# topological bases

Johan G. F. Belinfante
2011 March 24

```
In[1]:= SetDirectory["l:"]; << goedel.11mar23a

        :Package Title: goedel.11mar23a               2011 March 23 at 9:10 a.m.

        It is now:  2011 Mar 24 at 10:18

        Loading Simplification Rules

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

        weightlimit = 40
```

## summary

A collection **x** of sets is a **topological base** if **x** is a set, and the following holds:

```
In[2]:= forall[u, v, w, implies[and[member[u, x], member[v, x],
                    member[w, intersection[u, v]]],
                exists[t, and[member[t, x], member[w, t],
                        subclass[t, intersection[u, v]]]]]] // assert

Out[2]= subclass[image[CAP, cart[x, x]], Uclosure[x]]
```

It is shown in this notebook that if **x** is a topological base, then **Uclosure[x]** is a topology.

## references

Topological bases are discussed on pages 46-47 of the following reference.

```
In[3]:= "John L. Kelley, General Topology,
           D. Van Nostrand Co., Inc., Princeton, N. J., 1955.";
```

The results in this notebook are essentially the content of Kelley's theorem 11 on page 47. A slight variant of the definition given above was used by William McCune and Cynthia Wick in a remarkable paper which used **Otter** to derive results in topology when automated reasoning in set theory was still in its infancy.

```
In[4]:= "William W. McCune and Cynthia A. Wick,Automated reasoning about elementary point
           set topology, Journal of Automated Reasoning, vol. 5(1989), pp. 239–255.";
```

## derivation

Theorem.

```
In[5]:= SubstTest[implies, and[subclass[u, v], subclass[v, w]],
          subclass[u, w], {u → image[CAP, cart[Uclosure[x], Uclosure[x]]],
           v → Uclosure[image[CAP, cart[x, x]]], w → Uclosure[x]}] // Reverse

Out[5]= or[not[subclass[image[CAP, cart[x, x]], Uclosure[x]]],
          subclass[image[CAP, cart[Uclosure[x], Uclosure[x]]], Uclosure[x]]] == True

In[6]:= or[not[subclass[image[CAP, cart[x_, x_]], Uclosure[x_]]],
          subclass[image[CAP, cart[Uclosure[x_], Uclosure[x_]]], Uclosure[x_]]] := True
```

Lemma.

```
In[7]:= Map[implies[member[x, y], #] &,
          SubstTest[implies, and[member[t, V], subclass[image[CAP, cart[t, t]], t],
            equal[t, Uclosure[t]]], member[t, TOPS], t → Uclosure[x]]] // Reverse

Out[7]= or[member[Uclosure[x], TOPS], not[member[x, y]],
          not[subclass[image[CAP, cart[Uclosure[x], Uclosure[x]]], Uclosure[x]]]] == True

In[8]:= or[member[Uclosure[x_], TOPS], not[member[x_, y_]],
          not[subclass[image[CAP, cart[Uclosure[x_], Uclosure[x_]]], Uclosure[x_]]]] := True
```

Theorem. If **x** is a topological base, then **Uclosure[x]** is a topology.

```
In[9]:= Map[not, SubstTest[and, implies[p1, p2], not[implies[and[p0, p1], p3]],
            {p0 → member[x, y], p1 -> subclass[image[CAP, cart[x, x]], Uclosure[x]],
             p2 -> subclass[image[CAP, cart[Uclosure[x], Uclosure[x]]], Uclosure[x]],
             p3 → member[Uclosure[x], TOPS]}]] // Reverse

Out[9]= or[member[Uclosure[x], TOPS], not[member[x, y]],
          not[subclass[image[CAP, cart[x, x]], Uclosure[x]]]] == True

In[10]:= or[member[Uclosure[x_], TOPS], not[member[x_, y_]],
           not[subclass[image[CAP, cart[x_, x_]], Uclosure[x_]]]] := True
```

## a variable-free restatement

Lemma. A sethood result.

```
In[11]:= SubstTest[member, U[t], V, t → image[CAP, cart[x, y]]]

Out[11]= member[image[CAP, cart[x, y]], V] == member[intersection[U[x], U[y]], V]

In[12]:= member[image[CAP, cart[x_, y_]], V] := member[intersection[U[x], U[y]], V]
```

Lemma.

```
In[13]:= Map[implies[and[member[x, V], #], member[Uclosure[x], TOPS]] &,
          member[pair[cart[x, x], x], composite[inverse[UCLOSURE], S, IMAGE[CAP]]] // AssertTest]

Out[13]= or[member[Uclosure[x], TOPS],
           not[member[pair[cart[x, x], x], composite[inverse[UCLOSURE], S, IMAGE[CAP]]]]] == True

In[14]:= (% /. x → x_) /. Equal → SetDelayed
```

Theorem.

```
In[15]:= Map[equal[V, #] &,
          complement[dif[fix[composite[inverse[UCLOSURE], S, IMAGE[CAP], CART, DUP]],
            image[inverse[UCLOSURE], TOPS]]] // Normality]

Out[15]= subclass[image[UCLOSURE,
            fix[composite[inverse[UCLOSURE], S, IMAGE[CAP], CART, DUP]]], TOPS] == True

In[16]:= % /. Equal → SetDelayed
```

Lemma.

```
In[17]:= SubstTest[subclass, fix[composite[S, x]], fix[composite[inverse[UCLOSURE], S, x]],
           x -> composite[IMAGE[CAP], CART, DUP]] // Reverse

Out[17]= subclass[binclosed[CAP],
           fix[composite[inverse[UCLOSURE], S, IMAGE[CAP], CART, DUP]]] == True

In[18]:= subclass[binclosed[CAP],
           fix[composite[inverse[UCLOSURE], S, IMAGE[CAP], CART, DUP]]] := True
```

Lemma.

```
In[19]:= SubstTest[implies, subclass[u, v],
           subclass[image[t, u], image[t, v]], {t → UCLOSURE, u -> binclosed[CAP],
            v -> fix[composite[inverse[UCLOSURE], S, IMAGE[CAP], CART, DUP]]}] // Reverse

Out[19]= subclass[TOPS,
           image[UCLOSURE, fix[composite[inverse[UCLOSURE], S, IMAGE[CAP], CART, DUP]]]] == True

In[20]:= % /. Equal → SetDelayed
```

Theorem. A variable-free statement of the fact that the Uclosure of a topological base is a topology.

```
In[21]:= SubstTest[and, subclass[u, v], subclass[v, u], {u -> TOPS,
           v -> image[UCLOSURE, fix[composite[inverse[UCLOSURE], S, IMAGE[CAP], CART, DUP]]]}]

Out[21]= equal[TOPS,
           image[UCLOSURE, fix[composite[inverse[UCLOSURE], S, IMAGE[CAP], CART, DUP]]]] == True

In[22]:= image[UCLOSURE, fix[composite[inverse[UCLOSURE], S, IMAGE[CAP], CART, DUP]]] := TOPS
```

Finally, a variable-free statement will be derived of the following fact that was used in the derivation presented in the preceding section.

```
In[23]:= subclass[image[CAP, cart[Uclosure[x], Uclosure[y]]], Uclosure[image[CAP, cart[x, y]]]]

Out[23]= True
```

Lemma.

```
In[24]:= dif[composite[IMAGE[CAP], CART, cross[UCLOSURE, UCLOSURE]],
           composite[inverse[S], UCLOSURE, IMAGE[CAP], CART]] // ReifTriNormality

Out[24]= composite[intersection[composite[IMAGE[CAP], CART, cross[UCLOSURE, UCLOSURE]],
            composite[complement[inverse[S]], UCLOSURE, IMAGE[CAP], CART]], id[cart[V, V]]] == 0

In[25]:= % /. Equal → SetDelayed
```

Theorem.

```
In[26]:= SubstTest[empty, composite[dif[u, v], id[cart[V, V]]],
          {u -> composite[IMAGE[CAP], CART, cross[UCLOSURE, UCLOSURE]],
           v -> composite[inverse[S], UCLOSURE, IMAGE[CAP], CART]}]

Out[26]= subclass[composite[IMAGE[CAP], CART, cross[UCLOSURE, UCLOSURE]],
           composite[inverse[S], UCLOSURE, IMAGE[CAP], CART]] == True

In[27]:= subclass[composite[IMAGE[CAP], CART, cross[UCLOSURE, UCLOSURE]],
           composite[inverse[S], UCLOSURE, IMAGE[CAP], CART]] := True
```