image[inverse[PS], -] does not have cardinality 2

Johan G. F. Belinfante
2009 June 6

In[1]:= **SetDirectory["l:"]; << goedel.09jun05a;<< tools.m**

```
:Package Title: goedel.09jun05a                    2009 June 5 at 10:05 p.m.

It is now:  2009 Jun 6 at 15:32

Loading Simplification Rules

TOOLS.M                                        Revised 2009 June 1

weightlimit = 40
```

---

summary

The theorems derived in this notebook were suggested by the results obtained for low-rank examples. The constructor **ens[n]** was used to recursively generate hundreds of low-rank sets. For each of these sets one generates a set of the form **image[inverse[PS], x]**. Many of these are the same. The first hundred low-rank sets only yield the following eight examples.

In[2]:= **Map[image[inverse[PS], ens[#]] &, Range[0, 100]] // Union // TableForm**

Out[2]//TableForm=

```
0
set[0]
union[set[0], succ[set[set[0]]]]
union[set[0], succ[set[set[set[0]]]]]
union[set[set[set[set[0]]]], succ[set[0]]]
union[succ[set[0]], succ[set[set[set[0]]]]]
union[set[0], succ[set[set[0]]], succ[set[set[set[0]]]]]
union[set[set[set[set[0]]]], succ[set[0]], succ[set[set[0]]]]
```

Observe that for these examples one finds cardinalities 0, 1, 3 and 4, but not 2.

In[3]:= **Map[card, %] // Union**

Out[3]= `{0, set[0], succ[succ[set[0]]], succ[succ[succ[set[0]]]]}`

Upon reflection, it is not hard to understand why examples with cardinality 2 do not appear. If any member of **x** has more than one element, then that element alone contributes three elements to **image[inverse[PS], x]**, two of which are singletons, and if there are no such elements, then **image[inverse[PS], x]** is either **0** or **set[0]**. In this notebook a formal version of this argument is presented.

A similar result was found for the rank of **image[inverse[PS], x]**. No example with rank 2 was found.

```
In[4]:= Map[normalize[rank[#]] &, %%] // Union

Out[4]= {0, set[0], succ[succ[set[0]]], succ[succ[succ[set[0]]]]}
```

The latter result is somewhat less interesting because there are in fact only two sets with rank 2, namely **set[set[0]]** and **succ[set[0]]**.

```
In[5]:= image[inverse[RANK], set[succ[set[0]]]]

Out[5]= set[set[set[0]], succ[set[0]]]
```

The first is impossible because any set of the form **y = image[inverse[PS], x]** must satisfy **image[inverse[S], y] = y**, while the second is ruled out because it has cardinality 2.

the empty set

The only two sets **x** for which **image[inverse[PS], x]** is empty are **0** and **set[0]**.

```
In[6]:= empty[image[inverse[PS], x]]

Out[6]= subclass[x, set[0]]
```

Two variable-free expressions of this fact can be given. One of these is already available:

```
In[7]:= image[inverse[IMAGE[inverse[PS]]], set[0]]

Out[7]= succ[set[0]]
```

Note that the only sets satisfying **x ⊂ set[0]** are **x = 0** and **x = set[0]**. The pairset of these two cases is **set[0, set[0]] = succ[set[0]]**.

```
In[8]:= class[x, subclass[x, set[0]]]

Out[8]= succ[set[0]]
```

Corollary. If **x ⊂ set[0]**, then **image[inverse[PS], x]** = **0**.

```
In[9]:= ImageComp[IMAGE[inverse[PS]], inverse[IMAGE[inverse[PS]]], set[0]] // Reverse

Out[9]= image[IMAGE[inverse[PS]], succ[set[0]]] == set[0]

In[10]:= image[IMAGE[inverse[PS]], succ[set[0]]] := set[0]
```

the case of singletons

The only singleton among the examples of sets of the form **image[inverse[PS], x]** is **set[0]**. In this section it is shown that this is a theorem. There are actually many different sets **x** for which **image[inverse[PS], x]** = **set[0]**. The simplest of these is **x = set[set[0]]**.

```
In[11]:= image[inverse[PS], set[set[0]]]
```

```
Out[11]= set[0]
```

The class of all sets **x** for which **image[inverse[PS], x]** = **set[0]** is given by the following somewhat complicated expression:

```
In[12]:= image[inverse[IMAGE[inverse[PS]]], set[set[0]]] // Renormality
```

```
Out[12]= image[inverse[IMAGE[inverse[PS]]], set[set[0]]] ==
          intersection[complement[succ[set[0]]], P[union[range[SINGLETON], set[0]]]]
```

A simpler description is obtained for that class of sets x satisfying the weaker condition **image[inverse[PS], x]** ⊂ **set[0]**.

Theorem. The class **image[inverse[PS], x]** is either **0** or **set[0]** if and only if every nonempty member of **x** is a singleton.

```
In[13]:= image[inverse[IMAGE[inverse[PS]]], succ[set[0]]] // Renormality
```

```
Out[13]= image[inverse[IMAGE[inverse[PS]]], succ[set[0]]] == P[union[range[SINGLETON], set[0]]]
```

```
In[14]:= image[inverse[IMAGE[inverse[PS]]], succ[set[0]]] := P[union[range[SINGLETON], set[0]]]
```

Lemma. The singleton **set[0]** is a set of the form **image[inverse[PS], x]**.

```
In[15]:= member[set[0], range[IMAGE[inverse[PS]]]] // AssertTest
```

```
Out[15]= member[set[0], range[IMAGE[inverse[PS]]]] == True
```

```
In[16]:= member[set[0], range[IMAGE[inverse[PS]]]] := True
```

Corollary.

```
In[17]:= ImageComp[IMAGE[inverse[PS]], inverse[IMAGE[inverse[PS]]], succ[set[0]]] // Reverse
```

```
Out[17]= image[IMAGE[inverse[PS]], P[union[range[SINGLETON], set[0]]]] == succ[set[0]]
```

```
In[18]:= image[IMAGE[inverse[PS]], P[union[range[SINGLETON], set[0]]]] := succ[set[0]]
```

The key to explaining why **set[0]** is the only singleton found among the examples of sets of the form **image[inverse[PS], x]** is provided by the observation that if **image[inverse[PS], x]** is not empty, then **0** is one of its members.

```
In[19]:= implies[not[empty[image[inverse[PS], x]]], member[0, image[inverse[PS], x]]]
```

```
Out[19]= True
```

Theorem. If **image[inverse[PS], x]** is a singleton, then it must be **set[0]**. (Comment. To prevent rewrite rules kicking in unexpectedly and obscuring the argument, control is provided by introducing a temporary variable **t** which is eliminated at the end.)

```
In[20]:= Map[not,
           SubstTest[and, implies[p1, p2], implies[and[p0, p2], p3], implies[and[p1, p3], p4],
             not[implies[and[p0, p1], p4]], {p0 → equal[t, image[inverse[PS], x]],
              p1 → member[t, range[SINGLETON]], p2 → not[empty[t]], p3 → member[0, t],
              p4 → equal[t, set[0]]}] /. t → image[inverse[PS], x]] // Reverse

Out[20]= or[equal[image[inverse[PS], x], set[0]],
           not[member[image[inverse[PS], x], range[SINGLETON]]]] == True

In[21]:= or[equal[image[inverse[PS], x_], set[0]],
           not[member[image[inverse[PS], x_], range[SINGLETON]]]] := True
```

A variable-free expression of this can be obtained. The following lemma does this in the form of an inclusion which will later be replaced by a a simpler equation.

Lemma.

```
In[22]:= Map[equal[V, #] &,
          SubstTest[class, x, or[equal[image[s, x], set[0]], not[member[image[s, x], r]]],
           {r → range[SINGLETON], s → inverse[PS]}]]

Out[22]= subclass[U[image[inverse[IMAGE[inverse[PS]]], range[SINGLETON]]],
           union[range[SINGLETON], set[0]]] == True

In[23]:= % /. Equal → SetDelayed
```

One can clean this up by replacing the inclusion of the form **U[x] ⊂ y** with the equivalent inclusion **x ⊂ P[y]**.

Lemma.

```
In[24]:= SubstTest[implies, subclass[u, v], subclass[image[t, u], image[t, v]],
           {t → composite[id[range[SINGLETON]], IMAGE[inverse[PS]]],
            u -> image[inverse[IMAGE[inverse[PS]]], range[SINGLETON]],
            v → P[union[range[SINGLETON], set[0]]]}] // Reverse

Out[24]= subclass[intersection[range[SINGLETON], range[IMAGE[inverse[PS]]]], set[set[0]]] ==
          True

In[25]:= subclass[
           intersection[range[SINGLETON], range[IMAGE[inverse[PS]]]], set[set[0]]] := True
```

This inclusion is actually an equation that can be made into a rewrite rule.

Theorem.

```
In[26]:= equal[intersection[range[SINGLETON], range[IMAGE[inverse[PS]]]], set[set[0]]]

Out[26]= True

In[27]:= intersection[range[SINGLETON], range[IMAGE[inverse[PS]]]] := set[set[0]]
```

the case of cardinality 2

To show that no set of the form **image[inverse[PS], x]** can have cardinality 2, it will be supposed that such a set does exist, and a contradiction derived from this hypothesis.

Lemma. A set with two elements cannot be **0** or **set[0]**.

```
In[28]:= SubstTest[implies, and[member[x, y], subclass[y, z]], member[x, z],
           {y → image[PAIRSET, Di], z → complement[succ[set[0]]]}] // Reverse

Out[28]= or[not[member[x, image[PAIRSET, Di]]], not[subclass[x, set[0]]]] == True

In[29]:= or[not[member[x_, image[PAIRSET, Di]]], not[subclass[x_, set[0]]]] := True
```

To apply this lemma to the case of **image[inverse[PS], x]** one needs the following lemma.

Lemma.

```
In[30]:= subclass[image[inverse[PS], x], set[0]] // AssertTest

Out[30]= subclass[image[inverse[PS], x], set[0]] == subclass[x, union[range[SINGLETON], set[0]]]

In[31]:= subclass[image[inverse[PS], x_], set[0]] :=
          subclass[x, union[range[SINGLETON], set[0]]]
```

Theorem. If the cardinality of **image[inverse[PS], x]** were equal to 2, then some element of **x** would have to have more than one member.

```
In[32]:= SubstTest[implies, member[t, image[PAIRSET, Di]],
           not[subclass[t, set[0]]], t -> image[inverse[PS], x]] // Reverse

Out[32]= or[not[member[image[inverse[PS], x], image[PAIRSET, Di]]],
           not[subclass[x, union[range[SINGLETON], set[0]]]]] == True

In[33]:= (% /. x → x_) /. Equal → SetDelayed
```

The key to deriving exactly the opposite conclusion from the same hypothesis is the observation that for each member **s** of a non-singleton member **t** of a class **x** one obtains a singleton member **set[s]** of the class **image[inverse[PS], x]**. The following lemma provides this fact. (The derivation of this lemma is fiendishly sneaky.)

Lemma. If **t ∈ x** is not a singleton, and if **s ∈ t,** then **set[s] ∈ image[inverse[PS], x]**.

```
In[34]:= Map[implies[member[t, V], #] &,
          SubstTest[member, t, union[complement[x], image[V, intersection[w, set[set[s]]]],
            P[complement[set[s]]]], range[SINGLETON]], w → image[inverse[PS], x]]]

Out[34]= or[member[t, range[SINGLETON]],
           member[set[s], image[inverse[PS], x]], not[member[s, t]], not[member[t, x]]] == True

In[35]:= (% /. {s → s_, t → t_, x → x_}) /. Equal → SetDelayed
```

Eliminating the variable **s** yields the following statement.

Theorem.

```
In[36]:= Map[equal[V, #] &, SubstTest[class, s,
           or[member[t, r], member[set[s], w], not[member[s, t]], not[member[t, x]]],
           {r → range[SINGLETON], w → image[inverse[PS], x]}]]

Out[36]= or[member[t, range[SINGLETON]], not[member[t, x]],
           subclass[image[SINGLETON, t], image[inverse[PS], x]]] == True

In[37]:= (% /. {t → t_, x → x_}) /. Equal → SetDelayed
```

If the cardinality of **image[inverse[PS], x]** were 2, then one would expect one of these two elements to be **0**. Removing the element **0** leaves a set with only one element, so there can be only one singleton member of **image[inverse[PS], x]**.

Lemma.

```
In[38]:= Map[implies[member[t, x], #] &, SubstTest[implies, and[subclass[u, v], member[v, w]],
            member[u, image[inverse[S], w]], {u → image[SINGLETON, t],
             v → dif[image[inverse[PS], x], set[0]], w → range[SINGLETON]}]] // Reverse

Out[38]= or[member[t, range[SINGLETON]], not[member[t, x]], not[
            member[intersection[complement[set[0]], image[inverse[PS], x]], range[SINGLETON]]],
           not[subclass[image[SINGLETON, t], image[inverse[PS], x]]], subclass[t, 0]] == True

In[39]:= (% /. {t → t_, x → x_}) /. Equal → SetDelayed
```

Lemma. If the cardinality of **image[inverse[PS],x]** were equal to 2, then **dif[image[inverse[PS],x], set[0]]** would be a singleton.

```
In[40]:= SubstTest[implies, and[member[0, t], member[t, image[PAIRSET, Di]]],
           member[dif[t, set[0]], range[SINGLETON]], t → image[inverse[PS], x]] // Reverse

Out[40]= or[member[intersection[complement[set[0]], image[inverse[PS], x]], range[SINGLETON]],
           not[member[image[inverse[PS], x], image[PAIRSET, Di]]], subclass[x, set[0]]] == True

In[41]:= (% /. x → x_) /. Equal → SetDelayed
```

Lemma. If the cardinality of **image[inverse[PS],x]** were equal to 2, then **x** cannot be **0** or **set[0]**.

```
In[42]:= SubstTest[implies, member[t, image[PAIRSET, Di]],
           not[empty[t]], t → image[inverse[PS], x]] // Reverse

Out[42]= or[not[member[image[inverse[PS], x], image[PAIRSET, Di]]],
           not[subclass[x, set[0]]]] == True

In[43]:= (% /. x → x_) /. Equal → SetDelayed
```

Lemma.

```
In[44]:= Map[not, SubstTest[and, implies[p1, p2], implies[and[p1, p2], p3], implies[p3, p4],
           not[implies[p1, p4]], {p1 -> member[image[inverse[PS], x], image[PAIRSET, Di]],
            p2 → not[subclass[x, set[0]]], p3 -> member[
              intersection[complement[set[0]], image[inverse[PS], x]], range[SINGLETON]],
            p4 -> or[member[t, range[SINGLETON]], not[member[t, x]], not[subclass[
                image[SINGLETON, t], image[inverse[PS], x]]], subclass[t, 0]]}]] // Reverse

Out[44]= or[member[t, range[SINGLETON]], not[member[t, x]],
          not[member[image[inverse[PS], x], image[PAIRSET, Di]]],
          not[subclass[image[SINGLETON, t], image[inverse[PS], x]]], subclass[t, 0]] == True

In[45]:= (% /. {t → t_, x → x_}) /. Equal → SetDelayed
```

Combining these results yields:

Lemma.

```
In[46]:= Map[not, SubstTest[and, implies[p1, p3], implies[and[p1, p2, p3], p4],
           not[implies[and[p1, p2], p4]], {p1 → member[t, dif[x, range[SINGLETON]]],
            p2 -> member[image[inverse[PS], x], image[PAIRSET, Di]],
            p3 -> subclass[image[SINGLETON, t], image[inverse[PS], x]],
            p4 -> or[member[t, range[SINGLETON]], subclass[t, 0]]}]] // Reverse

Out[46]= or[member[t, range[SINGLETON]], not[member[t, x]],
          not[member[image[inverse[PS], x], image[PAIRSET, Di]]], subclass[t, 0]] == True

In[47]:= (% /. {t → t_, x → x_}) /. Equal → SetDelayed
```

Theorem. If the cardinality of **image[inverse[PS], x]** were equal to 2, then the only possible members of **x** are **0** and singletons.

```
In[48]:= Map[equal[V, #] &, SubstTest[class, t,
          or[subclass[t, 0], member[t, r], not[member[t, x]], not[member[u, v]]],
          {u -> image[inverse[PS], x], v → image[PAIRSET, Di], r → range[SINGLETON]}]]

Out[48]= or[not[member[image[inverse[PS], x], image[PAIRSET, Di]]],
          subclass[x, union[range[SINGLETON], set[0]]]] == True

In[49]:= (% /. x → x_) /. Equal → SetDelayed
```

Under the same hypothesis, the opposite conclusion was obtained previously, producing the desired contradiction.

Theorem. The class **image[inverse[PS], x]** can not have cardinality 2.

```
In[50]:= Map[not, SubstTest[and, implies[p1, p2], implies[p1, not[p2]],
          {p1 -> member[image[inverse[PS], x], image[PAIRSET, Di]],
           p2 -> subclass[x, union[range[SINGLETON], set[0]]]}]]

Out[50]= member[image[inverse[PS], x], image[PAIRSET, Di]] == False

In[51]:= member[image[inverse[PS], x_], image[PAIRSET, Di]] := False
```

A variable-free formulation can be given.

Theorem.

```
In[52]:= Map[image[IMAGE[inverse[PS]], #] &,
          image[inverse[IMAGE[inverse[PS]]], image[PAIRSET, Di]] // Normality]

Out[52]= intersection[image[PAIRSET, Di], range[IMAGE[inverse[PS]]]] == 0

In[53]:= intersection[image[PAIRSET, Di], range[IMAGE[inverse[PS]]]] := 0
```