similarity of inverses

Johan G. F. Belinfante
2010 February 27

```
In[1]:= SetDirectory["l:"]; << goedel.10feb26b; << tools.m

:Package Title: goedel.10feb26b                2010 February 26 at midnite

It is now:  2010 Feb 27 at 19:35

Loading Simplification Rules

TOOLS.M                                       Revised 2010 February 26

weightlimit = 40
```

summary

If relations are similar, then so are their inverses. A variable-free statement of this fact is that the similarity relation commutes with the function that takes any relation to its inverse. The derivation of this fact presented here is also entirely variable-free.

derivation

Lemma. This rewrite rule allows the **GOEDEL** program to recognize that **x ∘ y ∘ inverse[x]** is the same as **image[x ⊗ x, y]**.

```
In[2]:= Map[flip,
          composite[IMG, cross[composite[CROSS, DUP], Id]] // FastReifTriNormality // Reverse]

Out[2]= composite[COMPOSE,
          intersection[composite[inverse[FIRST], IMAGE[id[cart[V, V]]], SECOND],
           composite[inverse[SECOND], COMPOSE, cross[Id, IMAGE[SWAP]]]]] ==
         composite[IMG, SWAP, cross[Id, composite[CROSS, DUP]]]

In[3]:= composite[COMPOSE,
          intersection[composite[inverse[FIRST], IMAGE[id[cart[V, V]]], SECOND],
           composite[inverse[SECOND], COMPOSE, cross[Id, IMAGE[SWAP]]]]] :=
         composite[IMG, SWAP, cross[Id, composite[CROSS, DUP]]]
```

Lemma. The inverse of **x ∘ y ∘ inverse[x]** is **x ∘ inverse[y] ∘ inverse[x]**.

```
In[12]:= composite[INVERSE, IMG, cross[composite[CROSS, DUP], INVERSE]] // FastReifTriNormality

Out[12]= composite[INVERSE, IMG, cross[composite[CROSS, DUP], INVERSE]] ==
          composite[IMG, cross[composite[CROSS, DUP], id[P[cart[V, V]]]]]

In[13]:= composite[INVERSE, IMG, cross[composite[CROSS, DUP], INVERSE]] :=
          composite[IMG, cross[composite[CROSS, DUP], id[P[cart[V, V]]]]]
```

Lemma. A simplification rule.

```
In[14]:= Map[image[#, id[x]] &,
           composite[id[IMAGE[FIRST]], inverse[FIRST], CROSS] // FastReifTriNormality] // Reverse

Out[14]= composite[CART, DUP, IMAGE[FIRST], id[x], inverse[DUP], inverse[CROSS]] ==
          composite[IMAGE[FIRST], id[image[CROSS, id[x]]]]

In[15]:= composite[CART, DUP, IMAGE[FIRST], id[x_], inverse[DUP], inverse[CROSS]] :=
          composite[IMAGE[FIRST], id[image[CROSS, id[x]]]]
```

Theorem. If relations are similar, then so are their inverses.

```
In[17]:= (Map[composite[#, INVERSE] &, Assoc[INVERSE, composite[IMG,
               cross[composite[CROSS, DUP], Id], id[x], cross[Id, INVERSE]], inverse[SECOND]]] /.
            x → composite[inverse[S], CART, DUP, IMAGE[FIRST], id[BIJ]]) // Reverse

Out[17]= composite[SIMILAR, INVERSE] == composite[INVERSE, SIMILAR]

In[18]:= composite[SIMILAR, INVERSE] := composite[INVERSE, SIMILAR]
```