

# twisted addition and subtraction

Johan G. F. Belinfante  
2007 February 20

```
In[1]:= SetDirectory["1:"]; << goedel90.17a; << tools.m

:Package Title: goedel90.17a      2007 February 17 at 5:00 p.m.

It is now: 2007 Feb 20 at 18:18

Loading Simplification Rules

TOOLS.M                          Revised 2007 January 7

weightlimit = 40
```

---

## summary

Some analogs are derived for the following **TWIST** rule.

```
In[2]:= composite[NATADD, cross[NATADD, NATADD], TWIST]

Out[2]= composite[NATADD, cross[NATADD, NATADD]]
```

When one replaces **NATADD** with **rotate[NATADD]**, similar but slightly more complicated rules are obtained.

---

## corollary of the original formula

The following equation is a corollary of the existing **TWIST** formula.

```
In[3]:= Map[
  equal[#, image[NATADD, cart[image[NATADD, cart[w, y]], image[NATADD, cart[x, z]]]]] &,
  ImageComp[composite[NATADD, cross[NATADD, NATADD]],
  TWIST, cart[cart[w, x], cart[y, z]]]]

Out[3]= equal[image[NATADD, cart[image[NATADD, cart[w, x]], image[NATADD, cart[y, z]]]],
  image[NATADD, cart[image[NATADD, cart[w, y]], image[NATADD, cart[x, z]]]]] == True

In[4]:= equal[image[NATADD, cart[image[NATADD, cart[w_, x_]], image[NATADD, cart[y_, z_]]]],
  image[NATADD, cart[image[NATADD, cart[w_, y_]], image[NATADD, cart[x_, z_]]]]] := True
```

---

## sums of differences

Lemma.

```
In[5]:= Map[composite[#, id[cartsq[cartsq[omega]]]] &, SubstTest[reify, x,
  image[t, cart[cart[set[nat[first[first[x]]]], set[nat[second[first[x]]]]],
    cart[set[nat[first[second[x]]]], set[nat[second[second[x]]]]]]],
  t -> dif[composite[NATADD, cross[rotate[NATADD], rotate[NATADD]], TWIST],
    composite[rotate[NATADD], cross[NATADD, NATADD]]]]]

Out[5]= composite[intersection[composite[complement[rotate[NATADD]], cross[NATADD, NATADD]],
  composite[NATADD, cross[rotate[NATADD], rotate[NATADD]], TWIST]],
  id[cart[cart[omega, omega], cart[omega, omega]]]] = 0
```

```
In[6]:= % /. Equal -> SetDelayed
```

The above result can be cleaned up as follows.

```
In[7]:= AssInt[composite[complement[rotate[NATADD]], cross[NATADD, NATADD]],
  cart[cart[cart[omega, omega], cart[omega, omega]], V],
  composite[NATADD, cross[rotate[NATADD], rotate[NATADD]], TWIST]] // Reverse
```

```
Out[7]= intersection[composite[complement[rotate[NATADD]], cross[NATADD, NATADD]],
  composite[NATADD, cross[rotate[NATADD], rotate[NATADD]], TWIST]] = 0
```

```
In[8]:= % /. Equal -> SetDelayed
```

Theorem.

```
In[9]:= SubstTest[empty, dif[u, v],
  {u -> composite[NATADD, cross[rotate[NATADD], rotate[NATADD]], TWIST],
  v -> composite[rotate[NATADD], cross[NATADD, NATADD]]}]
```

```
Out[9]= subclass[composite[NATADD, cross[rotate[NATADD], rotate[NATADD]], TWIST],
  composite[rotate[NATADD], cross[NATADD, NATADD]]] = True
```

```
In[10]:= subclass[composite[NATADD, cross[rotate[NATADD], rotate[NATADD]], TWIST],
  composite[rotate[NATADD], cross[NATADD, NATADD]]] := True
```

Corollary.

```
In[11]:= SubstTest[implies, subclass[u, v], subclass[image[u, t], image[v, t]],
  {u -> composite[NATADD, cross[rotate[NATADD], rotate[NATADD]], TWIST],
  v -> composite[rotate[NATADD], cross[NATADD, NATADD]],
  t -> cart[cart[w, y], cart[x, z]]} // Reverse
```

```
Out[11]= subclass[image[NATADD,
  cart[image[image[inverse[NATADD], w], x], image[image[inverse[NATADD], y], z]]],
  image[image[inverse[NATADD], image[NATADD, cart[w, y]]],
  image[NATADD, cart[x, z]]] = True
```

```
In[12]:= subclass[image[NATADD,
  cart[image[image[inverse[NATADD], w_], x_], image[image[inverse[NATADD], y_], z_]],
  image[image[inverse[NATADD], image[NATADD, cart[w_, y_]]],
  image[NATADD, cart[x_, z_]]] := True
```

---

## differences of differences

Lemma.

```
In[13]:= Map[composite[#, id[cartsq[cartsq[omega]]] &, SubstTest[reify, x,
  image[t, cart[cart[set[nat[first[first[x]]], set[nat[second[first[x]]]],
  cart[set[nat[first[second[x]]], set[nat[second[second[x]]]]]]],
  t -> dif[composite[rotate[NATADD], cross[rotate[NATADD], rotate[NATADD]]],
  composite[rotate[NATADD], cross[NATADD, NATADD], TWIST, cross[Id, SWAP]]]]]
```

```
Out[13]= composite[
  intersection[composite[rotate[NATADD], cross[rotate[NATADD], rotate[NATADD]]],
  composite[complement[rotate[NATADD]], cross[NATADD, NATADD], TWIST,
  cross[Id, SWAP]]], id[cart[cart[omega, omega], cart[omega, omega]]] = 0
```

```
In[14]:= % /. Equal -> SetDelayed
```

Clean up as before:

```
In[15]:= AssInt[
  cart[cart[cart[omega, omega], cart[omega, omega]], V],
  composite[rotate[NATADD], cross[rotate[NATADD], rotate[NATADD]]],
  composite[complement[rotate[NATADD]],
  cross[NATADD, NATADD], TWIST, cross[Id, SWAP]] // Reverse
```

```
Out[15]= intersection[composite[rotate[NATADD], cross[rotate[NATADD], rotate[NATADD]]],
  composite[complement[rotate[NATADD]],
  cross[NATADD, NATADD], TWIST, cross[Id, SWAP]]] = 0
```

```
In[16]:= % /. Equal -> SetDelayed
```

Theorem.

```
In[17]:= SubstTest[empty, dif[u, v],
  {u -> composite[rotate[NATADD], cross[rotate[NATADD], rotate[NATADD]]],
  v -> composite[rotate[NATADD], cross[NATADD, NATADD], TWIST, cross[Id, SWAP]]}]
```

```
Out[17]= subclass[composite[rotate[NATADD], cross[rotate[NATADD], rotate[NATADD]]],
  composite[rotate[NATADD], cross[NATADD, NATADD], TWIST, cross[Id, SWAP]]] = True
```

```
In[18]:= subclass[composite[rotate[NATADD], cross[rotate[NATADD], rotate[NATADD]]],
  composite[rotate[NATADD], cross[NATADD, NATADD], TWIST, cross[Id, SWAP]]] := True
```

Corollary.

```
In[19]:= SubstTest[implies, subclass[u, v], subclass[image[u, t], image[v, t]],
  {u -> composite[rotate[NATADD], cross[rotate[NATADD], rotate[NATADD]]],
   v -> composite[rotate[NATADD], cross[NATADD, NATADD], TWIST, cross[Id, SWAP]],
   t -> cart[cart[w, x], cart[y, z]]} // Reverse

Out[19]= subclass[image[image[inverse[NATADD], image[image[inverse[NATADD], w], x]],
  image[image[inverse[NATADD], y], z]], image[
  image[inverse[NATADD], image[NATADD, cart[w, z]]], image[NATADD, cart[x, y]]]] == True

In[20]:= subclass[image[image[inverse[NATADD], image[image[inverse[NATADD], w_], x_]],
  image[image[inverse[NATADD], y_], z_]],
  image[image[inverse[NATADD], image[NATADD, cart[w_, z_]]],
  image[NATADD, cart[x_, y_]]]] := True
```