NATADD associative law

Johan G. F. Belinfante
2002 June 13

```
<< goedel52.o50; << tools.m
```

```
:Package Title: goedel52.o50          2002 June 13 at 3:00 p.m.

It is now:  2002 Jun 13 at 15:21

Loading Simplification Rules

TOOLS.M                   Revised 2002 June 12

weightlimit = 40
```

■ Introduction

In this notebook the associativity of addition of natural numbers is derived. The obvious place to start is the **AssocTest**.

```
Assoc[image[power[x], u], image[power[x], v], image[power[x], w]]
```

```
image[power[x], image[iterate[SUCC, w], image[iterate[SUCC, u], v]]] ==
 image[power[x], image[iterate[SUCC, w], image[iterate[SUCC, v], u]]]
```

We cannot make this into a rule, because it would loop. A variable−freeapproach is called for, just as for the commutative law.

■ The idea of the derivation

The derivation is inspired by this observation:

```
image[composite[NATADD, cross[NATADD, Id]], cart[cart[u, v], w]]
```

```
intersection[omega, image[iterate[SUCC, w], image[iterate[SUCC, v], u]]]
```

```
image[composite[NATADD, cross[Id, NATADD], ASSOC, cross[SWAP, Id]], cart[cart[u, v], w]]
```

```
intersection[omega, image[iterate[SUCC, w], image[iterate[SUCC, v], u]]]
```

The factor of **cross[SWAP,Id]** is a temporary nuisance that will be eliminated later using the commutative law of addition.

■ Normality for power[SUCC]

The most basic **Normality** result is this:

```
power[SUCC] // Normality // Reverse
```

```
U[intersection[P[cart[omega, V]], subvar[
    union[cross[SUCC, cross[Id, SUCC]], id[cart[singleton[0], Id]]]]]] == power[SUCC]
```

```
U[intersection[P[cart[omega, V]], subvar[
    union[cross[SUCC, cross[Id, SUCC]], id[cart[singleton[0], Id]]]]]] := power[SUCC]
```

Beyond this one often needs another rule:

```
composite[rotate[inverse[power[SUCC]]], SWAP] // VSTriNormality // Reverse
```

```
rotate[composite[
   complement[composite[Di, SECOND, intersection[composite[complement[inverse[E]],
       IMAGE[cross[SUCC, SUCC]]], inverse[E]]]], id[intersection[P[cart[omega, V]],
     subvar[union[cross[SUCC, SUCC], id[cart[singleton[0], V]]]]]], E]] ==
 composite[rotate[inverse[power[SUCC]]], SWAP]
```

```
rotate[composite[
   complement[composite[Di, SECOND, intersection[composite[complement[inverse[E]],
       IMAGE[cross[SUCC, SUCC]]], inverse[E]]]], id[intersection[P[cart[omega, V]],
     subvar[union[cross[SUCC, SUCC], id[cart[singleton[0], V]]]]]], E]] :=
 composite[rotate[inverse[power[SUCC]]], SWAP]
```

■ NATADD normality

Before starting we add this basic rule:

```
composite[NATADD, id[cart[omega, V]]] // TripleRotate
```

```
composite[NATADD, id[cart[omega, V]]] == NATADD
```

```
composite[NATADD, id[cart[omega, V]]] := NATADD
```

The following rule is only needed when **x** is **NATADD**, but the generality makes for a faster derivation due to sequestering.

```
composite[NATADD, cross[x, Id]] // VSTriNormality // Reverse
```

```
composite[SECOND, id[cart[V, omega]], intersection[composite[inverse[FIRST], x, FIRST],
   composite[inverse[E], id[intersection[P[cart[omega, V]], subvar[
       union[cross[SUCC, SUCC], id[cart[singleton[0], V]]]]]], complement[composite[
      intersection[E, composite[inverse[IMAGE[cross[SUCC, SUCC]]], complement[E]]],
      inverse[SECOND], Di]], SECOND]]] == composite[NATADD, cross[x, Id]]
```

```
composite[SECOND, id[cart[V, omega]], intersection[composite[inverse[FIRST], x_, FIRST],
   composite[inverse[E], id[intersection[P[cart[omega, V]], subvar[
       union[cross[SUCC, SUCC], id[cart[singleton[0], V]]]]]], complement[composite[
      intersection[E, composite[inverse[IMAGE[cross[SUCC, SUCC]]], complement[E]]],
      inverse[SECOND], Di]], SECOND]]] := composite[NATADD, cross[x, Id]]
```

The specialization to the case of interest is done like this:

```
Assoc[NATADD, cross[id[omega], Id], cross[rotate[inverse[power[SUCC]]], Id]] // Reverse
```

```
composite[NATADD, cross[rotate[inverse[power[SUCC]]], Id]] ==
 composite[NATADD, cross[NATADD, Id]]
```

```
composite[NATADD, cross[rotate[inverse[power[SUCC]]], Id]] :=
 composite[NATADD, cross[NATADD, Id]]
```

The derivation of the associative law presents a rare occasion to use **VSTerNormality**.

```
composite[NATADD, cross[Id, NATADD], ASSOC, cross[SWAP, Id]] // VSTerNormality
```

```
composite[NATADD, cross[Id, NATADD], ASSOC, cross[SWAP, Id]] ==
 composite[NATADD, cross[NATADD, Id]]
```

As promised, we use the commutative law to remove the nuisance factor of **FLIP = cross[SWAP,Id]**.

```
Map[composite[#, cross[SWAP, Id]] &, %]
```

```
composite[NATADD, cross[Id, NATADD], ASSOC] == composite[NATADD, cross[NATADD, Id]]
```

Because of the commutative law, we have also rotation invariance

```
Map[flip[composite[#, inverse[ASSOC]]] &, %] // Reverse
```

```
composite[NATADD, cross[NATADD, Id], ROT] == composite[NATADD, cross[NATADD, Id]]
```

Both of these rules will be kept.

```
composite[NATADD, cross[Id, NATADD], ASSOC] := composite[NATADD, cross[NATADD, Id]]
```

```
composite[NATADD, cross[NATADD, Id], ROT] := composite[NATADD, cross[NATADD, Id]]
```