Dedekind finiteness and finiteness

Johan G. F. Belinfante
2009 April 9

```
In[1]:= SetDirectory["l:"]; << goedel.09apr08a; << tools.m

        :Package Title: goedel.09apr08a            2009 April 8 at 11:30 a.m.

        It is now:  2009 Apr 9 at 14:27

        Loading Simplification Rules

        TOOLS.M                                    Revised 2009 April 6

        weightlimit = 40
```

summary

For ordinals one need not distinguish between finite and Dedekind finite. In general, in the absence of the axiom of choice, one only has an inclusion:

```
In[2]:= subclass[FINITE, DEDEKIND]

Out[2]= True
```

It is shown below that the reverse inclusion holds for ordinals even without assuming the axiom of choice. Comment: This fact was pointed out by Patrick Suppes in his book (page 225).

```
In[3]:= "Patrick Suppes, Axiomatic Set Theory, Dover Publications, Inc., New York, 1972.";
```

As a corollary, it follows that a set is finite if and only if it is Dedekind finite and equipollent to an ordinal.

derivation

The key fact needed is this:

Lemma.

```
In[4]:= SubstTest[implies, and[subclass[u, v], member[v, DEDEKIND]],
          member[u, DEDEKIND], {u → omega, v → ord[x]}] // Reverse

Out[4]= or[member[ord[x], omega], not[member[ord[x], DEDEKIND]]] == True

In[5]:= (% /. x → x_) /. Equal → SetDelayed
```

The converse also holds, and can be combined with the lemma to obtain a simple rewrite rule. (A similar rule holds with the class **DEDEKIND** replaced by the class **FINITE**.)

Theorem.

```
In[6]:= equiv[member[ord[x], omega], member[ord[x], DEDEKIND]]

Out[6]= True

In[7]:= member[ord[x_], DEDEKIND] := member[ord[x], omega]
```

Removing the **ord** wrapper yields:

Theorem. Every Dedekind finite ordinal is a natural number.

```
In[8]:= SubstTest[implies, equal[x, ord[t]],
          or[member[x, omega], not[member[x, DEDEKIND]]], t → x] // Reverse

Out[8]= or[member[x, omega], not[member[x, DEDEKIND]], not[member[x, OMEGA]]] == True

In[9]:= or[member[x_, omega], not[member[x_, DEDEKIND]], not[member[x_, OMEGA]]] := True
```

The set variable **x** can be eliminated:

Corollary.

```
In[10]:= Map[equal[#, V] &, SubstTest[class, x, implies[member[x, u], member[x, v]],
           {u → intersection[DEDEKIND, OMEGA], v → omega}]]

Out[10]= subclass[intersection[DEDEKIND, OMEGA], omega] == True

In[11]:= % /. Equal → SetDelayed
```

The reverse inclusion also holds, and therefore this inclusion can be strengthened to an equation, and then made into a rewrite rule.

```
In[12]:= SubstTest[and, subclass[u, v], subclass[v, u],
          {u → intersection[DEDEKIND, OMEGA], v → omega}]

Out[12]= equal[omega, intersection[DEDEKIND, OMEGA]] == True

In[13]:= intersection[DEDEKIND, OMEGA] := omega
```

a characterization of finiteness

Lemma.

```
In[15]:= SubstTest[member, t, intersection[u, v], {t → card[x], u → DEDEKIND, v → OMEGA}]

Out[15]= member[card[x], DEDEKIND] == member[x, FINITE]
```

```
In[16]:= member[card[x_], DEDEKIND] := member[x, FINITE]
```

Theorem.

```
In[19]:= SubstTest[implies, and[member[x, DEDEKIND], member[pair[x, y], Q]],
           member[y, DEDEKIND], y → card[x]] // Reverse

Out[19]= or[member[x, FINITE], not[member[x, DEDEKIND]], not[member[x, image[Q, OMEGA]]]] == True

In[20]:= or[member[x_, FINITE], not[member[x_, DEDEKIND]],
           not[member[x_, image[Q, OMEGA]]]] := True
```

Corollary. (Eliminating the set variable **x**.)

```
In[22]:= Map[equal[V, #] &, SubstTest[class, x, implies[member[x, u], member[x, v]],
           {u → intersection[DEDEKIND, image[Q, OMEGA]], v → FINITE}]]

Out[22]= subclass[intersection[DEDEKIND, image[Q, OMEGA]], FINITE] == True

In[23]:= (% /. Equal → SetDelayed)
```

Theorem. A set is finite if and only if it is Dedekind finite and equipollent to an ordinal.

```
In[24]:= SubstTest[and, subclass[u, v], subclass[v, u],
          {u -> intersection[DEDEKIND, image[Q, OMEGA]], v -> FINITE}]

Out[24]= equal[FINITE, intersection[DEDEKIND, image[Q, OMEGA]]] == True

In[26]:= intersection[DEDEKIND, image[Q, OMEGA]] := FINITE
```