

Compact Linear Types

John Skaller

November 23, 2022

1 The Ring of Power

1.1 Preliminary definitions

Definition 1 A semi-group is a set together with an associative binary operation; that is

$$(a \cdot b) \cdot c = a \cdot (b \cdot c)$$

where infix \cdot is taken as the symbol for the binary operation.

Associativity is a crucial property for an operator because it allows concurrent evaluation of arbitrary subsequences of a sequence of operands. For example given the operand expression

$$(((1 \cdot 2) \cdot 3) \cdot 4) \cdot 5$$

we can compute $1 \cdot 2$ and $4 \cdot 5$ concurrently:

$$(1 \cdot 2) \cdot 3 \cdot (4 \cdot 5)$$

then then combine 3 to either the LHS or RHS subterm before performing the final combination. Another way of looking at this property is that the values to be combined can be stored in the leaves of a tree and any bottom up visitation algorithm can be used to find the total combination.

Associativity means you can add or remove (balanced pairs of) parentheses freely. In particular it is common practice to leave out the parentheses entirely.

Definition 2 A monoid is a semigroup with a unit u , that is

$$x \cdot u = x \text{ and } u \cdot x = x$$

for all x in the set.

The existence of a unit means you can freely add or remove units from anywhere in your computation.

Definition 3 A group is a monoid in which every element has an inverse, that is, for all x there exists an element y such that

$$x \cdot y = u \text{ and } y \cdot x = u$$

where u is the unit of the underlying monoid. For integers of course, the additive inverse of a value is it's negation.

Definition 4 An operation is commutative if the result is the same with the operands reversed, that is, for all a and b .

$$a \cdot b = b \cdot a$$

A group is said to be commutative if the group operation is commutative.

Commutativity says you can switch the order of children in the tree representation of an expression.

If an operation is also associative, commutative, and has a unit, then the operation is well defined on a set of operands, taking the operation on the empty set to be the unit.

This means irrespective of what data structure you use to hold the values to be combined, and what algorithm you use to scan them, provided you visit each value exactly once, the result of the operation on them is invariant.

Definition 5 A ring is a set with two operations denoted by $+$ and $*$ such that the set with $+$ is a group, and the set excluding the additive unit is a monoid, and the following rule, called the distributive law holds for all a , b and c

$$a * (b + c) = a * b + a * c$$

If the multiplication operation is commutative then it is called a commutative ring.

1.2 The rings \mathbb{N}_n

Definition 6 Let \mathbb{N}_n be the subrange of the integers $0..n - 1$ with addition, subtraction, multiplication, division and remainder defined as the natural result modulo n . Then \mathbb{N}_n is a commutative ring called a natural ring.

The usual linear order is also defined. Negation is defined by

$$-x = n - x$$

Natural computations prior to finding the modular residual present an issue we resolve by performing these computations in a much larger ring.

1.3 Representation

Lemma 1 *The C data types*

`uint8_t uint16_t uint32_t uint64_t`

with C builtin operations for addition, subtraction, negation, and multiplication are the rings \mathbb{N}_{2^8} $\mathbb{N}_{2^{16}}$ $\mathbb{N}_{2^{32}}$ $\mathbb{N}_{2^{64}}$ respectively, with the usual comparison operations, unsigned integer division, and unsigned integer modulus.

Theorem 1 Representation Theorem. *The values of a ring \mathbb{N}_n can be represented by values of a ring \mathbb{N}_{n^2} and the operations addition, subtraction, negation multiplication and modulus computed by the respective operations modulo n . Comparisons work without modification.*

In particular we can use `uint64_t` to represent rings of index up to 2^{32} .

2 Ring Products

Definition 7 *Let R_i for i in \mathbb{N}_n be a tuple of n rings, then the tensor product of the rings, denoted by*

$$R_0 \otimes R_1 \otimes \dots \otimes R_{n-1}$$

is a ring with values tuples of corresponding elements, operations defined componentwise, comparisons defined by the usual lexicographic ordering, and iterators sequencing through values in the defined order.

The size of the ring is the product of the ring sizes.

Definition 8 *A ring is compact linear if it is a natural ring, a product of compact linear rings, or a sum of compact linear rings.*

Theorem 2 Compact Linear Product Representation. *A compact linear product can be represented by a single value $0..N-1$ where N is the product of the sizes of the rings. The encoding of a value $(v_0, v_1, \dots, v_{n-1})$ is given by*

$$v_0 * r_0 + v_1 * r_1 + \dots + r_{n-1} * v_{n-1}$$

where $r_{n-1} = 1$ and r_k for k in $0..n-2$ is the product of the sizes of the rings R_j for $j > k$.

The projection p_k of the k 'th ring is given by

$$v / r_k \mod |R_k|$$

where $|R|$ is the size of the ring R .

3 Ring Sums

Definition 9 Let R_i for i in \mathbb{N}_n be a tuple of n rings, then the sum of the rings, denoted by

$$R_0 \oplus R_1 \oplus \dots \oplus R_{n-1}$$

is a ring with values of one of the rings determined by the injection function that constructed the ring.

Theorem 3 Compact Linear Sum Representation. A compact linear sum can be represented by a single value $0..N - 1$ where N is the sum of the sizes of the rings. The injection of a value v_i of ring R_i into the sum is given by

$$v_i + s_j$$

where $s_{n-1} = 0$ and r_k for k in $0..n - 2$ is the sum of the sizes of the rings R_j for $j > k$.

Decoding the sum is achieved as follows. Starting with $i = 0$, if $v \geq s_i$, then the injection was for the value $v_i = v - s_i$ of the ring R_i , otherwise increment i and try again. Since for $i = n - 1$, the rightmost ring, $v \geq 0$ is true for all possible v , the iteration must terminate.

3.1 Size function

We define the **size** of a type as follows, using the usual $|\cdot|$ notation:

$$\begin{aligned} |\mathbb{N}_n| &= n \\ |\oplus_{i=0}^{n-1} R_i| &= \sum_i |X_i| \\ |\otimes_{i=0}^{n-1} r_i| &= \prod_i |X_i| \end{aligned}$$

This just says the size of a sum is the sum of the sizes of its components, and the size of a product is the product of the sizes of its components. Note that by the convention for nullary terms, the size of the empty sum is 0 whilst the size of the empty product is 1.

4 IGNORE

Theorem 4 Computations involving addition, subtraction, negation and multiplication in a small region close to zero produce the same results as if they were done with the integers.

This means, to perform calculations for a ring \mathbb{N}_n we can use a ring \mathbb{N}_m provided m is sufficiently greater than n : the result holds for division of positive values but fails for negative ones.

We need to check multiplication makes sense for negative values. The requirement is that if we multiply a negative by positive value than take the modular residual, we get the same result as taking the modular residual first then multiplying, and similarly for two negative values.

For the first case when k is negative

$$\begin{aligned} ((k+n) \bmod n)j &= (k+n)j && \text{since } k+n < n \\ &= kj + nj && \text{distributivity} \\ &= kj \bmod n && \text{since } nj \bmod n = 0 \end{aligned}$$

and the second case

$$\begin{aligned} ((k+n) \bmod n)((j+n) \bmod n) &\bmod n \\ &= (k+n)(j+n) \bmod n && \text{since } k+n < n \text{ and } j+n < n \\ &= (kj + kn + jn + n^2) \bmod n \\ &= kj \bmod n \end{aligned}$$

Note the result holds only when the product of the absolute values of j and k are small, in particular the product is less than n .

For example in \mathbb{N}_{16} we have $2 * 3 = (-2) * (-3) = 14 * 13 = 182 \bmod 16 = 6$.

A bound for the naturalness of a multiplication is that the operands of the product are less than \sqrt{n} . If n is a power of 2, this just means the operands have half the number of bits in their representation.

Definition 10

$$\mathbb{N} = \lim_{n \rightarrow \infty} \mathbb{N}_n$$

That is, we define the integers are the limit of a sequence of larger and larger rings. As n gets bigger, a larger region about 0 will behave like the natural numbers.

Let \mathbb{D} be a distributive category with two specified families of functors \sum_n for sums and \prod_n for products indexed by the ring \mathbb{N}_\times where the nullary sum is the empty object *void* also written 0, nullary product is a *unit* object, also written 1, the 1-ary sum of an object X is the identity 1_X viewed as an injection, and the 1-ary product X is the identity 1_X viewed as projection.

Define the *kernel* of the category \mathbb{D} as the full subcategory \mathbb{K} with only the objects generated by combinations of \sum and \prod and let $\sum|_{\mathbb{K}}$ and $\prod|_{\mathbb{K}}$ be the functor families restricted to \mathbb{K} .

Then the kernels of all distributive categories with chosen sum and product functor families are isomorphic. Sum and product formation is neither associative nor symmetric, but they are associative and symmetric up to isomorphism.

5 Theory

With just two type constructors for sums and products, we construct a value representation in the natural numbers. We first define some convenient auxiliary functions with comprehensible semantics, because the essence of the raw formulas is embodied in superscripts and subscript details and the meaning would soon be lost.