co-restrictions

Johan G. F. Belinfante
2006 October 18

```
In[1]:= SetDirectory["l:"]; << goedel86.17a; << tools.m

        :Package Title: goedel86.17a          2006 October 17 at 2:00 p.m.

        It is now:  2006 Oct 18 at 0:53

        Loading Simplification Rules

        TOOLS.M                      Revised 2006 October 12

        weightlimit = 40
```

summary

The class **RS[x]** of restrictions of a class **x** is the class of all sets of the form **composite[x, id[y]]**. This class is related to the power class **P[x]** by the formula

```
In[2]:= intersection[invar[composite[id[x], inverse[FIRST], FIRST]], P[composite[Id, x]]]

Out[2]= RS[x]
```

In this notebook a similar formula is derived for the class of all sets of the form **composite[id[w], x]**. These are inverses of restrictions of the inverse of **x**. The key to the derivation is the use of the class **transvar[x,y]**.

inverses of restrictions

The class of inverses of the elements of **z** is **image[IMAGE[SWAP], z]**. When **z** is a ckass of relations, one can replace **IMAGE[SWAP**] with **INVERSE**. This is the case for the class of restrictions of any class. To settle on a standard expression for this class, and also for the sake of brevity, the following rewrite rule will be adopted.

```
In[4]:= ImageComp[IMAGE[SWAP], id[P[cart[V, V]]], RS[x]] // Reverse

Out[4]= image[IMAGE[SWAP], RS[x]] == image[INVERSE, RS[x]]

In[5]:= image[IMAGE[SWAP], RS[x_]] := image[INVERSE, RS[x]]
```

transvar formulas

The class **transvar[x, y]** is defined by

```
In[6]:= class[w, subclass[image[x, w], image[y, w]]]

Out[6]= transvar[x, y]
```

This class is related to **invar[x]** by the formula

```
In[7]:= transvar[x, Id]

Out[7]= invar[x]
```

Replacing both **x** and **y** by their flips yields:

```
In[8]:= transvar[flip[x], flip[y]] // Normality

Out[8]= transvar[composite[x, SWAP], composite[y, SWAP]] ==
         image[inverse[IMAGE[SWAP]], transvar[x, y]]

In[9]:= transvar[composite[x_, SWAP], composite[y_, SWAP]] :=
         image[inverse[IMAGE[SWAP]], transvar[x, y]]
```

Similarly, composing with **SWAP** on the other side, one obtains:

```
In[10]:= transvar[composite[SWAP, x], composite[SWAP, y]] // Normality

Out[10]= transvar[composite[SWAP, x], composite[SWAP, y]] ==
          transvar[composite[id[cart[V, V]], x], y]

In[11]:= transvar[composite[SWAP, x_], composite[SWAP, y_]] :=
          transvar[composite[id[cart[V, V]], x], y]
```

Composing with **SWAP** on both sides, and replacing **y** with **Id** yields

```
In[12]:= SubstTest[transvar, composite[SWAP, x, SWAP], composite[SWAP, y, SWAP], y → Id]

Out[12]= invar[composite[SWAP, x, SWAP]] ==
          image[inverse[IMAGE[SWAP]], invar[composite[id[cart[V, V]], x]]]

In[13]:= invar[composite[SWAP, x_, SWAP]] :=
          image[inverse[IMAGE[SWAP]], invar[composite[id[cart[V, V]], x]]]
```

class of co-restrictions

Temporary lemma.

```
In[14]:= SubstTest[invar, composite[SWAP, z, SWAP],
          z -> composite[id[x], inverse[SECOND], SECOND]]

Out[14]= invar[composite[id[inverse[x]], inverse[FIRST], FIRST]] ==
          image[inverse[IMAGE[SWAP]], invar[composite[id[x], inverse[SECOND], SECOND]]]

In[15]:= invar[composite[id[inverse[x_]], inverse[FIRST], FIRST]] :=
          image[inverse[IMAGE[SWAP]], invar[composite[id[x], inverse[SECOND], SECOND]]]
```

Theorem.

```
In[16]:= Map[image[IMAGE[SWAP], #] &, SubstTest[intersection, P[composite[Id, y]],
           invar[composite[id[y], inverse[FIRST], FIRST]], y → inverse[x]]]

Out[16]= intersection[invar[composite[id[x], inverse[SECOND], SECOND]], P[composite[Id, x]]] ==
          image[INVERSE, RS[inverse[x]]]

In[17]:= intersection[invar[composite[id[x_], inverse[SECOND], SECOND]],
           P[composite[Id, x_]]] := image[INVERSE, RS[inverse[x]]]
```