Alexandrov topologies

Johan G. F. Belinfante
2013 December 23

```
In[1]:= SetDirectory["l:"]; << goedel.13dec19b

        :Package Title: goedel.13dec19b          2013 December 19 at 2:40 p.m.

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

        It is now:  2013 Dec 23 at 14:46

        Loading Simplification Rules

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

        weightlimit = 40

        Loading completed.

        It is now:  2013 Dec 23 at 15:3
```

summary

An **Alexandrov topology** is a collection of sets that is closed under arbitrary intersections and arbitrary unions. The class **ALEX** of Alexandrov topologies is defined in this notebook, and a few of its properties are derived.

reference

```
In[2]:= "Pavel S. Alexandroff, Discrete Spaces, Mat.
          Sb. (N. S.), vol. 2, pp. 501-528 (1937).  [in German]";
```

definition

The class **ALEX** of Alexandrov topologies is here defined using a **case** construction.

```
In[3]:= image[V, intersection[ALEX, set[x_]]] :=
         case[and[member[x, V], subclass[Aclosure[x], x], subclass[Uclosure[x], x]]]
```

Theorem. Normalization condition.

```
In[4]:= Map[class[x, #] &, SubstTest[equal, V, case[t], t → member[x, ALEX]]] // Reverse

Out[4]= intersection[fix[ACLOSURE], fix[UCLOSURE]] == ALEX
```

```
In[5]:= intersection[fix[ACLOSURE], fix[UCLOSURE]] := ALEX
```

Theorem. Every Alexandrov topology is a topology.

```
In[6]:= Map[equal[V, #] &, dif[ALEX, TOPS] // complement // Renormality]

Out[6]= subclass[ALEX, TOPS] == True

In[7]:= subclass[ALEX, TOPS] := True
```

inclusions

Some obvious inclusions are derived in this section.

Theorem.

```
In[8]:= SubstTest[subclass, intersection[u, v],
          u, {u -> fix[ACLOSURE], v -> fix[UCLOSURE]}] // Reverse

Out[8]= subclass[ALEX, fix[ACLOSURE]] == True

In[9]:= subclass[ALEX, fix[ACLOSURE]] := True
```

Corollary.

```
In[10]:= SubstTest[implies, and[subclass[u, v], subclass[v, w]],
           subclass[u, w], {u → ALEX, v → fix[ACLOSURE], w → binclosed[CAP]}] // Reverse

Out[10]= subclass[ALEX, binclosed[CAP]] == True

In[11]:= subclass[ALEX, binclosed[CAP]] := True
```

Theorem.

```
In[12]:= SubstTest[subclass, intersection[u, v],
           v, {u -> fix[ACLOSURE], v -> fix[UCLOSURE]}] // Reverse

Out[12]= subclass[ALEX, fix[UCLOSURE]] == True

In[13]:= subclass[ALEX, fix[UCLOSURE]] := True
```

Corollary.

```
In[14]:= SubstTest[implies, and[subclass[u, v], subclass[v, w]],
           subclass[u, w], {u → ALEX, v → fix[UCLOSURE], w → binclosed[CUP]}] // Reverse

Out[14]= subclass[ALEX, binclosed[CUP]] == True

In[15]:= subclass[ALEX, binclosed[CUP]] := True
```

Corollary. Every Alexandrov topology has a greatest member: **U[t] ∈ t**.

```
In[16]:= SubstTest[implies, and[subclass[u, v], subclass[v, w]],
           subclass[u, w], {u → ALEX, v → fix[UCLOSURE],
            w -> fix[composite[inverse[E], IMAGE[inverse[BIGCUP]]]]}] // Reverse

Out[16]= subclass[ALEX, fix[composite[inverse[E], IMAGE[inverse[BIGCUP]]]]] == True

In[17]:= subclass[ALEX, fix[composite[inverse[E], IMAGE[inverse[BIGCUP]]]]] := True
```

Corollary. The empty set belongs to every $t \in$ **ALEX**.

```
In[18]:= SubstTest[implies, and[subclass[u, v], subclass[v, w]], subclass[u, w],
           {u → ALEX, v → fix[UCLOSURE], w -> image[E, set[0]]}] // Reverse

Out[18]= member[0, A[ALEX]] == True

In[19]:= member[0, A[ALEX]] := True
```

a special case

Theorem. Every power set is an Alexandrov topology.

```
In[20]:= SubstTest[subclass, range[POWER], intersection[u, v],
           {u -> fix[ACLOSURE], v -> fix[UCLOSURE]}] // Reverse

Out[20]= subclass[range[POWER], ALEX] == True

In[21]:= subclass[range[POWER], ALEX] := True
```

Corollary. Every set is a member of an Alexandrov topology.

```
In[22]:= Map[assert, SubstTest[implies, subclass[u, v], subclass[image[t, u], image[t, v]],
            {t → inverse[E], u → range[POWER], v → ALEX}]] // Reverse

Out[22]= equal[V, U[ALEX]] == True

In[23]:= U[ALEX] := V
```

Corollary. The class **ALEX** is not a set.

```
In[24]:= Map[not, SubstTest[implies, equal[U[t], V], not[member[t, x]], t → ALEX]] // Reverse

Out[24]= member[ALEX, x] == False

In[25]:= member[ALEX, x_] := False
```

Corollary. Every set is a subset of an Alexandrov topology.

```
In[26]:= Map[assert, SubstTest[implies, subclass[u, v], subclass[image[t, u], image[t, v]],
            {t → inverse[S], u → range[POWER], v → ALEX}]] // Reverse

Out[26]= equal[V, image[inverse[S], ALEX]] == True
```

```
In[27]:= image[inverse[S], ALEX] := V
```

image[IRC, ALEX]

The function **IRC** takes any set **t** of sets to the set of their relative complements in **U[t]**. In particular, the function **IRC** takes the set of open sets for a topology to the set of closed sets for that topology. Vice versa, the function **IRC** also takes the set of closed sets back to the set of open sets.

```
In[28]:= APPLY[IRC, t]

Out[28]= union[complement[image[V, set[t]]], image[RC[U[t]], t]]
```

In this section it is shown that **IRC** takes any Alexandrov topology to another Alexandrov topology. In other words, the closed sets for any Alexandrov topology is the collection of open sets for another Alexandrov topology, and vice versa.

Lemma.

```
In[29]:= SubstTest[implies, and[subclass[u, v], subclass[v, w]], subclass[u, w],
           {u → image[IRC, ALEX], v → image[IRC, TOPS], w → fix[ACLOSURE]}] // Reverse

Out[29]= subclass[image[IRC, ALEX], fix[ACLOSURE]] == True

In[30]:= % /. Equal → SetDelayed
```

Lemma.

```
In[31]:= ImageComp[IRC, composite[IRC, id[fix[UCLOSURE]]], fix[ACLOSURE]] // Reverse

Out[31]= image[IRC, image[IRC, ALEX]] == ALEX

In[32]:= % /. Equal → SetDelayed
```

Theorem.

```
In[33]:= SubstTest[implies, subclass[u, v], subclass[image[t, u], image[t, v]],
           {t → IRC, u -> image[IRC, ALEX], v -> fix[ACLOSURE]}] // Reverse

Out[33]= subclass[ALEX, image[IRC, fix[ACLOSURE]]] == True

In[34]:= subclass[ALEX, image[IRC, fix[ACLOSURE]]] := True
```

Theorem.

```
In[35]:= SubstTest[subclass, ALEX, intersection[u, v, w],
           {u -> intersection[fix[ACLOSURE], binclosed[CUP]], v -> image[E, set[0]],
            w -> fix[composite[inverse[E], IMAGE[inverse[BIGCUP]]]]}] // Reverse

Out[35]= subclass[ALEX, image[IRC, TOPS]] == True

In[36]:= subclass[ALEX, image[IRC, TOPS]] := True
```

Lemma.

```
In[37]:= SubstTest[implies, subclass[u, v], subclass[image[t, u], image[t, v]],
           {t → IRC, u → ALEX, v -> image[IRC, TOPS]}] // Reverse

Out[37]= subclass[image[IRC, ALEX], TOPS] == True

In[38]:= % /. Equal → SetDelayed
```

Corollary.

```
In[39]:= SubstTest[implies, and[subclass[u, v], subclass[v, w]], subclass[u, w],
           {u -> image[IRC, ALEX], v → TOPS, w -> fix[UCLOSURE]}] // Reverse

Out[39]= subclass[image[IRC, ALEX], fix[UCLOSURE]] == True

In[40]:= % /. Equal → SetDelayed
```

Lemma.

```
In[41]:= SubstTest[subclass, image[IRC, ALEX],
           intersection[u, v], {u -> fix[ACLOSURE], v -> fix[UCLOSURE]}] // Reverse

Out[41]= subclass[image[IRC, ALEX], ALEX] == True

In[42]:= % /. Equal → SetDelayed
```

Lemma.

```
In[43]:= SubstTest[implies, subclass[u, v], subclass[image[t, u], image[t, v]],
           {t → IRC, u -> image[IRC, ALEX], v → ALEX}] // Reverse

Out[43]= subclass[ALEX, image[IRC, ALEX]] == True

In[44]:= % /. Equal → SetDelayed
```

Theorem.

```
In[45]:= SubstTest[and, subclass[u, v], subclass[v, u], {u -> ALEX, v -> image[IRC, ALEX]}]

Out[45]= equal[ALEX, image[IRC, ALEX]] == True

In[46]:= image[IRC, ALEX] := ALEX
```