constant unary and binary operations

Johan G. F. Belinfante
2006 June 11

```
In[1]:= SetDirectory["l:"]; << goedel82.10b; << tools.m

        :Package Title: goedel82.10b         2006 June 10 at 10:20 p.m.

        It is now:  2006 Jun 11 at 18:22

        Loading Simplification Rules

        TOOLS.M                       Revised 2006 June 6

        weightlimit = 40
```

summary

Formulas for non-empty constant unary and binary operations are derived.

lemmas

```
In[2]:= AssInt[range[CART], range[CART], FUNS]

Out[2]= intersection[CONST, range[CART]] == CONST

In[3]:= intersection[CONST, range[CART]] := CONST

In[4]:= AssInt[range[CART], FUNS, UNOPS]

Out[4]= intersection[UNOPS, range[CART]] == intersection[CONST, UNOPS]

In[5]:= intersection[UNOPS, range[CART]] := intersection[CONST, UNOPS]

In[6]:= AssInt[range[CART], FUNS, BINOPS]

Out[6]= intersection[BINOPS, range[CART]] == intersection[BINOPS, CONST]

In[7]:= intersection[BINOPS, range[CART]] := intersection[BINOPS, CONST]
```

constant unary operations

Theorem.

```
In[8]:= image[inverse[CART], UNOPS] // RelnNormality

Out[8]= image[inverse[CART], UNOPS] ==
         union[cart[V, set[0]], cart[set[0], V], composite[inverse[E], IMAGE[SINGLETON]]]

In[9]:= image[inverse[CART], UNOPS] :=
         union[cart[V, set[0]], cart[set[0], V], composite[inverse[E], IMAGE[SINGLETON]]]
```

A formula for nonempty constant unary operations can be obtained from this.

```
In[10]:= Map[equal[image[CART, composite[inverse[E], IMAGE[SINGLETON]]], dif[#, set[0]]] &,
          ImageComp[CART, inverse[CART], UNOPS]]

Out[10]= equal[image[CART, composite[inverse[E], IMAGE[SINGLETON]]],
           intersection[CONST, UNOPS, complement[set[0]]]] == True

In[11]:= image[CART, composite[inverse[E], IMAGE[SINGLETON]]] :=
          intersection[UNOPS, complement[set[0]], range[CART]]
```

Corollary.

```
In[12]:= SubstTest[U, image[CART, x], x -> composite[inverse[E], IMAGE[SINGLETON]]]

Out[12]= U[intersection[CONST, UNOPS]] == cart[V, V]

In[13]:= U[intersection[CONST, UNOPS]] := cart[V, V]
```

constant binary operations

Theorem.

```
In[14]:= image[inverse[CART], BINOPS] // RelnNormality

Out[14]= image[inverse[CART], BINOPS] == union[cart[V, set[0]], cart[set[0], V],
           composite[inverse[E], IMAGE[SINGLETON], inverse[DUP], inverse[CART]]]

In[15]:= image[inverse[CART], BINOPS] := union[cart[V, set[0]], cart[set[0], V],
           composite[inverse[E], IMAGE[SINGLETON], inverse[DUP], inverse[CART]]]
```

A formula for nonempty constant binary operations is obtained the same way as for unary operations.

```
In[16]:= Map[equal[
            image[CART, composite[inverse[E], IMAGE[SINGLETON], inverse[DUP], inverse[CART]]],
            dif[#, set[0]]] &, ImageComp[CART, inverse[CART], BINOPS]]

Out[16]= equal[
           image[CART, composite[inverse[E], IMAGE[SINGLETON], inverse[DUP], inverse[CART]]],
           intersection[BINOPS, CONST, complement[set[0]]]] == True

In[17]:= image[CART, composite[inverse[E], IMAGE[SINGLETON], inverse[DUP], inverse[CART]]] :=
          intersection[BINOPS, CONST, complement[set[0]]]
```

Corollary.

```
In[18]:= SubstTest[U, image[CART, x],
          x -> composite[inverse[E], IMAGE[SINGLETON], inverse[DUP], inverse[CART]]]

Out[18]= U[intersection[BINOPS, CONST]] == cart[cart[V, V], V]

In[19]:= U[intersection[BINOPS, CONST]] := cart[cart[V, V], V]
```