - {2, 15, 12, 8} → [2..15]

Example: Sign Analysis

- Source language

  e ::= i | e * e | -e | e + e

- Concrete (execution) semantics

  V : e → Int

  V(i) = i

  V(e1*e2) = V(e1) × V(e2)

  V(-e) = -V(e)

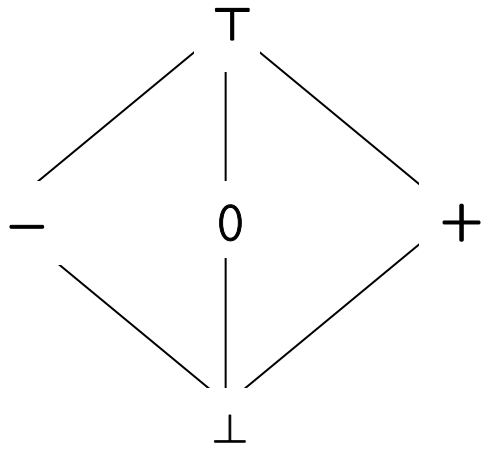
  V(e1+e2) = V(e1) + V(e2)

Sign Analysis

- Example
  - e = -45 * 30 + 23 * -50
  - V(e) = -2500
  - AV(e) = -
- Abstract semantics for sign analysis
  - Define Abstract domain
  - Define Abstract execution

Sign Analysis

- Abstract Domain: Lattice $\hat{A}$



$$2^{\mathbb{Z}} \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \hat{A}$$

- What are $\top$ and $\bot$?

- What are α and γ ?
  - α(abstraction)는 실제 값들을 요약하고
  - γ(concretization) 는 요약된 원소의 실제 의미를 정의한다.

Abstract semantics

- Abstract semantics

  $AV : e \rightarrow \{\bot, -, 0, +, \top\}$

  $$AV(i) = \begin{cases} + & \text{if } i > 0 \\ 0 & \text{if } i = 0 \\ - & \text{if } i < 0 \end{cases}$$

  $AV(-e) = \hat{-} AV(e)$

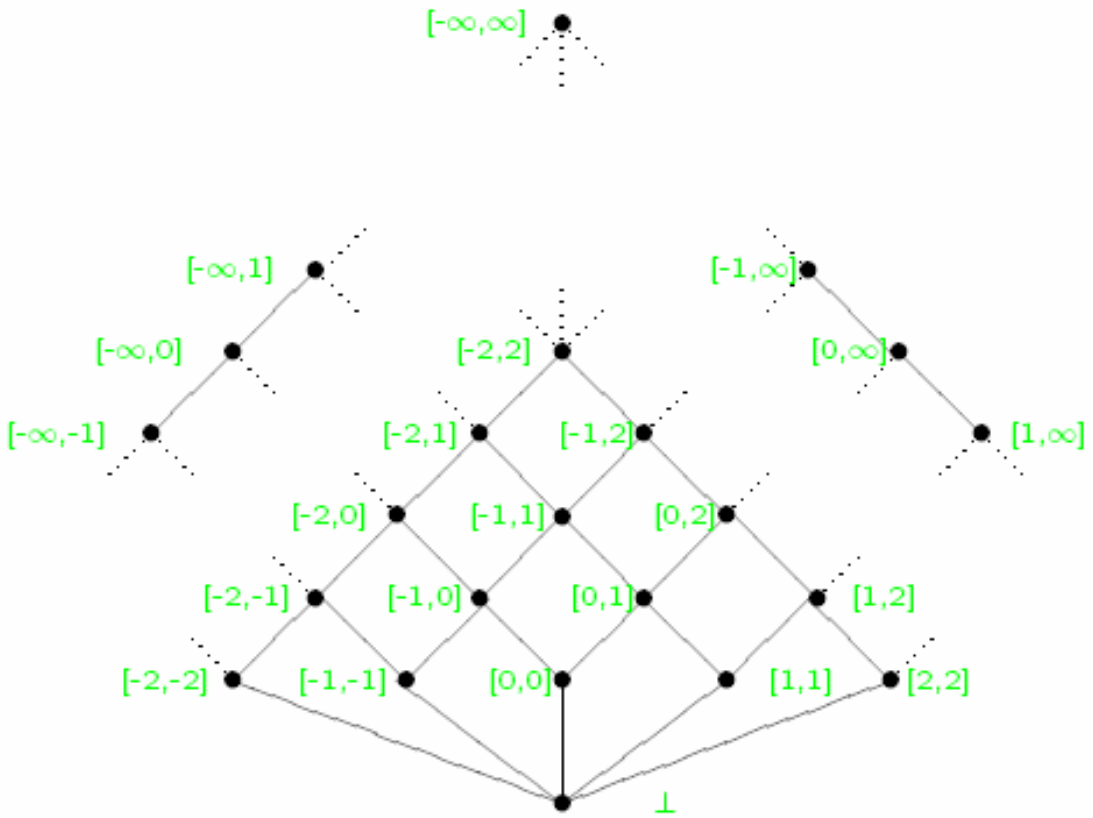
  $AV(e1+e2) = AV(e1) \hat{+} AV(e2)$

  $AV(e1*e2) = AV(e1) \hat{\times} AV(e2)$

- Abstract operator $\hat{a} \hat{+} \hat{b} = ?$
  - $+ \hat{+} + =$
  - $+ \hat{+} - =$
  - $\top \hat{+} - =$

Example: Interval Analysis

The complete lattice **Interval** = (**Interval**, $\sqsubseteq$)



Interval analysis

- Define an abstract semantics that
  - can approximate the concrete (execution) semantics.
- Abstract semantics for interval analysis
  - Interval to approximate concrete values of variables
  - Equation to approximate concrete execution flows.
- Program analysis
  - By computing the least fixpoint of the equation.

Example: interval analysis (1975) [5]

Program to be analyzed:

```
   x := 1;
1:
   while x < 10000 do
2:
        x := x + 1
3:
   od;
4:
```

[5] P. Cousot & R. Cousot, ISOP'76.

Example: interval analysis (1975) [5]

Equations (abstract interpretation of the semantics):

```
   x := 1;
1:
   while x < 10000 do
2:
      x := x + 1
3:
   od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Increasing chaotic iteration, initialization:

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = \emptyset \\ X_2 = \emptyset \\ X_3 = \emptyset \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Increasing chaotic iteration:

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = \emptyset \\ X_3 = \emptyset \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Increasing chaotic iteration:

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1,1] \\ X_3 = \emptyset \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Increasing chaotic iteration:

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1,1] \\ X_3 = [2,2] \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Increasing chaotic iteration:

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1,2] \\ X_3 = [2,2] \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Increasing chaotic iteration: **convergence?**

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1,2] \\ X_3 = [2,3] \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Increasing chaotic iteration: **convergence??**

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1,3] \\ X_3 = [2,3] \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Increasing chaotic iteration: **convergence???**

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1,3] \\ X_3 = [2,4] \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Increasing chaotic iteration: **convergence????**

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1,4] \\ X_3 = [2,4] \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Increasing chaotic iteration: **convergence?????**

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1,4] \\ X_3 = [2,5] \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Increasing chaotic iteration: **convergence??????**

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1,5] \\ X_3 = [2,5] \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Increasing chaotic iteration: **convergence???????**

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1,5] \\ X_3 = [2,6] \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Convergence speed-up by extrapolation:

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1,+\infty] \quad \Leftarrow \text{widening} \\ X_3 = [2,6] \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Decreasing chaotic iteration:

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1, +\infty] \\ X_3 = [2, +\infty] \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Decreasing chaotic iteration:

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1,9999] \\ X_3 = [2,+\infty] \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Decreasing chaotic iteration:

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1,9999] \\ X_3 = [2,+10000] \\ X_4 = \emptyset \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Final solution:

```
    x := 1;
1:
    while x < 10000 do
2:
        x := x + 1
3:
    od;
4:
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1,9999] \\ X_3 = [2,+10000] \\ X_4 = [+10000,+10000] \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Result of the interval analysis:

```
    x := 1;
1: {x = 1}
    while x < 10000 do
2: {x ∈ [1, 9999]}
        x := x + 1
3: {x ∈ [2, +10000]}
    od;
4: {x = 10000}
```

$$\begin{cases} X_1 = [1,1] \\ X_2 = (X_1 \cup X_3) \cap [-\infty, 9999] \\ X_3 = X_2 \oplus [1,1] \\ X_4 = (X_1 \cup X_3) \cap [10000, +\infty] \end{cases}$$

$$\begin{cases} X_1 = [1,1] \\ X_2 = [1,9999] \\ X_3 = [2,+10000] \\ X_4 = [+10000,+10000] \end{cases}$$

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

Example: interval analysis (1975) [5]

Exploitation of the result of the interval analysis:

```
    x := 1;
1: {x = 1}
    while x < 10000 do
2: {x ∈ [1, 9999]}
        x := x + 1          ⟵ no overflow
3: {x ∈ [2, +10000]}
    od;
4: {x = 10000}
```

[5] P. Cousot & R. Cousot, ISOP'1976, POPL'77.

요약 해석으로 정적 분석

1. 실제 실행 정의(Concrete semantics)
2. 요약 실행 정의(Abstract semantics)
3. 올바른 요약 실행 정의인지 확인
4. 정의된 요약 실행을 계산하는 방법

Concrete Semantics실제 실행

- The concrete (collecting) semantics
  - collects the set of traces(or states) that can reach a given program point
  - may be uncomputable.

- Concrete semantic domain

  CPO (D, ⊑, ⊔)

- Concrete semantic function

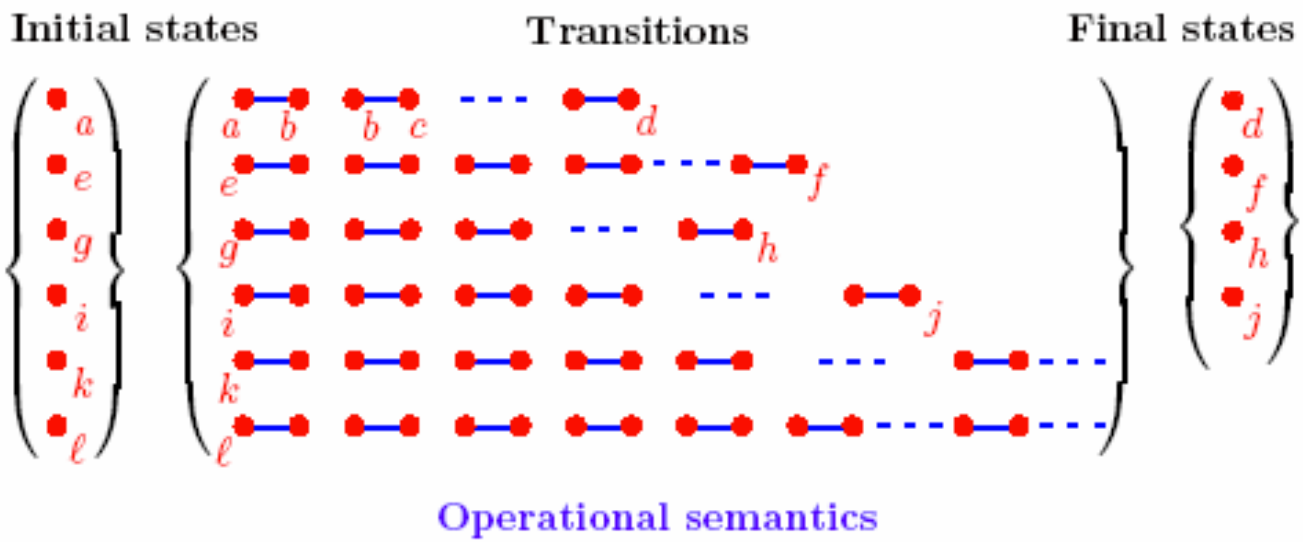
$$F \in D \to D$$

- Concrete semantics

  Least fixpoint of a continuous function $F \in D \to D$

$$lfpF = \bigsqcup_{i \in \mathbb{N}} F^i(\bot_D)$$

Concrete Semantics실제 실행



요약 해석으로 정적 분석

1. 실제 실행 정의(Concrete semantics)
2. 요약 실행 정의(Abstract semantics)
3. 올바른 요약 실행 정의인지 확인
4. 정의된 요약 실행을 계산하는 방법

Abstract Semantics요약 실행

- Abstract semantic domain CPO ($\hat{D}$, ⊑, ⊔)
  - Galois connection $D \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \hat{D}$

- Abstract semantic function
  - Monotone function $\hat{F} \in \hat{D} \to \hat{D}$

$$\forall \hat{x}, \hat{y} \in \hat{D} : x \sqsubseteq y \Rightarrow \hat{F}(x) \sqsubseteq \hat{F}(y)$$

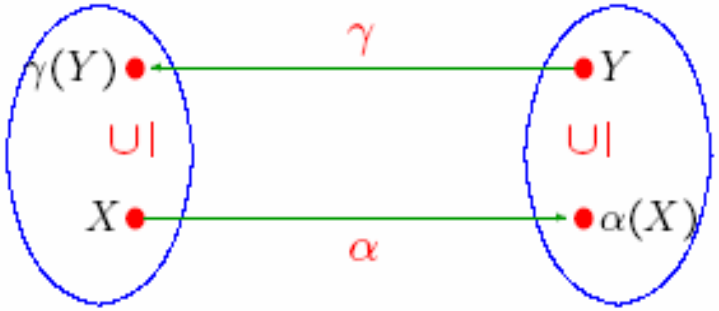
  - Approximation

$$\alpha \circ F \sqsubseteq \hat{F} \circ \alpha, \quad \text{or} \quad F \circ \gamma \sqsubseteq \gamma \circ \hat{F}$$

Galois connection갈로아 연결

- A Galois connection between two sets is a pair of (α, γ) of functions between the sets satisfying

$$X \subseteq \gamma(Y) \quad \Leftrightarrow \quad \alpha(X) \subseteq Y$$



α: abstraction function
γ: concretisation function

Concrete domain D Abstract domain $\hat{D}$

- Abstract values respects the order in the concrete domain

Abstract semantics요약 실행

- Abstract semantics

$$lfp\ \hat{F} = \bigsqcup_{i \in \mathbb{N}} \hat{F}^i(\hat{\perp})$$

- [Theorem]

  The abstract semantics is an approximation of the concrete semantics.

$$\alpha(lfpF) \sqsubseteq \bigsqcup_{i \in \mathbb{N}} \hat{F}^i(\hat{\perp}).$$

- Abstract semantics may be computable in finite time or uncomputable.

요약 해석으로 정적 분석

1. 실제 실행 정의(Concrete semantics)
2. 요약 실행 정의(Abstract semantics)
3. 올바른 요약 실행 정의인지 확인
4. 정의된 요약 실행을 계산하는 방법

Program Analysis

- Program analysis by
  - computing the abstract semantics

    $lfp\ \hat{F} = \bigsqcup_{i \in \mathbb{N}} \hat{F}^i(\hat{\perp})$ if it is computable in reasonable time

  - or, computing an upper bound $\hat{A}$ of the abstract semantics in finite time

  - then it approximates the concrete execution

$$\alpha(lfpF) \sqsubseteq \hat{A},$$
$$lfpF \sqsubseteq \gamma\hat{A}$$

Widening/Narrowing

- Compute the upperbound by widening and

$$\hat{X}_0 = \hat{\bot}$$

$$\hat{X}_{i+1} = \begin{cases} \hat{X}_i & \text{if } \hat{F}(\hat{X}_i) \sqsubseteq \hat{X}_i \\ \hat{X}_i \bigtriangledown \hat{F}(\hat{X}_i) & \text{otherwise} \end{cases}$$

- Narrowing
  - If the upperbound is $\hat{\mathcal{A}} \overset{\text{let}}{=} \lim_{i \in \mathbb{N}}(\hat{X}_i)$,
  - refine the upperbound $\hat{\mathcal{A}}$ by narrowing

$$\hat{Y}_0 = \hat{\mathcal{A}}$$

$$\hat{Y}_{i+1} = \hat{Y}_i \bigtriangleup \hat{F}(\hat{Y}_i)$$

Applications

응용 사례

- A. Deutsch uses abstract interpretation (including interval analysis) for the static analysis of the embedded ADA software of the Ariane 5 launcher [6];
- Automatic detection of the definiteness, potentiality, impossibility or inaccessibility of run-time errors [7];
- Success for the 502 & 503 flights and the ARD [8].

---

6 Flight software (60,000 lines of Ada code) and Inertial Measurement Unit (30,000 lines of Ada code).

7 such as scalar and floating-point overflows, array index errors, divisions by zero and related arithmetic exceptions, uninitialized variables, data races on shared data structures, etc.

8 Atmospheric Reentry Demonstrator: module coming back to earth.

응용 분야

- program transformation & optimization;
- abstract model-checking of infinite systems;
- abstract testing;
- type inference (for undecidable systems);
- mobile code communication topology;

상업화

- First research results: 1975;
- First industrializations:
  -  Connected Components Corporation (U.S.A.), L. Harrison, 1993;
  -  AbsInt Angewandte Informatik GmbH (Germany), R. Wilhelm, 1998;
  -  Polyspace Technologies (France), A. Deutsch & D. Pilaud, 1999.
- Domestic industrializations:
  - Sparrow
    K. Yi, 2007

참고문헌

- Cousot&Cousot
  - [Cousot&Cousot 77, 79]
  - [Cousot&Cousot 92]
  - [Cousot&Cousot 00]
  - …
- 이광근
  - 프로그래밍언어 이야기 I, 2004-2005
  - 관련 논문 리스트
    http://ropas.snu.ac.kr/~kwang/4541.664A/07/papers.html
- Nielson, Nielson and Hankin
  - Principles of Program Analysis, Springer-Verlag.