RIF rules

Johan G. F. Belinfante  
2002 June 6

```
<< goedel52.o32; << tools.m

:Package Title: goedel52.o32 2002 June 6 at 8:20 a.m.

It is now: 2002 Jun 6 at 21:52

Loading Simplification Rules

TOOLS.M Revised 2002 June 6

weightlimit = 40
```

## Introduction

The function **RIF** can be expressed in terms of **FIRST** and **SECOND**, so in principle we could eliminate it.

```math
\text{composite[cross[SECOND, FIRST], id[composite[\text{inverse}[SECOND], FIRST]]]}
```

**RIF**

The name **RIF** stands for rotation invariant function. Indeed:

```math
\text{FUNCTION[RIF]}
\text{True}
\text{rotate[RIF]}
\text{RIF}
```

The importance of **RIF** stems from its ability to transform cartesian products into composites:

```
\text{image[RIF, cart}[x, y][]}
\text{composite[\text{inverse}[y], \text{inverse}[x]]}
```

In fact we first became interested in **RIF** when we found it could be used to define the **COMPOSE** function in terms of **CART** as follows.

```
\text{composite[IMAGE[SWAP], IMAGE[RIF], CART]}
\text{COMPOSE}
```

Recently we found it useful to express the commutativity of powers in a succinct way:
composite[RIF, SWAP, cross[power[x], power[x]]]
composite[RIF, cross[power[x], power[x]]]

■ Rule 1

composite[SECOND, id[cart[x, y]], inverse[RIF]] // DoubleInverse

composite[SECOND, id[cart[x, y]], inverse[RIF]] ==
composite[SWAP, id[inverse[y]], cross[inverse[x], Id]]

composite[SECOND, id[cart[x_, y_]], inverse[RIF]] :=
composite[SWAP, id[inverse[y]], cross[inverse[x], Id]]

■ Double Inverse formula

composite[cross[SECOND, FIRST], inverse[RIF]] // DoubleInverse

composite[cross[SECOND, FIRST], inverse[RIF]] == id[cart[V, V]]

■ Rule 2

composite[RIF, id[cart[x, cart[V, V]]]] // TripleRotate

composite[RIF, id[cart[x, cart[V, V]]]] == composite[RIF, id[cart[x, V]]]

composite[RIF, id[cart[x_, cart[V, V]]]] := composite[RIF, id[cart[x, V]]]

■ RIF–RIF formula

composite[RIF, RIF, cross[cross[Id, inverse[SECOND]], cross[inverse[FIRST], Id]]] // TripleRotate

composite[RIF, RIF, cross[cross[Id, inverse[SECOND]], cross[inverse[FIRST], Id]]] == RIF

composite[RIF, RIF, cross[cross[x, inverse[SECOND]], cross[inverse[FIRST], y]]] // TripleRotate

composite[RIF, RIF, cross[cross[x, inverse[SECOND]], cross[inverse[FIRST], y]]] ==
composite[RIF, cross[Id, cross[Id, composite[inverse[x], y]]]]

■ Curious Rule

composite[RIF, cross[cross[Id, x], cross[y, Id]], inverse[RIF]] // VSTriNormality

composite[RIF, cross[cross[Id, x], cross[y, Id]], inverse[RIF]] == cross[x, y]
composite[RIF, cross[cross[u, x], cross[y, v]], inverse[RIF]] // VSTriNormality

composite[RIF, cross[cross[u, x], cross[y, v]], inverse[RIF]] ==
cross[x, composite[y, id[image[V, fix[composite[inverse[u], v]]]]]]