New York University, CIMS, CS, Course CSCI-GA.3140-001, Spring 2024

"Abstract Interpretation"

Ch. 16, Fixpoint, Deductive, Inductive, Structural, Coinductive, and Bi-inductive Definitions

Patrick Cousot

pcousot@cs.nyu.edu cs.nyu.edu/~pcousot

Class 3, Thursday, February 5th, 2024, 4:55-6:55 PM, WWH, Room CIWW 202

These slides are available at
http://cs.nyu.edu/~pcousot/courses/spring24/CSCI-GA.3140-001/slides/03--2024-02-05-structural-fixpoint-prefix-trace-semantics/slides-16--fixpoint-inductive-deductive-structural-definitions-AI.pdf

Chapter 16

Ch. 16, Fixpoint, Deductive, Inductive, Structural, Coinductive, and Bi-inductive Definitions

Set-theoretic formal definitions

The problem is to formally define a subset $D \in \wp(\mathbb{U})$ of a set $\mathbb{U}$ (called the universe).

Example 16.1 Define the odd numbers $\mathbb{O}\mathbb{d}$ as a subset of the natural numbers $\mathbb{N}$. Same for the even numbers $\mathbb{E}\mathbb{n}$.

Fixpoint definitions

Fixpoint definition

- Since $\langle \wp(\mathbb{U}),\ \subseteq,\ \varnothing,\ \mathbb{U},\ \cup,\ \cap \rangle$ is a complete lattice, D can be defined as the least fixpoint $D \triangleq \mathsf{lfp}^{\subseteq} F$ of an increasing function $F \in \wp(\mathbb{U}) \xrightarrow{\nearrow} \wp(\mathbb{U})$

- So D is as the $\subseteq$-least solution of the equation $X = F(X)$

- So D is as the $\subseteq$-least solution of the constraint $F(X) \subseteq X$.

- D is well-defined (i.e. exists and is unique) by Tarski's theorem 15.6

Example 16.3 Continuing example 16.1, in the universe $\mathbb{N}$, the odd numbers are $\mathbb{O}\mathbb{d} \triangleq \mathsf{lfp}^{\subseteq} F$ where $F(X) \triangleq \{1\} \cup \{n + 2 \mid n \in X\}$.

Applying Tarski-Kantorovich's fixpoint theorem 15.21, we get $\mathbb{O}\mathbb{d} = \varnothing \cup \{1\} \cup \{1, 3\} \cup \ldots \cup \{1, 3, \ldots, 2k + 1\} \cup \ldots$.

Fixpoint definition

Definition 16.4 The fixpoint definition of $D \in \wp(\mathbb{U})$ by a $\subseteq$-increasing function $F \in \wp(\mathbb{U}) \stackrel{\nearrow}{\longrightarrow} \wp(\mathbb{U})$ is $D \triangleq \mathsf{lfp}^{\subseteq} F$.

Fixpoint definitions are well-defined

Theorem 16.5 D in definition 16.4 is well-defined.

Proof By Tarski's theorem 15.6. $\square$

Deductive definitions

Deductive definition

- A deductive definition of $D \in \wp(\mathbb{U})$ is given by a set of *inference rules* $R = \{\frac{P_i}{c_i} \mid i \in \Delta\}$
- $P_i \in \wp_f(\mathbb{U})$ is the *finite premise* and $c_i \in \mathbb{U}$ is the *conclusion* of the rule.
- A rule $\frac{P_i}{c_i} \in R$ states that if $P_i \subseteq D$ then $c_i \in D$.
- If $P_i = \emptyset$, the rule is called an *axiom* and states that $c_i \in D$.

en.wikipedia.org/wiki/Deductive_reasoning
en.wikipedia.org/wiki/Hilbert_system
en.wikipedia.org/wiki/Axiom
en.wikipedia.org/wiki/Rule_of_inference

example 16.6

- Continuing example 16.3, in the universe $\mathbb{N}$, the odd numbers are

$$\left\{\frac{\varnothing}{1}\right\} \cup \left\{\frac{\{n\}}{n+2} \;\middle|\; n \in \mathbb{N}\right\}$$

- 1 is an axiom
- from n is odd, we infer that $n+2$ is odd.
- As a shorthand, this can be written symbolically in the form of

an axiom $1 \in \mathbb{O}\mathbb{d}$ and an inference rule schema $\dfrac{n \in \mathbb{O}\mathbb{d}}{n{+}2 \in \mathbb{O}\mathbb{d}}$

- The instantiation for all $n \in \mathbb{N}$ yields the rules $\frac{\varnothing}{1}, \frac{\{0\}}{2}, \frac{\{1\}}{3}, \frac{\{2\}}{4}, \frac{\{3\}}{5}, \frac{\{4\}}{6}, \ldots, \frac{\{n\}}{n{+}2}, \ldots.$
- Notice that the rules $\frac{\{0\}}{2}, \frac{\{2\}}{4}, \ldots$ are useless since their premises cannot be derived form the deductive definition (0 is not an axiom).

proof

- A proof of p by rules R is a finite sequence $t_0 \dots t_n$ of elements of $\mathbb{U}$ such that
  - each $t_i, i \in [0, n]$ is deduced from $t_0 \dots t_{i-1}$ by application of a rule of R
  - $t_n = p$.

- Formally

Definition 16.7
$\text{is-provable}(p, R) \triangleq \exists t_0 \dots t_n \in \mathbb{U} \,.\, (\forall i \in [0, n] \,.\, \exists \frac{P}{c} \in R \,.\, P \subseteq \{t_0, \dots, t_{i-1}\} \wedge t_i = c) \wedge t_n = p$.

en.wikipedia.org/wiki/Mathematical_proof
en.wikipedia.org/wiki/Formal_proof

example 16.8

With $R \triangleq \left\{\frac{\varnothing}{1}\right\} \cup \left\{\frac{\{n\}}{n+2} \;\middle|\; n \in \mathbb{N}\right\}$ of example 16.6

- The proof that 5 is odd is $1, 3, 5$.
- To prove that 4 is not odd
  - $\frac{\{2\}}{4}$ is the only rule allowing us to prove that 4 would be odd,
  - This rule requires to prove that 2 is odd
  - The only applicable rule is $\frac{\{0\}}{2}$.
  - It remains to prove that 0 is odd
  - This is impossible since there is no rule with 0 as conclusion.

Set specified by a deductive definition

The set D defined by a set of rules R is $D \triangleq \{p \in \mathbb{U} \mid \text{is-provable}(p, R)\}$.

Example 16.9

Let us prove that $R \triangleq \{\frac{\varnothing}{1}\} \cup \{\frac{\{n\}}{n+2} \mid n \in \mathbb{N}\}$ in example 16.6 defines $\mathbb{Od} = \{2k+1 \mid k \in \mathbb{N}\}$

- We must prove that $2k+1$ is provable for all $k \in \mathbb{N}$
- 1 is provable by rule $\frac{\varnothing}{1}$
- Assume, by recurrence hypothesis, that we have got a proof $1, 3, 5, \ldots, 2k+1$ of $2k+1$.
- A proof of $2(k+1)+1 = 2k+3$ is by the rule $\frac{\{2k+1\}}{2k+3}$ such that $\{2k+1\} \subseteq \{1, 3, 5, \ldots, 2k+1\}$. By recurrence all $2k+1$, $k \in \mathbb{N}$ are provable so $\{2k+1 \mid k \in \mathbb{N}\} \subseteq \mathbb{Od}$.
- For the inverse inclusion, we can use a reasoning by *reductio ad absurdum* as illustrated in example 16.8[1].

[1] More precisely, Fermat's infinite descent (`en.wikipedia.org/wiki/Proof_by_infinite_descent`)

Deductive definition

Definition 16.10 (deductive definition) The deductive definition of $D \in \wp(\mathbb{U})$ by a deductive system of rules $\frac{P}{c} \in R$ is $D \triangleq \{p \in \mathbb{U} \mid \text{is-provable}(p, R)\}$.

Equivalence of the least fixpoint and deductive definition methods

A deductive definition can be expressed as a fixpoint definition and conversely.

Deductive definition as a fixpoint definition (section 16.3.1)

Consequence operator

For a deductive definition by rules $R = \{\frac{P_i}{c_i} \mid i \in \Delta\}$, define

- the *consequence operator* $F_R(X) \triangleq \{c \mid \exists \frac{P}{c} \in R \,.\, P \subseteq X\}$
- $F_R(X)$ is the set of consequences provable by R when X has already been proved
- The consequence operator F_R does not necessarily preserve joins but is increasing

Equivalence of the deductive and fixpoint definitions

Theorem 16.12 We have $D = \{p \in \mathbb{U} \mid \text{is-provable}(p, R)\} = \mathsf{lfp}^{\subseteq} F_R$ where $F_R(X) \triangleq \{c \mid \exists \frac{P}{c} \in R \mathrel{.} P \subseteq X\}$ is the *consequence operator* of R.

Theorem 16.12 may not hold when considering rules which premises can be infinite sets.

Proof of theorem 16.12

Let us first prove that $D \subseteq \bigcup\{F_R^n \mid n \in \mathbb{N}\}$

- Let F_R^n be the iterates of F_R
- Let us prove that F_R^n contains all elements with a proof of length less than or equal to n (F_R^n may contains proofs of longer length.)
  - This holds for $n = 0$ since $F_R^0 = \emptyset$ and there is no proof of length 0 or less
  - F_R^1 contains all elements with a proof of length 1 obtained by applying an axiom
  - Assume that F_R^n contains all elements with a proof of length less than or equal to n
  - If c has a proof of length less than or equal to $n+1$ then it is deduced by a rule $\frac{P}{c} \in R$ where the elements of P are proved before c hence have proofs of length less than or equal to n
  - It follows that $c \in \{c \mid \exists \frac{P}{c} \in R \,.\, P \subseteq F_R^n\} = F_R(F_R^n) = F_R^{n+1}$
  - By recurrence, for all $n \in \mathbb{N}$, all elements c with a proof of length less than or equal to n belong to F_R^n $\square$

- Now all elements in $\{p \in \mathbb{U} \mid \text{is-provable}(p, R)\}$ have a proof of some length $n \in \mathbb{N}$ so belong to $\bigcup\{F_R^n \mid n \in \mathbb{N}\}$
- We conclude that $D \subseteq \bigcup\{F_R^n \mid n \in \mathbb{N}\}$. $\square$

Conversely, let us prove, by contradiction, that $\bigcup\{F_R^n \mid n \in \mathbb{N}\} \subseteq D$

- Assume that $\bigcup\{F_R^n \mid n \in \mathbb{N}\}$ contains an element c not in D
- Since the $\langle F_R^n,\ n \in \mathbb{N}\rangle$ form a $\subseteq$-increasing chain, there exists a smallest n such that c belongs to F_R^n but does not belong to any of the F_R^m, $m < n$
- Among the pairs $\langle c,\ n\rangle$ with this property, chose one which minimize n
- So all F_R^m, $m < n$, have provable elements only, hence in D
- By definition of the iterates $F_R^n = F_R(F_R^{n-1})$
- So, by definition of F_R, c has a proof of length n
- This is a contradiction
- So $\bigcup\{F_R^n \mid n \in \mathbb{N}\} \subseteq D$.

By anstisymmetry, we conclude that $D = \bigcup\{F_R^n \mid n \in \mathbb{N}\}$.

Let us prove that $\mathsf{lfp}^{\subseteq} F_R = \bigcup\{F_R^n \mid n \in \mathbb{N}\}$ using Tarski-Kantorovich's fixpoint theorem 15.21

- F_R is increasing since if $X \subseteq X'$ then $P \subseteq X$ implies $P \subseteq X'$ and so $c \in F_R(X)$ implies $c \in F_R(X')$, proving $F_R(X) \subseteq F_R(X')$
- Since $\langle \wp(\mathbb{U}), \subseteq \rangle$ is a complete lattice, the lub $\cup$ exists
- It remains to prove that $F_R(\bigcup\{F_R^n \mid n \in \mathbb{N}\}) = \bigcup\{F_R(F_R^n) \mid n \in \mathbb{N}\}$ i.e.. $F_R(D) = D$.

- Let us first prove the $\supseteq$ inclusion.

$$\begin{array}{ll}
& \forall n \in \mathbb{N} \,.\, F_R^n \subseteq \bigcup\{F_R^n \mid n \in \mathbb{N}\} & \wr\text{def. lub } \cup\int \\
\Rightarrow & \forall n \in \mathbb{N} \,.\, F_R(F_R^n) \subseteq F_R(\bigcup\{F_R^n \mid n \in \mathbb{N}\}) & \wr F_R \text{ is } \subseteq\text{-increasing}\int \\
\Rightarrow & \forall n \in \mathbb{N} \,.\, F_R^{n+1} \subseteq F_R(\bigcup\{F_R^n \mid n \in \mathbb{N}\}) & \wr\text{def. iterates}\int \\
\Rightarrow & \forall n \in \mathbb{N} \,.\, F_R^n \subseteq F_R(\bigcup\{F_R^n \mid n \in \mathbb{N}\}) & \wr\text{since } F_R^0(X) = \varnothing\int \\
\Rightarrow & \bigcup\{F_R^n \mid n \in \mathbb{N}\} \subseteq F_R(\bigcup\{F_R^n \mid n \in \mathbb{N}\}) & \wr\text{def. lub}\int \\
\Rightarrow & D \subseteq F_R(D) & \wr\text{by } D = \bigcup\{F_R^n \mid n \in \mathbb{N}\}\int
\end{array}$$

- Conversely, we have to prove $\subseteq$
- Assume by reductio ad absurdum that $F_R(D) \nsubseteq D$ so that $\exists c \in F_R(D) \;.\; c \notin D$.
- Since $c \notin D$ so there exists no finite proof of c
- By def. F_R, $\exists \frac{P}{c} \in R \;.\; P \subseteq D$
- Because $P \subseteq D$, all elements p of P have a proof.
- Since the premise P must be finite, P has a finite proof (which is the finite sequence of the proofs of the elements of P), and therefore, using the rule $\frac{P}{c}$, c has also a finite proof
- This is a contradiction.

By antisymmetry $F_R(D) = D = \bigcup\{F_R^n \mid n \in \mathbb{N}\}$ so $D = \mathsf{lfp}^{\subseteq} F_R$ by Tarski-Kantorovich's fixpoint theorem 15.21. $\square$

Fixpoint definition as a deductive definition

Equivalence of the fixpoint and deductive definitions

Theorem 16.16 For a fixpoint definition $\mathsf{lfp}^{\subseteq} F$ define $R = \{\frac{P}{c} \mid P \subseteq \mathbb{U} \wedge c \in F(P)\}$. Then $F = F_R$ so $\mathsf{lfp}^{\subseteq} F_R = \mathsf{lfp}^{\subseteq} F$.

Note that if R turns out to have finite premises only, then $\{p \in \mathbb{U} \mid \text{is-provable}(p, R)\} = \mathsf{lfp}^{\subseteq} F_R$.

Proof

$$
\begin{aligned}
& F_R(X) \\
\triangleq\ & \{c \mid \exists \frac{P}{c} \in R \,.\, P \subseteq X\} && \text{(def. } F_R\text{)} \\
=\ & \{c \mid \exists P \subseteq \mathbb{U} \,.\, c \in F(P) \wedge P \subseteq X\} && \text{(def. } R\text{)} \\
=\ & F(X)
\end{aligned}
$$

($\subseteq$) if $c \in F(P)$ and $P \subseteq X$ then $c \in F(X)$ since F is $\subseteq$-increasing.
($\supseteq$) if $c \in F(X)$ then $\exists P \subseteq \mathbb{U} \,.\, c \in F(P) \wedge P \subseteq X$ by choosing $P = X$. $\square$

Well-definedness of deductive definitions

Well-definedness of deductive definitions

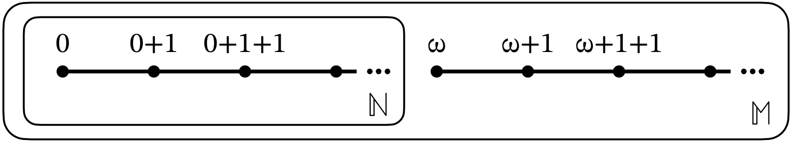
Theorem 16.16 D in definition 16.10 is well-defined.

Proof The deductive definition of D by rules R is equivalent to $D = \mathsf{lfp}^{\subseteq} F_R$ where F_R is $\subseteq$-increasing so is well-defined by Tarski's fixpoint theorem 15.6. $\square$

`en.wikipedia.org/wiki/Well-defined`

An aside on Peano definition of naturals $\mathbb{N}$

- Giuseppe Peano defined $\mathbb{N}$ as the set such that $0 \in \mathbb{N}$ and if $n \in \mathbb{N}$ then $n + 1 \in \mathbb{N}$.
- This is not well defined since there are many such sets such as



- One solution is to define $\mathbb{N}$ as the *smallest* set such that $0 \in \mathbb{N}$ and if $n \in \mathbb{N}$ then $n + 1 \in \mathbb{N}$ (which eliminates $\mathbb{M}$ strictly larger than $\mathbb{N}$)
- Another solution is to add that the induction rule $\frac{0 \in P, \quad n \in P \Rightarrow n+1 \in P}{\mathbb{N} \subseteq P}$ (which eliminates $\mathbb{M}$ by taking $P = \mathbb{N}$ which satisfies the premiss but not the conclusion $\mathbb{M} \subseteq P = \mathbb{N}$)
- The two solutions are equivalent.

Proof rule for a deductive definition

- Let $R = \{\frac{P_i}{c_i} \mid i \in \Delta\}$ be a deductive definition of a set D (as $\mathsf{lfp}^{\subseteq} F_R$ where F_R is the consequence operator for the rules R)
- Then the inductive proof method $\dfrac{\forall i \in \Delta \,.\, P_i \subseteq Q \Rightarrow c_i \in Q}{D \subseteq Q}$ is sound and complete
- *Soundness* (if the hypothesis of the proof rule holds then its conclusion holds): the premiss of the proof rule implies that $F_R(Q) \triangleq \{c_i \mid \exists i \in \Delta \,.\, P_i \subseteq Q\} \subseteq Q$ so by Tarski's fixpoint theorem $D = \mathsf{lfp}^{\subseteq} F_R \subseteq Q$.
- *Completeness* (if the conclusion holds then the hypothesis holds so the conclusion is provable by the proof rule): Assume $D = \mathsf{lfp}^{\subseteq} F_R \subseteq Q$ then strengthen Q to $Q' = \mathsf{lfp}^{\subseteq} F_R$, $F_R(Q') = \{c_i \mid \exists i \in \Delta \,.\, P_i \subseteq Q'\} = Q'$ so Q' satisfies the hypothesis of the proof rule. Then $Q' \subseteq Q$ implies $D = \mathsf{lfp}^{\subseteq} F_R \subseteq Q$ by transitivity.

Proof rule for a deductive definition

- Let $R = \{\frac{P_i}{c_i} \mid i \in \Delta\}$ be a deductive definition of a set D (as $\mathsf{lfp}^{\subseteq} F_R$ where F_R is the consequence operator for the rules R)
- Then the inductive proof method $\dfrac{\forall i \in \Delta \,.\, P_i \subseteq Q \Rightarrow c_i \in Q}{D \subseteq Q}$ is sound and complete
- *Soundness* (if the hypothesis of the proof rule holds then its conclusion holds): the premiss of the proof rule implies that $F_R(Q) \triangleq \{c_i \mid \exists i \in \Delta \,.\, P_i \subseteq Q\} \subseteq Q$ so by Tarski's fixpoint theorem $D = \mathsf{lfp}^{\subseteq} F_R \subseteq Q$.
- *Completeness* (if the conclusion holds then the hypothesis holds so the conclusion is provable by the proof rule): Assume $D = \mathsf{lfp}^{\subseteq} F_R \subseteq Q$ then strengthen Q to $Q' = \mathsf{lfp}^{\subseteq} F_R$, $F_R(Q') = \{c_i \mid \exists i \in \Delta \,.\, P_i \subseteq Q'\} = Q'$ so Q' satisfies the hypothesis of the proof rule. Then $Q' \subseteq Q$ implies $D = \mathsf{lfp}^{\subseteq} F_R \subseteq Q$ by transitivity.

So any deductive definition of a set D comes with a sound and complete method to prove that D satisfies a property P that is $D \subseteq P$.

Inductive definitions

Inductive definitions I

Inductive definitions are mathematical generalizations of recursive programs such as the factorial $\mathtt{f}(0) = 1$ and $\mathtt{f}(n) = n * \mathtt{f}(n-1)$ for $n \in \mathbb{Z}$.

```
$ cat factorial.c
#include <stdio.h>
int f(int n) {
    if (n==0) return 1;
    else return n * f(n - 1);
}
int main () {
    int n;
    scanf(”%d”, &n);
    printf(”%d! = %d\n”, n, f(n));
}
$ gcc factorial.c
$ echo ”7” | ./a.out
7! = 5040
$
```


Inductive definitions II

The difference is that at each recursive call a different function is called which parameters are all previously computed values.

$$
\begin{array}{lcl@{\quad}lcl}
\mathtt{f}(0) & = & 1 & D(0) & = & 1 \\
\mathtt{f}(1) & = & 1 * \mathtt{f}(0) & D(1) & = & F_1(\langle D(0)\rangle) \\
\mathtt{f}(2) & = & 2 * \mathtt{f}(1) & D(2) & = & F_2(\langle D(0),\ D(1)\rangle) \\
\mathtt{f}(3) & = & 3 * \mathtt{f}(2) & D(3) & = & F_3(\langle D(0),\ D(1),\ D(2)\rangle) \\
\ldots & & & \ldots & & \\
\mathtt{f}(n) & = & n * \mathtt{f}(n-1) & D(n) & = & F_n(\langle D(i),\ i \in [0, n-1]\rangle) \\
\ldots & & & \ldots & &
\end{array}
$$

Programmers would implement $F_n(\langle D(i),\ i \in [0, n-1]\rangle)$ by a function F taking n as a parameter of F and $\langle D(i),\ i \in [0, n-1]\rangle$ represented e.g. as a linear list of n elements.

Inductive definitions III

- The program might not terminate for negative values of the parameter
- The corresponding mathematical definition is not well-defined.

```
$ echo ”-7” | ./a.out
Segmentation fault: 11
$
```

Inductive definitions IV

Termination can be proved by recurrence.

- For $n = 0$, the function `f` returns the evaluation of expression 1, which terminates.
- Assume, by recurrence hypothesis that $\mathtt{f}(n)$ terminates.
- For the parameter $n + 1$, the function call returns the evaluation of $(n + 1) \times \mathtt{f}((n + 1) - 1)$. Since $\mathtt{f}(n)$ terminates by induction hypothesis, the evaluation of the expression terminates.
- By recurrence, all calls $\mathtt{f}(n)$, $n \in \mathbb{N}$ do terminate.

The corresponding reasoning on the mathematical inductive definition is by induction on a given well-founded relation $\preccurlyeq$ ($\preccurlyeq$ is $\leqslant$ on $\mathbb{N}$ for the factorial example).
Of course, computer integers are limited in size which leads to errors.

Inductive definitions V

```
$ echo ”20” | ./a.out
20! = -2102132736
$ echo ”40” | ./a.out
40! = 0
$ echo ”10000000” | ./a.out
Segmentation fault: 11
$
```

The corresponding mathematical reasoning must consider the universe $\mathbb{U}$ = [INT_MIN, INT_MAX] from C directive `#include <limits.h>`, not $\mathbb{U} = \mathbb{Z}$.

Well-founded relation

Definition 16.18 A relation $\preccurlyeq \in \wp(S \times S)$ on a set S is well-founded if and only there is no infinite (strictly decreasing chain if $\preccurlyeq$ is a partial order) sequence $x_0 \succ x_1 \succ x_2 \succ \ldots \succ x_n \succ x_{n+1} \succ \ldots$ of elements of S.

en.wikipedia.org/wiki/Well-founded_relation

Inductive proof

Theorem 16.19 Let $\preccurlyeq$ be a well-founded relation on S and $P \subseteq S$ be a property of the elements of S. We write $P(x)$ for $x \in P$. If

$$\forall x \in S \,.\, (\forall y \in S \,.\, (y \prec x) \Rightarrow P(y)) \Rightarrow P(x)$$

then $\forall x \in S \,.\, P(x)$.

Proof of theorem 16.19 By reductio ad absurdum, assuming $\exists x_0 \in S \,.\, \neg P(x_0)$, we construct an infinite sequence $x_0 \succ x_1 \succ x_2 \succ \ldots \succ x_n \succ x_{n+1} \succ \ldots$ of elements of S such that $\forall n \in \mathbb{N} \,.\, \neg P(x_n)$.

- Assume we have constructed the sequence up to x_n (i.e. x_0 to start with).
- Then, by contraposition, $\neg P(x_n)$ implies $\exists x_{n+1} \prec x_n \,.\, \neg P(x_{n+1})$.
- We get an infinite sequence of elements of S
- This is in contradiction with $\preccurlyeq$ is a well-founded relation on S. $\square$

en.wikipedia.org/wiki/Mathematical_induction

Inductive definition

Definition 16.20 The inductive definition of $D \in S \to \mathbb{U}$ where $\langle S, \preccurlyeq \rangle$ is well-founded has the form

(1) $D(m) \triangleq D_m$ where $D_m \in \mathbb{U}$ is a constant for minimal elements $m \in S$ (i.e. $\nexists s \in S \mathrel{.} s \prec m$);

(2) otherwise, $D(s) \triangleq F_s(\langle D(s'),\ s' \prec s \rangle)$ where $F_s \in (\{s' \in S \mid s' \prec s\} \to \mathbb{U}) \to \mathbb{U}$.

Most often, we use definition 16.20 for $\mathbb{U} = \wp(\mathbb{S})$ where $\mathbb{S}$ is a set.

Inductive definitions are well-defined

Theorem 16.21 D in definition 16.20 is well-defined.

Proof • We first observe that the first case is a special case of the second case by defining $F_m(\langle\rangle) = D_m$ for all $m \in S$ such that $\nexists s \in S \,.\, s \prec m$.

- The proof is by the induction proof theorem 16.19 on the well-found set $\langle S, \preccurlyeq\rangle$.
- Assume, by induction hypothesis that $D(s')$ is well-defined for all $s' \prec s$.
- Then $\langle D(s'),\ s' \prec s\rangle \in \{s' \in S \mid s' \prec s\} \to \mathbb{U}$ so $F_s(\langle D(s'),\ s' \prec s\rangle) \in \mathbb{U}$ is well-defined, proving that $D(s)$ is well-defined.
- By induction, $\forall s \in S \,.\, D(s) \in \mathbb{U}$ is well-defined.
- So $D \in S \to \mathbb{U}$ is well-defined. $\square$

Inductive definition as a fixpoint definition

Inductive definition can be expressed as an equivalent fixpoint definition

- Represent functions $f \in A \to B$ as a relation $\{\langle a,\ \mathsf{f}(a)\rangle \mid a \in A\}$
- The inductive definition 16.20 is $D = \mathsf{lfp}^{\subseteq} F$ where

$$\mathscr{F}(X) \triangleq \begin{array}{l} \{\langle m,\ D_m\rangle \mid \forall s' \in S \,.\, s' \nprec m\} \\ \cup \\ \{\langle s,\ F_s(\langle X(s'),\ s' \prec s\rangle)\rangle \mid \forall s' \prec s \,.\, s' \in \mathsf{dom}(X)\} \end{array}$$

Structural definitions

Structural definition

Definition 16.24 A structural definition is an inductive definition of the form

$$\begin{cases} \mathcal{D}[\![\mathsf{S}]\!] \quad \triangleq \quad f[\![\mathsf{S}]\!]\Big(\prod_{\mathsf{S}' \lhd \mathsf{S}} \mathcal{D}[\![\mathsf{S}']\!]\Big) \\ \mathsf{S} \in \mathbb{P}\mathbb{c} \end{cases} \tag{16.24}$$

where the well-founded order $\preccurlyeq$ is the syntactic order $\trianglelefteq$ on programs i.e. $\mathsf{S} \lhd \mathsf{S}'$ if and only if S is a strict syntactic component of S'.

The strict syntactic order $\lhd$ (example 16.25)

$$
\begin{array}{rcll}
\mathtt{P} & ::= & \mathtt{Sl}\ \ell & \mathtt{Sl} \lhd \mathtt{P} \\
\mathtt{S} & ::= & & \\
& & \mathtt{x} = \mathtt{E}\ ; & \mathtt{x} \lhd \mathtt{S}, \quad \mathtt{E} \lhd \mathtt{S} \\
& | & ; & \\
& | & \mathbf{if}\ (\mathtt{B})\ \mathtt{S}_t & \mathtt{B} \lhd \mathtt{S}, \quad \mathtt{S}_t \lhd \mathtt{S} \\
& | & \mathbf{if}\ (\mathtt{B})\ \mathtt{S}_t\ \mathbf{else}\ \mathtt{S}_f & \mathtt{B} \lhd \mathtt{S}, \quad \mathtt{S}_t \lhd \mathtt{S}, \quad \mathtt{S}_f \lhd \mathtt{S} \\
& | & \mathbf{while}\ (\mathtt{B})\ \mathtt{S}_b & \mathtt{B} \lhd \mathtt{S}, \quad \mathtt{S}_b \lhd \mathtt{S} \\
& | & \mathbf{break}\ ; & \\
& | & \{\ \mathtt{Sl}\ \} & \mathtt{Sl} \lhd \mathtt{S} \\
\mathtt{Sl} & ::= & \mathtt{Sl}'\ \mathtt{S} \mid \epsilon & \mathtt{Sl}' \lhd \mathtt{Sl}, \quad \mathtt{S} \lhd \mathtt{Sl}, \quad \epsilon \lhd \mathtt{Sl}
\end{array}
$$

- $\lhd$ is well-founded
- The *syntactic order* $\lhd$ is $\mathtt{S} \unlhd \mathtt{S}' \triangleq \mathtt{S} \lhd \mathtt{S}' \vee \mathtt{S} = \mathtt{S}'$.
- The *recursive syntactic order* $\lhd^+$ is the transitive closure of $\lhd$.
- The *recursive subcomponent partial order* $\lhd^*$ is the transitive closure of $\unlhd$ i.e. the reflexive transitive closure of $\lhd$.

Structural proofs

Corollary 16.31 If

$$\forall S \lhd^* P \,.\, (\forall S' \lhd^* P \,.\, (S' \lhd S) \Rightarrow \mathcal{P}(S')) \Rightarrow \mathcal{P}(S)$$

then $\forall S \lhd^* P \,.\, \mathcal{P}(S)$.

The structural induction hypothesis $\mathcal{P}(S')$ is assumed to hold for all $S' \lhd S$ when proving $\mathcal{P}(S)$.

Proof By theorem 16.19 for the syntactic order $\langle\{S \mid S \lhd^* P\}, \unlhd\rangle$ of example 16.25 which is well-founded. $\square$

Structural definitions are well-defined

Corollary 16.31 A structural definition 16.20 for the syntactic order $\langle\{S \mid S \lhd^* P\}, \unlhd\rangle$ is well-defined.

Proof By structural induction and corollary 16.31. $\square$

Structural proofs were originally introduced by Rod Burstall [Burstall, 1969] for recursively defined structures such as data types.

Coinductive definitions

Coinductive definitions

The same way that deductive definitions are equivalent to least fixpoint definitions by theorems 16.12 and 16.16, coinductive definitions are equivalent to greatest fixpoint definitions.

`en.wikipedia.org/wiki/Coinduction`

Coinductive definition

Definition 16.34 The coinductive definition of $D \in \wp(\mathbb{U})$ by a deductive system of rules $\frac{P}{c} \in R$ is $\mathsf{gfp}^{\subseteq} F R$ where $F_R(X) \triangleq \{c \mid \exists \frac{P}{c} \in R \,.\, P \subseteq X\}$ is the consequence operator of R.

Infinitary language (example 16.35) I

- Let $\mathbb{U}$ be the set of infinite strings on the alphabet $\{\mathsf{a}, \mathsf{b}\}$.
- Let $R = \left\{\frac{\{\sigma\}}{\mathsf{a}\sigma} \mid \sigma \in \mathbb{U}\right\}$
- This coinductive definition states that if $\sigma \in \mathbb{U}$ is an infinite string on the alphabet $\{\mathsf{a}, \mathsf{b}\}$ in D then $a\sigma$ is also an infinite string in D.
- This coinductive definition is equivalent to $\mathsf{gfp}^{\subseteq} FR$ where $F_R(X) \triangleq \{\mathsf{a}\sigma \mid \sigma \in X\}$.
- F_R preserves arbitrary meets
- So by the dual of Tarski-Kantorovich's fixpoint theorem 15.21, the greatest fixpoint is the limit of the following $\subseteq$-decreasing chain of iterates of F_R

Infinitary language (example 16.35) II

$$
\begin{array}{lcl}
F_R^0 & = & \mathbb{U} \\
F_R^1 & = & \{\mathsf{a}\sigma \mid \sigma \in \mathbb{U}\} \\
\ldots & & \\
F_R^n & = & \{\mathsf{a}^n\sigma \mid \sigma \in \mathbb{U}\} \\
\ldots & & \\
D & = & \mathsf{gfp}^{\subseteq} FR \quad = \quad \bigcap_{n\in\mathbb{N}} F_R^n \quad = \quad \{\mathsf{aaaaa}\ldots\} \quad = \quad \{\mathsf{a}^\omega\}
\end{array}
$$

- For the limit observe that
  - all iterates contains a^ω
  - if an infinite string contains a b, say at rank n in the string, then it does not belong to F_R^n hence not to the limit $\bigcap_{n\in\mathbb{N}} F_R^n$.

Bi-inductive definition

A combination of inductive and co-inductive definitions, see section 16.7.

Conclusion

Conclusion

- Fixpoint, deductive, inductive, and structural definitions are used to provide well-defined specifications of the semantics of programs and their abstractions.

- Context-free grammars are a particular case

Conclusion

- Fixpoint, deductive, inductive, structural, coinductive, and bi-inductive definitions are used in the definition of semantics, verification conditions, and static analysis of programs
- An overview of set-theoretic formal definitions is given in [Aczel, 1977].
- A generalization from $\wp(\mathbb{U})$ to complete partial orders is considered in [P. Cousot and R. Cousot, 1995].

Bibliography

Bibliography I

[redacted]

Aczel, Peter (1977). "An Introduction to Inductive Definitions.". In John Barwise, ed. *Handbook of Mathematical Logic.* Amsterdam: North–Holland. Chap. 7, pp. 739–782.

Burstall, Rod M. (1969). "Proving Properties of Programs by Structural Induction.". *Computer Journal.* 12.1, pp. 41–48.

Cousot, Patrick and Radhia Cousot (1995). "Compositional and Inductive Semantic Definitions in Fixpoint, Equational, Constraint, Closure–Condition, Rule–Based and Game–Theoretic Form.". In *CAV.* Vol. 939. Lecture Notes in Computer Science. Springer, pp. 293–308.

The End, Thank you