finite strict orders are well-founded

Johan G. F. Belinfante
2009 May 5

```
In[1]:= SetDirectory["l:"]; << goedel.09may04a;<< tools.m

        :Package Title: goedel.09may04a            2009 May 4 at 3:40 p.m.

        It is now:  2009 May 5 at 15:51

        Loading Simplification Rules

        TOOLS.M                                    Revised 2009 April 6

        weightlimit = 40
```

summary

A **strict order** is an irreflexive transitive relation. Finite strict orders are well-founded.

finite strict orders

Lemma.

```
In[2]:= SubstTest[implies, and[subclass[u, v], subclass[v, w]],
          subclass[u, w], {u -> intersection[FINITE, TRV, P[Di]],
           v -> intersection[FINITE, ACYCLIC], w -> WF}] // Reverse

Out[2]= subclass[intersection[FINITE, TRV, P[Di]], WF] == True

In[3]:= subclass[intersection[FINITE, TRV, P[Di]], WF] := True
```

Theorem.

```
In[4]:= SubstTest[implies, and[member[x, y], subclass[y, z]],
          member[x, z], {y → intersection[FINITE, TRV, P[Di]], z → WF}] // Reverse

Out[4]= or[not[equal[0, fix[x]]],
          not[member[x, FINITE]], not[TRANSITIVE[x]], WELLFOUNDED[x]] == True

In[5]:= or[not[equal[0, fix[x_]]], not[member[x_, FINITE]],
          not[TRANSITIVE[x_]], WELLFOUNDED[x_]] := True
```

Corollary.

```
In[6]:= SubstTest[implies, and[equal[0, fix[t]], member[t, FINITE], TRANSITIVE[t]],
          WELLFOUNDED[t], t → trv[fin[x]]] // Reverse

Out[6]= or[not[equal[0, fix[trv[fin[x]]]]], WELLFOUNDED[trv[fin[x]]]] == True

In[7]:= or[not[equal[0, fix[trv[fin[x_]]]]], WELLFOUNDED[trv[fin[x_]]]] := True
```

Lemma.

```
In[8]:= SubstTest[implies, and[thin[inverse[w]], WELLFOUNDED[w], subvariant[w, y]],
          empty[y], w → trv[fin[x]]] // Reverse

Out[8]= or[equal[0, y], not[subclass[y, image[trv[fin[x]], y]]],
          not[WELLFOUNDED[trv[fin[x]]]]] == True

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

Theorem. Every nonempty class has a minimal element for any finite strict order.

```
In[11]:= Map[not, SubstTest[and, implies[p1, p2], implies[p2, p3], not[implies[p1, p3]],
            {p1 → empty[fix[trv[fin[x]]]], p2 -> WELLFOUNDED[trv[fin[x]]],
             p3 → implies[subvariant[trv[fin[x]], y], empty[y]]}]] // Reverse

Out[11]= or[equal[0, y], not[equal[0, fix[trv[fin[x]]]]],
           not[subclass[y, image[trv[fin[x]], y]]]] == True

In[12]:= or[equal[0, y_], not[equal[0, fix[trv[fin[x_]]]]],
           not[subclass[y_, image[trv[fin[x_]], y_]]]] := True
```

Lemma.

```
In[21]:= SubstTest[implies, and[equal[x, fin[t]], TRANSITIVE[x]],
           equal[x, trv[fin[x]]], t → x] // Reverse

Out[21]= or[equal[x, trv[fin[x]]], not[member[x, FINITE]], not[TRANSITIVE[x]]] == True

In[22]:= (% /. x → x_) /. Equal → SetDelayed
```

Theorem. Double-wrapper removal rule.

```
In[23]:= equiv[equal[x, trv[fin[x]]], and[member[x, FINITE], TRANSITIVE[x]]]

Out[23]= True

In[25]:= equal[x_, trv[fin[x_]]] := and[member[x, FINITE], TRANSITIVE[x]]
```

Corollary. The only class subvariant under a finite strict order is the empty set.

```
In[26]:= SubstTest[implies, and[equal[x, trv[fin[t]]], empty[fix[x]], subvariant[x, y]],
           empty[y], t → x] // Reverse

Out[26]= or[equal[0, y], not[equal[0, fix[x]]], not[member[x, FINITE]],
           not[subclass[y, image[x, y]]], not[TRANSITIVE[x]]] == True
```

```
In[27]:= or[equal[0, y_], not[equal[0, fix[x_]]], not[member[x_, FINITE]],
           not[subclass[y_, image[x_, y_]]], not[TRANSITIVE[x_]]] := True
```

Corollary. If **x** is a finite partial order, then **Di** ⋂ **x** is well-founded.

```
In[38]:= SubstTest[implies, and[empty[fix[t]], member[t, FINITE], TRANSITIVE[t]],
           WELLFOUNDED[t], t → intersection[Di, po[fin[x]]]] // Reverse

Out[38]= WELLFOUNDED[intersection[Di, po[fin[x]]]] == True

In[39]:= WELLFOUNDED[intersection[Di, po[fin[x_]]]] := True
```

Corollary.

```
In[50]:= equal[subvar[intersection[Di, po[fin[x]]]], set[0]]

Out[50]= True

In[51]:= subvar[intersection[Di, po[fin[x_]]]] := set[0]
```

Introducing another variable, one obtains the following.

Theorem. Every non-empty class has a minimal element for a finite partial order.

```
In[52]:= SubstTest[implies, and[thin[inverse[w]], WELLFOUNDED[w], subvariant[w, y]],
           empty[y], w -> intersection[Di, po[fin[x]]]] // Reverse

Out[52]= or[equal[0, y], not[subclass[y, fix[composite[po[fin[x]], id[y], Di]]]]] == True

In[53]:= or[equal[0, y_], not[subclass[y_, fix[composite[po[fin[x_]], id[y_], Di]]]]] := True
```

Restatement.

```
In[54]:= implies[empty[minimal[po[fin[x]], y]], empty[y]]

Out[54]= True
```

The wrappers can be removed.

Theorem. If **x** is a finite partial order, then the corresponding strict order **Di** ⋂ **x** is well-founded.

```
In[43]:= SubstTest[implies, equal[x, po[fin[t]]],
           WELLFOUNDED[intersection[Di, x]], t → x] // Reverse

Out[43]= or[not[member[x, FINITE]],
           not[PARTIALORDER[x]], WELLFOUNDED[intersection[Di, x]]] == True

In[44]:= or[not[member[x_, FINITE]],
           not[PARTIALORDER[x_]], WELLFOUNDED[intersection[Di, x_]]] := True
```

The next theorem presents a variable-free restatement of this theorem.

Theorem. Intersecting a finite partial order with **Di** yields a well-founded relation.

```
In[45]:= Map[equal[V, #] &, SubstTest[class, x, implies[member[x, u], member[x, v]],
           {u → intersection[FINITE, PO], v → image[inverse[IMAGE[id[Di]]], WF]}]]

Out[45]= subclass[image[IMAGE[id[Di]], intersection[FINITE, PO]], WF] == True

In[46]:= subclass[image[IMAGE[id[Di]], intersection[FINITE, PO]], WF] := True
```

Theorem. If **x** is a finite partial order, then every nonempty class **y** has a minimal element.

```
In[56]:= SubstTest[implies,
           and[equal[x, po[fin[t]]], empty[minimal[x, y]]], empty[y], t → x] // Reverse

Out[56]= or[equal[0, y], not[member[x, FINITE]], not[PARTIALORDER[x]],
           not[subclass[y, fix[composite[x, id[y], Di]]]]] == True

In[57]:= or[equal[0, y_], not[member[x_, FINITE]], not[PARTIALORDER[x_]],
           not[subclass[y_, fix[composite[x, id[y_], Di]]]]] := True
```

The inverse of a finite partial order is also a finite partial order.

Corollary.

```
In[41]:= SubstTest[WELLFOUNDED, intersection[Di, po[fin[t]]], t → inverse[po[fin[x]]]] // Reverse

Out[41]= WELLFOUNDED[intersection[Di, inverse[po[fin[x]]]]] == True

In[42]:= WELLFOUNDED[intersection[Di, inverse[po[fin[x_]]]]] := True
```

Theorem.

```
In[70]:= SubstTest[implies, empty[minimal[po[fin[t]], y]],
           empty[y], t → inverse[po[fin[x]]]] // Reverse

Out[70]= or[equal[0, y], not[subclass[y, fix[composite[Di, id[y], po[fin[x]]]]]]] == True

In[71]:= or[equal[0, y_], not[subclass[y_, fix[composite[Di, id[y_], po[fin[x_]]]]]]] := True
```

Restatement. Every non-empty class has a maximal element for a finite partial order.

```
In[72]:= implies[empty[maximal[po[fin[x]], y]], empty[y]]

Out[72]= True
```

Theorem. Another formulation of for the existence of maximal elements for a finite partial order.

```
In[73]:= SubstTest[subvar, intersection[Di, po[fin[t]]], t → inverse[po[fin[x]]]] // Reverse

Out[73]= subvar[intersection[Di, inverse[po[fin[x]]]]] == set[0]

In[74]:= subvar[intersection[Di, inverse[po[fin[x_]]]]] := set[0]
```