1.5 Course of Values Recursion

1.5.1 Introduction. In this section we will study a simple generalization of primitive recursion called *course of values recursion.* In this new scheme the value at the $(n{+}1)$-stage depends not only on the value from the previous n-th stage but on the values from $<n$-th stages as well. We show that the class of primitive recursive functions are closed under course of values recursion by reducing it to primitive recursion.

1.5.2 Example. The scheme of course of values recursion is best explained with an example. Consider again the function $\mathrm{fib}(n)$ which yields the n-th element of the sequence of Fibonacci:

$$\begin{aligned} \mathrm{fib}(0) &= 0 \\ \mathrm{fib}(1) &= 1 \\ \mathrm{fib}(n+2) &= \mathrm{fib}(n+1) + \mathrm{fib}(n). \end{aligned}$$

These three identities can be re-written in a more compact form with the help of the case discrimination function D (see Par. 1.3.1) as follows:

$$\begin{aligned} \mathrm{fib}(0) &= 0 \\ \mathrm{fib}(n+1) &= D\big(n, \mathrm{fib}(n) + \mathrm{fib}(n \mathbin{\dot{-}} 1), 1\big). \end{aligned}$$

This is not a definition by primitive recursion since the value $\mathrm{fib}(n+1)$ depends not only on the value $\mathrm{fib}(n)$ but on the value $\mathrm{fib}(n \mathbin{\dot{-}} 1))$ as well.

1.5.3 Course of values recursive definitions. Suppose that

$$\rho[\vec{y}], \tau[x, \vec{z}, \vec{y}], \xi_1[x, \vec{y}], \dots, \xi_k[x, \vec{y}]$$

are terms which do not apply f with all their free variables indicated s.t.

$$\xi_1[x, \vec{y}] \le x \quad \dots \quad \xi_k[x, \vec{y}] \le x. \tag{1}$$

Consider the $(n{+}1)$-ary function symbol f satisfying

$$f(0, \vec{y}) = \rho[\vec{y}] \tag{2}$$

$$f(x+1, \vec{y}) = \tau\left[x, f\big(\xi_1[x, \vec{y}], \vec{y}\big), \dots, f\big(\xi_k[x, \vec{y}], \vec{y}\big), \vec{y}\right], \tag{3}$$

We say that f is defined by *course of values recursion.* The definition can be viewed as a function operator which takes all functions applied in the terms $\rho, \tau, \xi_1, \dots, \xi_k$ and yields the function f as a result.

Fixing notation. We keep the notation introduced in this paragraph fixed until the end of this section where we prove in Thm. 1.5.6 that the class of primitive recursive functions is closed under course of values recursion.

1.5.4 The outline of the proof. We wish to introduce as primitive recursive the course of values function $\overline{f}(x,\vec{y})$ yielding the course of values sequence for $f(x,\vec{y})$, i.e. we would like to have

$$\overline{f}(x,\vec{y}) = \langle f(x,\vec{y}), f(x-1,\vec{y}), \ldots, f(2,\vec{y}), f(1,\vec{y}), f(0,\vec{y}), 0\rangle.$$

Note that then the following holds for every $u \le x$:

$$f(u,\vec{y}) = \big(\overline{f}(x,\vec{y})\big)_{x \dot{-} u}.$$

The function f can be thus defined explicitly by

$$f(x,\vec{y}) = \big(\overline{f}(x,\vec{y})\big)_0.$$

1.5.5 Course of values function. We define the $(n{+}1)$-ary course of value function $\overline{f}(x,\vec{y})$ as primitive recursive by the primitive recursive definition:

$$\begin{aligned}\overline{f}(0,\vec{y}) &= \langle \rho[\vec{y}], 0\rangle\\ \overline{f}(x+1,\vec{y}) &= \Big\langle \tau\Big[x, \big(\overline{f}(x,\vec{y})\big)_{x \dot{-} \xi_1[x,\vec{y}]}, \ldots, \big(\overline{f}(x,\vec{y})\big)_{x \dot{-} \xi_k[x,\vec{y}]}, \vec{y}\Big], \overline{f}(x,\vec{y})\Big\rangle.\end{aligned}$$

The following holds for $i = 1,\ldots,k$:

$$\Big(\overline{f}\big(\xi_i[x,\vec{y}],\vec{y}\big)\Big)_0 = \big(\overline{f}(x,\vec{y})\big)_{x \dot{-} \xi_i[x,\vec{y}]}. \tag{1}$$

Proof. First note that we have

$$T \vdash \big(\overline{f}(x_1+x_2,\vec{y})\big)_{x_2} = \big(\overline{f}(x_1,\vec{y})\big)_0. \tag{$\dagger_1$}$$

This is proved by induction on x_2. The base case is obvious and the induction step follows from

$$\begin{aligned}\big(\overline{f}(x_1+x_2+1,\vec{y})\big)_{x_2+1} &= \Big(\big\langle \tau[\ldots], \overline{f}(x_1+x_2,\vec{y})\big\rangle\Big)_{x_2+1} =\\ &= \big(\overline{f}(x_1+x_2,\vec{y})\big)_{x_2} \overset{\text{IH}}{=} \big(\overline{f}(x_1,\vec{y})\big)_0.\end{aligned}$$

We are now ready to prove (1). We have $\xi_i[x,\vec{y}] \le x$ by 1.5.3(1) and thus

$$\xi_i[x,\vec{y}] + (x \dot{-} \xi_i[x,\vec{y}]) = x \tag{$\dagger_2$}$$

for every $i = 1,\ldots,k$. We obtain

$$\Big(\overline{f}\big(\xi_i[x,\vec{y}],\vec{y}\big)\Big)_0 \overset{(\dagger_1)}{=} \Big(\overline{f}\big(\xi_i[x,\vec{y}]+(x\dot{-}\xi_i[x,\vec{y}]),\vec{y}\big)\Big)_{x\dot{-}\xi_i[x,\vec{y}]} \overset{(\dagger_2)}{=}$$
$$= \big(\overline{f}(x,\vec{y})\big)_{x\dot{-}\xi_i[x,\vec{y}]}. \qquad \square$$

1.5.6 Theorem *Primitive recursive functions are closed under course of values recursion.*

Proof. Let f be defined by the course of values recursion as in Par. 1.5.3 from p.r. functions. Let further $\overline{f}$ be its course of values function as in Par. 1.5.5. We claim that we have

$$f(x,\vec{y}) = \big(\overline{f}(x,\vec{y})\big)_0.$$

The function $\overline{f}$ is primitive recursive and so is f.

The property is proved by complete induction on x. There are two cases to consider. If $x = 0$ then we have

$$f(0,\vec{y}) = \rho[\vec{y}] = \big(\langle\rho[\vec{y}],0\rangle\big)_0 = \big(\overline{f}(0,\vec{y})\big)_0.$$

If $x = x'+1$ for some x' then $\xi_i[x',\vec{y}] \le x' < x'+1$ by 1.5.3(1) and therefore

$$f(x'+1,\vec{y}) = \tau\big[x', f\big(\xi_1[x',\vec{y}],\vec{y}\big),\ldots,f\big(\xi_k[x',\vec{y}],\vec{y}\big),\vec{y}\big] \overset{k\times\mathrm{IH}}{=}$$
$$= \tau\Big[x',\Big(\overline{f}\big(\xi_1[x',\vec{y}],\vec{y}\big)\Big)_0,\ldots,\Big(\overline{f}\big(\xi_k[x',\vec{y}],\vec{y}\big)\Big)_0,\vec{y}\Big] \overset{1.5.5(1)}{=}$$
$$= \tau\Big[x',\big(\overline{f}(x',\vec{y})\big)_{x'\dot{-}\xi_1[x',\vec{y}]},\ldots,\big(\overline{f}(x',\vec{y})\big)_{x'\dot{-}\xi_k[x',\vec{y}]},\vec{y}\Big] = \big(\overline{f}(x'+1,\vec{y})\big)_0. \qquad \square$$