

# Octonions

Angeliki Koutsoukou-Argyraiki

September 13, 2023

## Abstract

We develop the basic theory of Octonions, including various identities and properties of the octonions and of the octonionic product, a description of 7D isometries and representations of orthogonal transformations. To this end we first develop the theory of the vector cross product in 7 dimensions. The development of the theory of Octonions is inspired by that of the theory of Quaternions by Lawrence Paulson. However, we do not work within the type class *real\_algebra\_1* because the octonionic product is not associative.

# Contents

<b>1</b>	<b>Vector Cross Product in 7 Dimensions</b>	<b>3</b>
1.1	Elementary auxiliary lemmas. . . . .	3
1.2	The definition of the 7D cross product and related lemmas . .	4
1.3	Continuity . . . . .	8
<b>2</b>	<b>Theory of Octonions</b>	<b>8</b>
2.1	Basic definitions . . . . .	8
2.2	Addition and Subtraction: An Abelian Group . . . . .	9
2.3	A Normed Vector Space . . . . .	10
2.4	The Octonionic product and related properties and lemmas .	13
2.5	Embedding of the Reals into the Octonions . . . . .	16
2.6	"Expansion" into the traditional notation . . . . .	20
2.7	Conjugate of an octonion and related properties. . . . .	20
2.8	Linearity and continuity of the components. . . . .	24
	2.8.1 Octonionic-specific theorems about sums. . . . .	26
	2.8.2 Bound results for real and imaginary components of limits. . . . .	26
2.9	Octonions for describing 7D isometries . . . . .	28
	2.9.1 The <i>HI</i> m operator . . . . .	28
	2.9.2 The <i>Hv</i> operator . . . . .	29
	2.9.3 Related basic identities . . . . .	30
2.10	Representing orthogonal transformations as conjugation or congruence with an octonion. . . . .	30

**Acknowledgements.** A.K.-A. was supported by the ERC Advanced Grant ALEXANDRIA (Project 742178) funded by the European Research Council and led by Professor Lawrence Paulson at the University of Cambridge, UK. Many thanks to Manuel Eberl, Wenda Li and Lawrence Paulson for their suggestions and improvements.

## 1 Vector Cross Product in 7 Dimensions

**theory** *Cross-Product-7*  
**imports** *HOL-Analysis.Multivariate-Analysis*  
**begin**

### 1.1 Elementary auxiliary lemmas.

**lemma** *exhaust-7*:  
**fixes**  $x :: 7$   
**shows**  $x = 1 \vee x = 2 \vee x = 3 \vee x = 4 \vee x = 5 \vee x = 6 \vee x = 7$   
*<proof>*

**lemma** *forall-7*:  $(\forall i::7. P i) \longleftrightarrow P 1 \wedge P 2 \wedge P 3 \wedge P 4 \wedge P 5 \wedge P 6 \wedge P 7$   
*<proof>*

**lemma** *vector-7* [*simp*]:  
 $(\text{vector } [x1, x2, x3, x4, x5, x6, x7] :: ('a::zero)^7)\$1 = x1$   
 $(\text{vector } [x1, x2, x3, x4, x5, x6, x7] :: ('a::zero)^7)\$2 = x2$   
 $(\text{vector } [x1, x2, x3, x4, x5, x6, x7] :: ('a::zero)^7)\$3 = x3$   
 $(\text{vector } [x1, x2, x3, x4, x5, x6, x7] :: ('a::zero)^7)\$4 = x4$   
 $(\text{vector } [x1, x2, x3, x4, x5, x6, x7] :: ('a::zero)^7)\$5 = x5$   
 $(\text{vector } [x1, x2, x3, x4, x5, x6, x7] :: ('a::zero)^7)\$6 = x6$   
 $(\text{vector } [x1, x2, x3, x4, x5, x6, x7] :: ('a::zero)^7)\$7 = x7$   
*<proof>*

**lemma** *forall-vector-7*:  
 $(\forall v::'a::zero^7. P v) \longleftrightarrow (\forall x1\ x2\ x3\ x4\ x5\ x6\ x7. P(\text{vector}[x1, x2, x3, x4, x5, x6, x7]))$   
*<proof>*

**lemma** *UNIV-7*:  $UNIV = \{1::7, 2::7, 3::7, 4::7, 5::7, 6::7, 7::7\}$   
*<proof>*

**lemma** *sum-7*:  $\text{sum } f \text{ (UNIV::7 set)} = f 1 + f 2 + f 3 + f 4 + f 5 + f 6 + f 7$   
*<proof>*

**lemma** *not-equal-vector7* :  
**fixes**  $x::\text{real}^7$  **and**  $y::\text{real}^7$   
**assumes**  $x = \text{vector}[x1, x2, x3, x4, x5, x6, x7]$  **and**  $y = \text{vector}[y1, y2, y3, y4, y5, y6, y7]$   
**and**  $x\$1 \neq y\$1 \vee x\$2 \neq y\$2 \vee x\$3 \neq y\$3 \vee x\$4 \neq y\$4 \vee x\$5 \neq y\$5 \vee x\$6 \neq y\$6 \vee x\$7 \neq y\$7$

**shows**  $x \neq y$  *<proof>*

**lemma** *equal-vector7*:

**fixes**  $x::\text{real}^7$  **and**  $y::\text{real}^7$

**assumes**  $x = \text{vector}[x1,x2,x3,x4,x5,x6,x7]$  **and**  $y = \text{vector}[y1,y2,y3,y4,y5,y6,y7]$   
**and**  $x = y$

**shows**  $x\$1 = y\$1 \wedge x\$2 = y\$2 \wedge x\$3 = y\$3 \wedge x\$4 = y\$4 \wedge x\$5 = y\$5 \wedge x\$6 = y\$6 \wedge x\$7 = y\$7$

*<proof>*

**lemma** *numeral-4-eq-4*:  $4 = \text{Suc}(\text{Suc}(\text{Suc}(\text{Suc} 0)))$

*<proof>*

**lemma** *numeral-5-eq-5*:  $5 = \text{Suc}(\text{Suc}(\text{Suc}(\text{Suc}(\text{Suc} 0))))$

*<proof>*

**lemma** *numeral-6-eq-6*:  $6 = \text{Suc}(\text{Suc}(\text{Suc}(\text{Suc}(\text{Suc}(\text{Suc} 0))))))$

*<proof>*

**lemma** *numeral-7-eq-7*:  $7 = \text{Suc}(\text{Suc}(\text{Suc}(\text{Suc}(\text{Suc}(\text{Suc}(\text{Suc} 0))))))$

*<proof>*

## 1.2 The definition of the 7D cross product and related lemmas

**context includes** *no-Set-Product-syntax*

**begin**

Note: in total there exist 480 equivalent multiplication tables for the definition, the following is based on the one most widely used:

**definition** *cross7* ::  $[\text{real}^7, \text{real}^7] \Rightarrow \text{real}^7$  (**infixr**  $\times_7$  80)

**where**  $a \times_7 b \equiv$

$\text{vector}[a\$2 * b\$4 - a\$4 * b\$2 + a\$3 * b\$7 - a\$7 * b\$3 + a\$5 * b\$6 - a\$6 * b\$5,$   
 $a\$3 * b\$5 - a\$5 * b\$3 + a\$4 * b\$1 - a\$1 * b\$4 + a\$6 * b\$7 - a\$7 * b\$6,$   
 $a\$4 * b\$6 - a\$6 * b\$4 + a\$5 * b\$2 - a\$2 * b\$5 + a\$7 * b\$1 - a\$1 * b\$7,$   
 $a\$5 * b\$7 - a\$7 * b\$5 + a\$6 * b\$3 - a\$3 * b\$6 + a\$1 * b\$2 - a\$2 * b\$1,$   
 $a\$6 * b\$1 - a\$1 * b\$6 + a\$7 * b\$4 - a\$4 * b\$7 + a\$2 * b\$3 - a\$3 * b\$2,$   
 $a\$7 * b\$2 - a\$2 * b\$7 + a\$1 * b\$5 - a\$5 * b\$1 + a\$3 * b\$4 - a\$4 * b\$3,$   
 $a\$1 * b\$3 - a\$3 * b\$1 + a\$2 * b\$6 - a\$6 * b\$2 + a\$4 * b\$5 - a\$5 * b\$4]$

**end**

**bundle** *cross7-syntax* **begin**

**notation** *cross7* (**infixr**  $\times_7$  80)

**no-notation** *Product-Type.Times* (**infixr**  $\times_7$  80)

**end**

**bundle** *no-cross7-syntax* **begin**

**no-notation** *cross7* (**infixr**  $\times_7$  80)

**notation** *Product-Type.Times* (**infixr**  $\times_7$  80)

**end**

**unbundle** *cross7-syntax*

**lemmas** *cross7-simps* = *cross7-def inner-vec-def sum-7 det-def vec-eq-iff vector-def algebra-simps*

**lemma** *dot-cross7-self*:  $x \cdot (x \times_7 y) = 0$   $x \cdot (y \times_7 x) = 0$   $(x \times_7 y) \cdot y = 0$   $(y \times_7 x) \cdot x = 0$

*<proof>*

**lemma** *orthogonal-cross7*: *orthogonal*  $(x \times_7 y)$  *x* *orthogonal*  $(x \times_7 y)$  *y*  
*orthogonal*  $y$   $(x \times_7 y)$  *orthogonal*  $(x \times_7 y)$  *x*

*<proof>*

**lemma** *cross7-zero-left* [*simp*]:  $0 \times_7 x = 0$

**and** *cross7-zero-right* [*simp*]:  $x \times_7 0 = 0$

*<proof>*

**lemma** *cross7-skew*:  $(x \times_7 y) = -(y \times_7 x)$

*<proof>*

**lemma** *cross7-refl* [*simp*]:  $x \times_7 x = 0$

*<proof>*

**lemma** *cross7-add-left*:  $(x + y) \times_7 z = (x \times_7 z) + (y \times_7 z)$

**and** *cross7-add-right*:  $x \times_7 (y + z) = (x \times_7 y) + (x \times_7 z)$

*<proof>*

**lemma** *cross7-mult-left*:  $(c *_R x) \times_7 y = c *_R (x \times_7 y)$

**and** *cross7-mult-right*:  $x \times_7 (c *_R y) = c *_R (x \times_7 y)$

*<proof>*

**lemma** *cross7-minus-left* [*simp*]:  $(-x) \times_7 y = -(x \times_7 y)$

**and** *cross7-minus-right* [*simp*]:  $x \times_7 -y = -(x \times_7 y)$

*<proof>*

**lemma** *left-diff-distrib*:  $(x - y) \times_7 z = x \times_7 z - y \times_7 z$

**and** *right-diff-distrib*:  $x \times_7 (y - z) = x \times_7 y - x \times_7 z$

*<proof>*

**hide-fact** (**open**) *left-diff-distrib right-diff-distrib*

**lemma** *cross7-triple1*:  $(x \times_7 y) \cdot z = (y \times_7 z) \cdot x$

**and** *cross7-triple2*:  $(x \times_7 y) \cdot z = x \cdot (y \times_7 z)$   
 ⟨proof⟩

**lemma** *scalar7-triple1*:  $x \cdot (y \times_7 z) = y \cdot (z \times_7 x)$   
**and** *scalar7-triple2*:  $x \cdot (y \times_7 z) = z \cdot (x \times_7 y)$   
 ⟨proof⟩

**lemma** *cross7-components*:

$$\begin{aligned}
 (x \times_7 y)_1 &= x_2 * y_4 - x_4 * y_2 + x_3 * y_7 - x_7 * y_3 + x_5 * y_6 \\
 &- x_6 * y_5 \\
 (x \times_7 y)_2 &= x_4 * y_1 - x_1 * y_4 + x_3 * y_5 - x_5 * y_3 + x_6 * y_7 \\
 &- x_7 * y_6 \\
 (x \times_7 y)_3 &= x_5 * y_2 - x_2 * y_5 + x_4 * y_6 - x_6 * y_4 + x_7 * y_1 \\
 &- x_1 * y_7 \\
 (x \times_7 y)_4 &= x_1 * y_2 - x_2 * y_1 + x_6 * y_3 - x_3 * y_6 + x_5 * y_7 \\
 &- x_7 * y_5 \\
 (x \times_7 y)_5 &= x_6 * y_1 - x_1 * y_6 + x_2 * y_3 - x_3 * y_2 + x_7 * y_4 \\
 &- x_4 * y_7 \\
 (x \times_7 y)_6 &= x_1 * y_5 - x_5 * y_1 + x_7 * y_2 - x_2 * y_7 + x_3 * y_4 \\
 &- x_4 * y_3 \\
 (x \times_7 y)_7 &= x_1 * y_3 - x_3 * y_1 + x_4 * y_5 - x_5 * y_4 + x_2 * y_6 \\
 &- x_6 * y_2 \\
 &\langle proof \rangle
 \end{aligned}$$

Nonassociativity of the 7D cross product showed using a counterexample

**lemma** *cross7-nonassociative*:

$$\neg (\forall (c::\text{real}^7) (a::\text{real}^7) (b::\text{real}^7) . c \times_7 (a \times_7 b) = (c \times_7 a) \times_7 b)$$

⟨proof⟩

The 7D cross product does not satisfy the Jacobi Identity (shown using a counterexample)

**lemma** *cross7-not-Jacobi*:

$$\neg (\forall (c::\text{real}^7) (a::\text{real}^7) (b::\text{real}^7) . (c \times_7 a) \times_7 b + (b \times_7 c) \times_7 a + (a \times_7 b) \times_7 c = 0)$$

⟨proof⟩

The vector triple product property fulfilled for the 3D cross product does not hold for the 7D cross product. Shown below with a counterexample.

**lemma** *cross7-not-vectortriple*:

$$\neg (\forall (c::\text{real}^7) (a::\text{real}^7) (b::\text{real}^7) . (c \times_7 a) \times_7 b = (c \cdot b) *_R a - (c \cdot a) *_R b)$$

⟨proof⟩

**lemma** *axis-nth-neq [simp]*:  $i \neq j \implies \text{axis } i \text{ } x \text{ } j = 0$

⟨proof⟩

**lemma** *cross7-basis*:

$$(\text{axis } 1 \text{ } 1) \times_7 (\text{axis } 2 \text{ } 1) = \text{axis } 4 \text{ } 1 \text{ } (\text{axis } 2 \text{ } 1) \times_7 (\text{axis } 1 \text{ } 1) = -(\text{axis } 4 \text{ } 1)$$

$$\begin{aligned}
(\text{axis } 2 \ 1) \times_7 (\text{axis } 3 \ 1) &= \text{axis } 5 \ 1 & (\text{axis } 3 \ 1) \times_7 (\text{axis } 2 \ 1) &= -(\text{axis } 5 \ 1) \\
(\text{axis } 3 \ 1) \times_7 (\text{axis } 4 \ 1) &= \text{axis } 6 \ 1 & (\text{axis } 4 \ 1) \times_7 (\text{axis } 3 \ 1) &= -(\text{axis } 6 \ 1) \\
(\text{axis } 2 \ 1) \times_7 (\text{axis } 4 \ 1) &= \text{axis } 1 \ 1 & (\text{axis } 4 \ 1) \times_7 (\text{axis } 2 \ 1) &= -(\text{axis } 1 \ 1) \\
(\text{axis } 4 \ 1) \times_7 (\text{axis } 5 \ 1) &= \text{axis } 7 \ 1 & (\text{axis } 5 \ 1) \times_7 (\text{axis } 4 \ 1) &= -(\text{axis } 7 \ 1) \\
(\text{axis } 3 \ 1) \times_7 (\text{axis } 5 \ 1) &= \text{axis } 2 \ 1 & (\text{axis } 5 \ 1) \times_7 (\text{axis } 3 \ 1) &= -(\text{axis } 2 \ 1) \\
(\text{axis } 4 \ 1) \times_7 (\text{axis } 6 \ 1) &= \text{axis } 3 \ 1 & (\text{axis } 6 \ 1) \times_7 (\text{axis } 4 \ 1) &= -(\text{axis } 3 \ 1) \\
(\text{axis } 5 \ 1) \times_7 (\text{axis } 7 \ 1) &= \text{axis } 4 \ 1 & (\text{axis } 7 \ 1) \times_7 (\text{axis } 5 \ 1) &= -(\text{axis } 4 \ 1) \\
(\text{axis } 4 \ 1) \times_7 (\text{axis } 1 \ 1) &= \text{axis } 2 \ 1 & (\text{axis } 1 \ 1) \times_7 (\text{axis } 4 \ 1) &= -(\text{axis } 2 \ 1) \\
(\text{axis } 5 \ 1) \times_7 (\text{axis } 2 \ 1) &= \text{axis } 3 \ 1 & (\text{axis } 2 \ 1) \times_7 (\text{axis } 5 \ 1) &= -(\text{axis } 3 \ 1) \\
(\text{axis } 6 \ 1) \times_7 (\text{axis } 3 \ 1) &= \text{axis } 4 \ 1 & (\text{axis } 3 \ 1) \times_7 (\text{axis } 6 \ 1) &= -(\text{axis } 4 \ 1) \\
(\text{axis } 7 \ 1) \times_7 (\text{axis } 4 \ 1) &= \text{axis } 5 \ 1 & (\text{axis } 4 \ 1) \times_7 (\text{axis } 7 \ 1) &= -(\text{axis } 5 \ 1) \\
(\text{axis } 5 \ 1) \times_7 (\text{axis } 6 \ 1) &= \text{axis } 1 \ 1 & (\text{axis } 6 \ 1) \times_7 (\text{axis } 5 \ 1) &= -(\text{axis } 1 \ 1) \\
(\text{axis } 6 \ 1) \times_7 (\text{axis } 7 \ 1) &= \text{axis } 2 \ 1 & (\text{axis } 7 \ 1) \times_7 (\text{axis } 6 \ 1) &= -(\text{axis } 2 \ 1) \\
(\text{axis } 7 \ 1) \times_7 (\text{axis } 1 \ 1) &= \text{axis } 3 \ 1 & (\text{axis } 1 \ 1) \times_7 (\text{axis } 7 \ 1) &= -(\text{axis } 3 \ 1) \\
(\text{axis } 6 \ 1) \times_7 (\text{axis } 1 \ 1) &= \text{axis } 5 \ 1 & (\text{axis } 1 \ 1) \times_7 (\text{axis } 6 \ 1) &= -(\text{axis } 5 \ 1) \\
(\text{axis } 7 \ 1) \times_7 (\text{axis } 2 \ 1) &= \text{axis } 6 \ 1 & (\text{axis } 2 \ 1) \times_7 (\text{axis } 7 \ 1) &= -(\text{axis } 6 \ 1) \\
(\text{axis } 1 \ 1) \times_7 (\text{axis } 3 \ 1) &= \text{axis } 7 \ 1 & (\text{axis } 3 \ 1) \times_7 (\text{axis } 1 \ 1) &= -(\text{axis } 7 \ 1) \\
(\text{axis } 1 \ 1) \times_7 (\text{axis } 5 \ 1) &= \text{axis } 6 \ 1 & (\text{axis } 5 \ 1) \times_7 (\text{axis } 1 \ 1) &= -(\text{axis } 6 \ 1) \\
(\text{axis } 2 \ 1) \times_7 (\text{axis } 6 \ 1) &= \text{axis } 7 \ 1 & (\text{axis } 6 \ 1) \times_7 (\text{axis } 2 \ 1) &= -(\text{axis } 7 \ 1) \\
(\text{axis } 3 \ 1) \times_7 (\text{axis } 7 \ 1) &= \text{axis } 1 \ 1 & (\text{axis } 7 \ 1) \times_7 (\text{axis } 3 \ 1) &= -(\text{axis } 1 \ 1)
\end{aligned}$$

*<proof>*

**lemma** *cross7-basis-zero:*

$$\begin{aligned}
u=0 &\implies (u \times_7 \text{axis } 1 \ 1 = 0) \wedge (u \times_7 \text{axis } 2 \ 1 = 0) \wedge (u \times_7 \text{axis } 3 \ 1 = 0) \\
&\wedge (u \times_7 \text{axis } 4 \ 1 = 0) \wedge (u \times_7 \text{axis } 5 \ 1 = 0) \wedge (u \times_7 \text{axis } 6 \ 1 = 0) \\
&\wedge (u \times_7 \text{axis } 7 \ 1 = 0)
\end{aligned}$$

*<proof>*

**lemma** *cross7-basis-nonzero:*

$$\begin{aligned}
&\neg (u \times_7 \text{axis } 1 \ 1 = 0) \vee \neg (u \times_7 \text{axis } 2 \ 1 = 0) \vee \neg (u \times_7 \text{axis } 3 \ 1 = 0) \\
&\vee \neg (u \times_7 \text{axis } 4 \ 1 = 0) \vee \neg (u \times_7 \text{axis } 5 \ 1 = 0) \vee \neg (u \times_7 \text{axis } 6 \ 1 = 0) \\
&\vee \neg (u \times_7 \text{axis } 7 \ 1 = 0) \implies u \neq 0
\end{aligned}$$

*<proof>*

Pythagorean theorem/magnitude

**lemma** *norm-square-vec-eq:*  $\text{norm } x \wedge 2 = (\sum_{i \in \text{UNIV}} x \ \$ \ i \wedge 2)$

*<proof>*

**lemma** *norm-cross7-dot-magnitude:*  $(\text{norm } (x \times_7 y))^2 = (\text{norm } x)^2 * (\text{norm } y)^2 - (x \cdot y)^2$

*<proof>*

**lemma** *cross7-eq-0:*  $x \times_7 y = 0 \iff \text{collinear } \{0, x, y\}$

*<proof>*

**lemma** *cross7-eq-self:*  $x \times_7 y = x \iff x = 0 \ x \times_7 y = y \iff y = 0$

*<proof>*

**lemma** *norm-and-cross7-eq-0:*

$x \cdot y = 0 \wedge x \times_7 y = 0 \iff x = 0 \vee y = 0$  (is ?lhs = ?rhs)  
 <proof>

**lemma** *bilinear-cross7*: *bilinear* ( $\times_7$ )  
 <proof>

### 1.3 Continuity

**lemma** *continuous-cross7*:  $\llbracket \text{continuous } F f; \text{continuous } F g \rrbracket \implies \text{continuous } F (\lambda x. f x \times_7 g x)$   
 <proof>

**lemma** *continuous-on-cross*:  
**fixes**  $f :: 'a::t2\text{-space} \Rightarrow \text{real}^\gamma$   
**shows**  $\llbracket \text{continuous-on } S f; \text{continuous-on } S g \rrbracket \implies \text{continuous-on } S (\lambda x. f x \times_7 g x)$   
 <proof>

end

## 2 Theory of Octonions

**theory** *Octonions*  
**imports** *Cross-Product-7*  
**begin**

### 2.1 Basic definitions

As with the complex numbers, coinduction is convenient.

**codatatype** *octo* =  
*Octo* (*Ree*: real) (*Im1*: real) (*Im2*: real) (*Im3*: real) (*Im4*: real)  
 (*Im5*: real) (*Im6*: real) (*Im7*: real)

**lemma** *octo-eqI* [*intro?*]:  
 $\llbracket \text{Ree } x = \text{Ree } y; \text{Im1 } x = \text{Im1 } y; \text{Im2 } x = \text{Im2 } y; \text{Im3 } x = \text{Im3 } y;$   
 $\text{Im4 } x = \text{Im4 } y; \text{Im5 } x = \text{Im5 } y; \text{Im6 } x = \text{Im6 } y; \text{Im7 } x = \text{Im7 } y \rrbracket \implies x = y$   
 <proof>

**lemma** *octo-eq-iff*:  
 $x = y \iff \text{Ree } x = \text{Ree } y \wedge \text{Im1 } x = \text{Im1 } y \wedge \text{Im2 } x = \text{Im2 } y \wedge \text{Im3 } x = \text{Im3 } y \wedge$   
 $\text{Im4 } x = \text{Im4 } y \wedge \text{Im5 } x = \text{Im5 } y \wedge \text{Im6 } x = \text{Im6 } y \wedge \text{Im7 } x = \text{Im7 } y$   
 <proof>

**context**  
**begin**

**primcorec** *octo-e0* :: *octo* (*e0*)



**where**  $Ree\ e0 = 1 \mid Im1\ e0 = 0 \mid Im2\ e0 = 0 \mid Im3\ e0 = 0$   
 $\mid Im4\ e0 = 0 \mid Im5\ e0 = 0 \mid Im6\ e0 = 0 \mid Im7\ e0 = 0$

**primcorec** *octo-e1* :: *octo* (e1)  
**where**  $Ree\ e1 = 0 \mid Im1\ e1 = 1 \mid Im2\ e1 = 0 \mid Im3\ e1 = 0$   
 $\mid Im4\ e1 = 0 \mid Im5\ e1 = 0 \mid Im6\ e1 = 0 \mid Im7\ e1 = 0$

**primcorec** *octo-e2* :: *octo* (e2)  
**where**  $Ree\ e2 = 0 \mid Im1\ e2 = 0 \mid Im2\ e2 = 1 \mid Im3\ e2 = 0$   
 $\mid Im4\ e2 = 0 \mid Im5\ e2 = 0 \mid Im6\ e2 = 0 \mid Im7\ e2 = 0$

**primcorec** *octo-e3* :: *octo* (e3)  
**where**  $Ree\ e3 = 0 \mid Im1\ e3 = 0 \mid Im2\ e3 = 0 \mid Im3\ e3 = 1$   
 $\mid Im4\ e3 = 0 \mid Im5\ e3 = 0 \mid Im6\ e3 = 0 \mid Im7\ e3 = 0$

**primcorec** *octo-e4* :: *octo* (e4)  
**where**  $Ree\ e4 = 0 \mid Im1\ e4 = 0 \mid Im2\ e4 = 0 \mid Im3\ e4 = 0$   
 $\mid Im4\ e4 = 1 \mid Im5\ e4 = 0 \mid Im6\ e4 = 0 \mid Im7\ e4 = 0$

**primcorec** *octo-e5* :: *octo* (e5)  
**where**  $Ree\ e5 = 0 \mid Im1\ e5 = 0 \mid Im2\ e5 = 0 \mid Im3\ e5 = 0$   
 $\mid Im4\ e5 = 0 \mid Im5\ e5 = 1 \mid Im6\ e5 = 0 \mid Im7\ e5 = 0$

**primcorec** *octo-e6* :: *octo* (e6)  
**where**  $Ree\ e6 = 0 \mid Im1\ e6 = 0 \mid Im2\ e6 = 0 \mid Im3\ e6 = 0$   
 $\mid Im4\ e6 = 0 \mid Im5\ e6 = 0 \mid Im6\ e6 = 1 \mid Im7\ e6 = 0$

**primcorec** *octo-e7* :: *octo* (e7)  
**where**  $Ree\ e7 = 0 \mid Im1\ e7 = 0 \mid Im2\ e7 = 0 \mid Im3\ e7 = 0$   
 $\mid Im4\ e7 = 0 \mid Im5\ e7 = 0 \mid Im6\ e7 = 0 \mid Im7\ e7 = 1$   
**end**

## 2.2 Addition and Subtraction: An Abelian Group

**instantiation** *octo* :: *ab-group-add*

**begin**

**primcorec** *zero-octo*  
**where**  $Ree\ 0 = 0 \mid Im1\ 0 = 0 \mid Im2\ 0 = 0 \mid Im3\ 0 = 0$   
 $\mid Im4\ 0 = 0 \mid Im5\ 0 = 0 \mid Im6\ 0 = 0 \mid Im7\ 0 = 0$

**primcorec** *plus-octo*  
**where**  
 $Ree\ (x + y) = Ree\ x + Ree\ y$   
 $\mid Im1\ (x + y) = Im1\ x + Im1\ y$   
 $\mid Im2\ (x + y) = Im2\ x + Im2\ y$   
 $\mid Im3\ (x + y) = Im3\ x + Im3\ y$   
 $\mid Im4\ (x + y) = Im4\ x + Im4\ y$

|  $Im5 (x + y) = Im5 x + Im5 y$   
|  $Im6 (x + y) = Im6 x + Im6 y$   
|  $Im7 (x + y) = Im7 x + Im7 y$

**primcorec** *uminus-octo*

**where**

$Ree (- x) = - Ree x$   
|  $Im1 (- x) = - Im1 x$   
|  $Im2 (- x) = - Im2 x$   
|  $Im3 (- x) = - Im3 x$   
|  $Im4 (- x) = - Im4 x$   
|  $Im5 (- x) = - Im5 x$   
|  $Im6 (- x) = - Im6 x$   
|  $Im7 (- x) = - Im7 x$

**primcorec** *minus-octo*

**where**

$Ree (x - y) = Ree x - Ree y$   
|  $Im1 (x - y) = Im1 x - Im1 y$   
|  $Im2 (x - y) = Im2 x - Im2 y$   
|  $Im3 (x - y) = Im3 x - Im3 y$   
|  $Im4 (x - y) = Im4 x - Im4 y$   
|  $Im5 (x - y) = Im5 x - Im5 y$   
|  $Im6 (x - y) = Im6 x - Im6 y$   
|  $Im7 (x - y) = Im7 x - Im7 y$

**instance**

*<proof>*

**end**

**lemma** *octo-eq-0-iff*:

$x = 0 \iff Ree x^2 + Im1 x^2 + Im2 x^2 + Im3 x^2 +$   
 $Im4 x^2 + Im5 x^2 + Im6 x^2 + Im7 x^2 = 0$

*<proof>*

## 2.3 A Normed Vector Space

**instantiation** *octo* :: *real-vector*

**begin**

**primcorec** *scaleR-octo*

**where**

$Ree (scaleR r x) = r * Ree x$   
|  $Im1 (scaleR r x) = r * Im1 x$   
|  $Im2 (scaleR r x) = r * Im2 x$   
|  $Im3 (scaleR r x) = r * Im3 x$   
|  $Im4 (scaleR r x) = r * Im4 x$

```

| Im5 (scaleR r x) = r * Im5 x
| Im6 (scaleR r x) = r * Im6 x
| Im7 (scaleR r x) = r * Im7 x

```

**instance**  
 ⟨proof⟩

**end**

**instantiation** *octo::one*  
**begin**  
**primcorec** *one-octo*

```

where
  Ree 1 = 1 | Im1 1 = 0 | Im2 1 = 0 | Im3 1 = 0 |
  Im4 1 = 0 | Im5 1 = 0 | Im6 1 = 0 | Im7 1 = 0

```

**instance** ⟨proof⟩  
**end**

**fun** *octo-proj*

```

where
  octo-proj x 0 = ( Ree (x) )
| octo-proj x (Suc 0) = ( Im1(x) )
| octo-proj x (Suc (Suc 0)) = ( Im2 ( x) )
| octo-proj x (Suc (Suc (Suc 0))) = ( Im3( x) )
| octo-proj x (Suc (Suc (Suc (Suc 0)))) = ( Im4( x) )
| octo-proj x (Suc(Suc (Suc (Suc (Suc 0)))))) = ( Im5( x) )
| octo-proj x (Suc(Suc (Suc (Suc (Suc (Suc 0)))))) = ( Im6( x) )
| octo-proj x (Suc( Suc(Suc (Suc (Suc (Suc (Suc 0)))))) ) = ( Im7( x) )

```

**lemma** *octo-proj-add*:

```

assumes  $i \leq 7$ 
shows  $octo-proj (x+y) i = octo-proj x i + octo-proj y i$ 

```

⟨proof⟩

**instantiation** *octo ::real-normed-vector*

**begin**

**definition**  $norm\ x = \sqrt{(Ree\ x)^2 + (Im1\ x)^2 + (Im2\ x)^2 + (Im3\ x)^2 + (Im4\ x)^2 + (Im5\ x)^2 + (Im6\ x)^2 + (Im7\ x)^2}$  **for**  $x::octo$

**definition**  $sgn\ x = x /_R\ norm\ x$  **for**  $x :: octo$

**definition**  $dist\ x\ y = norm\ (x - y)$  **for**  $x\ y :: octo$

**definition** [code del]:

(*uniformity* :: (octo × octo) filter) = (INF e∈{0 <..}. principal {(x, y). dist x y

< e})

**definition** [code del]:

*open* ( $U :: \text{octo set}$ )  $\longleftrightarrow (\forall x \in U. \text{eventually } (\lambda(x', y). x' = x \longrightarrow y \in U)$   
*uniformity*)

**lemma** *norm-eq-L2*: *norm*  $x = L2\text{-set}$  (*octo-proj*  $x$ ) {..7}

*<proof>*

**instance** *<proof>*

**end**

**lemma** *norm-octo-squared*:

$\text{norm } x^{\wedge} 2 = \text{Ree } x^{\wedge} 2 + \text{Im1 } x^{\wedge} 2 + \text{Im2 } x^{\wedge} 2 + \text{Im3 } x^{\wedge} 2 +$   
 $\text{Im4 } x^{\wedge} 2 + \text{Im5 } x^{\wedge} 2 + \text{Im6 } x^{\wedge} 2 + \text{Im7 } x^{\wedge} 2$

*<proof>*

**instantiation** *octo* :: *real-inner*

**begin**

**definition** *inner-octo* **where**

*inner-octo*  $x y = \text{Ree } x * \text{Ree } y + \text{Im1 } x * \text{Im1 } y + \text{Im2 } x * \text{Im2 } y + \text{Im3 } x * \text{Im3 } y$   
 $+ \text{Im4 } x * \text{Im4 } y + \text{Im5 } x * \text{Im5 } y + \text{Im6 } x * \text{Im6 } y + \text{Im7 } x * \text{Im7 } y$  **for**  
 $x y :: \text{octo}$

**instance**

*<proof>*

**end**

**lemma** *octo-inner-1* [simp]: *inner* 1  $x = \text{Ree } x$

**and** *octo-inner-1-right* [simp]: *inner*  $x$  1 =  $\text{Ree } x$

*<proof>*

**lemma** *octo-inner-e1-left* [simp]: *inner* e1  $x = \text{Im1 } x$

**and** *octo-inner-e1-right* [simp]: *inner*  $x$  e1 =  $\text{Im1 } x$

*<proof>*

**lemma** *octo-inner-e2-left* [simp]: *inner* e2  $x = \text{Im2 } x$

**and** *octo-inner-e2-right* [simp]: *inner*  $x$  e2 =  $\text{Im2 } x$

*<proof>*

**lemma** *octo-inner-e3-left* [simp]: *inner* e3  $x = \text{Im3 } x$

**and** *octo-inner-e3-right* [simp]: *inner*  $x$  e3 =  $\text{Im3 } x$

*<proof>*

**lemma** *octo-inner-e4-left* [simp]: *inner* e4  $x = \text{Im4 } x$

**and** *octo-inner-e4-right* [*simp*]:  $inner\ x\ e4 = Im4\ x$   
 ⟨*proof*⟩

**lemma** *octo-inner-e5-left* [*simp*]:  $inner\ e5\ x = Im5\ x$   
**and** *octo-inner-e5-right* [*simp*]:  $inner\ x\ e5 = Im5\ x$   
 ⟨*proof*⟩

**lemma** *octo-inner-e6-left* [*simp*]:  $inner\ e6\ x = Im6\ x$   
**and** *octo-inner-e6-right* [*simp*]:  $inner\ x\ e6 = Im6\ x$   
 ⟨*proof*⟩

**lemma** *octo-inner-e7-left* [*simp*]:  $inner\ e7\ x = Im7\ x$   
**and** *octo-inner-e7-right* [*simp*]:  $inner\ x\ e7 = Im7\ x$   
 ⟨*proof*⟩

**lemma** *octo-norm-pow-2-inner*:  $(norm\ x) \wedge 2 = inner\ x\ x$  **for**  $x::octo$   
 ⟨*proof*⟩

**lemma** *octo-norm-property*:  
 $inner\ x\ y = (1/2)*((norm(x+y))\wedge 2 - (norm(x))\wedge 2 - (norm(y))\wedge 2)$  **for**  $x\ y$   
 $::octo$   
 ⟨*proof*⟩

## 2.4 The Octonionic product and related properties and lemmas

The multiplication is defined following one of the 480 equivalent multiplication tables in an analogy to the definition of the 7D cross product.

**instantiation** *octo* :: *times*

**begin**

**definition** *times-octo* :: [*octo*, *octo*]  $\Rightarrow$  *octo*

**where**

$(a * b) = (let$   
 $t0 = Re\ a * Re\ b - Im1\ a * Im1\ b - Im2\ a * Im2\ b - Im3\ a * Im3\ b$   
 $\quad - Im4\ a * Im4\ b - Im5\ a * Im5\ b - Im6\ a * Im6\ b - Im7\ a * Im7\ b ;$   
 $t1 = Re\ a * Im1\ b + Im1\ a * Re\ b + Im2\ a * Im4\ b + Im3\ a * Im7\ b -$   
 $\quad Im4\ a * Im2\ b + Im5\ a * Im6\ b - Im6\ a * Im5\ b - Im7\ a * Im3\ b ;$   
 $t2 = Re\ a * Im2\ b - Im1\ a * Im4\ b + Im2\ a * Re\ b + Im3\ a * Im5\ b$   
 $\quad + Im4\ a * Im1\ b - Im5\ a * Im3\ b + Im6\ a * Im7\ b - Im7\ a * Im6\ b ;$   
 $t3 = Re\ a * Im3\ b - Im1\ a * Im7\ b - Im2\ a * Im5\ b + Im3\ a * Re\ b + Im4$   
 $a * Im6\ b$   
 $\quad + Im5\ a * Im2\ b - Im6\ a * Im4\ b + Im7\ a * Im1\ b ;$   
 $t4 = Re\ a * Im4\ b + Im1\ a * Im2\ b - Im2\ a * Im1\ b - Im3\ a * Im6\ b + Im4$   
 $a * Re\ b$   
 $\quad + Im5\ a * Im7\ b + Im6\ a * Im3\ b - Im7\ a * Im5\ b ;$   
 $t5 = Re\ a * Im5\ b - Im1\ a * Im6\ b + Im2\ a * Im3\ b - Im3\ a * Im2\ b - Im4$   
 $a * Im7\ b$   
 $\quad + Im5\ a * Re\ b + Im6\ a * Im1\ b + Im7\ a * Im4\ b ;$

$$\begin{aligned}
t6 &= \text{Ree } a * \text{Im6 } b + \text{Im1 } a * \text{Im5 } b - \text{Im2 } a * \text{Im7 } b + \text{Im3 } a * \text{Im4 } b - \text{Im4 } \\
& a * \text{Im3 } b \\
& - \text{Im5 } a * \text{Im1 } b + \text{Im6 } a * \text{Ree } b + \text{Im7 } a * \text{Im2 } b ; \\
t7 &= \text{Ree } a * \text{Im7 } b + \text{Im1 } a * \text{Im3 } b + \text{Im2 } a * \text{Im6 } b - \text{Im3 } a * \text{Im1 } b + \text{Im4 } \\
& a * \text{Im5 } b \\
& - \text{Im5 } a * \text{Im4 } b - \text{Im6 } a * \text{Im2 } b + \text{Im7 } a * \text{Ree } b \\
& \text{in Octo } t0 \ t1 \ t2 \ t3 \ t4 \ t5 \ t6 \ t7)
\end{aligned}$$

**instance**  $\langle \text{proof} \rangle$

**end**

**instantiation** *octo* :: *inverse*  
**begin**

**primcorec** *inverse-octo*

**where**

$$\begin{aligned}
& \text{Ree } (\text{inverse } x) = \text{Ree } x / (\text{Ree } x^{\wedge} 2 + \text{Im1 } x^{\wedge} 2 + \text{Im2 } x^{\wedge} 2 + \text{Im3 } x^{\wedge} 2 \\
& \quad + \text{Im4 } x^{\wedge} 2 + \text{Im5 } x^{\wedge} 2 + \text{Im6 } x^{\wedge} 2 + \text{Im7 } x^{\wedge} 2) \\
| \text{Im1 } (\text{inverse } x) &= - (\text{Im1 } x) / (\text{Ree } x^{\wedge} 2 + \text{Im1 } x^{\wedge} 2 + \text{Im2 } x^{\wedge} 2 + \text{Im3 } x^{\wedge} 2 \\
& \quad + \text{Im4 } x^{\wedge} 2 + \text{Im5 } x^{\wedge} 2 + \text{Im6 } x^{\wedge} 2 + \text{Im7 } x^{\wedge} 2) \\
| \text{Im2 } (\text{inverse } x) &= - (\text{Im2 } x) / (\text{Ree } x^{\wedge} 2 + \text{Im1 } x^{\wedge} 2 + \text{Im2 } x^{\wedge} 2 + \text{Im3 } x^{\wedge} 2 \\
& \quad + \text{Im4 } x^{\wedge} 2 + \text{Im5 } x^{\wedge} 2 + \text{Im6 } x^{\wedge} 2 + \text{Im7 } x^{\wedge} 2) \\
| \text{Im3 } (\text{inverse } x) &= - (\text{Im3 } x) / (\text{Ree } x^{\wedge} 2 + \text{Im1 } x^{\wedge} 2 + \text{Im2 } x^{\wedge} 2 + \text{Im3 } x^{\wedge} 2 \\
& \quad + \text{Im4 } x^{\wedge} 2 + \text{Im5 } x^{\wedge} 2 + \text{Im6 } x^{\wedge} 2 + \text{Im7 } x^{\wedge} 2) \\
| \text{Im4 } (\text{inverse } x) &= - (\text{Im4 } x) / (\text{Ree } x^{\wedge} 2 + \text{Im1 } x^{\wedge} 2 + \text{Im2 } x^{\wedge} 2 + \text{Im3 } x^{\wedge} 2 \\
& \quad + \text{Im4 } x^{\wedge} 2 + \text{Im5 } x^{\wedge} 2 + \text{Im6 } x^{\wedge} 2 + \text{Im7 } x^{\wedge} 2) \\
| \text{Im5 } (\text{inverse } x) &= - (\text{Im5 } x) / (\text{Ree } x^{\wedge} 2 + \text{Im1 } x^{\wedge} 2 + \text{Im2 } x^{\wedge} 2 + \text{Im3 } x^{\wedge} 2 \\
& \quad + \text{Im4 } x^{\wedge} 2 + \text{Im5 } x^{\wedge} 2 + \text{Im6 } x^{\wedge} 2 + \text{Im7 } x^{\wedge} 2) \\
| \text{Im6 } (\text{inverse } x) &= - (\text{Im6 } x) / (\text{Ree } x^{\wedge} 2 + \text{Im1 } x^{\wedge} 2 + \text{Im2 } x^{\wedge} 2 + \text{Im3 } x^{\wedge} 2 \\
& \quad + \text{Im4 } x^{\wedge} 2 + \text{Im5 } x^{\wedge} 2 + \text{Im6 } x^{\wedge} 2 + \text{Im7 } x^{\wedge} 2) \\
| \text{Im7 } (\text{inverse } x) &= - (\text{Im7 } x) / (\text{Ree } x^{\wedge} 2 + \text{Im1 } x^{\wedge} 2 + \text{Im2 } x^{\wedge} 2 + \text{Im3 } x^{\wedge} 2 \\
& \quad + \text{Im4 } x^{\wedge} 2 + \text{Im5 } x^{\wedge} 2 + \text{Im6 } x^{\wedge} 2 + \text{Im7 } x^{\wedge} 2)
\end{aligned}$$

**definition**  $x \text{ div } y = x * (\text{inverse } y)$  **for**  $x \ y :: \text{octo}$

**instance**  $\langle \text{proof} \rangle$

**end**

**lemma** *octo-mult-components*:

$$\begin{aligned}
\text{Ree } (x * y) &= \text{Ree } x * \text{Ree } y - \text{Im1 } x * \text{Im1 } y - \text{Im2 } x * \text{Im2 } y - \text{Im3 } x * \\
&\text{Im3 } y \\
&- \text{Im4 } x * \text{Im4 } y - \text{Im5 } x * \text{Im5 } y - \text{Im6 } x * \text{Im6 } y - \text{Im7 } x * \text{Im7 } y \\
\text{Im1 } (x * y) &= \text{Ree } x * \text{Im1 } y + \text{Im1 } x * \text{Ree } y + \text{Im2 } x * \text{Im4 } y + \text{Im3 } x * \\
&\text{Im7 } y - \\
&\text{Im4 } x * \text{Im2 } y + \text{Im5 } x * \text{Im6 } y - \text{Im6 } x * \text{Im5 } y - \text{Im7 } x * \text{Im3 } y \\
\text{Im2 } (x * y) &= \text{Ree } x * \text{Im2 } y - \text{Im1 } x * \text{Im4 } y + \text{Im2 } x * \text{Ree } y + \text{Im3 } x * \\
&* \text{Im5 } y \\
&+ \text{Im4 } x * \text{Im1 } y - \text{Im5 } x * \text{Im3 } y + \text{Im6 } x * \text{Im7 } y - \text{Im7 } x * \text{Im6 } y \\
\text{Im3 } (x * y) &= \text{Ree } x * \text{Im3 } y - \text{Im1 } x * \text{Im7 } y - \text{Im2 } x * \text{Im5 } y + \text{Im3 } x * \text{Ree } \\
&y + \text{Im4 } x * \text{Im6 } y \\
&+ \text{Im5 } x * \text{Im2 } y - \text{Im6 } x * \text{Im4 } y + \text{Im7 } x * \text{Im1 } y \\
\text{Im4 } (x * y) &= \text{Ree } x * \text{Im4 } y + \text{Im1 } x * \text{Im2 } y - \text{Im2 } x * \text{Im1 } y - \text{Im3 } x * \\
&\text{Im6 } y + \text{Im4 } x * \text{Ree } y \\
&+ \text{Im5 } x * \text{Im7 } y + \text{Im6 } x * \text{Im3 } y - \text{Im7 } x * \text{Im5 } y \\
\text{Im5 } (x * y) &= \text{Ree } x * \text{Im5 } y - \text{Im1 } x * \text{Im6 } y + \text{Im2 } x * \text{Im3 } y - \text{Im3 } x * \\
&\text{Im2 } y - \text{Im4 } x * \text{Im7 } y \\
&+ \text{Im5 } x * \text{Ree } y + \text{Im6 } x * \text{Im1 } y + \text{Im7 } x * \text{Im4 } y \\
\text{Im6 } (x * y) &= \text{Ree } x * \text{Im6 } y + \text{Im1 } x * \text{Im5 } y - \text{Im2 } x * \text{Im7 } y + \text{Im3 } x * \\
&\text{Im4 } y - \text{Im4 } x * \text{Im3 } y \\
&- \text{Im5 } x * \text{Im1 } y + \text{Im6 } x * \text{Ree } y + \text{Im7 } x * \text{Im2 } y \\
\text{Im7 } (x * y) &= \text{Ree } x * \text{Im7 } y + \text{Im1 } x * \text{Im3 } y + \text{Im2 } x * \text{Im6 } y - \text{Im3 } x * \\
&\text{Im1 } y + \text{Im4 } x * \text{Im5 } y \\
&- \text{Im5 } x * \text{Im4 } y - \text{Im6 } x * \text{Im2 } y + \text{Im7 } x * \text{Ree } y \\
&\langle \text{proof} \rangle
\end{aligned}$$

**lemma** *octo-distrib-left* :

$$a * (b + c) = a * b + a * c \text{ for } a \ b \ c :: \text{octo}$$

*<proof>*

**lemma** *octo-distrib-right* :

$$(b + c) * a = b * a + c * a \text{ for } a \ b \ c :: \text{octo}$$

*<proof>*

**lemma** *multiplicative-norm-octo*:  $\text{norm } (x * y) = \text{norm } x * \text{norm } y$  for  $x \ y :: \text{octo}$   
*<proof>*

**lemma** *mult-1-right-octo* [*simp*]:  $x * 1 = (x :: \text{octo})$

**and** *mult-1-left-octo* [*simp*]:  $1 * x = (x :: \text{octo})$   
*<proof>*

**instance** *octo* :: *power* *<proof>*

**lemma** *power2-eq-square-octo*:  $x \wedge 2 = (x * x :: \text{octo})$   
*<proof>*

**lemma** *octo-product-alternative-left*:  $x * (x * y) = (x * x :: \text{octo}) * y$   
*<proof>*

**lemma** *octo-product-alternative-right*:  $x * (y * y) = (x * y :: octo) * y$   
 ⟨proof⟩

**lemma** *octo-product-flexible*:  $(x * y) * x = x * (y * x :: octo)$   
 ⟨proof⟩

**lemma** *octo-power-commutes*:  $x \hat{=} y * x = x * (x \hat{=} y :: octo)$   
 ⟨proof⟩

**lemma** *octo-product-noncommutative*:  $\neg(\forall x y :: octo. (x * y = y * x))$   
 ⟨proof⟩

**lemma** *octo-product-nonassociative* :  
 $\neg(\forall x y z :: octo. x * (y * z) = (x * y) * z)$   
 ⟨proof⟩

## 2.5 Embedding of the Reals into the Octonions

**definition** *octo-of-real* ::  $real \Rightarrow octo$   
 where *octo-of-real*  $r = scaleR\ r\ 1$

**definition** *octo-of-nat* ::  $nat \Rightarrow octo$   
 where *octo-of-nat*  $r = scaleR\ r\ 1$

**definition** *octo-of-int* ::  $int \Rightarrow octo$   
 where *octo-of-int*  $r = scaleR\ r\ 1$

**lemma** *octo-of-nat-sel* [simp]:  
 $Ree\ (octo-of-nat\ x) = of-nat\ x$   $Im1\ (octo-of-nat\ x) = 0$   $Im2\ (octo-of-nat\ x) = 0$   
 $Im3\ (octo-of-nat\ x) = 0$   $Im4\ (octo-of-nat\ x) = 0$   $Im5\ (octo-of-nat\ x) = 0$   
 $Im6\ (octo-of-nat\ x) = 0$   $Im7\ (octo-of-nat\ x) = 0$   
 ⟨proof⟩

**lemma** *octo-of-real-sel* [simp]:  
 $Ree\ (octo-of-real\ x) = x$   $Im1\ (octo-of-real\ x) = 0$   $Im2\ (octo-of-real\ x) = 0$   
 $Im3\ (octo-of-real\ x) = 0$   $Im4\ (octo-of-real\ x) = 0$   $Im5\ (octo-of-real\ x) = 0$   
 $Im6\ (octo-of-real\ x) = 0$   $Im7\ (octo-of-real\ x) = 0$   
 ⟨proof⟩

**lemma** *octo-of-int-sel* [simp]:  
 $Ree\ (octo-of-int\ x) = of-int\ x$   $Im1\ (octo-of-int\ x) = 0$   $Im2\ (octo-of-int\ x) = 0$   
 $Im3\ (octo-of-int\ x) = 0$   $Im4\ (octo-of-int\ x) = 0$   $Im5\ (octo-of-int\ x) = 0$   
 $Im6\ (octo-of-int\ x) = 0$   $Im7\ (octo-of-int\ x) = 0$   
 ⟨proof⟩

**lemma** *scaleR-conv-octo-of-real*:  $scaleR\ r\ x = octo-of-real\ r * x$   
 ⟨proof⟩



**lemma** *octo-of-real-0* [*simp*]: *octo-of-real* 0 = 0  
⟨*proof*⟩

**lemma** *octo-of-real-1* [*simp*]: *octo-of-real* 1 = 1  
⟨*proof*⟩

**lemma** *octo-of-real-add* [*simp*]: *octo-of-real* (x + y) = *octo-of-real* x + *octo-of-real* y  
⟨*proof*⟩

**lemma** *octo-of-real-minus* [*simp*]: *octo-of-real* (- x) = - *octo-of-real* x  
⟨*proof*⟩

**lemma** *octo-of-real-diff* [*simp*]: *octo-of-real* (x - y) = *octo-of-real* x - *octo-of-real* y  
⟨*proof*⟩

**lemma** *octo-of-real-mult* [*simp*]: *octo-of-real* (x \* y) = *octo-of-real* x \* *octo-of-real* y  
⟨*proof*⟩

**lemma** *octo-of-real-sum* [*simp*]: *octo-of-real* (sum f s) = (∑ x∈s. *octo-of-real* (f x))  
⟨*proof*⟩

**lemma** *octo-of-real-power* [*simp*]:  
*octo-of-real* (x ^ y) = (*octo-of-real* x :: *octo*) ^ y  
⟨*proof*⟩

**lemma** *octo-of-real-eq-iff* [*simp*]: *octo-of-real* x = *octo-of-real* y ↔ x = y  
⟨*proof*⟩

**lemmas** *octo-of-real-eq-0-iff* [*simp*] = *octo-of-real-eq-iff* [*of - 0, simplified*]

**lemmas** *octo-of-real-eq-1-iff* [*simp*] = *octo-of-real-eq-iff* [*of - 1, simplified*]

**lemma** *minus-octo-of-real-eq-octo-of-real-iff* [*simp*]: -*octo-of-real* x = *octo-of-real* y ↔ -x = y  
⟨*proof*⟩

**lemma** *octo-of-real-eq-minus-of-real-iff* [*simp*]: *octo-of-real* x = -*octo-of-real* y ↔ x = -y  
⟨*proof*⟩

**lemma** *octo-of-real-of-nat-eq* [*simp*]: *octo-of-real* (*of-nat* x) = *octo-of-nat* x  
⟨*proof*⟩

**lemma** *octo-of-real-of-int-eq* [*simp*]: *octo-of-real* (*of-int* z) = *octo-of-int* z  
⟨*proof*⟩

**lemma** *octo-of-int-of-nat*: *octo-of-int* (*of-nat* n) = *octo-of-nat* n

*<proof>*

**lemma** *octo-of-nat-add* [*simp*]:  $octo-of-nat (a + b) = octo-of-nat a + octo-of-nat b$

**and** *octo-of-nat-mult* [*simp*]:  $octo-of-nat (a * b) = octo-of-nat a * octo-of-nat b$

**and** *octo-of-nat-diff* [*simp*]:  $b \leq a \implies octo-of-nat (a - b) = octo-of-nat a - octo-of-nat b$

**and** *octo-of-nat-0* [*simp*]:  $octo-of-nat 0 = 0$

**and** *octo-of-nat-1* [*simp*]:  $octo-of-nat 1 = 1$

**and** *octo-of-nat-Suc-0* [*simp*]:  $octo-of-nat (Suc 0) = 1$

*<proof>*

**lemma** *octo-of-int-add* [*simp*]:  $octo-of-int (a + b) = octo-of-int a + octo-of-int b$

**and** *octo-of-int-mult* [*simp*]:  $octo-of-int (a * b) = octo-of-int a * octo-of-int b$

**and** *octo-of-int-diff* [*simp*]:  $b \leq a \implies octo-of-int (a - b) = octo-of-int a - octo-of-int b$

**and** *octo-of-int-0* [*simp*]:  $octo-of-int 0 = 0$

**and** *octo-of-int-1* [*simp*]:  $octo-of-int 1 = 1$

*<proof>*

**instance** *octo* :: *numeral* *<proof>*

**lemma** *numeral-octo-conv-of-nat*:  $numeral x = octo-of-nat (numeral x)$

*<proof>*

**lemma** *numeral-octo-sel* [*simp*]:

*Ree* (*numeral n*) = *numeral n* *Im1* (*numeral n*) = 0 *Im2* (*numeral n*) = 0

*Im3* (*numeral n*) = 0 *Im4* (*numeral n*) = 0 *Im5* (*numeral n*) = 0

*Im6* (*numeral n*) = 0 *Im7* (*numeral n*) = 0

*<proof>*

**lemma** *octo-of-real-numeral* [*simp*]:  $octo-of-real (numeral w) = numeral w$

*<proof>*

**lemma** *octo-of-real-neg-numeral* [*simp*]:  $octo-of-real (- numeral w) = - numeral w$

*<proof>*

**lemma** *octo-of-real-times-commute*:  $octo-of-real r * q = q * octo-of-real r$

*<proof>*

**lemma** *octo-of-real-times-conv-scaleR*:  $octo-of-real x * y = scaleR x y$

*<proof>*

**lemma** *octo-mult-scaleR-left*:  $(r *_R x) * y = r *_R (x * y :: octo)$

*<proof>*

**lemma** *octo-mult-scaleR-right*:  $x * (r *_R y) = r *_R (x * y :: octo)$

*<proof>*

**lemma** *scaleR-octo-of-real* [simp]:  $\text{scaleR } r \ (\text{octo-of-real } s) = \text{octo-of-real } (r * s)$   
 ⟨proof⟩

**lemma** *octo-of-real-times-left-commute*:  $\text{octo-of-real } r * (x * q) = x * (\text{octo-of-real } r * q)$   
 ⟨proof⟩

**lemma** *nonzero-octo-of-real-inverse*:  
 $x \neq 0 \implies \text{octo-of-real } (\text{inverse } x) = \text{inverse } (\text{octo-of-real } x :: \text{octo})$   
 ⟨proof⟩

**lemma** *octo-of-real-inverse* [simp]:  
 $\text{octo-of-real } (\text{inverse } x) = \text{inverse } (\text{octo-of-real } x)$   
 ⟨proof⟩

**lemma** *nonzero-octo-of-real-divide*:  
 $y \neq 0 \implies \text{octo-of-real } (x / y) = (\text{octo-of-real } x / \text{octo-of-real } y :: \text{octo})$   
 ⟨proof⟩

**lemma** *octo-of-real-divide* [simp]:  
 $\text{octo-of-real } (x / y) = (\text{octo-of-real } x / \text{octo-of-real } y :: \text{octo})$   
 ⟨proof⟩

**lemma** *octo-of-real-inverse-collapse* [simp]:  
**assumes**  $c \neq 0$   
**shows**  $\text{octo-of-real } c * \text{octo-of-real } (\text{inverse } c) = 1$   
 $\text{octo-of-real } (\text{inverse } c) * \text{octo-of-real } c = 1$   
 ⟨proof⟩

**lemma** *octo-divide-numeral*:  
**fixes**  $x :: \text{octo}$  **shows**  $x / \text{numeral } y = x /_R \text{numeral } y$   
 ⟨proof⟩

**lemma** *octo-divide-numeral-sel* [simp]:  
 $\text{Ree } (x / \text{numeral } w) = \text{Ree } x / \text{numeral } w$   
 $\text{Im1 } (x / \text{numeral } w) = \text{Im1 } x / \text{numeral } w$   
 $\text{Im2 } (x / \text{numeral } w) = \text{Im2 } x / \text{numeral } w$   
 $\text{Im3 } (x / \text{numeral } w) = \text{Im3 } x / \text{numeral } w$   
 $\text{Im4 } (x / \text{numeral } w) = \text{Im4 } x / \text{numeral } w$   
 $\text{Im5 } (x / \text{numeral } w) = \text{Im5 } x / \text{numeral } w$   
 $\text{Im6 } (x / \text{numeral } w) = \text{Im6 } x / \text{numeral } w$   
 $\text{Im7 } (x / \text{numeral } w) = \text{Im7 } x / \text{numeral } w$   
 ⟨proof⟩

**lemma** *octo-norm-units* [simp]:  
 $\text{norm } \text{octo-e1} = 1 \ \text{norm } (\text{e2} :: \text{octo}) = 1 \ \text{norm } (\text{e3} :: \text{octo}) = 1$   
 $\text{norm } (\text{e4} :: \text{octo}) = 1 \ \text{norm } (\text{e5} :: \text{octo}) = 1 \ \text{norm } (\text{e6} :: \text{octo}) = 1 \ \text{norm } (\text{e7} :: \text{octo}) = 1$

*<proof>*

**lemma** *e1-nz* [*simp*]:  $e1 \neq 0$   
**and** *e2-nz* [*simp*]:  $e2 \neq 0$   
**and** *e3-nz* [*simp*]:  $e3 \neq 0$   
**and** *e4-nz* [*simp*]:  $e4 \neq 0$   
**and** *e5-nz* [*simp*]:  $e5 \neq 0$   
**and** *e6-nz* [*simp*]:  $e6 \neq 0$   
**and** *e7-nz* [*simp*]:  $e7 \neq 0$   
*<proof>*

## 2.6 "Expansion" into the traditional notation

**lemma** *octo-unfold*:

$q = (\text{Ree } q) *_R e0 + (\text{Im1 } q) *_R e1 + (\text{Im2 } q) *_R e2 + (\text{Im3 } q) *_R e3$   
 $+ (\text{Im4 } q) *_R e4 + (\text{Im5 } q) *_R e5 + (\text{Im6 } q) *_R e6 + (\text{Im7 } q) *_R e7$   
*<proof>*

**lemma** *octo-trad*: *Octo x y z w u v q g =*

$x *_R e0 + y *_R e1 + z *_R e2 + w *_R e3 + u *_R e4 + v *_R e5 + q *_R$   
 $e6 + g *_R e7$   
*<proof>*

**lemma** *octo-of-real-eq-Octo*: *octo-of-real a = Octo a 0 0 0 0 0 0*

*<proof>*

**lemma** *e1-squared* [*simp*]:  $e1 \wedge 2 = -1$   
**and** *e2-squared* [*simp*]:  $e2 \wedge 2 = -1$   
**and** *e3-squared* [*simp*]:  $e3 \wedge 2 = -1$   
**and** *e4-squared* [*simp*]:  $e4 \wedge 2 = -1$   
**and** *e5-squared* [*simp*]:  $e5 \wedge 2 = -1$   
**and** *e6-squared* [*simp*]:  $e6 \wedge 2 = -1$   
**and** *e7-squared* [*simp*]:  $e7 \wedge 2 = -1$   
*<proof>*

**lemma** *inverse-e1* [*simp*]: *inverse e1 = -e1*

**and** *inverse-e2* [*simp*]: *inverse e2 = -e2*  
**and** *inverse-e3* [*simp*]: *inverse e3 = -e3*  
**and** *inverse-e4* [*simp*]: *inverse e4 = -e4*  
**and** *inverse-e5* [*simp*]: *inverse e5 = -e5*  
**and** *inverse-e6* [*simp*]: *inverse e6 = -e6*  
**and** *inverse-e7* [*simp*]: *inverse e7 = -e7*  
*<proof>*

## 2.7 Conjugate of an octonion and related properties.

**primcorec** *cnj* :: *octo*  $\Rightarrow$  *octo*

**where**

$\text{Ree } (\text{cnj } z) = \text{Ree } z$   
 $\text{Im1 } (\text{cnj } z) = - \text{Im1 } z$

|  $Im2 (cnj z) = - Im2 z$   
|  $Im3 (cnj z) = - Im3 z$   
|  $Im4 (cnj z) = - Im4 z$   
|  $Im5 (cnj z) = - Im5 z$   
|  $Im6 (cnj z) = - Im6 z$   
|  $Im7 (cnj z) = - Im7 z$

**lemma** *cnj-cancel-iff* [*simp*]:  $cnj x = cnj y \longleftrightarrow x = y$

*<proof>*

**lemma** *cnj-cnj* [*simp*]:

$cnj(cnj q) = q$

*<proof>*

**lemma** *cnj-of-real* [*simp*]:  $cnj(octo-of-real x) = octo-of-real x$

*<proof>*

**lemma** *cnj-zero* [*simp*]:  $cnj 0 = 0$

*<proof>*

**lemma** *cnj-zero-iff* [*iff*]:  $cnj z = 0 \longleftrightarrow z = 0$

*<proof>*

**lemma** *cnj-one* [*simp*]:  $cnj 1 = 1$

*<proof>*

**lemma** *cnj-one-iff* [*simp*]:  $cnj z = 1 \longleftrightarrow z = 1$

*<proof>*

**lemma** *octo-norm-cnj* [*simp*]:  $norm(cnj q) = norm q$

*<proof>*

**lemma** *cnj-add* [*simp*]:  $cnj (x + y) = cnj x + cnj y$

*<proof>*

**lemma** *cnj-sum* [*simp*]:  $cnj (sum f S) = (\sum x \in S. cnj (f x))$

*<proof>*

**lemma** *cnj-diff* [*simp*]:  $cnj (x - y) = cnj x - cnj y$

*<proof>*

**lemma** *cnj-minus* [*simp*]:  $cnj (- x) = - cnj x$

*<proof>*

**lemma** *cnj-inverse* [*simp*]:  $cnj (inverse x) = inverse (cnj x)$  **for**  $x y :: octo$

*<proof>*

**lemma** *cnj-mult* [*simp*]:  $cnj (x * y) = cnj y * cnj x$  **for**  $x y :: octo$

$\langle proof \rangle$

**lemma** *cnj-divide* [*simp*]:  $cnj (x / y) = (inverse (cnj y)) * cnj x$   
for  $x y :: octo$   
 $\langle proof \rangle$

**lemma** *cnj-power* [*simp*]:  $cnj (x \hat{ } y) = (cnj x) \hat{ } y$  for  $x :: octo$   
 $\langle proof \rangle$

**lemma** *cnj-of-nat* [*simp*]:  $cnj (octo-of-nat x) = octo-of-nat (cnj x)$   
 $\langle proof \rangle$

**lemma** *cnj-of-int* [*simp*]:  $cnj (octo-of-int x) = octo-of-nat (cnj x)$   
 $\langle proof \rangle$

**lemma** *cnj-numeral* [*simp*]:  $cnj (numeral x) = numeral (cnj x)$   
 $\langle proof \rangle$

**lemma** *cnj-neg-numeral* [*simp*]:  $cnj (- numeral x) = - numeral (cnj x)$   
 $\langle proof \rangle$

**lemma** *cnj-scaleR* [*simp*]:  $cnj (scaleR r x) = scaleR r (cnj x)$   
 $\langle proof \rangle$

**lemma** *cnj-units* [*simp*]:  $cnj e1 = -e1$   $cnj e2 = -e2$   $cnj e3 = -e3$   
 $cnj e4 = -e4$   $cnj e5 = -e5$   $cnj e6 = -e6$   $cnj e7 = -e7$   
 $\langle proof \rangle$

**lemma** *cnj-eq-of-real*:  $cnj q = octo-of-real x \iff q = octo-of-real x$   
 $\langle proof \rangle$

**lemma** *octo-trad-cnj*:  $cnj q = (Ree q) *_R e0 - (Im1 q) *_R e1 - (Im2 q) *_R e2$   
 $- (Im3 q) *_R e3 - (Im4 q) *_R e4 - (Im5 q) *_R e5 - (Im6 q) *_R e6 - (Im7 q) *_R e7$  for  $q :: octo$   
 $\langle proof \rangle$

**lemma** *octonion-conjugate-property*:

$cnj x = -(1/6) *_R (x + (e1 * x) * e1 + (e2 * x) * e2 + (e3 * x) * e3 +$   
 $(e4 * x) * e4 + (e5 * x) * e5 + (e6 * x) * e6 + (e7 * x) * e7)$   
 $\langle proof \rangle$

**lemma** *octo-add-cnj*:  $q + cnj q = 2 *_R (Ree q) *_R e0$   $cnj q + q = 2 *_R (Ree q) *_R e0$   
 $\langle proof \rangle$

**lemma** *octo-add-cnj1*:  $q + cnj q = octo-of-real (2 *_R (Ree q))$   
 $cnj q + q = octo-of-real (2 *_R (Ree q))$   
 $\langle proof \rangle$

**lemma** *octo-subtract-cnj*:

$$q - \text{cnj } q = 2 *_{\text{R}} (\text{Im}1 \text{ } q *_{\text{R}} e1 + \text{Im}2 \text{ } q *_{\text{R}} e2 + \text{Im}3 \text{ } q *_{\text{R}} e3 + \\ \text{Im}4 \text{ } q *_{\text{R}} e4 + \text{Im}5 \text{ } q *_{\text{R}} e5 + \text{Im}6 \text{ } q *_{\text{R}} e6 + \text{Im}7 \text{ } q *_{\text{R}} e7)$$

*<proof>*

**lemma** *octo-mult-cnj-commute*:  $\text{cnj } x * x = x * \text{cnj } x$

*<proof>*

**lemma** *octo-cnj-mult-conv-norm*:  $\text{cnj } x * x = \text{octo-of-real } (\text{norm } x) \wedge 2$

*<proof>*

**lemma** *octo-mult-cnj-conv-norm*:  $x * \text{cnj } x = \text{octo-of-real } (\text{norm } x) \wedge 2$

*<proof>*

**lemma** *octo-mult-cnj-conv-norm-aux*:  $\text{octo-of-real } (\text{norm } x \wedge 2) = x * \text{cnj } x$

*<proof>*

**lemma** *octo-norm-conj*:  $\text{octo-of-real } (\text{inner } x \text{ } y) = (1/2) *_{\text{R}} (x * (\text{cnj } y) + y * (\text{cnj } x))$

*<proof>*

**lemma** *octo-inverse-cnj*:  $\text{inverse } x = \text{cnj } x /_{\text{R}} (\text{norm } x \wedge 2)$

*<proof>*

**lemma** *inverse-octo-1*:  $x \neq 0 \implies x * \text{inverse } x = (1 :: \text{octo})$

*<proof>*

**lemma** *inverse-octo-1-sym*:  $x \neq 0 \implies \text{inverse } x * x = (1 :: \text{octo})$

*<proof>*

**lemma** *inverse-0-octo [simp]*:  $\text{inverse } 0 = (0 :: \text{octo})$

*<proof>*

**lemma** *inverse-octo-commutes*:  $\text{inverse } x * x = x * (\text{inverse } x :: \text{octo})$

*<proof>*

**lemma** *octo-inverse-mult*:  $\text{inverse } (x * y) = \text{inverse } y * \text{inverse } x$  **for**  $x \text{ } y :: \text{octo}$

*<proof>*

**lemma** *octo-inverse-eq-cnj*:  $\text{norm } q = 1 \implies \text{inverse } q = \text{cnj } q$  **for**  $q :: \text{octo}$

*<proof>*

**lemma** *octo-in-Reals-if-Re*: **fixes**  $q :: \text{real}$  **shows**  $\text{Ree}(\text{octo-of-real}(q)) = q$

*<proof>*

**lemma** *octo-in-Reals-if-Re-con*: **assumes**  $\text{Ree}(\text{octo-of-real } q) = q$

**shows**  $q \in \text{Reals}$

*<proof>*

**lemma** *octo-in-Reals-if-cnj*: **fixes**  $q::\text{real}$  **shows**  $\text{cnj}(\text{octo-of-real}(q)) = \text{octo-of-real } q$

*<proof>*

**lemma** *octo-in-Reals-if-cnj-con*: **assumes**  $\text{cnj}(\text{octo-of-real}(q)) = \text{octo-of-real } q$   
**shows**  $q \in \text{Reals}$

*<proof>*

**lemma** *norm-power2*:  $\text{norm } q \wedge 2 = \text{Ree}(\text{cnj } q * q)$

*<proof>*

**lemma** *norm-power2-cnj*:  $\text{norm } q \wedge 2 = \text{Ree}(q * \text{cnj } q)$

*<proof>*

**lemma** *octo-norm-imaginary*:  $\text{Ree } x = 0 \implies x * x = -(\text{octo-of-real}(\text{norm } x))^2$

*<proof>*

## 2.8 Linearity and continuity of the components.

**lemma** *bounded-linear-Ree*: *bounded-linear Ree*

**and** *bounded-linear-Im1*: *bounded-linear Im1*

**and** *bounded-linear-Im2*: *bounded-linear Im2*

**and** *bounded-linear-Im3*: *bounded-linear Im3*

**and** *bounded-linear-Im4*: *bounded-linear Im4*

**and** *bounded-linear-Im5*: *bounded-linear Im5*

**and** *bounded-linear-Im6*: *bounded-linear Im6*

**and** *bounded-linear-Im7*: *bounded-linear Im7*

*<proof>*

**lemmas** *Cauchy-Ree* = *bounded-linear.Cauchy [OF bounded-linear-Ree]*

**lemmas** *Cauchy-Im1* = *bounded-linear.Cauchy [OF bounded-linear-Im1]*

**lemmas** *Cauchy-Im2* = *bounded-linear.Cauchy [OF bounded-linear-Im2]*

**lemmas** *Cauchy-Im3* = *bounded-linear.Cauchy [OF bounded-linear-Im3]*

**lemmas** *Cauchy-Im4* = *bounded-linear.Cauchy [OF bounded-linear-Im4]*

**lemmas** *Cauchy-Im5* = *bounded-linear.Cauchy [OF bounded-linear-Im5]*

**lemmas** *Cauchy-Im6* = *bounded-linear.Cauchy [OF bounded-linear-Im6]*

**lemmas** *Cauchy-Im7* = *bounded-linear.Cauchy [OF bounded-linear-Im7]*

**lemmas** *tendsto-Re* [*tendsto-intros*] = *bounded-linear.tendsto [OF bounded-linear-Ree]*

**lemmas** *tendsto-Im1* [*tendsto-intros*] = *bounded-linear.tendsto [OF bounded-linear-Im1]*

**lemmas** *tendsto-Im2* [*tendsto-intros*] = *bounded-linear.tendsto [OF bounded-linear-Im2]*

**lemmas** *tendsto-Im3* [*tendsto-intros*] = *bounded-linear.tendsto [OF bounded-linear-Im3]*

**lemmas** *tendsto-Im4* [*tendsto-intros*] = *bounded-linear.tendsto [OF bounded-linear-Im4]*

**lemmas** *tendsto-Im5* [*tendsto-intros*] = *bounded-linear.tendsto [OF bounded-linear-Im5]*

**lemmas** *tendsto-Im6* [*tendsto-intros*] = *bounded-linear.tendsto [OF bounded-linear-Im6]*

**lemmas** *tendsto-Im7* [*tendsto-intros*] = *bounded-linear.tendsto [OF bounded-linear-Im7]*

**lemmas** *isCont-Ree* [*simp*] = *bounded-linear.isCont [OF bounded-linear-Ree]*

**lemmas** *isCont-Im1* [*simp*] = *bounded-linear.isCont [OF bounded-linear-Im1]*



**lemmas** *isCont-Im2* [simp] = *bounded-linear.isCont* [OF *bounded-linear-Im2*]  
**lemmas** *isCont-Im3* [simp] = *bounded-linear.isCont* [OF *bounded-linear-Im3*]  
**lemmas** *isCont-Im4* [simp] = *bounded-linear.isCont* [OF *bounded-linear-Im4*]  
**lemmas** *isCont-Im5* [simp] = *bounded-linear.isCont* [OF *bounded-linear-Im5*]  
**lemmas** *isCont-Im6* [simp] = *bounded-linear.isCont* [OF *bounded-linear-Im6*]  
**lemmas** *isCont-Im7* [simp] = *bounded-linear.isCont* [OF *bounded-linear-Im7*]

**lemmas** *continuous-Ree* [simp] = *bounded-linear.continuous* [OF *bounded-linear-Ree*]  
**lemmas** *continuous-Im1* [simp] = *bounded-linear.continuous* [OF *bounded-linear-Im1*]  
**lemmas** *continuous-Im2* [simp] = *bounded-linear.continuous* [OF *bounded-linear-Im2*]  
**lemmas** *continuous-Im3* [simp] = *bounded-linear.continuous* [OF *bounded-linear-Im3*]  
**lemmas** *continuous-Im4* [simp] = *bounded-linear.continuous* [OF *bounded-linear-Im4*]  
**lemmas** *continuous-Im5* [simp] = *bounded-linear.continuous* [OF *bounded-linear-Im5*]  
**lemmas** *continuous-Im6* [simp] = *bounded-linear.continuous* [OF *bounded-linear-Im6*]  
**lemmas** *continuous-Im7* [simp] = *bounded-linear.continuous* [OF *bounded-linear-Im7*]

**lemmas** *continuous-on-Ree* [continuous-intros] = *bounded-linear.continuous-on* [OF *bounded-linear-Ree*]  
**lemmas** *continuous-on-Im1* [continuous-intros] = *bounded-linear.continuous-on* [OF *bounded-linear-Im1*]  
**lemmas** *continuous-on-Im2* [continuous-intros] = *bounded-linear.continuous-on* [OF *bounded-linear-Im2*]  
**lemmas** *continuous-on-Im3* [continuous-intros] = *bounded-linear.continuous-on* [OF *bounded-linear-Im3*]  
**lemmas** *continuous-on-Im4* [continuous-intros] = *bounded-linear.continuous-on* [OF *bounded-linear-Im4*]  
**lemmas** *continuous-on-Im5* [continuous-intros] = *bounded-linear.continuous-on* [OF *bounded-linear-Im5*]  
**lemmas** *continuous-on-Im6* [continuous-intros] = *bounded-linear.continuous-on* [OF *bounded-linear-Im6*]  
**lemmas** *continuous-on-Im7* [continuous-intros] = *bounded-linear.continuous-on* [OF *bounded-linear-Im7*]

**lemmas** *has-derivative-Ree* [derivative-intros] = *bounded-linear.has-derivative* [OF *bounded-linear-Ree*]  
**lemmas** *has-derivative-Im1* [derivative-intros] = *bounded-linear.has-derivative* [OF *bounded-linear-Im1*]  
**lemmas** *has-derivative-Im2* [derivative-intros] = *bounded-linear.has-derivative* [OF *bounded-linear-Im2*]  
**lemmas** *has-derivative-Im3* [derivative-intros] = *bounded-linear.has-derivative* [OF *bounded-linear-Im3*]  
**lemmas** *has-derivative-Im4* [derivative-intros] = *bounded-linear.has-derivative* [OF *bounded-linear-Im4*]  
**lemmas** *has-derivative-Im5* [derivative-intros] = *bounded-linear.has-derivative* [OF *bounded-linear-Im5*]  
**lemmas** *has-derivative-Im6* [derivative-intros] = *bounded-linear.has-derivative* [OF *bounded-linear-Im6*]  
**lemmas** *has-derivative-Im7* [derivative-intros] = *bounded-linear.has-derivative* [OF *bounded-linear-Im7*]

**lemmas** *sums-Ree* = *bounded-linear.sums* [OF *bounded-linear-Ree*]  
**lemmas** *sums-Im1* = *bounded-linear.sums* [OF *bounded-linear-Im1*]  
**lemmas** *sums-Im2* = *bounded-linear.sums* [OF *bounded-linear-Im2*]  
**lemmas** *sums-Im3* = *bounded-linear.sums* [OF *bounded-linear-Im3*]  
**lemmas** *sums-Im4* = *bounded-linear.sums* [OF *bounded-linear-Im4*]  
**lemmas** *sums-Im5* = *bounded-linear.sums* [OF *bounded-linear-Im5*]  
**lemmas** *sums-Im6* = *bounded-linear.sums* [OF *bounded-linear-Im6*]  
**lemmas** *sums-Im7* = *bounded-linear.sums* [OF *bounded-linear-Im7*]

### 2.8.1 Octonionic-specific theorems about sums.

**lemma** *Ree-sum* [simp]:  $\text{Ree} (\text{sum } f \ S) = \text{sum} (\lambda x. \ \text{Ree}(f \ x)) \ S$   
**and** *Im1-sum* [simp]:  $\text{Im1} (\text{sum } f \ S) = \text{sum} (\lambda x. \ \text{Im1} (f \ x)) \ S$   
**and** *Im2-sum* [simp]:  $\text{Im2} (\text{sum } f \ S) = \text{sum} (\lambda x. \ \text{Im2} (f \ x)) \ S$   
**and** *Im3-sum* [simp]:  $\text{Im3} (\text{sum } f \ S) = \text{sum} (\lambda x. \ \text{Im3} (f \ x)) \ S$   
**and** *Im4-sum* [simp]:  $\text{Im4} (\text{sum } f \ S) = \text{sum} (\lambda x. \ \text{Im4} (f \ x)) \ S$   
**and** *Im5-sum* [simp]:  $\text{Im5} (\text{sum } f \ S) = \text{sum} (\lambda x. \ \text{Im5} (f \ x)) \ S$   
**and** *Im6-sum* [simp]:  $\text{Im6} (\text{sum } f \ S) = \text{sum} (\lambda x. \ \text{Im6} (f \ x)) \ S$   
**and** *Im7-sum* [simp]:  $\text{Im7} (\text{sum } f \ S) = \text{sum} (\lambda x. \ \text{Im7} (f \ x)) \ S$   
 <proof>

### 2.8.2 Bound results for real and imaginary components of limits.

**lemma** *Ree-tendsto-upperbound*:

$\llbracket (f \longrightarrow \text{limit}) \ \text{net}; \ \forall_F \ x \ \text{in} \ \text{net}. \ \text{octo.Ree} (f \ x) \leq b; \ \text{net} \neq \text{bot} \rrbracket \implies \text{Ree} \ \text{limit} \leq b$   
 <proof>

**lemma** *Im1-tendsto-upperbound*:

$\llbracket (f \longrightarrow \text{limit}) \ \text{net}; \ \forall_F \ x \ \text{in} \ \text{net}. \ \text{Im1} (f \ x) \leq b; \ \text{net} \neq \text{bot} \rrbracket \implies \text{Im1} \ \text{limit} \leq b$   
 <proof>

**lemma** *Im2-tendsto-upperbound*:

$\llbracket (f \longrightarrow \text{limit}) \ \text{net}; \ \forall_F \ x \ \text{in} \ \text{net}. \ \text{Im2} (f \ x) \leq b; \ \text{net} \neq \text{bot} \rrbracket \implies \text{Im2} \ \text{limit} \leq b$   
 <proof>

**lemma** *Im3-tendsto-upperbound*:

$\llbracket (f \longrightarrow \text{limit}) \ \text{net}; \ \forall_F \ x \ \text{in} \ \text{net}. \ \text{Im3} (f \ x) \leq b; \ \text{net} \neq \text{bot} \rrbracket \implies \text{Im3} \ \text{limit} \leq b$   
 <proof>

**lemma** *Im4-tendsto-upperbound*:

$\llbracket (f \longrightarrow \text{limit}) \ \text{net}; \ \forall_F \ x \ \text{in} \ \text{net}. \ \text{Im4} (f \ x) \leq b; \ \text{net} \neq \text{bot} \rrbracket \implies \text{Im4} \ \text{limit} \leq b$   
 <proof>

**lemma** *Im5-tendsto-upperbound*:

$\llbracket (f \longrightarrow \text{limit}) \ \text{net}; \ \forall_F \ x \ \text{in} \ \text{net}. \ \text{Im5} (f \ x) \leq b; \ \text{net} \neq \text{bot} \rrbracket \implies \text{Im5} \ \text{limit} \leq b$   
 <proof>

**lemma** *Im6-tendsto-upperbound*:

$\llbracket (f \longrightarrow \text{limit}) \text{ net}; \forall_F x \text{ in net. Im6 } (f x) \leq b; \text{ net} \neq \text{bot} \rrbracket \Longrightarrow \text{Im6 limit} \leq b$   
 ⟨proof⟩

**lemma** *Im7-tendsto-upperbound*:

$\llbracket (f \longrightarrow \text{limit}) \text{ net}; \forall_F x \text{ in net. Im7 } (f x) \leq b; \text{ net} \neq \text{bot} \rrbracket \Longrightarrow \text{Im7 limit} \leq b$   
 ⟨proof⟩

**lemma** *Ree-tendsto-lowerbound*:

$\llbracket (f \longrightarrow \text{limit}) \text{ net}; \forall_F x \text{ in net. } b \leq \text{octo.Ree } (f x); \text{ net} \neq \text{bot} \rrbracket \Longrightarrow b \leq \text{Ree limit}$   
 ⟨proof⟩

**lemma** *Im1-tendsto-lowerbound*:

$\llbracket (f \longrightarrow \text{limit}) \text{ net}; \forall_F x \text{ in net. } b \leq \text{Im1 } (f x); \text{ net} \neq \text{bot} \rrbracket \Longrightarrow b \leq \text{Im1 limit}$   
 ⟨proof⟩

**lemma** *Im2-tendsto-lowerbound*:

$\llbracket (f \longrightarrow \text{limit}) \text{ net}; \forall_F x \text{ in net. } b \leq \text{Im2 } (f x); \text{ net} \neq \text{bot} \rrbracket \Longrightarrow b \leq \text{Im2 limit}$   
 ⟨proof⟩

**lemma** *Im3-tendsto-lowerbound*:

$\llbracket (f \longrightarrow \text{limit}) \text{ net}; \forall_F x \text{ in net. } b \leq \text{Im3 } (f x); \text{ net} \neq \text{bot} \rrbracket \Longrightarrow b \leq \text{Im3 limit}$   
 ⟨proof⟩

**lemma** *Im4-tendsto-lowerbound*:

$\llbracket (f \longrightarrow \text{limit}) \text{ net}; \forall_F x \text{ in net. } b \leq \text{Im4 } (f x); \text{ net} \neq \text{bot} \rrbracket \Longrightarrow b \leq \text{Im4 limit}$   
 ⟨proof⟩

**lemma** *Im5-tendsto-lowerbound*:

$\llbracket (f \longrightarrow \text{limit}) \text{ net}; \forall_F x \text{ in net. } b \leq \text{Im5 } (f x); \text{ net} \neq \text{bot} \rrbracket \Longrightarrow b \leq \text{Im5 limit}$   
 ⟨proof⟩

**lemma** *Im6-tendsto-lowerbound*:

$\llbracket (f \longrightarrow \text{limit}) \text{ net}; \forall_F x \text{ in net. } b \leq \text{Im6 } (f x); \text{ net} \neq \text{bot} \rrbracket \Longrightarrow b \leq \text{Im6 limit}$   
 ⟨proof⟩

**lemma** *Im7-tendsto-lowerbound*:

$\llbracket (f \longrightarrow \text{limit}) \text{ net}; \forall_F x \text{ in net. } b \leq \text{Im7 } (f x); \text{ net} \neq \text{bot} \rrbracket \Longrightarrow b \leq \text{Im7 limit}$   
 ⟨proof⟩

**lemma** *octo-of-real-continuous* [continuous-intros]:

$\text{continuous net } f \Longrightarrow \text{continuous net } (\lambda x. \text{octo-of-real } (f x))$   
 ⟨proof⟩

**lemma** *octo-of-real-continuous-on* [continuous-intros]:

$\text{continuous-on } S f \Longrightarrow \text{continuous-on } S (\lambda x. \text{octo-of-real } (f x))$   
 ⟨proof⟩

**lemma** *of-real-continuous-iff*:  $\text{continuous net } (\lambda x. \text{octo-of-real } (f x)) \longleftrightarrow \text{contin-}$

uous net  $f$   
 $\langle proof \rangle$

**lemma** *of-real-continuous-on-iff*:  
 $continuous-on\ S\ (\lambda x. octo-of-real(f\ x)) \longleftrightarrow continuous-on\ S\ f$   
 $\langle proof \rangle$

## 2.9 Octonions for describing 7D isometries

### 2.9.1 The $HIm$ operator

**definition**  $HIm :: octo \Rightarrow real^7$  **where**  
 $HIm\ q \equiv vector[Im1\ q, Im2\ q, Im3\ q, Im4\ q, Im5\ q, Im6\ q, Im7\ q]$

**lemma** *HIm-Octo*:  $HIm\ (Octo\ w\ x\ y\ z\ u\ v\ q\ g) = vector[x,y,z, u, v, q, g]$   
 $\langle proof \rangle$

**lemma** *him-eq*:  $HIm\ a = HIm\ b \longleftrightarrow Im1\ a = Im1\ b \wedge Im2\ a = Im2\ b \wedge Im3\ a = Im3\ b$   
 $\wedge Im4\ a = Im4\ b \wedge Im5\ a = Im5\ b \wedge Im6\ a = Im6\ b \wedge Im7\ a = Im7\ b$   
 $\langle proof \rangle$

**lemma** *him-of-real [simp]*:  $HIm(octo-of-real\ a) = 0$   
 $\langle proof \rangle$

**lemma** *him-0 [simp]*:  $HIm\ 0 = 0$   
 $\langle proof \rangle$

**lemma** *him-1 [simp]*:  $HIm\ 1 = 0$   
 $\langle proof \rangle$

**lemma** *him-cnj*:  $HIm(cnj\ q) = -\ HIm\ q$   
 $\langle proof \rangle$

**lemma** *him-mult-left [simp]*:  $HIm\ (a\ *_R\ q) = a\ *_R\ HIm\ q$   
 $\langle proof \rangle$

**lemma** *him-mult-right [simp]*:  $HIm\ (q\ *\ octo-of-real\ a) = HIm\ q\ *\ of-real\ a$   
 $\langle proof \rangle$

**lemma** *him-add [simp]*:  $HIm\ (x + y) = HIm\ x + HIm\ y$   
**and** *him-minus [simp]*:  $HIm\ (-x) = -\ HIm\ x$   
**and** *him-diff [simp]*:  $HIm\ (x - y) = HIm\ x - HIm\ y$   
 $\langle proof \rangle$

**lemma** *him-sum [simp]*:  $HIm\ (sum\ f\ S) = (\sum\ x \in S. HIm\ (f\ x))$   
 $\langle proof \rangle$

**lemma** *linear-him*: *linear HIm*

*<proof>*

## 2.9.2 The $Hv$ operator

**definition**  $Hv :: \text{real}^7 \Rightarrow \text{octo}$  where

$$Hv\ v \equiv \text{Octo } 0\ (v\$1)\ (v\$2)\ (v\$3)\ (v\$4)\ (v\$5)\ (v\$6)\ (v\$7)$$

**lemma**  $Hv\text{-sel}$  [*simp*]:

$$\begin{aligned} \text{Ree } (Hv\ v) &= 0 & \text{Im1 } (Hv\ v) &= v\ \$\ 1 & \text{Im2 } (Hv\ v) &= v\ \$\ 2 & \text{Im3 } (Hv\ v) &= v\ \$\ 3 \\ \text{Im4 } (Hv\ v) &= v\ \$\ 4 & \text{Im5 } (Hv\ v) &= v\ \$\ 5 & \text{Im6 } (Hv\ v) &= v\ \$\ 6 & \text{Im7 } (Hv\ v) &= v\ \$\ 7 \end{aligned}$$

*<proof>*

**lemma**  $hv\text{-vec}$ :  $Hv(\text{vec } r) = \text{Octo } 0\ r\ r\ r\ r\ r\ r\ r$

*<proof>*

**lemma**  $hv\text{-eq-zero}$  [*simp*]:  $Hv\ v = 0 \longleftrightarrow v = 0$

*<proof>*

**lemma**  $hv\text{-zero}$  [*simp*]:  $Hv\ 0 = 0$

*<proof>*

**lemma**  $hv\text{-vector}$  [*simp*]:  $Hv(\text{vector}[x,y,z,u,v,q,g]) = \text{Octo } 0\ x\ y\ z\ u\ v\ q\ g$

*<proof>*

**lemma**  $hv\text{-basis}$ :

$$\begin{aligned} Hv(\text{axis } 1\ 1) &= e1 & Hv(\text{axis } 2\ 1) &= e2 & Hv(\text{axis } 3\ 1) &= e3 \\ Hv(\text{axis } 4\ 1) &= e4 & Hv(\text{axis } 5\ 1) &= e5 & Hv(\text{axis } 6\ 1) &= e6 & Hv(\text{axis } 7\ 1) &= e7 \end{aligned}$$

*<proof>*

**lemma**  $hv\text{-add}$  [*simp*]:  $Hv(x + y) = Hv\ x + Hv\ y$

*<proof>*

**lemma**  $hv\text{-minus}$  [*simp*]:  $Hv(-x) = -Hv\ x$

*<proof>*

**lemma**  $hv\text{-diff}$  [*simp*]:  $Hv(x - y) = Hv\ x - Hv\ y$

*<proof>*

**lemma**  $hv\text{-cmult}$  [*simp*]:

$$Hv(\text{scaleR } a\ x) = \text{scaleR } a\ (Hv\ x)$$

*<proof>*

**lemma**  $hv\text{-sum}$  [*simp*]:  $Hv(\text{sum } f\ S) = (\sum x \in S. Hv\ (f\ x))$

*<proof>*

**lemma**  $hv\text{-inj}$ :  $Hv\ x = Hv\ y \longleftrightarrow x = y$

*<proof>*

**lemma**  $linear\text{-hv}$ :  $linear\ Hv$

*<proof>*

**lemma** *him-hv* [*simp*]:  $HIm(Hv\ x) = x$   
*<proof>*

**lemma** *cnj-hv* [*simp*]:  $cnj(Hv\ v) = -Hv\ v$   
*<proof>*

**lemma** *hv-him*:  $Hv(HIm\ q) = Octo\ 0\ (Im1\ q)\ (Im2\ q)\ (Im3\ q)\ (Im4\ q)\ (Im5\ q)\ (Im6\ q)\ (Im7\ q)$   
*<proof>*

**lemma** *hv-him-eq*:  $Hv(HIm\ q) = q \iff Re\ q = 0$   
*<proof>*

**lemma** *dot-hv* [*simp*]:  $Hv\ u \cdot Hv\ v = u \cdot v$   
*<proof>*

**lemma** *norm-hv* [*simp*]:  $norm\ (Hv\ v) = norm\ v$   
*<proof>*

### 2.9.3 Related basic identities

**lemma** *mult-hv-eq-cross-dot*:  $Hv\ x * Hv\ y = Hv(x \times_7 y) - octo-of-real\ (inner\ x\ y)$   
*<proof>*

**lemma** *octonion-identity1-cross7* :  
 $Hv\ (x \times_7 y) = (1/2) *R\ (Hv\ x * Hv\ y - Hv\ y * Hv\ x)$   
*<proof>*

**lemma** *octonion-identity2-cross7*:  
 $Hv\ (x \times_7 (y \times_7 z) + y \times_7 (z \times_7 x) + z \times_7 (x \times_7 y)) =$   
 $-(3/2) *R\ ((Hv\ x * Hv\ y) * Hv\ z - Hv\ x * (Hv\ y * Hv\ z))$   
*<proof>*

## 2.10 Representing orthogonal transformations as conjugation or congruence with an octonion.

**lemma** *HIm-nth* [*simp*]:  
 $HIm\ x\ \$\ 1 = Im1\ x\ HIm\ x\ \$\ 2 = Im2\ x\ HIm\ x\ \$\ 3 = Im3\ x\ HIm\ x\ \$\ 4 = Im4\ x$   
 $HIm\ x\ \$\ 5 = Im5\ x\ HIm\ x\ \$\ 6 = Im6\ x\ HIm\ x\ \$\ 7 = Im7\ x$   
*<proof>*

**lemma** *orthogonal-transformation-octo-congruence*:  
**assumes**  $norm\ q = 1$   
**shows** *orthogonal-transformation*  $(\lambda x. HIm(cn\ j\ q * Hv\ x * q))$   
*<proof>*

**lemma** *orthogonal-transformation-octo-conjugation:*  
**assumes**  $q \neq 0$   
**shows** *orthogonal-transformation* ( $\lambda x.$   $HIm$  (*inverse*  $q * Hv\ x * q$ ))  
*<proof>*

**unbundle** *no-cross7-syntax*

**bundle** *octonion-syntax*  
**begin**

**notation** *octo-e0* ( $e0$ )  
**notation** *octo-e1* ( $e1$ )  
**notation** *octo-e2* ( $e2$ )  
**notation** *octo-e3* ( $e3$ )  
**notation** *octo-e4* ( $e4$ )  
**notation** *octo-e5* ( $e5$ )  
**notation** *octo-e6* ( $e6$ )  
**notation** *octo-e7* ( $e7$ )

**end**

**bundle** *no-octonion-syntax*  
**begin**

**no-notation** *octo-e0* ( $e0$ )  
**no-notation** *octo-e1* ( $e1$ )  
**no-notation** *octo-e2* ( $e2$ )  
**no-notation** *octo-e3* ( $e3$ )  
**no-notation** *octo-e4* ( $e4$ )  
**no-notation** *octo-e5* ( $e5$ )  
**no-notation** *octo-e6* ( $e6$ )  
**no-notation** *octo-e7* ( $e7$ )

**end**

**unbundle** *no-octonion-syntax*  
**hide-const** (**open**) *Octonions.cnj*

**end**

## References

- [1] J. C. Baez. The octonions. *Bull. Amer. Math. Soc.*, 39:145–205, 2002.
- [2] H.-D. E. et al. (eds.). *Numbers*. Springer, New York, 1991.
- [3] P. Lounesto. *Clifford Algebras and Spinors*. Cambridge University Press, 2 edition, 2001.