two-sided restrictions to final segments

Johan G. F. Belinfante
2010 July 15

In[1]:= SetDirectory["l:" ]; << goedel.10jul13a; << tools.m

Package Title: goedel.10jul13a 2010 July 13 at 12:20 p.m.
It is now: 2010 Jul 15 at 14:45
Loading Simplification Rules
TOOLS.M Revised 2010 February 26
weightlimit = 40

summary

In the recently posted notebook trv-fnsg.nb a variable-free statement about transitivity for final segments was derived. In the present notebook another variable-free statement concerning the transitivity of final segments is derived in which the predicate TRANSITIVE occurs explicitly.

acknowledgment

Theorem 8.10 on page 32 in the following reference inspired this work.


temporary abbreviations

Two-sided restriction can be viewed as an action of sets on relations. If \( s \) is a set and \( r \) is a relation, then this action of \( s \) on \( r \) could be denoted by \( s \cdot r = \text{restrict}[r, s, s] \). Actually one can allow \( r \) to be any set. There is no need to require \( r \) to be a relation. The following temporary abbreviation for the binary function for this action will be used.

In[3]:= TWOSIDED := composite[CAP, cross[composite[CART, DUP], Id]]

The APPLY rule for this action simplifies when one uses setpart wrappers:

In[4]:= APPLY[TWOSIDED, PAIR[setpart[s], setpart[r]]]
Out[4]= composite[id[setpart[s]], setpart[r], id[setpart[s]]]
Harzheim defines a set \( s \) to be a **final segment** of a relation \( r \) if \( \text{image}[r, s] = s \). The following temporary abbreviation for the relation of being a final segment will be used here.

\[
\text{FINSEG} := \text{fix[composite[inverse[SECOND], IMG]]}
\]

The following membership rule for this relation helps explain its significance.

\[
\text{FNRS} := \text{composite[TWOSIDED, id[inverse[FINSEG]]]}
\]

It will be shown below that the relation \( \text{FNRS} \circ \text{inverse[SECOND]} \) is transitive. An ordered pair of relations \( \text{pair}[x, y] \) belongs to this transitive relation if \( y \) is the two-sided restriction of \( x \) to some final segment of \( x \).

---

**a quasi-associative law**

The **TWOSIDED** action satisfies the following quasi-associative law: \( u \cdot (v \cdot x) = (u \cap v) \cdot x \). This quasi-associative law simplifies for the restricted action **FNRS** in that \( u \cap v \) can be replaced with \( u \). Explicitly, if one writes \( r \cdot s \) for this restriction of the action \( r \cdot s \), the quasi-associative law effectively becomes the inclusion \( u \cdot (v \cdot x) \subseteq u \cdot x \). (Note that one must replace equality by inclusion here because the restricted action is no longer total. The following lemma formally expresses this simplification of the quasi-associative law for the action **FNRS** in the form of an implication involving two additional variables: \( y = v \cdot x \) and \( z = u \cdot y \Rightarrow z = u \cdot x \).

Lemma. A quasi-associative law for the action **FNRS** with five variables wrapped with \( \text{setpart} \).

\[
\text{implies[and[member[pair[setpart[u], setpart[x]], setpart[y]], \text{FNRS}], member[pair[setpart[v], setpart[y]], setpart[z]], \text{FNRS}], member[pair[setpart[v], setpart[x]], setpart[z]], \text{FNRS}] // NotNotTest}
\]

\[
\text{or[and[equal[composite[id[setpart[v]], setpart[x], id[setpart[v]]], setpart[z]], equal[identity[setpart[x]], setpart[v]], setpart[v]], not[equal[composite[id[setpart[u]], setpart[x], id[setpart[u]]], setpart[y]], not[equal[composite[id[setpart[v]], setpart[y], id[setpart[v]]], setpart[z]], not[equal[identity[setpart[x]], setpart[u]], setpart[u]], not[equal[identity[setpart[y]], setpart[v]], setpart[v]]] = True}
\]

\[
\text{/. Equal} \rightarrow \text{SetDelayed}
\]

The main problem encountered in eliminating the variables is that the complexity of the formula for **FNRS** causes various rewrite rules to produce complicated expressions. To shield **FNRS** from unwanted rewriting in the process of eliminating variables, yet another variable \( t \) and an equality literal will be introduced. Accordingly, the above simplified statement of the quasi-associative law will first be rewritten as follows:
Theorem. The composite of the functions \( \text{FNRS} \circ \text{LEFT}[u] \) and \( \text{FNRS} \circ \text{LEFT}[v] \) is contained in the left hand factor: \( \text{FNRS} \circ \text{LEFT}[u] \).

Lemma. (Elimination of three variables.)

This rather messy statement can be cleaned up as follows:

The goal is to eliminate all six variables to derive a variable-free statement of transitivity.

**eliminating x, y and z**

Lemma. (Elimination of three variables.)

This rather messy statement can be cleaned up as follows:
The following observation implies a corollary.

\[ \text{In[15]} := \text{subclass}[\text{LEFT}[u], \text{inverse}[\text{SECOND}]] \]
\[ \text{Out[15]} = \text{True} \]

Corollary. The composite of the functions \( \text{FNRS} \circ \text{LEFT}[u] \) and \( \text{FNRS} \circ \text{LEFT}[v] \) is contained in the relation \( \text{FNRS} \circ \text{inverse}[\text{SECOND}] \).

\[ \text{In[16]} := \text{Map}[\text{not}, \text{SubstTest}[\text{and}, \text{implies}[\text{p1}, \text{p2}], \text{implies}[\text{p2}, \text{p3}], \text{not}[\text{implies}[\text{p1}, \text{p3}]], \{\text{p1} \rightarrow \text{equal}[t, \text{FNRS}], \text{p2} \rightarrow \text{subclass}[\text{composite}[t, \text{LEFT}[\text{setpart}[u]]], \text{t}, \text{LEFT}[\text{setpart}[v]]], \text{composite}[t, \text{LEFT}[\text{setpart}[u]]], \text{p3} \rightarrow \text{subclass}[\text{composite}[t, \text{LEFT}[\text{setpart}[u]]], \text{t}, \text{LEFT}[\text{setpart}[v]]], \text{composite}[t, \text{inverse}[\text{SECOND}][]])] \] // Reverse
\[ \text{Out[16]} = \text{or}[\text{not}[\text{equal}[t, \text{composite}[\text{CAP}, \text{cross}[\text{composite}[\text{CART}, \text{DUP}], \text{id}], \text{id}[\text{inverse}[\text{fix}[\text{composite}[\text{inverse}[\text{SECOND}], \text{IMG}]]]]]]], \text{subclass}[\text{composite}[t, \text{LEFT}[\text{setpart}[u]]], \text{t}, \text{LEFT}[\text{setpart}[v]]], \text{composite}[t, \text{inverse}[\text{SECOND}][]])] = \text{True} \]

\[ \text{In[17]} := (\% / . \{t \rightarrow t_-, u \rightarrow u_-, v \rightarrow v_-\}) \text{/. SetDelayed} \]

After some experimentation it was discovered that it is much easier to eliminate the variables \( u \) and \( v \) if one works with right multiplications for the flipped action \( \text{flip[FNRS]} \) instead of left multiplications of \( \text{FNRS} \). Some additional steps are needed to do this.

Corollary.

\[ \text{In[18]} := \text{SubstTest}[\text{implies}, \text{equal}[s, \text{FNRS}], \text{subclass}[\text{composite}[s, \text{LEFT}[\text{setpart}[u]]], \text{s}, \text{LEFT}[\text{setpart}[v]]], \text{composite}[s, \text{inverse}[\text{SECOND}][]])] \] // Reverse
\[ \text{Out[18]} = \text{or}[\text{not}[\text{equal}[\text{composite}[t, \text{SWAP}], \text{composite}[\text{CAP}, \text{cross}[\text{composite}[\text{CART}, \text{DUP}], \text{id}], \text{id}[\text{inverse}[\text{fix}[\text{composite}[\text{inverse}[\text{SECOND}], \text{IMG}]]]]]]], \text{subclass}[\text{composite}[t, \text{RIGHT}[\text{setpart}[u]]], \text{t}, \text{RIGHT}[\text{setpart}[v]]], \text{composite}[t, \text{inverse}[\text{FIRST}][]])] = \text{True} \]

\[ \text{In[19]} := (\% / . \{t \rightarrow t_-, u \rightarrow u_-, v \rightarrow v_-\}) \text{/. SetDelayed} \]

Corollary.

\[ \text{In[20]} := (\text{Map}[\text{not}, \text{SubstTest}[\text{and}, \text{implies}[\text{and}[\text{p1}, \text{p2}], \text{p3}], \text{implies}[\text{and}[\text{p2}, \text{p3}], \text{p4}], \text{not}[\text{implies}[\text{and}[\text{p1}, \text{p2}], \text{p4}]], \{\text{p1} \rightarrow \text{equal}[s, \text{flip[FNRS]}], \text{p2} \rightarrow \text{equal}[s, \text{t}], \text{p3} \rightarrow \text{equal}[\text{flip}[t], \text{FNRS}], \text{p4} \rightarrow \text{subclass}[\text{composite}[t, \text{RIGHT}[\text{setpart}[u]]], \text{t}, \text{RIGHT}[\text{setpart}[v]]], \text{composite}[t, \text{inverse}[\text{FIRST}][]])] \] // Reverse) / . \text{s} \rightarrow \text{flip[FNRS]}
\[ \text{Out[20]} = \text{or}[\text{not}[\text{equal}[t, \text{composite}[\text{CAP}, \text{cross}[\text{id}, \text{composite}[\text{CART}, \text{DUP}]], \text{id}[\text{fix}[\text{composite}[\text{inverse}[\text{SECOND}], \text{IMG}]]]]]]], \text{subclass}[\text{composite}[t, \text{RIGHT}[\text{setpart}[u]]], \text{t}, \text{RIGHT}[\text{setpart}[v]]], \text{composite}[t, \text{inverse}[\text{FIRST}][]])] = \text{True} \]
All the remaining variables can now be eliminated all at once to obtain a transitive law.

Theorem. Two-sided restriction to final segments is transitive.

Restatement.

Comment. The relation \( \text{FNRS} \circ \text{inverse[SECOND]} \) is also antisymmetric.

The union of the domain and range of \( \text{FNRS} \circ \text{inverse[SECOND]} \) is the universal class \( V \).

The following statement also follows automatically now.