Abstract Interpretation

Steven Schäfer, Florian Benz

Saarland University

May 29, 2010
Overview

(i) Correctness of an analysis.
(ii) Computation and approximation of fixed points.
(iii) Construction of Galois connections.
(iv) Induced analysis functions.
Abstract Interpretation
Correctness of an Analysis

Notation

Program Semantics

A program $p$ transforms values from a set $V$. Let $v_1, v_2 \in V$:

$$p \vdash v_1 \leadsto v_2$$
Notation

**Program Semantics**
A program $p$ transforms values from a set $V$. Let $v_1, v_2 \in V$:

$$p \vdash v_1 \rightsquigarrow v_2$$

**Program Analysis**
An analysis computes properties of values. Let $L$ be the set of properties, $l_1, l_2 \in L$:

$$p \vdash l_1 \triangleright l_2$$
Abstract Interpretation
Correctness of an Analysis

Notation

Program Semantics
A program \( p \) transforms values from a set \( V \). Let \( v_1, v_2 \in V \):

\[
p \vdash v_1 \sim v_2
\]

Program Analysis
An analysis computes properties of values. Let \( L \) be the set of properties, \( l_1, l_2 \in L \):

\[
p \vdash l_1 \triangleright l_2
\]

An analysis should be deterministic, and therefore can be described by a function \( f_p : L \rightarrow L \).
Correctness

Correctness Relation

Let $R \subseteq V \times L$, so that $v \ R \ l \leftrightarrow v$ can be described by $l$.
Correctness

Correctness Relation

Let $R \subseteq V \times L$, so that $v \ R \ l \ \Leftrightarrow \ v$ can be described by $l$

An analysis is **correct** if $R$ is preserved under computation:

$$p \vdash v_1 \rightsquigarrow v_2$$

$$R \quad R$$

$$p \vdash l_1 \triangleright l_2$$
Correctness

Correctness Relation

Let $R \subseteq V \times L$, so that $v R l \iff v$ can be described by $l$

An analysis is correct if $R$ is preserved under computation:

\[
\begin{align*}
 p & \vdash v_1 \quad \leadsto \quad v_2 \\
 R & \quad \quad \quad R \\
 p & \vdash l_1 \quad \triangleright \quad l_2
\end{align*}
\]

(Formally: $v_1 R l_1 \land p \vdash v_2 \land p \vdash l_1 \triangleright l_2 \Rightarrow v_2 R l_2$)
Consider the program (with $V = \mathbb{Z}$):

$$\text{if } x = 0 \text{ then } 1 \text{ else } x$$
Example

Consider the program (with $V = \mathbb{Z}$):

\[
\text{if } x = 0 \text{ then } 1 \text{ else } x
\]

The analysis determines the possible signs of the results.
Let $L = \mathcal{P}(\text{Sign})$, Sign = \{-, 0, +\}.
Example

Consider the program (with $V = \mathbb{Z}$):

\[
\text{if } x = 0 \text{ then } 1 \text{ else } x
\]

The analysis determines the possible signs of the results. Let $L = \mathcal{P}(\text{Sign})$, $\text{Sign} = \{-, 0, +\}$.

The correctness relation is:

\[
R = \{(V, L) \in \mathbb{Z} \times \mathcal{P}(\text{Sign}) | \text{sgn}(v) \in L\}
\]


Example

Consider the program:

\[
\text{if } x = 0 \text{ then 1 else } x
\]

There are several possible analysis functions, which are correct according to \( R \):

(i) \( f_{p1}(S) = S[0 \rightarrow +] \)
Example

Consider the program:

\[
\text{if } x = 0 \text{ then } 1 \text{ else } x
\]

There are several possible analysis functions, which are correct according to \(R\):

(i) \(f_{p1}(S) = S[0 \rightarrow +]\)
(ii) \(f_{p2}(S) = (S - \{0\}) \cup \{+\}\)
Example

Consider the program:

\[
\text{if } x = 0 \text{ then } 1 \text{ else } x
\]

There are several possible analysis functions, which are correct according to \( R \):

(i) \( f_{p1}(S) = S[0 \rightarrow +] \)

(ii) \( f_{p2}(S) = (S - \{0\}) \cup \{+\} \)

(iii) \( f_{p3}(S) = \{-, 0, +\} \)
Let $\sqsubseteq$ be a partial order on $L$, so that $(L, \sqsubseteq, \sqcap, \sqcup, \top, \bot)$ forms a complete lattice.

“$\sqsubseteq$” describes the precision of a property.

$l_1 \sqsubseteq l_2 :\iff l_2$ describes at least the same values as $l_1$. 
Consistency with $R$

Impose the following requirements on $R$:

(i) $v R l_1 \land l_1 \subseteq l_2 \Rightarrow v R l_2$
Consistency with $R$

Impose the following requirements on $R$:

(i) $v \ R l_1 \land l_1 \subseteq l_2 \Rightarrow v \ R l_2$

(ii) $\forall l \in L' \subseteq L, v \ R l \Rightarrow v \ R (\bigcap L')$
Consistency with $R$

Impose the following requirements on $R$:

(i) $v R l_1 \land l_1 \sqsubseteq l_2 \Rightarrow v R l_2$

(ii) $\forall l \in L' \subseteq L, v R l \Rightarrow v R (\bigcap L')$

This also implies:

(i) $v R \top$
Consistency with $R$

Impose the following requirements on $R$:

(i) $v R l_1 \land l_1 \subseteq l_2 \Rightarrow v R l_2$

(ii) $\forall l \in L' \subseteq L, v R l \Rightarrow v R (\bigcap L')$

This also implies:

(i) $v R \top$

(ii) $v R l_1 \land v R l_2 \Rightarrow v R (l_1 \cap l_2)$
Example (cont’d)

Every powerset forms a complete lattice, in this example:
\((\mathcal{P}(\text{Sign}), \subseteq, \cap, \cup, \text{Sign}, \emptyset)\)

\[ f_{p1}(S) = S[0 \rightarrow +] \]
\[ f_{p2}(S) = (S - \{0\}) \cup \{+\} \]
\[ f_{p3}(S) = \{-, 0, +\} \]
are monotone functions on this lattice.
Let $L' = \{ l \mid v \ R \ l \}$. Since

$$\forall l \in L' \subseteq L, v \ R \ l \Rightarrow v \ R \ (\bigcap L')$$

...there should be a “most precise” property $\beta(v) = \bigcap L'$ to represent $v$. 

Representation Functions

Let $\beta : V \rightarrow L$ be a total function on $V$. 

Formally:

$\beta(v_1) \sqsubseteq l_1 \land p \vdash v_1 \xrightarrow{} v_2 \land p \vdash l_1 \sqsubseteq l_2 \Rightarrow \beta(v_2) \sqsubseteq l_2$
Representation Functions

Let $\beta : V \rightarrow L$ be a total function on $V$.

Express correctness through $\beta$ by:

$$
\begin{array}{c}
\frac{p \vdash v_1 \leadsto v_2}{\beta(v_1) \sqsubseteq \beta(v_2)} \\
\frac{p \vdash l_1 \triangleright v_2}{\quad \quad \quad \quad \quad \quad \quad}
\end{array}
$$
Representation Functions

**Representation Function**

Let $\beta : V \rightarrow L$ be a total function on $V$.

Express correctness through $\beta$ by:

\[
\begin{align*}
  p \vdash v_1 & \leadsto v_2 \\
  \beta(v_1) & \sqsubseteq \beta(v_2) \\
  p \vdash l_1 & \triangleright v_2
\end{align*}
\]

(Formally: $\beta(v_1) \sqsubseteq l_1 \land p \vdash v_1 \leadsto v_2 \land p \vdash l_1 \triangleright l_2 \Rightarrow \beta(v_2) \sqsubseteq l_2$)
Equivalence of correctness formulations

(i) $R_\beta = \{(v, l) \in V \times L \mid \beta(v) \sqsubseteq l\}$

(ii) $\beta_R(v) = \bigsqcap\{l \mid v \ R \ l\}$
Definition

Let \((L, \subseteq, \cap, \cup, \top, \bot)\) be a complete lattice. A function \(f : L \to L\) is monotone (order-preserving) if: \(l_1 \subseteq l_2 \Rightarrow f(l_1) \subseteq f(l_2)\).
Monotone functions on complete lattices

\[ f_{p2}(S) = (S - \{0\}) \cup \{+\} \]

\( f \) partitions the lattice in:

\[ \text{Red}(f) = \{S \mid f(S) \sqsubseteq S\} \]
\[ \text{Ext}(f) = \{S \mid S \sqsubseteq f(S)\} \]
\[ \text{Fix}(f) = \text{Red}(f) \cap \text{Ext}(f) \]

...and incomparable elements
Abstract Interpretation
Monotone functions on complete lattices

Properties

Lemma

Red(f), Ext(f) and Fix(f) are closed under f, that is:
l ∈ Red(f) ⇒ f(l) ∈ Red(f), etc.

Let l ∈ Red(f):

\[ f(l) \sqsubseteq l \Rightarrow f^2(l) \sqsubseteq f(l) \Rightarrow f(l) \in Red(f) \]

... the other cases are analogous.
Properties

Tarski’s fixed point theorem

$\text{Fix}(f)$ is a non-empty complete lattice. Specifically $f$ has a least and greatest fixed point, $\text{lfp}(f)$ and $\text{gfp}(f)$ respectively.
The fixed point theorem

It is possible to show that:

(i) $\text{lfp}(f) = \bigsqcap Fix(f) = \bigsqcap Red(f) \in Fix(f) \subseteq Red(f)$

(ii) $\text{gfp}(f) = \bigsqcup Fix(f) = \bigsqcup Ext(f) \in Fix(f) \subseteq Ext(f)$

The theorem follows from this, but the proof is non-constructive.
The fixed point theorem

Assume that $L$ has finite height: All totally ordered subsets of $L$ are finite.

The proof proceeds in two parts:

(i) There is a least fixed point.
(ii) Define a least upper bound in $\text{Fix}(f)$. 
The fixed point theorem

(i) Construct the least fixed point.

We always have \( \bot \in Ext(f) \).
The fixed point theorem

(i) Construct the least fixed point.

We always have $\bot \in Ext(f)$. It follows that $f^n(\bot) \sqsubseteq f^{n+1}(\bot)$. 
Abstract Interpretation
Monotone functions on complete lattices

The fixed point theorem

(i) Construct the least fixed point.

We always have $\bot \in Ext(f)$. It follows that $f^n(\bot) \sqsubseteq f^{n+1}(\bot)$. Since $\{f^n(\bot) \mid n \in \mathbb{N}\}$ is a totally ordered subset of $L$, it must be finite. Thus there is an $m \in \mathbb{N}$ with $f^m(\bot) = f^{m+1}(\bot)$.

$\Rightarrow f^m(\bot) \in Fix(f)$

$\Rightarrow Fix(f) \neq \emptyset$
(i) Construct the least fixed point.

Let \( e \in Fix(f) \). Since \( f \) is monotone:

\[
\bot \sqsubseteq e \Rightarrow f^m(\bot) \sqsubseteq f^m(e) = e
\]

\( \Rightarrow f^m(\bot) \) is the least fixed point of \( f \) (\( \text{lfp}(f) \)).
The fixed point theorem

(ii) Define a least upper bound in $\text{Fix}(f)$.

Let $F \subseteq \text{Fix}(f)$. 
The fixed point theorem

\[(ii)\] Define a least upper bound in \(\text{Fix}(f)\).

Let \(F \subseteq \text{Fix}(f)\).
Using \(\forall l \in F, l \sqsubseteq (\bigsqcup F)\) and the monotonicity of \(f\):

\[
\forall l \in F, l = f(l) \sqsubseteq f(\bigsqcup F)
\]

\(\ldots f(\bigsqcup F)\) is an upper bound on \(F\).
The fixed point theorem

(ii) Define a least upper bound in $Fix(f)$.

Let $F \subseteq Fix(f)$.
Using $\forall l \in F, l \sqsubseteq (\bigsqcup F)$ and the monotonicity of $f$:

$$\forall l \in F, l = f(l) \sqsubseteq f(\bigsqcup F)$$

$\ldots f(\bigsqcup F)$ is an upper bound on $F$.

Since $\bigsqcup F$ is the least upper bound on $F$, $\bigsqcup F \sqsubseteq f(\bigsqcup F)$ and thus:

$$\bigsqcup F \in Ext(f)$$
(ii) Define a least upper bound in $\text{Fix}(f)$.

As before, there is an $m \in \mathbb{N}$ for which $f^m(\bigcup F) \in \text{Fix}(f)$. This is the least fixed point of $f$ greater than $\bigcup F$. 
The fixed point theorem

By duality:

(i) There is a greatest fixed point $\text{gfp}(f)$.
(ii) Define a greatest lower bound in $\text{Fix}(f)$.

$\Rightarrow \text{Fix}(f)$ is a non-empty complete lattice.
Further properties

Hierarchy

\[ f^n(\perp) \sqsubseteq \bigsqcup_n f^n(\perp) \sqsubseteq \text{lfp}(f) \sqsubseteq \text{gfp}(f) \sqsubseteq \bigsqcap_n f^n(\top) \sqsubseteq f^n(\top) \]

In the general case, all inequalities can be strict.
The lattice of intervals
The lattice of intervals

Consider the set

\[ \text{Interval} = \{ \bot \} \cup \{ [z_1, z_2] \mid z_1 \leq z_2, z_1 \in \mathbb{Z} \cup \{-\infty\}, z_2 \in \mathbb{Z} \cup \{\infty\} \} \]

With

\[ l_1 \sqsubseteq l_2 :\Leftrightarrow (z \in l_1 \Rightarrow z \in l_2) \]

\[ l_1 \sqcup l_2 :\Leftrightarrow \left[ \min(LB(l_1), LB(l_2)), \max(UB(l_1), UB(l_2)) \right] \]

This lattice has infinite height:

\[ [1, 1] \sqsubseteq [1, 2] \sqsubseteq [1, 3] \sqsubseteq \ldots \]
The lattice of intervals

Consider the program:

\[(V_0) \ x = 0\]
\[(V_1) \ \text{while}(x < 100) :\]
\[(V_2) \ x = x + 1\]

...and an analysis for \textit{Interval} for \(x\) on the program:

\[V_0 = [0, 0]\]
\[V_1 = (V_0 \sqcup V_2) \cap [-\infty, 99]\]
\[V_2 = V_1 + [1, 1]\]
The lattice of intervals

\[(V_0) \quad x = 0 \quad V_0 = [0, 0]\]
\[(V_1) \quad \text{while}(x < 100) : \quad V_1 = (V_0 \sqcup V_2) \cap [-\infty, 99]\]
\[(V_2) \quad x = x + 1 \quad V_2 = V_1 + [1, 1]\]

To obtain a solution, iterate starting from \(\bot\):

<table>
<thead>
<tr>
<th>Iteration</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>\ldots</th>
<th>201</th>
</tr>
</thead>
<tbody>
<tr>
<td>(V_0)</td>
<td>[0, 0]</td>
<td>[0, 0]</td>
<td>[0, 0]</td>
<td>[0, 0]</td>
<td>[0, 0]</td>
<td>\ldots</td>
<td>[0, 0]</td>
</tr>
<tr>
<td>(V_1)</td>
<td>⊥</td>
<td>[0, 0]</td>
<td>[0, 0]</td>
<td>[0, 1]</td>
<td>[0, 1]</td>
<td>\ldots</td>
<td>[0, 99]</td>
</tr>
<tr>
<td>(V_2)</td>
<td>⊥</td>
<td>⊥</td>
<td>[1, 1]</td>
<td>[1, 1]</td>
<td>[1, 2]</td>
<td>\ldots</td>
<td>[1, 100]</td>
</tr>
</tbody>
</table>

By Tarski’s theorem, this is the most precise solution.
Approximation of Fixed Points

Computing $\text{lfp}(f)$ might be prohibitively expensive, or impossible. Different approaches:

(i) Approximate $\text{gfp}(f)$ by $(f^n(\top))_n$.

(ii) Find a conservative approximation of $\text{lfp}(f)$.

(iii) Use a different lattice.
Approximate $\text{gfp}(f)$

By Tarski’s theorem $\text{lfp}(f) = \bigsqcap Red(f)$. Using the properties of correctness relations:

$$\forall l \in Red(f), v \ R \ lfp(f) \land lfp(f) \sqsubseteq l \Rightarrow v \ R \ l$$

Since every element of $Red(f)$ is safe, a safe approximation of $\text{gfp}(f)$ can be obtained by any element of $(f^n(\top))_n$.
Example: The lattice of intervals

\[(V_0) \ \ x = 0 \quad \quad V_0 = [0, 0]\]
\[(V_1) \ \ \text{while}(x < 100): \quad V_1 = (V_0 \sqcup V_2) \cap [-\infty, 99]\]
\[(V_2) \ \ x = x + 1 \quad V_2 = V_1 + [1, 1]\]

Iterating from \(\top\) yields:

<table>
<thead>
<tr>
<th>Iteration</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>(V_0)</td>
<td>[0, 0]</td>
<td>[0, 0]</td>
<td>[0, 0]</td>
</tr>
<tr>
<td>(V_1)</td>
<td>(\top)</td>
<td>([-\infty, 99])</td>
<td>([-\infty, 99])</td>
</tr>
<tr>
<td>(V_2)</td>
<td>(\top)</td>
<td>(\top)</td>
<td>([-\infty, 100])</td>
</tr>
</tbody>
</table>

Which is clearly much worse than the least solution.
Conservative approximation of $\text{lfp}(f)$

Idea: Since $(f^n(\perp))_n$ might not eventually stabilise, construct a sequence that will.
Upper Bound Operators

Definition

An operator $\tilde{\sqcup}: L \times L \rightarrow L$ on a complete lattice $L$ is an “upper bound operator” if:

$$l_1 \sqsubseteq (l_1 \tilde{\sqcup} l_2) \sqsubseteq l_2$$

An upper bound operator is an upper approximation of the least upper bound: $l_1 \sqcup l_2 \sqsubseteq l_1 \tilde{\sqcup} l_2$. 
Folding

Let \((l_n)_n\) be a sequence in \(L\) and let \(\phi : L \times L \to L\) be an operator on \(L\).

\[
l_\phi^n = \begin{cases} 
    l_n, & \text{if } n = 0 \\
    l_{n-1} \phi l_n, & \text{if } n > 0 
\end{cases}
\]
Folding

Let \((l_n)_n\) be a sequence in \(L\) and let \(\phi : L \times L \to L\) be an operator on \(L\).

\[
  l^\phi_n = \begin{cases} 
    l_n, & \text{if } n = 0 \\
    l^\phi_{n-1} \phi l_n, & \text{if } n > 0
  \end{cases}
\]

If \(\bigsqcup\) is an upper bound operator then \((l^\bigsqcup_n)_n\) is an ascending chain and furthermore \(l^\bigsqcup_n = \bigsqcup\{l_0, l_1, \ldots, l_n\} \subseteq l^\bigsqcup_n\).
Widening Operators

Definition
An upper bound operator $\nabla : L \times L \rightarrow L$ is called a “widening operator” if for all ascending chains $(l_n)_n$ the ascending chain $(l_n^\nabla)_n$ eventually stabilises.
Folding (cont’d)

Let $f : L \rightarrow L$ be a monotone function. Define the sequence $(f^n_{\triangledown})_n$ by:

$$f^n_{\triangledown} = \begin{cases} \bot, & \text{if } n = 0 \\ f^{n-1}_{\triangledown}, & \text{if } n > 0 \land f(f^{n-1}_{\triangledown}) \sqsubseteq f^{n-1}_{\triangledown} \\ f^{n-1}_{\triangledown} \triangledown f(f^{n-1}_{\triangledown}), & \text{otherwise} \end{cases}$$
Widening Operators

Lemma

If $\nabla$ is a widening operator then the ascending chain $(f^n \nabla)_n$ eventually stabilises.

$lfp f^n \nabla$ is a computable upper approximation of $lfp f$. 
Example: Intervals (cont’d)

Define $\nabla : \text{Interval} \times \text{Interval} \rightarrow \text{Interval}$ by

$$[l_1, u_1] \nabla [l_2, u_2] = \begin{cases} [l_1, u_1], & \text{if } [l_2, u_2] \subseteq [l_1, u_1] \\ [-\infty, u_1], & \text{if } l_2 \leq l_1 \land u_2 \leq u_1 \\ [l_1, \infty], & \text{if } l_1 \leq l_2 \land u_1 \leq u_2 \\ [-\infty, \infty], & \text{otherwise} \end{cases}$$

$$\bot \nabla int_2 = int_2$$

It is straightforward to show that $\nabla$ is a widening operator.
Example: Intervals (cont’d)

Applying $\nabla$ on an ascending chain $(\bot, [l, u], [l', u'], \ldots)$ yields:

```
[\infty, \infty]  
\downarrow
[\infty, u] \quad [l, \infty]  
\uparrow \quad \uparrow
[l, u]  
\downarrow
\bot
```
Example: Intervals (cont’d)

\[ V_0 = [0, 0] \]
\[ V_1 = V_1 \nabla ((V_0 \sqcup V_2) \cap [-\infty, 99]) \]
\[ V_2 = V_1 + [1, 1] \]

Using widening on \( V_1 \) ensures convergence:

<table>
<thead>
<tr>
<th>Iteration</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>( V_0 )</td>
<td>[0, 0]</td>
<td>[0, 0]</td>
<td>[0, 0]</td>
<td>[0, 0]</td>
<td>[0, 0]</td>
</tr>
<tr>
<td>( V_1 )</td>
<td>⊥</td>
<td>[0, 0]</td>
<td>[0, 0]</td>
<td>[0, \infty]</td>
<td>[0, \infty]</td>
</tr>
</tbody>
</table>
Narrowing Operators

The approximation could be improved by iterating $f$. What about convergence?
Abstract Interpretation
Approximating fixed points

Narrowing Operators

Definition
An operator $\Delta : L \times L \to L$ is a “narrowing operator” if:

(i) $l_2 \sqsubseteq l_1 \Rightarrow l_2 \sqsubseteq (l_1 \Delta l_2) \sqsubseteq l_1$ for all $l_1, l_2 \in L$.

(ii) for all descending chains $(l_n)_n$, the sequence $(l_n^\Delta)_n$ eventually stabilises.
Narrowing Operators

Starting with $\text{lfp}_{\nabla}(f) = f^m_{\nabla}$ construct:

$$f^n_{\Delta} = \begin{cases} f^m_{\nabla}, & \text{if } n = 0 \\ f^{n-1}_{\Delta} \Delta f(f^{n-1}_{\Delta}), & \text{if } n > 0 \end{cases}$$
Abstract Interpretation
Approximating fixed points

Improving the approximation

Lemma

\((f^n_\Delta)_n\) is a descending chain in \(Red(f)\) and:

\[ f^n_\Delta \sqsubseteq f^n(f^m_\nabla) \sqsubseteq \text{lfp}(f) \]
Improving the approximation

Lemma

\((f^n_\Delta)_n\) is a descending chain in \(\text{Red}(f)\) and:

\[ f^n_\Delta \supseteq f^n(f^m) \supseteq \text{lfp}(f) \]

Lemma

If \(\Delta\) is a narrowing operator then the descending chain \((f^n_\Delta)_n\) eventually stabilises.
Example: Intervals (cont’d)

There are only two kinds of infinite descending chains on Interval:

\[ [z_1, \infty] \supseteq [z_2, \infty] \supseteq \ldots \]
\[ [-\infty, -z_1] \supseteq [-\infty, -z_2] \supseteq \ldots \]

Where \( z_1 < z_2 < \ldots \)

Define \( \Delta : L \times L \rightarrow L \) as:

\[ int_1 \Delta int_2 = \begin{cases} 
  int_2, & \text{if } int_2 \text{ is finite} \\
  int_1, & \text{otherwise}
\end{cases} \]
Example: Intervals (cont’d)

\[ V_0 = [0, 0] \]
\[ V_1 = V_1 \triangle ((V_0 \sqcup V_2) \sqcap [-\infty, 99]) \]
\[ V_2 = V_1 + [1, 1] \]

Continuing from before, using narrowing on \( V_1 \):

<table>
<thead>
<tr>
<th>Iteration</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>( V_0 )</td>
<td>[0, 0]</td>
<td>[0, 0]</td>
<td>[0, 0]</td>
</tr>
<tr>
<td>( V_1 )</td>
<td>[0, \infty]</td>
<td>[0, 99]</td>
<td>[0, 99]</td>
</tr>
<tr>
<td>( V_2 )</td>
<td>[1, \infty]</td>
<td>[1, \infty]</td>
<td>[1, 100]</td>
</tr>
</tbody>
</table>
The third option

Or use a different lattice.
Galois connection

\[ \alpha(\gamma(m)) \Rightarrow \gamma(\alpha(l)) \]
Galois connections

Definition

\((L, \alpha, \gamma, M)\) is a Galois connection between the complete lattices \((L, \sqsubseteq)\) and \((M, \sqsubseteq)\) iff \(\alpha : L \rightarrow M\) and \(\gamma : M \rightarrow L\) are monotone functions that satisfy:

\[
\gamma \circ \alpha \supseteq \lambda l. l \\
\alpha \circ \gamma \sqsubseteq \lambda m. m
\]
Example: Intervals (again)

Construct a Galois connection between $\mathcal{P}(\mathbb{Z})$ and $\text{Interval}$:

\[
\begin{align*}
\alpha(Z) &= [\inf Z, \sup Z] \\
\gamma(I) &= \{z \in \mathbb{Z} \mid LB(I) \leq z \leq UB(I)\}
\end{align*}
\]

Note that this implies $\bot = [\infty, -\infty]$
Adjunction

**Definition**

\((L, \alpha, \gamma, M)\) is an *adjunction* between the complete lattices \((L, \sqsubseteq)\) and \((M, \sqsubseteq)\) iff \(\alpha : L \to M\) and \(\gamma : M \to L\) are total functions that satisfy:

\[ \alpha(l) \sqsubseteq m \iff l \sqsubseteq \gamma(m) \]
Equivalence

\((L, \alpha, \gamma, M)\) is an adjunction iff it is a Galois connection.
Proof

Assume that \((L, \alpha, \gamma, M)\) is a Galois connection.

\[\alpha(l) \sqsubseteq m \Rightarrow \gamma(\alpha(l)) \sqsubseteq \gamma(m) \Rightarrow l \sqsubseteq \gamma(\alpha(l)) \sqsubseteq \gamma(m)\]

Where the last step follows from \(\gamma \circ \alpha \sqsubseteq \lambda l.l\).

The proof of \(l \sqsubseteq \gamma(m) \Rightarrow \alpha(l) \sqsubseteq m\) is analogous.
Assume that \((L, \alpha, \gamma, M)\) is an adjunction.

\[
\alpha(l) \sqsubseteq \alpha(l) \Rightarrow l \sqsubseteq \gamma(\alpha(l))
\]

The proof for \(\alpha \circ \gamma \sqsubseteq \lambda m.m\) is analogous.

To show that \(\alpha, \gamma\) are monotone, let \(l_1 \sqsubseteq l_2\). From \(\gamma \circ \alpha \sqsubseteq \lambda l.l\) it follows that

\[
l_1 \sqsubseteq l_2 \sqsubseteq \gamma(\alpha(l_2)) \Rightarrow \alpha(l_1) \sqsubseteq \alpha(l_2)
\]

The treatment of \(\gamma\) is again, analogous.
Galois connections defined by extraction functions

Let $\beta : V \rightarrow L$ be a representation function. This can be used to define a Galois connection $(\mathcal{P}(V), \alpha, \gamma, L)$, by:

$$\alpha(V') = \bigcup \{ \beta(v) | v \in V' \}$$

$$\gamma(l) = \{ v \in V | \beta(v) \subseteq l \}$$
Galois connections defined by extraction functions

Let $\beta : V \rightarrow L$ be a representation function. This can be used to define a Galois connection $(\mathcal{P}(V), \alpha, \gamma, L)$, by:

$$\alpha(V') = \bigsqcup \{ \beta(v) | v \in V' \}$$
$$\gamma(l) = \{ v \in V | \beta(v) \sqsubseteq l \}$$

This is an adjunction, since:

$$\alpha(V') \sqsubseteq l \iff \bigsqcup \{ \beta(v) | v \in V' \} \sqsubseteq l$$
$$\iff \forall v \in V', \beta(v) \sqsubseteq l$$
$$\iff V' \subseteq \gamma(l)$$
Extraction functions

If $L = \mathcal{P}(D)$ for some set $D$ and $\eta : V \rightarrow D$ is an “extraction function” mapping values to their descriptions in $D$, the representation function $\beta_\eta : V \rightarrow \mathcal{P}(D)$ can be defined by $\beta(v) = \{\eta(v)\}$. 
If $L = \mathcal{P}(D)$ for some set $D$ and $\eta : V \rightarrow D$ is an “extraction function” mapping values to their descriptions in $D$, the representation function $\beta_\eta : V \rightarrow \mathcal{P}(D)$ can be defined by $\beta(v) = \{\eta(v)\}$.

By the same construction as before, it is possible to define a Galois connection $(\mathcal{P}(V), \alpha_\eta, \gamma_\eta, \mathcal{P}(D))$ by:

$$\alpha_\eta(V') = \bigcup \{\beta_\eta(v) \mid v \in V'\} = \{\eta(v) \mid v \in V'\}$$
$$\gamma_\eta(D') = \{v \in V \mid \beta_\eta(v) \subseteq D'\} = \{v \mid \eta(v) \in D'\}$$
A Galois connection for $\mathcal{P}(\mathbb{Z})$ to $\mathcal{P}(\text{Sign})$ can be derived by taking $\text{sgn}$ to be the extraction function. This leads to a Galois connection $(\mathcal{P}(\mathbb{Z}), \alpha_{\text{sgn}}, \gamma_{\text{sgn}}, \mathcal{P}(\text{Sign}))$:

$$\alpha_{\text{sgn}}(V) = \{ \text{sgn}(v) \mid v \in V' \}$$

$$\gamma_{\text{sgn}}(D) = \{ v \mid \text{sgn}(v) \in D' \}$$
Properties of Galois connections

**Lemma**

Let \((L, \alpha, \gamma, M)\) be a Galois connection.

(i) \(\alpha\) uniquely determines \(\gamma\) by \(\gamma(m) = \bigcup \{l \mid \alpha(l) \subseteq m\}\) and

\(\gamma\) uniquely determines \(\alpha\) by \(\alpha(l) = \bigcap \{m \mid l \subseteq \gamma(m)\}\)

(ii) \(\alpha\) is completely additive and \(\gamma\) is completely multiplicative.
Proof

(i) $\alpha$ uniquely determines $\gamma$

Since a Galois connection is an adjunction:

$$\gamma(m) = \bigsqcup \{ l \mid l \subseteq \gamma(m) \} = \bigsqcup \{ l \mid \alpha(l) \subseteq m \}$$

The same argument holds for $\alpha$, which shows (i).
Proof (cont’d)

(ii) \( \alpha \) is completely additive.

A function \( \alpha \) on a complete lattice is **completely additive**, if for each \( L' \subseteq L \):

\[
\alpha(\bigsqcup L') = \bigsqcup \{ \alpha(l) | l \in L' \}
\]

This is equivalent to:

\[
\alpha(\bigsqcup L') \sqsubseteq m \iff \bigsqcup \{ \alpha(l) | l \in L' \} \sqsubseteq m
\]
Proof (cont’d)

(ii) \( \alpha \) is completely additive.

For an adjunction, this can be verified directly:

\[
\alpha\left(\bigsqcup L'\right) \subseteq m \iff \bigsqcup L' \subseteq \gamma(m) \\
\iff \forall l \in L', l \subseteq \gamma(m) \\
\iff \forall l \in L', \alpha(l) \subseteq m \\
\iff \bigsqcup\{\alpha(l) | l \in L'\} \subseteq m
\]

The proof for \( \gamma(\bigsqcap M') = \bigsqcap\{\gamma(m) | m \in M'\} \) is analogous.
Properties of Galois connections

Corollary
If $\alpha : L \to M$ is completely additive, then there exists $\gamma : M \to L$, so that $(L, \alpha, \gamma, M)$ is a Galois connection.
If $\gamma : M \to L$ is completely multiplicative, then there exists $\alpha : L \to M$, so that $(L, \alpha, \gamma, M)$ is a Galois connection.
Proof

Proof for $\alpha$ ($\gamma$ is analogous): Define $\gamma : M \rightarrow L$ as:

$$\gamma(m) = \bigsqcup \{ l' \mid \alpha(l') \sqsubseteq m \}$$
Proof

Proof for $\alpha$ ($\gamma$ is analogous): Define $\gamma : M \rightarrow L$ as:

$$\gamma(m) = \bigsqcup \{ l' \mid \alpha(l') \sqsubseteq m \}$$

Show that $(M, \alpha, \gamma, L)$ is an adjunction.

$$\alpha(l) \sqsubseteq m \Rightarrow l \in \{ l' \mid \alpha(l') \sqsubseteq m \} \Rightarrow l \sqsubseteq \gamma(m)$$
Proof (cont’d)

For the other direction, assume that $l \sqsubseteq \gamma(m)$.

$$\alpha(l) \sqsubseteq \alpha(\gamma(m))$$

$$= \alpha(\bigsqcup \{ l' \mid \alpha(l') \sqsubseteq m \})$$

$$= \bigsqcup \{ \alpha(l') \mid \alpha(l') \sqsubseteq m \}$$

$$\sqsubseteq m$$
Properties

Lemma

If \((M, \alpha, \gamma, L)\) is a Galois connection, then \(\gamma \circ \alpha \circ \gamma = \gamma\) and \(\alpha \circ \gamma \circ \alpha = \alpha\).
Properties

Lemma
If \((M, \alpha, \gamma, L)\) is a Galois connection, then \(\gamma \circ \alpha \circ \gamma = \gamma\) and \(\alpha \circ \gamma \circ \alpha = \alpha\).

This follows from antisymmetry: Since from

\[
\gamma \circ \alpha \sqsupseteq \lambda l.l \Rightarrow \alpha \circ (\gamma \circ \alpha) \sqsupseteq \alpha
\]
\[
\alpha \circ \gamma \sqsubseteq \lambda l.l \Rightarrow (\alpha \circ \gamma) \circ \alpha \sqsubseteq \alpha
\]

it follows that \(\alpha \circ \gamma \circ \alpha = \alpha\). The other case is analogous.
Example

To construct a Galois connection between \textbf{Interval} and $\mathcal{P}(\text{Sign})$, consider:

\[
\begin{align*}
\gamma(\{-, 0, +\}) &= [-\infty, \infty] & \gamma(\{-, 0\}) &= [-\infty, 0] \\
\gamma(\{-, +\}) &= [-\infty, \infty] & \gamma(\{0, +\}) &= [0, \infty] \\
\gamma(\{-\}) &= [-\infty, -1] & \gamma(\{0\}) &= [0, 0] \\
\gamma(\{+\}) &= [1, \infty] & \gamma(\emptyset) &= \perp
\end{align*}
\]

If $\gamma$ is completely multiplicative, this can be turned into a Galois connection.
Example

\[\gamma(\{-, 0, +\}) = [-\infty, \infty] \quad \gamma(\{-, 0\}) = [-\infty, 0]\]
\[\gamma(\{-, +\}) = [-\infty, \infty] \quad \gamma(\{0, +\}) = [0, \infty]\]
\[\gamma(\{-\}) = [-\infty, -1] \quad \gamma(\{0\}) = [0, 0]\]
\[\gamma(\{+\}) = [1, \infty] \quad \gamma(\emptyset) = \perp\]

\(\gamma\) is not completely multiplicative:

\[\gamma(\{-, 0\} \cap \{-, +\}) = \gamma(\{-\}) = [-\infty, -1]\]
\[\gamma(\{-, 0\}) \cap \gamma(\{-, +\}) = [-\infty, 0] \cap [-\infty, \infty] = [-\infty, 0]\]
Theorem

Let $R \subseteq V \times L$ be a correctness relation and $(L, \alpha, \gamma, M)$ be a Galois connection. Define $v \ S \ m \Leftrightarrow v \ R \ \gamma(m)$. Then $S$ is a correctness relation.
Induced representation function

**Corollary**

If $\beta : V \rightarrow L$ is a representation function generated by $R$, then $\alpha \circ \beta : V \rightarrow M$ is a representation function generated by $S$. 
Conclusion

(i) Abstract definition of semantics, correctness.
(ii) Tarski’s Theorem guarantees existence of best solutions.
(iii) Widening/Narrowing as a general method for approximating fixedpoints.
(iv) Galois connections and their properties (still much work to do).
Proof of correctness relation

The first property follows from the monotonicity of $\gamma$

\[(v \ S \ m_1) \land m_1 \sqsubseteq m_2 \Rightarrow (v \ R \ \gamma(m_1)) \land m_1 \sqsubseteq m_2\]

\[\Rightarrow (v \ R \ \gamma(m_1)) \land \gamma(m_1) \sqsubseteq \gamma(m_2)\]

\[\Rightarrow (v \ R \ \gamma(m_2))\]

\[\Rightarrow v \ S \ m_2\]
Proof (cont’d)

For the second condition, use the complete multiplicativity of $\gamma$:

$$\forall m \in M' \subseteq M, v \ S \ m \Rightarrow \forall m \in M' \subseteq M, v \ R \ \gamma(m)$$
$$\Rightarrow v \ R \left( \bigcap \{\gamma(m) \mid m \in M'\} \right)$$
$$\Rightarrow v \ R \ \gamma\left( \bigcap M' \right)$$
$$\Rightarrow v \ S \left( \bigcap M' \right)$$

This shows, that we need a Galois connection to preserve the accuracy of correctness relations.
Proof of Collorary

**Collorary**

If \( \beta : V \rightarrow L \) is a representation function generated by \( R \), then \( \alpha \circ \beta : V \rightarrow M \) is a representation function generated by \( S \).

Since \( (L, \alpha, \gamma, M) \) is an adjunction:

\[
  v \ S \ m \iff v \ R \ \gamma(m) \\
  \iff \beta(v) \sqsubseteq \gamma(m) \\
  \iff \alpha(\beta(v)) \sqsubseteq m \\
  \iff (\alpha \circ \beta)(v) \sqsubseteq m
\]