

# distributive law for product of a difference

*Johan G. F. Belinfante*  
2002 October 5

```
<< goedel52.p73; << tools.m
:Package Title: goedel52.p73          2002 October 4 at 5:35 p.m.

It is now: 2002 Oct 5 at 3:10

Loading Simplification Rules

TOOLS.M                               Revised 2002 September 16

weightlimit = 40
```

## ■ Summary

This notebook contains a conditional distributive law for **natmul[x,natsub[y,z]]**. The following example shows that an unconditional distributive law is not generally valid:

```
equal[natmul[x, natsub[y, z]], natsub[natmul[x, y], natmul[x, z]]] /.
  {x -> 0, y -> 0, z -> singleton[0]}

False
```

The law that will be derived contains an extra term to deal with this limitation.

## ■ derivation

The main step in the derivation is this:

```
SubstTest[natmul, x, natadd[w, z], w -> natsub[y, z]]

union[complement[image[V, intersection[omega, singleton[z]]]],
  complement[image[V, intersection[omega, image[S, singleton[z]], singleton[y]]]],
  natmul[x, y]] == natadd[natmul[x, z], natmul[x, natsub[y, z]]]
```

Next we subtract **natmul[x,z]** from both sides:

```
Map[natsub[#, natmul[x, z]] &, %]

union[complement[image[V, intersection[omega, singleton[z]]]],
  complement[image[V, intersection[omega, image[S, singleton[z]], singleton[y]]]],
  natsub[natmul[x, y], natmul[x, z]]] ==
  union[complement[image[V, intersection[omega, singleton[z]]], natmul[x, natsub[y, z]]]
```

Finally, everything on the right side is transposed to the left side as follows:

```

Map[equal[Last[%], #] &, %]

or[equal[natmul[x, natsub[y, z]], natsub[natmul[x, y], natmul[x, z]]],
  not[member[x, omega]], not[member[y, omega]],
  not[member[z, omega]], not[subclass[z, y]]] == True

```

This fact is added as a temporary rule:

```

or[equal[natmul[x_, natsub[y_, z_]], natsub[natmul[x_, y_], natmul[x_, z_]]],
  not[member[x_, omega]], not[member[y_, omega]],
  not[member[z_, omega]], not[subclass[z_, y_]]] := True

```

At this point the **GOEDEL** program recognizes the truth of this equation:

```

equal[natmul[x, natsub[y, z]],
  union[image[V, intersection[z, complement[y]]], natsub[natmul[x, y], natmul[x, z]]]]

True

```

On the basis of this, it is justified to add the following rewrite rule:

```

natmul[x_, natsub[y_, z_]] :=
  union[image[V, intersection[z, complement[y]]], natsub[natmul[x, y], natmul[x, z]]]

```