Principles of Abstract Interpretation

MIT press

Ch. **24**, Fixpoint induction

Patrick Cousot

pcousot.github.io

PrAbsInt@gmail.com   github.com/PrAbsInt/

These slides are available at
http://github.com/PrAbsInt/slides/slides-24--fixpoint-induction-PrAbsInt.pdf

Chapter 24

Ch. **24**, Fixpoint induction

General idea: from formal semantics to verification methods

- Proofs are by structural induction of the program abstract syntax;
- For iteration, the fixpoint definition of the semantics directly leads to proofs by fixpoint induction.

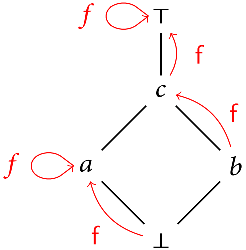
Fixpoint (or Park) induction [Cousot, 1978, p. 3.4.1], [Park, 1979, (2.3)]

Theorem (24.1) Let $f \in \mathcal{L} \stackrel{\nearrow}{\longrightarrow} \mathcal{L}$ be an increasing function on a complete lattice $\langle \mathcal{L}, \sqsubseteq, \bot, \top, \sqcap, \sqcup \rangle$ and $P \in \mathcal{L}$.
We have $\mathsf{lfp}^{\sqsubseteq} f \sqsubseteq P \Leftrightarrow \exists I \in \mathcal{L} \,.\, f(I) \sqsubseteq I \wedge I \sqsubseteq P$.

- I is called an *invariant* of f when $\mathsf{lfp}^{\sqsubseteq} f \sqsubseteq I$ and an *inductive invariant* when satisfying $f(I) \sqsubseteq I$.
- **Soundness** ($\Leftarrow$) states that if a statement is proved by the proof method then that statement is true.
- **Completeness** ($\Rightarrow$) states that the proof method is always applicable to prove a true statement.

Invariant versus inductive invariant

- An invariant is not necessarily inductive.
- Consider $f \in \mathcal{L} \xrightarrow{\nearrow} \mathcal{L}$ on the complete lattice L represented by the following Hasse diagram.



- $\mathsf{lfp}^{\sqsubseteq} f \sqsubseteq c$ so c is an invariant of f
- $f(c) = \top \not\sqsubseteq c$ so c is not an inductive invariant of f
- a and $\top$ are the only inductive invariants of f

Proof of the fixpoint induction Theorem 24.1

Theorem (24.1) Let $f \in \mathcal{L} \stackrel{\nearrow}{\longrightarrow} \mathcal{L}$ be an increasing function on a complete lattice $\langle \mathcal{L}, \sqsubseteq, \bot, \top, \sqcap, \sqcup \rangle$ and $P \in \mathcal{L}$.
We have $\mathsf{lfp}^{\sqsubseteq} f \sqsubseteq P \Leftrightarrow \exists I \in \mathcal{L} \,.\, f(I) \sqsubseteq I \wedge I \sqsubseteq P$.

Proof of Theorem 24.1 By Tarski fixpoint Theorem 15.6,
$\mathsf{lfp}^{\sqsubseteq} f = \bigsqcap\{x \in L \mid f(x) \sqsubseteq x\}$.
Soundness: If $I \in \mathcal{L}$ satisfies $f(I) \sqsubseteq I$ then $I \in \{x \in L \mid f(x) \sqsubseteq x\}$ so by definition of the glb $\bigsqcap$, $\mathsf{lfp}^{\sqsubseteq} f = \bigsqcap\{x \in L \mid f(x) \sqsubseteq x\} \sqsubseteq I \sqsubseteq P$.
Completeness: If $\mathsf{lfp}^{\sqsubseteq} f \sqsubseteq P$ then take $I = \mathsf{lfp}^{\sqsubseteq} f$ then $I = f(I)$ so $f(I) \sqsubseteq I$ by reflexivity and $I \sqsubseteq P$ by hypothesis. $\square$

Iteration (or Scott) induction [de Bakker and Scott, 1969]

Theorem (24.11) Let $f \in \mathcal{L} \xrightarrow{uc} \mathcal{L}$ be an upper-continuous function on a cpo $\langle \mathcal{L}, \sqsubseteq, \bot, \sqcup \rangle$ and $\mathcal{P} \in \wp(\mathcal{L})$. If $\bot \in \mathcal{P}$, $\forall x \in \mathcal{P} \,.\, f(x) \in \mathcal{P}$, and for any increasing chain $\{x_i \mid i \in \mathbb{N}\}$, $\forall i \in \mathbb{N} \,.\, x_i \in \mathcal{P}$ implies $\bigsqcup_{i\in\mathbb{N}} x_i \in \mathcal{P}$[a] then $\mathsf{lfp}^{\sqsubseteq} f \in \mathcal{P}$.

[a] $\mathcal{P}$ is said to be *admissible*.

Proof Let f^i be the iterates of f from $f^0 = \bot$. By recurrence, $\forall i \in \mathbb{N} \,.\, f^i \in \mathcal{P}$. f is increasing so $\{f^i \mid i \in \mathbb{N}\}$ is an increasing chain. By Theorem 15.26 and hypothesis, we conclude that $\mathsf{lfp}^{\sqsubseteq} f = \bigsqcup_{i\in\mathbb{N}} f^i \in \mathcal{P}$. □

- Note that the proof shows that the hypothesis is necessary only for the iterates of f.
- Generalized in [Cousot, 2019] to a sound *and* complete proof method (see the book version of this iteration induction Theorem 24.11).

Bibliography I

Cousot, Patrick (Mar. 21, 1978). "Méthodes itératives de construction et d'approximation de points fixes d'opérateurs monotones sur un treillis, analyse sémantique de programmes (in French)". Thèse d'État ès sciences mathématiques. Grenoble, France: Université de Grenoble Alpes.

— (2019). "On fixpoint/iteration/variant induction principles for proving total correctness of programs with denotational semantics". In: *LOPSTR 2019*. Vol. to appear. Lecture Notes in Computer Science. Springer. URL: `http://cs.unibo.it/projects/lopstr19/proceedingsLOPSTR19.pdf`.

de Bakker, Jacobus W. and Dana S. Scott (Aug. 1969). "A theory of programs". IBM Seminar Vienna, Austria (Unpublished notes).

Park, David Michael Ritchie (1979). "On the Semantics of Fair Parallelism". In: *Abstract Software Specifications*. Vol. 86. Lecture Notes in Computer Science. Springer, pp. 504–526.

Home work

Read Ch. **24** “Fixpoint induction” of

Principles of Abstract Interpretation
Patrick Cousot
MIT Press

The End, Thank you