max-min monus formulas

Johan G. F. Belinfante and Claudia D. Huang
2005 July 2

```
In[1]:= SetDirectory["i:"]; << goedel70.30a; << tools.m

        :Package Title: goedel70.30a       2005 June 30 at 9:00 a.m.

        It is now:  2005 Jul 2 at 18:1

        Loading Simplification Rules

        TOOLS.M                       Revised 2005 June 17

        weightlimit = 40
```

---

summary

In Art Quaife's development of the arithmetic of natural numbers, using William McCune's automated reasoning program **Otter**, extensive use is made of floored subtraction.

```
In[2]:= "Art Quaife, Automated Development of Fundamental Mathematical Theories,
          Kluwer Academic Publishers, Dordrecht, the Netherlands, 1992.";
```

Comparing Quaife's formalism with that of the **GOEDEL** program is facilitated by interpreting floored subtraction as **monus[x,y]**, defined by wrapping **nat** around **natsub[x,y]**.

```
In[3]:= monus[x, y]

Out[3]= nat[natsub[x, y]]
```

If **x** and **y** are natural numbers, then **monus[x,y]** is their difference when **y** is less than or equal to **x**, and is zero otherwise. Two formulas for **monus** are derived in this notebook which involve the minimum and maximum of two natural numbers. In Gödel's formalism, each natural number is the set of all lesser ones, and therefore the minimum and maximum of two natural numbers are simply their intersection and union, respectively. Quaife's definitions of the maximum and minimum of two numbers now become theorems, derived here as corollaries.

two formulas for monus

Lemma 1.

```
In[4]:= SubstTest[and, subclass[u, v], subclass[v, u],
          {u → natsub[nat[x], nat[y]], v -> natsub[nat[x], nat[z]]}] /.
         z → intersection[nat[x], nat[y]] // Reverse

Out[4]= equal[natsub[nat[x], intersection[nat[x], nat[y]]], natsub[nat[x], nat[y]]] ==
         not[member[nat[x], nat[y]]]

In[5]:= equal[natsub[nat[x_], intersection[nat[x_], nat[y_]]],
         natsub[nat[x_], nat[y_]]] := not[member[nat[x], nat[y]]]
```

Lemma 2.

```
In[6]:= SubstTest[implies, and[equal[u, v], equal[v, w]],
         equal[u, w], {u -> natsub[nat[x], intersection[nat[x], nat[y]]],
          v → natsub[nat[x], nat[y]], w -> monus[nat[x], nat[y]]}]

Out[6]= or[equal[nat[natsub[nat[x], nat[y]]],
          natsub[nat[x], intersection[nat[x], nat[y]]]],
         member[nat[x], nat[y]]] == True

In[7]:= (% /. {x → x_, y → y_}) /. Equal → SetDelayed
```

Lemma 3.

```
In[8]:= SubstTest[implies, and[equal[u, v], equal[v, w]],
         equal[u, w], {u -> natsub[nat[x], intersection[nat[x], nat[y]]],
          v → 0, w -> monus[nat[x], nat[y]]}]

Out[8]= or[equal[nat[natsub[nat[x], nat[y]]],
          natsub[nat[x], intersection[nat[x], nat[y]]]],
         member[nat[y], nat[x]]] == True

In[9]:= (% /. {x → x_, y → y_}) /. Equal → SetDelayed
```

The first of two formulas for **monus** now follows, and is made into a rewrite rule.

```
In[10]:= SubstTest[and, or[p1, p3], or[p2, p3], {p1 -> member[nat[x], nat[y]],
            p2 → member[nat[y], nat[x]], p3 -> equal[nat[natsub[nat[x], nat[y]]],
              natsub[nat[x], intersection[nat[x], nat[y]]]]}] // Reverse

Out[10]= equal[nat[natsub[nat[x], nat[y]]],
           natsub[nat[x], intersection[nat[x], nat[y]]]] == True
```

```
In[11]:= natsub[nat[x_], intersection[nat[x_], nat[y_]]] := nat[natsub[nat[x], nat[y]]]
```

The second formula for **monus** is obtained as a corollary, and also made into a rewrite rule.

```
In[12]:= SubstTest[equal, natadd[u, w], natadd[v, w],
           {u -> natsub[nat[x], intersection[nat[x], nat[y]]],
            v → natsub[union[nat[x], nat[y]], nat[y]], w → nat[y]}] // Reverse

Out[12]= equal[nat[natsub[nat[x], nat[y]]],
           natsub[union[nat[x], nat[y]], nat[y]]] == True

In[13]:= natsub[union[nat[x_], nat[y_]], nat[y_]] := nat[natsub[nat[x], nat[y]]]
```

Quaife's (MMDEF1)

In a previous notebook, dealing with Quaife's **DF** group, the following formula was derived for expressions with **monus** substituted inside **natadd**.

```
In[14]:= natadd[nat[x], monus[nat[y], nat[z]]]

Out[14]= union[intersection[image[V, intersection[nat[y], set[nat[z]]]],
           natsub[natadd[nat[x], nat[y]], nat[z]]], nat[x]]
```

This complicated rewrite rule is removed here, and will be replaced with a simpler rewrite rule that corresponds to Quaife's second equation of his **(MMDEF1)**.

```
In[15]:= natadd[nat[x_], nat[natsub[nat[y_], nat[z_]]]] =.
```

This equation, which was used by Quaife to define the maximum of two natural numbers, suffices for the **GOEDEL** program to recognize the validity of all theorems of Quaife's **DF** group. It can be quickly derived as follows:

```
In[16]:= SubstTest[natadd, nat[y], natsub[nat[z], nat[y]], z -> union[nat[x], nat[y]]]

Out[16]= natadd[nat[y], nat[natsub[nat[x], nat[y]]]] == union[nat[x], nat[y]]

In[17]:= natadd[nat[y_], nat[natsub[nat[x_], nat[y_]]]] := union[nat[x], nat[y]]
```

In a similar fashion, the first equation of Quaife's **(MMDEF1)**, which he used to define the minimum of two numbers, can be derived from the first formula for **monus** by double transposition:

```
In[18]:= SubstTest[implies, equal[natsub[nat[u], nat[v]], nat[w]],
          equal[natsub[nat[u], nat[w]], nat[v]],
          {u → nat[x], v → intersection[nat[x], nat[y]], w -> natsub[nat[x], nat[y]]}]

Out[18]= equal[intersection[nat[x], nat[y]],
           natsub[nat[x], nat[natsub[nat[x], nat[y]]]]] == True

In[19]:= natsub[nat[x_], nat[natsub[nat[x_], nat[y_]]]] := intersection[nat[x], nat[y]]
```