Observational Completeness on Abstract Interpretation

G. Amato F. Scozzari

Dipartimento di Scienze
Università "G. D'Annunzio" di Chieti-Pescara

WoLLIC 2009

Plan of the talk

What is abstract interpretation

Definition

A theory for approximating the behavior of a discrete dynamic system. [P. Cousot, R. Cousot 77]

Concrete Semantics

Concrete domain

$$\langle C, \leq_C \rangle$$

Concrete semantic function

$$f : C \rightarrow C$$

Concrete semantics

$$\mathcal{S} = \mathrm{lfp}\, f$$

Abstract Semantics

Abstract domain

$$\langle A, \leq_A \rangle$$

Abstract semantic function

$$f^\alpha : A \rightarrow A$$

Abstract semantics

$$\mathcal{S}^\alpha = \mathrm{lfp}\, f^\alpha$$

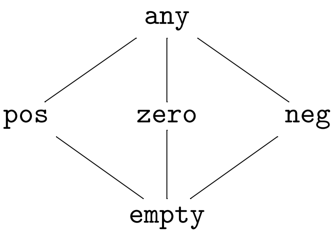
Concrete domain

$$(\wp(\mathbb{Z}), \subseteq)$$

Concretization function

$$\begin{aligned}
\gamma(\texttt{any}) &= \mathbb{Z} \\
\gamma(\texttt{pos}) &= \{x \in \mathbb{Z} \mid x > 0\} \\
\gamma(\texttt{zero}) &= \{0\} \\
\gamma(\texttt{neg}) &= \{x \in \mathbb{Z} \mid x < 0\} \\
\gamma(\texttt{empty}) &= \emptyset
\end{aligned}$$

Abstract domain



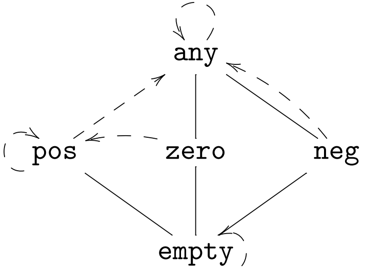
Abstraction function

$$\alpha(X) = \begin{cases} \texttt{bot} & \text{if } X = \emptyset \\ \texttt{zero} & \text{if } X = \{0\} \\ \texttt{pos} & \text{if } \forall x \in X . x > 0 \\ \texttt{neg} & \text{if } \forall x \in X . x < 0 \\ \texttt{any} & \text{otherwise} \end{cases}$$

Correct approximation

$a \in A$ is a correct approximation of $c \in C$ when $\alpha(c) \leq_A a$

Abstract domain and semantic function



Semantic function

$$f(X) = \{x + 1 \mid x \in X\}$$

Best correct abstraction

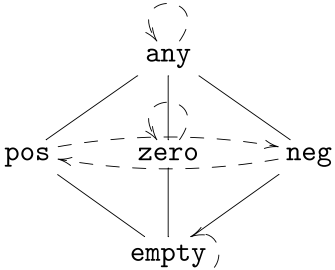
The abstract function f^α is induced by the abstraction: $f^\alpha = \alpha \circ f \circ \gamma$.

Concrete computation: $f(f(\{-1\})) = \{1\}$
Abstract computation: $f^\alpha(f^\alpha(\alpha(\{-1\}))) = \texttt{any}$

We are

- correct, since $\alpha(\{1\}) = \texttt{pos} \leq \texttt{any}$
- not very precise, since pos is a better approximation of $\{1\}$ than any.

Abstract domain and semantic function



Semantic function

$$f(X) = \{-x \mid x \in X\}$$

Concrete computation: $f(f(\{-1\})) = \{-1\}$
Abstract computation: $f^\alpha(f^\alpha(\alpha(\{-1\}))) = \texttt{neg}$

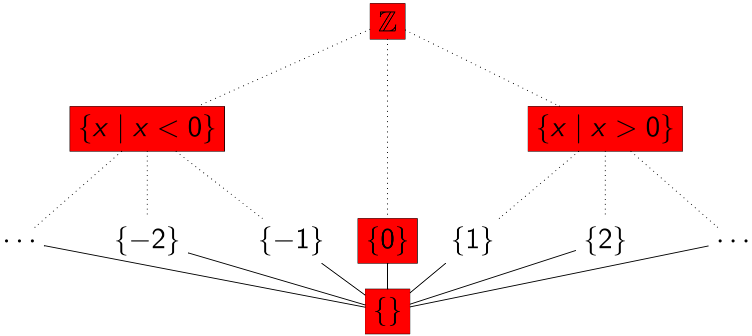
Definition (Completeness)

An abstract intepretation is complete when the result of any abstract computation is the best correct abstraction of the result of the concrete computation:

$$\alpha \circ f = f^\alpha \circ \alpha$$

From Galois insertions to Moore families

1. an abstract domain is a set of *names* for particular elements in C;
2. good for implementation, bad for theory;
3. we identify the abstract domain with its image trough the concretization function γ.



$$A = \{\{\}, \{0\}, \{x \mid x < 0\}, \{x \mid x > 0\}, \mathbb{Z}\}$$

We can ignore γ (which is just the identity for A). For example, the abstract semantic function f^α becomes

$$f^\alpha = \alpha \circ f \ .$$

Subsets of C corresponding to abstract domains are Moore families.

Definition (Moore family)

Given a complete lattice C, a Moore's family of C is a subset of C closed by arbitrary meets.

Theorem

- *The abstraction function α induces a Moore family $\alpha(C)$.*
- *The correspondence between α and $\alpha(C)$ is invertible.*

We use α either to denote the abstraction function or for the corresponding Moore family.

Observables

Let us fix a domain π which describes the properties we are interested in (observable domain).

Problem:

- We want $\pi(\text{lfp } f)$ which is either undecidable or too expensive to compute;
- We may compute $\text{lfp } f^{\pi}$ which is imprecise.

Solution:

1. Choose an intermediate computational domain

$$\pi \subseteq \alpha \subseteq C$$

2. Compute $\pi(\text{lfp } f^{\alpha})$.

How to choose the computational domain

Definition (Observational Completeness)

α is observationally complete (for π) when computing over α we do not lose precision, if we are only interested in observation made over π:

$$\pi \circ \underbrace{f^\alpha \circ \cdots \circ f^\alpha}_{n \text{ times}} \circ \alpha = \pi \circ \underbrace{f \circ \cdots \circ f}_{n \text{ times}}$$

for each $n \in \mathbb{N}$.

There are several observationally complete domains for π.

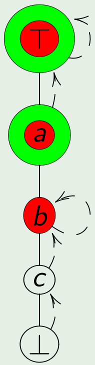
Example (Trivial)

The concrete domain C is observationally complete for any π.

We are interested in smaller domains.

Non-trivial observationally complete domains

Example (1)

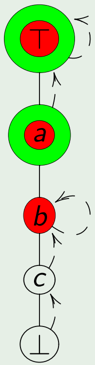


observable domain π computational domain α

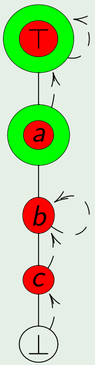
- Concrete computation: $(\pi \circ f \circ f)(\bot) = a$
- Abstract computation: $(\pi \circ \underbrace{f^\alpha}_{\alpha \circ f} \circ \underbrace{f^\alpha}_{\alpha \circ f} \circ \alpha)(\bot) = a$

Non-trivial observationally complete domains

Example (1)



Example (2)



observable domain π

computational domain α

- Supersets preserve observational completeness
- We want the smallest observationally complete domain (l.o.c.)

Continuity and least observationally complete domains

It is possible to show that, in the general case, the l.o.c. domain does not exist. However:

Theorem

If f is continuous, the least observationally complete domain exists.

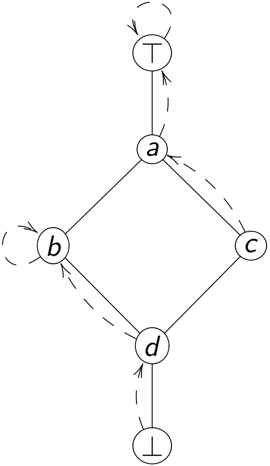
The l.o.c. domain is endowed with a constructive characterization:

$$\alpha = \mathcal{M}\Big(\bigcup \{ \max\{ x \in C \mid (\underbrace{f \circ \cdots \circ f}_{i \text{ times}})(x) \leq a \} \mid i \in \mathbb{N}, a \in \pi \} \Big) ,$$

where $\mathcal{M} : \wp(C) \rightarrow \wp(C)$ is the Moore's closure, i.e.

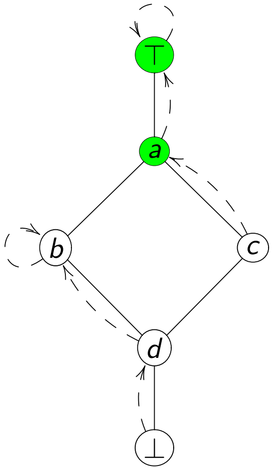
$$\mathcal{M}(S) = \{ \bigwedge X \mid X \subseteq S \} .$$

Example – Building the l.o.c. domain



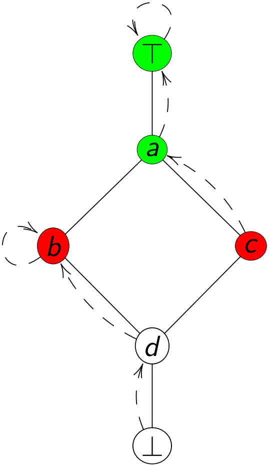
- concrete domain and semantic function f

Example – Building the l.o.c. domain



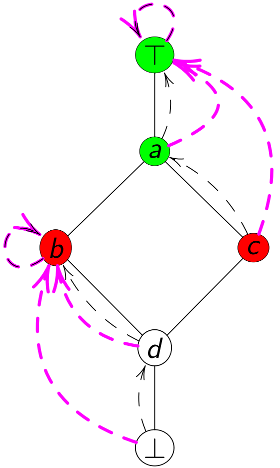
- concrete domain and semantic function f
- observable domain π

Example – Building the l.o.c. domain



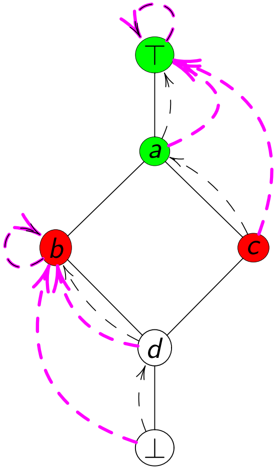
- concrete domain and semantic function f
- observable domain π
- step 1

Example – Building the l.o.c. domain



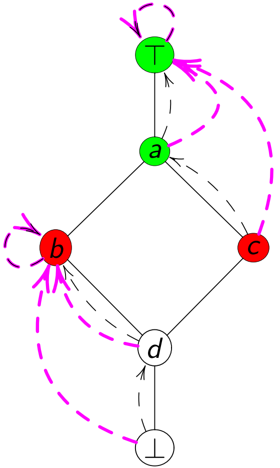
- concrete domain and semantic function f
- observable domain π
- step 1
- semantic function $f \circ f$

Example – Building the l.o.c. domain



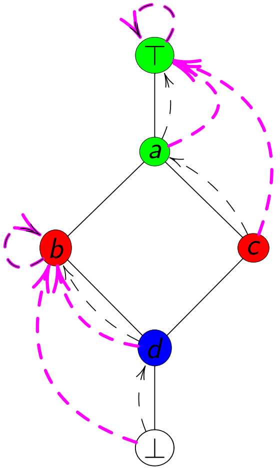
- concrete domain and semantic function f
- observable domain π
- step 1
- semantic function $f \circ f$
- step 2 (no new points)

Example – Building the l.o.c. domain



- concrete domain and semantic function f
- observable domain π
- step 1
- semantic function $f \circ f$
- step 2 (no new points)
- more steps...

Example – Building the l.o.c. domain



- concrete domain and semantic function f
- observable domain π
- step 1
- semantic function $f \circ f$
- step 2 (no new points)
- more steps. . .
- Moore closure

Completeness and Observational Completeness

Compare the two definitions of completeness and observational completeness:

- Completeness

$$f^\alpha \circ \alpha = \alpha \circ f$$

which is equivalent to

$$\underbrace{f^\alpha \circ \cdots \circ f^\alpha}_{n \text{ times}} \circ \alpha = \alpha \circ \underbrace{f \circ \cdots \circ f}_{n \text{ times}}$$

- Observational Completeness

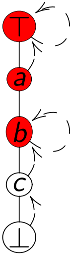
$$\pi \circ \underbrace{f^\alpha \circ \cdots \circ f^\alpha}_{n \text{ times}} \circ \alpha = \pi \circ \underbrace{f \circ \cdots \circ f}_{n \text{ times}}$$

They are similar but, for standard completeness:

- the observable domain is not fixed in advance;
- the short form is equivalent to the long form.

Completeness is not preserved by super-sets

Complete

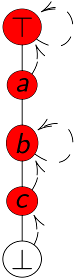


$$(\sigma \circ f \circ \sigma)(\perp) = b$$

and

$$(\sigma \circ f)(\perp) = b$$

Non-complete



$$(\sigma \circ f \circ \sigma)(\perp) = b$$

but

$$(\sigma \circ f)(\perp) = c$$

Why we prefer observational completeness

Theorem

If σ is complete, it is also observationally complete for every $\pi \subseteq \sigma$.

However,

- completeness is a strong property;
- completeness is not closed for supersets, which is very counter-intuitive.

If we want to find a precise computational domain for observing the properties in π, then observational completeness is the property to look for.

Complete shell

Definition (Complete shell)

Given an observable domain π, the least complete domain which includes π is the complete shell of π.

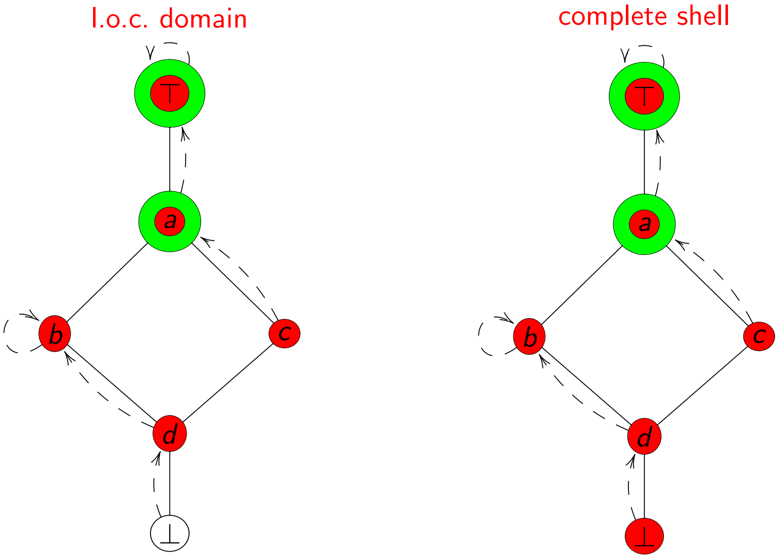
Theorem

If f is continuous, the complete shell of π is the least fixpoint of the refinement

$$\mathcal{R}(\alpha) = \mathcal{M}\Big(\pi \cup \bigcup_{a \in \alpha} \max(\{x \in C \mid f(x) \leq a\})\Big)$$

- it is easier to compute than the l.o.c. domain, since we do not need to consider all the possible compositions of f;
- what is the relationship between the l.o.c. domain and the complete shell?

The complete shell is bigger then the l.o.c. domain



L.o.c. domain and complete shell for additive functions

Theorem

If f is additive, then the l.o.c. domain and the complete shell coincide

- We can use the machinery already developed to compute complete shells for l.o.c. domains.
- Completeness and observational completeness are still different: only their least elements coincide.

Summary of results

Results

- we propose a different notion of completeness for abstract interpretation
- we argue that the new definition is the right one when we want to find out a computational domain which does not lose precision
- we show that, in general, the two notions of completeness are different
- we show that, when the semantic function is additive, the l.o.c. domain coincides with the complete shell

How are concrete and abstract domains related

An abstract interpretation is given by:

- a domain A of properties of elements of C;
- a way to relate A to C, such as a Galois insertions.

Definition (Galois insertion)

Given posets C and A, a Galois insertion $\langle \alpha, \gamma \rangle : C \leftrightarrow A$ is given by

- an abstraction function $\alpha : C \xrightarrow{m} A$ which maps every concrete object $c \in C$ to the strongest property it enjoys;
- a concretization function $\gamma : A \xrightarrow{m} C$ which maps every abstract property $a \in A$ to the biggest concrete object which enjoys the property;

such that

- $\gamma(\alpha(c)) \geq_C c$;
- $\alpha(\gamma(a)) = a$

Correctness and best correct abstraction

Definition (Correctness)

The abstract interpretation is correct when the result of any abstract computation is an correct approximation of the result of the corresponding concrete computation.

Correctness is preserved by fixpoints, i.e. the abstract semantics is a correct approximation of the concrete semantics:

$$\alpha(\mathrm{lfp}\, f) \leq \mathrm{lfp}\, f^{\alpha} \quad .$$

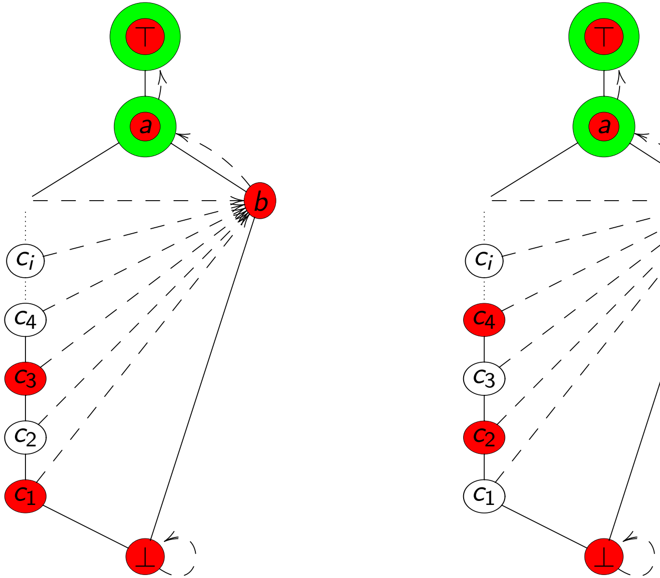
A trivial correct abstract semantic function is

$$f^{\alpha}(x) = \top_A \quad ,$$

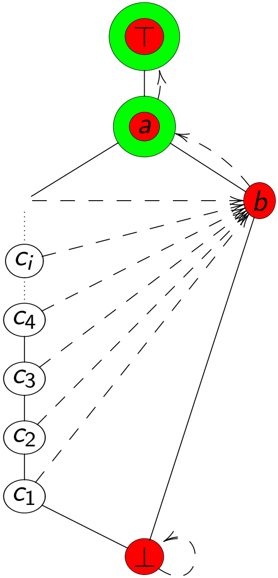
but it is very imprecise. The best correct abstraction is

$$f^{\alpha} = \alpha \circ f \circ \gamma \quad .$$

The least o.c.d. does not always exists 1/2



least o.c.d. does not always exists 2/2



- The intersection domain does not contain any of the c_i's
- It is not observationally complete

$$\pi\alpha f\alpha c_1 = \pi\alpha fa = \pi\alpha\top = \top$$

while

$$\pi fc_1 = \pi b = a$$