T2 separation condition in topology

Johan G. F. Belinfante and Tiffany Goble
2003 June 17

```
In[1]:= << goedel52.s06; << tools.m

        :Package Title: goedel52.s06       2003 June 16 at 12:15 midnite

        It is now:  2003 Jun 17 at 9:4

        Loading Simplification Rules

        TOOLS.M                      Revised 2003 June 13

        weightlimit = 40

        weightlimit = 40
```

■ summary

A Hausdorff topology is a topology that satisfies the T2 separation condition. In this notebook, a membership rule for the class of all sets satisfying the T2 condition is derived, and it is shown that every power set satisfies this condition.

■ definition of T2

The class of Hausdorff topologies is the intersection of the class **TOPS** of all topologies and the class of sets that satisfy the T2 separation condition:

```
In[2]:= T2 == class[t, forall[u, v, implies[and[member[u, U[t]], member[v, U[t]]],
            or[equal[u, v], exists[x, y, and[member[x, t], member[y, t],
               member[u, x], member[v, y], disjoint[x, y]]]]]]]

Out[2]= T2 == complement[
          fix[composite[inverse[BIGCUP], e, fix[composite[complement[composite[intersection[
                 composite[intersection[composite[inverse[FIRST], e], composite[inverse[
                     SECOND], inverse[e]]], DISJOINT], composite[inverse[FIRST], e]], e]],
              intersection[composite[Di, SECOND], composite[inverse[e], BIGCUP, FIRST]]]]]]]
```

To facilitate deriving a membership rule for this class, it helps to turn off the flags **cond** and **simplify**.

```
In[3]:= cond = False; simplify = False;
```

This yields a simple membership rule:

```
In[4]:= Map[assert[member[t, #]] &, %%]

Out[4]= member[t, T2] == and[member[t, V], subclass[
           composite[id[U[t]], Di, id[U[t]]], composite[inverse[e], id[t], DISJOINT, id[t], e]]]
```

This membership rule will be used to define the class **T2**.

```
In[5]:= member[t_, T2] := and[member[t, V], subclass[
          composite[id[U[t]], Di, id[U[t]]], composite[inverse[E], id[t], DISJOINT, id[t], E]]]
```

We attempt to verify that all power sets satisfy this condition using **Normality**:

```
In[6]:= Map[equal[V, #] &, image[inverse[POWER], T2] // Normality]

Out[6]= subclass[range[POWER], T2] ==
         subclass[composite[S, PAIRSET, id[Di]], composite[S, inverse[RCF], e, inverse[CART], e]]
```

The expression encountered can be simplified:

```
In[7]:= composite[inverse[RCF], E, inverse[CART], E] // VSRenormality

Out[7]= composite[inverse[RCF], e, inverse[CART], e] == composite[S, PAIRSET, id[Di]]

In[8]:= composite[inverse[RCF], E, inverse[CART], E] := composite[S, PAIRSET, id[Di]]
```

The following lemma is needed to proceed:

```
In[9]:= equal[composite[id[complement[singleton[0]]], intersection[E, x]], intersection[E, x]]

Out[9]= True

In[10]:= composite[id[complement[singleton[0]]], intersection[E, x_]] := intersection[E, x]
```

A further simplification:

```
In[11]:= composite[S, PAIRSET, id[Di], inverse[FIRST]] // VSNormality

Out[11]= composite[S, PAIRSET, id[Di], inverse[FIRST]] ==
          intersection[e, composite[Di, SINGLETON]]

In[12]:= composite[S, PAIRSET, id[Di], inverse[FIRST]] :=
          intersection[E, composite[Di, SINGLETON]]
```

The final step yields the desired result:

```
In[13]:= Map[equal[V, #] &, image[inverse[POWER], T2] // Normality // RotateFix]

Out[13]= subclass[range[POWER], T2] == True

In[14]:= subclass[range[POWER], T2] := True
```

To complete the proof that all discrete topologies are Hausdorff topologies one also needs to use the following fact:

```
In[15]:= subclass[range[POWER], TOPS]

Out[15]= True
```