

Abstract Interpretation

Wei Le

February 21, 2021

Acknowledgement: this lecture used slides from Profs Alex Aiken, David Schmidt

Outline

- ▶ What is abstract interpretation?
- ▶ Abstract domain
- ▶ Galois connection
- ▶ Design an abstract interpretation system
- ▶ Certify neural networks: abstract interpretation for artificial intelligence (ai², ai4ai)

History: Patrick Cousot, Radhia Cousot 1977

- ▶ Abstract Interpretation: A Unified Lattice Model for Static Analysis of Programs by Construction or Approximation of Fixpoints, 1977
- ▶ Methods and Logics for Proving Programs, 1990
- ▶ Completeness in Abstract Interpretation, 1995
- ▶ Directions for Research in Approximate System Analysis, 1999
- ▶ Probabilistic Abstract Interpretation, 2012
- ▶ An abstract interpretation framework for termination, 2012
- ▶ Abstract interpretation: past, present and future, 2014

What is abstract interpretation?

- ▶ Define an abstract domain and perform computation on abstract domain
- ▶ Soundness: the conclusions from abstract interpretation are correct comparing to the conclusions reached from concrete executions
- ▶ A theoretical framework to formalize *approximation*
- ▶ A sound approximation: the conclusion proved in the abstract domain will be held in the concrete domain
- ▶ Computing semantics on abstract domains: an application of abstraction to the semantics of programming languages
- ▶ abstract interpretation can lose information, meaning some conclusions that can be reached by the concrete executions but cannot be reached by abstract interpretation

An Example

See Prof. Alex Aiken's slide

Partitioning and Abstract Domain

Partition: abstract sets of environments

Abstract domain:

- Intervals (nonrelational):

$$x \Rightarrow [a, b], y \Rightarrow [a', b'], \dots$$

- Polyhedra (relational):

$$x + y - 2z \leq 10, \dots$$

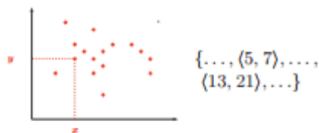
- Difference-bound matrices (weakly relational):

$$y - x \leq 5, z - y \leq 10, \dots$$

- ▶ Interval domain: upper and lower bound
- ▶ Congruence domain: measure density of its values

Abstract domain construction: What are the properties about D that I wish to calculate?

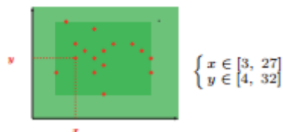
Partitioning and Abstract Domain



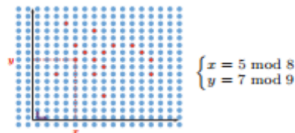
(a) [In]finite Set of Points



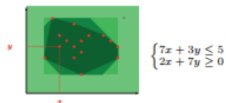
(b) Sign Abstraction



(c) Interval Abstraction



(d) Simple Congruence Abstraction



(b) Polyhedral Abstraction

Abstract interpretation: interval

```
1:  n = 0;  
2:  while n < 1000 do  
3:    n = n + 1;  
4:  end  
5:  exit
```

- Iteration 1: $E_2^\# = [0, 0]$
- Iteration 2: $E_2^\# = [0, 1]$
- Iteration 3: $E_2^\# = [0, 2]$
- Iteration 4: $E_2^\# = [0, 3]$
- ...

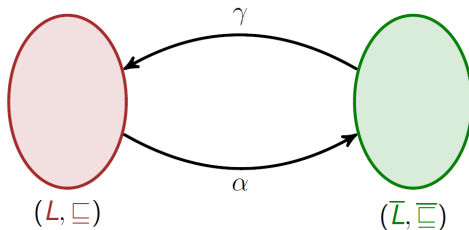
Galois Connection

Abstract interpretation and Galois connection

An abstract interpretation consists of:

- ▶ An abstract domain A ($+, -, 0$) and concrete domain D (Int)
- ▶ Concretization γ and abstraction functions σ , forming a *Galois connection*
- ▶ A (sound) abstract semantic function

Galois Connection: intuition



Concretization

γ is the **concretization** function.

$\gamma(\bar{y})$ is the concrete value in L that is **represented** by \bar{y} .

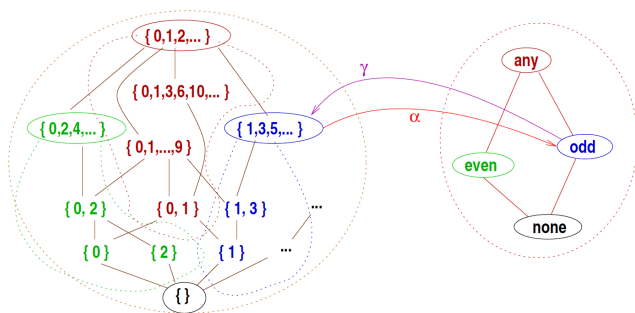
Abstraction

α is the **abstraction** function.

$\alpha(x)$ is the **most precise** abstract value in \bar{L} whose concretization **approximates** x .

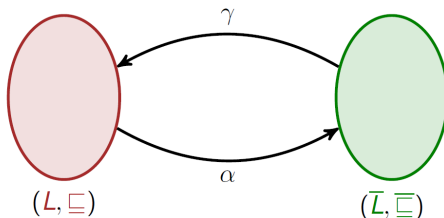
The "order" (property) is preserved: Soundness

Galois connection: example



That is, for all $c \in \gamma[A]$, $c = \gamma \circ \alpha(c)$; for all $a \in \alpha[C]$, $a = \alpha \circ \gamma(a)$.

Galois Connection: Definition



Definition

A **Galois connection** between a lattice (L, \subseteq) and a lattice (\bar{L}, \subseteq) is a pair of functions (α, γ) , with $\alpha : L \rightarrow \bar{L}$ and $\gamma : \bar{L} \rightarrow L$, satisfying:

$$\alpha(x) \subseteq \bar{y} \quad \text{iff} \quad x \subseteq \gamma(\bar{y}) \quad (\text{for all } x \in L, \bar{y} \in \bar{L})$$

Notation for Galois connections: $(L, \subseteq) \xrightleftharpoons[\alpha]{\gamma} (\bar{L}, \subseteq)$

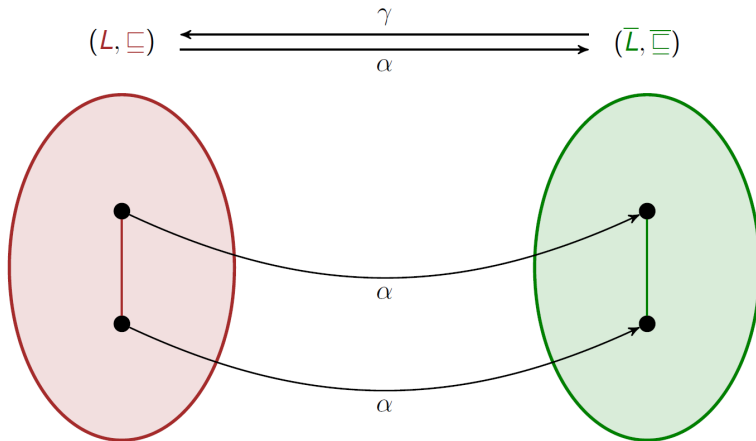
Galois Connection: monotonic function

Consider two lattices (L, \sqsubseteq) and $(\bar{L}, \bar{\sqsubseteq})$.

For any two functions $\alpha : L \rightarrow \bar{L}$ et $\gamma : \bar{L} \rightarrow L$, we have

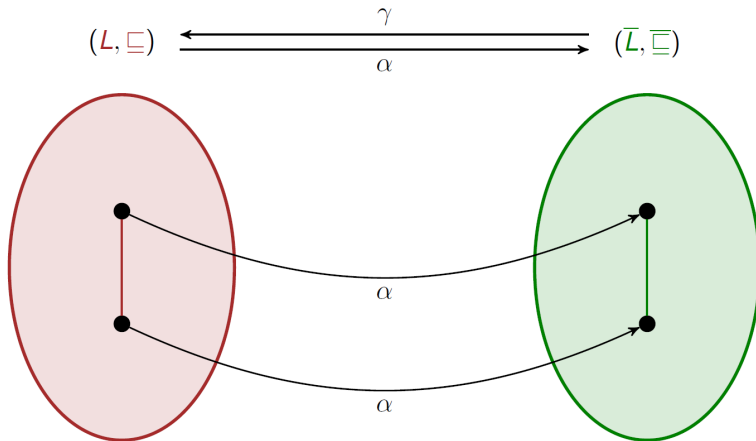
$$(L, \sqsubseteq) \xrightleftharpoons[\alpha]{\gamma} (\bar{L}, \bar{\sqsubseteq}) \quad \text{iff} \quad \left\{ \begin{array}{ll} x \sqsubseteq \gamma \circ \alpha(x) & (\text{for all } x \in L) \\ \alpha \circ \gamma(\bar{y}) \bar{\sqsubseteq} \bar{y} & (\text{for all } \bar{y} \in \bar{L}) \\ \alpha \text{ is monotonic} \\ \gamma \text{ is monotonic} \end{array} \right.$$

Galois Connection: monotonic function



α is monotonic

Galois Connection: monotonic function



α is monotonic

Designing an abstract interpretation system: an example

Now that we know how to model $c \in C$ by $\alpha(c) \in A$, we must model concrete computation steps, $f : C \rightarrow C$, by abstract computation steps, $f^\# : A \rightarrow A$.

Example: We have concrete domain, Nat , and concrete operation, $\text{succ} : \text{Nat} \rightarrow \text{Nat}$, defined as $\text{succ}(n) = n + 1$.

We have abstract domain, Parity , and abstract operation, $\text{succ}^\# : \text{Parity} \rightarrow \text{Parity}$, defined as

$$\text{succ}^\#(\text{even}) = \text{odd}, \quad \text{succ}^\#(\text{odd}) = \text{even}$$

$$\text{succ}^\#(\text{any}) = \text{any}, \quad \text{succ}^\#(\text{none}) = \text{none}$$

$\text{succ}^\#$ must be consistent (sound) with respect to succ :

$$\text{if } n \mathcal{R}_{\text{Nat}} a, \text{ then } \text{succ}(n) \mathcal{R}_{\text{Nat}} \text{succ}^\#(a)$$

where $\mathcal{R} \subseteq \text{Nat} \times \text{Parity}$ relates numbers to their parities (e.g., $2 \mathcal{R}_{\text{Nat}} \text{even}$, $5 \mathcal{R}_{\text{Nat}} \text{odd}$, etc.).

Designing an abstract interpretation system: an example

Example 1: $n \mathcal{R}_{\text{Nat}} a$ **implies** $\text{succ}(n) \mathcal{R}_{\text{Nat}} \text{succ}^\#(a)$

Galois connection: $\wp(\text{Nat}) \langle \alpha, \gamma \rangle \text{Parity}$

$\text{succ}^* : \wp(\text{Nat}) \rightarrow \wp(\text{Nat})$

$\text{succ}^*(S) = \{\text{succ}(n) \mid n \in S\}$

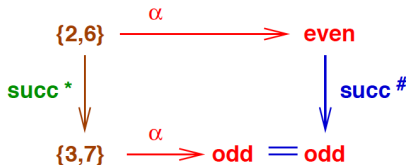
where $\text{succ}(n) = n + 1$

$\text{succ}^\# : \text{Parity} \rightarrow \text{Parity}$

$\text{succ}^\#(\text{even}) = \text{odd}, \quad \text{succ}^\#(\text{odd}) = \text{even}$

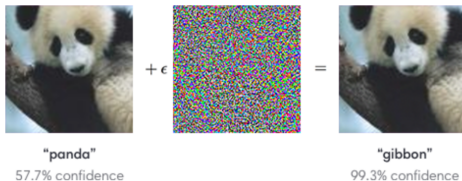
$\text{succ}^\#(\text{any}) = \text{any}, \quad \text{succ}^\#(\text{none}) = \text{none}$

We have that $\alpha \circ \text{succ}^* = \text{succ}^\# \circ \alpha$:



Abstract interpretation for robust neural networks

What does it mean to prove the robustness of a neural network?









Attack	Original	Perturbed	Diff
FGSM [12], $\epsilon = 0.3$			
Brightening, $\delta = 0.085$			

Fig. 1: Attacks applied to MNIST images [25].

Abstract interpretation for robust neural networks

Why can we use abstract interpretation?

- ▶ Deep Neural Nets: Affine transforms + Restricted nonlinearity
- ▶ Abstract Interpretation: Scalable and Precise Numerical Domains
- ▶ Abstract semantics should be defined on: Affine transformation (multiplication and addition), ReLu

Abstract interpretation for robust neural networks

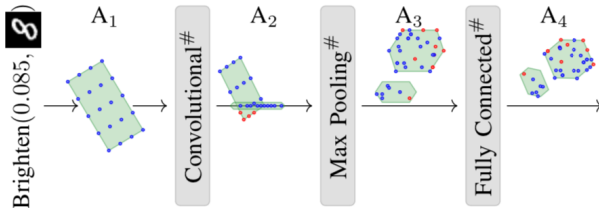
What is the abstract domain?

- ▶ Abstract domain: shapes expressible as a set of logical constraints
- ▶ Zonotope: a center-symmetric convex closed polyhedron [CAV09]

Abstract interpretation for robust neural networks

High level ideas:

- ▶ abstract element: here A1 is an abstract element that represent a group of inputs: a digit 8 and all the images generated through the perturbation (red indicates that additional images are also included)
- ▶ abstract layer: process abstract element
- ▶ abstract transformer: design abstract semantics for each concrete transformation available in the neural network
- ▶ A4 is an overapproximation of input of interest
- ▶ verify robustness on A4



Abstract interpretation for robust neural networks

Further reading:

AI2: Safety and Robustness Certification of Neural Networks with Abstract Interpretation