Theorem SBV−PS1

Johan G. F. Belinfante
2004 January 18

```
In[1]:= << goedel53.17b; << tools.m

        :Package Title: goedel53.17b      2004 January 17 at 8:35 p.m.

        It is now:  2004 Jan 21 at 13:46

        Loading Simplification Rules

        TOOLS.M                      Revised 2004 January 3

        weightlimit = 40
```

summary

Theorem **SBV−PS1** proved 2000 April 15 using **Otter** is generalized here by removing an unneeded hypothesis. The derivation here does not follow **Otter**'s proof, but uses a different method. A variable−free version of the theorem is obtained using **reify**.

a key lemma

In this section a key lemma is derived: **id[P[x]]** subcommutes with the proper subset relation **PS**.

```
In[2]:= dif[composite[id[P[x]], PS], composite[PS, id[P[x]]]] // VSNormality

Out[2]= composite[id[P[x]], PS, id[complement[P[x]]]] == 0

In[3]:= (% /. x -> x_) /. Equal -> SetDelayed
```

The statement of subcommutativity is transformed by rewrite rules to a somewhat cryptic form:

```
In[4]:= SubstTest[equal, 0, dif[u, v],
          {u -> composite[id[P[x]], PS], v -> composite[PS, id[P[x]]]}] // Reverse

Out[4]= subclass[U[fix[composite[Di, id[P[x]], S]]], x] == True

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

The following less crytic corollary is all that will be needed in the sequel.

```
In[6]:= SubstTest[implies, subclass[u, v], subclass[image[u, x], image[v, x]],
            {u -> composite[id[P[y]], PS], v -> composite[PS, id[P[y]]]}]

Out[6]= subclass[intersection[image[PS, x], P[y]], image[PS, intersection[x, P[y]]]] == True

In[7]:= subclass[intersection[image[PS, x_], P[y_]], image[PS, intersection[x_, P[y_]]]] := True
```

derivation of a generalization of Theorem SBV−PS1

Only a few simple steps are needed to derive the main theorem.

```
In[8]:= SubstTest[implies, and[subclass[u, v], subclass[v, w]], subclass[u, w],
           {u -> intersection[x, P[y]],
         v -> intersection[P[y], image[PS, x]], w -> image[PS, intersection[x, P[y]]]}]

Out[8]= or[not[subclass[intersection[x, P[y]], image[PS, x]]],
          subclass[intersection[x, P[y]], image[PS, intersection[x, P[y]]]]] == True

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

The following theorem generalizes Theorem **SBV−PS1** proved using **Otter** in that an unneeded hypothesis that **y** be a member of **x** has been removed.

```
In[10]:= Map[not, SubstTest[and, implies[p1, p2], implies[p2, p3], not[implies[p1, p3]],
              {p1 -> subvariant[PS, x], p2 -> subclass[intersection[x, P[y]], image[PS, x]],
          p3 -> subvariant[PS, intersection[x, P[y]]]}]]

Out[10]= or[not[subclass[x, image[PS, x]]],
           subclass[intersection[x, P[y]], image[PS, intersection[x, P[y]]]]] == True

In[11]:= or[not[subclass[x_, image[PS, x_]]],
           subclass[intersection[x_, P[y_]], image[PS, intersection[x_, P[y_]]]]] := True
```

To eliminate the variable **x**, one needs a lemma derivable using nothing more than double negation:

```
In[12]:= implies[member[x, subvar[PS]], member[intersection[x, P[y]], subvar[PS]]] // NotNotTest

Out[12]= or[and[member[intersection[x, P[y]], V],
            subclass[intersection[x, P[y]], image[PS, intersection[x, P[y]]]]],
           not[member[x, V]], not[subclass[x, image[PS, x]]]] == True

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

The final result is this:

```
In[14]:= Map[equal[V, #] &, SubstTest[class, y,
            implies[member[y, z], member[intersection[y, P[x]], z]], z -> subvar[PS]]] // Reverse

Out[14]= subclass[image[IMAGE[id[P[x]]], subvar[PS]], subvar[PS]] == True

In[15]:= subclass[image[IMAGE[id[P[x_]]], subvar[PS]], subvar[PS]] := True
```

removing the other variable

The remaining variable can be removed quickly by using **reify**:

```
In[16]:= Map[equal[0, #] &,
           SubstTest[reify, x, dif[image[IMAGE[id[P[x]]], z], z], z -> subvar[PS]]] // Reverse

Out[16]= subclass[image[CAP, cart[subvar[PS], range[POWER]]], subvar[PS]] == True

In[17]:= subclass[image[CAP, cart[subvar[PS], range[POWER]]], subvar[PS]] := True
```

This theorem says that subvariance under the proper subset relation is preseverved under intersections with power sets.