Chapter 8
Definable Elements and Constants

Abstract In mathematics, or at least in the mathematics inspired by logical methods, to know a structure means to know all sets that are definable in it. In this chapter we will take a look at the smallest nonempty sets—those that have only one element. This a specialized topic, and it is technical, but it will give us an opportunity to see in detail what domains of mathematical structures are made of and in what sense they are "given to us."

Keywords Definable elements · Constants · The theory of a structure · Definable real numbers · Parametric definability

8.1 Definable Elements

In practice, mathematicians do not scrutinize elements of their structures in such a meticulous way as we will do in this chapter. Asking each time about the exact nature of elements would be counterproductive. Instead, new elements are introduced in definitions that are based on previously defined concepts. It can be a long process at the end of which it is easy to lose track of all the steps taken on the way. In our discussion of structures, we will adopt an even more radical approach. We will not be asking what mathematical objects are made of. We will accept that arbitrary sets as somehow given to us, and then, all other structure will gain concreteness, as they will be built, or at least represented, as sets.

We say that an element a of a structure is *definable*, if the set $\{a\}$, is definable. According to Definition 7.2, a set $\{a\}$ is definable, if there is a formula $\varphi(x)$ in the first-order language for the structure such that a is the only element of the domain that has the property expressed by $\varphi(x)$. Hence, the definable elements of a structure are those that can be identified by a first-order property. What can we say about individual elements of a structure?

R. Kossak, *Mathematical Logic*, Springer Graduate Texts in Philosophy 3,
https://doi.org/10.1007/978-3-319-97298-5_8

In our context everything is a set.[1] What are elements of sets then? They are also sets. So what are elements of those sets? Sets as well. Can it go on like this forever; are there sets in sets, in sets, in sets, and so on? No. One of the axioms of ZF—the *foundation axiom*—implies that there are no infinite sequences $x_1, x_2, x_3, \ldots$, such that x_2 is a member of x_1, x_3 is a member of x_2, and so on. The axiom of *foundation* implies that there are no such sequences, although there are axiom systems that allow them. Think of a set as an envelope that contains its elements. We open such an envelope, and look inside. Since everything is a set, unless the set is empty, inside there are sets. Think of those other sets as envelopes as well. If there are nonempty ones, let us take one and look inside. The foundation axiom assures us that this cannot go on forever. Proceeding in this way, we always have to reach a bottom after finitely many steps. So what are those elements (envelopes) at the bottom? They cannot have any elements, so they are empty. But there is only one empty set,[2] and this is what we find there at the bottom of each set. In set theory, the way it is designed, everything is built up from the empty set. No structure, other than the empty set, whose existence is declared by the axiom of the empty set, is assumed to exist a priori. Everything has to be created according to what the axioms are allowing us to create.[3]

The whole discussion above explains how structures are built in set theory. The process begins with the empty set, and laboriously other larger sets are constructed, and once a set is built, then by selecting a set of relations on it, a structure is made. Each element of a structure thus built has a structure of its own, but for our considerations now, the set-theoretic structure of each element must be forgotten. In analyzing a structure, we do not need to know what the elements of the domain are made of. All we are interested in are the properties of those elements with respect to one another, and the only properties that matter initially are those that can be directly expressed in terms of the relations of the structure. This is not as unnatural as it may seem at first glance. If you want learn about the structure of a city, first you identify the elements you want to pay attention to: streets, buildings, bridges, parks. We do not need to know what the buildings are made of, or whether the trees in a park have been recently trimmed. We just need a map showing how the elements spatially relate to one another. It would obfuscate the picture if one also wanted to include architectural plans for all buildings, or a complete book catalog of all libraries.

In our approach to structures, initially one treats all elements of a structure as equally significant, but one of first goals of the logical analysis is to recognize those elements that are distinguished by their first-order properties. In the next section we will examine some specific examples to see how it is done, but first we need another digression on how structures are given to us in the context of set-theory.

---

[1] In more advanced presentations of mathematical logic, even the syntactic objects are sets. Instead of thinking of formulas as marks on paper, we define them as certain sequences of sets.

[2] This is where the envelope metaphor brakes down. Clearly, there are many empty envelopes, but is follows from the axiom of extensionality that there is only one empty set.

[3] This is not quite accurate. The axiom of infinity also declares the existence of a set, but, as we have seen, the axiom postulates the existence of a set that contains infinitely many elements that are constructed from the empty set in a particular manner (the set-theoretic natural numbers).

8.2 Databases, Oracles, and the Theory of a Structure

How can we know anything about infinite structures at all? It is not a trivial matter. In this section, we will examine just one, not very precise metaphor hoping that it will shed light on the idiosyncrasies of mathematical thinking.

We defined the addition of real numbers as a relation. It is the set of all triples (a, b, c) such that $a + b = c$. We can think of this set as a database. If we want to know what is $a + b$, we do not need to perform any operations. In most cases it is not even possible. If a and b are real numbers with infinite non-repeating decimal representations, we do not even know how to begin. Instead of computing, one can think of searching the database until we find a triple (a, b, c) is in it. We can imagine doing this, but there is a problem. We tend to think of all real numbers as somehow given to us, but this is an illusion. They are infinite objects; there is very little we can actually do with them in an effective manner. So instead of "searching the database," let us think of an oracle, an infinite mind that can directly see all relations of a structure. Instead of searching, we will be asking the oracle. We could ask it to find the number c such that $a + b = c$. Since the oracle sees the whole database, it will instantly see the answer. But there is another problem: Can we even ask our question? Most real numbers are infinite objects, either as infinite strings of digits, or as Dedekind cuts. How do we inform the oracle what numbers we have in mind? We do have names for some special irrational numbers, such as $\sqrt{2}$, π, or e, but those names are meant for communication between us. Those numbers are defined in special ways, and for each such number there are computational algorithms giving us the digits of their decimal expansions. Look up π in Google (search "pi"). You will see pages and pages of decimal digits. Some webpages are live, they compute digit after digit in real time. To communicate with the oracle, we would need a mechanism for transmitting information with infinite content. There is no such mechanism. In general, we cannot ask direct questions about particular elements of the domain. In this sense it was not quite right, as we claimed earlier, to think of relations on the structure as given to us directly. They are not. They are set-theoretic objects. How can we know anything about them? This question has a complicated answer.

Most structures we are interested in are constructed for a particular purpose, and since we know how they are constructed, we know something about them. Often, but not always, what we know about a structure is first-order expressible. But here is another set-theoretic twist. Instead of considering particular first-order properties, we can make an audacious move, and also think of the totality of all first-order statements that are true about a structure. Let $\mathfrak{A}$ be a structure. The *theory* of $\mathfrak{A}$, denoted $\mathrm{Th}(\mathfrak{A})$, is the set of all first-order sentences true about $\mathfrak{A}$. It is the complete collection of all properties of $\mathfrak{A}$ that can be expressed in the first-order language with the symbols for all relations of $\mathfrak{A}$. Some structures, such as $(\mathbb{N}, +, \cdot)$, have immensely complicated theories, and we only know small fragments of them. For other, such as $(\mathbb{Q}, <)$, we have complete descriptions. In any case, whether we know the theory of a given structure or not, it is a well-defined set. It exists as a set-theoretic object, and we can think of it as a truth oracle for the structure.

Although first-order logic has not been introduced for this purpose, it turns out to be a perfect formalism for queries about databases. We will not go into details, but it is not difficult to convince oneself that any first order statement in the language of a structure can be interpreted as a database query. For example, if R is a binary relation symbol and R is a binary relation on a domain A, then, when given the sentence $\forall x \exists y R(x, y)$, the oracle looks at all a in A, and if for each such a she can find a b such that $(a, b) \in \mathsf{R}$ she can declare the sentence to be true, and false otherwise. Now, if we want to learn something about that structure we can ask any question we want, but if they are in the form of first-order sentences, then we understand well how a yes/no answer can be obtained by querying the database. We know the process.

If our goal is to understand a given structure, how do we know what first-order questions to ask? If we know nothing about the structure, then there is not much we can do. We need to know something in advance, based on how the structure was built and how its domain and its relations are defined. In such cases, it is often not that difficult to identify at least some definable sets. Usually one begins with identifying some special elements of the domain, and uses those to identify specific definable sets. For example, in all number structures that we discussed, 0 and 1 are of special importance, and they are definable (we will see how in the next section). From 1 one can define all integers, from those, solution sets to algebraic equations with integer coefficients.

To show that an element of the domain of a structure is definable, one has to find its definition, or to prove that such a definition exists. This is usually not a very hard task. It is much more difficult to prove that an element is not definable. We will need special tools to do that, and they will be described in Chap. 9. Now we will take a closer look at definability of real numbers.

8.3 Defining Real Numbers

In general, if $\varphi(x)$ is a formula in the first-order language for a given structure, the sentence $\exists x \varphi(x)$ is true in the structure, and $\exists x \exists y[\neg(x = y) \wedge \varphi(x) \wedge \varphi(y)]$ is false, it means that there is only one element in the domain that has the property expressed by $\varphi(x)$; this element is defined by the property $\varphi(x)$.

Recall that $\mathfrak{R}$ denotes the field of real numbers $(\mathbb{R}, +, \cdot)$. Consider the formula $x + x = x$. The only real number with that property is 0.[4] Hence, the formula $x + x = x$ defines 0 in $\mathfrak{R}$. The numeral 0 is not a symbol of the language of $\mathfrak{R}$, but since 0 is definable, we can introduce a formal symbol for it, and use it in first-order statements, because the added symbol can always be eliminated. For example, the

[4]Since we are dealing with an infinite domain, one could still ask how we know that. Our understanding of how the number system is constructed helps here. We do understand addition in $\mathfrak{R}$ well enough to know that much.

formula

$$\neg(x = 0) \wedge x \cdot x = x \qquad (*)$$

can be translated into

$$\neg(x + x = x) \wedge x \cdot x = x.$$

The only number that has the property expressed by formula (∗) is 1, so 1 is also definable. Once we have 1, we can define other natural numbers. For example, 2 is the only number x such that $1 + 1 = x$, hence 2 is definable. If a is a definable number, then $-a$ is definable as well as the only number x such that $x + a = 0$. This shows that all integers are definable. Rational numbers are also definable; $1/2$ is the only x such that $2 \cdot x = 1$, $2/3$ is the only x such that $3 \cdot x = 2$, and, in general, m/n is defined by $m = \cdot n$. Since all rational numbers are definable, in first-order formulas of the language of $\mathfrak{R}$ we can use names for all those numbers.

Every rational number is definable in $\mathfrak{R}$, but this does not imply that set of rational numbers is definable in $\mathfrak{R}$. Each rational number has its own individual definition, but those definitions cannot be combined to define $\mathbb{Q}$ as a set. We cannot combine those definitions because there are infinitely many of them. This does not rule out the possibility that there may be some other definition that works, but we do know that there can be no such definitions. This is a consequence of the already mentioned Alfred Tarski's analysis of definability in $\mathfrak{R}$.

An example of a number that is not definable is π, but it is not easy to prove.[5]

The ordering, i.e. the "less than" relation, is crucial for understanding the structure of the real numbers. We did not include a symbol for it in the language for $\mathfrak{R}$. That is because the ordering of real numbers is already definable in $\mathfrak{R}$. It is logically visible. The formula that defines the relation "x is less than y" is

$$\exists z[\neg(z = 0) \wedge x + z \cdot z = y].$$

Why does it work? If x and y are real numbers, and x is less than y, then the difference $y - x$ is a positive real number. Since every positive real number has a square root, the difference $y - x$ has a square root z, and for this z, $x + z \cdot z = y$. Since the ordering of the real numbers is definable, we can introduce the "less than" symbol $<$ and use it freely when expressing first-order facts about $\mathfrak{R}$.

In the argument above, we used subtraction $y - x$, but there is no symbol in the language of $\mathfrak{R}$ for it. There is also no symbol for division. The reason is that both these operations can be defined in terms of $+$ and $\cdot$. Indeed, for all numbers x, y, and z

$$y - x = z \Longleftrightarrow z + x = y$$

[5]It follows from a theorem proved in 1882 by Ferdinand von Lindemann (1852–1939). The theorem states that π is not a solution of any polynomial equation with integer coefficients (it is *transcendental*).

and

$$y/x = z \Longleftrightarrow x \cdot z = y.$$

This all works well, with the exception of $x = 0$ and $y = 0$ in the second formula. If $x = 0$ and $y = 0$ were allowed, then $0/0 = z$ would be true for any number z. Notice that if x is zero and y is not, then the definition is still fine, because then $0 \cdot z = y$ for no number z, which means that division by zero is undefined.

With the aid of the ordering, we can now define more real numbers. For example, there are exactly two numbers x such that $x \cdot x = 2$. They are $\sqrt{2}$ and $-\sqrt{2}$. The formula $x \cdot x = 2$ defines the set $\{\sqrt{2}, -\sqrt{2}\}$, so it does not identify any of these two numbers uniquely, but the formula $x > 0 \wedge x \cdot x = 2$ defines $\sqrt{2}$, and $0 > x \wedge x \cdot x = 2$ defines $-\sqrt{2}$.

In a similar fashion, using algebraic equations, one can define many other real numbers, but one cannot define them all. The set of all first-order formulas is infinite, but small. It can be built in a step-by-step process. It is *countable*. The set of all real numbers is *uncountable*, it cannot be built step-by-step. It follows that not all numbers can have first-order definitions.

Talking about $\mathfrak{R}$ in a first-order way, we are allowed to use quantifiers that refer to the totality of all numbers, but, unless they are definable, we cannot refer to particular numbers. That does not mean that we cannot talk about undefinable numbers in a first-order way at all. In fact, we can, and we will, but to do it, we need to expand the language. For example, to talk about π, we can add to the language a new relation symbol $P(x)$, and interpret it as the set $\{\pi\}$. This gives us a new structure $\mathfrak{R}_\pi = (\mathbb{R}, +, \times, \{\pi\})$. π is not definable in $\mathfrak{R}$, but it is in $\mathfrak{R}_\pi$. A formula that defines it is $P(x)$. We can do more. For each real number r, we can add a new relation symbol $P_r(x)$ and interpret it in the set of real numbers as the set $\{r\}$. So now we have a new structure in which all real numbers are definable. It is a genuine first-order structure consisting of the set of real numbers, relations of addition and multiplication, and a huge infinite set of relations $\{r\}$, one for each real number r.

8.4 Definability With and Without Parameters

In everyday mathematical practice, one freely refers to arbitrary elements in structures. To do it in the first-order formalism, we expand the language of the structure so that we can use names for elements. In the previous section, we saw how it is done in the case of the field of real numbers. Let us see how it is done in general.

If an element a of the domain of a structure $\mathfrak{A}$ has a property that is expressible by a first-order formula $\psi(x)$, then we are tempted to say that $\psi(a)$ holds in $\mathfrak{A}$. The problem is that a is not a symbol in the language, and therefore $\psi(a)$ is not an expression of first-order logic. It mixes the two worlds: the syntactic and the semantic, and this is a serious violation. One could say; why don't we just add a symbol to the language to represent the element, and allow formal expressions involving the symbol? There is no problem if a is definable in $\mathfrak{A}$. Suppose that a is the only element of $\mathfrak{A}$ which has a first-order property $\varphi(x)$. If $\psi(x)$ is a formula of the language of the structure, then since a is the only element of the structure having the property $\varphi(x)$, the statement $\forall x[\varphi(x) \implies \psi(x)]$ expresses that a has the property $\psi(x)$. We can now introduce a new symbol s_a, and treat expressions such as $\psi(s_a)$ as abbreviations of $\forall x[\varphi(x) \implies \psi(x)]$. This way we can formally talk about definable elements without changing the formalism. We can do something similar for all elements of the domain, even if they are not definable.

For each element a of the domain of a structure $\mathfrak{A}$, we can add a new relation symbol $U_a(x)$ and interpret it in $\mathfrak{A}$ as the set $\{a\}$. Then $\mathfrak{A}$ expanded by adding all those new relations is a new structure in which each element is now definable. Indeed, each element a is defined by the formula $U_a(x)$. Then, to simplify notation, for each element a, we can introduce a constant symbol s_a, and treat expressions $\psi(s_a)$ as abbreviations of $\forall x[U_a(x) \implies \psi(x)]$. In this way, we can turn every structure $\mathfrak{A}$ into an expanded structure, which we will denote by $\overline{\mathfrak{A}}$, considered as a structure for the expanded language in which each element has a definition.

What is this formal fuss all about? In standard presentations of first-order logic, in addition to relation symbols, the syntax usually includes function symbols and constants. None of it is absolutely necessary, since functions can be considered as relations of a special type, and constants, as we have seen, can be treated as unary relations. For the presentation here, I have decided to use Occam's razor to reduce the formal apparatus to the minimum. In mathematical practice such frugality is not needed. In fact, one often considers structures in the first-order language including constants for all elements. The elements are treated as *parameters* and used freely in mathematical formulas. For structures whose domains are well-understood, such as the natural numbers or the rational numbers, it is not a problem at all, since for all elements we can use their standard representations as formal names. The approach is more problematic when applied to larger and more complex domains, such as the real numbers $\mathbb{R}$. Since the real numbers are uncountable, the set of their names has to be uncountable as well. This means that we can no longer claim that the set of all formulas of the extended language can be built in a step-by-step process. The set of formal expressions itself becomes a set-theoretic object, and it has to be investigated by set-theoretic means. There is no problem with doing all of that mathematically—modern mathematics is firmly based in set-theory—but notice that the distinction between syntax and semantics gets blurry, and we no longer stand on the firm (or firmer) ground, where the syntax is much simpler and better understood, than the structures it serves to describe. Another technical problem that has to be resolved is that if for a large domain we need a large set of constants, then where do those constants come from? Think of possible names for real numbers. Real numbers are represented by arbitrary Dedekind cuts, or finite or infinite sequences of digits. There is no rule or pattern for creating symbolic names for such objects, but there is an elegant way out. We can just declare that each element of a structure stands for its own name. After all, we never said what the symbols of first-order logic are supposed to be. Since there is no formal restriction, they can in fact be any kind of objects we like, so we can use elements of structures themselves. Why not? Technically, there is no problem, but the syntax vs. semantics division line gets even more difficult to demarcate.

In the next chapter, we will discuss some notions that are used to classify structures. They involve definability in languages that include names for all elements. Therefore let us finish this chapter with a formal definition. Compare it with Definition 7.2.

Definition 8.1 Let $\mathfrak{A}$ be a structure with domain A. For each natural number $n > 0$, a subset X of the Cartesian power A^n is called *parametrically definable* in $\mathfrak{A}$, if it is definable in the expanded structure $\overline{\mathfrak{A}}$.

Exercises

Exercise 8.1 *Write first-order formulas defining* 3, $\frac{1}{3}$, *and* $\frac{2}{3}$ *in* $\mathfrak{R}$, *using only symbols* $+$ *and* $\cdot$. *Hint:* $\frac{2}{3}$ *is a solution of the equation* $3x = 2$.

Exercise 8.2 *Write a first-order formula defining* $-\sqrt{2}$ *in* $\mathfrak{R}$, *using only symbols* $+$, $\cdot$, $<$, *and* 2. *Hint:* $\sqrt{2}$ *and* $-\sqrt{2}$ *are the only solutions of the equation* $x^2 = 2$.

Exercise 8.3 *Write a first-order formula defining* $\sqrt{2}$ *in* $\mathfrak{R}$, *using only symbols* $+$ *and* $\cdot$ *(no* $<$*). Hint: Recall that the ordering of the real numbers can be defined in* $(\mathbb{R}, +, \cdot)$.

Exercise 8.4 *Show that every element of the ordered set* $(\mathbb{N}, <)$ *is definable. Hint:* 0 *is the least element of* $\mathbb{N}$, 1 *is the least element of* $\mathbb{N}$ *that is greater than* 0, 2 *is the lest element of* $\mathbb{N}$ *that is greater than* 1, *and so on.*

Exercise 8.5 *Let* $\mathfrak{A}$ *and be a structure with domain* A. *Show that every finite subset of* A *is definable in the expanded structure* $\overline{\mathfrak{A}}$ *(see Definition 8.1).*