a partial order for bands

Johan G. F. Belinfante
2009 February 17

In[1]:= **SetDirectory["l:"]; << goedel.09feb17a; << tools.m**

```
:Package Title: goedel.09feb17a         2009 February 17 at 11:20 a.m.

It is now:  2009 Feb 17 at 15:33

Loading Simplification Rules

TOOLS.M                                       Revised 2009 February 17

weightlimit = 40
```

summary

For a band the left and right divisibility relations coincide, respectively, with their left and right neutrality relations. These relations are both reflexive and transitive. Their intersection is a partial order relation.

comparison with the hand proof

That divisibility in a band implies neutrality can easily be seen by hand. Suppose **x** is a band, and **u, v, w ∈ range[x]**. One needs to show that if **u · v = w**, then **u · w = w**. Indeed, using the associative and idempotent laws, one has:

$$\mathbf{u \cdot w = u \cdot (u \cdot v) = (u \cdot u) \cdot v = u \cdot v = w}$$

The associative law is an automatic consequence of rewrite rules when one uses the **band** wrapper. The entire computation done above becomes completely automatic:

In[2]:= **implies[and[equal[APPLY[band[x], PAIR[u, v]], w], equal[APPLY[band[x], PAIR[u, u]], u]], equal[APPLY[band[x], PAIR[u, w]], w]]**

Out[2]= True

The main task is not this computation, but rather the elimination of the variables **u**, **v** and **w**.

left and right neutrality relations

An element **u** is **left-neutral** for an element **v** for a binary operation **x** if **u · v = v**. The **left-neutrality relation** for a binary operation **x** is the class of **pair[u, v]** such that **u · v = v**. This class is:

```
In[3]:= class[pair[u, v], member[pair[pair[u, v], v], x]]

Out[3]= fix[composite[inverse[SECOND], x]]
```

The following abbreviation will be used on occasion:

```
In[4]:= leftneutrality[x_] := fix[composite[inverse[SECOND], x]]
```

The **rught-neutrality** relation for **x** is the left-neutrality relation for **flip[x]**.

```
In[5]:= leftneutrality[flip[x]]

Out[5]= inverse[fix[composite[inverse[FIRST], x]]]
```

This relation will be abbreviated as well.

```
In[6]:= rightneutrality[x_] := inverse[fix[composite[inverse[FIRST], x]]]
```

For a commutative binary operation, these relations coincide. This observation may sometimes be useful to fill in missing rewrite rules. For example:

```
In[7]:= SubstTest[rightneutrality, flip[t], t -> INTMUL]

Out[7]= fix[composite[inverse[SECOND], INTMUL]] ==
         union[cart[Z, set[id[omega]]], cart[set[composite[id[omega], SUCC]], Z]]

In[8]:= fix[composite[inverse[SECOND], INTMUL]] :=
         union[cart[Z, set[id[omega]]], cart[set[composite[id[omega], SUCC]], Z]]
```

---

divisibility

An element **u** is a **left-divisor** of an element **v** for a binary operation if there exists some element **w** such that **u · w = v**. The **left-divisibility** relation for **x** is the class of **pair[u, v]** such that **u** is a left-divisor of **v**. The left-divisibility relation for **x** is

```
In[9]:= class[pair[u, v], exists[w, member[pair[pair[u, w], v], x]]]

Out[9]= composite[x, inverse[FIRST]]
```

The following abbreviation will be used for this relation:

```
In[10]:= leftdivisibility[x_] := composite[x, inverse[FIRST]]
```

Again, the right-divisibility relation for **x** is the left-divisibility relation for **flip[x]**.

```
In[11]:= leftdivisibility[flip[x]]

Out[11]= composite[x, inverse[SECOND]]
```

An abbreviation for this will also be introduced:

```
In[12]:= rightdivisibility[x_] := composite[x, inverse[SECOND]]
```

In general **left-neutrality** implies **left-divisibility**.

```
In[13]:= subclass[leftneutrality[x], leftdivisibility[x]]

Out[13]= True

In[14]:= subclass[rightneutrality[x], rightdivisibility[x]]

Out[14]= True
```

More explicitly, this observation says:

```
In[15]:= subclass[fix[composite[inverse[SECOND], x]], composite[x, inverse[FIRST]]]

Out[15]= True

In[16]:= subclass[fix[composite[inverse[FIRST], x]], composite[SECOND, inverse[x]]]

Out[16]= True
```

If **x** is associative, both divisibility relations are transitive. In particular, this is the case for bands.

Theorem.   For a band, the left-divisibility relation is transitive.

```
In[17]:= SubstTest[TRANSITIVE, leftdivisibility[assoc[t]], t → band[x]] // Reverse

Out[17]= TRANSITIVE[composite[band[x], inverse[FIRST]]] == True

In[18]:= TRANSITIVE[composite[band[x_], inverse[FIRST]]] := True
```

Theorem. For a band, the right-divisibility relation is transitive.

```
In[19]:= SubstTest[TRANSITIVE, rightdivisibility[assoc[t]], t → band[x]] // Reverse

Out[19]= TRANSITIVE[composite[band[x], inverse[SECOND]]] == True

In[20]:= TRANSITIVE[composite[band[x_], inverse[SECOND]]] := True
```

---

eliminating APPLY

The elimination of variables is impeded by the occurrence of **APPLY**, but rewrite rules are helped by using **APPLY**. One needs to be able to introduce and then eliminate this constructor as needed.

Theorem. (Introducing **APPLY**.)

```
In[21]:= SubstTest[implies, member[pair[t, w], funpart[z]],
           equal[w, APPLY[funpart[z], t]], {t → pair[u, v], z → band[x]}] // Reverse

Out[21]= or[equal[w, APPLY[band[x], PAIR[u, v]]],
           not[member[pair[pair[u, v], w], band[x]]]] == True

In[22]:= or[equal[w_, APPLY[band[x_], PAIR[u_, v_]]],
           not[member[pair[pair[u_, v_], w_], band[x_]]]] := True
```

idempotence

One can formulate idempotence for elements of a band without using **APPLY**.

Temporary Lemma. (Simplification rule.)

```
In[23]:= SubstTest[member, pair[pair[u, u], u], composite[Id, t], t → band[x]]

Out[23]= and[member[u, V], member[pair[pair[u, u], u], band[x]]] ==
          member[pair[pair[u, u], u], band[x]]

In[24]:= and[member[u_, V], member[pair[pair[u_, u_], u_], band[x_]]] :=
          member[pair[pair[u, u], u], band[x]]
```

Theorem. A rewrite rule for idempotence.

```
In[25]:= SubstTest[member, u, fix[composite[semigp[t], DUP]], t → band[x]]

Out[25]= member[pair[pair[u, u], u], band[x]] == member[u, range[band[x]]]

In[26]:= member[pair[pair[u_, u_], u_], band[x_]] := member[u, range[band[x]]]
```

main theorem

Lemma.

```
In[27]:= (or[member[pair[pair[u, w], w], funpart[t]], not[member[w, V]],
             not[equal[w, APPLY[funpart[t], PAIR[u, w]]]]] // AssertTest) /. t → band[x]

Out[27]= or[member[pair[pair[u, w], w], band[x]],
           not[equal[w, APPLY[band[x], PAIR[u, w]]]], not[member[w, V]]] == True

In[28]:= or[member[pair[pair[u_, w_], w_], band[x_]],
           not[equal[w_, APPLY[band[x_], PAIR[u_, w_]]]], not[member[w_, V]]] := True
```

Theorem. A restatement of the fact that left-divisibility implies left-neutrality sans the **APPLY** constructor. (Extensive timing experiments have brought this derivation down to about 3 seconds.)

```
In[29]:= Map[not, SubstTest[and, implies[and[p3, p4], p5],
            implies[and[p0, p5], p6], not[implies[and[p0, p1, p2], p6]],
            {p0 → member[w, V], p1 -> member[pair[pair[u, u], u], band[x]],
             p2 -> member[pair[pair[u, v], w], band[x]],
             p3 -> equal[APPLY[band[x], PAIR[u, u]], u], p4 ->
              equal[APPLY[band[x], PAIR[u, v]], w], p5 -> equal[APPLY[band[x], PAIR[u, w]], w],
             p6 → member[pair[pair[u, w], w], band[x]]}]] // Reverse

Out[29]= or[member[pair[pair[u, w], w], band[x]], not[member[u, range[band[x]]]],
           not[member[w, V]], not[member[pair[pair[u, v], w], band[x]]]] == True

In[30]:= or[member[pair[pair[u_, w_], w_], band[x]], not[member[u_, range[band[x_]]]],
           not[member[w_, V]], not[member[pair[pair[u_, v_], w_], band[x_]]]] := True
```

---

removal of variables

The removal of variables is rather tricky because there are three of them.

Lemma. A needed simplification rule.

```
In[31]:= Assoc[band[x], id[domain[band[x]]], id[cart[range[band[x]], V]]] // Reverse

Out[31]= composite[band[x], id[cart[range[band[x]], V]]] == band[x]

In[32]:= composite[band[x_], id[cart[range[band[x_]], V]]] := band[x]
```

Theorem. (Removal of all three variables. A key trick here is the interchange of the variables **v** and **w** in the class-formation construction.)

```
In[33]:= Map[empty[composite[complement[#], id[cart[V, V]]]] &, SubstTest[class,
           pair[pair[u, w], v], or[member[pair[pair[u, w], w], t], not[member[u, range[t]]],
            not[member[w, V]], not[member[pair[pair[u, v], w], t]]], t → band[x]]]

Out[33]= subclass[composite[band[x], inverse[FIRST]],
           fix[composite[inverse[SECOND], band[x]]]] == True

In[34]:= (% /. x → x_) /. Equal → SetDelayed
```

Theorem. The left-divisibility relation coincides with the left-neutrality relation for a band.

```
In[35]:= SubstTest[and, subclass[u, v], subclass[v, u], {u -> composite[band[x], inverse[FIRST]],
           v -> fix[composite[inverse[SECOND], band[x]]]}]

Out[35]= equal[composite[band[x], inverse[FIRST]],
           fix[composite[inverse[SECOND], band[x]]]] == True

In[36]:= fix[composite[inverse[SECOND], band[x_]]] := composite[band[x], inverse[FIRST]]
```

Theorem. Application of duality.

```
In[37]:= Map[inverse, SubstTest[fix, composite[inverse[SECOND], band[t]], t → flip[band[x]]]] //
          Reverse

Out[37]= fix[composite[inverse[FIRST], band[x]]] == composite[SECOND, inverse[band[x]]]

In[38]:= fix[composite[inverse[FIRST], band[x_]]] := composite[SECOND, inverse[band[x]]]
```

reflexivity of left- and right-neutrality

Lemma.

```
In[39]:= SubstTest[implies, subclass[u, v],
            subclass[image[t, u], image[t, v]], {t → composite[FIRST, id[inverse[band[x]]]],
             u → DUP, v → inverse[FIRST]}] // Reverse // InvertFix

Out[39]= subclass[range[band[x]], fix[composite[band[x], inverse[FIRST]]]] == True

In[40]:= (% /. x → x_) /. Equal → SetDelayed
```

Theorem. An equation for a fixed-point class.

```
In[41]:= SubstTest[and, subclass[u, v], subclass[v, u],
          {u → fix[composite[band[x], inverse[FIRST]]], v → range[band[x]]}]

Out[41]= equal[fix[composite[band[x], inverse[FIRST]]], range[band[x]]] == True

In[42]:= fix[composite[band[x_], inverse[FIRST]]] := range[band[x]]
```

Corollary. Dual result.

```
In[44]:= SubstTest[fix, composite[band[t], inverse[FIRST]], t → flip[band[x]]] // Reverse

Out[44]= fix[composite[band[x], inverse[SECOND]]] == range[band[x]]

In[45]:= fix[composite[band[x_], inverse[SECOND]]] := range[band[x]]
```

Lemma. The left-divisibility relation of a band is contained in its domain.

```
In[46]:= SubstTest[subclass, t, cart[domain[t], range[t]],
           t -> composite[band[x], inverse[FIRST]]] // Reverse

Out[46]= subclass[composite[band[x], inverse[FIRST]], domain[band[x]]] == True

In[47]:= subclass[composite[band[x_], inverse[FIRST]], domain[band[x_]]] := True
```

Theorem. Left divisibility is reflexive.

```
In[48]:= SubstTest[subclass, t, cartsq[fix[t]], t → leftneutrality[band[x]]]

Out[48]= REFLEXIVE[composite[band[x], inverse[FIRST]]] == True

In[49]:= REFLEXIVE[composite[band[x_], inverse[FIRST]]] := True
```

Theorem. Dual result.

```
In[50]:= SubstTest[REFLEXIVE, composite[band[t], inverse[FIRST]], t → flip[band[x]]] // Reverse

Out[50]= REFLEXIVE[composite[band[x], inverse[SECOND]]] == True

In[51]:= REFLEXIVE[composite[band[x_], inverse[SECOND]]] := True
```

a partial order

The two-sided divisibility relation is the intersection of the left-divisibility and right-divisibility relations. Equivalently, this can be regarded as two-sided neutrality.

Lemma. Two-sided divisibility in a band is transitive.

```
In[52]:= SubstTest[TRANSITIVE, intersection[trv[u], trv[v]],
           {u -> composite[band[x], inverse[FIRST]],
            v -> composite[band[x], inverse[SECOND]]}] // Reverse

Out[52]= TRANSITIVE[intersection[composite[band[x], inverse[FIRST]],
            composite[band[x], inverse[SECOND]]]] == True

In[53]:= TRANSITIVE[intersection[composite[band[x_], inverse[FIRST]],
            composite[band[x_], inverse[SECOND]]]] := True
```

Lemma. Two-sided divisibility in a band is reflexive.

```
In[54]:= SubstTest[subclass, t, cartsq[fix[t]], t -> intersection[
            composite[band[x], inverse[FIRST]], composite[band[x], inverse[SECOND]]]]

Out[54]= REFLEXIVE[intersection[composite[band[x], inverse[FIRST]],
            composite[band[x], inverse[SECOND]]]] == True

In[55]:= REFLEXIVE[intersection[composite[band[x_], inverse[FIRST]],
            composite[band[x_], inverse[SECOND]]]] := True
```

Lemma.

```
In[56]:= SubstTest[fix, intersection[u, v],
          {u -> composite[inverse[FIRST], band[x]], v -> composite[inverse[SECOND], band[x]]}]

Out[56]= intersection[composite[SECOND, inverse[band[x]]],
          composite[band[x], inverse[FIRST]]] == id[range[band[x]]]

In[57]:= intersection[composite[SECOND, inverse[band[x_]]],
          composite[band[x_], inverse[FIRST]]] := id[range[band[x]]]
```

Theorem. For a band two-sided divisibility is a partial order.

```
In[58]:= SubstTest[and, REFLEXIVE[t], ANTISYMMETRIC[t], TRANSITIVE[t], t -> intersection[
            composite[band[x], inverse[FIRST]], composite[band[x], inverse[SECOND]]]]

Out[58]= PARTIALORDER[intersection[composite[band[x], inverse[FIRST]],
            composite[band[x], inverse[SECOND]]]] == True

In[59]:= PARTIALORDER[intersection[composite[band[x_], inverse[FIRST]],
            composite[band[x_], inverse[SECOND]]]] := True
```

Restatement. If one defines $\mathbf{u} \le \mathbf{v}$ to mean that $\mathbf{u} = \mathbf{u} \cdot \mathbf{v} = \mathbf{v} \cdot \mathbf{u}$ for elements $\mathbf{u}, \mathbf{v} \in$ **range[x]** of a band **x**, then this relation is a partial order.