Sums and Permutations

Noah Wedlich	12620484

January 22, 2025

Sums and basic properties

Definition of sums

We define ”RealSum” as a program constant:

```
(add-program-constant "RealSum" (py "nat=>nat=>(nat=>rea)=>rea"))
(add-computation-rules
    "RealSum n Zero xs"   "0"
    "RealSum n(Succ m)xs" "RealSum n m xs+xs(n+m)")
```

In classical notation we have:

$$\text{RealSum n m xs} = \sum_{i=n}^{n+m-1} (xs)_i$$

We procede to prove totality of ”RealSum” and introduce a non-normalizing construct ”cRSum” equivalent to ”RealSum”.

Basic properties

We now use the above definition to state some basic facts about finite real sums. We will not write out the proofs in this paper. The entirety of the proofs can be found in the file ”rseq.scm”. We will also omit some special cases for the sake of brevity.

RealSumReal

We have $\forall i : (xs)_i \in \mathbb{R} \implies \sum_{i=n}^{n+m-1} (xs)_i \in \mathbb{R}$:

```
all xs(all n Real(xs n) -> all n,m Real(RealSum n m xs))
```

RealSumZeroSucc

We have:

$$\sum_{i=0}^{m}(xs)_i = \sum_{i=0}^{m-1}(xs)_i + (xs)_m$$

In Minlog this is formulated as:

```
all m,xs(all n Real(xs n) ->
    RealSum Zero(Succ m)xs===RealSum Zero m xs+xs m)
```

RealLeMonSum

Sums are monotone, i.e.

$$\forall l = n, \ldots, n+m-1 : (xs)_i \leq (ys)_i \implies \sum_{i=n}^{n+m-1}(xs)_i \leq \sum_{i=n}^{n+m-1}(ys)_i$$

In Minlog this is formulated as:

```
all xs,ys,m,n(all l(n<=l -> l<n+m -> xs l<<=ys l) ->
    RealSum n m xs<<=RealSum n m ys)
```

Proof in Minlog!

RealSumCompat

Sums are compatible with equality:

$$\forall i : (xs)_i = (ys)_i \implies \sum_{i=n}^{n+m-1}(xs)_i = \sum_{i=n}^{n+m-1}(ys)_i$$

In Minlog this is formulated as:

```
all xs,ys,n,m(all l(n<=l -> l<n+m -> xs l===ys l) ->
    RealSum n m xs===RealSum n m ys)
```

Proof in Minlog!

RealSumShiftUp

We can "shift up" a sum:

$$\forall l \in \mathbb{N} : \sum_{i=n}^{n+m-1} (xs)_{i+l} = \sum_{i=n+l}^{n+l+m-1} (xs)_i$$

In Minlog we have:

```
all xs,l,n( all n Real(xs n) ->
    all m RealSum n m([n0]xs(l+n0))===RealSum(l+n)m xs)
```

Proof in Minlog!

RealSumShiftDown

We can "shift down" a sum:

$$\forall l \leq n : \sum_{i=n}^{n+m-1} (xs)_i = \sum_{i=n-l}^{n-l+m-1} (xs)_{i+l}$$

In Minlog:

```
all xs,l,n(all n Real(xs n) -> l<=n ->
    all m RealSum n m xs===RealSum(n--l)m([n0]xs(l+n0)))
```

RealTimesSumDistr

We can distributes real numbers:

$$\forall x \in \mathbb{R} : x \cdot \sum_{i=n}^{n+m-1} (xs)_i = \sum_{i=n}^{n+m-1} x \cdot (xs)_i$$

In Minlog:

```
all x,xs(Real x -> all n Real(xs n) -> all n,m(
    x*RealSum n m xs===RealSum n m([l]x*xs l)))
```

Analogously we get "RealTimesSumDistrLeft" for a right multiplication.

RealSumSplit

We can split sums in two:

$$\sum_{i=n}^{n+m-1}(xs)_i + \sum_{i=n+m}^{n+m+l-1}(xs)_i = \sum_{i=n}^{n+m+l-1}(xs)_i$$

In Minlog:

```
all xs(all n Real(xs n) -> all n,m,l(
    RealSum n m xs+RealSum(n+m)l xs===RealSum n(m+l)xs))
```

RealSumMinus

We can "trim of" the start of a sum:

$$\sum_{i=n}^{n+m+l-1}(xs)_i - \sum_{i=n}^{n+m-1}(xs)_i = \sum_{i=n+m}^{n+m+l-1}(xs)_i$$

In Minlog:

```
all xs(all n Real(xs n) -> all n,m,l(
    RealSum n(m+l)xs+ ~(RealSum n m xs)===RealSum(n+m)l xs))
```

RealSumOne

A sum of length 1 is just a real number:

$$\sum_{i=n}^{n}(xs)_i = (xs)_n \in \mathbb{R}$$

In Minlog:

```
all xs,n(Real(xs n) -> RealSum n(Succ Zero)xs===xs n)
```

RealSumPlus

Addition of finite sums:

$$\sum_{i=n}^{n+m-1}(xs)_i + \sum_{i=n}^{n+m-1}(ys_i) = \sum_{i=n}^{n+m-1}(xs)_i + (ys_i)$$

In Minlog:

```
all xs,ys(all n Real(xs n) -> all n Real(ys n) -> all m,n(
    RealSum n m xs+RealSum n m ys===RealSum n m([l]xs l+ys l)))
```

Proof in Minlog!

RealSumUMinus

We can invert sums:

$$-\sum_{i=n}^{n+m-1}(xs)_i = \sum_{i=n}^{n+m-1} -(xs)_i$$

In Minlog:

```
all xs(all n Real(xs n) -> all n,m(
    ~(RealSum n m xs)===RealSum n m([l]~(xs l))))
```

RealLeAbsSum

We have the triangle inequality:

$$\left|\sum_{i=n}^{n+m-1}(xs)_i\right| \leq \sum_{i=n}^{n+m-1} |(xs)_i|$$

RealSumZeros

The sum of zeros is again zero:

$$\sum_{i=n}^{n+m-1} 0 = 0$$

In Minlog:

```
all xs(all n xs n===0 -> all n,m RealSum n m xs===0)
```

RealSumZerosLR

We can omit zeros at the end and start:

$$\begin{aligned}\forall i \in &\{0,\ldots,m_1-1\}\cup\{m_1+m_2,\ldots,m_1+m_2+m_3-1\} : (xs)_i = 0\\ &\implies \sum_{n=0}^{m_1+m_2+m_3-1}(xs)_i = \sum_{i=m_1}^{m_2}(xs)_i\end{aligned}$$

In Minlog:

```
all m1,m2,m3,xs(
    all n Real(xs n) ->
        all l(l<m1 -> xs l===0) ->
            all l(m1+m2<=l -> l<m1+m2+m3 -> xs l===0) ->
                RealSum Zero(m1+m2+m3)xs===RealSum m1 m2 xs)
```

RealSumSplitTwo

We can split a sum based on a "mask":

$$\sum_{i=n}^{n+m-1}(xs)_i = \sum_{i=n}^{n+m-1}\delta_{ps}(xs)_i + \sum_{i=n}^{n+m-1}(1-\delta_{ps})(xs)_i$$

Here $ps : \mathbb{N} \to \{\text{True}, \text{False}\}$ is a mask and:

$$\delta_{ps}(n) := \begin{cases} 1 & ps(n) = \text{True} \\ 0 & ps(n) = \text{False} \end{cases}$$

In Minlog we can formulate this as:

```
all xs,ps(
    all n Real(xs n) ->
    all n,m
        RealSum n m xs===
            RealSum n m([l][if (ps l) (xs l) 0])+
            RealSum n m([l][if (ps l) 0 (xs l)]))
```

RatSumRealSum

Rational sums are also real sums:

```
all as,n,m RatSum n m as===RealSum n m as
```

RealSumTimes

Multiplication of sums. This uses root-based coding of pairs of natural numbers:

$$\left(\sum_{i=0}^{n-1}(xs)_i\right)\left(\sum_{i=0}^{n-1}(ys)_i\right) = \sum_{j,k=0}^{n-1}(xs)_j(ys)_k$$

In Minlog we formulate this as follows:

```
all xs,ys(
    all n Real(xs n) -> all n Real(ys n) -> all n
        RealSum Zero n xs*RealSum Zero n ys===
            RealSum Zero(n*n)([k]xs(lft(RtD k))*ys(rht(RtD k))))
```

Proof Sketch (IS):

$$\begin{aligned}
\sum_{i=0}^{n+1} x_i \cdot \sum_{i=0}^{n+1} y_i &= \left(\sum_{i=0}^{n} x_i + x_{n+1}\right)\left(\sum_{i=0}^{n} y_i + y_{n+1}\right) \\
&= \sum_{i=0}^{n} x_i \cdot \sum_{i=0}^{n} y_i + \left(x_{n+1} \cdot \sum_{i=0}^{n} y_i + \sum_{i=0}^{n} x_i \cdot y_{n+1} + x_{n+1} \cdot y_{n+1}\right) \\
&\stackrel{?}{=} \sum_{i,j=0}^{n+1} x_i \cdot y_j \\
&= \sum_{i,j=0}^{n} x_i \cdot y_j + \sum_{\substack{0 \leq i,j \leq n+1 \\ i=n+1 \vee j=n+1}} x_i \cdot y_j
\end{aligned}$$

RealSumDiags

We have the following identity:

$$\sum_{i=0}^{n} \left(\sum j + k = i (xs)_j (ys)_k\right) = \sum_{j+k \leq n} (xs)_j (ys)_k$$

In Minlog this is:

```
all xs,ys(
    all n Real(xs n) ->
    all n Real(ys n) ->
    all n
        RealSum Zero(Succ n)
        ([m]
        RealSum Zero((m+1)*(m+1))
        ([k][if (lft(RtD k)+rht(RtD k)=m)
            (xs(lft(RtD k))*ys(rht(RtD k)))
            0])) ===
        RealSum Zero((n+1)*(n+1))
        ([k][if (lft(RtD k)+rht(RtD k)<=n)
            (xs(lft(RtD k))*ys(rht(RtD k)))
            0]))
```

Permutation Pairs

0.1 Definition of Permutation Pairs

We ultimately want to prove the invariance of finite real sums under permutations. Before we can work on this, we need to introduce so called permutation pairs and prove some basic properties.

Definition of permutation pairs

A 4-tuple $(m, ps, f, f0)$ is a permuation pair if the functions $f, f0 : \mathbb{N} \to \mathbb{N}$ are mutually inverse, $ps : \mathbb{N} \to \{\text{True, False}\}$ masks out numbers over m and f is the identity on numbers masked out by ps. Thus we get the following definition:

```
(add-ids
(list (list "Pms"
        (make-arity (py "nat") (py "nat=>boole")
            (py "nat=>nat") (py "nat=>nat"))))
'("all m,ps,f,f0(
    all n f0(f n)=n  ->
    all n f(f0 n)=n  ->
    all n(m<n -> ps n -> F) ->
    all n((ps n -> F) -> f n=n) ->
    Pms m ps f f0)" "PmsIntro"))
```

Elimination properties

We now explicitely state that all permutation pairs have the properties enforced in the definition:

```
all m,ps,f,f0(Pms m ps f f0 -> all n f0(f n)=n)
all m,ps,f,f0(Pms m ps f f0 -> all n f(f0 n)=n)
all m,ps,f,f0(Pms m ps f f0 -> all n(m<n -> ps n -> F))
all m,ps,f,f0(Pms m ps f f0 -> all n((ps n -> F) -> f n=n))
```

Basic properties

We now show the necessary properties to later prove the invariance of real sums under permutations.

PmsSucc

A permutation pair on $\{0, \ldots, m\}$ is also a permutation pair on $\{0, \ldots, m+1\}$:

```
all m,ps,f,f0(Pms m ps f f0 -> Pms(Succ m)ps f f0)
```

Proof in Minlog!

PmsSuccInv

We can shrink the domain of definition of a permutation pair, if the last number is masked out.

```
all m,ps,f,f0(Pms(Succ m)ps f f0 ->
    (ps(Succ m) -> F) -> Pms m ps f f0)
```

PmsIn

The permutation mask is invariant under the permutation maps.

```
all m,ps,f,f0(Pms m ps f f0 -> all n(ps n -> ps(f n)))
```

PmsSym

One can swap the permutation maps:

```
all m,ps,f,f0(Pms m ps f f0 -> Pms m ps f0 f)
```

Sums under permutations

gPerm

In the later proof we will use induction on the length of permutations. To use the induction hypothesis we will need to construct shorter permutation. Consider a permutation pair $(l+1, ps, f, f_0)$ such that $ps(l+1) = \text{True}$ and $f, f_0(l+1) \neq l+1$. Then we construct:

$$g(n) := \begin{cases} n & n = l+1 \\ f(f(n)) & n = f_0(l+1) \\ f(n) \end{cases} \qquad g_0(n) := \begin{cases} n & n = l+1 \\ f_0(f_0(n)) & n = f(l+1) \\ f_0(n) \end{cases}$$

Furthermore define $ps_0(l+1) = \text{False}$ and $ps_0(n) = ps(n)$ for $n < l+1$. Then (l, ps_0, g, g_0) is a permutation pair.

```
    all l,ps,f,f0,ps0,g,g0(
        Pms(Succ l)ps f f0 -> ps(Succ l) ->
        (f(Succ l)=Succ l -> F) -> (f0(Succ l)=Succ l -> F) ->
        ps0 eqd([n][if (n=Succ l) False (ps n)]) ->
        g eqd([n][if (n=Succ l)
            n
            [if (n=f0(Succ l)) (f(f n)) (f n)]]) ->
        g0 eqd([n][if (n=Succ l)
            n
            [if (n=f(Succ l)) (f0(f0 n)) (f0 n)]]) ->
        Pms l ps0 g g0)
```

fSwapg

We show that f can be obtained from g (as defined above) by swapping $f(l+1)$ with $l+1$.

```
(add-program-constant "Swap" (py "nat=>nat=>nat=>nat"))
(add-computation-rules
    "Swap n m n0" "[if (n0=n) m [if (n0=m) n n0]]")
```

Then we can formulate the above proposition:

```
all l,ps,f,f0,g(
    Pms(Succ l)ps f f0 ->
    (f0(Succ l)=Succ l -> F) ->
    (Succ l=f(Succ l) -> F) ->
    g eqd([n][if (n=Succ l)
        n
        [if (n=f0(Succ l)) (f(f n)) (f n)]]) ->
    all n f n=Swap(f(Succ l))(Succ l)(g n))
```

RealSumSwap

We now want to show that any terms in a sum from zero can be swapped:

```
all xs,l,n,m(
    all n0 Real(xs n0) ->
    n<m ->
    m<Succ l ->
    RealSum Zero(Succ l)xs===
        RealSum Zero(Succ l)([n0]xs(Swap n m n0)))
```

To prove this we use the following preliminaries:

RealSumSplitCor1

$$\forall n \leq l : \sum_{i=0}^{l}(xs)_i = \sum_{i=0}^{n-1}(xs)_i + \sum_{i=n}^{l}(xs)_i$$

```
all xs,l,n(all n0(Real(xs n0)) -> n<Succ l ->
    RealSum Zero(Succ l)xs===
        RealSum Zero n xs+RealSum n(Succ l--n)xs)
```

RealSumSplitCor2

$$\forall n < m \leq l : \sum_{i=0}^{n-1}(xs)_i + \sum_{i=n}^{l}(xs)_i = \sum_{i=0}^{n-1}(xs)_i + \sum_{i=n}^{m}(xs)_i + \sum_{i=m+1}^{l}(xs)_i$$

```
all xs,l,n,m(
    all n0 Real(xs n0) ->
    n<m ->
    m<Succ l ->
    RealSum Zero n xs+RealSum n(Succ l--n)xs===
    RealSum Zero n xs+(RealSum n(Succ(m--n))xs
        +RealSum(Succ m)(l--m)xs))
```

RealSumSplitCor3

$$\forall n < m \leq l : \sum_{i=0}^{l} (xs)_{\mathrm{Swap}(n,m,i)} = \sum_{i=0}^{n} (xs)_{\mathrm{Swap}(n,m,i)} + \sum_{i=n}^{m} (xs)_{\mathrm{Swap}(n,m,i)} + \sum_{i=m+1}^{l} (xs)_{\mathrm{Swap}(n,m,i)}$$

```
all xs,l,n,m(
    all n0 Real(xs n0) ->
    n<m ->
    m<Succ l ->
    RealSum Zero(Succ l)([n0]xs(Swap n m n0))===
        RealSum Zero n([n0]xs(Swap n m n0))+
        RealSum n(Succ(m--n))([n0]xs(Swap n m n0))+
        RealSum(Succ m)(l--m)([n0]xs(Swap n m n0)))
```

RealSumSplitCor4

$$\forall n < m \leq l : \sum_{i=0}^{n-1} (xs)_{\mathrm{Swap}(n,m,i)} + \sum_{i=n}^{m} (xs)_{\mathrm{Swap}(n,m,i)} + \sum_{i=m+1}^{l} (xs)_{\mathrm{Swap}(n,m,i)} = \sum_{i=0}^{n-1} (xs)_i + \sum_{i=n}^{m} (xs)_{\mathrm{Swap}(n,m,i)} + \sum_{i=m+1}^{l} (xs)_i$$

```
all xs,l,n,m(
    all n0 Real(xs n0) ->
    n<m ->
    m<Succ l ->
    RealSum Zero n([n0]xs(Swap n m n0))+
    RealSum n(Succ(m--n))([n0]xs(Swap n m n0))+
    RealSum(Succ m)(l--m)([n0]xs(Swap n m n0))===
        RealSum Zero n xs+
        RealSum n(Succ(m--n))([n0]xs(Swap n m n0))+
        RealSum(Succ m)(l--m)xs)
```

RealSumSplitCor5

$$\forall n < m : \sum_{i=n}^{m} (xs)_i = \sum_{i=n}^{m} (xs)_{\mathrm{Swap}(n,m,i)}$$

```
all xs,n,m(
    all n0 Real(xs n0) ->
    n<m ->
    RealSum n(Succ(m--n))xs===
        RealSum n(Succ(m--n))([n0]xs(Swap n m n0)))
```

Proof Sketch (RealSumSwap)

$$\begin{aligned}\sum_{i=0}^{l} (xs)_i &\overset{C1}{=} \sum_{i=0}^{n-1} (xs)_i + \sum_{i=n}^{l} (xs)_i \\ &\overset{C2}{=} \sum_{i=0}^{n-1} (xs)_i + \sum_{i=n}^{m} (xs)_i + \sum_{i=m+1}^{l} (xs)_i \\ &\overset{C5}{=} \sum_{i=0}^{n-1} (xs)_i + \sum_{i=n}^{m} (xs)_{\mathrm{Swap}(n,m,i)} + \sum_{i=m+1}^{l} (xs)_i \\ &\overset{C4}{=} \sum_{i=0}^{n-1} (xs)_{\mathrm{Swap}(n,m,i)} + \sum_{i=n}^{m} (xs)_{\mathrm{Swap}(n,m,i)} + \sum_{i=m+1}^{l} (xs)_{\mathrm{Swap}(n,m,i)} \\ &\overset{C3}{=} \sum_{i=0}^{l} (xs)_{\mathrm{Swap}(n,m,i)}\end{aligned}$$

RealSumPms

Now using the above theorem we can prove invariance not only under swaps but also under any permutation map:

```
all l,xs(
    all n Real(xs n) ->
    all ps,f,f0(
    Pms l ps f f0 ->
    RealSum Zero(Succ l)xs===
        RealSum Zero(Succ l)([n]xs(f n))))
```