restrictions of functions

Johan G. F. Belinfante
2009 February 24

```
In[1]:= SetDirectory["l:"]; << goedel.09feb23a; << tools.m

        :Package Title: goedel.09feb23a        2009 February 23 at 4:15 p.m.

        It is now:  2009 Feb 24 at 10:33

        Loading Simplification Rules

        TOOLS.M                                      Revised 2009 February 18

        weightlimit = 40
```

summary

The function **IMAGE[composite[id[x], inverse[FIRST]]** is useful for describing classes of restrictions of a function **x**.

introduction

The relation **composite[id[x], inverse[FIRST]]** is a correspondence between the points in a relation **x** and the points of its domain.

```
In[2]:= class[pair[u, pair[v, w]], and[equal[u, v], member[pair[v, w], x]]]

Out[2]= composite[id[x], inverse[FIRST]]
```

This relation is a function only if the relational part of **x** is a function.

```
In[3]:= FUNCTION[composite[id[x], inverse[FIRST]]]

Out[3]= FUNCTION[composite[Id, x]]
```

Its inverse is always a function, and therefore when **x** is a function, **composite[id[x], inverse[FIRST]]** is one-to-one. This one-to-one correspondence implies, for example, that the cardinality of a function is the same as that of its domain.

```
In[4]:= card[domain[funpart[x]]]

Out[4]= card[funpart[x]]
```

classes of restrictions

The related function **IMAGE[composite[id[x], inverse[FIRST]]]** is useful for discussing classes of restrictions of **x**. Its range is the class of all restrictions of **x**.

```
In[5]:= range[IMAGE[composite[id[x], inverse[FIRST]]]]

Out[5]= RS[x]
```

Theorem. A simplification rule.

```
In[6]:= Assoc[id[P[x]], id[RS[x]], IMAGE[composite[id[x], inverse[FIRST]]]]

Out[6]= composite[id[P[x]], IMAGE[composite[id[x], inverse[FIRST]]]] ==
         IMAGE[composite[id[x], inverse[FIRST]]]

In[7]:= composite[id[P[x_]], IMAGE[composite[id[x_], inverse[FIRST]]]] :=
         IMAGE[composite[id[x], inverse[FIRST]]]
```

Observation. The domain of the function **IMAGE[composite[id[x], inverse[FIRST]]]** is **V** if **x** is thin, which is the case when **x** is a function.

```
In[8]:= domain[IMAGE[composite[id[x], inverse[FIRST]]]]

Out[8]= P[domain[VERTSECT[x]]]
```

Observation. The class of all restrictions of **funpart[x]** to sets belonging to a class **y** is:

```
In[9]:= class[z, exists[t, and[member[t, y], equal[z, composite[funpart[x], id[t]]]]]]

Out[9]= image[IMAGE[composite[id[funpart[x]], inverse[FIRST]]], y]
```

Rewrite rules corresponding to this observation will now be derived.

Theorem. Any member of the class **image[IMAGE[composite[id[x], inverse[FIRST]]], y]** is a restriction of **x**.

```
In[10]:= SubstTest[implies, member[w, image[t, y]], member[w, range[t]],
           t -> IMAGE[composite[id[x], inverse[FIRST]]]] // Reverse

Out[10]= or[equal[w, composite[x, id[domain[w]]]],
           not[member[w, image[IMAGE[composite[id[x], inverse[FIRST]]], y]]]] == True

In[11]:= or[equal[w_, composite[x_, id[domain[w_]]]],
           not[member[w_, image[IMAGE[composite[id[x_], inverse[FIRST]]], y_]]]] := True
```

To obtain an implication in the reverse direction, the following lemma proves to be useful.

Theorem. A general result about **IMAGE[oopart[x]]**.

```
In[12]:= composite[inverse[IMAGE[oopart[x]]], IMAGE[oopart[x]]] // RelnRenormality

Out[12]= composite[inverse[IMAGE[oopart[x]]], IMAGE[oopart[x]]] ==
          composite[inverse[IMAGE[id[domain[oopart[x]]]]], IMAGE[id[domain[oopart[x]]]]]

In[13]:= composite[inverse[IMAGE[oopart[x_]]], IMAGE[oopart[x_]]] :=
          composite[inverse[IMAGE[id[domain[oopart[x]]]]], IMAGE[id[domain[oopart[x]]]]]
```

Corollary. Application to the case of **IMAGE[composite[id[funpart[x]], inverse[FIRST]]]**.

```
In[14]:= SubstTest[composite, inverse[IMAGE[oopart[t]]], IMAGE[oopart[t]],
           t → composite[id[funpart[x]], inverse[FIRST]]] // Reverse

Out[14]= composite[inverse[IMAGE[composite[id[funpart[x]], inverse[FIRST]]]],
           IMAGE[composite[id[funpart[x]], inverse[FIRST]]]] ==
          composite[inverse[IMAGE[id[domain[funpart[x]]]]], IMAGE[id[domain[funpart[x]]]]]

In[15]:= composite[inverse[IMAGE[composite[id[funpart[x_]], inverse[FIRST]]]],
           IMAGE[composite[id[funpart[x_]], inverse[FIRST]]]] :=
          composite[inverse[IMAGE[id[domain[funpart[x]]]]], IMAGE[id[domain[funpart[x]]]]]
```

When **funpart[x]** is total, this implies that the function **IMAGE[composite[id[funpart[x]], inverse[FIRST]]]** is one-to-one. More generally, a certain restriction of this function is one-to-one.

Theorem. The restriction of the function **IMAGE[composite[id[funpart[x]], inverse[FIRST]]]** to **P[domain[funpart[x]]]** is one-to-one.

```
In[16]:= SubstTest[FUNCTION, composite[id[P[domain[oopart[t]]]], inverse[IMAGE[oopart[t]]]],
           t -> composite[id[funpart[x]], inverse[FIRST]]] // Reverse

Out[16]= FUNCTION[composite[id[P[domain[funpart[x]]]],
            inverse[IMAGE[composite[id[funpart[x]], inverse[FIRST]]]]]] == True

In[17]:= FUNCTION[composite[id[P[domain[funpart[x_]]]],
            inverse[IMAGE[composite[id[funpart[x_]], inverse[FIRST]]]]]] := True
```

Theorem. If **y ∈ z,** then the restriction of **funpart[x]** to **y** belongs to **image[IMAGE[composite[id[funpart[x]], inverse[-FIRST]]], z]**.

```
In[18]:= Map[implies[member[y, z], member[y, #]] &,
           ImageComp[inverse[IMAGE[composite[id[funpart[x]], inverse[FIRST]]]],
            IMAGE[composite[id[funpart[x]], inverse[FIRST]]], z]] // Reverse

Out[18]= or[member[composite[funpart[x], id[y]], image[
             IMAGE[composite[id[funpart[x]], inverse[FIRST]]], z]], not[member[y, z]]] == True

In[19]:= or[member[composite[funpart[x_], id[y_]],
            image[IMAGE[composite[id[funpart[x_]], inverse[FIRST]]], z_]],
           not[member[y_, z_]]] := True
```

Corollary. (Obtained by eliminating the **funpart** wrapper.)

```
In[20]:= SubstTest[implies, equal[x, funpart[t]],
           or[member[composite[x, id[y]], image[IMAGE[composite[id[x], inverse[FIRST]]], z]],
            not[member[y, z]]], t → x] // Reverse

Out[20]= or[member[composite[x, id[y]], image[IMAGE[composite[id[x], inverse[FIRST]]], z]],
           not[FUNCTION[x]], not[member[y, z]]] == True

In[21]:= or[member[composite[x_, id[y_]], image[IMAGE[composite[id[x_], inverse[FIRST]]], z_]],
           not[FUNCTION[x_]], not[member[y_, z_]]] := True
```

cutting down the class of restrictions

Theorem. One can always cut down the class **y** to a subclass of **P[domain[x]]**.

```
In[22]:= ImageComp[IMAGE[composite[id[x], inverse[FIRST]]], IMAGE[id[domain[x]]], y] // Reverse

Out[22]= image[IMAGE[composite[id[x], inverse[FIRST]]], image[IMAGE[id[domain[x]]], y]] ==
          image[IMAGE[composite[id[x], inverse[FIRST]]], y]

In[23]:= image[IMAGE[composite[id[x_], inverse[FIRST]]], image[IMAGE[id[domain[x_]]], y_]] :=
          image[IMAGE[composite[id[x], inverse[FIRST]]], y]
```

Corollary.

```
In[24]:= ImageComp[IMAGE[composite[id[x], inverse[FIRST]]], IMAGE[id[domain[x]]], V] // Reverse

Out[24]= image[IMAGE[composite[id[x], inverse[FIRST]]], P[domain[x]]] == RS[x]

In[25]:= image[IMAGE[composite[id[x_], inverse[FIRST]]], P[domain[x_]]] := RS[x]
```