« Mathematical foundations: (7) Approximation »

Patrick Cousot
Jerome C. Hunsaker Visiting Professor
Massachusetts Institute of Technology
Department of Aeronautics and Astronautics
cousot@mit.edu
www.mit.edu/~cousot

Course 16.399: "Abstract interpretation"

http://web.mit.edu/afs/athena.mit.edu/course/16/16.399/www/

THESE

présentée à

Université Scientifique et Médicale de Grenoble
Institut National Polytechnique de Grenoble

pour obtenir le grade de
DOCTEUR ES SCIENCES MATHEMATIQUES

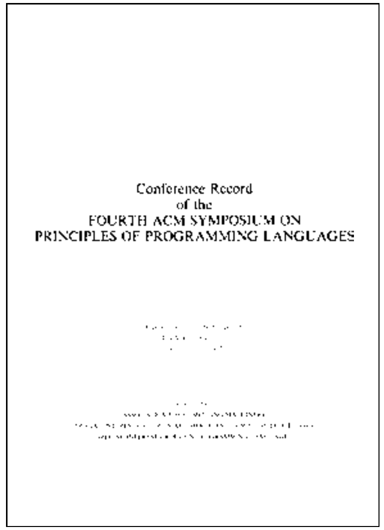
par

Patrick COUSOT

METHODES ITERATIVES DE CONSTRUCTION ET D'APPROXIMATION DE POINTS FIXES D'OPERATEURS MONOTONES SUR UN TREILLIS, ANALYSE SEMANTIQUE DES PROGRAMMES.

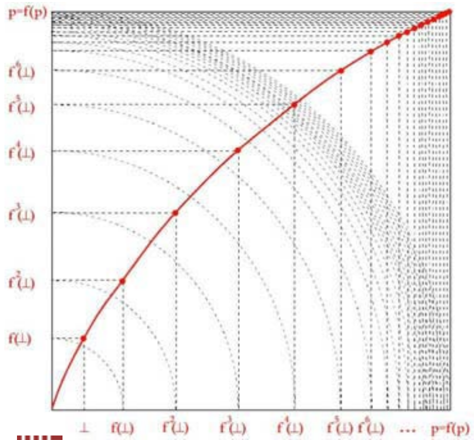
Thèse soutenue le 21 mars 1978 devant la Commission d'Examen :

Président : L. BOLLIET
Examinateurs : C. BENZAKEN, Ph. JORRAND, B. LORHO, C. PAIR, F. ROBERT, M. SINTZOFF



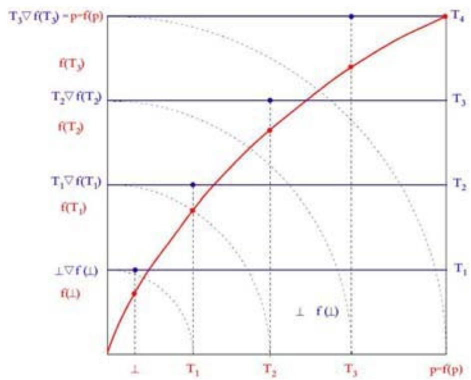
Conference Record
of the
FOURTH ACM SYMPOSIUM ON
PRINCIPLES OF PROGRAMMING LANGUAGES

Intuition for the iterative fixpoint computation of monotone/extensive operators (in general non convergent)



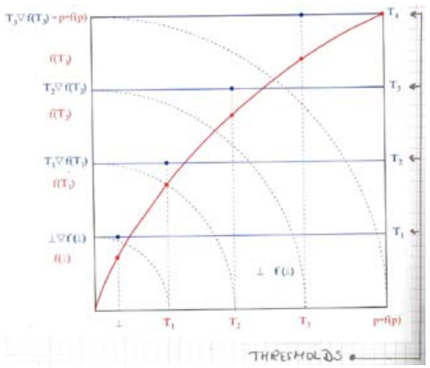
In general the iterates $\bot$, $f(\bot)$, ..., $f^n(\bot)$, ... are not convergent or converge mathematically in infinitely many steps.

Intuition for the iterative fixpoint computation with convergence acceleration by widening of monotone/extensive operators (with overapproximation)



The convergence of $\bot$, $f(\bot), \ldots, f^n(\bot), \ldots$ is accelerated as $x^0 \stackrel{\text{def}}{=} \bot, \ldots, x^{n+1} \stackrel{\text{def}}{=} x^x \,\nabla\, f^n(\bot), \ldots$ using a widening ∇.

Example of widening using finitely many thresholds



In this example, the widening is defined using thresholds $\{T_1, T_2, \ldots, T_n\}$ in finite number such that

- The thresholds include the supremum;
- $x \nabla y$ is the least threshold T_i greater than or equal to both x and y.

Approximate fixpoint abstraction

À la Kleene, Galois connection based, continuous transformer, fixpoint approximation

If
- $L(\sqsubseteq, \bot, \top, \sqcup, \sqcap)$ is a complete lattice;
- $F \in L \stackrel{\text{uc}}{\longmapsto} L$ is continuous for $\sqsubseteq$;
- $L^\sharp(\sqsubseteq^\sharp, \bot^\sharp, \top^\sharp, \sqcup^\sharp, \sqcap^\sharp)$ is a complete lattice;
- $L \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} L^\sharp$ is a Galois connection;
- $F^\sharp \in L^\sharp \mapsto L^\sharp$ is continuous for $\sqsubseteq^\sharp$;
- $\alpha \circ F \circ \gamma \sqsubseteq^\sharp F^\sharp$;

then:

$$\mathsf{lfp}\ F \sqsubseteq \gamma(\mathsf{lfp}\ F^\sharp)$$

PROOF. 1. α is monotonic:

$f \sqsubseteq g$	[hypothesis]
$\Longrightarrow f \sqsubseteq g \ \wedge\ \alpha(g) \sqsubseteq \alpha(g)$	[reflexivity]
$\Longrightarrow f \sqsubseteq g \ \wedge\ g \sqsubseteq \gamma(\alpha(g))$	[Galois connection]
$\Longrightarrow f \sqsubseteq \gamma(\alpha(g))$	[transitivity]
$\Longrightarrow \alpha(f) \sqsubseteq^\sharp \alpha(g)$	[Galois connection]

2.

$\bot \sqsubseteq \gamma(\bot^\sharp)$	[infimum]
$\Longrightarrow F^0(\bot) \sqsubseteq \gamma({F^\sharp}^0(\bot^\sharp))$	[def. iterates]
$\Longrightarrow \alpha(F^0(\bot)) \sqsubseteq^\sharp {F^\sharp}^0(\bot^\sharp)$	[Galois connection]

3. $\alpha(F^n(\bot)) \sqsubseteq^\sharp F^{\sharp^n}(\bot^\sharp)$ [induction hypothesis]

$\Longrightarrow F^n(\bot) \sqsubseteq \gamma(F^{\sharp^n}(\bot^\sharp))$ [Galois connection]

$\Longrightarrow F(F^n(\bot)) \sqsubseteq F \circ \gamma(F^{\sharp^n}(\bot^\sharp))$ [F monotonic]

$\Longrightarrow \alpha(F(F^n(\bot))) \sqsubseteq^\sharp \alpha \circ F \circ \gamma(F^{\sharp^n}(\bot^\sharp))$ [α monotonic 1.]

$\alpha \circ F \circ \gamma(F^{\sharp^n}(\bot^\sharp)) \sqsubseteq^\sharp F^\sharp(F^{\sharp^n}(\bot^\sharp))$ [hypothesis]

$\Longrightarrow \alpha(F(F^n(\bot))) \sqsubseteq^\sharp F^\sharp(F^{\sharp^n}(\bot^\sharp))$ [transitivity]

$\Longrightarrow \alpha(F^{n+1}(\bot)) \sqsubseteq^\sharp F^{\sharp^{n+1}}(\bot^\sharp)$ [def. iterates]

4. $\forall n : \alpha(F^n(\bot)) \sqsubseteq^\sharp F^{\sharp^n}(\bot^\sharp)$ [2., 3., recurrence]

$\Longrightarrow \forall n : \alpha(F^n(\bot)) \sqsubseteq^\sharp \sqcup^\sharp_{m\geq 0} F^{\sharp^m}(\bot^\sharp)$ [lub]

$\Longrightarrow \alpha(F^0(\bot)) \sqsubseteq^\sharp F^{\sharp^0}(\bot^\sharp)$ [Galois connection]

$\Longrightarrow \forall n : \alpha(F^n(\bot)) \sqsubseteq^\sharp \mathsf{lfp}\, F^\sharp$ [Tarski constructive th.]

$\Longrightarrow \forall n : F^n(\bot) \sqsubseteq^\sharp \gamma(\mathsf{lfp}\, F^\sharp)$ [Galois connection]

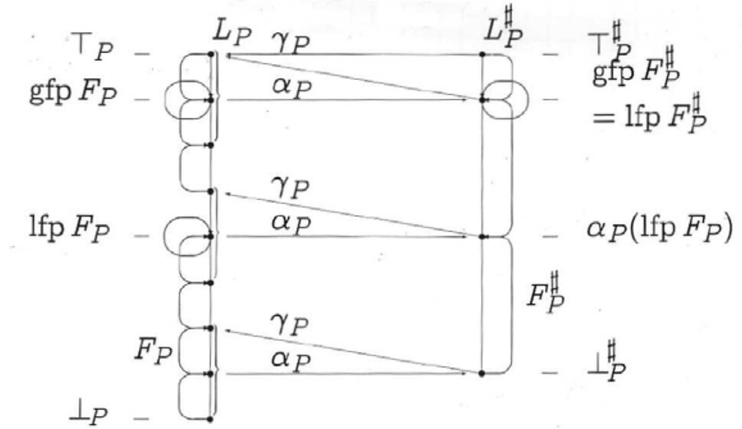
$\Longrightarrow \sqcup_{n\geq 0} F^n(\bot) \sqsubseteq^\sharp \gamma(\mathsf{lfp}\, F^\sharp)$ [least upper bound]

$\Longrightarrow \mathsf{lfp}\,(F) \sqsubseteq^\sharp \gamma(\mathsf{lfp}\, F^\sharp)$ [Tarski constructive th.]

□

Note: we need $\alpha \circ F \circ \gamma(X) \sqsubseteq^\sharp F^\sharp(X)$ only when $X = F^{\sharp^n}(\bot^\sharp)$, $n \in \mathbb{N}$ and so we can relax the hypothesis and assume e.g. $\forall X \sqsubseteq^\sharp \mathsf{lfp}\, F^\sharp : \alpha \circ F \circ \gamma(X) \sqsubseteq^\sharp F^\sharp(X)$

Example:



À la Kleene, Galois connection, monotone transformer-based fixpoint approximation

If
- $\langle L, \sqsubseteq, \sqcup\rangle$ is a cpo
- $F \in L \stackrel{m}{\longmapsto} L$ is monotonic for $\sqsubseteq$
- $a \in L$ is a prefixpoint of F, i.e.: $a \sqsubseteq F(a)$
- $\langle \overline{L}, \overline{\sqsubseteq}, \overline{\sqcup}\rangle$ is a cpo
- $\overline{F} \in \overline{L} \stackrel{m}{\longmapsto} \overline{L}$ is monotonic for $\overline{\sqsubseteq}$
- $\langle L, \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \overline{L}, \overline{\sqsubseteq}\rangle$ is a Galois connection
- $[\forall y \in \overline{L} : y \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)}\, \overline{F} \Longrightarrow \alpha \circ F \circ \gamma(y) \mathrel{\overline{\sqsubseteq}} \overline{F}(y)]$

$\Longleftrightarrow [\forall x \in L : \alpha(x) \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)}\, \overline{F} \Longrightarrow F(x) \sqsubseteq \gamma \circ \overline{F} \circ \alpha(x)]$

$$\Longleftrightarrow \quad [\forall x \in L : \alpha(x) \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F} \Longrightarrow \alpha \circ F(x) \mathrel{\overline{\sqsubseteq}} \overline{F} \circ \alpha(x)]$$

$$\Longleftrightarrow \quad [\forall y \in \overline{L} : y \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F} \Longrightarrow F \circ \gamma(y) \sqsubseteq \gamma \circ \overline{F}(y)]$$

then $\quad \mathsf{lfp}^{\sqsubseteq}_{a} F \sqsubseteq \gamma(\mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F})$

PROOF. – The equivalence of the different statements of overapproximation of F by $\overline{F}$ can be proved as follows:

$\forall y \in \overline{L} : y \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F} \Longrightarrow \alpha \circ F \circ \gamma(y) \mathrel{\overline{\sqsubseteq}} \overline{F}(y)$

$\Longrightarrow \forall x \in L : \alpha(x) \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F} \Longrightarrow \alpha \circ F \circ \gamma(\alpha(x)) \mathrel{\overline{\sqsubseteq}} \overline{F}(\alpha(x))$ $\wr$by letting $y = \alpha(x)$$\int$

$\Longrightarrow \forall x \in L : \alpha(x) \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F} \Longrightarrow \alpha \circ F(x) \mathrel{\overline{\sqsubseteq}} \overline{F} \circ \alpha(x)$ $\wr\gamma \circ \alpha$ is extensive, F and α are monotone, def. composition $\circ\int$

$\Longrightarrow \forall x \in L : \alpha(x) \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F} \Longrightarrow F(x) \sqsubseteq \gamma \circ \overline{F} \circ \alpha(x)$ $\wr$def. Galois connection$\int$

$\Longrightarrow \forall y \in \overline{L} : \alpha(\gamma(y)) \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F} \Longrightarrow F(\gamma(y)) \sqsubseteq \gamma \circ \overline{F} \circ \alpha(\gamma(y))$ $\wr$by letting $x = \gamma(y)$$\int$

$\Longrightarrow \forall y \in \overline{L} : y \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F} \Longrightarrow F(\gamma(y)) \sqsubseteq \gamma \circ \overline{F} \circ \alpha(\gamma(y))$ $\wr$since $\alpha \circ \gamma$ is reductive in a Galois connection and so $y \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F}$ implies $\alpha(\gamma(y)) \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F}$ by transitivity$\int$

$\Longrightarrow \forall y \in \overline{L} : y \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F} \Longrightarrow F \circ \gamma(y) \sqsubseteq \gamma \circ \overline{F} \circ \alpha \circ \gamma(y)$ $\wr$def. composition $\circ\int$

$\Longrightarrow \forall y \in \overline{L} : y \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F} \Longrightarrow F \circ \gamma(y) \sqsubseteq \gamma \circ \overline{F}(y)$ $\wr\alpha \circ \gamma$ is reductive, γ, $\overline{F}$ monotone, transitivity$\int$

$\Longrightarrow \forall y \in \overline{L} : y \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F} \Longrightarrow \alpha \circ F \circ \gamma(y) \mathrel{\overline{\sqsubseteq}} \alpha \circ \gamma \circ \overline{F}(y)$ $\wr\alpha$ is monotone$\int$

$\Longrightarrow \forall y \in \overline{L} : y \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F} \Longrightarrow \alpha \circ F \circ \gamma(y) \mathrel{\overline{\sqsubseteq}} \overline{F}(y)$ $\wr\alpha \circ \gamma$ is reductive, transitivity$\int$

- We let $\langle F^{\delta}, \delta \in \mathbb{O}\rangle$ be the iterates of F starting from a. By the hypothesis that $\langle L, \sqsubseteq, \sqcup\rangle$ is a cpo, F is monotonic and $a \sqsubseteq F(a)$, they are a well-defined increasing chain and an ordinal ϵ such that $\mathsf{lfp}^{\sqsubseteq}_{a} F = F^{\epsilon}$.
- We let $\langle \overline{F}^{\delta}, \delta \in \mathbb{O}\rangle$ be the iterates of $\overline{F}$ starting from $\alpha(a)$. Observe that $\alpha(a) \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F}$ and so $\alpha \circ F(a) \mathrel{\overline{\sqsubseteq}} \overline{F} \circ \alpha(a)$. We have $\alpha \circ F \mathrel{\dot{\overline{\sqsubseteq}}} \overline{F} \circ \alpha$, $F(a) \sqsupseteq a$ and α is monotone by $\langle L, \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \overline{L}, \overline{\sqsubseteq}\rangle$ and so $\overline{F}(\alpha(a)) \mathrel{\overline{\sqsupseteq}} \alpha(F(a)) \mathrel{\overline{\sqsupseteq}} \alpha(a)$ proving $\alpha(a) \sqsubseteq \overline{F}(a)$.
  $\alpha(a)$ is a prefixpoint of the monotone operator $\overline{F}$ on the cpo $\langle \overline{L}, \overline{\sqsubseteq}, \overline{\sqcup}\rangle$ proving, as shown in the constructive version of Tarski's fixpoint theorem, that they are a well-defined increasing chain and an ordinal ϵ' such that $\mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F} = \overline{F}^{\epsilon'}$ [1].
- We have
  - $F^{0} = a \sqsubseteq \gamma \circ \alpha(a) = \gamma(\overline{F}^{0})$

[1] Indeed $\langle L, \sqsubseteq, \sqcup\rangle$ and $\langle \overline{L}, \overline{\sqsubseteq}, \overline{\sqcup}\rangle$ need only be $\max(\epsilon, \epsilon')$-cpos.

- If $F^{\delta} \sqsubseteq \gamma(\overline{F}^{\delta})$ by induction hypothesis, then $\overline{F}^{\delta} \mathrel{\overline{\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F}$ and so $F^{\delta+1} = F(F^{\delta}) \sqsubseteq F \circ \gamma(\overline{F}^{\delta}) \sqsubseteq \gamma \circ \overline{F}(\overline{F}^{\delta}) = \gamma(\overline{F}^{\delta+1})$
- If λ is a limit ordinal and, by induction hypothesis, $\forall \delta < \lambda : F^{\delta} \sqsubseteq \gamma(\overline{F}^{\delta})$,

then $\alpha(F^{\delta}) \mathrel{\overline{\sqsubseteq}} \overline{F}^{\delta}$ and so

$$\alpha(F^{\lambda}) = \alpha(\bigsqcup_{\beta<\lambda} F^{\beta}) = \overline{\bigsqcup_{\beta<\lambda}} \alpha(F^{\beta}) \quad \wr\text{since } \alpha \text{ is a complete join morphism}\int$$

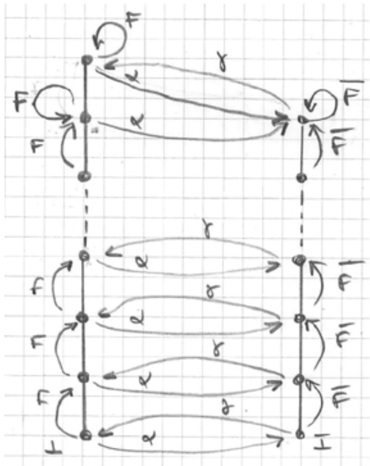
$$\mathrel{\overline{\sqsubseteq}} \overline{\bigsqcup_{\beta<\lambda}} \overline{F}^{\beta} = \overline{F}^{\lambda}$$

and so $F^{\lambda} \sqsubseteq \gamma(F^{\lambda})$.

- By transfinite induction, $\forall \delta \in \mathbb{O} : F^{\delta} \sqsubseteq \gamma(\overline{F}^{\delta})$.
- Finally, $\mathsf{lfp}^{\sqsubseteq}_{a} F = F^{\epsilon} = F^{\max(\epsilon,\epsilon')} \sqsubseteq \gamma(F^{\max(\epsilon,\epsilon')} = \gamma(F^{\epsilon'}) = \gamma(\mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)} \overline{F})$.

□

Example:



These theorems are used (respectively for continuous and monotone functions) in presence of best approximation when α selects the best possible abstraction.

Soundness and (in-)completeness of abstractions

To prove $\mathsf{lfp}_a^{\sqsubseteq} F \sqsubseteq P$, where the fixpoint or invariants are uncomputable, we must overapproximate $\mathsf{lfp}_a^{\sqsubseteq} F$ and underapproximate P. Since in practice this is very hard in non-trivial cases, we choose the abstract domain $\overline{L}$ to be expressive enough to express the properties $P = \gamma(\overline{P})$ to be proved. By the previous theorems, we get:

– Soundness:

$$\mathsf{lfp}_{\alpha(a)}^{\overline{\sqsubseteq}} \overline{F} \mathrel{\overline{\sqsubseteq}} \overline{P} \Longrightarrow \mathsf{lfp}_a^{\sqsubseteq} F \sqsubseteq \gamma(\overline{P})$$

We have also seen previously that the additional commutation condition $\overline{F} \circ \alpha = \alpha \circ F$ implies $\overline{F} = \alpha \circ F \circ \gamma$, $\alpha(\mathsf{lfp}_a^{\sqsubseteq} F) = \mathsf{lfp}_{\alpha(a)}^{\overline{\sqsubseteq}} \overline{F}$ and $\epsilon' \leq \epsilon$, so, in that case, we have

– Completeness:

$$\mathsf{lfp}_a^{\sqsubseteq} F \sqsubseteq \gamma(\overline{P}) \Longrightarrow \mathsf{lfp}_{\alpha(a)}^{\overline{\sqsubseteq}} \overline{F} \mathrel{\overline{\sqsubseteq}} \overline{P}$$

In case of incompleteness, the only way to get more precise abstractions is therefore to refine the abstract transformer $\overline{F}$ (choosing $\overline{F} = \alpha \circ F \circ \gamma$ instead of $\overline{F} \mathrel{\dot{\overline{\sqsupset}}} \alpha \circ F \circ \gamma$) and otherwise to refine the abstraction α.

À la Kleene, continuous abstraction function-based fixpoint approximation

If
- $\langle L, \sqsubseteq, \sqcup \rangle$ is a cpo
- $F \in L \overset{m}{\longmapsto} L$ is monotonic for $\sqsubseteq$
- $a \in L$ is a prefixpoint of F, i.e.: $a \sqsubseteq F(a)$
- $\langle \overline{L}, \overline{\sqsubseteq}, \overline{\sqcup} \rangle$ is a cpo
- $\overline{F} \in \overline{L} \overset{m}{\longmapsto} \overline{L}$ is monotonic for $\overline{\sqsubseteq}$
- $\alpha \in L \overset{uc}{\longmapsto} \overline{L}$ is *upper-continuous*
- $\alpha \circ F \mathrel{\dot{\overline{\sqsubseteq}}} \overline{F} \circ \alpha$

then

$$\alpha(\mathsf{lfp}_a^{\sqsubseteq} F) \mathsf{lfp}_{\alpha(a)}^{\overline{\sqsubseteq}} \overline{F}$$

PROOF. – We let $\langle F^\delta,\ \delta \in \mathbb{O}\rangle$ be the iterates of F starting from a. By the hypothesis that $\langle L,\ \sqsubseteq,\ \sqcup\rangle$ is a cpo, F is monotonic and $a \sqsubseteq F(a)$, they are a well-defined increasing chain and there exists an ordinal ϵ such that $\mathsf{lfp}^{\sqsubseteq}_a\ F = F^\epsilon$.

– We let $\langle \overline{F}^\delta,\ \delta \in \mathbb{O}\rangle$ be the iterates of $\overline{F}$ starting from $\alpha(a)$. We have $\overline{F}(\alpha(a)) \mathrel{\overline{\sqsupseteq}} \alpha(F(a)) \mathrel{\overline{\sqsupseteq}} \alpha(a)$ since $\alpha \circ F \mathrel{\dot{\overline{\sqsubseteq}}} \overline{F} \circ \alpha$, $F(a) \sqsupseteq a$ and α is upper-continuous whence monotone. So $\alpha(a)$ is a prefixpoint of the monotonic operator $\overline{F}$ on the cpo $\langle \overline{L},\ \overline{\sqsubseteq},\ \overline{\sqcup}\rangle$ proving, as shown in the constructive version of Tarski's fixpoint theorem, that they are a well-defined increasing chain and there exists an ordinal ϵ' such that $\mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)}\ \overline{F} = \overline{F}^{\epsilon'}$ [2].

– We have

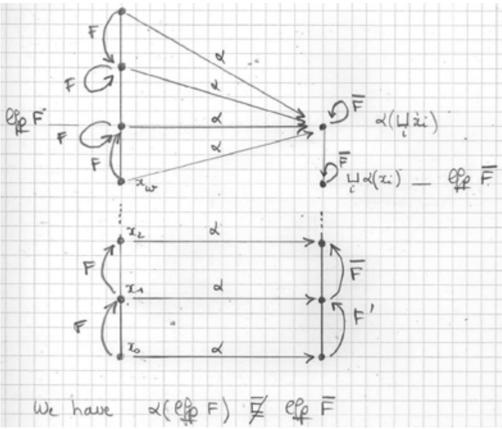
- $\alpha(F^0) = \alpha(a) = \overline{F}^0$
- If $\alpha(F^\delta) \mathrel{\overline{\sqsubseteq}} \overline{F}^\delta$ by induction hypothesis, then
  $\alpha(F^{\delta+1}) = \alpha(F(F^\delta)) \mathrel{\overline{\sqsubseteq}} \overline{F}(\alpha(F^\delta)) \mathrel{\overline{\sqsubseteq}} \overline{F}(\overline{F}^\delta) = \overline{F}^{\delta+1}$

[2] Again, $\langle L,\ \sqsubseteq,\ \sqcup\rangle$ and $\langle \overline{L},\ \overline{\sqsubseteq},\ \overline{\sqcup}\rangle$ need only be $\max(\epsilon, \epsilon')$-cpos.

- If λ is a limit ordinal and, by induction hypothesis, $\forall \delta < \lambda : \alpha(F^\delta) \mathrel{\overline{\sqsubseteq}} \overline{F}^\delta$ then
  $\alpha(F^\lambda) = \alpha(\bigsqcup_{\beta<\lambda} F^\beta) = \overline{\bigsqcup_{\beta<\lambda}} \alpha(F^\beta)$ ≀since $\langle F^\delta,\ \delta < \lambda\rangle$ is an increasing chain and α is a upper-continuous≀
  $\mathrel{\overline{\sqsubseteq}} \overline{\bigsqcup_{\beta<\lambda}} \overline{F}^\beta = \overline{F}^\lambda$
- By transfinite induction, $\forall \delta \in \mathbb{O} : \alpha(F^\delta) \mathrel{\overline{\sqsubseteq}} \overline{F}^\delta$.
- In conclusion, $\alpha(\mathsf{lfp}^{\sqsubseteq}_a\ F) = \alpha((F^\epsilon) = \alpha(F^{\max(\epsilon,\epsilon')}) \mathrel{\overline{\sqsubseteq}} \overline{F}^{\max(\epsilon,\epsilon')} = \overline{F}^{\epsilon'} = \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)}\ \overline{F}$.

□

A counter-example showing the continuity of the abstraction function is necessary



We have $\alpha(\mathsf{lfp}^{\sqsubseteq}_a\ F) \mathrel{\overline{\not\sqsubseteq}} \mathsf{lfp}^{\overline{\sqsubseteq}}_{\alpha(a)}\ \overline{F}$ where $a = \bot$ and $\alpha(a) = \overline{\bot}$.

This theorem is used in absence of best approximation when α selects among possible (minimal) abstractions

À la Kleene, monotone concretization-based fixpoint approximation

If
- $\langle L,\ \sqsubseteq,\ \sqcup\rangle$ is a cpo
- $F \in L \stackrel{m}{\longmapsto} L$ is monotonic for $\sqsubseteq$
- $\langle \overline{L},\ \overline{\sqsubseteq},\ \overline{\sqcup}\rangle$ is a cpo
- $\overline{F} \in \overline{L} \stackrel{m}{\longmapsto} \overline{L}$ is monotonic for $\overline{\sqsubseteq}$
- $\overline{a} \in \overline{L}$ is a prefixpoint of $\overline{F}$, i.e.: $\overline{a} \mathrel{\overline{\sqsubseteq}} \overline{F}(\overline{a})$
- $\gamma \in \overline{L} \stackrel{m}{\longmapsto} L$ is monotonic
- $\gamma(\overline{a})$ is a prefixpoint of $\overline{a}$, i.e. $\gamma(\overline{a}) \sqsubseteq F(\gamma(\overline{a}))$
- $F \circ \gamma \mathrel{\dot{\sqsubseteq}} \gamma \circ \overline{F}$

then $\mathsf{lfp}^{\sqsubseteq}_{\gamma(\overline{a})}\ F \sqsubseteq \gamma(\mathsf{lfp}^{\overline{\sqsubseteq}}_{\overline{a}}\ \overline{F})$

Proof. – Observe that the hypotheses that $\overline{a}$ and $\gamma(\overline{a})$ are respective prefixpoints of $\overline{F}$ and F are independent, as shown by the following examples:

– We let $\langle F^\delta, \delta \in \mathbb{O}\rangle$ be the iterates of F starting from $\gamma(\overline{a})$. By the hypothesis that $\langle L, \sqsubseteq, \sqcup\rangle$ is a cpo, F is monotonic and $\gamma(\overline{a}) \sqsubseteq F(\gamma(\overline{a}))$, they are a well-defined increasing chain and there is an ordinal ϵ such that $\mathsf{lfp}^{\sqsubseteq}_{\gamma(\overline{a})} F = F^\epsilon$.

– We let $\langle \overline{F}^\delta, \delta \in \mathbb{O}\rangle$ be the iterates of $\overline{F}$ starting from $\overline{a}$. By the hypothesis that $\overline{a}$ is a prefixpoint of the monotonic operator $\overline{F}$ on the cpo $\langle \overline{L}, \overline{\sqsubseteq}, \overline{\sqcup}\rangle$, that they are a well-defined increasing chain and, as shown in the constructive version of Tarski's fixpoint theorem, there is an ordinal ϵ' such that $\mathsf{lfp}^{\overline{\sqsubseteq}}_{\overline{a}} \overline{F} = \overline{F}^{\epsilon'}$ [3].

– We have

[3] Once again $\langle L, \sqsubseteq, \sqcup\rangle$ and $\langle \overline{L}, \overline{\sqsubseteq}, \overline{\sqcup}\rangle$ need only be $\max(\epsilon, \epsilon')$-cpos.

– $F^0 = \gamma(\overline{a}) \sqsubseteq \gamma(\overline{a}) = \overline{F}^0$

– If $F^\delta \sqsubseteq \gamma(\overline{F}^\delta)$ by induction hypothesis, then
$F^{\delta+1} = F(F^\delta) \sqsubseteq F \circ \gamma(\overline{F}^\delta) \sqsubseteq \gamma \circ \overline{F}(\overline{F}^\delta) = \gamma(\overline{F}^{\delta+1})$

– If λ is a limit ordinal and, by induction hypothesis, $\forall \delta < \lambda : F^\delta \sqsubseteq \gamma(\overline{F}^\delta)$, then

$$F^\lambda = \bigsqcup_{\beta<\lambda} F^\beta \sqsubseteq \bigsqcup_{\beta<\lambda} \gamma(\overline{F}^\delta) \sqsubseteq \gamma(\overline{\bigsqcup_{\beta<\lambda}} \overline{F}^\delta) \quad \text{(since } \gamma \text{ is monotone)}$$

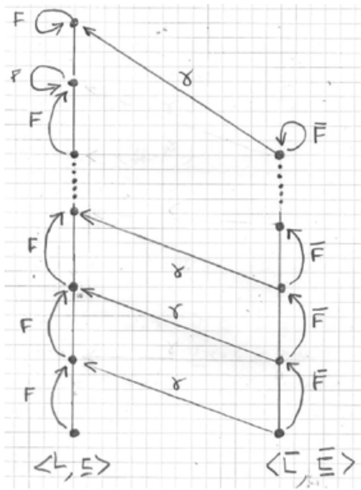
$= \gamma(\overline{F}^\lambda)$ and so $F^\lambda \sqsubseteq \gamma(\overline{F}^\lambda)$.

– By transfinite induction, $\forall \delta \in \mathbb{O} : F^\delta \sqsubseteq \gamma(\overline{F}^\delta)$.

– In conclusion, $\mathsf{lfp}^{\sqsubseteq}_{\gamma(\overline{a})} F = F^\epsilon = F^{\max(\epsilon,\epsilon')} \sqsubseteq \gamma(F^{\max(\epsilon,\epsilon')}) = \gamma(F^{\epsilon'}) = \gamma(\mathsf{lfp}^{\overline{\sqsubseteq}}_{\overline{a}} \overline{F})$.

□

Example:



γ monotonic, $F \circ \gamma \mathrel{\dot{\sqsubseteq}} \gamma \circ \overline{F}$

This theorem is used in absence of best approximation, when a concretization function is only available (e.g. polyhedral analysis, Cousot & Halbwachs, POPL 1978).

À la Tarski, abstraction function-based fixpoint approximation

If $\langle \mathcal{D}^\natural, \sqsubseteq^\natural, \perp^\natural, \sqcup^\natural\rangle$ and $\langle \mathcal{D}^\sharp, \sqsubseteq^\sharp, \perp^\sharp, \sqcup^\sharp\rangle$ are complete lattices, $F^\natural \in \mathcal{D}^\natural \overset{m}{\longmapsto} \mathcal{D}^\natural$, $F^\sharp \in \mathcal{D}^\sharp \overset{m}{\longmapsto} \mathcal{D}^\sharp$ are monotonic and

– α is monotonic (a)

– $\forall y \in \mathcal{D}^\sharp : F^\sharp(y) \sqsubseteq^\sharp y$
$\Longrightarrow \exists x \in \mathcal{D}^\natural : \alpha(x) \sqsubseteq^\sharp y \wedge F^\natural(x) \sqsubseteq^\natural x$ (b)

then

$$\alpha(\mathsf{lfp}^{\sqsubseteq^\natural} F^\natural) \sqsubseteq^\sharp \mathsf{lfp}^{\sqsubseteq^\sharp} F^\sharp$$

Proof.

$$
\begin{array}{lll}
& \alpha(\mathsf{lfp}^{\sqsubseteq^\natural} F^\natural) & \\
= & \alpha(\bigsqcap^\natural\{x \in \mathcal{D}^\natural \mid F^\natural(x) \sqsubseteq^\natural x\}) & \wr \text{Tarski} \wr \\
\sqsubseteq^\sharp & \bigsqcap^\sharp\{\alpha(x) \mid x \in \mathcal{D}^\natural \wedge F^\natural(x) \sqsubseteq^\natural x\} & \wr \alpha \text{ monotone} \wr \\
\sqsubseteq^\sharp & \bigsqcap^\sharp\{y \in \mathcal{D}^\sharp \mid \wedge F^\sharp(y) \sqsubseteq^\sharp y\} & \wr \text{by (b)} \wr \\
= & \mathsf{lfp}^{\sqsubseteq^\sharp} F^\sharp & \wr \text{Tarski} \wr
\end{array}
$$

□

Sufficient conditions for iterative fixpoint computation convergence

– Given a language $\mathcal{L}$, we have seen that program properties can be defined in fixpoint form as

$$\mathsf{lfp}_{\perp[\![P]\!]}^{\sqsubseteq[\![P]\!]} F[\![P]\!]$$

where $F[\![P]\!]$ is a monotone operator on a cpo

$$\langle L[\![P]\!],\ \sqsubseteq[\![P]\!],\ \perp[\![P]\!],\ \sqcup[\![P]\!]\rangle$$

defined by structural induction on the syntactic structure of the program P

– The encoding of $F[\![P]\!]$ is essentially in two forms:
  - as a term, encoded in some data structure, together with an abstract interpreter which, when applied to the term representing $F[\![P]\!]$ and an argument $X \in L[\![P]\!]$ will return $F[\![P]\!](X)$
  - as a function, which can be directly applied to an argument $X \in L[\![P]\!]$ (this requires a functional language or code generation and is often called *abstract compilation*)

– A static analyzer is specified by an abstraction:

$$\langle L[\![P]\!],\ \sqsubseteq[\![P]\!]\rangle \underset{\alpha[\![P]\!]}{\overset{\gamma[\![P]\!]}{\leftrightarrows}} \langle \overline{L}[\![P]\!],\ \overline{\sqsubseteq}[\![P]\!]\rangle$$

and an abstract transformer:

$$\overline{F} \mathrel{\dot{\sqsupseteq}} \alpha[\![P]\!] \circ F[\![P]\!] \circ \gamma[\![P]\!]$$

which are both defined compositionally, by induction on the syntactic structure of $P \in \mathcal{L}$.

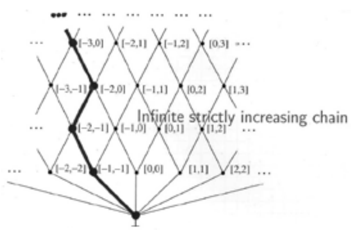
– The static analyzer has the form

```
⟨L[[P]], ⊑[[P]], ⊥[[P]], ⊔[[P]], F̄[[P]]⟩
                          := syntax_analysis(P);
X := ⊥[[P]];
repeat
    Y := F̄[[P]](X);
    stable := Y ⊑[[P]] X;
    X := Y
until stable:
diagnostic(P,X)
```


– Since $\bot[\![P]\!] \sqsubseteq[\![P]\!][\![P]\!] \overline{F}[\![P]\!](\bot[\![P]\!]$ and $\overline{F}[\![P]\!]$ is monotone, the successive values of X form a $\sqsubset[\![P]\!]$ increasing chain (but maybe for the last iterate where equality can hold). The stabilization test implies, if and when the loop exists, that $X = \overline{F}[\![P]\!](X)$. So upon termination, if ever, $X = \mathsf{lfp}^{\sqsubseteq[\![P]\!]}_{\bot[\![P]\!]} \overline{F}[\![P]\!]$ so that we can apply the soundness result.

– As far as termination is concerned, it follows that
  - The iteration terminates if the lattice $\overline{L}$ is finite
  - The iteration terminates if the lattice $\overline{L}$ satisfies the ascending chain condition (ACC).

– However, the iteration may not terminate, or terminate after a huge number of iterations. An example is the abstraction of $\wp(\mathbb{Z})$ by intervals:



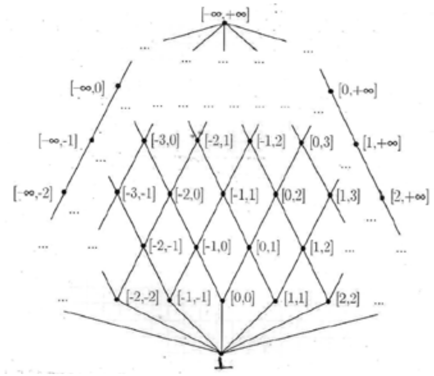
– In this case, one case choose a coarser abstraction α (in an abstract domain satisfying the ACC)

– We will later show that it is preferable to use convergence acceleration by widening/narrowing (the rôle of α is then to ensure that abstract properties have efficient computer representations, while convergence is treated otherwise, by widening/narrowing).

Iteration acceleration by extrapolation

Example of abstraction into a lattice not satisfying the ascending chain condition (ACC)

– The lattice of intervals is



– The lattice of intervals abstracts $\wp(\mathbb{Z})$:

$$\langle \wp(\mathbb{Z}), \subseteq \rangle \underset{\alpha_{\mathbb{I}}}{\overset{\gamma_{\mathbb{I}}}{\leftrightarrows}} \langle \mathbb{I}, \sqsubseteq \rangle$$

where

- $\mathbb{I} \stackrel{\text{def}}{=} \{[a,b] \mid a,b \in \mathbb{Z} \cup \{-\infty, +\infty\} \wedge a \leq b\}$
- $\bot \sqsubseteq \bot \sqsubseteq [a,b]$
- $[a,b] \sqsubseteq [a',b']) \stackrel{\text{def}}{=} (a' \leq a \wedge b \leq b')$
- $\alpha_{\mathbb{I}}(\emptyset) \stackrel{\text{def}}{=} \bot$, when $X \neq \emptyset$, $\alpha_{\mathbb{I}}(X) \stackrel{\text{def}}{=} [\min X, \max X]$ where $\min X = -\infty$ when X has no minimum in $\mathbb{Z}$ and $\max X = +\infty$ when X has no maximum in $\mathbb{Z}$.

– The lattice of intervals provides a classical example of infinite lattice not satisfying the ascending chain condition, for which iterative fixpoint computations may not be convergent

– In practice, one can choose $-\infty$ = `min_int` and $+\infty$ = `max_int` but then the convergence, although always guaranteed is so slow that it cannot be of any practical use, but for programs with very few program variables.

Example of non-convergent iterative fixpoint computation

– Let us consider the program:

```
0: x := 1;
1: while true do
     2: x := (x + 1);
3: od
4:
```

– The states are $\Sigma \stackrel{\text{def}}{=} \{0, 1, 2, 3, 4\} \times [\texttt{min\_int}; \texttt{max\_int}]$

– The abstraction is

$$\alpha_p \in \wp(\Sigma) \mapsto \prod_{i=0}^{4} \wp([\texttt{min\_int}; \texttt{max\_int}])$$

$$\alpha_p(X) \stackrel{\text{def}}{=} \prod_{i=0}^{4} \{x \mid \langle i,\ x\rangle \in X\}$$

$$\alpha \in \wp(\Sigma) \mapsto \prod_{i=0}^{4} \{\bot\} \cup \{[\ell, h] \mid a, b \in \mathbb{Z} \wedge \texttt{min\_int} \leqslant \ell \leqslant h \leqslant \texttt{max\_int}\}$$

$$\alpha(X) \stackrel{\text{def}}{=} \prod_{i=0}^{4} (\!|\, \alpha_p(X)_i = \emptyset \;?\; \bot \;\text{¿}\; [\min \alpha_p(X)_i,\ \max \alpha_p(X)_i] \,|\!)$$

– The abstract reachable state transformer for the interval abstraction (without bounded non-modular arithmetics) can be encoded as a system of equations involving symbolic term (which can e.g. be encoded by their syntax trees)

$$\begin{cases} X0 = \{\texttt{x} \to [\texttt{min\_int}, \texttt{max\_int}]\} \\ X1 = \{\texttt{x} \to [1, 1]\} \,\dot{\sqcup}\, X3 \\ X2 = X1 \,\dot{\sqcap}\, \{\texttt{x} \to [\texttt{min\_int}, \texttt{max\_int}]\} \\ X3 = (\!|\, X2 = \bot \;?\; \bot \;\text{¿}\; \text{let } [a, b] = X2 \text{ in} \\ \qquad\qquad [\min(a+1, \texttt{max\_int}), \min(b+1, \texttt{max\_int})] \,|\!) \\ X4 = X1 \,\dot{\sqcap}\, \{\texttt{x} \to \bot\} \end{cases}$$

– A functional encoding in OCaml could be:

```
type interval = BOT | INT of (int * int);;
let less x y = match x,y with
| BOT, _ -> true
| _, BOT -> false
| INT (a,b), INT (c,d) -> (a<=c)&&(b<=d);;
let join x y = match x,y with
| BOT, _ -> y
| _, BOT -> x
| INT (a,b), INT (c,d) -> INT (min a c,max b d);;
let meet x y = match x,y with
| BOT, _ -> BOT
| _, BOT -> BOT
| INT (a,b), INT (c,d) ->
    if (b<c) or (d<a) then BOT
    else INT (max a c,min b d);;
let f (x0,x1,x2,x3,x4) =
  (INT (min_int,max_int),
   join (INT (1,1)) x3,
   meet x1 (INT (min_int,max_int)),
```

```
   (match x2 with
    | BOT -> BOT
    | INT (a,b) ->
        let a' = if a<max_int then a+1 else max_int in
          let b' = if b<max_int then b+1 else max_int in
            INT (a',b')),
   meet x1 BOT);;
let pless (x0,x1,x2,x3,x4) (x'0,x'1,x'2,x'3,x'4) =
 (less x0 x'0) && (less x1 x'1) && (less x2 x'2) && (less x3 x'3)
  && (less x4 x'4);;
let lfp leq a f =
  let rec iterate x =
    let y = f x in
      if leq y x then x
      else iterate y
  in iterate a;;
lfp pless (BOT,BOT,BOT,BOT,BOT) f;;
```


– After a few hours of computation, the result is:

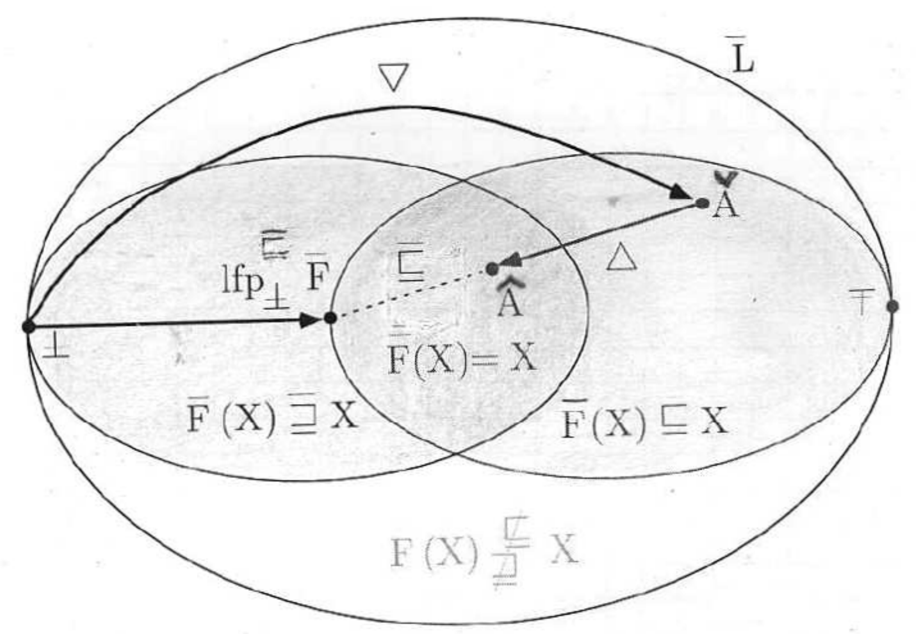
```
- : interval * interval * interval * interval * interval =
(INT (-1073741824, 1073741823), INT (1, 1073741823), INT (1, 1073741823),
 INT (2, 1073741823), BOT)
```


– The chaotic iterates from the infimum $\perp^5$ are as follows

```
X0 = { x:[-1073741824, 1073741823]
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] }
X1 = { x:[1, 5] }
X2 = { x:[1, 5] }
...
X1 = { x:[1, 1073741823] }
```


Intuition for convergence acceleration

1. Speed-up the convergence of the increasing iteration $X^0 = \perp, \ldots, X^{n+1} = F(X^n), \ldots, \breve{A}$ in order to reach a postfixpoint $\breve{A} : F(\breve{A}) \sqsubseteq \breve{A}$ so that by Tarski: $\mathsf{lfp}\, F \sqsubseteq \breve{A}$
   $\leadsto$ WIDENING ∇
2. Speed up the convergence of the decreasing iteration $Y^0 = \breve{A}, \ldots : Y^{n+1} = F(Y^n), \ldots, \hat{A}$ so as to stay <u>above</u> the least fixpoint $\mathsf{lfp}\, F \sqsubseteq \hat{A}$
   $\leadsto$ NARROWING Δ



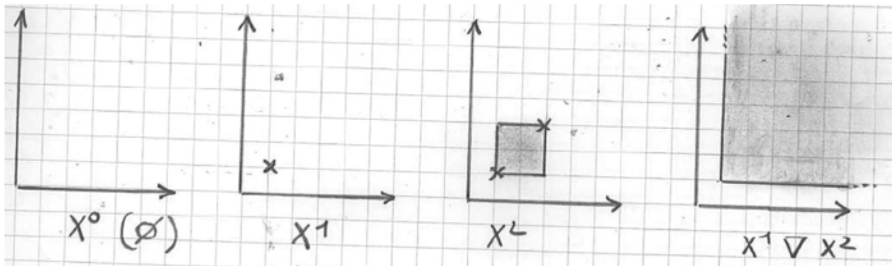
Widening

Example of widening for interval analysis

- $\overline{L} = \{\perp\} \cup \{[\ell, u] \mid \ell \in \mathbb{Z} \cup \{-\infty\} \wedge u \in \mathbb{Z} \cup \{+\infty\} \wedge \ell \leq u\}$
- The widening extrapolates unstable bounds to infinity:

$$\begin{aligned}
\perp \nabla X &= X \\
X \nabla \perp &= X \\
[\ell_0, u_0] \nabla [\ell_1, u_1] &= [(\ell_1 < \ell_0\ ?\ -\infty\ \wr\ \ell_0), \\
&\quad\ (u_1 > u_0\ ?\ +\infty\ \wr\ u_0)]
\end{aligned}$$

- Example:



- Not monotone. For example $[0, 1] \sqsubseteq [0, 2]$ but $[0, 1] \nabla [0, 2] = [0, +\infty] \not\sqsubseteq [0, 2] = [0, 2] \nabla [0, 2]$

Example of upward iteration with widening to upper-approximate a least-fixpoint by a post-fixpoint

– The analysis of the output of the following PROLOG II program:

```
program   -> init(x,1) while(x);
init(x,x) -> ;
while(x)  -> val(inf(x,100),1) out(x) line val(add(x,2),y)\\
             while(y);
```

consists in solving the equation:

$$X = \Big([1,\ 1] \sqcup (X \oplus [2,\ 2])\Big) \sqcap [-\infty,\ 99]$$

where $\emptyset \oplus I = I \oplus \emptyset = \emptyset$ and $[a,\ b] \oplus [c,\ d] = [a \dotplus c,\ b \dotplus d]$ with $-\infty \dotplus x = x \dotplus -\infty = -\infty$ and $+\infty \dotplus x = x \dotplus +\infty = +\infty$.

– Ascending abstract iteration sequence with widening:

$$\begin{aligned}
\hat{X}^0 &= \emptyset \\
\hat{X}^1 &= \hat{X}^0 \nabla \Big(([1,\ 1] \sqcup (\hat{X}^0 \oplus [2,\ 2])) \sqcap [-\infty,\ 99]\Big) \\
&= \emptyset \nabla [1,\ 1] \\
&= [1,\ 1] \\
\hat{X}^2 &= \hat{X}^1 \nabla \Big(([1,\ 1] \sqcup (\hat{X}^1 \oplus [2,\ 2])) \sqcap [-\infty,\ 99]\Big) \\
&= [1,\ 1] \nabla [1,\ 3] \\
&= [1,\ +\infty] \\
\hat{X}^3 &= \hat{X}^2 \nabla \Big(([1,\ 1] \sqcup (\hat{X}^2 \oplus [2,\ 2])) \sqcap [-\infty,\ 99]\Big) \\
&= [1,\ +\infty] \nabla [1,\ 99] \\
&= [1,\ +\infty]
\end{aligned}$$

Definition of a widening

A widening $\nabla \in P \times P \mapsto P$ 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 $y^0 \stackrel{\text{def}}{=} x^0, \ldots, y^{n+1} \stackrel{\text{def}}{=} y^n \nabla x^{n+1}, \ldots$ is <u>not</u> strictly increasing.

Two different main uses:

– Approximate missing lubs.

– Convergence acceleration [4];

[4] A widening operator can be used to effectively compute an upper approximation of the least fixpoint of $\overline{F} \in \overline{L} \xrightarrow{\text{m}} \overline{L}$ starting from below when $\overline{L}$ is computer representable but does not satisfy the ascending chain condition.

Upward iteration with widening

– Let F be an operator on a poset $\langle P,\ \sqsubseteq\rangle$;

– Let $\nabla \in P \times P \mapsto P$ be a widening;

– The iteration sequence with widening ∇ for F from $\bot$ is $X^n, n \in \mathbb{N}$:

- $X^0 = \bot$
- $X^{n+1} = X^n$ if $F(X^n) \sqsubseteq (X^n)$
- $X^{n+1} = X^n \nabla F(X^n)$ if $F(X^n) \not\sqsubseteq X^n$

Correctness of the upward iteration with widening to upper-approximate a least-fixpoint by a post-fixpoint

If
- $L(\sqsubseteq, \bot, \sqcup)$ is a poset,
- $\varphi \in L \stackrel{\text{uc}}{\longmapsto} L$ and
- ∇ is a widening operator

then the increasing chain:
- $\hat{X}^0 = \bot$,
- $\hat{X}^{k+1} = \hat{X}^k$ if $\varphi(\hat{X}^k) \sqsubseteq \hat{X}^k$
- $\hat{X}^{k+1} = \hat{X}^k \nabla \varphi(\hat{X}^k)$ otherwise

for $k \in \mathbb{N}$ is stationary with limit $\hat{X}^\ell$ such that $\mathsf{lfp}\,\varphi \sqsubseteq \hat{X}^\ell$.

Proof. – $\varphi^0 = \bot \sqsubseteq \bot = \hat{X}^0 \wedge \varphi^0 \sqsubseteq \varphi^1$ [$\sqsubseteq$ reflexive, $\bot$ infimum]

– $\varphi^k \sqsubseteq \hat{X}^k \wedge \varphi^k \sqsubseteq \varphi^{k+1}$ [induction hypothesis]

$\Longrightarrow \varphi^{k+1} = \varphi(\varphi^k) \sqsubseteq \varphi(\hat{X}^k) \wedge \varphi^{k+1} \sqsubseteq \varphi^{k+2}$ [monotony]

- if $\varphi(\hat{X}^k) \sqsubseteq \hat{X}^k$ then $\hat{X}^{k+1} = \hat{X}^k$

  $\Longrightarrow \varphi^{k+1} \sqsubseteq \hat{X}^{k+1}$ [transitivity]

- else $\hat{X}^{k+1} = \hat{X}^k \nabla \varphi(\hat{X}^k)$

  $\Longrightarrow \hat{X}^k \sqsubseteq \hat{X}^{k+1} \wedge \varphi^{k+1} \sqsubseteq \hat{X}^{k+1}$ [(b)]

$\Longrightarrow$ the chain $\hat{X}^k$, $k \in \mathbb{N}$ is increasing and [by induction]

(1) $\forall k \in \mathbb{N} : \varphi^k \sqsubseteq \hat{X}^k$.

$\Longrightarrow$ the chain $\varphi(\hat{X}^k)$, $k \in \mathbb{N}$ is increasing [monotony]

$\Longrightarrow$ the chain $\hat{X}^k$, $k \in \mathbb{N}$ is stationary. [(c)]

– For the limit $\hat{X}^\ell$ where $\ell \in \mathbb{N}$, we have:

(2) - $\forall k \leq \ell$: $\hat{X}^k \sqsubseteq \hat{X}^\ell$ [increasing chain]

- $m \geq \ell \wedge \hat{X}^m = \hat{X}^\ell$ [induction hypothesis]

  $\Longrightarrow \hat{X}^{m+1} = \hat{X}^m = \hat{X}^\ell$ [if $\varphi(\hat{X}^m) \sqsubseteq \hat{X}^m$]

  or $\Longrightarrow \hat{X}^{m+1} = \hat{X}^m \nabla \varphi(\hat{X}^m)$ [otherwise]

  $= \hat{X}^\ell \nabla \varphi(\hat{X}^\ell) = \hat{X}^\ell$

  $\Longrightarrow \hat{X}^{m+1} = \hat{X}^\ell$ [by cases]

(3) $\Longrightarrow \forall m \geq \ell : \hat{X}^m = \hat{X}^\ell$ [by induction]

$\Longrightarrow \forall k \in \mathbb{N} : \varphi^k \sqsubseteq \hat{X}^k \sqsubseteq \hat{X}^\ell$ [(1) (2) (3)]

$\Longrightarrow \mathsf{lfp}\,\varphi = \bigsqcup_{k\in\mathbb{N}} \varphi^k \sqsubseteq \hat{X}^\ell$ [Tarski constructive and lubs]

□

The generalization to a monotonic $\varphi \in L \stackrel{\text{m}}{\longmapsto} L$ is straightforward.

In summary:

- Any iteration sequence with widening is increasing and stationary after finitely many iteration steps;
- Its limit F^{∇} is a post-fixpoint of F, whence an upper-approximation of the least fixpoint $\mathsf{lfp}^{\sqsubseteq} F$ [5]:

$$\mathsf{lfp}^{\sqsubseteq} F \sqsubseteq F^{\nabla}$$

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

Example of convergence acceleration of an upward iterative fixpoint computation by widening

- Program:

```
0: x := 1;
2: while (x < 1000) do
     3: x := (x + 1);
4: od {(x >= 1000))}
6:
```


- Forward abstract equations for interval analysis with widening:

$$\begin{cases} X0 = \{\mathtt{x} \to [\mathtt{min\_int}, \mathtt{max\_int}]\} \\ X2 = \{\mathtt{x} \to [1, 1]\} \mathbin{\dot{\sqcup}} X4 \\ X3 = X2 \mathbin{\dot{\sqcap}} \{\mathtt{x} \to [\mathtt{min\_int}, 999]\} \\ X4 = (\!| X3 = \bot \mathrel{?} \bot \mathrel{⁝} \text{let } [a, b] = X3 \text{ in} \\ \qquad\qquad [\min(a + 1, \mathtt{max\_int}), \min(b + 1, \mathtt{max\_int})] |\!) \\ X6 = X2 \mathbin{\dot{\sqcap}} \{\mathtt{x} \to [1000, \mathtt{max\_int}]\} \end{cases}$$

- Iteration with widening from X0 = X2 = X3 = X4 = X6 = {x → ⊥}:

```
X2 = { x:_|_ }
widening at 2 by { x:[1, 1] }
X2 = { x:[1, 1] }
widening at 3 by { x:[1, 1] }
X3 = { x:[1, 1] }
widening at 4 by { x:[2, 2] }
X4 = { x:[2, 2] }
widening at 2 by { x:[1, 2] }
X2 = { x:[1, +oo] }
widening at 3 by { x:[1, 999] }
X3 = { x:[1, +oo] }
widening at 4 by { x:[2, +oo] }
X4 = { x:[2, +oo] }
widening at 6 by { x:[1000, +oo] }
X6 = { x:[1000, +oo] }
stable
```

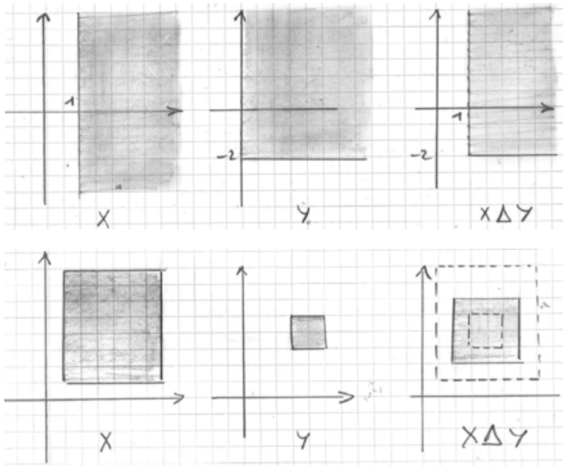
Narrowing

Example of narrowing for interval analysis

– The narrowing improves infinite bounds only:

$$\bot \,\Delta\, X = \bot$$
$$[\ell_0,\, u_0] \,\Delta\, [\ell_1,\, u_1] = [(\ell_0 = -\infty \;?\; \ell_1 \;¿\; \ell_0),\; (u_0 = +\infty \;?\; u_1 \;¿\; u_0)]$$

– Other examples of narrowings:



Example of downward iteration with narrowing to improve a post-fixpoint approximation of a (least) fixpoint

– Equation (contn'd):

$$X = \Big([1,\, 1] \sqcup (X \oplus [2,\, 2])\Big) \sqcap [-\infty,\, 99]$$

where $\emptyset \oplus I = I \oplus \emptyset = \emptyset$ and $[a,\, b] \oplus [c,\, d] = [a \dotplus c,\, b \dotplus d]$ with $-\infty \dotplus x = x \dotplus -\infty = -\infty$ and $+\infty \dotplus x = x \dotplus +\infty = +\infty$.

– Descending abstract iteration sequence with narrowing starting from $\widetilde{X}^3 = [1, +\infty]$:

$$
\begin{aligned}
\check{X}^0 &= \hat{X}^3 \\
&= [1,\ +\infty] \\
\check{X}^1 &= \check{X}^0 \,\Delta\, \Big(([1,\ 1] \sqcup (\check{X}^0 \oplus [2,\ 2])) \sqcap [-\infty,\ 99]\Big) \\
&= [1,\ +\infty] \,\Delta\, [1,\ 99] \\
&= [1,\ 99] \\
\check{X}^2 &= \check{X}^1 \,\Delta\, \Big(([1,\ 1] \sqcup (\check{X}^1 \oplus [2,\ 2])) \sqcap [-\infty,\ 99]\Big) \\
&= [1,\ 99] \,\Delta\, [1,\ 99] \\
&= [1,\ 99]
\end{aligned}
$$

The analysis time does not depend upon the number of iterations in the while-loop.

Definition of the narrowing

– Since we have got a postfixpoint F^{∇} of $F \in P \mapsto P$, its iterates $F^n(F^{\nabla})$ are all upper approximations of $\mathsf{lfp}\ F$.

– To accelerate convergence of this decreasing chain, we use a narrowing $\nabla \in P \times P \mapsto P$ on the poset $\langle P, \sqsubseteq\rangle$ satisfying:

- $\forall x, y \in P : y \sqsubseteq x \Longrightarrow y \sqsubseteq x \,\Delta\, y \sqsubseteq x$
- For all decreasing chains $x^0 \sqsupseteq x^1 \sqsupseteq \ldots$ the decreasing chain $y^0 \stackrel{\text{def}}{=} x^0, \ldots, y^{n+1} \stackrel{\text{def}}{=} y^n \,\Delta\, x^{n+1}, \ldots$ is not strictly decreasing.

Decreasing Iteration Sequence with Narrowing

– Let F be a monotonic operator on a poset $\langle P, \sqsubseteq\rangle$;

– Let $\Delta \in P \times P \mapsto P$ be a narrowing;

– The iteration sequence with narrowing Δ for F from the postfixpoint P[6] is $Y^n, n \in \mathbb{N}$:

- $Y^0 = P$
- $Y^{n+1} = Y^n \quad$ if $F(Y^n) = Y^n$
- $Y^{n+1} = Y^n \,\Delta\, F(Y^n) \quad$ if $F(Y^n) \neq Y^n$

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

Correctness of the downward iteration with narrowing to improve a post-fixpoint approximation of a (least) fixpoint

If

- $L(\sqsubseteq)$ is a poset,
- $\varphi \in L \stackrel{\text{m}}{\longmapsto} L$,
- $\Delta \in L \times L \mapsto L$ is a narrowing operator and
- $\varphi(x) = x \sqsubseteq y,\ \varphi(y) \sqsubseteq y$,

then the decreasing chain:

- $\check{X}^0 = y$,
- $\check{X}^{k+1} = \check{X}^k \,\Delta\, \varphi(\check{X}^k)$

for $k \in \mathbb{N}$ is stationary with limit $\check{X}^\ell, \ell \in \mathbb{N}$ such that $x \sqsubseteq \check{X}^\ell \sqsubseteq y$.

PROOF. – $x \sqsubseteq \check{X}^0$ [hypothesis and transitivity]

– $x \sqsubseteq \check{X}^k$ [induction hypothesis]

$\Longrightarrow x = \varphi(x) \sqsubseteq \varphi(\check{X}^k)$ [monotony]

$\Longrightarrow x \sqsubseteq \check{X}^{k+1} = \check{X}^k \Delta \varphi(\check{X}^k) \sqsubseteq \check{X}^k$ [(e) and (f)]

$\Longrightarrow \forall k \in \mathbb{N} : x \sqsubseteq \check{X}^k$ and [by induction]

the chain $\check{X}^k$, $k \in \mathbb{N}$ is decreasing for $\sqsubseteq$

$\Longrightarrow$ the chain $\varphi(\check{X}^k)$, $k \in \mathbb{N}$ is decreasing for $\sqsubseteq$ [monotony]

$\Longrightarrow \check{X}^k$, $k \in \mathbb{N}$ has a limit $\check{X}^\ell$ [(g)]

$\Longrightarrow x \sqsubseteq \check{X}^\ell \sqsubseteq \check{X}^0 = y$.

□

In summary:

– Any iteration sequence with narrowing starting from a postfixpoint P of F[7] is decreasing and stationary after finitely many iteration steps;

– if $\mathsf{lfp}^{\sqsubseteq} F$ does exist[8] and $\mathsf{lfp}^{\sqsubseteq} F \sqsubseteq P$ then its limit F^{Δ} is a fixpoint of F, whence an upper-approximation of the least fixpoint $\mathsf{lfp}^{\sqsubseteq} F$:

$$\mathsf{lfp}^{\sqsubseteq} F \sqsubseteq F^{\Delta} \sqsubseteq P$$

– The downward iteration sequence can jump over <u>no</u> fixpoint (hence cannot jump over the [unknown] least fixpoint), which ensures that we have an approximation from above.

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

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

Example of convergence acceleration of a downward iteration with narrowing to improve a post-fixpoint approximation of a (least) fixpoint

– Program:

```
0: x := 1;
2: while (x < 1000) do
     3: x := (x + 1);
4: od {(x >= 1000))}
6:
```


– Forward abstract equations for interval analysis with widening:

$$\begin{cases} X0 = \{\mathtt{x} \to [\mathtt{min\_int}, \mathtt{max\_int}]\} \\ X2 = \{\mathtt{x} \to [1,1]\} \dot{\sqcup} X4 \\ X3 = X2 \dot{\sqcap} \{\mathtt{x} \to [\mathtt{min\_int}, 999]\} \\ X4 = (X3 = \bot \mathrel{?} \bot \mathrel{\text{¦}} \text{let } [a,b] = X3 \text{ in} \\ \qquad\qquad [\min(a+1, \mathtt{max\_int}), \min(b+1, \mathtt{max\_int})]) \\ X6 = X2 \dot{\sqcap} \{\mathtt{x} \to [1000, \mathtt{max\_int}]\} \end{cases}$$

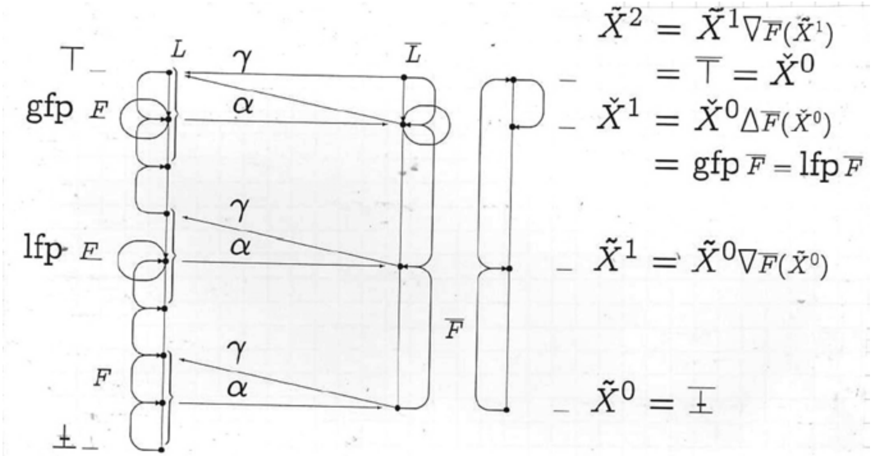
```
iterations with narrowing from:
X0 = { x:_O_ }
X2 = { x:[1,+oo] }
X3 = { x:[1,+oo] }
X4 = { x:[2,+oo] }
X6 = { x:[1000,+oo] }
-
narrowing at 0 by { x:[-oo,+oo]}
X0 = { x:[-oo,+oo]}
narrowing at 2 by { x:[1,+oo] }
X2 = { x:[1,+oo] }
narrowing at 3 by { x:[1,999] }
X3 = { x:[1,999] }
narrowing at 4 by { x:[2,1000] }
X4 = { x:[2,1000] }
narrowing at 6 by { x:[1000,+oo] }
X6 = { x:[1000,+oo] }
narrowing at 0 by { x:[-oo,+oo] }
X0 = { x:[-oo,+oo]}
narrowing at 2 by { x:[1,1000] }
X2 = { x:[1,1000] }
narrowing at 3 by { x:[1,999] }
X3 = { x:[1,999] }
narrowing at 4 by { x:[2,1000] }
X4 = { x:[2,1000] }
narrowing at 6 by { x:[1000,1000] }
X6 = { x:[1000,1000] }
stable
```

– Obviously narrowing at each program point can be replaced by a narrowing at loop heads (see later). Was the same for widenings.

Static Analysis with Widening/Narrowing

Iteration convergence acceleration

– Intuition:



– A non-trivial example of automatic interval analysis with widening/narrowing:

```
function F(X : integer) : integer;
begin
   if X > 100 then begin
      F := X - 10
      { X:101..maxint & F:91..maxint - 10 }
   end else begin
      F := F(F(X + 11))
      { X:minint..100 & F = 91 }
   end;
end;
```

(simple ideas can be effective but in general more refined widenings should be used, as shown later).

On fixpoint approximation using widening/narrowing operators

- The approximation is done a priori, once for all ($L \stackrel{\gamma}{\underset{\alpha}{\leftrightarrows}} \overline{L}$, ∇ and Δ).
- The approximation α may be precise while ∇ may be very rough.
- Usefulness of the approximation is shown by experience (precision/cost can be tuned with ∇).
- The approximation is applied at each iteration step for $\overline{F}$.
- The approximation depends upon the iterates.

- $\overline{L}$ need not satisfy the ascending chain condition (since ∇ will be used to enforce convergence).

Schema of a static program analyzer with widening/narrowing

$\langle \overline{\bot}, \overline{F} \rangle$:= syntactic_analysis(Program);
%% $\alpha(\bot) \sqsubseteq \overline{\bot} \wedge \alpha \circ F \circ \gamma \sqsubseteq \overline{F}$
$X := \overline{\bot}$;
repeat
$\quad Y := X$;
$\quad X := \overline{F}(X)$;
$\quad$ **if** $X \sqsubseteq Y$ **then** $C :=$ true
$\quad$ **else** $C :=$ false; $X := Y \mathbin{\nabla} X$ **fi**
until C;
%% $\mathsf{lfp}^{\overline{\bot}}\, \overline{F} \sqsubseteq X = \overline{F}(Y) \sqsubseteq Y \wedge \mathsf{lfp}^{\bot}\, F \sqsubseteq \gamma(Y)$

%% $\mathsf{lfp}^{\overline{\bot}}\, \overline{F} \sqsubseteq X = \overline{F}(Y) \sqsubseteq Y$
while $X \neq Y$ **do**
$\quad Y := Y \mathbin{\Delta} X$;
$\quad X := \overline{F}(Y)$
od;
%% $\mathsf{lfp}^{\overline{\bot}}\, \overline{F} \sqsubseteq \overline{F}(X) = X \wedge \mathsf{lfp}^{\bot}\, F \sqsubseteq \gamma(X)$

In practice, chaotic or asynchronous iterations (with memory).

Galois-connection based static program analyzer with widening/narrowing

- The Galois connection approach is the basic method of abstract interpretation.
- With its variants (e.g. concretization function only in absence of best approximation), its always applicable to a poset with ACC;
- However, combination with the widening/narrowing is the key to success:
  - Rich domain of information (whence not satisfying the ACC),
  - Convergence acceleration.
- In practice, a much better compromise than just weakening the expressiveness of the abstract domain using a coarser Galois connection.

Properties of Widening/Narrowing

Widening/narrowing are not dual; Dual widening/narrowing

- The iteration with widening starts from below the least fixpoint and stabilizes above to a postfixpoint;
- The iteration with narrowing starts from above the least fixpoint and stabilizes above;
- The iteration with dual widening starts from above the greatest fixpoint and stabilizes below to a prefixpoint;
- The iteration with dual narrowing starts from below the greatest fixpoint and stabilizes below;

	Iteration starts from	Iteration stabilizes
Widening ∇	below	above
Narrowing Δ	above	above
Dual widening $\widetilde{\nabla}$	above	below
Dual narrowing $\widetilde{\Delta}$	below	below

Whence that's four different notions.

An example of static analysis of a simple program for automatic determination of interval invariant by fixpoint approximation with convergence acceleration by widening/narrowing

```
program P;
    var I : integer;
begin
{1:}
    I := 1;
{2:}
    while { I ∈ X } I <= 100 do begin
{3:}
        I := I + 2;
{4:}
    end;
{5:} { I ∈ Y }
end.
```


– Interval equations:
  - $X = [1, 1] \cup ((X \cap [-\infty, 100]) \mp [2, 2])$
  - $Y = X \cap [101, +\infty]$

– Upwards iteration from the infimum without widening

$$
\begin{array}{llll}
X^0 = \bot & & Y^0 = \bot \\
X^1 = [1, 1] & & Y^1 = \bot \\
X^2 = [1, 3] & & Y^2 = \bot \\
X^3 = [1, 5] & & Y^3 = \bot \\
\ldots = \ldots & & \ldots = \ldots \\
X^{50} = [1, 99] & & Y^{50} = \bot \\
X^{51} = [1, 101] & & Y^{51} = [101, 101] \\
X^{52} = [1, 101] & & Y^{52} = [101, 101]
\end{array}
$$

Convergence could have been very slow (even impossible without the test `I <= 100` when using bignums)!

– Upwards iteration from the infimum with widening

$$
\begin{array}{l}
\hat{X}^0 = \bot \\
\hat{X}^1 = \hat{X}^0 \nabla [1, 1] = [1, 1] \\
\hat{X}^2 = \hat{X}^1 \nabla [1, 3] = [1, +\infty] \\
\hat{X}^3 = \hat{X}^2 \nabla [1, 102] = [1, +\infty] \\
\hat{Y}^3 = [101, +\infty]
\end{array}
$$

Convergence is accelerated hence the loss of precision!

– Downward iteration with narrowing

$$
\begin{array}{ll}
\check{X}^0 = [1, +\infty] & \check{Y}^0 = [101, +\infty] \\
\check{X}^1 = \check{X}^0 \,\Delta\, [1, 102] = [1, 102] & \\
\check{X}^2 = \check{X}^1 \,\Delta\, [1, 102] = [1, 102] & \\
 & \check{Y}^3 = [101, 102]
\end{array}
$$

– The narrowing is not always able to recapture the information lost by the widening

– It's therefore better not to loose too much information by widening in the first upward iteration

A parameterized meta-example of interval invariant by fixpoint approximation with convergence acceleration by widening/narrowing

– The analyzer will behave in exactly the same way for all programs of the form

```
program P;
    var I : integer;
begin
{1:}
    I := 1;
{2:}
    while { I ∈ X } I <= n do begin
{3:}
        I := I + 2;
{4:}
    end;
{5:} { I ∈ Y }
end.
```


where n is a mathematical variable denoting any program constant, $n \geq 1$. By instantiating n to all possible naturals ≥ 1, one gets an infinite family of programs, which are similar up to n and have therefore similar analyzes.

– The fixpoint interval equations are all of the same form:
  - $X = [1,\ 1] \cup ((X \cap [-\infty,\ n]) \mp [2,\ 2]) \quad X \in [1,\ n+1]$
  - $Y = X \cap [n+1,\ +\infty] \qquad Y \in [n+1,\ +\infty]$

– The upward iteration with widening is now (in parametric form):

$$
\begin{aligned}
\hat{X}^0 &= \bot\\
\hat{X}^1 &= \hat{X}^0 \,\nabla\, ([1,\ 1] \cup ((\hat{X}^0 \cap [-\infty,\ n]) \mp [2,\ 2]))\\
&= \bot \,\nabla\, [1,\ 1] \cup \bot\\
&= [1,\ 1]\\
\hat{X}^2 &= \hat{X}^1 \,\nabla\, ([1,\ 1] \cup ((\hat{X}^1 \cap [-\infty,\ n]) \mp [2,\ 2]))\\
&= [1,\ 1] \,\nabla\, ([1,\ 1] \cup (([1,\ 1] \cap [-\infty,\ n]) \mp [2,\ 2]))\\
&= [1,\ 1] \,\nabla\, [1,\ 3]\\
&= [1,\ +\infty]\\
\hat{X}^3 &= \hat{X}^2 \,\nabla\, ([1,\ 1] \cup ((\hat{X}^2 \cap [-\infty,\ n]) \mp [2,\ 2]))\\
&= [1,\ +\infty] \,\nabla\, ([1,\ 1] \cup (([1,\ +\infty] \cap [-\infty,\ n]) \mp [2,\ 2]))\\
&= [1,\ +\infty] \,\nabla\, ([1,\ 1] \cup ([3,\ n+2])\\
&= [1,\ +\infty] \,\nabla\, ([1, n+2])\\
&= [1,\ +\infty]
\end{aligned}
$$

– The downward iteration sequence from $[1,\ +\infty]$ with narrowing will now be as follows (always in parameterized form, to be instantiated for any particular value of n):

$$
\begin{aligned}
\check{X}^0 &= [1,\ +\infty]\\
\check{X}^1 &= \check{X}^0 \,\Delta\, ([1,\ 1] \cup ((\check{X}^0 \cap [-\infty,\ n]) \mp [2,\ 2]))\\
&= [1,\ +\infty] \,\Delta\, ([1,\ 1] \cup (([1,\ +\infty] \cap [-\infty,\ n]) \mp [2,\ 2]))\\
&= [1,\ +\infty] \,\Delta\, ([1,\ 1] \cup (([1,\ n]) \mp [2,\ 2]))\\
&= [1,\ +\infty] \,\Delta\, ([1,\ 1] \cup [2,\ n+2])\\
&= [1,\ +\infty] \,\Delta\, ([1, n+2])\\
&= [1, n+2])\\
\check{X}^2 &= \check{X}^1 \,\Delta\, [1,\ n+2]\\
&= [1,\ n+2]
\end{aligned}
$$

$$
\begin{aligned}
\check{Y}^2 &= \check{X}^2 \cap [n+1,\ +\infty] \\
&= [1,\ n+2] \cap [n+1,\ +\infty] \\
&= [n+1,\ n+2]
\end{aligned}
$$

– This proves that for all programs in the family (parameterized by n), the analysis with widening/narrowing will always discover the interval invariant

$$
\begin{cases} X = [1,\ n+2] \\ Y = [n+1,\ n+2] \end{cases}
$$

for the given n corresponding to each particular program in the family.

Finitary nature of static analysis with widening/narrowing

THEOREM. Given any specific program, and given specific infinite abstract domain together with a specific widening, it is possible to find a finite lattice and a Galois connection which will produce exactly the same analysis results for that given program. ∎

PROOF. – Assume that we are given a program P and that the problem is to overapproximate $\mathsf{lfp}_{\bot}^{\sqsubseteq} F$ where F is a concrete monotonic transformer $F \in L \overset{m}{\longmapsto} L$ on the cpo $\langle L,\ \sqsubseteq,\ \bot,\ \top,\ \sqcup\rangle$. We assume L to contain a supremum $\top$ [9]

[9] to be able to express "I don't know" in the concrete.

– The analyzer makes use of an abstract domain $\langle \overline{L},\ \overline{\sqsubseteq}\rangle$ such that $\langle L,\ \sqsubseteq\rangle \underset{\alpha}{\overset{\gamma}{\leftarrows\!\!\twoheadrightarrow}} \langle \overline{L},\ \overline{\sqsubseteq}\rangle$, a monotonic abstract transformer $\overline{F}\ \dot{\overline{\sqsupseteq}}\ \alpha \circ F \circ \gamma$ and a widening $\overline{\nabla}$.

– Because α is surjective [10], $\langle \overline{L},\ \overline{\sqsubseteq},\ \overline{\bot},\ \overline{\top},\ \overline{\sqcup}\rangle$ is indeed a cpo with supremum $\overline{\top} = \alpha(\top)$.

– The analysis computes iterates $y^0 = \alpha(\bot),\ \ldots,\ y^{n+1} = y^n\ \overline{\nabla}\ \overline{F}(y^n), \ldots, y^\ell$ where the limit y^ℓ is a postfixpoint $\overline{F}(y^\ell)\ \overline{\sqsubseteq}\ y^\ell$

– Let us define the abstract domain $\overline{\overline{L}} = \{y^0, \ldots, y^n, \ldots, y^\ell, \overline{\top}\}$ with ordering $\overline{\overline{\sqsubseteq}}$ which is $\overline{\sqsubseteq}$ restricted to $\overline{\overline{L}}$.

– Because the iterates are a finite increasing chain and $\overline{\top}\}$ is the supremum, $\langle \overline{\overline{L}},\ \overline{\overline{\sqsubseteq}}\rangle$ is a finite chain whence a complete lattice.

– Let us define the abstraction

$$
\begin{aligned}
\overline{\overline{\alpha}} &\in L \mapsto \overline{\overline{L}} \\
\overline{\overline{\alpha}}(x) &\stackrel{\text{def}}{=} \overline{\overline{\bigsqcap}}\{y \in \overline{\overline{L}} \mid \alpha(x)\ \overline{\sqsubseteq}\ y\}\ ^{11}
\end{aligned}
$$

[10] Otherwise we choose $\overline{L} = \alpha(L)$.

[11] So that $\alpha(x) = \overline{\top}$ if x is not comparable to any of the iterates y^i, $i = 0, \ldots, \ell$.

and the concretization

$$
\begin{aligned}
\overline{\overline{\alpha}} &\in \overline{\overline{L}} \mapsto L \\
\overline{\overline{\gamma}} &\stackrel{\text{def}}{=} \gamma
\end{aligned}
$$

– We have a Galois connection $\langle L,\ \sqsubseteq\rangle \underset{\overline{\overline{\alpha}}}{\overset{\overline{\overline{\gamma}}}{\leftarrows\!\!\twoheadrightarrow}} \langle \overline{\overline{L}},\ \overline{\overline{\sqsubseteq}}\rangle$.

PROOF.

$$
\begin{aligned}
& \overline{\overline{\alpha}}(x)\ \overline{\overline{\sqsubseteq}}\ y \\
\Longleftrightarrow\ & \overline{\overline{\alpha}}(x)\ \overline{\sqsubseteq}\ y && \text{(def. } \overline{\overline{\sqsubseteq}}\text{)} \\
\Longleftrightarrow\ & \overline{\overline{\bigsqcap}}\{y \in \overline{\overline{L}} \mid \alpha(x)\ \overline{\sqsubseteq}\ y\}\ \overline{\sqsubseteq}\ y && \text{(def. } \overline{\overline{\alpha}}\text{)} \\
\Longrightarrow\ & && \text{(We have } \alpha(x)\ \overline{\sqsubseteq}\ \overline{\top} \in \overline{\overline{L}} \text{ so, since } \overline{\overline{L}} \text{is a finite strictly decreasing} \\
& && \text{chain, there is a smallest } y^n \in \overline{\overline{L}} : \alpha(x)\ \overline{\sqsubseteq}\ y^n\text{, whence } \alpha(x)\ \overline{\sqsubseteq}\ y' \\
& && \text{implies } y^n\ \overline{\sqsubseteq}\ y' \text{ so } y^n = \overline{\overline{\bigsqcap}}\{y \in \overline{\overline{L}} \mid \alpha(x)\ \overline{\sqsubseteq}\ y\}\ \overline{\sqsubseteq}\ y\text{. It follows that:)} \\
& \alpha(x)\ \overline{\sqsubseteq}\ y^n\ \overline{\overline{\sqsubseteq}}\ y \\
\Longrightarrow\ & x \sqsubseteq \gamma(y)
\end{aligned}
$$

$$\Longrightarrow x \sqsubseteq \overline{\overline{\gamma}}(y) \qquad \wr\text{since } \gamma = \overline{\overline{\gamma}}\wr$$

Conversely

$$x \sqsubseteq \overline{\overline{\gamma}}(y)$$
$$\Longrightarrow x \sqsubseteq \gamma(y) \qquad \wr\text{since } \gamma = \overline{\overline{\gamma}}\wr$$
$$\Longrightarrow \alpha(x) \overline{\sqsubseteq} y$$
$$\Longrightarrow \overline{\overline{\bigsqcap}}\{y' \in \overline{\overline{L}} \mid \alpha(x) \overline{\sqsubseteq} y'\} \overline{\sqsubseteq} y$$
$$\Longrightarrow \overline{\overline{\alpha}}(x) \overline{\sqsubseteq} y$$
$$\Longrightarrow \overline{\overline{\alpha}}(x) \overline{\overline{\sqsubseteq}} y$$

□

– Let us define:

$$\overline{\overline{F}} \in \overline{\overline{L}} \mapsto \overline{\overline{L}}$$
$$\overline{\overline{F}} \stackrel{\text{def}}{=} \lambda y \cdot (\!| \, y = \top \; ? \; \top \; \| \; y = y^{\ell} \; ? \; \overline{F}(y^{\ell}) \; ¿ \; y \, \nabla \, \overline{F}(y) \, |\!)$$

– We have $\overline{\overline{F}} \mathrel{\dot{\sqsupseteq}} \overline{F} \mathrel{\dot{\sqsupseteq}} \overline{\overline{\alpha}} \circ F \circ \overline{\overline{\gamma}}$

PROOF. We proceed pointwise on $\overline{\overline{L}} \subseteq \overline{L}$. We already know that $\overline{F} \mathrel{\dot{\sqsupseteq}} \overline{\overline{\alpha}} \circ F \circ \overline{\overline{\gamma}}$.

- This is obvious for $\top$ since $\overline{\overline{F}}(\top) = \top \sqsupseteq \overline{F}(\top)$ since $\top$ is the supremum of $\overline{L}$
- This holds for y^{ℓ} since $\overline{\overline{F}}(y^{\ell}) = \overline{F}(y^{\ell}) \sqsupseteq \overline{F}(y^{\ell})$ by reflexivity.
- For the other elements $y \in \overline{\overline{L}} \setminus \{\top, y^{\ell}\}$, we have $\overline{\overline{F}}(y) = y \, \nabla \, \overline{F}(y) \sqsupseteq \overline{F}(y)$ by def. ∇ which is an upper bound.
- Observe that the iterates of $\overline{\overline{F}}$ from $\alpha(\bot)$ are exactly $y^0 = \alpha(\bot)$, ..., $y^n = \overline{\overline{F}}(y^{n-1}) = y^{n-1} \, \nabla \, \overline{F}(y^{n-1})$, ..., y^{ℓ} since $\overline{\overline{F}}(y^{\ell}) = \overline{F}(y^{\ell}) \sqsubseteq y^{\ell}$, which is the convergence condition.
- These iterates are therefore convergent (despite the fact that the widening ∇ hence $\overline{\overline{F}}$ is not monotone since $\overline{\overline{F}}$ is extensive, but for y^{ℓ}, which is the convergence point).

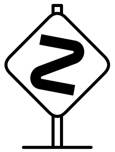
- In conclusion, the analysis of the given program can be done in the finite (complete) lattice $\langle \overline{\overline{L}}, \overline{\overline{\sqsubseteq}}, \overline{\overline{\bot}}, \overline{\overline{\sqcup}} \rangle$ by computing the limit (y^{ℓ}) of the finitely convergent iterates of $\overline{\overline{F}} \mathrel{\dot{\sqsupseteq}} \overline{\overline{\alpha}} \circ F \circ \overline{\overline{\gamma}}$.

□

□

An incorrect common believe about the uselessness of widenings

Because of this theorem, some (a.o. [1]) conclude that:



> The widening approach to program static analysis is useless since it is always possible to perform an iterative static analysis using a finite abstract domain.

This is ERRONEOUS [2]

Reference

[1] R.B. Kieburtz and M. Napierala. Abstract semantics. In S. Abramsky and C. Hankin, eds., Abstract Interpretation of Declarative Languages, chapter 7, pp. 143–180. Ellis Horwood, Chichester, U.K., 1987.

[2] P. Cousot and R. Cousot. Comparing the Galois Connection and Widening/Narrowing Approaches to Abstract Interpretation, invited paper. In M. Bruynooghe and M. Wirsing, eds., *Programming Language Implementation and Logic Programming, Proc. 4th In. Symp., PLILP'92*, Leuven, Belgium, 13–17 Aug. 1992, LNCS 631, p. 269–295. Springer-Verlag, 1992.

Proof that the common believe about the uselessness of widenings is erroneous

– This is due to the confusion between analyzing a given program, as opposed to any program in a given programming language

– We exhibit a counter-example, using interval analysis, showing that infinitely many $[0, n], n \geq 0$ are needed

– For any given $n = 0, 1, 2, \ldots$, we have seen that the interval analysis with widening will produce the following analysis (given as comments between $\{\ldots\}$):


```
program P;
    var I : integer;
begin
{1:}
      I := 1;
{2:}
      while { I ∈ X } I <= 100 do begin
{3:}
         I := I + 2;
{4:}
      end;
{5:} { I ∈ Y }
end.
```

– So when considering all programs $P(n)$, for all $n \geq 0$, we have to have all necessary inductive invariants $[0, n + 1]$, $n \geq 0$ (otherwise the analysis can only be less precise if this invariant is not expressible)

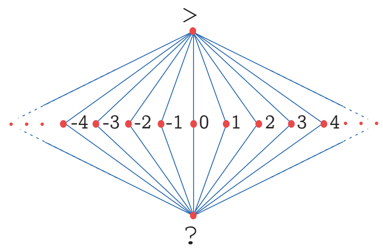
– So the abstract domain with which the static analysis can produce this result for all $P(n)$, $n \geq 0$ must contain an infinite strictly increasing chain $[0, 1] \sqsubseteq [0, 2] \sqsubseteq \ldots \sqsubseteq [0, n] \sqsubseteq \ldots$

– Analyzing iteratively a program like `while true do I := I + 2; end` will definitely require a widening to converge

– Another hope would be to guess the constants n, $\ldots$ by a simple syntactic inspection of the program text (by "simple" we exclude a static analysis with widening and similar sophisticated analyzes!)

– However practice show that this is extremely difficult

– A first example is Kildall's constant propagation using the lattice:



for which an equally precise analysis has to guess all necessary constants, including those not appearing explicitly in the program text

– A second example, using interval analysis

```
program Variant_of_function_91_of_McCarthy;
   var X, Y : integer;
   function F(X : integer) : integer;
   begin
      if X > 100 then
         F := X - 10      { F ∈ [91, maxint - 10] }
      else
         F := F(F(F(F(X + 33))));     { F ∈ [91, 93] }
      { F ∈ [91, maxint - 10] }
   end;
begin
   readln(X);   Y := F(X);
   { Y ∈ [91, maxint - 10] }
end.
```

shows that the intermediate intervals for the recursive calls cannot be easily guessed on a syntactic basic.

A correct statement about the usefulness of widenings

The power of the widening/narrowing approach to static program analysis by abstract interpretation is more precisely stated as follows:

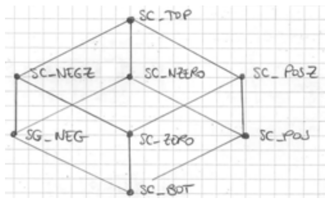
1. For each program there exists a finite lattice which can be used for this program to obtain results equivalent to those obtained using widening/narrowing operators;

2. No such a finite lattice (more precisely, satisfying the ascending chain condition) will do for all programs;
3. For all programs, infinitely many abstract values are necessary;
4. For a particular program it is not possible to infer the set of needed abstract values by a simple inspection of the text of the program.

Another incorrect common believe about the precision of widenings

– It can be thought that an analysis using a more precise abstraction (with widening on an infinite abstract domain not satisfying the ACC) is always more precise than an analysis using a less precise abstraction (e.g. in a finite abstract domain)

– Here is a *counter-example*, using a sign analysis:



– An example of sign analysis is the following:

```
0: { x: SC_TOP}
  x := 1;
2: { x: SC_POS}
  while (x <> 0) do
    4: {x : SC_POS}
      if (0 < x) then
        5: {X: SC_POS}
          x := 0
        6: { x: SC_ZERO}
      else
        7: SC_BOT
          skip
        8: SC_BOT
      fi;
    10: { x: SC_ZERO}
  od
11: { x: SC_ZERO}
```


– The interval analysis with abstract domain

$$\overline{L} \stackrel{\text{def}}{=} \{[\ell,\ u] \mid \ell \in \mathbb{Z} \cup \{-\infty\} \wedge u \in \mathbb{Z} \cup \{+\infty\} \ \wedge\ \ell \leq u\} \cup \{\bot\}$$

and widening extrapolating unstable bounds to infinity:

$$\bot \,\nabla\, X = X$$
$$X \,\nabla\, \bot = X$$
$$[\ell_0,\ u_0] \,\nabla\, [\ell_1,\ u_1] = [(\ell_1 < \ell_0 \ ?\ -\infty : \ell_0),\ (u_1 > u_0 \ ?\ +\infty \ell u_0)]$$

is less precise!

– Program analysis:

```
0: { x:[-∞,+∞]
  x := 1;
2: { x:[-∞, 1]
  while (x <> 0) do
    4: {x:[-∞,1]}
      if (0 < x) then
        5: { x:[1,1] }
          x := 0
        6: { x:[0,0] }
      else
        7:{x:[-∞,0]}
          skip
        8:{x:[-∞,0]}
      fi;
    10: { x:[-∞,0]}
  od
11: { x:[0,0]}
```

```
0 : {x : [-∞,+∞]}
2 : {x : [1,1]}
4 : {x : [1,1]}
5 : {x : [1,1]}
6 : {x : [0,0]}
10 : {x : [0,1]}
2 : {x : [-∞,1]}
4 : {x : [-∞,1]}
5 : {x : [1,1]}
6 : {x : [0,0]}
7 : {x : [-∞,0]}
8 : {x : [-∞,0]}
10 : {x : [-∞,0]}
2 : {x : [-∞,1] ∇ [-∞,1]}
  = {x : [-∞,1]}
stable
```


– The widening is at the origin of the loss of precision:

	previous iterate X	$F(X) = X - 1$	next iterate
signs	SC_POS	SC_POSZ	SC_POSZ
intervals	$[1, +\infty]$	$[0, +\infty]$	$[-\infty, +\infty]$

The 0 threshold is missed by the widening but catched by the sign analysis.

– The interval analysis with improved widening is as precise or more precise than the sign analysis:

```
0: { x:_O_ }
  x := 1;
2: { x:[0,1] }
  while (x <> 0) do
    4: {                  x:[1,1] }        ←    x = 1
      if (0 < x) then
        5: { x:[1,1] }
          x := 0
        6: { x:[0,0] }
      else {(x <= 0)}
        7: _|_
          skip
        8: _|_
      fi;
    10: { x:[0,0] }
  od {(x = 0)}
11: { x:[0,0] }
```


A correct statement about the relative precision of widenings

Theorem. Assume that $\langle L, \sqsubseteq\rangle$, $\langle \overline{L}, \overline{\sqsubseteq}\rangle$ and $\langle \overline{\overline{L}}, \overline{\overline{\sqsubseteq}}\rangle$ are posets such that

(a) $\overline{F} \in \overline{L} \mapsto \overline{L}$, $\overline{\overline{F}} \in \overline{\overline{L}} \mapsto \overline{\overline{L}}$

(b) $\overline{\gamma} \in \overline{L} \mapsto L$, $\overline{\overline{\gamma}} \in \overline{\overline{L}} \mapsto L$

(c) $\overline{a} \in \overline{L}$, $\overline{\overline{a}} \in \overline{\overline{L}}$ with $\overline{\gamma}(\overline{a}) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{a}})$

(d) $\overline{\nabla} \in \overline{L} \times \overline{L} \mapsto \overline{L}$, $\overline{\overline{\nabla}} \in \overline{\overline{L}} \times \overline{\overline{L}} \mapsto \overline{\overline{L}}$ satisfy

(d.1) $[\overline{\gamma}(X) \sqsubseteq \overline{\overline{\gamma}}(X') \wedge \overline{\gamma}(Y) \sqsubseteq \overline{\overline{\gamma}}(Y')] \Longrightarrow [\overline{\gamma}(X \mathbin{\overline{\nabla}} Y) \sqsubseteq \overline{\overline{\gamma}}(X' \mathbin{\overline{\overline{\nabla}}} Y')]$

(d.2) $[Y \mathrel{\overline{\sqsubseteq}} X \Longrightarrow X \mathbin{\overline{\nabla}} Y = X]$, $[Y \mathrel{\overline{\overline{\sqsubseteq}}} X \Longrightarrow X \mathbin{\overline{\overline{\nabla}}} Y = X]$

(e) $[\overline{\gamma}(X) \sqsubseteq \overline{\overline{\gamma}}(X')] \Longrightarrow [\overline{\gamma} \circ \overline{F}(X) \sqsubseteq \overline{\overline{\gamma}} \circ \overline{\overline{F}}(X')]$

then the limit $\overline{X}^{\ell}$ of the iterates of $\overline{F}$ from $\overline{a}$ with widening $\overline{\nabla}$ is more precise than the limit $\overline{\overline{X}}^{\ell}$ of the iterates of $\overline{\overline{F}}$ from $\overline{\overline{a}}$ with widening $\overline{\overline{\nabla}}$:

$$\overline{\gamma}(\overline{X}^{\ell}) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^{\ell})$$

■

Proof. We let $\langle \overline{X}^k, k \geq 0\rangle$ be defined as follows:

$$\begin{cases} \overline{X}^0 = \overline{a} & \\ \overline{X}^{n+1} = \overline{X}^n & \text{if } \overline{F}(\overline{X}^n) \mathrel{\overline{\sqsubseteq}} \overline{X}^n \\ \overline{X}^{n+1} = \overline{X}^n \mathbin{\overline{\nabla}} \overline{F}(\overline{X}^n) & \text{otherwise} \end{cases}$$

and similarly $\langle \overline{\overline{X}}^k, k \geq 0\rangle$ is defined as follows:

$$\begin{cases} \overline{\overline{X}}^0 = \overline{\overline{a}} & \\ \overline{\overline{X}}^{n+1} = \overline{\overline{X}}^n & \text{if } \overline{\overline{F}}(\overline{\overline{X}}^n) \mathrel{\overline{\overline{\sqsubseteq}}} \overline{\overline{X}}^n \\ \overline{\overline{X}}^{n+1} = \overline{\overline{X}}^n \mathbin{\overline{\overline{\nabla}}} \overline{\overline{F}}(\overline{\overline{X}}^n) & \text{otherwise} \end{cases}$$

– We have $\overline{\gamma}(\overline{X}^0) = \overline{\gamma}(\overline{a}) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{a}}) = \overline{\overline{\gamma}}(\overline{\overline{X}}^0)$ by (c).

– Assume by induction hypothesis that $\overline{\gamma}(\overline{X}^n) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^n)$

  - If both iterates have converged then $\overline{\gamma}(\overline{X}^{n+1}) = \overline{\gamma}(\overline{X}^n) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^n) = \overline{\overline{\gamma}}(\overline{\overline{X}}^{n+1})$

  - If $\langle \overline{X}^k, k \geq 0\rangle$ has converged at rank n, but not $\langle \overline{\overline{X}}^k, k \geq 0\rangle$. We have

$\overline{\gamma}(\overline{X}^n) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^n)$ $\wr$ind. hyp.$\wr$

$\Longrightarrow \overline{\gamma}(\overline{X}^n) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^n \mathbin{\overline{\overline{\nabla}}} \overline{\overline{F}}(\overline{\overline{X}}^n))$ $\wr$by (d.1)$\wr$

$\Longrightarrow \overline{\gamma}(\overline{X}^{n+1}) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^{n+1})$ $\wr$by def. iterates$\wr$

- If $\langle \overline{\overline{X}}^k,\ k \geq 0\rangle$ has converged at rank $\overline{\ell} \leq n$ but not $\langle \overline{X}^k,\ k \geq 0\rangle$, we have

$\overline{\gamma}(\overline{X}^n) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^n)$ $\wr$ind. hyp.$\wr$

$\Longrightarrow \overline{\gamma}(\overline{F}(\overline{X}^n)) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{F}}(\overline{\overline{X}}^n))$ $\wr$by (e)$\wr$

$\Longrightarrow \overline{\gamma}(\overline{X}^n \mathbin{\overline{\nabla}} \overline{F}(\overline{X}^n)) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^n \mathbin{\overline{\overline{\nabla}}} \overline{\overline{F}}(\overline{\overline{X}}^n))$ $\wr$by (d.1)$\wr$

$\Longrightarrow \overline{\gamma}(\overline{X}^n) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^n \mathbin{\overline{\overline{\nabla}}} \overline{\overline{F}}(\overline{\overline{X}}^n))$ $\wr$by (d.2) since $\langle \overline{\overline{X}}^k,\ k \geq 0\rangle$ has converged at rank $\overline{\ell} \leq n$ and so $\overline{X}^{\overline{\ell}} = \ldots = \overline{X}^n$ and $\overline{F}(\overline{X}^n) \mathrel{\overline{\sqsubseteq}} \overline{X}^n$$\wr$

$\Longrightarrow \overline{\gamma}(\overline{X}^{n+1}) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^{n+1})$ $\wr$by def. iterates$\wr$

- If $\langle \overline{X}^k,\ k \geq 0\rangle$ has converged at rank $\overline{\ell} \leq n$ but not $\langle \overline{\overline{X}}^k,\ k \geq 0\rangle$, we have

$\overline{\gamma}(\overline{X}^n) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^n)$ $\wr$ind. hyp.$\wr$

$\Longrightarrow \overline{\gamma}(\overline{F}(\overline{X}^n)) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{F}}(\overline{\overline{X}}^n))$ $\wr$by (e)$\wr$

$\Longrightarrow \overline{\gamma}(\overline{X}^n \mathbin{\overline{\nabla}} \overline{F}(\overline{X}^n)) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^n \mathbin{\overline{\overline{\nabla}}} \overline{\overline{F}}(\overline{\overline{X}}^n))$ $\wr$by (d.1)$\wr$

$\Longrightarrow$ $\wr$by (d.2) since $\overline{\overline{X}}^{\overline{\overline{\ell}}} = \ldots = \overline{\overline{X}}^n$ with $\overline{\overline{F}}(\overline{\overline{X}}^n) \mathrel{\overline{\overline{\sqsubseteq}}} \overline{\overline{X}}^n$$\wr$
$\overline{\gamma}(\overline{X}^n \mathbin{\overline{\nabla}} \overline{F}(\overline{X}^n)) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^n)$

$\Longrightarrow \overline{\gamma}(\overline{X}^{n+1}) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^{n+1})$ $\wr$def. iterates$\wr$

- If none of the $\langle \overline{X}^k,\ k \geq 0\rangle$ and $\langle \overline{\overline{X}}^k,\ k \geq 0\rangle$ have converged at rank n, then:

$\overline{\gamma}(\overline{X}^n) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^n)$ $\wr$ind. hyp.$\wr$

$\Longrightarrow \overline{\gamma}(\overline{F}(\overline{X}^n)) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{F}}(\overline{\overline{X}}^n))$ $\wr$by (e)$\wr$

$\Longrightarrow \overline{\gamma}(\overline{X}^n \mathbin{\overline{\nabla}} \overline{F}(\overline{X}^n)) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^n \mathbin{\overline{\overline{\nabla}}} \overline{\overline{F}}(\overline{\overline{X}}^n))$ $\wr$by (e)$\wr$

$\Longrightarrow \overline{\gamma}(\overline{X}^{n+1}) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^{n+1})$ $\wr$def. iterates$\wr$

– By recurrence, $\forall n \in \mathbb{N} : \overline{\gamma}(\overline{X}^n) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^n)$

– If $\langle \overline{X}^k,\ k \geq 0\rangle$ converges at rank $\overline{\ell}$ and and similarly $\langle \overline{\overline{X}}^k,\ k \geq 0\rangle$ converges at rank $\overline{\overline{\ell}}$, we have $\overline{\gamma}(\overline{X}^{\overline{\ell}} = \overline{\gamma}(\overline{X}^{\max(\overline{\ell},\overline{\overline{\ell}})}) \sqsubseteq \overline{\overline{\gamma}}(\overline{\overline{X}}^{\max(\overline{\ell},\overline{\overline{\ell}})}) = \overline{\overline{\gamma}}(\overline{\overline{X}}^{\overline{\overline{\ell}}})$.

$\square$

Note: If one iterate has no widening, we can just replace it by the join since F and $\lambda X \cdot X \sqcup F(X)$ have identical iterates when starting from a prefixpoint and F is monotone or F is extensive.

Weakening the hypotheses on widenings (expression of the upper bound overapproximation in term of concretization, no need for lub overapproximation)

– We have shown that for a monotonic $\overline{F}$ on a cpo $\langle \overline{L},\ \overline{\sqsubseteq},\ \overline{\bot},\ \overline{\sqcup}\rangle$, **lfp** $\overline{F}$ is overapproximated by the limit of an upper iteration of $\overline{F}$ from $\overline{\bot}$ with widening $\overline{\nabla}$

– With this point of view, the correctness conditions for the widening are expressed in the abstract

$$X \mathrel{\overline{\sqsubseteq}} X \mathbin{\overline{\nabla}} Y$$
$$Y \mathrel{\overline{\sqsubseteq}} X \mathbin{\overline{\nabla}} Y$$

– In practice, we only want to compare iterations of $F \in L \stackrel{m}{\longmapsto} L$ on the cpo $\langle L, \sqsubseteq, \bot, \sqcup\rangle$ with the iterates of $\overline{F} \in \overline{L} \mapsto \overline{L}$ with widening $\overline{\nabla}$.
– Then the above overapproximation hypotheses can be replaced with

$$\gamma(X) \sqsubseteq \gamma(X \mathbin{\overline{\nabla}} Y)$$
$$\gamma(Y) \sqsubseteq \gamma(X \mathbin{\overline{\nabla}} Y)$$

(together with $F \circ \gamma \mathrel{\dot{\sqsubseteq}} \gamma \circ \overline{F}$)

– These hypotheses may be useful when the widening is used both to
- overapproximate non-existent lubs
- accelerate the convergence of the iterates
– The widening $\overline{\nabla}$ is used to generate induction hypotheses which are checked by the convergence condition $\overline{F}(X) \mathrel{\overline{\sqsubseteq}} X$ so no condition on $\overline{\nabla}$ relative to soundness is indeed needed!

Revisiting the soundness of increasing iterations with widening (enforcing convergence without (concrete) lub overapproximation)

THEOREM. [12] Let $F \in L \stackrel{m}{\longmapsto} L$ be a monotone operator on the cpo $\langle L, \sqsubseteq, \sqcap\rangle$. Assume $\bot \in L$ satisfies $\bot \sqsubseteq F(\bot)$. Let $\langle \overline{L}, \overline{\sqsubseteq}\rangle$ be a poset and $\overline{F} \in \overline{L} \mapsto \overline{L}$ such that $F \circ \gamma \mathrel{\dot{\sqsubseteq}} \gamma \circ \overline{F}$ where $\gamma \in \overline{L} \stackrel{m}{\longmapsto} L$. Assume that the widening $\overline{\nabla} \in \overline{L} \times \overline{L} \mapsto \overline{L}$ satisfies:

– $\forall x, y \in \overline{L} : \gamma(y) \sqsubseteq \gamma(x \mathbin{\overline{\nabla}} y)$;
– $\forall x, y \in \overline{L} : \gamma(x) \sqsubseteq \gamma(x \mathbin{\overline{\nabla}} y)$ or $\overline{F}$ is extensive, i.e.: $\forall x, y \in \overline{L} : x \mathrel{\overline{\sqsubseteq}} \overline{F}(x)$.

Assume that the widening iteration sequence for $\overline{F}$ from $\overline{\bot}$ (satisfying $\bot \sqsubseteq \gamma(\overline{\bot})$) is $\langle X^n, n \in \mathbb{N}\rangle$, which is defined as follows:

– $\overline{X}^0 = \overline{\bot}$ (a)
– $\overline{X}^{n+1} = \overline{X}^n$ if $\overline{F}(\overline{X}^n) \mathrel{\overline{\sqsubseteq}} \overline{X}^n$ (b)
– $\overline{X}^{n+1} = \overline{X}^n \mathbin{\overline{\nabla}} \overline{F}(\overline{X}^n)$ otherwise (c)

is ultimately stationary at rank $\overline{\ell} \in \mathbb{N}$. Then $\gamma(\overline{F}(\overline{X}^{\ell})) \sqsubseteq \gamma(\overline{X}^{\ell})$ and $\mathsf{lfp}_{\bot}^{\sqsubseteq} F \sqsubseteq \gamma(\overline{X}^{\overline{\ell}})$. ■

PROOF. First observe that $\langle \gamma(\overline{X}^n), n \in \mathbb{N}\rangle$ is an increasing chain since for $\overline{X}^n$ either (b) holds in which case this is trivial by reflexivity since $\gamma(\overline{X}^n) = \gamma(\overline{X}^{n+1})$ or (c) holds, in which case either $\gamma(\overline{X}^n) \sqsubseteq \gamma(\overline{X}^n \mathbin{\overline{\nabla}} \overline{F}(\overline{X}^n)) = \gamma(\overline{X}^{n+1})$ or $\overline{X}^n \mathrel{\overline{\sqsubseteq}} \overline{F}(\overline{X}^n)$ by extensivity and so by monotony $\gamma(\overline{X}^n) \sqsubseteq \gamma(\overline{F}(\overline{X}^n)) \sqsubseteq \gamma(\overline{X}^n \mathbin{\overline{\nabla}} \overline{F}(\overline{X}^n)) = \gamma(\overline{X}^{n+1})$.

[12] Observe that the absence of lub existence hypotheses in $\overline{L}$, that $\overline{F}$ is not assumed to be monotone or extensive and that the widening is only assumed to ensure convergence not to overapproximate lubs.

$\overline{X}^{\overline{\ell}}$ exists by the convergence enforcement hypothesis on the widening. Moreover $\overline{\ell} \geq 1$ since at least one iteration is necessary to check for stability. In case $\overline{X}^{\overline{\ell}}$ satisfies (b), we have

$$\overline{F}(\overline{X}^{\overline{\ell}}) \sqsubseteq \overline{X}^{\overline{\ell}}$$
$$\Longrightarrow \gamma \circ \overline{F}(\overline{X}^{\overline{\ell}}) \sqsubseteq \gamma(\overline{X}^{\overline{\ell}}) \quad \wr \gamma \text{ monotone} \wr$$
$$\Longrightarrow F \circ \gamma(\overline{X}^{\overline{\ell}}) \sqsubseteq \gamma(\overline{X}^{\overline{\ell}}) \quad \wr \text{since } F \circ \gamma \,\dot{\sqsubseteq}\, \gamma \circ \overline{F} \wr$$
$$\Longrightarrow \mathsf{lfp}^{\sqsubseteq}_{\perp} F \sqsubseteq \gamma(\overline{X}^{\overline{\ell}})$$

by transfinite induction on the iterates of F from $\perp$ as follows:

- $X^0 = \perp \sqsubseteq \gamma(\overline{\perp}) = \overline{X}^0 \sqsubseteq \gamma(\overline{X}^{\overline{\ell}})$ by hypothesis and $\langle \gamma(\overline{X}^n),\ n \in \mathbb{N} \rangle$ is an increasing chain
- If $X^\delta \sqsubseteq \gamma(\overline{X}^{\overline{\ell}})$ by induction hypothesis then by monotony of F, we have $X^{\delta+1} = F(X^\delta) \sqsubseteq F \circ \gamma(\overline{X}^{\overline{\ell}}) \sqsubseteq \gamma \circ \overline{F}(\overline{X}^{\overline{\ell}}) \sqsubseteq \gamma(\overline{X}^{\overline{\ell}})$ by the convergence condition $\overline{F}(\overline{X}^n) \,\overline{\sqsubseteq}\, \overline{X}^n$ and γ monotone.
- If λ is a limit ordinal and $X^\delta \sqsubseteq \gamma(\overline{X}^{\overline{\ell}})$ for all $\delta < \lambda$ by induction hypothesis, then $X^\lambda = \bigsqcup_{\delta<\lambda} X^\delta \sqsubseteq \gamma(\overline{X}^{\overline{\ell}})$ by def. of lubs which exist for chains in cpos.
- There exists ϵ such that $\mathsf{lfp}^{\sqsubseteq}_{\perp} F = X^\epsilon \sqsubseteq \gamma(\overline{X}^{\overline{\ell}})$.

Otherwise $\overline{X}^{\overline{\ell}}$ satisfies (c) and we have

$$\overline{X}^{\ell} = \overline{X}^{\ell} \,\overline{\nabla}\, \overline{F}(\overline{X}^{\ell}) \quad \wr \text{by (c) since } \overline{X}^{\overline{\ell+1}} = \overline{X}^{\overline{\ell}} \wr$$
$$\Longrightarrow \gamma(\overline{F}(\overline{X}^{\overline{\ell}})) \sqsubseteq \gamma(\overline{X}^{\ell}) \quad \wr \text{since } \forall x, y \in \overline{L} : \gamma(y) \sqsubseteq \gamma(x \,\overline{\nabla}\, y) \wr$$
$$\Longrightarrow \mathsf{lfp}^{\sqsubseteq}_{\perp} F \sqsubseteq \gamma(\overline{X}^{\overline{\ell}}) \quad \wr \text{as shown above} \wr$$

□

Why widenings cannot be monotone

- Let X and Y be such that $X \sqsubseteq Y$ (e.g. $X \sqsubseteq Y = F(X)$ since the iterates for F with widening ∇ are increasing)
- Assume that ∇ is monotone, we have

$$X \,\nabla\, Y \sqsubseteq Y \,\nabla\, Y$$

- We have seen that is reasonable to assume that $(Y \sqsubseteq X) \Longrightarrow (X \,\nabla\, Y = Y)$ (since e.g. if $Y = F(X) \sqsubseteq X$ then we have converged so their should be no other loss of information)

- In particular for $X = Y$, we have

$$Y \,\nabla\, Y = Y$$

- It follows that

$$X \,\nabla\, Y \sqsubseteq Y$$

which prevents extrapolations!

Example of non-monotone widening

– the classical widening on intervals is:

$$\bot \nabla X = X \nabla \bot = X$$
$$[\ell_0,\ u_0] \nabla [\ell_1,\ u_1] = [(\ell_1 < \ell_0\ ?\ -\infty \ ¿\ \ell_0),\ (u_1 > u_0\ ?\ +\infty\ ¿\ u_0)]$$

– Not monotone in its first argument: $[0,1] \sqsubseteq [0,2]$ but $[0,1] \nabla [0,2] = [0,+\infty] \not\sqsubseteq [0,2] = [0,2] \nabla [0,2]$

– Monotone in its second parameter: $(I' \sqsubseteq I'') \Longrightarrow (I \nabla I' \sqsubseteq I \nabla I'')$

Proof. - If $I = \bot$: $(I \nabla I' = \bot \nabla I' = I' \sqsubseteq I'' = \bot \nabla I'' = I \nabla I''$

- Else $I = [a,b] \neq \bot$. Then:
  · If $I' = \bot$ then $I \nabla I' = I \nabla \bot = I \sqsubseteq I \nabla I''$
  · Else $I' = [a',b'] \neq \bot$ so $I' \sqsubseteq I''$ implies $I'' = [a'',b''] \neq \bot$ with $a'' \leq a'$ and $b' \leq b''$.
    - For the lower bound, we have:
      - If $a' < a$ so $a'' \leq a' < a$ hence we have $I \nabla I' = [a,\_] \nabla [a',\_] = [-\infty,\_] \sqsubseteq I \nabla I'' = [a,\_] \nabla [a'',\_] = [-\infty,\_]$
      - Else, $a'' \geq a$, hence $I \nabla I' = [a,\_] \nabla [a',\_] = [a,\_] \sqsubseteq I \nabla I'' = [a,\_] \nabla [a'',\_] = (a'' \geq a\ ?\ [a,\_]\ ¿\ [-\infty,\_])$
    - Idem, for the upper bound.

□

Consequences of the absence of monotony of the widening

A *local improvement* of a static analysis may lead to a *global deterioration* of the precision.

– Example:

```
    X := 0;
{1}
    while true do
{2}
       if X = 0 then
{3}
         X := 1
{4}
       else
{5}
          X := 2
{6}
       fi
{7}
    od
```


– Analysis 1: $X = 0 \Longrightarrow X \in [0,2]$, locally imprecise

  1 $\rightarrow X \in [0,2]$ because of local imprecision
  2 $\rightarrow X \in \bot \nabla [0,2] = [0,2]$
  3 $\rightarrow X \in [0,0]$
  4 $\rightarrow X \in [1,1]$
  5 $\rightarrow X \in [1,2]$
  6 $\rightarrow X \in [2,2]$
  7 $\rightarrow X \in [1,1] \sqcup [2,2] = [1,2]$
  2 $\rightarrow X \in [0,2] \nabla [1,2] = [0,2]$, stable!

- Analysis 2: $X = 0 \Longrightarrow X \in [0,0]$, locally precise
  - $1 \rightarrow X \in [0,0]$ because of local precision
  - $3 \rightarrow X \in [0,0]$
  - $4 \rightarrow X \in [1,1]$
  - $5 \rightarrow \bot$
  - $6 \rightarrow \bot$
  - $7 \rightarrow X \in \bot \sqcup [1,1] = [1,1]$
  - $2 \rightarrow X \in [0,0] \nabla ([0,0] \sqcup [1,1]) = [0,+\infty]$
  - $3 \rightarrow X \in [0,0]$
  - $4 \rightarrow X \in [1,1]$
  - $5 \rightarrow [1,+\infty]$
  - $6 \rightarrow [2,2]$

  - $7 \rightarrow X \in [1,1] \sqcup [2,2] = [1,2]$
  - $2 \rightarrow X \in [0,+\infty] \nabla ([0,0] \sqcup [1,2]) = [0,+\infty]$, stable
- Remedies:
  - Use a *narrowing*, if possible
    - In the example, $[0,+\infty] \Delta [0,2] = [0,2]$ so that the final result is exact
    - The narrowing is not always able to compensate for the lack of precision of the widening, so a stable bound can be missed
  - Choose a *more precise widening*

Revisiting the soundness of decreasing iterations with narrowing

THEOREM. [13] Let $F \in L \stackrel{m}{\longmapsto} L$ be a monotone operator on the cpo $\langle L, \sqsubseteq, \sqcap\rangle$. Let $\langle \overline{L}, \overline{\sqsubseteq}\rangle$ be a poset. Let $\gamma \in \overline{L} \stackrel{m}{\longmapsto} L$ be a monotone concretization function. Let $\overline{F} \in \overline{L} \mapsto \overline{L}$ such that $F \circ \gamma \dot{\sqsubseteq} \gamma \circ \overline{F}$. Let $A \in \overline{L}$ be such that $\mathsf{lfp}_{\bot}^{\sqsubseteq} F \sqsubseteq \gamma(A)$. Assume that the narrowing $\Delta \in \overline{L} \times \overline{L} \mapsto \overline{L}$ satisfies:

- $\forall x, y \in \overline{L} : \gamma(y) \sqsubseteq \gamma(x \,\Delta\, y)$.

(which can be restricted to the case $y \,\overline{\sqsubseteq}\, x$ and even $y = \overline{F}(x)$)
Assume that the narrowing iteration sequence for $\overline{F}$ from A defined as

- $\overline{Y}^0 = \overline{F}(A)$ [14] (a)

- $\overline{Y}^{n+1} = \overline{Y}^n$ if $\overline{Y}^n \,\overline{\sqsubseteq}\, \overline{F}(\overline{Y}^n)$ (b)
- $\overline{Y}^{n+1} = \overline{Y}^n \,\overline{\Delta}\, \overline{F}(\overline{Y}^n)$ otherwise (c)

is ultimately stationary at rank $\ell \in \mathbb{N}$. Then $\mathsf{lfp}_{\bot}^{\sqsubseteq} F \sqsubseteq \gamma(\overline{Y}^{\ell})$. ■

PROOF. We prove that $\forall n \in \mathbb{N} : \mathsf{lfp}_{\bot}^{\sqsubseteq} F \sqsubseteq \gamma(\overline{Y}^n)$ so that the narrowing iteration can be stopped at any iteration rank [15].

- For the basis, we have $\mathsf{lfp}_{\bot}^{\sqsubseteq} F \sqsubseteq \gamma(A)$ by hypothesis (which results from the widening phase) and, so by fixpoint property, monotony of F, $F \circ \gamma \dot{\sqsubseteq} \gamma \circ \overline{F}$ and (a), we have $\mathsf{lfp}_{\bot}^{\sqsubseteq} F = F(\mathsf{lfp}_{\bot}^{\sqsubseteq} F) \sqsubseteq F(\gamma(A)) \sqsubseteq \gamma \circ \overline{F}(A) = \gamma(\overline{Y}^0)$

[13] Observe that $\overline{F}$ is not assumed to be monotone. If it is extensive, a narrowing is of no interest.
[14] $\overline{F}(A)$ has already been computed to stop the widening iteration so that it would be less efficient to restart from A.
[15] so e.g. the narrowing can be $x \,\Delta\, y = y$ and the iteration restricted to one step.

– Assume that $n \leq \ell$ and, by induction hypothesis, $\mathsf{lfp}_{\bot}^{\sqsubseteq} F \sqsubseteq \gamma(\overline{Y}^n)$. There are two cases according to the definition of the iterates:
  - If $\overline{Y}^n \mathrel{\overline{\sqsubseteq}} \overline{F}(\overline{Y}^n)$ then $\overline{Y}^{n+1} = \overline{Y}^n$ so that by induction hypothesis $\mathsf{lfp}_{\bot}^{\sqsubseteq} F \sqsubseteq \gamma(\overline{Y}^{n+1})$ and indeed, $n = \ell$.

– Otherwise, $n < \ell$ and $\overline{Y}^n \mathrel{\Delta} \overline{F}(\overline{Y}^n)$. In that case, we have

$$
\begin{array}{ll}
\mathsf{lfp}_{\bot}^{\sqsubseteq} F & \\
= F(\mathsf{lfp}_{\bot}^{\sqsubseteq} F) & \wr \text{fixpoint property} \wr \\
\sqsubseteq F(\gamma(\overline{Y}^n)) & \wr \text{ind. hyp. and } F \text{ monotone} \wr \\
\sqsubseteq \gamma(\overline{F}(\overline{Y}^n)) & \wr \text{by } F \circ \gamma \mathrel{\dot{\sqsubseteq}} \gamma \circ \overline{F} \text{ and transitivity} \wr \\
\sqsubseteq \gamma(\overline{Y}^n \mathrel{\Delta} \overline{F}(\overline{Y}^n)) & \wr \text{by } \forall x, y \in \overline{L} : \gamma(y) \sqsubseteq \gamma(x \mathrel{\Delta} y) \wr \\
= \gamma(\overline{Y}^{n+1}) & \wr \text{def. iterates} \wr
\end{array}
$$

We conclude by recurrence on n, noting that the iterates are stationary beyond ℓ. □

Design of Widening/Narrowing

Strategies to improve the precision of iterations with widening/narrowing (iteration threshold, unrolling, cut-points, history-based extrapolation)

– *Iteration threshold*: do not widen/narrow in the first iterations (e.g. in a loop), up to some threshold n

– *Unrolling*: semantically unroll the first iterates of a loop, so that, e.g.:

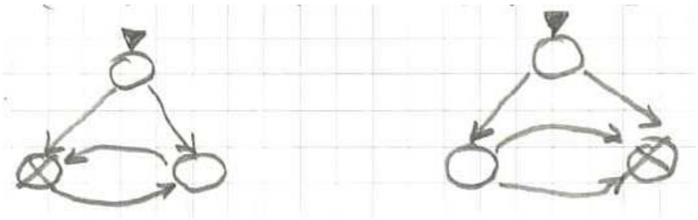
```
B := true;
while true do if B then I else C; B := false od
```

as found in some automatically generated code will be handled as:

```
I; while true do C od
```


Done n times, this is more precise than a temporization with iteration threshold since no join is performed at all in the first iterations

– Widening/narrowing/stabilization checks at *cut-points* only
  - Minimal number of cut-points
  - A cut point within each loop (more precisely, within each circular dependency)
  - The choice may not be unique (for irreducible dependence graphs)



– *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" [16] since the last widening/narrowing.

– Example:

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

- With widening/narrowing at cut-points:

```
{ i:_O_; j:_O_ }
  i := 1;
{ i:[1,+oo]; j:[1,+oo]? }
  while (i < 1000) do
    { i:[1,999]; j:[1,+oo]? }
      j := 1;
    { i:[1,+oo]; j:[1,+oo] }
      while (j < i) do
        { i:[2,+oo]; j:[1,1073741822] }
          j := (j + 1)
        { i:[2,+oo]; j:[2,+oo] }
      od;
    { i:[1,+oo]; j:[1,+oo] }
      i := (i + 1);
    { i:[2,+oo]; j:[1,+oo] }
  od
{ i:[1000,+oo]; j:[1,+oo]? }
```


- With history-based widening/narrowing:

```
{ i:_O_; j:_O_ }
  i := 1;
{ i:[1,1000]; j:[1,999]? }
  while (i < 1000) do
    { i:[1,999]; j:[1,999]? }
      j := 1;
    { i:[1,999]; j:[1,999] }
      while (j < i) do
        { i:[2,999]; j:[1,998] }
          j := (j + 1)
        { i:[2,999]; j:[2,999] }
      od;
    { i:[1,999]; j:[1,999] }
      i := (i + 1);
    { i:[2,1000]; j:[1,999] }
  od
{ i:[1000,1000]; j:[1,999]? }
```

– More generally, the extrapolation is more precise if we:
  - widen up to constants, ranges, ... given by declarations, tests, ...;
  - have the widening depend upon the iteration step, e.g. by:
    · introducing a threshold under which the least upper bound is used and above which widening is enforced;
    · awaiting for regular behaviors before widening within loops:
      - do not widen on the first iterate,
      - do not widen if a new branch of a test has just been taken within the loop body.

Thresholded/layered widening

Let $\langle L, \sqsubseteq, \bot, \top, \sqcup, \sqcap\rangle$ be a complete lattice. Let $T_1 \sqsubset T_2 \sqsubset \ldots : \sqsubset T_n = \top$ be finitely many elements of L. Define $T = \{T_1, \ldots, T_n\}$. The widening with thresholds T is

$$X \mathbin{\nabla_T} Y = (\!| Y \sqsubseteq X \mathrel{?} X \mathrel{¿} T_i |\!)$$
$$\text{where} \quad X \sqcup Y \sqsubseteq T_i$$
$$\text{and} \quad \forall T_j \in T : X \sqcup Y \sqsubseteq T_j \Longrightarrow T_i \sqsubseteq T_j$$

THEOREM. $X \mathbin{\nabla_T} Y$ is a widening. ■

PROOF. 1. ∇_T is a upper bound.
– If $Y \sqsubseteq X$ then $X \mathbin{\nabla_T} Y = X = X \sqcup Y$;
– Otherwise, $Y \sqsubseteq X = T_i \sqsupseteq X \sqcup Y$.

2. ∇_T enforces convergence. Given $\langle X_i,\ i \in \mathbb{N}\rangle$ define $Y_0 = X_0, \ldots, Y_{i+1} = Y_i \mathbin{\nabla} X_i$. Assume that $\langle Y_i,\ i \in \mathbb{N}\rangle$ is a strictly increasing chain. We have $Y_1 = X_0 \mathbin{\nabla_T} X_1 = T_{i_1}$ (since otherwise $X_1 \sqsubseteq X_0$ and $Y_1 = X_0$ so $\langle Y_i,\ i \in \mathbb{N}\rangle$ would not be a strictly increasing chain). Assume by induction hypothesis that $Y_k = T_{i_k}$ with $T_{i_1} \sqsubset \ldots \sqsubset T_{i_k}$. Then $Y_{k+1} = T_{i_k} \mathbin{\nabla_T} X_{k+1}$ since we cannot have $X_{k+1} \sqsubseteq Y_{k+1}$ which would imply that $Y_{k+1} = T_{i_k}$ in contradiction with the hypothesis that $\langle Y_i,\ i \in \mathbb{N}\rangle$ is a strictly increasing chain. So $Y_{k+1} = T_{i_{k+1}}$ with $T_{i_{k+1}} \sqsupseteq T_{i_k}$. But $T_{i_{k+1}} \neq T_{i_k}$ since otherwise $\langle Y_i,\ i \in \mathbb{N}\rangle$ would not be a strictly increasing chain. It follows, by recurrence that $\forall k \in \mathbb{N} : Y_k = T_{i_k}$ so $\langle T_{i_k},\ k \in \mathbb{N}\rangle$ is strictly increasing, a contradiction.

□

Widenings for pairs/tuples

– If
  - ∇_1 is a widening for $\langle L_1, \sqsubseteq_1\rangle$, and
  - ∇_2 is a widening for $\langle L_2, \sqsubseteq_2\rangle$,

  then

$$\langle x,\ y\rangle \mathbin{\nabla} \langle x',\ y'\rangle \stackrel{\text{def}}{=} \langle x \mathbin{\nabla_1} x',\ y \mathbin{\nabla_2} y'\rangle$$

  is a widening for $\langle L_1 \times L_2, \sqsubseteq_1 \times \sqsubseteq_2\rangle$ where $\sqsubseteq_1 \times \sqsubseteq_2$ is the componentwise ordering
– Idem for narrowing
– Idem for tuples

First-order functional widening

As we have seen, if:

- $f \in L \stackrel{\mathrm{m}}{\longmapsto} L$, ∇ is a widening on a poset $\langle L, \sqsubseteq\rangle$

then

$$\mathsf{lfp}^{\sqsubseteq} f \sqsubseteq \mathsf{lfp}^{\sqsubseteq} \lambda x \cdot x \mathbin{\nabla} f(x)$$

(and the second fixpoint can be computed iteratively starting from a prefixpoint $\bot \sqsubseteq f(\bot)$ in finitely many steps).

Example: Interval Analysis of Functions

Solve the second-order equation:

$$f = F(f) \text{ where } f(x) = [1,1] \sqcup (f(x) + [2,2])$$

for the argument $[0,0]$.

So we approximate by:

$$f = f \mathbin{\nabla} F(f) \text{ for argument } [0,0],$$

that is:

$$f([0,0]) = f([0,0]) \mathbin{\nabla_1} F(f([0,0]))$$
$$= f([0,0]) \mathbin{\nabla_1} F([1,1] \sqcup (f([0,0]) + [2,2]))$$

We let $X \stackrel{\mathrm{def}}{=} f([0,0])$ so we get a first order equation:

$$X = X \mathbin{\nabla_1} ([1,1] \sqcup (X + [2,2]))$$

The equation is solved iteratively as follows:

$X^0 = \bot$
$X^1 = X^0 \mathbin{\nabla_1} ([1,1] \sqcup (X^0 + [2,2])) = [1,1]$
$X^2 = X^1 \mathbin{\nabla_1} ([1,1] \sqcup (X^1 + [2,2]))$
$= [1,1] \mathbin{\nabla_1} ([1,1] \sqcup ([1,1] + [2,2]))$
$= [1,1] \mathbin{\nabla_1} [1,3] = [1,+\infty]$
$X^2 = [1,1] \mathbin{\nabla_1} [1,+\infty] = [1,+\infty]$
proving that $f(0)$ is greater than 1.

Second-order functional widening – I – finite domains

If

- $F \in (S \mapsto L) \stackrel{\mathrm{m}}{\longmapsto} (S \mapsto L)$, pointwise ordering;
- ∇ is a widening for $\langle L, \sqsubseteq\rangle$;
- S is a finite set

then

$$\mathsf{lfp}^{\dot\sqsubseteq} F \mathrel{\dot\sqsubseteq} \mathsf{lfp}^{\dot\sqsubseteq} \lambda f \cdot \lambda x \cdot f(x) \mathbin{\nabla} F(f)(x)$$

Note:
This can be seen as a system of equations:

$$\begin{cases} X_i = F_i(X_1, \ldots, X_n) \\ \quad i = 1, \ldots, n \end{cases}$$

where $S = \{1, \ldots, n\}$ and X_i is $f(i)$.
This is solved as:

$$\begin{cases} X_i = X \mathbin{\nabla} F_i(X_1, \ldots, X_n) \\ \quad i = 1, \ldots, n \end{cases}$$

with the usual remark that ∇ is needed only once around cycles of the dependence graph.

Possible divergence for infinite domains

– Note: the previous widening strategy fails for

$$f(x) = [1,1] \sqcup (f(x + [1,1]) + [2,2])$$

since $f([0,0])$ needs $f([1,1])$ which needs $f([2,2])$, etc.

Second-order functional widening – II – infinite domains

– If
- ∇_1 is a widening on $\langle L_1, \sqsubseteq_1 \rangle$,
- ∇_2 is a widening on $\langle L_2, \sqsubseteq_2 \rangle$
- $F \in (L_1 \stackrel{\mathrm{m}}{\longmapsto} L_2) \stackrel{\mathrm{m}}{\longmapsto} (L_1 \stackrel{\mathrm{m}}{\longmapsto} L_2)$

then

$$\mathsf{lfp}^{\dot{\sqsubseteq}_2} F \mathrel{\dot{\sqsubseteq}_2} \mathsf{lfp}\, \lambda f \cdot \lambda x \cdot f(x) \mathbin{\nabla_2} F(\lambda y \cdot f(x \mathbin{\nabla_1} y))(x)$$

where $\dot{\sqsubseteq}_2$ is the pointwise ordering on $L_1 \stackrel{\mathrm{m}}{\longmapsto} L_2$.

Reference

[3] P. Cousot and R. Cousot. Static determination of dynamic properties of recursive procedures. In *IFIP Conf. on Formal Description of Programming Concepts, St-Andrews, N.B., CA*, E.J. Neuhold (Ed.), pages 237–277, St-Andrews, N.B., Canada, 1977. North-Holland Publishing Company (1978).

Example of second-order functional widenings in infinite domains

$$F = \lambda f \cdot \lambda x \in 1, 1 \cdot \sqcup (f(x + [1,1]) + [2,2])$$

$\mathsf{lfp}^{\dot{\sqsubseteq}}_{\lambda x \cdot \bot} F$ is approximated as the least solution to:

$$\begin{aligned} f(x) &= f(x) \mathbin{\nabla_2} F(\lambda y \cdot f(x \mathbin{\nabla_1} y))(x) \\ &= f(x) \mathbin{\nabla_2} ([1,1] \sqcup (\lambda y \cdot f(x \mathbin{\nabla_1} y) x + [1,1]) + [2,2])) \\ &= f(x) \mathbin{\nabla_2} ([1,1] \sqcup (f(x \mathbin{\nabla_1} (x + [1,1])) + [2,2])) \end{aligned}$$

In order to compute $f([0,0])$ we follow a chatotic iteration strategy (see [3]):

– A table of pairs $\langle a,\ f(a)\rangle$ is maintained for needing arguments only, starting from $\langle [0,0],\ \bot\rangle$;

– We recompute $f^{n+1}(a)$ for the pair $\langle a,\ f^n(a)\rangle$ using $f^n(a)$ as the current approximation to $f(a)$

as long as:

– no new argument a' is needed;

– all needed pairs $\langle a,\ f(a)\rangle$ are stable.

- $f^0([0,0]) = \bot$
- $f^1([0,0])$
$= f^0([0,0])\,\nabla_2([1,1] \sqcup (f^0([0,0]\,\nabla_1([0,0]+[1,1]))+[2,2]))$
$= \bot\,\nabla_2\,([1,1] \sqcup (f^0([0,0]\ \nabla_1\ ([0,0]+[1,1]))+[2,2]))$
$= ([1,1] \sqcup (f^0([0,0]\ \nabla_1\ [1,1]) + [2,2]))$
$= ([1,1] \sqcup (f^0([0,+\infty]) + [2,2]))$
$= ([1,1] \sqcup \bot)$
$= [1,1]$
since $f([0,+\infty])$ has not yet been computed, hence:
- $f^0([0,+\infty]) = \bot$ ...

- $f^1([0,+\infty])$
$= f^0([0,+\infty])\nabla_2([1,1] \sqcup (f^0([0,+\infty]\nabla_1([0,+\infty]+[1,1]))+[2,2]))$
$= \bot\,\nabla_2\,([1,1] \sqcup (f^0([0,+\infty]\ \nabla_1\ [1,+\infty]) + [2,2]))$
$= ([1,1] \sqcup (f^0([0,+\infty]\ \nabla_1\ [1,+\infty]) + [2,2]))$
$= ([1,1] \sqcup (f^0([0,+\infty]) + [2,2]))$
$= ([1,1] \sqcup \bot)$
$= [1,1]$ ...

- $f^2([0,+\infty])$
$= f^1([0,+\infty])\nabla_2([1,1] \sqcup (f^1([0,+\infty]\nabla_1([0,+\infty]+[1,1]))+[2,2]))$
$= [1,1]\ \nabla_2\,([1,1] \sqcup (f^1([0,+\infty]) + [2,2]))$
$= [1,1]\ \nabla_2\,([1,1] \sqcup ([1,1] + [2,2]))$
$= [1,1]\ \nabla_2\,([1,1] \sqcup ([3,3]))$
$= [1,1]\ \nabla_2\,[1,3])$
$= [1,+\infty])$ ...

- $f^3([0,+\infty])$
$= f^2([0,+\infty])\nabla_2([1,1]\sqcup(f^2([0,+\infty]\nabla_1([0,+\infty]+[1,1]))+[2,2]))$
$= [1,\infty]\ \nabla_2\ ([1,1]\sqcup(f^2([0,+\infty]) + [2,2]))$
$= [1,\infty]\ \nabla_2\ ([1,1]\sqcup([1,+\infty] + [2,2]))$
$= [1,\infty]\ \nabla_2\ ([1,1]\sqcup([3,\infty]))$
$= [1,\infty]\ \nabla_2\ [1,\infty])$
$= [1,+\infty])$ ...

- $f^2([0,0])$
$= f^1([0,0])\nabla_2([1,1]\sqcup(f^1([0,0]\nabla_1([0,0]+[1,1]))+[2,2]))$
$= [1,1]\ \nabla_2\ ([1,1]\sqcup(f^0([0,\infty]) + [2,2]))$
$= [1,1]\ \nabla_2\ [1,+\infty])$
$= [1,+\infty]$

Everything needed is stable.

Note: This chaotic iteration strategy from [3] was used can be chosen as a semantics of procedures (in the finite case so no widening is needed) by Jones & Mycroft [4] under the popular name "minimal function graphs".

Reference

[4] N. Jones and A. Mycroft. Data flow analysis of applicative programs using minimal function graphs. Annual Symposium on Principles of Programming Languages archive Proceedings of the 13th ACM SIGACT-SIGPLAN symposium on Principles of programming languages table of contents St. Petersburg Beach, Florida, pp. 296–306, 1986.

Commented bibliography

- The very first report on static analysis in infinite abstract domains not satisfying the ACC with widening/narrowing:

[5] P. Cousot and R. Cousot. Static verification of dynamic type properties of variables. Res. rep. R.R. 25, Laboratoire IMAG, Université scientifique et médicale de Grenoble, Grenoble, France. Nov. 1975, 18 p.

- The first published paper on the subject:

[6] P. Cousot and R. Cousot. Static determination of dynamic properties of programs. In *Proceedings of the Second International Symposium on Programming*, pages 106–130. Dunod, Paris, France, 1976.

- The first published paper on the subject in the US and most cited reference:

[7] 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.

- Extension to recursive procedures:

[8] P. Cousot and R. Cousot. Static determination of dynamic properties of recursive procedures. In *IFIP Conf. on Formal Description of Programming Concepts, St-Andrews, N.B., CA*, E.J. Neuhold (Ed.), pages 237–277, St-Andrews, N.B., Canada, 1977. North-Holland Publishing Company (1978).

- Presentation using transition systems (i.e. language independent semantics and equational analyzers, if you read french):

[9] P. Cousot. «*Méthodes itératives de construction et d'approximation de points fixes d'opérateurs monotones sur un treillis, analyse sémantique de programmes*». Thèse d'État ès sciences mathématiques, Université scientifique et médicale de Grenoble, Grenoble, 21 mars 1978.

otherwise, see

[10] P. Cousot. Semantic foundations of program analysis. In S.S. Muchnick and N.D. Jones, editors, *Program Flow Analysis: Theory and Applications*, chapter 10, pages 303–342. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, USA, 1981.

– Showing that widening/narrowing is strictly more powerful than abstraction/concretization (and therefore abstract model-checking which is a particular case):

[11] P. Cousot and R. Cousot. Comparing the Galois Connection and Widening/Narrowing Approaches to Abstract Interpretation, invited paper. In M. Bruynooghe and M. Wirsing, eds., *Programming Language Implementation and Logic Programming, Proc. 4th In. Symp., PLILP'92*, Leuven, Belgium, 13–17 Aug. 1992, LNCS 631, p. 269–295. Springer-Verlag, 1992.

– The first use of a concretization-only framework (which is used in this course in absence of abstract lubs whence of best approximations):

[12] P. Cousot and N. Halbwachs. Automatic discovery of linear restraints among variables of a program. In *Conference Record of the Fifth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 84–97, Tucson, Arizona, 1978. ACM Press, New York, NY, U.S.A.

– Making clear the possible choice of abstraction-concretization, abstraction-only and concretization-only frameworks:

[13] P. Cousot and R. Cousot. Abstract Interpretation Frameworks. *Journal of Logic and Computation*, 2(4):511–547, August 1992.

– Notes on a course on static analysis (in equational form for finite domains):

[14] P. Cousot. The Calculational Design of a Generic Abstract Interpreter. Course notes for the International Summer School Marktoberdorf (Germany) on Calculational System Design from July 28 to August 9, 1998 organized by F.L. Bauer, M. Broy, E.W. Dijkstra, D. Gries and C.A.R. Hoare.

published in structural form:

[15] P. Cousot. The Calculational Design of a Generic Abstract Interpreter. In M. Broy and R. Steinbrüggen (eds.): *Calculational System Design*. NATO ASI Series F. Amsterdam: IOS Press, 1999.

THE END

My MIT web site is http://www.mit.edu/~cousot/

The course web site is http://web.mit.edu/afs/athena.mit.edu/course/16/16.399/www/.