Exponentiation is elementarily definable from addition and multiplication on finite structures

Steven Lindell
Department of Computer Science
Haverford College
Haverford, PA 19041-1392
slindell@haverford.edu

Abstract: *We explain how to define powering from plus and times in first-order logic on finite structures.*

Introduction

Many people have asked me this question at one time or another, so I have provided a sketch of elementarily defining exponentiation from addition and multiplication.

The construction outlined below is not very difficult, quite clever, and not mine. My explanation is based on the discussion starting on page 294 in [H&P] which Sam Buss pointed me to. Ken Regan had suggested looking at [S] where, in addition to the great discussions of mathematical history etc., he includes a discussion of Pell's equation to get an exponential Diophantine relation, which can then presumably be used to define ^ from + and * in a first-order way. I did not end up following this rather lengthy, though quite interesting treatment, ending my search with a more direct proof which the first reference provides. However, I highly recommend Smorynski for his lucid interesting style (throughout the book). It is one of the most enjoyable academic texts I have looked at in a long time (i.e. even if you skip the mathematics, it is worth reading).

Construction

Here is the construction, sketched roughly. Some details are omitted. My acknowledgements go to Scott Weinstein for mentioning a simplification which avoids the use of the pairing function and its attendant reduction in the size of numbers which can be quantified over. However, Smorynski's discussion of the Cantor pairing function and its unicity over real polynomials is quite fascinating (section 1.3. p.14-). Hájek & Pudlák use pairing.

We want to define the relation of exponentiation $a^b = c$ for $0 \leq a, b, c < n$ in a first-order way. I will ignore the easy cases where $a = 0, 1$. Our intent will be to construct a quickly growing sequence of powers based on the binary representation of b (an old trick in computer arithmetic).

Let $b = (b_1 \ldots b_m)$ in base 2, each b_i either 0 or 1. Applying Horner's method for polynomial evaluation yields the formula:

$$b = 2*(\ldots(2*(2*(b_1) + b_2) + b_3) + \ldots) + b_m$$

which can be computed by the iterative algorithm:

$b := 0$	$b_0 = 0$
FOR $i = 1$ **TO** m	
$\quad b := 2*b + b[i]$	$2*(\ldots) + b_i$

Using this idea we can define a sequence $1 = c_0 \ldots c_m = c$ of powers reaching a^b more quickly:

$c := 1$	$c_0 = 1$
FOR $i = 1$ **TO** m	
$\quad c := c * c$	$c_{i+1} = c_i^2 \ldots$
$\quad$ **IF** $b[i]$ **THEN** $c := c * a$	$* \; a^{b_i}$

At each stage, we square the previous number, multiplying by the base a when necessary. The correctness of this method follows easily from the formula for b when it appears in the exponent.

The major challenge is to define a pair of sequences we can quantify over: one to calculate b; the other to calculate c (according to the above iterations). In particular, we define a triple of numbers u, v and w. Although we do not have a way of determining the binary expansion of a number, we can visualize v as the concatenation of the sequence $\langle c_m \ldots c_0 \rangle$ where each c_i is written in binary, and u as the concatenated binary sequence of values resulting from the evaluation of the formula for b. The number w is visualized as a binary sequence whose 1's indicate demarcations which separate the binary components within both u and v. Although u and v are arbitrary numbers, we can insist that w begin and end with a 1. This can be achieved by saying that w is odd, and that it is at least as big as the highest power of two. The predicate "x is a power of two" can be defined by saying that every prime divisor of x is 2. We can visualize this:

$u =$
$v =$ $\quad c_m \;,\; c_{m-1} \;, \ldots ,\; c_0$
$w =$ $\quad \mathbf{1}0\ldots0\mathbf{1}0\ldots0\mathbf{1}\ldots\mathbf{1}0\ldots0\mathbf{1}$

Think of the 1's of w as providing the commas which separate element strings of both sequences.

Without the ability to address the i^{th} element of either sequence (which would be tantamount to having 2^i), we must instead quantify over adjacent elements. We start to do this by defining what it means for two ones (i.e. their positions) in w to be *adjacent* (i.e. ignoring zeros):

> "$z < z'$ are powers of two such that (w div z) and (w div z') are both odd,
> and there is no power of two in between them with this property."

Here, div is integer division. Note that we are using the property that the powers of two are linearly ordered as a subset of the domain. Now, we can extract the element of the sequence demarcated by these ones in either u or v by using [(u mod z') div z] or [(v mod z') div z] respectively.

To say two elements are adjacent in u or v, we say that there are three powers of two, $z < z' < z''$ which determine adjacent 1's in w. However, the key idea is that we can look at corresponding adjacent elements in both u and v simultaneously. Suppose x and x' are adjacent elements in u, and y and y' are the corresponding adjacent elements of v (determined by the same z, z', z''). State that the first element of u is 0, and the first element of v is 1. Then state for all adjacency pairs:

$$x' = 2x + 0 \mid 1 \quad \text{iff} \quad y' = y^2 * 1 \mid a$$

where the vertical bar | separates the two respective cases. Finally, state that the last element of u is b, and the last element of v is c.

It remains to be checked that v does not exceed the strict bound of n. However, it does by a small amount (n^2). This can be remedied by either shortening the sequence (leaving c_m to be computed later), using two variables (to quantify up to n^2), or just defining ^ for smaller numbers.

References

[H&P] Petr Hájek, Pavel Pudlák, *Metamathematics of First-Order Arithmetic*, Springer-Verlag, 1993.

[L] This manuscript, 1995

[S] Craig Smorynski, *Logical Number Theory I - An Introduction*, Springer-Verlag, 1991.