A Fortran-90 Implementation of Rotation Representations

Marc De Graef

08/18/14

1 Introduction

This document describes the rotations.f90 module, a module containing functions and subroutines for handling rotations and transformations between rotations in seven different representations/parameterizations:

- Euler angles (eu);
- Rotation/orientation matrix (om);
- Axis-angle pair (ax);
- Rodrigues-Frank vector (ro);
- Quaternion (qu);
- Homochoric vector (ho);
- Cubochoric vector (cu).

The two letter abbreviation after each representation is the short-hand string that we will use throughout this document to describe rotations.

While rotations are an old, well-understood subject, a number of potential pitfalls arise when one attempts to implement all the above representations and the transformations between them. The pitfalls mainly occur because there are multiple occasions where the user has to make a sign choice: left-handed vs. right-handed; positive vs. negative rotation angles; active vs. passive interpretation. These sign choices are highlighted in this document as ***Conventions***.

In this document we present a consistent approach for the numerical implementation of all seven rotation representations. The document is accompanied by a number of fortran-90 modules that have been extensively tested and validated. Implementation in other programming languages should not pose any significant problems, although one must be careful with the following potential issues:

1. Fortran-90 is a column-major language when it comes to storage of multidimensional arrays; implementations in C/C++, Mathematica, Pascal, and Python, to name just a few, are row-major languages, so care must be taken when converting routines/functions that employ orientation matrices.

2. Fortran-90 allows for the explicit use of the number *infinity* in value assignments and logical comparisons; this occurs mostly in the Rodrigues-Frank representation for rotation angles that equal π. Conversion to languages that do not support this IEEE floating point standard will require careful rewriting of those parts of the module.

We begin this document with a number of definitions and conventions upon which the entire implementation is based. Then we describe all the transformation routines between the seven representations, followed by a description of the testing routines. The text is somewhat verbose and explicit in all its definitions; experience has shown that it is not a bad idea to spell out every single detail.

2 Definitions and Conventions

2.1 Reference frames

Convention 1: All non-crystallographic reference frames are right-handed orthonormal (cartesian) frames of the form $\mathbf{e}_i$ $(i = 1 \ldots 3)$, such that $\mathbf{e}_1 \cdot (\mathbf{e}_2 \times \mathbf{e}_3) > 0$.

Amongst the possible cartesian reference frames, there is one that has a special relation to the crystallographic reference frames, which are described in the following subsections. The special cartesian frame is known as the *default cartesian frame* and is described in subsection 2.1.3.

2.1.1 Crystallographic reference frames: direct space

A Bravais lattice is described by means of three basis vectors $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$, or $\mathbf{a}_i$, $i = 1 \ldots 3$. The basis vectors form a right handed reference frame, i.e., the mixed product $\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c})$ is positive. The lengths of the basis vectors are represented by a_i or (a, b, c); the angles between vectors are denoted by (α, β, γ), with α the angle between $\mathbf{b}$ and $\mathbf{c}$, β between $\mathbf{c}$ and $\mathbf{a}$, and γ between $\mathbf{a}$ and $\mathbf{b}$.

Crystallographic quantities in direct space (distances, angles) are often expressed in terms of the direct metric tensor, g_{ij}, which is defined in terms of the basis vector dot products:

$$g_{ij} \equiv \mathbf{a}_i \cdot \mathbf{a}_j = |\mathbf{a}_i|\,|\mathbf{a}_j| \cos\omega_{ij}.$$

The determinant of the metric tensor is equal to the square of the unit cell volume, $V^2 = \det|g_{ij}|$.

2.1.2 Crystallographic reference frames: reciprocal space

The dual or reciprocal lattice is defined by the basis vectors $\mathbf{a}^*$, $\mathbf{b}^*$, and $\mathbf{c}^*$, or $\mathbf{a}_i^*$. The reciprocal basis vectors are defined by the relation:

$$\mathbf{a}_i \cdot \mathbf{a}_j^* = \delta_{ij},$$

where δ_{ij} is the identity matrix. One can show that the following relations hold:

$$\mathbf{a}^* = \frac{\mathbf{b} \times \mathbf{c}}{V};$$
$$\mathbf{b}^* = \frac{\mathbf{c} \times \mathbf{a}}{V};$$
$$\mathbf{c}^* = \frac{\mathbf{a} \times \mathbf{b}}{V}.$$

The reciprocal metric tensor is defined as:

$$g_{ij}^* \equiv \mathbf{a}_i^* \cdot \mathbf{a}_j^*.$$

One can show that $g_{ij}^* = g_{ij}^{-1}$, so that $V^* = V^{-1}$.

2.1.3 Crystallographic reference frames: default cartesian frame

The direct and reciprocal crystallographic basis vectors do not, in general, form orthonormal reference frames. For computational work, it is often desirable to express crystallographic quantities with respect to a standard orthonormal (cartesian) reference frame. According to the International Tables for Crystallography, such a reference frame can be defined by taking $\mathbf{e}_x \parallel \mathbf{a}$, $\mathbf{e}_z \parallel \mathbf{c}^*$, and completing the right handed frame with $\mathbf{e}_y = \mathbf{e}_z \times \mathbf{e}_x$:

$$\begin{aligned}\mathbf{e}_x &= \frac{\mathbf{a}}{a};\\ \mathbf{e}_y &= \mathbf{e}_z \times \mathbf{e}_x;\\ \mathbf{e}_z &= \frac{\mathbf{c}^*}{c^*}.\end{aligned}$$

Direct space vectors $\mathbf{r}$ with components r_i expressed in terms of the Bravais basis vectors can be transformed into the cartesian basis by means of the direct structure matrix a_{ij}:

$$x_i = a_{ij} r_j,$$

where x_i are the cartesian components, and

$$\begin{aligned} a_{ij} &= \begin{pmatrix} \sqrt{g_{11}} & \frac{g_{21}}{\sqrt{g_{11}}} & \frac{g_{31}}{\sqrt{g_{11}}} \\ 0 & V\sqrt{\frac{g^*_{33}}{g_{11}}} & -\frac{V g^*_{32}}{\sqrt{g^*_{33} g_{11}}} \\ 0 & 0 & \frac{1}{\sqrt{g^*_{33}}} \end{pmatrix}; \\ &= \begin{pmatrix} a & b\cos\gamma & c\cos\beta \\ 0 & b\sin\gamma & -\frac{c\mathcal{F}(\beta,\gamma,\alpha)}{\sin\gamma} \\ 0 & 0 & \frac{V}{ab\sin\gamma} \end{pmatrix}, \end{aligned} \tag{1}$$

where $\mathcal{F}(\alpha,\beta,\gamma) = \cos\alpha\cos\beta - \cos\gamma$.

The same matrix can be used to determine the reciprocal structure matrix, b_{ij}, which expresses the components k_i of a vector $\mathbf{k}$ in the reciprocal Bravais frame in terms of the same cartesian reference frame:

$$q_i = b_{ij} k_j.$$

One can show that the relation between direct and reciprocal structure matrices is given by $b^T = a^{-1}$, where T denotes the transpose. Furthermore, the product of the direct structure matrix and its transpose is equal to the direct metric tensor:

$$g_{ij} = a_{ik}(a^T)_{kj},$$

with a similar relation between the reciprocal metric tensor and the reciprocal structure matrix.

The cartesian reference frame is important because it is used to define the "standard crystallographic orientation", i.e., the identity rotation. In a typical electron back-scatter diffraction experiment, the cartesian basis vectors are taken to be parallel to the conventional (RD, TD, ND) sample reference frame for the identity rotation. It should be noted that not every EBSD OEM treats the reference frames in the same way.

2.2 Rotations

In this section we define a general rotation, followed by a description of commonly used rotation representations/parameterizations, as well as a brief description of the new cubochoric parameterization.

2.2.1 Definition

The turning of an object or a coordinate system by a given angle ω about a fixed point $\mathbf{d}$ is referred to as a *rotation*. Such an operation preserves the distances between the object points as well as the handedness of the object. Each rotation has an invariant line, represented by the unit vector $\hat{\mathbf{n}}$, which is known as the rotation axis; if the rotation is represented by the operator $\mathcal{R}$, then the action $\mathcal{R}(\hat{\mathbf{n}})$ produces $\hat{\mathbf{n}}$ again.

Consider a rotation characterized by $\mathbf{d}$, $\hat{\mathbf{n}}$, and ω. The rotation angle ω is taken to be positive for a counterclockwise rotation when looking from the end point of $\hat{\mathbf{n}}$ towards the fixed point $\mathbf{d}$. For simplicity, and without loss of generality, we will from here on take the fixed point to coincide with the origin of the reference frame, $\mathbf{d} = \mathbf{0}$.

Convention 2: A rotation angle ω is taken to be positive for a counterclockwise rotation when viewing from the end point of the axis unit vector $\hat{\mathbf{n}}$ towards the origin.

2.2.2 Active and passive interpretations

A rotation can be viewed as operating on the object, which is the *active* interpretation, or operating on the reference frame, which is the *passive* interpretation. An active rotation transforms object coordinates to new coordinates in the same reference frame; for the passive interpretation, the initial and final reference frames are different. This is illustrated for a rotation with invariant axis $\hat{\mathbf{n}}$ normal to the plane of the drawing in Fig. **??**.

Here we describe explicitly the difference between active and passive interpretations. Consider the "old" reference frame $\mathbf{e}_i$ along with the vector $\mathbf{q}$, as shown in Fig. **??**. This vector has components q_i with respect to the reference frame $\mathbf{e}_i$. The "new" reference frame is defined by the vectors $\mathbf{e}'_j$, and the vector $\mathbf{q}$ has components q'_j with respect to these vectors. The vector $\mathbf{q}$ can be written as:

$$\mathbf{q} = q_x\mathbf{e}_x + q_y\mathbf{e}_y + q_z\mathbf{e}_z.$$

When we take the dot product with $\mathbf{e}_x$, we obtain:

$$\mathbf{q}\cdot\mathbf{e}_x = q_x\mathbf{e}_x\cdot\mathbf{e}_x + q_y\mathbf{e}_y\cdot\mathbf{e}_x + q_z\mathbf{e}_z\cdot\mathbf{e}_x;$$

since the basis vectors are orthonormal, we have

$$\mathbf{q}\cdot\mathbf{e}_x = q_x,$$

and a similar expression for q_y and q_z. Therefore, we can write the vector $\mathbf{q}$ as:

$$\mathbf{q} = (\mathbf{q}\cdot\mathbf{e}_x)\mathbf{e}_x + (\mathbf{q}\cdot\mathbf{e}_y)\mathbf{e}_y + (\mathbf{q}\cdot\mathbf{e}_z)\mathbf{e}_z.$$

This is valid for every vector $\mathbf{q}$, so in particular it must be valid for $\mathbf{q} = \mathbf{e}'_x$:

$$\mathbf{e}'_x = (\mathbf{e}'_x\cdot\mathbf{e}_x)\mathbf{e}_x + (\mathbf{e}'_x\cdot\mathbf{e}_y)\mathbf{e}_y + (\mathbf{e}'_x\cdot\mathbf{e}_z)\mathbf{e}_z.$$

A similar relation is found for the other vectors $\mathbf{e}'_j$. Combining the equations into matrix form, we find:

$$\begin{pmatrix} \mathbf{e}'_x \\ \mathbf{e}'_y \\ \mathbf{e}'_z \end{pmatrix} = \begin{pmatrix} \mathbf{e}'_x \cdot \mathbf{e}_x & \mathbf{e}'_x \cdot \mathbf{e}_y & \mathbf{e}'_x \cdot \mathbf{e}_z \\ \mathbf{e}'_y \cdot \mathbf{e}_x & \mathbf{e}'_y \cdot \mathbf{e}_y & \mathbf{e}'_y \cdot \mathbf{e}_z \\ \mathbf{e}'_z \cdot \mathbf{e}_x & \mathbf{e}'_z \cdot \mathbf{e}_y & \mathbf{e}'_z \cdot \mathbf{e}_z \end{pmatrix} \begin{pmatrix} \mathbf{e}_x \\ \mathbf{e}_y \\ \mathbf{e}_z \end{pmatrix}.$$

We define the rotation matrix α_{ij} as:

$$\alpha_{ij} = \begin{pmatrix} \mathbf{e}'_x \cdot \mathbf{e}_x & \mathbf{e}'_x \cdot \mathbf{e}_y & \mathbf{e}'_x \cdot \mathbf{e}_z \\ \mathbf{e}'_y \cdot \mathbf{e}_x & \mathbf{e}'_y \cdot \mathbf{e}_y & \mathbf{e}'_y \cdot \mathbf{e}_z \\ \mathbf{e}'_z \cdot \mathbf{e}_x & \mathbf{e}'_z \cdot \mathbf{e}_y & \mathbf{e}'_z \cdot \mathbf{e}_z \end{pmatrix}.$$

and the transformation rule becomes:

$$\mathbf{e}'_i = (\mathbf{e}'_i \cdot \mathbf{e}_j)\mathbf{e}_j = \alpha_{ij}\mathbf{e}_j.$$

The matrix α_{ij} represents a passive rotation, since it transforms (rotates) the old reference frame to the new reference frame.

Next we consider an arbitrary vector $\mathbf{p}$, which can be written as $\mathbf{p} = p'_i\mathbf{e}'_i$. Using the transformation rule for basis vectors, we obtain:

$$\mathbf{p} = p'_i\alpha_{ij}\mathbf{e}_j = p_j\mathbf{e}_j$$

from which we derive:

$$\begin{aligned} p_j &= p'_i\alpha_{ij}; \\ p_j(\alpha^{-1})_{jk} &= p'_i\alpha_{ij}(\alpha^{-1})_{jk}; \\ p_j(\alpha^T)_{jk} &= p'_i\delta_{ik}; \\ \alpha_{kj}p_j &= p'_k. \end{aligned}$$

We have made use of the fact that the transpose of a rotation matrix is equal to the inverse of that matrix, and also of the fact that the transpose of a product is the product of the transposes in reverse order. The final result expresses the new vector components in terms of the old ones as:

$$p'_i = \alpha_{ij}p_j.$$

Since α_{ij} represents a passive rotation, the components p'_i are the components of the vector $\mathbf{p}$ in the new reference frame.

For an active rotation, we keep the reference frame stationary, and rotate in the opposite direction, i.e., we replace the rotation angle ω by $-\omega$, or, equivalently, we keep ω unchanged and replace $\hat{\mathbf{n}}$ by $-\hat{\mathbf{n}}$. This is equivalent with taking the transpose of the matrix α_{ij} in the transformation rule:

$$p'_i = \alpha_{ji}p_j = p_j\alpha_{ji} \qquad \text{(active)}$$

Note that the components p_j and p'_i are now with respect to the same reference frame.

Rotation matrices, such as α_{ij}, belong to the set of special orthonormal matrices, with the following properties:

- the determinant of a rotation matrix equals 1;
- the transpose of a rotation matrix is equal to the inverse matrix;
- the sum of the squares of the entries in each column/row equals 1.

This set of matrices occurs frequently in many branches of science and has received a special name: SO(3) (Special Orthonormal 3×3 matrices). One can show that SO(3) is isomorphic with the set of 3D rotations, hence SO(3) is a representation of the set of 3D rotations.

2.3 Rotation parameterizations

2.3.1 Euler angles

Euler has shown that any arbitrary 3D rotation can be decomposed into three successive rotations around the coordinate axes. Since there are three independent axes, denoted xyz for simplicity, there are several possibilities for the decomposition. Note that the rightmost symbol corresponds to the first rotation, the middle symbol refers to the rotated y axis, and the first symbol refers to the twice rotated x axis.

The name *Euler angles* is reserved for those decompositions for which two of the three axes symbols are equal, as in zxz or yzy. Hence, there are six possible Euler angle conventions. When the three axes are different, one refers to the angle triplet as *Tayt-Briant angles*; while they are used infrequently in materials science, they find frequent application in other fields, for instance in aviation (roll, pitch, and yaw motions of an airplane).

In texture analysis, the most commonly used definition of the Euler angles is the Bunge convention, which has an axis triplet zxz, with corresponding rotation angles $(\varphi_1, \Phi, \varphi_2)$ (applied from left to right).

Convention 3: Euler angle triplets $\boldsymbol{\theta} = (\varphi_1, \Phi, \varphi_2)$ are implemented using the Bunge convention, with the angular ranges as $\varphi_1 \in [0, 2\pi]$, $\Phi \in [0, \pi]$, and $\varphi_2 \in [0, 2\pi]$, and operate on the reference frame (passive interpretation).

To obtain the active interpretation, one must apply the rotations with opposite angles, in the opposite order, $(-\varphi_2, -\Phi, -\varphi_1)$. The range of the Euler angle triplet defined above results in a total volume in Euler space of $8\pi^2 = \iiint \sin\Phi \mathrm{d}\varphi_1 \mathrm{d}\Phi \mathrm{d}\varphi_2$.

2.3.2 Rotation matrices

Rotation matrices were defined in the previous section.

Convention 4: Rotation matrices convert the old reference frame into the new reference frame and are hence always regarded in the passive interpretation.

2.3.3 Axis angle pair

The axis angle pair representation, $(\hat{\mathbf{n}}, \omega)$ provides a convenient way of writing down a rotation in terms of the angle ω and the axis $\hat{\mathbf{n}}$, which is typically represented either by a crystallographic direction $[uvw]$ or by a set of direction cosines $[c_1 c_2 c_3]$ (which are essentially normalized direction indices). Note that for non-cubic crystal systems, the normalization must be carried out on direction indices that have been transformed to the cartesian reference frame by means of the direct structure matrix. Often, the axis angle pair is written in the form $\omega@\hat{\mathbf{n}}$.

Convention 5: The rotation angle ω is limited to the interval $[0, \pi]$.

For angles in the range $[\pi, 2\pi[$, the sign of the unit axis vector $\hat{\mathbf{n}}$ must be reversed, and ω replaced by $2\pi - \omega$. For angles outside the range $[0, 2\pi[$, the angle must first be reduced to the interval $[0, 2\pi[$ by adding or subtracting the appropriate integer multiple of 2π. The main reason for limiting ω to the interval $[0, \pi]$ is to allow for a seamless conversion to the Rodrigues-Frank vector.

2.3.4 Neo-Eulerian: Rodrigues-Frank vector

The term *neo-Eulerian* was coined by Frank and refers to rotation representations of the form $\hat{\mathbf{n}} f(\omega)$, where f is a suitably chosen function of the rotation angle. The Rodrigues-Frank vector

representation is obtained by setting $f(\omega) = \tan(\omega/2)$:

$$\boldsymbol{\rho} = \hat{\mathbf{n}} \tan \frac{\omega}{2}.$$

Since the rotation angle belongs to the interval $[0, \pi]$, all rotations around an axis $\hat{\mathbf{n}}$ are represented by points along the line formed by this unit vector, with a rotation by π represented by the point at infinity $(+\infty)$. Rotation angles in the range $[-\pi, 0]$ are equivalent to setting the unit vector equal to its opposite, $-\hat{\mathbf{n}}$, and using a positive angle. Restricting the rotation angle to $[0, \pi]$ has the added advantage that $\tan(\omega/2)$ is always positive and is thus equal to the length of the Rodrigues-Frank vector $\boldsymbol{\rho}$. Angles outside this range may cause a negative value for the tangent, which would conflict with the positive nature of vector norms. Note that in the source code, a Rodrigues-Frank vector is stored as a four-component vector, first the unit axis vector $\hat{\mathbf{n}}$, and the vectror length $\tan(\omega/2)$ in the fourth position. This is necessary, so that a 180° rotation around an arbitrary axis can be correctly represented.

Convention 6: The length of a Rodrigues-Frank vector $\boldsymbol{\rho}$ always results in a rotation angle $\omega = 2 \arctan(|\boldsymbol{\rho}|)$ in the range $[0, \pi]$.

While the infinite range of the Rodrigues space may pose certain numerical difficulties, the Rodrigues-Frank representation is of particular importance in the texture field due to the fact that incorporation of crystallographic symmetry is relatively straightforward. Note that the volume of Rodrigues space is infinite. In the fortran-90 implementation described later on in this document, the IEEE floating point standard is employed whenever the rotation angle equals $\omega = \pi$; in that case, the value of *infinity* is assigned to the vector length. While one cannot perform meaningful mathematical operations with this value, it is possible to assign it to a variable and to perform logical operations on it.

The following code snippet shows how one can define the number $+\infty$ for use in a fortran-90 program.

```
INTEGER,private  :: inf
REAL,public      :: infty
EQUIVALENCE (inf,infty) !Stores two variable at the same address
DATA inf/z'7f800000'/ !Hex for +Infinity
```

One can now use the variable infty to assign this value to a variable:

```
REAL  :: q
q = infty
```

or to perform logical tests:

```
if (q.eq.infty) then
  do something
else
  do something else
end if
```

2.3.5 Quaternion

Quaternions are four-dimensional complex numbers, typically written in the form

$$q = q_0 + \mathrm{i}q_1 + \mathrm{j}q_2 + \mathrm{k}q_3$$

where q_0 is the scalar part, $\mathbf{q} = (q_1, q_2, q_3)$ the vector part, and the imaginary units i, j, and k satisfy the following relations:

$$\mathrm{i}^2 = \mathrm{j}^2 = \mathrm{k}^2 = \mathrm{ijk} = -1;$$

$$\mathrm{ij} = -\mathrm{ji} = \mathrm{k}; \mathrm{jk} = -\mathrm{kj} = \mathrm{i}; \mathrm{ki} = -\mathrm{ik} = \mathrm{j};$$

as a result of these relations, quaternion multiplication is non-commutative. If one represents the quaternion as $q = (q_0, \mathbf{q})$, then quaternion addition and multiplication can be written succinctly as:

$$p + q = (p_0 + q_0, \mathbf{p} + \mathbf{q});$$

$$pq = (p_0 q_0 - \mathbf{p} \cdot \mathbf{q}, p_0 \mathbf{q} + q_0 \mathbf{p} + \mathbf{p} \times \mathbf{q}).$$

The norm of a quaternion is defined as

$$|p| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}$$

and the inverse of a quaternion can be computed via the following relation:

$$q^{-1} = \frac{q^*}{|q|^2} = (q_0, -\mathbf{q})/(q_0^2 + q_1^2 + q_2^2 + q_3^2)$$

The asterisk denotes quaternion conjugation, which changes the sign of the vector part of the quaternion.

In the f90 code, the quaternion is stored with the scalar part in the first position, followed by the vector part, i.e., a four-component array with entries (q_0, q_1, q_2, q_3).

Quaternions with unit norm are known as *unit quaternions* or *versors*, and they have a special relationship with 3D rotations, in a similar way to the relation between regular complex numbers and 2D rotations. Unit quaternions are located on the sphere $\mathcal{S}^3$ inside the 4D quaternion space. The 3D surface area of this sphere is equal to $2\pi^2$. One can show that the unit quaternions q and $-q$ represent the same rotation, so that all rotations can be represented by selecting either the Northern ($q_0 \geq 0$) or the Southern ($q_0 \leq 0$) hemisphere, which has surface area π^2.

Convention 7: The scalar part q_0 of a unit quaternion representing a rotation will always be positive (or 0 for rotations with rotation angle π).

Pure quaternions have a zero scalar part and they will turn out to be useful to carry out rotations on vectors. Despite the more complicated definition for quaternion multiplication, the quaternion representation of rotations is numerically very convenient and widely used.

2.3.6 Neo-Eulerian: homochoric

The 3D nature of the unit quaternion sphere embedded in 4D space is rather inconvenient when graphical representations are needed. The homochoric representation is defined as a generalization of the standard Lambert equal-area mapping, which projects the 2D sphere $\mathcal{S}^2$ (i.e., the collection of all 3D vectors with unit norm) onto a 2D disk. The generalization takes the 3D unit quaternion hemisphere and maps it onto a 3D ball with a radius selected so that the ball volume is equal to the surface area of the Northern hemisphere of the unit quaternion sphere.

The homochoric representation of rotations is defined by the following form of the function $f(\omega)$:

$$f(\omega) = \left[\frac{3}{4}(\omega - \sin\omega)\right]^{\frac{1}{3}} \qquad \text{for } \omega \in [0, \pi].$$

The maximum value of $f(\omega)$ is given by $f(\pi) = (3\pi/4)^{1/3}$, which is the radius of the homochoric ball; hence its volume equals π^2, which equals the volume of the Northern hemisphere of the unit quaternion sphere. Note that for $\omega \in [\pi, 2\pi[$, one would have to use the following definition for the function $f(\omega)$:

$$f(\omega) = \left[\frac{3}{4}(2\pi - \omega + \sin\omega)\right]^{\frac{1}{3}} \qquad \text{for } \omega \in [\pi, 2\pi[.$$

The homochoric vector, $\mathbf{h}$ is then defined as:

$$\mathbf{h} = \hat{\mathbf{n}} \left[\frac{3}{4}(\omega - \sin\omega)\right]^{\frac{1}{3}}.$$

Note that the sign limitations on the axis-angle pair are sufficient to guarantee that the above definition is all that is needed. The homochoric representation is an equal-volume representation, which, at least in principle, makes it easy to sample orientation space uniformly. The homochoric ball is isomorphic with SO(3).

2.3.7 Cubochoric

The homochoric representation is an equal-volume parameterization of rotation space. This means that a uniform sampling of the homochoric ball will provide a uniform sampling of rotation space. While there are several numerical approaches that provide uniform samplings of SO(3), they are not straightforward to work with, in particular in the context of crystallographic symmetry. The cubochoric mapping provides a simpler way to uniformly sample rotation space, by starting from a 3D uniform cubic grid, which is trivial to construct numerically. The cubochoric mapping is an equal-volume mapping from the cube to the homochoric ball, and therefore the volume of the cube is also equal to π^2. In this approach, rotations are represented by points inside the cube of edge length $\pi^{2/3}$; points on the cube outer surface correspond to 180° rotations, and the identity operation is located at the center of the cube. Concentric cubes have constant rotation angles. The cubochoric vector, $\mathbf{c}$ can be derived from the homochoric vector $\mathbf{h}$ by means of the transformation rule described in section 3.6.6.

3 Transformations between rotation representations

In this section, we list all the relevant transformation relations between rotation representations. While there are many direct transformation equations, in some cases it is easier to transform via an intermediate rotation representation. Note that each section starts with the name of a function that carries out the transformation; each name is of the form *xx2yy*, where *xx* is the two-letter designation of the original representation and *yy* the desired representation. In some cases, the text will refer to a transformation routine that will be defined in a later section.

The table below shows a summary of all the available transformation routines, using one-letter abbreviations; a checkmark indicates that a direct conversion is available, based on some analytical expression provided in one of the following sections; a letter or sequence of letters indicates that the transformation requires one or more intermediate representations. From a computational point of view, it is obviously better if there are fewer letters in an entry. Out of the 42 transformation functions, 21 are currently available using direct analytical implementations, the others use intermediate steps. All transformations have been implemented in the *rotations.f90* module, in both single and double precision versions.

Table 1: Available transformation routines between rotation representations.

↓ from \ to →	e	o	a	r	q	h	c
e	–	✓	✓	✓	✓	a	ah
o	✓	–	✓	e	✓	a	ah
a	o	✓	–	✓	✓	✓	h
r	o	a	✓	–	a	✓	h
q	✓	✓	✓	✓	–	✓	h
h	ao	a	✓	a	a	–	✓
c	hao	ha	h	ha	ha	✓	–

Note that in several of the equations on the following pages there is a red P present. This is explained in detail section 4. For now it is sufficient to state that $P = \pm 1$, with $P = +1$ corresponding to the relations in Morawiec's book.

3.1 Euler angles to other representations

3.1.1 *eu2om*: Euler angles to rotation/orientation matrix

The Euler angle triplet $\boldsymbol{\theta} = (\varphi_1, \Phi, \varphi_2)$ can be converted into a rotation matrix by multiplication of the individual (passive) rotation matrices:

$$\mathcal{R}_z(\varphi_1) = \begin{pmatrix} \cos\varphi_1 & \sin\varphi_1 & 0 \\ -\sin\varphi_1 & \cos\varphi_1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\mathcal{R}_{x'}(\Phi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\Phi & \sin\Phi \\ 0 & -\sin\Phi & \cos\Phi \end{pmatrix}$$

$$\mathcal{R}_{z''}(\varphi_2) = \begin{pmatrix} \cos\varphi_2 & \sin\varphi_2 & 0 \\ -\sin\varphi_2 & \cos\varphi_2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The overall rotation matrix is then given by:

$$\begin{aligned} \mathcal{R}(\boldsymbol{\theta}) &= \mathcal{R}_{z''}(\varphi_2)\mathcal{R}_{x'}(\Phi)\mathcal{R}_z(\varphi_1); \\ &= \begin{pmatrix} c_1c_2 - s_1c\,s_2 & s_1c_2 + c_1c\,s_2 & ss_2 \\ -c_1s_2 - s_1c\,c_2 & -s_1s_2 + c_1c\,c_2 & sc_2 \\ s_1\,s & -c_1\,s & c \end{pmatrix} \end{aligned}$$

where $c_i = \cos\varphi_i$, $s_i = \sin\varphi_i$, $c = \cos\Phi$, and $s = \sin\Phi$. This definition of the rotation matrix agrees with the one in Morawiec's book, page 28, and represents a passive rotation matrix; i.e., if the matrix is post-multiplied by a coordinate triplet, then the new coordinates will describe the vector in the rotated reference frame.

Special cases: There should not be any special cases, assuming that we keep the Euler angles within their respective range intervals (see Convention 3). Note that only positive rotation angles should be used.

3.1.2 *eu2ax*: Euler angles to axis-angle pair

The axis-angle pair $(\hat{\mathbf{n}}, \omega)$ can be obtained from the Bunge Euler angles by using the following relation:

$$(\hat{\mathbf{n}}, \omega) = \left(-\frac{P}{\tau} t\cos\delta, -\frac{P}{\tau} t\sin\delta, -\frac{P}{\tau}\sin\sigma, \alpha \right)$$

where

$$t = \tan\frac{\Phi}{2}; \quad \sigma = \frac{1}{2}(\varphi_1 + \varphi_2); \quad \delta = \frac{1}{2}(\varphi_1 - \varphi_2); \quad \tau = \sqrt{t^2 + \sin^2\sigma}; \quad \alpha = 2\arctan\tau.$$

Special cases: If $\alpha \leq \epsilon$, where ϵ is a threshold set to 10^{-16} for single and 10^{-10} for double prevision, then $\hat{\mathbf{n}} = [001]$, and $\omega = 0$ (identity operation). If $\alpha > \pi$, then the axis angle pair is given by:

$$(\hat{\mathbf{n}}, \omega) = \left(\frac{P}{\tau} t\cos\delta, \frac{P}{\tau} t\sin\delta, \frac{P}{\tau}\sin\sigma, 2\pi - \alpha \right).$$

3.1.3 *eu2ro*: Euler angles to Rodrigues-Frank vector

The Rodrigues-Frank vector $\boldsymbol{\rho}$ has components ρ_i with respect to a cartesian reference frame. In terms of the Bunge Euler angles (in radians), the components are determined by first converting the Euler angles to an axis-angle pair, using the relations in section 3.1.2. Then the fourth component is converted to $\tan(\omega/2)$, taking care of the $\omega = \pi$ case. To allow for the proper handling of infinity, while maintaining the direction cosines of the rotation axis, the Rodrigues-Frank vector is stored as a four-component vector, with n_i in the first three spots, and $\tan(\omega/2)$ in the last spot.

3.1.4 *eu2qu*: Euler angle to quaternion

Starting from Euler angles in radians, we define

$$\sigma = \frac{1}{2}(\varphi_1 + \varphi_2); \quad \delta = \frac{1}{2}(\varphi_1 - \varphi_2); \quad c = \cos\frac{\Phi}{2}; \quad s = \sin\frac{\Phi}{2};$$

The quaternion is then given by:

$$q = (c\cos\sigma, -Ps\cos\delta, -Ps\sin\delta, -Pc\sin\sigma)\,.$$

Special cases: If q_0 becomes negative, then the sign of the entire quaternion has to be reversed, $q \rightarrow -q$, so that the resulting quaternion will lie in the Northern hemisphere of the unit quaternion sphere.

3.1.5 *eu2ho*: Euler angle to homochoric

First we apply the *eu2ax* routine (section 3.1.2) to obtain the axis-angle pair representation; then we apply the *ax2ho* routine (section 3.3.5) to obtain the homochoric vector $\mathbf{h}$.

Special cases: None, because *eu2ax* should never return a negative angle.

3.1.6 *eu2cu*: Euler angle to cubochoric vector

First we apply the *eu2ho* routine (section 3.1.5) to obtain the homochoric vector $\mathbf{h}$. Then transform the homochoric vector to the cubochoric vector, $\mathbf{c}$, using *ho2cu* (section 3.6.6).

Special cases: None.

3.2 Rotation/orientation matrix to other representations

3.2.1 *om2eu*: Rotation/orientation matrix to Euler angles

Let the rotation matrix be given by R_{ij}, then, if $|R_{33}| \neq 1$, compute ζ as

$$\zeta = \frac{1}{\sqrt{1 - R_{33}^2}}.$$

Then use the following relations to determine the angles:

$$\boldsymbol{\theta} = (\mathrm{atan2}(R_{31}\zeta, -R_{32}\zeta), \arccos(R_{33}), \mathrm{atan2}(R_{13}\zeta, R_{23}\zeta))\,.$$

If $|R_{33}| = 1$, i.e. $\Phi = 0$, then we can not determine a unique value for φ_1 and φ_2. So, we will use the convention that the entire angle is represented by φ_1, and we set $\varphi_2 = 0$. If $R_{33} = 1$ then

$$\boldsymbol{\theta} = (\mathrm{atan2}(R_{12}, R_{11}), 0, 0)\,;$$

if $R_{33} = -1$, then

$$\boldsymbol{\theta} = (-\mathrm{atan2}(-R_{12}, R_{11}), \pi, 0)\,.$$

3.2.2 *om2ax*: Rotation/orientation matrix to axis-angle pair

We assume that the rotation matrix represents a passive rotation. Starting from the rotation matrix R_{ij}, we can compute the rotation angle via the trace of the matrix:

$$\omega = \arccos\left(\frac{1}{2}(\mathrm{Tr}(R) - 1)\right)$$

If the rotation angle equals zero, we return the unit vector [001] and $\omega = 0$.

For the direction cosines of the rotation axis, we determine the eigenvector of the rotation matrix corresponding to the eigenvalue +1; it is well known that every rotation matrix has three eigenvalues $(1, \mathrm{e}^{\mathrm{i}\omega}, \mathrm{e}^{-\mathrm{i}\omega})$. We use the LAPACK routines sgeev and dgeev to compute eigenvalues and eigenvectors, then look for the eigenvector corresponding to the eigenvalue +1.

For each component of the (right) eigenvector, res, we must then verify the sign as follows:

```
if ((om(2,3)-om(3,2)).ne.0.D0) res(1) = sign(res(1),-epsijk*(om(2,3)-om(3,2)))
if ((om(3,1)-om(1,3)).ne.0.D0) res(2) = sign(res(2),-epsijk*(om(3,1)-om(1,3)))
if ((om(1,2)-om(2,1)).ne.0.D0) res(3) = sign(res(3),-epsijk*(om(1,2)-om(2,1)))
```

The rotation matrix is represented by om, and res$(1 \ldots 3)$ are the unit vector components. The f90 sign function takes two arguments, and returns the first argument with the sign of the second argument. Note that the variable epsijk$= P$.

Special cases: There should not be any special cases; using the eigenvector approach should capture any special cases automatically, without the need for additional code.

3.2.3 *om2ro*: Rotation/orientation matrix to Rodrigues-Frank vector

First convert the matrix to Euler angles using *om2eu* (section 3.2.1), then convert the Euler angle triplet to a Rodrigues vector using *eu2ro* (section 3.1.3).

3.2.4 *om2qu*: Rotation/orientation matrix to quaternion

Consider a rotation matrix R_{ij}. The corresponding quaternion can be determined as follows:

- compute the quaternion components (one should check the square root argument to make sure it is positive):

$$\begin{aligned} q_0 &= \frac{1}{2}\sqrt{1 + R_{11} + R_{22} + R_{33}}; \\ q_1 &= \frac{1}{2}\sqrt{1 + R_{11} - R_{22} - R_{33}}; \\ q_2 &= \frac{1}{2}\sqrt{1 - R_{11} + R_{22} - R_{33}}; \\ q_3 &= \frac{1}{2}\sqrt{1 - R_{11} - R_{22} + R_{33}}. \end{aligned}$$

- Change the component signs if necessary:

$$\begin{aligned} &\text{if } R_{32} < R_{23} \rightarrow q_1 = -P q_1; \\ &\text{if } R_{13} < R_{31} \rightarrow q_2 = -P q_2; \\ &\text{if } R_{21} < R_{12} \rightarrow q_3 = -P q_3. \end{aligned}$$

- normalize the quaternion:

$$q = \frac{q}{|q|}$$

with $|q| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}$.

Using this approach will not always generate the correct sign for the unit vector. To make sure that the sign is correct, we call the *om2ax* routine (3.2.2), and then compare the sign of the unit vector components with the sign of the corresponding quaternion component. There maybe a more efficient way of implementing this.

Special cases: None.

3.2.5 *om2ho*: Rotation/orientation matrix to homochoric vector

First convert the matrix to Euler angles using *om2eu* (section 3.2.1), then convert the Euler angle triplet to a homochoric vector using *eu2ho* (section 3.1.5).

3.2.6 *om2cu*: Rotation/orientation matrix to cubochoric vector

First convert the matrix to a homochoric vector using *om2ho* (section 3.2.5), then transform to a cubochoric vector using *ho2cu* (section 3.6.6).

3.3 Axis-angle pair to other representations

3.3.1 *ax2eu*: Axis-angle pair to Euler angle

First convert axis-angle pair to an orientation matrix using *ax2om* in section 3.3.2, then convert to Euler angles using *om2eu* in section 3.2.1.

Special cases: None.

3.3.2 *ax2om*: Axis-angle pair to rotation/orientation matrix

For the axis angle pair $(\hat{\mathbf{n}}, \omega)$, compute $c = \cos(\omega)$, $s = \sin(\omega)$; then compute

$$\mathcal{R} = \begin{pmatrix} c + (1-c)n_1^2 & (1-c)n_1n_2 + s\,n_3 & (1-c)n_1n_3 - s\,n_2 \\ (1-c)n_1n_2 - s\,n_3 & c + (1-c)n_2^2 & (1-c)n_2n_3 + s\,n_1 \\ (1-c)n_1n_3 + s\,n_2 & (1-c)n_2n_3 - s\,n_1 & c + (1-c)n_3^2 \end{pmatrix}$$

Special cases: None.

3.3.3 *ax2ro*: Axis-angle pair to Rodrigues-Frank vector

For an axis-angle pair $(\hat{\mathbf{n}}, \omega)$, compute

$$\boldsymbol{\rho} = \hat{\mathbf{n}} \tan \frac{\omega}{2}$$

Special cases: For $\omega = \pi$, this would result in infinity, so we need to intercept this case using the floating point infinity handling described earlier in this document.

3.3.4 *ax2qu*: Axis-angle pair to quaternion

Given the axis-angle pair $(\hat{\mathbf{n}}, \omega)$, the quaternion is given by:

$$q = \left(\cos\frac{\omega}{2}, \hat{\mathbf{n}}\sin\frac{\omega}{2}\right)$$

Special cases: There should not be any special cases, since the angle $\omega \in [0, \pi]$, so that the q_0 component should always be positive.

3.3.5 *ax2ho*: Axis-angle pair to homochoric vector

Given the axis-angle pair $(\hat{\mathbf{n}}, \omega)$, compute the following parameter:

$$f = \left(\frac{3}{4}(\omega - \sin\omega)\right)^{\frac{1}{3}};$$

the homochoric vector is then given by

$$\mathbf{h} = \hat{\mathbf{n}}\, f$$

Special cases: There should not be any special cases, since the angle $\omega \in [0, \pi]$.

3.3.6 *ax2cu*: Axis-angle pair to cubochoric vector

First transform the axis-angle pair $(\hat{\mathbf{n}}, \omega)$ into the homochoric vector, $\mathbf{h}$ using *ax2ho* (section 3.3.5); then transform this vector to the cubochoric representation $\mathbf{c}$ using *ho2cu* (section 3.6.6).

Special cases: None.

3.4 Rodrigues vector to other representations

3.4.1 *ro2eu*: Rodrigues vector to Euler angles

First convert the Rodrigues-Frank vector to an orientation matrix using *ro2om* (section 3.4.2), then convert the matrix to Euler angles using *om2eu* (section 3.2.1).

Special cases: None.

3.4.2 *ro2om*: Rodrigues vector to rotation/orientation matrix

First convert the Rodrigues-Frank vector to an axis-angle pair using *ro2ax* (section 3.4.3), then convert the axis-angle pair to an orientation matrix using *ax2om* (section 3.3.2).

Special cases: None.

3.4.3 *ro2ax*: Rodrigues vector to axis-angle pair

Consider the Rodrigues-Frank vector $\boldsymbol{\rho}$. First define

$$\rho = |\boldsymbol{\rho}|.$$

Then compute the axis-angle pair as:

$$(\hat{\mathbf{n}}, \omega) = \left(\frac{\boldsymbol{\rho}}{\rho}, 2\arctan\rho\right)$$

Special cases: if $\rho = 0$, then return $\hat{\mathbf{n}} = [001]$; is $\rho = \infty$, then return a rotation angle of π.

3.4.4 *ro2qu*: Rodrigues vector to quaternion

First convert the Rodrigues-Frank vector to an axis-angle pair using *ro2ax* (section 3.4.3), then convert the axis-angle pair to a quaternion using *ax2qu* (section 3.3.4).

Special cases: None.

3.4.5 *ro2ho*: Rodrigues vector to homochoric vector

For a Rodrigues-Frank vector $\boldsymbol{\rho}$, consider the fourth argument, ρ. if $\rho = 0$ then return $\mathbf{h} = (0, 0, 0)$. If $\rho = \infty$, then set $f = 3\pi/4$, otherwise compute $\omega = 2\arctan(\rho)$ and $f = 3(\omega - \sin\omega)/4$; then return

$$\mathbf{h} = \hat{\mathbf{n}} f^{\frac{1}{3}}.$$

Special cases: None.

3.4.6 *ro2cu*: Rodrigues vector to cubochoric vector

First convert the Rodrigues-Frank vector to a homochoric vector using *ro2ho* (section 3.4.5), then convert the homochoric vector to a cubochoric vector using *ho2cu* (section 3.6.6).

Special cases: None.

3.5 Quaternion to other representations

3.5.1 *qu2eu*: Quaternion to Euler angles

For a given unit quaternion q, compute $q_{03} = q_0^2 + q_3^2$, $q_{12} = q_1^2 + q_3^2$, and $\chi = \sqrt{q_{03} q_{12}}$. Distinguish between the following cases:

- If $\chi = 0$ and $q_{12} = 0$, then

$$\boldsymbol{\theta} = \big(\text{atan2}(-2Pq_0q_3, q_0^2 - q_3^2), 0, 0\big)$$

- If $\chi = 0$ and $q_{03} = 0$, then

$$\boldsymbol{\theta} = \big(\text{atan2}(2q_1q_2, q_1^2 - q_2^2), \pi, 0\big)$$

- If $\chi \neq 0$, then

$$\boldsymbol{\theta} = \left(\text{atan2}\left(\frac{q_1q_3 - Pq_0q_2}{\chi}, \frac{-Pq_0q_1 - q_2q_3}{\chi}\right), \text{atan2}(2\chi, q_{03} - q_{12}), \text{atan2}\left(\frac{Pq_0q_2 + q_1q_3}{\chi}, \frac{q_2q_3 - Pq_0q_1}{\chi}\right)\right)$$

Special cases: None.

3.5.2 *qu2om*: Quaternion to rotation/orientation matrix

Consider a unit quaternion q and define $\bar{q} = q_0^2 - (q_1^2 + q_2^2 + q_3^2)$. Then the rotation matrix is given by:

$$\mathcal{R} = \begin{pmatrix} \bar{q} + 2q_1^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & \bar{q} + 2q_2^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & \bar{q} + 2q_3^2 \end{pmatrix}$$

If $P = -1$, then transpose the matrix.

Special cases: None.

3.5.3 *qu2ax*: Quaternion to axis-angle pair

First compute the rotation angle ω from

$$\omega = 2\arccos(q_0).$$

If $\omega = 0$, then return $\hat{\mathbf{n}} = [001]$. Otherwise:

- if $q_0 = 0$, return $\hat{\mathbf{n}} = [q_1, q_2, q_3]$ and $\omega = \pi$
- if $q_0 \neq 0$, compute

$$s = \frac{\text{sign}(q_0)}{\sqrt{q_1^2 + q_2^2 + q_3^2}}$$

  and return $\hat{\mathbf{n}} = [sq_1, sq_2, sq_3]$ along with ω.

Special cases: None.

3.5.4 *qu2ro*: Quaternion to Rodrigues-Frank vector

If $q_0 \leq \epsilon$, with $\epsilon = 10^{-8}$ for single precision and $\epsilon = 10^{-10}$ for double precision, then return $\boldsymbol{\rho} = [q_1, q_2, q_3, \infty]$. Otherwise, compute

$$s = \sqrt{q_1^2 + q_2^2 + q_3^2}$$

If $s \leq \epsilon$, return $\boldsymbol{\rho} = [0, 0, 0, 0]$, otherwise compute $t = \tan(\arccos(q_0))$ and return $\boldsymbol{\rho}[q_1/s, q_2/s, q_3/s, t]$ (recall that the Rodrigues-Frank vector is stored as a four-component vector).

Special cases: None.

3.5.5 *qu2ho*: Quaternion to homochoric vector

Compute $\omega = 2\arccos(q_0)$; if $\omega = 0$, return $\mathbf{h} = \mathbf{0}$, otherwise compute $s = 1/\sqrt{q_1^2 + q_2^2 + q_3^2}$, put $\hat{\mathbf{n}} = [sq_1, sq_2, sq_3]$, set $f = 3(\omega - \sin\omega)/4$ and return

$$\mathbf{h} = \hat{\mathbf{n}} f^{\frac{1}{3}}.$$

Special cases: None.

3.5.6 *qu2cu*: Quaternion to cubochoric vector

First convert the quaternion to homochoric vector using *qu2ho* in section 3.5.5, then convert to cubochoric vector using *ho2cu* in section 3.6.6.

Special cases: None.

3.6 Homochoric vector to other representations

3.6.1 *ho2eu*: Homochoric vector to Euler angles

First convert the homochoric vector to an axis-angle pair using *ho2ax* in section 3.6.3, then convert to Euler angles using *ax2eu* in section 3.3.1.

Special cases: None.

3.6.2 *ho2om*: Homochoric vector to rotation/orientation matrix

First convert the homochoric vector to an axis-angle pair using *ho2ax* in section 3.6.3, then convert to an orientation matrix using *ax2om* in section 3.3.2.

Special cases: None.

3.6.3 *ho2ax*: Homochoric vector to axis-angle pair

Define a 16-component constant vector $\boldsymbol{\gamma}$ with the following values:

$$\begin{aligned}\boldsymbol{\gamma} = [&1.0000000000018852D0, -0.5000000002194847D0, -0.024999992127593126D0,\\ &-0.003928701544781374D0, -0.0008152701535450438D0, -0.0002009500426119712D0,\\ &-0.00002397986776071756D0, -0.0000820286892660584 1D0, +0.0001244871504209009 2D0,\\ &-0.0001749114214822577D0, +0.0001703481934140054D0, -0.0001206206500411682 8D0,\\ &+0.0000597197058686608 26D0, -0.0000198075672396564 7D0, +0.0000039537146842128 74D0,\\ &-0.0000003655500143971954 4D0]\end{aligned}$$

- compute $h = |\mathbf{h}|^2$;
- if $h = 0$, then return $(0, 0, 1; 0)$ as the axis-angle pair;
- else put $\mathbf{h}' = \mathbf{h}/\sqrt{h}$ and compute the sum

$$s = \sum_{i=1}^{16} \gamma_i \, h^{i-1};$$

then return $(\mathbf{h}', 2\arccos(s))$ as the axis-angle pair.

Note that this routine replaces a slower version in which the angle ω is solved for by means of an iterative approach.

Special cases: None.

3.6.4 *ho2ro*: Homochoric vector to Rodrigues-Frank vector

First convert the homochoric vector to an axis-angle pair using *ho2ax* in section 3.6.3, then convert to a Rodrigues-Frank vector using *ax2ro* in section 3.3.3.

3.6.5 *ho2qu*: Homochoric vector to quaternion

First convert the homochoric vector to an axis-angle pair using *ho2ax* in section 3.6.3, then convert to a quaternion using *ax2qu* in section 3.3.4.

3.6.6 *ho2cu*: Homochoric vector to cubochoric vector

Given the homochoric vector, $\mathbf{h}$, one can obtain the cubochoric vector, $\mathbf{c}$, by means of the following steps:

1. Make sure that $|\mathbf{h}| \leq (3\pi/4)^{1/3}$; if not, return zero-vector and error status 1.
2. If this is the identity rotation, then return the zero-vector and error status 0.

3. Determine which pyramidal section $\mathbf{h}$ is part of, and rearrange the components h_i accordingly: pyramids $1,2 \rightarrow$ no change; $3,4 \rightarrow \mathbf{h} = (h_2, h_3, h_1)$; $5,6 \rightarrow \mathbf{h} = (h_3, h_1, h_2)$.

4. Using the rearranged $\mathbf{h}$, put $q = (2|\mathbf{h}|/(|\mathbf{h}| + |h_3|))^{1/2}$; then compute

$$\mathbf{h}' = \left(qh_1, qh_2, \text{sign}(h_3)|\mathbf{h}|/\sqrt{6/\pi}\right)$$

5. Set $qxy = h_1'^2 + h_2'^2$; if $h_1' \neq 0 \rightarrow sx = \text{sign}(h_1')$, else $sx = 1$; if $h_2' \neq 0 \rightarrow sy = \text{sign}(h_2')$, else $sy = 1$;

6. if $|h_2'| \leq |h_1'|$ then compute

$$\begin{aligned} q2xy &= qxy + h_1'^2; \\ sq2xy &= \sqrt{q2xy}; \\ q &= \frac{\pi^{5/6}4^{1/3}}{2\sqrt{2}\,(3\pi)^{1/3}6^{1/6}}\sqrt{\frac{q2xy\,qxy}{(q2xy - |h_1'|sq2xy)}}; \\ ac &= \arccos((h_2'^2 + |h_1'|sq2xy)/\sqrt{2}/qxy); \\ T1inv &= q\,sx; \\ T2inv &= 12q\,sy\,ac/\pi. \end{aligned}$$

else $(|h_1'| \leq |h_2'|)$

$$\begin{aligned} qx2y &= qxy + h_2'^2; \\ sqx2y &= \sqrt{qx2y}; \\ q &= \frac{\pi^{5/6}4^{1/3}}{2\sqrt{2}\,(3\pi)^{1/3}6^{1/6}}\sqrt{\frac{qx2y\,qxy}{(qx2y - |h_2'|sqx2y)}}; \\ ac &= \arccos((h_1'^2 + |h_2'|sqx2y)/\sqrt{2}/qxy); \\ T1inv &= 12q\,sx\,ac/\pi; \\ T2inv &= q\,sy. \end{aligned}$$

Then set $\mathbf{h}'' = (6/\pi)^{1/6}(T1inv, T2inv, h_3')$

7. Finally, revert the components back to the original order (undoing the pyramid case): $1,2 \rightarrow \mathbf{c} = \mathbf{h}''$; $3,4 \rightarrow \mathbf{c} = (h_3'', h_1'', h_2'')$; and $5,6 \rightarrow \mathbf{c} = (h_2'', h_3'', h_1'')$. Return $\mathbf{c}$ and error status 0.

Special cases: None (already dealt with in steps 1 and 2)

3.7 Cubochoric vector to other representations

3.7.1 *cu2eu*: Cubochoric vector to Euler angles

First convert the cubochoric vector to a homochoric vector using *cu2ho* in section 3.7.6, then convert to a Euler angles using *ho2eu* in section 3.6.1.

Special cases: None.

3.7.2 *cu2om*: Cubochoric vector to rotation/orientation matrix

First convert the cubochoric vector to a homochoric vector using *cu2ho* in section 3.7.6, then convert to an orientation matrix using *ho2om* in section 3.6.2.

Special cases: None.

3.7.3 *cu2ax*: Cubochoric vector to axis-angle pair

First convert the cubochoric vector to a homochoric vector using *cu2ho* in section 3.7.6, then convert to an axis-angle pair using *ho2ax* in section 3.6.3.

Special cases: None.

3.7.4 *cu2ro*: Cubochoric vector to Rodrigues-Frank vector

First convert the cubochoric vector to a homochoric vector using *cu2ho* in section 3.7.6, then convert to a Rodrigues-Frank vector using *ho2ro* in section 3.6.4.

Special cases: None.

3.7.5 *cu2qu*: Cubochoric vector to quaternion

First convert the cubochoric vector to a homochoric vector using *cu2ho* in section 3.7.6, then convert to a quaternion using *ho2qu* in section 3.6.5.

Special cases: None.

3.7.6 *cu2ho*: Cubochoric vector to homochoric vector

Consider the cubochoric vector $\mathbf{c}$. First check if the maximum value of the absolute value of the components is less than the semi-edge length of the cube, which is $\pi^{2/3}/2$; if not, then return $\mathbf{h} = \mathbf{0}$ and error status 1.

1. Rearrange the components of $\mathbf{c}$ according to the pyramid in which it lies: $1, 2 \rightarrow$ no change needed; $3, 4 \rightarrow \mathbf{c} = (c_2, c_3, c_1)$; $5, 6 \rightarrow \mathbf{c} = (c_3, c_1, c_2)$.
2. Scale $\mathbf{c}$ to $\mathbf{C} = \mathbf{c}(\pi/6)^{1/6}$;
3. if $\mathbf{C} = \mathbf{0}$ then set $\boldsymbol{\lambda} = \mathbf{0}$;
4. else if $\mathbf{C}$ lies along the z-axis (first two components are zero) then $\boldsymbol{\lambda} = (0, 0, C_3\sqrt{6/\pi})$
5. in all other cases we have a general grid point; if $|C_2| \leq |C_1|$ then

$$
\begin{aligned}
q &= \frac{\pi}{12}\frac{C_2}{C_1}; \\
c &= \cos(q); \\
s &= \sin(q); \\
q &= \sqrt{\frac{6}{\pi}} 2^{\frac{1}{4}} \frac{C_1}{\sqrt{\sqrt{2} - c}}; \\
T1 &= (c\sqrt{2} - 1)q; \\
T2 &= s\, q\, \sqrt{2}.
\end{aligned}
$$

else ($|C_2| > |C_1|$)

$$
\begin{aligned}
q &= \frac{\pi}{12}\frac{C_1}{C_2};\\
c &= \cos(q);\\
s &= \sin(q);\\
q &= \sqrt{\frac{6}{\pi}}2^{\frac{1}{4}}\frac{C_2}{\sqrt{\sqrt{2}-c}};\\
T1 &= s\,q\,\sqrt{2};\\
T2 &= (c\sqrt{2}-1)q.
\end{aligned}
$$

and transform to the sphere grid:

$$
\begin{aligned}
c &= T1^2 + T2^2;\\
s &= \frac{\pi c}{24C_3^2};\\
c &= \sqrt{\frac{\pi}{24}}\frac{c}{C_3};\\
q &= \sqrt{1-s};\\
\boldsymbol{\lambda} &= (q\,T1, q\,T2, \sqrt{\frac{6}{\pi}}C_3 - c).
\end{aligned}
$$

6. Finally, rearrange the components of $\boldsymbol{\lambda}$ back into the correct order depending on the pyramid number: $1, 2 \rightarrow \mathbf{h} = \boldsymbol{\lambda}$; $3, 4 \rightarrow \mathbf{h} = (\lambda_3, \lambda_1, \lambda_2)$; $5, 6 \rightarrow \mathbf{h} = (\lambda_2, \lambda_3, \lambda_1)$. Return $\mathbf{h}$ and error status 0.

4 The final convention

The red symbol P in the equations of the previous section requires a bit of explanation. In the original version of the rotations.f90 module, all the relations between the various rotation parameterizations were based on the equations (and hence the sign conventions) in Adam Morawiec's book. When the transformation equations are applied with $P = +1$, then all representations are related to each other according to the Morawiec book. While this is a correct and self-consistent way of transforming between representations, some of the results that are obtained are a bit counter-intuitive, as we will now illustrate.

Consider the Euler angle triplet $\boldsymbol{\theta} = (90^\circ, 0^\circ, 0^\circ)$; the derived quantities using the relations in the text with $P = +1$, are:

- Passive rotation matrix:
$$\mathcal{R}_{\boldsymbol{\theta}} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$
- Quaternion:
$$q_{\boldsymbol{\theta}} = \left(\frac{1}{\sqrt{2}}, 0, 0, -\frac{1}{\sqrt{2}} \right)$$
- Axis-angle pair:
$$a = (\hat{\mathbf{n}}, \omega) = (0, 0, -1; 90^\circ)$$
- Rodrigues-Frank vector:
$$\boldsymbol{\rho} = (0, 0, -1)$$

Note that the axis angle pair has a unit vector of $[00\bar{1}]$! Intuitively, the Euler triplet $(90^\circ, 0^\circ, 0^\circ)$ corresponds to a counterclockwise rotation by 90° around the z or $[001]$ axis, so we would be inclined to state that the resulting axis-angle pair should be $([001]; 90^\circ)$, and yet the Morawiec transformation relations produce $[00\bar{1}]$ for the rotation axis. As long as all transformation relations are used consistently (i.e., all with $P = +1$), there are no problems, and all computations will give the correct result, but at the expense of sometimes producing counter-intuitive results.

If we set $P = -1$ everywhere, we obtain once again an internally consistent set of transformation relations, and this time the rotation axis for the Euler angle triplet $(90^\circ, 0^\circ, 0^\circ)$ results in $[001]$, which is in agreement with our intuition.

Convention 8: Setting the parameter P equal to $+1$ produces transformation relations that are consistent with the relations listed in Morawiec's book; setting $P = -1$ produces a second consistent set of transformation relations. It should be left up to the user to decide which of these should be used.

In the rotations.f90 module, the selection of P is accomplished by setting two parameters in the accompanying constants.f90 module. This module defines, among other things, two floating point parameters epsijk and epsijkd (d for double precision) which should be set equal to $+1$ to obtain the Morawiec convention, and -1 for the alternative convention. The parameters should be set at compile time.

5 Active and passive transformations

In this final section we describe several functions that implement passive and active coordinate transformations. The decision on whether to apply an active or a passive transformation ultimately resides with the user, so each transformation routine (for vectors, tensors, etc) has an input parameter that expresses this selection: the parameter is a single character ap, which is set to 'a' for active and 'p' for passive interpretation. The transformation routines are aware of the choice of P, so that the user does not have to worry about it.

The rotations.f90 module implements two different rotation function for vectors, one using the rotation matrix, the other using the quaternion rotation. Single and double precision versions are automatically selected based on the argument type. The module presents the user with a single routine, RotateVector(vec,rot,ap), where vec is a 3-component vector, rot is a 3×3 rotation matrix or a quaternion, and ap is a single character equal to 'a' or 'p'. Internally, the module defines two different transformation routines, one for the rotation matrix and one for the quaternion (single and double precision versions for each). As an example, here is the double precision transformation routine for a rotation matrix om:

```
recursive function RotVec_om_d(vec,om,ap) result(res)

use local

real(kind=dbl),INTENT(IN)       :: vec(3)
real(kind=dbl),INTENT(IN)       :: om(3,3)
character(1),INTENT(IN)         :: ap

real(kind=dbl)                  :: res(3)

if (ap.eq.'p') then
 res = matmul(om,vec)
else
 res = matmul(transpose(om),vec)
end if

end function RotVec_om_d
```

Since a rotation matrix is always defined as a passive matrix, *regardless of the sign of* P, the passive transformation is carried out by post-multiplying the matrix by the vector, using the f90 routine matmul, which multiplies the first entry by the second one. For active rotations, the matrix is first transposed before carrying out the multiplication.

Since the sign choice for P does affect the quaternion representation, the vector transformation routine is a bit more involved, as shown here for the double precision version:

```
recursive function RotVec_qu_d(vec,qu,ap) result(res)

use local
use quaternions
use constants

real(kind=dbl),INTENT(IN)       :: vec(3)
real(kind=dbl),INTENT(IN)       :: qu(4)
character(1),INTENT(IN)         :: ap
```

```
real(kind=dbl)                  :: res(3)
real(kind=dbl)                  :: rq(4), rr(4)

rq = (/ 0.D0, vec(1), vec(2), vec(3) /)

if (epsijk.lt.0) then
  if (ap.eq.'a') then
    rr = quat_mult(qu,quat_mult(rq,conjg(qu)) )
  else
    rr = quat_mult(conjg(qu),quat_mult(rq,qu) )
  end if
else
  if (ap.eq.'p') then
    rr = quat_mult(qu,quat_mult(rq,conjg(qu)) )
  else
    rr = quat_mult(conjg(qu),quat_mult(rq,qu) )
  end if
end if

res = rr(2:4)

end function RotVec_qu_d
```

These routines have been tested extensively for both signs of P (epsijk) and a variety of rotation parameters.

6 Tests

The rotations.f90 module was tested extensively using a series of Euler angle triplets as input. For each triplet, all equivalent representations were computed, and those were then subjected to pairwise tests and triplet tests.

- *pairwise tests:* these tests use a sequence of two transformations, out=yy2xx(xx2yy(in)), where both xx and yy represent any one of the seven representations. This leads to a total of 42 pairwise tests. The overall pairwise test is succesful if the largest value of the 42 differences between in and out is a small number, say of order 10^{-10} or so.

- *triplet tests:* these tests use a sequence of three consecutive transformation routines, as in out=zz2xx(yy2zz(xx2yy(in))), for a total of 210 tests; once again, the largest value of the 210 differences between in and out must be a small number, say of order 10^{-10} or so.

This set of tests is then repeated for a series of Euler angle input triplets with general and special values of the angles. The basic test takes 75 input triplets in which each Euler angle is incremented in steps of $\pi/2$ over the entire range for each angle, and all 75 triplets are then fed to the testing program. The test is deemed successful if the maximum differences are of order 10^{-10} or smaller for all 75 test triplets. All tests were carried out for both sign conventions for P.

As a reference, the following pages show all 75 Euler triplets with all equivalent representations, as well as the maximum differences for $P = -1$; to obtain the results for $P = +1$, the sign of the unit axis vector $\hat{\mathbf{n}}$ needs to be changed for the axis-angle, Rodrigues-Frank, quaternion, homochoric and cubochoric representations. Note that the Rodrigues-Frank vector is shown as a three-component vector when the rotation angle is different from π; when $\omega = \pi$, the first three components are the components of $\hat{\mathbf{n}}$, and the fourth entry is set to infinity.

```
Euler angles [deg]               :   0.0000000    0.0000000    0.0000000
Axis angle pair [n; angle (deg)] :   0.0000000    0.0000000    1.0000000 ;    0.0000000
Rodrigues vector                 :       0.0000000        0.0000000        0.0000000
Homochoric vector                :   0.0000000    0.0000000    0.0000000
Cubochoric vector                :   0.0000000    0.0000000    0.0000000
Quaternion [scalar, vector]      :   1.0000000    0.0000000    0.0000000    0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   1.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    0.0000000000000000
  Maximum difference in triplet tests  :    0.0000000000000000

 ------------------------------------------------------
Euler angles [deg]               :  90.0000000    0.0000000    0.0000000
Axis angle pair [n; angle (deg)] :   0.0000000    0.0000000    1.0000000 ;   90.0000000
Rodrigues vector                 :       0.0000000        0.0000000        1.0000000
Homochoric vector                :   0.0000000    0.0000000    0.7536693
Cubochoric vector                :   0.0000000    0.0000000    0.6074544
Quaternion [scalar, vector]      :   0.7071068    0.0000000    0.0000000    0.7071068
                                   /  0.0000   1.0000   0.0000 \
Orientation Matrix               : | -1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    4.8346515679753566E-010
  Maximum difference in triplet tests  :    4.8346520658749818E-010

 ------------------------------------------------------
Euler angles [deg]               : 180.0000000    0.0000000    0.0000000
Axis angle pair [n; angle (deg)] :   0.0000000    0.0000000    1.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.0000000        0.0000000        1.0000000 ;        Infinity
Homochoric vector                :   0.0000000    0.0000000    1.3306700
Cubochoric vector                :   0.0000000    0.0000000    1.0725147
Quaternion [scalar, vector]      :   0.0000000    0.0000000    0.0000000    1.0000000
                                   / -1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000  -1.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    2.2204460492503131E-016
  Maximum difference in triplet tests  :    2.2204460492503131E-016

 ------------------------------------------------------
Euler angles [deg]               : 270.0000000    0.0000000    0.0000000
Axis angle pair [n; angle (deg)] :   0.0000000   -0.0000000   -1.0000000 ;   90.0000000
Rodrigues vector                 :       0.0000000       -0.0000000       -1.0000000
Homochoric vector                :   0.0000000   -0.0000000   -0.7536693
Cubochoric vector                :   0.0000000    0.0000000   -0.6074544
Quaternion [scalar, vector]      :   0.7071068    0.0000000   -0.0000000   -0.7071068
                                   /  0.0000  -1.0000   0.0000 \
Orientation Matrix               : |  1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    4.8346526781983812E-010
  Maximum difference in triplet tests  :    4.8346526781983812E-010
```

```
 ------------------------------------------------------
Euler angles [deg]               : 360.0000000    0.0000000    0.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000   -0.0000000   -1.0000000 ;   -0.0000000
Rodrigues vector                 :       0.0000000        0.0000000        0.0000000
Homochoric vector                :  -0.0000000   -0.0000000   -0.0000000
Cubochoric vector                :   0.0000000    0.0000000    0.0000000
Quaternion [scalar, vector]      :   1.0000000    0.0000000   -0.0000000   -0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   1.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    1.2246467991473532E-016
  Maximum difference in triplet tests  :    7.2278920375157370E-015

 ------------------------------------------------------
Euler angles [deg]               :   0.0000000   90.0000000    0.0000000
Axis angle pair [n; angle (deg)] :   1.0000000    0.0000000    0.0000000 ;   90.0000000
Rodrigues vector                 :       1.0000000        0.0000000        0.0000000
Homochoric vector                :   0.7536693    0.0000000    0.0000000
Cubochoric vector                :   0.6074544    0.0000000    0.0000000
Quaternion [scalar, vector]      :   0.7071068    0.7071068    0.0000000    0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   0.0000   1.0000 |
                                   \  0.0000  -1.0000   0.0000 /


  Maximum difference in pairwise tests  :    4.8346515679753566E-010
  Maximum difference in triplet tests  :    4.8346520658749818E-010

 ------------------------------------------------------
Euler angles [deg]               :  90.0000000   90.0000000    0.0000000
Axis angle pair [n; angle (deg)] :   0.5773503    0.5773503    0.5773503 ;  120.0000000
Rodrigues vector                 :       1.0000000        1.0000000        1.0000000
Homochoric vector                :   0.5617842    0.5617842    0.5617842
Cubochoric vector                :   0.7842653    0.7842653    0.7842653
Quaternion [scalar, vector]      :   0.5000000    0.5000000    0.5000000    0.5000000
                                   /  0.0000   1.0000   0.0000 \
Orientation Matrix               : |  0.0000   0.0000   1.0000 |
                                   \  1.0000   0.0000   0.0000 /


  Maximum difference in pairwise tests  :    1.3439116486324565E-010
  Maximum difference in triplet tests  :    1.3439205304166535E-010

 ------------------------------------------------------
Euler angles [deg]               : 180.0000000   90.0000000    0.0000000
Axis angle pair [n; angle (deg)] :   0.0000000    0.7071068    0.7071068 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.0000000        0.7071068        0.7071068 ;        Infinity
Homochoric vector                :   0.0000000    0.9409258    0.9409258
Cubochoric vector                :   0.0000000    1.0725147    1.0725147
Quaternion [scalar, vector]      :   0.0000000    0.0000000    0.7071068    0.7071068
                                   / -1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   0.0000   1.0000 |
                                   \  0.0000   1.0000   0.0000 /


  Maximum difference in pairwise tests  :    6.6856799808063715E-015
  Maximum difference in triplet tests  :    6.6856799808063715E-015
```

```
 ----------------------------------------------------
Euler angles [deg]               : 270.0000000   90.0000000    0.0000000
Axis angle pair [n; angle (deg)] :   0.5773503   -0.5773503   -0.5773503 ;  120.0000000
Rodrigues vector                 :       1.0000000       -1.0000000       -1.0000000
Homochoric vector                :   0.5617842   -0.5617842   -0.5617842
Cubochoric vector                :   0.7842653   -0.7842653   -0.7842653
Quaternion [scalar, vector]      :   0.5000000    0.5000000   -0.5000000   -0.5000000
                                   /  0.0000  -1.0000   0.0000 \
Orientation Matrix               : |  0.0000   0.0000   1.0000 |
                                   \ -1.0000   0.0000   0.0000 /


  Maximum difference in pairwise tests  :    1.3440182300428205E-010
  Maximum difference in triplet tests  :    1.3442225110793515E-010

 ----------------------------------------------------
Euler angles [deg]               : 360.0000000   90.0000000    0.0000000
Axis angle pair [n; angle (deg)] :   1.0000000   -0.0000000   -0.0000000 ;   90.0000000
Rodrigues vector                 :       1.0000000       -0.0000000       -0.0000000
Homochoric vector                :   0.7536693   -0.0000000   -0.0000000
Cubochoric vector                :   0.6074544   -0.0000000   -0.0000000
Quaternion [scalar, vector]      :   0.7071068    0.7071068   -0.0000000   -0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   0.0000   1.0000 |
                                   \  0.0000  -1.0000   0.0000 /


  Maximum difference in pairwise tests  :    4.8346515679753566E-010
  Maximum difference in triplet tests  :    4.8346520658749818E-010

 ----------------------------------------------------
Euler angles [deg]               :   0.0000000  180.0000000    0.0000000
Axis angle pair [n; angle (deg)] :   1.0000000    0.0000000    0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       1.0000000        0.0000000        0.0000000 ;        Infinity
Homochoric vector                :   1.3306700    0.0000000    0.0000000
Cubochoric vector                :   1.0725147    0.0000000    0.0000000
Quaternion [scalar, vector]      :   0.0000000    1.0000000    0.0000000    0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000  -1.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    2.2204460492503131E-016
  Maximum difference in triplet tests  :    2.2204460492503131E-016

 ----------------------------------------------------
Euler angles [deg]               :  90.0000000  180.0000000    0.0000000
Axis angle pair [n; angle (deg)] :   0.7071068    0.7071068    0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.7071068        0.7071068        0.0000000 ;        Infinity
Homochoric vector                :   0.9409258    0.9409258    0.0000000
Cubochoric vector                :   1.0725147    1.0725147    0.0000000
Quaternion [scalar, vector]      :   0.0000000    0.7071068    0.7071068    0.0000000
                                   /  0.0000   1.0000   0.0000 \
Orientation Matrix               : |  1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    4.4408920985006262E-016
```

```
  Maximum difference in triplet tests  :    4.4408920985006262E-016

 ----------------------------------------------------
Euler angles [deg]               : 180.0000000  180.0000000    0.0000000
Axis angle pair [n; angle (deg)] :   0.0000000    1.0000000    0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.0000000        1.0000000        0.0000000 ;        Infinity
Homochoric vector                :   0.0000000    1.3306700    0.0000000
Cubochoric vector                :   0.0000000    1.0725147    0.0000000
Quaternion [scalar, vector]      :   0.0000000    0.0000000    1.0000000    0.0000000
                                   / -1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   1.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    2.2204460492503131E-016
  Maximum difference in triplet tests  :    2.2204460492503131E-016

 ----------------------------------------------------
Euler angles [deg]               : 270.0000000  180.0000000    0.0000000
Axis angle pair [n; angle (deg)] :   0.7071068   -0.7071068   -0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.7071068       -0.7071068       -0.0000000 ;        Infinity
Homochoric vector                :   0.9409258   -0.9409258   -0.0000000
Cubochoric vector                :   1.0725147   -1.0725147   -0.0000000
Quaternion [scalar, vector]      :   0.0000000    0.7071068   -0.7071068   -0.0000000
                                   /  0.0000  -1.0000   0.0000 \
Orientation Matrix               : | -1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    1.9317880628477724E-014
  Maximum difference in triplet tests  :    1.9317880628477724E-014

 ----------------------------------------------------
Euler angles [deg]               : 360.0000000  180.0000000    0.0000000
Axis angle pair [n; angle (deg)] :   1.0000000   -0.0000000   -0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       1.0000000       -0.0000000       -0.0000000 ;        Infinity
Homochoric vector                :   1.3306700   -0.0000000   -0.0000000
Cubochoric vector                :   1.0725147   -0.0000000   -0.0000000
Quaternion [scalar, vector]      :   0.0000000    1.0000000   -0.0000000   -0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000  -1.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    9.7852813551359461E-015
  Maximum difference in triplet tests  :    9.7852813551359461E-015

 ----------------------------------------------------
Euler angles [deg]               :   0.0000000    0.0000000   90.0000000
Axis angle pair [n; angle (deg)] :   0.0000000   -0.0000000    1.0000000 ;   90.0000000
Rodrigues vector                 :       0.0000000       -0.0000000        1.0000000
Homochoric vector                :   0.0000000   -0.0000000    0.7536693
Cubochoric vector                :   0.0000000    0.0000000    0.6074544
Quaternion [scalar, vector]      :   0.7071068    0.0000000   -0.0000000    0.7071068
                                   /  0.0000   1.0000   0.0000 \
Orientation Matrix               : | -1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /
```

```
  Maximum difference in pairwise tests  :    4.8346515679753566E-010
  Maximum difference in triplet tests  :    4.8346520658749818E-010

 ------------------------------------------------------
Euler angles [deg]               :  90.0000000    0.0000000   90.0000000
Axis angle pair [n; angle (deg)] :   0.0000000    0.0000000    1.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.0000000        0.0000000        1.0000000 ;        Infinity
Homochoric vector                :   0.0000000    0.0000000    1.3306700
Cubochoric vector                :   0.0000000    0.0000000    1.0725147
Quaternion [scalar, vector]      :   0.0000000    0.0000000    0.0000000    1.0000000
                                   / -1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000  -1.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    2.2204460492503131E-016
  Maximum difference in triplet tests  :    2.2204460492503131E-016

 ------------------------------------------------------
Euler angles [deg]               : 180.0000000    0.0000000   90.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000   -0.0000000   -1.0000000 ;   90.0000000
Rodrigues vector                 :      -0.0000000       -0.0000000       -1.0000000
Homochoric vector                :  -0.0000000   -0.0000000   -0.7536693
Cubochoric vector                :   0.0000000    0.0000000   -0.6074544
Quaternion [scalar, vector]      :   0.7071068   -0.0000000   -0.0000000   -0.7071068
                                   /  0.0000  -1.0000   0.0000 \
Orientation Matrix               : |  1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    4.8346526781983812E-010
  Maximum difference in triplet tests  :    4.8346526781983812E-010

 ------------------------------------------------------
Euler angles [deg]               : 270.0000000    0.0000000   90.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000   -0.0000000   -1.0000000 ;   -0.0000000
Rodrigues vector                 :       0.0000000        0.0000000        0.0000000
Homochoric vector                :  -0.0000000   -0.0000000   -0.0000000
Cubochoric vector                :   0.0000000    0.0000000    0.0000000
Quaternion [scalar, vector]      :   1.0000000   -0.0000000   -0.0000000   -0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   1.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    1.2246467991473532E-016
  Maximum difference in triplet tests  :    7.2278920375157370E-015

 ------------------------------------------------------
Euler angles [deg]               : 360.0000000    0.0000000   90.0000000
Axis angle pair [n; angle (deg)] :   0.0000000   -0.0000000    1.0000000 ;   90.0000000
Rodrigues vector                 :       0.0000000       -0.0000000        1.0000000
Homochoric vector                :   0.0000000   -0.0000000    0.7536693
Cubochoric vector                :   0.0000000    0.0000000    0.6074544
Quaternion [scalar, vector]      :   0.7071068    0.0000000   -0.0000000    0.7071068
                                   /  0.0000   1.0000   0.0000 \
Orientation Matrix               : | -1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /
```

```
  Maximum difference in pairwise tests  :    4.8346537884214058E-010
  Maximum difference in triplet tests  :    4.8346537884214058E-010

 ----------------------------------------------------
Euler angles [deg]               :   0.0000000   90.0000000   90.0000000
Axis angle pair [n; angle (deg)] :   0.5773503   -0.5773503    0.5773503 ;  120.0000000
Rodrigues vector                 :       1.0000000       -1.0000000        1.0000000
Homochoric vector                :   0.5617842   -0.5617842    0.5617842
Cubochoric vector                :   0.7842653   -0.7842653    0.7842653
Quaternion [scalar, vector]      :   0.5000000    0.5000000   -0.5000000    0.5000000
                                   /  0.0000   0.0000   1.0000 \
Orientation Matrix               : | -1.0000   0.0000   0.0000 |
                                   \  0.0000  -1.0000   0.0000 /


  Maximum difference in pairwise tests  :    1.3439116486324565E-010
  Maximum difference in triplet tests  :    1.3439205304166535E-010

 ----------------------------------------------------
Euler angles [deg]               :  90.0000000   90.0000000   90.0000000
Axis angle pair [n; angle (deg)] :   0.7071068    0.0000000    0.7071068 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.7071068        0.0000000        0.7071068 ;        Infinity
Homochoric vector                :   0.9409258    0.0000000    0.9409258
Cubochoric vector                :   1.0725147    0.0000000    1.0725147
Quaternion [scalar, vector]      :   0.0000000    0.7071068    0.0000000    0.7071068
                                   /  0.0000   0.0000   1.0000 \
Orientation Matrix               : |  0.0000  -1.0000   0.0000 |
                                   \  1.0000   0.0000   0.0000 /


  Maximum difference in pairwise tests  :    4.4408920985006262E-016
  Maximum difference in triplet tests  :    4.4408920985006262E-016

 ----------------------------------------------------
Euler angles [deg]               : 180.0000000   90.0000000   90.0000000
Axis angle pair [n; angle (deg)] :  -0.5773503   -0.5773503   -0.5773503 ;  120.0000000
Rodrigues vector                 :      -1.0000000       -1.0000000       -1.0000000
Homochoric vector                :  -0.5617842   -0.5617842   -0.5617842
Cubochoric vector                :  -0.7842653   -0.7842653   -0.7842653
Quaternion [scalar, vector]      :   0.5000000   -0.5000000   -0.5000000   -0.5000000
                                   /  0.0000   0.0000   1.0000 \
Orientation Matrix               : |  1.0000   0.0000   0.0000 |
                                   \  0.0000   1.0000   0.0000 /


  Maximum difference in pairwise tests  :    1.3440182300428205E-010
  Maximum difference in triplet tests  :    1.3440182300428205E-010

 ----------------------------------------------------
Euler angles [deg]               : 270.0000000   90.0000000   90.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000   -1.0000000   -0.0000000 ;   90.0000000
Rodrigues vector                 :      -0.0000000       -1.0000000       -0.0000000
Homochoric vector                :  -0.0000000   -0.7536693   -0.0000000
Cubochoric vector                :  -0.0000000   -0.6074544   -0.0000000
Quaternion [scalar, vector]      :   0.7071068   -0.0000000   -0.7071068   -0.0000000
                                   /  0.0000   0.0000   1.0000 \
Orientation Matrix               : |  0.0000   1.0000   0.0000 |
                                   \ -1.0000   0.0000   0.0000 /
```

```
  Maximum difference in pairwise tests  :    4.8346515679753566E-010
  Maximum difference in triplet tests  :    4.8346520658749818E-010

 ------------------------------------------------------
Euler angles [deg]               : 360.0000000   90.0000000   90.0000000
Axis angle pair [n; angle (deg)] :   0.5773503   -0.5773503    0.5773503 ;  120.0000000
Rodrigues vector                 :       1.0000000       -1.0000000        1.0000000
Homochoric vector                :   0.5617842   -0.5617842    0.5617842
Cubochoric vector                :   0.7842653   -0.7842653    0.7842653
Quaternion [scalar, vector]      :   0.5000000    0.5000000   -0.5000000    0.5000000
                                   /  0.0000   0.0000   1.0000 \
Orientation Matrix               : | -1.0000   0.0000   0.0000 |
                                   \  0.0000  -1.0000   0.0000 /


  Maximum difference in pairwise tests  :    1.3439116486324565E-010
  Maximum difference in triplet tests  :    1.3439205304166535E-010

 ------------------------------------------------------
Euler angles [deg]               :   0.0000000  180.0000000   90.0000000
Axis angle pair [n; angle (deg)] :   0.7071068   -0.7071068    0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.7071068       -0.7071068        0.0000000 ;        Infinity
Homochoric vector                :   0.9409258   -0.9409258    0.0000000
Cubochoric vector                :   1.0725147   -1.0725147    0.0000000
Quaternion [scalar, vector]      :   0.0000000    0.7071068   -0.7071068    0.0000000
                                   /  0.0000  -1.0000   0.0000 \
Orientation Matrix               : | -1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    1.9317880628477724E-014
  Maximum difference in triplet tests  :    1.9317880628477724E-014

 ------------------------------------------------------
Euler angles [deg]               :  90.0000000  180.0000000   90.0000000
Axis angle pair [n; angle (deg)] :   1.0000000    0.0000000    0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       1.0000000        0.0000000        0.0000000 ;        Infinity
Homochoric vector                :   1.3306700    0.0000000    0.0000000
Cubochoric vector                :   1.0725147    0.0000000    0.0000000
Quaternion [scalar, vector]      :   0.0000000    1.0000000    0.0000000    0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000  -1.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    2.2204460492503131E-016
  Maximum difference in triplet tests  :    2.2204460492503131E-016

 ------------------------------------------------------
Euler angles [deg]               : 180.0000000  180.0000000   90.0000000
Axis angle pair [n; angle (deg)] :  -0.7071068   -0.7071068   -0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :      -0.7071068       -0.7071068       -0.0000000 ;        Infinity
Homochoric vector                :  -0.9409258   -0.9409258   -0.0000000
Cubochoric vector                :  -1.0725147   -1.0725147   -0.0000000
Quaternion [scalar, vector]      :   0.0000000   -0.7071068   -0.7071068   -0.0000000
                                   /  0.0000   1.0000   0.0000 \
Orientation Matrix               : |  1.0000   0.0000   0.0000 |
```

```
                                \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    4.4408920985006262E-016
  Maximum difference in triplet tests  :    6.6613381477509392E-016

 ----------------------------------------------------
Euler angles [deg]               : 270.0000000  180.0000000   90.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000   -1.0000000   -0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :      -0.0000000       -1.0000000       -0.0000000 ;        Infinity
Homochoric vector                :  -0.0000000   -1.3306700   -0.0000000
Cubochoric vector                :  -0.0000000   -1.0725147   -0.0000000
Quaternion [scalar, vector]      :   0.0000000   -0.0000000   -1.0000000   -0.0000000
                                  / -1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   1.0000   0.0000 |
                                  \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    3.1579718625741035E-016
  Maximum difference in triplet tests  :    3.1579718625741035E-016

 ----------------------------------------------------
Euler angles [deg]               : 360.0000000  180.0000000   90.0000000
Axis angle pair [n; angle (deg)] :   0.7071068   -0.7071068    0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.7071068       -0.7071068        0.0000000 ;        Infinity
Homochoric vector                :   0.9409258   -0.9409258    0.0000000
Cubochoric vector                :   1.0725147   -1.0725147    0.0000000
Quaternion [scalar, vector]      :   0.0000000    0.7071068   -0.7071068    0.0000000
                                  /  0.0000  -1.0000   0.0000 \
Orientation Matrix               : | -1.0000   0.0000   0.0000 |
                                  \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    1.9317880628477724E-014
  Maximum difference in triplet tests  :    1.9317880628477724E-014

 ----------------------------------------------------
Euler angles [deg]               :   0.0000000    0.0000000  180.0000000
Axis angle pair [n; angle (deg)] :   0.0000000   -0.0000000    1.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.0000000       -0.0000000        1.0000000 ;        Infinity
Homochoric vector                :   0.0000000   -0.0000000    1.3306700
Cubochoric vector                :   0.0000000    0.0000000    1.0725147
Quaternion [scalar, vector]      :   0.0000000    0.0000000   -0.0000000    1.0000000
                                  / -1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000  -1.0000   0.0000 |
                                  \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    2.2204460492503131E-016
  Maximum difference in triplet tests  :    2.2204460492503131E-016

 ----------------------------------------------------
Euler angles [deg]               :  90.0000000    0.0000000  180.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000    0.0000000   -1.0000000 ;   90.0000000
Rodrigues vector                 :      -0.0000000        0.0000000       -1.0000000
Homochoric vector                :  -0.0000000    0.0000000   -0.7536693
Cubochoric vector                :   0.0000000    0.0000000   -0.6074544
Quaternion [scalar, vector]      :   0.7071068   -0.0000000    0.0000000   -0.7071068
                                  /  0.0000  -1.0000   0.0000 \
```

```
Orientation Matrix               : |  1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    4.8346526781983812E-010
  Maximum difference in triplet tests  :    4.8346526781983812E-010

 ---------------------------------------------------
Euler angles [deg]               : 180.0000000    0.0000000  180.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000   -0.0000000   -1.0000000 ;   -0.0000000
Rodrigues vector                 :       0.0000000        0.0000000        0.0000000
Homochoric vector                :  -0.0000000   -0.0000000   -0.0000000
Cubochoric vector                :   0.0000000    0.0000000    0.0000000
Quaternion [scalar, vector]      :   1.0000000   -0.0000000   -0.0000000   -0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   1.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    1.2246467991473532E-016
  Maximum difference in triplet tests  :    7.2278920375157370E-015

 ---------------------------------------------------
Euler angles [deg]               : 270.0000000    0.0000000  180.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000   -0.0000000    1.0000000 ;   90.0000000
Rodrigues vector                 :      -0.0000000       -0.0000000        1.0000000
Homochoric vector                :  -0.0000000   -0.0000000    0.7536693
Cubochoric vector                :   0.0000000    0.0000000    0.6074544
Quaternion [scalar, vector]      :   0.7071068   -0.0000000   -0.0000000    0.7071068
                                   /  0.0000   1.0000   0.0000 \
Orientation Matrix               : | -1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    4.8346537884214058E-010
  Maximum difference in triplet tests  :    4.8346537884214058E-010

 ---------------------------------------------------
Euler angles [deg]               : 360.0000000    0.0000000  180.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000   -0.0000000    1.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :      -0.0000000       -0.0000000        1.0000000 ;        Infinity
Homochoric vector                :  -0.0000000   -0.0000000    1.3306700
Cubochoric vector                :   0.0000000    0.0000000    1.0725147
Quaternion [scalar, vector]      :   0.0000000   -0.0000000   -0.0000000    1.0000000
                                   / -1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000  -1.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    2.2204460492503131E-016
  Maximum difference in triplet tests  :    4.4408920985006262E-016

 ---------------------------------------------------
Euler angles [deg]               :   0.0000000   90.0000000  180.0000000
Axis angle pair [n; angle (deg)] :   0.0000000   -0.7071068    0.7071068 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.0000000       -0.7071068        0.7071068 ;        Infinity
Homochoric vector                :   0.0000000   -0.9409258    0.9409258
Cubochoric vector                :   0.0000000   -1.0725147    1.0725147
Quaternion [scalar, vector]      :   0.0000000    0.0000000   -0.7071068    0.7071068
```

```
                                / -1.0000   0.0000   0.0000 \
Orientation Matrix              : |  0.0000   0.0000  -1.0000 |
                                \  0.0000  -1.0000   0.0000 /


  Maximum difference in pairwise tests  :    6.6856799808063715E-015
  Maximum difference in triplet tests  :    6.6856799808063715E-015

 ----------------------------------------------------
Euler angles [deg]              :  90.0000000   90.0000000  180.0000000
Axis angle pair [n; angle (deg)] :  -0.5773503    0.5773503   -0.5773503 ;  120.0000000
Rodrigues vector                :      -1.0000000        1.0000000       -1.0000000
Homochoric vector               :  -0.5617842    0.5617842   -0.5617842
Cubochoric vector               :  -0.7842653    0.7842653   -0.7842653
Quaternion [scalar, vector]     :   0.5000000   -0.5000000    0.5000000   -0.5000000
                                /  0.0000  -1.0000   0.0000 \
Orientation Matrix              : |  0.0000   0.0000  -1.0000 |
                                \  1.0000   0.0000   0.0000 /


  Maximum difference in pairwise tests  :    1.3440182300428205E-010
  Maximum difference in triplet tests  :    1.3440182300428205E-010

 ----------------------------------------------------
Euler angles [deg]              : 180.0000000   90.0000000  180.0000000
Axis angle pair [n; angle (deg)] :  -1.0000000   -0.0000000   -0.0000000 ;   90.0000000
Rodrigues vector                :      -1.0000000       -0.0000000       -0.0000000
Homochoric vector               :  -0.7536693   -0.0000000   -0.0000000
Cubochoric vector               :  -0.6074544    0.0000000   -0.0000000
Quaternion [scalar, vector]     :   0.7071068   -0.7071068   -0.0000000   -0.0000000
                                /  1.0000   0.0000   0.0000 \
Orientation Matrix              : |  0.0000   0.0000  -1.0000 |
                                \  0.0000   1.0000   0.0000 /


  Maximum difference in pairwise tests  :    4.8346515679753566E-010
  Maximum difference in triplet tests  :    4.8346520658749818E-010

 ----------------------------------------------------
Euler angles [deg]              : 270.0000000   90.0000000  180.0000000
Axis angle pair [n; angle (deg)] :  -0.5773503   -0.5773503    0.5773503 ;  120.0000000
Rodrigues vector                :      -1.0000000       -1.0000000        1.0000000
Homochoric vector               :  -0.5617842   -0.5617842    0.5617842
Cubochoric vector               :  -0.7842653   -0.7842653    0.7842653
Quaternion [scalar, vector]     :   0.5000000   -0.5000000   -0.5000000    0.5000000
                                /  0.0000   1.0000   0.0000 \
Orientation Matrix              : |  0.0000   0.0000  -1.0000 |
                                \ -1.0000   0.0000   0.0000 /


  Maximum difference in pairwise tests  :    1.3439116486324565E-010
  Maximum difference in triplet tests  :    1.3439205304166535E-010

 ----------------------------------------------------
Euler angles [deg]              : 360.0000000   90.0000000  180.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000   -0.7071068    0.7071068 ;  180.0000000
Rodrigues vector [n; tan(w/2)]  :      -0.0000000       -0.7071068        0.7071068 ;        Infinity
Homochoric vector               :  -0.0000000   -0.9409258    0.9409258
Cubochoric vector               :  -0.0000000   -1.0725147    1.0725147
```

```
Quaternion [scalar, vector]      :   0.0000000   -0.0000000   -0.7071068    0.7071068
                                   / -1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   0.0000  -1.0000 |
                                   \  0.0000  -1.0000   0.0000 /


  Maximum difference in pairwise tests  :    6.6856799808063715E-015
  Maximum difference in triplet tests  :    6.6856799808063715E-015

 ----------------------------------------------------
Euler angles [deg]               :   0.0000000  180.0000000  180.0000000
Axis angle pair [n; angle (deg)] :   0.0000000   -1.0000000    0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.0000000       -1.0000000        0.0000000 ;        Infinity
Homochoric vector                :   0.0000000   -1.3306700    0.0000000
Cubochoric vector                :   0.0000000   -1.0725147    0.0000000
Quaternion [scalar, vector]      :   0.0000000    0.0000000   -1.0000000    0.0000000
                                   / -1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   1.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    1.7237823112685768E-014
  Maximum difference in triplet tests  :    1.7237823112685768E-014

 ----------------------------------------------------
Euler angles [deg]               :  90.0000000  180.0000000  180.0000000
Axis angle pair [n; angle (deg)] :  -0.7071068    0.7071068   -0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :      -0.7071068        0.7071068       -0.0000000 ;        Infinity
Homochoric vector                :  -0.9409258    0.9409258   -0.0000000
Cubochoric vector                :  -1.0725147    1.0725147   -0.0000000
Quaternion [scalar, vector]      :   0.0000000   -0.7071068    0.7071068   -0.0000000
                                   /  0.0000  -1.0000   0.0000 \
Orientation Matrix               : | -1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    1.9317880628477724E-014
  Maximum difference in triplet tests  :    1.9317880628477724E-014

 ----------------------------------------------------
Euler angles [deg]               : 180.0000000  180.0000000  180.0000000
Axis angle pair [n; angle (deg)] :  -1.0000000   -0.0000000   -0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :      -1.0000000       -0.0000000       -0.0000000 ;        Infinity
Homochoric vector                :  -1.3306700   -0.0000000   -0.0000000
Cubochoric vector                :  -1.0725147    0.0000000   -0.0000000
Quaternion [scalar, vector]      :   0.0000000   -1.0000000   -0.0000000   -0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000  -1.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    9.9568920889096810E-015
  Maximum difference in triplet tests  :    9.9568920889096810E-015

 ----------------------------------------------------
Euler angles [deg]               : 270.0000000  180.0000000  180.0000000
Axis angle pair [n; angle (deg)] :  -0.7071068   -0.7071068    0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :      -0.7071068       -0.7071068        0.0000000 ;        Infinity
Homochoric vector                :  -0.9409258   -0.9409258    0.0000000
```

```
Cubochoric vector               :  -1.0725147   -1.0725147    0.0000000
Quaternion [scalar, vector]     :   0.0000000   -0.7071068   -0.7071068    0.0000000
                                  /  0.0000   1.0000   0.0000 \
Orientation Matrix              : |  1.0000   0.0000   0.0000 |
                                  \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    4.4408920985006262E-016
  Maximum difference in triplet tests  :    6.6613381477509392E-016

 ----------------------------------------------------
Euler angles [deg]              : 360.0000000  180.0000000  180.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000   -1.0000000    0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]  :      -0.0000000       -1.0000000        0.0000000 ;        Infinity
Homochoric vector               :  -0.0000000   -1.3306700    0.0000000
Cubochoric vector               :  -0.0000000   -1.0725147    0.0000000
Quaternion [scalar, vector]     :   0.0000000   -0.0000000   -1.0000000    0.0000000
                                  / -1.0000   0.0000   0.0000 \
Orientation Matrix              : |  0.0000   1.0000   0.0000 |
                                  \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    2.2204460492503131E-016
  Maximum difference in triplet tests  :    2.2204460492503131E-016

 ----------------------------------------------------
Euler angles [deg]              :   0.0000000    0.0000000  270.0000000
Axis angle pair [n; angle (deg)] :   0.0000000    0.0000000   -1.0000000 ;   90.0000000
Rodrigues vector                :       0.0000000        0.0000000       -1.0000000
Homochoric vector               :   0.0000000    0.0000000   -0.7536693
Cubochoric vector               :   0.0000000    0.0000000   -0.6074544
Quaternion [scalar, vector]     :   0.7071068    0.0000000    0.0000000   -0.7071068
                                  /  0.0000  -1.0000   0.0000 \
Orientation Matrix              : |  1.0000   0.0000   0.0000 |
                                  \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    4.8346526781983812E-010
  Maximum difference in triplet tests  :    4.8346526781983812E-010

 ----------------------------------------------------
Euler angles [deg]              :  90.0000000    0.0000000  270.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000   -0.0000000   -1.0000000 ;   -0.0000000
Rodrigues vector                :       0.0000000        0.0000000        0.0000000
Homochoric vector               :  -0.0000000   -0.0000000   -0.0000000
Cubochoric vector               :   0.0000000    0.0000000    0.0000000
Quaternion [scalar, vector]     :   1.0000000   -0.0000000    0.0000000   -0.0000000
                                  /  1.0000   0.0000   0.0000 \
Orientation Matrix              : |  0.0000   1.0000   0.0000 |
                                  \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    1.2246467991473532E-016
  Maximum difference in triplet tests  :    7.2278920375157370E-015

 ----------------------------------------------------
Euler angles [deg]              : 180.0000000    0.0000000  270.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000    0.0000000    1.0000000 ;   90.0000000
Rodrigues vector                :      -0.0000000        0.0000000        1.0000000
```

```
Homochoric vector               :  -0.0000000    0.0000000    0.7536693
Cubochoric vector               :   0.0000000    0.0000000    0.6074544
Quaternion [scalar, vector]     :   0.7071068   -0.0000000    0.0000000    0.7071068
                                  /  0.0000   1.0000   0.0000 \
Orientation Matrix              : | -1.0000   0.0000   0.0000 |
                                  \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    4.8346537884214058E-010
  Maximum difference in triplet tests   :    4.8346537884214058E-010

 ----------------------------------------------------
Euler angles [deg]              : 270.0000000    0.0000000  270.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000   -0.0000000    1.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]  :      -0.0000000       -0.0000000        1.0000000 ;        Infinity
Homochoric vector               :  -0.0000000   -0.0000000    1.3306700
Cubochoric vector               :   0.0000000    0.0000000    1.0725147
Quaternion [scalar, vector]     :   0.0000000   -0.0000000   -0.0000000    1.0000000
                                  / -1.0000   0.0000   0.0000 \
Orientation Matrix              : |  0.0000  -1.0000   0.0000 |
                                  \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    2.2204460492503131E-016
  Maximum difference in triplet tests   :    4.4408920985006262E-016

 ----------------------------------------------------
Euler angles [deg]              : 360.0000000    0.0000000  270.0000000
Axis angle pair [n; angle (deg)] :   0.0000000    0.0000000   -1.0000000 ;   90.0000000
Rodrigues vector                :       0.0000000        0.0000000       -1.0000000
Homochoric vector               :   0.0000000    0.0000000   -0.7536693
Cubochoric vector               :   0.0000000    0.0000000   -0.6074544
Quaternion [scalar, vector]     :   0.7071068    0.0000000    0.0000000   -0.7071068
                                  /  0.0000  -1.0000   0.0000 \
Orientation Matrix              : |  1.0000   0.0000   0.0000 |
                                  \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    4.8346526781983812E-010
  Maximum difference in triplet tests   :    4.8346526781983812E-010

 ----------------------------------------------------
Euler angles [deg]              :   0.0000000   90.0000000  270.0000000
Axis angle pair [n; angle (deg)] :   0.5773503    0.5773503   -0.5773503 ;  120.0000000
Rodrigues vector                :       1.0000000        1.0000000       -1.0000000
Homochoric vector               :   0.5617842    0.5617842   -0.5617842
Cubochoric vector               :   0.7842653    0.7842653   -0.7842653
Quaternion [scalar, vector]     :   0.5000000    0.5000000    0.5000000   -0.5000000
                                  /  0.0000   0.0000  -1.0000 \
Orientation Matrix              : |  1.0000   0.0000   0.0000 |
                                  \  0.0000  -1.0000   0.0000 /


  Maximum difference in pairwise tests  :    1.3440182300428205E-010
  Maximum difference in triplet tests   :    1.3442225110793515E-010

 ----------------------------------------------------
Euler angles [deg]              :  90.0000000   90.0000000  270.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000    1.0000000   -0.0000000 ;   90.0000000
```

```
Rodrigues vector                :      -0.0000000        1.0000000       -0.0000000
Homochoric vector               :  -0.0000000    0.7536693   -0.0000000
Cubochoric vector               :  -0.0000000    0.6074544   -0.0000000
Quaternion [scalar, vector]     :   0.7071068   -0.0000000    0.7071068   -0.0000000
                                  /  0.0000   0.0000  -1.0000 \
Orientation Matrix              : |  0.0000   1.0000   0.0000 |
                                  \  1.0000   0.0000   0.0000 /


  Maximum difference in pairwise tests  :    4.8346515679753566E-010
  Maximum difference in triplet tests  :    4.8346520658749818E-010

 ------------------------------------------------------
Euler angles [deg]              : 180.0000000   90.0000000  270.0000000
Axis angle pair [n; angle (deg)] :  -0.5773503    0.5773503    0.5773503 ;  120.0000000
Rodrigues vector                :      -1.0000000        1.0000000        1.0000000
Homochoric vector               :  -0.5617842    0.5617842    0.5617842
Cubochoric vector               :  -0.7842653    0.7842653    0.7842653
Quaternion [scalar, vector]     :   0.5000000   -0.5000000    0.5000000    0.5000000
                                  /  0.0000   0.0000  -1.0000 \
Orientation Matrix              : | -1.0000   0.0000   0.0000 |
                                  \  0.0000   1.0000   0.0000 /


  Maximum difference in pairwise tests  :    1.3439116486324565E-010
  Maximum difference in triplet tests  :    1.3439205304166535E-010

 ------------------------------------------------------
Euler angles [deg]              : 270.0000000   90.0000000  270.0000000
Axis angle pair [n; angle (deg)] :  -0.7071068   -0.0000000    0.7071068 ;  180.0000000
Rodrigues vector [n; tan(w/2)]  :      -0.7071068       -0.0000000        0.7071068 ;        Infinity
Homochoric vector               :  -0.9409258   -0.0000000    0.9409258
Cubochoric vector               :  -1.0725147    0.0000000    1.0725147
Quaternion [scalar, vector]     :   0.0000000   -0.7071068   -0.0000000    0.7071068
                                  /  0.0000   0.0000  -1.0000 \
Orientation Matrix              : |  0.0000  -1.0000   0.0000 |
                                  \ -1.0000   0.0000   0.0000 /


  Maximum difference in pairwise tests  :    4.4408920985006262E-016
  Maximum difference in triplet tests  :    1.9356228213530656E-014

 ------------------------------------------------------
Euler angles [deg]              : 360.0000000   90.0000000  270.0000000
Axis angle pair [n; angle (deg)] :   0.5773503    0.5773503   -0.5773503 ;  120.0000000
Rodrigues vector                :       1.0000000        1.0000000       -1.0000000
Homochoric vector               :   0.5617842    0.5617842   -0.5617842
Cubochoric vector               :   0.7842653    0.7842653   -0.7842653
Quaternion [scalar, vector]     :   0.5000000    0.5000000    0.5000000   -0.5000000
                                  /  0.0000   0.0000  -1.0000 \
Orientation Matrix              : |  1.0000   0.0000   0.0000 |
                                  \  0.0000  -1.0000   0.0000 /


  Maximum difference in pairwise tests  :    1.3439116486324565E-010
  Maximum difference in triplet tests  :    1.3441758817123173E-010

 ------------------------------------------------------
Euler angles [deg]              :   0.0000000  180.0000000  270.0000000
```

```
Axis angle pair [n; angle (deg)] :   0.7071068    0.7071068   -0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.7071068        0.7071068       -0.0000000 ;        Infinity
Homochoric vector                :   0.9409258    0.9409258   -0.0000000
Cubochoric vector                :   1.0725147    1.0725147   -0.0000000
Quaternion [scalar, vector]      :   0.0000000    0.7071068    0.7071068   -0.0000000
                                   /  0.0000   1.0000   0.0000 \
Orientation Matrix               : |  1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    6.6613381477509392E-016
  Maximum difference in triplet tests  :    6.6613381477509392E-016

 ------------------------------------------------------
Euler angles [deg]               :  90.0000000  180.0000000  270.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000    1.0000000   -0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :      -0.0000000        1.0000000       -0.0000000 ;        Infinity
Homochoric vector                :  -0.0000000    1.3306700   -0.0000000
Cubochoric vector                :  -0.0000000    1.0725147   -0.0000000
Quaternion [scalar, vector]      :   0.0000000   -0.0000000    1.0000000   -0.0000000
                                   / -1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   1.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    1.7253034593166964E-014
  Maximum difference in triplet tests  :    1.7253034593166964E-014

 ------------------------------------------------------
Euler angles [deg]               : 180.0000000  180.0000000  270.0000000
Axis angle pair [n; angle (deg)] :  -0.7071068    0.7071068    0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :      -0.7071068        0.7071068        0.0000000 ;        Infinity
Homochoric vector                :  -0.9409258    0.9409258    0.0000000
Cubochoric vector                :  -1.0725147    1.0725147    0.0000000
Quaternion [scalar, vector]      :   0.0000000   -0.7071068    0.7071068    0.0000000
                                   /  0.0000  -1.0000   0.0000 \
Orientation Matrix               : | -1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    1.9317880628477724E-014
  Maximum difference in triplet tests  :    1.9317880628477724E-014

 ------------------------------------------------------
Euler angles [deg]               : 270.0000000  180.0000000  270.0000000
Axis angle pair [n; angle (deg)] :  -1.0000000   -0.0000000    0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :      -1.0000000       -0.0000000        0.0000000 ;        Infinity
Homochoric vector                :  -1.3306700   -0.0000000    0.0000000
Cubochoric vector                :  -1.0725147    0.0000000    0.0000000
Quaternion [scalar, vector]      :   0.0000000   -1.0000000   -0.0000000    0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000  -1.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    2.2204460492503131E-016
  Maximum difference in triplet tests  :    2.2204460492503131E-016

 ------------------------------------------------------
```

```
Euler angles [deg]               : 360.0000000  180.0000000  270.0000000
Axis angle pair [n; angle (deg)] :   0.7071068    0.7071068   -0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.7071068        0.7071068       -0.0000000 ;        Infinity
Homochoric vector                :   0.9409258    0.9409258   -0.0000000
Cubochoric vector                :   1.0725147    1.0725147   -0.0000000
Quaternion [scalar, vector]      :   0.0000000    0.7071068    0.7071068   -0.0000000
                                   /  0.0000   1.0000   0.0000 \
Orientation Matrix               : |  1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    4.4408920985006262E-016
  Maximum difference in triplet tests  :    4.4408920985006262E-016

 ------------------------------------------------
Euler angles [deg]               :   0.0000000    0.0000000  360.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000   -0.0000000   -1.0000000 ;   -0.0000000
Rodrigues vector                 :       0.0000000        0.0000000        0.0000000
Homochoric vector                :  -0.0000000   -0.0000000   -0.0000000
Cubochoric vector                :   0.0000000    0.0000000    0.0000000
Quaternion [scalar, vector]      :   1.0000000    0.0000000    0.0000000   -0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   1.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    1.2246467991473532E-016
  Maximum difference in triplet tests  :    7.2278920375157370E-015

 ------------------------------------------------
Euler angles [deg]               :  90.0000000    0.0000000  360.0000000
Axis angle pair [n; angle (deg)] :   0.0000000    0.0000000    1.0000000 ;   90.0000000
Rodrigues vector                 :       0.0000000        0.0000000        1.0000000
Homochoric vector                :   0.0000000    0.0000000    0.7536693
Cubochoric vector                :   0.0000000    0.0000000    0.6074544
Quaternion [scalar, vector]      :   0.7071068    0.0000000    0.0000000    0.7071068
                                   /  0.0000   1.0000   0.0000 \
Orientation Matrix               : | -1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    4.8346537884214058E-010
  Maximum difference in triplet tests  :    4.8346537884214058E-010

 ------------------------------------------------
Euler angles [deg]               : 180.0000000    0.0000000  360.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000    0.0000000    1.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :      -0.0000000        0.0000000        1.0000000 ;        Infinity
Homochoric vector                :  -0.0000000    0.0000000    1.3306700
Cubochoric vector                :   0.0000000    0.0000000    1.0725147
Quaternion [scalar, vector]      :   0.0000000   -0.0000000    0.0000000    1.0000000
                                   / -1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000  -1.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    2.2204460492503131E-016
  Maximum difference in triplet tests  :    4.4408920985006262E-016
```

```
 ------------------------------------------------------
Euler angles [deg]               : 270.0000000    0.0000000  360.0000000
Axis angle pair [n; angle (deg)] :   0.0000000   -0.0000000   -1.0000000 ;   90.0000000
Rodrigues vector                 :       0.0000000       -0.0000000       -1.0000000
Homochoric vector                :   0.0000000   -0.0000000   -0.7536693
Cubochoric vector                :   0.0000000    0.0000000   -0.6074544
Quaternion [scalar, vector]      :   0.7071068    0.0000000   -0.0000000   -0.7071068
                                   /  0.0000  -1.0000   0.0000 \
Orientation Matrix               : |  1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    4.8346526781983812E-010
  Maximum difference in triplet tests  :    4.8346526781983812E-010

 ------------------------------------------------------
Euler angles [deg]               : 360.0000000    0.0000000  360.0000000
Axis angle pair [n; angle (deg)] :   0.0000000    0.0000000    1.0000000 ;    0.0000000
Rodrigues vector                 :       0.0000000        0.0000000        0.0000000
Homochoric vector                :   0.0000000    0.0000000    0.0000000
Cubochoric vector                :   0.0000000    0.0000000    0.0000000
Quaternion [scalar, vector]      :   1.0000000    0.0000000    0.0000000   -0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   1.0000   0.0000 |
                                   \  0.0000   0.0000   1.0000 /


  Maximum difference in pairwise tests  :    2.4492935982947064E-016
  Maximum difference in triplet tests  :    7.3503567174304722E-015

 ------------------------------------------------------
Euler angles [deg]               :   0.0000000   90.0000000  360.0000000
Axis angle pair [n; angle (deg)] :   1.0000000    0.0000000   -0.0000000 ;   90.0000000
Rodrigues vector                 :       1.0000000        0.0000000       -0.0000000
Homochoric vector                :   0.7536693    0.0000000   -0.0000000
Cubochoric vector                :   0.6074544    0.0000000   -0.0000000
Quaternion [scalar, vector]      :   0.7071068    0.7071068    0.0000000   -0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   0.0000   1.0000 |
                                   \  0.0000  -1.0000   0.0000 /


  Maximum difference in pairwise tests  :    4.8346515679753566E-010
  Maximum difference in triplet tests  :    4.8346520658749818E-010

 ------------------------------------------------------
Euler angles [deg]               :  90.0000000   90.0000000  360.0000000
Axis angle pair [n; angle (deg)] :   0.5773503    0.5773503    0.5773503 ;  120.0000000
Rodrigues vector                 :       1.0000000        1.0000000        1.0000000
Homochoric vector                :   0.5617842    0.5617842    0.5617842
Cubochoric vector                :   0.7842653    0.7842653    0.7842653
Quaternion [scalar, vector]      :   0.5000000    0.5000000    0.5000000    0.5000000
                                   /  0.0000   1.0000   0.0000 \
Orientation Matrix               : |  0.0000   0.0000   1.0000 |
                                   \  1.0000   0.0000   0.0000 /


  Maximum difference in pairwise tests  :    1.3439116486324565E-010
  Maximum difference in triplet tests  :    1.3439205304166535E-010
```

```
 ----------------------------------------------------
Euler angles [deg]               : 180.0000000   90.0000000  360.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000    0.7071068    0.7071068 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :      -0.0000000        0.7071068        0.7071068 ;        Infinity
Homochoric vector                :  -0.0000000    0.9409258    0.9409258
Cubochoric vector                :  -0.0000000    1.0725147    1.0725147
Quaternion [scalar, vector]      :   0.0000000   -0.0000000    0.7071068    0.7071068
                                   / -1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   0.0000   1.0000 |
                                   \  0.0000   1.0000   0.0000 /


  Maximum difference in pairwise tests  :    6.6856799808063715E-015
  Maximum difference in triplet tests  :    6.6856799808063715E-015

 ----------------------------------------------------
Euler angles [deg]               : 270.0000000   90.0000000  360.0000000
Axis angle pair [n; angle (deg)] :   0.5773503   -0.5773503   -0.5773503 ;  120.0000000
Rodrigues vector                 :       1.0000000       -1.0000000       -1.0000000
Homochoric vector                :   0.5617842   -0.5617842   -0.5617842
Cubochoric vector                :   0.7842653   -0.7842653   -0.7842653
Quaternion [scalar, vector]      :   0.5000000    0.5000000   -0.5000000   -0.5000000
                                   /  0.0000  -1.0000   0.0000 \
Orientation Matrix               : |  0.0000   0.0000   1.0000 |
                                   \ -1.0000   0.0000   0.0000 /


  Maximum difference in pairwise tests  :    1.3439116486324565E-010
  Maximum difference in triplet tests  :    1.3441758817123173E-010

 ----------------------------------------------------
Euler angles [deg]               : 360.0000000   90.0000000  360.0000000
Axis angle pair [n; angle (deg)] :   1.0000000    0.0000000   -0.0000000 ;   90.0000000
Rodrigues vector                 :       1.0000000        0.0000000       -0.0000000
Homochoric vector                :   0.7536693    0.0000000   -0.0000000
Cubochoric vector                :   0.6074544    0.0000000   -0.0000000
Quaternion [scalar, vector]      :   0.7071068    0.7071068    0.0000000   -0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   0.0000   1.0000 |
                                   \  0.0000  -1.0000   0.0000 /


  Maximum difference in pairwise tests  :    4.8346515679753566E-010
  Maximum difference in triplet tests  :    4.8346520658749818E-010

 ----------------------------------------------------
Euler angles [deg]               :   0.0000000  180.0000000  360.0000000
Axis angle pair [n; angle (deg)] :   1.0000000    0.0000000   -0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       1.0000000        0.0000000       -0.0000000 ;        Infinity
Homochoric vector                :   1.3306700    0.0000000   -0.0000000
Cubochoric vector                :   1.0725147    0.0000000   -0.0000000
Quaternion [scalar, vector]      :   0.0000000    1.0000000    0.0000000   -0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000  -1.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    2.2204460492503131E-016
```

```
  Maximum difference in triplet tests  :    2.2204460492503131E-016

 ----------------------------------------------------
Euler angles [deg]               :  90.0000000  180.0000000  360.0000000
Axis angle pair [n; angle (deg)] :   0.7071068    0.7071068    0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.7071068        0.7071068        0.0000000 ;        Infinity
Homochoric vector                :   0.9409258    0.9409258    0.0000000
Cubochoric vector                :   1.0725147    1.0725147    0.0000000
Quaternion [scalar, vector]      :   0.0000000    0.7071068    0.7071068    0.0000000
                                   /  0.0000   1.0000   0.0000 \
Orientation Matrix               : |  1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    6.6613381477509392E-016
  Maximum difference in triplet tests  :    6.6613381477509392E-016

 ----------------------------------------------------
Euler angles [deg]               : 180.0000000  180.0000000  360.0000000
Axis angle pair [n; angle (deg)] :  -0.0000000    1.0000000    0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :      -0.0000000        1.0000000        0.0000000 ;        Infinity
Homochoric vector                :  -0.0000000    1.3306700    0.0000000
Cubochoric vector                :  -0.0000000    1.0725147    0.0000000
Quaternion [scalar, vector]      :   0.0000000   -0.0000000    1.0000000    0.0000000
                                   / -1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000   1.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    1.7237823112685768E-014
  Maximum difference in triplet tests  :    1.7237823112685768E-014

 ----------------------------------------------------
Euler angles [deg]               : 270.0000000  180.0000000  360.0000000
Axis angle pair [n; angle (deg)] :   0.7071068   -0.7071068   -0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       0.7071068       -0.7071068       -0.0000000 ;        Infinity
Homochoric vector                :   0.9409258   -0.9409258   -0.0000000
Cubochoric vector                :   1.0725147   -1.0725147   -0.0000000
Quaternion [scalar, vector]      :   0.0000000    0.7071068   -0.7071068   -0.0000000
                                   /  0.0000  -1.0000   0.0000 \
Orientation Matrix               : | -1.0000   0.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /


  Maximum difference in pairwise tests  :    1.9317880628477724E-014
  Maximum difference in triplet tests  :    1.9317880628477724E-014

 ----------------------------------------------------
Euler angles [deg]               : 360.0000000  180.0000000  360.0000000
Axis angle pair [n; angle (deg)] :   1.0000000    0.0000000   -0.0000000 ;  180.0000000
Rodrigues vector [n; tan(w/2)]   :       1.0000000        0.0000000       -0.0000000 ;        Infinity
Homochoric vector                :   1.3306700    0.0000000   -0.0000000
Cubochoric vector                :   1.0725147    0.0000000   -0.0000000
Quaternion [scalar, vector]      :   0.0000000    1.0000000    0.0000000   -0.0000000
                                   /  1.0000   0.0000   0.0000 \
Orientation Matrix               : |  0.0000  -1.0000   0.0000 |
                                   \  0.0000   0.0000  -1.0000 /
```

```
Maximum difference in pairwise tests  :    9.9568920889096810E-015
Maximum difference in triplet tests  :    9.9568920889096810E-015
```