f and f' in $\mathcal{F}_{p+1}$ by the conditions

$$\begin{aligned}
f(x_1, x_2, \ldots, x_p, 0) &= g(x_1, x_2, \ldots, x_p);\\
f'(x_1, x_2, \ldots, x_p, 0) &= g'(x_1, x_2, \ldots, x_p);\\
f(x_1, x_2, \ldots, x_p, y+1) &= h(x_1, x_2, \ldots, x_p, y, f(x_1, x_2, \ldots, x_p, y),\\
&\qquad f'(x_1, x_2, \ldots, x_p, y));\\
f'(x_1, x_2, \ldots, x_p, y+1) &= h'(x_1, x_2, \ldots, x_p, y, f(x_1, x_2, \ldots, x_p, y),\\
&\qquad f'(x_1, x_2, \ldots, x_p, y)).
\end{aligned}$$

We will show that if all four of g, g', h, h' are primitive recursive, then so are f and f'. To do this, let us introduce the function $k = \alpha_2(f, f')$. This function is definable by recursion as follows:

$$\begin{aligned}
&k(x_1, x_2, \ldots, x_p, 0) = \alpha_2(g(x_1, x_2, \ldots, x_p), g'(x_1, x_2, \ldots, x_p));\\
&k(x_1, x_2, \ldots, x_p, y+1)\\
&\quad = \alpha_2(h(x_1, x_2, \ldots, x_p, y, \beta_2^1(k(x_1, x_2, \ldots, x_p, y)), \beta_2^2(k(x_1, x_2, \ldots, x_p, y))),\\
&\qquad\quad h'(x_1, x_2, \ldots, x_p, y, \beta_2^1(k(x_1, x_2, \ldots, x_p, y)), \beta_2^2(k(x_1, x_2, \ldots, x_p, y)))).
\end{aligned}$$

Thus the function k is primitive recursive; hence $f = \beta_2^1 \circ k$ and $f' = \beta_2^2 \circ k$ are as well.

5.2 Recursive functions

5.2.1 Ackerman's function

Our aim in this subsection is to give an example of a function that is effectively computable, in the intuitive sense of the word, but that is not primitive recursive. This will justify all the extra work that we will demand of the reader in the future. We define a function (which we call **Ackerman's function** even though it is in fact a slight variant of the one Ackerman defined originally) of two variables that we will denote by ξ as follows:

(i) for every integer x, $\xi(0, x) = 2^x$;

(ii) for every integer y, $\xi(y, 0) = 1$;

(iii) for all integers x and y, $\xi(y+1, x+1) = \xi(y, \xi(y+1, x))$.

For each integer n, let ξ_n denote the function $\lambda x.\xi(n, x)$. Then $\xi_0(x) = 2^x$ and, by invoking clause (iii), it is easy to show that for all positive n, ξ_n is defined by recursion from ξ_{n-1} by

$$\xi_n(0) = 1 \quad \text{and} \quad \xi_n(x+1) = \xi_{n-1}(\xi_n(x)).$$

This shows, first of all, that there is a unique function ξ satisfying the given conditions. Moreover, all the functions ξ_n are primitive recursive (this is proved by induction on n). On the contrary, nothing permits us to affirm that the function ξ

itself is primitive recursive; this is fortunate since we are about to show that it is not. However, we can effectively compute $\xi(x, y)$ for any values of x and y, as the reader should easily be convinced. We must next prove a few easy but annoying lemmas concerning the function ξ.

Lemma 5.6 *For every n and for every x, $\xi_n(x) > x$.*

Proof Our proof will involve two interleaved inductions. By induction on n, we will show that for all x, $\xi_n(x) > x$. This is clear for $n = 0$. Now fix an $n > 0$ and assume that the assertion

$$\text{for every integer } x, \quad \xi_{n-1}(x) > x$$

is true. We must then prove the assertion

$$\text{for every integer } x, \quad \xi_n(x) > x.$$

To do this, we will now argue by induction on x. For $x = 0$, this is clear since $\xi_n(0) = 1$. Next, assuming that $\xi_n(x) > x$, we will prove that $\xi_n(x+1) > x+1$. We know that $\xi_n(x+1) = \xi_{n-1}(\xi_n(x))$ and so, by the first induction hypothesis, we see that

$$\xi_n(x+1) > \xi_n(x), \text{ or, equivalently, } \xi_n(x+1) \geq \xi_n(x) + 1.$$

Now, according to the second induction hypothesis, $\xi_n(x) > x$; so the lemma is proved. ∎

Lemma 5.7 *For every integer n, the function ξ_n is strictly increasing.*

Proof This is clear for $n = 0$. For positive n, it follows immediately from the previous lemma and from the formula $\xi_n(x+1) = \xi_{n-1}(\xi_n(x))$. ∎

Lemma 5.8 *For all $n \geq 1$ and for all x, $\xi_n(x) \geq \xi_{n-1}(x)$.*

Proof This is clear for $x = 0$. For $x + 1$, since $\xi_n(x) \geq x + 1$ and since ξ_{n-1} is increasing, $\xi_{n-1}(\xi_n(x)) \geq \xi_{n-1}(x+1)$; it now suffices to apply the formula

$$\xi_n(x+1) = \xi_{n-1}(\xi_n(x)).$$ ∎

If k is an integer, let ξ_n^k denote the function ξ_n iterated k times (i.e. $\xi_n^0 = \lambda x.x$, $\xi_n^1 = \xi_n$, and $\xi_n^{k+1} = \xi_n \circ \xi_n^k$). The following lemma is now a collection of trivialities.

Lemma 5.9 *The functions ξ_n^k are all strictly increasing. Moreover, for all m, n, k, and x,*

$$\xi_n^k(x) < \xi_n^{k+1}(x), \qquad \xi_n^k(x) \geq x, \qquad \xi_n^k \circ \xi_n^h = \xi_n^{k+h}$$

and, if $m \leq n$, then $\xi_m^k(x) \leq \xi_n^k(x)$.

Next, let us give a definition.

Definition 5.10 *Suppose that $f \in \mathcal{F}_1$ and $g \in \mathcal{F}_p$. We say that f **dominates** g if there exists an integer A such that, for all $(x_1, x_2, \ldots, x_p)$,*

$$g(x_1, x_2, \ldots, x_p) \leq f(\sup(x_1, x_2, \ldots, x_p, A)).$$

In particular, when f is strictly increasing, f dominates g if and only if $g(x_1, x_2, \ldots, x_p) \leq f(\sup(x_1, x_2, \ldots, x_p))$ holds for all but finitely many p-tuples $(x_1, x_2, \ldots, x_p)$.

Let C_n denote the set of functions that are dominated by at least one iterate of ξ_n:

$$C_n = \{g : \text{there exists a } k \text{ such that } \xi_n^k \text{ dominates } g\}.$$

It is obvious that the following functions belong to C_0: the projection functions P_n^i, the constant functions, the successor function S, the function

$$\lambda x_1 x_2 \ldots x_p . \sup(x_1, x_2, \ldots, x_p),$$

the function $\lambda xy.x + y$, and the functions $\lambda x.kx$ where k is an arbitrary integer. Also, the function ξ_n belongs to C_n. Finally, if f and g both belong to $\mathcal{F}_p$, if $g \in C_n$, and if for all $x_1, x_2, \ldots, x_p$, $f(x_1, x_2, \ldots, x_p) \leq g(x_1, x_2, \ldots, x_p)$, then $f \in C_n$.

We will now establish

Lemma 5.11 *For every integer n, the set C_n is closed under composition.*

Proof Let $f_1, f_2, \ldots, f_m$ be functions of p variables and let g be a function of m variables and suppose all these functions are in C_n. We need to prove that $g(f_1, f_2, \ldots, f_m)$ is also in C_n. We know that there exist integers $A, A_1, A_2, \ldots, A_m, k, k_1, k_2, \ldots, k_m$ such that, for all $y_1, y_2, \ldots, y_m$,

$$g(y_1, y_2, \ldots, y_m) \leq \xi_n^k(\sup(y_1, y_2, \ldots, y_m, A)),$$

and for all $x_1, x_2, \ldots, x_p$ and for all i between 1 and m inclusive,

$$f_i(x_1, x_2, \ldots, x_p) \leq \xi_n^{k_i}(\sup(x_1, x_2, \ldots, x_p, A_i)).$$

Set $B = \sup(A, A_1, A_2, \ldots, A_m)$ and $h = \sup(k_1, k_2, \ldots, k_m)$. By invoking Lemma 5.9, we can now see that, for all $x_1, x_2, \ldots, x_p$,

$$\begin{aligned} &g(f_1(x_1, x_2, \ldots, x_p), f_2(x_1, x_2, \ldots, x_p), \ldots, f_m(x_1, x_2, \ldots, x_p)) \\ &\quad \leq \xi_n^k(\xi_n^h(\sup(x_1, x_2, \ldots, x_p, B))), \end{aligned}$$

and hence that

$$g(f_1(x_1, x_2, \dots, x_p), f_2(x_1, x_2, \dots, x_p), \dots, f_m(x_1, x_2, \dots, x)).$$
$$\leq \xi_n^{k+h}(\sup(x_1, x_2, \dots, x_p, B)). \qquad \blacksquare$$

Lemma 5.12 *For all integers n, k, and x,*

$$\xi_n^k(x) \leq \xi_{n+1}(x + k).$$

Proof The proof is by induction on k. For k equal to 0 or 1, it is obvious. Assume it is true for k; then it is also true for $k + 1$ because

$$\begin{aligned} \xi_n^{k+1}(x) &= \xi_n(\xi_n^k(x)) \\ &\leq \xi_n(\xi_{n+1}(x + k)) \quad \text{(by the induction hypothesis)} \\ &= \xi_{n+1}(x + k + 1) \quad \text{(by the definition of } \xi\text{).} \end{aligned} \qquad \blacksquare$$

Lemma 5.13 *Suppose that $g \in \mathcal{F}_p$, that $h \in \mathcal{F}_{p+2}$ and that g and h both belong to C_n ($n \geq 0$). Then the function f defined by recursion from g and h belongs to C_{n+1}.*

Proof We begin by translating the hypotheses. First, the definition of f:

$$\begin{aligned} f(x_1, x_2, \dots, x_p, 0) &= g(x_1, x_2, \dots, x_p), \\ f(x_1, x_2, \dots x_p, y + 1) &= h(x_1, x_2, \dots, x_p, y, f(x_1, x_2, \dots, x_p, y)); \end{aligned}$$

next, the domination conditions:

there exist A_1, A_2, k_1, k_2 such that, for all $x_1, x_2, \dots, x_p, y$,
$g(x_1, x_2, \dots, x_p) \leq \xi_n^{k_1}(\sup(x_1, x_2, \dots, x_p, A_1))$ and
$h(x_1, x_2, \dots, x_p, y, z) \leq \xi_n^{k_2}(\sup(x_1, x_2, \dots, x_p, y, z, A_2))$.

We will now prove by induction on y that, for all $x_1, x_2, \dots, x_p, y$,

$$f(x_1, x_2, \dots, x_p, y) \leq \xi_n^{k_1 + yk_2}(\sup(x_1, x_2, \dots, x_p, y, A_1, A_2)). \qquad (*)$$

For $y = 0$, this is clear. If it is true for y, then it is also true for $y + 1$ because

$$f(x_1, x_2, \dots, x_p, y + 1) = h(x_1, x_2, \dots, x_p, y, f(x_1, x_2, \dots, x_p, y));$$
$$f(x_1, x_2, \dots, x_p, y + 1) \leq \xi_n^{k_2}(\sup(x_1, x_2, \dots, x_p, y, f(x_1, x_2, \dots, x_p, y), A_2)).$$

So, using the induction hypothesis ($*$) and Lemma 5.9,

$$f(x_1, x_2, \dots, x_p, y + 1) \leq \xi_n^{k_2}(\xi_n^{k_1 + yk_2}(\sup(x_1, x_2, \dots, x_p, y, A_1, A_2))),$$

which proves the assertion. Now, we invoke Lemma 5.12 to get

$$f(x_1, x_2, \ldots, x_p, y) \leq \xi_{n+1}(\sup(x_1, x_2, \ldots, x_p, y, A_1, A_2) + k_1 + yk_2).$$

Note that the function

$$\lambda x_1 x_2 \ldots x_p y.\xi_{n+1}(\sup(x_1, x_2, \ldots, x_p, y, A_1, A_2) + k_1 + yk_2)$$

is obtained by composition from functions belonging to C_{n+1}; so it too belongs to C_{n+1} and so does f. ■

We are now in a position to assert:

Corollary 5.14 *The set $\bigcup_{n \in \mathbb{N}} C_n$ contains all primitive recursive functions.*

Proof Indeed, this set contains the constant functions, the projections, and the successor function; also, it is closed under composition, and under definitions by recursion. ■

This brings us to the main theorem of this subsection.

Theorem 5.15 *Ackerman's function is not primitive recursive.*

Proof Suppose, to the contrary, that Ackerman's function is primitive recursive; then so is the function $\lambda x.\xi(x, 2x)$. So, there exist integers n, k, and A such that for all $x > A$, $\xi(x, 2x) \leq \xi_n^k(x)$. Thus, for all $x > A$, we have

$$\xi(x, 2x) \leq \xi_n^k(x) \leq \xi_{n+1}(x + k)$$

(by Lemma 5.12), and, if $x > \sup(A, k, n + 1)$,

$$\xi_{n+1}(x + k) < \xi_{n+1}(2x) < \xi_x(2x) = \xi(x, 2x)$$

(by Lemma 5.9), which is absurd. ■

In fact, we can see that the function $\lambda x.\xi(x, x)$ dominates all the primitive recursive functions.

5.2.2 The μ-operator and the partial recursive functions

We must therefore define a larger class which we will call the class of recursive functions. We will accomplish this by allowing a new definition scheme, the unbounded μ-operator. The idea is as follows: given a subset A of $\mathbb{N}^{p+1}$, this scheme permits us to define the function $f \in \mathcal{F}_p$ which, with the p-tuple $(x_1, x_2, \ldots, x_p)$, associates the least integer z such that $(x_1, x_2, \ldots, x_p, z) \in A$. The problem with this is immediately apparent: what happens if there does not exist an integer z such that $(x_1, x_2, \ldots, x_p) \in A$? Observe that it is not possible in this situation to do what we did for the bounded μ-operator and simply set $f(x_1, x_2, \ldots, x_p) = 0$. Indeed, assuming, as we must, that we have an algorithm at our disposal which computes the characteristic function χ_A of A, the only way we can imagine for computing $f(x_1, x_2, \ldots, x_p)$ is to calculate $\chi_A(x_1, x_2, \ldots, x_p, 0)$. If the result is 1, we may stop; if not, then calculate $\chi_A(x_1, x_2, \ldots, x_p, 1)$, then $\chi_A(x_1, x_2, \ldots, x_p, 2)$,