Principles of Program Analysis:

Abstract Interpretation

Transparencies based on Chapter 4 of the book: Flemming Nielson, Hanne Riis Nielson and Chris Hankin: Principles of Program Analysis. Springer Verlag 2005.

Correctness Relations

$$R : V \times L \rightarrow \{\textit{true}, \textit{false}\}$$

Idea: $v\ R\ l$ means that the value v is described by the property l.

Correctness criterion: R is preserved under computation:

$$\begin{array}{cccccc}
p & \vdash & v_1 & \boxed{\rightsquigarrow} & v_2 \\
 & & \vdots & & \vdots \\
 & & R & \Rightarrow & R \\
 & & \vdots & & \vdots \\
p & \vdash & l_1 & \boxed{\rhd} & l_2
\end{array}$$

logical relation:

$$(p \vdash \cdot \boxed{\rightsquigarrow} \cdot)\ (R \twoheadrightarrow R)\ (p \vdash \cdot \boxed{\rhd} \cdot)$$

Admissible Correctness Relations

$$v \; R \; l_1 \;\wedge\; l_1 \sqsubseteq l_2 \;\Rightarrow\; v \; R \; l_2$$

$$(\forall l \in L' \subseteq L : v \; R \; l) \;\Rightarrow\; v \; R \; (\bigsqcap L') \qquad (\{l \mid v \; R \; l\} \text{ is a Moore family})$$

Two consequences:

$$v \; R \; \top$$

$$v \; R \; l_1 \;\wedge\; v \; R \; l_2 \;\Rightarrow\; v \; R \; (l_1 \sqcap l_2)$$

Assumption: $(L, \sqsubseteq)$ is a complete lattice.

Representation Functions

$$\beta : V \to L$$

Idea: β maps a value to the *best* property describing it.

Correctness criterion:

$$\begin{array}{ccccc}
p & \vdash & v_1 & \leadsto & v_2 \\
 & & \big\downarrow \beta & \Rightarrow & \big\downarrow \beta \\
 & & \sqsubseteq & & \sqsubseteq \\
p & \vdash & l_1 & \rhd & l_2
\end{array}$$

Equivalence of Correctness Criteria

Given a representation function β we define a correctness relation R_β by

$$v \; R_\beta \; l \quad \text{iff} \quad \beta(v) \sqsubseteq l$$

Given a correctness relation R we define a representation function β_R by

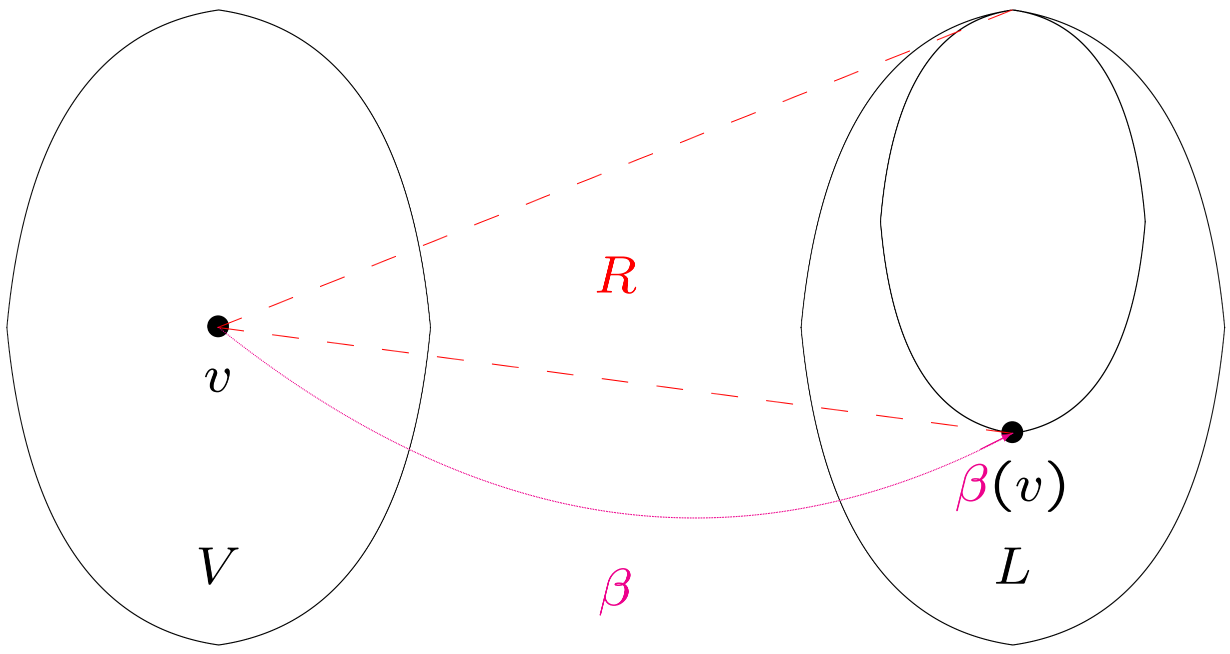
$$\beta_R(v) \;=\; \bigsqcap \{l \mid v \; R \; l\}$$

Lemma:

(i) Given $\beta : V \to L$, then the relation $R_\beta : V \times L \to \{\mathit{true}, \mathit{false}\}$ is an admissible correctness relation such that $\beta_{R_\beta} = \beta$.

(ii) Given an admissible correctness relation $R : V \times L \to \{\mathit{true}, \mathit{false}\}$, then β_R is well-defined and $R_{\beta_R} = R$.

Equivalence of Criteria: R is *generated* by β



Galois connections

$$L \underset{\alpha}{\overset{\gamma}{\rightleftarrows}} M$$

α: *abstraction function*

γ: *concretisation function*

is a Galois connection if and only if

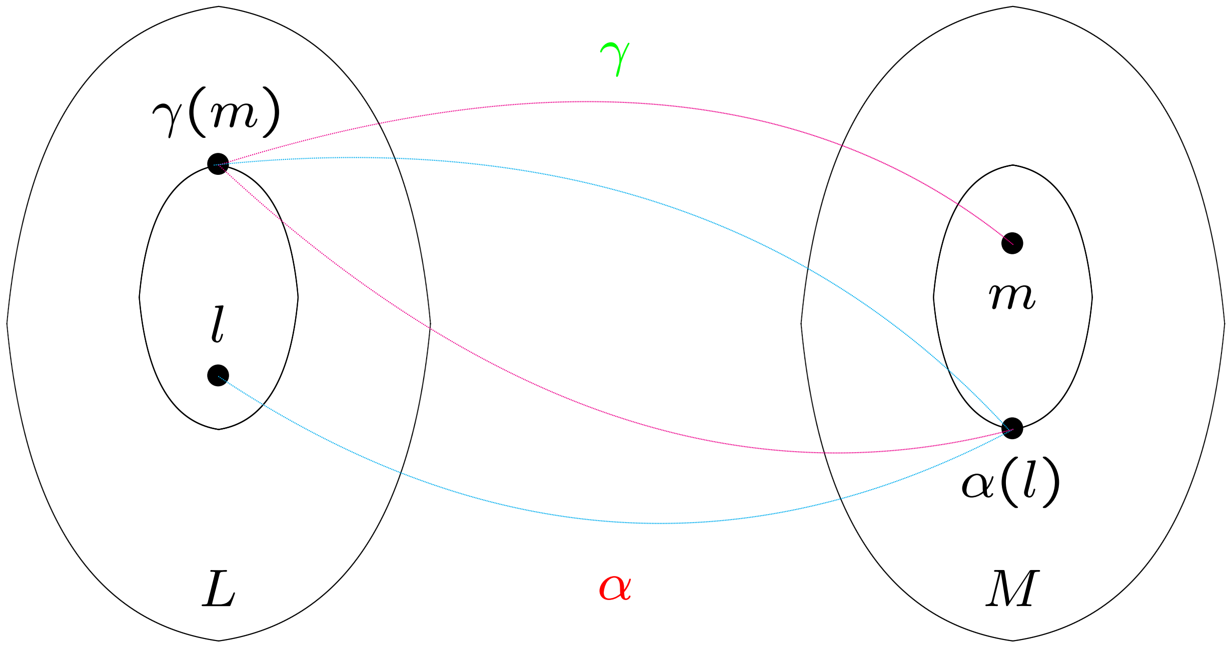
α and γ are monotone functions

that satisfy

$$\gamma \circ \alpha \sqsupseteq \lambda l.l$$

$$\alpha \circ \gamma \sqsubseteq \lambda m.m$$

Galois connections



$\gamma \circ \alpha \sqsupseteq \lambda l.l$ $\qquad$ $\alpha \circ \gamma \sqsubseteq \lambda m.m$

Adjunctions

$$L \underset{\alpha}{\overset{\gamma}{\rightleftarrows}} M$$

is an *adjunction* if and only if

$$\alpha : L \to M \text{ and } \gamma : M \to L \text{ are total functions}$$

that satisfy

$$\alpha(l) \sqsubseteq m \quad \underline{\text{iff}} \quad l \sqsubseteq \gamma(m)$$

for all $l \in L$ and $m \in M$.

Proposition: (α, γ) is an adjunction iff it is a Galois connection.

Galois connections from extraction functions

An *extraction function*

$$\eta : V \to D$$

maps the values of V to their best descriptions in D.

It gives rise to a representation function $\beta_\eta : V \to \mathcal{P}(D)$ (corresponding to $L = (\mathcal{P}(D), \subseteq)$) defined by

$$\beta_\eta(v) = \{\eta(v)\}$$

The associated Galois connection is

$$(\mathcal{P}(V), \alpha_\eta, \gamma_\eta, \mathcal{P}(D))$$

where

$$\begin{array}{rcl c l}
\alpha_\eta(V') & = & \bigcup\{\beta_\eta(v) \mid v \in V'\} & = & \{\eta(v) \mid v \in V'\} \\
\gamma_\eta(D') & = & \{v \in V \mid \beta_\eta(v) \subseteq D'\} & = & \{v \mid \eta(v) \in D'\}
\end{array}$$

Properties of Galois Connections

Lemma: If (L, α, γ, M) is a Galois connection then:

- α uniquely determines γ by $\gamma(m) = \bigsqcup\{l \mid \alpha(l) \sqsubseteq m\}$
- γ uniquely determines α by $\alpha(l) = \bigsqcap\{m \mid l \sqsubseteq \gamma(m)\}$
- α is completely additive and γ is completely multiplicative

In particular $\alpha(\bot) = \bot$ and $\gamma(\top) = \top$.

Lemma:

- If $\alpha : L \to M$ is completely additive then there exists (an upper adjoint) $\gamma : M \to L$ such that (L, α, γ, M) is a Galois connection.
- If $\gamma : M \to L$ is completely multiplicative then there exists (a lower adjoint) $\alpha : L \to M$ such that (L, α, γ, M) is a Galois connection.

Fact: If (L, α, γ, M) is a Galois connection then

- $\alpha \circ \gamma \circ \alpha = \alpha$ and $\gamma \circ \alpha \circ \gamma = \gamma$

The mundane approach: correctness relations

Assume

- $R : V \times L \to \{\textit{true}, \textit{false}\}$ is an admissible correctness relation
- (L, α, γ, M) is a Galois connection

Then $S : V \times M \to \{\textit{true}, \textit{false}\}$ defined by

$$v \; S \; m \quad \underline{\text{iff}} \quad v \; R \; (\gamma(m))$$

is an admissible correctness relation between V and M

$$\begin{array}{cccccc}
p & \vdash & v_1 & \rightsquigarrow & v_2 & \\
 & & \vdots & & \vdots & \\
 & & R & \Rightarrow & R & \\
 & & \vdots & & \vdots & \\
p & \vdash & l_1 & \rhd & l_2 & \\
 & \gamma & \uparrow & & \uparrow & \gamma \\
p & \vdash & m_1 & \rhd & m_2 &
\end{array}$$

The mundane approach: representation functions

Assume

- $R : V \times L \to \{\mathit{true}, \mathit{false}\}$ is *generated by* $\beta : V \to L$
- (L, α, γ, M) is a Galois connection

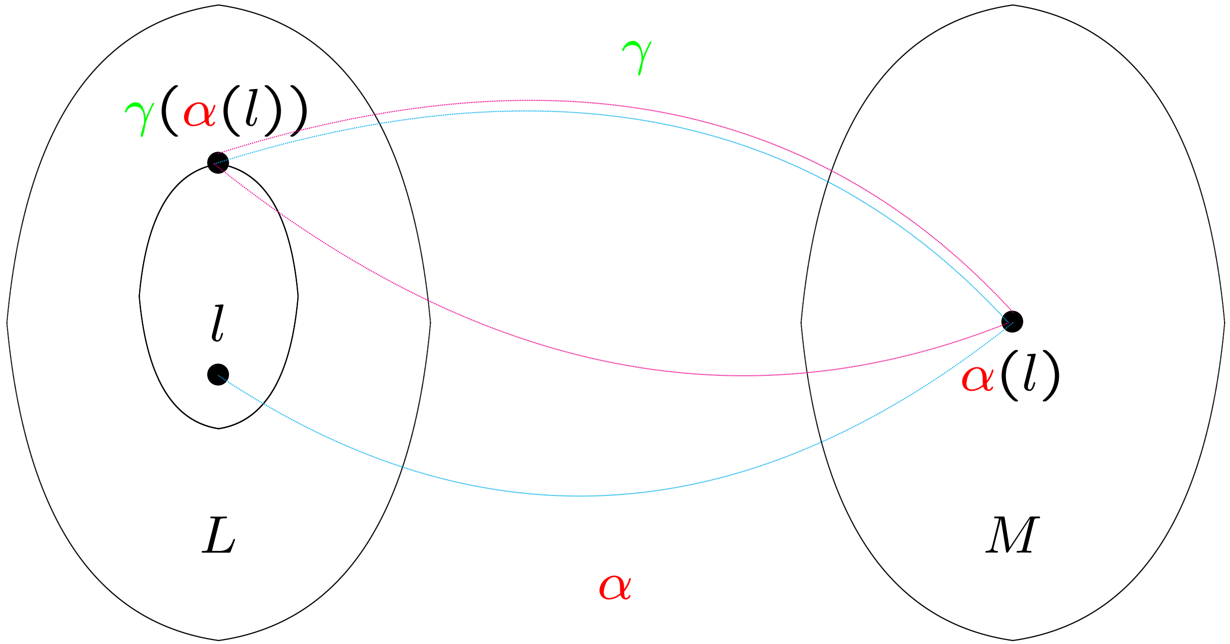
Then $S : V \times M \to \{\mathit{true}, \mathit{false}\}$ defined by

$$v \; S \; m \qquad \underline{\text{iff}} \qquad v \; R \; (\gamma(m))$$

is *generated by* $\alpha \circ \beta : V \to M$

$$\begin{tikzcd}
p & \vdash & v_1 \arrow[d, "\beta"'] & \rightsquigarrow & v_2 \arrow[d, "\beta"] \\
 & & \sqsubseteq & \Rightarrow & \sqsubseteq \\
p & \vdash & l_1 \arrow[d, "\alpha"'] & \rhd & l_2 \arrow[d, "\alpha"] \\
p & \vdash & m_1 & \rhd & m_2
\end{tikzcd}$$

Galois Insertions



Monotone functions satisfying: $\gamma \circ \alpha \sqsupseteq \lambda l.l \qquad \alpha \circ \gamma = \lambda m.m$

Reduction Operators

Given a Galois connection (L, α, γ, M) it is always possible to obtain a Galois insertion by enforcing that the concretisation function γ is injective.

Idea: remove the superfluous elements from M using a *reduction operator*

$$\varsigma : M \to M$$

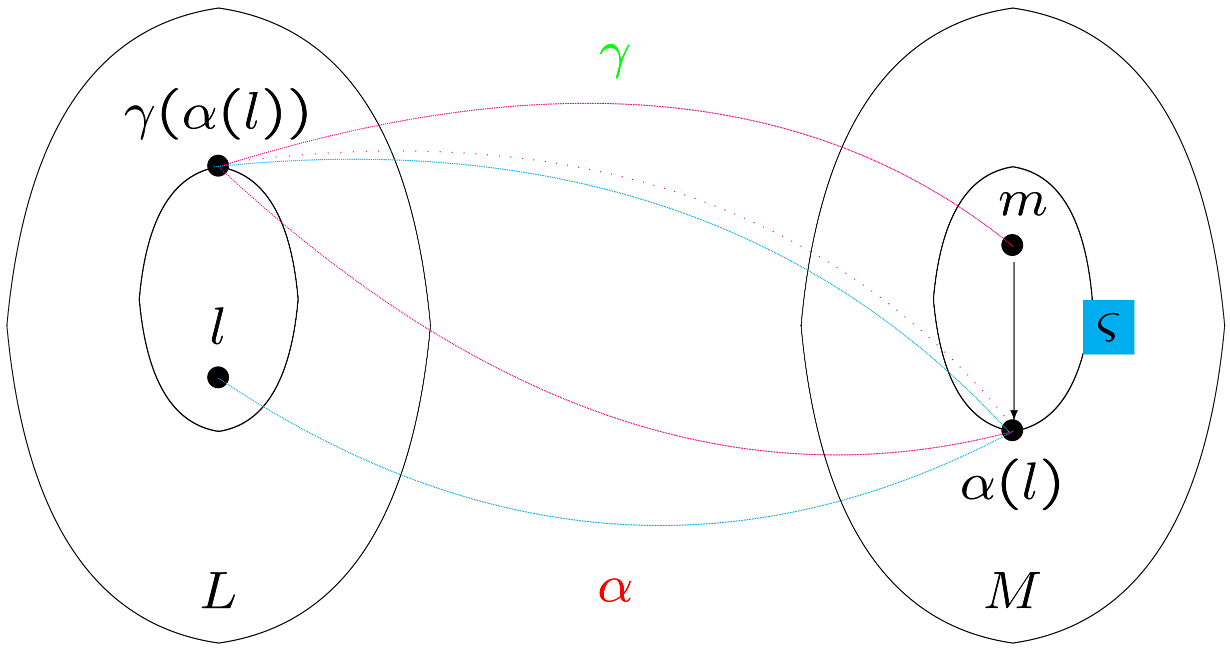
defined from the Galois connection.

Proposition: Let (L, α, γ, M) be a Galois connection and define the reduction operator $\varsigma : M \to M$ by

$$\varsigma(m) = \bigsqcap\{m' \mid \gamma(m) = \gamma(m')\}$$

Then $\varsigma[M] = (\{\varsigma(m) \mid m \in M\}, \sqsubseteq_M)$ is a complete lattice and $(L, \alpha, \gamma, \varsigma[M])$ is a Galois insertion.

The reduction operator $\boxed{\varsigma}$: $M \to M$



Systematic Design of Galois Connections

The "functional composition" (or "sequential composition") of two Galois connections is also a Galois connection:

$$L_0 \underset{\alpha_1}{\overset{\gamma_1}{\leftrightarrows}} L_1 \underset{\alpha_2}{\overset{\gamma_2}{\leftrightarrows}} L_2 \underset{\alpha_3}{\overset{\gamma_3}{\leftrightarrows}} \cdots \underset{\alpha_k}{\overset{\gamma_k}{\leftrightarrows}} L_k$$

A catalogue of techniques for combining Galois connections:

- independent attribute method
- direct product
- reduced product
- total function space
- relational method
- direct tensor product
- reduced tensor product
- monotone function space

Running Example: Array Bound Analysis

Approximation of the difference in magnitude between two numbers (typically the index and the bound):

- a Galois connection for approximating pairs (z_1, z_2) of integers by their difference $|z_1| - |z_2|$
- a Galois connection for approximating integers using a finite lattice $\{\texttt{<-1}, \texttt{-1}, \texttt{0}, \texttt{+1}, \texttt{>+1}\}$
- a Galois connection for their functional composition

Example: Difference in Magnitude

$$(\mathcal{P}(\mathbf{Z} \times \mathbf{Z}), \alpha_{\mathsf{diff}}, \gamma_{\mathsf{diff}}, \mathcal{P}(\mathbf{Z}))$$

where the extraction function $\mathsf{diff} : \mathbf{Z} \times \mathbf{Z} \to \mathbf{Z}$ calculates the difference in magnitude:

$$\mathsf{diff}(z_1, z_2) = |z_1| - |z_2|$$

The abstraction and concretisation functions are

$$\begin{aligned} \alpha_{\mathsf{diff}}(ZZ) &= \{|z_1| - |z_2| \mid (z_1, z_2) \in ZZ\} \\ \gamma_{\mathsf{diff}}(Z) &= \{(z_1, z_2) \mid |z_1| - |z_2| \in Z\} \end{aligned}$$

for $ZZ \subseteq \mathbf{Z} \times \mathbf{Z}$ and $Z \subseteq \mathbf{Z}$.

Example: Finite Approximation

$$(\mathcal{P}(\mathbf{Z}), \alpha_{\mathsf{range}}, \gamma_{\mathsf{range}}, \mathcal{P}(\mathbf{Range}))$$

where $\mathbf{Range} = \{\texttt{<-1}, \texttt{-1}, \texttt{0}, \texttt{+1}, \texttt{>+1}\}$
and the extraction function $\mathsf{range} : \mathbf{Z} \to \mathbf{Range}$ is

$$\mathsf{range}(z) = \begin{cases} \texttt{<-1} & \text{if } z < -1 \\ \texttt{-1} & \text{if } z = -1 \\ \texttt{0} & \text{if } z = 0 \\ \texttt{+1} & \text{if } z = 1 \\ \texttt{>+1} & \text{if } z > 1 \end{cases}$$

The abstraction and concretisation functions are

$$\begin{array}{rcl} \alpha_{\mathsf{range}}(Z) & = & \{\mathsf{range}(z) \mid z \in Z\} \\ \gamma_{\mathsf{range}}(R) & = & \{z \mid \mathsf{range}(z) \in R\} \end{array}$$

for $Z \subseteq \mathbf{Z}$ and $R \subseteq \mathbf{Range}$.

Example: Functional Composition

$$(\mathcal{P}(\mathbf{Z} \times \mathbf{Z}), \alpha_{\mathsf{R}}, \gamma_{\mathsf{R}}, \mathcal{P}(\textbf{Range}))$$

where

$$\begin{aligned} \alpha_{\mathsf{R}} &= \alpha_{\mathsf{range}} \circ \alpha_{\mathsf{diff}} \\ \gamma_{\mathsf{R}} &= \gamma_{\mathsf{diff}} \circ \gamma_{\mathsf{range}} \end{aligned}$$

The explicit formulae for the abstraction and concretisation functions

$$\begin{aligned} \alpha_{\mathsf{R}}(ZZ) &= \{\mathsf{range}(|z_1| - |z_2|) \mid (z_1, z_2) \in ZZ\} \\ \gamma_{\mathsf{R}}(R) &= \{(z_1, z_2) \mid \mathsf{range}(|z_1| - |z_2|) \in R\} \end{aligned}$$

correspond to the extraction function $\mathsf{range} \circ \mathsf{diff}$.

Approximation of Pairs

Independent Attribute Method

Let $(L_1, \alpha_1, \gamma_1, M_1)$ and $(L_2, \alpha_2, \gamma_2, M_2)$ be Galois connections.

The *independent attribute method* gives a Galois connection

$$(L_1 \times L_2, \alpha, \gamma, M_1 \times M_2)$$

where

$$\begin{array}{rcl} \alpha(l_1, l_2) & = & (\alpha_1(l_1), \alpha_2(l_2)) \\ \gamma(m_1, m_2) & = & (\gamma_1(m_1), \gamma_2(m_2)) \end{array}$$

Example: Detection of Signs Analysis

Given

$$(\mathcal{P}(\mathbf{Z}), \alpha_{\mathsf{sign}}, \gamma_{\mathsf{sign}}, \mathcal{P}(\mathbf{Sign}))$$

using the extraction function sign.

The independent attribute method gives

$$(\mathcal{P}(\mathbf{Z}) \times \mathcal{P}(\mathbf{Z}), \alpha_{\mathsf{SS}}, \gamma_{\mathsf{SS}}, \mathcal{P}(\mathbf{Sign}) \times \mathcal{P}(\mathbf{Sign}))$$

where

$$\begin{aligned}
\alpha_{\mathsf{SS}}(Z_1, Z_2) &= (\{\mathsf{sign}(z) \mid z \in Z_1\}, \{\mathsf{sign}(z) \mid z \in Z_2\}) \\
\gamma_{\mathsf{SS}}(S_1, S_2) &= (\{z \mid \mathsf{sign}(z) \in S_1\}, \{z \mid \mathsf{sign}(z) \in S_2\})
\end{aligned}$$

Motivating the Relational Method

The independent attribute method often leads to imprecision!

Semantics: The expression `(x,-x)` may have a value in

$$\{(z, -z) \mid z \in \mathbf{Z}\}$$

Analysis: When we use $\mathcal{P}(\mathbf{Z}) \times \mathcal{P}(\mathbf{Z})$ to represent sets of pairs of integers we cannot do better than representing $\{(z, -z) \mid z \in \mathbf{Z}\}$ by

$$(\mathbf{Z}, \mathbf{Z})$$

Hence the best property describing it will be

$$\alpha_{\mathsf{SS}}(\mathbf{Z}, \mathbf{Z}) = (\{\text{-}, 0, \text{+}\}, \{\text{-}, 0, \text{+}\})$$

Relational Method

Let $(\mathcal{P}(V_1), \alpha_1, \gamma_1, \mathcal{P}(D_1))$ and $(\mathcal{P}(V_2), \alpha_2, \gamma_2, \mathcal{P}(D_2))$ be Galois connections.

The *relational method* will give rise to the Galois connection

$$(\mathcal{P}(V_1 \times V_2), \alpha, \gamma, \mathcal{P}(D_1 \times D_2))$$

where

$$\begin{array}{rcl} \alpha(VV) & = & \bigcup\{\alpha_1(\{v_1\}) \times \alpha_2(\{v_2\}) \mid (v_1, v_2) \in VV\} \\ \gamma(DD) & = & \{(v_1, v_2) \mid \alpha_1(\{v_1\}) \times \alpha_2(\{v_2\}) \subseteq DD\} \end{array}$$

Generalisation to arbitrary complete lattices: use *tensor products*.

Relational Method from Extraction Functions

Assume that the Galois connections $(\mathcal{P}(V_i), \alpha_i, \gamma_i, \mathcal{P}(D_i))$ are given by *extraction functions* $\eta_i : V_i \to D_i$ as in

$$\begin{aligned} \alpha_i(V_i') &= \{\eta_i(v_i) \mid v_i \in V_i'\} \\ \gamma_i(D_i') &= \{v_i \mid \eta_i(v_i) \in D_i'\} \end{aligned}$$

Then the Galois connection $(\mathcal{P}(V_1 \times V_2), \alpha, \gamma, \mathcal{P}(D_1 \times D_2))$ has

$$\begin{aligned} \alpha(\mathit{VV}) &= \{(\eta_1(v_1), \eta_2(v_2)) \mid (v_1, v_2) \in \mathit{VV}\} \\ \gamma(\mathit{DD}) &= \{(v_1, v_2) \mid (\eta_1(v_1), \eta_2(v_2)) \in \mathit{DD}\} \end{aligned}$$

which also can be obtained directly from the extraction function $\eta : V_1 \times V_2 \to D_1 \times D_2$ defined by

$$\eta(v_1, v_2) = (\eta_1(v_1), \eta_2(v_2))$$

Example: Detection of Signs Analysis

Using the relational method we get a Galois connection

$$(\mathcal{P}(\mathbf{Z} \times \mathbf{Z}), \alpha_{\mathsf{SS'}}, \gamma_{\mathsf{SS'}}, \mathcal{P}(\mathbf{Sign} \times \mathbf{Sign}))$$

where

$$\begin{array}{rcl} \alpha_{\mathsf{SS'}}(ZZ) & = & \{(\mathsf{sign}(z_1), \mathsf{sign}(z_2)) \mid (z_1, z_2) \in ZZ\} \\ \gamma_{\mathsf{SS'}}(SS) & = & \{(z_1, z_2) \mid (\mathsf{sign}(z_1), \mathsf{sign}(z_2)) \in SS\} \end{array}$$

corresponding to an extraction function $\mathsf{twosigns} : \mathbf{Z} \times \mathbf{Z} \to \mathbf{Sign} \times \mathbf{Sign}$ defined by

$$\mathsf{twosigns}(z_1, z_2) = (\mathsf{sign}(z_1), \mathsf{sign}(z_2))$$

Advantages of the Relational Method

Semantics: The expression `(x,-x)` may have a value in

$$\{(z,-z) \mid z \in \mathbf{Z}\}$$

In the present setting $\{(z,-z) \mid z \in \mathbf{Z}\}$ is an element of $\mathcal{P}(\mathbf{Z} \times \mathbf{Z})$.

Analysis: The best “relational” property describing it is

$$\alpha_{\mathsf{SS}'}(\{(z,-z) \mid z \in \mathbf{Z}\}) = \{(\text{-},\text{+}),(0,0),(\text{+},\text{-})\}$$

whereas the best “independent attribute” property was

$$\alpha_{\mathsf{SS}}(\mathbf{Z},\mathbf{Z}) = (\{\text{-},0,\text{+}\},\{\text{-},0,\text{+}\})$$

Function Spaces

Total Function Space

Let (L, α, γ, M) be a Galois connection and let S be a set.

The Galois connection for the *total function space*

$$(S \to L, \alpha', \gamma', S \to M)$$

is defined by

$$\alpha'(f) = \alpha \circ f \qquad\qquad \gamma'(g) = \gamma \circ g$$

Do we need to assume that S is non-empty?

Monotone Function Space

Let $(L_1, \alpha_1, \gamma_1, M_1)$ and $(L_2, \alpha_2, \gamma_2, M_2)$ be Galois connections.

The Galois connection for the *monotone function space*

$$(L_1 \to L_2, \alpha, \gamma, M_1 \to M_2)$$

is defined by

$$\alpha(f) = \alpha_2 \circ f \circ \gamma_1 \qquad\qquad \gamma(g) = \gamma_2 \circ g \circ \alpha_1$$

$$\begin{tikzcd}
L_1 \arrow[r, "f"] & L_2 \arrow[d, "\alpha_2"] \\
M_1 \arrow[u, "\gamma_1"] \arrow[r, "\alpha(f)"'] & M_2
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
L_1 \arrow[r, "\gamma(g)"] \arrow[d, "\alpha_1"'] & L_2 \\
M_1 \arrow[r, "g"'] & M_2 \arrow[u, "\gamma_2"']
\end{tikzcd}$$

Performing Analyses Simultaneously

Direct Product

Let $(L, \alpha_1, \gamma_1, M_1)$ and $(L, \alpha_2, \gamma_2, M_2)$ be Galois connections.

The *direct product* is the Galois connection

$$(L, \alpha, \gamma, M_1 \times M_2)$$

defined by

$$\begin{aligned}
\alpha(l) &= (\alpha_1(l), \alpha_2(l)) \\
\gamma(m_1, m_2) &= \gamma_1(m_1) \sqcap \gamma_2(m_2)
\end{aligned}$$

Example:

Combining the detection of signs analysis for pairs of integers with the analysis of difference in magnitude.

We get the Galois connection

$$(\mathcal{P}(\mathbf{Z} \times \mathbf{Z}), \alpha_{\mathsf{SSR}}, \gamma_{\mathsf{SSR}}, \mathcal{P}(\mathbf{Sign} \times \mathbf{Sign}) \times \mathcal{P}(\mathbf{Range}))$$

where

$$\begin{array}{rcl} \alpha_{\mathsf{SSR}}(ZZ) & = & (\{(\mathsf{sign}(z_1), \mathsf{sign}(z_2)) \mid (z_1, z_2) \in ZZ\}, \\ & & \{\mathsf{range}(|z_1| - |z_2|) \mid (z_1, z_2) \in ZZ\}) \\ \gamma_{\mathsf{SSR}}(SS, R) & = & \{(z_1, z_2) \mid (\mathsf{sign}(z_1), \mathsf{sign}(z_2)) \in SS\} \\ & \cap & \{(z_1, z_2) \mid \mathsf{range}(|z_1| - |z_2|) \in R\} \end{array}$$

Motivating the Direct Tensor Product

The expression `(x, 3*x)` may have a value in

$$\{(z, 3 * z) \mid z \in \mathbf{Z}\}$$

which is described by

$$\alpha_{\mathsf{SSR}}(\{(z, 3 * z) \mid z \in \mathbf{Z}\}) = (\{(\text{-}, \text{-}), (0, 0), (\text{+}, \text{+})\}, \{0, \text{<-1}\})$$

But

- any pair described by $(0, 0)$ will have a difference in magnitude described by 0
- any pair described by $(\text{-}, \text{-})$ or $(\text{+}, \text{+})$ will have a difference in magnitude described by <-1

and the analysis cannot express this.

Direct Tensor Product

Let $(\mathcal{P}(V), \alpha_1, \gamma_1, \mathcal{P}(D_1))$ and $(\mathcal{P}(V), \alpha_2, \gamma_2, \mathcal{P}(D_2))$ be Galois connections.

The *direct tensor product* is the Galois connection

$$(\mathcal{P}(V), \alpha, \gamma, \mathcal{P}(D_1 \times D_2))$$

defined by

$$\begin{aligned} \alpha(V') &= \bigcup\{\alpha_1(\{v\}) \times \alpha_2(\{v\}) \mid v \in V'\} \\ \gamma(DD) &= \{v \mid \alpha_1(\{v\}) \times \alpha_2(\{v\}) \subseteq DD\} \end{aligned}$$

Direct Tensor Product from Extraction Functions

Assume that the Galois connections $(\mathcal{P}(V), \alpha_i, \gamma_i, \mathcal{P}(D_i))$ are given by *extraction functions* $\eta_i : V \rightarrow D_i$ as in

$$
\begin{aligned}
\alpha_i(V') &= \{\eta_i(v) \mid v \in V'\} \\
\gamma_i(D_i') &= \{v \mid \eta_i(v) \in D_i'\}
\end{aligned}
$$

The Galois connection $(\mathcal{P}(V), \alpha, \gamma, \mathcal{P}(D_1 \times D_2))$ has

$$
\begin{aligned}
\alpha(V') &= \{(\eta_1(v), \eta_2(v)) \mid v \in V'\} \\
\gamma(DD) &= \{v \mid (\eta_1(v), \eta_2(v)) \in DD\}
\end{aligned}
$$

corresponding to the extraction function $\eta : V \rightarrow D_1 \times D_2$ defined by

$$
\eta(v) = (\eta_1(v), \eta_2(v))
$$

Example:

Using the direct tensor product to combine the detection of signs analysis for pairs of integers with the analysis of difference in magnitude.

$$(\mathcal{P}(\mathbf{Z} \times \mathbf{Z}), \alpha_{\mathsf{SSR}'}, \gamma_{\mathsf{SSR}'}, \mathcal{P}(\mathbf{Sign} \times \mathbf{Sign} \times \mathbf{Range}))$$

is given by

$$\begin{array}{rcl} \alpha_{\mathsf{SSR}'}(ZZ) & = & \{(\mathsf{sign}(z_1), \mathsf{sign}(z_2), \mathsf{range}(|z_1| - |z_2|)) \mid (z_1, z_2) \in ZZ\} \\ \gamma_{\mathsf{SSR}'}(SSR) & = & \{(z_1, z_2) \mid (\mathsf{sign}(z_1), \mathsf{sign}(z_2), \mathsf{range}(|z_1| - |z_2|)) \in SSR\} \end{array}$$

corresponding to twosignsrange : $\mathbf{Z} \times \mathbf{Z} \to \mathbf{Sign} \times \mathbf{Sign} \times \mathbf{Range}$ given by

$$\mathsf{twosignsrange}(z_1, z_2) = (\mathsf{sign}(z_1), \mathsf{sign}(z_2), \mathsf{range}(|z_1| - |z_2|))$$

Advantages of the Direct Tensor Product

The expression `(x,3*x)` may have a value in $\{(z, 3*z) \mid z \in \mathbf{Z}\}$ which in the direct tensor product can be described by

$$\alpha_{\mathsf{SSR}'}(\{(z, 3*z) \mid z \in \mathbf{Z}\}) = \{(\text{-},\text{-},\text{<-1}), (\text{0},\text{0},\text{0}), (\text{+},\text{+},\text{<-1})\}$$

compared to the direct product that gave

$$\alpha_{\mathsf{SSR}}(\{(z, 3*z) \mid z \in \mathbf{Z}\}) = (\{(\text{-},\text{-}), (\text{0},\text{0}), (\text{+},\text{+})\}, \{\text{0},\text{<-1}\})$$

Note that the Galois connection is *not* a Galois insertion because

$$\gamma_{\mathsf{SSR}'}(\emptyset) = \emptyset = \gamma_{\mathsf{SSR}'}(\{(\text{0},\text{0},\text{<-1})\})$$

so $\gamma_{\mathsf{SSR}'}$ is not injective and hence we do not have a Galois insertion.

From Direct to Reduced

Reduced Product

Let $(L, \alpha_1, \gamma_1, M_1)$ and $(L, \alpha_2, \gamma_2, M_2)$ be Galois connections.

The *reduced product* is the Galois *insertion*

$$(L, \alpha, \gamma, \varsigma[M_1 \times M_2])$$

defined by

$$\begin{aligned}
\alpha(l) &= (\alpha_1(l), \alpha_2(l)) \\
\gamma(m_1, m_2) &= \gamma_1(m_1) \sqcap \gamma_2(m_2) \\
\varsigma(m_1, m_2) &= \bigsqcap \{(m_1', m_2') \mid \gamma_1(m_1) \sqcap \gamma_2(m_2) = \gamma_1(m_1') \sqcap \gamma_2(m_2')\}
\end{aligned}$$

Reduced Tensor Product

Let $(\mathcal{P}(V), \alpha_1, \gamma_1, \mathcal{P}(D_1))$ and $(\mathcal{P}(V), \alpha_2, \gamma_2, \mathcal{P}(D_2))$ be Galois connection.

The *reduced tensor product* is the Galois *insertion*

$$(\mathcal{P}(V), \alpha, \gamma, \varsigma[\mathcal{P}(D_1 \times D_2)])$$

defined by

$$\begin{array}{rcl} \alpha(V') & = & \bigcup\{\alpha_1(\{v\}) \times \alpha_2(\{v\}) \mid v \in V'\} \\ \gamma(\mathsf{DD}) & = & \{v \mid \alpha_1(\{v\}) \times \alpha_2(\{v\}) \subseteq \mathsf{DD}\} \\ \varsigma(\mathsf{DD}) & = & \bigcap\{\mathsf{DD}' \mid \gamma(\mathsf{DD}) = \gamma(\mathsf{DD}')\} \end{array}$$

Example: Array Bounds Analysis

The superfluous elements of $\mathcal{P}(\mathbf{Sign} \times \mathbf{Sign} \times \mathbf{Range})$ will be removed when we use a reduced tensor product:

The reduction operator $\varsigma_{\mathsf{SSR'}}$ amounts to

$$\varsigma_{\mathsf{SSR'}}(SSR) = \bigcap\{SSR' \mid \gamma_{\mathsf{SSR'}}(SSR) = \gamma_{\mathsf{SSR'}}(SSR')\}$$

where $SSR, SSR' \subseteq \mathbf{Sign} \times \mathbf{Sign} \times \mathbf{Range}$.

The singleton sets constructed from the following 16 elements

$$\begin{array}{llll}
(-, 0, <\!-1), & (-, 0, -1), & (-, 0, 0), & \\
(0, -, 0), & (0, -, +1), & (0, -, >\!+1), & \\
(0, 0, <\!-1), & (0, 0, -1), & (0, 0, +1), & (0, 0, >\!+1), \\
(0, +, 0), & (0, +, +1), & (0, +, >\!+1), & \\
(+, 0, <\!-1), & (+, 0, -1), & (+, 0, 0) &
\end{array}$$

will be mapped to the empty set (as they are useless).

Example (cont.): Array Bounds Analysis

The remaining 29 elements of $\mathbf{Sign} \times \mathbf{Sign} \times \mathbf{Range}$ are

$$\begin{array}{lllll}
(-,-,<\!-1), & (-,-,-1), & (-,-,0), & (-,-,+1), & (-,-,>\!+1), \\
(-,0,+1), & (-,0,>\!+1), & & & \\
(-,+,<\!-1), & (-,+,-1), & (-,+,0), & (-,+,+1), & (-,+,>\!+1), \\
(0,-,<\!-1), & (0,-,-1), & (0,0,0), & (0,+,<\!-1), & (0,+,-1), \\
(+,-,<\!-1), & (+,-,-1), & (+,-,0), & (+,-,+1), & (+,-,>\!+1), \\
(+,0,+1), & (+,0,>\!+1), & & & \\
(+,+,<\!-1), & (+,+,-1), & (+,+,0), & (+,+,+1), & (+,+,>\!+1)
\end{array}$$

and they describe disjoint subsets of $\mathbf{Z} \times \mathbf{Z}$.

Any collection of properties can be descibed in 4 bytes.

Summary

The Array Bound Analysis has been designed from three simple Galois connections specified by extraction functions:

(i) an analysis approximating integers by their sign,

(ii) an analysis approximating pairs of integers by their difference in magnitude, and

(iii) an analysis approximating integers by their closeness to 0, 1 and -1.

These analyses have been combined using:

(iv) the relational product of analysis (i) with itself,

(v) the functional composition of analyses (ii) and (iii), and

(vi) the reduced tensor product of analyses (iv) and (v).

Induced Operations

Given: Galois connections $(L_i, \alpha_i, \gamma_i, M_i)$ so that M_i is more approximate than (i.e. is coarser than) L_i.

Aim: Replace an existing analysis over L_i with an analysis making use of the coarser structure of M_i.

Methods:

- Inducing along the abstraction function: move the computations from L_i to M_i.
- Application to Data Flow Analysis.
- Inducing along the concretisation function: move a widening from M_i to L_i.

Application to Data Flow Analysis

A *generalised Monotone Framework* consists of:

- the property space: a complete lattice $L = (L, \sqsubseteq)$;
- the set $\mathcal{F}$ of monotone functions from L to L.

An *instance* A of a generalised Monotone Framework consists of:

- a finite flow, $F \subseteq \mathbf{Lab} \times \mathbf{Lab}$;
- a finite set of extremal labels, $E \subseteq \mathbf{Lab}$;
- an extremal value, $\iota \in L$; and
- a mapping $f_\cdot$ from the labels $\mathbf{Lab}$ of F and E to monotone transfer functions from L to L.

Application to Data Flow Analysis

Let (L, α, γ, M) be a Galois connection.

Consider an instance B of the generalised Monotone Framework M that satisfies

- the mapping $g_{.}$ from the labels $\mathbf{Lab}$ of F and E to monotone transfer functions of $M \to M$ satisfies $g_\ell \;\sqsupseteq\; \alpha \circ f_\ell \circ \gamma$ for all ℓ; and
- the extremal value $\jmath$ satisfies $\gamma(\jmath) = \iota$;

and otherwise B is as A.

One can show that a solution to the B-constraints gives rise to a solution to the A-constraints:

$$(B_\circ, B_\bullet) \models \mathsf{B}^{\sqsupseteq} \quad \text{implies} \quad (\gamma \circ B_\circ, \gamma \circ B_\bullet) \models \mathsf{A}^{\sqsupseteq}$$

The Mundane Approach to Semantic Correctness

Here $F = \mathit{flow}(S_\star)$ and $E = \{\mathit{init}(S_\star)\}$.

Correctness of every solution to $\mathsf{A}^{\sqsupseteq}$ amounts to:

Assume $(A_\circ, A_\bullet) \models \mathsf{A}^{\sqsupseteq}$ and $\langle S_\star, \sigma_1\rangle \to^* \sigma_2$.

Then $\beta(\sigma_1) \sqsubseteq \iota$ implies $\beta(\sigma_2) \sqsubseteq \bigsqcup\{A_\bullet(\ell) \mid \ell \in \mathit{final}(S_\star)\}$.

where $\beta : \mathbf{State} \to L$.

One can then prove the correctness result for B:

Assume $(B_\circ, B_\bullet) \models \mathsf{B}^{\sqsupseteq}$ and $\langle S_\star, \sigma_1\rangle \to^* \sigma_2$.

Then $(\alpha \circ \beta)(\sigma_1) \sqsubseteq \jmath$ implies $(\alpha \circ \beta)(\sigma_2) \sqsubseteq \bigsqcup\{B_\bullet(\ell) \mid \ell \in \mathit{final}(S_\star)\}$.

Sets of States Analysis

Generalised Monotone Framework over $(\mathcal{P}(\mathbf{State}), \subseteq)$.
Instance SS for $S_\star$:

- the flow F is *flow*$(S_\star)$;
- the set E of extremal labels is $\{$*init*$(S_\star)\}$;
- the extremal value ι is $\mathbf{State}$; and
- the transfer functions are given by $f_\cdot^{\mathsf{SS}}$:

$$
\begin{array}{lrcl}
[x := a]^\ell & f_\ell^{\mathsf{SS}}(\Sigma) & = & \{\sigma[x \mapsto \mathcal{A}[\![a]\!]\sigma] \mid \sigma \in \Sigma\} \\
[\mathtt{skip}]^\ell & f_\ell^{\mathsf{SS}}(\Sigma) & = & \Sigma \\
[b]^\ell & f_\ell^{\mathsf{SS}}(\Sigma) & = & \Sigma
\end{array}
$$

where $\Sigma \subseteq \mathbf{State}$.

Correctness: Assume $(\mathit{SS}_\circ, \mathit{SS}_\bullet) \models \mathsf{SS}^{\supseteq}$ and $\langle S_\star, \sigma_1\rangle \to^* \sigma_2$.
Then $\sigma_1 \in \mathbf{State}$ implies $\sigma_2 \in \bigcup\{\mathit{SS}_\bullet(\ell) \mid \ell \in$ *final*$(S_\star)\}$.

Constant Propagation Analysis

Generalised Monotone Framework over $\widehat{\mathbf{State}}_{\mathsf{CP}} = ((\mathbf{Var} \to \mathbf{Z}^{\top})_{\perp}, \sqsubseteq)$.
Instance CP for $S_\star$:

- the flow F is *flow*$(S_\star)$;
- the set E of extremal labels is $\{$*init*$(S_\star)\}$;
- the extremal value ι is $\lambda x.\top$; and
- the transfer functions are given by the mapping $f^{\mathsf{CP}}_{\cdot}$:

$$
\begin{array}{lrcl}
[x := a]^\ell : & f^{\mathsf{CP}}_\ell(\widehat{\sigma}) & = & \begin{cases} \perp & \text{if } \widehat{\sigma} = \perp \\ \widehat{\sigma}[x \mapsto \mathcal{A}_{\mathsf{CP}}[\![a]\!]\widehat{\sigma}] & \text{otherwise} \end{cases} \\
[\mathtt{skip}]^\ell : & f^{\mathsf{CP}}_\ell(\widehat{\sigma}) & = & \widehat{\sigma} \\
[b]^\ell : & f^{\mathsf{CP}}_\ell(\widehat{\sigma}) & = & \widehat{\sigma}
\end{array}
$$

Galois Connection

The representation function $\beta_{\mathsf{CP}} : \mathbf{State} \to \widehat{\mathbf{State}}_{\mathsf{CP}}$ is defined by

$$\beta_{\mathsf{CP}}(\sigma) = \sigma$$

This gives rise to a Galois connection

$$(\mathcal{P}(\mathbf{State}), \alpha_{\mathsf{CP}}, \gamma_{\mathsf{CP}}, \widehat{\mathbf{State}}_{\mathsf{CP}})$$

where $\alpha_{\mathsf{CP}}(\Sigma) = \bigsqcup\{\beta_{\mathsf{CP}}(\sigma) \mid \sigma \in \Sigma\}$ and $\gamma_{\mathsf{CP}}(\widehat{\sigma}) = \{\sigma \mid \beta_{\mathsf{CP}}(\sigma) \sqsubseteq \widehat{\sigma}\}$.

One can show that for all labels ℓ

$\boxed{f_\ell^{\mathsf{CP}} \sqsupseteq \alpha_{\mathsf{CP}} \circ f_\ell^{\mathsf{SS}} \circ \gamma_{\mathsf{CP}}}$ as well as $\boxed{\gamma_{\mathsf{CP}}(\lambda x.\top) = \mathbf{State}}$

It follows that CP is an upper approximation to the analysis induced from SS and the Galois connection; therefore it is correct.

Widening Operators

Problem: We cannot guarantee that $(f^n(\bot))_n$ eventually stabilises nor that its least upper bound necessarily equals *lfp*(f).

Idea: We replace $(f^n(\bot))_n$ by a new sequence $(f^n_\nabla)_n$ that is known to eventually stabilise and to do so with a value that is a safe (upper) approximation of the least fixed point.

The new sequence is parameterised on the widening operator ∇: an upper bound operator satisfying a finiteness condition.

Upper bound operators

$\breve{\sqcup} : L \times L \to L$ is an *upper bound operator* iff

$$l_1 \ \sqsubseteq \ l_1 \mathbin{\breve{\sqcup}} l_2 \ \sqsupseteq \ l_2$$

for all $l_1, l_2 \in L$.

Let $(l_n)_n$ be a sequence of elements of L. Define the sequence $(l_n^{\breve{\sqcup}})_n$ by:

$$l_n^{\breve{\sqcup}} = \begin{cases} l_n & \text{if } n = 0 \\ l_{n-1}^{\breve{\sqcup}} \mathbin{\breve{\sqcup}} l_n & \text{if } n > 0 \end{cases}$$

Fact: If $(l_n)_n$ is a sequence and $\breve{\sqcup}$ is an upper bound operator then $(l_n^{\breve{\sqcup}})_n$ is an ascending chain; furthermore $l_n^{\breve{\sqcup}} \sqsupseteq \bigsqcup\{l_0, l_1, \cdots, l_n\}$ for all n.

Widening operators

An operator $\nabla : L \times L \to L$ is a *widening operator* iff

- it is an upper bound operator, and
- for all ascending chains $(l_n)_n$ the ascending chain $(l_n^{\nabla})_n$ eventually stabilises.

Widening operators

Given a monotone function $f : L \to L$ and a widening operator ∇ define the sequence $(f^n_\nabla)_n$ by

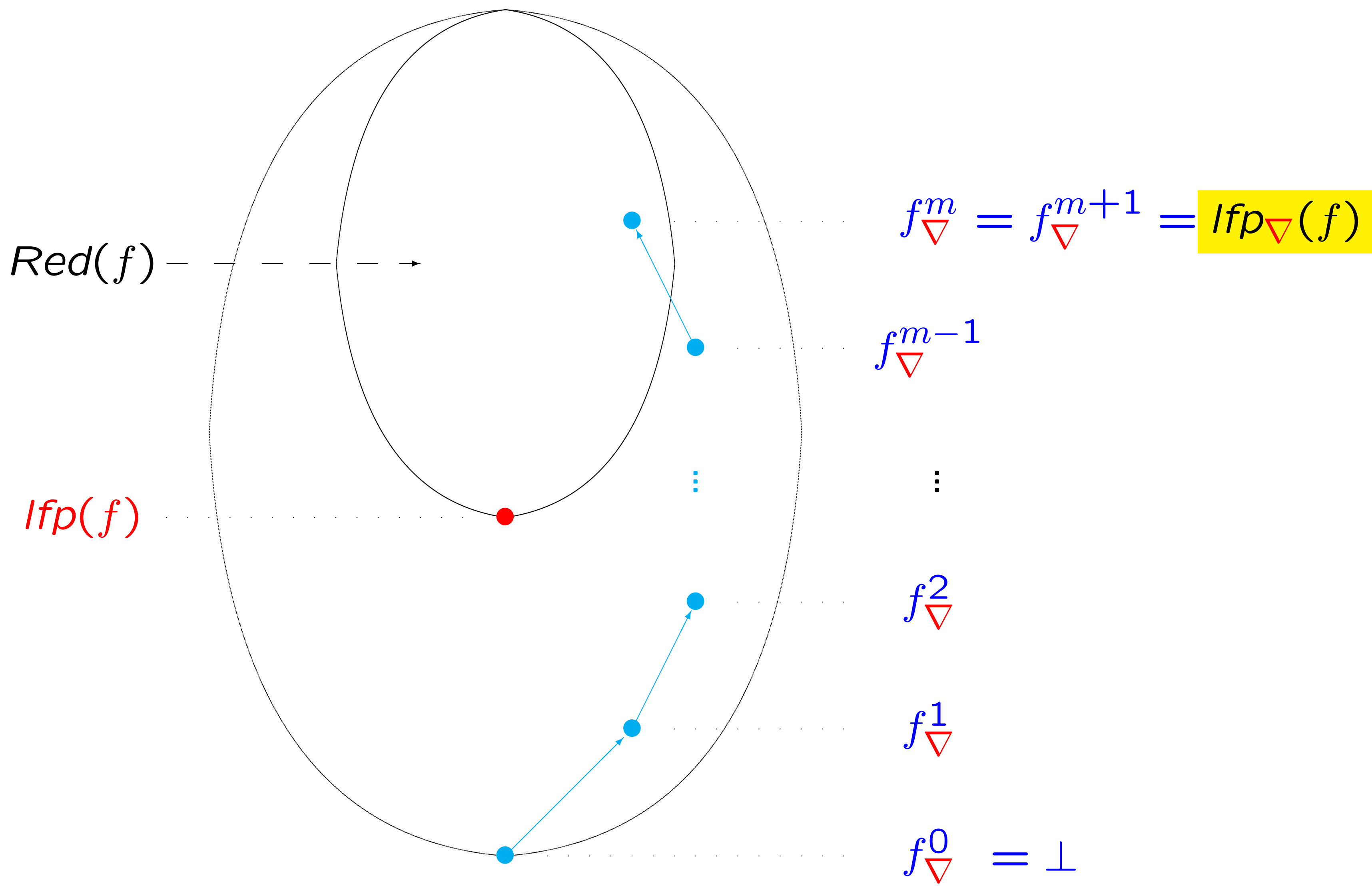
$$f^n_\nabla = \begin{cases} \bot & \text{if } n = 0 \\ f^{n-1}_\nabla & \text{if } n > 0 \ \wedge \ f(f^{n-1}_\nabla) \sqsubseteq f^{n-1}_\nabla \\ f^{n-1}_\nabla \ \nabla \ f(f^{n-1}_\nabla) & \text{otherwise} \end{cases}$$

One can show that:

- $(f^n_\nabla)_n$ is an ascending chain that eventually stabilises
- it happens when $f(f^m_\nabla) \sqsubseteq f^m_\nabla$ for some value of m
- Tarski's Theorem then gives $f^m_\nabla \sqsupseteq \mathit{lfp}(f)$

$$\boxed{\mathit{lfp}_\nabla(f) = f^m_\nabla}$$

The widening operator ∇ applied to f



Inducing along the Concretisation Function

Given an upper bound operator

$$\nabla_M : M \times M \to M$$

and a Galois connection (L, α, γ, M).

Define an upper bound operator

$$\nabla_L : L \times L \to L$$

by

$$l_1 \ \nabla_L \ l_2 = \gamma(\ \alpha(l_1) \ \nabla_M \ \alpha(l_2) \ )$$

It defines a widening operator if one of the following conditions holds:

(i) M satisfies the Ascending Chain Condition, or

(ii) (L, α, γ, M) is a Galois insertion and $\nabla_M : M \times M \to M$ is a widening.

Precision of the Induced Widening Operator

Lemma: Let (L, α, γ, M) be a Galois insertion such that $\gamma(\bot_M) = \bot_L$ and let $\nabla_M : M \times M \to M$ be a widening operator.

Then the widening operator $\nabla_L : L \times L \to L$ defined by

$$l_1 \ \nabla_L \ l_2 = \gamma(\alpha(l_1) \ \nabla_M \ \alpha(l_2))$$

satisfies

$$\mathit{lfp}_{\nabla_L}(f) = \gamma(\mathit{lfp}_{\nabla_M}(\alpha \circ f \circ \gamma))$$

for all monotone functions $f : L \to L$.

Precision of the Induced Widening Operator

Corollary: Let M be of finite height, let (L, α, γ, M) be a Galois insertion (such that $\gamma(\bot_M) = \bot_L$), and let ∇_M equal the least upper bound operator $\sqcup_M$.

Then the above lemma shows that $\mathit{lfp}_{\nabla_L}(f) = \gamma(\mathit{lfp}(\alpha \circ f \circ \gamma))$.

This means that $\mathit{lfp}_{\nabla_L}(f)$ *equals* the result we would have obtained if we decided to work with $\alpha \circ f \circ \gamma : M \to M$ instead of the given $f : L \to L$; furthermore the number of iterations needed turn out to be the same. However, for all other operations the increased precision of L is available.