Introduction to Program Analysis

5. Abstract Interpretation

Kihong Heo

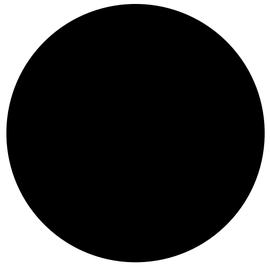
KAIST

Abstract Interpretation

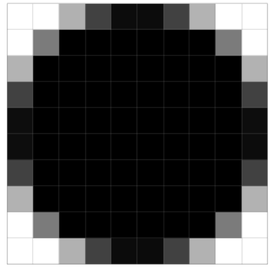
- A **powerful framework** for designing **correct** static analysis
  - Framework: given some inputs, a static analysis comes out
  - Powerful: all static analyses are understood in this framework
    (e.g., type systems, data-flow analysis, etc)
  - Correct: mathematically proven
- Estcabilished by Patrick and Radhia Cousot
  - *Abstract Interpretation: A Unified Lattice Model for Static Analysis of Programs by Construction or Approximation of Fixpoints*, 1977
  - *Systematic Design of Program Analysis Frameworks*, 1979

Abstract?

- Concrete (execution, dynamic) vs Abstract (analysis, static)
- Without abstraction, it is undecidable to subsume all possible behavior of SW
  - Recall the Rice's theorem and the Halting problem



Concrete



Abstract

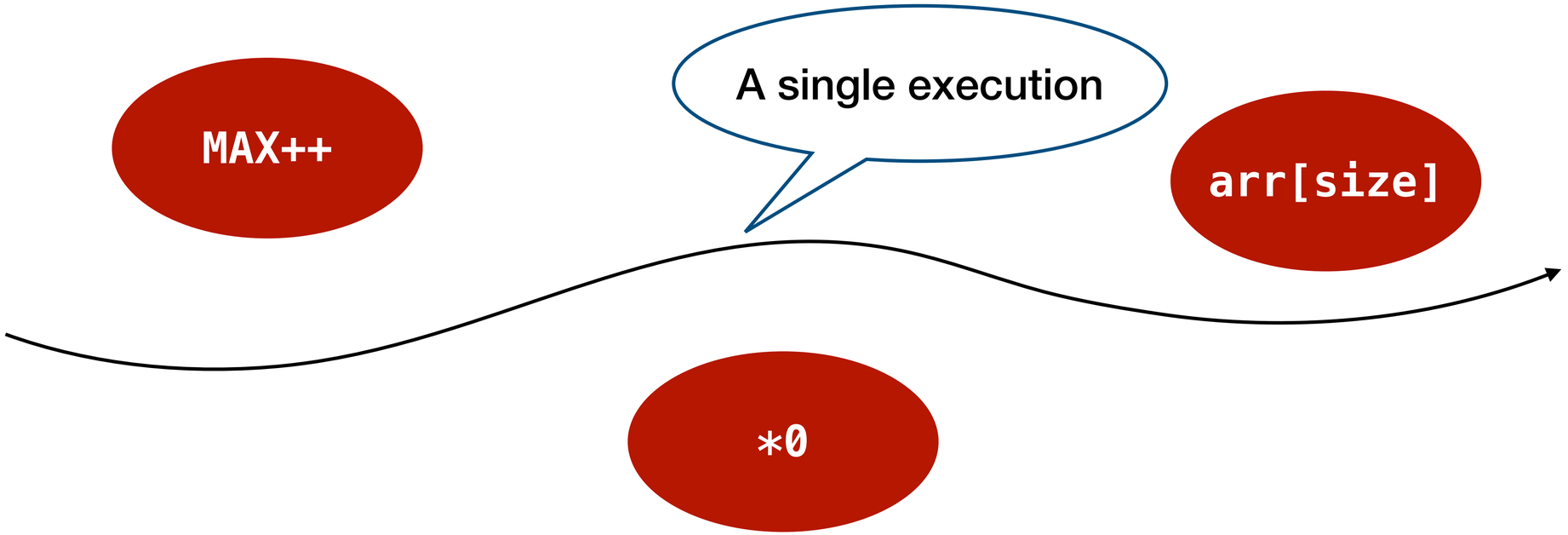
Example

```
x = 3;
while (*) {
  x += 2;
}
x -= 1;
print(x);
```

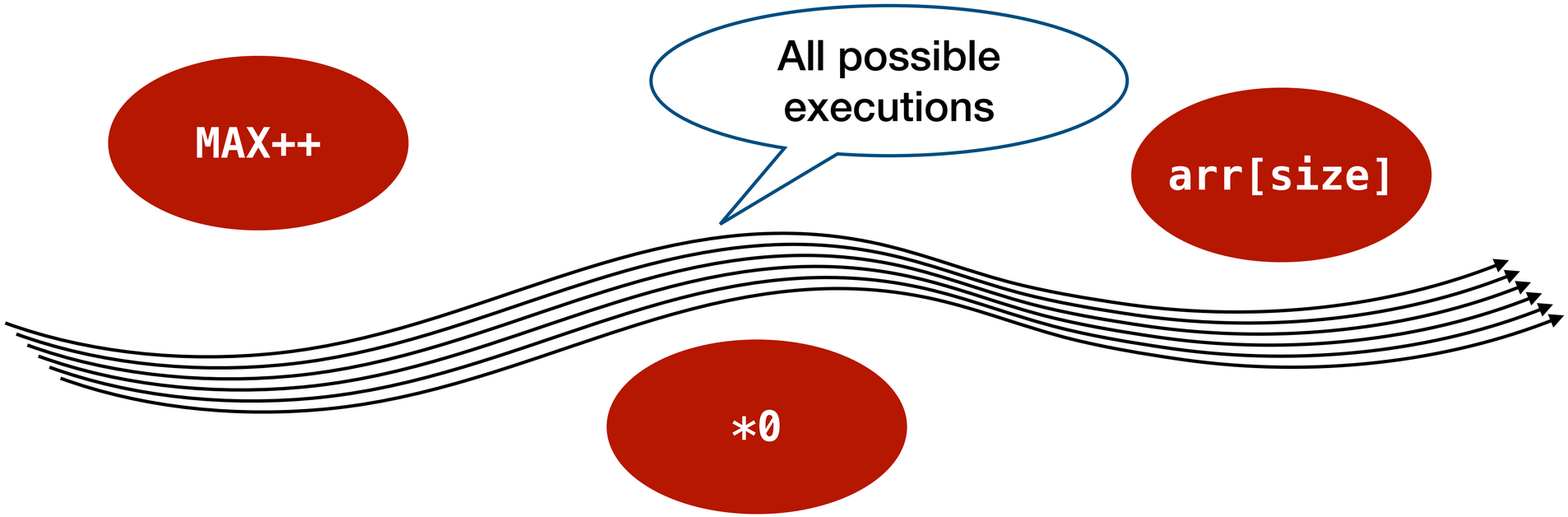
Q: What are the possible output values?

- Concrete interpretation : 2, 4, …, uncomputable (infinitely many possibilities)
- Abstract interpretation 1 : “integers” (good)
- Abstract interpretation 2 : “positive integers” (better)
- Abstract interpretation 3 : “positive even integers” (best)

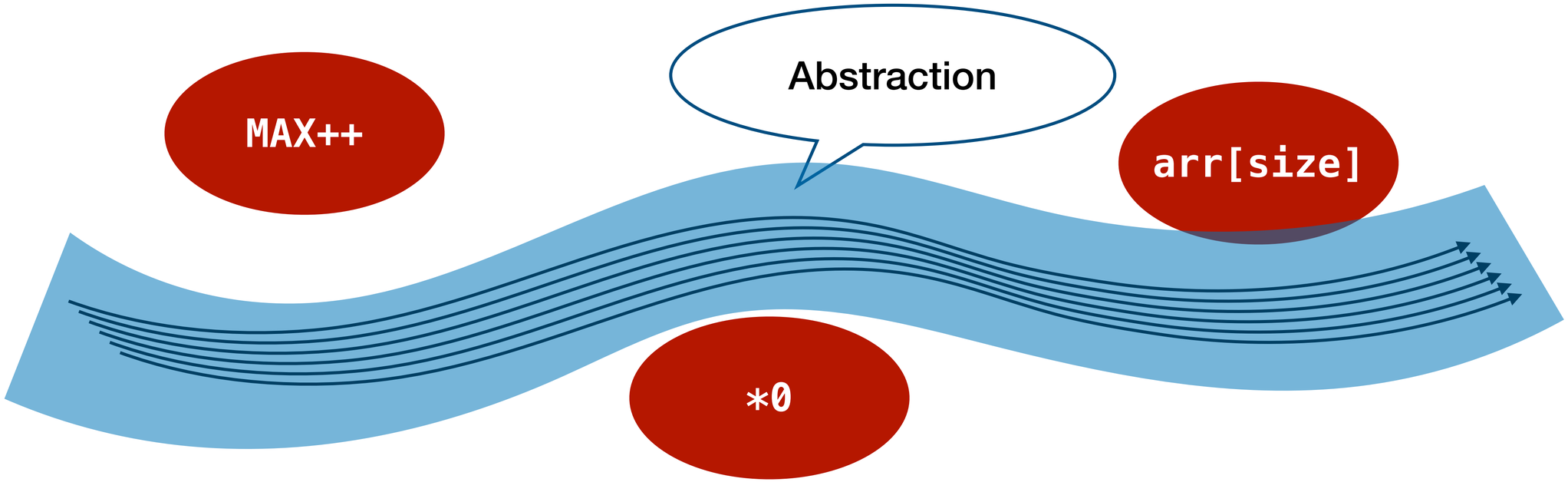
Abstraction of Executions



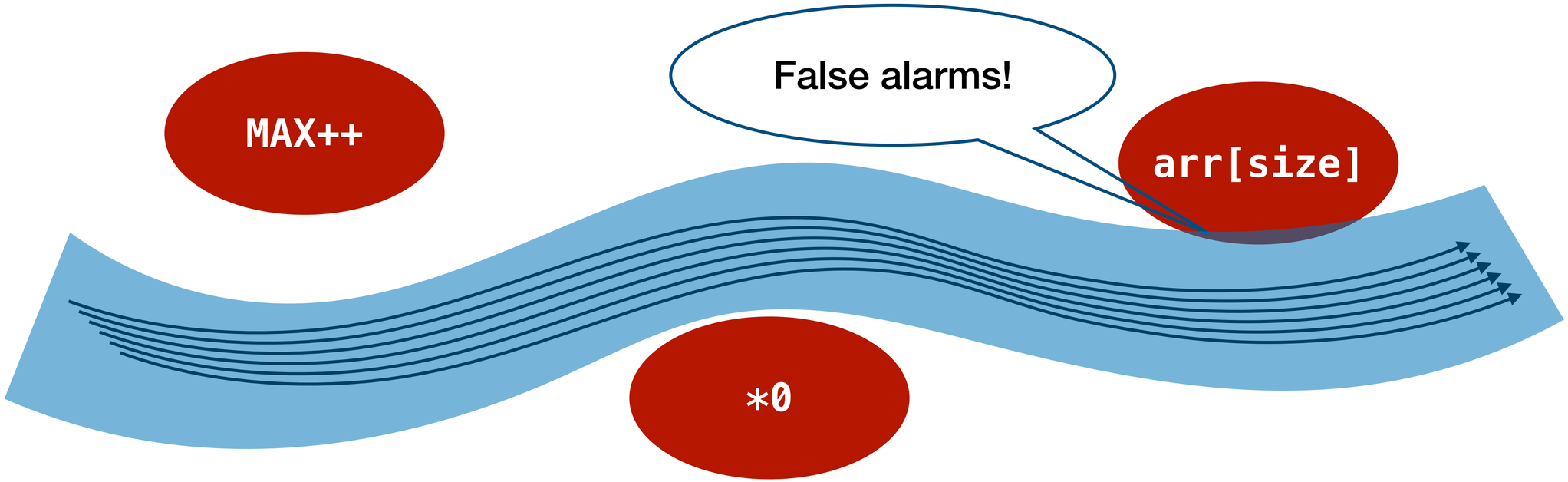
Abstraction of Executions



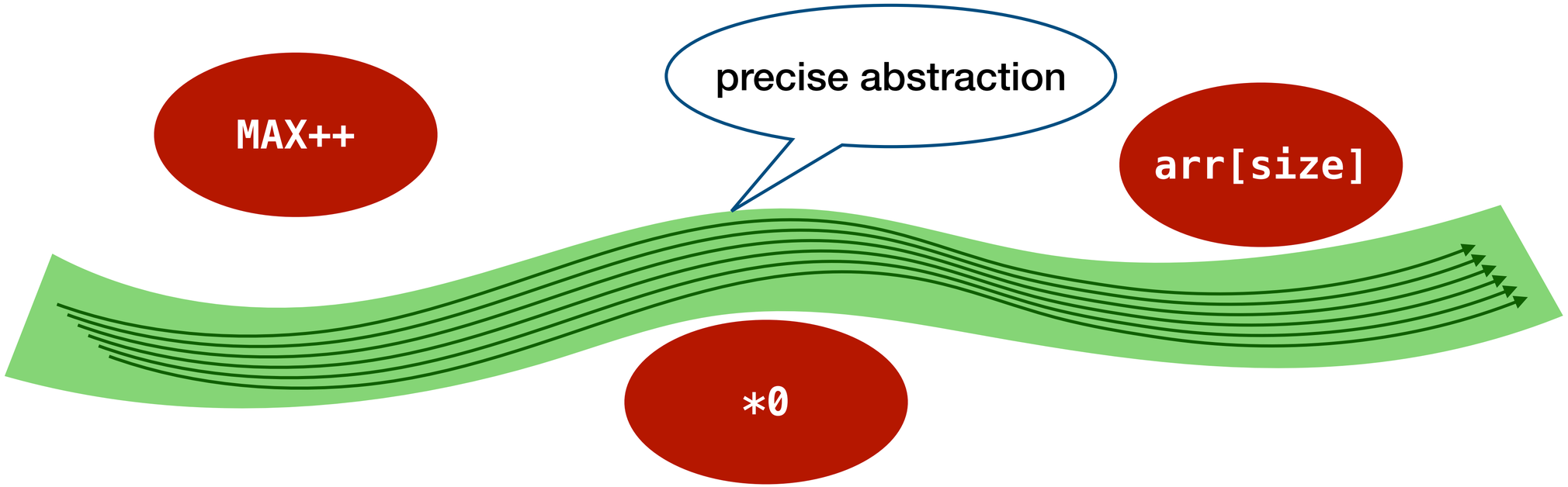
Abstraction of Executions



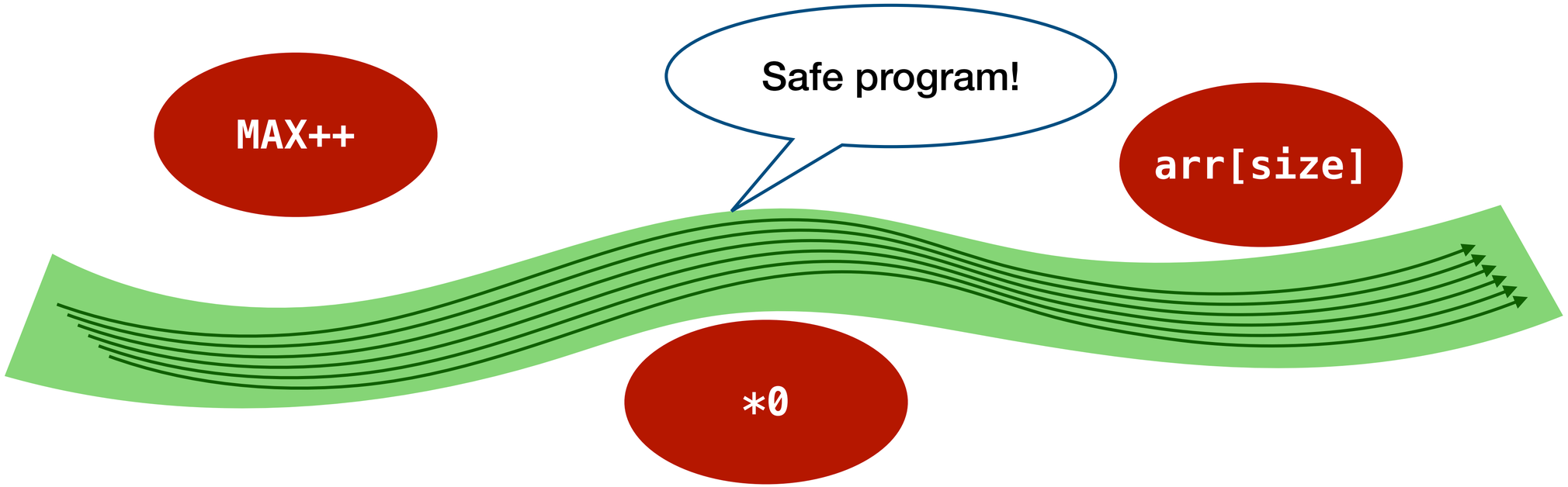
Abstraction of Executions



Abstraction of Executions



Abstraction of Executions



How to analyze?

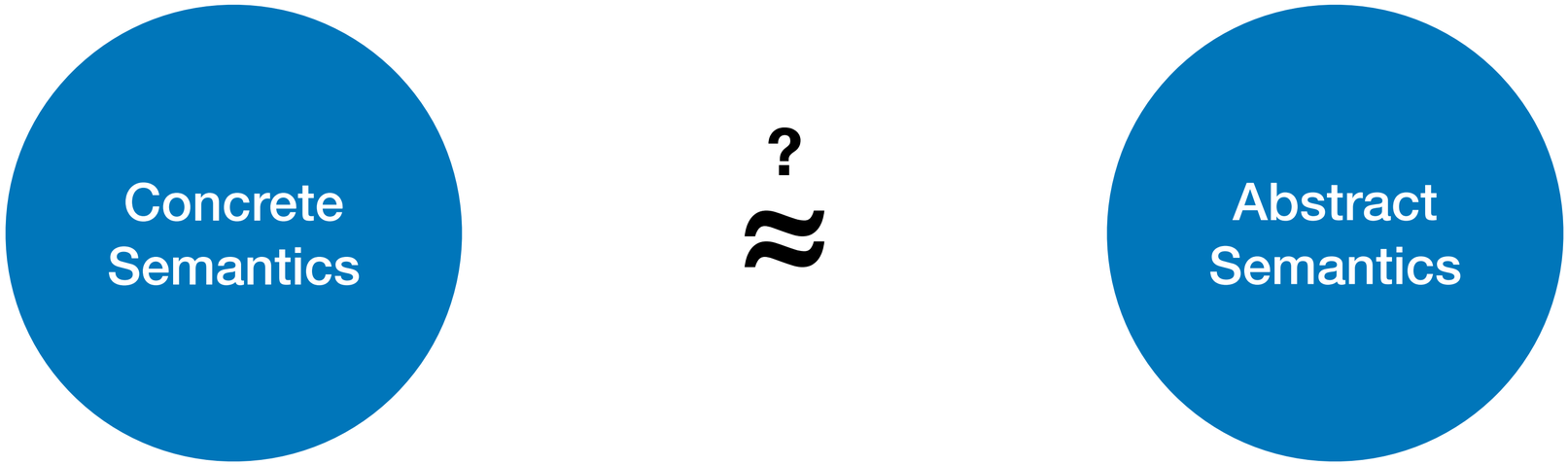
- Interpret the target program
  - with abstract semantics (= analyzer's concern)
  - not concrete semantics (= interpreter's and compiler's concern)

- Example

```
x = 3;
while (*) {
  x += 2;
}
x -= 1;
print(x);
```

Concrete	Abstract 1	Abstract 2	Abstract 3
{3}	Int	Pos	PosOdd
{3, 5, 7, ...}	Int	Pos	PosOdd
{2, 4, 6, ...}	Int	Pos	PosEven

Principles



- How to guarantee soundness?
- How to guarantee termination?
- How to design more precise abstraction?
- How to compute abstract semantics?

Practice

- Guidance for a lot of design choices in practice such as
  - Soundness vs Scalability vs Precision vs Usability vs …
  - Characteristics of target programs and properties
  - Optimizations of program analyzers

Abstract Interpretation Framework

- Abstract interpretation concerns
  - Concrete semantics: $[\![C]\!] = \mathsf{lfp} F \in \mathbb{D}$
  - Abstract semantics: $[\![C]\!]^\sharp = \bigsqcup_{i \geq 0} F^{\sharp i}(\bot) \in \mathbb{D}^\sharp$
- Requirements:
  - Relationship between $\mathbb{D}$ and $\mathbb{D}^\sharp$
  - Relationship between $F \in \mathbb{D} \to \mathbb{D}$ and $F^\sharp \in \mathbb{D}^\sharp \to \mathbb{D}^\sharp$
- Guarantees:
  - Correctness (soundness): $[\![C]\!] \approx [\![C]\!]^\sharp$
  - Computability: $[\![C]\!]^\sharp$ is computable within finite time

Design of Static Analysis

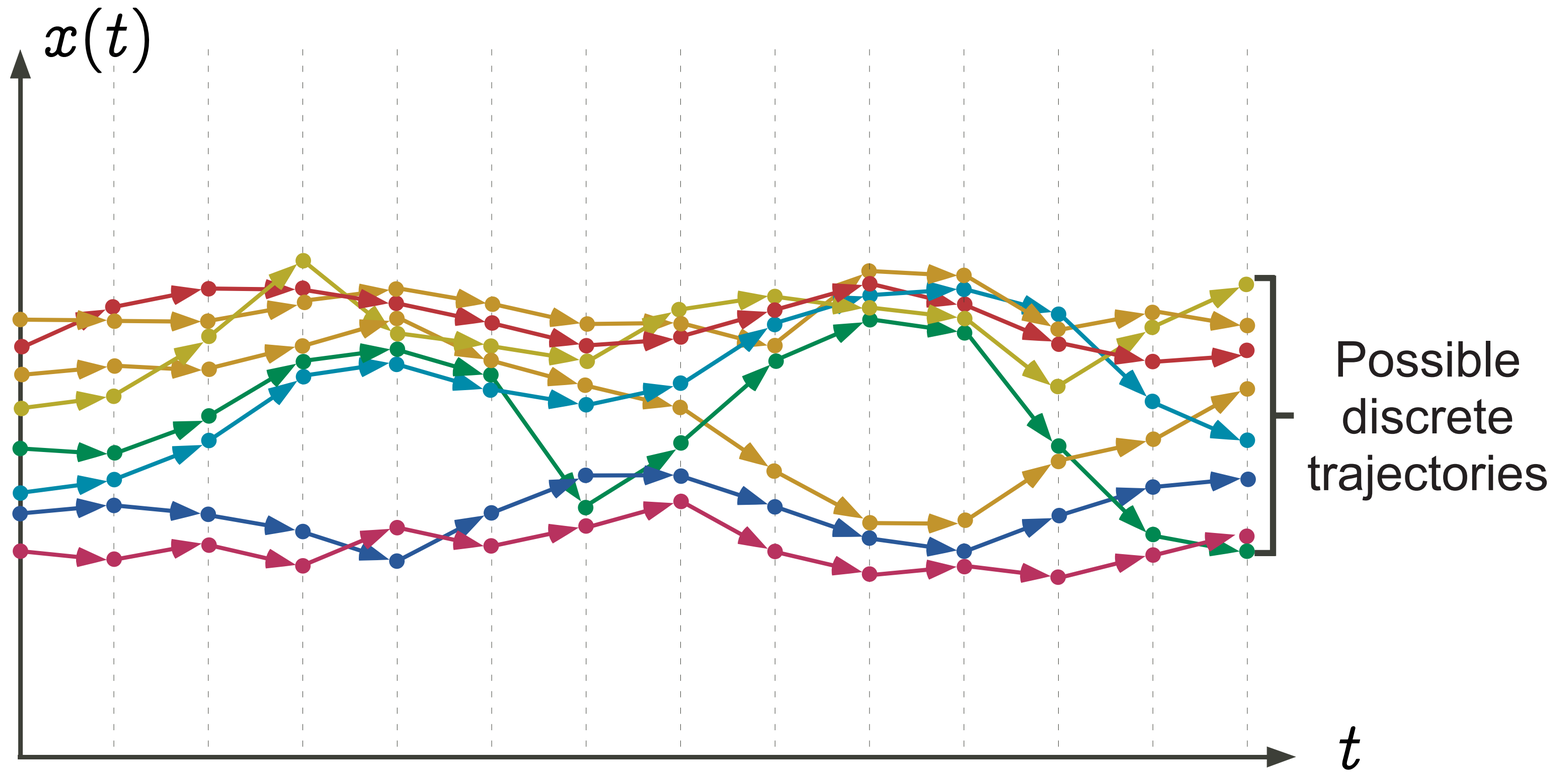
- Goal: conservative and terminating static analysis
- Design principles:
  - Define concrete semantics
  - Define abstract semantics (sound w.r.t the concrete semantics)
- Computation & implementation:
  - Abstract semantics of a program: the least fixed point of the semantic function
  - Static analyzer: compute the least fixed point within finite time

Define Standard Semantics

- Formalization of a **single program execution**
  - Recall Lecture 2 and 3 (operational and denotation semantics)
- What to describe: different choices depending on the purpose
  - E.g., denotational, operational, etc
- In this lecture, we will use denotational semantics
  - Recall the denotational semantics the simple imperative language

$$[\![C]\!] : \mathbb{M} \to \mathbb{M}$$

Define Standard Semantics



*from Patrick Cousot's slides

Standard Semantics

- Define a semantic domain $\mathbb{D} = \mathbb{M} \to \mathbb{M}$ (CPO)
- Define a semantic function $F : \mathbb{D} \to \mathbb{D}$ (continuous)
- Semantics of a program: the least fixed point of F

$$\mathbf{lfp}F = \bigsqcup_{i \geq 0} F^i(\bot)$$

Standard Semantics of Commands

$$
\begin{aligned}
[\![C]\!] &: \mathbb{M} \to \mathbb{M} \\
[\![\texttt{skip}]\!] &= \lambda m.m \\
[\![C_0\,\texttt{;}\,C_1]\!] &= \lambda m.[\![C_1]\!]([\![C_0]\!](m)) \\
[\![x\,\texttt{:=}\,E]\!] &= \lambda m.m\{x \mapsto [\![E]\!](m)\} \\
[\![\texttt{input}(x)]\!] &= \lambda m.m\{x \mapsto n\} \\
[\![\texttt{if}\ B\ \texttt{then}\ C_1\ \texttt{else}\ C_2]\!] &= \lambda m. \begin{cases} [\![C_1]\!](m) & \text{if } [\![B]\!](m) = \texttt{true} \\ [\![C_2]\!](m) & \text{if } [\![B]\!](m) = \texttt{false} \end{cases} \\
[\![\texttt{while}\ B\ C]\!] &= \mathbf{lfp}\lambda X. \left(\lambda m. \begin{cases} X([\![C]\!](m)) & \text{if } [\![B]\!](m) = \texttt{true} \\ m & \text{if } [\![B]\!](m) = \texttt{false} \end{cases} \right)
\end{aligned}
$$

Standard Semantics of Programs

- What is the semantic function F?
  - Straightforward from the definition of $[\![C]\!]$ (because of compositionally!)

$$F : (\mathbb{M} \to \mathbb{M}) \to (\mathbb{M} \to \mathbb{M})$$
$$F = \lambda X.[\![C]\!]$$

- Then, the semantics of a program C

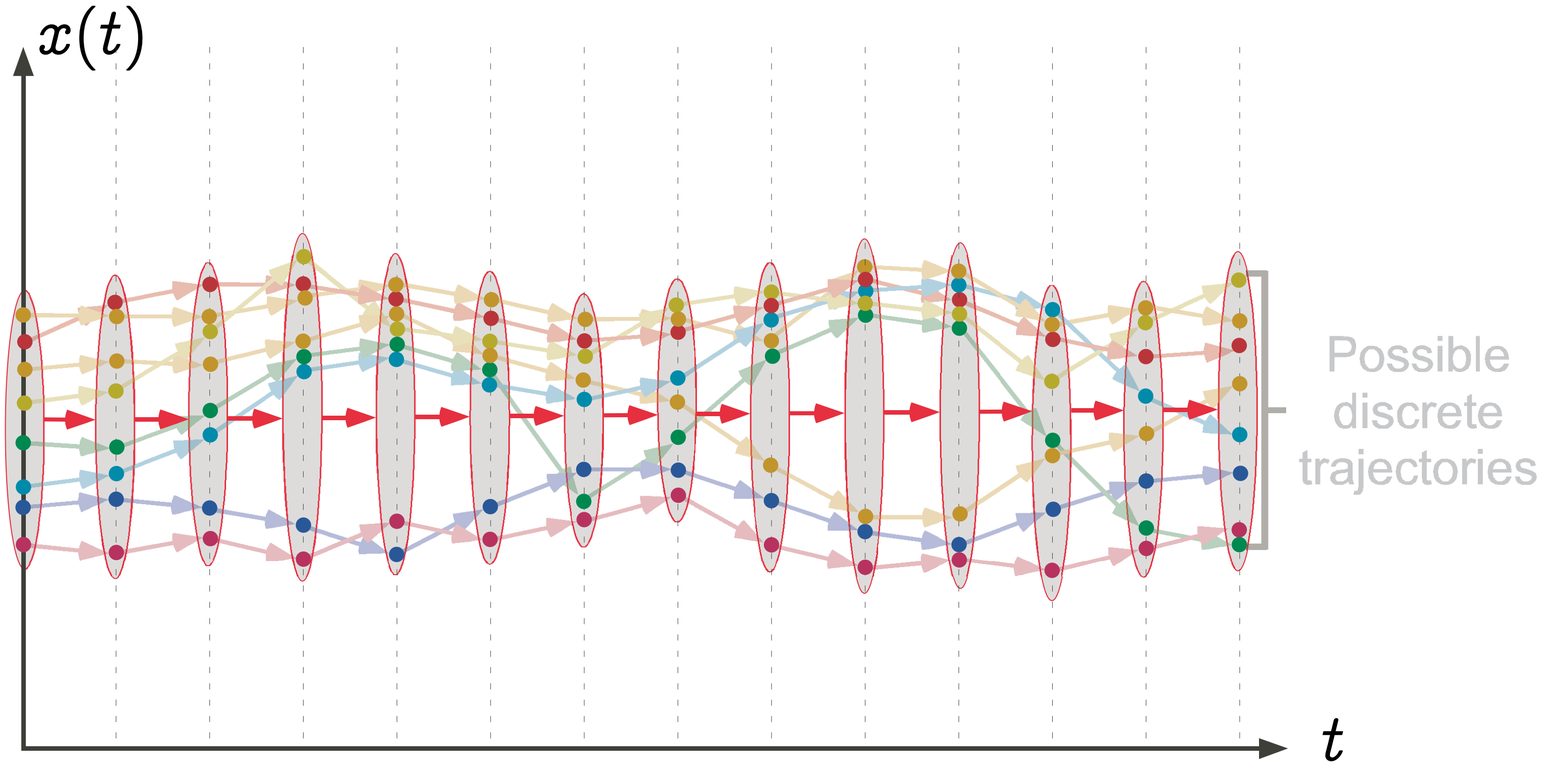
$$\mathbf{lfp}F : \mathbb{M} \to \mathbb{M}$$
$$\mathbf{lfp}F = [\![C]\!]$$

- In this lecture, we will consider only $[\![C]\!]$

Define Concrete Semantics

- Formalization of **all possible** program executions
  - So-called collecting semantics
  - Usually a simple extension of the standard semantics
- What to describe: different choices depending on the purposes (recall, property)
  - Some are more expressive than others
  - E.g., traces (sequence of states), reachable states (set of states), etc
- In this lecture, we will use reachable states for concrete semantics

Transitions of Sets of States

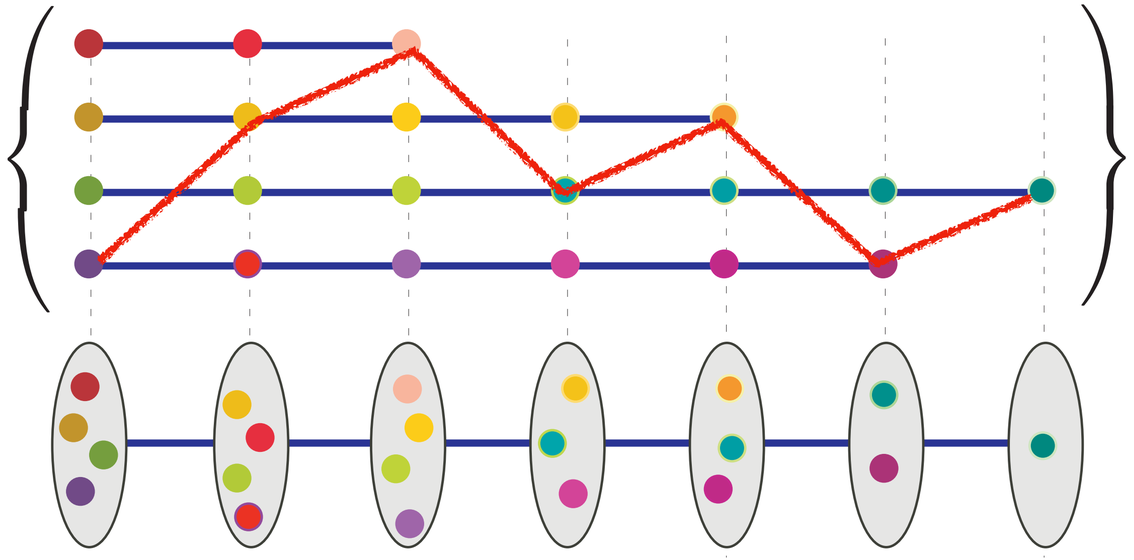


*from Patrick Cousot's slides

Traces vs Reachable States

Traces

Reachable States



Can Answer:

- Can variable `p` be `NULL` at line 10?
- Can buffer index `i` be larger than size `s`?
- ...

Can't Answer:

- Does the red trace exist?
- ...

*from Patrick Cousot's slides

Concrete Semantics

- Define a concrete domain $\mathbb{D}$ (CPO)
- Define a semantic function $F : \mathbb{D} \to \mathbb{D}$ (continuous)
- Then the concrete semantics is defined as the least fixed point of the semantic function F :

$$\mathbf{lfp}F = \bigsqcup_{i \geq 0} F^i(\bot)$$

Example

- Define a concrete semantics of the simple language using denotational semantics
  - Concrete domain $\mathbb{D} = \wp(\mathbb{M}) \to \wp(\mathbb{M})$
  - Define a semantic function $F : \mathbb{D} \to \mathbb{D}$
  - Concrete semantics lfp$F \in \mathbb{D}$
- Q: How to define F?

Concrete Semantics of Expressions

$$
\begin{aligned}
[\![E]\!]_{\wp} &: \wp(\mathbb{M}) \to \wp(\mathbb{Z}) \\
[\![n]\!]_{\wp} &= \lambda M.\{n\} \\
[\![x]\!]_{\wp} &= \lambda M.\{m(x) \mid m \in M\} \\
[\![E_1 \odot E_2]\!]_{\wp} &= \lambda M.\{v_1 \odot v_2 \mid v_1 \in [\![E_1]\!]_{\wp}(M), v_2 \in [\![E_2]\!]_{\wp}(M)\}
\end{aligned}
$$

$$
\begin{aligned}
[\![B]\!]_{\wp} &: \wp(\mathbb{M}) \to \wp(\mathbb{M}) \\
[\![\mathsf{true}]\!]_{\wp} &= \lambda M.M \\
[\![\mathsf{false}]\!]_{\wp} &= \lambda M.\emptyset \\
[\![B_1 \circledcirc B_2]\!]_{\wp} &= \lambda M.\{m \in M \mid [\![B_1]\!](m) \circledcirc [\![B_2]\!](m) = \mathsf{true}\}
\end{aligned}
$$

Concrete Semantics of Commands

$$
\begin{aligned}
[\![C]\!]_{\wp} &: \wp(\mathbb{M}) \to \wp(\mathbb{M}) \\
[\![\texttt{skip}]\!]_{\wp} &= \lambda M.M \\
[\![C_0\,\texttt{;}\,C_1]\!]_{\wp} &= \lambda M.[\![C_1]\!]_{\wp} \circ [\![C_0]\!]_{\wp}(M) \\
[\![x\,\texttt{:=}\,E]\!]_{\wp} &= \lambda M.\{m\{x \mapsto [\![E]\!]_{\wp}(m)\} \mid x \in M\} \\
[\![\texttt{input}(x)]\!]_{\wp} &= \lambda M.\{m\{x \mapsto n\} \mid m \in M, n \in \mathbb{Z}\} \\
[\![\texttt{if}\ B\ \texttt{then}\ C_1\ \texttt{else}\ C_2]\!]_{\wp} &= \lambda M.[\![C_1]\!]_{\wp} \circ [\![B]\!]_{\wp}(M) \cup [\![C_2]\!]_{\wp} \circ [\![\neg B]\!]_{\wp}(M) \\
[\![\texttt{while}\ B\ C]\!]_{\wp} &= \lambda M.[\![\neg B]\!]_{\wp}\big(\mathbf{lfp}\lambda X.M \cup [\![C]\!]_{\wp} \circ [\![B]\!]_{\wp}(X)\big)
\end{aligned}
$$

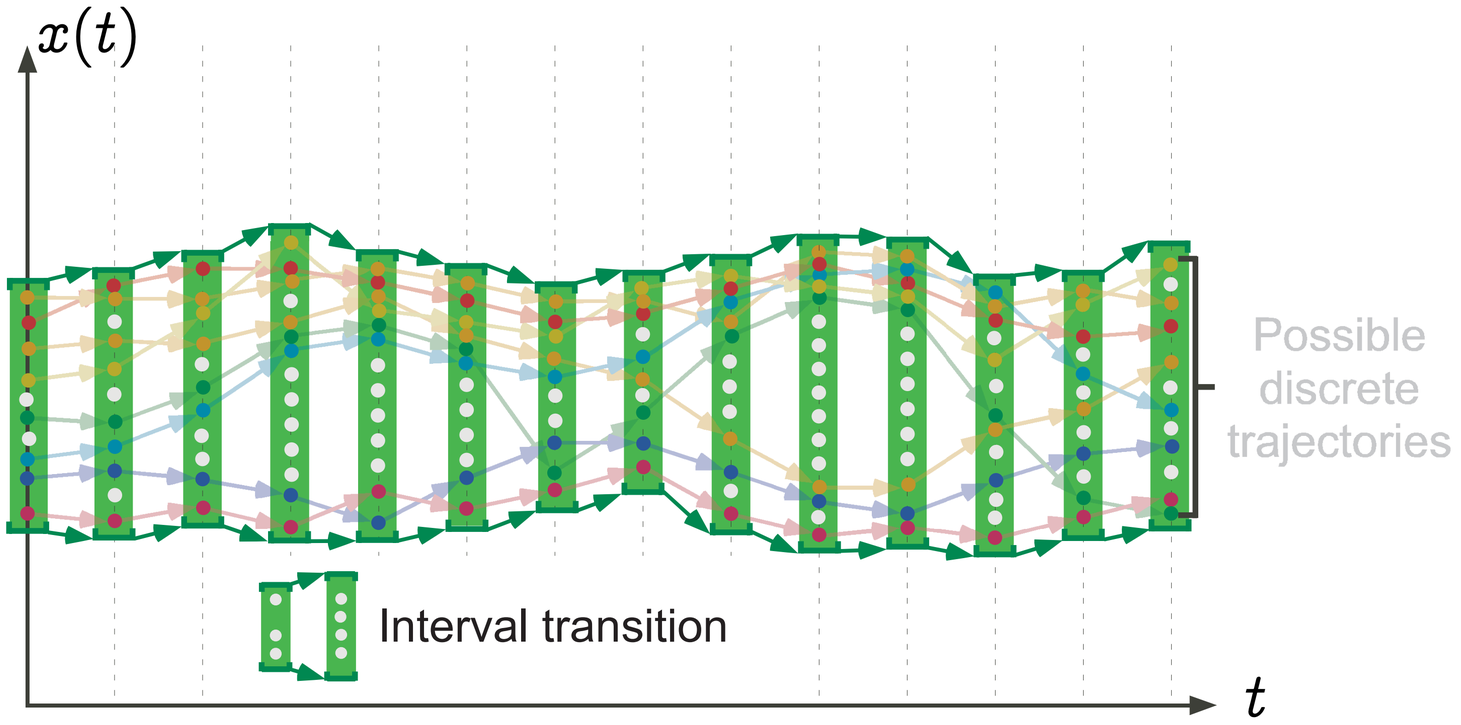
Design of Static Analysis

- Goal: conservative and terminating static analysis
- Design principles:
  - Define concrete semantics
  - Define abstract semantics (sound w.r.t the concrete semantics)
- Computation & implementation:
  - Abstract semantics of a program: the least fixed point of the semantic function
  - Static analyzer: compute the least fixed point within finite time

Step 2: Design Abstract Semantics

- Formalization of **abstract** program executions
  - Soundly subsume concrete executions
- How to subsume: different choices depending on the purposes
  - Some are more expressive than others
- Example: abstraction of {1, 3, 5, 7}
  - Integer, Positive, Odd, [1, 7], etc

Transitions of Abstract States



*from Patrick Cousot's slides

Abstract Semantics

- Define an abstract domain $\mathbb{D}^\sharp$ (CPO)
- Define an abstract semantic function $F^\sharp : \mathbb{D}^\sharp \to \mathbb{D}^\sharp$ (monotone or extensive)

(Monotone) $\forall x^\sharp, y^\sharp \in \mathbb{D}^\sharp.\ x^\sharp \sqsubseteq y^\sharp \implies F^\sharp(x^\sharp) \sqsubseteq F^\sharp(y^\sharp)$

(Extensive) $\forall x^\sharp \in \mathbb{D}.\ x^\sharp \sqsubseteq F^\sharp(x^\sharp)$

- Static analysis is to compute an upper bound of the chain:

$$\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp)$$

Q. How to ensure that the abstract semantics soundly subsume the concrete semantics?

Requirement 1: Galois Connection

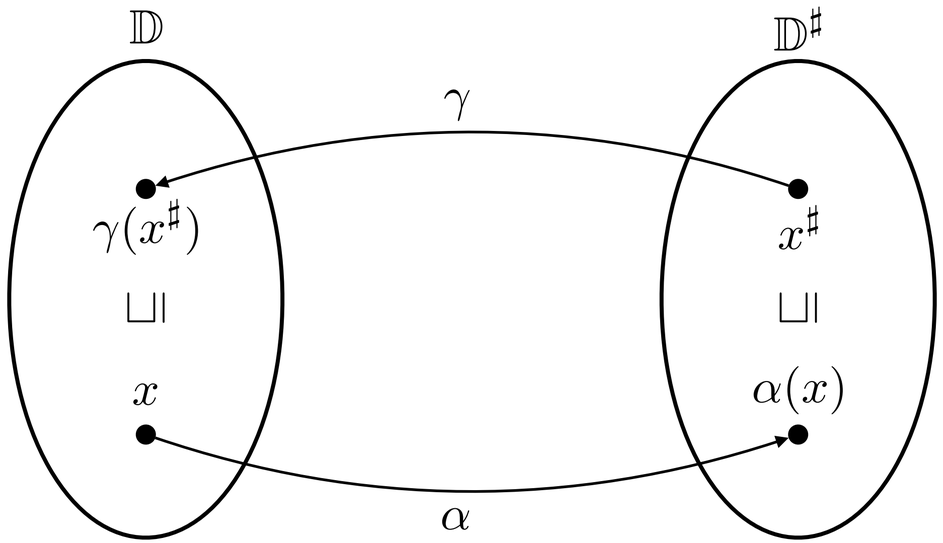
$$\mathbb{D} \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{D}^\sharp$$

- $\mathbb{D}$ and $\mathbb{D}^\sharp$ must be related with a Galois connection where
  - Abstraction function: $\alpha \in \mathbb{D} \to \mathbb{D}^\sharp$
  - Concretization function: $\gamma \in \mathbb{D}^\sharp \to \mathbb{D}$

$$\forall x \in \mathbb{D}, x^\sharp \in \mathbb{D}^\sharp.\ \alpha(x) \sqsubseteq x^\sharp \iff x \sqsubseteq \gamma(x^\sharp)$$

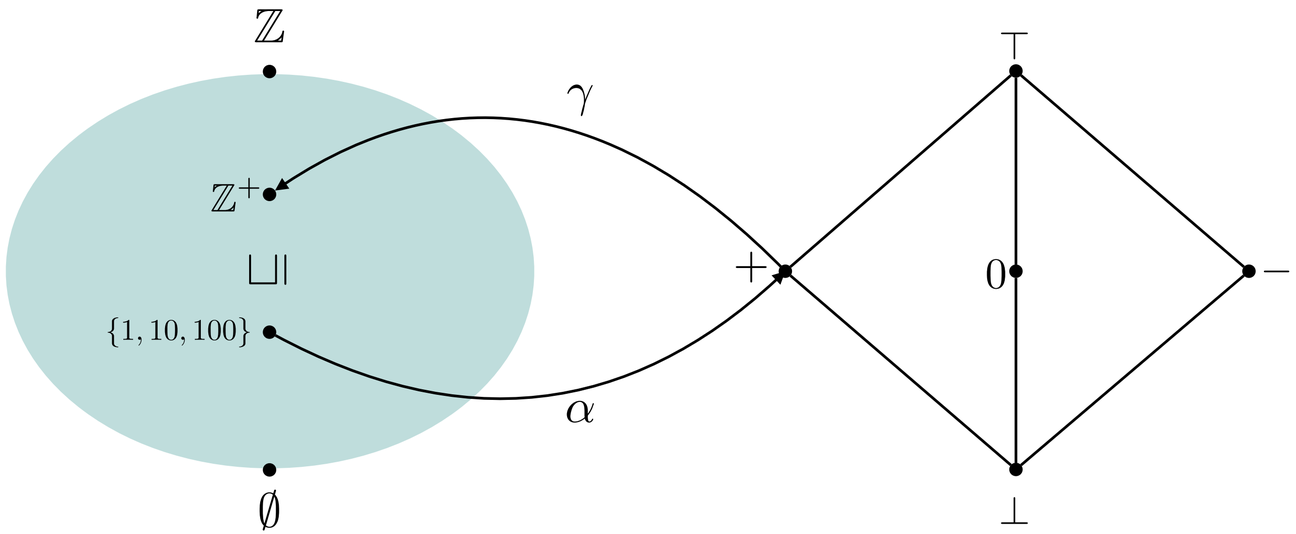
Requirement 1: Galois Connection

- Intuition: order preservation between two semantic domains



Example: Sign Abstraction

$$\wp(\mathbb{Z}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \{\bot, -, 0, +, \top\}$$



Example: Sign Abstraction

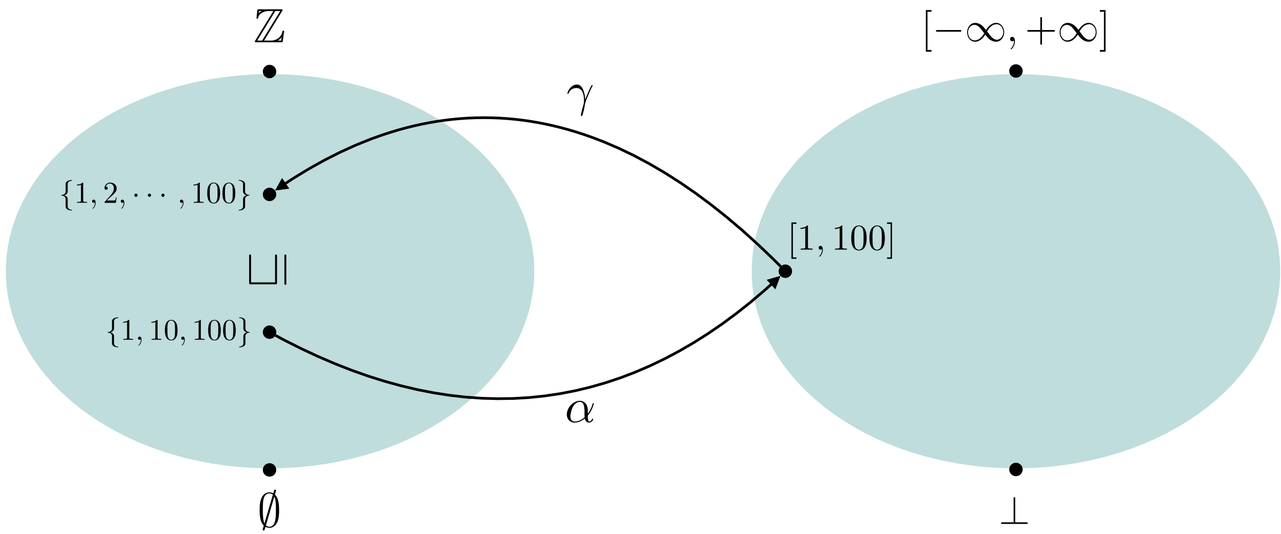
$$\wp(\mathbb{Z}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \{\bot, -, 0, +, \top\}$$

$$\alpha(Z) = \begin{cases} \bot & Z = \emptyset \\ + & \forall z \in Z.\ z > 0 \\ 0 & Z = \{0\} \\ - & \forall z \in Z.\ z < 0 \\ \top & \text{otherwise} \end{cases}$$

$$\gamma(\bot) = \emptyset$$
$$\gamma(+) = \{z \in \mathbb{Z} \mid z > 0\}$$
$$\gamma(0) = \{0\}$$
$$\gamma(-) = \{z \in \mathbb{Z} \mid z < 0\}$$
$$\gamma(\top) = \mathbb{Z}$$

Example: Interval Abstraction

$$\wp(\mathbb{Z}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \{\bot\} \cup \{[a,b] \mid a \in \mathbb{Z} \cup \{-\infty\}, b \in \mathbb{Z} \cup \{+\infty\}\}$$



Example: Interval Abstraction

$$\wp(\mathbb{Z}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \{\bot\} \cup \{[a,b] \mid a \in \mathbb{Z} \cup \{-\infty\}, b \in \mathbb{Z} \cup \{+\infty\}\}$$

$$\begin{aligned}
\alpha(\emptyset) &= \bot \\
\alpha(X) &= [\min X, \max X] \\
\\
\gamma(\bot) &= \emptyset \\
\gamma([a,b]) &= \{x \in \mathbb{Z} \mid a \leq x \leq b\}
\end{aligned}$$

Properties of Galois Connection

$$\forall x \in \mathbb{D}, x^\sharp \in \mathbb{D}^\sharp.\ \alpha(x) \sqsubseteq x^\sharp \iff x \sqsubseteq \gamma(x^\sharp)$$

- $id \sqsubseteq \gamma \circ \alpha$

$$\begin{array}{rrcll} & \alpha(x) & \sqsubseteq & \alpha(x) & \\ \iff & x & \sqsubseteq & \gamma(\alpha(x)) & \text{(by Galois connection)} \end{array}$$

- $\alpha \circ \gamma \sqsubseteq id$

$$\begin{array}{rrcll} & \gamma(x^\sharp) & \sqsubseteq & \gamma(x^\sharp) & \\ \iff & \alpha(\gamma(x^\sharp)) & \sqsubseteq & x^\sharp & \text{(by Galois connection)} \end{array}$$

- α is monotone

$$\begin{array}{rrcll} & x & \sqsubseteq & y & \\ \implies & x & \sqsubseteq & \gamma(\alpha(y)) & (id \sqsubseteq \gamma \circ \alpha) \\ \iff & \alpha(x) & \sqsubseteq & \alpha(y) & \text{(by Galois connection)} \end{array}$$

- γ is monotone

$$\begin{array}{rrcll} & x^\sharp & \sqsubseteq & y^\sharp & \\ \implies & \alpha(\gamma(x^\sharp)) & \sqsubseteq & y^\sharp & (\alpha \circ \gamma \sqsubseteq id) \\ \iff & \gamma(x^\sharp) & \sqsubseteq & \gamma(x^\sharp) & \text{(by Galois connection)} \end{array}$$

Deriving Galois Connections (1)

- Pointwise lifting:

$$\text{Given a Galois connection } \mathbb{D} \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{D}^\sharp \text{ and a set } \mathbb{S}$$

$$\mathbb{S} \to \mathbb{D} \underset{\alpha'}{\overset{\gamma'}{\leftrightarrows}} \mathbb{S} \to \mathbb{D}^\sharp$$

$$\text{where } \alpha'(f) = \lambda x \in \mathbb{S}.\ \alpha(f(x)) \text{ and } \gamma'(f^\sharp) = \lambda x \in \mathbb{S}.\ \gamma(f^\sharp(x))$$

- Composition:

$$\text{Given two galois connections } \mathbb{D}_1 \underset{\alpha_1}{\overset{\gamma_1}{\leftrightarrows}} \mathbb{D}_2 \underset{\alpha_2}{\overset{\gamma_2}{\leftrightarrows}} \mathbb{D}_3$$

$$\mathbb{D}_1 \underset{\alpha_2 \circ \alpha_1}{\overset{\gamma_1 \circ \gamma_2}{\leftrightarrows}} \mathbb{D}_3$$

Deriving Galois Connections (2)

Given two Galois connections $\mathbb{D}_1 \underset{\alpha_1}{\overset{\gamma_1}{\leftrightarrows}} \mathbb{D}_1^\sharp$ and $\mathbb{D}_2 \underset{\alpha_2}{\overset{\gamma_2}{\leftrightarrows}} \mathbb{D}_2^\sharp$

- $\mathbb{D}_1 \times \mathbb{D}_2 \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{D}_1^\sharp \times \mathbb{D}_2^\sharp$ where $\alpha = \lambda\langle x, y\rangle.\langle \alpha_1(x), \alpha_2(y)\rangle$
- $\mathbb{D}_1 + \mathbb{D}_2 \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{D}_1^\sharp + \mathbb{D}_2^\sharp$ where $\alpha = \lambda x. \begin{cases} \alpha_1(x) & \text{if } x \in \mathbb{D}_1 \\ \alpha_2(x) & \text{if } x \in \mathbb{D}_2 \end{cases}$
- $\mathbb{D}_1 \to \mathbb{D}_2 \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{D}_1^\sharp \to \mathbb{D}_2^\sharp$ where $\alpha = \lambda f.\alpha_2 \circ f \circ \gamma_1$

Deriving Galois Connections (3)

Given two Galois connections $\wp(\mathbb{A}) \underset{\alpha_1}{\overset{\gamma_1}{\leftrightarrows}} \mathbb{D}_1^\sharp$ and $\wp(\mathbb{B}) \underset{\alpha_2}{\overset{\gamma_2}{\leftrightarrows}} \mathbb{D}_2^\sharp$

- $\wp(\mathbb{A} \times \mathbb{B}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{D}_1^\sharp \times \mathbb{D}_2^\sharp$ where $\alpha = \lambda X.\langle \alpha_1(\{a \mid \langle a, b\rangle \in X\}), \alpha_2(\{b \mid \langle a, b\rangle \in X\})\rangle$
- $\wp(\mathbb{A} \times \mathbb{B}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{A}' \to \mathbb{D}_2^\sharp$ where $\alpha = \lambda X.\{a \mapsto \alpha_2(S) \mid \langle a, b\rangle \in X, S = \{b \mid \langle a, b\rangle \in X\}\}$ and $\mathbb{A}' \subseteq \mathbb{A}$
- $\wp(\mathbb{A} + \mathbb{B}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{D}_1^\sharp \times \mathbb{D}_2^\sharp$ where $\alpha = \lambda X.\langle \alpha_1(X \cap \mathbb{A}), \alpha_2(X \cap \mathbb{B})\rangle$
- $\wp(\mathbb{A}) \to \wp(\mathbb{B}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{D}_1^\sharp \to \mathbb{D}_2^\sharp$ where $\alpha = \lambda f.\alpha_2 \circ f \circ \gamma_1$

Example

- Concrete domain: $\mathbb{D} = \wp(\mathbb{M}) \to \wp(\mathbb{M})$ where $\mathbb{M} = \mathbb{X} \to \mathbb{Z}$
- Abstract domain: $\mathbb{D}^\sharp = \mathbb{M}^\sharp \to \mathbb{M}^\sharp$ where $\mathbb{M}^\sharp = \mathbb{X} \to \mathbb{Z}^\sharp$
- Galois connection: $\wp(\mathbb{D}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{D}^\sharp$
  - Memory abstraction: $\wp(\mathbb{M}) \underset{\alpha_{\mathbb{M}}}{\overset{\gamma_{\mathbb{M}}}{\leftrightarrows}} \mathbb{M}^\sharp$ via $\wp(\mathbb{X} \to \mathbb{Z}) \underset{\alpha_{\mathbb{M}}}{\overset{\gamma_{\mathbb{M}}}{\leftrightarrows}} \mathbb{X} \to \mathbb{Z}^\sharp$

$$\gamma_{\mathbb{M}} : \mathbb{M}^\sharp \to \wp(\mathbb{M})$$
$$\gamma_{\mathbb{M}} = \lambda m^\sharp.\{m \mid \forall x.m(x) \in \gamma_{\mathbb{Z}}(m^\sharp(x))\}$$

  - Value abstraction: $\wp(\mathbb{Z}) \underset{\alpha_{\mathbb{Z}}}{\overset{\gamma_{\mathbb{Z}}}{\leftrightarrows}} \mathbb{Z}^\sharp$

Requirement 2: F and $F^\sharp$

- $F^\sharp$ is a sound abstraction of F (option 1)

$$F \circ \gamma \sqsubseteq \gamma \circ F^\sharp$$

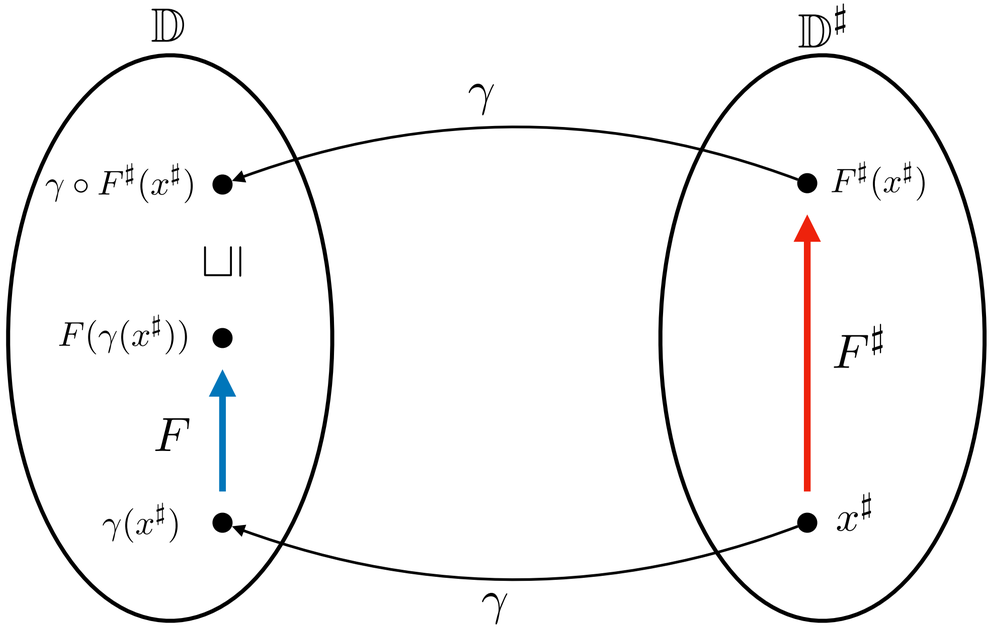
- $F^\sharp$ is a sound abstraction of F (option 2)

$$x \sqsubseteq \gamma(x^\sharp) \implies F(x) \sqsubseteq \gamma(F^\sharp(x^\sharp))$$

Intuition: the result of one-step abstract execution ($F^\sharp$) subsumes that of one-step concrete execution (F)

Sound Abstract Semantics (1)

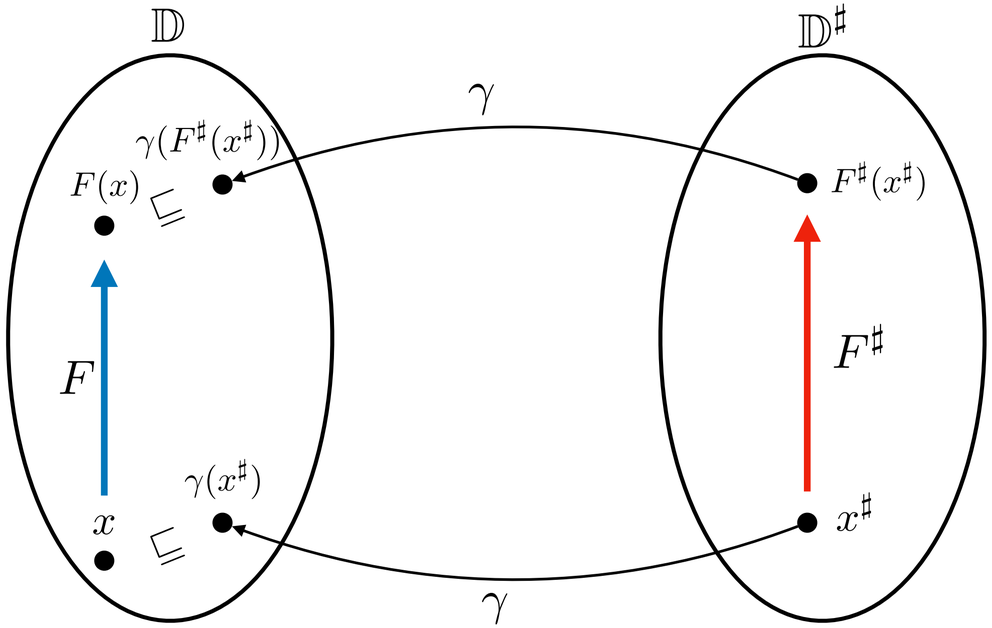
$$F \circ \gamma \sqsubseteq \gamma \circ F^\sharp$$



Intuition: the result of one-step abstract execution ($F^\sharp$) subsumes that of one-step concrete execution (F)

Sound Abstract Semantics (2)

$$x \sqsubseteq \gamma(x^\sharp) \implies F(x) \sqsubseteq \gamma(F^\sharp(x^\sharp))$$



Intuition: the result of one-step abstract execution ($F^\sharp$) subsumes that of one-step concrete execution (F)

Soundness

- Static analysis result $\bigsqcup_{i \geq 0} F^{\sharp i}(\bot)$ soundly subsumes all possible executions

$$\mathbf{lfp} F \sqsubseteq \gamma(\bigsqcup_{i \geq 0} F^{\sharp i}(\bot))$$

- How to guarantee the soundness?
- How to compute the sound result within finite time?

Fixpoint Transfer Theorems

- With option 1

Theorem (Fixpoint Transfer 1)**.** *Let* $\mathbb{D}$ *and* $\mathbb{D}^\sharp$ *be related by Galois connection* $\mathbb{D} \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{D}^\sharp$*. Let* $F : \mathbb{D} \to \mathbb{D}$ *be a continuous function and* $F^\sharp : \mathbb{D}^\sharp \to \mathbb{D}^\sharp$ *be a monotone or extensive function such that* $F \circ \gamma \sqsubseteq \gamma \circ F^\sharp$*. Then,*

$$\mathbf{lfp} F \sqsubseteq \gamma(\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp)).$$

- With option 2

Theorem (Fixpoint Transfer 2)**.** *Let* $\mathbb{D}$ *and* $\mathbb{D}^\sharp$ *be related by Galois connection* $\mathbb{D} \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{D}^\sharp$*. Let* $F : \mathbb{D} \to \mathbb{D}$ *be a continuous function and* $F^\sharp : \mathbb{D}^\sharp \to \mathbb{D}^\sharp$ *be a monotone or extensive function such that* $x \sqsubseteq \gamma(x^\sharp) \implies F(x) \sqsubseteq \gamma(F^\sharp(x^\sharp))$*. Then,*

$$\mathbf{lfp} F \sqsubseteq \gamma(\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp)).$$

Proof

Theorem (Fixpoint Transfer 1). *Let $\mathbb{D}$ and $\mathbb{D}^\sharp$ be related by Galois connection $\mathbb{D} \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{D}^\sharp$. Let $F : \mathbb{D} \to \mathbb{D}$ be a continuous function and $F^\sharp : \mathbb{D}^\sharp \to \mathbb{D}^\sharp$ be a monotone or extensive function such that $F \circ \gamma \sqsubseteq \gamma \circ F^\sharp$. Then,*

$$\mathbf{lfp} F \sqsubseteq \gamma(\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp)).$$

Proof. First we prove $\forall n \in \mathbb{N}.\ F^n(\bot) \sqsubseteq \gamma(F^{\sharp n}(\bot^\sharp))$ by induction. The base case is trivial. The inductive case is as follows:

$$\begin{aligned} F^{n+1}(\bot) &= F \circ F^n(\bot) \\ &\sqsubseteq F \circ \gamma(F^{\sharp n}(\bot^\sharp)) \quad \text{(by induction hypothesis and monotonicity of } F) \\ &\sqsubseteq \gamma \circ F^\sharp \circ F^{\sharp n}(\bot^\sharp) \quad \text{(by assumption } F \circ \gamma \sqsubseteq \gamma \circ F^\sharp) \\ &= \gamma(F^{\sharp n+1}(\bot^\sharp)) \end{aligned}$$

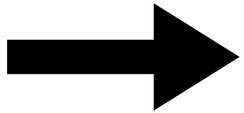
$\{F^i(\bot)\}_i$ is a chain because F is continuous (so monotone). Then, the least upper bound of the chain $\bigsqcup_{i \geq 0} F^i(\bot)$ exists because $\mathbb{D}$ is a CPO. $\{F^{\sharp i}(\bot^\sharp)\}_i$ is a chain because $F^\sharp$ is monotone or extensive. Then, $\{\gamma(F^{\sharp i}(\bot^\sharp))\}_i$ is also a chain because γ is monotone. Therefore, the least upper bound of the chain $\bigsqcup_{i \geq 0} \{\gamma(F^{\sharp i}(\bot^\sharp))\}_i$ exists.

$$\begin{aligned} \mathbf{lfp} F = \bigsqcup_{i \geq 0} F^i(\bot) &\sqsubseteq \bigsqcup_{i \geq 0} \gamma(F^{\sharp i}(\bot^\sharp)) \\ &\sqsubseteq \gamma(\bigsqcup_{i \geq 0} (F^{\sharp i}(\bot^\sharp))) \quad \text{(by monotonicity of } \gamma) \end{aligned}$$

Abstraction of Compositions

- Composition of sound semantic functions is also sound

$$
\begin{array}{ccc}
\mathbb{D}_1 & \underset{\alpha_1}{\overset{\gamma_1}{\leftrightarrows}} & \mathbb{D}_1^\sharp \\
f \downarrow & & \downarrow f^\sharp \\
\mathbb{D}_2 & \underset{\alpha_2}{\overset{\gamma_2}{\leftrightarrows}} & \mathbb{D}_2^\sharp \\
g \downarrow & & \downarrow g^\sharp \\
\mathbb{D}_3 & \underset{\alpha_3}{\overset{\gamma_3}{\leftrightarrows}} & \mathbb{D}_3^\sharp
\end{array}
$$



$$g \circ f \circ \gamma_1 \sqsubseteq \gamma_3 \circ g^\sharp \circ f^\sharp$$

- Implication: soundness proof can be compositional

Abstract Semantics of Expressions

$$
\begin{aligned}
[\![E]\!]^\sharp &: \mathbb{M}^\sharp \to \mathbb{Z}^\sharp \\
[\![n]\!]^\sharp &= \lambda m^\sharp.\alpha(\{n\}) \\
[\![x]\!]^\sharp &= \lambda m^\sharp.m^\sharp(x) \\
[\![E_1 \odot E_2]\!]^\sharp &= \lambda m^\sharp.[\![E_1]\!]^\sharp(m^\sharp) \odot^\sharp [\![E_2]\!]^\sharp(m^\sharp)
\end{aligned}
$$

- Example
  - Sign domain $\mathbb{Z}^\sharp = \{\bot, -, 0, +, \top\}$
  - Interval domain $\mathbb{Z}^\sharp = \{\bot\} \cup \{[a, b] \mid a \in \mathbb{Z} \cup \{-\infty\}, b \in \mathbb{Z} \cup \{+\infty\}\}$
- Soundness: $[\![E]\!]_\wp \circ \gamma_\mathbb{M} \subseteq \gamma_\mathbb{Z} \circ [\![E]\!]^\sharp$

Abstract Semantics of Conditions

$$
\begin{aligned}
[\![B]\!]^\sharp &: \mathbb{M}^\sharp \to \mathbb{M}^\sharp \\
[\![\mathsf{true}]\!]^\sharp &= \lambda m^\sharp . m^\sharp \\
[\![\mathsf{false}]\!]^\sharp &= \lambda m^\sharp . \bot
\end{aligned}
$$

- Example
  - Sign domain $[\![x < 0]\!]^\sharp = \lambda m^\sharp . \begin{cases} \bot & \text{if } m^\sharp(x) \in \{+, 0, \bot\} \\ m^\sharp\{x \mapsto -\} & \text{o.w.} \end{cases}$
  - Interval domain $[\![x < n]\!]^\sharp = \lambda m^\sharp . \begin{cases} \bot & \text{if } m^\sharp(x) = [a, b] \ \wedge \ a > n \\ m^\sharp\{x \mapsto [a, n-1]\} & \text{if } a \leq n \leq b \\ m^\sharp & \text{if } b < n \end{cases}$
- Soundness: $[\![B]\!]_\wp \circ \gamma_{\mathbb{M}} \subseteq \gamma_{\mathbb{M}} \circ [\![B]\!]^\sharp$

Abstract Semantics of Commands

$$
\begin{aligned}
[\![C]\!]^\sharp &: \mathbb{M}^\sharp \to \mathbb{M}^\sharp \\
[\![\texttt{skip}]\!]^\sharp &= \lambda m^\sharp . m^\sharp \\
[\![C_0 \texttt{;} C_1]\!]^\sharp &= \lambda m^\sharp . [\![C_1]\!]^\sharp \circ [\![C_0]\!]^\sharp (m^\sharp) \\
[\![x \texttt{:=} E]\!]^\sharp &= \lambda m^\sharp . m^\sharp \{x \mapsto [\![E]\!]^\sharp (m^\sharp)\} \\
[\![\texttt{input}(x)]\!]^\sharp &= \lambda m^\sharp . m^\sharp \{x \mapsto \alpha(\mathbb{Z})\} \\
[\![\texttt{if } B \texttt{ then } C_1 \texttt{ else } C_2]\!]^\sharp &= \lambda m^\sharp . [\![C_1]\!]^\sharp \circ [\![B]\!]^\sharp (m^\sharp) \sqcup [\![C_2]\!]^\sharp \circ [\![\neg B]\!]^\sharp (m^\sharp) \\
[\![\texttt{while } B \ C]\!]^\sharp &= \lambda m^\sharp . [\![\neg B]\!]^\sharp \big(\mathbf{lfp} \lambda X . M^\sharp \sqcup [\![C]\!]^\sharp \circ [\![B]\!]^\sharp (M^\sharp) \big)
\end{aligned}
$$

- Soundness: $[\![C]\!]_\wp \circ \gamma_{\mathbb{M}} \subseteq \gamma_{\mathbb{M}} \circ [\![C]\!]^\sharp$

Design of Static Analysis

- Goal: conservative and terminating static analysis
- Design principles:
  - Define concrete semantics
  - Define abstract semantics (sound w.r.t the concrete semantics)
- Computation & implementation:
  - Abstract semantics of a program: the least fixed point of the semantic function
  - Static analyzer: compute the least fixed point within finite time

Computing Abstract Semantics

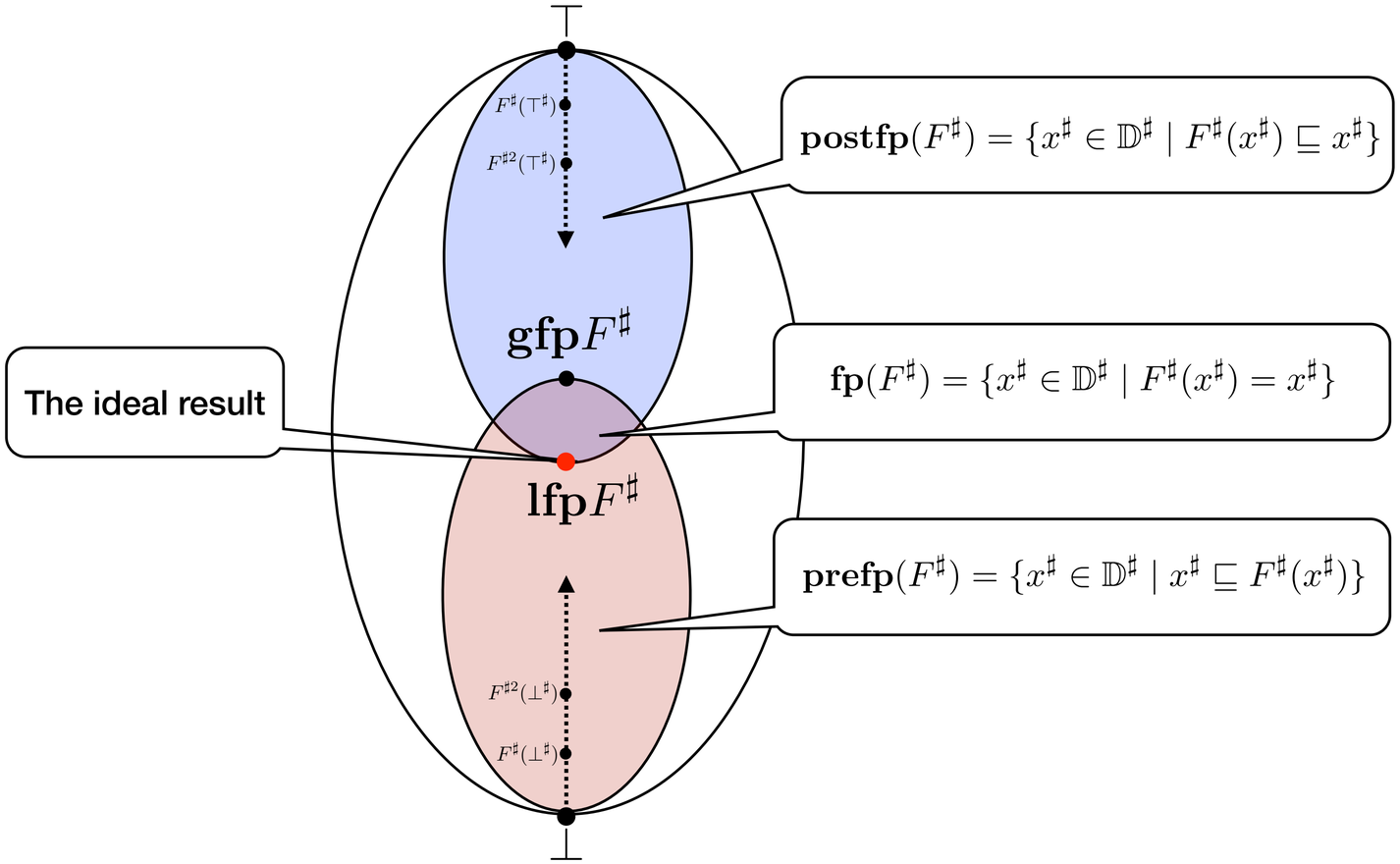
- If the abstract domain $\mathbb{D}^\sharp$ has **finite** height (i.e., all chains are finite)

$$\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp)$$

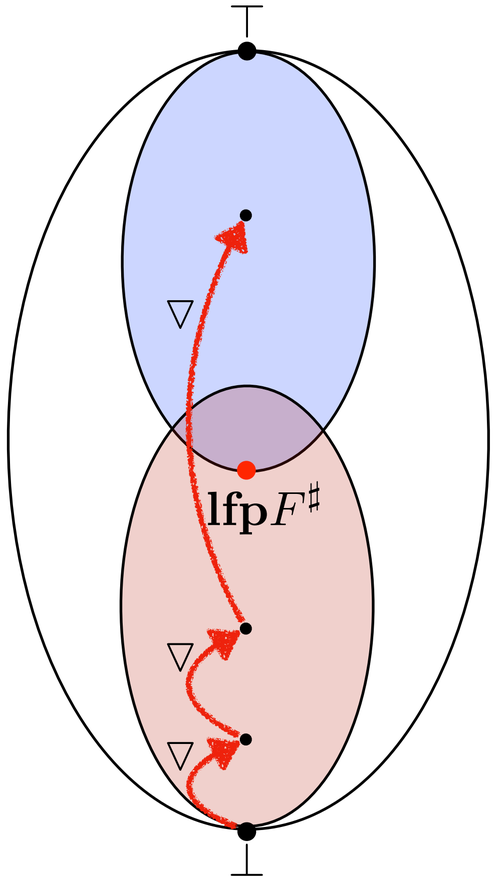
- If the abstract domain $\mathbb{D}^\sharp$ has **infinite** height, we compute a finite chain $X_0^\sharp \sqsubseteq X_1^\sharp \sqsubseteq X_2^\sharp \sqsubseteq \ldots \sqsubseteq X_{\text{lim}}^\sharp$ such that

$$\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp) \sqsubseteq X_{\text{lim}}^\sharp$$

Fixed Points



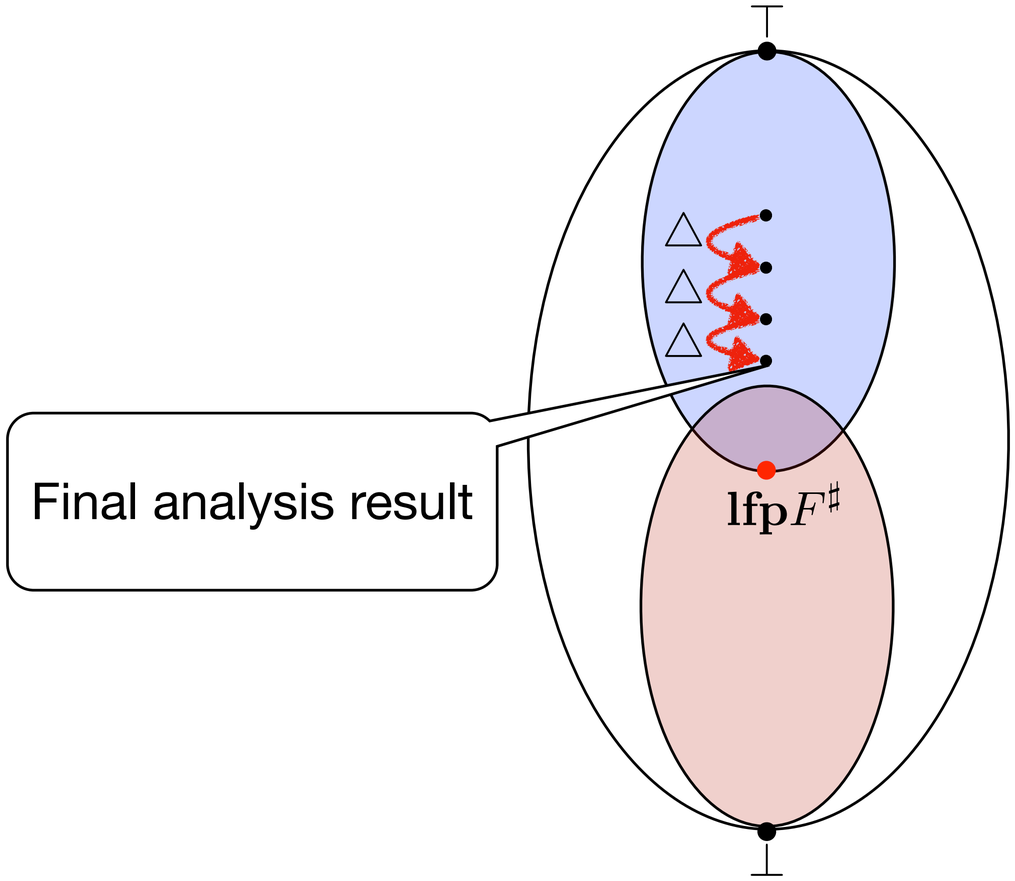
Widening



$$\nabla : \mathbb{D}^\sharp \times \mathbb{D}^\sharp \to \mathbb{D}^\sharp$$

Widening: enforcing the convergence of fix point iterations

Narrowing



$$\triangle : \mathbb{D}^\sharp \times \mathbb{D}^\sharp \to \mathbb{D}^\sharp$$

Narrowing: refining the analysis results with widening

Overshooting by Widening

$$\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp) \sqsubseteq Y^\sharp_{\text{lim}}$$

- Define finite chain $\{Y_i\}_i$ by an widening operator $\triangledown \in \mathbb{D}^\sharp \times \mathbb{D}^\sharp \to \mathbb{D}^\sharp$:

$$\begin{array}{rcl} Y_0^\sharp & = & \bot^\sharp \\ Y_{i+1}^\sharp & = & \begin{cases} Y_i^\sharp & \text{if } F^\sharp(Y_i^\sharp) \sqsubseteq Y_i^\sharp \\ Y_i^\sharp \ \triangledown \ F^\sharp(Y_i^\sharp) & \text{otherwise} \end{cases} \end{array}$$

Finite Increasing Chain with Widening

$$\vdots$$
$$|$$
$$F^{\sharp 2}(\bot^\sharp)$$
$$|$$
$$F^\sharp(\bot^\sharp)$$
$$|$$
$$\bot^\sharp$$

Original Chain

$$Y^\sharp_{N+1} = Y^\sharp_N$$
$$|$$
$$Y^\sharp_N = Y^\sharp_{N-1} \triangledown F^\sharp(Y^\sharp_{N-1})$$
$$\vdots$$
$$Y^\sharp_2 = Y^\sharp_1 \triangledown F^\sharp(Y^\sharp_1)$$
$$|$$
$$Y^\sharp_1 = Y^\sharp_0 \triangledown F^\sharp(Y^\sharp_0)$$
$$|$$
$$Y^\sharp_0 = \bot^\sharp$$

New Chain with Widening

Q. What conditions are required to ensure

$$\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp) \sqsubseteq Y^\sharp_{\text{lim}}$$

Safety of Widening Operator

- Conditions on widening operator:
  - $\forall a, b \in \mathbb{D}^\sharp.\ (a \sqsubseteq a \triangledown b) \ \wedge \ (b \sqsubseteq a \triangledown b)$
  - $\forall$increasing chain $\{x_i\}_i$, the following increasing chain $\{y_i\}_i$ is finite:

$$y_0 = x_0$$
$$y_{i+1} = y_i \triangledown x_{i+1}$$

- Then,
  - Chain $\{Y_i^\sharp\}_i$ is finite
  - $\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp) \sqsubseteq Y^\sharp_{\text{lim}}$

Proof

Theorem (Widening's Safety). *Let $\mathbb{D}^\sharp$ be a CPO, $F^\sharp : \mathbb{D}^\sharp \to \mathbb{D}^\sharp$ be a monotone function, and $\triangledown : \mathbb{D}^\sharp \times \mathbb{D}^\sharp \to \mathbb{D}^\sharp$ be a widening operator. Then, chain $\{Y_i^\sharp\}_i$ eventually stabilizes and*

$$\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp) \sqsubseteq Y^\sharp_{\text{lim}}$$

where $Y^\sharp_{\text{lim}}$ is the greatest element of the chain.

Proof. First we prove chain $\{Y_i^\sharp\}_i$ is finite. According to the second condition on widening operator, it is enough to show that chain $\{F^\sharp(Y_i^\sharp)\}_i$ is increasing. The chain is increasing because 1) $F^\sharp(Y^\sharp_{i+1})$ is either $F^\sharp(Y_i^\sharp)$ or $F^\sharp(Y_i^\sharp \triangledown F^\sharp(Y_i^\sharp))$, 2) $Y_i^\sharp \sqsubseteq Y_i^\sharp \triangledown F^\sharp(Y_i^\sharp)$ according to the first condition on widening, and 3) $F^\sharp$ is monotone.

Second, we prove $\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp) \sqsubseteq Y^\sharp_{\text{lim}}$. It is enough to show that $\forall i \in \mathbb{N}.\ F^{\sharp i}(\bot^\sharp) \sqsubseteq Y_i^\sharp$ that can be proven by induction. The base case is trivial. The inductive case is as follows:

$$\begin{aligned} F^{\sharp i+1}(\bot^\sharp) &= F^\sharp(F^{\sharp i}(\bot^\sharp)) \\ &\sqsubseteq F^\sharp(Y_i^\sharp) \qquad \text{(by induction hypothesis and monotonicity of } F^\sharp) \end{aligned}$$

If $F^\sharp(Y_i^\sharp) \sqsubseteq Y_i^\sharp$, then $Y^\sharp_{i+1} = Y_i^\sharp$ by definition. Therefore, $F^{\sharp i+1}(\bot^\sharp) \sqsubseteq Y^\sharp_{i+1}$.

If $F^\sharp(Y_i^\sharp) \sqsupset Y_i^\sharp$, then $Y^\sharp_{i+1} = Y_i^\sharp \triangledown F^\sharp(Y_i^\sharp)$ by definition. According to the first condition on widening, $F^\sharp(Y_i^\sharp) \sqsubseteq Y_i^\sharp \triangledown F^\sharp(Y_i^\sharp)$. Therefore, $F^{\sharp i+1}(\bot^\sharp) \sqsubseteq Y^\sharp_{i+1}$.

Proof (A Generalized Ver.)

Theorem (Widening's Safety). *Let* $\mathbb{D} \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \mathbb{D}^\sharp$ *be a Galois connection. Let* $F : \mathbb{D} \to \mathbb{D}$ *and* $F^\sharp : \mathbb{D}^\sharp \to \mathbb{D}^\sharp$ *be a concrete and abstract semantic function. Let* $\triangledown : \mathbb{D}^\sharp \times \mathbb{D}^\sharp \to \mathbb{D}^\sharp$ *be a widening operator. Then, chain* $\{Y_i^\sharp\}_i$ *eventually stabilizes and*

$$\bigsqcup_{i \geq 0} F(\bot) \sqsubseteq \gamma(Y_{\text{lim}}^\sharp)$$

where $Y_{\text{lim}}^\sharp$ *is the greatest element of the chain.*

Proof. First we prove chain $\{Y_i^\sharp\}_i$ is finite. According to the second condition on widening operator, it is enough to show that chain $\{F^\sharp(Y_i^\sharp)\}_i$ is increasing. The chain is increasing because 1) $F^\sharp(Y_{i+1}^\sharp)$ is either $F^\sharp(Y_i^\sharp)$ or $F^\sharp(Y_i^\sharp \triangledown F^\sharp(Y_i^\sharp))$, 2) $Y_i^\sharp \sqsubseteq Y_i^\sharp \triangledown F^\sharp(Y_i^\sharp)$ according to the first condition on widening, and 3) $F^\sharp$ is monotone or extensive.

Second, we prove $\bigsqcup_{i \geq 0} F^i(\bot) \sqsubseteq \gamma(Y_{\text{lim}}^\sharp)$. It is enough to show that $\forall i \in \mathbb{N}.\ F^i(\bot) \sqsubseteq \gamma(Y_i^\sharp)$ that can be proven by induction. The base case is trivial. The inductive case is as follows:

$$\begin{aligned} F^{i+1}(\bot) &= F \circ F^i(\bot) \\ &\sqsubseteq F \circ \gamma(Y_i^\sharp) && \text{(by induction hypothesis and monotonicity of } F) \\ &\sqsubseteq \gamma \circ F^\sharp(Y_i^\sharp) && \text{(by } F \circ \gamma \sqsubseteq \gamma \circ F^\sharp) \end{aligned}$$

If $F^\sharp(Y_i^\sharp) \sqsubseteq Y_i^\sharp$, then $Y_{i+1}^\sharp = Y_i^\sharp$ by definition. Therefore, $F^{i+1}(\bot) \sqsubseteq \gamma(Y_{i+1}^\sharp)$ by monotonicity of γ.

If $F^\sharp(Y_i^\sharp) \sqsupset Y_i^\sharp$, then $Y_{i+1}^\sharp = Y_i^\sharp \triangledown F^\sharp(Y_i^\sharp)$ by definition. According to the first condition on widening, $F^\sharp(Y_i^\sharp) \sqsubseteq Y_i^\sharp \triangledown F^\sharp(Y_i^\sharp)$. Therefore, $F^{\sharp i+1}(\bot^\sharp) \sqsubseteq Y_{i+1}^\sharp$ by monotonicity of γ.

Refinement by Narrowing

$$\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp) \sqsubseteq Z^\sharp_{\text{lim}}$$

- Define finite chain $\{Z^\sharp_i\}_i$ by an narrowing operator $\triangle \in \mathbb{D}^\sharp \times \mathbb{D}^\sharp \to \mathbb{D}^\sharp$:

$$\begin{array}{rcl} Z^\sharp_0 & = & Y^\sharp_{\text{lim}} \\ Z^\sharp_{i+1} & = & Z^\sharp_i \ \triangle \ F^\sharp(Z^\sharp_i) \end{array}$$

Finite Decreasing Chain with Narrowing

$$
\begin{array}{c}
Y^\sharp_{\text{lim}} \\
| \\
F^\sharp(Y^\sharp_{\text{lim}}) \\
| \\
F^{\sharp 2}(Y^\sharp_{\text{lim}}) \\
\vdots
\end{array}
$$

Original Chain

$$
\begin{array}{l}
Z^\sharp_0 = Y^\sharp_{\text{lim}} \\
| \\
Z^\sharp_1 = Z^\sharp_0 \mathbin{\triangle} F^\sharp(Z^\sharp_0) \\
| \\
Z^\sharp_2 = Z^\sharp_1 \mathbin{\triangle} F^\sharp(Z^\sharp_1) \\
\vdots \\
Z^\sharp_N = Z^\sharp_{N-1} \mathbin{\triangle} F^\sharp(Z^\sharp_{N-1}) \\
| \\
Z^\sharp_{N+1} = Z^\sharp_N
\end{array}
$$

New Chain with Narrowing

Q. What conditions are required to ensure

$$\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp) \sqsubseteq Z^\sharp_{\text{lim}}$$

Safety of Narrowing Operator

- Conditions on narrowing operator:
  - $\forall a, b \in \mathbb{D}^\sharp.\ a \sqsupseteq b \implies a \sqsupseteq (a \mathbin{\triangle} b) \sqsupseteq b$
  - For all decreasing chain $\{y_i\}_i$, the following decreasing chain $\{z_i\}_i$ is finite

$$z_0 = y_0$$
$$z_{i+1} = z_i \mathbin{\triangle} y_{i+1}$$

- Then,
  - Decreasing chain $\{Z_i^\sharp\}_i$ is finite
  - $\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp) \sqsubseteq Z_{\text{lim}}$

Proof

Theorem (Narrowing's Safety). *Let $\mathbb{D}^\sharp$ be a CPO, $F^\sharp : \mathbb{D}^\sharp \to \mathbb{D}^\sharp$ be a monotone function, and $\triangle : \mathbb{D}^\sharp \times \mathbb{D}^\sharp \to \mathbb{D}^\sharp$ be a narrowing operator. Then, chain $\{Z_i^\sharp\}_i$ eventually stabilizes and*

$$\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp) \sqsubseteq Z^\sharp_{\text{lim}}$$

where $Z^\sharp_{\text{lim}}$ is the least element of the chain.

Proof. First we prove chain $\{Z_i^\sharp\}_i$ is finite. According to the second condition on narrowing operator, it is enough to show that chain $\{F^\sharp(Z_i^\sharp)\}_i$ is decreasing. The chain is decreasing if $\forall i \in \mathbb{N}.\ Z_i^\sharp \sqsupseteq F^\sharp(Z_i^\sharp)$, because

$$\begin{aligned}
& Z_i^\sharp \sqsupseteq F^\sharp(Z_i^\sharp) \\
\Longrightarrow\ & Z_i^\sharp \sqsupseteq (Z_i^\sharp \triangle F^\sharp(Z_i^\sharp)) \sqsupseteq F^\sharp(Z_i^\sharp) && \text{(by the first condition on narrowing)} \\
\Longrightarrow\ & F^\sharp(Z_i^\sharp) \sqsupseteq F^\sharp(Z_i^\sharp \triangle F^\sharp(Z_i^\sharp)) && \text{(by monotonicity of } F^\sharp\text{)} \\
\Longrightarrow\ & F^\sharp(Z_i^\sharp) \sqsupseteq F^\sharp(Z_{i+1}^\sharp) && \text{(by definition of } Z_{i+1}^\sharp\text{)}
\end{aligned}$$

We prove $\forall i \in \mathbb{N}.\ Z_i^\sharp \sqsupseteq F^\sharp(Z_i^\sharp)$ by induction. The base case is true by definition of $Y^\sharp_{\text{lim}}$ from the increasing chain by widening. The inductive case is as follows:

$$\begin{aligned}
& Z_i^\sharp \sqsupseteq F^\sharp(Z_i^\sharp) && \text{(by induction hypothesis)} \\
\Longrightarrow\ & Z_i^\sharp \sqsupseteq Z_i^\sharp \triangle F^\sharp(Z_i^\sharp) \sqsupseteq F^\sharp(Z_i^\sharp) && \text{(by the first condition on narrowing)} \\
\Longrightarrow\ & Z_i^\sharp \sqsupseteq Z_{i+1}^\sharp \sqsupseteq F^\sharp(Z_i^\sharp) && \text{(by definition)} \\
\Longrightarrow\ & Z_i^\sharp \sqsupseteq F^\sharp(Z_i^\sharp) \sqsupseteq F^\sharp(Z_{i+1}^\sharp) && \text{(by monotonicity of } F^\sharp\text{)} \\
\Longrightarrow\ & Z_i^\sharp \sqsupseteq F^\sharp(Z_{i+1}^\sharp)
\end{aligned}$$

Proof

Theorem (Narrowing's Safety). *Let $\mathbb{D}^\sharp$ be a CPO, $F^\sharp : \mathbb{D}^\sharp \to \mathbb{D}^\sharp$ be a monotone function, and $\triangle : \mathbb{D}^\sharp \times \mathbb{D}^\sharp \to \mathbb{D}^\sharp$ be a narrowing operator. Then, chain $\{Z_i^\sharp\}_i$ eventually stabilizes and*

$$\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp) \sqsubseteq Z^\sharp_{\text{lim}}$$

where $Z^\sharp_{\text{lim}}$ is the least element of the chain.

(Cont'd) Second we prove chain $\bigsqcup_{i \geq 0} F^{\sharp i}(\bot^\sharp) \sqsubseteq Z^\sharp_{\text{lim}}$. It is enough to show that $\forall i \in \mathbb{N}.\ F^{\sharp i}(\bot^\sharp) \sqsubseteq Z_i^\sharp$ that can be proven by induction. The base case is trivial. The inductive case is as follows:

$$\begin{aligned}
F^{\sharp i+1}(\bot^\sharp) &= F^\sharp \circ F^{\sharp i}(\bot^\sharp) \\
&\sqsubseteq F^\sharp(Z_i^\sharp) && \text{(by induction hypothesis and monotonicity of } F^\sharp) \\
&\sqsubseteq Z_i^\sharp \ \triangle\ F^\sharp(Z_i^\sharp) && \text{(by condition } \forall i \in \mathbb{N}.\ Z_i^\sharp \sqsupseteq F^\sharp(Z_i^\sharp)) \\
&= Z_{i+1}^\sharp && \text{(by definition)}
\end{aligned}$$

Summary

- Abstract interpretation: a **framework** for designing correct static analysis
- **Soundness** guarantee by Fixpoint transfer theorem
  - Galois connection, sound abstract semantic function
- Widening: **termination** guarantee
- Narrowing: **refinement** of widening results