

the box topology

Johan G. F. Belinfante
2011 March 5

```
In[1]:= SetDirectory["1:"]; << goedel.11mar04b

:Package Title: goedel.11mar04b          2011 March 4 at 2:10 p.m.

It is now: 2011 Mar 5 at 3:56

Loading Simplification Rules

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

weightlimit = 40
```

summary

For any two topologies **top[x]** and **top[y]**, the collection **image[CART, cart[top[x], top[y]]]** is a basis for a topology, called the **box topology**.

derivation

Theorem. Basis for the box topology.

```
In[3]:= ImageComp[CAP, composite[cross[CART, CART], TWIST],
             cart[cartsq[top[x]], cartsq[top[y]]]] // Reverse

Out[3]= image[CAP, cart[image[CART, cart[top[x], top[y]]], image[CART, cart[top[x], top[y]]]]] ==
         image[CART, cart[top[x], top[y]]]

In[4]:= image[CAP, cart[image[CART, cart[top[x_], top[y_]]],
                     image[CART, cart[top[x_], top[y_]]]]] := image[CART, cart[top[x], top[y]]]
```

Theorem. The box topology.

```
In[6]:= SubstTest[implies, and[member[t, u], subclass[u, v]],
                member[t, v], {t -> image[CART, cart[top[x], top[y]]],
                               u -> binclosed[CAP], v -> image[inverse[UCLOSURE], TOPS]}] // Reverse

Out[6]= member[Uclosure[image[CART, cart[top[x], top[y]]], TOPS] == True

In[7]:= member[Uclosure[image[CART, cart[top[x_], top[y_]]], TOPS] := True
```

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

```
In[8]:= SubstTest[implies, and[equal[x, top[u]], equal[y, top[v]]],
  member[Uclosure[image[CART, cart[x, y]]], TOPS], {u → x, v → y}] // Reverse
```

```
Out[8]= or[member[Uclosure[image[CART, cart[x, y]]], TOPS],
  not[member[x, TOPS]], not[member[y, TOPS]]] == True
```

```
In[9]:= or[member[Uclosure[image[CART, cart[x_, y_]]], TOPS],
  not[member[x_, TOPS]], not[member[y_, TOPS]]] := True
```

Theorem. A variable-free restatement. The class of topologies is binary closed under the box operation.

```
In[12]:= Map[empty[composite[Id, complement[#]]] &, complement[dif[cartsq[TOPS],
  image[inverse[composite[UCLOSURE, IMAGE[CART], CART]], TOPS]]] // ReInNormality]
```

```
Out[12]= subclass[image[UCLOSURE, image[IMAGE[CART], image[CART, cart[TOPS, TOPS]]]], TOPS] ==
  True
```

```
In[13]:= subclass[
  image[UCLOSURE, image[IMAGE[CART], image[CART, cart[TOPS, TOPS]]]], TOPS] := True
```