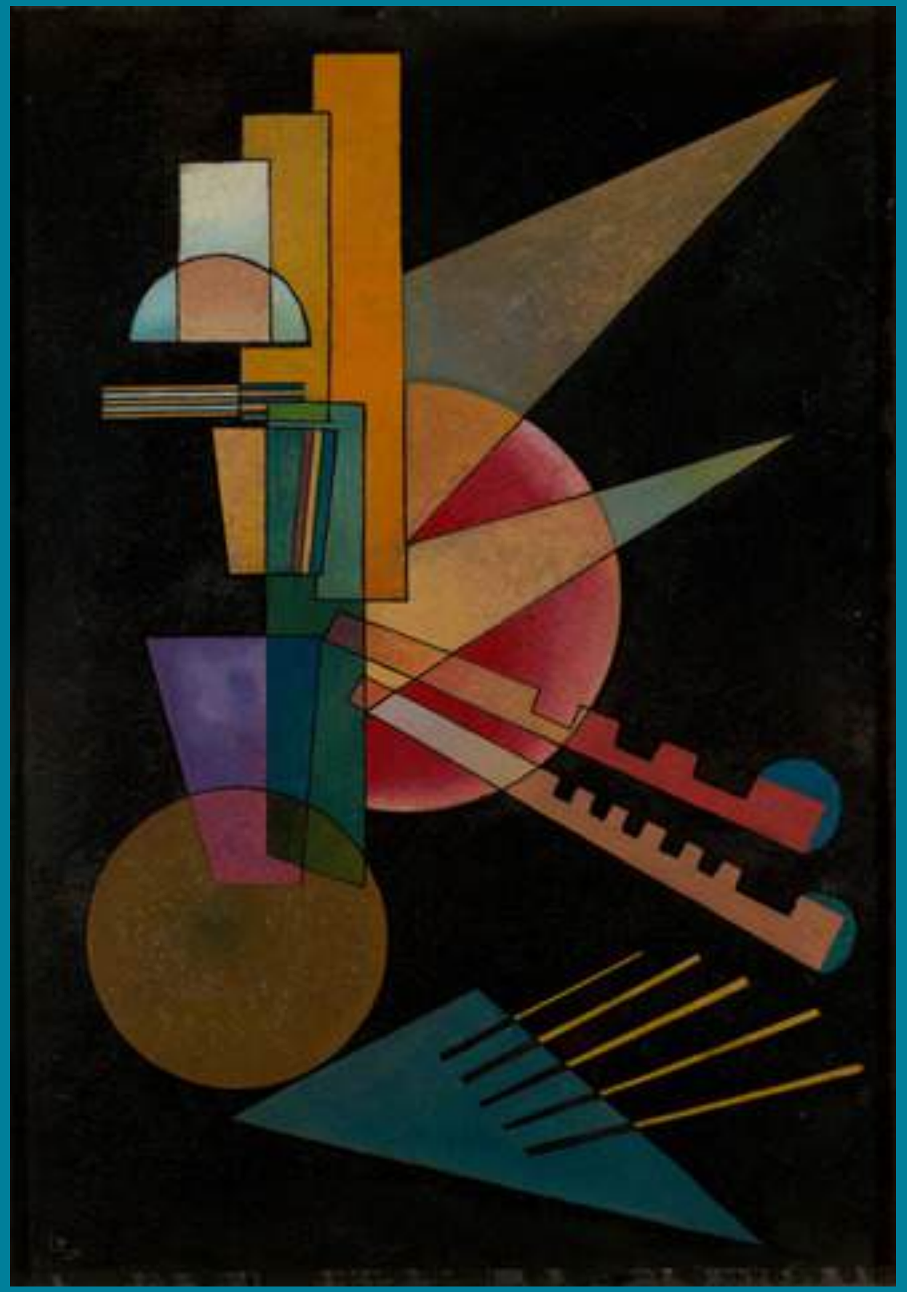


Kandinsky - Abstract Interpretation, 1925

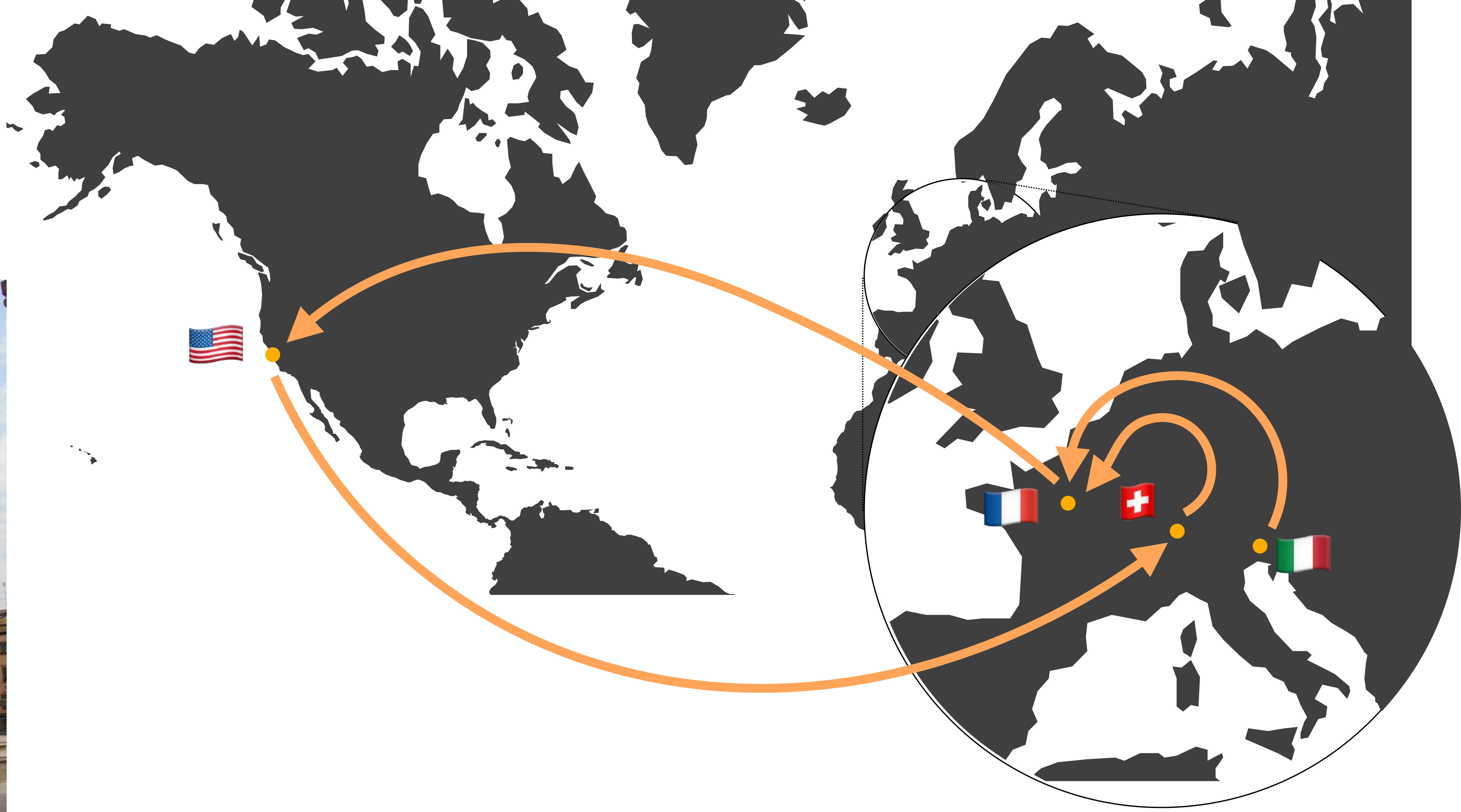
Abstract Interpretation

and Applications in Security, Data Science, and Machine Learning

OPLSS 2025

Caterina Urban
Inria & École Normale Supérieure | Université PSL

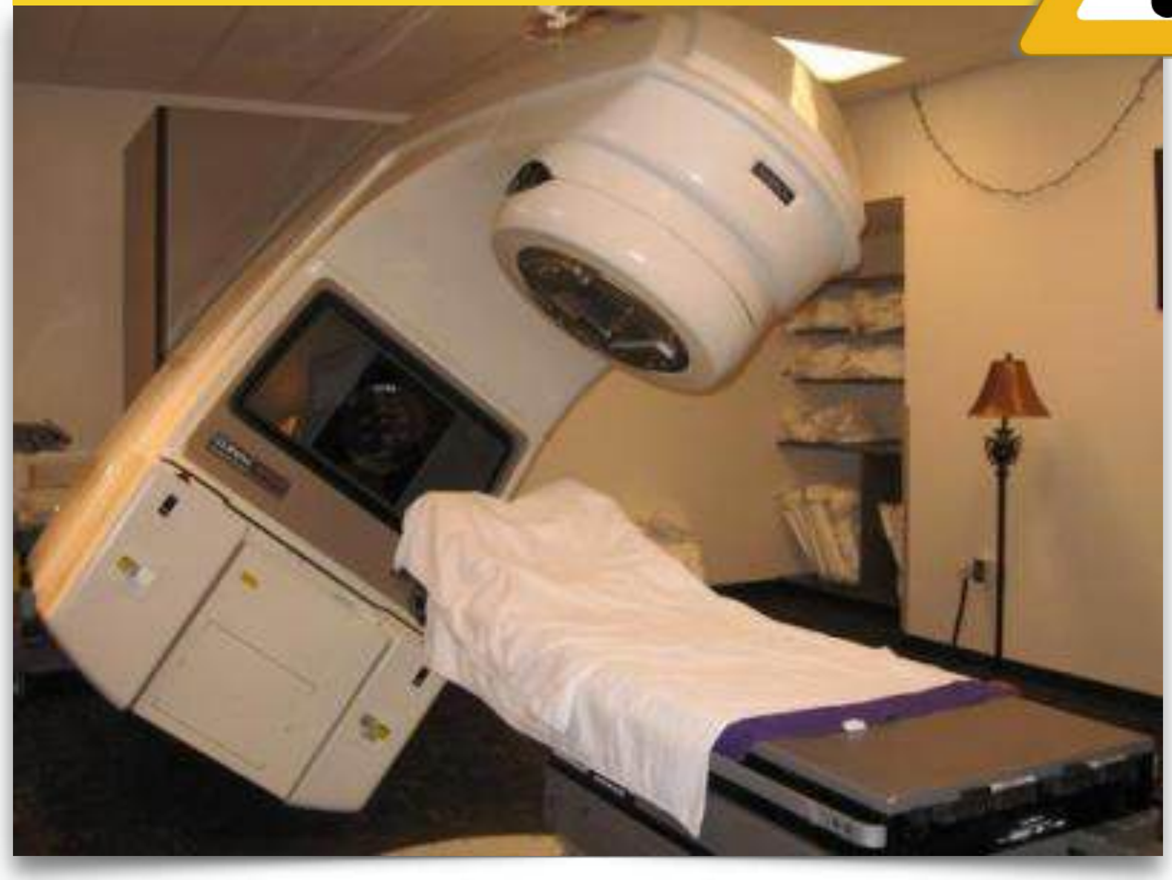
Who am I?



1987	Udine, Italie	
2006 - 2011	Università degli Studi di Udine	BSc, MSc
2011 - 2015	École Normale Supérieure	PhD
2015	NASA & Carnegie Mellon University	Internship
2015 - 2019	ETH Zurich	Postdoc
Since 2019	Inria	

The Cost of Software Failure

4 dead, 2 lifelong injuries



Therac-25, 1985-1987

loss of more than $370 000 000

Ariane 5, 4 June 1996

at least 89 deaths

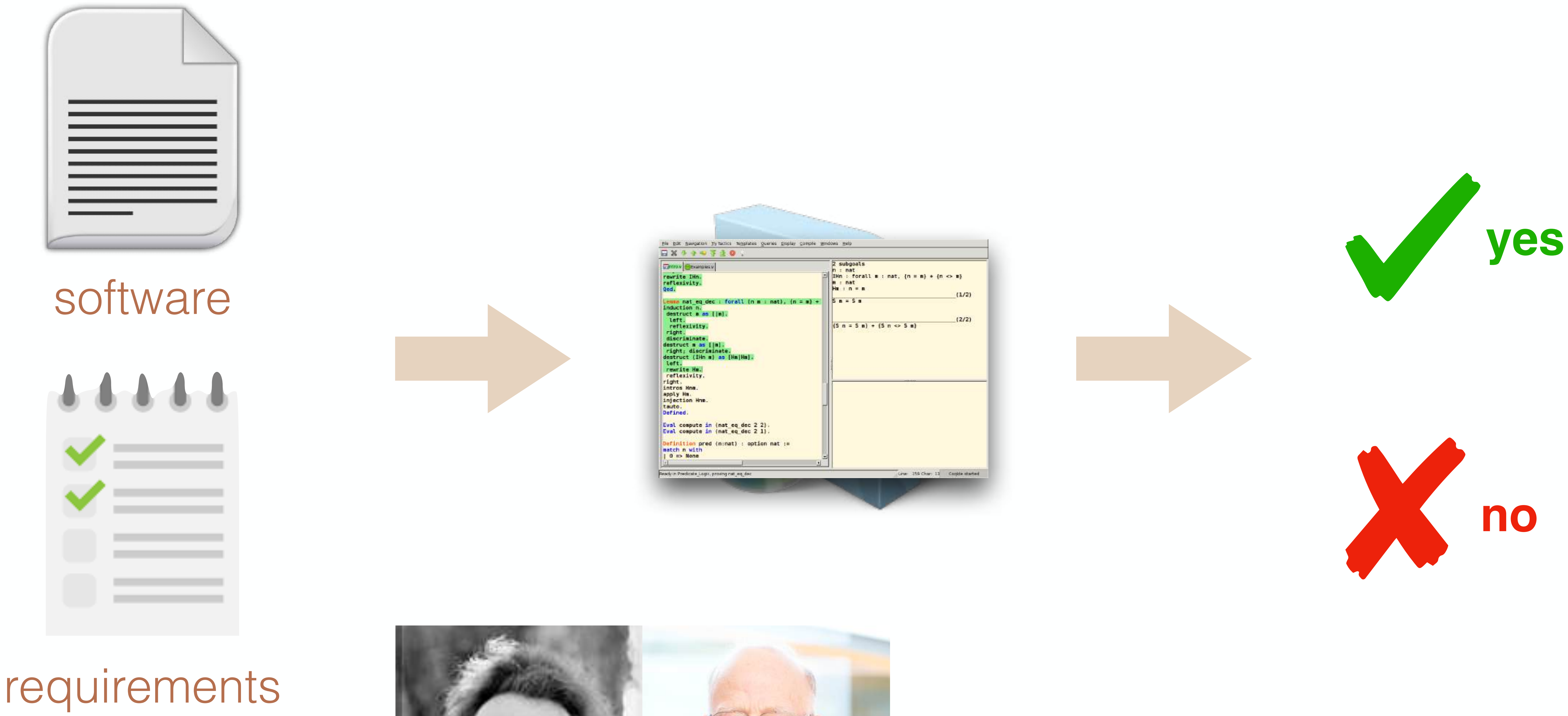
Toyota, 2000-2010

Correctness Guarantees

A Mathematically Proven Hard Problem

Formal Methods

Deductive Verification

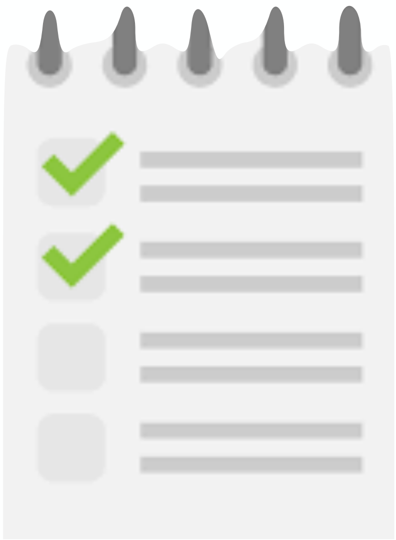


- extremely **expressive**
- **relies on the user** to guide the proof

Formal Methods

Model Checking

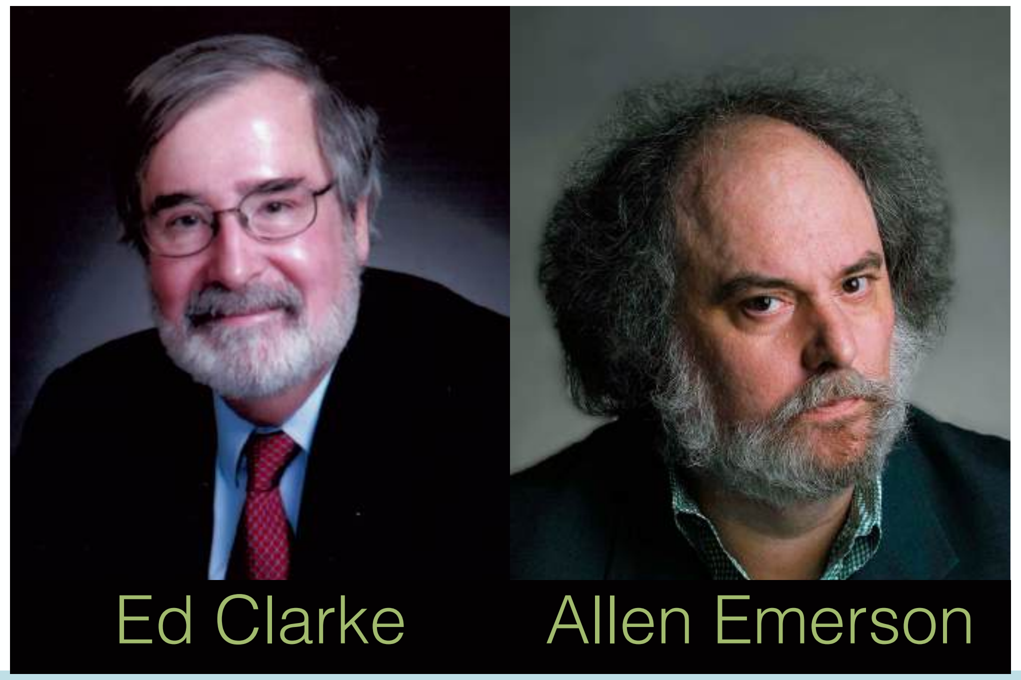
model



requirements

yes

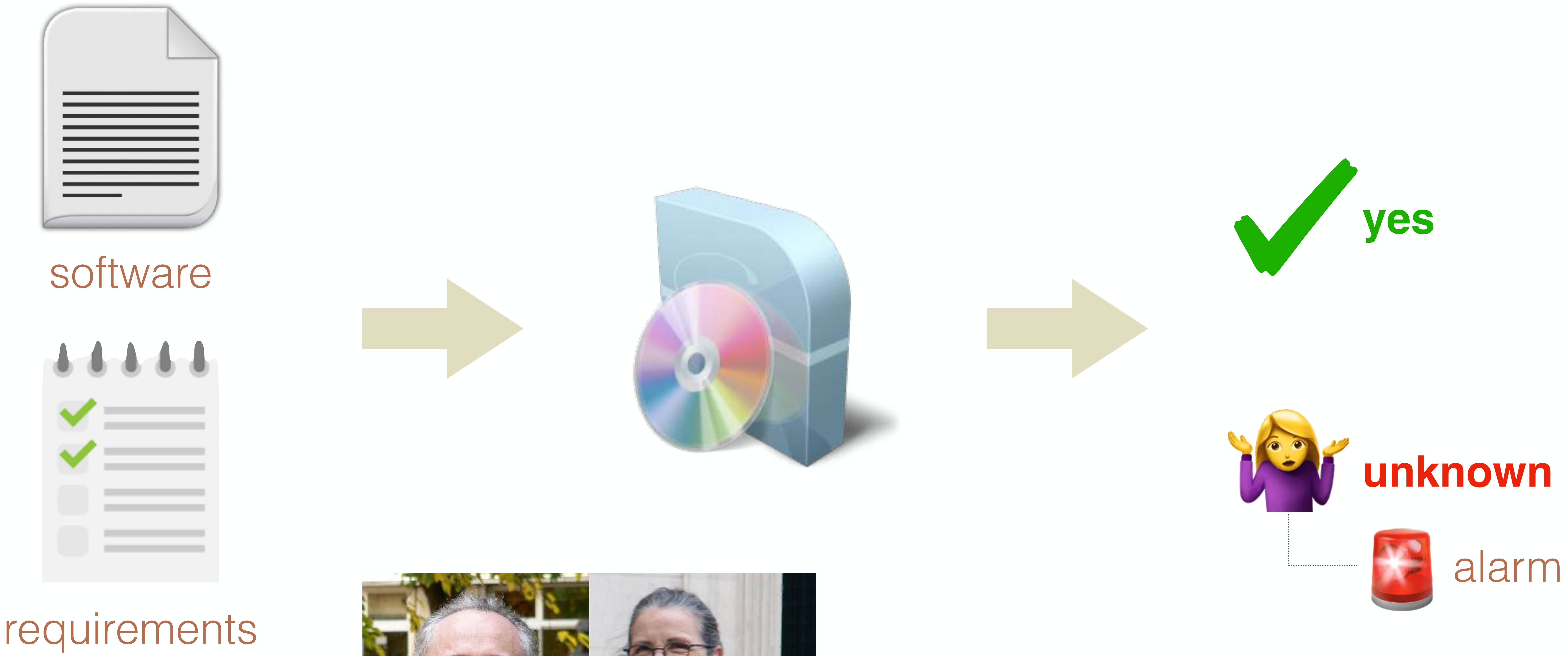
no



- analysis of a **model** of the software
- **sound and complete with respect to the model**

Formal Methods

Static Analysis by Abstract Interpretation



- analysis of the **source or object code**
- fully **automatic** and **sound** by construction
- generally **not complete**

Static Analysis by Abstract Interpretation

Static Analysis by Abstract Interpretation

ABSTRACTION

PROPERTY OF INTEREST

SOUNDNESS

$ 10 +
$ 40 +
$ 30 +
$ 10

$ 90

~~**COMPLETENESS**~~

$ 9.95 +
$ 35.85 +
$ 24.95 +
$ 4.85

$ 75.60

FALSE ALARM

Abstract Interpretation Today

integral part of the development of safety-critical software

AIRBUS

aviation software

esa

space software

HELBAKO

automotive software

AREVA

nuclear software

successfully employed by software companies

Google

Meta

Infer

Microsoft

Course Plan

The Art of Losing Precision

No Surprises, Please

What Could
Possibly Go Right?

It's Complicated

Principles of Abstract Interpretation

The Art of Losing Precision

No Surprises, Please

What Could Possibly Go Right?

It's Complicated

Language Syntax

Numeric Expressions

$$
\begin{array}{lll}
expr ::= & X & \text{(variable, } X \in \mathbb{X}\text{)} \\
 & \mid\ c & \text{(constant, } c \in \mathbb{Z}\text{)} \\
 & \mid\ [c_1, c_2] & \text{(non-deterministic input, } c_1, c_2 \in \mathbb{Z} \cup \{-\infty, +\infty\}\text{)} \\
 & \mid\ -expr & \text{(negation)} \\
 & \mid\ expr \diamond expr & \text{(binary operation, } \diamond \in \{+, -, \dots\}\text{)}
\end{array}
$$

Statements

$$
\begin{array}{lll}
stmt ::= & {}^{\ell}X \leftarrow expr^{\ell} & \text{(assignment, } \ell \in \mathscr{L}\text{)} \\
 & \mid\ \textbf{if } {}^{\ell}expr \bowtie 0 \textbf{ then } stmt \textbf{ end}^{\ell} & \text{(conditional, } \bowtie \in \{=, \leq, \dots\}\text{)} \\
 & \mid\ \textbf{while } {}^{\ell}expr \bowtie 0 \textbf{ do } stmt \textbf{ done}^{\ell} & \text{(loop)} \\
 & \mid\ stmt; stmt & \text{(sequence)}
\end{array}
$$

Example

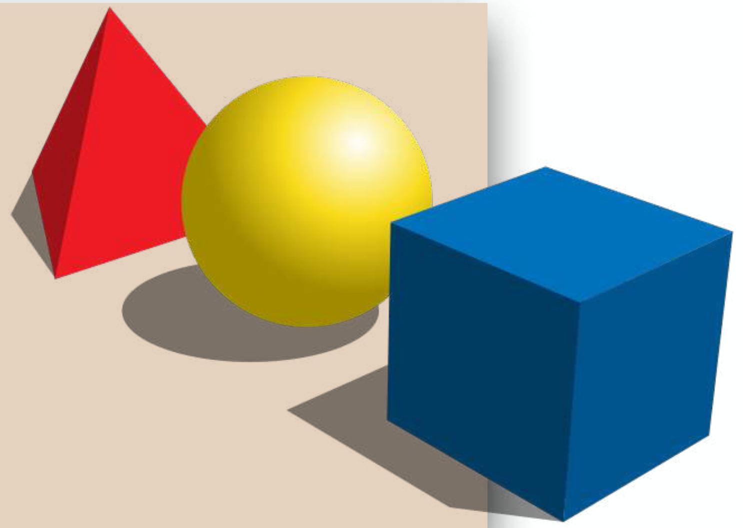
```
1
a ← [0, +∞]
2
b ← [0, +∞]
3
q ← 0
4
r ← a
5
while 6(r ≥ b) do
    7
    r ← r − b
    8
    q ← q + 1
    9
done
10
```

Static Analysis by Abstract Interpretation

3-Step Recipe

practical tools
targeting specific programs

abstract semantics, abstract domains
algorithmic approaches to decide program properties



concrete semantics
mathematical models of the program behavior

Static Analysis by Abstract Interpretation

Program Semantics

practical tools
targeting specific programs

abstract semantics, abstract domains
algorithmic approaches to decide program properties

concrete semantics
mathematical models of the program behavior

Static Analysis by Abstract Interpretation

Expression Semantics

- $E[\![expr]\!] : \underbrace{(\mathbb{X} \to \mathbb{Z})}_{\rho \in \mathscr{E}} \to \mathscr{P}(\mathbb{Z})$ memory state

$$
\begin{aligned}
expr ::= \; & X \\
\mid \; & c \\
\mid \; & [c_1, c_2] \\
\mid \; & -expr \\
\mid \; & expr \diamond expr
\end{aligned}
$$

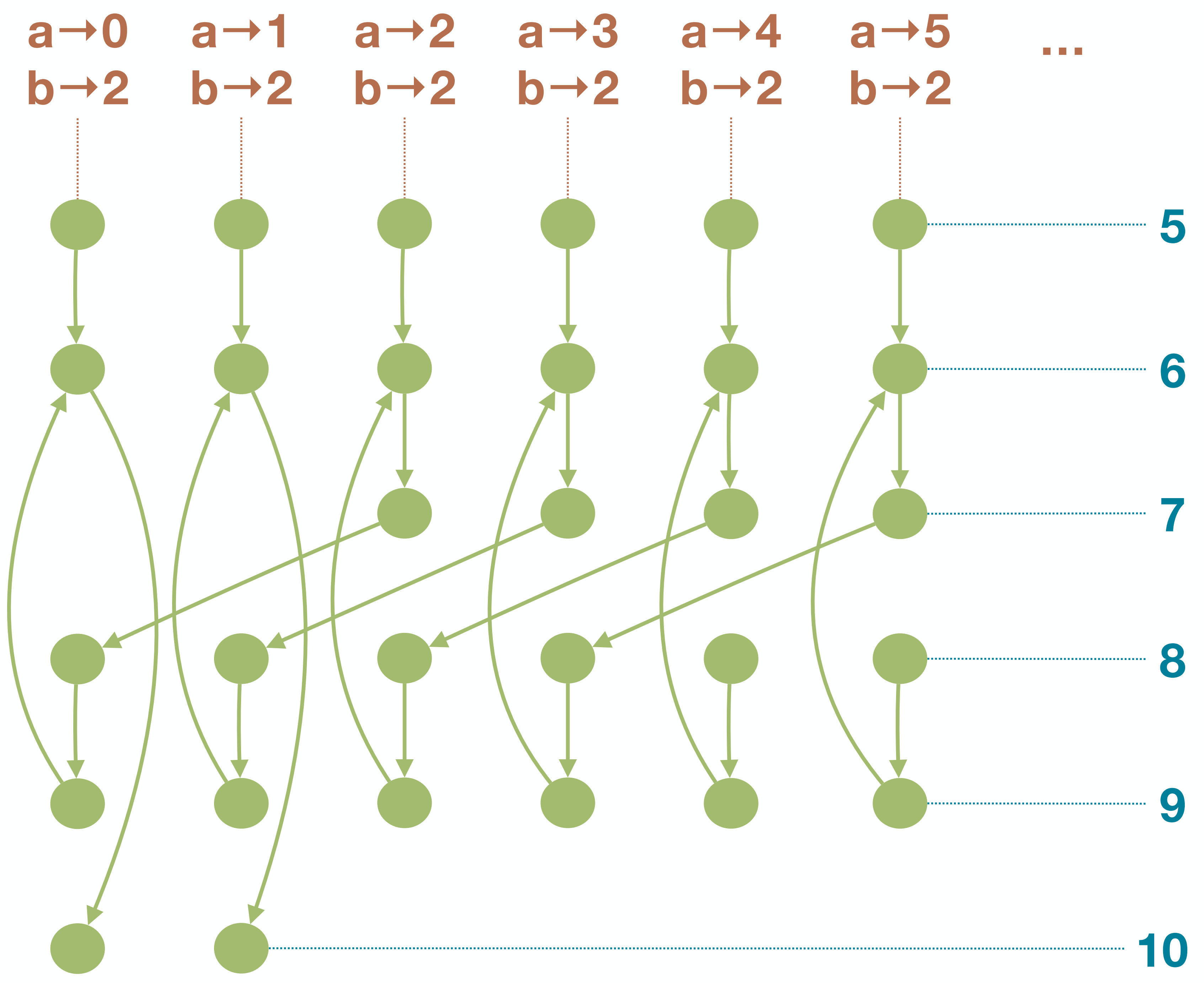
$$
\begin{aligned}
E[\![X]\!]\rho &\stackrel{\text{def}}{=} \{\rho(X)\} \\
E[\![c]\!]\rho &\stackrel{\text{def}}{=} \{c\} \\
E[\![[c_1, c_2]]\!]\rho &\stackrel{\text{def}}{=} \{x \in \mathbb{Z} \mid c_1 \leq x \leq c_2\} \\
E[\![-e]\!]\rho &\stackrel{\text{def}}{=} \{-x \mid x \in E[\![e]\!]\rho\} \\
E[\![e_1 + e_2]\!]\rho &\stackrel{\text{def}}{=} \{x_1 + x_2 \mid x_1 \in E[\![e]\!]\rho, x_2 \in E[\![e]\!]\rho\} \\
E[\![e_1 - e_2]\!]\rho &\stackrel{\text{def}}{=} \{x_1 - x_2 \mid x_1 \in E[\![e]\!]\rho, x_2 \in E[\![e]\!]\rho\} \\
&\vdots
\end{aligned}
$$

Transition Semantics

Program Executions as Discrete Transitions between States

- states: $\Sigma \stackrel{\text{def}}{=} \mathscr{L} \times (\mathbb{X} \rightarrow \mathbb{Z})$
- transition relation: $\tau \subseteq \Sigma \times \Sigma$

```
3
q ← 0
4
r ← a
5
while 6(r ≥ b) do
    7
    r ← r − b
    8
    q ← q + 1
    9
od
10
```



Transition Semantics

$$\begin{aligned} stmt ::=\ & {}^{\ell}X \leftarrow expr^{\ell} \\ \mid\ & \textsf{if}\ {}^{\ell}expr \bowtie 0\ \textsf{then}\ stmt\ \textsf{end}^{\ell} \\ \mid\ & \textsf{while}\ {}^{\ell}expr \bowtie 0\ \textsf{do}\ stmt\ \textsf{done}^{\ell} \\ \mid\ & stmt; stmt \end{aligned}$$

- transition relation: $\tau \subseteq \Sigma \times \Sigma$

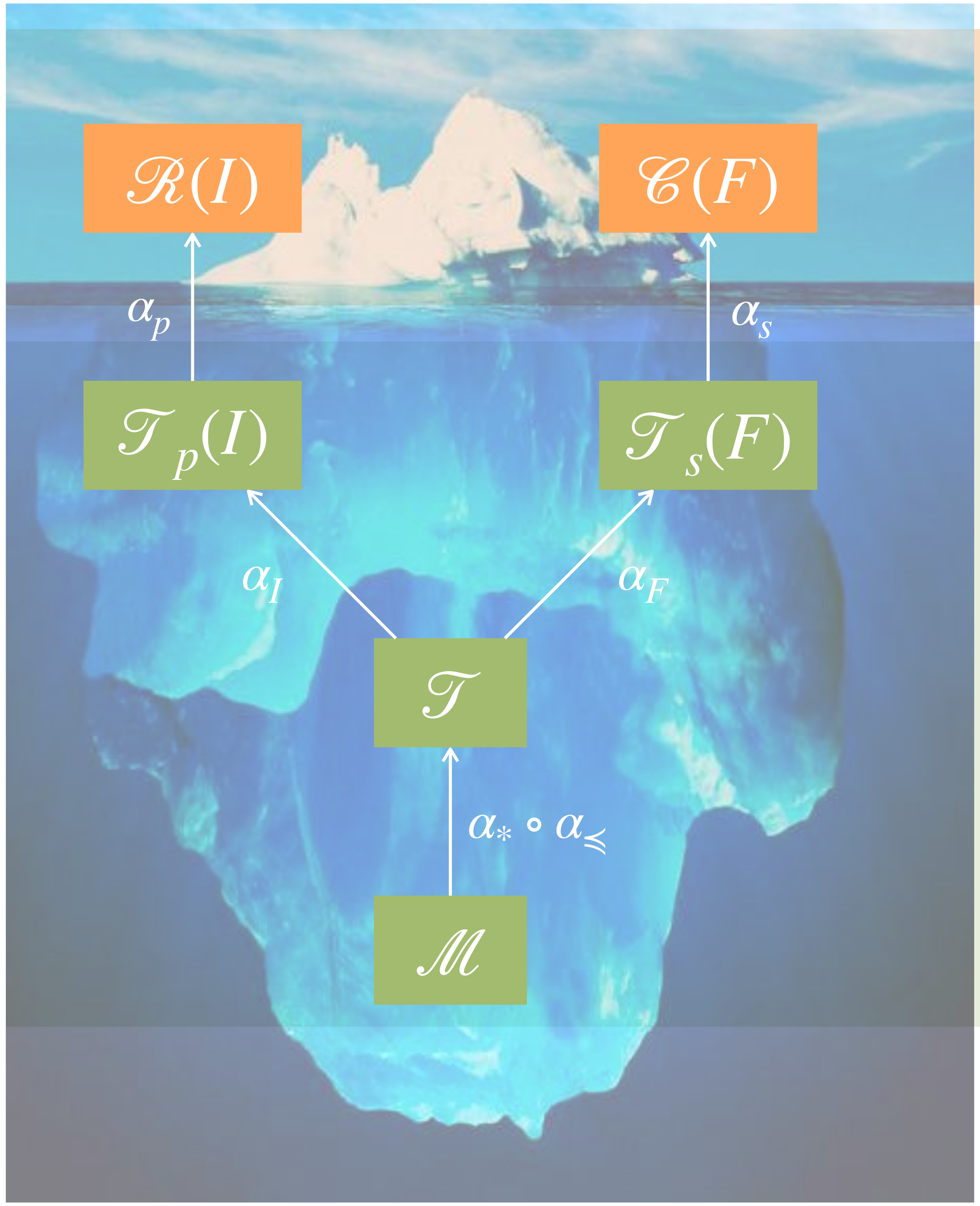
$$\tau[\![{}^{\ell_1}X \leftarrow e^{\ell_2}]\!] \stackrel{\text{def}}{=} \{((\ell_1, \rho), (\ell_2, \rho[X \mapsto v])) \mid \rho \in \mathscr{E}, v \in E[\![e]\!]\rho\}$$

$$\tau[\![\textbf{if}\ {}^{\ell_1}e \bowtie 0\ \textbf{then}\ {}^{\ell_2}s^{\ell_3}\ \textbf{end}^{\ell_4}]\!] \stackrel{\text{def}}{=}$$
$$\{((\ell_1, \rho), (\ell_2, \rho)) \mid \rho \in \mathscr{E}, \exists v \in E[\![e]\!]\rho : v \bowtie 0\} \cup \tau[\![{}^{\ell_2}s^{\ell_3}]\!]\rho \cup \{((\ell_3, \rho), (\ell_4, \rho)) \mid \rho \in \mathscr{E}\} \cup$$
$$\{((\ell_1, \rho), (\ell_4, \rho)) \mid \rho \in \mathscr{E}, \exists v \in E[\![e]\!]\rho : v \not\bowtie 0\}$$

$$\tau[\![\textbf{while}\ {}^{\ell_1}e \bowtie 0\ \textbf{then}\ {}^{\ell_2}s^{\ell_3}\ \textbf{done}^{\ell_4}]\!] \stackrel{\text{def}}{=}$$
$$\{((\ell_1, \rho), (\ell_2, \rho)) \mid \rho \in \mathscr{E}, \exists v \in E[\![e]\!]\rho : v \bowtie 0\} \cup \tau[\![{}^{\ell_2}s^{\ell_3}]\!]\rho \cup \{((\ell_3, \rho), (\ell_1, \rho)) \mid \rho \in \mathscr{E}\} \cup$$
$$\{((\ell_1, \rho), (\ell_4, \rho)) \mid \rho \in \mathscr{E}, \exists v \in E[\![e]\!]\rho : v \not\bowtie 0\}$$

$$\tau[\![s_1; s_2]\!] \stackrel{\text{def}}{=} \tau[\![s_1]\!] \cup \tau[\![s_2]\!]$$

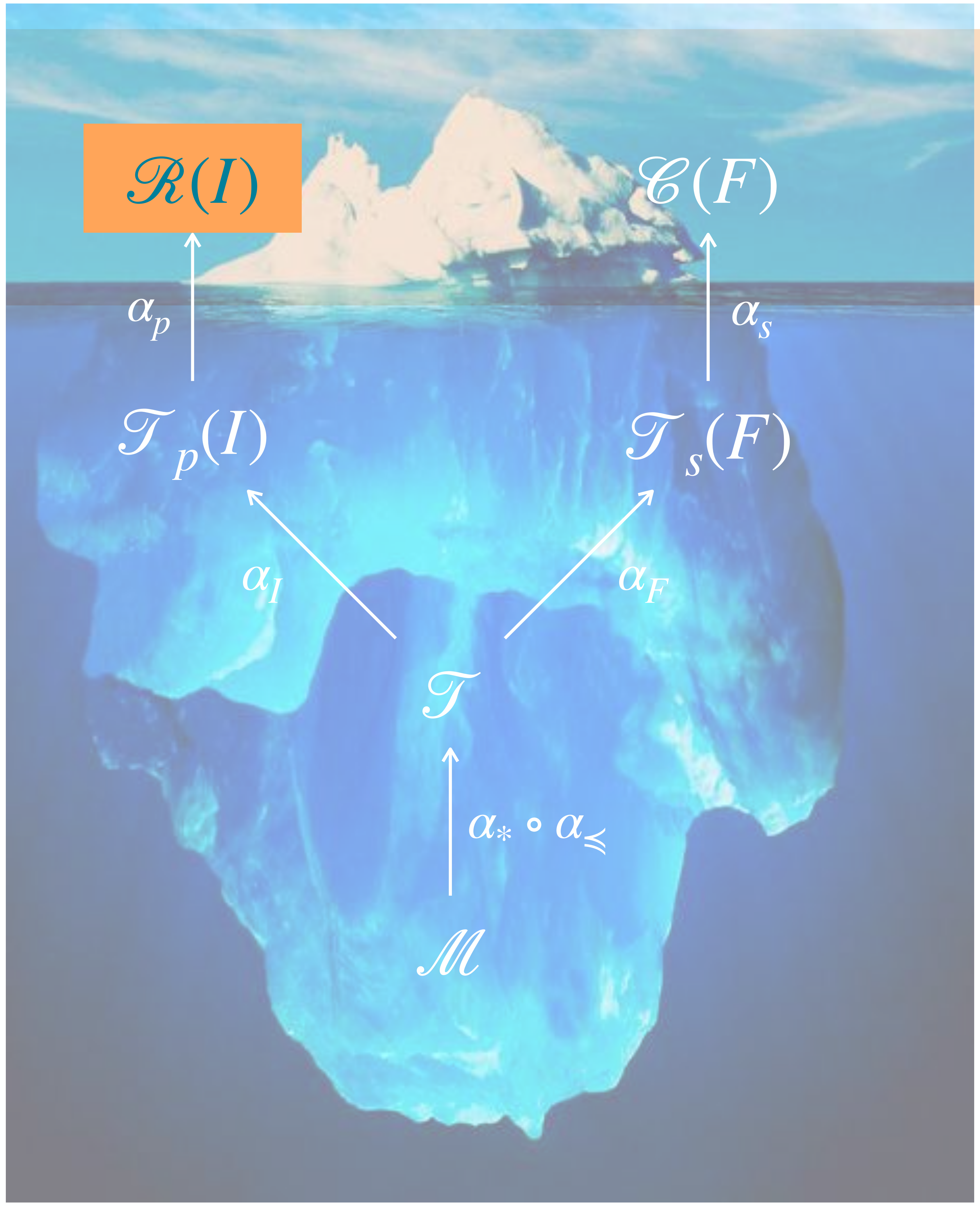
Hierarchy of Semantics



State Semantics

Trace Semantics

Forward Reachability Semantics



State Semantics

Order Theory

Partial Orders and Partially Ordered Sets

A binary relation $\sqsubseteq \in X \times X$ over a set X is called a **partial order** when it is:

- **reflexive** $\forall x \in X \colon x \sqsubseteq x$
- **antisymmetric** $\forall x, y \in X \colon (x \sqsubseteq y) \wedge (y \sqsubseteq x) \Rightarrow x = y$
- **transitive** $\forall x, y, z \in X \colon (x \sqsubseteq y) \wedge (y \sqsubseteq z) \Rightarrow x \sqsubseteq z$

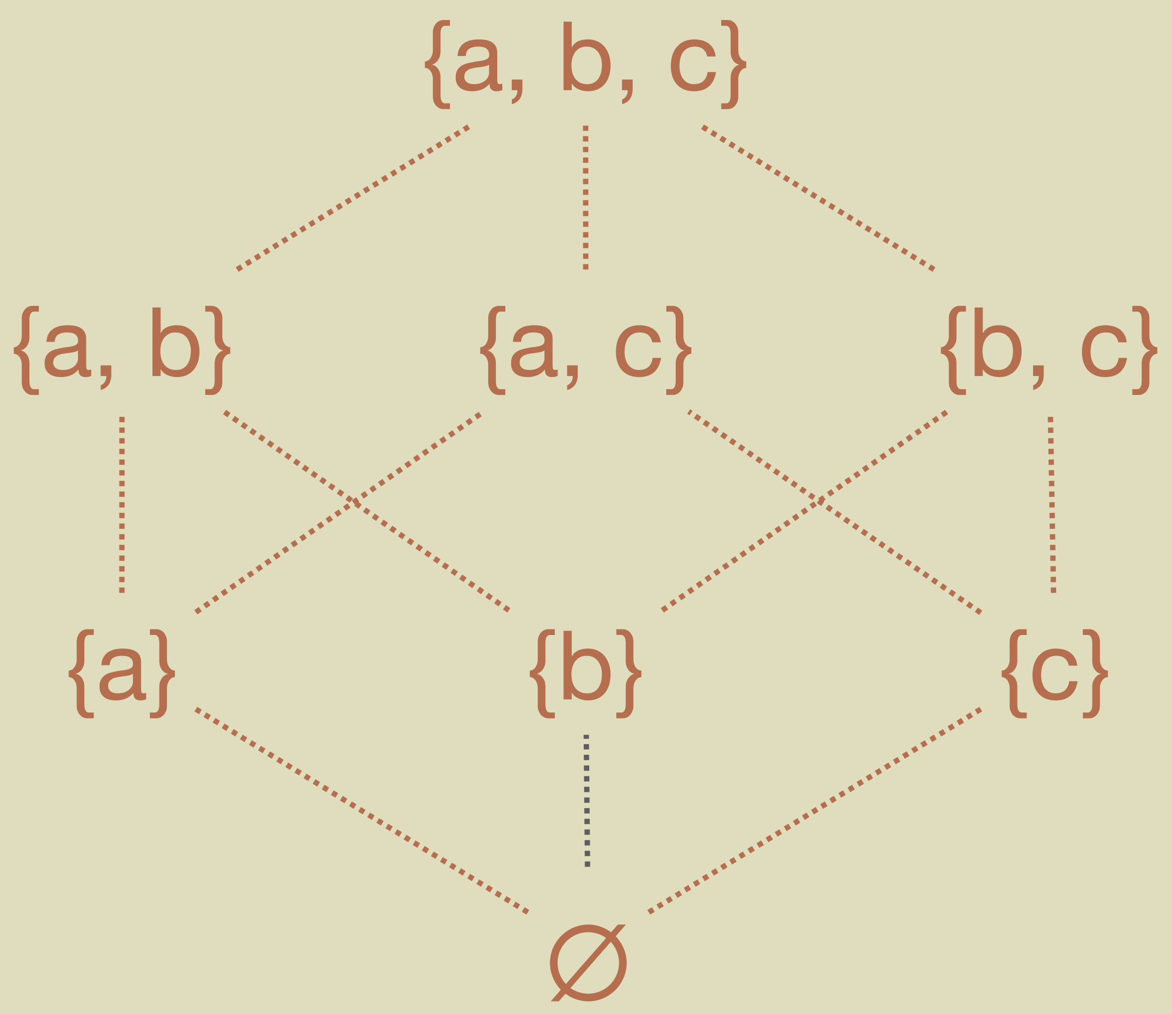
$\langle X, \sqsubseteq \rangle$ is a **partially ordered set** or **poset**

Example

$\langle \mathscr{P}(\Sigma), \subseteq \rangle$ **powerset** of the set of program states ordered by **set inclusion**

Order Theory

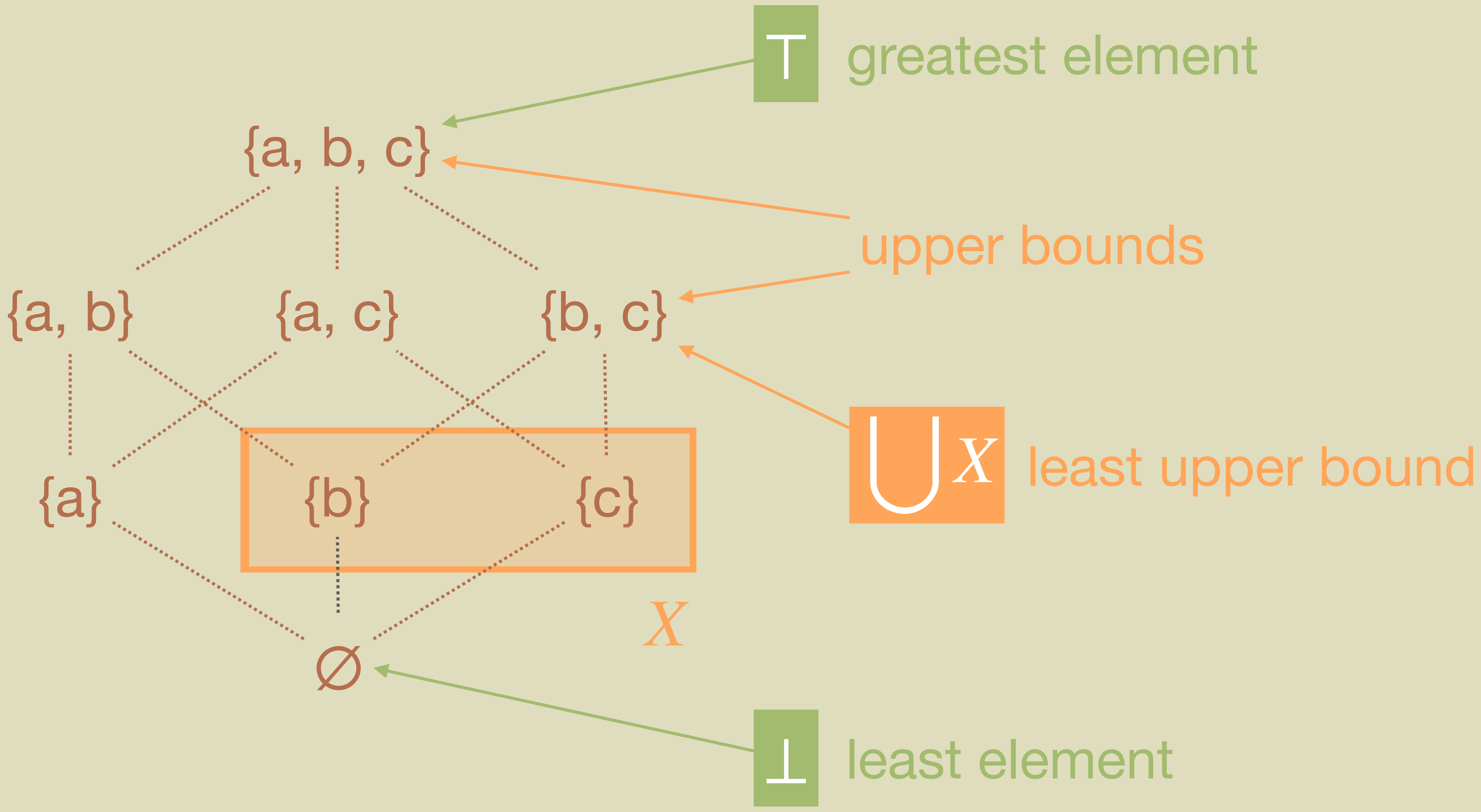
Hasse Diagram



$\langle \mathscr{P}(\{a, b, c\}), \subseteq \rangle$

Order Theory

(Least) Upper Bounds



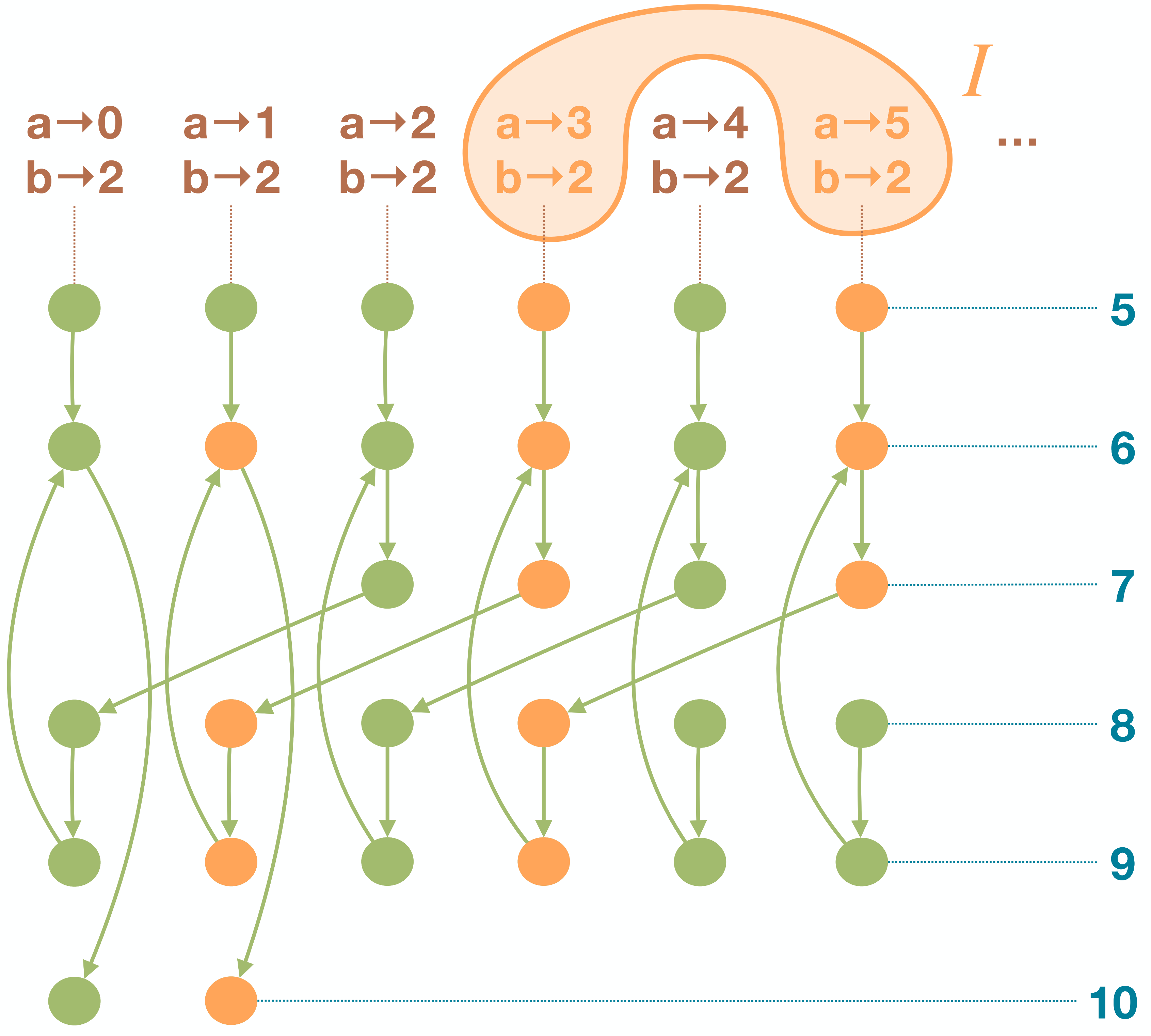
$\langle \mathscr{P}(\{a, b, c\}), \subseteq \rangle$

Forward Reachability Semantics

Program States Reachable From $I \in \mathscr{P}(\Sigma)$

- $\mathscr{R}(I) \in \mathscr{P}(\Sigma)$ $\quad\quad \mathscr{R}(I) \stackrel{\text{def}}{=} \{s \mid \exists n \geq 0, s_0, \ldots, s_n : s_0 \in I \land s = s_n \land \forall i : \langle s_i, s_{i+1} \rangle \in \tau\}$

```
3
q ← 0
4
r ← a
5
while 6(r ≥ b) do
   7
   r ← r − b
   8
   q ← q + 1
   9
od
10
```



Order Theory

Fixpoints

Given a partially ordered set $\langle X, \sqsubseteq \rangle$ and a function $f\colon X \to X$

- a **fixpoint** of f is an element $x \in X$ such that $x = f(x)$
- a **pre-fixpoint** of f is an element $x \in X$ such that $x \sqsubseteq f(x)$
- a **post-fixpoint** of f is an element $x \in X$ such that $f(x) \sqsubseteq x$

$$\mathbf{fp}(f) \stackrel{\text{def}}{=} \{x \in X \mid x = f(x)\}$$

$$\mathbf{lfp}_x^{\sqsubseteq} f \stackrel{\text{def}}{=} \min_{\sqsubseteq}\{y \in \text{fp}(f) \mid x \sqsubseteq y\}$$

$$\mathbf{gfp}_x^{\sqsubseteq} f \stackrel{\text{def}}{=} \max_{\sqsubseteq}\{y \in \text{fp}(f) \mid y \sqsubseteq x\}$$

Forward Reachability Semantics

Least Fixpoint Formulation

computational order

$$\mathscr{R}(I) = \mathbf{lfp}^{\subseteq}_{\emptyset} F_r$$

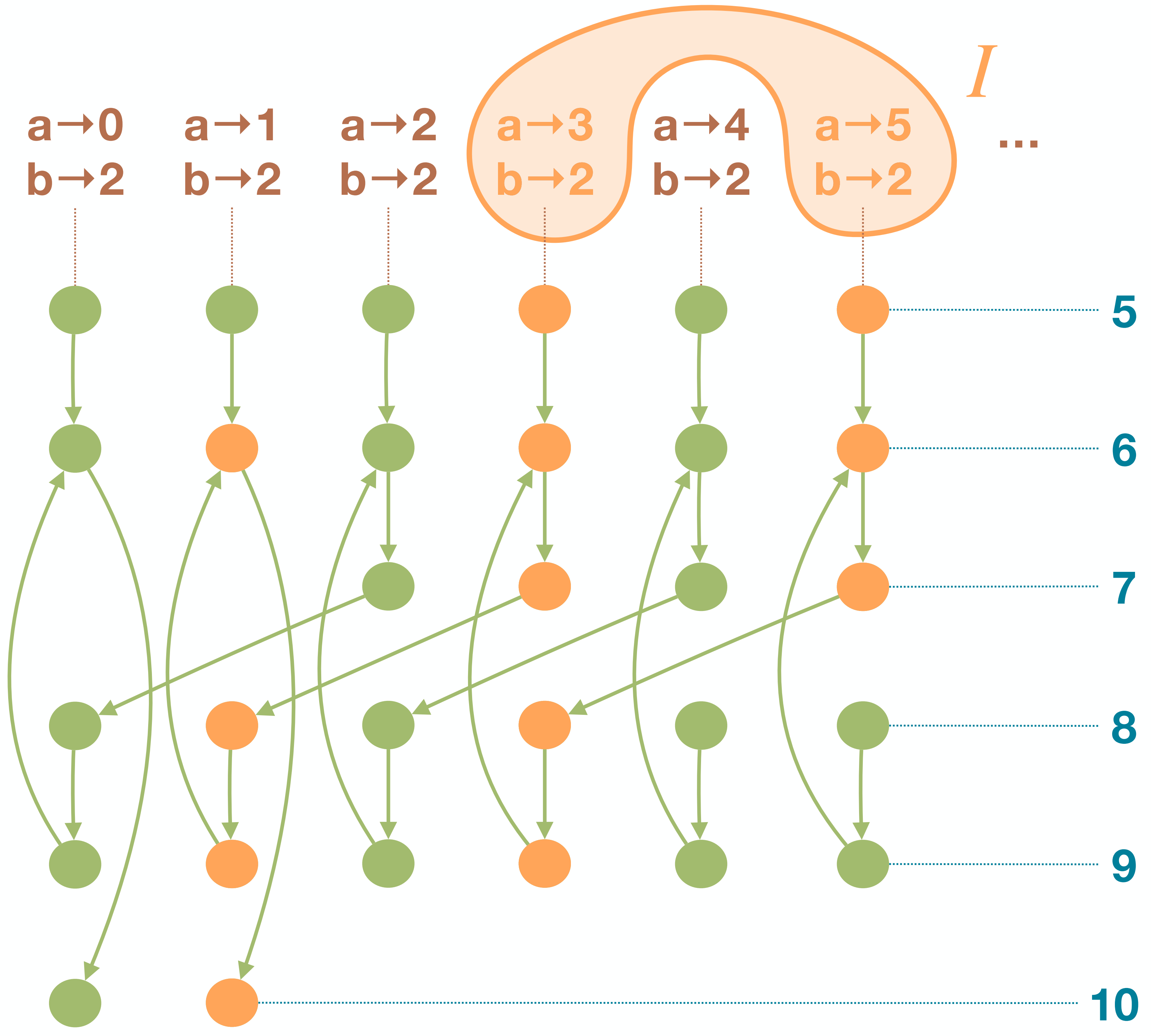
$$F_r(S) \stackrel{\text{def}}{=} I \cup \mathbf{post}(S)$$

Definition

Given a transition system $\langle \Sigma, \tau \rangle$, the **image function post** $: \mathscr{P}(\Sigma) \rightarrow \mathscr{P}(\Sigma)$ maps a set of program states $X \in \mathscr{P}(\Sigma)$ to the set of their successors with respect to the transition relation τ:

$$\mathbf{post}(X) \stackrel{\text{def}}{=} \{s' \in \Sigma \mid \exists s \in X \colon \langle s, s' \rangle \in \tau\}$$

```
3
q ← 0
4
r ← a
5
while 6(r ≥ b) do
    7
    r ← r − b
    8
    q ← q + 1
    9
od
10
```



Forward Reachability

Denotational Formulation

$$stmt ::= {}^{\ell}X \leftarrow expr^{\ell}$$
$$\mid \textbf{if } {}^{\ell}expr \bowtie 0 \textbf{ then } stmt \textbf{ end}^{\ell}$$
$$\mid \textbf{while } {}^{\ell}expr \bowtie 0 \textbf{ do } stmt \textbf{ done}^{\ell}$$
$$\mid stmt; stmt$$

$$\mathscr{R}[\![{}^{\ell_1}X \leftarrow e^{\ell_2}]\!]S \stackrel{\text{def}}{=} \{(\ell_2, \rho[X \mapsto v]) \mid (\ell_1, \rho) \in S, v \in E[\![e]\!]\rho\}$$

$$\mathscr{R}[\![\textbf{if } {}^{\ell_1} e \bowtie 0 \textbf{ then } {}^{\ell_2}s^{\ell_3} \textbf{ end}^{\ell_4}]\!]S \stackrel{\text{def}}{=}$$
$$\{(\ell_4, \rho) \mid (\ell_3, \rho) \in \mathscr{R}[\![{}^{\ell_2}s^{\ell_3}]\!]\{(\ell_2, \rho) \mid (\ell_1, \rho) \in S, \exists v \in E[\![e]\!]\rho : v \bowtie 0\}\} \cup$$
$$\{(\ell_4, \rho) \mid (\ell_1, \rho) \in S, \exists v \in E[\![e]\!]\rho : v \not\bowtie 0\}$$

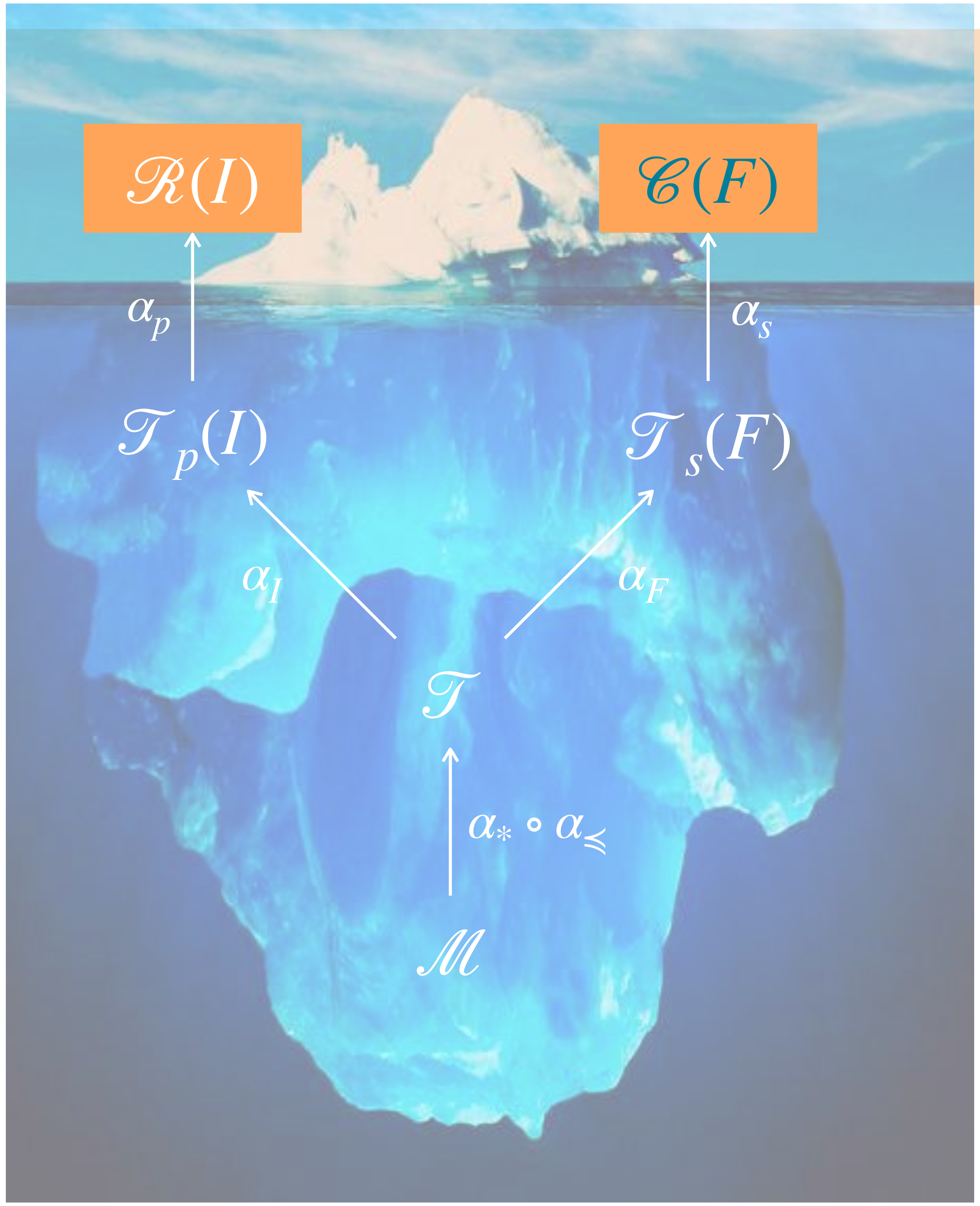
$$\mathscr{R}[\![\textbf{while } {}^{\ell_1} e \bowtie 0 \textbf{ then } {}^{\ell_2}s^{\ell_3} \textbf{ done}^{\ell_4}]\!]S \stackrel{\text{def}}{=}$$
$$\{(\ell_4, \rho) \mid (\ell_1, \rho) \in \mathsf{lfp}^{\subseteq}_{\emptyset} F_r, \exists v \in E[\![e]\!]\rho : v \not\bowtie 0\}$$

where

$$F_r(Y) \stackrel{\text{def}}{=} S \cup \{(\ell_1, \rho) \mid (\ell_3, \rho) \in \mathscr{R}[\![{}^{\ell_2}s^{\ell_3}]\!]\{(\ell_2, \rho) \mid (\ell_1, \rho) \in Y, \exists v \in E[\![e]\!]\rho : v \bowtie 0\}\}$$

$$\mathscr{R}[\![s_1; s_2]\!]S \stackrel{\text{def}}{=} \mathscr{R}[\![s_2]\!](\mathscr{R}[\![s_1]\!]S)$$

Backward Reachability Semantics



State Semantics

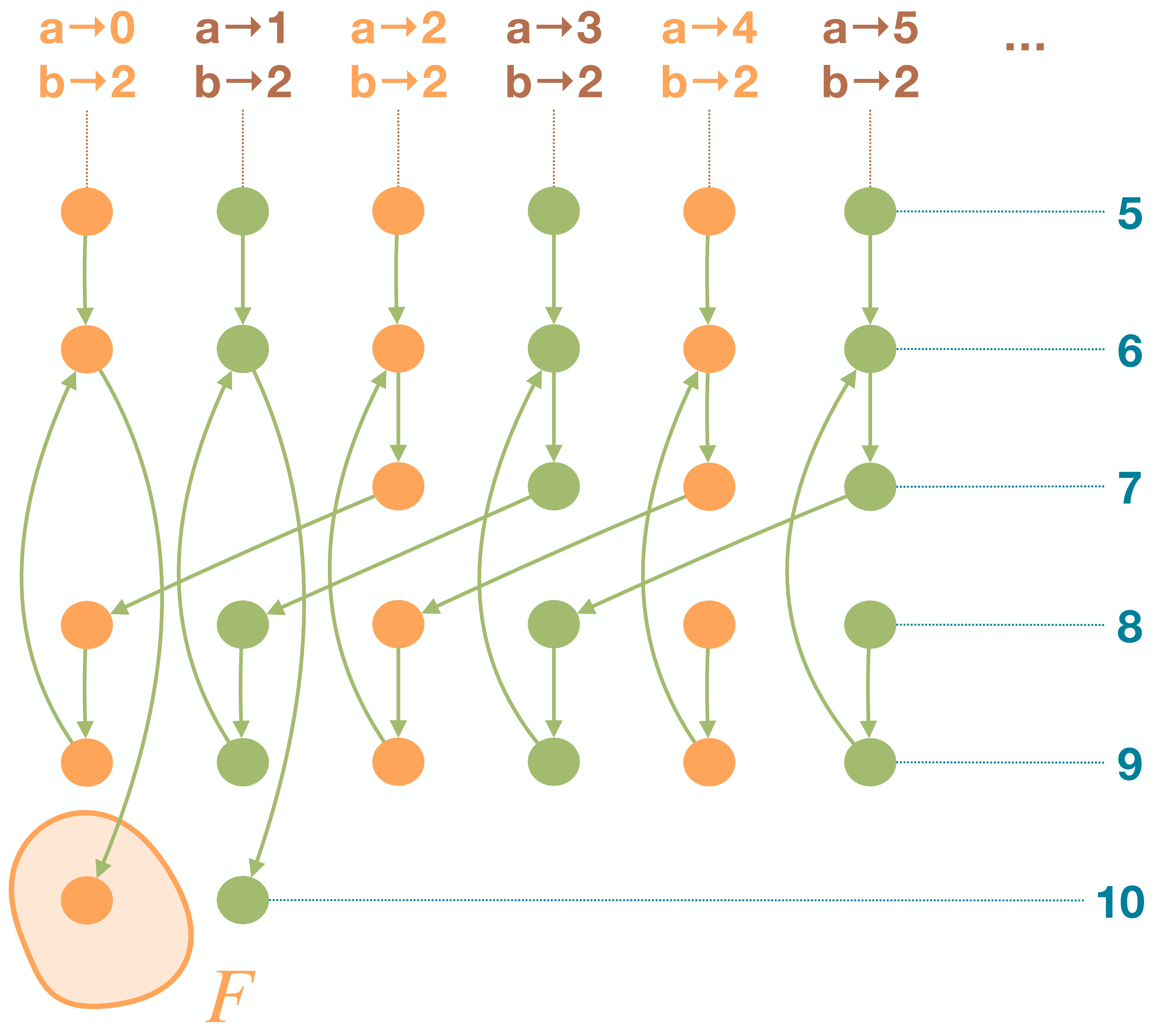
Backward Reachability Semantics

Program States Reaching $F \in \mathscr{P}(\Sigma)$

- $\mathscr{C}(F) \in \mathscr{P}(\Sigma)$

$$\mathscr{C}(F) \stackrel{\text{def}}{=} \{s \mid \exists n \geq 0, s_0, \ldots, s_n : s = s_0 \wedge s_n \in F \wedge \forall i : \langle s_i, s_{i+1} \rangle \in \tau\}$$

```
3
q ← 0
4
r ← a
5
while 6(r ≥ b) do
    7
    r ← r − b
    8
    q ← q + 1
    9
od
10
```



Backward Reachability Semantics

Least Fixpoint Formulation

$$\mathscr{C}(F) = \mathsf{lfp}^{\subseteq}_{\emptyset} F_c$$

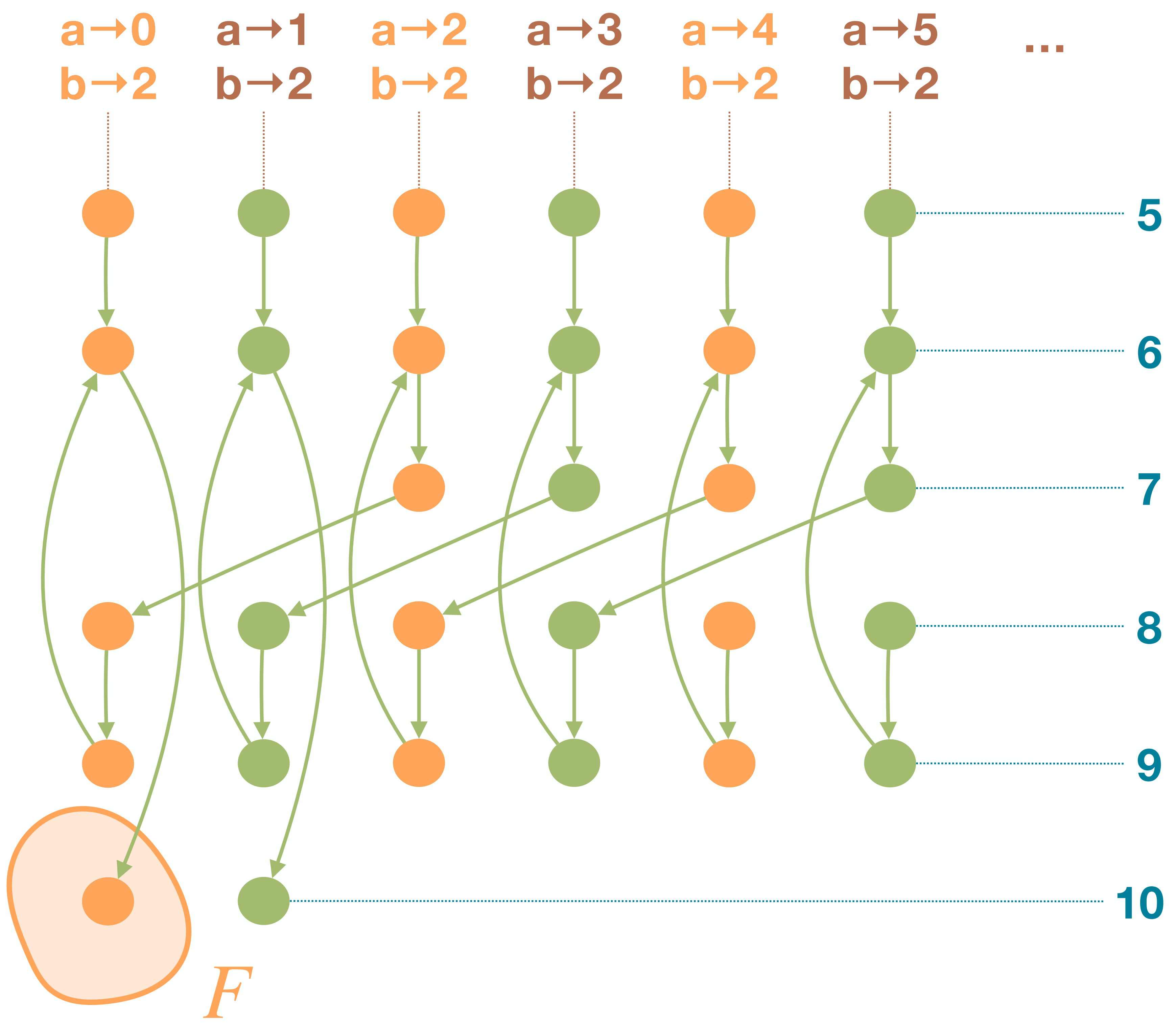
$$F_c(S) \stackrel{\text{def}}{=} F \cup \mathbf{pre}(S)$$

Definition

Given a transition system $\langle \Sigma, \tau \rangle$, the **preimage function pre** $: \mathscr{P}(\Sigma) \rightarrow \mathscr{P}(\Sigma)$ maps a set of program states $X \in \mathscr{P}(\Sigma)$ to the set of their predecessors with respect to the transition relation τ:

$$\mathbf{pre}(X) \stackrel{\text{def}}{=} \{ s \in \Sigma \mid \exists s' \in X : \langle s, s' \rangle \in \tau \}$$

```
3
q ← 0
4
r ← a
5
while 6(r ≥ b) do
    7
    r ← r − b
    8
    q ← q + 1
    9
od
10
```



Backward Reachability

Denotational Formulation

$$stmt ::= {}^{\ell}X \leftarrow expr^{\ell} \mid \textbf{if } {}^{\ell}expr \bowtie 0 \textbf{ then } stmt \textbf{ end}^{\ell} \mid \textbf{while } {}^{\ell}expr \bowtie 0 \textbf{ do } stmt \textbf{ done}^{\ell} \mid stmt; stmt$$

$$\mathscr{C}[\![{}^{\ell_1}X \leftarrow e^{\ell_2}]\!]S \stackrel{\text{def}}{=} \{(\ell_1, \rho) \mid (\ell_2, \rho[X \mapsto v]) \in S, v \in E[\![e]\!]\rho\}$$

$$\mathscr{C}[\![\textbf{if } {}^{\ell_1} e \bowtie 0 \textbf{ then } {}^{\ell_2}s^{\ell_3} \textbf{ end}^{\ell_4}]\!]S \stackrel{\text{def}}{=}$$
$$\{(\ell_1, \rho) \mid (\ell_2, \rho) \in \mathscr{C}[\![{}^{\ell_2}s^{\ell_3}]\!]\{(\ell_3, \rho) \mid (\ell_4, \rho) \in S\}, \exists v \in E[\![e]\!]\rho : v \bowtie 0\} \cup$$
$$\{(\ell_1, \rho) \mid (\ell_4, \rho) \in S, \exists v \in E[\![e]\!]\rho : v \not\bowtie 0\}$$

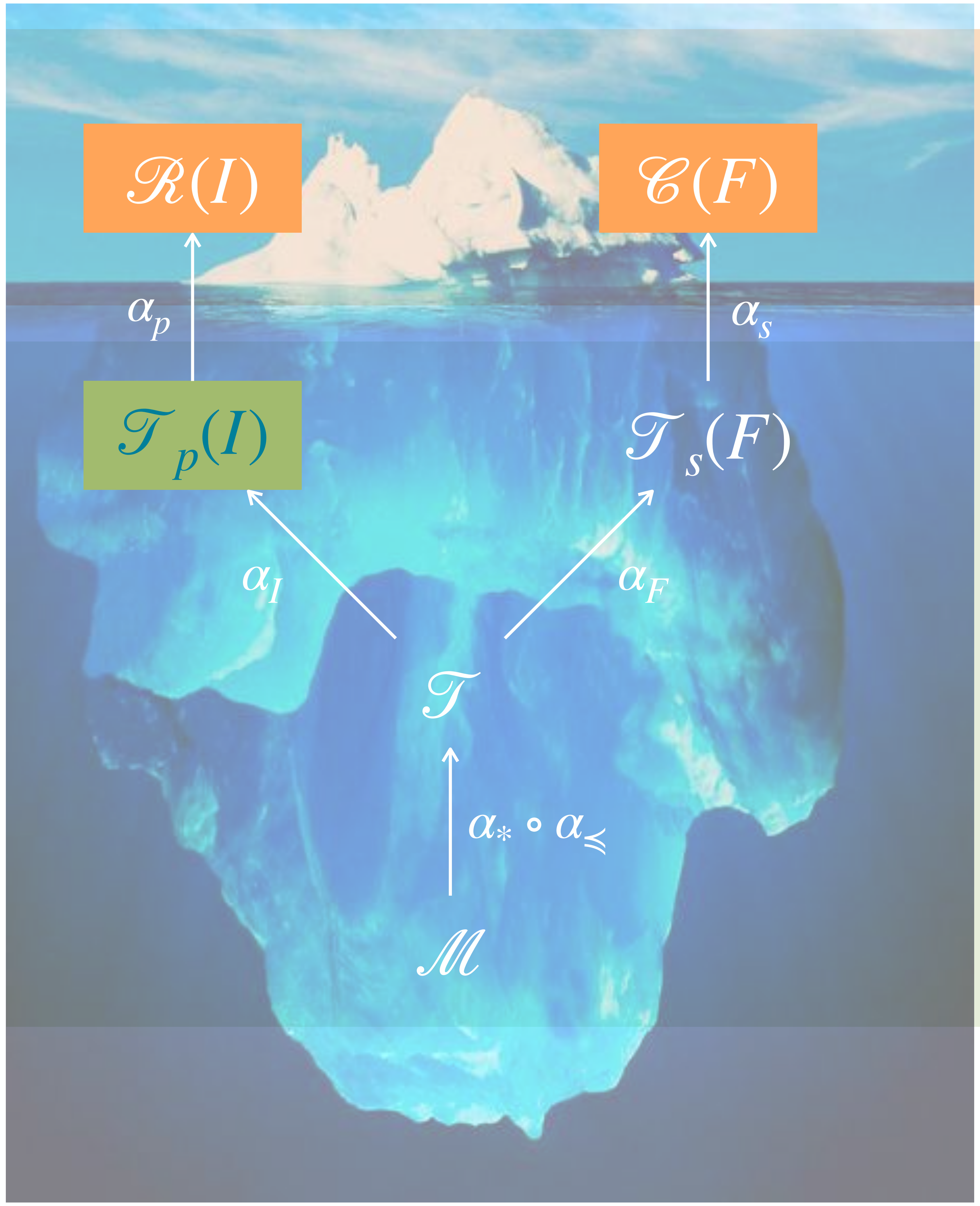
$$\mathscr{C}[\![\textbf{while } {}^{\ell_1} e \bowtie 0 \textbf{ then } {}^{\ell_2}s^{\ell_3} \textbf{ done}^{\ell_4}]\!]S \stackrel{\text{def}}{=} \mathsf{lfp}^{\subseteq}_{\emptyset} F_c$$

where

$$F_c(Y) \stackrel{\text{def}}{=} \{(\ell_1, \rho) \mid (\ell_4, \rho) \in S, \exists v \in E[\![e]\!]\rho : v \not\bowtie 0\} \cup$$
$$\{(\ell_1, \rho) \mid (\ell_2, \rho) \in \mathscr{C}[\![{}^{\ell_2}s^{\ell_3}]\!]\{(\ell_3, \rho) \mid (\ell_1, \rho) \in Y\}, \exists v \in E[\![e]\!]\rho : v \bowtie 0\}$$

$$\mathscr{C}[\![s_1; s_2]\!]S \stackrel{\text{def}}{=} \mathscr{C}[\![s_1]\!](\mathscr{C}[\![s_2]\!]S)$$

Prefix Trace Semantics



State Semantics

Trace Semantics

Program State Sequences

- ϵ **empty** sequence
- $s_0, \ldots, s_{n-1}$ sequence of **length n**
- Σ^n set of **sequences of length n**
- $\Sigma^* \stackrel{\text{def}}{=} \bigcup_{i \in \mathbb{N}} \Sigma^i$ set of **all finite sequences**
- Σ^ω set of **all infinite sequences**
- $\Sigma^\infty \stackrel{\text{def}}{=} \Sigma^* \cup \Sigma^\omega$ set of **all sequences**

Operations on Sequences

- **concatenation**: $(s_0, \ldots, s_n) \cdot (s'_0, \ldots, s'_n) \stackrel{\text{def}}{=} s_0, \ldots, s_n s'_0, \ldots, s'_n$
  $A \cdot B \stackrel{\text{def}}{=} \{a \cdot b \mid a \in A \wedge b \in B\}$
- **merging**: $(s_0, \ldots, s) \,;\, (s, s'_1, \ldots, s'_n) \stackrel{\text{def}}{=} s_0, \ldots, s s'_1, \ldots, s'_n$
  $A \,;\, B \stackrel{\text{def}}{=} \{a \,;\, b \mid a \in A \wedge b \in B\}$

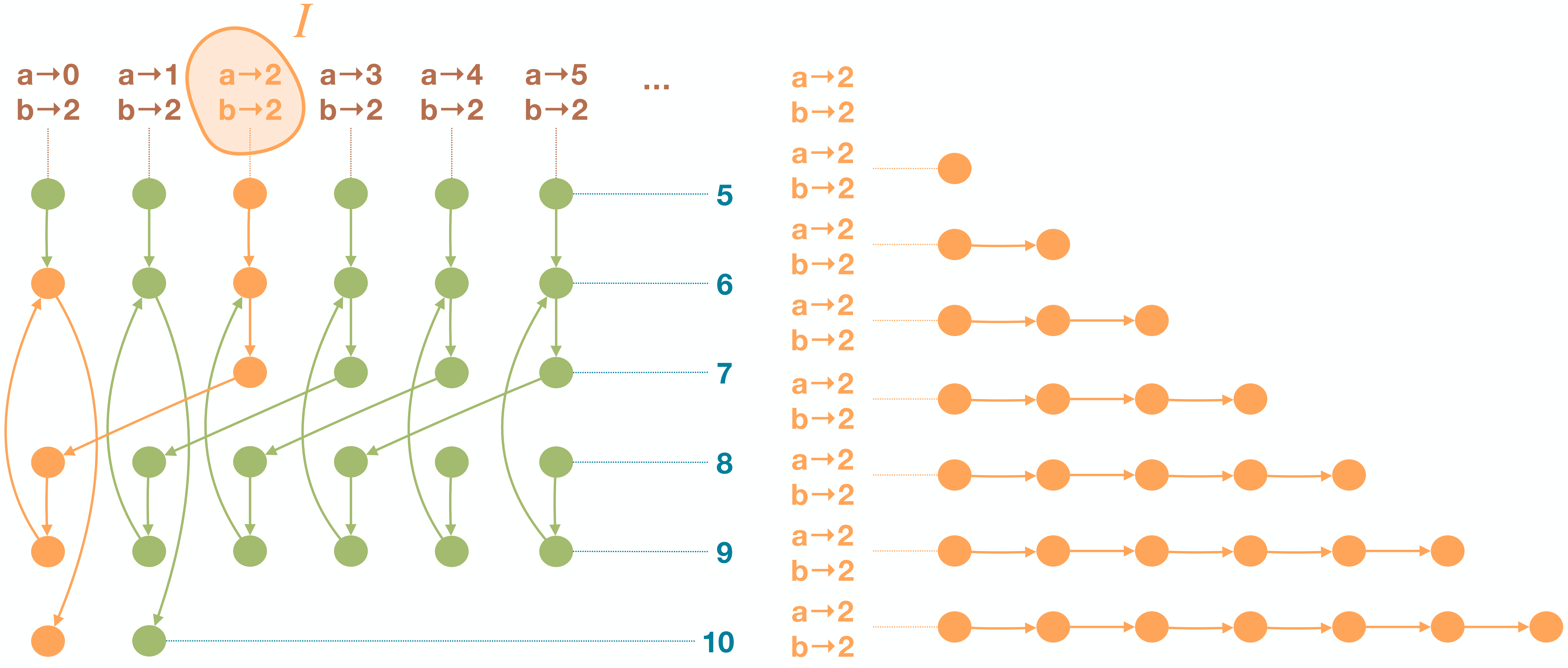
Prefix Trace Semantics

Finite Partial Program Traces Starting From $I \in \mathscr{P}(\Sigma)$

- $\mathscr{T}_p(I) \in \mathscr{P}(\Sigma^*)$

$$\mathscr{T}_p(I) \stackrel{\text{def}}{=} \{s_0, \ldots, s_n \mid n \geq 0 \land s_0 \in I \land \forall i : \langle s_i, s_{i+1} \rangle \in \tau\}$$

```
3
q ← 0
4
r ← a
5
while 6(r ≥ b) do
    7
    r ← r − b
    8
    q ← q + 1
    9
od
10
```

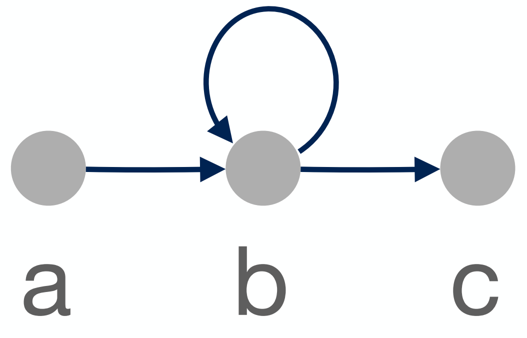


Prefix Trace Semantics

Least Fixpoint Formulation

$$\mathcal{T}_p(I) = \mathbf{lfp}^{\subseteq}_{\emptyset} F_p$$

$$F_p(T) \stackrel{\text{def}}{=} I \cup T ; \tau$$

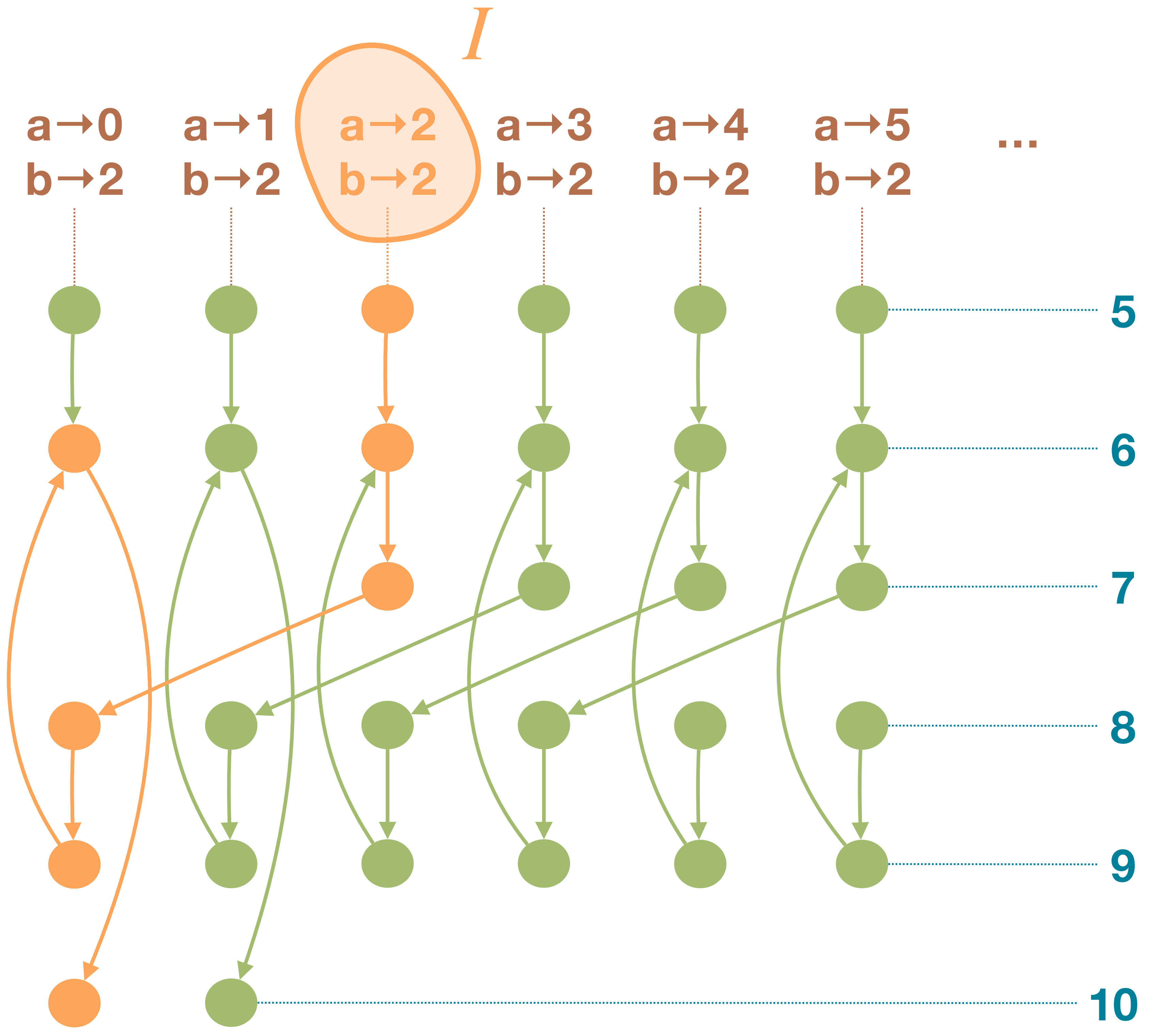


$I \stackrel{\text{def}}{=} \{a\}$

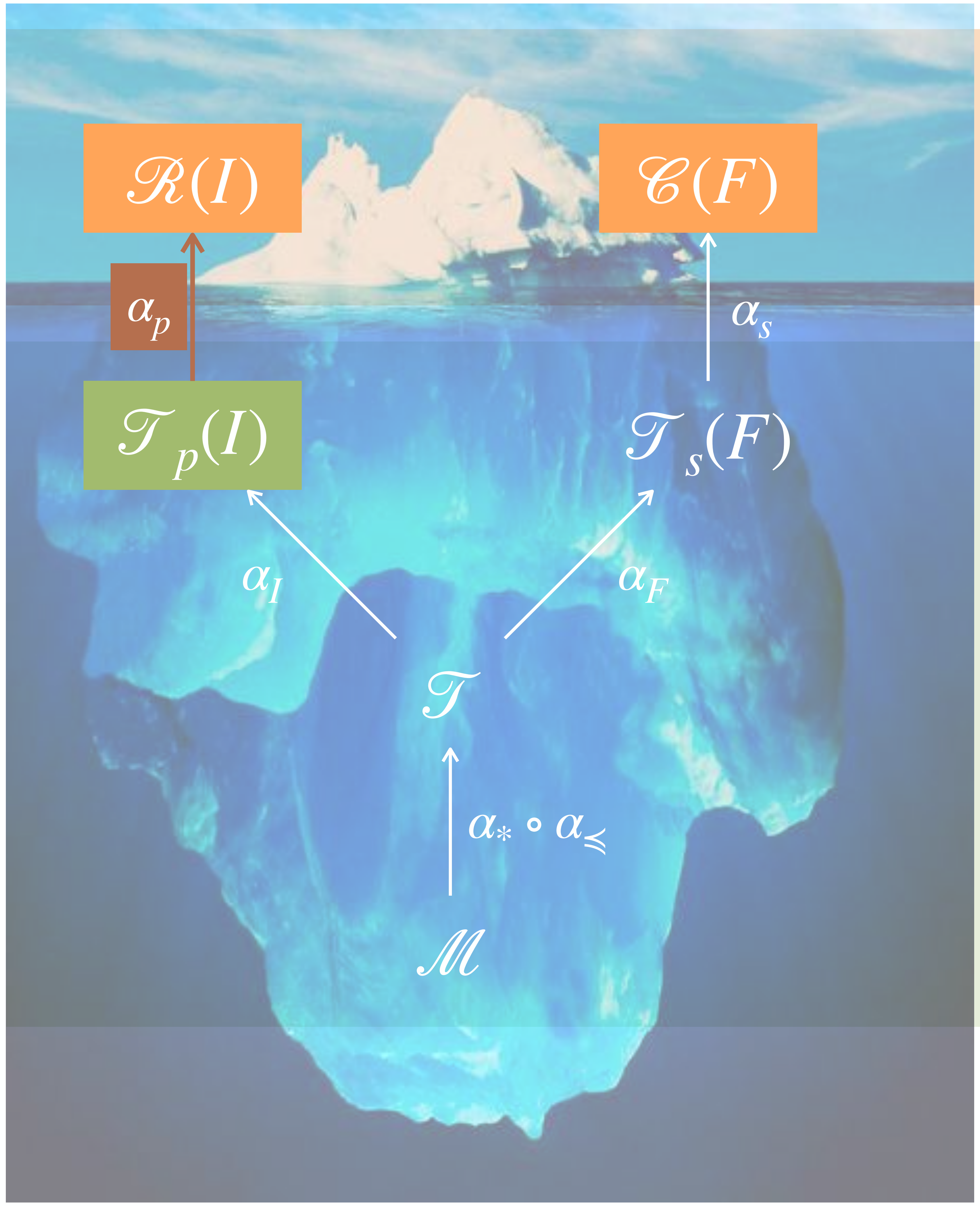
- $F_p^0(\emptyset) = \emptyset$
- $F_p^1(F_p^0) = I = \{a\}$
- $F_p^2(F_p^1) = \{a, ab\}$
- $F_p^3(F_p^2) = \{a, ab, abb, abc\}$

$$\mathcal{T}_p(I) = \{a, ab^i, ab^ic \mid i \geq 1\}$$

```
3
q ← 0
4
r ← a
5
while 6(r ≥ b) do
    7
    r ← r − b
    8
    q ← q + 1
    9
od
10
```



Forward Reachable State Abstraction



State Semantics

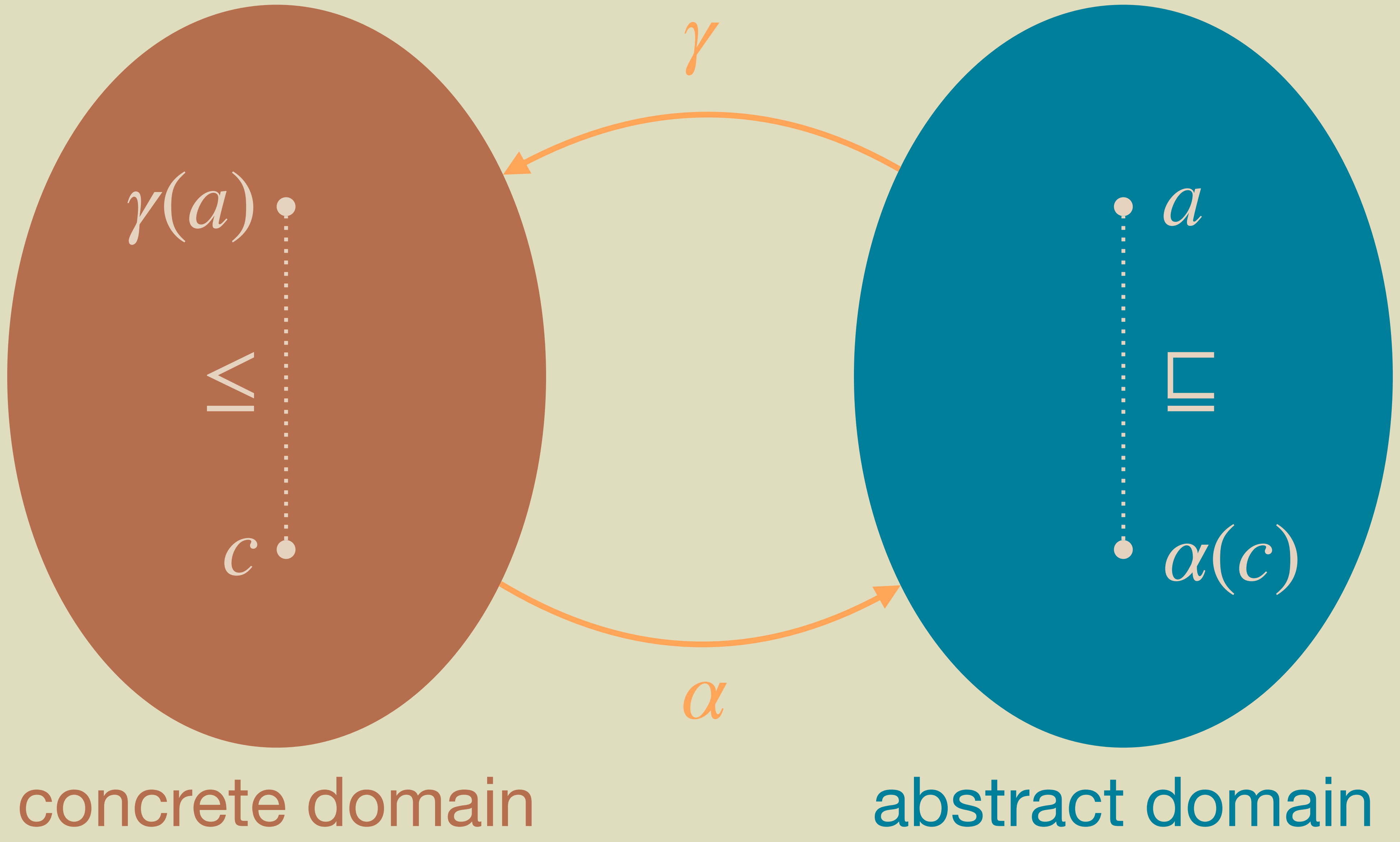
Trace Semantics

Order Theory

Galois Connections

A **Galois connection** between two posets $\langle C, \leq \rangle$ and $\langle A, \sqsubseteq \rangle$ is a pair of an **lower adjoint** or **abstraction function** $\alpha\colon C \rightarrow A$ and a **upper adjoint** or **concretization function** $\gamma\colon A \rightarrow C$ such that:

$$\forall c \in C, a \in A : \alpha(c) \sqsubseteq a \Leftrightarrow c \leq \gamma(a)$$



Order Theory

Galois Connections

Example:

$$\begin{tikzcd}[column sep=huge] \langle \mathscr{P}(\mathbb{Z}), \subseteq \rangle \arrow[r, bend right, "\alpha"'] & \langle \mathbb{I}, \sqsubseteq \rangle \arrow[l, bend right, "\gamma"'] \end{tikzcd}$$

$$\mathbb{I} \stackrel{\text{def}}{=} (\mathbb{Z} \cup \{-\infty\}) \times (\mathbb{Z} \cup \{\infty\})$$

$$(a, b) \sqsubseteq (c, d) \stackrel{\text{def}}{\Leftrightarrow} c \leq a \wedge b \leq d$$

$$\alpha(X) \stackrel{\text{def}}{=} (\min X, \max X)$$

$$\gamma((a, b)) \stackrel{\text{def}}{=} \{x \in \mathbb{Z} \mid a \leq x \leq b\}$$

Forward Reachable State Abstraction

a state in the forward reachability semantics corresponds to a partial program trace ending in this state

approximation order

$$\begin{tikzcd} \langle \mathscr{P}(\Sigma^*), \subseteq \rangle \arrow[r, bend right, "\alpha_p"'] & \langle \mathscr{P}(\Sigma), \subseteq \rangle \arrow[l, bend right, "\gamma_p"'] \end{tikzcd}$$

$$\alpha_p(T) \stackrel{\text{def}}{=} \{s \in \Sigma \mid \exists s_0, \ldots, s \in T\}$$

$$\gamma_p(S) \stackrel{\text{def}}{=} \{s_0, \ldots, s \in \Sigma^* \mid s \in S\}$$

Order Theory

Total Orders

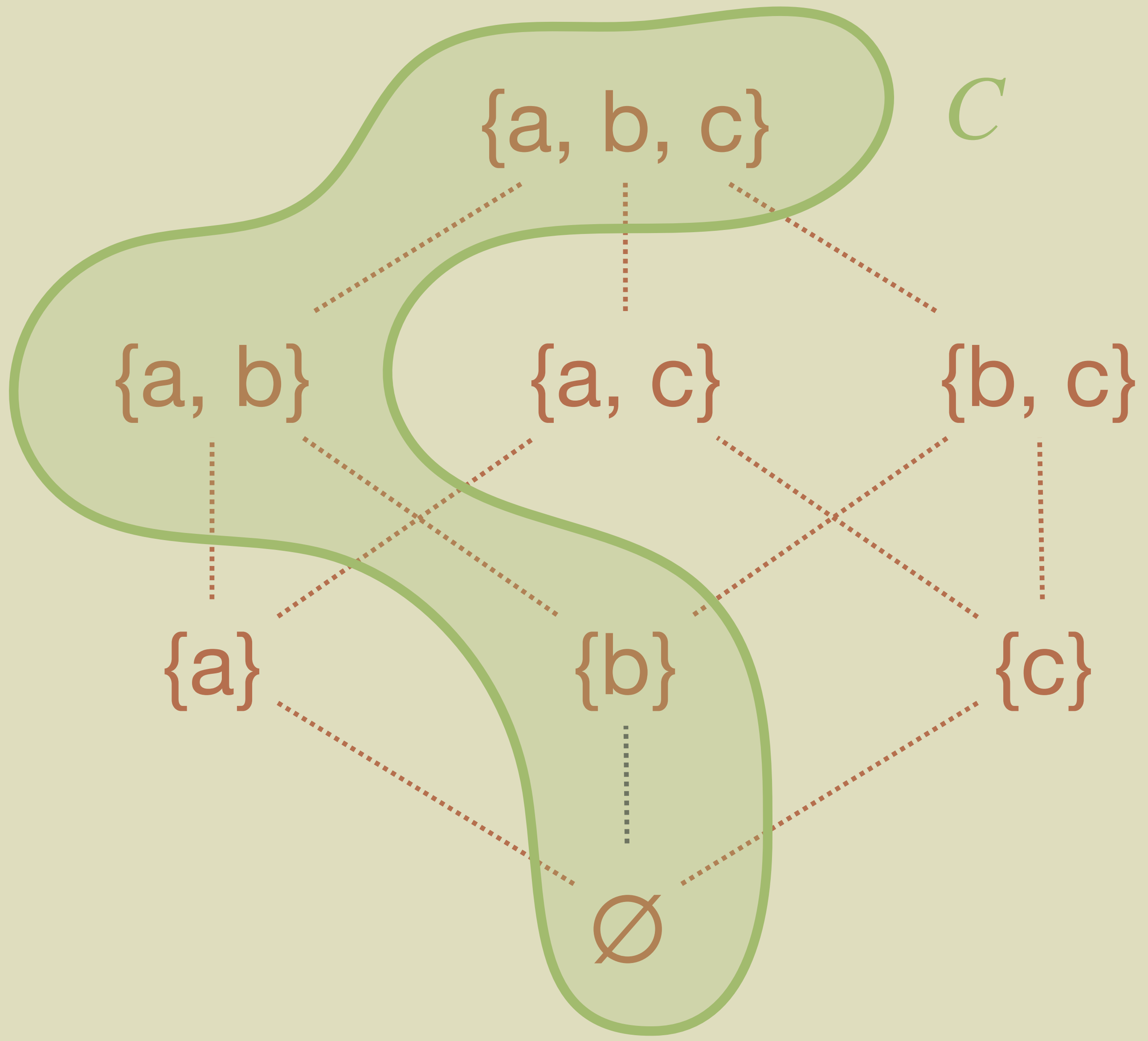
A binary relation $\sqsubseteq \in X \times X$ over a set X is called a **total order** when it is:

- **reflexive** $\forall x \in X\colon x \sqsubseteq x$
- **antisymmetric** $\forall x, y \in X\colon (x \sqsubseteq y) \wedge (y \sqsubseteq x) \Rightarrow x = y$
- **transitive** $\forall x, y, z \in X\colon (x \sqsubseteq y) \wedge (y \sqsubseteq z) \Rightarrow x \sqsubseteq z$
- **total** $\forall x, y, z \in X\colon (x \sqsubseteq y) \vee (y \sqsubseteq x)$

Order Theory

Chains and Complete Partial Orders

A **chain** is a totally ordered subset C of a poset $\langle X, \sqsubseteq \rangle$



$\langle X, \sqsubseteq \rangle$ is a **complete partial order** if every chain $C \subseteq X$ has a least upper bound $\bigsqcup C$

Order Theory

Monotonic and Scott-Continuous Functions

A function $f : X_1 \rightarrow X_2$ between posets $\langle X_1, \leq \rangle$ and $\langle X_2, \sqsubseteq \rangle$ is

- **monotonic** when $\forall x, y \in X_1 : x \leq y \Rightarrow f(x) \sqsubseteq f(y)$
- **Scott-continuous** when it preserves least upper bounds:
  for each chain $X \subseteq X_1$, if $\bigvee X$ exists, then $f(\bigvee X) = \bigsqcup \{f(x) \mid x \in X\}$

Order Theory

Kleenian Fixpoint Transfer

Theorem

Let $\langle C, \leq \rangle$ and $\langle A, \sqsubseteq \rangle$ be **complete partial orders**, let $f\colon C \to C$ and $f^{\#}\colon A \to A$ be **monotonic functions**, and let $\alpha\colon C \to A$ be a **Scott-continuous abstraction function** that satisfies the commutation condition $\alpha \circ f = f^{\#} \circ \alpha$. Then, given $c \in C$, we have $\alpha(\mathsf{lfp}_c^{\leq} f) = \mathsf{lpf}_{\alpha(c)}^{\sqsubseteq} f^{\#}$

Prefix Trace to Forward Reachability

Kleenian Fixpoint Transfer

Prefix Trace Semantics

$$\mathscr{T}_p(I) = \mathsf{lfp}^{\subseteq}_{\emptyset} F_p$$

$$F_p(T) \stackrel{\text{def}}{=} I \cup T \,;\, \tau$$

monotonic

Prefix State Abstraction

$$\langle \mathscr{P}(\Sigma^*), \subseteq \rangle \underset{\alpha_p}{\overset{\gamma_p}{\leftrightarrows}} \langle \mathscr{P}(\Sigma), \subseteq \rangle$$

complete partial order

complete partial order

$$\alpha_p(T) \stackrel{\text{def}}{=} \{s \in \Sigma \mid \exists s_0, \ldots, s \in T\}$$

Exercise: prove this 🙂 → Scott-continuous

$\alpha_p \circ F_p = F_r \circ \alpha_p$

Forward Reachability Semantics

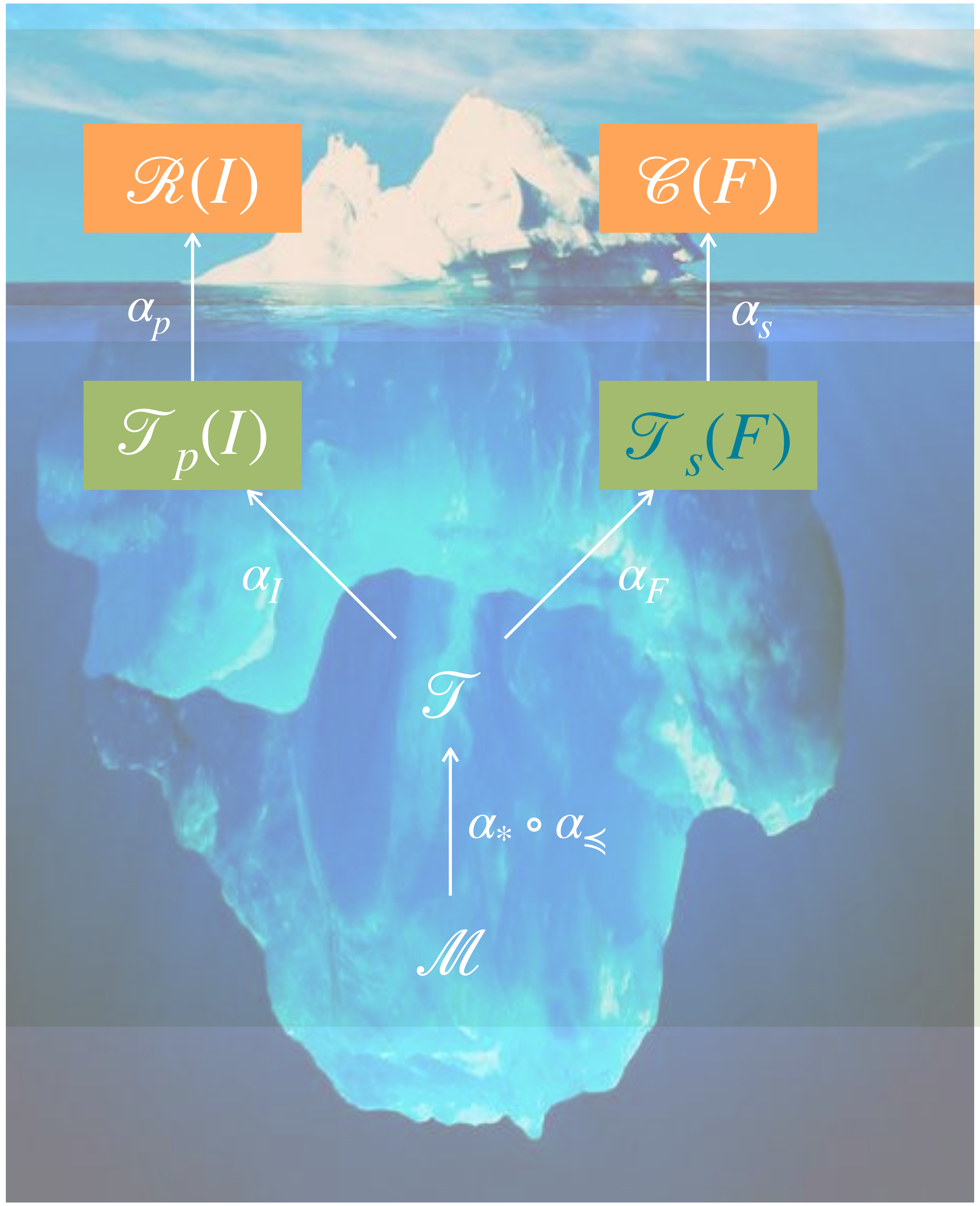
$$\mathscr{R}(I) = \mathsf{lfp}^{\subseteq}_{\emptyset} F_r$$

$$F_r(S) \stackrel{\text{def}}{=} I \cup \mathsf{post}(S)$$

monotonic

$$\alpha_p(\mathscr{T}_p(I)) = \alpha_p(\mathsf{lfp}^{\subseteq}_{\emptyset} F_p) = \mathsf{lfp}^{\subseteq}_{\emptyset} F_r = \mathscr{R}(I)$$

Suffix Trace Semantics



State Semantics

Trace Semantics

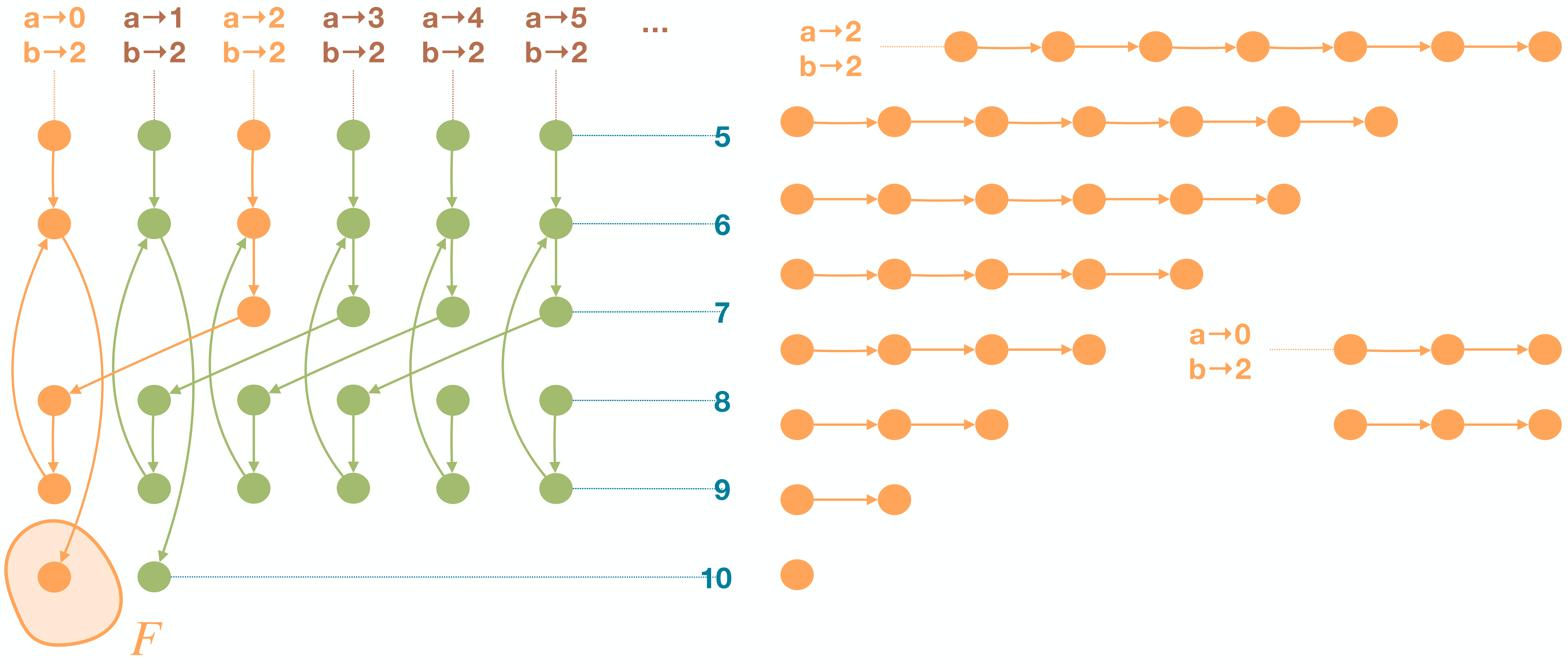
Suffix Trace Semantics

Finite Partial Program Traces Ending in $F \in \mathscr{P}(\Sigma)$

- $\mathscr{T}_s(F) \in \mathscr{P}(\Sigma^*)$

$$\mathscr{T}_s(F) \stackrel{\text{def}}{=} \{s_0, \ldots, s_n \mid n \geq 0 \land s_n \in F \land \forall i : \langle s_i, s_{i+1} \rangle \in \tau\}$$

3
q ← 0
4
r ← a
5
while 6(r ≥ b) **do**
7
r ← r − b
8
q ← q + 1
9
od
10

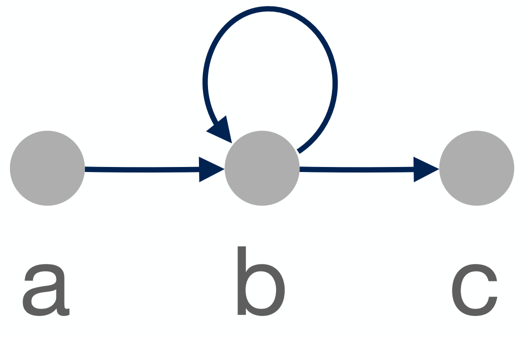


Suffix Trace Semantics

Least Fixpoint Formulation

$$\mathscr{T}_s(F) = \mathsf{lfp}^{\subseteq}_{\emptyset}\, F_s$$

$$F_s(T) \stackrel{\text{def}}{=} F \cup \tau \,;\, T$$

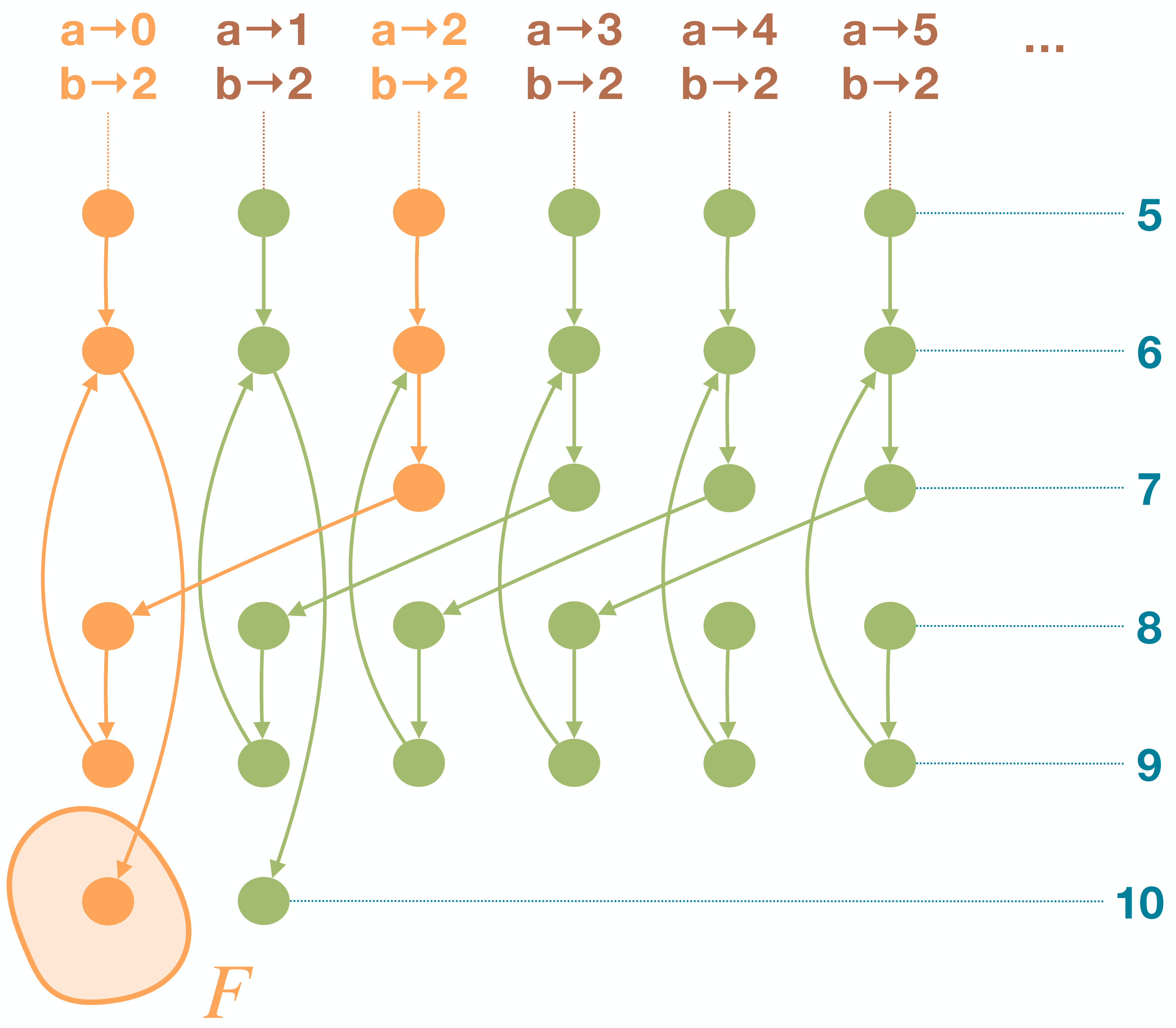


$$F \stackrel{\text{def}}{=} \{c\}$$

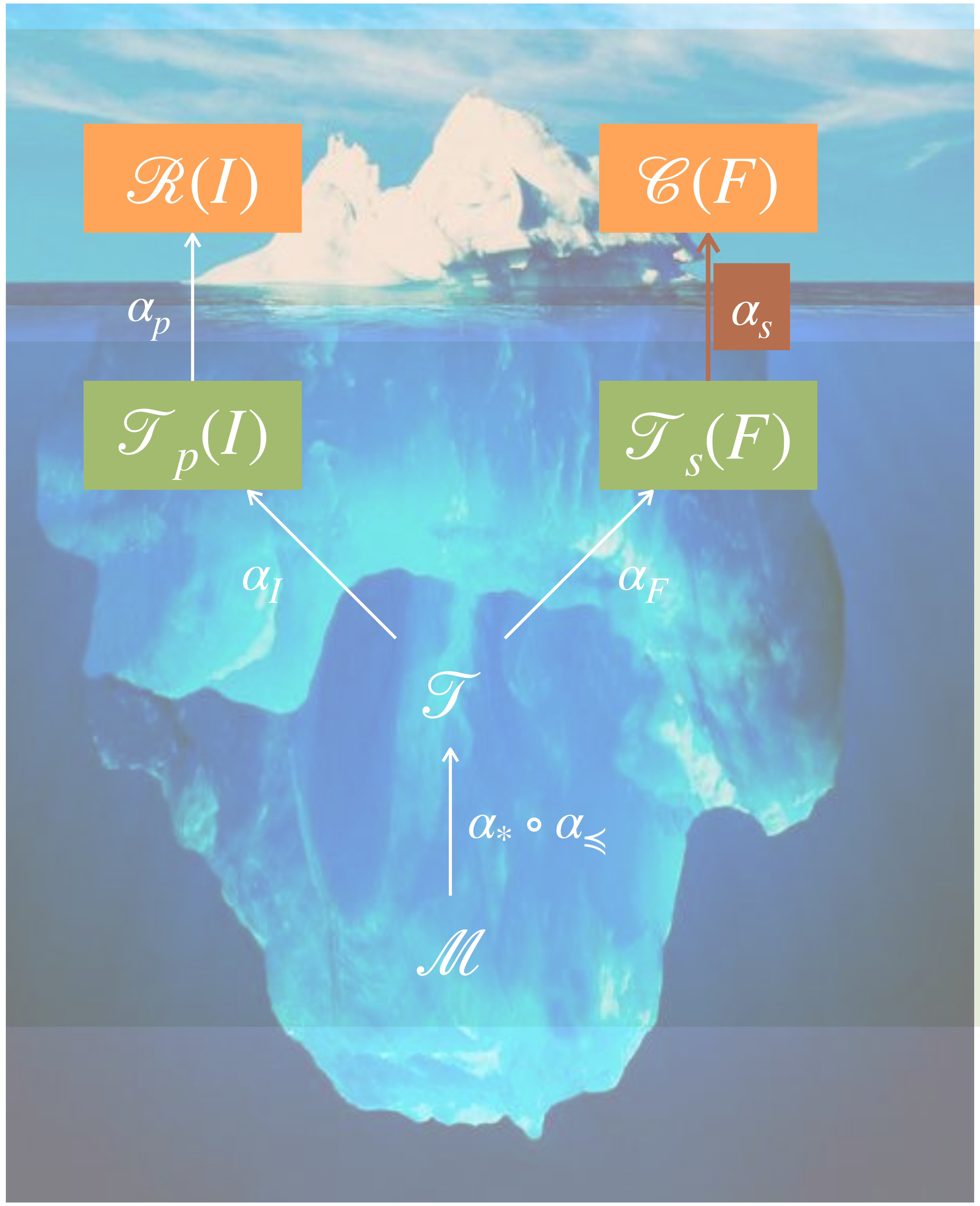
- $F_s^0(\emptyset) = \emptyset$
- $F_s^1(F_s^0) = F = \{c\}$
- $F_s^2(F_s^1) = \{c, bc\}$
- $F_s^3(F_s^3) = \{c, bc, bbc, abc\}$

$$\mathscr{T}_s(I) = \{c, b^i c, ab^i c \mid i \geq 1\}$$

```
3
q ← 0
4
r ← a
5
while 6(r ≥ b) do
    7
    r ← r − b
    8
    q ← q + 1
    9
od
10
```



Backward Reachable State Abstraction



State Semantics

Trace Semantics

Backward Reachable State Abstraction

a state in the backward reachability semantics corresponds to a partial program trace starting in this state

$$\langle \mathscr{P}(\Sigma^*), \subseteq \rangle \underset{\alpha_s}{\overset{\gamma_s}{\leftrightarrows}} \langle \mathscr{P}(\Sigma), \subseteq \rangle$$

$$\alpha_s(T) \stackrel{\text{def}}{=} \{s \in \Sigma \mid \exists s, \ldots, s_n \in T\}$$

$$\gamma_s(S) \stackrel{\text{def}}{=} \{s, \ldots, s_n \in \Sigma^* \mid s \in S\}$$

Suffix Trace to Backward Reachability

Kleenian Fixpoint Transfer

Suffix Trace Semantics

$\mathcal{T}_s(F) = \textbf{lfp}_{\emptyset}^{\subseteq} F_s$

$F_s(T) \stackrel{\text{def}}{=} F \cup \tau ; T$

monotonic

Suffix State Abstraction

$\langle \mathscr{P}(\Sigma^*), \subseteq \rangle \underset{\alpha_s}{\overset{\gamma_s}{\leftrightarrows}} \langle \mathscr{P}(\Sigma), \subseteq \rangle$

complete partial order

complete partial order

$\alpha_s(T) \stackrel{\text{def}}{=} \{s \in \Sigma \mid \exists s, \ldots, s_n \in T\}$

Scott-continuous

$\alpha_s \circ F_s = F_c \circ \alpha_s$

Exercise: prove this 🙂

Backward Reachability Semantics

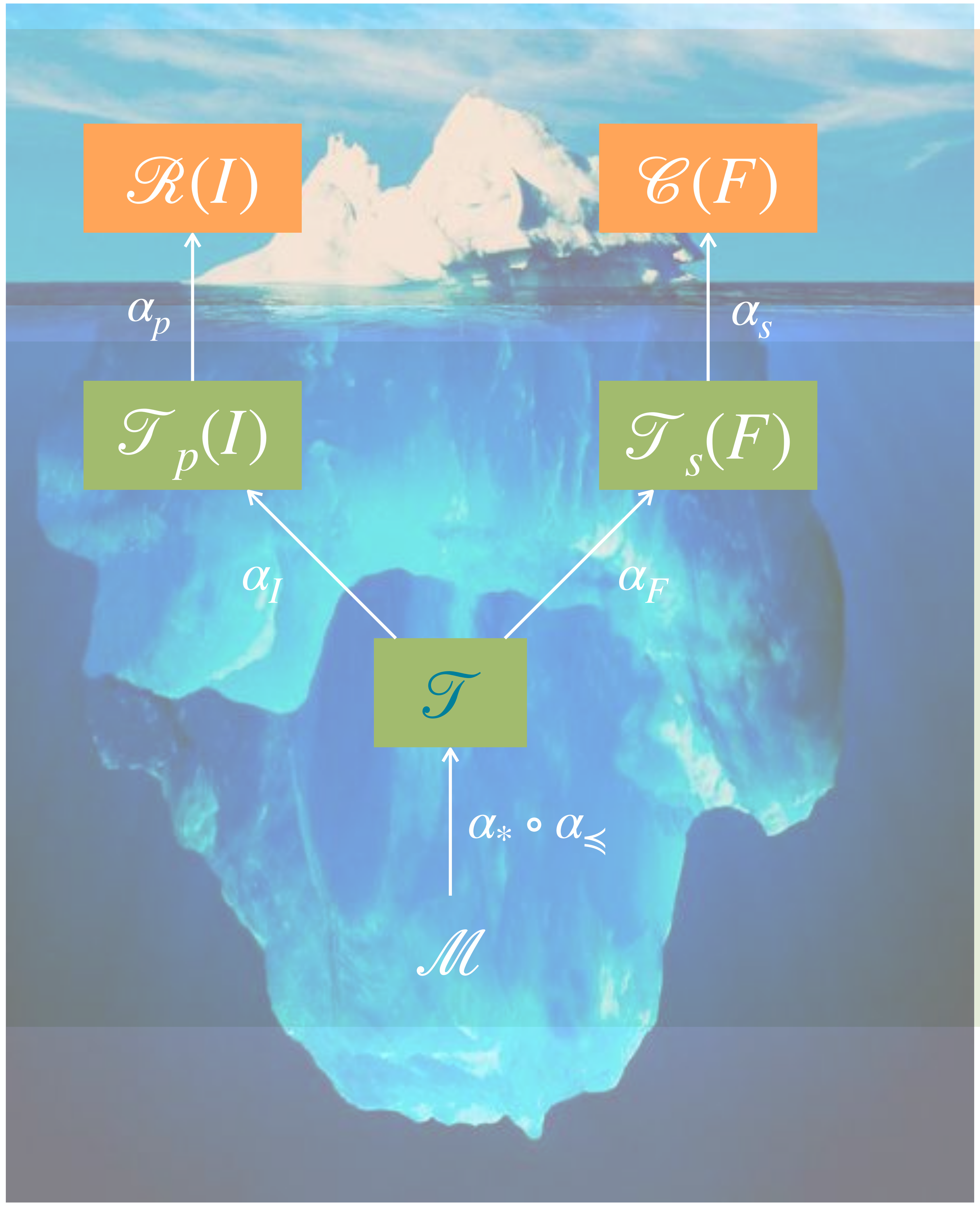
$\mathscr{C}(F) = \textbf{lfp}_{\emptyset}^{\subseteq} F_c$

$F_c(S) \stackrel{\text{def}}{=} F \cup \textbf{pre}(S)$

monotonic

$$\alpha_s(\mathcal{T}_s(F)) = \alpha_s(\textbf{lfp}_{\emptyset}^{\subseteq} F_s) = \textbf{lfp}_{\emptyset}^{\subseteq} F_c = \mathscr{C}(F)$$

Partial Finite Trace Semantics



State Semantics

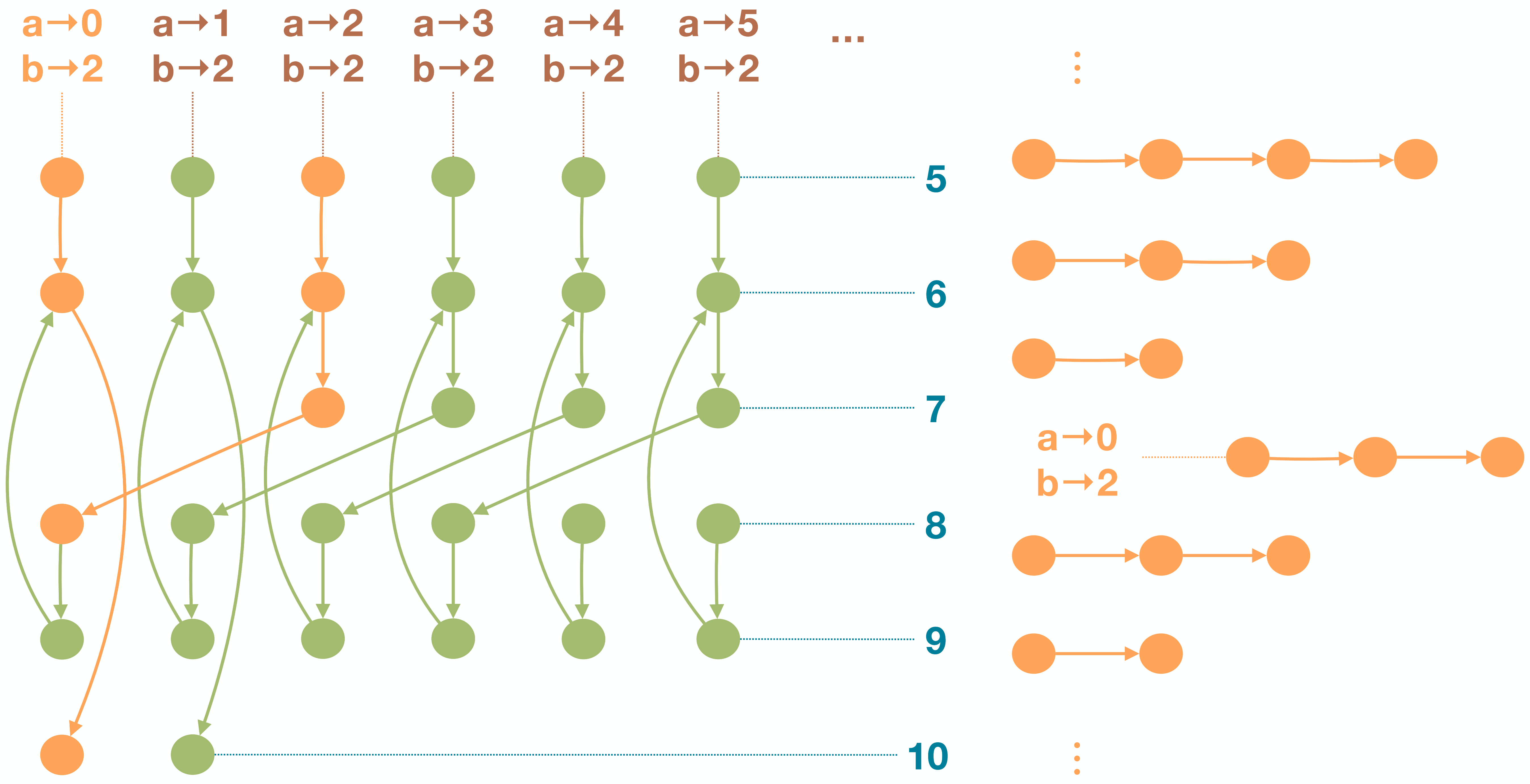
Trace Semantics

Partial Finite Trace Semantics

Partial Finite Program Traces

- $\mathcal{T} \in \mathcal{P}(\Sigma^*)$ $\qquad\qquad \mathcal{T} \stackrel{\text{def}}{=} \{s_0, \ldots, s_n \mid n \geq 0 \land \forall i\colon \langle s_i, s_{i+1} \rangle \in \tau\}$

```
3
q ← 0
4
r ← a
5
while 6(r ≥ b) do
    7
    r ← r − b
    8
    q ← q + 1
    9
od
10
```



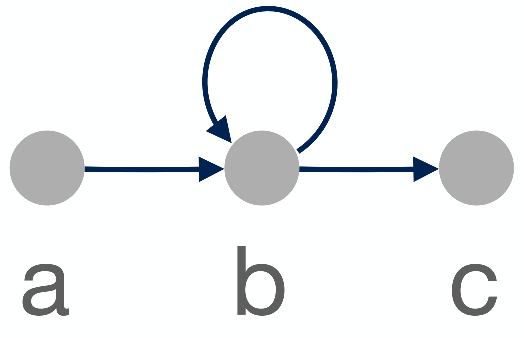
Partial Finite Trace Semantics

Least Fixpoint Formulation

Forward Formulation

$$\mathcal{T} = \mathsf{lfp}_{\emptyset}^{\subseteq} F_{p*}$$

$$F_{p*}(T) \stackrel{\text{def}}{=} \Sigma \cup T \,;\, \tau$$



- $F_{p*}^{0}(\emptyset) = \emptyset$
- $F_{p*}^{1}(F_{p*}^{0}) = \Sigma = \{a, b, c\}$
- $F_{p*}^{2}(F_{p*}^{1}) = \{a, ab, b, bb, bc, c\}$

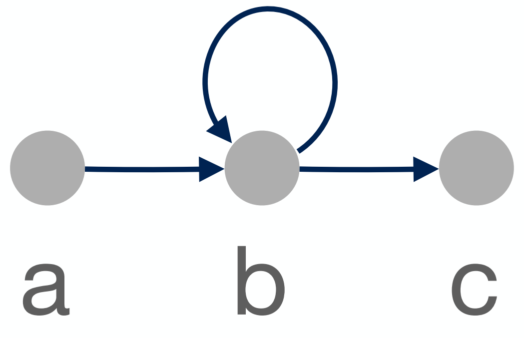
$\vdots$

$$\mathcal{T} = \{a, ab^i, ab^ic, b^i, b^ic, c \mid i \geq 1\}$$

Backward Formulation

$$\mathcal{T} = \mathsf{lfp}_{\emptyset}^{\subseteq} F_{s*}$$

$$F_{s*}(T) \stackrel{\text{def}}{=} \Sigma \cup \tau \,;\, T$$

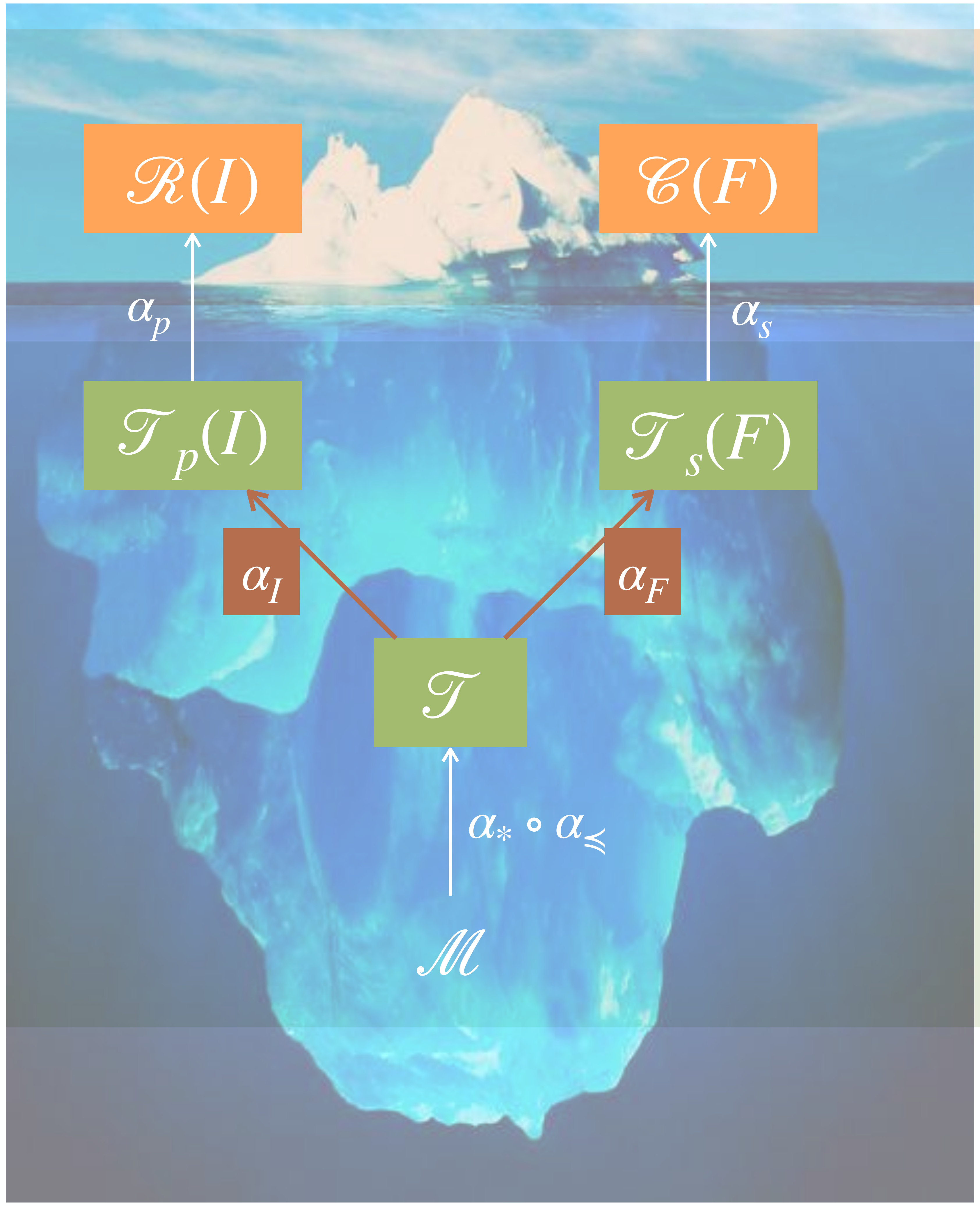


- $F_{s*}^{0}(\emptyset) = \emptyset$
- $F_{s*}^{1}(F_{s*}^{0}) = \Sigma = \{a, b, c\}$
- $F_{s*}^{2}(F_{s*}^{1}) = \{a, b, ab, bb, c, bc\}$

$\vdots$

$$\mathcal{T} = \{a, ab^i, ab^ic, b^i, b^ic, c \mid i \geq 1\}$$

Prefix/Suffix Trace Abstraction



State Semantics

Trace Semantics

Prefix Trace Abstraction

$$\begin{tikzcd} \langle \mathscr{P}(\Sigma^*), \subseteq \rangle \arrow[r, bend right, "\alpha_I"'] & \langle \mathscr{P}(\Sigma^*), \subseteq \rangle \arrow[l, bend right, "\gamma_I"'] \end{tikzcd}$$

$$\gamma_I(T) \overset{\text{def}}{=} T \cup ((\Sigma \setminus I) \cdot \Sigma^*)$$

$$\alpha_I(T) \overset{\text{def}}{=} T \cap (I \cdot \Sigma^*)$$

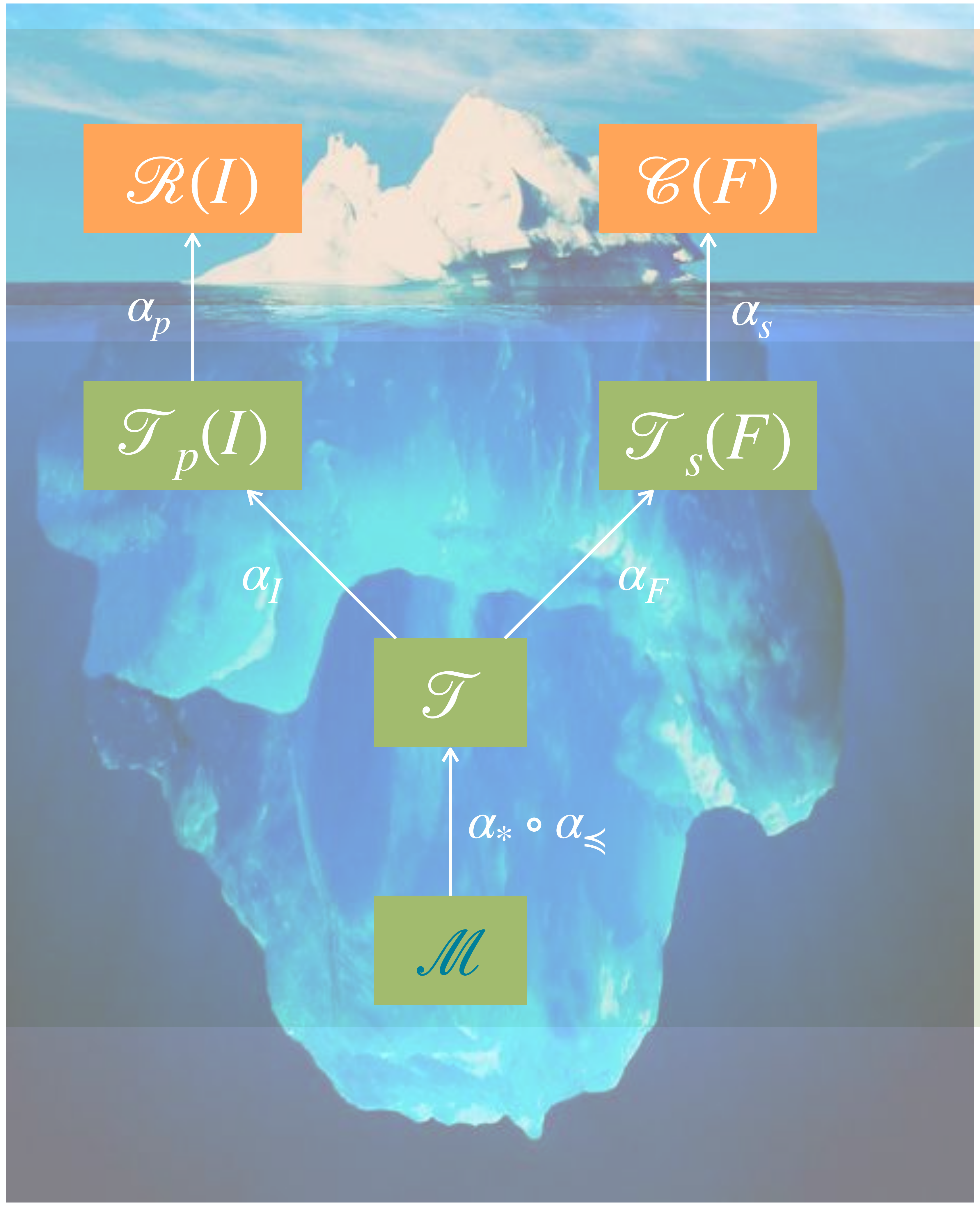
$$\alpha_F(T) \overset{\text{def}}{=} T \cap (\Sigma^* \cdot F)$$

$$\gamma_F(T) \overset{\text{def}}{=} T \cup (\Sigma^* \cdot (\Sigma \setminus F))$$

$$\begin{tikzcd} \langle \mathscr{P}(\Sigma^*), \subseteq \rangle \arrow[r, bend right, "\alpha_F"'] & \langle \mathscr{P}(\Sigma^*), \subseteq \rangle \arrow[l, bend right, "\gamma_F"'] \end{tikzcd}$$

Suffix Trace Abstraction

Maximal Trace Semantics



State Semantics

Trace Semantics

Maximal Trace Semantics

Finite and Infinite Program Traces

- $\mathscr{M} \in \mathscr{P}(\Sigma^\infty)$

$$\mathscr{M} \overset{\text{def}}{=} \{s_0, \ldots, s_n \in \Sigma^* \mid s_n \in \mathscr{B} \land \forall i \colon \langle s_i, s_{i+1} \rangle \in \tau\} \cup \{s_0, \ldots \in \Sigma^\omega \mid \forall i \colon \langle s_i, s_{i+1} \rangle \in \tau\}$$

$$\mathscr{B} \overset{\text{def}}{=} \{s \in \Sigma \mid \forall s' \in \Sigma \colon \langle s, s' \rangle \notin \tau\}$$

Order Theory

Lattices

A **lattice** $\langle X, \sqsubseteq, \sqcup, \sqcap \rangle$ is a partially ordered set with:

- a **least upper bound** $a \sqcup b$ for every pair of elements $a, b \in X$
- a **greatest lower bound** $a \sqcap b$ for every pair of elements $a, b \in X$

Order Theory

Complete Lattices

A **complete lattice** $\langle X, \sqsubseteq, \sqcup, \sqcap, \bot, \top \rangle$ is a partially ordered set with:

- a **least upper bound** $\bigsqcup S$ for every $S \subseteq X$ (and thus $\bot \stackrel{\text{def}}{=} \bigsqcup \emptyset$)
- a **greatest lower bound** $\bigsqcap S$ for every $S \subseteq X$ (thus $\top \stackrel{\text{def}}{=} \bigsqcap \emptyset = \bigsqcup D$)

Example $\langle \mathscr{P}(\Sigma^\infty), \sqsubseteq, \sqcup, \sqcap, \Sigma^\omega, \Sigma^* \rangle$

$$A \sqsubseteq B \stackrel{\text{def}}{\Leftrightarrow} (A \cap \Sigma^*) \subseteq (B \cap \Sigma^*) \wedge (A \cap \Sigma^\omega) \supseteq (B \cap \Sigma^\omega)$$

$$A \sqcup B \stackrel{\text{def}}{=} ((A \cap \Sigma^*) \cup (B \cap \Sigma^*)) \cup ((A \cap \Sigma^\omega) \cap (B \cap \Sigma^\omega))$$

$$A \sqcap B \stackrel{\text{def}}{=} ((A \cap \Sigma^*) \cap (B \cap \Sigma^*)) \cup ((A \cap \Sigma^\omega) \cup (B \cap \Sigma^\omega))$$

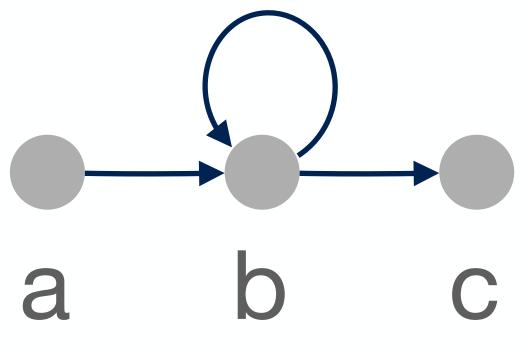
Maximal Trace Semantics

Least Fixpoint Formulation

computational order

$$\mathscr{M} = \mathsf{lfp}^{\sqsubseteq}_{\Sigma^\omega} F$$

$$F(T) \stackrel{\text{def}}{=} \mathscr{B} \cup \tau \,;\, T$$



- $F^0(\emptyset) = \Sigma^\omega$
- $F^1(F^0) = \{c\} \cup \{ab\Sigma^\omega, bb\Sigma^\omega, bc\Sigma^\omega\}$
- $F^2(F^1) = \{bc, c\} \cup \{abb\Sigma^\omega, bbb\Sigma^\omega, abc\Sigma^\omega, bbc\Sigma^\omega\}$
- $F_p^3(F_p^2) = \{abc, bbc, bc, c\} \cup \{abbb\Sigma^\omega, bbbb\Sigma^\omega, abbc\Sigma^\omega, bbbc\Sigma^\omega\}$

$$\mathscr{M} = \{ab^ic, b^ic, c \mid i \geq 1\} \cup \{ab^\omega, b^\omega\}$$

Maximal Trace Semantics

Denotational Formulation

$$\begin{aligned} stmt ::= \; & {}^{\ell}X \leftarrow expr^{\ell} \\ \mid \; & \textbf{if } {}^{\ell}expr \bowtie 0 \textbf{ then } stmt \textbf{ end}^{\ell} \\ \mid \; & \textbf{while } {}^{\ell}expr \bowtie 0 \textbf{ do } stmt \textbf{ done}^{\ell} \\ \mid \; & stmt; stmt \end{aligned}$$

$$\mathscr{M}[\![{}^{\ell_1}X \leftarrow e^{\ell_2}]\!]T \stackrel{\text{def}}{=} \{(\ell_1, \rho)(\ell_2, \rho[X \mapsto v])\sigma \mid \sigma \in \Sigma^{\infty}, (\ell_2, \rho[X \mapsto v])\sigma \in T, v \in E[\![e]\!]\rho\}$$

$$\mathscr{M}[\![\textbf{if } {}^{\ell_1}e \bowtie 0 \textbf{ then } {}^{\ell_2}s^{\ell_3} \textbf{ end}^{\ell_4}]\!]T \stackrel{\text{def}}{=}$$

$$\{(\ell_1, \rho)(\ell_2, \rho)\sigma \mid \sigma \in \Sigma^{\infty}, (\ell_2, \rho)\sigma \in \mathscr{C}[\![{}^{\ell_2}s^{\ell_3}]\!]\{(\ell_3, \rho')\sigma' \mid \sigma' \in T\}, \exists v \in E[\![e]\!]\rho : v \bowtie 0\} \cup$$
$$\{(\ell_1, \rho)(\ell_4, \rho)\sigma \mid \sigma \in \Sigma^{\infty}, (\ell_4, \rho)\sigma \in T, \exists v \in E[\![e]\!]\rho : v \not\bowtie 0\}$$

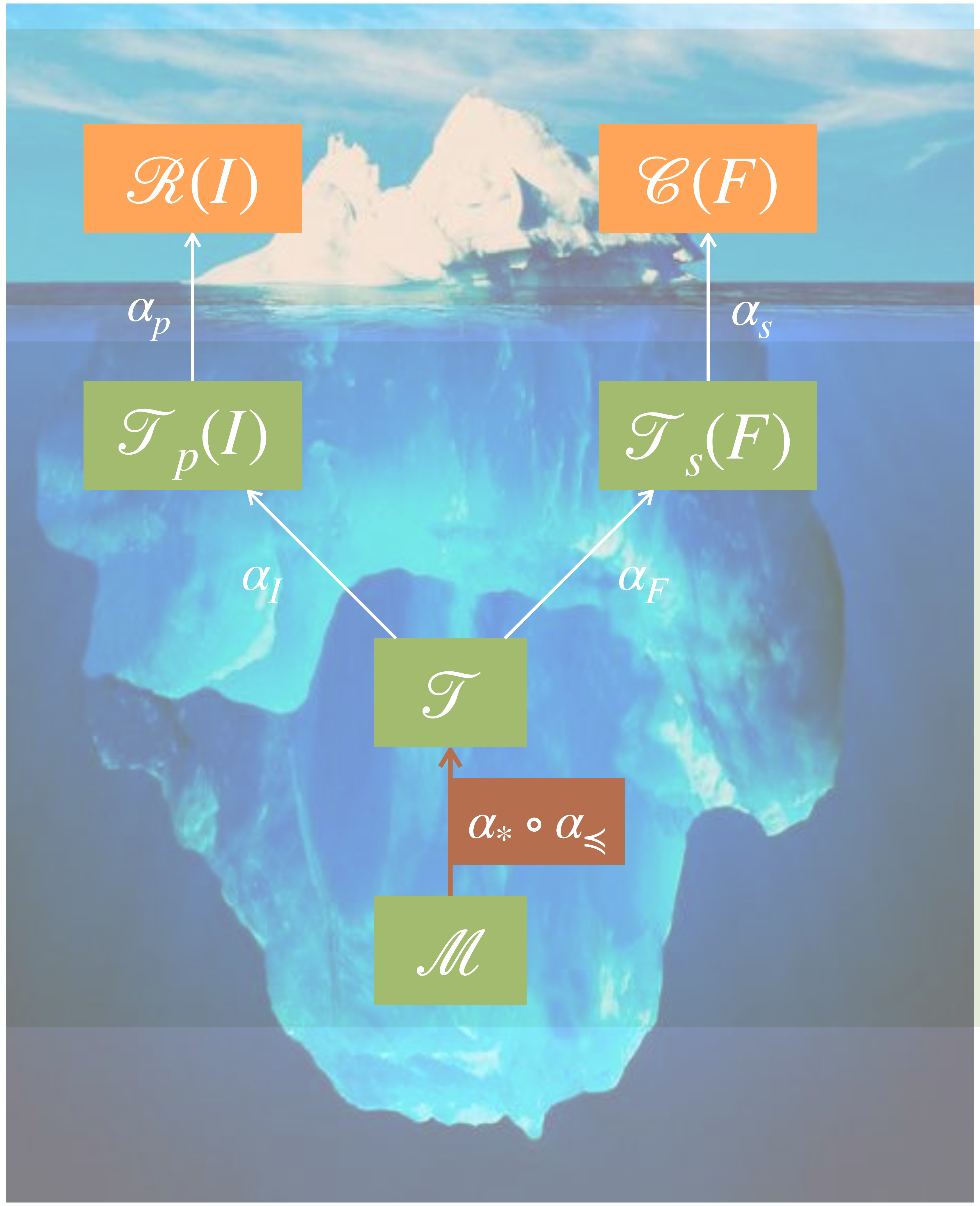
$$\mathscr{M}[\![\textbf{while } {}^{\ell_1}e \bowtie 0 \textbf{ then } {}^{\ell_2}s^{\ell_3} \textbf{ done}^{\ell_4}]\!]T \stackrel{\text{def}}{=} \mathsf{lfp}^{\sqsubseteq}_{\Sigma^{\omega}} F$$

where

$$F(Y) \stackrel{\text{def}}{=} \{(\ell_1, \rho)(\ell_4, \rho)\sigma \mid \sigma \in \Sigma^{\infty}, (\ell_4, \rho)\sigma \in T, \exists v \in E[\![e]\!]\rho : v \not\bowtie 0\} \cup$$
$$\{(\ell_1, \rho)(\ell_2, \rho)\sigma \mid \sigma \in \Sigma^{\infty}, (\ell_2, \rho)\sigma \in \mathscr{M}[\![{}^{\ell_2}s^{\ell_3}]\!]\{(\ell_3, \rho')\sigma' \mid \sigma' \in Y\}, \exists v \in E[\![e]\!]\rho : v \bowtie 0\}$$

$$\mathscr{M}[\![s_1; s_2]\!]T \stackrel{\text{def}}{=} \mathscr{M}[\![s_1]\!](\mathscr{M}[\![s_2]\!]S)$$

Partial Finite Trace Abstraction

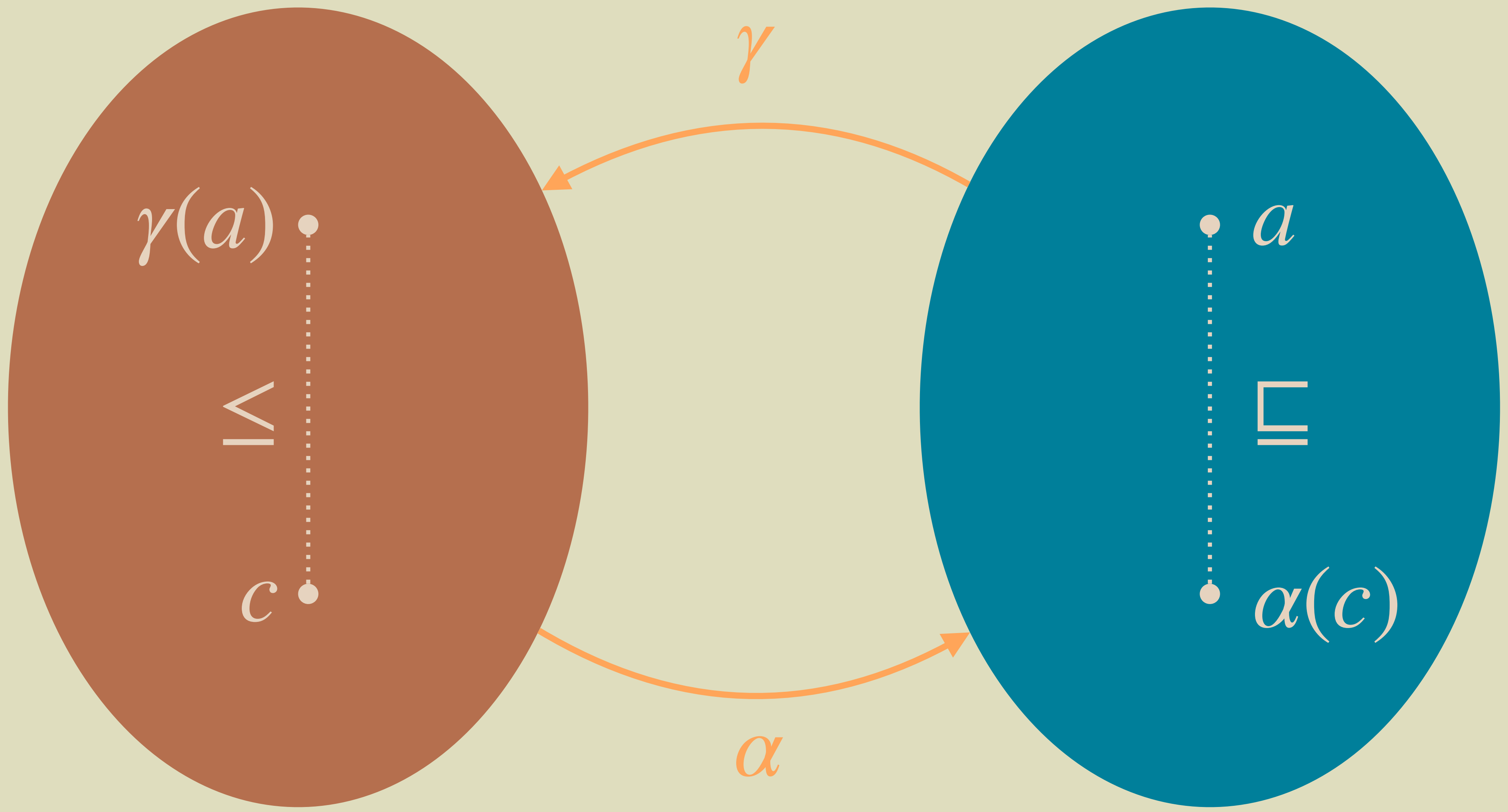


State Semantics

Trace Semantics

Order Theory

Galois Connections



Given a **Galois connection** between two posets $\langle C, \leq \rangle$ and $\langle A, \sqsubseteq \rangle$, each adjoint can be **uniquely defined** in term of the other:

$$\forall c \in C, \alpha(c) = \sqcap \{a \in A \mid c \leq \gamma(a)\}$$

$$\forall a \in A, \gamma(a) = \vee \{c \in C \mid \alpha(c) \sqsubseteq a\}$$

Partial Finite Trace Abstraction

approximation order

$$\begin{tikzcd}[column sep=huge] \langle \mathscr{P}(\Sigma^\infty), \subseteq \rangle \arrow[r, bend right, "\alpha_{\preccurlyeq}"'] & \langle \mathscr{P}(\Sigma^\infty), \subseteq \rangle \arrow[l, bend right, "\gamma_{\preccurlyeq}"'] \arrow[r, bend right, "\alpha_*"'] & \langle \mathscr{P}(\Sigma^*), \subseteq \rangle \arrow[l, bend right, "\gamma_*"'] \end{tikzcd}$$

Partial Trace Abstraction

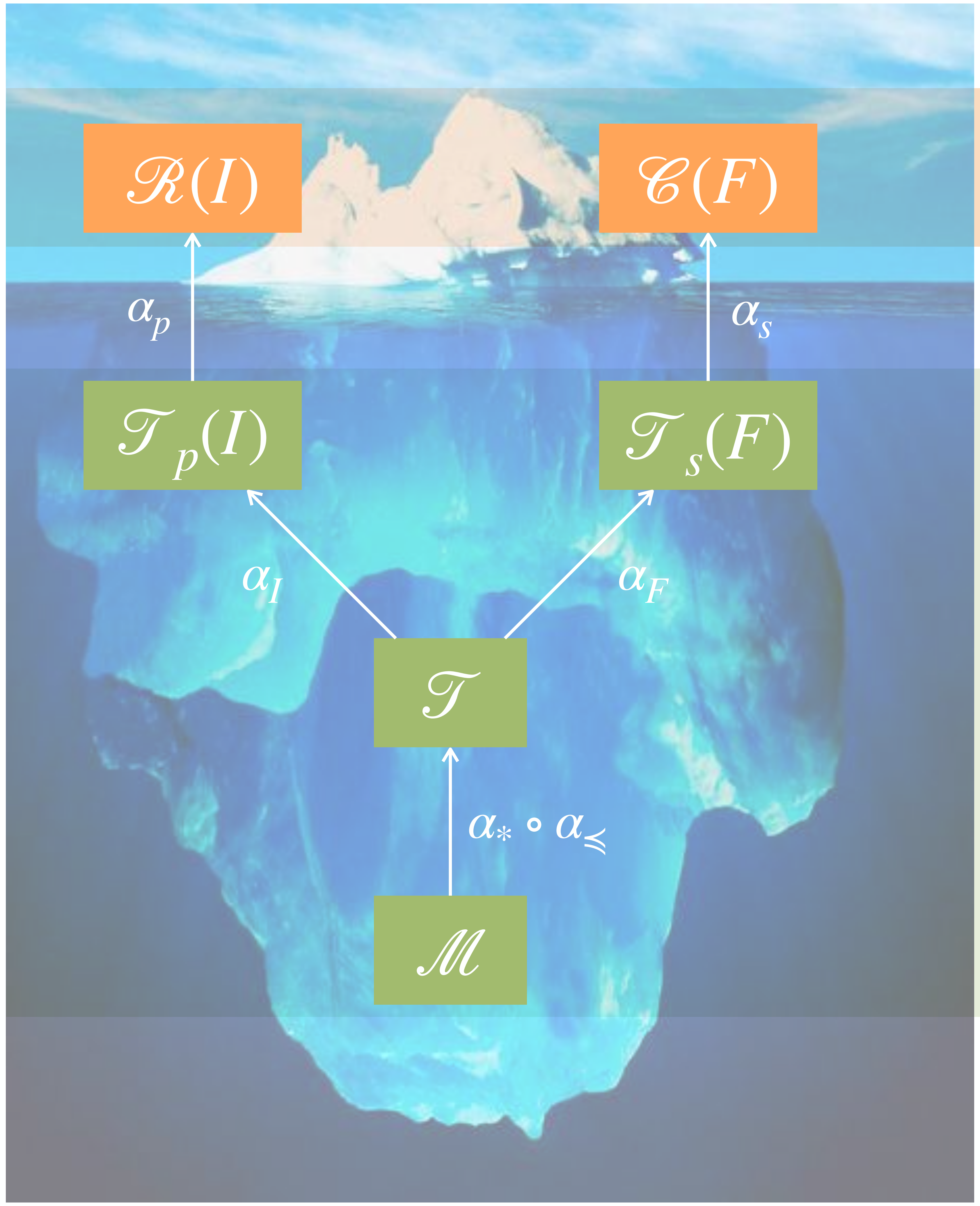
$\alpha_{\preccurlyeq}(T) \stackrel{\text{def}}{=} \{\sigma \in \Sigma^\infty \mid \exists \sigma' \in \Sigma^\infty : \sigma \circ \sigma' \in T\}$ ⬅ Exercise: derive $\gamma_{\preccurlyeq}$

Finite Trace Abstraction

$\alpha_*(T) \stackrel{\text{def}}{=} T \cap \Sigma^*$

$\gamma_*(S) \stackrel{\text{def}}{=} T \cup \Sigma^\omega$

Hierarchy of Semantics



Forward/Backward Reachability Semantics

Forward/Backward Reachable State Abstraction

Prefix/Suffix Trace Semantics

Prefix/Suffix Trace Abstraction

Partial Finite Trace Semantics

Partial Finite Trace Abstraction

Maximal Trace Semantics

Reading Suggestion

ELSEVIER

Theoretical Computer Science 277 (2002) 47–103

Theoretical Computer Science

www.elsevier.com/locate/tcs

Constructive design of a hierarchy of semantics of a transition system by abstract interpretation

Patrick Cousot[1]

Département d'Informatique, École Normale Supérieure, 45 rue d'Ulm, 75230 Paris cedex 05, France

Abstract

We construct a hierarchy of semantics by successive abstract interpretations. Starting from the maximal trace semantics of a transition system, we derive the big-step semantics, termination and nontermination semantics, Plotkin's natural, Smyth's demoniac and Hoare's angelic relational semantics and equivalent nondeterministic denotational semantics (with alternative powerdomains to the Egli–Milner and Smyth constructions), D. Scott's deterministic denotational semantics, the generalized and Dijkstra's conservative/liberal predicate transformer semantics, the generalized/total and Hoare's partial correctness axiomatic semantics and the corresponding proof methods. All the semantics are presented in a uniform fixpoint form and the correspondences between these semantics are established through composable Galois connections, each semantics being formally calculated by abstract interpretation of a more concrete one using Kleene and/or Tarski fixpoint approximation transfer theorems.

Static Analysis by Abstract Interpretation

Program Properties

practical tools
targeting specific programs

abstract semantics, abstract domains
algorithmic approaches to decide program properties

concrete semantics
mathematical models of the program behavior

Static Analysis by Abstract Interpretation

PROPERTY OF INTEREST

State Properties

$S \in \mathscr{P}(\Sigma)$

```
1
a ← [0, +∞]
2
b ← [0, +∞]
3
q ← 0
4
r ← a
5
while 6(r ≥ b) do
    7
    r ← r − b
    8
    q ← q + 1
    9
done
10
```

Example

- $S \stackrel{\text{def}}{=} \{\langle \ell, \rho \rangle \in \Sigma \mid \ell \in \mathscr{L}, \rho \in \mathscr{E}, \rho(r) \geq 0\}$

State Property Verification

$\mathscr{R}(I) \subseteq S$

$\mathscr{C}(F) \subseteq S$

Trace Properties

$T \in \mathscr{P}(\Sigma^\infty)$

Example

- Termination: $T \stackrel{\text{def}}{=} \Sigma^*$
- Non-Termination: $T \stackrel{\text{def}}{=} \Sigma^\omega$
- Any State Property $S \in \mathscr{P}(\Sigma)$: $T \stackrel{\text{def}}{=} S^\infty$

Trace Property Verification

$\mathscr{M} \subseteq T$

Trace Properties

$T \in \mathscr{P}(\Sigma^\infty)$

Safety Properties = "Nothing Bad Ever Happens"

Example

- Any State Property $S \in \mathscr{P}(\Sigma)$: $T \stackrel{\text{def}}{=} S^\infty$

Safety Property Verification

- T can be **verified** by **exhaustive testing**

$\mathscr{T}_p(I) \subseteq T$

- T can be **falsified** by finding **a single finite execution not in** T

Trace Properties

$T \in \mathscr{P}(\Sigma^\infty)$

Liveness Properties = “Something Good Eventually Happens”

Example

- Termination: $T \stackrel{\text{def}}{=} \Sigma^*$

Liveness Property Verification

- T cannot be **verified** by **testing**

$\mathscr{M} \subseteq T$

- falsifying T requires finding **an infinite execution not in** T

Program Properties

$P \in \mathscr{P}(\mathscr{P}(\Sigma^\infty))$

Example

- Determinism: $P \stackrel{\text{def}}{=} \{\{\sigma\} \mid \sigma \in \Sigma^\infty\}$

Program Property Verification

$$\mathscr{M} \in P \Leftrightarrow \underbrace{\{\mathscr{M}\}}_{\text{Collecting Semantics}} \subseteq P$$

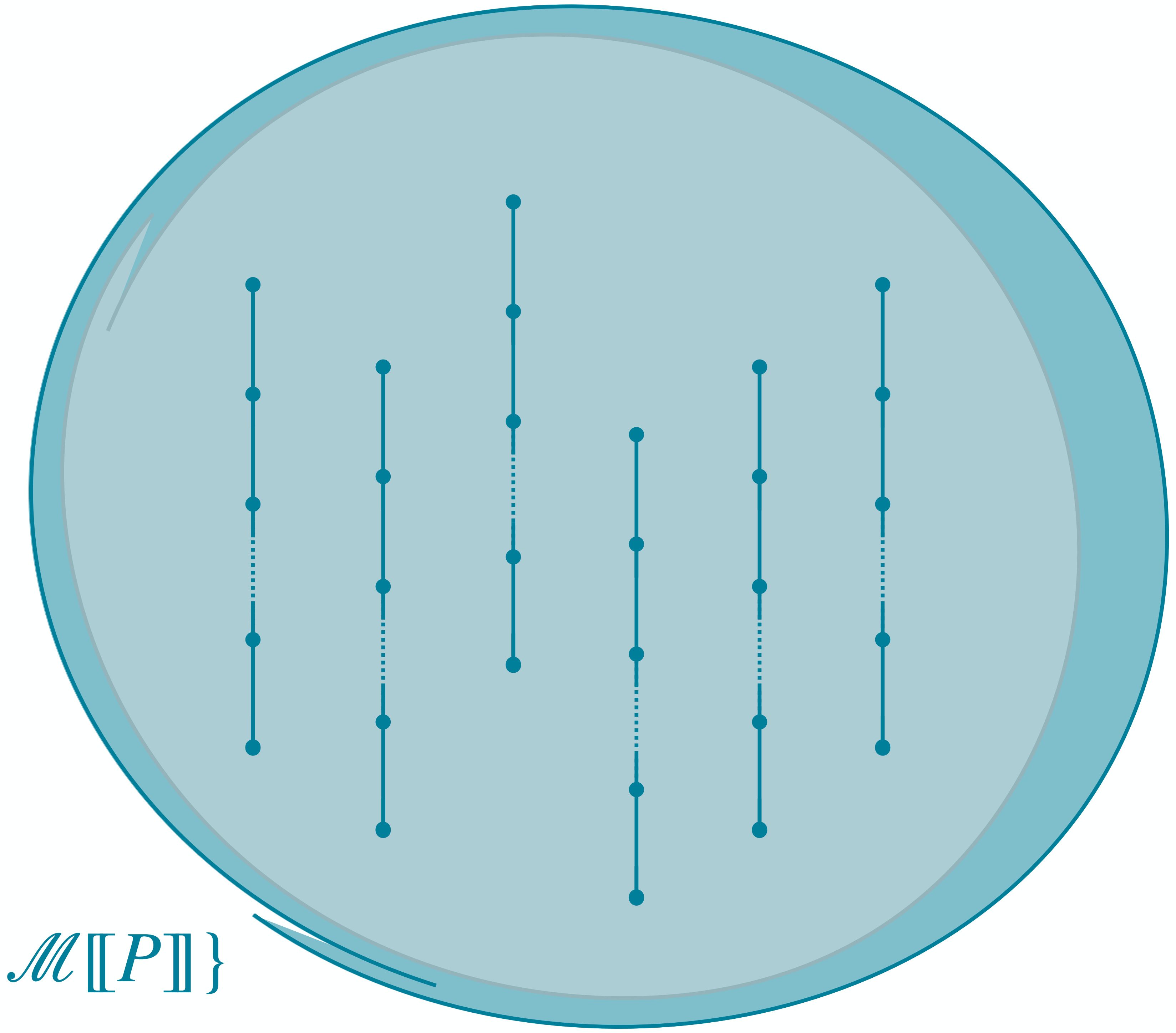
Collecting Semantics

Intuition

Property (by extension): set of elements that have that property

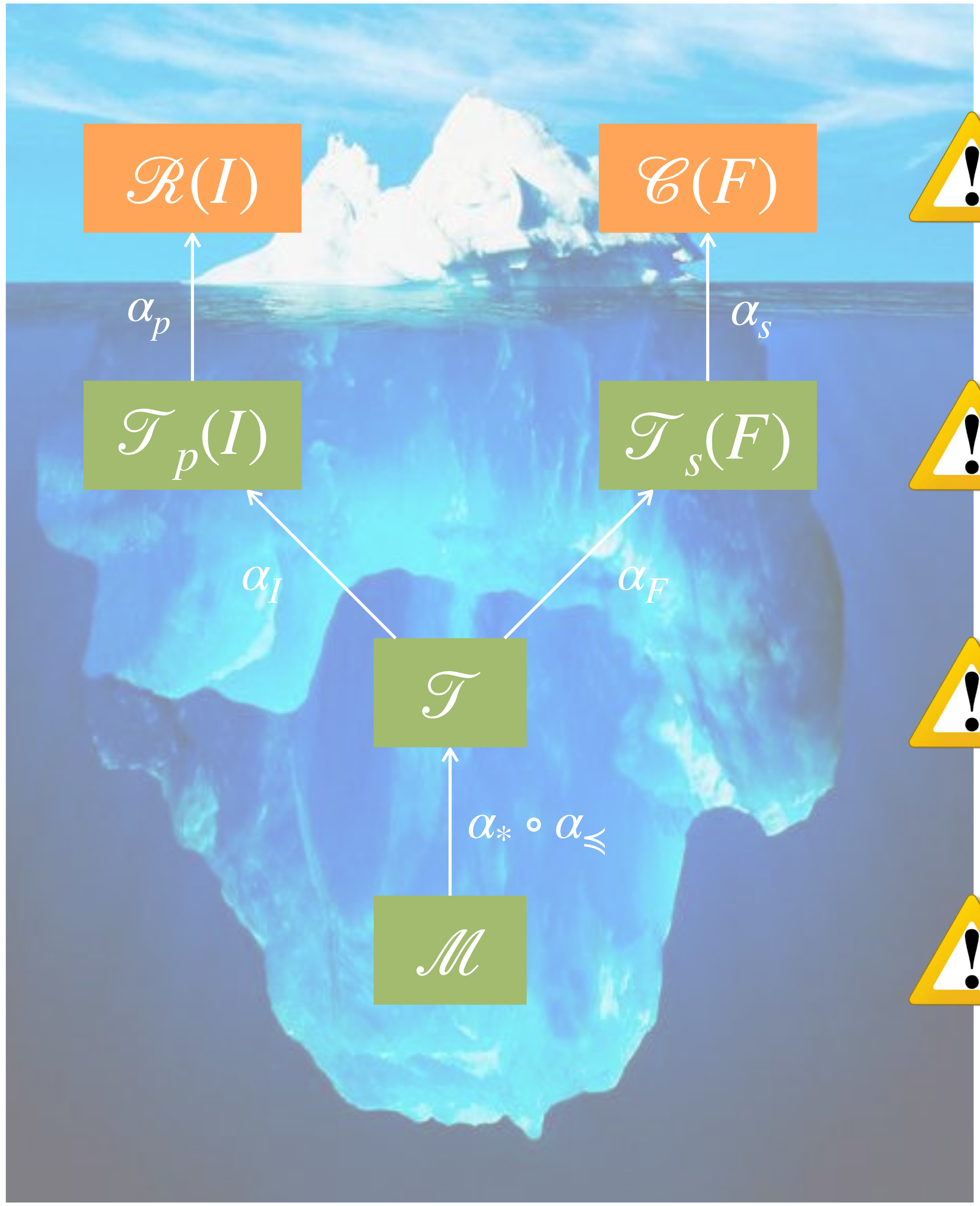
Property “being Zena”

Property “being program P”



$\{\mathscr{M}[\![P]\!]\}$

Hierarchy of Mathematical Objects



NOT COMPUTABLE!

NOT COMPUTABLE!

NOT COMPUTABLE!

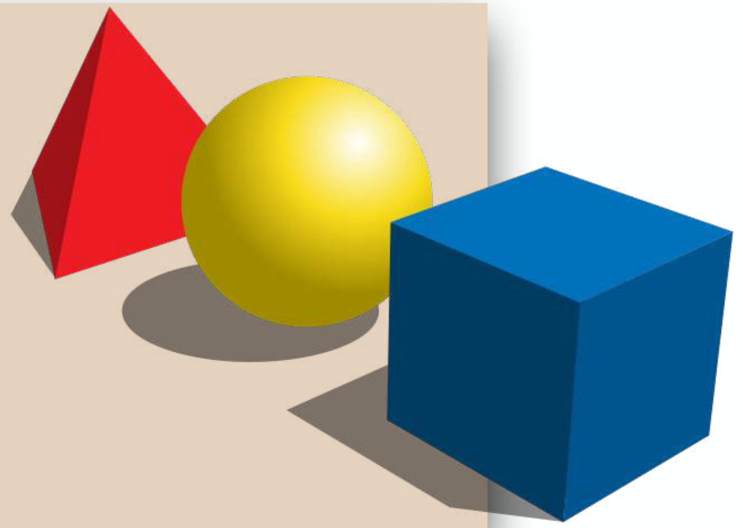
NOT COMPUTABLE!

Static Analysis by Abstract Interpretation

Abstract Program Semantics

practical tools
targeting specific programs

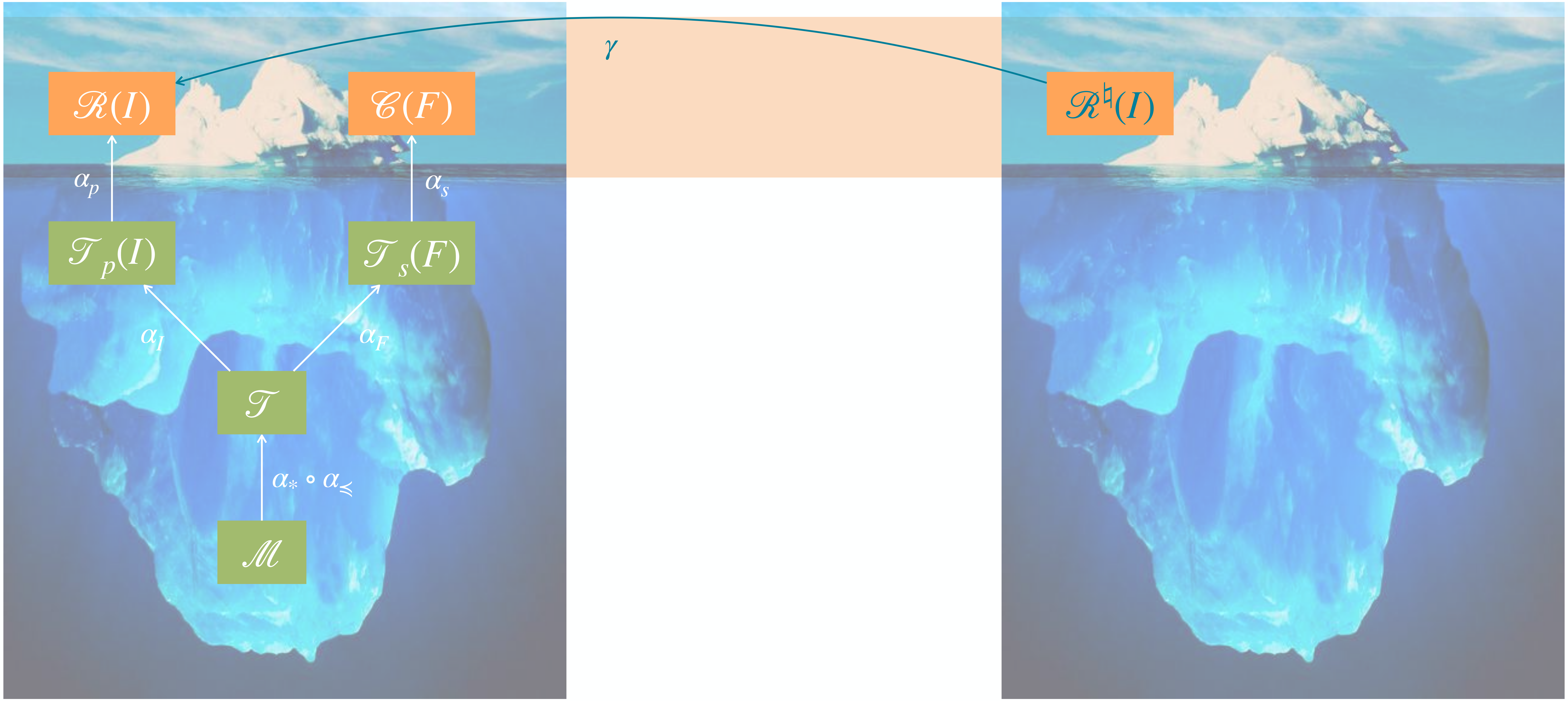
abstract semantics, abstract domains
algorithmic approaches to decide program properties



concrete semantics
mathematical models of the program behavior

Static Analysis by Abstract Interpretation

Abstract Forward Reachability Semantics



Abstract Forward Reachability Semantics

Over-Approximation of Program States Reachable From $I \in \mathscr{P}(\Sigma)$

- $\mathscr{R}(I) \subseteq \gamma(\mathscr{R}^{\#}(I)) \in \mathscr{P}(\Sigma)$

Abstract Forward Reachability Semantics

Denotational Formulation

$$\mathcal{R}^\#[\![{}^{\ell_1}X \leftarrow e^{\ell_2}]\!](\ell_1, a) \stackrel{\text{def}}{=} (\ell_2, \text{ASSIGN}_A[\![X \leftarrow e]\!]a)$$

$$\mathcal{R}^\#[\![\textbf{if } {}^{\ell_1} e \bowtie 0 \textbf{ then } {}^{\ell_2}s^{\ell_3} \textbf{ end}^{\ell_4}]\!](\ell_1, a) \stackrel{\text{def}}{=} (\ell_4, a') \sqcup_A (\ell_4, \text{FILTER}_A[\![e \not\bowtie 0]\!]a)$$

where $(\ell_3, a') = \mathcal{R}^\#[\![{}^{\ell_2}s^{\ell_3}]\!](\ell_2, \text{FILTER}_A[\![e \bowtie 0]\!]a))$

$$\mathcal{R}[\![\textbf{while } {}^{\ell_1} e \bowtie 0 \textbf{ then } {}^{\ell_2}s^{\ell_3} \textbf{ done}^{\ell_4}]\!](\ell_1, a) \stackrel{\text{def}}{=} (\ell_4, \text{FILTER}_A[\![e \not\bowtie 0]\!]a')$$

where $(\ell_1, a') = \text{lfp}^{\#\sqsubseteq_A}_{(\ell_1, \bot_A)} F_r^\#$

$$F_r^\natural((\ell_1, y)) \stackrel{\text{def}}{=} (\ell_1, a) \sqcup_A (\ell_1, a'')$$

$$(\ell_3, a'') = \mathcal{R}^\#[\![{}^{\ell_2}s^{\ell_3}]\!](\ell_2, \text{FILTER}_A[\![e \bowtie 0]\!]y))$$

$$\mathcal{R}^\#[\![s_1; s_2]\!](\ell_1, a) \stackrel{\text{def}}{=} \mathcal{R}\#[\![s_2]\!](\mathcal{R}\#[\![s_1]\!](\ell_1, a))$$

$$\begin{aligned} stmt ::= \; & {}^{\ell}X \leftarrow expr^{\ell} \\ | \; & \textbf{if } {}^{\ell}expr \bowtie 0 \textbf{ then } stmt \textbf{ end}^{\ell} \\ | \; & \textbf{while } {}^{\ell}expr \bowtie 0 \textbf{ do } stmt \textbf{ done}^{\ell} \\ | \; & stmt; stmt \end{aligned}$$

Static Analysis by Abstract Interpretation

SOUNDNESS

$ 10 +
$ 40 +
$ 30 +
$ 10

$ 90

~~COMPLETENESS~~

$ 9.95 +
$ 35.85 +
$ 24.95 +
$ 4.85

$ 75.60

FALSE ALARM

State Properties

$S \in \mathscr{P}(\Sigma)$

```
1
a ← [0, +∞]
2
b ← [0, +∞]
3
q ← 0
4
r ← a
5
while 6(r ≥ b) do
    7
    r ← r − b
    8
    q ← q + 1
    9
done
10
```

Example

- $S \stackrel{\text{def}}{=} \{\langle \ell, \rho \rangle \in \Sigma \mid \ell \in \mathscr{L}, \rho \in \mathscr{E}, \rho(r) \geq 0\}$

Sound State Property Verification

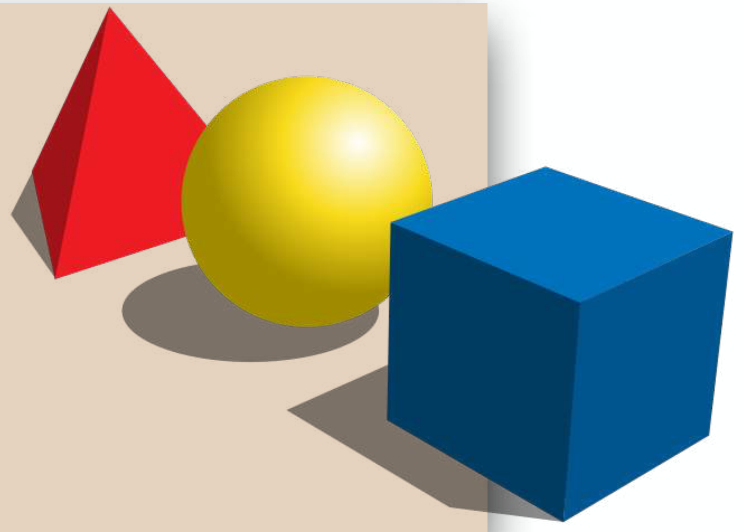
$\mathscr{R}(I) \subseteq \gamma(\mathscr{R}^\natural(I)) \subseteq S$

Static Analysis by Abstract Interpretation

Abstract Domains

practical tools
targeting specific programs

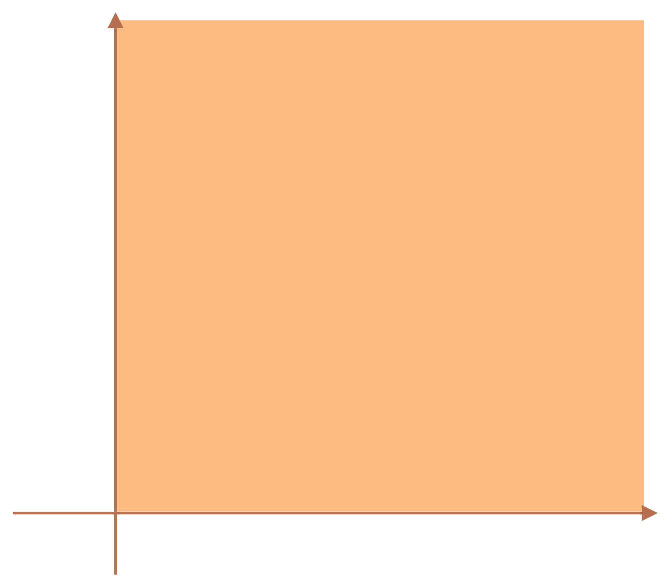
abstract semantics, abstract domains
algorithmic approaches to decide program properties



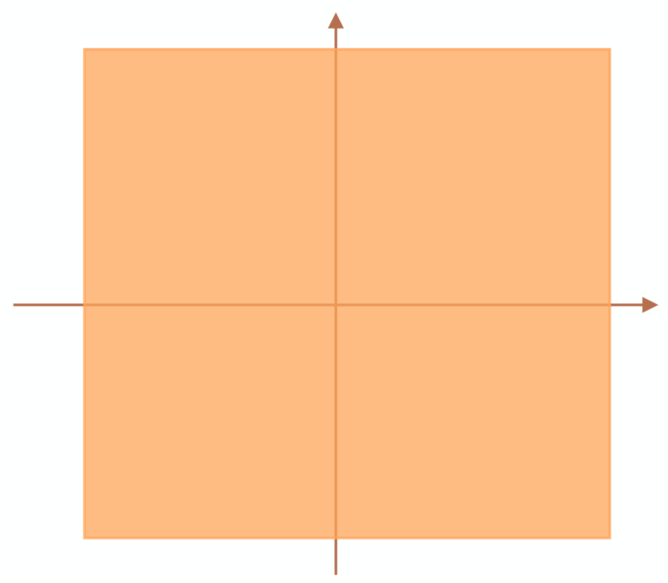
concrete semantics
mathematical models of the program behavior

Numerical Abstract Domains

Non-Relational Domains

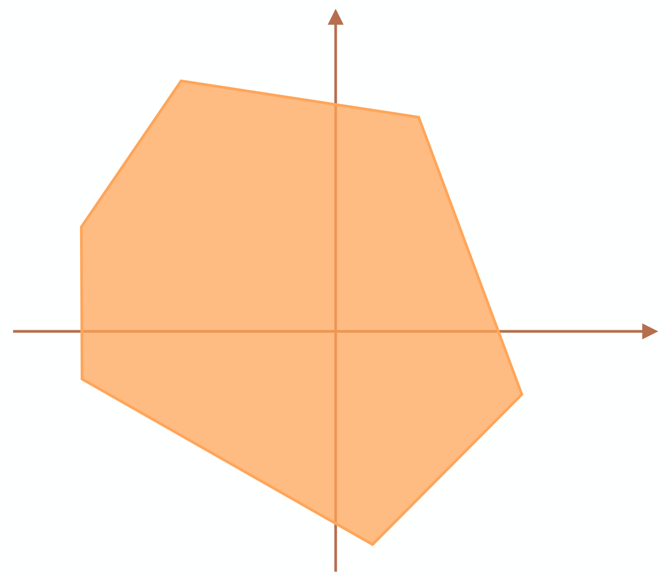


Sign Domain



Interval Domain

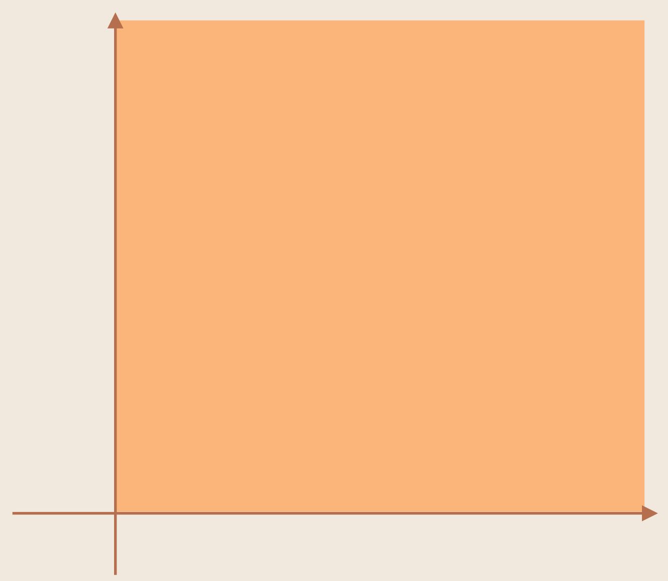
Relational Domains



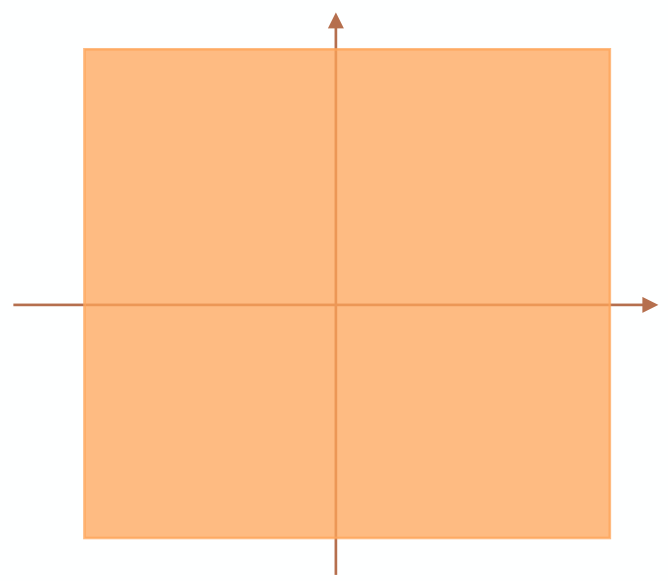
Polyhedra Domain

Numerical Abstract Domains

Non-Relational Domains

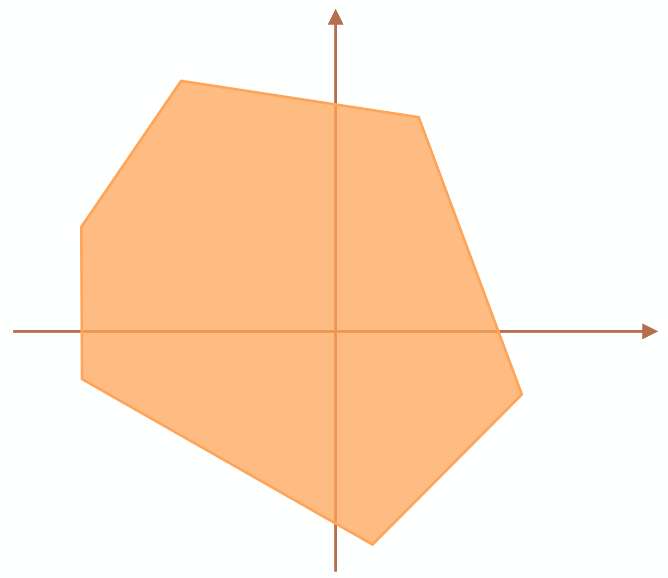


Sign Domain



Interval Domain

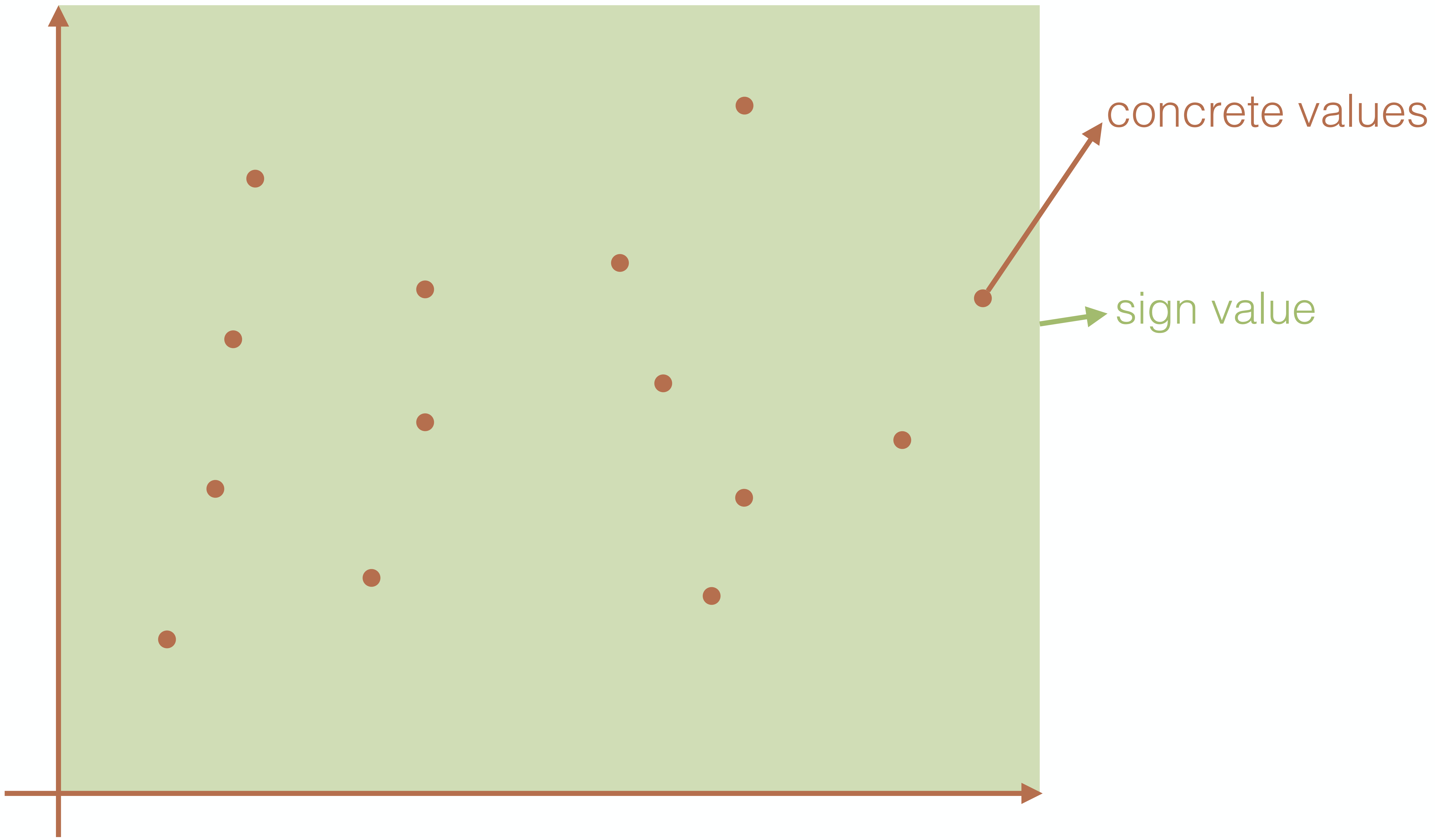
Relational Domains



Polyhedra Domain

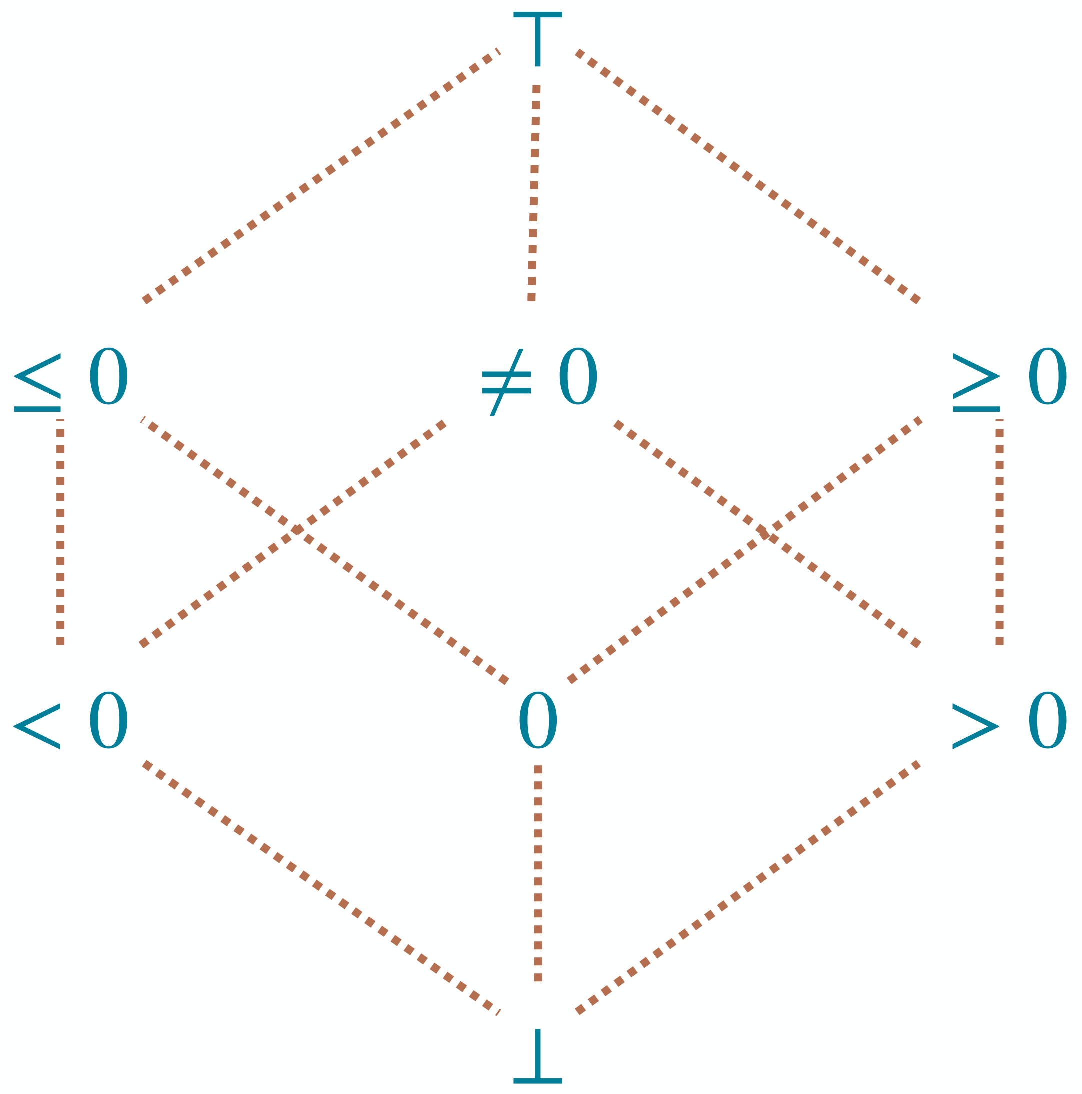
Sign Abstract Domain

Concrete Value are Replaced with Sign Values



Sign Abstract Domain

$A: \mathbb{X} \to \text{Sign}$ maps **variables** to their **sign**



$\sqsubseteq_A$ defined by the diagram

$\sqcup_A$ defined by the diagram

$\sqcap_A$ defined by the diagram

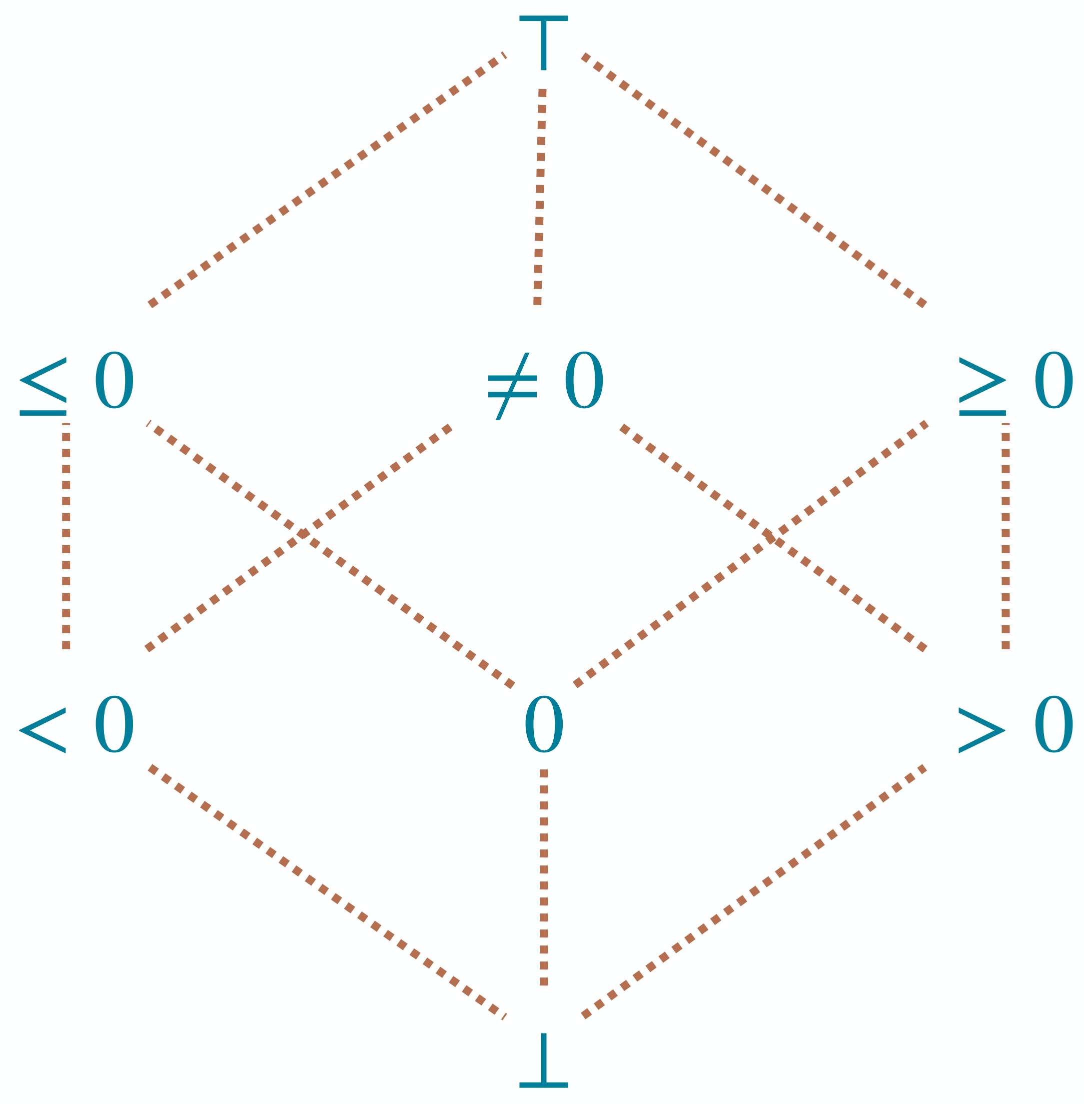
$\text{ASSIGN}_A[\![X \leftarrow e]\!]a$ maps X to e evaluated according to the **sign rules**

$$
\begin{gathered}
>0 + >0 = >0 - <0 = >0 \\
<0 + <0 = <0 - >0 = <0 \\
\vdots
\end{gathered}
$$

$\text{FILTER}_A[\![e \bowtie 0]\!]a$ modifies a to satisfy $e \bowtie 0$

Sign Abstract Domain

$A : \mathbb{X} \to \text{Sign}$ maps **variables** to their **sign**



$\gamma_{\text{Sign}} : \text{Sign} \to \mathscr{P}(\mathbb{Z})$

$$\gamma_{\text{Sign}}(\bot) \stackrel{\text{def}}{=} \emptyset$$

$$\gamma_{\text{Sign}}(0) \stackrel{\text{def}}{=} \{0\}$$

$$\vdots$$

$$\gamma_{\text{Sign}}(\top) \stackrel{\text{def}}{=} \mathbb{Z}$$

$\gamma' : A \to \mathscr{P}(\mathscr{E})$ $\quad \gamma'(a) \stackrel{\text{def}}{=} \{\rho \in \mathscr{E} \mid \forall x \in \mathbb{X} : \rho(x) \in \gamma_{\text{Sign}}(a(x))\}$

$\gamma : \mathscr{L} \times A \to \mathscr{P}(\Sigma)$ $\quad \gamma((\ell, a)) \stackrel{\text{def}}{=} \{(\ell, \rho) \in \Sigma \mid \rho(x) \in \gamma'(a)\}$

Sign Static Analysis

```
1
a ← [0, +∞]
2
b ← [0, +∞]
3
q ← 0
4
r ← a
5
while 6(r ≥ b) do
    7
    r ← r − b
    8
    q ← q + 1
    9
done
10
```

$$S \stackrel{\text{def}}{=} \{\langle \ell, \rho \rangle \in \Sigma \mid \ell \in \mathscr{L}, \rho \in \mathscr{E}, \rho(r) \geq 0\}$$

3: $a \mapsto \geq 0 \quad b \mapsto \geq 0$

4: $a \mapsto \geq 0 \quad b \mapsto \geq 0 \quad q \mapsto 0$

5: $a \mapsto \geq 0 \quad b \mapsto \geq 0 \quad q \mapsto 0 \quad r \mapsto \geq 0$

6: $a \mapsto \geq 0 \quad b \mapsto \geq 0 \quad q \mapsto 0 \quad r \mapsto \geq 0$

7: $a \mapsto \geq 0 \quad b \mapsto \geq 0 \quad q \mapsto 0 \quad r \mapsto \geq 0$

8: $a \mapsto \geq 0 \quad b \mapsto \geq 0 \quad q \mapsto 0 \quad r \mapsto \top$

9: $a \mapsto \geq 0 \quad b \mapsto \geq 0 \quad q \mapsto > 0 \quad r \mapsto \top$

6: $a \mapsto \geq 0 \quad b \mapsto \geq 0 \quad q \mapsto \geq 0 \quad r \mapsto \top$

7: $a \mapsto \geq 0 \quad b \mapsto \geq 0 \quad q \mapsto \geq 0 \quad r \mapsto \geq 0$

8: $a \mapsto \geq 0 \quad b \mapsto \geq 0 \quad q \mapsto \geq 0 \quad r \mapsto \top$

9: $a \mapsto \geq 0 \quad b \mapsto \geq 0 \quad q \mapsto > 0 \quad r \mapsto \top$

6: $a \mapsto \geq 0 \quad b \mapsto \geq 0 \quad q \mapsto \geq 0 \quad r \mapsto \top$

10: $a \mapsto \geq 0 \quad b \mapsto \geq 0 \quad q \mapsto \geq 0 \quad r \mapsto \top$

$$\gamma(\mathscr{R}^\natural(I)) \nsubseteq S$$

FALSE ALARM

Static Analysis by Abstract Interpretation

~~$9.95~~ $10

~~$35.85~~ $40

~~$24.95~~ $25

~~$4.85~~ $5

$ 10 +
$ 40 +
$ 25 +
$ 5

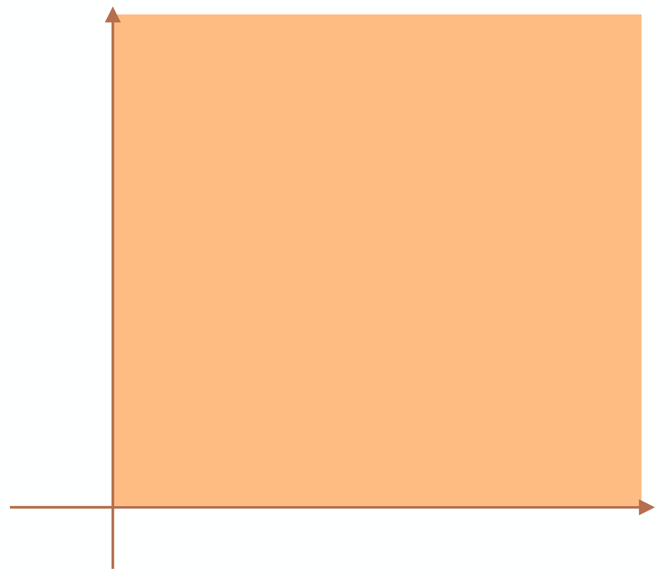
$ 80

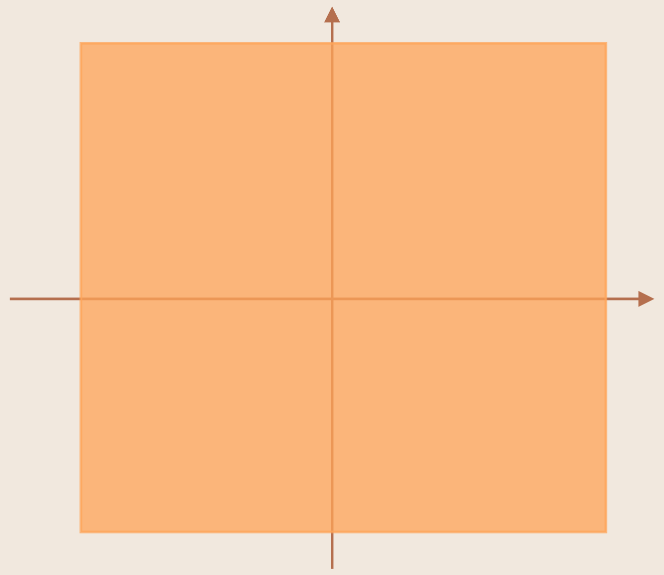
$ 9.95 +
$ 35.85 +
$ 24.95 +
$ 4.85

$ 75.60

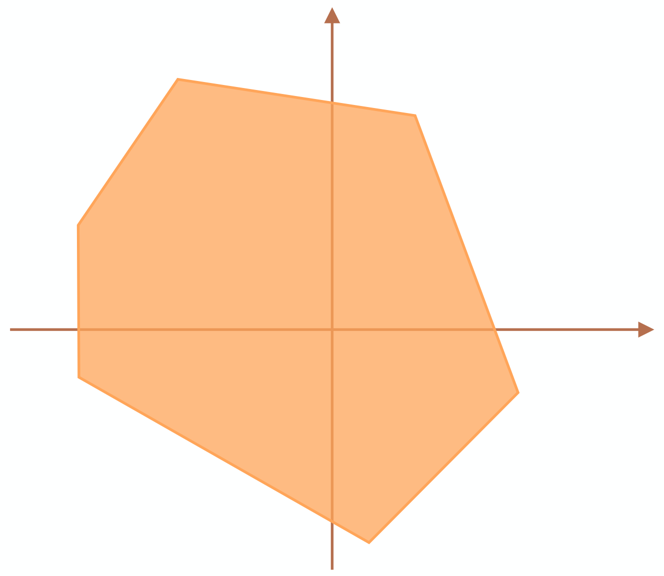
Numerical Abstract Domains

Non-Relational Domains


Sign Domain

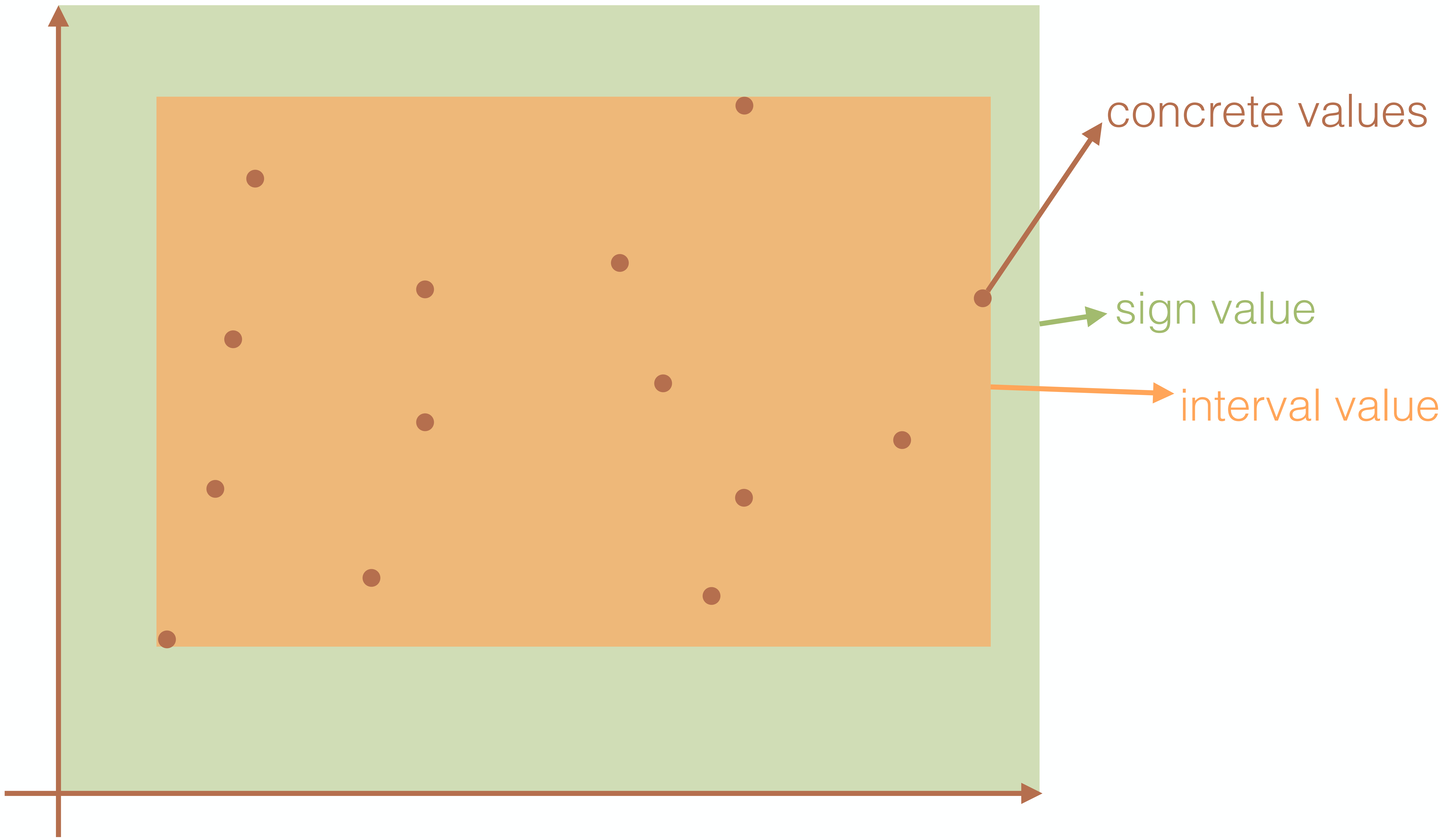

Interval Domain

Relational Domains


Polyhedra Domain

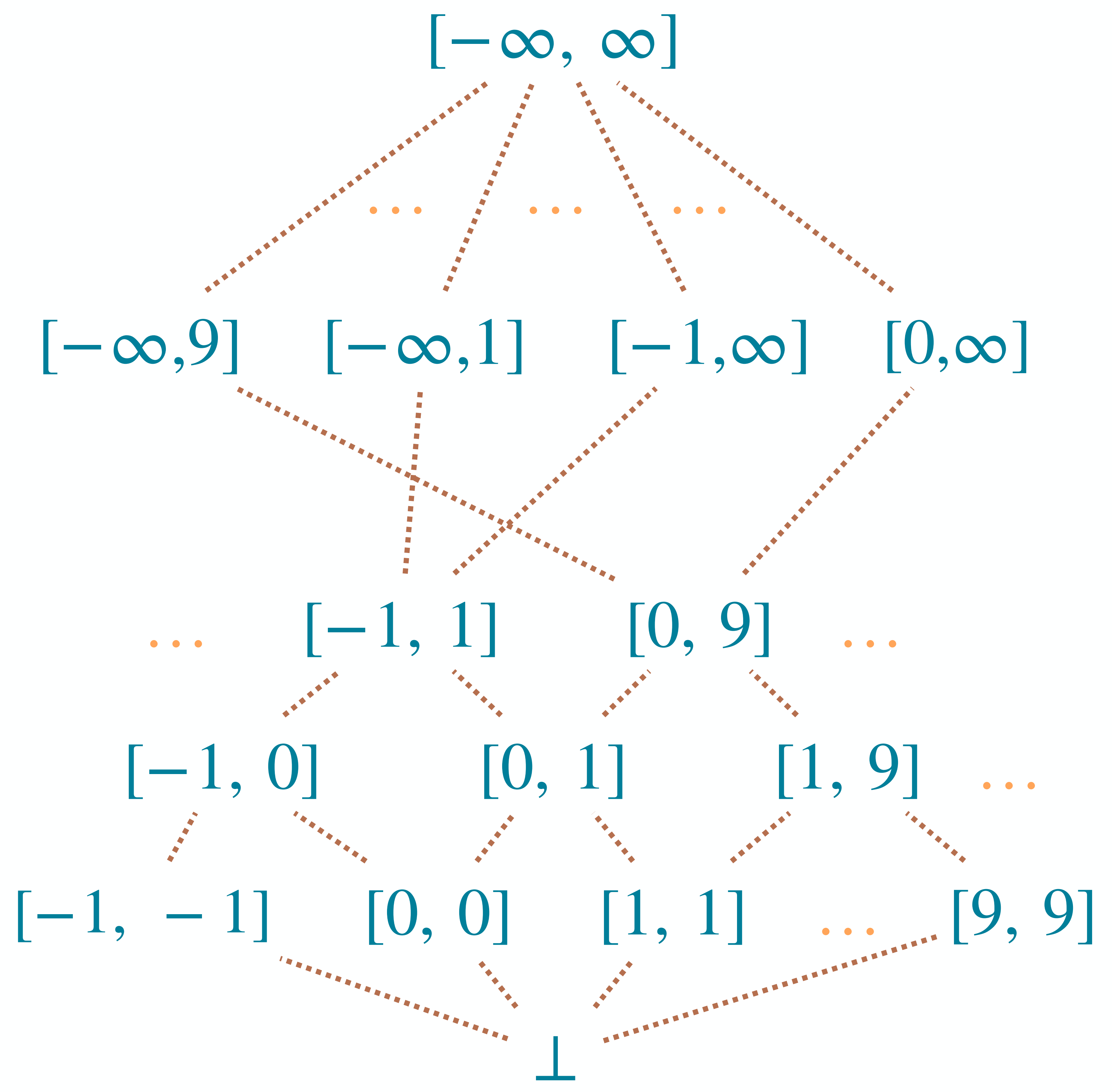
Interval Abstract Domain

Concrete Value are Replaced with Range Values



Interval Abstract Domain

$A : \mathbb{X} \to \mathsf{Itv}$ maps **variables** to their **lower and upper bounds**



$\sqsubseteq_A$ defined by the diagram

$\sqcup_A$ defined by the diagram

$\sqcap_A$ defined by the diagram

$\text{ASSIGN}_A[\![X \leftarrow e]\!]a$ maps X to e evaluated with **interval arithmetic**

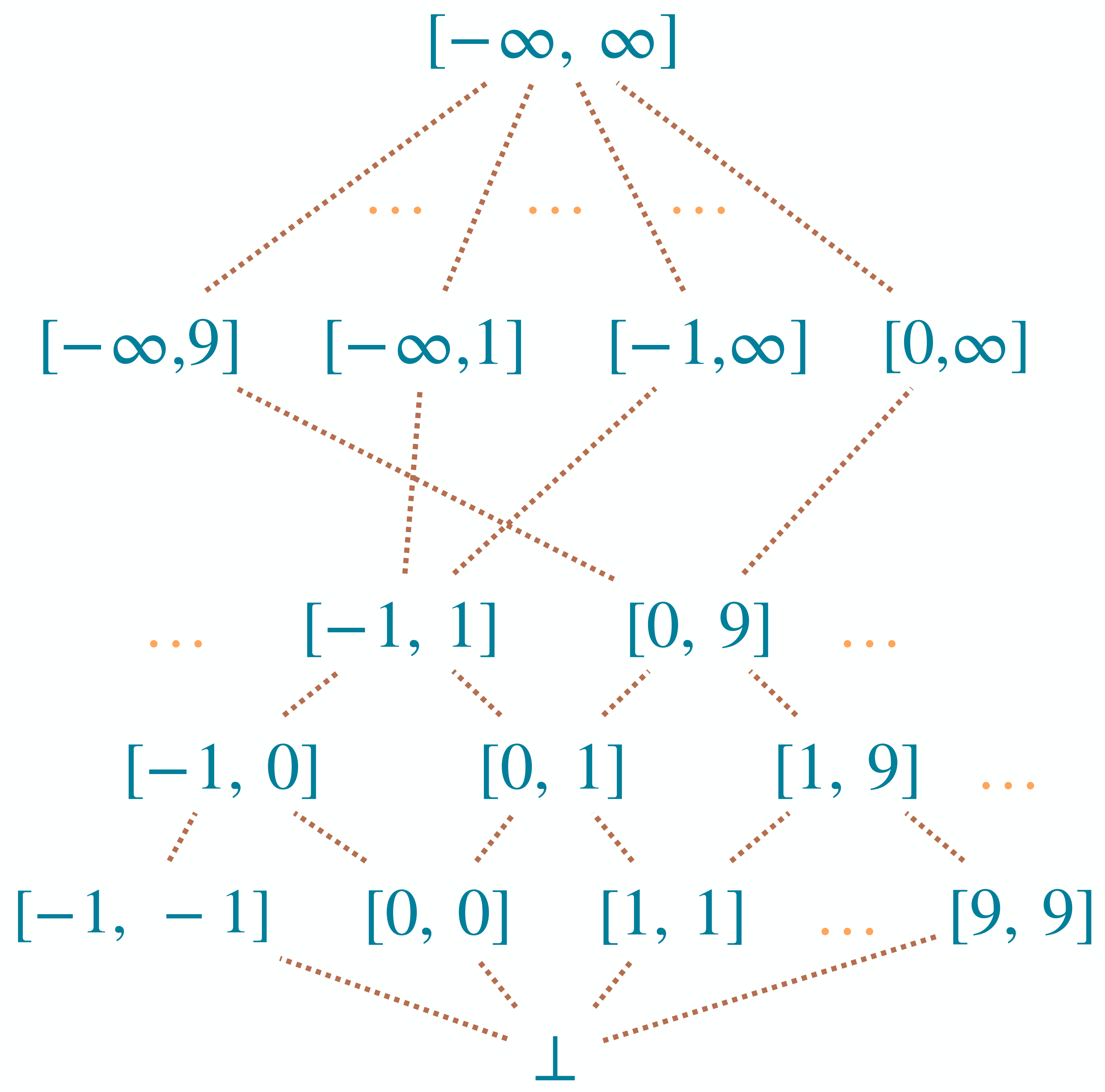
$\text{FILTER}_A[\![e \bowtie 0]\!]a$ modifies a to satisfy $e \bowtie 0$

$\triangledown_A$ loosens the unstable bounds

$$[0, 1] \triangledown_A [0, 2] = [0, \infty]$$
$$[0, 1] \triangledown_A [-1, 1] = [-\infty, 1]$$

Interval Abstract Domain

$A: \mathbb{X} \to \mathsf{Itv}$ maps **variables** to their **lower and upper bounds**



$\gamma_{\mathsf{Itv}}: \mathsf{Itv} \to \mathscr{P}(\mathbb{Z})$

$$\gamma_{\mathsf{Itv}}(\perp) \stackrel{\text{def}}{=} \emptyset$$
$$\gamma_{\mathsf{Itv}}([a, b]) \stackrel{\text{def}}{=} \{x \in \mathbb{Z} \mid a \leq x \leq b\}$$
$$\vdots$$
$$\gamma_{\mathsf{Itv}}([\infty, \infty]) \stackrel{\text{def}}{=} \mathbb{Z}$$

$\gamma': A \to \mathscr{P}(\mathscr{E})$

$$\gamma'(a) \stackrel{\text{def}}{=} \{\rho \in \mathscr{E} \mid \forall x \in \mathbb{X}: \rho(x) \in \gamma_{\mathsf{Itv}}(a(x))\}$$

$\gamma: \mathscr{L} \times A \to \mathscr{P}(\Sigma)$

$$\gamma((\ell, a)) \stackrel{\text{def}}{=} \{(\ell, \rho) \in \Sigma \mid \rho(x) \in \gamma'(a)\}$$

Interval Static Analysis

1

a ← [0, +∞]

2

b ← [0, +∞]

3

q ← 0

4

r ← a

5

while 6(r ≥ b) **do**

7

r ← r − b

8

q ← q + 1

9

done

10

$S \stackrel{\text{def}}{=} \{\langle \ell, \rho \rangle \in \Sigma \mid \ell \in \mathscr{L}, \rho \in \mathscr{E}, \rho(r) \geq 0\}$

$\gamma(\mathscr{R}^{\natural}(I)) \nsubseteq S$

FALSE ALARM

3: $a \mapsto [0,\infty]$ $b \mapsto [0,\infty]$

4: $a \mapsto [0,\infty]$ $b \mapsto [0,\infty]$ $q \mapsto [0,0]$

5: $a \mapsto [0,\infty]$ $b \mapsto [0,\infty]$ $q \mapsto [0,0]$ $r \mapsto [0,\infty]$

6: $a \mapsto [0,\infty]$ $b \mapsto [0,\infty]$ $q \mapsto [0,0]$ $r \mapsto [0,\infty]$

7: $a \mapsto [0,\infty]$ $b \mapsto [0,\infty]$ $q \mapsto [0,0]$ $r \mapsto [0,\infty]$

8: $a \mapsto [0,\infty]$ $b \mapsto [0,\infty]$ $q \mapsto [0,0]$ $\boxed{r \mapsto [-\infty,\infty]}$

9: $a \mapsto [0,\infty]$ $b \mapsto [0,\infty]$ $q \mapsto [1,1]$ $\boxed{r \mapsto [-\infty,\infty]}$

6: $a \mapsto [0,\infty]$ $b \mapsto [0,\infty]$ $q \mapsto [0,1]$ $\boxed{r \mapsto [-\infty,\infty]}$

7: $a \mapsto [0,\infty]$ $b \mapsto [0,\infty]$ $q \mapsto [0,1]$ $r \mapsto [0,\infty]$

8: $a \mapsto [0,\infty]$ $b \mapsto [0,\infty]$ $q \mapsto [0,1]$ $\boxed{r \mapsto [-\infty,\infty]}$

9: $a \mapsto [0,\infty]$ $b \mapsto [0,\infty]$ $q \mapsto [1,2]$ $\boxed{r \mapsto [-\infty,\infty]}$

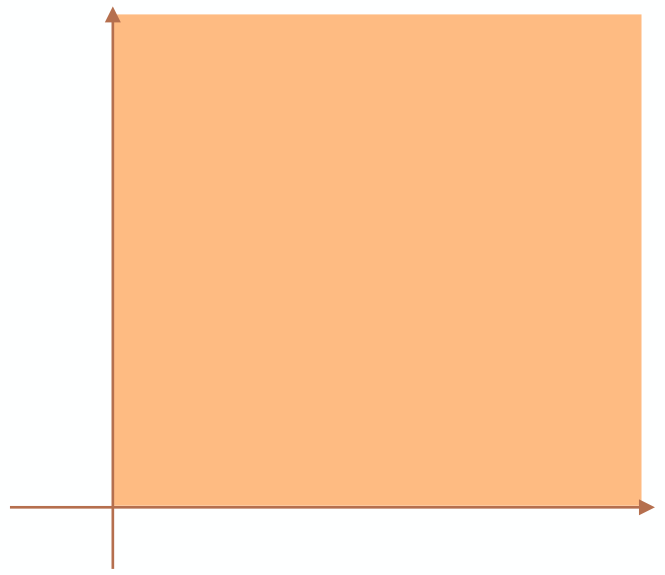
6: $a \mapsto [0,\infty]$ $b \mapsto [0,\infty]$ $q \mapsto [0,\infty]$ $\boxed{r \mapsto [-\infty,\infty]}$

⋮

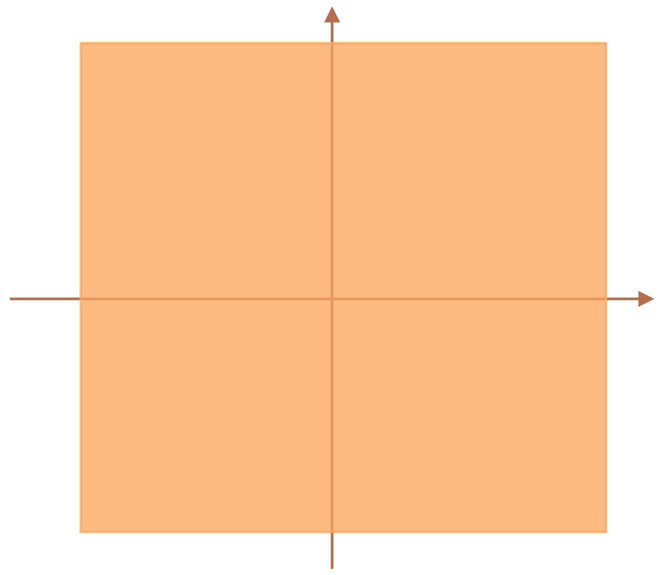
10: $a \mapsto [0,\infty]$ $b \mapsto [0,\infty]$ $q \mapsto [0,\infty]$ $\boxed{r \mapsto [-\infty,\infty]}$

Numerical Abstract Domains

Non-Relational Domains

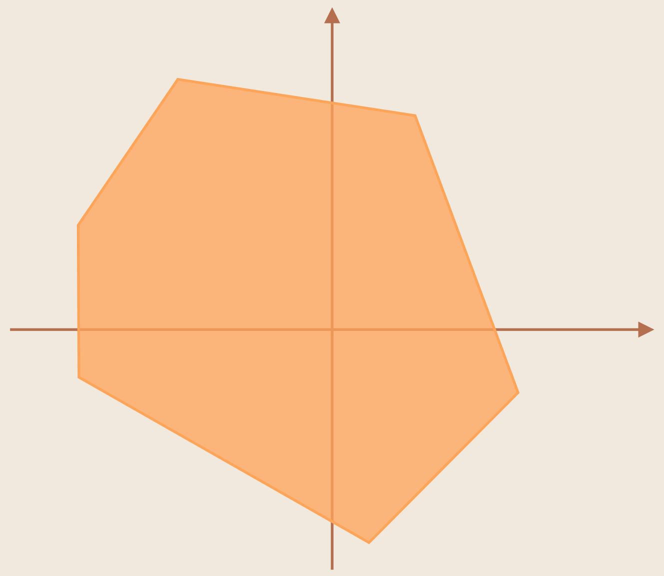


Sign Domain



Interval Domain

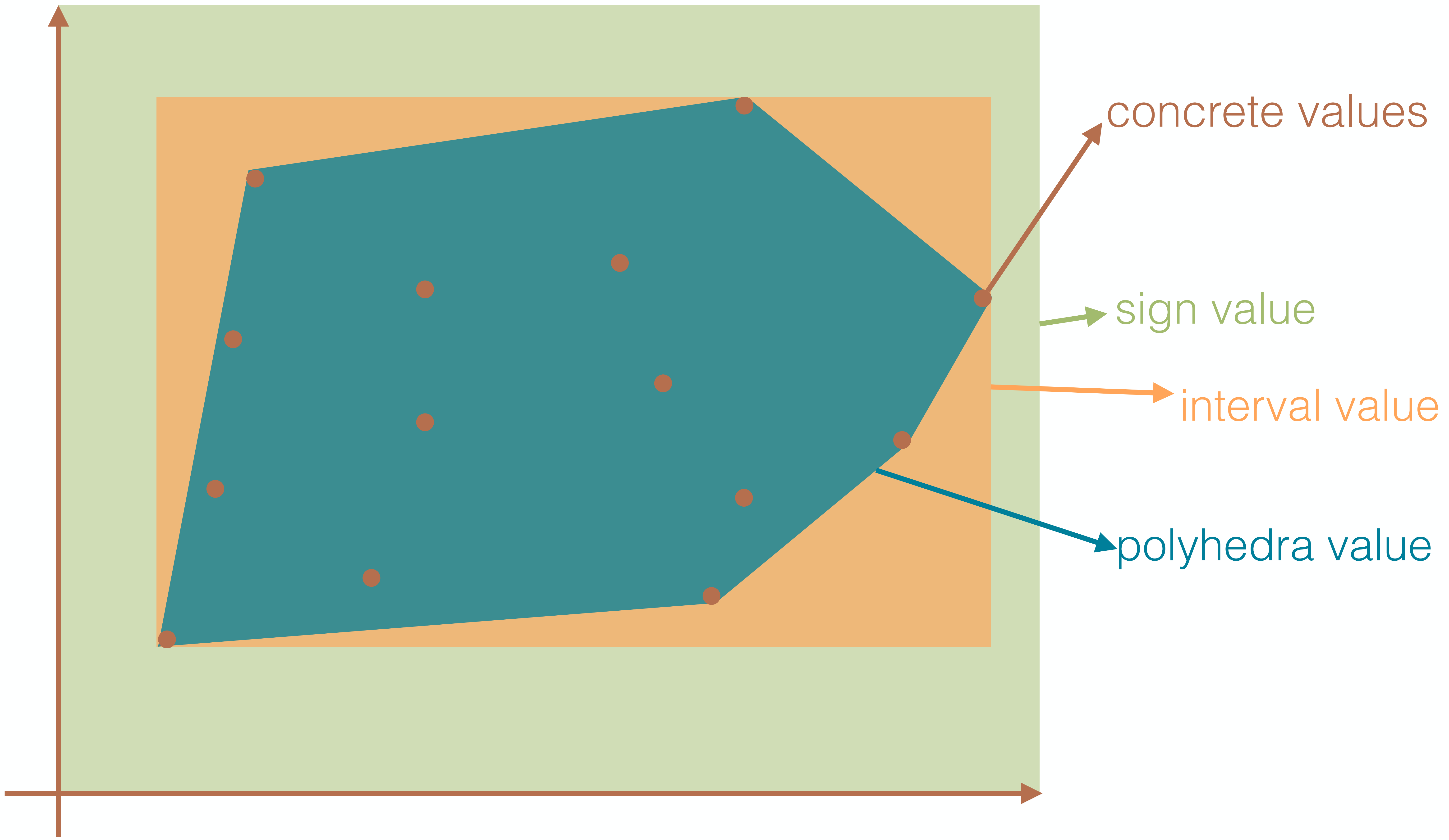
Relational Domains



Polyhedra Domain

Polyhedra Abstract Domain

Concrete Value are Replaced with Conjunctions of Linear Inequalities

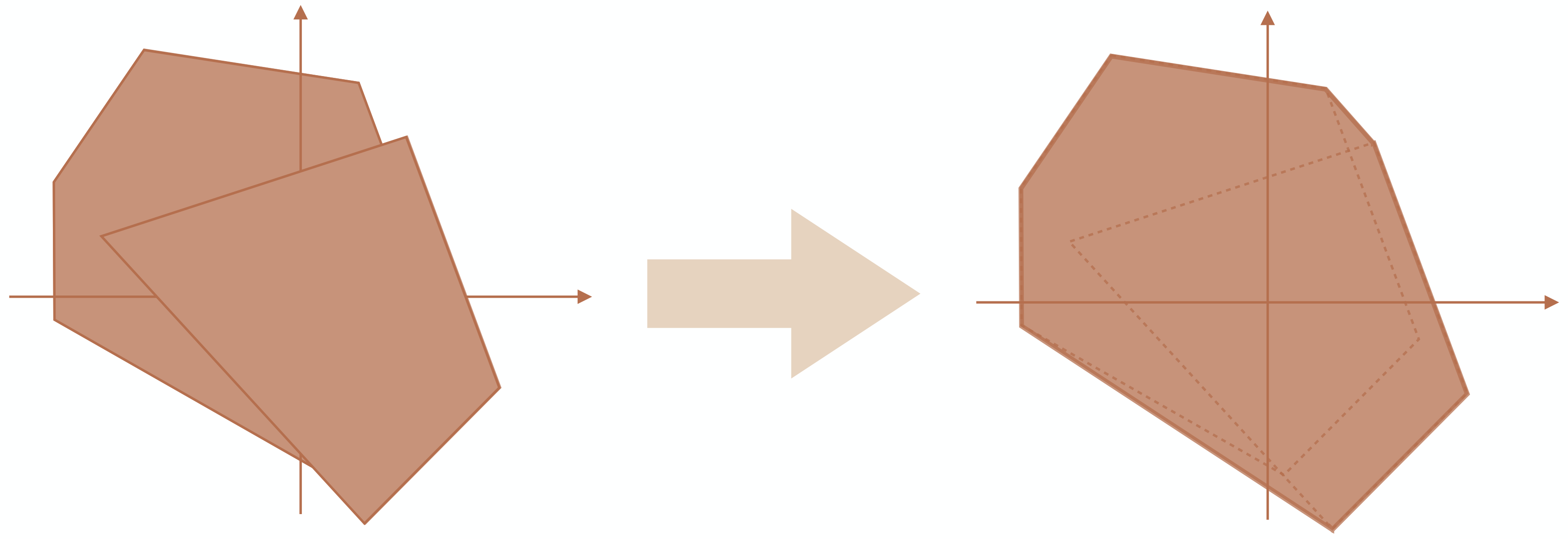


Polyhedra Abstract Domain

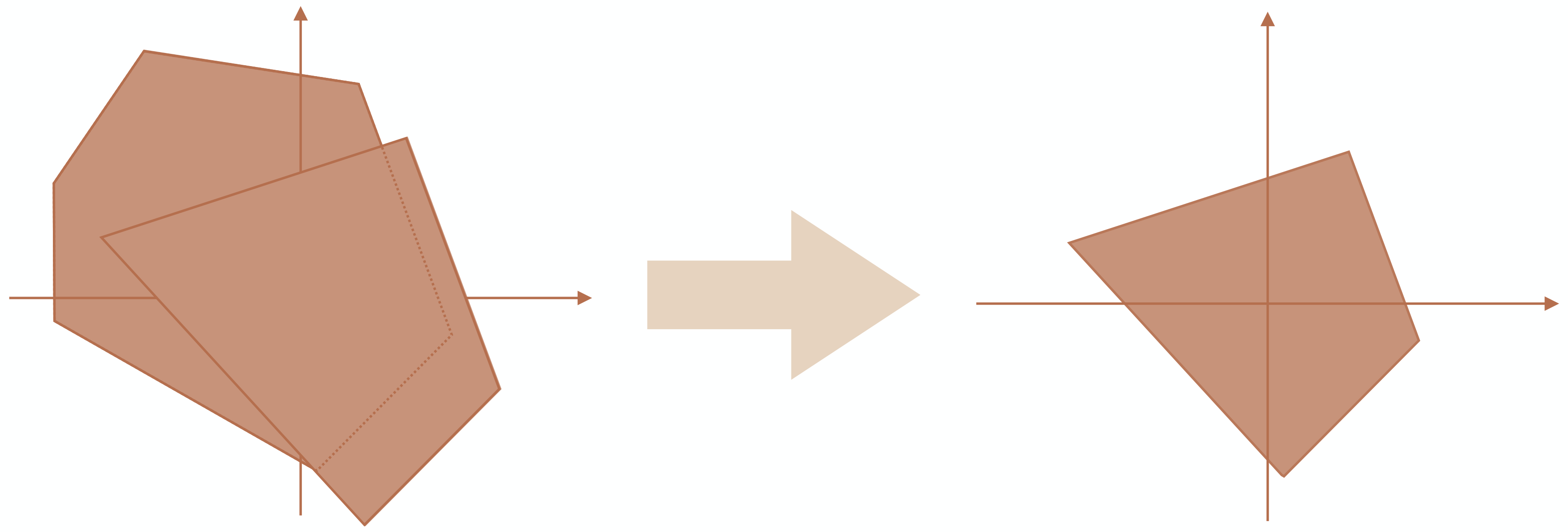
A set of **convex polyhedra**

$\sqsubseteq_A$ inclusion check

$\sqcup_A$ convex hull



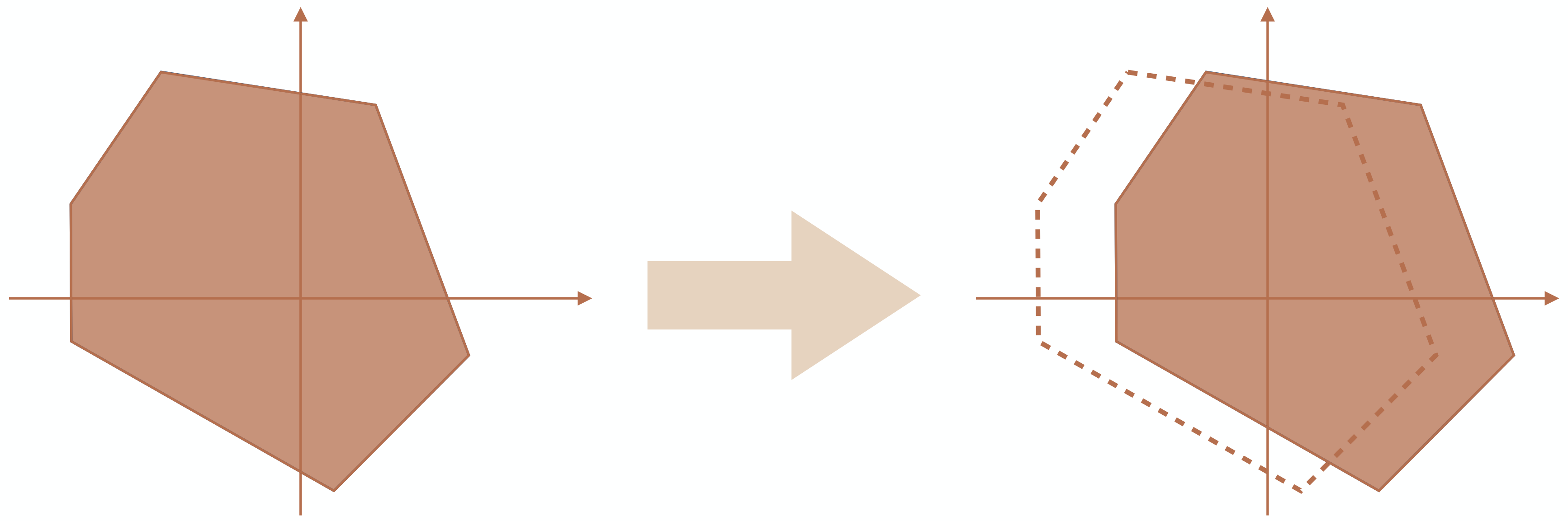
$\sqcap_A$ intersection



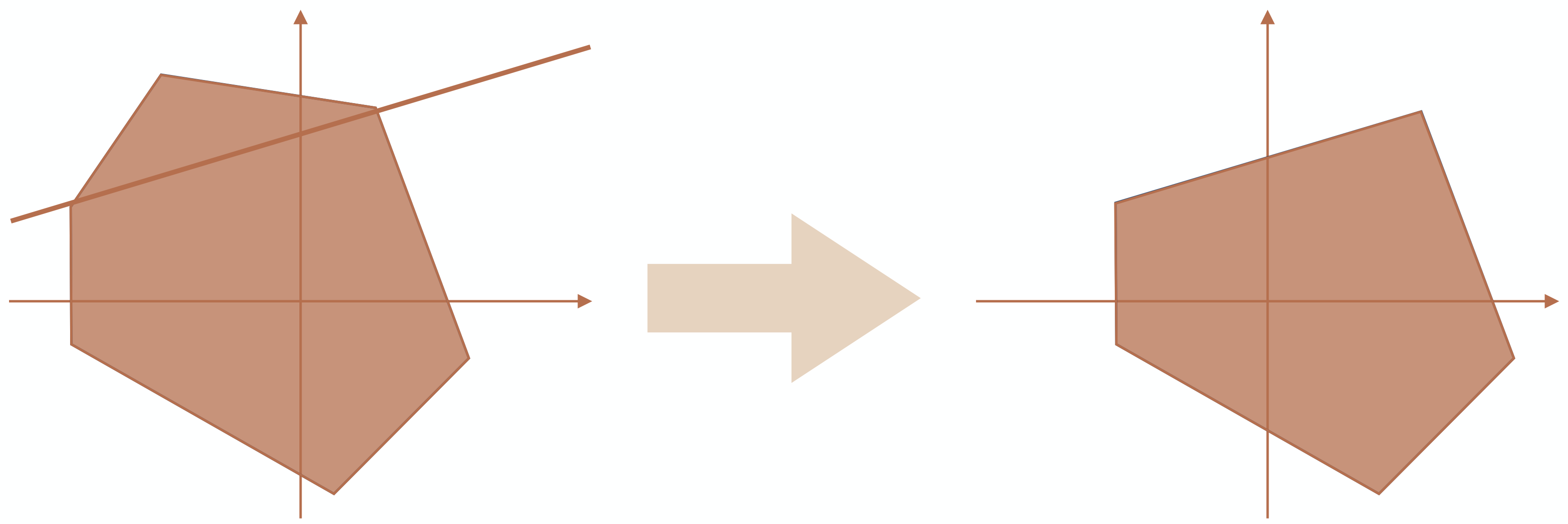
$\mathrm{ASSIGN}_A[\![X \leftarrow e]\!]a$ affine transformation

Example

$\mathrm{ASSIGN}_A[\![X \leftarrow X + 1]\!]a$



$\mathrm{FILTER}_A[\![e]\!]a$ intersection with e



$\triangledown_A$ remove unstable constraints

Polyhedra Abstract Domain

A set of **convex polyhedra**

$\gamma' \colon A \to \mathscr{P}(\mathscr{E})$ set of memory states within the polyhedra

$\gamma \colon \mathscr{L} \times A \to \mathscr{P}(\Sigma)$ $\gamma((\ell, a)) \stackrel{\text{def}}{=} \{(\ell, \rho) \in \Sigma \mid \rho(x) \in \gamma'(a)\}$

Polyhedra Static Analysis

$S \stackrel{\text{def}}{=} \{\langle \ell, \rho \rangle \in \Sigma \mid \ell \in \mathscr{L}, \rho \in \mathscr{E}, \rho(r) \geq 0\}$

$\mathscr{R}(I) \subseteq \gamma(\mathscr{R}^\natural(I)) \subseteq S$

👌

```
1
a ← [0, +∞]
2
b ← [0, +∞]
3
q ← 0
4
r ← a
5
while 6(r ≥ b) do
    7
    r ← r − b
    8
    q ← q + 1
    9
done
10
```

3: $a \geq 0 \wedge b \geq 0$

4: $a \geq 0 \wedge b \geq 0 \wedge q = 0$

5: $a \geq 0 \wedge b \geq 0 \wedge q = 0 \wedge r = a$

6: $a \geq 0 \wedge b \geq 0 \wedge q = 0 \wedge r = a$

7: $a \geq 0 \wedge b \geq 0 \wedge q = 0 \wedge r \geq b$

8: $a \geq 0 \wedge b \geq 0 \wedge q = 0 \wedge r \geq 0$

9: $a \geq 0 \wedge b \geq 0 \wedge q = 1 \wedge r \geq 0$

6: $a \geq 0 \wedge b \geq 0 \wedge 0 \leq q \leq 1 \wedge r \geq 0$

7: $a \geq 0 \wedge b \geq 0 \wedge 0 \leq q \leq 1 \wedge r \geq b$

8: $a \geq 0 \wedge b \geq 0 \wedge 0 \leq q \leq 1 \wedge r \geq 0$

9: $a \geq 0 \wedge b \geq 0 \wedge 1 \leq q \leq 2 \wedge r \geq 0$

6: $a \geq 0 \wedge b \geq 0 \wedge q \geq 0 \wedge r \geq 0$

⋮

10: $a \geq 0 \wedge b \geq 0 \wedge q \geq 0 \wedge b > r \geq 0$

Reading Suggestion

Foundations and Trends® in Programming Languages
Vol. 4, No. 3-4 (2017) 120–372

DOI: 10.1561/2500000034

now
the essence of knowledge

Tutorial on Static Inference of Numeric Invariants by Abstract Interpretation

Antoine Miné
Sorbonne Universités, UPMC Univ. Paris 06, CNRS, LIP6
antoine.mine@lip6.fr

Static Analysis by Abstract Interpretation

Static Analyzers

practical tools
targeting specific programs

abstract semantics, abstract domains
algorithmic approaches to decide program properties

concrete semantics
mathematical models of the program behavior

Astrée

ENS / AbsInt

astree.ens.fr

The Astrée Static Analyzer

Centre National de la Recherche Scientifique — École Normale Supérieure — INRIA (since Sep. 2007)

Participants:

Patrick Cousot (project leader), Radhia Cousot, Jérôme Feret, Antoine Miné, Xavier Rival

Who uses Astrée?

Since 2003, Airbus France has been using Astrée in the development of safety-critical software for various aircraft series, including the A380.

In 2018, Bosch Automotive Steering replaced their legacy tools with Astrée and RuleChecker, resulting in significant savings thanks to faster analyses, higher accuracy, and optimized licensing and support costs.

Framatome employs Astrée for verification of their safety-critical TELEPERM XS platform that is used for engineering, testing, commissioning, operating and troubleshooting nuclear reactors.

The global automotive supplier Helbako in Germany is using Astrée to guarantee that no runtime errors can occur in their electronic control software and to demonstrate MISRA compliance of the code.

In 2008, Astrée proved the absence of any runtime errors in a C version of the automatic docking software of the Jules Verne Automated Transfer Vehicle, enabling ESA to transport payloads to the International Space Station.

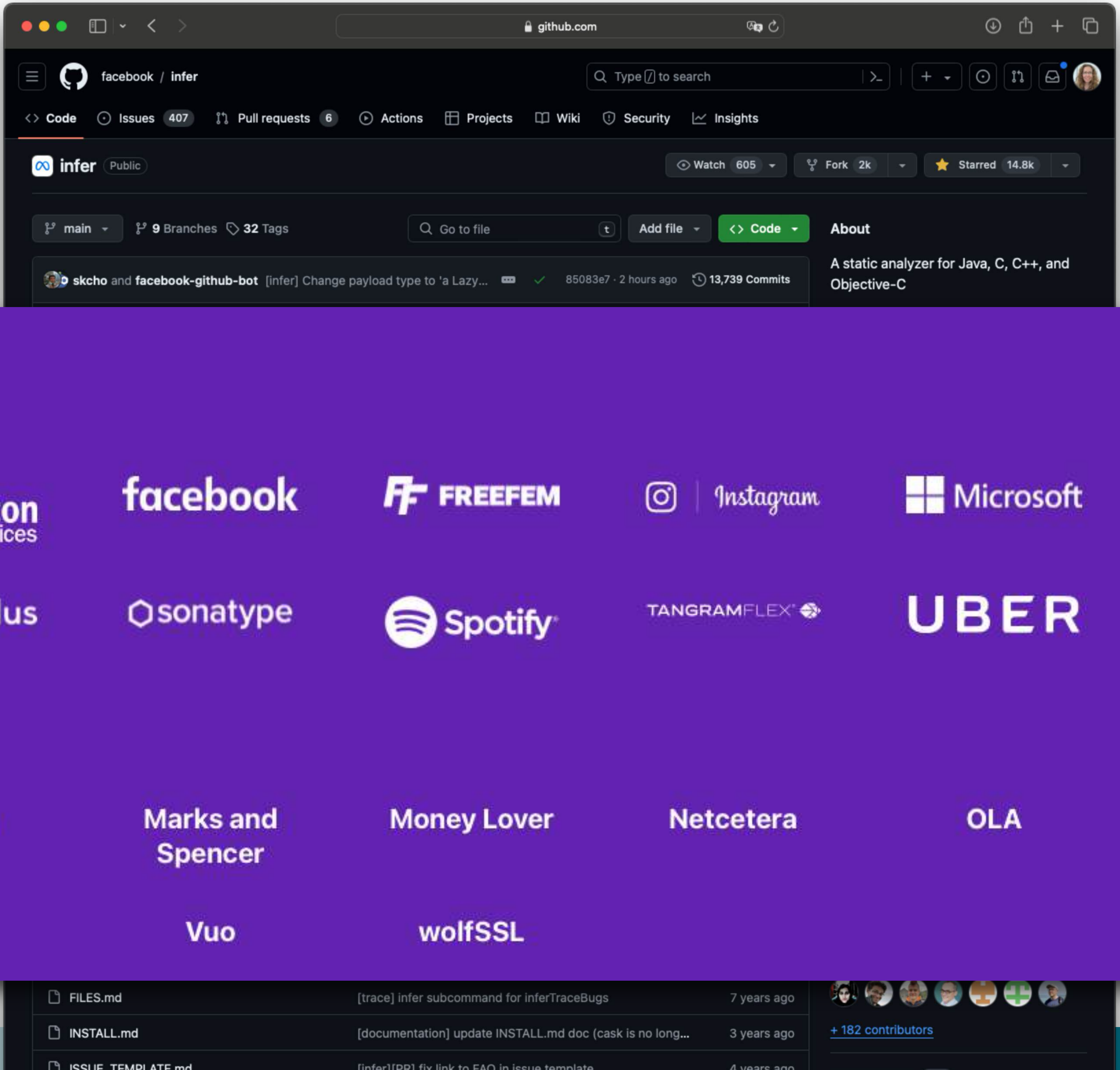
A world leader in motors and ventilators for air-conditioning and refrigeration systems, ebm-papst is using Astrée for fully automatic continuous verification of safety-critical interrupt-driven control software for commutating high-efficiency EC motors for ventilator systems.

Exploitation license of Astrée

Starting Dec. 2009, Astrée is available from AbsInt Angewandte Informatik (www.absint.de/astree/).

Infer

Facebook / Meta



Who Uses Infer?

AdaCore, amazon web services, facebook, FREEFEM, Instagram, Microsoft

mozilla, oculus, sonatype, Spotify, TANGRAMFLEX, UBER

WhatsApp

CodeAI, JD.com, Marks and Spencer, Money Lover, Netcetera, OLA

Sky, Tile, Vuo, wolfSSL

Apron

Library