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

In general the iterates $\bot$, $f(\bot), \ldots, f^n(\bot), \ldots$ 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 $x^0 \triangleq \bot$, $\ldots$, $x^{n+1} \triangleq x^n \triangledown f^n(\bot), \ldots$ using a widening $\triangledown$. 
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;
- \( z \uparrow y \) is the least threshold \( T_i \) greater than or equal to both \( z \) and \( y \).

\[ \text{Approximate fixpoint abstraction} \]

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

If
- \( L(\sqsubseteq, \perp, \top, \sqcup, \sqcap) \) is a complete lattice;
- \( F \in L \xrightarrow{\text{uc}} L \) is continuous for \( \sqsubseteq \);
- \( L^\parallel(\sqsubseteq^\parallel, \perp^\parallel, \top^\parallel, \sqcup^\parallel, \sqcap^\parallel) \) is a complete lattice;
- \( L \xrightarrow{\alpha} L^\parallel \) is a Galois connection;
- \( F^\parallel \in L^\parallel \rightarrow L^\parallel \) is continuous for \( \sqsubseteq^\parallel \);
- \( \alpha \circ F \circ \alpha \sqsubseteq F^\parallel \);

then:
\[
\llcorner_F F \sqsubseteq \alpha(\llcorner_F F^\parallel)
\]

\[ \text{Proof.} \]
1. \( \alpha \) is monotonic:
   \[
   f \sqsubseteq g \quad \implies f \sqsubseteq g \land \alpha(g) \sqsubseteq \alpha(g)
   \]
   \[
   \implies f \sqsubseteq g \land g \sqsubseteq \gamma(\alpha(g))
   \]
   \[
   \implies f \sqsubseteq \gamma(\alpha(g))
   \]
   \[
   \implies \alpha(f) \sqsubseteq^\parallel \alpha(g)
   \]

2. \( \perp \sqsubseteq \gamma(\perp^\parallel) \)
   \[
   \implies F^0(\perp) \sqsubseteq \gamma(F^0(\perp^\parallel))
   \]
   \[
   \implies \alpha(F^0(\perp)) \sqsubseteq^\parallel F^0(\perp^\parallel)
   \]
3. \( \alpha(F^n(\bot)) \sqSupset F_{\bot}^{\#}(\bot) \) \[\text{induction hypothesis}\]
\[\implies F^n(\bot) \sqSupset \gamma(F_{\bot}^{\#}(\bot)) \]  
[Galoi connection]
\[\implies F(F^n(\bot)) \sqSupset F \circ \gamma(F_{\bot}^{\#}(\bot)) \]  
[F monotonic]
\[\implies \alpha(F(F^n(\bot))) \sqSupset \alpha \circ F \circ \gamma(F_{\bot}^{\#}(\bot)) \]  
[\(\alpha\) monotonic]
\[\alpha \circ F \circ \gamma(F_{\bot}^{\#}(\bot)) \sqSupset F_{\bot}^{\#}(\bot) \]  
[hypothesis]
\[\implies \alpha(F(F^n(\bot))) \sqSupset F_{\bot}^{\#}(\bot) \]  
[transitivity]
\[\implies \alpha(F^{n+1}(\bot)) \sqSupset F_{\bot}^{\#+1}(\bot) \]  
[def. iterates]

4. \( \forall n: \alpha(F^n(\bot)) \sqSupset F_{\bot}^{\#n}(\bot) \) \[\text{[2., 3., recurrence]}\]
\[\implies \forall n: \alpha(F^n(\bot)) \sqSupset \bot_{m \geq 0} F_{\bot}^{\#m}(\bot) \]  
[lub]
\[\implies \alpha(F^0(\bot)) \sqSupset F_{\bot}^{\#0}(\bot) \]  
[Galoi connection]
\[\implies \forall n: \alpha(F^n(\bot)) \sqSupset \text{ifp } F_{\bot}^{\#} \]  
[Tarski constructive th.]
\[\implies \forall n: F^n(\bot) \sqSupset \gamma(\text{ifp } F_{\bot}^{\#}) \]  
[Galoi connection]
\[\implies \bot_{n \geq 0} F^n(\bot) \sqSupset \gamma(\text{ifp } F_{\bot}^{\#}) \]  
[least upper bound]
\[\implies \text{ifp } (F) \sqSupset \gamma(\text{ifp } F_{\bot}^{\#}) \]  
[Tarski constructive th.]

Note: we need \( \alpha \circ F \circ \gamma(X) \sqSupset F_{\bot}(X) \) only when \( X = F_{\bot}^{\#n}(\bot) \), \( n \in \mathbb{N} \) and so we can relax the hypothesis and assume e.g. \( \forall X \sqSupset \) ifp \( F_{\bot}^{\#} : \alpha \circ F \circ \gamma(X) \sqSupset F_{\bot}(X) \)

---

**Example:**

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

- \((L, \sqSupset, \sqSupset)\) is a cpo
- \(F \in L \overset{m}{\rightarrow} L\) is monotonic for \(\sqSupset\)
- \(a \in L\) is a prefixpoint of \(F\), i.e.: \(a \sqSupset F(a)\)
- \((L, \sqSupset, \sqSupset)\) is a cpo
- \(F \in L \overset{m}{\rightarrow} L\) is monotonic for \(\sqSupset\)
- \((L, \sqSupset) \overset{\gamma}{\leftarrow} \alpha \) is a Galois connection
- \(\forall y \in L : y \sqSupset \text{ifp } \alpha(a) \implies \alpha \circ F \circ \gamma(y) \sqSupset F(y)\)

\[\iff \forall x \in L : \alpha(x) \sqSupset \text{ifp } \alpha(a) \implies F(x) \sqSupset \gamma \circ F \circ \alpha(x)\]
\[ \iff \forall x \in L : \alpha(x) \sqsupseteq \mathcal{F}(a) \implies \alpha \circ \mathcal{F}(x) \sqsubseteq \mathcal{F} \circ \alpha(x) \]
\[ \iff \forall y \in L : y \sqsupseteq \mathcal{F}(a) \implies \alpha \circ \mathcal{F}(y) \sqsubseteq \gamma \circ \mathcal{F}(y) \]
\[ \text{then } \mathcal{F}(a) \sqsubseteq \gamma(\mathcal{F}(a)) \]

**Proof.** – The equivalence of the different statements of overapproximation

Of \( \mathcal{F} \) by \( \mathcal{F} \) can be proved as follows:

\[ \forall y \in L : y \sqsupseteq \mathcal{F}(a) \implies \alpha \circ y \circ \mathcal{F}(y) \]
\[ \implies \forall x \in L : \alpha(x) \sqsupseteq \mathcal{F}(a) \implies \alpha \circ \mathcal{F}(x) \sqsubseteq \mathcal{F}(x) \]
\[ \alpha = \gamma \text{ is extensive, } \mathcal{F} \]
\[ \text{and } \alpha \text{ are monotone, def. composition } \}

---

- We let \( \langle F^\delta, \delta \in \mathbb{D} \rangle \) be the iterates of \( F \) starting from \( a \). By the hypothesis that \( \langle L, \sqsubseteq, \sqcup \rangle \) is a cpo, \( F \) is monotonic and \( a \sqsubseteq \mathcal{F}(a) \), they are a well-defined increasing chain and an ordinal \( \epsilon \) such that \( \mathcal{F}^\epsilon = F^\delta \).

- We let \( \langle F^\delta, \delta \in \mathbb{D} \rangle \) be the iterates of \( \mathcal{F} \) starting from \( \alpha(a) \). Observe that \( \alpha(a) \sqsupseteq \mathcal{F}(a) \) and so \( \alpha \circ \mathcal{F}(a) \sqsubseteq \mathcal{F} \circ \alpha(a) \). We have \( \alpha \circ \mathcal{F} \sqsubseteq \mathcal{F} \circ \alpha \) and is monotone by \( \langle L, \sqsubseteq, \sqcup \rangle \) and \( \mathcal{F} \) are defined in the constructive version of Taraki’s fixpoint theorem, that they are a well-defined increasing chain and an ordinal \( \epsilon' \) such that \( \mathcal{F}^\epsilon = \mathcal{F}^\delta \).

- We have

\[ F^\delta = \alpha \circ \gamma \circ \mathcal{F}(a) = \gamma(\mathcal{F}^\delta) \]

---

- If \( \gamma(\mathcal{F}^\delta) \) by induction hypothesis, then \( \mathcal{F}^\delta \sqsubseteq \mathcal{F}(a) \) and so

\[ \mathcal{F}^{\delta+1} = \mathcal{F}(\mathcal{F}^\delta) \sqsubseteq \gamma \circ \mathcal{F}(\mathcal{F}^\delta) = \mathcal{F}(\mathcal{F}^{\delta+1}) \]

- If \( \delta \) is a limit ordinal and, by induction hypothesis, \( \forall \delta < \lambda : F^\delta \sqsubseteq \gamma(\mathcal{F}^\delta) \), then \( F^\delta \sqsubseteq \mathcal{F}^\lambda \) and so

\[ \alpha(\mathcal{F}^\delta) = \alpha(\bigcup_{\beta < \lambda} \mathcal{F}^\beta) = \bigcup_{\beta < \lambda} \alpha(\mathcal{F}^\beta) \]

- By transfinite induction, \( \forall \delta \in \mathbb{D} : F^\delta \sqsubseteq \gamma(\mathcal{F}^\delta) \).

- Finally, \( \mathcal{F}(a) \subseteq \mathcal{F}^\delta \subseteq \gamma(\mathcal{F}^\delta) \).

\[ \square \]
Example:

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

---

**Soundness and (in-)completeness of abstractions**

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

- **Soundness:**
  \[ \text{fix}_\alpha \subseteq F \subseteq \gamma(\overline{P}) \]

---

**À la Kleene, continuous abstraction function-based fixpoint approximation**

If

- \( \langle L, \sqsubseteq, \sqcap \rangle \) is a cpo
- \( F \in L \xrightarrow{m} L \) is monotonic for \( \sqsubseteq \)
- \( a \in L \) is a prefixpoint of \( F \), i.e.: \( a \sqsubseteq F(a) \)
- \( \langle \overline{L}, \overline{\sqsubseteq}, \overline{\sqcap} \rangle \) is a cpo
- \( \overline{F} \in \overline{L} \xrightarrow{m} \overline{L} \) is monotonic for \( \overline{\sqsubseteq} \)
- \( \alpha \in L \xrightarrow{uc} \overline{L} \) is upper-continuous
- \( \alpha \circ F \in \overline{\text{fix}}_{\alpha} \circ \alpha \)

then

\[ \alpha(\text{fix}_\alpha \subseteq F) \subseteq \text{fix}_\alpha \subseteq \gamma(\overline{F}) \]
Proof. – We let $\langle F^\delta, \delta \in \mathbb{D} \rangle$ 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 $\varepsilon$ such that $\operatorname{Ifp}^\varepsilon_a F = F^\varepsilon$.

- We let $\langle F^\delta, \delta \in \mathbb{D} \rangle$ be the iterates of $F$ starting from $a(a)$. We have $\overline{F}(a(a)) \supseteq \overline{a}(F(a)) \supseteq \overline{a}(a)$ since $a = F \sqsubseteq F = a$, $F(a) \sqsubseteq a$ and $a$ is upper-continuous whence monotone. So $a(a)$ is a prefixpoint of the monotonic operator $F$ on the cpo $\langle L, \sqsubseteq, \sqcup \rangle$ proving, as in the constructive version of Tarski’s fixpoint theorem, that they are a well-defined increasing chain and there exists an ordinal $\varepsilon'$ such that $\operatorname{Ifp}^\varepsilon_{a(a)} F = F^{\varepsilon'}$.

- We have
  - $\overline{a}(F^\varepsilon) = \overline{a}(a) = F^\varepsilon$
  - If $\overline{a}(F^\delta) \supseteq F^\delta$ by induction hypothesis, then
    $\overline{a}(F^{\delta + 1}) = \overline{a}(F(F^\delta)) \supseteq \overline{F}(\overline{a}(F^\delta)) \supseteq \overline{F}(F^\delta) = F^{\delta + 1}$

\[ \text{Again, } \langle L, \sqsubseteq, \sqcup \rangle \text{ and } F \sqsubseteq F \text{ need only be max}(\varepsilon, \varepsilon')\text{-cpos.} \]

---

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

We have $\overline{a}(\operatorname{Ifp}^{\varepsilon \varepsilon}_a F) \supseteq \overline{F}$ where $a = \bot$ and $\overline{a}(a) = \bot$.

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

À la Kleene, monotone concretization-based fixpoint approximation

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

then $\operatorname{Ifp}^{\varepsilon \varepsilon}_{\gamma(a)} F \sqsubseteq \gamma(\operatorname{Ifp}^{\varepsilon \varepsilon}_a F)$
\textbf{Proof.} – Observe that the hypotheses that $\mathcal{M}$ and $\gamma(\mathcal{M})$ are respective prefixpoints of $\mathcal{F}$ and $\mathcal{F}$ are independent, as shown by the following examples:

- We let $\langle F^\delta, \delta \in \Omega \rangle$ be the iterates of $F$ starting from $\gamma(\mathcal{M})$. By the hypothesis that $\langle L, \sqsubseteq, \sqcup \rangle$ is a cpo, $F$ is monotonic and $\gamma(\mathcal{M}) \sqsubseteq F(\gamma(\mathcal{M}))$, they are a well-defined increasing chain and there is an ordinal $\epsilon$ such that $\lf \gamma(\mathcal{M}) \quad F = F^\epsilon$.

- We let $\langle \tilde{F}^\delta, \delta \in \Omega \rangle$ be the iterates of $\mathcal{F}$ starting from $\mathcal{M}$. By the hypothesis that $\mathcal{M}$ is a prefixpoint of the monotonic operator $\mathcal{F}$ on the cpo $\langle L, \sqsubseteq, \sqcup \rangle$, that they are a well-defined increasing chain and, as shown in the constructive version of Tarski’s fixpoint theorem, there is an ordinal $\epsilon'$ such that $\lf \mathcal{M} \quad \tilde{F} = \mathcal{F}^\epsilon'$.

- We have $\mathcal{M}$ since again $\langle L, \sqsubseteq, \sqcup \rangle$ and $\langle L, \sqsubseteq, \sqcup \rangle$ need only be $\max(\epsilon, \epsilon')$-cpos.

\hspace{1cm} \rhd

\begin{example}
\end{example}

\textbf{Example:}

\begin{tikzpicture}
\node (A) {$F$};
\node (B) [below of=A] {$\mathcal{F}$};
\node (C) [below of=B] {$\mathcal{M}$};
\draw [->] (A) to node [above] {$\gamma$} (B);
\draw [->] (B) to node [above] {$\gamma$} (C);
\draw [->] (A) to node [above] {$F$} (B);
\end{tikzpicture}

$\gamma$ monotonic, $F = \gamma \sqsubseteq \gamma$.

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

\begin{itemize}
\item $F^0 = \gamma(\mathcal{M}) \sqsubseteq \gamma(\mathcal{M}) = \mathcal{F}^0$
\item If $F^\delta \sqsubseteq \gamma(\mathcal{F}^\delta)$ by induction hypothesis, then $F^{\delta + 1} = F(F^\delta) \sqsubseteq F \circ \gamma(\mathcal{F}^\delta) \sqsubseteq \gamma(\mathcal{F}^{\delta + 1})$.
\item If $\lambda$ is a limit ordinal and, by induction hypothesis, $\forall \delta < \lambda : F^\delta \sqsubseteq \gamma(\mathcal{F}^\delta)$, then
\[
\gamma(\mathcal{F}^\lambda) = \bigcup_{\beta < \lambda} \bigcup_{\beta < \lambda} \gamma(\mathcal{F}^\beta) \sqsubseteq \gamma(\bigcup_{\beta < \lambda} \bigcup_{\beta < \lambda} \mathcal{F}^\beta)
\]
\hspace{1cm} (since $\gamma$ is monotone)
\item $\mathcal{M}$ and so $\mathcal{M} \sqsubseteq \gamma(\mathcal{F}^\dot{\lambda})$.
\item By transfinite induction, $\forall \delta \in \Omega : F^\delta \sqsubseteq \gamma(\mathcal{F}^\delta)$.
\item In conclusion, $\lf \gamma(\mathcal{M}) \quad F = \mathcal{F}^\dot{\lambda} = \mathcal{F}^{\max(\epsilon, \epsilon')} \sqsubseteq \gamma(\mathcal{F}^{\max(\epsilon, \epsilon')}) = \gamma(\mathcal{F}^{\dot{\epsilon}'}) = \gamma(\lf F)$. \hfill $\Box$
\end{itemize}
\[\alpha(F^\mathbb{P})\]
\[= \alpha(\bigcap \{x \in \mathcal{D}^\mathbb{P} \mid F^\mathbb{P}(x) \sqsubseteq x\}) \tag{Tarski}\]
\[\sqsubseteq \bigcap \{\alpha(x) \mid x \in \mathcal{D}^\mathbb{P} \land F^\mathbb{P}(x) \sqsubseteq x\} \tag{\alpha \text{ monotone}}\]
\[\sqsubseteq \bigcap \{y \in \mathcal{D}^\mathbb{P} \mid \land F^\mathbb{P}(y) \sqsubseteq y\} \tag{by (b)}\]
\[= F^\mathbb{P} \sqsubseteq \mathbb{P} \tag{Tarski} \]
\[
\textbf{Sufficient conditions for iterative fixpoint computation convergence}
\]
- Given a language \(\mathcal{L}\), we have seen that program properties can be defined in fixpoint form as
\[
\downarrow \lfloor P \rfloor \sqsubseteq \uparrow \downarrow F \downarrow \lfloor P \rfloor
\]
where \(F \downarrow \lfloor P \rfloor\) is a monotone operator on a cpo
\[
\langle L \downarrow \lfloor P \rfloor, \sqsubseteq \downarrow \lfloor P \rfloor, \sqsubseteq \uparrow \downarrow \lfloor P \rfloor \rangle
\]
defined by structural induction on the syntactic structure of the program \(P\).

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

- A static analyzer is specified by an abstraction:
\[
\langle L \downarrow \lfloor P \rfloor, \sqsubseteq \downarrow \lfloor P \rfloor \rangle \xrightarrow{\gamma \downarrow \lfloor P \rfloor} \langle \overline{L} \downarrow \lfloor P \rfloor, \sqsubseteq \downarrow \lfloor P \rfloor \rangle
\]
and an abstract transformer:
\[
\overline{F} \uparrow \gamma \downarrow \lfloor P \rfloor \circ F \downarrow \lfloor P \rfloor \circ \gamma \downarrow \lfloor P \rfloor
\]
which are both defined compositionally, by induction on the syntactic structure of \(P \in \mathcal{L}\).
- The static analyzer has the form
  \[
  \langle \mathcal{L}[P], \sqsubseteq[P], \perp[P], \sqcup[P], \overline{\mathcal{F}}[P]\rangle \\
  := \text{syntax\_analysis}(P);
  \]
  \[
  X := \perp[P]; \\
  \text{repeat} \\
  Y := \overline{\mathcal{F}}[P](X); \\
  \text{stable} := Y \sqsubseteq[P] X; \\
  X := Y \\
  \text{until stable;} \\
  \text{diagnostic}(P, X)
  \]

- Since \( \perp[P] \sqsubseteq[P][P] \overline{\mathcal{F}}[P](\perp[P]) \) and \( \overline{\mathcal{F}}[P] \) is monotone, the successive values of \( X \) form a \( \sqsubseteq[P] \) increasing chain (but maybe for the last iterate where equality can hold). The stabilization test implies, if and when the loop exists, that \( X = \overline{\mathcal{F}}[P](X) \). So upon termination, if ever, \( X = \llbracket P \rrbracket \overline{\mathcal{F}}[P] \) so that we can apply the soundness result.

- As far as termination is concerned, it follows that
  - The iteration terminates if the lattice \( \mathcal{L} \) is finite
  - The iteration terminates if the lattice \( \overline{\mathcal{L}} \) 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 \( \rho(Z) \) by intervals:

- In this case, one can choose a coarser abstraction \( \alpha \) (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 \( \alpha \) is then to ensure that abstract properties have efficient computer representations, while convergence is treated otherwise, by widening/narrowing).
Iteration acceleration by extrapolation

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

\[
\langle \wp(\mathbb{Z}), \subseteq \rangle \xleftarrow{\gamma_I} \langle \mathbb{I}, \subseteq \rangle
\]

where

- \( \mathbb{I} \overset{\text{def}}{=} \{ [a, b] \mid a, b \in \mathbb{Z} \cup \{-\infty, +\infty\} \land a \leq b \} \)
- \( \bot \subseteq \sqsubseteq \subseteq [a, b] \)
- \( [a, b] \subseteq [a', b'] \) \( \overset{\text{def}}{=} (a' \leq a \land b \leq b') \)
- \( \alpha_{\mathbb{I}}(\emptyset) \overset{\text{def}}{=} \bot, \) when \( X \neq \emptyset, \alpha_{\mathbb{I}}(X) \overset{\text{def}}{=} [\min X, \max X] \)

where \( \min X = -\infty \) when \( X \) has no minimum in \( \mathbb{Z} \) and \( \max X = +\infty \) when \( X \) has no maximum in \( \mathbb{Z} \).

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

- The lattice of intervals is

- 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 \(-\infty = \text{min_int} \) and \(+\infty = \text{max_int} \) 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 \( \Sigma \equiv \{0, 1, 2, 3, 4\} \times \text{[min\_int; max\_int]} \)

- The abstraction is

\[
\alpha_p \in \wp(\Sigma) \mapsto \prod_{i=0}^{4} \wp([\text{min\_int}; \text{max\_int}])
\]
\[
\alpha_p(X) \equiv \prod_{i=0}^{4} \{x \mid (i, x) \in X\}
\]
\[
\alpha(X) \equiv \prod_{i=0}^{4} (\alpha_p(X)_i = 0 \land \lnot \exists \min \alpha_p(X)_i, \max \alpha_p(X)_i)
\]

- A functional encoding in OCaml could be:

```ocaml
1 type interval = BUT | INT of (int * int);;
2 let less x y = match x,y with
3 | BUT, _ -> true
4 | _, BUT -> false
5 | INT (a,b), INT (c,d) -> (a < c) && (b < d);;
6 let join x y = match x,y with
7 | BUT, _ -> y
8 | _, BUT -> x
9 | INT (a,b), INT (c,d) -> INT (min a c, max b d);;
10 let meet x y = match x,y with
11 | BUT, _ -> BUT
12 | _, BUT -> BUT
13 | INT (a,b), INT (c,d) ->
14 | if (b < c) or (d < a) then BUT
15 | else INT (max a c, min b d);;
16 let f (x0,x1,x2,x3,x4) =
17 | (INT (min_int,max_int),
18 | join (INT (1,1)) x3,
19 | meet x1 (INT (min_int,max_int)),
```

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

\[
\begin{align*}
\text{INT} &\rightarrow \text{BST} \\
\text{INT} (a, b) & \\
\text{let } a' = \text{ if } a < \text{max_int} \text{ then } a+1 \text{ else max_int in} \\
\text{let } b' = \text{ if } b < \text{max_int} \text{ then } b+1 \text{ else max_int in} \\
\text{INT} (a', b')
\end{align*}
\]

- The chaotic iterates from the infimum \( \bot^5 \) are as follows

\[
\begin{align*}
x_0 &= \{ x : [-402341824, -402341823] \} \\
x_1 &= \{ x : [1, 4] \} \\
x_2 &= \{ x : [1, 3] \} \\
x_3 &= \{ x : [2, 4] \} \\
x_4 &= \{ x : [1, 5] \} \\
x_5 &= \{ x : [1, 6] \} \\
\ldots
\end{align*}
\]

**Intuition for convergence acceleration**

1. Speed-up the convergence of the increasing iteration \( X^0 = \bot, \ldots, X^{n+1} = F(X^n), \ldots \), \( \hat{A} \) in order to reach a post-fixpoint \( \hat{A} : F(\hat{A}) \sqsubseteq \hat{A} \) so that by Tarski:
   \[
   \text{lfp } F \sqsubseteq \hat{A} \sim \text{WIDENING } \bigwedge
   \]

2. Speed up the convergence of the decreasing iteration \( Y^0 = \hat{A}, \ldots, Y^{n+1} = F(Y^n), \ldots \), \( \hat{A} \) so as to stay above the least fixpoint \( \text{lfp } F \sqsubseteq \hat{A} \sim \text{NARROWING } \bigtriangleup \)
Example of widening for interval analysis

- \( \overline{L} = \{ \bot \} \cup \{ [l, u] | l \in \mathbb{Z} \cup \{-\infty\} \land u \in \mathbb{Z} \cup \{+\infty\} \land l \leq u \} \)

- The widening extrapolates unstable bounds to infinity:

\[
\begin{align*}
\bot \triangledown X &= X \\
X \triangledown \bot &= X \\
[l_0, u_0] \triangledown [l_1, u_1] &= \left( \begin{array}{l}
(l_1 < l_0 \iff -\infty \leq l_0), \\
(u_1 > u_0 \iff +\infty \leq u_0) \end{array} \right)
\end{align*}
\]

- Example:

- Not monotone. For example \([0, 1] \subseteq [0, 2]\) but \([0, 1] \triangledown [0, 2] = [0, +\infty]  \not\subseteq [0, 2] = [0, 2] \triangledown [0, 2] \)
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)\$
    \text{while(y)};
  ```

  consists in solving the equation:

  $$X = ([1, 1] \cup (X \oplus [2, 2])) \cap [-\infty, 99]$$

  where \(0 \oplus I = I \oplus 0 = 0\) and \([a, b] \oplus [c, d] = [a + c, b + d]\) with \(-\infty + x = x - \infty = -\infty\) and \(+\infty + x = x + \infty = +\infty\).

---

Definition of a widening

A widening \(\triangledown \in \mathcal{P} \times \mathcal{P} \mapsto \mathcal{P}\) on a poset \(\langle \mathcal{P}, \sqsubseteq \rangle\) satisfies:

- \(\forall x, y \in \mathcal{P} : x \sqsubseteq (x \triangledown y) \land y \sqsubseteq (x \triangledown y)\)

- For all increasing chains \(x^0 \sqsubseteq x^1 \sqsubseteq \ldots\) the increasing chain \(y^0 \overset{\text{def}}{=} x^0, \ldots, y^{n+1} \overset{\text{def}}{=} y^n \triangledown x^{n+1}, \ldots\) is not strictly increasing.

Two different main uses:

- Approximate missing lub.

- Convergence acceleration\(^4\);

---

Ascending abstract iteration sequence with widening:

\[
\begin{align*}
\hat{X}^0 &= \emptyset \\
\hat{X}^1 &= \hat{X}^0 \triangledown \left( ([1, 1] \cup (\hat{X}^0 \oplus [2, 2])) \cap [-\infty, 99] \right) \\
&= \emptyset \triangledown [1, 1] \\
&= [1, 1] \\
\hat{X}^2 &= \hat{X}^1 \triangledown \left( ([1, 1] \cup (\hat{X}^1 \oplus [2, 2])) \cap [-\infty, 99] \right) \\
&= [1, 1] \triangledown [1, 3] \\
&= [1, +\infty] \\
\hat{X}^3 &= \hat{X}^2 \triangledown \left( ([1, 1] \cup (\hat{X}^2 \oplus [2, 2])) \cap [-\infty, 99] \right) \\
&= [1, +\infty] \triangledown [1, 99] \\
&= [1, +\infty]
\end{align*}
\]

---

Upward iteration with widening

- Let \(F\) be an operator on a poset \(\langle \mathcal{P}, \sqsubseteq \rangle\);

- Let \(\triangledown \in \mathcal{P} \times \mathcal{P} \mapsto \mathcal{P}\) be a widening;

- The iteration sequence with widening \(\triangledown\) for \(F\) from \(\bot\) is \(X^n, n \in \mathbb{N}\):

  - \(X^0 = \bot\)

  - \(X^{n+1} = X^n \triangledown F(X^n)\) if \(F(X^n) \sqsubseteq (X^n)\)

  - \(X^{n+1} = X^n \triangledown F(X^n)\) if \(F(X^n) \not\sqsubseteq X^n\)

---

\(^4\) A widening operator can be used to effectively compute an upper approximation of the least fixpoint of \(F \in \mathcal{T} \mapsto \mathcal{T}\) starting from below when \(\mathcal{T}\) is computer representable but does not satisfy the ascending chain condition.
Correctness of the upward iteration with widening to upper-approximate a least-fixpoint by a post-fixpoint

If
- \( L(\sqsubseteq, \sqcap, \sqcup) \) is a poset,
- \( \varphi \in L \xrightarrow{m} L \) and
- \( \triangledown \) is a widening operator
then the increasing chain:
- \( \hat{X}^0 = \bot \),
- \( \hat{X}^{k+1} = \hat{X}^k \) if \( \varphi(\hat{X}^k) \sqsubseteq \hat{X}^k \)
- \( \hat{X}^{k+1} = \hat{X}^k \triangledown \varphi(\hat{X}^k) \) otherwise
for \( k \in \mathbb{N} \) is stationary with limit \( \hat{X}^\ell \) such that \( \text{lfp} \varphi \sqsubseteq \hat{X}^\ell \).

\[ \begin{align*}
\Rightarrow & \text{ the chain } \varphi(\hat{X}^k), \, k \in \mathbb{N} \text{ is increasing} \quad [\text{monotony}] \\
\Rightarrow & \text{ the chain } \hat{X}^k, \, k \in \mathbb{N} \text{ is stationary.} \quad [(c)] \\
\Rightarrow & \text{ For the limit } \hat{X}^\ell \text{ where } \ell \in \mathbb{N}, \text{ we have:} \\
\Rightarrow & \quad \forall k \leq \ell : \hat{X}^k \sqsubseteq \hat{X}^\ell \quad [\text{increasing chain}] \\
\Rightarrow & \quad m \geq \ell \land \hat{X}^m = \hat{X}^\ell \quad [\text{induction hypothesis}]
\Rightarrow & \quad \hat{X}^{m+1} = \hat{X}^m = \hat{X}^\ell \quad [\text{if } \varphi(\hat{X}^m) \sqsubseteq \hat{X}^m]
\text{or} & \quad \hat{X}^{m+1} = \hat{X}^m \triangledown \varphi(\hat{X}^m) \quad [\text{otherwise}]
\Rightarrow & \quad \hat{X}^{m+1} = \hat{X}^\ell \triangledown \varphi(\hat{X}^\ell) = \hat{X}^\ell
\Rightarrow & \quad \forall m \geq \ell : \hat{X}^m = \hat{X}^\ell \quad [\text{by cases}]
\end{align*} \]

The generalization to a monotonic \( \varphi \in L \xrightarrow{m} L \) is straightforward.
In summary:
- Any iteration sequence with widening is increasing and stationary after finitely many iteration steps;
- Its limit $F^\uparrow$ is a post-fixpoint of $F$, whence an upper-approximation of the least fixpoint $\text{ifp} \subseteq F^\uparrow$.

\[ \text{ifp} \subseteq F \subseteq F^\uparrow \]

\[ \text{ifp} \text{ exists e.g. if } (P, \sqsubseteq, \sqsupseteq, \\downarrow) \text{ is a qpo.} \]

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 \( \{(x \geq 1000)\} \)
6:

- Forward abstract equations for interval analysis with widening:

\[
X0 = \{ x \mapsto [\text{min\_int}, \text{max\_int}] \}
X2 = \{ x \mapsto [1, 1] \} \sqcup X4
X3 = X2 \cap \{ x \mapsto [\text{min\_int}, 999] \}
X4 = \{ X3 = \bot \land \bot; \text{let } [a, b] = X3 \text{ in} \}
\quad [\text{min}(a + 1, \text{max\_int}), \text{min}(b + 1, \text{max\_int})] \}
X6 = X2 \cap \{ x \mapsto [1000, \text{max\_int}] \}
\]

- Iteration with widening from $X0 = X2 = X3 = X4 = X6 = \{ x \mapsto \bot \}$:

\[
X2 = \{ x = [1, 1] \}
\text{widening at 2 by } \{ x = [1, 1] \}
X2 = \{ x = [1, 1] \}
\text{widening at 3 by } \{ x = [1, 1] \}
X3 = \{ x = [1, 1] \}
\text{widening at 4 by } \{ x = [2, 2] \}
X4 = \{ x = [2, 2] \}
\text{widening at 2 by } \{ x = [1, 2] \}
X2 = \{ x = [1, +\infty] \}
\text{widening at 3 by } \{ x = [1, 999] \}
X3 = \{ x = [1, +\infty] \}
\text{widening at 4 by } \{ x = [2, +\infty] \}
X4 = \{ x = [2, +\infty] \}
\text{widening at 6 by } \{ x = [1000, +\infty] \}
X6 = \{ x = [1000, +\infty] \}
s\text{stable}
\]
Example of narrowing for interval analysis

- The narrowing improves infinite bounds only:

\[
\bot \Delta X = \bot
\]

\[
[l_0, u_0] \Delta [l_1, u_1] = \begin{cases} (l_0 = -\infty \ ? l_1 \neq l_0), & \text{if } l_1 < l_0 \\ (u_0 = +\infty \ ? u_1 \neq u_0) & \text{if } u_1 < u_0 \end{cases}
\]

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

- Equation (cont’d):

\[
X = \left( \left[ 1, 1 \right] \sqcup \left( X \oplus \left[ 2, 2 \right] \right) \right) \cap \left( -\infty, 99 \right]
\]

where \( \emptyset \oplus I = I \oplus \emptyset = \emptyset \) and \( \left[ a, b \right] \oplus \left[ c, d \right] = \left[ a + c, b + d \right] \) with

\[-\infty + x = x + -\infty = -\infty \text{ and } +\infty + x = x + +\infty = +\infty.\]

- Other examples of narrowings:
Descending abstract iteration sequence with narrowing starting from $\bar{X}^3 = [1, +\infty]$:

$\bar{X}^0 = \bar{X}^3$

$\quad = [1, +\infty]$

$\bar{X}^1 = \bar{X}^0 \Delta \left( (\bar{X}^0 \oplus [2, 2]) \cap [-\infty, 99] \right)$

$\quad = [1, +\infty] \Delta [1, 99]$

$\bar{X}^2 = \bar{X}^1 \Delta \left( (\bar{X}^1 \oplus [2, 2]) \cap [-\infty, 99] \right)$

$\quad = [1, 99] \Delta [1, 99]$

$\quad = [1, 99]$

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

---

**Definition of the narrowing**

- Since we have got a post-fixpoint $F^\nabla$ of $F \in P \mapsto P$, its iterates $F^n(F^\nabla)$ are all upper approximations of $\text{lfp } F$.

- 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:
  - $\forall x, y \in P : y \sqsubseteq x \implies y \sqsubseteq x \Delta y \sqsubseteq x$
  - For all decreasing chains $x^0 \sqsubseteq x^1 \sqsubseteq \ldots$ the decreasing chain $y^0 \overset{\text{def}}{=} x^0, \ldots, y^{n+1} \overset{\text{def}}{=} y^n \Delta x^{n+1}, \ldots$ 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 $\Delta$ for $F$ from the post-fixpoint $P^a$ is $Y^n, n \in \mathbb{N}$:
  - $Y^0 = P$
  - $Y^{n+1} = Y^n$ if $F(Y^n) = Y^n$
  - $Y^{n+1} = Y^n \Delta F(Y^n)$ if $F(Y^n) \neq Y^n$

---

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

If

- $L(\sqsubseteq)$ is a poset,
- $\varphi \in L \mapsto L$,
- $\Delta \in L \times L \mapsto L$ is a narrowing operator and
- $\varphi(x) = x \sqsubseteq y, \varphi(y) \sqsubseteq y$,

then the decreasing chain:

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

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

---
Proof. \( x \sqsubseteq \hat{X}^0 \) \quad \text{[hypothesis and transitivity]}
- \( x \sqsubseteq \hat{X}^k \) \quad \text{[induction hypothesis]}
\( \implies x = \varphi(x) \sqsubseteq \varphi(\hat{X}^k) \) \quad \text{[monotony]}
\( \implies x \sqsubseteq \hat{X}^{k+1} = \hat{X}^k \triangle \varphi(\hat{X}^k) \sqsubseteq \hat{X}^k \) \quad \text{[(e) and (f)]}
\( \implies \forall k \in \mathbb{N} : x \sqsubseteq \hat{X}^k \) and \quad \text{[by induction]}

the chain \( \hat{X}^k, k \in \mathbb{N} \) is decreasing for \( \sqsubseteq \)
\( \implies \) the chain \( \varphi(\hat{X}^k), k \in \mathbb{N} \) is decreasing for \( \sqsubseteq \) \quad \text{[monotony]}
\( \implies \hat{X}^k, k \in \mathbb{N} \) has a limit \( \hat{X}^\ell \) \quad \text{[(g)]}
\( \implies x \sqsubseteq \hat{X}^\ell \sqsubseteq \hat{X}^0 = y. \)

\( \square \)

In summary:
- Any iteration sequence with narrowing starting from a post-fixpoint \( P \) of \( F \) is decreasing and stationary after finitely many iteration steps;
- if \( \text{ifp} \subseteq F \) does exist and \( \text{ifp} \subseteq F \sqsubseteq P \) then its limit \( F_\Delta \) is a fixpoint of \( F \), whence an upper-approximation of the least fixpoint \( \text{ifp} \subseteq F \):

\[
\text{ifp} \subseteq F \sqsubseteq F_\Delta \sqsubseteq P
\]

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

---

### 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:

\[
\begin{cases}
X0 = \{x \mapsto [\text{min\_int}, \text{max\_int}]\} \\
X2 = \{x \mapsto [1, 1]\} \sqcup X4 \\
X3 = X2 \triangle \{x \mapsto [\text{min\_int}, 999]\} \\
X4 = (X3 = \bot ? \bot : \text{let } [a, b] = X3 \text{ in } [\text{min}(a + 1, \text{max\_int}), \text{min}(b + 1, \text{max\_int})]) \\
X6 = X2 \triangle \{x \mapsto [1000, \text{max\_int}]\}
\end{cases}
\]
iterations with narrowing from:

- X0 = \{ x:0 \} narrowing at 4 by \{ x:[2,1000] \}
- X2 = \{ x:[1,\infty) \} narrowing at 6 by \{ x:[1000,\infty) \}
- X3 = \{ x:[1,\infty) \} narrowing at 0 by \{ x:[-\infty,\infty) \}
- X4 = \{ x:[2,\infty) \} narrowing at 0 by \{ x:[-\infty,\infty) \}
- X6 = \{ x:[1000,\infty) \}
- 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:

\[ X^2 = X^1 P(X^1) \]
\[ X^1 = X^0 \Delta P(X^0) \]
\[ X^0 = \top \]

---

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

```plaintext
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 \( (L \xleftarrow{\gamma}{\alpha}, L, \nabla \text{ and } \Delta) \).
- The approximation \( \alpha \) may be precise while \( \nabla \) may be very rough.
- Usefulness of the approximation is shown by experience (precision/cost can be tuned with \( \nabla \)).
- The approximation is applied at each iteration step for \( \mathcal{F} \).
- The approximation depends upon the iterates.

Schema of a static program analyzer with widening/narrowing

\[
\langle I, \mathcal{F} \rangle := \text{syntactic _analysis}(\text{Program});
\]

\[
\text{while } X \neq Y \text{ do }
\]

\[
Y := X;
X := \mathcal{F}(X);
\]

\[
\text{if } X \sqsubseteq Y \text{ then } C := \text{true} \text{ else } C := \text{false}; \]

\[
X := Y \nabla X \text{ fi}
\]

until \( C \);

\[
\text{In practice, chaotic or asynchronous iterations (with memory).}
\]
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

<table>
<thead>
<tr>
<th>Widening/narrowing are not dual; Dual widening/narrowing</th>
</tr>
</thead>
<tbody>
<tr>
<td>- The iteration with <strong>widening</strong> starts from below the least fixpoint and stabilizes above to a postfixpoint;</td>
</tr>
<tr>
<td>- The iteration with <strong>narrowing</strong> starts from above the least fixpoint and stabilizes above;</td>
</tr>
<tr>
<td>- The iteration with <strong>dual widening</strong> starts from above the greatest fixpoint and stabilizes below to a prefixpoint;</td>
</tr>
<tr>
<td>- The iteration with <strong>dual narrowing</strong> starts from below the greatest fixpoint and stabilizes below;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Iteration starts from</th>
<th>Iteration stabilizes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Widening ▼</td>
<td>below</td>
</tr>
<tr>
<td>Narrowing △</td>
<td>above</td>
</tr>
<tr>
<td>Dual widening ▼</td>
<td>above</td>
</tr>
<tr>
<td>Dual narrowing △</td>
<td>below</td>
</tr>
</tbody>
</table>

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

```plaintext
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:
  - \( X = [1, 1] \cup (X \cap [\infty, 100]) \cap [2, 2] \)
  - \( Y = X \cap [101, +\infty] \)
  - Upwards iteration from the infimum without widening
    \[
    \begin{align*}
    X^0 &= \bot \\
    X^1 &= [1, 1] \\
    X^2 &= [1, 3] \\
    X^3 &= [1, 5] \\
    \ldots &= \ldots
    \end{align*}
    \]
    \[
    \begin{align*}
    Y^0 &= \bot \\
    Y^1 &= [1, 1] \\
    Y^2 &= [1, 3] \\
    Y^3 &= [1, 5] \\
    \ldots &= \ldots
    \end{align*}
    \]
    \[
    \begin{align*}
    X^{50} &= [1, 99] \\
    X^{51} &= [1, 101] \\
    X^{52} &= [1, 101]
    \end{align*}
    \]
  - Downward iteration with narrowing
    \[
    \begin{align*}
    \bar{X}^0 &= [1, +\infty] \\
    \bar{X}^1 &= \bar{X}^0 \Delta [1, 102] = [1, 102] \\
    \bar{X}^2 &= \bar{X}^1 \Delta [1, 102] = [1, 102] \\
    \bar{Y}^0 &= [101, +\infty] \\
    \bar{Y}^1 &= [101, 102] \\
    \bar{Y}^2 &= [101, 102]
    \end{align*}
    \]
    - 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

Convergence could have been very slow (even impossible without the test \( I \leq 100 \) when using bignums)!

- Upwards iteration from the infimum with widening
  \[
  \begin{align*}
  \bar{X}^0 &= \bot \\
  \bar{X}^1 &= \bar{X}^0 \vee [1, 1] = [1, 1] \\
  \bar{X}^2 &= \bar{X}^1 \vee [1, 3] = [1, +\infty] \\
  \bar{X}^3 &= \bar{X}^2 \vee [1, 102] = [1, +\infty] \\
  \bar{Y}^3 &= [101, +\infty]
  \end{align*}
  \]
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:

```plaintext
program P;
var I : integer;
begindt
\{1:\} I := 1;
\{2:\} while \(I \in X\) \(I \leq n\) do begin
\{3:\} I := I + 2;
\{4:\} end;
\{5:\} \{I \in Y\}
end.
```

where \(n\) is a mathematical variable denoting any program constant, \(n \geq 1\). By instantiating \(n\) to all possible naturals \(\geq 1\), 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:

\[
\begin{align*}
X &= [1, 1] \cup ((X \cap [-\infty, n]) \uplus [2, 2]) \\
Y &= X \cap [n + 1, +\infty] \\
\end{align*}
\]

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

\[
\begin{align*}
\hat{X}_0 &= \bot \\
\hat{X}_1 &= \hat{X}_0 \triangledown ([1, 1] \cup ((\hat{X}_0 \cap [-\infty, n]) \uplus [2, 2])) \\
&= \bot \triangledown [1, 1] \cup \bot \\
&= [1, 1] \\
\hat{X}_2 &= \hat{X}_1 \triangledown ([1, 1] \cup ((\hat{X}_1 \cap [-\infty, n]) \uplus [2, 2])) \\
&= [1, 1] \triangledown ([1, 1] \cup ([1, 1] \cap [-\infty, n]) \uplus [2, 2])) \\
&= [1, 1] \triangledown [1, 3] \\
&= [1, +\infty] \\
\hat{X}_3 &= \hat{X}_2 \triangledown ([1, 1] \cup ((\hat{X}_2 \cap [-\infty, n]) \uplus [2, 2])) \\
&= [1, +\infty] \triangledown ([1, 1] \cup ([1, +\infty] \cap [-\infty, n]) \uplus [2, 2])) \\
&= [1, +\infty] \triangledown [1, n + 2] \\
&= [1, +\infty] \\
\end{align*}
\]

- 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 \(n\)):

\[
\begin{align*}
\hat{X}_0 &= [1, +\infty] \\
\hat{X}_1 &= \hat{X}_0 \Delta ([1, 1] \cup ((\hat{X}_0 \cap [-\infty, n]) \uplus [2, 2])) \\
&= [1, +\infty] \Delta ([1, 1] \cup (([1, +\infty] \cap [-\infty, n]) \uplus [2, 2])) \\
&= [1, +\infty] \Delta ([1, 1] \uplus ([1, n] \uplus [2, 2])) \\
&= [1, +\infty] \Delta ([1, n + 2]) \\
&= [1, n + 2] \\
\hat{X}_2 &= \hat{X}_1 \Delta [1, n + 2] \\
&= [1, n + 2] \\
\end{align*}
\]
\[Y^2 = X^2 \cap [n + 1, +\infty] \]
\[= [1, n + 2] \cap [n + 1, +\infty] \]
\[= [n + 1, n + 2] \]

- This proves that for all programs in the family (parameterized by \(n\)), the analysis with widening/narrowing will always discover the interval invariant
\[
\begin{align*}
X &= [1, n + 2] \\
Y &= [n + 1, n + 2]
\end{align*}
\]

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 over-approximate \(\wp^\gamma F\) where \(F\) is a concrete monotonic transformer \(F \in L \xrightarrow{\gamma} L\) on the cpo \(\langle L, \sqsubseteq, \bot, \top, \cup, \sqcap \rangle\). 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 \(\langle \overline{L}, \sqsubseteq \rangle\) such that \(\langle L, \sqsubseteq \rangle \triangleleft \frac{\gamma}{\overline{\gamma}} \langle \overline{L}, \overline{\sqsubseteq} \rangle\), a monotonic abstract transformer \(\overline{F} \sqsubseteq \alpha \circ F \circ \gamma\) and a widening \(\overline{\gamma}\).
- Because \(\alpha\) is surjective\(^10\), \(\langle \overline{L}, \overline{\sqsubseteq}, \top, \bot \rangle\) is indeed a cpo with supremum \(\overline{\top} = \alpha(\top)\).
- The analysis computes iterates \(y^0 = \alpha(\bot), \ldots, y^{n+1} = y^n \lor F(y^n), \ldots, y^\ell\) where the limit \(y^\ell\) is a postfixpoint \(\overline{F}(y^\ell) \sqsubseteq y^\ell\).
- Let us define the abstract domain \(\overline{L} = \{y^0, \ldots, y^n, \ldots, y^\ell, \overline{\top}\}\) with ordering \(\overline{\sqsubseteq}\) which is \(\sqsubseteq\) restricted to \(\overline{L}\).
- Because the iterates are a finite increasing chain and \(\overline{\top}\) is the supremum, \(\langle \overline{L}, \overline{\sqsubseteq}\rangle\) is a finite chain whence a complete lattice.
- Let us define the abstraction
\[
\overline{\alpha}(x) \triangleq \bigwedge\{y \in \overline{L} \mid \alpha(x) \sqsubseteq y\} \quad 11
\]

\(^10\) Otherwise we choose \(\overline{F} = \alpha(L)\).

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

---

and the concretization
\[
\overline{\alpha}(x) \sqsubseteq y \quad \iff \quad \overline{\alpha}(x) \sqsubseteq y \quad \iff \quad \bigwedge\{y \in \overline{L} \mid \alpha(x) \sqsubseteq y\} \quad \iff \quad \{\text{def. } \overline{\sqsubseteq}\}
\]

---

We have a Galois connection \(\langle L, \sqsubseteq \rangle \triangleleft \frac{\gamma}{\overline{\gamma}} \langle \overline{L}, \overline{\sqsubseteq} \rangle\).

**Proof.**

\[
\begin{align*}
\overline{\alpha}(x) & \sqsubseteq y \quad \iff \quad \alpha(x) \sqsubseteq y \quad \text{(def. } \overline{\sqsubseteq}) \\
\bigwedge\{y \in \overline{L} \mid \alpha(x) \sqsubseteq y\} & \sqsubseteq y \quad \text{(def. } \overline{\sqsubseteq})
\end{align*}
\]

\[\implies \quad \langle \overline{\alpha}(x) \sqsubseteq y \rangle \implies \exists \text{We have } \alpha(x) \sqsubseteq \overline{\top} \in \overline{L} \text{ so, since } \overline{\top} \text{ is a finite strictly decreasing chain, there is a smallest } y^n \in \overline{L} : \alpha(x) \sqsubseteq y^n, \text{ whence } \alpha(x) \sqsubseteq y \text{ implies } y^n \sqsubseteq y \text{ so } y^n = \bigwedge\{y \in \overline{L} \mid \alpha(x) \sqsubseteq y\} \sqsubseteq y. \text{ It follows that:}
\]

\[\alpha(x) \sqsubseteq y^n \sqsubseteq y \implies x \sqsubseteq \gamma(y) \quad \text{(def. } \overline{\sqsubseteq})
\]

---
\[ \Rightarrow x \subseteq \mathcal{T}(y) \]

Conversely
\[ x \subseteq \gamma(y) \]

\[ \Rightarrow x \subseteq \gamma(y) \]

\[ \Rightarrow \alpha(x) \subseteq y \]

\[ \Rightarrow \exists \gamma' \in \mathcal{L} \mid x \subseteq \gamma' \subseteq y \]

\[ \Rightarrow \varphi(x) \subseteq y \]

\[ \Rightarrow \varphi(x) \subseteq y \]

\[ \Rightarrow \mathcal{L} \]

Let us define:
\[ \mathcal{F} \in \mathcal{L} \rightarrow \mathcal{L} \]
\[ \mathcal{F} \equiv \lambda y. (y = \top \rightarrow \top \mid y = y' \uplus \mathcal{F}(y') ; y \triangledown \mathcal{F}(y)) \]

- We have \( \mathcal{F} \supseteq \mathcal{F} \supseteq \mathcal{F} \supseteq \mathcal{F} \).

Proof. We proceed pointwise on \( \mathcal{F} \subseteq \mathcal{L} \). We already know that \( \mathcal{F} \supseteq \mathcal{F} \supseteq \mathcal{F} \).

- This is obvious for \( \top \) since \( \mathcal{F}(\top) = \top \uplus \mathcal{F}(\top) \) since \( \top \) is the supremum of \( \mathcal{L} \).

- This holds for \( y' \) since \( \mathcal{F}(y') = \mathcal{F}(y') \supseteq \mathcal{F}(y') \) by reflexivity.

- For the other elements \( y \in \mathcal{L} \setminus \{ \top, y' \} \), we have \( \mathcal{F}(y) = y \uplus \mathcal{F}(y) \supseteq \mathcal{F}(y) \) by def. \( \triangledown \) which is an upper bound.

- Observe that the iterates of \( \mathcal{F} \) from \( \alpha(\bot) \) are exactly \( y = \alpha(\bot), \ldots, y^n = \mathcal{F}(y^{n-1}) = y^{n-1} \uplus \mathcal{F}(y^{n-1}), \ldots, y \) since \( \mathcal{F}(y') = \mathcal{F}(y') \subseteq y' \), which is the convergence condition.

- These iterates are therefore convergent (despite the fact that the widening \( \triangledown \) hence \( \mathcal{F} \) is not monotone since \( \mathcal{F} \) is extensive, but for \( y' \), which is the convergence point).

In conclusion, the analysis of the given program can be done in the finite (complete) lattice \( \mathcal{L} \) by computing the limit \( (y') \) of the finitely convergent iterates of \( \mathcal{F} \supseteq \mathcal{F} \supseteq \mathcal{F} \).

- 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
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\}):

\[
\begin{aligned}
\text{program } P; \\
\quad \text{var } I : \text{integer}; \\
\quad \text{begin} \\
\quad \quad \{1:\} \\
\quad \quad \quad I := 1; \\
\quad \quad \{2:\} \\
\quad \quad \quad \text{while } \{ I \in X \} I \leq 100 \text{ do begin} \\
\quad \quad \quad \quad \{3:\} \\
\quad \quad \quad \quad \quad I := I + 2; \\
\quad \quad \quad \quad \{4:\} \\
\quad \quad \quad \quad \quad \text{end;} \\
\quad \quad \quad \{5:\} \{ I \in Y \} \text{ end.}
\end{aligned}
\]

- So when considering all programs \(P(n), \text{ for all } n \geq 0\), 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 \([0, 1] \subseteq [0, 2] \subseteq \ldots \subseteq [0, n] \subseteq \ldots\).
- Analyzing iteratively a program like \texttt{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:

\[
\begin{array}{c}
\begin{array}{c}
\text{for which an equally precise analysis has to guess all necessary constants, including those not appearing explicitly in the program text.}
\end{array}
\end{array}
\]
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(X + 33)))); { F ∈ [91, 99] }
  end;
begin
  readin(X);  Y := F(X);
  if Y ∈ [91, maxint - 10] then
end.
```

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

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.

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;

Another incorrect common belief 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_Tot }
  x := 1;
2: { x : SC_POS }
  while (x ≠ 0) do
    if (0 < x) then
      5: { x : SC_POS }
      x := 0;
    6: { x : SC_ZERO }
  else
    7: SC_BOT
      skip;
  fi;
10: { x : SC_ZERO }
11: { x : SC_ZERO }
```

- The interval analysis with abstract domain

\[
\mathcal{L} \overset{\text{def}}{=} \{ [l, u] \mid l \in \mathbb{Z} \cup \{-\infty\} \land u \in \mathbb{Z} \cup \{+\infty\} \land l \leq u \} \cup \{\bot\}
\]

and widening extrapolating unstable bounds to infinity:

\[
\bot \triangledown X = X \\
X \triangledown \bot = X \\
[l_0, u_0] \triangledown [l_1, u_1] = ((l_1 < l_0 \implies -\infty \land l_0), \\
(\l_1 > u_0 \implies +\infty \lor u_0))
\]

is less precise!

- Program analysis:

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

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

<table>
<thead>
<tr>
<th>previous iterate $X$</th>
<th>$F(X) = X - 1$</th>
<th>next iterate</th>
</tr>
</thead>
<tbody>
<tr>
<td>signs</td>
<td>SC_POS</td>
<td>SC_POS</td>
</tr>
<tr>
<td>intervals</td>
<td>$[1, +\infty]$</td>
<td>$[0, +\infty]$</td>
</tr>
</tbody>
</table>

The 0 threshold is missed by the widening but caught by the sign analysis.
The interval analysis with improved widening is as precise or more precise than the sign analysis:

0: { x: 0 }  
1: x := 1;  
2: { x:[0,1] }  
while (x < 0) do  
  4: { x:[1,1] }  
    if (0 < x) then  
      5: { x:[1,1] }  
        x := 0  
      6: { x:[0,0] }  
    else if (x == 0) then  
      7: x := 0  
      skip  
  8: { x:[0,0] }  
11: { x:[0,0] }

A correct statement about the relative precision of widenings

Theorem. Assume that \((L, \sqsubseteq), (\overline{L}, \sqsubseteq)\) and \((\overline{L}, \sqsubseteq)\) are posets such that

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

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

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

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

(d.1) \([\gamma(X) \sqsubseteq \overline{\gamma}(X') \land \gamma(Y) \sqsubseteq \overline{\gamma}(Y')] \implies [\gamma(X \overline{\nabla} Y) \sqsubseteq \overline{\gamma}(X' \overline{\nabla} Y')]\)

Proof. We let \(\overline{X}^k, k \geq 0\) be defined as follows:

\[
\begin{cases}
\overline{X}^0 = \overline{x} \\
\overline{X}^{n+1} = \overline{X}^n \overline{\nabla} F(X') & \text{if } F(X') \sqsubseteq X^n \\
\overline{X}^{n+1} = \overline{X}^n \overline{\nabla} F(X') & \text{otherwise}
\end{cases}
\]

and similarly \(\overline{X}^k, k \geq 0\) is defined as follows:

\[
\begin{cases}
\overline{X}^0 = \overline{x} \\
\overline{X}^{n+1} = \overline{X}^n \overline{\nabla} \overline{F}(X') & \text{if } \overline{F}(X') \sqsubseteq \overline{X}^n \\
\overline{X}^{n+1} = \overline{X}^n \overline{\nabla} \overline{F}(X') & \text{otherwise}
\end{cases}
\]

- We have \(\overline{\gamma}(\overline{X}^k) = \overline{\gamma}(\overline{x}) \sqsubseteq \overline{\gamma}(\overline{x}) = \overline{\gamma}(\overline{X}^0)\) by (c).
- Assume by induction hypothesis that \(\overline{\gamma}(\overline{X}^n) \sqsubseteq \overline{\gamma}(\overline{X}^n)\)
  - If both iterates have converged then \(\overline{\gamma}(\overline{X}^{n+1}) = \overline{\gamma}(\overline{X}^n) \sqsubseteq \overline{\gamma}(\overline{X}^n) = \overline{\gamma}(\overline{X}^{n+1})\)
  - If \((X^k, k \geq 0)\) has converged at rank \(n\), but not \(\overline{X}^k, k \geq 0\). We have
\( \tau(X^n) \subseteq \overline{\tau(X^n)} \)  \hspace{1cm} \{ \text{ind. hyp.} \}

\[ \Rightarrow \tau(X^n) \subseteq \overline{\tau(X^n \setminus F(X^n))} \]  \hspace{1cm} \{ \text{by (d.1)} \}

\[ \Rightarrow \tau(X^{n+1}) \subseteq \overline{\tau(X^{n+1})} \]  \hspace{1cm} \{ \text{by def. iterates} \}

- If \( \langle X^k, k \geq 0 \rangle \) has converged at rank \( \ell \leq n \) but not \( \langle X^k, k \geq 0 \rangle \), we have

\[ \tau(X^n) \subseteq \overline{\tau(X^n)} \]  \hspace{1cm} \{ \text{ind. hyp.} \}

\[ \Rightarrow \tau(F(X^n)) \subseteq \overline{\tau(F(X^n))} \]  \hspace{1cm} \{ \text{by (e)} \}

\[ \Rightarrow \tau(X^n \setminus F(X^n)) \subseteq \overline{\tau(X^n \setminus F(X^n))} \]  \hspace{1cm} \{ \text{by (d.1)} \}

\[ \Rightarrow \tau(X^n) \subseteq \overline{\tau(X^n \setminus F(X^n))} \]  \hspace{1cm} \{ \text{by def. iterates} \}

- If \( \langle X^k, k \geq 0 \rangle \) has converged at rank \( \ell \leq n \) but not \( \langle X^k, k \geq 0 \rangle \), we have

\[ \tau(X^n) \subseteq \overline{\tau(X^n)} \]  \hspace{1cm} \{ \text{ind. hyp.} \}

\[ \Rightarrow \tau(F(X^n)) \subseteq \overline{\tau(F(X^n))} \]  \hspace{1cm} \{ \text{by (e)} \}

\[ \Rightarrow \tau(X^n \setminus F(X^n)) \subseteq \overline{\tau(X^n \setminus F(X^n))} \]  \hspace{1cm} \{ \text{by def. iterates} \}

- By recurrence, \( \forall n \in \mathbb{N} : \tau(X^n) \subseteq \overline{\tau(X^n)} \)

\[ \Rightarrow \tau(X^n \setminus F(X^n)) \subseteq \overline{\tau(X^n \setminus F(X^n))} \]  \hspace{1cm} \{ \text{by (d.1)} \}

\[ \Rightarrow \tau(X^{n+1}) \subseteq \overline{\tau(X^{n+1})} \]  \hspace{1cm} \{ \text{def. iterates} \}

- If none of the \( \langle X^k, k \geq 0 \rangle \) and \( \langle X^k, k \geq 0 \rangle \) have converged at rank \( n \), then:

\[ \tau(X^n) \subseteq \overline{\tau(X^n)} \]  \hspace{1cm} \{ \text{ind. hyp.} \}

\[ \Rightarrow \tau(F(X^n)) \subseteq \overline{\tau(F(X^n))} \]  \hspace{1cm} \{ \text{by (e)} \}

\[ \Rightarrow \tau(X^n \setminus F(X^n)) \subseteq \overline{\tau(X^n \setminus F(X^n))} \]  \hspace{1cm} \{ \text{by (e)} \}

\[ \Rightarrow \tau(X^{n+1}) \subseteq \overline{\tau(X^{n+1})} \]  \hspace{1cm} \{ \text{def. iterates} \}

- 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 \( \langle \overline{I}, \overline{\sqcap}, \overline{I}, \overline{\sqcup} \rangle \), \( \overline{F} \) is overapproximated by the limit of an upper iteration of \( \overline{F} \) from \( \overline{I} \) with widening \( \overline{\sqcap} \)

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

\[ X \sqsubseteq X \sqcap Y \]

\[ Y \sqsubseteq X \sqcap Y \]
- In practice, we only want to compare iterations of $F \in L \xrightarrow{m} L$ on the cpo $\langle L, \sqsubseteq, \perp, \sqcup \rangle$ 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

\[
\begin{align*}
\gamma(X) &\subseteq \gamma(X \overline{\nabla} Y) \\
\gamma(Y) &\subseteq \gamma(X \overline{\nabla} Y)
\end{align*}
\]

(together with $F \circ \gamma \subseteq \gamma \circ \overline{F}$)

- 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 hypothe-
ses which are checked by the convergence condition $F(X) \sqsubseteq X$ so no condition on $\nabla$ relative to soundness is indeed needed!

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

**Theorem.** \(^{12}\) Let $F \in L \xrightarrow{m} L$ be a monotone operator on the cpo $\langle L, \sqsubseteq, \perp \rangle$. Assume $\perp \in L$ satisfies $\perp \sqsubseteq F(\perp)$. Let $\langle L, \sqsubseteq \rangle$ be a poset and $\overline{F} \in \overline{L} \mapsto \overline{L}$ such that $F \circ \gamma \subseteq \gamma \circ \overline{F}$ where $\gamma \in \overline{L} \xrightarrow{m} L$. Assume that the widening $\overline{\nabla} \in \overline{L} \times \overline{L} \mapsto \overline{L}$ satisfies:

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

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

- $X^0 = \perp$
- $X^{n+1} = X^n$ if $\overline{F}(X^n) \sqsubseteq X^n$ (a)
- $X^{n+1} = X^n \overline{\nabla} \overline{F}(X^n)$ otherwise (b)

is ultimately stationary at rank $\bar{l} \in \mathbb{N}$. Then $\gamma(\overline{F}(X^{\bar{l}})) \subseteq \gamma(X^{\bar{l}})$ and $\forall n \sqsubseteq F \sqsubseteq \gamma(X^{\bar{l}})$.

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

\(^{12}\) Observe that the absence of lub existence hypotheses in $L$, that $F$ is not assumed to be monotone or extensive and that the widening is only assumed to ensure convergence not to overapproximate lubs.
\(X^t\) exists by the convergence enforcement hypothesis on the widening. Moreover \(t \geq 1\) since at least one iteration is necessary to check for stability. In case \(X^t\) satisfies (b), we have

\[ F(X^t) \subseteq X^t \]
\[ \implies \gamma(F(X^t)) \subseteq \gamma(X^t) \]
\[ \implies F \cdot \gamma(X^t) \subseteq \gamma(X^t) \]
\[ \implies \Phi^* F \subseteq \gamma(X^t) \]

by transfinite induction on the iterates of \(F\) from \(\bot\) as follows:

- \(X^0 = \bot \subseteq \gamma(\bot) = X^0 \subseteq \gamma(X^t)\) by hypothesis and \(\gamma(\bot)\) is an increasing chain
- If \(X^\lambda \subseteq \gamma(X^t)\) by induction hypothesis then by monotonicity of \(F\), we have \(X^{\lambda+1} = F(X^\lambda) \subseteq \gamma(X^t)\) by the convergence condition \(F(X^\lambda) \subseteq X^\lambda\) and \(\gamma\) monotone.
- If \(\lambda\) is a limit ordinal and \(X^\delta \subseteq \gamma(X^t)\) for all \(\delta < \lambda\) by induction hypothesis, then \(X^\lambda = \bigsqcup_{\delta < \lambda} X^\delta \subseteq \gamma(X^t)\) by def. of limits which exist for chains in cpos.
- There exists \(\varepsilon\) such that \(\Phi^* F = X^\varepsilon \subseteq \gamma(X^t)\).

Otherwise \(X^t\) satisfies (c) and we have

\[ X^t = X^t \uparrow F(X^t) \]
\[ \implies \gamma(F(X^t)) \subseteq \gamma(X^t) \]
\[ \implies 165 F \subseteq \gamma(X^t) \]

(by (c) since \(X^{t+1} = X^t\))

\[ \gamma(\forall v \in L: \gamma(v) \subseteq \gamma(x \downarrow y)) \]
\[ \text{(as shown above)} \]

\[ \square \]

---

Why widenings cannot be monotone

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

\[ X \uparrow Y \subseteq Y \uparrow Y \]

- We have seen that it is reasonable to assume that \((Y \subseteq X) \implies (X \uparrow Y = Y)\) (since e.g. if \(Y = F(X) \subseteq X\) then we have converged so their should be no other loss of information)

\[ \square \]

---

In particular for \(X = Y\), we have

\[ Y \uparrow Y = Y \]

- It follows that

\[ X \uparrow Y \subseteq Y \]

which prevents extrapolations!
Example of non-monotone widening

- the classical widening on intervals is:

$$\bot \triangledown X = X \triangledown \bot = X$$

$$[\ell_0, u_0] \triangledown [\ell_1, u_1] = \left(\ell_1 < \ell_0 \iff -\infty < \ell_0\right),$$

$$\left( u_1 > u_0 \iff +\infty > u_0\right)$$

- Not monotone in its first argument: $[0, 1] \subseteq [0, 2]$ but $[0, 1] \triangledown [0, 2] = [0, +\infty] \not\subseteq [0, 2] = [0, 2] \triangledown [0, 2]$

- Monotone in its second parameter: $(I' \subseteq I'') \implies (I \triangledown I' \subseteq I \triangledown I'')$

Proof. - If $I = \bot$: $(I \triangledown I' = \bot \triangledown I' = I' \subseteq I'' = \bot \triangledown I'' = I \triangledown I'')$

- Else $I = [a, b] \neq \bot$. Then:

  - If $I' = \bot$ then $I \triangledown I' = I \triangledown \bot = \bot \subseteq I \triangledown I''$

  - Else $I' = [a', b'] \neq \bot$ so $I' \subseteq I''$ implies $I'' = [a'', b''] \neq \bot$ with $a'' \leq a'$ and $b' \leq b''$.

- For the lower bound, we have:

  - If $a' < a$ so $a'' < a'$ hence we have $I \triangledown I' = [a, \_] \triangledown [a', \_] = [\_a, \_] \subseteq I \triangledown I'' = [a, \_] \triangledown [a'', \_] = [\_a, \_a]$.

  - Else, $a'' \geq a$, hence $I \triangledown I' = [a, \_] \triangledown [a', \_] = [a, \_] \subseteq I \triangledown I'' = [a, \_] \triangledown [a'', \_] = (a'' \geq a \iff [a, \_] \subseteq [\_a, \_])$

  - Idem, for the upper bound. \hfill \Box

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:

```c
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: $X = 0 \implies X \in [0, 2]$, locally imprecise

  - $1 \rightarrow X \in [0, 2]$ because of local imprecision

  - $2 \rightarrow X \in \bot \triangledown [0, 2] = [0, 2]$

  - $3 \rightarrow X \in [0, 0]$

  - $4 \rightarrow X \in [1, 1]$

  - $5 \rightarrow X \in [1, 2]$

  - $6 \rightarrow X \in [2, 2]$

  - $7 \rightarrow X \in [1, 1] \cup [2, 2] = [1, 2]$

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

- Remedies:
  - Use a narrowing, if possible
    - In the example, \( [0, +\infty] \nabla [0, 2] = [0, 2] \) 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 \( F \in L \xrightarrow{m} L \) be a monotone operator on the cpo \( \langle L, \sqsubseteq, \sqcap \rangle \). Let \( \langle \underline{L}, \sqsubseteq \rangle \) be a poset. Let \( \gamma \in \underline{L} \xrightarrow{m} L \) be a monotone concretization function. Let \( \mathcal{F} \in L \rightarrow \underline{L} \) such that \( F \circ \gamma \sqsubseteq \gamma \circ \mathcal{F} \). Let \( A \in \underline{L} \) be such that \( \forall^\gamma_{\downarrow} F \sqsubseteq \gamma(A) \). Assume that the narrowing \( \Delta \in \underline{L} \times \underline{L} \rightarrow \underline{L} \) satisfies:

\[- \forall x, y \in \underline{L} : \gamma(y) \sqsubseteq \gamma(x \Delta y).\]

(which can be restricted to the case \( y \sqsubseteq x \) and even \( y = \mathcal{F}(x) \))

Assume that the narrowing iteration sequence for \( \mathcal{F} \) from \( A \) defined as

\[- \mathcal{F}^0 = \mathcal{F}(A) \]

\[- \mathcal{F}^{n+1} = \mathcal{F}^n \quad \text{if} \quad \mathcal{F}^n \sqsubseteq \gamma(\mathcal{F}^n) \]

\[- \mathcal{F}^{n+1} = \mathcal{F}^n \nabla \mathcal{F}(\mathcal{F}^n) \quad \text{otherwise} \]

is ultimately stationary at rank \( \ell \in \mathbb{N} \). Then \( \forall^\gamma_{\downarrow} F \sqsubseteq \gamma(\mathcal{F}^\ell) \). \( ^{\blacksquare} \)

**Proof.** We prove that \( \forall n \in \mathbb{N} : \forall^\gamma_{\downarrow} F \sqsubseteq \gamma(\mathcal{F}^n) \) so that the narrowing iteration can be stopped at any iteration rank \( ^{15} \).

- For the basis, we have \( \forall^\gamma_{\downarrow} F \sqsubseteq \gamma(A) \) by hypothesis (which results from the widening phase) and, so by fixpoint property, monotony of \( F \), \( F \circ \gamma \sqsubseteq \gamma \circ \mathcal{F} \) and \( \mathcal{F}^0 \), we have \( \forall^\gamma_{\downarrow} F = F(\forall^\gamma_{\downarrow} F) \sqsubseteq \gamma(\mathcal{F}(\mathcal{F}^0)) \sqsubseteq \gamma(A) = \gamma(\mathcal{F}^0) \)

---

\( ^{13} \) Observe that \( F \) is not assumed to be monotone. If it is extensive, a narrowing is of no interest.

\( ^{14} \) \( \mathcal{F}(A) \) 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 \( y \Delta y = y \) and the iteration restricted to one step.
- Assume that \( n \leq \ell \) and, by induction hypothesis, \( \mathcal{H}_n \subseteq \gamma(\mathcal{Y}^n) \). There are two cases according to the definition of the iterates:
  - If \( \mathcal{Y}^n \subseteq \mathcal{F}(\mathcal{Y}^n) \) then \( \mathcal{Y}^{n+1} = \mathcal{Y}^n \) so that by induction hypothesis \( \mathcal{H}_{n+1} \subseteq \gamma(\mathcal{Y}^{n+1}) \) and indeed, \( n = \ell \).
  - Otherwise, \( n < \ell \) and \( \mathcal{Y}^n \Delta \mathcal{F}(\mathcal{Y}^n) \). In that case, we have

\[
\mathcal{H}_{n+1} \subseteq \mathcal{F} = \mathcal{F}(\mathcal{H}_n) \subseteq \mathcal{F}(\gamma(\mathcal{Y}^n)) \subseteq \gamma(\mathcal{Y}^n) \subseteq \gamma(\mathcal{Y}^n \Delta \mathcal{F}(\mathcal{Y}^n)) \subseteq \gamma(\mathcal{Y}^{n+1})
\]

We conclude by recurrence on \( n \), noting that the iterates are stationary beyond \( \ell \). \( \square \)

---

**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” since the last widening/narrowing.

- Example:

- With widening/narrowing at cut-points:

- With history-based widening/narrowing:
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 \( \langle L, \sqsubseteq, \sqcap, \sqcup, \top, \sqcap \rangle \) be a complete lattice. Let \( T_1 \sqsubseteq T_2 \sqsubseteq \ldots \sqsubseteq T_n = \top \) be finitely many elements of \( L \). Define \( T = \{ T_1, \ldots, T_n \} \). The widening with thresholds \( T \) is

\[
X \nabla_T Y = (Y \sqsubseteq X ? X \uplus T) \\
where \quad X \sqcup Y \sqsubseteq T_i \\
\text{and} \quad \forall T_j \in T : X \sqcup Y \sqsubseteq T_j \implies T_i \sqsubseteq T_j
\]

**Theorem.** \( X \nabla_T Y \) is a widening. \( \square \)

Widenings for pairs/tuples

- If
  - \( \nabla_1 \) is a widening for \( \langle L_1, \sqsubseteq_1 \rangle \), and
  - \( \nabla_2 \) is a widening for \( \langle L_2, \sqsubseteq_2 \rangle \),
  then
    \[
    \langle x, y \rangle \nabla \langle x', y' \rangle := \langle x \nabla_1 x', y \nabla_2 y' \rangle
    \]
    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:

\[ f \in L \xrightarrow{\text{m}} L, \nabla \] is a widening on a poset \( \langle L, \sqsubseteq \rangle \)

then

\[ \text{lfp} \subseteq f \sqsubseteq \text{lfp} \subseteq \lambda x. x \nabla f(x) \]

(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:

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

for the argument \([0, 0]\).

So we approximate by:

\[ f = f \nabla F(f) \]

for argument \([0, 0]\), that is:

\[
\begin{align*}
  f([0, 0]) &= f([0, 0]) \nabla_1 F([0, 0]) \\
  &= f([0, 0]) \nabla_1 F([1, 1] \sqcup (f([0, 0]) + [2, 2]))
\end{align*}
\]

Second-order functional widening – I – finite domains

If

- \( F \in (S \mapsto L) \xrightarrow{\text{m}} (S \mapsto L) \), pointwise ordering;
- \( \nabla \) is a widening for \( \langle L, \sqsubseteq \rangle \);
- \( S \) is a finite set

then

\[ \text{lfp} \sqsubseteq F \sqsubseteq \text{lfp} \sqsubseteq \lambda x. f(x) \nabla F(f)(x) \]
Note:
This can be seen as a system of equations:
\[
\begin{cases}
X_i = F_i(X_1, \ldots, X_n) \\
i = 1, \ldots, n
\end{cases}
\]
where \( S = \{1, \ldots, n\} \) and \( X_i \) is \( f(i) \).
This is solved as:
\[
\begin{cases}
X_i = X \nabla F_i(X_1, \ldots, X_n) \\
i = 1, \ldots, n
\end{cases}
\]
with the usual remark that \( \nabla \) is needed only once around cycles of the dependence graph.

Possible divergence for infinite domains
- Note: the previous widening strategy fails for
\[
f(x) = [1, 1] \cup (f(x + [1, 1]) + [2, 2])
\]
since \( f([0, 0]) \) needs \( f([1, 1]) \) which needs \( f([2, 2]) \), etc.

Second-order functional widening – II – infinite domains
- If
  - \( \nabla_1 \) is a widening on \( \langle L_1, \sqsubseteq_1 \rangle \),
  - \( \nabla_2 \) is a widening on \( \langle L_2, \sqsubseteq_2 \rangle \)
  - \( F \in (L_1 \mapsto L_2) \mapsto (L_1 \mapsto L_2) \)
then
\[
\text{\textbf{if} } \quad F \sqsubseteq_2 \quad \text{\textbf{if} } \lambda f \cdot \lambda x \cdot f(x) \nabla_2 F(\lambda y \cdot f(x \nabla_1 y))(x)
\]
where \( \sqsubseteq_2 \) is the pointwise ordering on \( L_1 \mapsto L_2 \).

Example of second-order functional widenings in infinite domains
\[
F = \lambda f \cdot \lambda x \in 1, 1 \cdot \sqcup (f(x + [1, 1]) + [2, 2])
\]
\[
\text{\textbf{if} } \quad F \sqsubseteq_2 \quad \text{\textbf{if} } \lambda f \cdot \lambda x \cdot f(x) \nabla_2 F(\lambda y \cdot f(x \nabla_1 y))(x)
\]
where \( \sqsubseteq_2 \) is the pointwise ordering on \( L_1 \mapsto L_2 \).

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

References
In order to compute \( f([0, 0]) \) we follow a chaotic 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]) \cup (f^0([0, +\infty]) \cup \{[1, 1]\}) + [2, 2])
  = \bot \cup (f^0([0, +\infty]) \cup \{[1, 1]\}) + [2, 2])
  = ([1, 1] \cup (f^0([0, +\infty]) + [2, 2]))
  = ([1, 1] \cup [2, 2])
  = [1, 1]

since \( f([0, +\infty]) \) has not yet been computed, hence:
- \( f^0([0, +\infty]) = \bot \)

- \( f^2([0, +\infty]) = f^1([0, +\infty]) \cup (f^1([0, +\infty]) \cup \{[1, 1]\}) + [2, 2])
  = ([1, 1] \cup (f^1([0, +\infty]) + [2, 2]))
  = ([1, 1] \cup [3, 3])
  = [1, +\infty] \)
\[ f^3([0, +\infty]) = f^2([0, +\infty]) \nabla_2 (\{1, 1\} \cup (f^2([0, +\infty]) + [2, 2])) \]
\[ = [1, \infty) \nabla_2 (\{1, 1\} \cup (f^2([0, +\infty]) + [2, 2])) \]
\[ = [1, \infty) \nabla_2 (\{1, 1\} \cup (\{1, +\infty\} + [2, 2])) \]
\[ = [1, \infty) \nabla_2 (\{1, 1\} \cup (\{3, \infty\})) \]
\[ = [1, \infty) \nabla_2 [1, +\infty) \] \]

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".

Referece


Commented bibliography

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

- The first published paper on the subject:

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

- Extension to recursive procedures:

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

otherwise, see

- Presentation using transition systems (i.e. language independent semantics and computational analyzers, if you read french):
- Showing that widening/narrowing is strictly more powerful than abstraction/concretization (and therefore abstract model-checking which is a particular case):


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


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


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


published in structural form:


---

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/.