cardinality of the range of a function

Johan G. F. Belinfante
2012 June 11

In[1]:= **SetDirectory["l:"]; << goedel.12jun10a**

```
:Package Title: goedel.12jun10a                    2012 June 10 at 6:20 p.m.

Loading takes about seventeen minutes, half that time due to builtin pauses.

It is now:  2012 Jun 11 at 12:37

Loading Simplification Rules

TOOLS.M is now incorporated in the GOEDEL program as of 2010 September 3

weightlimit = 40

Loading completed.

It is now:  2012 Jun 11 at 12:54
```

---

summary

If the axiom of choice holds, then the range of any small function is equipollent to a subset of its domain. The following is a variable free statement of this fact.

In[2]:= **implies[axch, subclass[image[DORA, FUNS], composite[Q, inverse[S]]]]**

Out[2]= True

In the **GOEDEL** program, the axiom of choice is not automatically assumed to hold, but one can obtain a similar result if one adds a further hypothesis. In particular, it was shown 2005 November 20 in the posted notebook **acdorafu.nb** that the range of a function is equipollent to a subset of its domain if the inverse of the function admits a cross-section. (A cross-section of a set **x** is a function that is a subset of **x** and has the same domain as **x**.) The following result is a simplified version of this result. Here the class **SELECT** holds all sets that admit a cross-section.

In[3]:= **subclass[image[DORA, intersection[FUNS, image[INVERSE, SELECT]]], composite[Q, inverse[S]]]**

Out[3]= True

In this notebook a corollary of this is derived that is expected to be useful for applications to arithmetic. In particular, it is shown that if the domain of a function is countable, then so is its range.

derivation

The axiom of choice is equivalent to the statement that every set is equipollent to an ordinal:

```
In[4]:= equal[image[Q, OMEGA], V]

Out[4]= axch
```

It is natural to replace the axiom of choice by a hypothesis that some particular set is equipollent to an ordinal. It will be recalled that any set whose range is equipollent to an ordinal admits a cross-section.

```
In[5]:= subclass[image[inverse[IMAGE[SECOND]], image[Q, OMEGA]], SELECT]

Out[5]= True
```

Lemma.

```
In[46]:= Map[subclass[#, image[inverse[IMAGE[SWAP]], SELECT]] &,
          IminComp[IMAGE[SECOND], IMAGE[SWAP], image[Q, OMEGA]]]

Out[46]= subclass[image[IMAGE[SWAP], image[inverse[IMAGE[FIRST]], image[Q, OMEGA]]], SELECT] ==
          True

In[47]:= % /. Equal → SetDelayed
```

Theorem.

```
In[49]:= SubstTest[implies, and[subclass[u, v], subclass[v, w]],
           subclass[u, w], {u -> image[inverse[IMAGE[FIRST]], image[Q, OMEGA]],
            v -> image[inverse[IMAGE[SWAP]], SELECT], w -> union[complement[FUNS],
              image[inverse[DORA], composite[Q, inverse[S]]]]}] // Reverse

Out[49]= subclass[image[DORA, U[image[MAP, cart[image[Q, OMEGA], V]]]],
           composite[Q, inverse[S]]] == True

In[50]:= subclass[image[DORA, U[image[MAP, cart[image[Q, OMEGA], V]]]],
           composite[Q, inverse[S]]] := True
```

Corollary.

```
In[55]:= SubstTest[implies, and[member[x, y], subclass[y, z]],
           member[x, z], {y -> U[image[MAP, cart[image[Q, OMEGA], V]]],
            z → image[inverse[DORA], composite[Q, inverse[S]]]}] // Reverse

Out[55]= or[member[range[x], image[Q, P[domain[x]]]], not[FUNCTION[x]],
           not[member[x, V]], not[member[domain[x], image[Q, OMEGA]]]] == True

In[56]:= (% /. x → x_) /. Equal → SetDelayed
```

The sethood hypothesis here is redundant, and will now be removed.

Theorem. If the domain of a function is equipollent to an ordinal, then its range is equipollent to a subset of its domain.

```
In[59]:= Map[not, SubstTest[and, implies[and[p1, p2], p3], not[implies[and[p1, p2], p4]],
           {p1 → FUNCTION[x], p2 -> member[domain[x], image[Q, OMEGA]], p3 → member[x, V],
            p4 -> member[range[x], image[Q, P[domain[x]]]]}]] // Reverse

Out[59]= or[member[range[x], image[Q, P[domain[x]]]],
          not[FUNCTION[x]], not[member[domain[x], image[Q, OMEGA]]]] == True

In[60]:= or[member[range[x_], image[Q, P[domain[x_]]]],
          not[FUNCTION[x_]], not[member[domain[x_], image[Q, OMEGA]]]] := True
```

Corollary. If the domain of a function is equipollent to an ordinal, then its range is equipollent to a subset of the function.

```
In[70]:= Map[not, SubstTest[and, implies[and[p1, p2], p3], implies[and[p1, p2], p4],
           implies[and[p1, p4], p5], implies[and[p3, p5], p6], not[implies[and[p1, p2], p6]],
           {p1 -> FUNCTION[x], p2 -> member[domain[x], image[Q, OMEGA]],
            p3 -> member[range[x], image[Q, P[domain[x]]]], p4 → member[x, V], p5 →
             member[pair[domain[x], x], Q], p6 -> member[range[x], image[Q, P[x]]]}]] // Reverse

Out[70]= or[member[range[x], image[Q, P[x]]],
          not[FUNCTION[x]], not[member[domain[x], image[Q, OMEGA]]]] == True

In[71]:= or[member[range[x_], image[Q, P[x_]]],
          not[FUNCTION[x_]], not[member[domain[x_], image[Q, OMEGA]]]] := True
```

Theorem. A variable-free restatement of the corollary.

```
In[72]:= Map[equal[V, domain[#]] &,
          SubstTest[reify, x, case[or[member[range[x], image[Q, P[x]]], not[FUNCTION[x]],
            not[member[domain[x], t]]]], t -> image[Q, OMEGA]]]

Out[72]= subclass[U[image[MAP, cart[image[Q, OMEGA], V]]],
          fix[composite[Q, S, IMAGE[SECOND]]]] == True

In[73]:= subclass[U[image[MAP, cart[image[Q, OMEGA], V]]],
          fix[composite[Q, S, IMAGE[SECOND]]]] := True
```

functions with countable domain

A set is **countable** if it is finite or countably infinite. The **GOEDEL** program rewrites this statement as follows:

```
In[77]:= or[member[x, FINITE], equal[card[x], omega]]

Out[77]= not[member[omega, card[x]]]
```

Theorem. If the domain of a function is countably infinite, then its range is countable.

```
In[76]:= Map[not, SubstTest[and, implies[p2, p3], implies[and[p1, p3], p4],
            implies[and[p2, p4], p5], not[implies[and[p1, p2], p5]], {p1 → FUNCTION[x],
             p2 → equal[card[domain[x]], omega], p3 → member[domain[x], image[Q, OMEGA]],
             p4 → member[range[x], image[Q, P[domain[x]]]],
             p5 -> member[range[x], image[Q, P[omega]]]}]] // Reverse

Out[76]= or[not[equal[omega, card[domain[x]]]],
          not[FUNCTION[x]], not[member[omega, card[range[x]]]]] == True

In[78]:= or[not[equal[omega, card[domain[x_]]]],
          not[FUNCTION[x_]], not[member[omega, card[range[x_]]]]] := True
```

Lemma. If a function is finite, then its range is countable.

```
In[85]:= Map[not, SubstTest[and, implies[and[p1, p2], p3], implies[p3, p4],
            not[implies[and[p1, p2], p4]], {p1 → FUNCTION[x], p2 → member[domain[x], FINITE],
             p3 → member[range[x], FINITE], p4 -> not[member[omega, card[range[x]]]]}]] // Reverse

Out[85]= or[not[FUNCTION[x]], not[member[omega, card[range[x]]]], not[member[x, FINITE]]] == True

In[86]:= (% /. x → x_) /. Equal → SetDelayed
```

Note that in the above lemma, the following rewrite rule kicked in automatically:

```
In[87]:= and[FUNCTION[x], member[domain[x], FINITE]]

Out[87]= and[FUNCTION[x], member[x, FINITE]]
```

Theorem. If the domain of a function is countable, then so is its range.

```
In[89]:= Map[not, SubstTest[and, implies[p1, or[p3, p4]], not[implies[and[p1, p2], p5]],
            {p1 → not[member[omega, card[domain[x]]]], p2 → FUNCTION[x],
             p3 → member[domain[x], FINITE], p4 → equal[omega, card[domain[x]]],
             p5 -> not[member[omega, card[range[x]]]]}]] // Reverse

Out[89]= or[member[omega, card[domain[x]]],
          not[FUNCTION[x]], not[member[omega, card[range[x]]]]] == True

In[90]:= or[member[omega, card[domain[x_]]],
          not[FUNCTION[x_]], not[member[omega, card[range[x_]]]]] := True
```

It is to be emphasized once again that all of the results derived in this notebook are true independently of whether or not the axiom of choice holds.