cover relation of a connex strict order

Johan G. F. Belinfante
2010 February 7

```
In[1]:= SetDirectory["l:"]; << goedel.10feb06a;<< tools.m
        :Package Title: goedel.10feb06a                2010 February 6 at 3:50 p.m.
        It is now:  2010 Feb 7 at 10:35
        Loading Simplification Rules
        TOOLS.M                                    Revised 2010 January 29
        weightlimit = 40
```

summary

The cover relation of a connex strict order is a function.

simplification rules

Lemma. Simplification rule.

```
In[2]:= ImageComp[COVER, IMAGE[id[cart[V, V]]], x] // Reverse
Out[2]= image[COVER, image[IMAGE[id[cart[V, V]]], x]] == image[COVER, x]
```

```
In[3]:= image[COVER, image[IMAGE[id[cart[V, V]]], x_]] := image[COVER, x]
```

Lemma. Simplification rule.

```
In[4]:= ImageComp[COVER, IMAGE[id[Di]], x] // Reverse
Out[4]= image[COVER, image[IMAGE[id[Di]], x]] == image[COVER, x]
```

```
In[5]:= image[COVER, image[IMAGE[id[Di]], x_]] := image[COVER, x]
```

derivation for the special case of sets

For the case of sets, the entire derivation can be done using variable-free rewrite rules.

Theorem. The cover relation of the reflexive hull of a relation is the same as that of the relation itself.

```
In[6]:= Assoc[COVER, IMAGE[id[Di]], HULL[RFX]] // Reverse
Out[6]= composite[COVER, HULL[RFX]] == composite[COVER, id[P[cart[V, V]]]]

In[7]:= composite[COVER, HULL[RFX]] := composite[COVER, id[P[cart[V, V]]]]
```

Theorem. The cover relation of a connex strict order is a function.

```
In[8]:= SubstTest[implies, and[subclass[u, v], subclass[v, w]], subclass[u, w],
          {u -> image[HULL[RFX], intersection[CONNEX, TRV, P[Di]]], v -> TO,
           w → image[inverse[IMAGE[id[Di]]], image[inverse[COVER], FUNS]]}] // Reverse
Out[8]= subclass[image[COVER, intersection[CONNEX, TRV, P[Di]]], FUNS] == True

In[9]:= subclass[image[COVER, intersection[CONNEX, TRV, P[Di]]], FUNS] := True
```

The following special case follows from the fact that the reflexive hull of a connex well-founded relation is a well-order.

Corollary. The cover relation of a connex well-founded relation is a function.

```
In[10]:= SubstTest[implies, and[subclass[u, v], subclass[v, w]],
           subclass[u, w], {u -> image[HULL[RFX], intersection[CONNEX, WF]], v → WO,
            w → image[inverse[IMAGE[id[Di]]], image[inverse[COVER], FUNS]]}] // Reverse
Out[10]= subclass[image[COVER, intersection[CONNEX, WF]], FUNS] == True

In[11]:= subclass[image[COVER, intersection[CONNEX, WF]], FUNS] := True
```

---

a more general result

The restriction to sets is removed in this section. The basic idea is exactly the same, but this time one needs to introduce variables.

Lemma. Simplification rule.

```
In[12]:= SubstTest[cover, intersection[Di, t], t → union[x, id[y]]]
Out[12]= cover[union[x, id[y]]] == cover[x]

In[13]:= cover[union[x_, id[y_]]] := cover[x]
```

Theorem.

```
In[14]:= SubstTest[implies, TOTALORDER[t], FUNCTION[cover[t]], t → union[x, id[y]]] // Reverse
Out[14]= or[FUNCTION[cover[x]], not[TOTALORDER[union[x, id[y]]]]] == True

In[15]:= or[FUNCTION[cover[x_]], not[TOTALORDER[union[x_, id[y_]]]]] := True
```

Theorem. The cover relation of a connex strict order is a function.

```
In[16]:= Map[not, SubstTest[and, implies[p1, p2], implies[p2, p3], not[implies[p1, p3]],
           {p1 -> and[STRICTORDER[x], subclass[cart[union[domain[x], range[x]],
                union[domain[x], range[x]]], union[Id, x, inverse[x]]]],
            p2 -> TOTALORDER[union[x, id[udora[x]]]], p3 → FUNCTION[cover[x]]}]] // Reverse

Out[16]= or[FUNCTION[cover[x]], not[equal[0, fix[x]]],
           not[subclass[cart[union[domain[x], range[x]], union[domain[x], range[x]]],
             union[Id, x, inverse[x]]]], not[TRANSITIVE[x]]] == True

In[17]:= or[FUNCTION[cover[x_]], not[equal[0, fix[x_]]],
           not[subclass[cart[union[domain[x_], range[x_]], union[domain[x_], range[x_]]],
             union[Id, x_, inverse[x_]]]], not[TRANSITIVE[x_]]] := True
```

Lemma.

```
In[18]:= SubstTest[implies, WELLORDER[t], FUNCTION[cover[t]], t → union[x, id[y]]] // Reverse

Out[18]= or[FUNCTION[cover[x]], not[WELLORDER[union[x, id[y]]]]] == True

In[19]:= or[FUNCTION[cover[x_]], not[WELLORDER[union[x_, id[y_]]]]] := True
```

Theorem. The cover relation of a connex well-founded relation is a function.

```
In[20]:= Map[not, SubstTest[and, implies[p1, p2], implies[p2, p3], not[implies[p1, p3]],
           {p1 -> and[WELLFOUNDED[x], subclass[cart[union[domain[x], range[x]],
                union[domain[x], range[x]]], union[Id, x, inverse[x]]]],
            p2 → WELLORDER[union[x, id[udora[x]]]], p3 → FUNCTION[cover[x]]}]] // Reverse

Out[20]= or[FUNCTION[cover[x]],
           not[subclass[cart[union[domain[x], range[x]], union[domain[x], range[x]]],
             union[Id, x, inverse[x]]]], not[WELLFOUNDED[x]]] == True

In[21]:= or[FUNCTION[cover[x_]],
           not[subclass[cart[union[domain[x_], range[x_]], union[domain[x_], range[x_]]],
             union[Id, x_, inverse[x_]]]], not[WELLFOUNDED[x_]]] := True
```