finite acyclic => wellfounded

Johan G. F. Belinfante and Ming Li
2006 February 16

```
In[1]:= SetDirectory["l:"]; << goedel78.15b; << tools.m

        :Package Title: goedel78.15b           2006 February 15 at 6:10 p.m.

        It is now:  2006 Feb 16 at 12:21

        Loading Simplification Rules

        TOOLS.M                      Revised 2006 February 3

        weightlimit = 40
```

summary

Finite acyclic relations are wellfounded.

a lemma

If the range of a finite acyclic relation is contained in its domain, the relation is empty. Replacing **x** by its inverse implies that a similar result holds when the domain is contained in the range.

```
In[2]:= SubstTest[implies, and[member[y, FINITE], equal[0, fix[trv[y]]],
          subclass[y, cart[V, V]], subclass[range[y], domain[y]]], equal[0, y], y → inverse[x]]

Out[2]= or[equal[0, domain[x]], not[equal[0, fix[trv[x]]]], not[member[domain[x], FINITE]],
          not[member[range[x], FINITE]], not[subclass[domain[x], range[x]]]] == True

In[3]:= (% /. x → x_) /. Equal → SetDelayed

In[4]:= Map[not, SubstTest[and, implies[p2, p5],
          implies[p2, p6], implies[and[p1, p3, p4, p5, p6], p7],
          implies[and[p3, p7], p8], not[implies[and[p1, p2, p3, p4], p8]],
          {p1 -> equal[0, fix[trv[x]]], p2 → member[x, FINITE], p3 → subclass[x, cart[V, V]],
           p4 -> subclass[domain[x], range[x]], p5 -> member[domain[x], FINITE],
           p6 -> member[range[x], FINITE], p7 -> equal[0, domain[x]], p8 → equal[0, x]}]]

Out[4]= or[equal[0, x], not[equal[0, fix[trv[x]]]], not[member[x, FINITE]],
          not[subclass[x, cart[V, V]]], not[subclass[domain[x], range[x]]]] == True

In[5]:= or[equal[0, x_], not[equal[0, fix[trv[x_]]]], not[member[x_, FINITE]],
          not[subclass[domain[x_], range[x_]]], not[subclass[x_, cart[V, V]]]] := True
```

subvariance corollary

Replacing **x** by **composite[x, id[y]]** in the theorem derived in the preceding section yields:

```
In[6]:= SubstTest[implies,
         and[member[z, FINITE], equal[0, fix[trv[z]]], subclass[z, cart[V, V]],
          subclass[domain[z], range[z]]], equal[0, z], z → composite[x, id[y]]]

Out[6]= or[equal[0, intersection[y, domain[x]]], not[equal[0, fix[trv[composite[x, id[y]]]]]],
          not[member[composite[x, id[y]], FINITE]],
          not[subclass[intersection[y, domain[x]], image[x, y]]]] == True

In[7]:= (% /. {x → x_, y → y_}) /. Equal → SetDelayed
```

The following lemma holds because any subclass of a finite relation is finite.

```
In[8]:= SubstTest[implies, and[subclass[w, x], member[x, FINITE]],
         member[w, FINITE], w → composite[x, id[y]]]

Out[8]= or[member[composite[x, id[y]], FINITE], not[member[x, FINITE]]] == True

In[9]:= (% /. {x → x_, y → y_}) /. Equal → SetDelayed
```

In the same vein, the following is a consequence of the fact that any subclass of an acyclic relation is acyclic.

```
In[10]:= SubstTest[implies, and[subclass[w, x], equal[0, fix[trv[x]]]],
          equal[0, fix[trv[w]]], w → composite[x, id[y]]]

Out[10]= or[equal[0, fix[trv[composite[x, id[y]]]]], not[equal[0, fix[trv[x]]]]] == True

In[11]:= (% /. {x → x_, y → y_}) /. Equal → SetDelayed
```

The only class that is subvariant under **x** and disjoint from **domain[x]** is the empty set.

```
In[12]:= Map[implies[#, empty[y]] &, SubstTest[and, subclass[y, z], equal[0, z], z → image[x, y]]]

Out[12]= or[equal[0, y], not[equal[0, intersection[y, domain[x]]]],
           not[subclass[y, image[x, y]]]] == True

In[13]:= (% /. {x → x_, y → y_}) /. Equal → SetDelayed
```

These lemmas can be combined to obtain the following.

```
In[14]:= Map[not, SubstTest[and, implies[p1, p4],
           implies[p2, p5], implies[p3, p6], implies[and[p1, p4, p5, p6], p7],
           implies[and[p3, p7], p8], not[implies[and[p1, p2, p3], p8]],
           {p1 → member[x, FINITE], p2 → equal[0, fix[trv[x]]], p3 → subvariant[x, y], p4 ->
             member[composite[x, id[y]], FINITE], p5 → equal[0, fix[trv[composite[x, id[y]]]]],
            p6 -> subclass[intersection[y, domain[x]], image[x, y]],
            p7 → disjoint[y, domain[x]], p8 → equal[0, y]}]]

Out[14]= or[equal[0, y], not[equal[0, fix[trv[x]]]],
           not[member[x, FINITE]], not[subclass[y, image[x, y]]]] == True

In[15]:= (% /. {x → x_, y → y_}) /. Equal → SetDelayed
```

Eliminating the variable **y** now yields:

```
In[16]:= Map[equal[V, #] &, SubstTest[class, y,
            or[equal[0, y], not[equal[0, u]], not[member[x, v]], not[subclass[y, image[x, y]]]],
            {u → fix[trv[x]], v → FINITE}]] // Reverse

Out[16]= or[not[equal[0, fix[trv[x]]]],
           not[member[x, FINITE]], WELLFOUNDED[composite[Id, x]]] == True

In[17]:= (% /. x → x_) /. Equal → SetDelayed
```

Theorem.

```
In[18]:= SubstTest[implies, and[equal[y, composite[Id, x]],
           member[x, FINITE], equal[0, fix[trv[x]]]], WELLFOUNDED[y], y → x]

Out[18]= or[not[equal[0, fix[trv[x]]]], not[member[x, FINITE]],
           not[subclass[x, cart[V, V]]], WELLFOUNDED[x]] == True

In[19]:= or[not[equal[0, fix[trv[x_]]]], not[member[x_, FINITE]],
           not[subclass[x_, cart[V, V]]], WELLFOUNDED[x_]] := True
```

Variable-free restatement:

```
In[20]:= Map[equal[V, #] &, complement[dif[intersection[FINITE, ACYCLIC], WF]] // Normality]

Out[20]= subclass[intersection[ACYCLIC, FINITE], WF] == True

In[21]:= subclass[intersection[ACYCLIC, FINITE], WF] := True
```

infinitely divisible relations

Corollary.

```
In[22]:= SubstTest[implies, subclass[u, v], subclass[image[w, u], image[w, v]],
          {u -> intersection[ACYCLIC, FINITE], v -> WF, w → id[IDEM]}]

Out[22]= subclass[intersection[FINITE, IDEM, P[Di]], set[0]] == True
```

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

This can be restated as an equation.

```
In[24]:= equal[intersection[FINITE, IDEM, P[Di]], set[0]]

Out[24]= True
```

```
In[25]:= intersection[FINITE, IDEM, P[Di]] := set[0]
```

An idempotent irreflexive relation is infinitely divisible. Between any two points one can insert another: more precisely, if **pair[u,v]** belongs to such a relation, then there exists **w** such that **pair[u,w]** and **pair[w,v]** also belong to the relation, and **u, v, w** are all distinct. Since this process can be repeated over and over, it is intuitively clear that such a relation must be infinite, which is precisely what is stated by this equation.