the PS machine versus VERTSECT

Johan G. F. Belinfante
2004 April 22

```
In[1]:= << goedel56.20a; << tools.m;

        :Package Title: goedel56.20a         2004 April 20 at 5:20 p.m.

        It is now:  2004 Apr 22 at 10:53

        Loading Simplification Rules

        TOOLS.M                      Revised 2004 April 18

        weightlimit = 40
```

summary

The use of **PS** and **VERTSECT** both provide methods of cancelling factors of **S** and **inverse[S]** from formulas for functions. The formulas used in the **Otter** work to define **BIGCAP** and **POWER** made use of the proper subset relation **PS**. This notebook compares these constructions with alternate formulas using **VERTSECT** that would perhaps have led to simpler definitions.

three new rewrite rules

To simplify the discussion it is useful to add a few new rewrite rules. The following is just the inverse of an existing rule:

```
In[2]:= intersection[complement[composite[inverse[PS], inverse[S]]], inverse[S]] //
         DoubleInverse

Out[2]= intersection[complement[composite[inverse[PS], inverse[S]]], inverse[S]] == Id

In[3]:= intersection[complement[composite[inverse[PS], inverse[S]]], inverse[S]] := Id
```

The following identities involve **VERTSECT**.

```
In[4]:= complement[composite[complement[E], x]] // VSNormality

Out[4]= composite[Id, complement[composite[complement[E], x]]] == composite[S, VERTSECT[x]]
```

It is all right to omit the factor of **Id** here.

```
In[5]:= composite[complement[composite[complement[E], x_]]] := composite[S, VERTSECT[x]]
```

One could add a rule of this sort, but one can do better:

```
In[6]:=  complement[composite[complement[E], x]] // Normality

Out[6]=  complement[composite[complement[E], x]] ==
          union[complement[cart[V, V]], composite[S, VERTSECT[x]]]

In[7]:=  complement[composite[complement[E], x_]] :=
          union[complement[cart[V, V]], composite[S, VERTSECT[x]]]

In[8]:=  intersection[composite[S, VERTSECT[x]], composite[complement[E], x]] // VSNormality

Out[8]=  intersection[composite[S, VERTSECT[x]], composite[complement[E], x]] == 0

In[9]:=  intersection[composite[S, VERTSECT[x_]], composite[complement[E], x_]] := 0
```

the Otter definitions of BIGCAP and POWER

The definition of **BIGCAP** in the **Otter** work amounts to this formula:

```
In[10]:=  dif[LB[S], composite[inverse[PS], LB[S]]]

Out[10]=  BIGCAP
```

The relation **inverse[LB[S]]** was called **LOWERBOUND** in the **Otter** work. The function **BIGCAP** could also have been recovered from **LB[S]** as follows:

```
In[11]:=  composite[BIGCUP, VERTSECT[LB[S]]]

Out[11]=  BIGCAP
```

Both of these constructions are examples of general methods for cancelling a factor of **inverse[S]**:

```
In[12]:=  composite[BIGCUP, VERTSECT[#]] & @ composite[inverse[S], ff]

Out[12]=  union[ff, cart[complement[domain[ff]], singleton[0]]]

In[13]:=  dif[#, composite[inverse[PS], #]] & @ composite[inverse[S], ff]

Out[13]=  ff
```

A simpler formula for **BIGCAP** is this:

```
In[14]:=  VERTSECT[LB[E]]

Out[14]=  BIGCAP
```

A similar technique could have been used to define **POWER**, but the **Otter** work but did not use this:

```
In[15]:=  dif[#, composite[inverse[PS], #]] & @ inverse[UB[S]]

Out[15]=  POWER
```

Instead the following relation was used in the **Otter** work.

```
In[16]:=  SUPERPOWER == composite[Id, complement[composite[complement[E], inverse[S]]]]

Out[16]=  SUPERPOWER == composite[S, POWER]
```

The following proper−subset relation was used to cancel the factor of **S**, using the following general observation:

```
In[17]:=  intersection[#, complement[composite[PS, #]]] & @ composite[S, ff]

Out[17]=  ff
```

Thus, in light of the rewrite rule derived in the preceding section, the **Otter** formula amounts to this:

```
In[18]:=  intersection[#, complement[composite[PS, #]]] & @ composite[S, VERTSECT[inverse[S]]]

Out[18]=  POWER
```

A simpler method would have been to use the following formula that appears in work by Formisano and Omeodeo:

```
In[19]:=  VERTSECT[inverse[S]]

Out[19]=  POWER
```

The reference is this:

```
        "Andrea Formisano, Eugenio G.Omodeo: An Equational Re-engineering of Set Theories.
         in Automated Deduction in Classical and Non-Classical Logics: Selected Papers
        Editors: R. Caferra, G. Salzer (Eds.):  FTP (LNCS vol 1761/2000) 1998: pp.175-190"
```

The function **BIGCUP** could have been constructed in a similar way, using a double application of **complement**.

```
In[20]:=  VERTSECT[complement[LB[complement[E]]]]

Out[20]=  BIGCUP
```

Note however that

```
In[21]:=  VERTSECT[complement[LB[complement[x]]]]

Out[21]=  IMAGE[inverse[x]]
```

So this amounts to using the formula that was used in the **Otter** work:

```
In[22]:=  IMAGE[inverse[E]]

Out[22]=  BIGCUP
```