a compact topology on omega

Johan G. F. Belinfante
2004 October 12

summary

This notebook contains a simple construction of a compact topology on the natural numbers, as well as some other related results. A class \( x \) is compact if every coarser collection is coarser than a finite collection. In this context, a coarser collection means a set \( y \) satisfying subclass\( [y, x] \) and \( U[y] = U[x] \), that is,

\[
\text{In[2]:= member[pair[y, x], COARSER]}
\]

\[
\text{Out[2]= and[equal[U[x], U[y]], member[x, V], subclass[y, x]]}
\]

The definition of compact collection does not require that it be a topology. In the case that \( x \) is a topology, the underlying space is \( U[x] \). In this case, any collection \( y \) coarser than \( x \) is conventionally called an open cover of the space, and a collection coarser than \( y \) is called a subcover. Note that in this context it is the class \( U[x] \) that is being covered, and not \( x \) itself.

omega

Every finite collection of sets is compact, so in particular every natural number is compact:
omega is not compact

Since any collection is coarser than itself, any compact collection must be coarser than a finite one.

As a corollary, one obtains:

From this it follows that no collection coarser than omega can be finite.

The fact that omega is not compact follows:
The class `omega` is coarser than `succ[omega]`, but there is no finite set coarser than `omega`. Consequently, `succ[omega]` is not compact.

On the other hand, `succ[omega]` is a topology, and so this provides a simple example of a topology that is not compact.

Since `singleton[omega]` is a finite collection that is coarser than `succ[omega]`, one has:

The class `COMPACT` is sandwiched between `FINITE` and `image[COARSER, FINITE]`. Examples will be given to show that both of these inclusions are proper.
Since `succ[omega]` belongs to `image[COARSER,FINITE]` but not to `COMPACT`, it follows that the second inclusion is proper.

```plaintext
In[20]:= Map[not, SubstTest[implies, and[member[u, v], equal[v, w]], member[u, w], 
{u → succ[omega], v → image[COARSER, FINITE], w → COMPACT}]]
Out[20]= equal[COMPACT, image[COARSER, FINITE]] == False

In[21]:= equal[COMPACT, image[COARSER, FINITE]] := False
```

Membership in `COMPACT` means every coarser collection is coarser than a finite collection, whereas membership in `image[COARSER, FINITE]` is the weaker requirement that there is a coarser collection which is finite.

---

**an infinite compact collection**

The class `image[RC[omega], omega]` provides an example of an infinite compact collection. Each member of this class is the relative complement in `omega` of a natural number, that is, the set of all natural numbers greater than or equal to a given one. The proof that this collection is compact follows from the observation that any collection coarser than `image[RC[omega],omega]` must hold the set `omega` because that is the only set in the collection which holds the empty set. It follows that `singleton[omega]` is a finite collection coarser than any collection coarser than `image[-RC[omega],omega]`.

```plaintext
In[22]:= member[0, A[intersection[omega, x]]] // AssertTest
Out[22]= member[0, A[intersection[omega, x]]] == not[member[0, x]]

In[23]:= member[0, A[intersection[omega, x_]]] := not[member[0, x]]
```

Write `image[RC[omega],omega]` as the union of the complement of `0` and complements of nonzero numbers:

```plaintext
In[24]:= SubstTest[image, w, union[u, v], 
{u → dif[omega, singleton[0]], v → singleton[0], w → RC[omega]}] // Reverse
Out[24]= union[image[RC[omega], intersection[omega, complement[singleton[0]]]]],
    singleton[omega]] == image[RC[omega], omega]

In[25]:= union[image[RC[omega], intersection[omega, complement[singleton[0]]]]],
    singleton[omega]] := image[RC[omega], omega]
```
The latter complements do not hold 0.

\[
\text{In[26]} := \text{SubstTest[subclass, union[u, x], union[v, y],}
\]
\[
\{ u \rightarrow \text{image[RC[omega], \text{dif[omega, singleton[0]]}]},
\]
\[
v \rightarrow \text{P[complement[singleton[0]]]},
\]
\[
x \rightarrow \text{image[RC[omega], singleton[0]]},
\]
\[
y \rightarrow \text{singleton[omega]}
\]

\[
\text{Out[26]} := \text{subclass[image[RC[omega], omega]},
\]
\[
\text{union[P[complement[singleton[0]]], singleton[omega]]} = \text{True}
\]

\[
\text{In[27]} :=\ % / . \ \text{Equal} \rightarrow \text{SetDelayed}
\]

This inclusion can be sharpened to an equation:

\[
\text{In[28]} := \text{equal[intersection[complement[\text{P[complement[singleton[0]]]}],}
\]
\[
\text{image[RC[omega], omega]}], \text{singleton[omega]} \} / / \text{AssertTest}
\]

\[
\text{Out[28]} := \text{equal[intersection[complement[\text{P[complement[singleton[0]]]}],}
\]
\[
\text{image[RC[omega], omega]}], \text{singleton[omega]}\} = \text{True}
\]

\[
\text{In[29]} := \text{intersection[complement[\text{P[complement[singleton[0]]]}],}
\]
\[
\text{image[RC[omega], omega]} := \text{singleton[omega]}
\]

Introducing a variable \( x \), one obtains:

\[
\text{In[30]} := \text{Map[\text{implies}[\#, \text{equal[omega, x]]} \&, \text{SubstTest[member, x, intersection[y, z],}
\]
\[
\{ y \rightarrow \text{complement[\text{P[complement[singleton[0]]]}]},
\]
\[
z \rightarrow \text{image[RC[omega], omega]}\} \} / / \text{Reverse}
\]

\[
\text{Out[30]} := \text{or[equal[omega, x]}, \text{not[member[0, x]]},
\]
\[
\text{not[member[intersection[omega, \text{complement[x]}], omega]}],
\]
\[
\text{not[subclass[x, omega]]} = \text{True}
\]

\[
\text{In[31]} :=\ % / . \ \text{Equal} \rightarrow \text{SetDelayed}
\]

This says that \( \text{omega} \) is the only member of \( \text{image[RC[omega],omega]} \) that holds 0.

\[
\text{In[32]} := \text{implies[}
\]
\[
\text{and[member[x, image[RC[omega], omega]]], member[0, x]], equal[omega, x]}\]

\[
\text{Out[32]} := \text{True}
\]

From this will deduce that any collection \( x \) coarser than \( \text{image[RC[omega],omega]} \)
must hold \( \text{omega} \). The argument is this: if \( \text{U[x]} \) is omega, then 0 belongs to \( \text{U[x]} \).
So 0 must belong to some member of \( x \). But that member of \( x \) must belong to \( \text{image[RC[omega],omega]} \), and the only member of the latter that holds 0 is \( \text{omega} \). So \( \text{omega} \) must belong to \( x \).
The variable \( y \) can be eliminated as follows:

\[
\text{In[37]:=} \quad \text{Map[equal[V, \#] \&, SubstTest[class, y,}
\quad \quad \quad \text{or[member[omega, x], not[member[0, y]], not[member[y, x]],}
\quad \quad \quad \text{not[subclass[x, z]]], z \rightarrow \text{image[RC[omega], omega]]}]} // \text{Reverse}
\]

\[
\text{Out[37]=} \quad \text{or[member[omega, x], not[member[0, U[x]]],}
\quad \quad \quad \text{not[subclass[x, \text{image[RC[omega], omega]]]]} = \text{True}
\]

\[
\text{In[38]=} \quad (\% / . (x \rightarrow x_\_)) / \text{. Equal} \rightarrow \text{SetDelayed}
\]

If a covering of \( \omega \) that holds \( \omega \) has a finite subcovering, namely \( \text{singleton[omega]} \).

\[
\text{In[39]=} \quad \text{SubstTest[implies, and[member[pair[u, v], composite[Id, z]], member[u, y]],}
\quad \quad \quad \text{member[v, image[z, y]], \{u \rightarrow \text{singleton[omega]}, v \rightarrow x, y \rightarrow \text{FINITE, z \rightarrow \text{COARSER}}\}]]}
\]

\[
\text{Out[39]=} \quad \text{or[member[x, image[\text{COARSER}, \text{FINITE}]], not[equal[omega, U[x]]],}
\quad \quad \quad \text{not[member[omega, x]], not[member[x, V]]]} = \text{True}
\]

\[
\text{In[40]=} \quad (\% / . (x \rightarrow x_\_)) / \text{. Equal} \rightarrow \text{SetDelayed}
\]

These results can be combined:
The set \( \text{image[RC[omega], omega]} \) is a compact collection.

This compact set is not finite.

It follows that the inclusion \( \text{subclass[FINITE, COMPACT]} \) is a proper one.

a basis for an infinite compact topology

The class \( \text{image[RC[omega], omega]} \) is totally ordered by inclusion. To derive this fact, one begins with the fact that \( \text{omega} \) is totally ordered by inclusion:
In[49]:= SubstTest[implies, subclass[u, v], subclass[\{\text{image}[w, u], \text{image}[w, v]\}, \{\text{cart}[\text{omega}, \text{omega}], v \rightarrow \text{union}[S, \text{inverse}[S]], w \rightarrow \text{cross}[\text{RC}[\text{omega}}, \text{RC}[\text{omega}])\}]

Out[49]= subclass[\{\text{cart}[\text{image}[\text{RC}[\text{omega}]], \text{omega}], \text{image}[\text{RC}[\text{omega}]], \text{omega}\}], \text{union}[\text{composite}[\text{id}[\text{P}[\text{omega}]], S], \text{composite}[\text{inverse}[\text{S}], \text{id}[\text{P}[\text{omega}])], \text{True}]

In[50]:= \% /. Equal \rightarrow \text{SetDelayed}

This result just needs to be cleaned up:

In[51]:= SubstTest[implies, and[subclass[u, v], subclass[w, v]], subclass[u, w], \{u \rightarrow \text{cart}[\text{image}[\text{RC}[\text{omega}]], \text{omega}], \text{image}[\text{RC}[\text{omega}]], \text{omega}\}, v \rightarrow \text{union}[\text{composite}[\text{id}[\text{P}[\text{omega}]], S], \text{composite}[\text{inverse}[\text{S}], \text{id}[\text{P}[\text{omega}])], w \rightarrow \text{union}[S, \text{inverse}[S]])]

Out[51]= subclass[\{\text{cart}[\text{image}[\text{RC}[\text{omega}]], \text{omega}], \text{image}[\text{RC}[\text{omega}]], \text{omega}\}], \text{union}[S, \text{inverse}[S]]) = \text{True}

In[52]:= subclass[\{\text{cart}[\text{image}[\text{RC}[\text{omega}]], \text{omega}], \text{image}[\text{RC}[\text{omega}]], \text{omega}\}], \text{union}[S, \text{inverse}[S]]) := \text{True}

It follows that this class is closed under binary intersections, and is therefore a basis for a topology.

In[53]:= SubstTest[implies, and\[member[u, v], subclass[v, w]]}, member[u, w], \{u \rightarrow \text{image}[\text{RC}[\text{omega}]], \text{omega}], v \rightarrow \text{cliques}[\text{union}[S, \text{inverse}[S]]], w \rightarrow \text{binclosed}[\text{CAP}]\}]

Out[53]= subclass[\{\text{image}[\text{CAP}], \text{cart}[\text{image}[\text{RC}[\text{omega}]], \text{omega}], \text{image}[\text{RC}[\text{omega}]], \text{omega}\}], \text{image}[\text{RC}[\text{omega}]], \text{omega}]) = \text{True}

In[54]:= subclass[\{\text{image}[\text{CAP}], \text{cart}[\text{image}[\text{RC}[\text{omega}]], \text{omega}], \text{image}[\text{RC}[\text{omega}]], \text{omega}\}], \text{image}[\text{RC}[\text{omega}]], \text{omega}]) := \text{True}

This class is also closed under binary unions:

In[55]:= SubstTest[implies, and\[member[u, v], subclass[v, w]]}, member[u, w], \{u \rightarrow \text{image}[\text{RC}[\text{omega}]], \text{omega}], v \rightarrow \text{cliques}[\text{union}[S, \text{inverse}[S]]], w \rightarrow \text{binclosed}[\text{CUP}]\}]

Out[55]= subclass[\{\text{image}[\text{CUP}], \text{cart}[\text{image}[\text{RC}[\text{omega}]], \text{omega}], \text{image}[\text{RC}[\text{omega}]], \text{omega}\}], \text{image}[\text{RC}[\text{omega}]], \text{omega}]) = \text{True}

In[56]:= subclass[\{\text{image}[\text{CUP}], \text{cart}[\text{image}[\text{RC}[\text{omega}]], \text{omega}], \text{image}[\text{RC}[\text{omega}]], \text{omega}\}], \text{image}[\text{RC}[\text{omega}]], \text{omega}]) := \text{True}
Although the class \texttt{image[RC[omega],omega]} is closed under binary unions, it is not closed under arbitrary unions for the simple fact that it fails to hold the union of the empty set. To obtain a topology, one therefore needs to add the empty set.

topology

One obtains a topology from a topological base by applying \texttt{Uclosure}:

\begin{verbatim}
In[57]:= Map[implies[member[x, y], #] &, 
    SubstTest[implies, and[member[x, u], subclass[u, v]], member[x, v], 
{u \rightarrow \text{binclosed}[\text{CAP}], v \rightarrow \text{image}[\text{inverse}[\text{UCLOSE}], \text{TOPS}]}]]
Out[57]= or[member[Uclosure[x], \text{TOPS}], not[member[x, y]], 
not[subclass[\text{image}[\text{CAP}, \text{cart}[x, x]], x]]] := \text{True}

In[58]:= or[member[Uclosure[x_], \text{TOPS}], not[member[x_, y_]], 
not[subclass[\text{image}[\text{CAP}, \text{cart}[x_, x_]], x_]]] := \text{True}
\end{verbatim}

To carry out this idea, one needs a formula for the \texttt{Uclosure} of the class \texttt{image[-RC[omega], omega]}. The net result, as we shall see momentarily, amounts to adding the empty set.

\begin{verbatim}
In[59]:= SubstTest[image, RC[omega], union[omega, x], x \rightarrow \text{singleton[omega]}]
Out[59]= image[RC[omega], \text{succ[omega]]} := \text{union[image[RC[omega], \text{omega}], \text{singleton[0]}}]

In[60]:= image[RC[omega], \text{succ[omega]}] := \text{union[image[RC[omega], \text{omega}], \text{singleton[0]}}
\end{verbatim}

Some rewrite rules will be needed to simplify various expressions that appear in the derivation.

\begin{verbatim}
In[61]:= equal[\text{intersection}[P[omega], \text{succ[omega]}], \text{succ[omega]}]
Out[61]= \text{True}
\end{verbatim}

This fact is made into a rewrite rule:

\begin{verbatim}
In[62]:= \text{intersection}[P[omega], \text{succ[omega]]} := \text{succ[omega]}
\end{verbatim}

Here is another such fact, which is also made into a rewrite rule.
For finite sets, there is no difference between \texttt{Aclosure[x]} and \texttt{fix[HULL[x]].} In particular:

\begin{verbatim}
In[65]:- SubstTest[implies, member[x, V],
    equal[fix[HULL[x]], Aclosure[x], x \rightarrow succ[omega]]
Out[65]: equal[fix[HULL[succ[omega]]], succ[omega]] = True
In[66]:- fix[HULL[succ[omega]]]) := succ[omega]
\end{verbatim}

The needed \texttt{Uclosure} formula now follows:

\begin{verbatim}
In[67]:- Map[fix, Map[composite[RC[omega], #, RC[omega]] & SubstTest[HULL,
    image[RC[omega], x], x \rightarrow image[RC[omega], succ[omega]]]]] // Reverse
Out[67]: Uclosure[image[RC[omega], omega]] :=
            union[image[RC[omega], omega], singleton[0]]
In[68]:- Uclosure[image[RC[omega], omega]] :=
            union[image[RC[omega], omega], singleton[0]]
\end{verbatim}

This yields the promised example of a compact topology on the natural numbers:

\begin{verbatim}
In[69]:- SubstTest[implies, member[x, binclosed[CAP]],
    member[Uclosure[x], TOPS], x \rightarrow image[RC[omega], omega]]
Out[69]: member[union[image[RC[omega], omega], singleton[0]], TOPS] = True
In[70]:- member[union[image[RC[omega], omega], singleton[0]], TOPS] := True
\end{verbatim}

This is indeed a topology on the set \texttt{omega}:

\begin{verbatim}
In[71]:- U[union[image[RC[omega], omega], singleton[0]]]
Out[71]: omega
\end{verbatim}

The statement that this topology is compact is recognized automatically from a rewrite rule that says that compactness is not affected by adding the empty set.
This topology, is of course, not finite:

Comment. The topology that has been constructed in this notebook bears some resemblance to the cofinite topology, \texttt{Uclosure[image[RC[omega], FINITE]]}. The latter is also compact, but the derivation of that fact is not as simple.