Algebraic Foundations of Computer Science.

Closures.

Ferucio Laurenţiu Ţiplea

Department of Computer Science
“AL.I.Cuza” University of Iaşi
Iaşi, Romania
E-mail: fltiplea@info.uaic.ro

Spring 2013

Outline

Closures – example

Example 1

Let A be a set of atomic propositions. The set $PF(A)$ of propositional formulas over A is the least set which fulfills the following properties:

- *$a \in PF(A)$, for any $a \in A$ (that is, $A \subseteq PF(A)$);*
- *if α and β are propositional formulas over A, then*

$$\neg\alpha,\ (\alpha \vee \beta),\ (\alpha \wedge \beta),\ (\alpha \Rightarrow \beta),\ \text{and } (\alpha \Leftrightarrow \beta)$$

  are propositional formulas over A.

The three key features of $PF(A)$:

1. "includes A"
2. "closed under" $\neg$, $\vee$, $\wedge$, $\Rightarrow$, $\Leftrightarrow$
3. "least set" with the above properties

Constructors and closures

An n-ary constructor over a set V is a relation r from V^n to V. That is, the elements of r are of the form $((a_1, \ldots, a_n), a)$.

Given an n-ary constructor r and a set A, denote by $r(A)$ the set:

$$r(A) = \{a | (\exists a_1, \ldots, a_n \in A)(((a_1, \ldots, a_n), a) \in r)\}$$

Definition 2

Let A be a set and $\mathcal{R}$ be a set of constructors. The closure of A under $\mathcal{R}$ is the least set $B \subseteq V$ with the properties:

- *$A \subseteq B$;*
- *B is closed under $\mathcal{R}$, i.e., $r(B) \subseteq B$, for any $r \in \mathcal{R}$.*

Existence of Closures

Theorem 3 (Existence of closures)

Given a set A and a set $\mathcal{R}$ of constructors, the closure of A under $\mathcal{R}$ exists and it is unique. Moreover, if $\mathcal{R}[A]$ denotes the closure of A under $\mathcal{R}$, then

$$\mathcal{R}[A] = \bigcup_{m\geq 0} B_m,$$

where

- $B_0 = A$;
- $B_{m+1} = B_m \cup \bigcup_{r\in\mathcal{R}} r(B_m)$, for any $m \geq 0$;

The closure of A under $\mathcal{R}$ is the union of a chain of sets:

$$B_0 = A, B_1 = B_0 \cup \mathcal{R}(B_0), B_2 = B_1 \cup \mathcal{R}(B_1), \ldots, \bigcup_{m\geq 0} B_m = \mathcal{R}[A],$$

where $\mathcal{R}(B_i) = \bigcup_{r\in\mathcal{R}} r(B_i)$.

The set of natural numbers as a closure

Definition 4

The successor of a set x, denoted $S(x)$, is the set $S(x) = x \cup \{x\}$.

Recall that natural numbers are defined as follows:

- $0 = \emptyset$;
- $1 = S(0) = \{0\} = \{\emptyset\}$;
- $2 = S(1) = \{0, 1\} = \{\emptyset, \{\emptyset\}\}$ etc.

Therefore, $\mathbb{N}$ is the closure of $\{0\}$ under $\mathcal{R} = \{S\}$.

Closures of a binary relation

Definition 5

*The **reflexive closure** of a binary relation $\rho \subseteq A \times A$ is the least reflexive binary relation $r(\rho)$ which includes ρ.*

Fact: $r(\rho)$ can be computed as follows:

$$r(\rho) = \rho \cup \iota_A$$

Definition 6

*The **symmetric closure** of a binary relation $\rho \subseteq A \times A$ is the least symmetric binary relation $s(\rho)$ which includes ρ.*

Fact: $s(\rho)$ can be computed as follows:

$$s(\rho) = \rho \cup \rho^{-1}$$

Closures of a binary relation

Definition 7

The transitive closure of a binary relation $\rho \subseteq A \times A$ is the least transitive binary relation $t(\rho)$ which includes ρ.

Fact: $t(\rho)$, also denoted by ρ^+, can be computed as follows:

$$t(\rho) = \rho^+ = \bigcup_{m \geq 1} \rho^m,$$

where

- $\rho^1 = \rho$ and
- $\rho^{m+1} = \rho \circ \rho^m$, for all $m \geq 1$.

Definition 8

The reflexive and transitive closure of a binary relation $\rho \subseteq A \times A$ is the least reflexive and transitive binary relation ρ^* which includes ρ.

Fact: ρ^* can be computed as follows:

$$\rho^* = t(r(\rho)) = r(t(\rho)) = \bigcup_{m \geq 0} \rho^m,$$

where

- $\rho^0 = \iota_A$ and
- $\rho^{m+1} = \rho \circ \rho^m$, for all $m \geq 0$.

Closures of a binary relation

Definition 9

The closure under equivalence of a binary relation $\rho \subseteq A \times A$ is the least equivalence relation $equiv(\rho)$ which includes ρ.

Fact: $equiv(\rho)$ can be computed as follows:

$$equiv(\rho) = t(s(r(\rho))) = t(r(s(\rho))) = r(t(s(\rho))).$$

Remark 1

In general, $s(t(\rho)) \neq t(s(\rho))$.

Structural induction

Theorem 10 (Structural induction)

Let $B = \mathcal{R}[A]$ be the closure of A under $\mathcal{R}$ and let P be a property such that:

- $P(a)$, for any $a \in A$;
- $(P(a_1) \wedge \cdots \wedge P(a_n) \Rightarrow P(a))$, for any $r \in \mathcal{R}$ and $a_1, \ldots, a_n, a \in B$ with $((a_1, \ldots, a_n), a) \in r$.

Then, P is satisfied by any $a \in B$.

Remark 2

1. Structural induction is equivalent to mathematical induction.
2. Structural induction is more appropriate for proving properties of closures than mathematical induction.

Structural induction – example

Example 11

Let A be a set of atomic propositions. The set $PF(A)$ of propositional formulas as defined in Example 1 is the closure of A under some set of constructors (prove it!).

Let $P(\alpha)$ be the following property:

$$P(\alpha): \quad \alpha \text{ has as many left brackets as right brackets.}$$

By structural induction we can easily prove that P is satisfied by all propositional formulas over A (prove it!).

Definitions by induction

Definition 12

A set B is **inductively defined by A and $\mathcal{R}$** *if $B = \mathcal{R}[A]$.*

If $B = \mathcal{R}[A]$, then B is obtained as follows:

- $B_0 = A$;
- $B_{m+1} = B_m \cup \mathcal{R}(B_m)$, for all $m \geq 0$;
- $B = \bigcup_{m \geq 0} B_m$.

If the chain

$$B_0, B_1, B_2, \ldots, B_m, B_{m+1} = B_m, B_{m+2} = B_m, \ldots$$

stabilizes to some set B_m, then its union is B_m and, therefore, $B = B_m$.

Definitions by induction

A definition by induction corresponds to the following while-loop (that might be non-terminating):

Algorithm 1: Computing closures

---

input : set A and set $\mathcal{R}$ of constructors;
output: $B = \mathcal{R}[A]$;

begin
  $B := A$;
  while $\mathcal{R}(B) \not\subseteq B$ **do**
    $B := B \cup \mathcal{R}(B)$

Definitions by recursion

Assume that B is inductively defined by A and $\mathcal{R}$. It would be a good idea to define functions f on B in a recursive way as follows:

- define f for any $a \in A$;
- if $((a_1, \ldots, a_n), a) \in r$ and the function has already been defined for $a_1, \ldots, a_n$, then define the function for a as a combinations of the values $f(a_1), \ldots, f(a_n)$ in the form

$$h(r)(f(a_1), \ldots, f(a_n)),$$

where h associates a (partial) function $h(r)$ to r.

Definitions by recursion

The definition above has a main drawback: it could not work for some sets B. Just think that the element a above might be defined in at least two different ways,

$$((a_1, \ldots, a_n), a) \in r$$

and

$$((a'_1, \ldots, a'_m), a) \in r'.$$

In such a case, you must be assured that

$$h(r)(f(a_1), \ldots, f(a_n)) = h(r')(f(a'_1), \ldots, f(a'_m)).$$

The easiest way to have this property fulfilled is to ask for each element $a \in B$ to have exactly one inductive construction of it from A and $\mathcal{R}$. If B has this property then it is called a free inductively defined set.

However, for inductively defined sets we can prove the following result:

Lemma 13

Let $B = \mathcal{R}[A]$, C a set, $g : A \to C$, and h a function which associates a partial function $h(r) : C^n \to C$ to each $r \in \mathcal{R}$, where n is the arity of r. Then, there exists a unique relation $f \subseteq B \times C$ such that:

(1) $(a, g(a)) \in f$, for any $a \in A$;

(2) if $(a_1, b_1), \ldots, (a_n, b_n) \in f$, $((a_1, \ldots, a_n), a) \in r$ and $h(r)(b_1, \ldots, b_n)\downarrow$, then $(a, h(r)(b_1, \ldots, b_n)) \in f$;

(3) f is the least relation from B to C which satisfies (1) and (2).

Definitions by recursion

Definition 14

A set B is called **free inductively defined** *by A and $\mathcal{R}$ if, for any $a \in B$,*

- *either $a \in A$,*
- *or there exists a unique $r \in \mathcal{R}$ and a unique n-tuple $(a_1, \ldots, a_n)$ such that $((a_1, \ldots, a_n), a) \in r$, where n is the arity of r (for $n = 0$ we understand that $a \in r$).*

Now, we can obtain the following important result.

Theorem 15 (Recursion theorem)

Let B, C, g, and h as in Lemma 13. If B is free inductively defined by A and $\mathcal{R}$, then the binary relation f from Lemma 13 is a function.

Definitions by recursion

A slight extension of the recursion theorem is the following:

Theorem 16

Let $B = \mathcal{R}[A]$, C a set, $g : A \to C$, and h a function which associates a partial function $h(r) : B^n \times C^n \to C$ to each $r \in \mathcal{R}$, where n is the arity of r. If B is free inductively defined by A and $\mathcal{R}$, then there exists a unique function $f : B \to C$ such that:

(1) $f(a) = g(a)$, for any $a \in A$;

(2) $f(a) = h(r)(a_1, \ldots, a_n, f(a_1), \ldots, f(a_n))$, for any $a, a_1, \ldots, a_n$ with $((a_1, \ldots, a_n), a) \in r$ and $h(r)(a_1, \ldots, a_n, f(a_1), \ldots, f(a_{n_r}))\downarrow$, where n is the arity of r.

Definitions by recursion – example

Example 17

Let $PF(A)$ be the set of propositional formulas over A. It is easy to see that this set is free inductively defined.
Define a function $f : PF(A) \to \mathbb{N}$ in a recursive way as follows:

- $f(a) = 1$, *for any* $a \in A$;
- $f(\neg\alpha) = f(\alpha)$, *for any* $\alpha \in PF(A)$;
- $f((\alpha \vee \beta)) = f(\alpha) + f(\beta)$, *for any* $\alpha, \beta \in PF(A)$;
- $f((\alpha \wedge \beta)) = f(\alpha) + f(\beta)$, *for any* $\alpha, \beta \in PF(A)$;
- $f((\alpha \Rightarrow \beta)) = f(\alpha) + f(\beta)$, *for any* $\alpha, \beta \in PF(A)$;
- $f((\alpha \Leftrightarrow \beta)) = f(\alpha) + f(\beta)$, *for any* $\alpha, \beta \in PF(A)$.

The function f returns the length of propositional formulas.

Definitions by recursion – more examples

Pick up your favorite programming language and:

- show that its set of arithmetic and logic expressions is inductively defined;
- define recursively the length of an arithmetic expression;
- define inductively the set of variables of an arithmetic expression;
- define recursively the "height" of an arithmetic expression.

Definitions by recursion

Algebraic Foundations of Computer Science (AFCS)

Prof.Dr. F.L. Tiplea

Closures

Structural induction

Definitions by induction

Definitions by recursion

Course readings

An important particular case of the recursion theorem:

Theorem 18 (Recursion theorem for $\mathbb{N}$)

Let A be a set, $a \in A$, and $h : \mathbb{N} \times A \to A$ be a function. Then, there exists a unique function $f : \mathbb{N} \to A$ such that:

(1) $f(0) = a$;

(2) $f(n+1) = h(n, f(n))$, for any n.

This result can be strengthen to:

Theorem 19 (Parametric recursion theorem for $\mathbb{N}$)

Let A and P be sets, and $g : P \to A$ and $h : P \times \mathbb{N} \times A \to A$ be two functions. Then, there exists a unique function $f : P \times \mathbb{N} \to A$ such that:

(1) $f(p, 0) = g(p)$, for any $p \in P$;

(2) $f(p, n+1) = h(p, n, f(p, n))$, for any $p \in P$ and $n \in \mathbb{N}$.

Definitions by recursion

Addition, multiplication, and exponentiation on natural numbers are defined by recursion:

- Addition:
  - $x + 0 = x$
  - $x + (n + 1) = (x + n) + 1$;
- Multiplication:
  - $x \cdot 0 = 0$
  - $x \cdot (n + 1) = (x \cdot n) + x$;
- Exponentiation:
  - $x^0 = 1$
  - $x^{n+1} = (x^n) \cdot x$.

Definitions by recursion

In some cases the value of a function f at a natural number n may depend on the values of f at $0, \ldots, n-1$ (Fibonacci's sequence is such an example).

The recursion in such cases is called hereditary.

Theorem 20 (Hereditary recursion theorem)

Let A be a set, $S = \bigcup_{n \in \mathbb{N}} A^n$, and $h : \mathbb{N} \times S \to A$ be a function. Then, there exists a unique function $f : \mathbb{N} \to A$ such that

$$f(n) = h(n, f|_n),$$

for any $n \in \mathbb{N}$ (recall that $f|_0 = f|_\emptyset = \emptyset \in A^0$).

Exercise: Develop a parametric version of the hereditary recursion theorem.

Course readings

1. F.L. Ţiplea: *Fundamentele Algebrice ale Informaticii*, Ed. Polirom, Iaşi, 2006, pag. 70–79.

2. F.L. Ţiplea: *Introducere în Teoria Mulţimilor*, Ed. Univesităţii "Al.I.Cuza", Iaşi, 1998, pag. 83–90.