unions of chains of squares

Johan G. F. Belinfante
2011 November 30

In[1]:= SetDirectory["l:”; << goedel.11nov29a

:Package Title: goedel.11nov29a 2011 November 29 at 10:20 a.m.

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

It is now: 2011 Nov 30 at 12:22

Loading Simplification Rules

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

weightlimit = 40

Loading completed.

It is now: 2011 Nov 30 at 12:34

summary

The union of a chain of squares is a square. The following concise variable-free statement of this is already available.

In[2]:= Uchains[image[CART, Id]]

Out[2]= image[CART, Id]

In this notebook a more general result is derived, dropping the requirement that the chain be a set. Some related results are also derived. Several derivations can be speeded up by a few seconds if one omits certain proof steps. The omitted steps are indicated by (* ... *).

derivation

It is convenient to introduce the following temporary wrapper for collections of squares.

In[3]:= sq[x_] := image[CART, id[image[IMAGE[inverse[DUP]], x]]]

The introduction rule for this wrapper is already available.

In[4]:= subclass[sq[x], image[CART, Id]]

So is the elimination rule.

\[
\text{In}[5] := \text{equal}[x, \text{sq}[x]]
\]

\[
\text{Out}[5] := \text{subclass}[x, \text{image}[\text{CART}, \text{Id}]]
\]

Two immediate applications of this wrapper will now be given for arbitrary collections of squares.

Theorem. The union of any collection of squares is a reflexive relation.

\[
\text{In}[6] := \text{SubstTest}[\text{implies, equal}[x, \text{sq}[t]], \text{REFLEXIVE}[\text{U}[x]], t \rightarrow x] \text{ // Reverse}
\]

\[
\text{Out}[6] := \text{or}[\text{not}[\text{subclass}[x, \text{image}[\text{CART}, \text{Id}]]], \text{REFLEXIVE}[\text{U}[x]]] = \text{True}
\]

\[
\text{In}[7] := \text{or}[\text{not}[\text{subclass}[x_, \text{image}[\text{CART}, \text{Id}]]], \text{REFLEXIVE}[\text{U}[x_]]] := \text{True}
\]

Theorem. The union of any collection of squares is a symmetric relation.

\[
\text{In}[8] := \text{SubstTest}[\text{implies, equal}[x, \text{sq}[t]], \text{SYMMETRIC}[\text{U}[x]], t \rightarrow x] \text{ // Reverse}
\]

\[
\text{Out}[8] := \text{or}[\text{equal}[\text{inverse}[\text{U}[x]], \text{U}[x]], \text{not}[\text{subclass}[x, \text{image}[\text{CART}, \text{Id}]]]] = \text{True}
\]

\[
\text{In}[9] := \text{or}[\text{equal}[\text{inverse}[\text{U}[x_]], \text{U}[x_]], \text{not}[\text{subclass}[x_, \text{image}[\text{CART}, \text{Id}]]]] := \text{True}
\]

Theorem. Symmetric rectangles are square.

\[
\text{In}[10] := \text{Map}[\text{not, SubstTest}[\text{and, implies[and[p1, p2], p3], not[implies[and[p1, p2], p4]],}
\]

\[
\text{p1} \rightarrow \text{equal}[x, \text{cart}[\text{domain}[x], \text{range}[x]]], \text{p2} \rightarrow \text{equal}[x, \text{inverse}[x]],
\]

\[
\text{p3} \rightarrow \text{equal}[\text{domain}[x], \text{range}[x]], \text{p4} \rightarrow \text{equal}[x, \text{cart}[\text{fix}[x], \text{fix}[x]]]]] \text{ // Reverse}
\]

\[
\text{Out}[10] := \text{or}[\text{equal}[x, \text{cart}[\text{fix}[x], \text{fix}[x]]], \text{not}[\text{equal}[x, \text{inverse}[x]]]] = \text{True}
\]

\[
\text{In}[11] := \text{or}[\text{equal}[\text{cart}[\text{fix}[x_], \text{fix}[x_]], x_], \text{not}[\text{equal}[\text{cart}[\text{domain}[x_], \text{range}[x_]], x_]], \text{not}[\text{equal}[\text{inverse}[x_], x_]]] := \text{True}
\]

Theorem. The union of a chain of squares is a square.

\[
\text{In}[12] := \text{Map}[\text{not, SubstTest}[\text{and, (\text{implies[and[p1,p3],* implies[and[p2, p3], p4],}}]
\]

\[
\text{implies[and[p1, p2], p5], (\text{implies[and[p4,p5],p6],* not[implies[and[p1, p2], p6]]},}
\]

\[
\text{p1} \rightarrow \text{subclass}[x, \text{image}[\text{CART}, \text{Id}]], \text{p2} \rightarrow \text{subclass}[\text{P}[x], \text{chains}[S]],
\]

\[
\text{p3} \rightarrow \text{subclass}[x, \text{range}[\text{CART}]], \text{p4} \rightarrow \text{RECTANGLE}[\text{U}[x]],
\]

\[
\text{p5} \rightarrow \text{SYMMETRIC}[\text{U}[x]], \text{p6} \rightarrow \text{equal}[\text{U}[x], \text{cartsq[fix[U[x]]]}})]] \text{ // Reverse}
\]

\[
\text{Out}[12] := \text{or}[\text{equal}[\text{cart}[\text{fix[U[x]]}, \text{fix[U[x]]]], \text{U}[x]], \text{not}[\text{subclass}[x, \text{image}[\text{CART}, \text{Id}]]],
\]

\[
\text{not}[\text{subclass}[\text{cart}[x, x], \text{union}[S, \text{inverse}[S]]]]] = \text{True}
\]

\[
\text{In}[13] := \text{or}[\text{equal}[\text{cart}[\text{fix[U[x_]]}, \text{fix[U[x_]]]], \text{U}[x_]], \text{not}[\text{subclass}[\text{cart}[x_, x_], \text{union}[S, \text{inverse}[S]]]],
\]

\[
\text{not}[\text{subclass}[x_, \text{image}[\text{CART}, \text{Id}]]]] := \text{True}
\]