monotonicity of CARD

Johan G. F. Belinfante
2008 May 17

```
In[1]:= SetDirectory["l:"]; << goedel.08may15a; << tools.m

        :Package Title: goedel.08may15a                  2008 May 15 at 2:05 p.m.

        It is now:  2008 May 17 at 22:28

        Loading Simplification Rules

        TOOLS.M                                          Revised 2008 February 12

        weightlimit = 40
```

summary

In this notebook it is shown that the function **CARD** is monotone.

use of the wob wrapper

The axiom of choice is not automatically assumed in the **GOEDEL** program. Because of this, the cardinality function is not total. Its domain is the class of sets that are equipollent to some ordinal:

```
In[2]:= domain[CARD]

Out[2]= image[Q, OMEGA]
```

Many theorems about cardinality require the axiom of choice, but the monotonicity of **CARD** does not. Since cardinals are defined as initial ordinals, they automatically satisfy dichotomy, but the relation **Q** ∘ **S** need not. The monotonicity theorem amounts to the statement that a subset **x** of a set **y** cannot have greater cardinality than that of **y** when both **x** and **y** are equipollent to ordinals. The main ingredients of the proof are dichotomy for cardinals and the Schröder-Bernstein theorem for the relation **Q** ∘ **S**. The condition that **x** and **y** are both equipollent to ordinals is most conveniently imposed by using the **wob** wrapper. This wrapper satisfies the following basic properties:

```
In[3]:= member[wob[x], image[Q, OMEGA]]

Out[3]= True

In[4]:= equal[x, wob[x]]

Out[4]= member[x, image[Q, OMEGA]]
```

derivation

Theorem. Since **S** is contained in **Q** ∘ **S**, one has:

```
In[5]:= Map[implies[member[y, z], #] &, SubstTest[implies, and[member[u, v], subclass[v, w]],
           member[u, w], {u → pair[x, y], v → S, w → composite[Q, S]}]] // Reverse

Out[5]= or[member[x, image[Q, P[y]]], not[member[y, z]], not[subclass[x, y]]] == True

In[6]:= or[member[x_, image[Q, P[y_]]], not[member[y_, z_]], not[subclass[x_, y_]]] := True
```

Corollary. The sethood hypothesis is not needed if **y** is wrapped with **wob**.

```
In[7]:= SubstTest[implies, and[member[u, v], subclass[v, w]],
          member[u, w], {u → pair[x, wob[y]], v → S, w → composite[Q, S]}] // Reverse

Out[7]= or[member[x, image[Q, P[wob[y]]]], not[subclass[x, wob[y]]]] == True

In[8]:= or[member[x_, image[Q, P[wob[y_]]]], not[subclass[x_, wob[y_]]]] := True
```

Lemma. A monotonicity property for cardinality, using **wob** wrappers. (Note that **CARD** is contained in **Q**.)

```
In[9]:= SubstTest[implies, and[member[u, v], subclass[v, w]],
          member[u, w], {u → pair[wob[x], wob[y]],
           v → composite[inverse[CARD], S, CARD], w → composite[Q, S]}] // Reverse

Out[9]= or[member[card[wob[y]], card[wob[x]]], member[wob[x], image[Q, P[wob[y]]]]] == True

In[10]:= (% /. {x → x_, y → y_}) /. Equal → SetDelayed
```

Lemma. A corollary of Schröder-Bernstein theorem.

```
In[11]:= SubstTest[implies, subclass[u, v], subclass[image[t, u], image[t, v]],
           {t → id[composite[Q, S]], u -> composite[inverse[CARD], inverse[S], CARD],
            v -> composite[Q, inverse[S]]}] // Reverse

Out[11]= subclass[
           intersection[composite[Q, S], composite[inverse[CARD], inverse[S], CARD]], Q] == True

In[12]:= % /. Equal → SetDelayed
```

Lemma. A monotonicity property for cardinality, using **wob** wrappers, in the reverse direction.

```
In[13]:= SubstTest[implies, and[member[u, v], subclass[v, w]],
           member[u, w], {u → pair[wob[x], wob[y]], v → intersection[composite[Q, S],
              composite[inverse[CARD], inverse[S], CARD]], w → Q}] // Reverse

Out[13]= or[not[member[card[wob[y]], card[wob[x]]]],
           not[member[wob[x], image[Q, P[wob[y]]]]]] == True

In[14]:= (% /. {x → x_, y → y_}) /. Equal → SetDelayed
```

Theorem. (Combining the implications in both directions.)

```
In[15]:= equiv[member[wob[x], image[Q, P[wob[y]]]], not[member[card[wob[y]], card[wob[x]]]]]

Out[15]= True

In[16]:= member[wob[x_], image[Q, P[wob[y_]]]] := not[member[card[wob[y]], card[wob[x]]]]
```

Corollary.

```
In[17]:= SubstTest[implies, subclass[t, wob[y]],
           member[t, image[Q, P[wob[y]]]], t → wob[x]] // Reverse

Out[17]= or[not[member[card[wob[y]], card[wob[x]]]], not[subclass[wob[x], wob[y]]]] == True

In[18]:= or[not[member[card[wob[y_]], card[wob[x_]]]], not[subclass[wob[x_], wob[y_]]]] := True
```

Corollary. (Removing the **wob** wrappers.)

```
In[19]:= SubstTest[implies, and[equal[x, wob[u]], equal[y, wob[v]], subclass[x, y]],
            not[member[card[y], card[x]]], {u → x, v → y}] // Reverse // MapNotNot

Out[19]= or[not[member[x, image[Q, OMEGA]]],
           not[member[card[y], card[x]]], not[subclass[x, y]]] == True

In[20]:= or[not[member[x_, image[Q, OMEGA]]],
           not[member[card[y_], card[x_]]], not[subclass[x_, y_]]] := True
```

Corollary. (The special case of ordinals.)

```
In[22]:= SubstTest[implies, subclass[wob[u], wob[v]],
           subclass[card[wob[u]], card[wob[v]]], {u → ord[y], v → ord[x]}] // Reverse

Out[22]= or[member[ord[x], ord[y]], not[member[card[ord[x]], card[ord[y]]]]] == True

In[23]:= or[member[ord[x_], ord[y_]], not[member[card[ord[x_]], card[ord[y_]]]]] := True
```

monotonicity of CARD

The monotonicity of the function **CARD** is derived in this section by eliminating the variables **x** and **y** from the theorem derived in the preceding section.

Lemma. (Temporary simplification rule.)

```
In[24]:= AssInt[S, composite[inverse[CARD], inverse[E], CARD],
           cartsq[image[Q, OMEGA]]] // Reverse

Out[24]= composite[id[image[Q, OMEGA]],
           intersection[S, composite[inverse[CARD], inverse[E], CARD]], id[image[Q, OMEGA]]] ==
          intersection[S, composite[inverse[CARD], inverse[E], CARD]]
```

```
In[25]:= % /. Equal → SetDelayed
```

Lemma. (Removing the variables **x** and **y**.)

```
In[26]:= Map[composite[Id, complement[#]] &, SubstTest[class, pair[x, y], or[not[member[x, t]],
            not[member[card[y], card[x]]], not[subclass[x, y]]], t → image[Q, OMEGA]]]

Out[26]= intersection[S, composite[inverse[CARD], inverse[E], CARD]] == 0

In[27]:= % /. Equal → SetDelayed
```

Lemma.

```
In[28]:= SubstTest[empty, dif[u, v],
          {u → intersection[S, cartsq[image[Q, OMEGA]]], v -> composite[inverse[CARD], S, CARD]}]

Out[28]= subclass[composite[id[image[Q, OMEGA]], S, id[image[Q, OMEGA]]],
           composite[inverse[CARD], S, CARD]] == True

In[29]:= % /. Equal → SetDelayed
```

Theorem. Monotonicity of **CARD**.

```
In[30]:= SubstTest[implies, subclass[u, v], subclass[image[t, u], image[t, v]],
           {t → cross[CARD, CARD], u -> composite[id[image[Q, OMEGA]], S, id[image[Q, OMEGA]]],
            v -> composite[inverse[CARD], S, CARD]}] // Reverse

Out[30]= subclass[composite[CARD, S, inverse[CARD]], S] == True

In[31]:= subclass[composite[CARD, S, inverse[CARD]], S] := True
```