Discrete Fixpoint Approximation Methods in Program Static Analysis

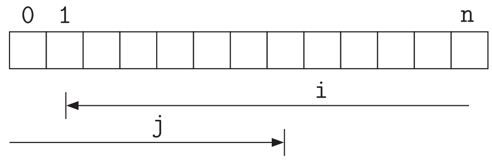
P. Cousot

Département de Mathématiques et Informatique
École Normale Supérieure – Paris
<cousot@dmi.ens.fr>
<http://www.dmi.ens.fr/~cousot>

Static Program analysis

- Automatic determination of runtime properties of infinite state programs
- Applications:
  - compilation (dataflow analysis, type inference),
  - program transformation (partial evaluation, parallelization/vectorization, . . .)
  - program verification (test generation, abstract debugging, . . .)
- Problems:
  - text inspection only (excluding executions or simulations)
  - undecidable
  - necessarily approximate

Example




```
{ n:Ω[1]; i:Ω; j:Ω }
read_int(n);
{ n:!![2][0,+∞[3]]; i:Ω; j:Ω }
i := n;
{ n:[0,+∞]; i:[0,+∞]; j:[1,+∞]?[4] }
while (i <> 0) do
   { n:[0,+∞]; i:[1,+∞]; j:[1,+∞]? }
    j := 0;
   { n:[0,+∞]; i:[1,+∞]; j:[0,+∞] }
    while (j <> i) do
       { n:[0,+∞]; i:[1,+∞]; j:[0,1073741822]!!
}
        j := (j + 1)
       { n:[0,+∞]; i:[1,+∞]; j:[1,+∞] }
    od;
   { n:[0,+∞]; i:[1,+∞]; j:[1,+∞] }
    i := (i - 1);
   { n:[0,+∞]; i:[0,1073741822]; j:[1,+∞] }
od;
{ n:[0,+∞]; i:[0,0]; j:[1,+∞]? }
```

[1] Ω denotes uninitialization.
[2] !! denotes inevitable error when the invariant is violated.
[3] $+\infty$ = 1073741823, $-\infty$ = -1073741824.
[4] This questionmark indicates possible uninitialization.

Abstract interpretation

Abstract interpretation [1, 2]:

- design method for static analysis algorithms;
- effective approximation of the semantics of programs;
- often, the semantics maps the program text to a model of computation obtained as the least fixpoint of an operator on a partially ordered semantic domain;
- effective approximation of fixpoints of posets;

References

[1] P. Cousot and R. Cousot. Abstract interpretation: a unified lattice model for static analysis of programs by construction or approximation of fixpoints. In *Conference Record of the Fourth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 238–252, Los Angeles, California, 1977. ACM Press, New York, New York, USA.

[2] P. Cousot and R. Cousot. Systematic design of program analysis frameworks. In *Conference Record of the Sixth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 269–282, San Antonio, Texas, 1979. ACM Press, New York, New York, USA.

Fixpoint semantics

Program semantics can be defined as least fixpoints [3]:

$$\mathrm{lfp}^{\sqsubseteq}\, \mathcal{F}$$

where

$$\begin{aligned} \mathcal{F}(\mathrm{lfp}^{\sqsubseteq}\, \mathcal{F}) &= \mathrm{lfp}^{\sqsubseteq}\, \mathcal{F} \\ \mathcal{F}(x) = x &\Longrightarrow \mathrm{lfp}^{\sqsubseteq}\, \mathcal{F} \sqsubseteq x \end{aligned}$$

of amonotonic operator $\mathcal{F} \in \mathcal{L} \xmapsto{\mathrm{m}} \mathcal{L}$ on a complete partial order (CPO):

$$\langle \mathcal{L},\, \sqsubseteq,\, \perp,\, \sqcup \rangle$$

where $\langle \mathcal{L},\, \sqsubseteq \rangle$ is a poset with infimum $\perp$ and the least upper bound (lub) $\sqcup$ of increasing chains exists.

Reference

[3] P. Cousot. Design of semantics by abstract interpretation, invited address. In *Mathematical Foundations of Programming Semantics, Thirteenth Annual Conference (MFPS XIII)*, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA, 23–26 March 1997.

Tarski's Fixpoint Theorem

A monotonic map $\varphi \in L \longmapsto L$ on a complete lattice:

$$\langle L,\, \sqsubseteq,\, \perp,\, \top,\, \sqcup,\, \sqcap \rangle$$

has a least fixpoint:

$$\mathrm{lfp}\, \varphi = \sqcap \{x \in L \mid \varphi(x) \sqsubseteq x\}$$

and, dually, a greatest fixpoint:

$$\mathrm{gfp}\, \varphi = \sqcup \{x \in L \mid x \sqsubseteq \varphi(x)\}$$

Kleenian fixpoint theorem [5]

- A map $\varphi \in L \xmapsto{\mathrm{c}} L$ on a cpo $\langle L,\, \sqsubseteq,\, \perp,\, \sqcup \rangle$ is upper-continuous iff it preserves lubs of increasing chains $x_i, i \in \mathbb{N}$:

$$\varphi(\bigsqcup_{i \in \mathbb{N}} x_i) = \bigsqcup_{i \in \mathbb{N}} \varphi(x_i)\ ;$$

- The least fixpoint of an <u>upper-continuous</u> map $\varphi \in L \xmapsto{\mathrm{c}} L$ on a cpo $\langle L,\, \sqsubseteq,\, \perp,\, \sqcup \rangle$ is:

$$\mathrm{lfp}\, \varphi = \bigsqcup_{n \geq 0} \varphi^n(\perp)$$

where the iterates $\varphi^n(x)$ of φ from x are:
  - $\varphi^0(x) \stackrel{\mathrm{def}}{=} x$;
  - $\varphi^{n+1}(x) \stackrel{\mathrm{def}}{=} \varphi(\varphi^n(x))$ for all $x \in L$.

[5] Can be generalized to monotonic non-continuous maps by considering transfinite iterates.

Chaotic/asynchronous iterations

- Convergent iterates $L = \bigsqcup_{n \geq 0} F^n(P)$ of a monotonic system of equations on a poset:

$$X = F(X) \qquad \begin{cases} X_1 = F_1(X_1, \ldots, X_n) \\ \ldots \\ X_n = F_n(X_1, \ldots, X_n) \end{cases}$$

starting from a prefixpoint ($P \mathrel{\dot{\sqsubseteq}} F(P)$) always converge to the same limit L whichever chaotic or asynchronous iteration strategy is used.

Example: reachability analysis

- Program:

```
{ X1 }
x := 1;
{ X2 }
while (x < 1000) do
   { X3 }
    x := x + 1;
   { X4 }
od;
{ X5 }
```

- System of equations:

$$\begin{cases} X_1 = \{\Omega\} \\ X_2 = \{1\} \cup X_4 \\ X_3 = \{x \in X_2 \mid x < 1000\} \\ X_4 = \{x+1 \mid x \in X_3\} \\ X_5 = \{x \in X_2 \mid x \geq 1000\} \end{cases}$$

- Reachable states:

$$\begin{cases} X_1 = \{\Omega\} \\ X_2 = \{x \mid 1 \leq x \leq 1000\} \\ X_3 = \{x \mid 1 \leq x < 1000\} \\ X_4 = \{x+1 \mid x \in X_3\} \\ X_5 = \{1000\} \end{cases}$$

Effective fixpoint approximation

- Simplify the fixpoint system of semantic equations: Galois connections;
- Accelerate convergence of the iterates: widening/narrowing;

Definition of Galois connections

Given posets $\langle \mathcal{P}, \sqsubseteq \rangle$ and $\langle \mathcal{Q}, \preceq \rangle$, a Galois connection is a pair of maps such that:

$$\alpha \in \mathcal{P} \longmapsto \mathcal{Q}$$
$$\gamma \in \mathcal{Q} \longmapsto \mathcal{P}$$
$$\forall x \in \mathcal{P} : \forall y \in \mathcal{Q} : \alpha(x) \preceq y \Leftrightarrow x \sqsubseteq \gamma(y)$$

in which case we write:

$$\langle \mathcal{P}, \sqsubseteq \rangle \xleftrightarrow[\alpha]{\gamma} \langle \mathcal{Q}, \preceq \rangle$$

Equivalent definition of Galois connections

$\langle \mathcal{P}, \sqsubseteq \rangle \xleftrightarrow[\alpha]{\gamma} \langle \mathcal{Q}, \preceq \rangle$ Galois connection

$\Longleftrightarrow$

$\left[\alpha \in \langle \mathcal{D}^\natural, \sqsubseteq \rangle \overset{\mathrm{m}}{\longmapsto} \langle \mathcal{Q}, \preceq \rangle\right] \wedge$ α monotone

$\left[\gamma \in \langle \mathcal{Q}, \preceq \rangle \overset{\mathrm{m}}{\longmapsto} \langle \mathcal{P}, \sqsubseteq \rangle\right] \wedge$ γ monotone

$[\forall x \in \mathcal{P} : x \sqsubseteq \gamma \circ \alpha(x)] \wedge$ $\gamma \circ \alpha$ extensive

$[\forall y \in \mathcal{Q} : \alpha \circ \gamma(y) \preceq y]$ $\alpha \circ \gamma$ reductive

Duality principle

- We write $\leq^{-1}$ or $\geq$ for the inverse of the partial order $\leq$.
- Observe that:
  $$\langle \mathcal{P},\, \sqsubseteq \rangle \xleftarrow[\alpha]{\gamma}\!\!\!\!\!\!\!\!\xrightarrow{} \langle \mathcal{Q},\, \preceq \rangle$$
  if and only if
  $$\mathcal{Q}(\succeq) \xleftarrow[\gamma]{\alpha}\!\!\!\!\!\!\!\!\xrightarrow{} \mathcal{P}(\sqsupseteq)$$
- duality principle: if a theorem is true for all posets, then so is its dual obtained by substituting $\geq$, $>$, $\top$, $\bot$, $\vee$, $\wedge$, α, γ etc. respectively for $\leq$, $<$, $\bot$, $\top$, $\wedge$, $\vee$, γ, α, etc.

Example 1 of Galois connection

If

- $@ \in \mathcal{P} \longmapsto \mathcal{Q}$
- $\alpha \in \wp(\mathcal{P}) \longmapsto \wp(\mathcal{Q})$ direct image
  $\alpha(X) \stackrel{\text{def}}{=} \{@(x) \mid x \in X\}$
- $\gamma \in \wp(\mathcal{Q}) \longmapsto \wp(\mathcal{P})$ inverse image
  $\gamma(Y) \stackrel{\text{def}}{=} \{x \mid @(x) \in Y\}$

then

$$\langle \wp(\mathcal{P}),\, \subseteq \rangle \xleftarrow[\alpha]{\gamma}\!\!\!\!\!\!\!\!\xrightarrow{} \langle \wp(\mathcal{Q}),\, \subseteq \rangle$$

Example 2 of Galois connection

If

- $\rho \subseteq \mathcal{P} \times \mathcal{Q}$
- $\alpha \in \wp(\mathcal{P}) \longmapsto \wp(\mathcal{Q})$
  $\alpha(X) = \mathrm{post}[\rho]X$ post-image
  $\stackrel{\text{def}}{=} \{y \mid \exists x \in X : \langle x,\, y \rangle \in \rho\}$
- $\gamma \in \wp(\mathcal{Q}) \longmapsto \wp(\mathcal{P})$
  $\gamma(Y) = \widetilde{\mathrm{pre}}[\rho]Y$ dual pre-image
  $\stackrel{\text{def}}{=} \{x \mid \forall y : \langle x,\, y \rangle \in \rho \Rightarrow y \in Y\}$

then

$$\langle \wp(\mathcal{P}),\, \subseteq \rangle \xleftarrow[\alpha]{\gamma}\!\!\!\!\!\!\!\!\xrightarrow{} \langle \wp(\mathcal{Q}),\, \subseteq \rangle$$

Example 3 of Galois connections

If S and T are sets then

$$\langle \wp(S \longmapsto T),\, \subseteq \rangle \xleftarrow[\alpha]{\gamma}\!\!\!\!\!\!\!\!\xrightarrow{} \langle S \longmapsto \wp(T),\, \dot{\subseteq} \rangle$$

where:

$$\alpha(F) \stackrel{\text{def}}{=} \lambda x \cdot \{f(x) \mid f \in F\}$$
$$\gamma(\varphi) \stackrel{\text{def}}{=} \{f \in S \longmapsto T \mid \forall x \in S : f(x) \in \varphi(x)\}$$

Moore families

- A Moore family is a subset of a complete lattice $\langle L, \sqsubseteq, \bot, \top, \sqcup, \sqcap\rangle$ containing $\top$ and closed under arbitrary glbs $\sqcap$;
- If $\langle \mathcal{P}, \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathcal{Q}, \preceq\rangle$ and $\langle \mathcal{P}, \sqsubseteq, \bot, \top, \sqcap, \sqcup\rangle$ is a complete lattice then $\gamma(\mathcal{Q})$ is a Moore family.
- A consequence is that one can reason upon the abstract semantics using only $\mathcal{P}$ and the image of $\mathcal{P}$ by the upper closure operator $\gamma \circ \alpha$ (instead of $\mathcal{Q}$).
- Intuition:
  - The upper-approximation of $x \in \mathcal{P}$ is any $y \in \gamma(\mathcal{Q})$ such that $x \sqsubseteq y$;
  - The best approximation of x is $\gamma \circ \alpha(x)$.

Unique adjoint

In a Galois connection, one function uniquely determines the other:

- If $\langle \mathcal{P}, \sqsubseteq\rangle \underset{\alpha_1}{\overset{\gamma_1}{\leftrightarrows}} \langle \mathcal{Q}, \preceq\rangle$ and $\langle \mathcal{P}, \sqsubseteq\rangle \underset{\alpha_2}{\overset{\gamma_2}{\leftrightarrows}} \langle \mathcal{Q}, \preceq\rangle$, then $(\alpha_1 = \alpha_2)$ if and only if $(\gamma_1 = \gamma_2)$.

$$\begin{aligned} \forall x \in \mathcal{P} : \alpha(x) &= \sqcap\{y \mid x \sqsubseteq \gamma(y)\} \\ \forall y \in \mathcal{Q} : \gamma(y) &= \sqcup\{x \mid \alpha(x) \preceq y\} \end{aligned}$$

Preservation of lubs/glbs

- If $\langle \mathcal{P}, \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathcal{Q}, \preceq\rangle$, then α preserves existing lubs: if $\sqcup X$ exists, then $\alpha\big(\sqcup X\big)$ is the lub of $\{\alpha(x) \mid x \in X\}$.

By the duality principle:

- If $\langle \mathcal{P}, \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathcal{Q}, \preceq\rangle$ then γ preserves existing glbs: if $Y \subseteq \mathcal{Q}$ and $\sqcap Y$ exists, then $\gamma\big(\sqcap Y\big)$ is the glb of $\{\gamma(y) \mid y \in Y\}$.

Complete join preserving abstraction function and complete meet preserving concretization function

- Let $\langle \mathcal{P}, \sqsubseteq\rangle$ and $\langle \mathcal{Q}, \preceq\rangle$ be posets.
- If
  1. $\alpha \in \mathcal{P}(\sqcup) \stackrel{a}{\longmapsto} \mathcal{Q}(\sqcup)$
  2. $\sqcup\{x \mid \alpha(x) \preceq y\}$ exists for all $y \in \mathcal{Q}$,

  then
  $$\langle \mathcal{P}, \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathcal{Q}, \preceq\rangle$$
  where $\forall y \in \mathcal{Q} : \gamma(y) = \sqcup\{x \mid \alpha(x) \preceq y\}$
- By duality, if
  1. $\gamma \in \mathcal{Q}(\sqcap) \stackrel{a}{\longmapsto} \mathcal{P}(\sqcap)$
  2. $\sqcap\{y \mid x \sqsubseteq \gamma(y)\}$ exists for all $x \in \mathcal{P}$

  then
  $$\langle \mathcal{P}, \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathcal{Q}, \preceq\rangle$$
  where $\forall x \in \mathcal{P} : \alpha(x) = \sqcap\{y \mid x \sqsubseteq \gamma(y)\}$

Galois surjection & injection

If $\langle \mathcal{P},\ \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathcal{Q},\ \preceq\rangle$, then:

α is onto

iff γ is one-to-one

iff $\alpha \circ \gamma$ is the identity

By the duality principle, if $\langle \mathcal{P},\ \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathcal{Q},\ \preceq\rangle$, then:

α is one-to-one

iff γ is onto

iff $\gamma \circ \alpha$ is the identity

Notation:

$\langle \mathcal{P},\ \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathcal{Q},\ \preceq\rangle$	Galois connection
$\langle \mathcal{P},\ \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\longleftarrow\!\!\!\twoheadrightarrow}} \langle \mathcal{Q},\ \preceq\rangle$	Galois surjection
$\langle \mathcal{P},\ \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\twoheadleftarrow\!\!\!\longrightarrow}} \langle \mathcal{Q},\ \preceq\rangle$	Galois injection
$\langle \mathcal{P},\ \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\twoheadleftarrow\!\!\!\twoheadrightarrow}} \langle \mathcal{Q},\ \preceq\rangle$	Galois bijection

with $\longleftarrow$ denoting 'into' and $\twoheadrightarrow$ denoting 'onto'.

The image of a Cpo by a Galois surjection is a Cpo

- If $\langle \mathcal{P},\ \sqsubseteq,\ \bot,\ \sqcup\rangle$ is a cpo, $\langle \mathcal{Q},\ \preceq\rangle$ is a poset and
$$\langle \mathcal{P},\ \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\longleftarrow\!\!\!\twoheadrightarrow}} \langle \mathcal{Q},\ \preceq\rangle$$
then
$$\langle \mathcal{Q},\ \preceq,\ 0,\ \vee\rangle$$
is a cpo with:
$$0 \stackrel{\text{def}}{=} \alpha(\bot)$$
$$\vee X \stackrel{\text{def}}{=} \alpha(\bigsqcup_{x\in X} \gamma(x))$$

The image of a complete lattice by a Galois surjection is a complete lattice

- If $\langle \mathcal{P},\ \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\longleftarrow\!\!\!\twoheadrightarrow}} \langle \mathcal{Q},\ \preceq\rangle$ and $\langle \mathcal{P},\ \sqsubseteq,\ \bot,\ \top,\ \sqcap,\ \sqcup\rangle$ is a complete lattice, then so is $\langle \mathcal{Q},\ \preceq\rangle$ with

$0 = \alpha(\bot)$	infimum
$1 = \alpha(\top)$	supremum
$\vee Y = \alpha(\bigsqcup_{y\in Y} \gamma(y))$	lub
$\wedge Y = \alpha(\bigsqcap_{y\in Y} \gamma(y))$	glb

Pointwise extension of Galois connections

- If $\langle \mathcal{P},\ \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathcal{Q},\ \preceq\rangle$ then:

$$\langle \mathcal{S} \longmapsto \mathcal{P},\ \dot{\sqsubseteq}\rangle \underset{\dot{\alpha}}{\overset{\dot{\gamma}}{\leftrightarrows}} \langle \mathcal{S} \longmapsto \mathcal{Q},\ \dot{\preceq}\rangle$$

where:

$$\dot{\alpha}(f) \stackrel{\text{def}}{=} \alpha \circ f$$
$$\dot{\gamma}(g) \stackrel{\text{def}}{=} \gamma \circ g$$

Lifting Galois connections at higher-order

If

$$\langle \mathcal{P}_1,\ \sqsubseteq_1\rangle \underset{\alpha_1}{\overset{\gamma_1}{\leftrightarrows}} \langle \mathcal{Q}_1,\ \preceq_1\rangle$$

$$\langle \mathcal{P}_2,\ \sqsubseteq_2\rangle \underset{\alpha_2}{\overset{\gamma_2}{\leftrightarrows}} \langle \mathcal{Q}_2,\ \preceq_2\rangle$$

then

$$\langle \mathcal{P}_1 \overset{\mathrm{m}}{\longmapsto} \mathcal{P}_2,\ \dot{\sqsubseteq}_2\rangle \underset{\vec{\alpha}}{\overset{\vec{\gamma}}{\leftrightarrows}} \langle \mathcal{Q}_1 \overset{\mathrm{m}}{\longmapsto} \mathcal{Q}_2,\ \dot{\preceq}_2\rangle$$

where

$$\varphi \mathrel{\dot{\sqsubseteq}} \psi \overset{\mathrm{def}}{=} \forall x : \varphi(x) \sqsubseteq \psi(x)$$

$$\vec{\alpha}(\varphi) \overset{\mathrm{def}}{=} \alpha_2 \circ \varphi \circ \gamma_1$$

$$\vec{\gamma}(\psi) \overset{\mathrm{def}}{=} \gamma_2 \circ \psi \circ \alpha_1$$

Composition of Galois connections

The composition of Galois connections is a Galois connection:

$$\Big(\langle \mathcal{P}^\flat,\ \sqsubseteq^\flat\rangle \underset{\alpha_1}{\overset{\gamma_1}{\leftrightarrows}} \langle \mathcal{P},\ \sqsubseteq^\natural\rangle \wedge$$

$$\langle \mathcal{P},\ \sqsubseteq^\natural\rangle \underset{\alpha_2}{\overset{\gamma_2}{\leftrightarrows}} \langle \mathcal{Q},\ \sqsubseteq^\sharp\rangle\Big)$$

$$\Rightarrow$$

$$\langle \mathcal{P}^\flat,\ \sqsubseteq^\flat\rangle \underset{\alpha_2 \circ \alpha_1}{\overset{\gamma_1 \circ \gamma_2}{\leftrightarrows}} \langle \mathcal{Q},\ \sqsubseteq^\sharp\rangle$$

Example: interval analysis

- Concrete/exact:

$D \overset{\mathrm{def}}{=} \{x \in \mathbb{N} \mid \texttt{min\_int} \leq x \leq \texttt{max\_int}\}$

$D_\Omega \overset{\mathrm{def}}{=} D \cup \{\Omega\}$ values & uninitialization

$n \geq 1$ program points

V variables

$S \overset{\mathrm{def}}{=} [1, n] \longmapsto (V \longmapsto D_\Omega)$ states

- Abstract/approximate:

$I \overset{\mathrm{def}}{=} \{[a, b] \mid \{x \in \mathbb{N} \mid a \leq x \leq b\}$ intervals

$\gamma(\Omega) \overset{\mathrm{def}}{=} \{\Omega\}$ concretization

$\gamma([a, b]) \overset{\mathrm{def}}{=} \{x \in \mathbb{N} \mid a \leq x \leq b\}$

$\gamma(\langle\Omega,\ [a, b]\rangle) \overset{\mathrm{def}}{=} \gamma(\Omega) \cup \gamma([a, b])$

$L \overset{\mathrm{def}}{=} [1, n] \longmapsto (V \longmapsto A)$ abstract domain

$\gamma \in A \longmapsto \wp(D_\Omega)$ concretization

$\gamma(P) \overset{\mathrm{def}}{=} \{\rho \mid \forall i \in [1, n] : \forall v \in V :$
$\rho(i)(v) \in \gamma(P(i)(v))\}$

$P \mathrel{\ddot{\sqsubseteq}} Q \overset{\mathrm{def}}{=} \gamma(P) \subseteq \gamma(Q)$ ordering

- Galois connexion:

$$\langle \wp(S),\ \subseteq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle L,\ \ddot{\sqsubseteq}\rangle$$

Kleenian fixpoint abstraction

If $\langle \mathcal{D},\ \sqsubseteq,\ \bot,\ \sqcup\rangle$ is a cpo, $\langle \mathcal{Q},\ \preceq\rangle$ is a poset, $F \in \mathcal{P} \overset{\mathrm{m}}{\longmapsto} \mathcal{D}$, $F^\sharp \in \mathcal{Q} \overset{\mathrm{m}}{\longmapsto} \mathcal{Q}$, and

$$F^\sharp \circ \alpha \ = \ \alpha \circ F$$

$$\langle \mathcal{D},\ \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathcal{D}^\sharp,\ \preceq\rangle$$

then

$$\alpha(\mathrm{lfp}^{\sqsubseteq} F) \ = \ \mathrm{lfp}^{\preceq} F^\sharp$$

Kleenian fixpoint approximation

If $\langle \mathcal{D},\ \sqsubseteq,\ \bot,\ \sqcup \rangle$ is a cpo, $\langle \mathcal{Q},\ \preceq \rangle$ is a poset, $F \in \mathcal{P} \overset{\mathrm{m}}{\longmapsto} \mathcal{D}$, $F^\sharp \in \mathcal{A} \overset{\mathrm{m}}{\longmapsto} \mathcal{A}$, and

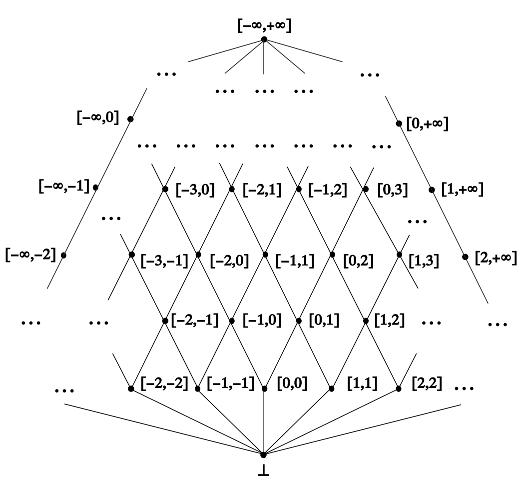
$$F^\sharp \circ \alpha \ \dot{\preceq}\ \alpha \circ F$$

$$\langle \mathcal{D},\ \sqsubseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathcal{D}^\sharp,\ \preceq \rangle$$

then

$$\alpha(\mathrm{lfp}^{\sqsubseteq} F) \preceq \mathrm{lfp}^{\preceq} F^\sharp$$

Interval lattice



Infinite strictly increasing chains

- Because of infinite (or very long) strictly increasing chains, the fixpoint iterates may not converge (or very slowly);
- Because of infinite (or very long) strictly decreasing chains, the local decreasing iterates may not converge (or not rapidly enough);
- The design strategy of using a more abstract domain satisfying the ACC often yields too imprecise results;
- It is often both more precise and faster to speed up convergence using widenings along increasing chains and narrowings along deceasing ones.

Slow fixpoint iterations

```
-- program:
0: x := 1;
1: while true do
    2: x := (x + 1)
    3: od {false}
4:
-- forward abstract equations:
X0 = (INIT 0)
X1 = assign[|x, 1|](X0) U X3
X2 = assert[|true|](X1)
X3 = assign[|x, (x + 1)|](X2)
X4 = assert[|false|](X1)
-- iterations from:
X0 = { x:_O_ }   X1 = _|_   X2 = _|_
X3 = _|_         X4 = _|_
--
X0 = { x:_O_ }
X1 = { x:[1,1] }
X2 = { x:[1,1] }
X3 = { x:[2,2] }
X1 = { x:[1,2] }
X2 = { x:[1,2] }
X3 = { x:[2,3] }
X1 = { x:[1,3] }
X2 = { x:[1,3] }
X3 = { x:[2,4] }
X1 = { x:[1,4] }
X2 = { x:[1,4] }
X3 = { x:[2,5] }
...
```

Widening

definition: A widening $\nabla \in P \times P \longmapsto P$ on a poset $\langle P,\ \sqsubseteq\rangle$ satisfies:

- $\forall x, y \in P : x \sqsubseteq (x \nabla y) \wedge y \sqsubseteq (x \nabla y)$
- For all increasing chains $x^0 \sqsubseteq x^1 \sqsubseteq \ldots$ the increasing chain $y^0 \stackrel{\text{def}}{=} x^0, \ldots, y^{n+A} \stackrel{\text{def}}{=} y^n \nabla x^{n+1}, \ldots$ is not strictly increasing.

use:

- Approximate missing lubs.
- Convergence acceleration;

Iteration sequence with widening

- Let F be a monotonic operator on a poset $\langle P,\ \sqsubseteq\rangle$;
- Let $\nabla \in P \times P \longmapsto P$ be a widening;
- The iteration sequence with widening ∇ for F from $\bot$ is $X^n, n \in \mathbb{N}$:
  - $X^0 = \bot$
  - $X^{n+1} = X^n \quad \text{if } F(X^n) \sqsubseteq (X^n)$
  - $X^{n+1} = X^n \nabla F(X^n) \quad \text{if } F(X^n) \not\sqsubseteq X^n$

Fixpoint upper approximation by widening

- Any iteration sequence with widening is increasing and stationary after finitely many iteration steps;
- Its limit L^{∇} is a post-fixpoint of F, whence an upper-approximation of the least fixpoint $\mathrm{lfp}^{\sqsubseteq} F$[6]:

$$\mathrm{lfp}^{\sqsubseteq} F \sqsubseteq L^{\nabla}$$

[6] if $\mathrm{lfp}^{\sqsubseteq} F$ does exist e.g. if $\langle P,\ \sqsubseteq,\ \bot,\ \cup\rangle$ is a cpo.

Example of widening for intervals

$$\begin{array}{l}
[a, b] \nabla [a', b'] \stackrel{\text{def}}{=} \\
\quad [(a' >= a\ ?\ a \mid a' >= 1\ ?\ 1 \\
\quad\ \mid a' >= 0\ ?\ 0 \mid a' >= -1\ ?\ -1 \\
\quad\ \mid \texttt{min_int}), \\
\quad\ \ (b' <= b\ ?\ b \mid b' <= -1\ ?\ -1 \\
\quad\ \ \mid b' <= 0\ ?\ 0 \mid b' <= 1\ ?\ 1 \\
\quad\ \ \mid \texttt{max_int}]
\end{array}$$

$$\begin{array}{rcl}
\bot \nabla y & \stackrel{\text{def}}{=} & y \\
x \nabla \bot & \stackrel{\text{def}}{=} & x \\
\Omega \nabla \Omega & \stackrel{\text{def}}{=} & \Omega \\
\Omega \nabla [a, b] & \stackrel{\text{def}}{=} & \langle \Omega,\ [a, b]\rangle \\
\Omega \nabla \langle \Omega,\ [a, b]\rangle & \stackrel{\text{def}}{=} & \langle \Omega,\ [a, b]\rangle \\
[a, b] \nabla \Omega & \stackrel{\text{def}}{=} & \langle \Omega,\ [a, b]\rangle \\
\langle \Omega,\ [a, b]\rangle \nabla \Omega & \stackrel{\text{def}}{=} & \langle \Omega,\ [a, b]\rangle \\
[a, b] \nabla \langle \Omega,\ [a', b']\rangle & \stackrel{\text{def}}{=} & \langle \Omega,\ [a, b] \nabla [a', b']\rangle \\
\langle \Omega,\ [a, b]\rangle \nabla [a', b'] & \stackrel{\text{def}}{=} & \langle \Omega,\ [a, b] \nabla [a', b']\rangle \\
\langle \Omega,\ [a, b]\rangle \nabla \langle \Omega,\ [a', b']\rangle & \stackrel{\text{def}}{=} & \langle \Omega,\ [a, b] \nabla [a', b']\rangle
\end{array}$$

Widening for systems of equations

A very rough idea:

- compute the dependence graph of the system of equations;
- widen at cut-points;
- iterate according to the weak topological ordering

Example

```
labelled program:
--
0: x := 1;
1: y := 1000;
2: while (x < y) do
3:   x := (x + 1)
4: od
5:
--
forward abstract equations:
--
X0 = (INIT 0)
X1 = assign[|x, 1|](X0)
X2 = assign[|y, 1000|](X1) U X4
X3 = assert[|(x < y)|](X2)
X4 = assign[|x, (x + 1)|](X3)
X5 = assert[|((y < x) | (x = y))|](X2)
--
forward graph with 6 vertices:
  0 : {1}
  1 : {2}
  2 : {3, 5}
  3 : {4}
  4 : {2}
  5 : {}
--
forward weak topological order: 0 1 ( 2 3 4 ) 5
forward cut & check points: {2}
```


Interval program analysis example with widening

```
labelled program:
--
0: x := 1;
1: y := 1000;
2: while (x < y) do
3:   x := (x + 1)
4: od
5:
--
iterations with widening from:
   X0 = { x:_O_; y:_O_ }   X1 = _|_   X2 = _|_
   X3 = _|_                X4 = _|_   X5 = _|_
--
X0 = { x:_O_; y:_O_ }
X1 = { x:[1,1]; y:_O_ }
widening at 2 by { x:[1,1]; y:[1000,1000] }
X2 = { x:[1,1]; y:[1000,1000] }
X3 = { x:[1,1]; y:[1000,1000] }
X4 = { x:[2,2]; y:[1000,1000] }
widening at 2 by { x:[1,2]; y:[1000,1000] }
X2 = { x:[1,+oo]; y:[1000,1000] }
X3 = { x:[1,999]; y:[1000,1000] }
X4 = { x:[2,1000]; y:[1000,1000] }
X2 = { x:[1,1000]; y:[1000,1000] }
X3 = { x:[1,999]; y:[1000,1000] }
X4 = { x:[2,1000]; y:[1000,1000] }
X5 = { x:[1000,1000]; y:[1000,1000] }
--
```


Narrowing

- Since we have got a postfixpoint $L^{\triangledown}$ of $F \in P \longmapsto P$, its iterates $F^n(L^{\triangledown})$ are all upper approximations of $\mathrm{lfp}\, F$.
- To accelerate convergence of this decreasing chain, we use a narrowing $\triangledown \in P \times P \longmapsto P$ on the poset $\langle P,\ \sqsubseteq\rangle$ satisfying:
  - $\forall x, y \in P : y \sqsubseteq x \Longrightarrow y \sqsubseteq x \vartriangle y \sqsubseteq x$
  - For all decreasing chains $x^0 \sqsupseteq x^1 \sqsupseteq \dots$ the decreasing chain $y^0 \stackrel{\text{def}}{=} x^0, \dots, y^{n+A} \stackrel{\text{def}}{=} y^n \vartriangle x^{n+1}, \dots$ is <u>not</u> strictly decreasing.

Decreasing iteration sequence with narrowing

- Let F be a monotonic operator on a poset $\langle P, \sqsubseteq \rangle$;
- Let $\triangle \in P \times P \longmapsto P$ be a narrowing;
- The iteration sequence with narrowing $\triangle$ for F from the postfixpoint P[7] is $Y^n, n \in \mathbb{N}$:
  - $Y^0 = P$
  - $Y^{n+1} = Y^n \quad \text{if } F(X^n) = X^n$
  - $Y^{n+1} = Y^n \triangle F(X^n) \quad \text{if } F(X^n) \neq X^n$

[7] $F(P) \sqsubseteq P$.

Fixpoint upper approximation by narrowing

- Any iteration sequence with narrowing starting from a postfixpoint P of F[8] is decreasing and stationary after finitely many iteration steps;
- if $\text{lfp}^{\sqsubseteq} F$ does exist[9] and $\text{lfp}^{\sqsubseteq} F \sqsubseteq P$ then its limit $L^{\triangle}$ is a fixpoint of F, whence an upper-approximation of the least fixpoint $\text{lfp}^{\sqsubseteq} F$:

$$\text{lfp}^{\sqsubseteq} F \sqsubseteq L^{\triangle} \sqsubseteq P$$

[8] $F(P) \sqsubseteq P$

[9] e.g. if $\langle P, \sqsubseteq, \bot, \cup \rangle$ is a cpo.

Example of narrowing for intervals

if $x \leq x' \leq y' \leq y$ then $[x, y] \triangle [x', y'] =$ `narrow` $x\ y\ x'\ y'$

```
let narrow x y x' y' =
 (if (x = min_int) then x' else x),
   (if (y = max_int) then y' else y) ;;
```

Trivially extended to initialization & interval analysis.

Program analysis example with narrowing

```
labelled program:
--
0: x := 1;
1: y := 1000;
2: while (x < y) do
3:    x := (x + 1)
4: od {((y < x) | (x = y))}
5:
--
iterations with narrowing from:
--
X0 = { x:_O_; y:_O_ }
X1 = { x:[1,1]; y:_O_ }
X2 = { x:[1,1000]; y:[1000,1000] }
X3 = { x:[1,999]; y:[1000,1000] }
X4 = { x:[2,1000]; y:[1000,1000] }
X5 = { x:[1000,1000]; y:[1000,1000] }
--
X0 = { x:_O_; y:_O_ }
X1 = { x:[1,1]; y:_O_ }
narrowing at 2 by { x:[1,1000]; y:[1000,1000] }
X2 = { x:[1,1000]; y:[1000,1000] }
X3 = { x:[1,999]; y:[1000,1000] }
X4 = { x:[2,1000]; y:[1000,1000] }
X5 = { x:[1000,1000]; y:[1000,1000] }
--
stable
```

Widenings and narrowings are not dual

- The iteration with widening starts from below the least fixpoints and stabilizes above;
- The iteration with narrowing starts from above the least fixpoints and stabilizes above;
- In general, widenings and narrowing are not monotonic.

Conclusion

- A very elementary introduction to abstract interpretation;
- For more details, see e.g.

  `http://www.dmi.ens.fr/~cousot`

Improving the precision of widenings/narrowings

- Threshold;
- Widening/narrowing (and stabilization checks) at cut points;
- Computation history-based extrapolation: A simple example:
  - Do not widen/narrow if a component of the system of fixpoint equations was computed for the first time since the last widening/narrowing ;
  - Otherwise, do not widen/narrow the abstract values of variables which were not "assigned to" [10] since the last widening / narrowing.

[10] more precisely which did not appear in abstract equations corresponding to an assignment to these variables.