finite subspaces of a Hausdorff space

Johan G. F. Belinfante
2014 April 19

```
In[1]:= SetDirectory["l:"]; << goedel.14apr18a

:Package Title: goedel.14apr18a                    2014 April 18 at 3:50 p.m.

Loading takes about seventeen minutes, half that time due to builtin pauses.

It is now:  2014 Apr 19 at 18:32

Loading Simplification Rules

TOOLS.M is now incorporated in the GOEDEL program as of 2010 September 3

weightlimit = 40

Loading completed.

It is now:  2014 Apr 19 at 18:49
```

summary

Finite subspaces of a Hausdorff space are closed.

derivation

The following theorem is an application of **FINITE** induction to the class of closed subsets of a topological space. The presence of the **top[t]** wrapper causes a rewrite rule to introduce the cover relation **K** as follows:

```
In[2]:= subclass[image[SINGLETON, U[top[t]]], image[RC[U[top[t]]], top[t]]]

Out[2]= subclass[image[inverse[K], set[U[top[t]]]], top[t]]
```

Theorem. If every singleton subspace of a topological space is closed, then every finite subspace is closed.

```
In[3]:= SubstTest[implies, and[member[0, u], subclass[image[SINGLETON, v], u],
           subclass[image[CUP, cart[u, u]], u]], subclass[intersection[FINITE, P[v]], u],
          {u → image[RC[U[top[t]]], top[t]], v → U[top[t]]}] // Reverse

Out[3]= or[not[subclass[image[inverse[K], set[U[top[t]]]], top[t]]],
          subclass[image[RC[U[top[t]]], FINITE], top[t]]] == True

In[4]:= or[not[subclass[image[inverse[K], set[U[top[t_]]]], top[t_]]],
          subclass[image[RC[U[top[t_]]], FINITE], top[t_]]] := True
```

The cover relation can be eliminated by replacing the **top[t]** wrapper with a literal **t ∈ TOPS**.

Corollary.

```
In[5]:= SubstTest[implies, equal[t, top[w]],
          implies[subclass[image[SINGLETON, U[t]], image[RC[U[t]], t]],
           subclass[intersection[FINITE, P[U[t]]], image[RC[U[t]], t]]], w → t] // Reverse

Out[5]= or[not[member[t, TOPS]], not[subclass[image[SINGLETON, U[t]], image[RC[U[t]], t]]],
          subclass[intersection[FINITE, P[U[t]]], image[RC[U[t]], t]]] == True

In[6]:= or[not[member[t_, TOPS]], not[subclass[image[SINGLETON, U[t_]], image[RC[U[t_]], t_]]],
          subclass[intersection[FINITE, P[U[t_]]], image[RC[U[t_]], t_]]] := True
```

Theorem. Every **T1** topology is point-closed.

```
In[7]:= SubstTest[implies, and[subclass[u, v], subclass[v, w]], subclass[u, w],
          {u → TOPS, v → fix[UCLOSURE], w → union[PointClosed, complement[T1]]}] // Reverse

Out[7]= subclass[intersection[T1, TOPS], PointClosed] == True

In[8]:= subclass[intersection[T1, TOPS], PointClosed] := True
```

Corollary.

```
In[9]:= SubstTest[implies, and[member[t, u], subclass[u, v]],
          member[t, v], {u -> intersection[T1, TOPS], v -> PointClosed}] // Reverse

Out[9]= or[member[t, PointClosed], not[member[t, T1]], not[member[t, TOPS]]] == True

In[10]:= or[member[t_, PointClosed], not[member[t_, T1]], not[member[t_, TOPS]]] := True
```

Observation. If **t** is point-closed, then every singleton subspace is closed.

```
In[11]:= implies[member[t, PointClosed], subclass[image[SINGLETON, U[t]], image[RC[U[t]], t]]]

Out[11]= True
```

Theorem. If **t** is a **T1** topology, then all singleton subspaces are closed.

```
In[12]:= Map[not, SubstTest[and, implies[p1, p2], implies[p2, p3], not[implies[p1, p3]],
           {p1 -> and[member[t, TOPS], member[t, T1]], p2 -> member[t, PointClosed],
            p3 -> subclass[image[SINGLETON, U[t]], image[RC[U[t]], t]]}]] // Reverse

Out[12]= or[not[member[t, T1]], not[member[t, TOPS]],
           subclass[image[SINGLETON, U[t]], image[RC[U[t]], t]]] == True

In[13]:= or[not[member[t_, T1]], not[member[t_, TOPS]],
           subclass[image[SINGLETON, U[t_]], image[RC[U[t_]], t_]]] := True
```

Theorem. If **t** is a **T1** topology, then any finite subspace is closed.

```
In[14]:= Map[not, SubstTest[and, implies[p1, p2], implies[and[p1, p2], p3],
            not[implies[p1, p3]], {p1 -> and[member[t, TOPS], member[t, T1]],
             p2 -> subclass[image[SINGLETON, U[t]], image[RC[U[t]], t]],
             p3 -> subclass[intersection[FINITE, P[U[t]]], image[RC[U[t]], t]]}]] // Reverse

Out[14]= or[not[member[t, T1]], not[member[t, TOPS]],
           subclass[intersection[FINITE, P[U[t]]], image[RC[U[t]], t]]] == True

In[15]:= or[not[member[t_, T1]], not[member[t_, TOPS]],
           subclass[intersection[FINITE, P[U[t_]]], image[RC[U[t_]], t_]]] := True
```

Corollary. Any finite subspace of a Hausdorff space is closed.

```
In[16]:= Map[not, SubstTest[and, implies[p1, p2], implies[p2, p3], not[implies[p1, p3]],
            {p1 → member[t, T2], p2 → member[t, T1], p3 -> or[not[member[t, TOPS]],
               subclass[intersection[FINITE, P[U[t]]], image[RC[U[t]], t]]]}]] // Reverse

Out[16]= or[not[member[t, T2]], not[member[t, TOPS]],
           subclass[intersection[FINITE, P[U[t]]], image[RC[U[t]], t]]] == True

In[17]:= or[not[member[t_, T2]], not[member[t_, TOPS]],
           subclass[intersection[FINITE, P[U[t_]]], image[RC[U[t_]], t_]]] := True
```

The conclusion can be rewritten using the following.

Theorem.

```
In[18]:= SubstTest[implies, subclass[u, v], subclass[image[r, u], image[r, v]],
           {r → RC[x], u -> intersection[y, P[x]], v -> image[RC[x], z]}] // Reverse

Out[18]= or[not[subclass[intersection[y, P[x]], image[RC[x], z]]],
           subclass[image[RC[x], y], z]] == True

In[19]:= or[not[subclass[intersection[y_, P[x_]], image[RC[x_], z_]]],
           subclass[image[RC[x_], y_], z_]] := True
```

The reverse implication also holds when **x** is a set. This equivalent formulation becomes automatic when one introduces a **top[t]** wrapper.

Theorem.

```
In[20]:= SubstTest[or, not[member[w, T1]], not[member[w, TOPS]],
           subclass[intersection[FINITE, P[U[w]]], image[RC[U[w]], w]], w → top[t]] // Reverse

Out[20]= or[not[member[top[t], T1]], subclass[image[RC[U[top[t]]], FINITE], top[t]]] == True

In[21]:= or[not[member[top[t_], T1]], subclass[image[RC[U[top[t_]]], FINITE], top[t_]]] := True
```

Corollary. (Eliminating the **top** wrapper.)

```
In[22]:= SubstTest[implies, equal[t, top[w]],
           or[not[member[t, T1]], subclass[image[RC[U[t]], FINITE], t]], w → t] // Reverse

Out[22]= or[not[member[t, T1]], not[member[t, TOPS]],
           subclass[image[RC[U[t]], FINITE], t]] == True

In[23]:= or[not[member[t_, T1]], not[member[t_, TOPS]],
           subclass[image[RC[U[t_]], FINITE], t_]] := True
```

Corollary. Cofinite subspaces of a Hausdorff space are open.

```
In[24]:= Map[not, SubstTest[and, implies[p1, p2], implies[p2, p3],
            not[implies[p1, p3]], {p1 → member[t, T2], p2 → member[t, T1],
             p3 -> or[not[member[t, TOPS]], subclass[image[RC[U[t]], FINITE], t]]}]] // Reverse

Out[24]= or[not[member[t, T2]], not[member[t, TOPS]],
           subclass[image[RC[U[t]], FINITE], t]] == True

In[25]:= or[not[member[t_, T2]], not[member[t_, TOPS]],
           subclass[image[RC[U[t_]], FINITE], t_]] := True
```

introducing a variable for the subspace

Theorem. A general result.

```
In[26]:= Map[or[#, member[intersection[x, complement[y]], t]] &,
           SubstTest[implies, and[member[y, u], subclass[u, v]], member[y, v],
            {u -> intersection[z, P[x]], v -> image[RC[x], t]}]] // Reverse

Out[26]= or[member[intersection[x, complement[y]], t], not[member[y, z]],
           not[subclass[y, x]], not[subclass[intersection[z, P[x]], image[RC[x], t]]]] == True

In[27]:= or[member[intersection[x_, complement[y_]], t_],
           not[member[y_, z_]], not[subclass[y_, x_]],
           not[subclass[intersection[z_, P[x_]], image[RC[x_], t_]]]] := True
```

Theorem. Any finite subspace **x** of a space with a **T1** topology **t** is closed.

```
In[28]:= Map[not, SubstTest[and, implies[and[p1, p3], p4], implies[p2, p3],
            not[implies[and[p1, p2], p4]], {p1 → and[member[x, FINITE], subclass[x, U[t]]],
             p2 → member[t, intersection[T1, TOPS]],
             p3 -> subclass[intersection[FINITE, P[U[t]]], image[RC[U[t]], t]],
             p4 -> member[intersection[complement[x], U[t]], t]}]] // Reverse

Out[28]= or[member[intersection[complement[x], U[t]], t], not[member[t, T1]],
           not[member[t, TOPS]], not[member[x, FINITE]], not[subclass[x, U[t]]]] == True

In[29]:= or[member[intersection[complement[x_], U[t_]], t_], not[member[t_, T1]],
           not[member[t_, TOPS]], not[member[x_, FINITE]], not[subclass[x_, U[t_]]]] := True
```

Corollary. Any finite subspace **x** of a space with a Hausdorff topology **t** is closed.

```
In[30]:= Map[not, SubstTest[and, implies[p1, p2], implies[p2, p3],
            not[implies[p1, p3]], {p1 → member[t, T2], p2 → member[t, T1],
             p3 -> or[member[intersection[complement[x], U[t]], t], not[member[t, TOPS]],
               not[member[x, FINITE]], not[subclass[x, U[t]]]]}]] // Reverse

Out[30]= or[member[intersection[complement[x], U[t]], t], not[member[t, T2]],
           not[member[t, TOPS]], not[member[x, FINITE]], not[subclass[x, U[t]]]] == True

In[31]:= or[member[intersection[complement[x_], U[t_]], t_], not[member[t_, T2]],
           not[member[t_, TOPS]], not[member[x_, FINITE]], not[subclass[x_, U[t_]]]] := True
```

Lemma.

```
In[32]:= SubstTest[or, member[intersection[complement[y], U[t]], t],
           not[member[t, T1]], not[member[t, TOPS]], not[member[y, FINITE]],
           not[subclass[y, U[t]]], y → dif[U[t], x]] // Reverse

Out[32]= or[member[intersection[x, U[t]], t], not[member[t, T1]], not[member[t, TOPS]],
           not[member[intersection[complement[x], U[t]], FINITE]]] == True

In[33]:= (% /. {t → t_, x → x_}) /. Equal → SetDelayed
```

Corollary. Any cofinite subspace of a **T1** topological space is open.

```
In[34]:= SubstTest[implies, equal[w, intersection[x, U[t]]],
           or[member[w, t], not[member[t, T1]], not[member[t, TOPS]],
            not[member[intersection[complement[x], U[t]], FINITE]]], w → x] // Reverse

Out[34]= or[member[x, t], not[member[t, T1]], not[member[t, TOPS]],
           not[member[intersection[complement[x], U[t]], FINITE]],
           not[subclass[x, U[t]]]] == True

In[35]:= or[member[x_, t_], not[member[t_, T1]], not[member[t_, TOPS]],
           not[member[intersection[complement[x_], U[t_]], FINITE]],
           not[subclass[x_, U[t_]]]] := True
```