connected subspaces

Johan G. F. Belinfante
2014 March 20

```
In[1]:= SetDirectory["l:"]; << goedel.14mar19a

:Package Title: goedel.14mar19a                    2014 March 20 at 6:30 a.m.

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

It is now:  2014 Mar 20 at 14:36

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 Mar 20 at 14:53
```

summary

If **x** is a connected subspace and if **y** is an open and closed subspace of a topological space, then $\mathbf{x} \subset \mathbf{y}$ or $\mathbf{x} \cap \mathbf{y} = \mathbf{0}$. In fact, no topology axioms are needed for this result.

reference

The result derived in this notebook is Lemma 1.2 on page 149 in the following reference.

```
In[53]:= "James Raymond Munkres, Topology,
           Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1975.";
```

derivation

Lemma.

```
In[3]:= SubstTest[or, equal[0, w], equal[x, w], not[member[w, image[IMAGE[id[x]], t]]],
          not[member[image[IMAGE[id[x]], t], CONNECTED]],
          not[member[intersection[x, complement[w]], image[IMAGE[id[x]], t]]],
          not[subclass[x, U[t]]], w → intersection[x, y]] // Reverse

Out[3]= or[equal[0, intersection[x, y]], not[member[image[IMAGE[id[x]], t], CONNECTED]],
          not[member[intersection[x, y], image[IMAGE[id[x]], t]]],
          not[member[intersection[x, complement[y]], image[IMAGE[id[x]], t]]],
          not[subclass[x, U[t]]], subclass[x, y]] == True

In[4]:= or[equal[0, intersection[x_, y_]], not[member[image[IMAGE[id[x_]], t_], CONNECTED]],
          not[member[intersection[complement[y_], x_], image[IMAGE[id[x_]], t_]]],
          not[member[intersection[x_, y_], image[IMAGE[id[x_]], t_]]],
          not[subclass[x_, U[t_]]], subclass[x_, y_]] := True
```

Theorem. If **x** is a connected subspace, and if **y** is an open and closed subspace of a space with topology **t**, then $\mathbf{x} \subset \mathbf{y}$ or $\mathbf{x} \cap \mathbf{y} = \mathbf{0}$.

```
In[14]:= Map[not, SubstTest[and, implies[p1, p5],
            implies[and[p3, p4], p6], implies[p1, p7], implies[and[p1, p2, p3, p5], p8],
            (*implies[and[p3,p4,p7,p8],p9],*) not[implies[and[p1, p2, p3, p4], p9]],
            {p1 -> member[y, t], p2 -> member[dif[U[t], y], t], p3 -> subclass[x, U[t]],
             p4 -> member[image[IMAGE[id[x]], t], CONNECTED], p5 → subclass[y, U[t]],
             p6 → member[x, V], p7 -> member[intersection[x, y], image[IMAGE[id[x]], t]],
             p8 -> member[intersection[x, complement[y]], image[IMAGE[id[x]], t]],
             p9 -> or[equal[0, intersection[x, y]], subclass[x, y]]}]] // Reverse

Out[14]= or[equal[0, intersection[x, y]],
           not[member[y, t]], not[member[image[IMAGE[id[x]], t], CONNECTED]],
           not[member[intersection[complement[y], U[t]], t]],
           not[subclass[x, U[t]]], subclass[x, y]] == True

In[16]:= or[equal[0, intersection[x_, y_]], not[member[image[IMAGE[id[x_]], t_], CONNECTED]],
           not[member[intersection[complement[y_], U[t_]], t_]],
           not[member[y_, t_]], not[subclass[x_, U[t_]]], subclass[x_, y_]] := True
```

Lemma. Eliminate the variable **y**.

```
In[20]:= Map[equal[domain[#], V] &,
          SubstTest[reify, y, case[or[equal[0, intersection[x, y]], not[member[y, s]],
             not[member[u, v]], not[subclass[x, U[t]]], subclass[x, y]]],
           {s → intersection[t, image[RC[U[t]], t]], u -> image[IMAGE[id[x]], t], v -> CONNECTED}]]

Out[20]= or[not[equal[intersection[image[IMAGE[id[x]], t], image[RC[intersection[x, U[t]]],
               image[IMAGE[id[x]], t]]], set[0, intersection[x, U[t]]]]],
           not[member[intersection[x, U[t]], V]], not[subclass[x, U[t]]],
           subclass[intersection[t, image[RC[U[t]], t]],
            union[image[S, set[x]], P[complement[x]]]]] == True

In[26]:= (% /. {t → t_, x → x_, y → y_}) /. Equal → SetDelayed
```

Lemma.

```
In[27]:= SubstTest[implies, equal[w, intersection[x, U[t]]],
           or[not[equal[intersection[image[IMAGE[id[x]], t],
               image[RC[w], image[IMAGE[id[x]], t]]], set[0, w]]], not[member[w, V]],
            not[subclass[x, U[t]]], subclass[intersection[t, image[RC[U[t]], t]],
             union[image[S, set[x]], P[complement[x]]]]]], w → x] // Reverse

Out[27]= or[
           not[equal[intersection[image[IMAGE[id[x]], t], image[RC[x], image[IMAGE[id[x]], t]]],
             set[0, x]]], not[member[x, V]],
           not[subclass[x, U[t]]], subclass[intersection[t, image[RC[U[t]], t]],
            union[image[S, set[x]], P[complement[x]]]]] == True

In[28]:= (% /. {t → t_, x → x_, y → y_}) /. Equal → SetDelayed
```

Corollary. (Restatement of the theorem with one less variable.)

```
In[32]:= Map[or[not[#], not[subclass[x, U[t]]], subclass[
             intersection[t, image[RC[U[t]], t]], union[image[S, set[x]], P[complement[x]]]]]] &,
          (member[s, CONNECTED] // AssertTest) /. s -> image[IMAGE[id[x]], t]]

Out[32]= or[not[member[image[IMAGE[id[x]], t], CONNECTED]],
           not[subclass[x, U[t]]], subclass[intersection[t, image[RC[U[t]], t]],
            union[image[S, set[x]], P[complement[x]]]]] == True

In[33]:= or[not[member[image[IMAGE[id[x_]], t_], CONNECTED]],
           not[subclass[x_, U[t_]]], subclass[intersection[t_, image[RC[U[t_]], t_]],
            union[image[S, set[x_]], P[complement[x_]]]]] := True
```