Unlimited Release
Printed May 9, 2002

ROTATION:

A review of useful theorems involving proper orthogonal matrices referenced to three-dimensional physical space.

Rebecca M. Brannon[†] and coauthors to be determined
[†]Computational Physics and Mechanics
Sandia National Laboratories
Albuquerque, NM 87185-0820

Abstract

Useful and/or little-known theorems involving 3×3 proper orthogonal matrices are reviewed. Orthogonal matrices appear in the transformation of tensor components from one orthogonal basis to another. The distinction between an orthogonal direction cosine matrix and a rotation operation is discussed. Among the theorems and techniques presented are (1) various ways to characterize a rotation including proper orthogonal tensors, dyadics, Euler angles, axis/angle representations, series expansions, and quaternions; (2) the Euler-Rodrigues formula for converting axis and angle to a rotation tensor; (3) the distinction between rotations and reflections, along with implications for "handedness" of coordinate systems; (4) non-commutivity of sequential rotations, (5) eigenvalues and eigenvectors of a rotation; (6) the polar decomposition theorem for expressing a general deformation as a sequence of shape and volume changes in combination with pure rotations; (7) mixing rotations in Eulerian hydrocodes or interpolating rotations in discrete field approximations; (8) Rates of rotation and the difference between spin and vorticity, (9) Random rotations for simulating crystal distributions; (10) The principle of material frame indifference (PMFI); and (11) a tensor-analysis presentation of classical rigid body mechanics, including direct notation expressions for momentum and energy and the extremely compact direct notation formulation of Euler's equations (i.e., Newton's law for rigid bodies). Computer source code is provided for several rotation-related algorithms.

A draft version of this document is available at http://www.me.unm.edu/~rmbrann/gobag.html

Intentionally Left Blank

Figures

ROTATION

Rebecca Brannon

Adjunct Professor, University of New Mexico
and Senior Staff, Sandia National Laboratories

Introduction

This rotation tutorial is written for engineers, not mathematicians or physicists. Hence, all discussions are limited to ordinary 3-dimensional vectors of the type discussed in first calculus courses, and no proofs are provided except where the proof itself clarifies a concept.

This document is an informal overview of fundamental theorems and concepts involving rotation. Physically, a rotation is a reorientation of a body without changing the body's size or shape. Then the distance between any two particular points in the body remain unchanged after the rotation. As will be soon seen, a rotation may be described mathematically by a special kind of *orthogonal* tensor. By using the "special kind" proviso, we are implying that rotation tensors are always orthogonal, but orthogonal tensors are not necessarily rotations. A matrix $[Q]$ is orthogonal if and only if

$$[Q]^T[Q] = [I], \tag{0.1}$$

where $[I]$ is the identity matrix. In indicial form,

$$Q_{ki}Q_{kj} = \delta_{ij} \tag{0.2}$$

where δ_{ij} is the Kronecker delta,* and (as explained in the Appendix B) repeated indices are to be summed from 1 to 3. The matrix $[Q]$ corresponds to a rotation if its determinant is equal to $+1$.

The topics in this tutorial are arranged as follows:

- Section 1 reviews how orthogonal **direction cosine matrices** are used to transform vector and tensor components from on orthonormal basis to another.
- Section 2 points out the distinction between **rotation and coordinate transformation**. The main idea is that rotation is a matter of perspective. You can rotate the object you are looking at, while you stay still, or you can keep the object fixed while you rotate yourself. It's important to be aware of which of these perspectives applies for your problem of interest.
- Section 3 describes how to **convert an axis and angle** of rotation into a

* i.e., components of the 3×3 identity matrix. Thus δ_{ij} equals 1 if $i=j$ and 0 if $i \neq j$.

rotation tensor.

- Section 4 reiterates that a rotation is characterized by a *proper* orthogonal tensor — i.e., one having a determinate equal to +1. This section also explains that an orthogonal tensor with a determinate equal to –1 should be regarded (in general) as a rotation *in combination with* a reflection.
- Section 5 presents the representation of a rotation in terms of a unit **quaternion**. Quaternions are rather old fashioned predecessors to modern vectors. The quaternion needed to describe rotation may be regarded as a point on a four-dimensional sphere.
- Section 6 describes how any linear operator can be expressed as a sum of three dyads, and this result is specialized to rotations.
- Section 7 shows that rotations may be expressed as a (non-commuting) sequence of three rotation operations. When defining a rotation in this way, it is essential to know whether subsequent rotations are defined about fixed axes or follower axes. The first view shows the structure of the rotation matrix for **sequential rotations applied about the fixed laboratory basis**. The next view (**Euler angles**) shows the rotation that results when the sequential rotations are applied about a triad that *moves with the body.*
- Section 8 provides the series expansion representation of a rotation.
- Section 9 derives the **eigenvalues and eigenvectors** of any rotation tensor.
- Section 10 presents the **polar decomposition** theorem.
- Section 11 discusses possible solutions to the problem of **"mixing" rotations** in Eulerian physics codes or in Lagrangian codes that remap the mesh.
- Section 12 shows how the **rate of a rotation** tensor is related to the more conventional angular rotation vector.
- Section 13 shows how to generate a uniformly **random rotation** tensor, which is useful for generating grain orientations for microscale simulations. We also discuss how to find the average of a tensor over all possible (uniform) rotations of that tensor.
- Section 15 provides an elementary introduction to the principle of material frame indifference, which requires that material constitutive models must be invariant under rigid rotation.
- Section 16 specializes the laws mechanics as applied to rigid body mechanics. By applying the full power of tensor analysis, this presentation distinguishes itself from the derivations normally found in elementary dynamics textbooks. For example, with direct-notation tensor analysis, Euler's equations can be presented in a single line.
- Appendix A provides **FORTRAN subroutines** that perform most of the computations presented in this report.

This report presumes that the reader has elementary knowledge of vector and tensor analysis. The notation conventions used in this report are summarized in Appendix B.

1. Orthogonal basis & coordinate transformations

A rigid body is an idealized collection of points (continuous or discrete) for which the distance between any two points is fixed. Consequently the angle between any two material line segments is also fixed. A rigid body is capable of only uniform translation and rotation. A rotation is simply a reorientation of a body without distortion (i.e., without bending or stretching). This section focuses strictly on rotation, not translation.*

Let {X,Y,Z} be a *fixed* "laboratory" orthogonal coordinate triad, and let $\{x, y, z\}$ be an orthogonal "embedded" triad that moves with the rigid body. The two triads are taken to be initially coincident. The rotation may be described mathematically by the angles that the embedded triad makes with the fixed laboratory triad.

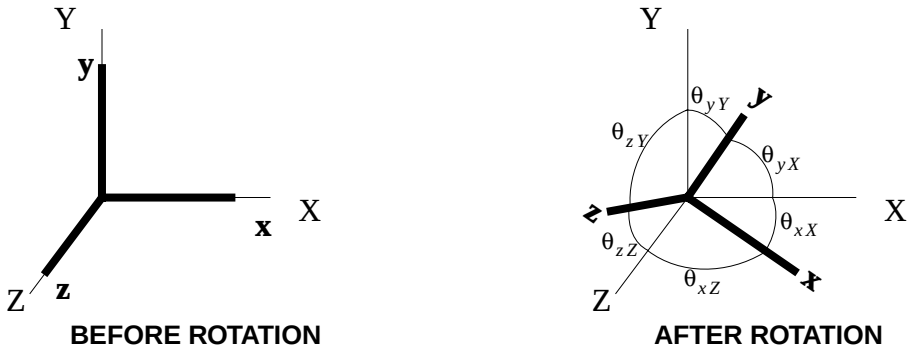


Figure 1.1. A rigid rotation. The rotation of a body may be described by the rotation of an *embedded* triad that moves with the body. Rotation may be described by the angles that the embedded triad axes make with the originally coincident laboratory axes. This figure shows six of the nine possible angles.

Motivational discussion: Principal basis. Most engineering students eventually learn that any symmetric matrix $[A]$ can be decomposed as

$$[A] = [Q]^T[D][Q], \tag{1.1}$$

where $[D]$ is a diagonal matrix containing the eigenvalues of $[A]$, and $[Q]$ is an orthogonal matrix whose columns consist of the corresponding eigenvectors of $[A]$. This theorem from matrix analysis has an interpretation in tensor analysis as a change-of-basis operation. As we will soon see, the above matrix decomposition tells us the following:

* To include translation, the laboratory triad is merely reinterpreted as a triad that translates with a particular (convenient) point in the body, but does not change orientation.

$$\text{If} \quad \underset{\sim}{\boldsymbol{A}} = \sum_{i=1}^{3}\sum_{j=1}^{3} A_{ij}\underset{\sim}{\boldsymbol{E}}_i\underset{\sim}{\boldsymbol{E}}_j \quad \text{and} \quad A_{ij} = A_{ji},$$

$$\text{then} \quad \underset{\sim}{\boldsymbol{A}} = \sum_{K=1}^{3} \lambda_K \underset{\sim}{\boldsymbol{p}}_K \underset{\sim}{\boldsymbol{p}}_K \tag{1.2}$$

where λ_K are the eigenvalues and $\underset{\sim}{\boldsymbol{p}}_K$ are the orthonormal eigenvectors. Stated differently, if a tensor $\underset{\sim}{\boldsymbol{A}}$ has components A_{ij} with respect to the laboratory $\{\underset{\sim}{\boldsymbol{E}}_1, \underset{\sim}{\boldsymbol{E}}_2, \underset{\sim}{\boldsymbol{E}}_3\}$ basis, then it has components

$$\begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \tag{1.3}$$

with respect to the $\{\underset{\sim}{\boldsymbol{p}}_1, \underset{\sim}{\boldsymbol{p}}_2, \underset{\sim}{\boldsymbol{p}}_3\}$ principal basis. This result motivates the general study of changes of basis. For example, if A_{ij} are the components of a tensor with respect to the laboratory $\{\underset{\sim}{\boldsymbol{E}}_1, \underset{\sim}{\boldsymbol{E}}_2, \underset{\sim}{\boldsymbol{E}}_3\}$ basis, then what are the components $\tilde{A}_{ij}$ with respect to some *different* $\{\underset{\sim}{\boldsymbol{e}}_1, \underset{\sim}{\boldsymbol{e}}_2, \underset{\sim}{\boldsymbol{e}}_3\}$?

Orthonormal basis transformations. In Fig. 1.1, the angle that the embedded x-axis makes with the fixed X-axis is denoted θ_{xX}, and the angle that the embedded x-axis makes with the fixed Z-axis is denoted θ_{xZ}, and so on. There are nine such angles, but only three are independent, as discussed below. The orientation of the rotated coordinate system can be described through the use of a coordinate *transformation matrix* $[L]$ defined

$$[L] = \begin{bmatrix} \cos\theta_{xX} & \cos\theta_{xY} & \cos\theta_{xZ} \\ \cos\theta_{yX} & \cos\theta_{yY} & \cos\theta_{yZ} \\ \cos\theta_{zX} & \cos\theta_{zY} & \cos\theta_{zZ} \end{bmatrix} \tag{1.4}$$

The components of $[L]$ are called *direction cosines*. Even though we have phrased these components in terms of cosines of the angles between fixed and rotated axes, it's conceptually easier to recognize that the i^{th} row of $[L]$ contains the components of the i^{th} base vector of the rotated coordinate system expressed in terms of the fixed system. We will later espouse representing the rotated orientation by using a alternative direction cosine matrix $[Q]$ equal to the transpose of $[L]$. Hence, the i^{th} *column* of $[Q]$ contains the components of the i^{th} rotated base vector. Examples will be provided soon.

Let $\{\underset{\sim}{\boldsymbol{E}}_X, \underset{\sim}{\boldsymbol{E}}_Y, \underset{\sim}{\boldsymbol{E}}_Z\}$ be the orthonormal set of "laboratory" base vectors associated with the {X,Y,Z} system. Let $\{\underset{\sim}{\boldsymbol{e}}_x, \underset{\sim}{\boldsymbol{e}}_y, \underset{\sim}{\boldsymbol{e}}_z\}$ be the orthonormal set of "laboratory" base vectors associated with the $\{x, y, z\}$ system. For any two vectors, $\underset{\sim}{\boldsymbol{u}}$ and $\underset{\sim}{\boldsymbol{v}}$, the vector inner product (single dot) can be defined

$$\underset{\sim}{\boldsymbol{u}} \bullet \underset{\sim}{\boldsymbol{v}} = |\underset{\sim}{\boldsymbol{u}}||\underset{\sim}{\boldsymbol{v}}|\cos\theta_{uv}, \tag{1.5}$$

where $|\underset{\sim}{\boldsymbol{u}}|$ and $|\underset{\sim}{\boldsymbol{v}}|$ are the magnitudes of the vectors, $\underset{\sim}{\boldsymbol{u}}$ and $\underset{\sim}{\boldsymbol{v}}$, and θ_{uv} is the angle between. Therefore, recognizing that the base vectors have unit magnitude, the direction cosine matrix may be written

$$[L] = \begin{bmatrix} \underset{\sim}{\boldsymbol{e}}_x \bullet \underset{\sim}{\boldsymbol{E}}_X & \underset{\sim}{\boldsymbol{e}}_x \bullet \underset{\sim}{\boldsymbol{E}}_Y & \underset{\sim}{\boldsymbol{e}}_x \bullet \underset{\sim}{\boldsymbol{E}}_Z \\ \underset{\sim}{\boldsymbol{e}}_y \bullet \underset{\sim}{\boldsymbol{E}}_X & \underset{\sim}{\boldsymbol{e}}_y \bullet \underset{\sim}{\boldsymbol{E}}_Y & \underset{\sim}{\boldsymbol{e}}_y \bullet \underset{\sim}{\boldsymbol{E}}_Z \\ \underset{\sim}{\boldsymbol{e}}_z \bullet \underset{\sim}{\boldsymbol{E}}_X & \underset{\sim}{\boldsymbol{e}}_z \bullet \underset{\sim}{\boldsymbol{E}}_Y & \underset{\sim}{\boldsymbol{e}}_z \bullet \underset{\sim}{\boldsymbol{E}}_Z \end{bmatrix} \tag{1.6}$$

Instead of using the symbols $\{x, y, z\}$ as subscripts, it is common practice to instead identify the coordinates by *number* $\{1, 2, 3\}$. With this convention, the above equation becomes

$$[L] = \begin{bmatrix} \underset{\sim}{\boldsymbol{e}}_1 \bullet \underset{\sim}{\boldsymbol{E}}_1 & \underset{\sim}{\boldsymbol{e}}_1 \bullet \underset{\sim}{\boldsymbol{E}}_2 & \underset{\sim}{\boldsymbol{e}}_1 \bullet \underset{\sim}{\boldsymbol{E}}_3 \\ \underset{\sim}{\boldsymbol{e}}_2 \bullet \underset{\sim}{\boldsymbol{E}}_1 & \underset{\sim}{\boldsymbol{e}}_2 \bullet \underset{\sim}{\boldsymbol{E}}_2 & \underset{\sim}{\boldsymbol{e}}_2 \bullet \underset{\sim}{\boldsymbol{E}}_3 \\ \underset{\sim}{\boldsymbol{e}}_3 \bullet \underset{\sim}{\boldsymbol{E}}_1 & \underset{\sim}{\boldsymbol{e}}_3 \bullet \underset{\sim}{\boldsymbol{E}}_2 & \underset{\sim}{\boldsymbol{e}}_3 \bullet \underset{\sim}{\boldsymbol{E}}_3 \end{bmatrix} \tag{1.7}$$

which can be written very compactly as

$$\boxed{L_{ij} = \underset{\sim}{\boldsymbol{e}}_i \bullet \underset{\sim}{\boldsymbol{E}}_j}, \tag{1.8}$$

The subscripts (called indices) vary from 1 to 3.

It will soon be shown that the $[L]$ matrix is orthogonal,* which means the inverse of $[L]$ is just the transpose of $[L]$. That is,

$$[L][L]^T = [L]^T[L] = [I] \tag{1.9}$$

Here, $[I]$ denotes the identity matrix. The above matrix equation is written in indicial form as

$$\sum_{k=1}^{3} L_{ik}L_{jk} = \sum_{k=1}^{3} L_{ki}L_{kj} = \delta_{ij} \tag{1.10}$$

where δ_{ij} is the Kronecker delta (equal to 1 if $i=j$ and 0 if $i \neq j$).

* This property holds only when both bases are orthonormal. An orthonormal basis is right-handed if crossing the first base vector into the second base vector gives the third base vector. Otherwise, if the third base vector points the opposite way, then the basis is left-handed. The determinate of $[L]$ will be +1 if both bases have the same handedness, or –1 if the bases have different handedness.

Because the laboratory basis $\{\underset{\sim}{\boldsymbol{E}}_1, \underset{\sim}{\boldsymbol{E}}_2, \underset{\sim}{\boldsymbol{E}}_3\}$ is a basis, we know that there exist coefficients v_j such that any vector $\underset{\sim}{\boldsymbol{v}}$ can be written as a linear combination of the lab base vectors. Let's denote the j^{th} component of $\underset{\sim}{\boldsymbol{v}}$ with respect to the laboratory basis by v_j. This component may be computed by dotting the vector $\underset{\sim}{\boldsymbol{v}}$ into the j^{th} base vector $\underset{\sim}{\boldsymbol{E}}_j$. Stated mathematically, there exist vector "components" v_j such that

$$\underset{\sim}{\boldsymbol{v}} = \sum_{j=1}^{3} v_j \underset{\sim}{\boldsymbol{E}}_j, \text{ where } v_j = \underset{\sim}{\boldsymbol{v}} \bullet \underset{\sim}{\boldsymbol{E}}_j \tag{1.11}$$

Equation (1.11) is true for *any* vector $\underset{\sim}{\boldsymbol{v}}$. As a special instance, we may consider the vector $\underset{\sim}{\boldsymbol{v}}$ to be one of the rotated base vectors $\underset{\sim}{\boldsymbol{e}}_i$. Then we know that each $\underset{\sim}{\boldsymbol{e}}_i$ can be expressed as a linear combination of the lab base vectors. Let's denote the component of $\underset{\sim}{\boldsymbol{e}}_i$ with respect to the j^{th} laboratory base vector by L_{ij}. Then Eq. (1.11) states that each rotated base vector $\underset{\sim}{\boldsymbol{e}}_i$ can be expressed in the form

$$\boxed{\underset{\sim}{\boldsymbol{e}}_i = \sum_{j=1}^{3} L_{ij} \underset{\sim}{\boldsymbol{E}}_j}, \text{ where } L_{ij} = \underset{\sim}{\boldsymbol{e}}_i \bullet \underset{\sim}{\boldsymbol{E}}_j \tag{1.12}$$

To remember this equation, note that the first index on L_{ij} is the same as the index on the rotated basis $\underset{\sim}{\boldsymbol{e}}_i$, and the second index on L_{ij} matches the index on the laboratory basis $\underset{\sim}{\boldsymbol{E}}_j$. Summation occurs over the repeated index j.

The rotated vectors $\{\underset{\sim}{\boldsymbol{e}}_1, \underset{\sim}{\boldsymbol{e}}_2, \underset{\sim}{\boldsymbol{e}}_3\}$ form a basis too. Consequently, we know that there exist coefficients $\tilde{v}_j$ such that any vector $\underset{\sim}{\boldsymbol{v}}$ can be written

$$\underset{\sim}{\boldsymbol{v}} = \sum_{j=1}^{3} \tilde{v}_j \underset{\sim}{\boldsymbol{e}}_j, \text{ where } \tilde{v}_j = \underset{\sim}{\boldsymbol{v}} \bullet \underset{\sim}{\boldsymbol{e}}_j \tag{1.13}$$

As a special case, we may assert the existence of coefficients λ_{ij} such that each laboratory base vector $\underset{\sim}{\boldsymbol{E}}_i$ can be written as a linear combination of the rotated basis:

$$\underset{\sim}{\boldsymbol{E}}_i = \sum_{j=1}^{3} \lambda_{ij} \underset{\sim}{\boldsymbol{e}}_j, \text{ where } \lambda_{ij} = \underset{\sim}{\boldsymbol{E}}_i \bullet \underset{\sim}{\boldsymbol{e}}_j \tag{1.14}$$

For any vectors, $\underset{\sim}{\boldsymbol{v}}$ and $\underset{\sim}{\boldsymbol{w}}$, we know that the dot product is commutative (i.e., $\underset{\sim}{\boldsymbol{v}} \bullet \underset{\sim}{\boldsymbol{w}} = \underset{\sim}{\boldsymbol{w}} \bullet \underset{\sim}{\boldsymbol{v}}$). Consequently, $\lambda_{ij} = \underset{\sim}{\boldsymbol{E}}_i \bullet \underset{\sim}{\boldsymbol{e}}_j = \underset{\sim}{\boldsymbol{e}}_j \bullet \underset{\sim}{\boldsymbol{E}}_i = L_{ji}$. Here, the final step came from applying the definition of L_{ij} in Eq. $(1.12)_2$. In matrix notation, this says that

$$[\lambda] = [L]^T. \tag{1.15}$$

Now that we know that $\lambda_{ij} = L_{ji}$, we see in hindsight that it's not really necessary to introduce a new set of coefficients λ_{ij}; the $[L]$ matrix contains all of the information needed to express one basis with respect to the other. In terms of the $[L]$ coefficients, Eq. (1.14) becomes

$$\boxed{\underset{\sim}{\boldsymbol{E}}_i = \sum_{j=1}^{3} L_{ji}\underset{\sim}{\boldsymbol{e}}_j}, \text{ where } L_{ji} = \underset{\sim}{\boldsymbol{e}}_j \bullet \underset{\sim}{\boldsymbol{E}}_i \tag{1.16}$$

As before, the first index on L_{ji} is identical to the index on the rotated basis $\underset{\sim}{\boldsymbol{e}}_j$, and the second index on L_{ji} matches the index on the laboratory basis $\underset{\sim}{\boldsymbol{E}}_i$. Summation occurs over the repeated index j.

The coordinate transformation matrix $[L]$ is often used for relating the components of vectors with respect to different bases. Suppose $\{v_1, v_2, v_3\}$ are the components of a vector $\underset{\sim}{\boldsymbol{v}}$ with respect to the fixed $\underset{\sim}{\boldsymbol{E}}_i$ laboratory basis. Then $\underset{\sim}{\boldsymbol{v}}$ can be written

$$\underset{\sim}{\boldsymbol{v}} = \sum_{i=1}^{3} v_i\underset{\sim}{\boldsymbol{E}}_i, \text{ where } v_i = \underset{\sim}{\boldsymbol{v}} \bullet \underset{\sim}{\boldsymbol{E}}_i \tag{1.17}$$

The *same* vector $\underset{\sim}{\boldsymbol{v}}$ can alternatively be written as a linear combination of the *rotated* $\underset{\sim}{\boldsymbol{e}}_j$ base vectors. If $\{\tilde{v}_1, \tilde{v}_2, \tilde{v}_3\}$ are the components of the same vector $\underset{\sim}{\boldsymbol{v}}$ with respect to the rotated $\underset{\sim}{\boldsymbol{e}}_j$ coordinate system, then

$$\underset{\sim}{\boldsymbol{v}} = \sum_{j=1}^{3} \tilde{v}_j\underset{\sim}{\boldsymbol{e}}_j, \text{ where } \tilde{v}_j = \underset{\sim}{\boldsymbol{v}} \bullet \underset{\sim}{\boldsymbol{e}}_j \tag{1.18}$$

Equations (1.17) and (1.18) must *both* represent the same vector. We can substitute Eq. $(1.17)_1$ into $(1.18)_2$ to obtain

$$\tilde{v}_j = \underset{\sim}{\boldsymbol{v}} \bullet \underset{\sim}{\boldsymbol{e}}_j = \sum_{i=1}^{3} v_i\underset{\sim}{\boldsymbol{E}}_i \bullet \underset{\sim}{\boldsymbol{e}}_j = \sum_{i=1}^{3} v_i\underset{\sim}{\boldsymbol{e}}_j \bullet \underset{\sim}{\boldsymbol{E}}_i \tag{1.19}$$

In the last step, we have used the property that $\underset{\sim}{\boldsymbol{a}} \bullet \underset{\sim}{\boldsymbol{b}} = \underset{\sim}{\boldsymbol{b}} \bullet \underset{\sim}{\boldsymbol{a}}$ for any vectors $\underset{\sim}{\boldsymbol{a}}$ and $\underset{\sim}{\boldsymbol{b}}$ and therefore $\underset{\sim}{\boldsymbol{E}}_i \bullet \underset{\sim}{\boldsymbol{e}}_j = \underset{\sim}{\boldsymbol{e}}_j \bullet \underset{\sim}{\boldsymbol{E}}_i$. Observing that $\underset{\sim}{\boldsymbol{e}}_j \bullet \underset{\sim}{\boldsymbol{E}}_i = L_{ji}$, Eq. (1.19) may be written

$$\boxed{\tilde{v}_j = \sum_{i=1}^{3} L_{ji}v_i}. \qquad \text{In matrix form,} \quad \{\tilde{v}\} = [L]\{v\} \tag{1.20}$$

A similar procedure of substituting Eq. $(1.18)_1$ into $(1.17)_2$ gives

$$v_i = \underset{\sim}{\boldsymbol{v}} \bullet \underset{\sim}{\boldsymbol{E}}_i = \sum_{j=1}^{3} \tilde{v}_j\underset{\sim}{\boldsymbol{e}}_j \bullet \underset{\sim}{\boldsymbol{E}}_i \tag{1.21}$$

Recognizing that $\underset{\sim}{\boldsymbol{e}}_j \bullet \underset{\sim}{\boldsymbol{E}}_i = L_{ji}$, this may be written

$$\boxed{v_i = \sum_{j=1}^{3} L_{ji}\tilde{v}_j}. \qquad \text{In matrix form,} \quad \{v\} = [L]^T\{\tilde{v}\} \tag{1.22}$$

Not only are Eqs. (1.20) and (1.22) important results in their own right, they also demonstrate that the matrix $[L]$ must be orthogonal because both matrix equations can be true only if

$$\boxed{[L]^{-1} = [L]^T} \tag{1.23}$$

All vector coordinate transformations are in the form of either Eq. (1.20) or (1.22). It can be difficult to remember which equation to use unless you use the following mnemonic observation: Note from Eq. (1.8) that the first subscript on L_{ij} corresponds to the *rotated* basis, and the second subscript corresponds to the *laboratory* basis. The same rule applies in the vector coordinate transformations of Eqs. (1.20) and (1.22). The first subscript on $[L]$ always matches the subscript on the rotated $\{\tilde{v}\}$ components and the second subscript on $[L]$ always matches the subscript on the lab $\{v\}$ components, regardless of where these components appear in the transformation equation. The summation always occurs over the *repeated* index. This mnemonic rule also applies to coordinate transformations for tensors. Let $\underset{\approx}{\boldsymbol{T}}$ denote a second-order tensor.* Let T_{ij} denote the components with respect to the laboratory basis. Let $\tilde{T}_{kl}$ denote the components with respect to the rotated basis. There are two subscripts on the components, so the $[L]$ matrix will appear twice. The mnemonic rule says that the subscripts on the rotated components must match the *first* subscripts on the L's, and the subscripts on the laboratory components must match the *second* subscripts on the L's. Finally, any subscripts that appear twice in a single term must be summed. Thus, the coordinate transformation rules for tensors are

$$\tilde{T}_{kl} = \sum_{i=1}^{3}\sum_{j=1}^{3} L_{ki}L_{lj}T_{ij}, \text{ or in matrix form, } [\tilde{T}] = [L][T][L]^T \tag{1.24}$$

and

$$T_{ij} = \sum_{k=1}^{3}\sum_{l=1}^{3} L_{ki}L_{lj}T_{kl}, \text{ or in matrix form, } [T] = [L]^T[\tilde{T}][L] \tag{1.25}$$

Note that is easier to *first* write the component form, and *then* construct the corresponding matrix form. Doing it that way relieves you from having to recall where the transpose goes.

---

* We emphasize the tensorial order of a quantity by the number of underlines. For example, s is a scalar, $\underset{\sim}{\boldsymbol{v}}$ is a vector, $\underset{\approx}{\boldsymbol{T}}$ is a second-order tensor, etc.

Alternative direction cosine matrix. There is no divine law that says the direction cosines must be arranged in matrix form as was done in Eq. (1.8). Quite often, they are arranged in the *transpose* of the configuration used for $[L]$. In other words, many people define an alternative direction cosine matrix:

$$\boxed{Q_{ij} = \underset{\sim}{\boldsymbol{E}}_i \bullet \underset{\sim}{\boldsymbol{e}}_j}, \tag{1.26}$$

This is a perfectly legitimate alternative, and a mnemonic rule still holds except, with this approach, the first subscript on Q_{ij} corresponds to the laboratory components, and the second subscript corresponds to the rotated components. With this direction cosine definition, the key results of this section may be written

$$\boxed{[Q]^{-1} = [Q]^T} \tag{1.27}$$

$$\boxed{\underset{\sim}{\boldsymbol{e}}_i = \sum_{j=1}^{3} Q_{ji} \underset{\sim}{\boldsymbol{E}}_j} \tag{1.28}$$

$$\boxed{\underset{\sim}{\boldsymbol{E}}_i = \sum_{j=1}^{3} Q_{ij} \underset{\sim}{\boldsymbol{e}}_j} \tag{1.29}$$

$$\boxed{\tilde{v}_j = \sum_{i=1}^{3} Q_{ij} v_i}. \qquad \text{In matrix form,} \quad \{\tilde{v}\} = [Q]^T \{v\} \tag{1.30}$$

$$\boxed{v_i = \sum_{j=1}^{3} Q_{ij} \tilde{v}_j}. \qquad \text{In matrix form,} \quad \{v\} = [Q]\{\tilde{v}\} \tag{1.31}$$

$$\tilde{T}_{kl} = \sum_{i=1}^{3}\sum_{j=1}^{3} Q_{ik} Q_{jl} T_{ij}, \text{ or in matrix form, } [\tilde{T}] = [Q]^T [T][Q] \tag{1.32}$$

$$T_{ij} = \sum_{k=1}^{3}\sum_{l=1}^{3} Q_{ik} Q_{jl} \tilde{T}_{kl}, \text{ or in matrix form, } [T] = [Q][\tilde{T}][Q]^T \tag{1.33}$$

Coordinate transformations. The laboratory coordinates of a point in space are denoted $\{X, Y, Z\}$ or, using the numbered notation, $\{X_1, X_2, X_3\}$. The coordinates of that same point with respect to the rotated system are denoted $\{x, y, z\}$ or $\{x_1, x_2, x_3\}$. Because both coordinate systems are Cartesian, these position *coordinates* are identically equal to the position *components*. Thus, the above vector transformation relation also serve as coordinate transforma-

tion relations. Specifically, replacing the *lab* components $\{v_1, v_2, v_3\}$ by the *lab* coordinates $\{X_1, X_2, X_3\}$, and replacing the *rotated* components $\{\tilde{v}_1, \tilde{v}_2, \tilde{v}_3\}$ by the *rotated* components $\{x_1, x_2, x_3\}$, Eqs. (1.30) and (1.31) give us the coordinate transformation rules:

$$x_j = \sum_{i=1}^{3} Q_{ij} X_i \,. \qquad \text{In matrix form,} \quad \{x\} = [Q]^T \{X\} \tag{1.34}$$

$$X_i = \sum_{j=1}^{3} Q_{ij} x_j \,. \qquad \text{In matrix form,} \quad \{X\} = [Q]\{x\} \tag{1.35}$$

2. Rotation operations

We now introduce a subtle but important distinction between coordinate transformations and rotation operations. In a coordinate transformation, the *same object* is viewed from a new perspective. In Eqs. (1.17) and (1.18), the same vector $\underset{\sim}{\boldsymbol{v}}$ is viewed from two different coordinate systems. Eq. (1.20) shows how the components of $\underset{\sim}{\boldsymbol{v}}$ in each system are related — that is, how one *transforms* the components of the single vector $\underset{\sim}{\boldsymbol{v}}$ from one system to another.

A similar-looking, but fundamentally different, concept is that of an *operation* that changes one vector to another vector. An operation can be quite general, but here we limit the discussion to "right-handed linear orthogonal operations," which are described through the use of a *rotation tensor,* $\underset{\approx}{\boldsymbol{R}}$. The equation

$$\hat{\underset{\sim}{\boldsymbol{w}}} = \underset{\approx}{\boldsymbol{R}} \bullet \underset{\sim}{\boldsymbol{w}} \tag{2.1}$$

is a symbolic way of saying that the rotation operation $\underset{\approx}{\boldsymbol{R}}$ operates on a vector $\underset{\sim}{\boldsymbol{w}}$, giving a new vector $\hat{\underset{\sim}{\boldsymbol{w}}}$. Because $\underset{\approx}{\boldsymbol{R}}$ is a rotation operation, the new vector $\hat{\underset{\sim}{\boldsymbol{w}}}$ has the same magnitude as $\underset{\sim}{\boldsymbol{w}}$ but a different orientation. The laboratory components of $\hat{\underset{\sim}{\boldsymbol{w}}}$ are related to the laboratory components of $\underset{\sim}{\boldsymbol{w}}$ by

$$\hat{w}_i = \sum_{j=1}^{3} R_{ij} w_j \tag{2.2}$$

Here, we have *two different vectors*, $\underset{\sim}{\boldsymbol{w}}$ and $\hat{\underset{\sim}{\boldsymbol{w}}}$, expressed in *one* coordinate system. For coordinate transformations, we are dealing with *one* vector expressed in *two* coordinate systems.

If a rotated triad of base vectors $\{\underset{\sim}{\boldsymbol{e}}_1, \underset{\sim}{\boldsymbol{e}}_2, \underset{\sim}{\boldsymbol{e}}_3\}$ is obtained by applying a rotation tensor $\underset{\approx}{\boldsymbol{R}}$ to the laboratory base vectors $\{\underset{\sim}{\boldsymbol{E}}_1, \underset{\sim}{\boldsymbol{E}}_2, \underset{\sim}{\boldsymbol{E}}_3\}$, then it's straightforward to show that the laboratory components of the rotation tensor are simply the transpose of the associated transformation matrix $[L]$. Hence, the laboratory components of $\underset{\approx}{\boldsymbol{R}}$ are *identical* to the components of the alternative direction cosine matrix $[Q]$ from page 10, which is a good argument in favor $[Q]$. The columns of the component matrix for $\underset{\approx}{\boldsymbol{R}}$ contain the rotated base vectors (written in terms of the laboratory basis). Ordinarily the components of a tensor change upon a change of basis. Remarkably, however, it turns out that the components of $\underset{\approx}{\boldsymbol{R}}$ with respect to the rotated basis are identical to the components with respect to the laboratory basis![*] Thus,

$$\tilde{R}_{ij} = R_{ij} \tag{2.3}$$

---

* Later on, we show that a rotation operation can be viewed in terms of an axis and angle of rotation. The components of $\underset{\approx}{\boldsymbol{R}}$ will be unchanged with respect to *any* basis obtained by rotating the laboratory basis by *any* angle about the *same* rotation axis as associated with $\underset{\approx}{\boldsymbol{R}}$.

Other important relationships between the rotated basis and the transformation matrix are

$$\underset{\sim}{\boldsymbol{e}}_i = \underset{\approx}{\boldsymbol{R}} \bullet \underset{\sim}{\boldsymbol{E}}_i \tag{2.4}$$

$$L_{mn} \equiv \underset{\sim}{\boldsymbol{e}}_m \bullet \underset{\sim}{\boldsymbol{E}}_n = Q_{mn} \tag{2.5}$$

$$R_{ij} = \underset{\sim}{\boldsymbol{E}}_i \bullet \underset{\approx}{\boldsymbol{R}} \bullet \underset{\sim}{\boldsymbol{E}}_j = L_{ji} \tag{2.6}$$

$$\tilde{R}_{ij} = \underset{\sim}{\boldsymbol{e}}_i \bullet \underset{\approx}{\boldsymbol{R}} \bullet \underset{\sim}{\boldsymbol{e}}_j = L_{ji}$$

$$\underset{\sim}{\boldsymbol{e}}_i = \sum_{j=1}^{3} L_{ij} \underset{\sim}{\boldsymbol{E}}_j \tag{2.7}$$

$$\underset{\sim}{\boldsymbol{e}}_i = \sum_{j=1}^{3} R_{ji} \underset{\sim}{\boldsymbol{E}}_j$$

A reader has requested some more detailed/intuitive explanation of these equations. I'll try to expand discussion added in future versions. -- RMB

(2.8)

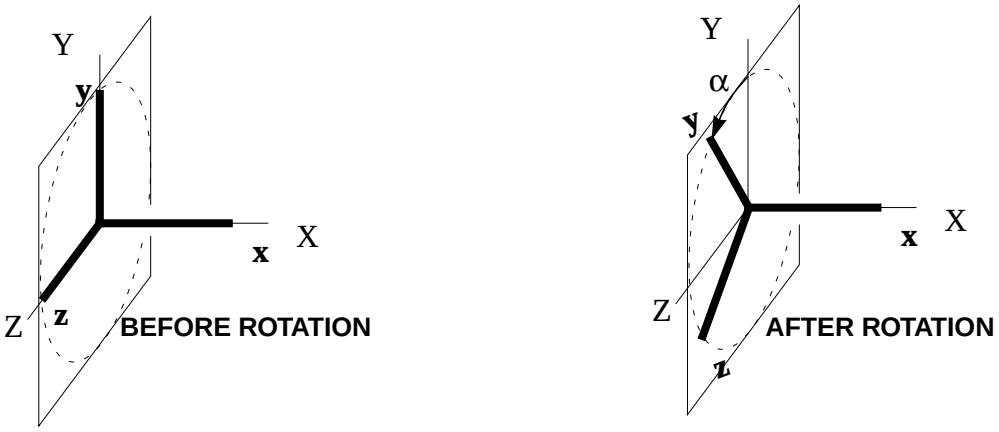
Note the distinction between Eqs. (2.4) and (2.8). We have *not* gotten the subscripts on R_{ji} accidentally reversed! To obtain Eq. (2.8) from Eq. (2.4), first note that the j^{th} component of $\underset{\sim}{\boldsymbol{e}}_i$ with respect to the lab basis is obtained by dotting $\underset{\sim}{\boldsymbol{E}}_j$ into $\underset{\sim}{\boldsymbol{e}}_i$. Dotting both sides of Eq. (2.4) from the left by $\underset{\sim}{\boldsymbol{E}}_j$ shows that the j^{th} component of $\underset{\sim}{\boldsymbol{e}}_i$ with respect to the lab basis therefore must be R_{ji}, which is exactly what appears in Eq. (2.4).

Equation (1.20) involves a single vector expressed in two coordinate systems whereas Eq. (2.2) involves two vectors expressed in a single coordinate system. In practice, this distinction is often confusing, and it is usually easiest to tell whether you need $\underset{\approx}{\boldsymbol{R}}$ or $[L]$ by considering a simple rotation for which you know the desired behavior (such as a 90° rotation about one of the coordinate axes). It should make some intuitive sense that the components of $\underset{\approx}{\boldsymbol{R}}$ are just the transpose of the components of $[L]$, which since $[L]$ is orthogonal, are components of the inverse of $[L]$. The transformation matrix $[L]$ represents how vectors look to an observer attached to the rotated system. Thus, if the tensor operation $\underset{\approx}{\boldsymbol{R}}$ rotates the system, say, 20° clockwise, then stationary vectors will appear to have rotated 20° *counterclockwise* to an observer in the new system*. Thus $\underset{\approx}{\boldsymbol{R}}$ and $[L]$ are related by rotations that are equal in magnitude, but opposite in direction.

* Keep in mind, though, the vector itself is *the same* no matter what coordinate system is used. Both the components and the base vectors change under coordinate transformation, but the *sum* of components times base vectors is invariant.

Equation (2.5) shows that the components of a rotation tensor are the same as the direction cosines of the rotated base vectors — the columns of $[R_{ij}]$ simply contain the rotated basis! Hence, to be a rotation, the columns must form a right-handed orthonormal triad. A routine for testing whether a matrix is a rotation is provided in **Listing 1 (Testing whether a matrix is a rotation) on page A-1**.

Example: rotation of an angle α about the X-axis**.**



For this rotation about X, the x axis remains aligned with the X-axis, so $\cos\theta_{xX} = 1$, $\cos\theta_{xY} = 0$, and $\cos\theta_{xZ} = 0$. The angle that the rotated y-axis makes with fixed Y-axis is θ, so $\cos\theta_{yY} = \cos\alpha$. The angle that y makes with Z is $\pi/2$-θ, so $\cos\theta_{yZ} = \cos(\pi/2 - \alpha) = \sin\alpha$, and so on. Thus, referring to Eqs. (1.4) and (2.6), the transformation matrix and the rotation tensor are

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & -\sin\alpha & \cos\alpha \end{bmatrix} \qquad [\underset{\approx}{\boldsymbol{R}}] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \tag{2.9}$$

Recall that the matrix $[\underset{\approx}{\boldsymbol{R}}]$ is the same as the matrix $[Q]$. Also recall that the components of position with respect to the lab system are denoted $\{X_1, X_2, X_3\}$ or $\{X, Y, Z\}$. Likewise, the components of that same location with respect to the rotated system are denoted $\{x_1, x_2, x_3\}$ or $\{x, y, z\}$. Thus, Eq. (1.34) becomes

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & -\sin\alpha & \cos\alpha \end{bmatrix} \begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix} \tag{2.10}$$

or

$$\begin{aligned} x &= X \\ y &= (\cos\alpha)\,Y + (\sin\alpha)\,Z \\ z &= (-\sin\alpha)\,Y + (\cos\alpha)\,Z \end{aligned} \tag{2.11}$$

Rotations about a particular axis will always involve a submatrix of the form

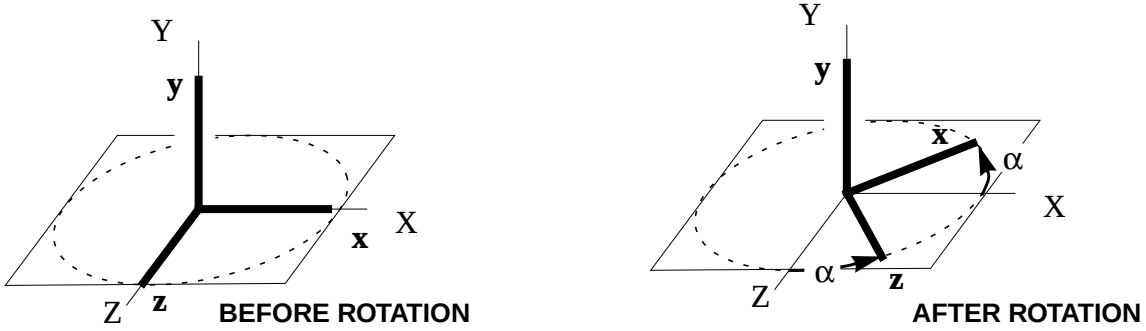
$$\begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix} \tag{2.12}$$

If you are ever unsure whether you have placed the "negative sign" on correct one of the two $\sin\alpha$, an easy way to check is to verify that you get the result when the rotation angle exactly equals $90°$. In this special limiting case, Eq. (2.11) becomes

$$\begin{aligned} x &= X \\ y &= Z \\ z &= -Y \end{aligned} \tag{2.13}$$

A visual inspection of the rotation sketch for a ninety degree rotation reveals that the y axis ends up aligned with the Z axis (so that $y = Z$) and the z axis ends up parallel to the Y axis but oppositely pointed (so that $z = -Y$). Thus, Eq. (2.13) gave the right result, which means we placed the negative sign correctly in Eq. (2.11). This simple "$90°$ limiting case" test can be a lifesaver.

Example: rotation of an angle α about the Y-axis.



Referring to Eqs. (1.4) and (2.6), the transformation matrix and the rotation matrix are:

$$L = \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix} \qquad [\underset{\approx}{\boldsymbol{R}}] = \begin{bmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix} \tag{2.14}$$

Note the relative difference between Eqs. (2.14) and (2.9).

Recall that the matrix $[\underset{\approx}{\boldsymbol{R}}]$ is the same as the matrix $[Q]$. Also recall that the components of position with respect to the lab system are denoted $\{X_1, X_2, X_3\}$ or $\{X, Y, Z\}$. Likewise, the components of that same location with respect to the rotated system are denoted $\{x_1, x_2, x_3\}$ or $\{x, y, z\}$. Thus, Eq. (1.34) becomes

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix} \begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix} \tag{2.15}$$

or

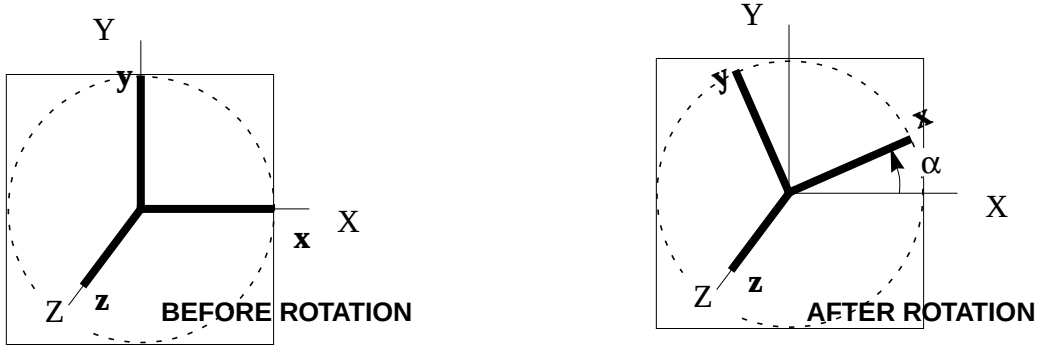
$$\begin{aligned} x &= (\cos\alpha)X - (\sin\alpha)Z \\ y &= Y \\ z &= (\sin\alpha)X + (\cos\alpha)Z \end{aligned} \tag{2.16}$$

For the special case of a $90°$ rotation, this becomes

$$\begin{aligned} x &= -Z \\ y &= Y \\ z &= X, \end{aligned} \tag{2.17}$$

which is consistent with visual inspection of the rotation figure. Namely, upon a ninety degree rotation, little x ends up pointing in the $-Z$ direction, little y stays equal to big Y, and little z ends up pointing in the big X direction. Since these observations are consistent with Eq. (2.17), we can feel assured that the negative sign in Eq. (2.16) was placed in the correct position.

Example: rotation of an angle α about the Z-axis.



Referring to Eqs. (1.4) and (2.4), the transformation matrix and the rotation matrix are:

$$L = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad [\underset{\approx}{\boldsymbol{R}}] = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.18}$$

Note the relative differences between Eqs. (2.18), (2.14) and (2.9).

Where does that negative sign go? Note how the negative in front of the sine seems to be in a "different relative spot" when we compare equations Eqs. (2.9), (2.14), and (2.18). The proper way to think about the location of the negative is *not* to consider the submatrix of sines and cosines. Instead consider the placement of the negative when moving cyclically right* one column and cyclically up* one row from the solitary "1" on the diagonal. In all cases, the negative in $[L]$ is up and right from the "1"; the negative in $\underset{\approx}{\boldsymbol{R}}$ is cyclically down and left from the "1". Another (often easier) way to remember is to simply consider a simple 90° rotation for which you know the answer and can see where the negative needs to be.

Faster way to write down the matrix of a rotation tensor. We've already mentioned that you can construct $\underset{\approx}{\boldsymbol{R}}$ by computing the cosines of the angles between the laboratory axes and the embedded axes. This viewpoint can be somewhat tedious. A better (and equivalent) procedure is to simply recognize that the columns of $[\underset{\approx}{\boldsymbol{R}}]$ are given by the rotated basis when expressed in terms of the lab coordinates.

Consider the above example in which we rotate by an angle θ rotation about the Z-axis. The three *laboratory* base vectors are

$$\underset{\sim}{\boldsymbol{E}}_X = \begin{Bmatrix} 1 \\ 0 \\ 0 \end{Bmatrix}, \qquad \underset{\sim}{\boldsymbol{E}}_Y = \begin{Bmatrix} 0 \\ 1 \\ 0 \end{Bmatrix}, \quad \text{and} \quad \underset{\sim}{\boldsymbol{E}}_Z = \begin{Bmatrix} 0 \\ 0 \\ 1 \end{Bmatrix} \tag{2.19}$$

BEFORE ROTATION **AFTER ROTATION**

* Moving cyclically right from the last column means looping back to the first column. Similarly, moving cyclically up from the top row means looping back down to the bottom row.

Referring to the above figure, the *rotated* base vectors are

$$\underset{\sim}{\boldsymbol{e}}_x = \cos\alpha\,\underset{\sim}{\boldsymbol{E}}_X + \sin\alpha\,\underset{\sim}{\boldsymbol{E}}_Y \tag{2.20}$$

$$\underset{\sim}{\boldsymbol{e}}_y = -\sin\alpha\,\underset{\sim}{\boldsymbol{E}}_X + \cos\alpha\,\underset{\sim}{\boldsymbol{E}}_Y \tag{2.21}$$

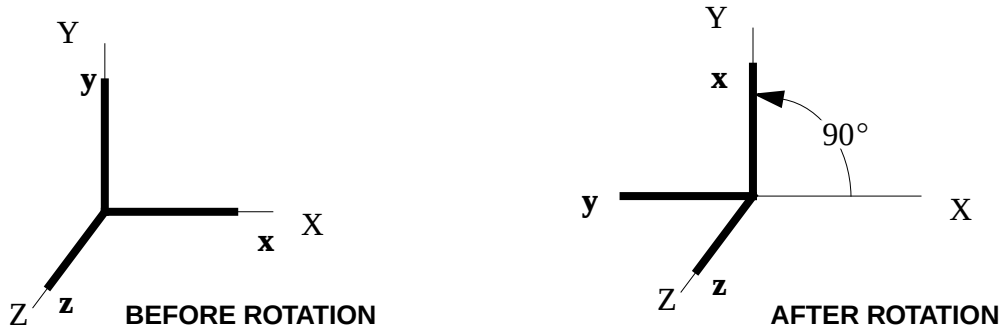
$$\underset{\sim}{\boldsymbol{e}}_x = \underset{\sim}{\boldsymbol{E}}_Z \tag{2.22}$$

or, written as column vectors,

$$\underset{\sim}{e}_x = \begin{Bmatrix} \cos\alpha \\ \sin\alpha \\ 0 \end{Bmatrix},\ \underset{\sim}{e}_y = \begin{Bmatrix} -\sin\alpha \\ \cos\alpha \\ 0 \end{Bmatrix},\ \text{and}\ \underset{\sim}{e}_z = \begin{Bmatrix} 0 \\ 0 \\ 1 \end{Bmatrix} \tag{2.23}$$

These three vectors assemble together to form the columns of $\underset{\approx}{\boldsymbol{R}}$ in Eq. (2.18). That's easy!

Specific example: A 90˚ rotation about the Z-axis.



Substituting $\theta = 90°$ into Eq. (2.18) gives

$$L = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad [\underset{\approx}{\boldsymbol{R}}] = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.24}$$

When writing down the rotation matrix for a rotation of angle α about the Z-axis, it is often useful to quickly verify that your matrix gives the desired result for the special case of a 90˚ rotation. This simple sanity check is very useful for ensuring that the negative is placed on the correct off-diagonal "sine" component.

3. Axis and angle of rotation

As explained above, a rotation may be described by the direction cosines of the rotation or transformation. These direction cosines are not all independent. Euler's theorem states that any rotation may be specified by an axis and angle of rotation. The axis of rotation, $\underset{\sim}{\boldsymbol{a}}$, is a unit vector and the angle of rotation, θ, is defined by the right hand rule with respect to the axis $\underset{\sim}{\boldsymbol{a}}$. The axis of rotation has two independent components (the third being restricted to ensure that $||\underset{\sim}{\boldsymbol{a}}|| = 1$). Hence, with the angle of rotation, only three numbers are needed to completely specify a rotation and therefore only three of the nine direction cosines in Eq. (1.4) are independent.

Because, according to Euler's theorem, any rotation can be described by a unit vector $\underset{\sim}{\boldsymbol{a}}$ and a rotation angle α, one can permissibly state that the "pseudo" rotation vector defined $\underset{\sim}{\boldsymbol{s}} = \alpha\underset{\sim}{\boldsymbol{a}}$ fully describes a rotation. This is a legal construction, but one must be cautioned that rotation operations do not commute. Thus, if you apply one rotation followed by a second rotation, then the result will be different if you apply the rotations in the opposite order. Thus, even though each individual rotation can be described through individual pseudo rotation vectors, $\underset{\sim}{\boldsymbol{s}}^{(1)}$ and $\underset{\sim}{\boldsymbol{s}}^{(2)}$, the pseudo rotation vector corresponding to sequentially applied rotations will not generally equal $\underset{\sim}{\boldsymbol{s}}^{(1)} + \underset{\sim}{\boldsymbol{s}}^{(2)}$. This issue is discussed in much greater detail in Chapter 7. For the discussion of the present chapter, we will keep the rotation angle α and the rotation axis $\underset{\sim}{\boldsymbol{a}}$ clearly separated.

Euler-Rodrigues formula

Euler's theorem* states that the general displacement of a rigid body with one point fixed is a rotation about some axis that passes through that fixed point. For convenience in this section, we place our origin at the fixed point.

Consider a rotation operation $\underset{\approx}{\boldsymbol{R}}$ about a (unit) axis $\underset{\sim}{\boldsymbol{a}}$. Consider an arbitrary position vector $\underset{\sim}{\boldsymbol{x}}$ that points from the fixed origin to an arbitrary point on the rigid body. Letting α denote the rotation angle, the sketch in Fig. 3.1 shows how $\underset{\sim}{\boldsymbol{x}}$ is transformed to become a new rotated vector $\underset{\approx}{\boldsymbol{R}} \bullet \underset{\sim}{\boldsymbol{x}}$.

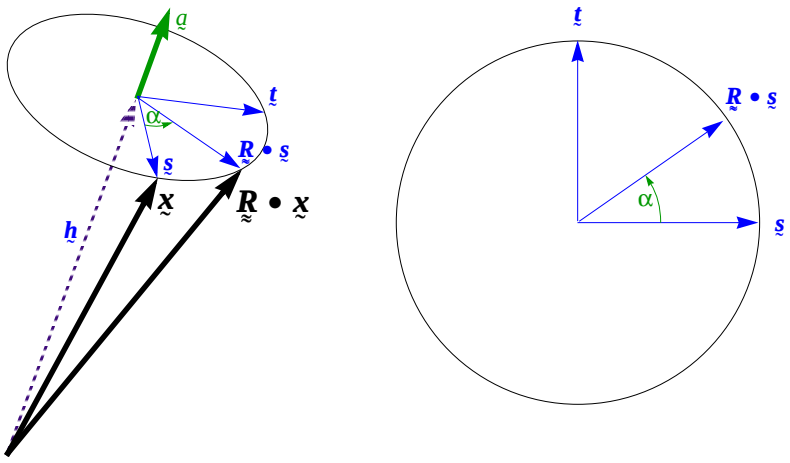


Figure 3.1. Rotation of a vector $\underset{\sim}{\boldsymbol{x}}$ by a known angle α about a known axis $\underset{\sim}{\boldsymbol{a}}$. In this figure, we have introduced some helper vectors: $\underset{\sim}{\boldsymbol{h}}$ is the part of $\underset{\sim}{\boldsymbol{x}}$ in the direction of $\underset{\sim}{\boldsymbol{a}}$, while $\underset{\sim}{\boldsymbol{s}}$ is the perpendicular part. The vector $\underset{\sim}{\boldsymbol{t}}$ is simply defined to be perpendicular to both $\underset{\sim}{\boldsymbol{a}}$ and $\underset{\sim}{\boldsymbol{x}}$ as shown.

The vector $\underset{\sim}{\boldsymbol{h}}$ shown in Fig. 3.1 is the part of $\underset{\sim}{\boldsymbol{x}}$ in the direction of $\underset{\sim}{\boldsymbol{a}}$, so it must be given by

$$\underset{\sim}{\boldsymbol{h}} = (\underset{\sim}{\boldsymbol{x}} \bullet \underset{\sim}{\boldsymbol{a}})\underset{\sim}{\boldsymbol{a}} \tag{3.1}$$

The vector $\underset{\sim}{\boldsymbol{s}}$ is defined to be the part of $\underset{\sim}{\boldsymbol{x}}$ that is perpendicular to $\underset{\sim}{\boldsymbol{a}}$, so it must be given by

$$\begin{aligned} \underset{\sim}{\boldsymbol{s}} &= \underset{\sim}{\boldsymbol{x}} - \underset{\sim}{\boldsymbol{h}} \\ &= \underset{\sim}{\boldsymbol{x}} - (\underset{\sim}{\boldsymbol{x}} \bullet \underset{\sim}{\boldsymbol{a}})\underset{\sim}{\boldsymbol{a}} \end{aligned} \tag{3.2}$$

The vector $\underset{\sim}{\boldsymbol{t}}$ is defined to be perpendicular to both $\underset{\sim}{\boldsymbol{a}}$ and $\underset{\sim}{\boldsymbol{s}}$, but with the same magnitude as $\underset{\sim}{\boldsymbol{s}}$. Recalling that $\underset{\sim}{\boldsymbol{a}}$ is a unit vector, we therefore conclude that

$$\begin{aligned} \underset{\sim}{\boldsymbol{t}} &= \underset{\sim}{\boldsymbol{a}} \times \underset{\sim}{\boldsymbol{s}} \\ &= \underset{\sim}{\boldsymbol{a}} \times (\underset{\sim}{\boldsymbol{x}} - \underset{\sim}{\boldsymbol{h}}) \\ &= \underset{\sim}{\boldsymbol{a}} \times \underset{\sim}{\boldsymbol{x}} \end{aligned} \tag{3.3}$$

* one of many!

In the last step, we have noted that $\underset{\sim}{a}\times\underset{\sim}{h}=\underset{\sim}{0}$ because $\underset{\sim}{h}$ is parallel to $\underset{\sim}{a}$. Looking strictly in the plane of rotation (right side of Fig. 3.1), we note that the transformed location of $\underset{\sim}{s}$ is given by

$$\begin{aligned}\underset{\approx}{R}\bullet\underset{\sim}{s} &= (\cos\alpha)\underset{\sim}{s}+(\sin\alpha)\underset{\sim}{t}\\ &= (\cos\alpha)[\underset{\sim}{x}-(\underset{\sim}{x}\bullet\underset{\sim}{a})\underset{\sim}{a}]+(\sin\alpha)[\underset{\sim}{a}\times\underset{\sim}{x}]\end{aligned} \tag{3.4}$$

From the figure on the left side of Fig. 3.1, we note that the rotated vector $\underset{\approx}{R}\bullet\underset{\sim}{x}$ can be broken into two parts as the vector sum of $\underset{\sim}{h}$ and $\underset{\approx}{R}\bullet\underset{\sim}{s}$:

$$\begin{aligned}\underset{\approx}{R}\bullet\underset{\sim}{x} &= \underset{\sim}{h}+\underset{\approx}{R}\bullet\underset{\sim}{s}\\ &= (\underset{\sim}{x}\bullet\underset{\sim}{a})\underset{\sim}{a}+(\cos\alpha)[\underset{\sim}{x}-(\underset{\sim}{x}\bullet\underset{\sim}{a})\underset{\sim}{a}]+(\sin\alpha)[\underset{\sim}{a}\times\underset{\sim}{x}]\end{aligned} \tag{3.5}$$

Slight rearrangement gives the so-called **Euler-Rodrigues formula**:*

$$\boxed{\underset{\approx}{R}\bullet\underset{\sim}{x} = (\cos\alpha)[\underset{\sim}{x}-\underset{\sim}{a}(\underset{\sim}{a}\bullet\underset{\sim}{x})]+\underset{\sim}{a}(\underset{\sim}{a}\bullet\underset{\sim}{x})+(\sin\alpha)[\underset{\sim}{a}\times\underset{\sim}{x}]} \tag{3.6}$$

In the next section, we will we will use this formula to extract a direct notation expression for the rotation tensor. Toward this end, it is useful to recognize that the cross product $\underset{\sim}{a}\times\underset{\sim}{x}$ is linear with respect to $\underset{\sim}{x}$. Consequently, there must exist a tensor $\underset{\approx}{A}$ — called the "axial" tensor — such that

$$\underset{\sim}{a}\times\underset{\sim}{x} = \underset{\approx}{A}\bullet\underset{\sim}{x} \tag{3.7}$$

This equation may be written in (Cartesian) component form as

$$-\varepsilon_{ijk}a_j x_k = A_{ik}x_k \tag{3.8}$$

where ε_{ijk} is the permutation symbol defined by

$$\varepsilon_{ijk} = \begin{cases}+1 & \text{if } ijk=123,\ 231,\ \text{or } 312\\ -1 & \text{if } ijk=321,\ 132,\ \text{or } 213\\ 0 & \text{otherwise}\end{cases} \tag{3.9}$$

Eliminating the common factor of x_k in Eq. (3.8) gives the direct notation and component formulas for the axial tensor, which we write below along with the indicial and matrix expressions to ensure clarity:

$$\underset{\approx}{A}\equiv-\underset{\approx\approx}{\varepsilon}\bullet\underset{\sim}{a}=\underset{\sim}{a}\times\underset{\approx}{I}\quad\Leftrightarrow\quad A_{ij}=-\varepsilon_{ijk}a_k\quad\Leftrightarrow\quad [A]=\begin{bmatrix}0 & -a_3 & a_2\\ a_3 & 0 & -a_1\\ -a_2 & a_1 & 0\end{bmatrix} \tag{3.10}$$

* In a footnote about this eponymic designation, Goldstein [1], states that Hamel (*Theoretische Mechanik*, p. 103) ascribes this theorem to the French mathematician O. Rodrigues (1794-1851), but Goldstein claims this might be an error, suggesting that Gibbs was the first to put it in vector form (*Vector Analysis*, p. 338). The underlying formula is apparently much older still.

The axial tensor is skew-symmetric because the permutation symbol has the property that $\varepsilon_{ijk} = -\varepsilon_{jik}$. By the cyclic property of the permutation symbol, the component A_{ij} is often written with the free index in the center of the permutation symbol and the negative sign removed. In other words, the alternative expression, $A_{ij} = \varepsilon_{ikj} a_k$, is equivalent to Eq. (3.10).

Incidentally, note the following trigonometric identity

$$\cos\alpha = 1 - 2\sin^2\left(\frac{\alpha}{2}\right) \tag{3.11}$$

Also note the vector identity

$$\underset{\sim}{a} \times (\underset{\sim}{a} \times \underset{\sim}{x}) = \underset{\sim}{a}(\underset{\sim}{a} \bullet \underset{\sim}{x}) - \underset{\sim}{x}(\underset{\sim}{a} \bullet \underset{\sim}{a}) = \underset{\sim}{a}(\underset{\sim}{a} \bullet \underset{\sim}{x}) - \underset{\sim}{x}, \tag{3.12}$$

In the final step, we used the fact that $\underset{\sim}{a}$ is a unit vector to write $\underset{\sim}{a} \bullet \underset{\sim}{a} = 1$. By using the above two identities, Eq. (3.6) may be written as

$$\boxed{\underset{\approx}{R} \bullet \underset{\sim}{x} = \underset{\sim}{x} + (\sin\alpha)(\underset{\sim}{a} \times \underset{\sim}{x}) + 2\sin^2\left(\frac{\alpha}{2}\right)(\underset{\sim}{a} \times (\underset{\sim}{a} \times \underset{\sim}{x}))} \tag{3.13}$$

This is the form of the Euler-Rodrigues identity cited by Argyris [2].

Computing the rotation tensor given axis and angle

Given the (unit) axis $\underset{\sim}{a}$ and angle α, what is the rotation tensor? The answer is found by simply differentiating both sides of the **Euler-Rodrigues formula** of Eq. (3.6) by $\underset{\sim}{x}$ to obtain

$$\boxed{\underset{\approx}{R} = \cos\alpha\ (\underset{\approx}{I} - \underset{\sim}{a}\,\underset{\sim}{a}) + \underset{\sim}{a}\,\underset{\sim}{a} + \sin\alpha\ \underset{\approx}{A}} \tag{3.14}$$

Alternatively, using the trigonometric identity $\cos\alpha = 1 - 2\sin^2(\alpha/2)$, Eq. (3.14) can be written as

$$\underset{\approx}{R} = \underset{\approx}{I} + 2\sin^2\left(\frac{\alpha}{2}\right)(\underset{\sim}{a}\,\underset{\sim}{a} - \underset{\approx}{I}) + \sin\alpha\ \underset{\approx}{A} \tag{3.15}$$

Here, $\underset{\approx}{I}$ is the identity tensor (represented in Cartesian coordinates by the 3×3 identity matrix). As explained in appendix B, symbol $\underset{\sim}{a}\,\underset{\sim}{a}$ denotes a vector-vector dyad (often called the *outer product*), which is simply a tensor corresponding to a 3×3 Cartesian matrix with components

$$(\underset{\sim}{a}\,\underset{\sim}{a})_{ij} = a_i a_j \tag{3.16}$$

written out,

$$[\underset{\sim}{a}\,\underset{\sim}{a}] = \begin{bmatrix} a_1a_1 & a_1a_2 & a_1a_3 \\ a_2a_1 & a_2a_2 & a_2a_3 \\ a_3a_1 & a_3a_2 & a_3a_3 \end{bmatrix} \tag{3.17}$$

The tensor $\underset{\approx}{A}$ is the skew symmetric "axial" tensor associated with the axis $\underset{\sim}{a}$. In direct notation, the tensor $\underset{\approx}{A}$ is defined such that $\underset{\approx}{A} \bullet \underset{\sim}{u} = \underset{\sim}{a} \times \underset{\sim}{u}$ for any vector $\underset{\sim}{u}$. It was shown in the previous section that $\underset{\approx}{A} \equiv -\underset{\approx\approx}{\varepsilon} \bullet \underset{\sim}{a}$, where the symbol $\underset{\approx\approx}{\varepsilon}$ denotes the third-order alternating tensor.* Expanding the operation $-\underset{\approx\approx}{\varepsilon} \bullet \underset{\sim}{a}$ shows that the 3×3 Cartesian matrix for $\underset{\approx}{A}$ is

$$[\underset{\approx}{A}] = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{3.18}$$

Thus, substituting Eqs. (3.17) and (3.18) into (3.14) gives the component form of the Euler-Rodrigues (tensor) formula:

$$[\underset{\approx}{R}] = \cos\alpha \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + (1-\cos\alpha) \begin{bmatrix} a_1a_1 & a_1a_2 & a_1a_3 \\ a_2a_1 & a_2a_2 & a_2a_3 \\ a_3a_1 & a_3a_2 & a_3a_3 \end{bmatrix} + \sin\alpha \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{3.19}$$

This equation is coded in **Listing 2 (Converting axis and angle to direction cosines) on page A-2**. In mathematica, this operation can be defined by the following function

```
rotationMatrix[axis_, angle_] := Cos[angle] IdentityMatrix[3]
+ (1 - Cos[angle]) Outer[Times, axis, axis]
+ Sin[angle] {{0, -axis[[3]], axis[[2]]}, {axis[[3]], 0, -axis[[1]]},
{-axis[[2]], axis[[1]], 0}};
```

Example. Consider a rotation of 120 degrees about an axis that passes through the point (1,1,1).

The axis of rotation is just a unit vector that points from the origin to the point (1,1,1):

$$\{\underset{\sim}{a}\} = \frac{1}{\sqrt{3}} \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} \tag{3.20}$$

* The alternating tensor is a third order tensor with Cartesian components defined as follows: $\varepsilon_{ijk}=1$ if ijk = 123, 231, or 312; $\varepsilon_{ijk}=-1$ if ijk = 321, 132, or 213; and $\varepsilon_{ijk}=0$ for all other ijk.

The sine and cosine of the angle are $\sin 120° = \sqrt{3}/2$ and $\cos 120° = -1/2$. Thus, using Eq. (3.19),

$$[\underset{\approx}{\boldsymbol{R}}] = \left(-\frac{1}{2}\right)\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix} + \left(\frac{3}{2}\right)\begin{bmatrix}1/3&1/3&1/3\\1/3&1/3&1/3\\1/3&1/3&1/3\end{bmatrix} + \frac{\sqrt{3}}{2}\begin{bmatrix}0&-1/\sqrt{3}&1/\sqrt{3}\\1/\sqrt{3}&0&-1/\sqrt{3}\\-1/\sqrt{3}&1/\sqrt{3}&0\end{bmatrix} \tag{3.21}$$

or

$$[\underset{\approx}{\boldsymbol{R}}] = \begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix} \tag{3.22}$$

This result could be obtained by using the mathematica function defined on page 23. Namely, `rotationMatrix[{1, 1, 1}/Sqrt[3], 120 Degree]`.

Another example. Consider a rotation of 180 degrees about an axis that passes through the point (1,1,1):

Again we have the following unit vector representing the axis of rotation:

$$\{\underset{\sim}{\boldsymbol{a}}\} = \frac{1}{\sqrt{3}}\begin{Bmatrix}1\\1\\1\end{Bmatrix} \tag{3.23}$$

The sine and cosine of the angle are $\sin 180° = 0$ and $\cos 180° = -1$. Thus, using Eq. (3.19),

$$[\underset{\approx}{\boldsymbol{R}}] = (-1)\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix} + 2\begin{bmatrix}1/3&1/3&1/3\\1/3&1/3&1/3\\1/3&1/3&1/3\end{bmatrix} = \begin{bmatrix}-1/3&2/3&2/3\\2/3&-1/3&2/3\\2/3&2/3&-1/3\end{bmatrix} \tag{3.24}$$

Note that this rotation tensor is symmetric. It is straightforward to prove that this will happen if and only if the angle of rotation is an integral multiple of 180 degrees.

This result could be obtained by using the mathematica function defined on page 23 by executing... `rotationMatrix[{1, 1, 1}/Sqrt[3], 180 Degree]`.

Another similar example. Consider a rotation of 180 degrees about an axis that passes through the point (3,4,12):

We have the following *unit* vector representing the axis of rotation:

$$\{\underset{\sim}{\boldsymbol{a}}\} = \frac{1}{13}\begin{Bmatrix}3\\4\\12\end{Bmatrix} \tag{3.25}$$

The sine and cosine of the angle are $\sin 180° = 0$ and $\cos 180° = -1$. Thus, using Eq. (3.19),

$$[\underset{\approx}{\boldsymbol{R}}] = (-1)\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix} + \frac{2}{(13)^2}\begin{bmatrix}9&12&36\\12&16&48\\36&48&144\end{bmatrix} = \frac{1}{(13)^2}\begin{bmatrix}-151&24&72\\24&-137&96\\72&96&-119\end{bmatrix} \tag{3.26}$$

Note that this rotation tensor is symmetric. As previously asserted, this will happen if and only if the angle of rotation is an integral multiple of 180 degrees.

This result could be obtained by using the mathematica function defined on page 23 by executing... `rotationMatrix[{3, 4, 12}/13, 180 Degree]`.

Numerical example (for testing computer codes). Consider a rotation of 76˚ about an axis that passes through the point (1, 3.2, 7).

The axis of rotation is a unit vector that points from the origin to the point (1, 3.2, 7), namely

$$a=\{0.128842,\ 0.412294,\ 0.901894\} \tag{3.27}$$

The sine and cosine of the angle are

$$\begin{aligned}\sin(76^\circ)&=0.970296\\ \cos(76^\circ)&=0.241922\end{aligned} \tag{3.28}$$

Using Eq. (3.19),

$$\begin{aligned}[\underset{\approx}{\boldsymbol{R}}] \;=\; & 0.241922\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}\\ & +(0.758078)\begin{bmatrix}0.0166003 & 0.0531208 & 0.116202\\ 0.0531208 & 0.169987 & 0.371846\\ 0.116202 & 0.371846 & 0.813413\end{bmatrix}\\ & +0.970296\begin{bmatrix}0 & -0.901894 & 0.412294\\ 0.901894 & 0 & -0.128842\\ -0.412294 & 0.128842 & 0\end{bmatrix}\end{aligned} \tag{3.29}$$

or

$$[\underset{\approx}{\boldsymbol{R}}] \;=\; \begin{bmatrix}0.254506 & -0.834834 & 0.488138\\ 0.915374 & 0.370785 & 0.156873\\ -0.311957 & 0.406903 & 0.858552\end{bmatrix} \tag{3.30}$$

This result could be obtained by using the mathematica function defined on page 23: `rotationMatrix[{1, 3.2, 7}/Sqrt[1+3.2^2+7^2], 76 Degree]`.

Alternative way to construct the rotation tensor. The Euler-Rodrigues formula of Eq. (3.14) is the most elegant means of constructing the matrix of the rotation tensor, given axis and angle. However, you might encounter the following alternative algorithm, which we present here for completeness:

(i) Assume the user has supplied the components $\{a_1, a_2, a_3\}$ of the rotation axis unit vector $\underset{\sim}{\boldsymbol{a}}$ with respect to the laboratory $\{\underset{\sim}{\boldsymbol{E}}_1, \underset{\sim}{\boldsymbol{E}}_2, \underset{\sim}{\boldsymbol{E}}_3\}$ basis. First ensure that the rotation axis is a unit vector by replacing $\underset{\sim}{\boldsymbol{a}}$ by $\underset{\sim}{\boldsymbol{a}}/|\underset{\sim}{\boldsymbol{a}}|$.

(ii) Determine the smallest component of $\underset{\sim}{\boldsymbol{a}}$. If this component is a_K, then set $\hat{\underset{\sim}{\boldsymbol{s}}} = \underset{\sim}{\boldsymbol{E}}_K$. This vector will be linearly independent of $\underset{\sim}{\boldsymbol{a}}$, so you can construct a unit vector $\underset{\sim}{\boldsymbol{b}}$ that is perpendicular to $\underset{\sim}{\boldsymbol{a}}$ by:

$$\underset{\sim}{\boldsymbol{b}} = \frac{\underset{\sim}{\boldsymbol{a}} \times \hat{\underset{\sim}{\boldsymbol{s}}}}{|\underset{\sim}{\boldsymbol{a}} \times \hat{\underset{\sim}{\boldsymbol{s}}}|} \tag{3.31}$$

(iii) Construct a third vector $\underset{\sim}{\boldsymbol{c}}$ that is right-perpendicular to both $\underset{\sim}{\boldsymbol{a}}$ and $\underset{\sim}{\boldsymbol{b}}$ by

$$\underset{\sim}{\boldsymbol{c}} = \underset{\sim}{\boldsymbol{a}} \times \underset{\sim}{\boldsymbol{b}} \tag{3.32}$$

(iv) Construct a real unitary (proper orthogonal) matrix $[L]$ whose columns contain the vectors $\{\underset{\sim}{\boldsymbol{b}}, \underset{\sim}{\boldsymbol{c}}, \underset{\sim}{\boldsymbol{a}}\}$:

$$[L] = \begin{bmatrix} b_1 & c_1 & a_1 \\ b_2 & c_2 & a_2 \\ b_3 & c_3 & a_3 \end{bmatrix} \tag{3.33}$$

(v) Then the matrix for the rotation tensor is

$$[\underset{\approx}{\boldsymbol{R}}] = [L] \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} [L]^T \tag{3.34}$$

In essence, this procedure:

(1) constructs the transformation matrix $[L]$ that temporarily transforms to a basis for which $\underset{\sim}{\boldsymbol{a}}$ is in the 3-direction, then
(2) performs the rotation about this new 1-direction, and finally
(3) transforms back to the laboratory configuration.

Although this approach is perfectly legitimate, direct application of the Euler-Rodrigues formula seems more elegant due to its invariant nature. A major advantage of the Euler-Rodrigues formula is that one may take its rate to obtain the material time derivative $\dot{\underset{\approx}{\boldsymbol{R}}}$ as a *direct notation* function of α, $\underset{\sim}{\boldsymbol{a}}$, $\dot{\alpha}$, and $\dot{\underset{\sim}{\boldsymbol{a}}}$ (see Eq. 12.15).

Some properties of the axial tensor. This section (which the reader may skip without loss) summarizes several useful formulas involving the axial tensor $\underset{\approx}{\boldsymbol{A}}$ that was defined in Eq. (3.10):

$$\underset{\approx}{\boldsymbol{A}} \equiv -\underset{\approx}{\boldsymbol{\varepsilon}} \bullet \underset{\sim}{\boldsymbol{a}} = \underset{\sim}{\boldsymbol{a}} \times \underset{\approx}{\boldsymbol{I}} \quad \Leftrightarrow \quad A_{ij} = -\varepsilon_{ijk} a_k \quad \Leftrightarrow \quad [A] = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{3.35}$$

If the axial *tensor* is known, then the axial *vector* $\underset{\sim}{\boldsymbol{a}}$ can be computed immediately by

$$a_1 = A_{32}, \tag{3.36a}$$

$$a_2 = A_{13} \tag{3.36b}$$

$$a_3 = A_{21} \tag{3.36c}$$

This operation may be written in direct notation as

$$\underset{\sim}{\boldsymbol{a}} = -\frac{1}{2}\underset{\approx}{\boldsymbol{\varepsilon}} : \underset{\approx}{\boldsymbol{A}}, \tag{3.37}$$

The indicial form is

$$a_i = -\varepsilon_{ijk} A_{jk} \tag{3.38}$$

For many calculations, it's useful to note that

$$\underset{\sim}{\boldsymbol{a}} \times \underset{\sim}{\boldsymbol{x}} = \underset{\approx}{\boldsymbol{A}} \bullet \underset{\sim}{\boldsymbol{x}} \qquad \text{for any vector } \underset{\sim}{\boldsymbol{x}} \tag{3.39}$$

Taking $\underset{\sim}{\boldsymbol{x}} = \underset{\sim}{\boldsymbol{a}}$ as a special case shows that

$$\underset{\approx}{\boldsymbol{A}} \bullet \underset{\sim}{\boldsymbol{a}} = \underset{\sim}{\boldsymbol{0}}. \tag{3.40}$$

The axial tensor is skew symmetric. That is,

$$\underset{\approx}{\boldsymbol{A}}^T = -\underset{\approx}{\boldsymbol{A}} \tag{3.41}$$

Axial tensors have interesting properties when they are raised to various powers. For example, even though $\underset{\approx}{\boldsymbol{A}}$ is skew-symmetric, its *square* $\underset{\approx}{\boldsymbol{A}}^2$ turns out to be symmetric.* Specifically:

$$\boxed{\underset{\approx}{\boldsymbol{A}}^2 = \underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}} - \underset{\approx}{\boldsymbol{I}}} \tag{3.42}$$

If applied to an arbitrary vector $\underset{\sim}{\boldsymbol{x}}$, this linear operator gives the *negative* projection of $\underset{\sim}{\boldsymbol{x}}$ onto the plane whose normal is $\underset{\sim}{\boldsymbol{a}}$. Replacing $\underset{\sim}{\boldsymbol{x}}$ by $\underset{\sim}{\boldsymbol{a}} \times \underset{\sim}{\boldsymbol{x}}$ in Eq. (3.39) shows that

$$\underset{\sim}{\boldsymbol{a}} \times (\underset{\sim}{\boldsymbol{a}} \times \underset{\sim}{\boldsymbol{x}}) = \underset{\approx}{\boldsymbol{A}}^2 \bullet \underset{\sim}{\boldsymbol{x}} \qquad \text{for any vector } \underset{\sim}{\boldsymbol{x}} \tag{3.43}$$

* This is a nice counterexample to demonstrate that the "square root" of a symmetric tensor does not necessarily have to be symmetric!

If we take the third power of the axial tensor, we obtain

$$\underset{\approx}{\boldsymbol{A}}^3 = \underset{\approx}{\boldsymbol{A}} \bullet \underset{\approx}{\boldsymbol{A}}^2 = \underset{\approx}{\boldsymbol{A}} \bullet (\underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}} - \underset{\approx}{\boldsymbol{I}}) = (\underset{\approx}{\boldsymbol{A}} \bullet \underset{\sim}{\boldsymbol{a}})\underset{\sim}{\boldsymbol{a}} - \underset{\approx}{\boldsymbol{A}} \tag{3.44}$$

or, using Eq. (3.40),

$$\boxed{\underset{\approx}{\boldsymbol{A}}^3 = -\underset{\approx}{\boldsymbol{A}}} \tag{3.45}$$

Thus, the cube of the axial tensor is just the negative of the axial tensor itself. Higher powers are computed similarly and alternate between being skew-symmetric axial tensors and (negative or positive) symmetric projectors.

Recall the Euler-Rodrigues expression for the rotation:

$$\underset{\approx}{\boldsymbol{R}} = \cos\alpha\ (\underset{\approx}{\boldsymbol{I}} - \underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}}) + \underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}} + \sin\alpha\ \underset{\approx}{\boldsymbol{A}} \tag{3.46}$$

Applying the above identities, we note that

$$\underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\boldsymbol{A}} = \underset{\approx}{\boldsymbol{A}} \bullet \underset{\approx}{\boldsymbol{R}} = (\cos\alpha)\underset{\approx}{\boldsymbol{A}} + \sin\alpha(\underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}} - \underset{\approx}{\boldsymbol{I}}) \tag{3.47}$$

Some proofs are most easily performed when the 3-direction is aligned with the rotation axis $\underset{\sim}{\boldsymbol{a}}$, in which case

$$[\underset{\approx}{\boldsymbol{R}}] = \begin{bmatrix} c & -s & 0 \\ s & c & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ and } [\underset{\approx}{\boldsymbol{A}}] = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.48}$$

where $c = \cos\alpha$ and $s = \sin\alpha$.

Argyris's form of the Euler-Rodrigues formula. Recalling Eqs. (3.39) and (3.43), a simpler (but computationally less efficient) form for the Euler-Rodrigues formula follows from Eq. 3.13:

$$\boxed{\underset{\approx}{\boldsymbol{R}} = \underset{\approx}{\boldsymbol{I}} + \sin\alpha\ \underset{\approx}{\boldsymbol{A}} + 2\sin^2\left(\frac{\alpha}{2}\right)\ \underset{\approx}{\boldsymbol{A}}^2} \tag{3.49}$$

or

$$\boxed{\underset{\approx}{\boldsymbol{R}} = \underset{\approx}{\boldsymbol{I}} + \sin\alpha\ \underset{\approx}{\boldsymbol{A}} + (1-\cos\alpha)\ \underset{\approx}{\boldsymbol{A}}^2} \tag{3.50}$$

Argyris [2] introduces a "pseudo-rotation vector" defined by

$$\underset{\sim}{\boldsymbol{s}} \equiv \alpha\,\underset{\sim}{\boldsymbol{a}}\,. \tag{3.51}$$

Using the trigonometric identity $\cos\alpha = 1 - 2\sin^2(\alpha/2)$, Eq. (3.14) can be written in terms of this pseudo rotation vector as

$$\underset{\approx}{\boldsymbol{R}} = \underset{\approx}{\boldsymbol{I}} + 2\sin^2\left(\frac{\alpha}{2}\right)(\underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}} - \underset{\approx}{\boldsymbol{I}}) + \sin\alpha\ \underset{\approx}{\boldsymbol{A}} \tag{3.52}$$

Of course, Argyris emphasizes the caveats that two successive rotations cannot by represented by simply adding the pseudo-rotation vectors. Nonetheless, the product $\alpha\underset{\sim}{\boldsymbol{a}}$ is certainly well-defined and (in some instances) more convenient.

Argyris introduces a corresponding skew-symmetric tensor $\underset{\approx}{\boldsymbol{S}}$ associated with $\underset{\sim}{\boldsymbol{s}}$ defined such that

$$\underset{\sim}{\boldsymbol{s}} \times \underset{\sim}{\boldsymbol{x}} = \underset{\approx}{\boldsymbol{S}} \bullet \underset{\sim}{\boldsymbol{x}} \text{ for any vector } \underset{\sim}{\boldsymbol{x}} \tag{3.53}$$

Comparing this definition with Eq. (3.39), we note that

$$\underset{\approx}{\boldsymbol{A}} = \frac{\underset{\approx}{\boldsymbol{S}}}{\alpha} \tag{3.54}$$

Thus, in terms of Argyris's pseudo-rotation tensor, Eq. (3.49) can be written

$$\boxed{\underset{\approx}{\boldsymbol{R}} = \underset{\approx}{\boldsymbol{I}} + \frac{\sin\alpha}{\alpha}\ \underset{\approx}{\boldsymbol{S}} + \frac{1}{2}\frac{\sin^2(\alpha/2)}{(\alpha/2)^2}\ \underset{\approx}{\boldsymbol{S}}^2} \tag{3.55}$$

We shall later see that this expression is most convenient when expanding the rotation tensor with respect to the rotation angle.

Corollary to the Euler-Rodrigues formula: Existence of a preferred basis

Recall Eq. (3.19) for the Euler-Rodrigues expression for a rotation tensor expressed solely in terms of the angle α and axis $\underset{\sim}{\boldsymbol{a}}$ of rotation:

$$[\underset{\approx}{\boldsymbol{R}}] = \cos\alpha \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + (1-\cos\alpha) \begin{bmatrix} a_1a_1 & a_1a_2 & a_1a_3 \\ a_2a_1 & a_2a_2 & a_2a_3 \\ a_3a_1 & a_3a_2 & a_3a_3 \end{bmatrix} + \sin\alpha \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{3.56}$$

The above expression holds no matter what basis is used. If, however, one sets up a basis such that the 3-direction is coincident with $\underset{\sim}{\boldsymbol{a}}$ then

$$a_1 = a_2 = 0 \text{ and } a_3 = 1, \tag{3.57}$$

In this case, the Euler-Rodrigues formula simplifies to

$$[\underset{\approx}{\boldsymbol{R}}] = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ when the 3-direction is aligned with } \underset{\sim}{\boldsymbol{a}}. \tag{3.58}$$

In other words, it is always possible to set up a basis such that the component matrix of the rotation is given by the above expression. This can be very useful in many situations. For example, we can take the trace of Eq. (3.58) to obtain

$$\text{tr}\,\underset{\approx}{\boldsymbol{R}} = 1 + 2\cos\alpha \tag{3.59}$$

Even though this formula was obtained by using the expression for $\underset{\approx}{\boldsymbol{R}}$ in its preferred basis, we know that it holds for *any* basis because the trace operation is an invariant. Hence, as exploited in the next section, the above formula provides a means of computing the angle of rotation when only the rotation matrix is known.

Computing axis and angle given the rotation tensor.

Now consider the reverse problem. Suppose you have the rotation tensor $\boldsymbol{\underset{\approx}{R}}$ and you want the axis and angle of rotation. Actually, saying "the axis" is misleading. More correctly, we should say "an axis." After all, if the rotation angle is zero, then the axis is arbitrary (there's an infinite number of solutions). If a rotation angle α is measured by right-hand-rule about a rotation direction $\boldsymbol{\underset{\sim}{a}}$, then an angle of $-\alpha$ about $-\boldsymbol{\underset{\sim}{a}}$ will produce the *same* rotation tensor. Therefore, the solution for the axis and angle of rotation is not unique. For nonzero rotation angles, there are always two solutions. We know that either $\boldsymbol{\underset{\sim}{a}}$ or $-\boldsymbol{\underset{\sim}{a}}$ serves equally well as the rotation axis. With one choice, the rotation angle ranges from 0 to π (plus or minus an integral multiple of 2π). For the other choice, the rotation angle will range from 0 to -2π (again plus or minus an integral multiple of 2π). The **principal solution** will be the one such that

$$0 \le \alpha \le \pi, \text{ measured by right-hand-rule about } \boldsymbol{\underset{\sim}{a}}. \tag{3.60}$$

Once this principal solution is found, then the family of general solutions for the axis and angle of rotation are of the form

$$\begin{aligned} &2\pi m + \alpha \text{ measured by right-hand-rule about } \boldsymbol{\underset{\sim}{a}}, \text{ and} \\ &2\pi m - \alpha \text{ measured by right-hand-rule about } -\boldsymbol{\underset{\sim}{a}} \end{aligned} \tag{3.61}$$

where m is any integer.

Finding the principal rotation angle. Equation (3.59) gives

$$\boxed{\cos\alpha = \frac{\operatorname{tr}\boldsymbol{\underset{\approx}{R}} - 1}{2} = \frac{1}{2}(R_{11} + R_{22} + R_{33} - 1)} \tag{3.62}$$

The inverse cosine has *two* solutions on the interval from 0 to 2π, so simply knowing the cosine is usually not sufficient to determine an angle. However, the inverse cosine *does* uniquely determine the *principal* rotation angle of Eq. (3.60) because that angle is required to range from 0 to π. Thus, the above expression determines the *principal* angle of rotation. When using the *principal* angle of rotation, it is essential to select the direction of the axis ($\boldsymbol{\underset{\sim}{a}}$ or $-\boldsymbol{\underset{\sim}{a}}$) consistently. You have to match the principal rotation angle to the principal axis. Matching the principal rotation angle with an oppositely-oriented axis will give the wrong result if you try to reconstruct the rotation tensor using the Euler-Rodrigues formula.

Finding the principal rotation axis. Recall the Euler-Rodrigues expression for the rotation:

$$\underset{\approx}{\boldsymbol{R}} = (\cos\alpha)\underset{\approx}{\boldsymbol{I}} + (1-\cos\alpha)\underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}} + \sin\alpha\ \underset{\approx}{\boldsymbol{A}} \tag{3.63}$$

First we are going to show how to invert this formula to obtain $\underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}}$ as a function of $\underset{\approx}{\boldsymbol{R}}$. Then we are going to explain why knowing *only* $\underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}}$ is not adequate for deducing the *principal* axis $\underset{\sim}{\boldsymbol{a}}$. Then we will move on to invert the above formula for $\underset{\approx}{\boldsymbol{A}}$ as a function of $\underset{\approx}{\boldsymbol{R}}$, which in turn gives the correct formula for the *principal* axis.

Recalling that the axial tensor $\underset{\approx}{\boldsymbol{A}}$ is skew-symmetric, note that the symmetric part of the rotation (i.e., $\operatorname{sym}\underset{\approx}{\boldsymbol{R}} \equiv \frac{1}{2}(\underset{\approx}{\boldsymbol{R}} + \underset{\approx}{\boldsymbol{R}}^T)$) is given by

$$\operatorname{sym}\underset{\approx}{\boldsymbol{R}} = (\cos\alpha)\underset{\approx}{\boldsymbol{I}} + (1-\cos\alpha)\underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}} \tag{3.64}$$

solving for the dyad $\underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}}$ gives

$$\underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}} = \frac{\operatorname{sym}\underset{\approx}{\boldsymbol{R}} - (\cos\alpha)\underset{\approx}{\boldsymbol{I}}}{(1-\cos\alpha)} \tag{3.65}$$

In the degenerate case of a zero rotation angle, we know that $\cos\alpha = 1$ and the rotation tensor is just the identity tensor. For this special case, Eq. (3.65) is an indeterminate 0/0 form. However, we know that the rotation axis is arbitrary whenever the rotation angle is zero. Hence, it may be set equal to $\underset{\sim}{\boldsymbol{e}}_1$ without loss. The remainder of this discussion concerns the nontrivial case of nonzero rotation angles. Recall from Eq. (3.62) that

$$\cos\alpha = \frac{\operatorname{tr}\underset{\approx}{\boldsymbol{R}} - 1}{2} \tag{3.66}$$

Hence, we obtain an expression for the dyad $\underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}}$:

$$\boxed{\underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}} = \frac{\underset{\approx}{\boldsymbol{R}} + \underset{\approx}{\boldsymbol{R}}^T + (1-\operatorname{tr}\underset{\approx}{\boldsymbol{R}})\underset{\approx}{\boldsymbol{I}}}{3-\operatorname{tr}\underset{\approx}{\boldsymbol{R}}}} \tag{3.67}$$

Recall from Eq. (3.17), that

$$[\underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}}] = \begin{bmatrix} a_1a_1 & a_1a_2 & a_1a_3 \\ a_2a_1 & a_2a_2 & a_2a_3 \\ a_3a_1 & a_3a_2 & a_3a_3 \end{bmatrix} \tag{3.68}$$

Hence, any non-zero column (or row) of $[\underset{\sim}{a}\,\underset{\sim}{a}]$ must be parallel to the rotation axis $\underset{\sim}{a}$. Since we must select a nonzero column, we know that we will be required to compute the magnitude of the column. Hence, it is computationally more efficient to ignore the denominator in Eq. (3.67) and simply say that the rotation axis can be obtained by normalizing any nonzero column (or row) of the numerator in Eq. (3.67):

$$\underset{\approx}{R} + \underset{\approx}{R}^T + (1 - \mathrm{tr}\,\underset{\approx}{R})\,\underset{\approx}{I} \qquad \leftarrow \textit{see warning below!} \tag{3.69}$$

WARNING: A key disadvantage of solving the dyad $\underset{\sim}{a}\,\underset{\sim}{a}$ for the axis $\underset{\sim}{a}$ is that it does not give any information about the *direction* of the axis vector $\underset{\sim}{a}$. If the principal inverse cosine is to be used to find the rotation angle α from Eq. (3.66), then the result will be an angle that ranges from 0 to π. Hence, we *must* select the direction of the axial vector $\underset{\sim}{a}$ consistently. After normalizing a nonzero column of Eq. (3.69) to obtain a *trial* rotation axis $\underset{\sim}{a}$, we might be required to change the answer to $-\underset{\sim}{a}$ in order to be consistent with our assumption that the rotation angle lies between 0 and π. Consider, for example, a rotation obtained by a right-handed rotation of 270° about the $\underset{\sim}{e}_3$ axis:

$$[\underset{\approx}{R}] = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.70}$$

We know that 270° about the $\underset{\sim}{e}_3$ axis is *not* the principal solution for the angle and axis of rotation because 270° does not lie in the interval from 0 to 180°. However, Eq. (3.70) is equivalent to a 90° rotation about the $-\underset{\sim}{e}_3$ direction, and *this* is the principal solution that we seek.

First, applying Eq. (3.66) gives

$$\alpha = \mathrm{ArcCos}\left(\frac{\mathrm{tr}\,\underset{\approx}{R} - 1}{2}\right) = \mathrm{ArcCos}(0) = 90°, \tag{3.71}$$

which correctly corresponds to the principal rotation angle. The expression in Eq. (3.69) becomes

$$[\underset{\approx}{R} + \underset{\approx}{R}^T + (1 - \mathrm{tr}\,\underset{\approx}{R})\,\underset{\approx}{I}] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 2 \end{bmatrix} \tag{3.72}$$

Normalizing the only nonzero column gives

$$\begin{Bmatrix} 0 \\ 0 \\ 1 \end{Bmatrix} \tag{3.73}$$

which is the wrong result. The correct answer for the *principal* rotation axis is

$$\begin{Bmatrix} 0 \\ 0 \\ -1 \end{Bmatrix}, \tag{3.74}$$

At this point, we do not have any criterion that indicates that the sign of Eq. (3.73) needs to be changed! Hence, attempting to find $\underset{\sim}{\boldsymbol{a}}$ from only the dyad $\underset{\sim}{\boldsymbol{a}}\underset{\sim}{\boldsymbol{a}}$ is a dead end. Of course, one could always use the Euler-Rodrigues formula to reconstruct the rotation tensor using Eq. (3.73). Then the axis direction could be reversed if the result does not equal the original rotation tensor. Again, this approach entails numerical round-off problems when trying to assess equality. We now discuss a second approach.

We have seen that solving the dyad $\underset{\sim}{\boldsymbol{a}}\underset{\sim}{\boldsymbol{a}}$ for $\underset{\sim}{\boldsymbol{a}}$ is not a satisfactory method for obtaining the *principal* rotation axis. Below, we discuss an alternative method that instead finds the *axial tensor* $\underset{\approx}{\boldsymbol{A}}$, from which the principal rotation axis may be found by applying Eq. (3.36). Again recall the Euler-Rodrigues expression for the rotation:

$$\underset{\approx}{\boldsymbol{R}} = (\cos\alpha)\underset{\approx}{\boldsymbol{I}} + (1-\cos\alpha)\underset{\sim}{\boldsymbol{a}}\underset{\sim}{\boldsymbol{a}} + \sin\alpha\ \underset{\approx}{\boldsymbol{A}} \tag{3.75}$$

This time, we will take the *skew-symmetric part* (i.e., $\operatorname{skw}\underset{\approx}{\boldsymbol{R}} \equiv \frac{1}{2}(\underset{\approx}{\boldsymbol{R}} - \underset{\approx}{\boldsymbol{R}}^T)$ to obtain

$$\operatorname{skw}\underset{\approx}{\boldsymbol{R}} = \frac{1}{2}(\underset{\approx}{\boldsymbol{R}} - \underset{\approx}{\boldsymbol{R}}^T) = (\sin\alpha)\underset{\approx}{\boldsymbol{A}} \tag{3.76}$$

Thus,

$$\underset{\approx}{\boldsymbol{A}} = \frac{1}{2(\sin\alpha)}(\underset{\approx}{\boldsymbol{R}} - \underset{\approx}{\boldsymbol{R}}^T) \tag{3.77}$$

Applying Eq. (3.36) gives the principal rotation axis:

$$a_1 = \frac{1}{2(\sin\alpha)}(R_{32} - R_{23}) \tag{3.78a}$$

$$a_2 = \frac{1}{2(\sin\alpha)}(R_{13} - R_{31}) \tag{3.78b}$$

$$a_3 = \frac{1}{2(\sin\alpha)}(R_{21} - R_{12}) \tag{3.78c}$$

The above formula gives the correct answer for the rotation axis regardless of whether or not the rotation angle is the *principal* angle. When we seek the *principal* axis, then we know that $\sin\alpha > 0$ and it may therefore be computed by the *positive* square root in the following formula

$$\sin\alpha = +\sqrt{1-\cos^2\alpha} = +\sqrt{1-\left(\frac{\operatorname{tr}\underset{\approx}{\boldsymbol{R}}-1}{2}\right)^2} \tag{3.79}$$

For the special case that $\sin\alpha = 0$ Eq. (3.78) becomes an indeterminate 0/0 form. In this case, we know that the rotation angle is either 0 (in which case the rotation axis is arbitrary) or the rotation angle is 180°, and the directional sense of rotation axis is irrelevant. For this special case, the method of Eq. (3.69) may be applied. We note that the rotation tensor is symmetric whenever the rotation angle is exactly 180° so the rotation axis may be take as the normalization of any nonzero row of $\boldsymbol{\underset{\approx}{R}} + \boldsymbol{\underset{\approx}{I}}$.

Method 1 algorithm for axis and angle of rotation. Given $\boldsymbol{\underset{\approx}{R}}$, below we provide an algorithm for finding the principal axis $\boldsymbol{\underset{\sim}{a}}$ and principal angle α of rotation:

STEP 1. Compute the cosine of the angle of rotation:

$$c = \frac{1}{2}(R_{11} + R_{22} + R_{33} - 1) \tag{3.80}$$

STEP 2. Decide whether the angle is an integral multiple of 180° by checking the cosine and follow one of these branches:

- If $c \neq \pm 1$, proceed to step 3.
- If $c = 1$ then $\boldsymbol{\underset{\approx}{R}} = \boldsymbol{\underset{\approx}{I}}$. Hence, the angle of rotation is 0 and the axis of rotation is arbitrary (so you can set it to {1,0,0} or any other convenient unit vector}. Go to step 6.
- If $c = -1$, then the angle of rotation is 180° (= π radians). To find the axis of rotation, simply normalize any *nonzero* column of $\boldsymbol{\underset{\approx}{R}} + \boldsymbol{\underset{\approx}{I}}$. Go to step 6.

STEP 3. Compute the angle of rotation:

$$\alpha = \text{ArcCos}(c) \tag{3.81}$$

Note: in this step, we are forcing $0 \le \alpha < 180°$ because that is the range of the arc-cosine function on all computing platforms. This restriction entails no loss in generality because larger or negative angles about some particular axis $\boldsymbol{\underset{\sim}{a}}$ can always be expressed as an angle between 0 and 180° about an *oppositely oriented* axis, $-\boldsymbol{\underset{\sim}{a}}$. The steps below ensure that we select the axis orientation correctly.

STEP 4. Compute the sine of the angle:

$$s = +\sqrt{1 - c^2} \tag{3.82}$$

STEP 5. Compute the axis of rotation:

$$a_1 = \frac{1}{2s}(R_{32} - R_{23}) \tag{3.83a}$$

$$a_2 = \frac{1}{2s}(R_{13} - R_{31}) \tag{3.83b}$$

$$a_3 = \frac{1}{2s}(R_{21} - R_{12}) \tag{3.83c}$$

(Note: The check in step 2 guarantees that s will not be zero.)

STEP 6. Stop.

This algorithm is coded in **Listing 3 (Converting direction cosines to axis and angle) on page A-3**.

Example. Let's use this algorithm to recover the axis and angle of rotation from Eq. (3.22). We are given

$$[\underset{\approx}{\boldsymbol{R}}] = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{3.84}$$

Follow the steps of the algorithm:

STEP 1. Compute the cosine of the angle:

$$c = \frac{1}{2}(0 + 0 + 0 - 1) = -\frac{1}{2} \tag{3.85}$$

STEP 2. Note: $c \neq \pm 1$; therefore proceed.

STEP 3. Compute the angle of rotation:

$$\alpha = \operatorname{acos}\left(-\frac{1}{2}\right) = 120^\circ \tag{3.86}$$

STEP 4. Compute the sine of the angle:

$$s = \sqrt{1 - \left(-\frac{1}{2}\right)^2} = \frac{\sqrt{3}}{2} \tag{3.87}$$

STEP 5. Compute the axis of rotation $\underset{\sim}{\boldsymbol{a}}$:

$$a_1 = \frac{1}{2(\sqrt{3}/2)}(1 - 0) \tag{3.88a}$$

$$a_2 = \frac{1}{2(\sqrt{3}/2)}(1 - 0) \tag{3.88b}$$

$$a_3 = \frac{1}{2(\sqrt{3}/2)}(1 - 0) \tag{3.88c}$$

or

$$\{\underset{\sim}{\boldsymbol{a}}\} = \frac{1}{\sqrt{3}}\begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} \tag{3.89}$$

Another example. Let's use the algorithm to recover the axis and angle of rotation from Eq. (3.24). We are given

$$[\underset{\approx}{\boldsymbol{R}}] = \begin{bmatrix} -1/3 & 2/3 & 2/3 \\ 2/3 & -1/3 & 2/3 \\ 2/3 & 2/3 & -1/3 \end{bmatrix} \tag{3.90}$$

STEP 1.Compute the cosine of the angle:

$$c = \frac{1}{2}\left(-\frac{1}{3}-\frac{1}{3}-\frac{1}{3}-1\right) = -1 \tag{3.91}$$

STEP 2.Note that $c = -1$, so the angle of rotation is 180˚. To find the axis of rotation, we must construct the matrix $\underset{\approx}{\boldsymbol{R}} + \underset{\approx}{\boldsymbol{I}}$:

$$[\underset{\approx}{\boldsymbol{R}} + \underset{\approx}{\boldsymbol{I}}] = \begin{bmatrix} -1/3 & 2/3 & 2/3 \\ 2/3 & -1/3 & 2/3 \\ 2/3 & 2/3 & -1/3 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 2/3 & 2/3 & 2/3 \\ 2/3 & 2/3 & 2/3 \\ 2/3 & 2/3 & 2/3 \end{bmatrix} \tag{3.92}$$

Thus, the axis of rotation is a unit vector in the direction of {2/3,2/3,2/3}. In other words, $\{\underset{\sim}{\boldsymbol{a}}\} = \{1,1,1\}/\sqrt{3}$. In this particular case *all* the columns turned out to be nonzero, but that won't be true in general. Nor will they typically be identical.

Numerical Example (for testing codes). Now consider the numerical rotation tensor of Eq. (3.30):

$$[\underset{\approx}{\boldsymbol{R}}] = \begin{bmatrix} 0.254506 & -0.834834 & 0.488138 \\ 0.915374 & 0.370785 & 0.156873 \\ -0.311957 & 0.406903 & 0.858552 \end{bmatrix} \tag{3.93}$$

We wish to recover the axis and angle of rotation.

Following the algorithm:

STEP 1. Compute the cosine of the angle:

$$c = \frac{1}{2}(0.254506 + 0.370785 + 0.858552 - 1) = 0.241992 \tag{3.94}$$

STEP 2. Note: $c \neq \pm 1$; therefore proceed.

STEP 3. Compute the angle:

$$\alpha = \text{acos}(0.241992) = 1.32645 \text{ radians} = 76 \text{ degrees} \tag{3.95}$$

STEP 4. Compute the sine of the angle:

$$s = \sqrt{1 - (0.241992)^2} = 0.970296 \tag{3.96}$$

STEP 5. Compute the axis of rotation:

$$a_1 = \frac{1}{2(0.970296)}(0.406903 - 0.156873) = 0.128842 \tag{3.97a}$$

$$a_2 = \frac{1}{2(0.970296)}(0.488138 - (-0.311957)) = 0.412294 \tag{3.97b}$$

$$a_3 = \frac{1}{2(0.970296)}(0.915374 - (-0.834834)) = 0.901894 \tag{3.97c}$$

Again, we wish to emphasize that the solution for the axis is unique only within a numerical sign. Changing the sign of the axis and angle will produce the same rotation tensor. In other words, the combination $(-\alpha, -\underset{\sim}{\boldsymbol{a}})$ is always a second solution.

Method 2 algorithm for computing axis and angle. One disadvantage of the above algorithm is that special treatment required when the angle is 0 or 180° degrees. One might expect some numerical round-off problems when the angle is *nearly* equal to these special cases. The following alternative algorithm better protects against these round-off errors and has no special cases, but it is less efficient. This algorithm computes the axis by using Eq. (3.67), and then assigns the axis orientation so that the rotation angle will range from 0 to 180°.

STEP 1. Compute the cosine of the angle of rotation:

$$c = \frac{1}{2}(R_{11} + R_{22} + R_{33} - 1) \tag{3.98}$$

STEP 2. Compute the angle of rotation:

$$\alpha = \text{ArcCos}(c) \tag{3.99}$$

STEP 3. Construct the matrix

$$\underset{\approx}{\boldsymbol{R}} + \underset{\approx}{\boldsymbol{R}}^T - 2c\underset{\approx}{\boldsymbol{I}} \tag{3.100}$$

STEP 4. Let $\underset{\sim}{\boldsymbol{x}}$ be the column of the above matrix having the largest magnitude.

STEP 5. Compute the axis of rotation:

$$a_1 = \frac{1}{x}\text{sign}(x_1, R_{32} - R_{23}) \tag{3.101}$$

$$a_2 = \frac{1}{x}\text{sign}(x_1, R_{13} - R_{31}) \tag{3.102}$$

$$a_3 = \frac{1}{x}\text{sign}(x_1, R_{21} - R_{12}) \tag{3.103}$$

Here,

$$x \equiv \sqrt{x_1^2 + x_2^2 + x_3^2} \tag{3.104}$$

Here, the "sign" function is defined by

$$\text{sign}(z, w) = \begin{cases} |z| & \text{if} \quad w \geq 0 \\ -|z| & \text{if} \quad w < 0 \end{cases} \tag{3.105}$$

This algorithm is coded in **Listing 3 (Converting direction cosines to axis and angle) on page A-3**.

4. Rotations contrasted with reflections

A tensor $\underset{\approx}{\boldsymbol{Q}}$ is orthogonal (or "idempotent") if and only if $\underset{\approx}{\boldsymbol{Q}}^T \bullet \underset{\approx}{\boldsymbol{Q}} = \underset{\approx}{\boldsymbol{I}}$. In other words, $\underset{\approx}{\boldsymbol{Q}}^{-1} = \underset{\approx}{\boldsymbol{Q}}^T$. Note that $\det \underset{\approx}{\boldsymbol{Q}} = \pm 1$. A *rotation* tensor $\underset{\approx}{\boldsymbol{R}}$ (also known as a *proper* orthogonal tensor) is a *special kind* of orthogonal tensor for which the determinate is positive. Thus, $\underset{\approx}{\boldsymbol{R}}^T \bullet \underset{\approx}{\boldsymbol{R}} = \underset{\approx}{\boldsymbol{I}}$ and $\det \underset{\approx}{\boldsymbol{R}} = +1$. It is often wrongly claimed that if $\underset{\approx}{\boldsymbol{Q}}$ is orthogonal with a negative determinant, then it must be a reflection. Before we can rebut this claim, we need to define what *reflection* means. Intuitively, if $\underset{\sim}{\boldsymbol{y}}$ is a reflection of $\underset{\sim}{\boldsymbol{x}}$, then $\underset{\sim}{\boldsymbol{x}}$ should be a reflection of $\underset{\sim}{\boldsymbol{y}}$. Mathematically, if $\underset{\sim}{\boldsymbol{y}} = \underset{\approx}{\Gamma} \bullet \underset{\sim}{\boldsymbol{x}}$, then the operator $\underset{\approx}{\Gamma}$ is a reflection if $\underset{\approx}{\Gamma} \bullet \underset{\sim}{\boldsymbol{y}} = \underset{\sim}{\boldsymbol{x}}$; i.e., a reflection is an operation $\underset{\approx}{\Gamma}$ whose inverse $\underset{\approx}{\Gamma}^{-1}$ is the operation $\underset{\approx}{\Gamma}$ itself:

$$\underset{\approx}{\Gamma} \bullet \underset{\approx}{\Gamma} = \underset{\approx}{\boldsymbol{I}}. \quad \text{or more compactly,} \quad \underset{\approx}{\Gamma}^2 = \underset{\approx}{\boldsymbol{I}} \tag{4.1}$$

Thus, a reflection is a "square root" of the identity tensor. There are an infinite number of such tensors. Even though the identity is symmetric, its square root does not have to be symmetric. For example,

$$[\underset{\sim}{\Gamma}] = \begin{bmatrix} 2 & -3 & 0 \\ 1 & -2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.2}$$

is a non-symmetric reflection tensor that satisfies Eq. (4.1). Reflection tensors are fundamentally different from orthogonal tensors. A tensor $\underset{\approx}{\boldsymbol{Y}}$ is a *proper* reflection (or an *improper* orthogonal tensor) if and only if it is an orthogonal reflection with a negative determinant:

$$\underset{\approx}{\boldsymbol{Y}} \bullet \underset{\approx}{\boldsymbol{Y}} = \underset{\approx}{\boldsymbol{I}} \quad \text{and} \quad \underset{\approx}{\boldsymbol{Y}}^T \bullet \underset{\approx}{\boldsymbol{Y}} = \underset{\approx}{\boldsymbol{I}} \quad \det \underset{\approx}{\boldsymbol{Y}} = -1 \tag{4.3}$$

To satisfy the first two expressions, a proper reflection $\underset{\approx}{\boldsymbol{Y}}$ must be symmetric. Even if we seek only symmetric square roots of the identity tensor, there are still an infinite number of answers. For tensors in 3D space, however, they are all expressible in two possible forms: (1) reflections about the origin $\underset{\approx}{\Gamma} = -\underset{\approx}{\boldsymbol{I}}$ and (2) reflections across a plane with unit normal $\underset{\sim}{\boldsymbol{n}}$ for which $\underset{\approx}{\Gamma} = \underset{\approx}{\boldsymbol{I}} - 2\underset{\sim}{\boldsymbol{n}}\underset{\sim}{\boldsymbol{n}}$. For a reflection about the origin, all three eigenvalues of $\underset{\approx}{\Gamma}$ equal -1 and the operation $\underset{\approx}{\Gamma} \bullet \underset{\sim}{\boldsymbol{x}}$ results in simply $-\underset{\sim}{\boldsymbol{x}}$. For a reflection across a plane, exactly one eigenvalue of $\underset{\approx}{\Gamma}$ equals -1 while the other two equal $+1$, and $\underset{\approx}{\Gamma} \bullet \underset{\sim}{\boldsymbol{x}}$ results in $\underset{\sim}{\boldsymbol{x}} - 2(\underset{\sim}{\boldsymbol{x}} \bullet \underset{\sim}{\boldsymbol{n}})\underset{\sim}{\boldsymbol{n}}$, which is like the mirror image of $\underset{\sim}{\boldsymbol{x}}$ across the plane perpendicular to $\underset{\sim}{\boldsymbol{n}}$. Note that $\underset{\approx}{\boldsymbol{I}} - 2\underset{\sim}{\boldsymbol{n}}\underset{\sim}{\boldsymbol{n}}$ can be written as $-\underset{\approx}{\boldsymbol{R}}^{(180n)}$, where $\underset{\approx}{\boldsymbol{R}}^{(180n)}$ is a proper rotation of 180° about $\underset{\sim}{\boldsymbol{n}}$.

Now that we have defined what is meant by a reflection, we can state that, for tensors in 3D space, any improper orthogonal tensor $\underset{\approx}{\boldsymbol{Q}}$ can be written in the form $-\underset{\approx}{\boldsymbol{R}}$, where $\underset{\approx}{\boldsymbol{R}}$ is a proper rotation. Thus any improper orthogonal tensor (i.e., one for which $\det\underset{\approx}{\boldsymbol{Q}} = -1$) can be written as a reflection across the origin *in combination with a proper rotation*.

5. Quaternion representation of a rotation

Rotation matrices have three independent components. This can be readily verified by recalling that the matrix is wholly determined by a unit axis of rotation (two independent variables) and an angle of rotation (the third independent variable). In this section we present the representation of a rotation in terms of a quaternion.*

Shoemake's form [3]. Any rotation operation may be represented by a quaternion $u_o + \boldsymbol{i}u_1 + \boldsymbol{j}u_2 + \boldsymbol{k}u_3$, for which the corresponding rotation matrix may be written in the form

$$[\underset{\approx}{\boldsymbol{R}}] = \begin{bmatrix} 1-2(u_2^2+u_3^2) & 2u_1u_2-2u_ou_3 & 2u_1u_3+2u_ou_2 \\ 2u_2u_1+2u_ou_3 & 1-2(u_3^2+u_1^2) & 2u_2u_3-2u_ou_1 \\ 2u_3u_1-2u_ou_2 & 2u_3u_3+2u_ou_1 & 1-2(u_1^2+u_2^2) \end{bmatrix} \tag{5.1}$$

where the four numbers $\{u_o, u_1, u_2, u_3\}$ satisfy

$$u_1^2 + u_2^2 + u_3^2 + u_o^2 = 1 \tag{5.2}$$

In other words, (u_o, u_1, u_2, u_3) is a point on a four-dimensional hypersphere.

A more structural direct form. The matrix in Eq. (5.1) may be written in alternatively as

$$[\underset{\approx}{\boldsymbol{R}}] = (2u_o^2-1)\begin{bmatrix} 1&0&0\\0&1&0\\0&0&1 \end{bmatrix} + 2\begin{bmatrix} u_1^2 & u_1u_2 & u_1u_3 \\ u_2u_1 & u_2^2 & u_2u_3 \\ u_3u_1 & u_3u_2 & u_3^2 \end{bmatrix} + 2u_o\begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix} \tag{5.3}$$

The corresponding direct notation is

$$\underset{\approx}{\boldsymbol{R}} = (2u_o^2-1)\underset{\approx}{\boldsymbol{I}} + 2\underset{\sim}{\boldsymbol{u}}\,\underset{\sim}{\boldsymbol{u}} - 2u_o\underset{\approx}{\boldsymbol{\varepsilon}} \bullet \underset{\sim}{\boldsymbol{u}}$$

$$\text{where } \underset{\sim}{\boldsymbol{u}} \equiv u_1\underset{\approx}{\boldsymbol{E}}_1 + u_2\underset{\approx}{\boldsymbol{E}}_2 + u_3\underset{\approx}{\boldsymbol{E}}_3 \tag{5.4}$$

For later use, note that

$$\underset{\sim}{\boldsymbol{u}} \bullet \underset{\sim}{\boldsymbol{u}} = 1 - u_o^2 \tag{5.5}$$

---

* A quaternion (also known as a "hypercomplex number") is the division algebra over the real numbers generated by the elements i, j, k subject to the relations $i^2=j^2=k^2=-1$, $ij=(-ji)=k$, $jk=(-kj)=i$, and $ki=(-ik)=j$. For our purposes, a quaternion is a four-dimensional vector.

Relationship between quaternion and axis/angle forms

Note from Eq. (5.4) that

$$\text{tr}\,\underset{\approx}{\boldsymbol{R}} = 4u_o^2 - 1 \tag{5.6}$$

Since the trace is invariant, we also know from Eq. (3.80) that

$$\text{tr}\,\underset{\approx}{\boldsymbol{R}} = 2\cos\alpha + 1 \tag{5.7}$$

Equating the last two expressions gives a relationship between u_o and α:

$$\cos\alpha = 2u_o^2 - 1 \tag{5.8}$$

The axial vector associated with $\underset{\approx}{\boldsymbol{R}}$ is

$$\text{ax}\,\underset{\approx}{\boldsymbol{R}} = 2u_o\underset{\sim}{\boldsymbol{u}} \tag{5.9}$$

We also know that the axial vector of $\underset{\approx}{\boldsymbol{R}}$ is given by

$$\text{ax}\,\underset{\approx}{\boldsymbol{R}} = (\sin\alpha)\underset{\sim}{\boldsymbol{a}} \tag{5.10}$$

Equating the last two equations shows that

$$\sin\alpha = 2u_o\sqrt{\underset{\sim}{\boldsymbol{u}}\bullet\underset{\sim}{\boldsymbol{u}}} = 2u_o\sqrt{1-u_o^2} \tag{5.11}$$

By virtue of Eq. (5.2), we may introduce **4-dimensional spherical coordinates** $\{\beta, \theta, \phi\}$ such that

$$\begin{aligned} u_o &= \cos\beta \\ u_1 &= \sin\beta(\sin\theta\cos\phi) \\ u_2 &= \sin\beta(\sin\theta\sin\phi) \\ u_3 &= \sin\beta(\cos\theta) \end{aligned} \tag{5.12}$$

Eqs. (5.8) and (5.11) therefore imply that

$$\beta = \frac{\alpha}{2} \tag{5.13}$$

and

$$\underset{\sim}{\boldsymbol{u}} = \left(\sin\frac{\alpha}{2}\right)\underset{\sim}{\boldsymbol{a}} \quad \text{if } \alpha \neq 0$$

($\underset{\sim}{\boldsymbol{u}}$ and $\underset{\sim}{\boldsymbol{a}}$ are both arbitrary if the angle of rotation is zero) (5.14)

Equations (5.12) through (5.14) provide a means of converting from axis/angle to quaternion form, and vice versa. The rotation angle α may permissibly take a full range of values from $-\pi$ to $+\pi$:

$$\text{If } -\pi \le \alpha \le \pi, \text{ then } -\frac{\pi}{2} \le \beta \le \frac{\pi}{2} \tag{5.15}$$

With this viewpoint, a rotation can be described as a point (unit quaternion) on a four-dimensional hypersphere with coordinates (u_o, u_1, u_2, u_3).

Without loss in generality it is always possible (by changing the direction of the rotation axis) to presume that the rotation angle satisfies $\sin\alpha \geq 0$. Thus,

$$\text{If } 0 \leq \alpha \leq \pi, \text{ then } 0 \leq \beta \leq \frac{\pi}{2} \tag{5.16}$$

In this case, a rotation is described by a unit quaternion lying on a spherical *hemisphere* in four dimensional space. Keep in mind that both viewpoints are legitimate, and both give the same unique rotations.

6. Dyad form of an invertible linear operator

Suppose that $\{\underset{\sim}{\boldsymbol{b}}_1, \underset{\sim}{\boldsymbol{b}}_2, \underset{\sim}{\boldsymbol{b}}_3\}$ is a set of linearly independent vectors. Suppose that some operator f is known to be linear and its action on the "b" vectors is known. Specifically, let

$$\underset{\sim}{\boldsymbol{c}}_k = f(\underset{\sim}{\boldsymbol{b}}_k) \tag{6.1}$$

Since the operation is known to be linear, then we know there must exist a tensor $\underset{\approx}{\boldsymbol{F}}$ such that the above expression may be written

$$\underset{\sim}{\boldsymbol{c}}_k = \underset{\approx}{\boldsymbol{F}} \bullet \underset{\sim}{\boldsymbol{b}}_k \tag{6.2}$$

If the six vectors $\{\underset{\sim}{\boldsymbol{b}}_1, \underset{\sim}{\boldsymbol{b}}_2, \underset{\sim}{\boldsymbol{b}}_3\}$ and $\{\underset{\sim}{\boldsymbol{c}}_1, \underset{\sim}{\boldsymbol{c}}_2, \underset{\sim}{\boldsymbol{c}}_3\}$ are known, then the tensor $\underset{\approx}{\boldsymbol{F}}$ is expressible as the following sum of dyads:

$$\underset{\approx}{\boldsymbol{F}} = \underset{\sim}{\boldsymbol{c}}_k \underset{\sim}{\boldsymbol{b}}^k \tag{6.3}$$

Here, $\{\underset{\sim}{\boldsymbol{b}}^1, \underset{\sim}{\boldsymbol{b}}^2, \underset{\sim}{\boldsymbol{b}}^3\}$ are the so-called "dual" vectors associated with the $\{\underset{\sim}{\boldsymbol{b}}_1, \underset{\sim}{\boldsymbol{b}}_2, \underset{\sim}{\boldsymbol{b}}_3\}$ basis. Specifically, the dual basis is defined by

$$\underset{\sim}{\boldsymbol{b}}^i \bullet \underset{\sim}{\boldsymbol{b}}_j = \delta^i_j, \tag{6.4}$$

where δ^i_j is the Kronecker delta. It follows that

$$\underset{\sim}{\boldsymbol{b}}^1 = \frac{\underset{\sim}{\boldsymbol{b}}_2 \times \underset{\sim}{\boldsymbol{b}}_3}{\underset{\sim}{\boldsymbol{b}}_1 \bullet (\underset{\sim}{\boldsymbol{b}}_2 \times \underset{\sim}{\boldsymbol{b}}_3)}, \qquad \underset{\sim}{\boldsymbol{b}}^2 = \frac{\underset{\sim}{\boldsymbol{b}}_3 \times \underset{\sim}{\boldsymbol{b}}_1}{\underset{\sim}{\boldsymbol{b}}_1 \bullet (\underset{\sim}{\boldsymbol{b}}_2 \times \underset{\sim}{\boldsymbol{b}}_3)}, \text{ and } \quad \underset{\sim}{\boldsymbol{b}}^3 = \frac{\underset{\sim}{\boldsymbol{b}}_1 \times \underset{\sim}{\boldsymbol{b}}_2}{\underset{\sim}{\boldsymbol{b}}_1 \bullet (\underset{\sim}{\boldsymbol{b}}_2 \times \underset{\sim}{\boldsymbol{b}}_3)} \tag{6.5}$$

SPECIAL CASE: lab basis. If the $\{\underset{\sim}{\boldsymbol{b}}_1, \underset{\sim}{\boldsymbol{b}}_2, \underset{\sim}{\boldsymbol{b}}_3\}$ basis happens to be orthonormal, then the dual basis is the same as the $\{\underset{\sim}{\boldsymbol{b}}_1, \underset{\sim}{\boldsymbol{b}}_2, \underset{\sim}{\boldsymbol{b}}_3\}$ basis. In particular, if the $\{\underset{\sim}{\boldsymbol{b}}_1, \underset{\sim}{\boldsymbol{b}}_2, \underset{\sim}{\boldsymbol{b}}_3\}$ basis is taken to be the lab basis $\{\underset{\sim}{\boldsymbol{E}}_1, \underset{\sim}{\boldsymbol{E}}_2, \underset{\sim}{\boldsymbol{E}}_3\}$, then

$$\underset{\approx}{\boldsymbol{F}} = \underset{\sim}{\boldsymbol{c}}_k \underset{\sim}{\boldsymbol{E}}_k, \tag{6.6}$$

where

$$\underset{\sim}{\boldsymbol{c}}_k = \underset{\approx}{\boldsymbol{F}} \bullet \underset{\sim}{\boldsymbol{E}}_k \tag{6.7}$$

This means that the k^{th} column of $[\underset{\approx}{\boldsymbol{F}}]$ with respect to the lab basis is given by the lab components of $\underset{\sim}{\boldsymbol{c}}_k$. Stated differently, if you know how $\underset{\approx}{\boldsymbol{F}}$ transforms the three lab base vectors, then you can immediately construct $[\underset{\approx}{\boldsymbol{F}}]$. Importantly, if $[F_{ij}]$ are the lab components of $\underset{\approx}{\boldsymbol{F}}$, then Eq. (6.8) may be written

$$\underset{\sim}{\boldsymbol{c}}_k = F_{jk}\underset{\sim}{\boldsymbol{E}}_j \tag{6.8}$$

Carefully note that Eqs. (6.7) and (6.8) express the *same result* in two different forms. There is *not* a transpose error on the subscripts of F_{jk} in Eq. (6.8).

SPECIAL CASE: rotation. Now to specialize this result to rotation, suppose that an orthonormal triad $\{\hat{\underset{\sim}{\boldsymbol{c}}}_1, \hat{\underset{\sim}{\boldsymbol{c}}}_2, \hat{\underset{\sim}{\boldsymbol{c}}}_3\}$ is obtained by applying a rotation $\underset{\approx}{\boldsymbol{R}}$ to an orthonormal triad $\{\hat{\underset{\sim}{\boldsymbol{b}}}_1, \hat{\underset{\sim}{\boldsymbol{b}}}_2, \hat{\underset{\sim}{\boldsymbol{b}}}_3\}$. Applying Eq. 6.3 (noting that $\hat{\underset{\sim}{\boldsymbol{b}}}^k = \hat{\underset{\sim}{\boldsymbol{b}}}_k$ for orthonormal triads), the dyad representation of the rotation is

$$\boxed{\underset{\approx}{\boldsymbol{R}} = \hat{\underset{\sim}{\boldsymbol{c}}}_k\hat{\underset{\sim}{\boldsymbol{b}}}_k}, \tag{6.9}$$

where

$$\hat{\underset{\sim}{\boldsymbol{c}}}_k = \underset{\approx}{\boldsymbol{R}} \bullet \hat{\underset{\sim}{\boldsymbol{b}}}_k \qquad \Leftrightarrow \qquad \hat{\underset{\sim}{\boldsymbol{c}}}_k = R_{jk}\hat{\underset{\sim}{\boldsymbol{b}}}_j \tag{6.10}$$

This result can be used to construct the rotation relating two configurations. If, for example, the edges of a cube are defined by three unit vectors $\{\hat{\underset{\sim}{\boldsymbol{b}}}_1, \hat{\underset{\sim}{\boldsymbol{b}}}_2, \hat{\underset{\sim}{\boldsymbol{b}}}_3\}$ and that cube is rotated to a new orientation so that the original three vectors become $\{\hat{\underset{\sim}{\boldsymbol{c}}}_1, \hat{\underset{\sim}{\boldsymbol{c}}}_2, \hat{\underset{\sim}{\boldsymbol{c}}}_3\}$ then the rotation tensor can be constructed using Eq. (6.9).

7. Sequential Rotations

Sequential rotations about fixed (laboratory) axes.

If a body is rigidly rotated by a rotation tensor $\underset{\approx}{\boldsymbol{R}}_1$ and then subsequently rotated by another rotation operation $\underset{\approx}{\boldsymbol{R}}_2$, then the total overall rotation tensor is given by

$$\underset{\approx}{\boldsymbol{R}} = \underset{\approx}{\boldsymbol{R}}_2 \bullet \underset{\approx}{\boldsymbol{R}}_1 \tag{7.1}$$

Any general rotation $\underset{\approx}{\boldsymbol{R}}$ can always be described in terms of three *sequential* rotations about the fixed laboratory axes $\{X, Y, Z\}$ through angles $\{\Theta_X, \Theta_Y, \Theta_Z\}$. These angles are not equivalent to Euler angles (which are discussed in the next section).

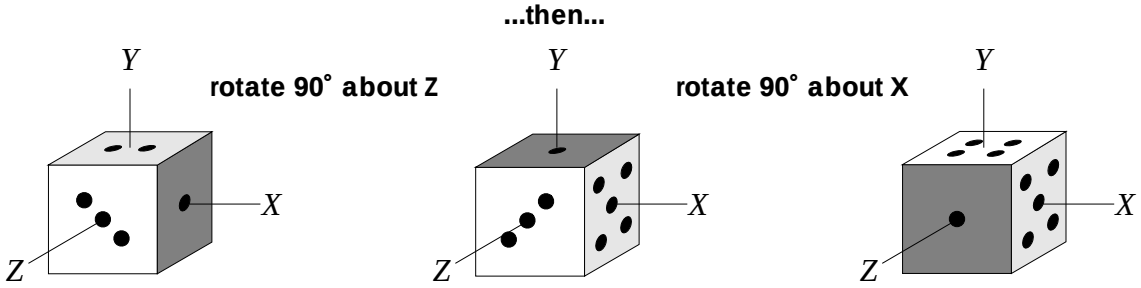


Figure 7.1. Sequential rotation about fixed axes Note how the block rotates relative to the fixed laboratory triad. The number of dots on opposite sides of honest dice always sum to seven (hence, the side with four dots is opposite the side with three dots).

Figure 7.1 shows a simple example of sequential laboratory-referenced rotations in which a block is rotated 90˚ about the fixed laboratory Z-axis and then 90˚ about the fixed X-axis. Using Eq. (2.18) for the first rotation about Z and Eq. (2.9) for the second rotation about X, and then combining them in Eq. (7.1), the total rotation tensor is

$$[R_{ij}] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix} \tag{7.2}$$

The overall axis and angle of rotation may be determined by using the algorithm on page 32.

Sequential rotation operations do not commute. That is, as shown in Fig. 7.2, when the rotation operations are applied in a different order, the final orientation of the block will generally be different. Namely, the rotation tensor corresponding to Fig. 7.2 is

$$[R_{ij}] = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \tag{7.3}$$

which is not equal to the rotation matrix given in Eq. (7.2). Many writers use this counterexample as "proof" that rotation is not a vector. The proper conclusion is simply that sequentially applied rotation operations do not commute. We have already shown that, through the Euler-Rodrigues formula, that rotation *can* be represented by a vector. The lack of commutativity of rotations simply proves that the rotation vector of a *sequential* rotation is not in general equal to the sum of the individual rotation vectors. On the other hand, as discussed later, *simultaneously* applied rotations *do* commute.

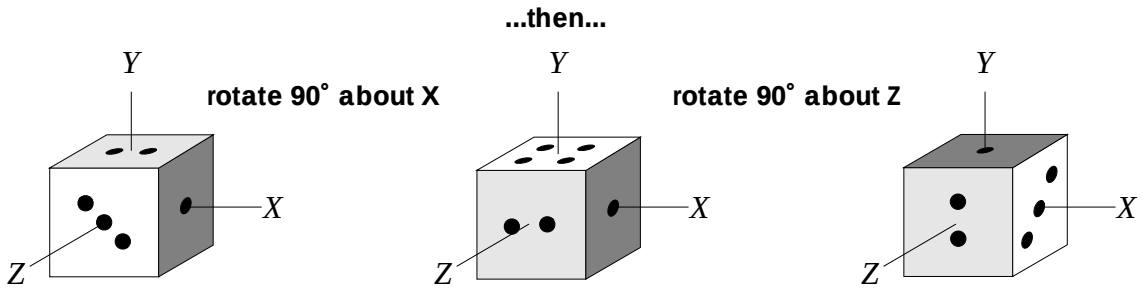


Figure 7.2. A different sequence for the rotations. Note that the final orientation of the block does not match that of Fig. 7.1, which proves that rotation operations do not commute.

EULER ANGLES: Sequential rotations about "follower" axes.

Euler angles are a commonly used — but awkward — way to describe any general rotation. Rather than defining the rotation relative to the fixed laboratory axes, Euler angles use the embedded triad $\{x, y, z\}$ of axes that *follow* the material as it rotates. The Euler angles $\{\varphi, \vartheta, \psi\}$ are defined [4] as follows: First rotate the embedded triad an angle φ about the z-axis (this causes x and y to move to new orientations while z remains unchanged). Then rotate the triad an angle ϑ about its new x-axis (this causes y and z to move to new orientations while x remains unchanged). Finally, rotate the triad an angle ψ about its new z-axis.

Converting Euler angles to direction cosines. The matrix of components of the rotation tensor $\underset{\approx}{\boldsymbol{R}}$ *with respect to the laboratory triad* is the product of the three *relative* rotation matrices as follows

$$[R_{ij}] = \begin{bmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\vartheta & -\sin\vartheta \\ 0 & \sin\vartheta & \cos\vartheta \end{bmatrix} \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{7.4}$$

See **Listing 4 (Converting Euler angles to direction cosines.) on page A-5**. Unlike Eq. (7.1) the above three matrices are multiplied *in the order of operation*. This distinction arises because Euler angle matrices are defined with respect to the *embedded* follower axes whereas Eq. (7.1) is defined with respect to the fixed *laboratory* axes.

Converting a direction cosine matrix back to Euler angles is more difficult and the solution isn't necessarily unique. One solution is provided in **Listing 5 (Converting direction cosines to Euler angles.) on page A-6.**

Example: Consider the particular set of Euler angles $\{\varphi, \vartheta, \psi\}=\{90°, 90°, 0\}$, illustrated in Fig. 7.3. Euler angles are based on rotations about the *embedded* follower triad. Note how Fig. 7.3 differs from Fig. 7.1 which uses the same set of rotation angles, but referenced to a *fixed* triad.

Using Eq. (7.4), the rotation tensor corresponding to Fig. 7.3 is

$$[R_{ij}] = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{7.5}$$

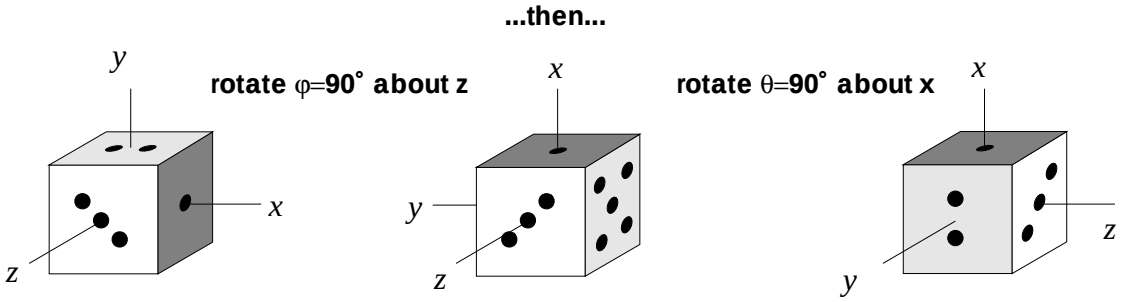


Figure 7.3. Example of Euler Angles. $\{\varphi, \vartheta, \psi\}=\{90°, 90°, 0\}$. The final rotation is quite different from the one defined with respect to the fixed laboratory triad.

Converting Euler angles to axis and angle. To convert from Euler angles to a single axis and angle of rotation, simply compute the rotation tensor using Eq. (7.4), and then use the algorithm on page 32. For example, the rotation described in Fig. 7.3 gave the to the rotation tensor of Eq. (7.5), which is identical to Eq. (3.84) for which the angle and axis were found to be 120˚ and $\{1,1,1\}/\sqrt{3}$.

8. Series expression for a rotation

Recall Eq. (3.55), which is Argyris's version of the Euler-Rodigues formula:

$$\boxed{\underset{\approx}{\boldsymbol{R}} = \underset{\approx}{\boldsymbol{I}} + \frac{\sin\alpha}{\alpha}\,\underset{\approx}{\boldsymbol{S}} + \frac{1}{2}\frac{\sin^2(\alpha/2)}{(\alpha/2)^2}\,\underset{\approx}{\boldsymbol{S}}^2} \tag{8.1}$$

This expression is valid for arbitrarily large rotation angles. This section is devoted to expanding the above expression with respect to the rotation angle.

Keep in mind that Argyris's pseudo-tensor $\underset{\approx}{\boldsymbol{S}}$ is defined

$$\underset{\approx}{\boldsymbol{S}} = \alpha\underset{\approx}{\boldsymbol{A}}, \tag{8.2}$$

where $A_{ij} = -\varepsilon_{ijk}a_k$. Thus, when forming expansions, it is important to keep in mind that $\underset{\approx}{\boldsymbol{S}}$ itself is proportional to (and therefore first order with respect to) the rotation angle α. Using the indicial expression for $\underset{\approx}{\boldsymbol{A}}$, we note that

$$\begin{aligned}
(\underset{\approx}{\boldsymbol{A}}^2)_{ij} &= A_{ip}A_{pj} \\
&= (-\varepsilon_{ipk}a_k)(-\varepsilon_{pjs}a_s) \\
&= (\varepsilon_{ipk}\varepsilon_{pjs})a_k a_s \\
&= (\delta_{kj}\delta_{is} - \delta_{ks}\delta_{ij})a_k a_s \\
&= a_i a_j - a_s a_s \delta_{ij} \\
&= a_i a_j - \delta_{ij}
\end{aligned} \tag{8.3}$$

In other words,

$$\boxed{\underset{\approx}{\boldsymbol{A}}^2 = \underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}} - \underset{\approx}{\boldsymbol{I}}} \tag{8.4}$$

Dotting both sides by $\underset{\approx}{\boldsymbol{A}}$ gives

$$\underset{\approx}{\boldsymbol{A}}^3 = (\underset{\approx}{\boldsymbol{A}} \bullet \underset{\sim}{\boldsymbol{a}})\underset{\sim}{\boldsymbol{a}} - \underset{\approx}{\boldsymbol{A}} \tag{8.5}$$

Now recall that, for *any* vector $\underset{\sim}{\boldsymbol{x}}$,

$$\underset{\approx}{\boldsymbol{A}} \bullet \underset{\sim}{\boldsymbol{x}} = \underset{\sim}{\boldsymbol{a}} \times \underset{\sim}{\boldsymbol{x}} \tag{8.6}$$

This holds for *any* vector $\underset{\sim}{\boldsymbol{x}}$, so we are allowed to consider the case in which $\underset{\sim}{\boldsymbol{x}}$ is the rotation axis $\underset{\sim}{\boldsymbol{a}}$ itself. Thus

$$\underset{\approx}{\boldsymbol{A}} \bullet \underset{\sim}{\boldsymbol{a}} = \underset{\sim}{\boldsymbol{0}} \tag{8.7}$$

Consequently, Eq. (8.5) becomes

$$\underset{\approx}{\boldsymbol{A}}^3 = -\underset{\approx}{\boldsymbol{A}} \tag{8.8}$$

Continuing in this manner, we find

$$\underset{\approx}{A}^4 = -\underset{\approx}{A}^2, \quad \underset{\approx}{A}^6 = \underset{\approx}{A}^2, \quad \underset{\approx}{A}^8 = -\underset{\approx}{A}^2, \quad \underset{\approx}{A}^{10} = \underset{\approx}{A}^2,$$
$$\underset{\approx}{A}^3 = -\underset{\approx}{A}, \quad \underset{\approx}{A}^5 = \underset{\approx}{A}, \quad \underset{\approx}{A}^7 = -\underset{\approx}{A}, \quad \underset{\approx}{A}^9 = \underset{\approx}{A}, \quad \text{etc.} \tag{8.9}$$

In other words, all odd powers of $\underset{\approx}{A}$ are equal to $\pm\underset{\approx}{A}$ and all even powers of $\underset{\approx}{A}$ are equal to $\pm\underset{\approx}{A}^2$. Recalling that $\underset{\approx}{S} = \alpha\underset{\approx}{A}$, this means that

$$\underset{\approx}{S}^4 = -\alpha^2\underset{\approx}{S}^2, \quad \underset{\approx}{S}^6 = \alpha^4\underset{\approx}{S}^2, \quad \underset{\approx}{S}^8 = -\alpha^6\underset{\approx}{S}^2, \quad \underset{\approx}{S}^{10} = \alpha^8\underset{\approx}{S}^2,$$
$$\underset{\approx}{S}^3 = -\alpha^2\underset{\approx}{S}, \quad \underset{\approx}{S}^5 = \alpha^4\underset{\approx}{S}, \quad \underset{\approx}{S}^7 = -\alpha^6\underset{\approx}{S}, \quad \underset{\approx}{S}^9 = \alpha^8\underset{\approx}{S}, \quad \text{etc.} \tag{8.10}$$

These expressions will prove useful as we expand the rotation tensor with respect to the rotation angle.

The first-order expansion of Eq. (8.1) for small rotation angles is

$$\underset{\approx}{R} = \underset{\approx}{I} + \underset{\approx}{S} \quad + \; O(\alpha^2) \tag{8.11}$$

The second order expansion is

$$\underset{\approx}{R} = \underset{\approx}{I} + \underset{\approx}{S} + \tfrac{1}{2}\underset{\approx}{S}^2 + O(\alpha^3) \tag{8.12}$$

In fact, Argyris shows that the infinite series expansion is

$$\boxed{\underset{\approx}{R} = \underset{\approx}{I} + \underset{\approx}{S} + \tfrac{1}{2!}\underset{\approx}{S}^2 + \tfrac{1}{3!}\underset{\approx}{S}^3 + \ldots + \tfrac{1}{n!}\underset{\approx}{S}^n + \ldots} \tag{8.13}$$

More compactly,

$$\boxed{\underset{\approx}{R} = e^{\underset{\sim}{S}} = \exp\underset{\approx}{S}} \tag{8.14}$$

This result may be deduced by arguments based on Lie's group theory as is done in quantum mechanics, but Argyris rightly asks "why use a steam hammer to crack a nut?"

Though intriguing and intoxicatingly compact, the above result is fraught with hidden pitfalls that limit its usefulness. Identities that apply for exponentials of scalars do not generalize for exponentials of tensors. For example,

$$e^{\underset{\sim}{S}_2} \bullet e^{\underset{\sim}{S}_1} \neq e^{\underset{\sim}{S}_2 + \underset{\sim}{S}_1} \neq e^{\underset{\sim}{S}_1} \bullet e^{\underset{\sim}{S}_2} \text{ unless } \underset{\approx}{S}_1 \text{ and } \underset{\approx}{S}_2 \text{ commute!} \tag{8.15}$$

The above statement is expressing the known fact that, given two rotations, $\underset{\approx}{R}_1 = e^{\underset{\sim}{S}_1}$ and $\underset{\approx}{R}_2 = e^{\underset{\sim}{S}_2}$, the *sequential* rotation $\underset{\approx}{R}_2 \bullet \underset{\approx}{R}_1$ is not generally equal to $\underset{\approx}{R}_1 \bullet \underset{\approx}{R}_2$ unless both rotations share the same axis of rotation.

9. Spectrum of a rotation

The eigenproblem for a rotation tensor requires determination of all eigenvectors $\underset{\sim}{p}$ and eigenvalues λ for which

$$\underset{\approx}{R} \bullet \underset{\sim}{p} = \lambda \underset{\sim}{p} \tag{9.1}$$

Suppose we set up a preferred basis $\{\underset{\sim}{b}_1, \underset{\sim}{b}_2, \underset{\sim}{b}_3\}$ for which the 3-direction is aligned with the axis of rotation $\underset{\sim}{a}$. Then the corresponding components of $\underset{\approx}{R}$ are

$$[\underset{\approx}{R}] = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{with respect to basis } \{\underset{\sim}{b}_1, \underset{\sim}{b}_2, \underset{\sim}{b}_3 = \underset{\sim}{a}\} \tag{9.2}$$

From this, we immediately see that $\underset{\sim}{p} = \underset{\sim}{a}$ is an eigenvector with associated eigenvalue $\lambda = 1$. For the remaining eigenpairs, we focus on the upper 2×2 submatrix. The characteristic equation for this submatrix is

$$\lambda^2 - 2(\cos\alpha)\lambda + 1 = 0 \tag{9.3}$$

Applying the quadratic formula gives a complex-conjugate pair of solutions:

$$\lambda = \cos\theta \pm i \sin\alpha = e^{\pm i\theta} \tag{9.4}$$

where $i \equiv \sqrt{-1}$. The corresponding eigenvectors are $\{\pm i, 1\}$. Thus, the spectrum of the rotation tensor is:

Table 9.1: Spectrum of a rotation tensor

eigenvalue	eigenvector
$\lambda_1 = e^{i\alpha}$	$\underset{\sim}{p}_1 = \frac{1}{\sqrt{2}}(i\underset{\sim}{b}_1 + \underset{\sim}{b}_2)$
$\lambda_2 = e^{-i\alpha}$	$\underset{\sim}{p}_2 = \frac{1}{\sqrt{2}}(-i\underset{\sim}{b}_1 + \underset{\sim}{b}_2)$
$\lambda_3 = 1$	$\underset{\sim}{p}_3 = \underset{\sim}{b}_3 \quad (= \underset{\sim}{a})$

It can be readily verified that the rotation tensor can be reconstructed by

$$\underset{\approx}{R} = \lambda_1(\underset{\sim}{p}_1 \otimes \overline{\underset{\sim}{p}}_1) + \lambda_2(\underset{\sim}{p}_2 \otimes \overline{\underset{\sim}{p}}_2) + \lambda_3(\underset{\sim}{p}_3 \otimes \overline{\underset{\sim}{p}}_3), \tag{9.5}$$

where the superposed "bar" denotes the conjugate. Furthermore, noting that $\lambda_2 = \bar{\lambda}_1$ and $\underset{\sim}{p}_2 = \overline{\underset{\sim}{p}}_1$, the above equation may be written

$$\underset{\approx}{R} = 2\mathrm{Re}[\lambda_1(\underset{\sim}{p}_1 \otimes \overline{\underset{\sim}{p}}_1)] + \lambda_3(\underset{\sim}{p}_3 \otimes \overline{\underset{\sim}{p}}_3). \tag{9.6}$$

Incidentally, for any nonzero rotation angle, the eigenvalues are distinct and the eigenvectors $\underset{\sim}{\boldsymbol{p}}_1$ and $\underset{\sim}{\boldsymbol{p}}_2$ are therefore unique to within a complex multiple even though it might appear otherwise in light of the following observation. Let $\underset{\approx}{\boldsymbol{Q}}$ be an arbitrary real rotation about $\underset{\sim}{\boldsymbol{a}}$. Then, because $\underset{\approx}{\boldsymbol{R}}$ and $\underset{\approx}{\boldsymbol{Q}}$ share the same rotation axis, we know that they commute:

$$\underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\boldsymbol{Q}} = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{R}} \tag{9.7}$$

Let $\underset{\sim}{\boldsymbol{p}}$ be an eigenvector of $\underset{\approx}{\boldsymbol{R}}$ with eigenvalue λ. Dotting both sides of Eq. (9.7) from the right by $\underset{\sim}{\boldsymbol{p}}$ gives

$$(\underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\boldsymbol{Q}}) \bullet \underset{\sim}{\boldsymbol{p}} = (\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{R}}) \bullet \underset{\sim}{\boldsymbol{p}} \tag{9.8}$$

or

$$\underset{\approx}{\boldsymbol{R}} \bullet (\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{p}}) = \underset{\approx}{\boldsymbol{Q}} \bullet (\underset{\approx}{\boldsymbol{R}} \bullet \underset{\sim}{\boldsymbol{p}}) = \underset{\approx}{\boldsymbol{Q}} \bullet (\lambda \underset{\sim}{\boldsymbol{p}}) = \lambda(\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{p}}) \tag{9.9}$$

This proves that $\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{p}}$ is an eigenvector of $\underset{\approx}{\boldsymbol{R}}$ corresponding to the eigenvalue λ. Stated differently, the choice of base vectors $\underset{\sim}{\boldsymbol{b}}_1$ and $\underset{\sim}{\boldsymbol{b}}_2$ is arbitrary so long as they are orthonormal and orthogonal to the rotation axis $\underset{\sim}{\boldsymbol{a}}$.

Sanity check. We know from elementary matrix analysis [*cf.*, Ref. 5] that the number of linearly independent eigenvectors must be no greater than the algebraic multiplicity of the eigenvalue. So long as $\sin\alpha \neq 0$, we know that the eigenvalues are all distinct. In this case, there can be only *one single* linearly independent eigenvector associated with each eigenvector. If $\underset{\sim}{\boldsymbol{p}}$ is an eigenvector, then Eq. (9.9) states that $\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{p}}$ is also an eigenvector associated with the same eigenvalue. Thus, Eq. (9.9) makes it *appear* that there are an infinite number of linearly independent eigenvectors associated with the complex eigenvalues. When a vector $\underset{\sim}{\boldsymbol{v}}$ is real, then $\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{v}}$ is generally linearly independent of $\underset{\sim}{\boldsymbol{v}}$. However, the eigenvector $\underset{\sim}{\boldsymbol{p}}$ is complex, so this property does not hold. We need to demonstrate that $\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{p}}$ and $\underset{\sim}{\boldsymbol{p}}$ are linearly dependent for all rotations $\underset{\approx}{\boldsymbol{Q}}$ about the rotation axis $\underset{\sim}{\boldsymbol{a}}$. Denoting the rotation angle for $\underset{\approx}{\boldsymbol{Q}}$ by by q, it's straightforward to demonstrate that $\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{p}} = \Lambda \underset{\sim}{\boldsymbol{p}}$, where $\Lambda = \{e^{iq}, e^{-iq}, \text{or } 1\}$. In other words, $\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{p}}$ and $\underset{\sim}{\boldsymbol{p}}$ differ only by a (possibly complex) scalar multiple, which proves that they are indeed linearly dependent.

10. Polar decomposition

A special tensor called the "deformation gradient $\underset{\approx}{\boldsymbol{F}}$" plays a pivotal role in the field of continuum mechanics [6, 7]. This tensor carries all information about the *deformation* of a material element from its known reference state to its current "spatial" state. The polar decomposition theorem uniquely quantifies how any general deformation can be viewed as a combination of material re-orientation (*i.e.* rotation), material distortion (*i.e.* a change in shape), and material dilation (*i.e.* a change in size).

Difficult definition of the deformation gradient

In this section, we provide the mathematically rigorous definition of the term "deformation gradient" normally found in textbooks. Later, we provide a more intuitive (less rigorous) definition.

Suppose that the location $\underset{\sim}{\boldsymbol{X}}$ of each material particle in a body is known at some reference state (usually time zero). This reference position is regarded as the "name" of the particle. For example, when we say "point $\underset{\sim}{\boldsymbol{X}}$," we mean "the particle whose reference position is $\underset{\sim}{\boldsymbol{X}}$."

The complete deformation of the body may be described by a mapping function χ such that the deformed location $\underset{\sim}{\boldsymbol{x}}$ of the point $\underset{\sim}{\boldsymbol{X}}$ is given by

$$\underset{\sim}{\boldsymbol{x}} = \chi(\underset{\sim}{\boldsymbol{X}}) \tag{10.1}$$

The "motion" of a body is a sequence of deformed states over time. Hence, the motion of a body is described by allowing the above mapping function to additionally vary with time:

$$\underset{\sim}{\boldsymbol{x}} = \chi(\underset{\sim}{\boldsymbol{X}}, t) \tag{10.2}$$

The deformation gradient tensor $\underset{\approx}{\boldsymbol{F}}$ is mathematically defined to be the gradient of the mapping function:

$$\underset{\approx}{\boldsymbol{F}} = \frac{\partial \underset{\sim}{\boldsymbol{x}}}{\partial \underset{\sim}{\boldsymbol{X}}} \quad \Rightarrow \quad F_{ij} = \frac{\partial x_i}{\partial X_j} \Rightarrow \quad [F] = \begin{bmatrix} \frac{\partial x_1}{\partial X_1} & \frac{\partial x_1}{\partial X_2} & \frac{\partial x_1}{\partial X_3} \\ \frac{\partial x_2}{\partial X_1} & \frac{\partial x_2}{\partial X_2} & \frac{\partial x_2}{\partial X_3} \\ \frac{\partial x_3}{\partial X_1} & \frac{\partial x_3}{\partial X_2} & \frac{\partial x_3}{\partial X_3} \end{bmatrix} \tag{10.3}$$

While this definition is rigorously precise, it offers virtually no intuitive insight into the real meaning of information contained in a matrix of F_{ij} components. If someone showed you a deformation gradient matrix, would you be able to interpret it physically? Probably not if you only understand it by the above definition.

One way to begin to gain insight into this definition is to consider some simple specialized mapping functions. Suppose that $\underset{\sim}{\boldsymbol{X}}^{(p)}$ denotes the initial location of a particle of interest. We want to gain better understanding of where that particle translates and how material in the vicinity of that particle deforms. Then it makes sense to consider a Taylor series expansion of the mapping function, centered about the particle $\underset{\sim}{\boldsymbol{X}}^{(p)}$. Namely, to second-order accuracy, the mapping function near the particle can be approximated by

$$\begin{aligned}\underset{\sim}{\boldsymbol{x}} = \chi(\underset{\sim}{\boldsymbol{X}}, t) \approx \quad & \chi(\underset{\sim}{\boldsymbol{X}}^{(p)}, t) \\ & + \left[\frac{\partial \chi(\underset{\sim}{\boldsymbol{X}}, t)}{\partial \underset{\sim}{\boldsymbol{X}}}\right]_{\text{evaluated at } \underset{\sim}{\boldsymbol{X}} = \underset{\sim}{\boldsymbol{X}}^{(p)}} \bullet (\underset{\sim}{\boldsymbol{X}} - \underset{\sim}{\boldsymbol{X}}^{(p)}) \\ & + \frac{1}{2}\left[\frac{\partial^2 \chi(\underset{\sim}{\boldsymbol{X}}, t)}{\partial \underset{\sim}{\boldsymbol{X}} \partial \underset{\sim}{\boldsymbol{X}}}\right]_{\text{evaluated at } \underset{\sim}{\boldsymbol{X}} = \underset{\sim}{\boldsymbol{X}}^{(p)}} :(\underset{\sim}{\boldsymbol{X}} - \underset{\sim}{\boldsymbol{X}}^{(p)})(\underset{\sim}{\boldsymbol{X}} - \underset{\sim}{\boldsymbol{X}}^{(p)}) \\ & + \ldots \end{aligned} \tag{10.4}$$

The first term in this expansion, $\chi(\underset{\sim}{\boldsymbol{X}}^{(p)}, t)$, is the deformed location of the particle $\underset{\sim}{\boldsymbol{X}}^{(p)}$, so it makes sense to denote this quantity by simply $\underset{\sim}{\boldsymbol{x}}^{(p)}$:

$$\underset{\sim}{\boldsymbol{x}}^{(p)} \equiv \chi(\underset{\sim}{\boldsymbol{X}}^{(p)}, t) \tag{10.5}$$

Recalling Eq. (10.3), the derivative in the second term of the above expansion is the value of the deformation gradient tensor evaluated at the particle of interest. Hence, it makes sense for us to denote this derivative by a more compact symbol,

$$\underset{\approx}{\boldsymbol{F}}^{(p)} \equiv \left[\frac{\partial \chi(\underset{\sim}{\boldsymbol{X}}, t)}{\partial \underset{\sim}{\boldsymbol{X}}}\right]_{\text{evaluated at } \underset{\sim}{\boldsymbol{X}} = \underset{\sim}{\boldsymbol{X}}^{(p)}} \tag{10.6}$$

We included the third term in the expansion of Eq. (10.4) merely to emphasize that we could continue the expansion to as high an order of accuracy as we desire so long as the mapping function is adequately smooth. However, in the analysis of general non-linear mappings, you are unlikely to ever encounter expansions that go beyond first order. Using our more compact notation defined in Eqs. (10.5) and (10.6), the first-order mapping expansion in the neighborhood of a particle $\underset{\sim}{\boldsymbol{X}}^{(p)}$ can be written

$$\underset{\sim}{\boldsymbol{x}} \approx \underset{\sim}{\boldsymbol{x}}^{(p)} + \underset{\approx}{\boldsymbol{F}}^{(p)} \bullet (\underset{\sim}{\boldsymbol{X}} - \underset{\sim}{\boldsymbol{X}}^{(p)}) + \ldots \tag{10.7}$$

Expressions like this are similar to writing a first order expansion of a highly non-linear function centered about a particular point of interest. What you get is an equation of a straight line that passes through the function at that point and which is tangent to the curve at that point. Anyone good mathematician will tell you that the key to solving or interpreting a nonlinear equation is to

first understand *linear* functions -- that's because all functions can be regarded as an ensemble of linear line segments (the more you use, the better the approximation, and in the limit of an infinite number of line segments, the technique becomes known by the name *calculus*!)

Since understanding linear functions (straight lines) is an essential first step to understanding non-linear functions, one might suspect that a good place to begin understanding general non-linear mappings is to first study linear mappings of the form

$$\underset{\sim}{\boldsymbol{x}} \approx \underset{\sim}{\boldsymbol{x}}^{(p)} + \underset{\approx}{\boldsymbol{F}}^{(p)} \bullet (\underset{\sim}{\boldsymbol{X}} - \underset{\sim}{\boldsymbol{X}}^{(p)}) \tag{10.8}$$

If the tensor $\underset{\approx}{\boldsymbol{F}}^{(p)}$ equals the identity tensor, then this mapping describes a pure rigid translation. If the tensor $\underset{\approx}{\boldsymbol{F}}^{(p)}$ is a rotation tensor, then this mapping describes translation in combination with rotation. If the tensor $\underset{\approx}{\boldsymbol{F}}^{(p)}$ is not orthogonal, then this mapping is capable of describing simultaneous translation, rotation, and shape change.

We should make amends for our sloppy terminology. Above, we said that a straight line in the plane can be described through a *linear* function. Most people would say that a linear function is expressible in the form $y = mx + b$. A careful mathematician will tell you that this function is not linear -- it is "*affine*." A linear function $y(x)$ must satisfy the property that $y(\alpha_1 x_1 + \alpha_2 x_2) = \alpha_1 y(x_1) + \alpha_2 y(x_2)$ for all scalars $\{\alpha_1, x_1, \alpha_2, x_2\}$. If this condition must hold for all choices of the scalars α_1 and α_2, then it must hold for the particular choice $\alpha_1 = \alpha_2 = 0$. Thus, if a function is linear then it must have the property that $y(0) = 0$. The function $y = mx + b$ does not satisfy this property if b is nonzero, so this function must not be linear! A truly linear function must be expressible as $y = mx$ for some scalar x. An affine function is simply a linear function plus some constant.

The function in Eq. (10.8) is not *linear* -- it is *affine*. However, we can always rewrite this equation in the form $\underset{\sim}{\boldsymbol{x}} - \underset{\sim}{\boldsymbol{x}}^{(p)} = \underset{\approx}{\boldsymbol{F}}^{(p)} \bullet (\underset{\sim}{\boldsymbol{X}} - \underset{\sim}{\boldsymbol{X}}^{(p)})$. Thus, the *relative* deformed position vector $\underset{\sim}{\boldsymbol{x}} - \underset{\sim}{\boldsymbol{x}}^{(p)}$ can always be expressed as a linear function of the *relative* initial position vector $\underset{\sim}{\boldsymbol{X}} - \underset{\sim}{\boldsymbol{X}}^{(p)}$. For this reason, it is sufficient for us to study the very specialized case of *linear* mapping functions:

$$\underset{\sim}{\boldsymbol{x}} = \underset{\approx}{\boldsymbol{F}} \bullet \underset{\sim}{\boldsymbol{X}} \tag{10.9}$$

We have already explained why a general mapping can be locally approximated by a linear mapping so long as the position vectors are defined *relative* to the *local* point of interest. The hallmark of a *nonlinear* mapping is that the local deformation gradient tensor $\underset{\approx}{\boldsymbol{F}}$ varies with position. This is similar to the fact that the local slope of a tangent line varies with position along the nonlinear curve. If, as a very specialized case, the deformation gradient tensor $\underset{\approx}{\boldsymbol{F}}$ does *not* vary with position, then the mapping is said to be *homogenous*. As

will be explained below, a homogenous mapping has the property that initially straight lines deform to straight lines, and flat planes deform to flat planes. Consequently, cubes will always deform to parallelepipeds. Since *inhomogenous* deformations can be regarded as *locally* linear, we conclude that *infinitesimal* cubes in general mappings will deform to *infinitesimal* parallelepipeds, but the size, orientation, and distortion level of these deformed parallelepipeds will vary in space. These ideas are illustrated in Fig. 10.1.

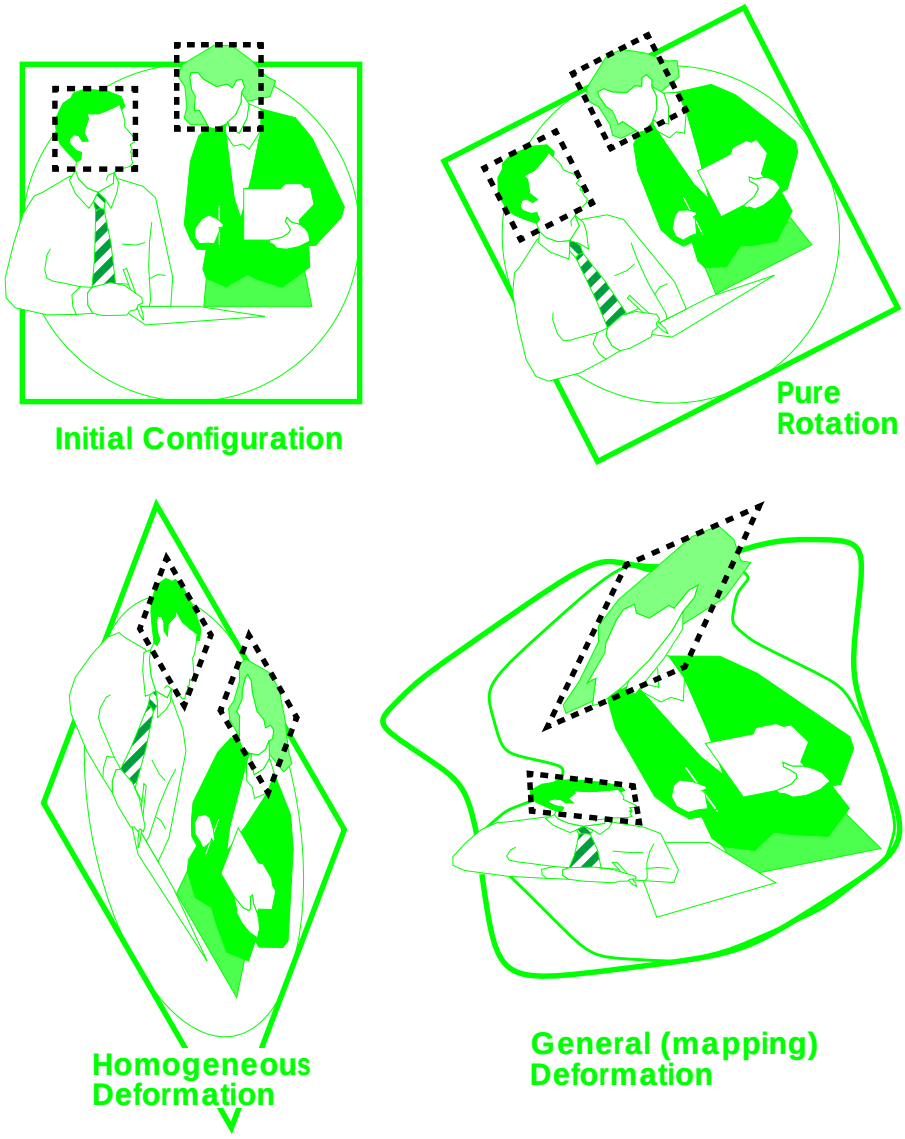


Figure 10.1. Increasingly complex deformations. The simplest deformation, rotation, does not involve any length changes -- cubes remain cubes of exactly the same size but of different orientation. The next easiest deformation is "homogenous" deformation where the changes in shape and orientation are the same everywhere throughout the body -- all reference cubes deform to parallelepipeds of the same shape and orientation. For the most general (inhomogeneous) deformation, infinitesimal cubes still deform to parallelepipeds, but the orientation and distortion of these parallelepipeds varies in space.

Intuitive definition of the deformation gradient

To better understand the deformation gradient, imagine a small cube-shaped element of material whose sides are aligned with the reference basis $\{\underset{\sim}{\boldsymbol{E}}_1, \underset{\sim}{\boldsymbol{E}}_2, \underset{\sim}{\boldsymbol{E}}_3\}$. Upon deformation, the cube deforms to a parallelepiped. If the deformation is 2-dimensional, then a square deforms to a parallelogram, as

sketched in Fig.10.2. The vectors on the sides of the element deform to new vectors $\{\underset{\sim}{g}_1, \underset{\sim}{g}_2, \underset{\sim}{g}_3\}$. These so-called **convected base vectors** are not generally orthogonal nor of unit length.

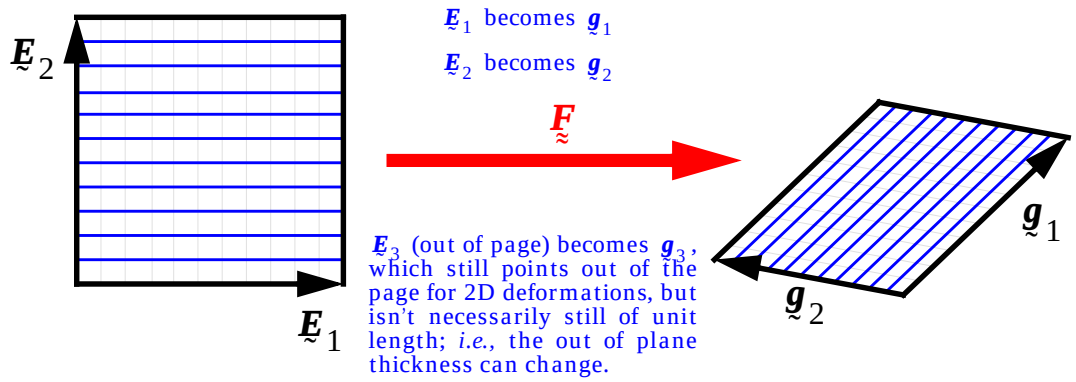


Figure 10.2. The deformation gradient tensor. A cube deforms to a parallelepiped. This deformation is shown in the plane for simplicity. For planar deformations, squares deform to parallelograms; of course, the out-of-plane thickness is allowed to change as well. Note that this deformation involves a significant amount of counter-clockwise rotation.

With a bit of thought, one recognizes that an intuitive (non-rigorous) definition of the deformation gradient tensor $\underset{\approx}{F}$ states that columns of the lab component matrix $[F]$ simply contain the lab components of the convected $\{\underset{\sim}{g}_1, \underset{\sim}{g}_2, \underset{\sim}{g}_3\}$ vectors. We write this statement symbolically as

$$[F] = [\{\underset{\sim}{g}_1\}\{\underset{\sim}{g}_2\}\{\underset{\sim}{g}_3\}]. \tag{10.10}$$

In direct notation, this may be expressed by a sum of dyads:*

$$\underset{\approx}{F} = \underset{\sim}{g}_1\underset{\sim}{E}_1 + \underset{\sim}{g}_2\underset{\sim}{E}_2 + \underset{\sim}{g}_3\underset{\sim}{E}_3. \tag{10.11}$$

Consider, for example, that the deformation shown in Fig. 10.2, which occurs in the plane so that $\underset{\sim}{E}_3$ points out of the page. Knowing that $\underset{\sim}{E}_1$ and $\underset{\sim}{E}_2$ are of unit length, we can use a ruler to measure the vectors in Fig. 10.2 to obtain

* Here, two vectors written side by side $\underset{\sim}{a}\underset{\sim}{b}$ are multiplied dyadically so that $[\underset{\sim}{a}\underset{\sim}{b}]_{ij} = a_i b_j$. Therefore, the dyad $\underset{\sim}{g}_1\underset{\sim}{E}_1$ is represented by a 3×3 matrix whose components are all zero except that the first column has the same components as the lab components of the $\underset{\sim}{g}_1$ vector.

$$\underset{\sim}{g}_1 \approx (0.62)\underset{\sim}{E}_1 + (0.59)\underset{\sim}{E}_2 \qquad \rightarrow \qquad \{\underset{\sim}{g}_1\} = \begin{Bmatrix} 0.62 \\ 0.59 \\ 0 \end{Bmatrix}$$

$$\underset{\sim}{g}_2 \approx (-0.71)\underset{\sim}{E}_1 + (0.13)\underset{\sim}{E}_2 \qquad \rightarrow \qquad \{\underset{\sim}{g}_2\} = \begin{Bmatrix} -0.71 \\ 0.13 \\ 0 \end{Bmatrix} \tag{10.12}$$

In addition to the deformation shown in Fig. 10.2, let's further suppose that the out-of-plane thickness of the material increases by a factor of 3 so that

$$\underset{\sim}{g}_3 = (3.0)\underset{\sim}{E}_3 \qquad \rightarrow \qquad \{\underset{\sim}{g}_3\} = \begin{Bmatrix} 0 \\ 0 \\ 3.0 \end{Bmatrix} \tag{10.13}$$

The deformation gradient matrix* corresponding to Fig. 10.2 is constructed by assembling these three vectors into the columns of $[F]$:

$$[F] = \begin{bmatrix} 0.62 & -0.71 & 0 \\ 0.59 & 0.13 & 0 \\ 0 & 0 & 3.0 \end{bmatrix} \tag{10.14}$$

Converse problem: interpreting a deformation gradient matrix. The intuitive definition of the deformation gradient may be applied in reverse. Suppose, for example, that the deformation gradient matrix is known to be

$$[F] = \begin{bmatrix} \mathbf{1.0} & \mathbf{1.0} & 0 \\ 0 & \mathbf{2.0} & 0 \\ 0 & 0 & \mathbf{0.8} \end{bmatrix} \tag{10.15}$$

From this, we would identify the three deformed base vectors by the columns of this matrix. Namely,

$$\{\underset{\sim}{g}_1\} = \begin{Bmatrix} 1.0 \\ 0 \\ 0 \end{Bmatrix}, \qquad \{\underset{\sim}{g}_2\} = \begin{Bmatrix} 1.0 \\ 2.0 \\ 0 \end{Bmatrix}, \qquad \{\underset{\sim}{g}_3\} = \begin{Bmatrix} 0 \\ 0 \\ 0.8 \end{Bmatrix} \tag{10.16}$$

In basis notation, these would be written

$$\underset{\sim}{g}_1 = \underset{\sim}{E}_1, \qquad \underset{\sim}{g}_2 = \underset{\sim}{E}_1 + 2\underset{\sim}{E}_2, \qquad \underset{\sim}{g}_3 = (0.8)\underset{\sim}{E}_3 \tag{10.17}$$

These vectors define sides of a deformed parallelepiped as drawn in Fig. 10.3. Not shown in that figure is the fact that $\underset{\sim}{g}_3 = (0.8)\underset{\sim}{E}_3$, meaning that this deformation has reduced the out-of-plane thickness by 20%.

* with respect to the lab basis $\{\underset{\sim}{E}_1, \underset{\sim}{E}_2, \underset{\sim}{E}_3\}$.

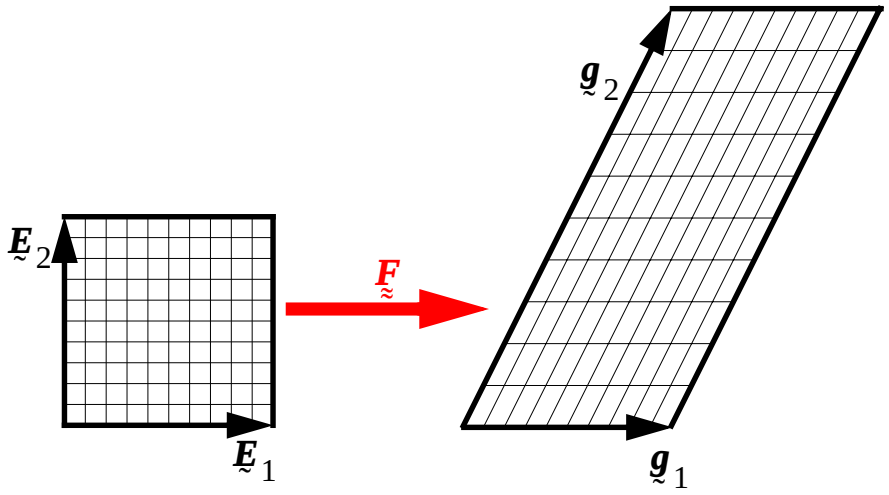


Figure 10.3. Physically interpreting a deformation gradient matrix.

Importantly, the intuitive definition of the deformation gradient applies only to *homogeneous* deformations in which all material elements within the body deform in exactly the same way. This means that the value of $\underset{\approx}{\boldsymbol{F}}$ is the same for every point in the body. For general inhomogeneous deformations, each material point can have its own different deformation gradient tensor $\underset{\approx}{\boldsymbol{F}}$. In this case, the intuitive definition of the deformation gradient must be applied to each *infinitesimal* material element.

The Jacobian of the deformation

A material element that deforms according to a deformation gradient $\underset{\approx}{\boldsymbol{F}}$ generally changes from a reference volume dV_o to a current volume dV. The Jacobian of the deformation is denoted J and is defined to equal the ratio

$$J = \frac{dV}{dV_o} \tag{10.18}$$

It can be shown that

$$J = \det \underset{\approx}{\boldsymbol{F}} \tag{10.19}$$

For example, for the deformation shown in Fig. 10.2, the Jacobian is obtained by taking the determinant of the matrix in Eq. (10.14):

$$J = \det \begin{bmatrix} 0.62 & -0.71 & 0 \\ 0.59 & 0.13 & 0 \\ 0 & 0 & 3.0 \end{bmatrix} = 1.43 \tag{10.20}$$

which means that the material element has increased in volume by 43%. Although the element appears to be smaller in size in Fig. 10.2, the overall volume has increased because of the large out-of-the-page expansion.

Invertibility of a deformation

In order for the deformation to make physical sense, the mapping function $\chi(\underset{\sim}{\boldsymbol{X}}, t)$ must be "one-to-one," meaning that any two distinct points must deform to two distinct points — they must not map to the same location (otherwise, the material will have interpenetrated itself, which is physically undesirable). Likewise, a single material point must not map to two different points — such behavior would imply the generation of new free surfaces.* A deformation is called "globally invertible" if its mapping function is one-to-one.

A deformation is locally invertible if and only if the deformation gradient $\underset{\approx}{\boldsymbol{F}}$ is invertible. In other words, the Jacobian must be nonzero: $J \neq 0$. Global invertibility implies local invertibility, but not vice versa (see Fig. 10.4). This is why many numerical methods require special treatment to prevent material interpenetration.

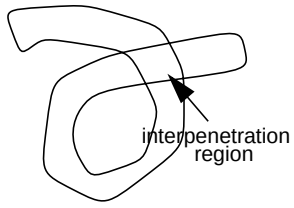


Figure 10.4. Example of a deformation that is locally invertible, but not globally invertible.

For physically real deformations, both the initial and the deformed volumes in Eq. (10.18) must be positive. Hence, the condition of local invertibility may be replaced by the more restrictive condition,

$$J > 0 \tag{10.21}$$

Sequential deformations

If a material element is deformed by a deformation gradient, $\underset{\approx}{\boldsymbol{F}}_1$, followed by a second deformation gradient $\underset{\approx}{\boldsymbol{F}}_2$, then the total deformation gradient is given by

$$\underset{\approx}{\boldsymbol{F}} = \underset{\approx}{\boldsymbol{F}}_2 \bullet \underset{\approx}{\boldsymbol{F}}_1 \tag{10.22}$$

Note that the tensors appear in the composition in reverse order of application. Any differential material vector $d\underset{\sim}{\boldsymbol{X}}$ in the reference configuration deforms to a new vector $d\underset{\sim}{\boldsymbol{x}}$ in the spatial configuration by the operation

$$d\underset{\sim}{\boldsymbol{x}} = \underset{\approx}{\boldsymbol{F}} \bullet d\underset{\sim}{\boldsymbol{X}}\,. \tag{10.23}$$

If $\underset{\approx}{\boldsymbol{F}}$ is given by Eq. (10.22), then $\underset{\approx}{\boldsymbol{F}}_1$ first acts on $d\underset{\sim}{\boldsymbol{X}}$ and *then* $\underset{\approx}{\boldsymbol{F}}_2$ acts on $\underset{\approx}{\boldsymbol{F}}_1 \bullet d\underset{\sim}{\boldsymbol{X}}$. This is why the sequential deformation gradients appear in reverse order of application.

---

* which would be desirable if one wishes to model, say, void nucleation or the formation of cracks. In that case, the one-to-one condition would be relaxed.

Matrix analysis version of the polar decomposition theorem

An important theorem from matrix analysis states that any square matrix $[F]$ can be decomposed into the following form

$$[F] = [S][D][T]^T, \tag{10.24}$$

where $[S]$ and $[T]$ are proper orthogonal and $[D]$ is diagonal. The above equation may be written as

$$[F] = [R][U] = [V][R], \tag{10.25}$$

where

$$[R] = [S][T]^T \tag{10.26}$$

$$[U] = [T][D][T]^T \tag{10.27}$$

$$[V] = [S][D][S]^T \tag{10.28}$$

Note that $[R]$ is orthogonal, while $[U]$ and $[V]$ are symmetric tensors whose eigenvalues are given by the diagonal components of $[D]$.

In indicial form,

$$F_{ij} = R_{ik}U_{kj} = V_{ik}R_{kj}, \tag{10.29}$$

where

$$R_{ij} = S_{ik}T_{jk} \tag{10.30}$$

$$U_{ij} = T_{im}T_{jn}D_{mn} \tag{10.31}$$

$$V_{ij} = S_{im}S_{jn}D_{mn} \tag{10.32}$$

Multiplying these equations by the laboratory basis dyads gives

$$\underset{\approx}{\boldsymbol{F}} = \underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\boldsymbol{U}} = \underset{\approx}{\boldsymbol{V}} \bullet \underset{\approx}{\boldsymbol{R}}, \tag{10.33}$$

where $\underset{\approx}{\boldsymbol{U}}$ and $\underset{\approx}{\boldsymbol{V}}$ are symmetric tensors having the same eigenvalues. Furthermore, the eigenvectors of $\underset{\approx}{\boldsymbol{U}}$ and $\underset{\approx}{\boldsymbol{V}}$ are respectively

$$\underset{\sim}{\delta}^U_m = T_{im}\underset{\sim}{\boldsymbol{E}}_i \quad \text{and} \quad \underset{\sim}{\delta}^V_m = S_{im}\underset{\sim}{\boldsymbol{E}}_i$$

Then the dyad representation of the rotation tensor is

$$\underset{\approx}{\boldsymbol{R}} = \underset{\sim}{\delta}^V_k \underset{\sim}{\delta}^U_k \tag{10.34}$$

These conclusions will now be presented directly from the tensor analysis perspective.

The polar decomposition theorem — a hindsight intuitive introduction

The essential idea behind the polar decomposition theorem is illustrated in Fig. 10.5 where we have shown that the deformation $\underset{\approx}{\boldsymbol{F}}$ from Fig. 10.2 can be decomposed into a sequence of two separate steps:

- A stretch $\underset{\approx}{\boldsymbol{U}}$ followed by a rotation $\underset{\approx}{\boldsymbol{R}}$, so that $\underset{\approx}{\boldsymbol{F}} = \underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\boldsymbol{U}}$. This is the top path in Fig. 10.5. The tensor $\underset{\approx}{\boldsymbol{U}}$ is called the right stretch because it appears on the right in the expression $\underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\boldsymbol{U}}$.

Alternatively, the deformation may be decomposed in similar steps applied in the opposite order:

- A rotation $\underset{\approx}{\boldsymbol{R}}$ followed by a stretch $\underset{\approx}{\boldsymbol{V}}$, so that $\underset{\approx}{\boldsymbol{F}} = \underset{\approx}{\boldsymbol{V}} \bullet \underset{\approx}{\boldsymbol{R}}$. This is the bottom path in Fig. 10.5. The tensor $\underset{\approx}{\boldsymbol{V}}$ is called the left stretch.

In Fig. 10.2, a circumscribed square and an inscribed shaded circle* have been "painted" onto the deforming material to help illustrate the meaning of the term "**stretch**." A stretch is a special type of deformation for which the deformation gradient is both symmetric *and positive definite*. So a stretch is any deformation for which $\underset{\approx}{\boldsymbol{R}}=\underset{\approx}{\boldsymbol{I}}$. For any pure stretch, there always exists three directions (the eigenvectors of the stretch tensor) along which in which the material is stretched or compressed, but not rotated. Material fibers that are originally aligned with the principal stretch directions may change length, but they do not change orientation under a pure stretch. The polar decomposition theorem says that any general deformation can be decomposed into a pure stretch in combination with a rigid rotation.

The directions marked $\underset{\sim}{\delta}_k^U$ are the principal stretch directions for the right stretch $\underset{\approx}{\boldsymbol{U}}$. They change length but don't change orientation during the application of $\underset{\approx}{\boldsymbol{U}}$. The directions marked $\underset{\sim}{\delta}_k^V$ are the principal stretch directions for the left stretch $\underset{\approx}{\boldsymbol{V}}$. They change length but don't change orientation during the application of $\underset{\approx}{\boldsymbol{V}}$.

Incidentally, note that

$$\underset{\sim}{\delta}_k^V = \underset{\approx}{\boldsymbol{R}} \bullet \underset{\sim}{\delta}_k^U \tag{10.35}$$

* which would be a sphere for 3D deformations

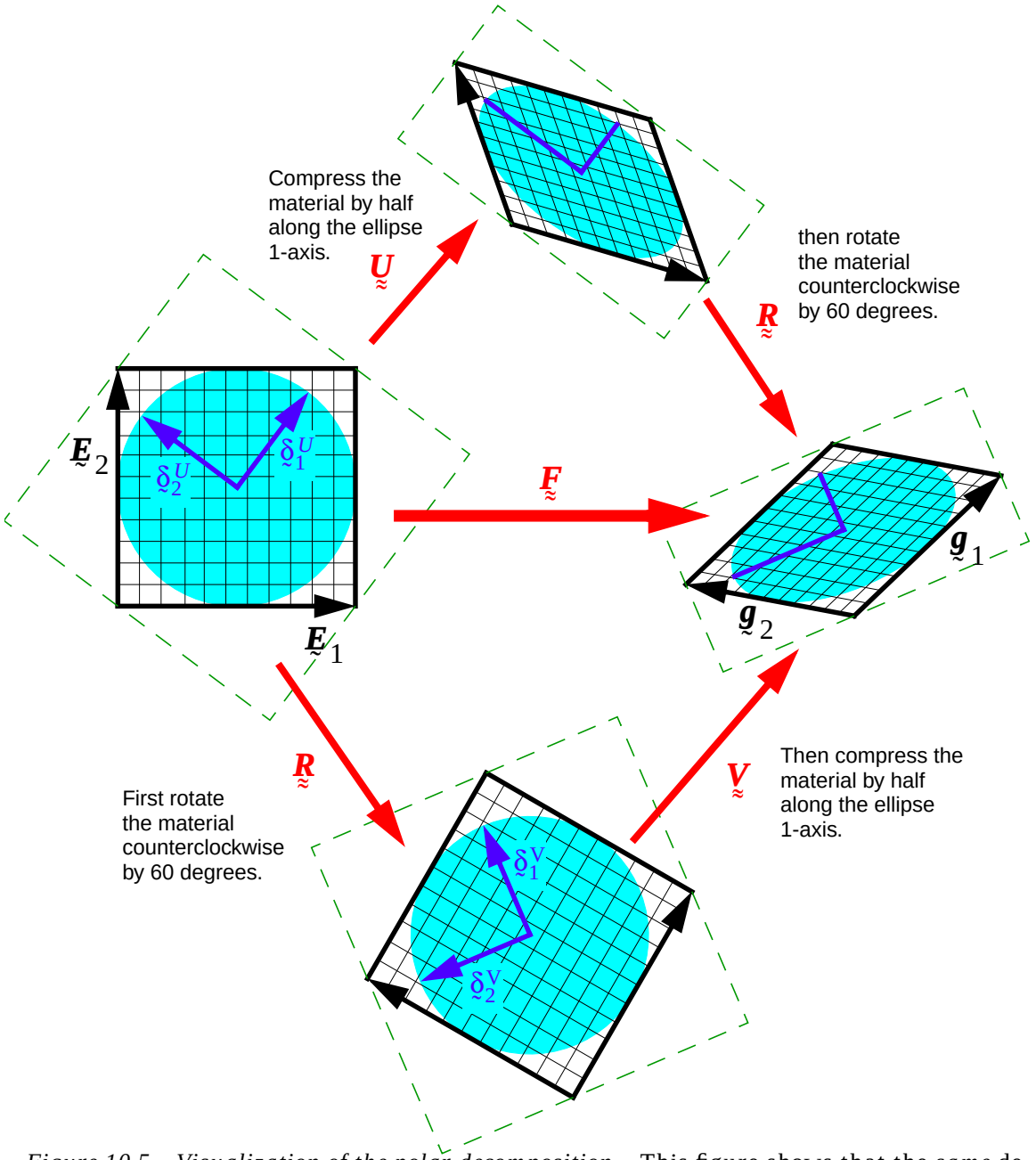


Figure 10.5. Visualization of the polar decomposition. This figure shows that the *same* deformation $\underset{\sim}{\boldsymbol{F}}$ from Fig. 10.2 can be visualized as two different paths, each of which involves an intermediate configuration. The upper $\underset{\sim}{\boldsymbol{R}} \bullet \underset{\sim}{\boldsymbol{U}}$ path first compresses the material by a factor of $1/2$ in the direction of the vector labeled $\underset{\sim}{\delta}_1^U$ and then rotates counterclockwise by $60°$. The same deformation is achieved on the bottom path by rotating by $60°$ and *then* compressing by a factor of $1/2$ in the direction of the vector labeled $\underset{\sim}{\delta}_1^V$. Note that $\underset{\sim}{\delta}_1^V$ is obtained by rotating $\underset{\sim}{\delta}_1^U$ by $60°$. In these figures, we have "painted" a circle (or sphere in 3D) on the reference cube to show how it deforms into an ellipse (or ellipsoid in 3D). The vectors $\underset{\sim}{\delta}_k^U$ and $\underset{\sim}{\delta}_k^V$ lie on the major axes of the ellipse.

This section has been called a "hindsight" introduction because the principal directions of the stretch are not normally known *a priori*. Finding these directions is a key task for the polar decomposition theorem.

In Fig. 10.5, the orthonormal triad of vectors $\underset{\sim}{\delta}_1^U$, $\underset{\sim}{\delta}_2^U$, and $\underset{\sim}{\delta}_3^U$ (which points out of the page for planar deformations) are the principal directions of the right stretch $\underset{\approx}{\boldsymbol{U}}$. These principal stretch vectors do not change orientation during the application of $\underset{\approx}{\boldsymbol{U}}$. They change direction only upon application of the rotation. Note that other material fibers which are *not* aligned with principal directions of $\underset{\approx}{\boldsymbol{U}}$ generally do change orientation even before the rotation is applied. Thus, the rotation tensor describes *overall* material reorientation, but not the reorientation of individual material fibers.

A visual inspection of Fig. 10.5 shows that material fibers originally in the direction of $\underset{\sim}{\delta}_1^U$ appear to change length by a factor of about $1/2$ for that particular example. The fibers in the direction of $\underset{\sim}{\delta}_2^U$ do not appear to change length. Stated differently, the ratio of their deformed length to their undeformed length is 1. If we suppose that the out-of-plane thickness increases by a factor of 3, then fibers in the direction of $\underset{\sim}{\delta}_3^U$ will change length by a factor of 3. The matrix for $\underset{\approx}{\boldsymbol{U}}$ is diagonal in the principal stretch basis. The components are the ratios of deformed length to undeformed length. Thus,

$$[\underset{\approx}{\boldsymbol{U}}] = \begin{bmatrix} \frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 3 \end{bmatrix} \quad \textit{with respect to the } \underset{\sim}{\delta}_k^U \textit{ triad} \tag{10.36}$$

In order to express this stretch tensor in terms of the *reference* basis we must perform a coordinate transformation. To do this, first we use a ruler to measure the components of each of the $\underset{\sim}{\delta}_k^U$ with respect to the $\underset{\sim}{\boldsymbol{E}}_k$ reference basis to obtain:

$$\{\underset{\sim}{\delta}_1^U\} = \begin{Bmatrix} 0.6 \\ 0.8 \\ 0 \end{Bmatrix}, \qquad \{\underset{\sim}{\delta}_2^U\} = \begin{Bmatrix} -0.8 \\ 0.6 \\ 0 \end{Bmatrix} \qquad \{\underset{\sim}{\delta}_3^U\} = \begin{Bmatrix} 0 \\ 0 \\ 1 \end{Bmatrix} \tag{10.37}$$

These arrays are assembled into columns to form the direction cosine matrix $[Q]$ from Eq. (1.26):

$$[Q_{ij}] = [\underset{\sim}{\boldsymbol{E}}_i \bullet \underset{\sim}{\delta}_j^U] = [\{\underset{\sim}{\delta}_1^U\}\{\underset{\sim}{\delta}_2^U\}\{\underset{\sim}{\delta}_3^U\}] = \begin{bmatrix} 0.6 & -0.8 & 0 \\ 0.8 & 0.6 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{10.38}$$

Applying Eq. (1.33) gives us the components of $\underset{\approx}{\boldsymbol{U}}$ with respect to the *reference* basis:

$$[\underset{\approx}{\boldsymbol{U}}] = \begin{bmatrix} 0.6 & -0.8 & 0 \\ 0.8 & 0.6 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 3 \end{bmatrix} \begin{bmatrix} 0.6 & -0.8 & 0 \\ 0.8 & 0.6 & 0 \\ 0 & 0 & 1 \end{bmatrix}^T$$

$$= \begin{bmatrix} 0.82 & -0.24 & 0 \\ -0.24 & 0.68 & 0 \\ 0 & 0 & 3.0 \end{bmatrix} \textit{ with respect to the } \underset{\sim}{\boldsymbol{E}}_k \textit{ reference triad} \qquad (10.39)$$

This tensor is symmetric. It is also positive definite. It is therefore a stretch tensor.

To construct the rotation tensor, we would use a protractor to measure the angle through which the ellipse in Fig. (10.5) has rotated. The result appears to be about $60°$ and, because the deformation is planar, the axis of rotation is in the $\underset{\sim}{\boldsymbol{E}}_3$ direction. Hence, the rotation tensor $\underset{\approx}{\boldsymbol{R}}$ must be

$$\underset{\approx}{\boldsymbol{R}} = \begin{bmatrix} \cos 60° & -\sin 60° & 0 \\ \sin 60° & \cos 60° & 0 \\ 0 & 0 & 1 \end{bmatrix} \textit{ with respect to the } \underset{\sim}{\boldsymbol{E}}_k \textit{ reference triad} \qquad (10.40)$$

Multiplying the above two tensors gives

$$[\underset{\approx}{\boldsymbol{F}}] = [\underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\boldsymbol{U}}] = \begin{bmatrix} \cos 60° & -\sin 60° & 0 \\ \sin 60° & \cos 60° & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0.82 & -0.24 & 0 \\ -0.24 & 0.68 & 0 \\ 0 & 0 & 3.0 \end{bmatrix}$$

$$= \begin{bmatrix} 0.618 & -7.09 & 0. \\ 0.590 & 0.132 & 0. \\ 0. & 0. & 3. \end{bmatrix} \textit{ with respect to the } \underset{\sim}{\boldsymbol{E}}_k \textit{ reference triad,} \qquad (10.41)$$

which is the same as the matrix derived directly in Eq. (10.14). Thus, we have demonstrated that the decomposition into a stretch followed by a rotation does indeed exist for the sample deformation in Fig. 10.5.

This section was labeled “hindsight” because we produced the principal stretch directions by serendipity (*i.e.*, “out of thin air”) and then we demonstrated that they did indeed permit the decomposition into stretch followed by rotation.

The next section presents the polar decomposition in a more classical manner.

A more rigorous (classical) presentation of the polar

decomposition theorem

THEOREM: For each invertible tensor $\underset{\approx}{\boldsymbol{F}}$ there exists a unique orthogonal tensor $\underset{\approx}{\boldsymbol{R}}$, a unique symmetric positive definite "right stretch" $\underset{\approx}{\boldsymbol{U}}$, and a unique symmetric positive definite "left stretch" $\underset{\approx}{\boldsymbol{V}}$ such that

$$\underset{\approx}{\boldsymbol{F}} = \underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\boldsymbol{U}} = \underset{\approx}{\boldsymbol{V}} \bullet \underset{\approx}{\boldsymbol{R}} \tag{10.42}$$

WARNING: the descriptors "symmetric" and "positive definite" are *requirements*, not consequences. Keep in mind that a stretch must be both symmetric and positive definite — a tensor can be symmetric without being positive definite. It is depressingly common for researchers to erroneously declare a deformation gradient to be a stretch simply because it is symmetric. However, if only symmetry were tested, the rotation matrix of Eq. (3.24) would be wrongly identified as a stretch. Checking symmetry is not enough to declare a symmetric matrix to be a stretch — one must also prove positive definiteness! It can be shown that the deformation gradient $\underset{\approx}{\boldsymbol{F}}$ is symmetric if and only if the rotation angle is an integral multiple of $180°$.

COMMENT: The determinant of $\underset{\approx}{\boldsymbol{R}}$ will have the same sign as the determinant of $\underset{\approx}{\boldsymbol{F}}$. Hence, the orthogonal tensor $\underset{\approx}{\boldsymbol{R}}$ will be a proper rotation if and only if $\det \underset{\approx}{\boldsymbol{F}} > 0$. We are interested in second-order tensors in 3D space so that $[\underset{\approx}{\boldsymbol{F}}]$ is a 3×3 matrix. If the determinant of $\underset{\approx}{\boldsymbol{F}}$ is negative, then we can always perform the polar decomposition of $-\underset{\approx}{\boldsymbol{F}}$, which has a positive determinant in 3D space. Thus, without loss in generality, we may assume that the orthogonal tensor $\underset{\approx}{\boldsymbol{R}}$ in Eq. (10.42) is a *proper* rotation.

PROOF: It is important to understand the flow of the following proof. The polar decomposition theorem has the following structure:

GIVEN: $\underset{\approx}{\boldsymbol{F}}$ is invertible

THEN: there exist tensors $\underset{\approx}{\boldsymbol{U}}$, $\underset{\approx}{\boldsymbol{V}}$, and $\underset{\approx}{\boldsymbol{R}}$ such that

(i) $\underset{\approx}{\boldsymbol{F}} = \underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\boldsymbol{U}} = \underset{\approx}{\boldsymbol{V}} \bullet \underset{\approx}{\boldsymbol{R}}$,

(ii) $\underset{\approx}{\boldsymbol{U}}$ and $\underset{\approx}{\boldsymbol{V}}$ are symmetric

(iii) $\underset{\approx}{\boldsymbol{U}}$ and $\underset{\approx}{\boldsymbol{V}}$ are positive definite

(iv) $\underset{\approx}{\boldsymbol{R}}$ is orthogonal (*i.e.*, $\underset{\approx}{\boldsymbol{R}}^{-1} = \underset{\approx}{\boldsymbol{R}}^{T}$

Corollary: $\underset{\approx}{\boldsymbol{U}}$, $\underset{\approx}{\boldsymbol{V}}$, and $\underset{\approx}{\boldsymbol{R}}$ are unique. They are the only tensors that satisfy all of the conditions *(i)* through *(iv)*.

To prove the theorem, we will first assume that $\underset{\approx}{\boldsymbol{U}}$, $\underset{\approx}{\boldsymbol{V}}$, and $\underset{\approx}{\boldsymbol{R}}$ do indeed exist. We will then derive explicit formulas for them as functions of $\underset{\approx}{\boldsymbol{F}}$. Since such formulas will have been derived on the *premise* that $\underset{\approx}{\boldsymbol{U}}$, $\underset{\approx}{\boldsymbol{V}}$, and $\underset{\approx}{\boldsymbol{R}}$ exist, we will need to then verify satisfaction of all of the conditions *(i)* through *(iv)* listed above. Assume for now that the decomposition *(i)* $\underset{\approx}{\boldsymbol{F}}=\underset{\approx}{\boldsymbol{R}}\bullet\underset{\approx}{\boldsymbol{U}}$ does indeed exist. Then $\underset{\approx}{\boldsymbol{F}}^T=\underset{\approx}{\boldsymbol{U}}^T\bullet\underset{\approx}{\boldsymbol{R}}^T=\underset{\approx}{\boldsymbol{U}}\bullet\underset{\approx}{\boldsymbol{R}}^{-1}$, where we have used the properties that *(iii)* $\underset{\approx}{\boldsymbol{U}}$ is symmetric and *(iv)* $\underset{\approx}{\boldsymbol{R}}$ is orthogonal. Consequently the product $\underset{\approx}{\boldsymbol{F}}^T\bullet\underset{\approx}{\boldsymbol{F}}$ must equal $\underset{\approx}{\boldsymbol{U}}^2$, and the stretch $\underset{\approx}{\boldsymbol{U}}$ must be given by

$$\underset{\approx}{\boldsymbol{U}} = \underset{\approx}{\boldsymbol{C}}^{1/2}, \text{ where } \underset{\approx}{\boldsymbol{C}} \equiv \underset{\approx}{\boldsymbol{F}}^T\bullet\underset{\approx}{\boldsymbol{F}} \tag{10.43}$$

The helper tensor $\underset{\approx}{\boldsymbol{C}}$ is positive definite because

$$\underset{\sim}{\boldsymbol{u}}\bullet\underset{\approx}{\boldsymbol{C}}\bullet\underset{\sim}{\boldsymbol{u}} = \underset{\sim}{\boldsymbol{u}}\bullet\underset{\approx}{\boldsymbol{F}}^T\bullet\underset{\approx}{\boldsymbol{F}}\bullet\underset{\sim}{\boldsymbol{u}} = (\underset{\approx}{\boldsymbol{F}}\bullet\underset{\sim}{\boldsymbol{u}})\bullet(\underset{\approx}{\boldsymbol{F}}\bullet\underset{\sim}{\boldsymbol{u}})$$
$$= \left\|(\underset{\approx}{\boldsymbol{F}}\bullet\underset{\sim}{\boldsymbol{u}})\right\|^2$$
$$> 0 \quad \text{for all nonzero vectors } \underset{\sim}{\boldsymbol{u}} \tag{10.44}$$

Of course the magnitude of *any* vector is non-negative, and therefore $\left\|(\underset{\approx}{\boldsymbol{F}}\bullet\underset{\sim}{\boldsymbol{u}})\right\|^2\geq 0$. Our assertion that $\left\|(\underset{\approx}{\boldsymbol{F}}\bullet\underset{\sim}{\boldsymbol{u}})\right\|^2>0$ is a stronger condition. To rule out the possibility that $\left\|(\underset{\approx}{\boldsymbol{F}}\bullet\underset{\sim}{\boldsymbol{u}})\right\|=0$, we must recall that (by premise) $\underset{\approx}{\boldsymbol{F}}$ is invertible and therefore $\underset{\approx}{\boldsymbol{F}}\bullet\underset{\sim}{\boldsymbol{u}}$ can never be zero for any nonzero $\underset{\sim}{\boldsymbol{u}}$. Note that the helper tensor $\underset{\approx}{\boldsymbol{C}}$ is symmetric. It is therefore diagonal in its principal basis. By finding the eigenvalues α_i and eigenvectors $\underset{\sim}{\delta}_i^C$ of $\underset{\approx}{\boldsymbol{C}}$, we may write

$$[\underset{\approx}{\boldsymbol{C}}] = \begin{bmatrix}\alpha_1 & 0 & 0\\ 0 & \alpha_2 & 0\\ 0 & 0 & \alpha_3\end{bmatrix} \text{ with respect to the principal } \underset{\sim}{\delta}_i^C \text{ triad}$$
where each eigenvalue α_i is strictly positive. (10.45)

In basis notation,

$$\underset{\approx}{\boldsymbol{C}} = \alpha_1\underset{\sim}{\delta}_1^C\underset{\sim}{\delta}_1^C + \alpha_2\underset{\sim}{\delta}_2^C\underset{\sim}{\delta}_2^C + \alpha_3\underset{\sim}{\delta}_3^C\underset{\sim}{\delta}_3^C, \text{ where each } \alpha_i>0 \tag{10.46}$$

We must be very careful when defining the meaning of the "square root" $\underset{\sim}{\boldsymbol{C}}^{1/2}$ in Eq. (10.43). A positive definite and symmetric 3×3 matrix generally has an *infinite* number of square roots, of which at least 8 are symmetric and only one is both symmetric and positive definite.* To ensure that our Eq. (10.43) defines a *unique* stretch $\underset{\sim}{\boldsymbol{U}}$, we demand that the "square root" $\underset{\sim}{\boldsymbol{C}}^{1/2}$ must be interpreted as the unique positive definite square root. Namely, the right stretch tensor $\underset{\sim}{\boldsymbol{U}}$ is given by

$$\underset{\sim}{\boldsymbol{U}} = \lambda_1 \underset{\sim}{\delta}_1^U \underset{\sim}{\delta}_1^U + \lambda_2 \underset{\sim}{\delta}_2^U \underset{\sim}{\delta}_2^U + \lambda_3 \underset{\sim}{\delta}_3^U \underset{\sim}{\delta}_3^U ,$$
$$\text{where } \lambda_i \equiv +\sqrt{\alpha_i} \text{ and } \underset{\sim}{\delta}_i^U = \underset{\sim}{\delta}_1^C \tag{10.47}$$

In matrix notation,

$$[\underset{\sim}{\boldsymbol{U}}] = +[\underset{\sim}{\boldsymbol{C}}^{1/2}] = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \text{ with respect to the principal } \underset{\sim}{\delta}_i^C \text{ triad}$$
$$\text{where } \lambda_i \equiv +\sqrt{\alpha_i} \tag{10.48}$$

In practice, this result is phrased in terms of the laboratory basis as

$$[\underset{\sim}{\boldsymbol{U}}] = [Q] \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} [Q]^T \text{ with respect to the lab } \underset{\sim}{\boldsymbol{E}}_i \text{ triad}$$
$$\text{where } \lambda_i \equiv +\sqrt{\alpha_i} \text{ and } Q_{ij} = \underset{\sim}{\delta}_i^C \bullet \underset{\sim}{\boldsymbol{E}}_j \tag{10.49}$$

The i^{th} column of the transformation matrix $[Q]$ contains the lab components of the $\underset{\sim}{\delta}_i^C$ eigenvector. There is only one square root of $\underset{\sim}{\boldsymbol{C}}$ that is both symmetric and positive definite. Hence, the stretch defined in Eq. (10.48) is the *only* stretch that satisfies the premise of the theorem, so we have proved that if $\underset{\sim}{\boldsymbol{U}}$ exists, then it is unique. Once $\underset{\sim}{\boldsymbol{U}}$ is known, the rotation tensor can be found by

$$\underset{\sim}{\boldsymbol{R}} = \underset{\sim}{\boldsymbol{F}} \bullet \underset{\sim}{\boldsymbol{U}}^{-1} \tag{10.50}$$

This is a single valued operation. Hence, uniqueness of $\underset{\sim}{\boldsymbol{U}}$ guarantees uniqueness of $\underset{\sim}{\boldsymbol{R}}$. The tensor $\underset{\sim}{\boldsymbol{R}}$ is indeed orthogonal because

$$\underset{\sim}{\boldsymbol{R}}^T \bullet \underset{\sim}{\boldsymbol{R}} = \underset{\sim}{\boldsymbol{U}}^{-1} \bullet \underset{\sim}{\boldsymbol{F}}^T \bullet \underset{\sim}{\boldsymbol{F}} \bullet \underset{\sim}{\boldsymbol{U}}^{-1} = \underset{\sim}{\boldsymbol{U}}^{-1} \bullet \underset{\sim}{\boldsymbol{U}}^2 \bullet \underset{\sim}{\boldsymbol{U}}^{-1} = \underset{\sim}{\boldsymbol{I}} \tag{10.51}$$

By similar arguments, the left stretch $\underset{\sim}{\boldsymbol{V}}$ must be given by

$$\underset{\sim}{\boldsymbol{V}} = +\underset{\sim}{\boldsymbol{B}}^{1/2}, \qquad \text{where} \qquad \underset{\sim}{\boldsymbol{B}} \equiv \underset{\sim}{\boldsymbol{F}} \bullet \underset{\sim}{\boldsymbol{F}}^T \tag{10.52}$$

---

* To understand the reason for these statements, consider the following symmetric and nonsymmetric square roots of the 2×2 identity matrix: $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $\begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$, $\begin{bmatrix} +1 & 0 \\ 0 & -1 \end{bmatrix}$, $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$, $\begin{bmatrix} 2 & 3 \\ -1 & -2 \end{bmatrix}$

As before, to ensure that $\underset{\approx}{\boldsymbol{V}}$ will be a proper stretch, $\underset{\approx}{\boldsymbol{B}}^{1/2}$ must be taken to be the unique symmetric and positive-definite square root. Once $\underset{\approx}{\boldsymbol{V}}$ is known, an orthogonal tensor $\underset{\approx}{\boldsymbol{Q}}$ can be found by

$$\hat{\underset{\approx}{\boldsymbol{R}}} = \underset{\approx}{\boldsymbol{V}}^{-1} \bullet \underset{\approx}{\boldsymbol{F}} \tag{10.53}$$

So far, we have proved existence of stretches, $\underset{\approx}{\boldsymbol{U}}$ and $\underset{\approx}{\boldsymbol{V}}$, and rotations, $\underset{\approx}{\boldsymbol{R}}$ and $\hat{\underset{\approx}{\boldsymbol{R}}}$, such that

$$\underset{\approx}{\boldsymbol{F}} = \underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\boldsymbol{U}} = \underset{\approx}{\boldsymbol{V}} \bullet \hat{\underset{\approx}{\boldsymbol{R}}} \tag{10.54}$$

All that remains is to prove that the orthogonal tensor from the left stretch is the same as the orthogonal tensor from the right stretch. In other words, we must prove that $\hat{\underset{\approx}{\boldsymbol{R}}} = \underset{\approx}{\boldsymbol{R}}$. Applying the definition of the left stretch, we know that $\underset{\approx}{\boldsymbol{U}}^2 = \underset{\approx}{\boldsymbol{F}}^T \bullet \underset{\approx}{\boldsymbol{F}} = (\underset{\approx}{\boldsymbol{V}} \bullet \hat{\underset{\approx}{\boldsymbol{R}}})^T \bullet \underset{\approx}{\boldsymbol{V}} \bullet \hat{\underset{\approx}{\boldsymbol{R}}} = \hat{\underset{\approx}{\boldsymbol{R}}}^T \bullet \underset{\approx}{\boldsymbol{V}}^2 \bullet \hat{\underset{\approx}{\boldsymbol{R}}}$ from which it follows that $\underset{\approx}{\boldsymbol{U}} = \hat{\underset{\approx}{\boldsymbol{R}}}^T \bullet \underset{\approx}{\boldsymbol{V}} \bullet \hat{\underset{\approx}{\boldsymbol{R}}}$. Substituting this result back into Eq. (10.54), shows that $\underset{\approx}{\boldsymbol{R}} \bullet \hat{\underset{\approx}{\boldsymbol{R}}}^T = \underset{\approx}{\boldsymbol{I}}$ and therefore

$$\hat{\underset{\approx}{\boldsymbol{R}}} = \underset{\approx}{\boldsymbol{R}}. \tag{10.55}$$

Working with the inverse gradient. For some applications (such as simple shear), it is easier to work with the *inverse* of the deformation gradient. In such cases, the stretches may be computed by

$$\underset{\approx}{\boldsymbol{U}} = (\underset{\approx}{\boldsymbol{F}}^{-1} \bullet \underset{\approx}{\boldsymbol{F}}^{-T})^{-1/2} \qquad \text{and} \qquad \underset{\approx}{\boldsymbol{V}} = (\underset{\approx}{\boldsymbol{F}}^{-T} \bullet \underset{\approx}{\boldsymbol{F}}^{-1})^{-1/2} \tag{10.56}$$

Since $\underset{\approx}{\boldsymbol{R}} = \underset{\approx}{\boldsymbol{R}}^{-T}$, the rotation can be computed by

$$\underset{\approx}{\boldsymbol{R}} = \underset{\approx}{\boldsymbol{F}}^{-T} \bullet \underset{\approx}{\boldsymbol{U}} \qquad \text{and} \qquad \underset{\approx}{\boldsymbol{R}} = \underset{\approx}{\boldsymbol{V}} \bullet \underset{\approx}{\boldsymbol{F}}^{-T} \tag{10.57}$$

A good example of a situation where you would work with the inverse gradient is in finite element calculations. In a Lagrangian calculation, the nodes move with the material. Consequently, the reference position $\underset{\sim}{\boldsymbol{X}}$ of each node is constant throughout time for a Lagrangian code. The current position $\underset{\sim}{\boldsymbol{x}}$ of a node varies in time. From a computational standpoint, it is often easier to use the element shape functions to compute the gradient of $\underset{\sim}{\boldsymbol{X}}$ with respect to $\underset{\sim}{\boldsymbol{x}}$. This gradient is the inverse deformation gradient:

$$\underset{\approx}{\boldsymbol{F}}^{-1} = \frac{\partial \underset{\sim}{\boldsymbol{X}}}{\partial \underset{\sim}{\boldsymbol{x}}} \tag{10.58}$$

Equations (10.56) and (10.57) show that it is not necessary to invert Eq. (10.58) in order to perform a polar decomposition.

The *FAST* way to do a polar decomposition in two dimensions

For planar deformations, the deformation gradient tensor is of the form

$$[F] = \begin{bmatrix} F_{11} & F_{12} & 0 \\ F_{21} & F_{22} & 0 \\ 0 & 0 & F_{33} \end{bmatrix}, \text{ where } F_{33} > 0 \tag{10.59}$$

We claim by serendipity* that the polar decomposition can be performed rapidly by the following formula:

$$[R] = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \text{and} \quad [U] = [R]^T[F] \tag{10.60}$$

where

$$\cos\theta = \frac{F_{11} + F_{22}}{\sqrt{(F_{11} + F_{22})^2 + (F_{21} - F_{12})^2}}$$
$$\sin\theta = \frac{F_{21} - F_{12}}{\sqrt{(F_{11} + F_{22})^2 + (F_{21} - F_{12})^2}} \tag{10.61}$$

Beware! You must define $\cos\theta$ and $\sin\theta$ *separately* in order to uniquely determine the rotation angle. It is certainly true that

$$\tan\theta = (F_{21} - F_{12})/(F_{11} + F_{22}), \quad \leftarrow \text{not enough!} \tag{10.62}$$

This relation does not *uniquely* define the rotation angle because there are always *two* angles θ in the range from 0 to 2π that satisfy the above equation. By contrast there is only *one* angle in the range from 0 to 2π that satisfies Eq. (10.61).

* It's easy to verify that our formulas yield an orthogonal [R] and a symmetric [U]. It is straightforward, but tedious, to also prove that our formula gives a *positive definite* [U] matrix. This property is essential in order to confirm the validity of our serendipitous formulas.

For computer applications, the following coding will very rapidly perform a planar polar decomposition:

```
c = f(1,1)+f(2,2)
s = f(2,1)-f(1,2)
d = sqrt(c*c + s*s)
c = c/d
s = s/d

R(1,1)=c
R(2,2)=c
R(1,2)=-s
R(2,1)=s

U(1,1)=c*F(1,1)+s*F(2,1)
U(1,2)=c*F(1,2)+s*F(2,2)
U(2,1)=c*F(2,1)-s*F(1,1)
U(2,2)=c*F(2,2)-s*F(1,2)

R(3,3)=1.
R(1,3)=0.
R(2,3)=0.
R(3,1)=0.
R(3,2)=0.

U(3,3)=F(3,3)
U(1,3)=0.
U(2,3)=0.
U(3,1)=0.
U(3,2)=0.
```

11. "Mixing" or interpolating rotations

Eulerian physics codes (or Lagrangian codes that remap quantities) must routinely handle the task of mixing two or more materials in a computational cell and assigning reasonable homogenized values for the field variables of the mixture. In the absence of specific arguments to the contrary, field variables are typically mixed by simple mass or volume averaging. This approach, however, can be unsatisfactory when mixing rotation tensors because the sum of two orthogonal tensors is not itself generally orthogonal. A similar problem occurs when attempting to interpolate between two rotations in a discrete representation of a field.

Code architects are faced with an interesting situation: should they allow the mixed value of orthogonal tensors to be nonorthogonal, or should they develop a rule for *forcing* the mixed rotation to be orthogonal. Surprisingly, there are some arguments in favor of allowing a mixture of rotations to not be a rotation itself (the line of reasoning is that, statistically speaking, averages shouldn't be expected to be representative of the things being averaged*). However, subroutines that require a rotation often presume that the rotation supplied will indeed be orthogonal, and therefore, one needs a rule for mixing rotations that will always give an orthogonal result.

Below we list possible solutions to this problem, starting with what we believe will give the best results, and working down to less viable solutions.

proposal #1: Map and re-compute the polar decomposition

The *initial* position vector $\underset{\sim}{\boldsymbol{X}}$ may be regarded as a field in its own right. In terms of the mapping function of Eq. (10.1), it is given by

$$\underset{\sim}{\boldsymbol{X}} = \chi^{-1}(\underset{\sim}{\boldsymbol{x}}) \tag{11.1}$$

The vector $\underset{\sim}{\boldsymbol{X}}$ may be advected just like any other field. This is particularly easy to do since the *Lagrangian* time rate of $\underset{\sim}{\boldsymbol{X}}$ is always zero. Thus, the value of $\underset{\sim}{\boldsymbol{X}}$ at a *Lagrangian* tracer particle or at a *Lagrangian* mesh point remains fixed over time. For Lagrangian finite element calculations, the value of $\underset{\sim}{\boldsymbol{X}}$ at any point p may always be determined by using the shape functions. When it is determined that the mesh should be moved in a *non*-Lagrangian manner (as for remeshing or for ALE calculations), the value of $\underset{\sim}{\boldsymbol{X}}$ can be mapped to the new locations of the nodes just as any other variable would be mapped to the nodes.

* For example, the average location of a point in a body (i.e., its centroid) need not be a point that is actually in the body. More colorfully, the Surgeon General should allocate research funding based on a composite "average" human being who has one breast and one testicle even though there is no one who fits this description!

In short, the field for $\underset{\sim}{\boldsymbol{X}}$ is very easy to monitor for all time. To determine the polar rotation and stretch, we recommend recomputing the *inverse* deformation gradient tensor by taking the spatial gradient of the $\underset{\sim}{\boldsymbol{X}}$ field. Namely,

$$\underset{\approx}{\boldsymbol{F}}^{-1} = \frac{\partial \underset{\sim}{\boldsymbol{X}}}{\partial \underset{\sim}{\boldsymbol{x}}} \quad \Rightarrow \quad F^{-1}_{ij} = \frac{\partial X_i}{\partial x_j} \Rightarrow \quad [F]^{-1} = \begin{bmatrix} \frac{\partial X_1}{\partial x_1} & \frac{\partial X_1}{\partial x_2} & \frac{\partial X_1}{\partial x_3} \\ \frac{\partial X_2}{\partial x_1} & \frac{\partial X_2}{\partial x_2} & \frac{\partial X_2}{\partial x_3} \\ \frac{\partial X_3}{\partial x_1} & \frac{\partial X_3}{\partial x_2} & \frac{\partial X_3}{\partial x_3} \end{bmatrix} \tag{11.2}$$

One could always invert this matrix to obtain the new deformation gradient, but that would entail unnecessary computational cost. It is much more efficient to work directly with $\underset{\approx}{\boldsymbol{F}}^{-1}$. Recall that

$$\underset{\approx}{\boldsymbol{F}} = \underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\boldsymbol{U}} = \underset{\approx}{\boldsymbol{V}} \bullet \underset{\approx}{\boldsymbol{R}} \tag{11.3}$$

Noting that $\underset{\approx}{\boldsymbol{R}}^{-1} = \underset{\approx}{\boldsymbol{R}}^{T}$, taking the inverse gives

$$\underset{\approx}{\boldsymbol{F}}^{-1} = \underset{\approx}{\boldsymbol{U}}^{-1} \bullet \underset{\approx}{\boldsymbol{R}}^{T} = \underset{\approx}{\boldsymbol{R}}^{T} \bullet \underset{\approx}{\boldsymbol{V}}^{-1} \tag{11.4}$$

The stretch $\underset{\approx}{\boldsymbol{U}}$ can be obtained directly from $\underset{\approx}{\boldsymbol{F}}^{-1}$ by

$$\boxed{\underset{\approx}{\boldsymbol{U}} = (\underset{\approx}{\boldsymbol{F}}^{-1} \bullet \underset{\approx}{\boldsymbol{F}}^{-T})^{-1/2}} \tag{11.5}$$

and then the rotation is obtained directly from Eq. (11.4) as

$$\boxed{\underset{\approx}{\boldsymbol{R}} = \underset{\approx}{\boldsymbol{F}}^{-T} \bullet \underset{\approx}{\boldsymbol{U}}} \tag{11.6}$$

The above boxed calculations are fairly easy to perform since $\underset{\approx}{\boldsymbol{F}}^{-1}$ would be already available. With this approach, it is never necessary to invert a nonsymmetric matrix. For planar calculations, the shortcut of Eq. (10.59) can also be appropriately modified to work directly with $\underset{\approx}{\boldsymbol{F}}^{-1}$.

proposal #2: Discard the "stretch" part of a mixed rotation.

A second technique for directly mapping/mixing rotations would be to go ahead and average the rotation tensors to obtain a new tensor $\underset{\approx}{\boldsymbol{F}}$ that is not orthogonal, but which is probably "nearly" orthogonal. We would then project the nonorthogonal tensor $\underset{\approx}{\boldsymbol{F}}$ to the "nearest" orthogonal tensor. This would involve a minimization calculation. Namely, given a tensor $\underset{\approx}{\boldsymbol{A}}$ (the nonorthogonal mixed rotation), find the tensor $\underset{\approx}{\boldsymbol{R}}$ such that $\underset{\approx}{\boldsymbol{R}}^{T} \bullet \underset{\approx}{\boldsymbol{R}} = \underset{\approx}{\boldsymbol{I}}$ and $\left\| \underset{\approx}{\boldsymbol{R}} - \underset{\approx}{\boldsymbol{A}} \right\|$ is minimized. The difficulty here is that, to our knowledge, this minimization problem has no known solution. I suspect that the polar rotation tensor $\underset{\approx}{\boldsymbol{R}}$ cor-

responding to the tensor $\underset{\approx}{\boldsymbol{F}}$ is the solution to this minimization problem. If so, then this proposal suggests that the orthogonal tensors be mixed and the polar decomposition should be applied to the resulting (generally non-orthogonal) mixture.

Consider an initial tensor $\underset{\approx}{\boldsymbol{F}}$ that is almost (but not quite) orthogonal. Specifically, suppose that the largest eigenvalue of $\underset{\approx}{\boldsymbol{I}} - \underset{\approx}{\boldsymbol{F}}^T \underset{\approx}{\boldsymbol{F}}$ is significantly smaller than unity. Then the associated polar rotation tensor is approximately

$$\underset{\approx}{\boldsymbol{R}} \approx \frac{1}{2} \underset{\approx}{\boldsymbol{F}} \bullet [3\underset{\approx}{\boldsymbol{I}} - \underset{\approx}{\boldsymbol{F}}^T \bullet \underset{\approx}{\boldsymbol{F}}] \tag{11.7}$$

If the above formula is applied recursively to obtain increasingly better estimates for the rotation, then quadratically to the nearest orthogonal matrix in tensor space. Typically, one iteration will suffice. Of course, if the process converges, then it converges to an orthogonal result. However, we are not certain whether or not the converged result will necessarily equal the actual rotation tensor associated with the ORIGINAL F tensor.

Proof:*. Our goal is to find the orthogonal tensor $\underset{\approx}{\boldsymbol{R}}$ that is closest to a given (nearly orthogonal) tensor $\underset{\approx}{\boldsymbol{F}}$.

The proof begins with at Taylor expansion:

$$(1-x)^a = 1 - ax + O(x^2) \tag{11.8}$$

The substitution $x = 1 - y$ yields

$$y^a = 1 - a(1-y) + O(1-y)^2 \tag{11.9}$$

The above expansion was for scalar arguments, but it also extends to any symmetric tensor $\underset{\approx}{\boldsymbol{Y}}$. In particular, if we pick the exponent $a = -1/2$, we get

$$\underset{\approx}{\boldsymbol{Y}}^{-1/2} = \underset{\approx}{\boldsymbol{I}} + \frac{1}{2}(\underset{\approx}{\boldsymbol{I}} - \underset{\approx}{\boldsymbol{Y}}) + O(\underset{\approx}{\boldsymbol{I}} - \underset{\approx}{\boldsymbol{Y}})^2 \tag{11.10}$$

Let's turn our attention to the polar decomposition of $\underset{\approx}{\boldsymbol{F}}$:

$$\underset{\approx}{\boldsymbol{F}} = \underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\boldsymbol{U}} \tag{11.11}$$

Define

$$\underset{\approx}{\boldsymbol{C}} = \underset{\approx}{\boldsymbol{F}}^T \underset{\approx}{\boldsymbol{F}} = \underset{\approx}{\boldsymbol{U}}^2 \tag{11.12}$$

Therefore

$$\underset{\approx}{\boldsymbol{U}}^{-1} = (\underset{\approx}{\boldsymbol{F}}^T \underset{\approx}{\boldsymbol{F}})^{-1/2} \tag{11.13}$$

We know that

* This proof follows the basic ideas communicated in an email to me from Glynne Casteel, though he used the SVD decomposition from matrix analysis. Our presentation recognizes that such a decomposition is equivalent to the polar decomposition.

$$\underset{\approx}{\boldsymbol{R}} = \underset{\approx}{\boldsymbol{F}} \bullet \underset{\approx}{\boldsymbol{U}}^{-1} = \underset{\approx}{\boldsymbol{F}}\underset{\approx}{\boldsymbol{C}}^{-1/2} \tag{11.14}$$

If the eigenvalues of $\underset{\approx}{\boldsymbol{U}}$ are close to unity (i.e., if $\underset{\approx}{\boldsymbol{F}}$ is very nearly orthogonal), then the eigenvalues of $\underset{\approx}{\boldsymbol{C}}$ are also close to unity. By Taylor expansion,

$$\underset{\approx}{\boldsymbol{C}}^{-1/2} \approx \underset{\approx}{\boldsymbol{I}} + \frac{1}{2}(\underset{\approx}{\boldsymbol{I}} - \underset{\approx}{\boldsymbol{C}}) = \underset{\approx}{\boldsymbol{I}} + \frac{1}{2}(\underset{\approx}{\boldsymbol{I}} - \underset{\approx}{\boldsymbol{F}}^T\underset{\approx}{\boldsymbol{F}}) = \frac{3}{2}\underset{\approx}{\boldsymbol{I}} - \frac{1}{2}\underset{\approx}{\boldsymbol{F}}^T\underset{\approx}{\boldsymbol{F}} \tag{11.15}$$

Putting this into Eq. (11.14) shows that, to second order,

$$\underset{\approx}{\boldsymbol{R}} \approx \frac{1}{2}\underset{\approx}{\boldsymbol{F}} \bullet [3\underset{\approx}{\boldsymbol{I}} - \underset{\approx}{\boldsymbol{F}}^T\underset{\approx}{\boldsymbol{F}}] \tag{11.16}$$

proposal #3: mix the pseudo-rotation-vectors

Another strategy for mixing rotation tensors is to use a conventional mass or volume average of the rotation *vectors* and to then convert the average rotation vector back into a rotation tensor. The algorithm is as follows:

STEP 1. For each rotation tensor $\underset{\approx}{\boldsymbol{R}}^{(k)}$ in the mixture, use the technique described on page 32 to compute the rotation vector $\underset{\sim}{\boldsymbol{r}}^{(k)}$.

STEP 2. Use ordinary volume or mass averaging to compute the average rotation vector.

STEP 3. Use Eq. (3.14) to construct the associated mixture rotation tensor $\underset{\approx}{\boldsymbol{R}}$.

The above scheme for averaging rotations basically assumes that all the individual rotations are applied simultaneously, as discussed on page 87. The fact that angle and axis are not unique could cause problems with this method.

proposal #4: mix the quaternions

Similar to the preceding proposal, the quaternions for each rotation could be computed and interpreted as points on a four-dimensional unit sphere. The normalized vector sum of the rotation quaternions would probably do a decent job at assigning a mixed rotation.

12. Rates of rotation

The "spin" tensor

Consider a vector $\underset{\sim}{b}$ that is embedded in the rigidly rotating body. Let $\underset{\sim}{\bar{b}}$ denote the initial position of $\underset{\sim}{b}$. Then

$$\underset{\sim}{b} = \underset{\approx}{R} \bullet \underset{\sim}{\bar{b}} \tag{12.1}$$

Suppose that the rotation of the body is constantly changing. Then $\underset{\approx}{R}=\underset{\approx}{R}(t)$. Note that $\underset{\sim}{\bar{b}}$ is not a function of time (it is simply the *initial* position, which is forever constant). Thus, the time rate of the embedded vector $\underset{\sim}{b}$ is

$$\dot{\underset{\sim}{b}} = \dot{\underset{\approx}{R}} \bullet \underset{\sim}{\bar{b}} \tag{12.2}$$

From Eq (12.1), we know that $\underset{\sim}{\bar{b}} = \underset{\approx}{R}^{-1} \bullet \underset{\sim}{b} = \underset{\approx}{R}^{T} \bullet \underset{\sim}{b}$. Substituting this result in Eq (12.2) gives

$$\dot{\underset{\sim}{b}} = \underset{\approx}{\Omega} \bullet \underset{\sim}{b} \tag{12.3}$$

where the second-order tensor $\underset{\approx}{\Omega}$ is called the *spin tensor** and is defined

$$\underset{\approx}{\Omega} \equiv \dot{\underset{\approx}{R}} \bullet \underset{\approx}{R}^{T} \tag{12.4}$$

For later use, $\dot{\underset{\approx}{R}}$ may be expressed in terms of $\underset{\approx}{\Omega}$ by

$$\dot{\underset{\approx}{R}} = \underset{\approx}{\Omega} \bullet \underset{\approx}{R} \tag{12.5}$$

Consider the orthogonality condition

$$\underset{\approx}{R} \bullet \underset{\approx}{R}^{T} = \underset{\approx}{I} \tag{12.6}$$

Taking the time rate gives:

$$\dot{\underset{\approx}{R}} \bullet \underset{\approx}{R}^{T} + \underset{\approx}{R} \bullet \dot{\underset{\approx}{R}}^{T} = \underset{\approx}{0} \tag{12.7}$$

Substituting Eq. (12.5) into Eq. (12.7) and imposing Eq. (12.6) shows that

$$\underset{\approx}{\Omega} + \underset{\approx}{\Omega}^{T} = \underset{\approx}{0} \tag{12.8}$$

In other words, the spin tensor is skew-symmetric. Consequently, as discussed in the next section, it has an associated axial vector $\underset{\sim}{\omega}$ which is defined such that Eq. (12.3) becomes

$$\dot{\underset{\sim}{b}} = \underset{\sim}{\omega} \times \underset{\sim}{b} \tag{12.9}$$

---

* If the rotation tensor $\underset{\approx}{R}$ is the *polar* rotation, the spin $\underset{\approx}{\Omega}$ is called the **polar spin**.

The angular velocity vector

Just as rotations are associated with an axis and angle of rotation, a spin tensor $\underset{\approx}{\Omega}$ also has an associated angular velocity vector, which is given by the so-called "axial" or "dual" vector:

$$\underset{\sim}{\omega} = -\tfrac{1}{2}\underset{\approx}{\underset{\approx}{\varepsilon}}:\underset{\approx}{\Omega}, \qquad \text{or} \qquad \omega_i = -\tfrac{1}{2}\varepsilon_{ijk}\Omega_{jk} = \tfrac{1}{2}\varepsilon_{jik}\Omega_{jk} \tag{12.10}$$

The operation is reversible. Given the angular velocity vector, the associated angular velocity tensor is computed by.

$$\underset{\approx}{\Omega} = -\underset{\approx}{\underset{\approx}{\varepsilon}} \bullet \underset{\sim}{\omega} \qquad \text{or} \qquad \Omega_{ij} = -\varepsilon_{ijk}\omega_k = \varepsilon_{ikj}\omega_k \tag{12.11}$$

The explicit relationships are

$$\omega_1 = -\Omega_{23} = \Omega_{32}, \quad \omega_2 = -\Omega_{31} = \Omega_{13}, \quad \omega_3 = -\Omega_{12} = \Omega_{21} \tag{12.12}$$

The matrix relationship is

$$[\underset{\approx}{\Omega}] = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \tag{12.13}$$

Dotting the spin tensor $\underset{\approx}{\Omega}$ onto an arbitrary vector $\underset{\sim}{\boldsymbol{u}}$ is equivalent to crossing the angular velocity vector $\underset{\sim}{\omega}$ into $\underset{\sim}{\boldsymbol{u}}$. That is,

$$\underset{\approx}{\Omega} \bullet \underset{\sim}{\boldsymbol{u}} \equiv \underset{\sim}{\omega} \times \underset{\sim}{\boldsymbol{u}} \tag{12.14}$$

As a matter of fact, the angular velocity vector was constructed specifically so that this relationship would hold.

Final question: what relationship, if any, does the angular velocity vector have with the angular rotation vector? If the rotation axis $\underset{\sim}{\boldsymbol{a}}$ is held fixed, then $\underset{\sim}{\omega} = \dot{\alpha}\underset{\sim}{\boldsymbol{a}}$. However, if the axis of rotation is a function of time, the relationship between the rotation vector and the angular velocity vector is highly nonlinear due to the presence of coriolis-type terms, as discussed in the next section.

Angular velocity in terms of axis and angle of rotation

Taking the time rate of Eq. (3.14) gives $\dot{\underset{\approx}{R}}$ as

$$\dot{\underset{\approx}{R}} = -s\dot{\alpha}(\underset{\approx}{I} - \underset{\sim}{a}\underset{\sim}{a}) - c\dot{\alpha}\underset{\approx\approx}{\varepsilon} \bullet \underset{\sim}{a} + (1-c)[\dot{\underset{\sim}{a}}\underset{\sim}{a} + \underset{\sim}{a}\dot{\underset{\sim}{a}}] - s\underset{\approx\approx}{\varepsilon} \bullet \dot{\underset{\sim}{a}} \tag{12.15}$$

where $c \equiv \cos\alpha$ and $s \equiv \sin\alpha$. The transpose of Eq. (3.14) is

$$\underset{\approx}{R}^T = c(\underset{\approx}{I} - \underset{\sim}{a}\underset{\sim}{a}) + \underset{\sim}{a}\underset{\sim}{a} + s\,\underset{\approx\approx}{\varepsilon} \bullet \underset{\sim}{a} \tag{12.16}$$

After much simplification,* the spin ($\underset{\approx}{\Omega} \equiv \dot{\underset{\approx}{R}} \bullet \underset{\approx}{R}^T$) is given by

$$\underset{\approx}{\Omega} = \dot{\alpha}\underset{\approx}{A} + c(1-c)[\underset{\sim}{a}\dot{\underset{\sim}{a}} - \dot{\underset{\sim}{a}}\underset{\sim}{a}] + s\dot{\underset{\approx}{A}} \tag{12.17}$$

where

$$\underset{\approx}{A} = -\underset{\approx\approx}{\varepsilon} \bullet \underset{\sim}{a}, \quad \text{and} \quad \dot{\underset{\approx}{A}} = -\underset{\approx\approx}{\varepsilon} \bullet \dot{\underset{\sim}{a}} \tag{12.18}$$

Taking the axial vector of Eq. (12.17) gives the angular velocity vector as

$$\boxed{\underset{\sim}{\omega} = \dot{\alpha}\underset{\sim}{a} + c(1-c)(\dot{\underset{\sim}{a}} \times \underset{\sim}{a}) + s\dot{\underset{\sim}{a}}} \tag{12.19}$$

Difference between vorticity and polar spin.

In continuum mechanics, the velocity gradient tensor $\underset{\approx}{L}$ is the spatial derivative of the velocity field. Thus, if $\underset{\sim}{v}$ is velocity and $\underset{\sim}{x}$ is the spatial position vector, then

$$L_{ij} = \frac{\partial v_i}{\partial x_j} \tag{12.20}$$

The so-called "rate" of deformation tensor $\underset{\approx}{D}$ is defined to equal the symmetric part of the velocity gradient:

$$D_{ij} = \frac{1}{2}(L_{ij} + L_{ji}) = \frac{1}{2}\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right) \tag{12.21}$$

The vorticity tensor $\underset{\approx}{W}$ is defined to be the skew-symmetric part of the velocity gradient tensor:

$$W_{ij} = \frac{1}{2}(L_{ij} - L_{ji}) = \frac{1}{2}\left(\frac{\partial v_i}{\partial x_j} - \frac{\partial v_j}{\partial x_i}\right) \tag{12.22}$$

The vorticity vector $\underset{\sim}{w}$ is the axial vector associated with $\underset{\approx}{W}$:

$$\underset{\sim}{w} = -\tfrac{1}{2}\underset{\approx\approx}{\varepsilon} : \underset{\approx}{W} \tag{12.23}$$

Written out in Cartesian components,

* It is critical to note that $\underset{\sim}{a} \bullet \underset{\sim}{a} = 1$ and therefore $\dot{\underset{\sim}{a}} \bullet \underset{\sim}{a} = 0$.

$$w_1 = \frac{1}{2}\left(\frac{\partial v_3}{\partial x_2} - \frac{\partial v_2}{\partial x_3}\right)$$
$$w_2 = \frac{1}{2}\left(\frac{\partial v_1}{\partial x_3} - \frac{\partial v_3}{\partial x_1}\right)$$
$$w_3 = \frac{1}{2}\left(\frac{\partial v_2}{\partial x_1} - \frac{\partial v_1}{\partial x_2}\right) \tag{12.24}$$

A fundamental theorem from continuum mechanics states that the velocity gradient is related to the deformation gradient tensor $\underset{\approx}{\boldsymbol{F}}$ by

$$\underset{\approx}{\boldsymbol{L}} = \dot{\underset{\approx}{\boldsymbol{F}}} \bullet \underset{\approx}{\boldsymbol{F}}^{-1} \tag{12.25}$$

Recall the polar decomposition:

$$\underset{\approx}{\boldsymbol{F}} = \underset{\approx}{\boldsymbol{V}} \bullet \underset{\approx}{\boldsymbol{R}}, \tag{12.26}$$

where $\underset{\approx}{\boldsymbol{R}}$ is the orthogonal rotation tensor and $\underset{\approx}{\boldsymbol{V}}$ is the symmetric positive definite left stretch. Substituting Eq. (12.26) into (12.25) and noting that $\underset{\approx}{\boldsymbol{L}} = \underset{\approx}{\boldsymbol{D}} + \underset{\approx}{\boldsymbol{W}}$ gives

$$\underset{\approx}{\boldsymbol{D}} + \underset{\approx}{\boldsymbol{W}} = \dot{\underset{\approx}{\boldsymbol{V}}} \bullet \underset{\approx}{\boldsymbol{V}}^{-1} + \underset{\approx}{\boldsymbol{V}} \bullet \underset{\approx}{\Omega} \bullet \underset{\approx}{\boldsymbol{V}}^{-1} \tag{12.27}$$

where the polar spin $\underset{\approx}{\Omega}$ is defined

$$\underset{\approx}{\Omega} \equiv \dot{\underset{\approx}{\boldsymbol{R}}} \bullet \underset{\approx}{\boldsymbol{R}}^{T}, \tag{12.28}$$

The polar angular velocity vector $\underset{\sim}{\omega}$ is defined

$$\underset{\sim}{\omega} = -\tfrac{1}{2}\underset{\approx}{\varepsilon} : \underset{\approx}{\Omega} \tag{12.29}$$

The purpose of the remainder of this section is to summarize Dienes's derivation [8] of the difference between $\underset{\sim}{\boldsymbol{w}}$ and $\underset{\sim}{\omega}$. The transpose of Eq. (12.27) is

$$\underset{\approx}{\boldsymbol{D}} - \underset{\approx}{\boldsymbol{W}} = \underset{\approx}{\boldsymbol{V}}^{-1} \bullet \dot{\underset{\approx}{\boldsymbol{V}}} - \underset{\approx}{\boldsymbol{V}}^{-1} \bullet \underset{\approx}{\Omega} \bullet \underset{\approx}{\boldsymbol{V}} \tag{12.30}$$

Post multiplying Eq. (12.27) by $\underset{\approx}{\boldsymbol{V}}$ gives

$$\underset{\approx}{\boldsymbol{D}} \bullet \underset{\approx}{\boldsymbol{V}} + \underset{\approx}{\boldsymbol{W}} \bullet \underset{\approx}{\boldsymbol{V}} = \dot{\underset{\approx}{\boldsymbol{V}}} + \underset{\approx}{\boldsymbol{V}} \bullet \underset{\approx}{\Omega} \tag{12.31}$$

Taking the skew-symmetric part of both sides gives

$$\boxed{\underset{\approx}{\boldsymbol{Z}} = \underset{\approx}{\boldsymbol{V}} \bullet \underset{\approx}{\boldsymbol{H}} + \underset{\approx}{\boldsymbol{H}} \bullet \underset{\approx}{\boldsymbol{V}}} \tag{12.32}$$

where

$$\boxed{\underset{\approx}{\boldsymbol{Z}} \equiv \underset{\approx}{\boldsymbol{D}} \bullet \underset{\approx}{\boldsymbol{V}} - \underset{\approx}{\boldsymbol{V}} \bullet \underset{\approx}{\boldsymbol{D}}} \tag{12.33}$$

and

$$\boxed{\underset{\approx}{\boldsymbol{H}} \equiv \underset{\approx}{\Omega} - \underset{\approx}{\boldsymbol{W}}} \tag{12.34}$$

We are interested in the difference between the vorticity and the polar spin. In other words, we seek to solve Eq. (12.32) for the tensor $\underset{\approx}{\boldsymbol{H}}$. The tensor $\underset{\approx}{\boldsymbol{H}}$ is skew-symmetric, so there exists an axial vector $\underset{\sim}{\boldsymbol{h}}$ such that

$$H_{ij} = \varepsilon_{ikj} h_k \tag{12.35}$$

Similarly, there exists an axial vector $\underset{\sim}{\boldsymbol{z}}$ such that

$$Z_{ij} = \varepsilon_{ikj} z_k \tag{12.36}$$

Thus, the component form of Eq. (12.32) can be written

$$\varepsilon_{ikj} z_k = V_{im}\varepsilon_{mpj} h_p + \varepsilon_{iqn} h_q V_{nj} \tag{12.37}$$

Multiplying both sides by ε_{irj} and using the identity $\varepsilon_{irj}\varepsilon_{ikj} = 2\delta_{rk}$ gives

$$2\delta_{rk} z_k = \varepsilon_{irj} V_{im}\varepsilon_{mpj} h_p + \varepsilon_{irj}\varepsilon_{iqn} h_q V_{nj} \tag{12.38}$$

Noting that $\varepsilon_{irj}\varepsilon_{mpj} = \delta_{im}\delta_{rp} + \delta_{ip}\delta_{rm}$ and $\varepsilon_{irj}\varepsilon_{iqn} = \delta_{rq}\delta_{jn} + \delta_{rn}\delta_{jq}$, this becomes

$$2\delta_{rk} z_k = (\delta_{im}\delta_{rp} + \delta_{ip}\delta_{rm}) V_{im} h_p + (\delta_{rq}\delta_{jn} + \delta_{rn}\delta_{jq}) h_q V_{nj} \tag{12.39}$$

or, eliminating all Kronecker deltas,

$$2 z_r = (V_{mm} h_r + V_{pr} h_p) + (h_r V_{jj} + h_j V_{rj}) \tag{12.40}$$

This may be written in direct notation as

$$\underset{\sim}{\boldsymbol{z}} = [(\mathrm{tr}\underset{\approx}{\boldsymbol{V}})\underset{\approx}{\boldsymbol{I}} + \underset{\approx}{\boldsymbol{V}}] \bullet \underset{\sim}{\boldsymbol{h}} \tag{12.41}$$

Recalling Eq. (12.34), we note that $\underset{\sim}{\boldsymbol{h}} = \underset{\sim}{\omega} - \underset{\sim}{\boldsymbol{w}}$ and therefore the above equation may be inverted to give

$$\boxed{\underset{\sim}{\omega} = \underset{\sim}{\boldsymbol{w}} + [(\mathrm{tr}\underset{\approx}{\boldsymbol{V}})\underset{\approx}{\boldsymbol{I}} + \underset{\approx}{\boldsymbol{V}}]^{-1} \bullet \underset{\sim}{\boldsymbol{z}}} \tag{12.42}$$

From this we see that the polar spin equals the vorticity if and only if $\underset{\sim}{\boldsymbol{z}} = \underset{\sim}{\boldsymbol{0}}$. Recalling Eq. (12.33), this will occur if and only if $\underset{\approx}{\boldsymbol{D}}$ and $\underset{\approx}{\boldsymbol{V}}$ commute,* which implies that there must exist a triad of orthonormal vectors that are principal directions of *both* $\underset{\approx}{\boldsymbol{D}}$ and $\underset{\approx}{\boldsymbol{V}}$. This is naturally the case for rigid motion where $\underset{\approx}{\boldsymbol{D}}=\underset{\approx}{\boldsymbol{0}}$ and $\underset{\approx}{\boldsymbol{V}}=\underset{\approx}{\boldsymbol{I}}$. Another (less trivial) way for $\underset{\approx}{\boldsymbol{D}}$ and $\underset{\approx}{\boldsymbol{V}}$ to commute occurs when the principal directions of $\underset{\approx}{\boldsymbol{V}}$ remain constant throughout all time. In this case both the vorticity and the spin are zero. Finally, note that $\underset{\approx}{\boldsymbol{D}} \bullet \underset{\approx}{\boldsymbol{V}} - \underset{\approx}{\boldsymbol{V}} \bullet \underset{\approx}{\boldsymbol{D}} = \underset{\approx}{\boldsymbol{D}}' \bullet \underset{\approx}{\boldsymbol{V}}' - \underset{\approx}{\boldsymbol{V}}' \bullet \underset{\approx}{\boldsymbol{D}}'$ where a prime denotes the deviatoric part. This identity implies that there is a difference between the vorticity and spin only if the deformation involves distortion — isotropic expansion or contraction has no influence on the distinction between vorticity and spin. For this

* i.e, $\underset{\sim}{\boldsymbol{z}} = \underset{\sim}{\boldsymbol{0}}$ if and only if $\underset{\approx}{\boldsymbol{D}} \bullet \underset{\approx}{\boldsymbol{V}} = \underset{\approx}{\boldsymbol{V}} \bullet \underset{\approx}{\boldsymbol{D}}$.

reason, Dienes [8] considered simple shear as a counterexample to demonstrate that the Jaumann stress rate (based on vorticity) predicts anomalous oscillatory shear stresses when used with a constant modulus hypoelastic constitutive law. The polar rate predicts an intuitive monotonically-increasing shear stress.

The (commonly mis-stated) Gosiewski's theorem

In this section we prove the following theorem:

> *If* $\underset{\sim}{\boldsymbol{d}}$ *is a unit material fiber that is instantaneously coincident with a principal direction of the "rate" of deformation* $\underset{\approx}{\boldsymbol{D}}$*, then its instantaneous rate of rotation is given by the vorticity; i.e., at this instant,* $\dot{\underset{\sim}{\boldsymbol{d}}} = \underset{\approx}{\boldsymbol{W}} \bullet \underset{\sim}{\boldsymbol{d}}$.

This theorem is often misinterpreted to mean that the principal directions of $\underset{\approx}{\boldsymbol{D}}$ rotate according to the vorticity $\underset{\approx}{\boldsymbol{W}}$. To the contrary, it is only the *material fibers* parallel to the principal directions of $\underset{\approx}{\boldsymbol{D}}$ that rotate according to the vorticity. These material fibers do not generally remain aligned with the principal directions of $\underset{\approx}{\boldsymbol{D}}$. Hence, the above theorem (1) applies only to material fibers and (2) only at the instant when they pass over principal directions of $\underset{\approx}{\boldsymbol{D}}$ — any time thereafter, they rotate at a speed that is *not* related to the vorticity.

Before proceeding with the proof of the correct theorem, let's first provide a counterexample that discredits the common false interpretation. We wish to prove that the principal directions of $\underset{\approx}{\boldsymbol{D}}$ do *not* rotate according to the vorticity. To do this, we use simple shear as a counterexample. For this canonical motion, the vorticity is a nonzero constant tensor, but the $\underset{\approx}{\boldsymbol{D}}$ tensor is also constant, so its principal directions can't possibly rotate. Under simple shear, the deformation gradient is of the form

$$[\underset{\approx}{\boldsymbol{F}}] = \begin{bmatrix} 1 & 2\varepsilon & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{12.43}$$

where ε is a time-varying measure of the amount of shear. The corresponding rate of deformation and vorticity tensors are

$$[\underset{\approx}{\boldsymbol{D}}] = \begin{bmatrix} 0 & \dot{\varepsilon} & 0 \\ \dot{\varepsilon} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad [\underset{\approx}{\boldsymbol{W}}] = \begin{bmatrix} 0 & \dot{\varepsilon} & 0 \\ -\dot{\varepsilon} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{12.44}$$

Note that the principal directions of the rate of deformation $\underset{\approx}{\boldsymbol{D}}$ are $\{1, 1, 0\}$, $\{1, -1, 0\}$ and $\{0, 0, 1\}$ which *never change with time*. According to the misinterpretation of Gosiewski's theorem, this would be possible only if the vorticity were zero, which is false. This completes the proof that $\underset{\approx}{\boldsymbol{W}}$ does *not* govern the rotation rate of the principal directions of $\underset{\approx}{\boldsymbol{D}}$.

Proof of the correct theorem. For clarity, we consider a homogenous deformation field. Let $\underset{\sim}{f}$ denote a material fiber. By this, we mean that there exists a material vector $\underset{\sim}{f}_o$ in the reference configuration such that

$$\underset{\sim}{f} = \underset{\approx}{F} \bullet \underset{\sim}{f}_o \tag{12.45}$$

The reference material fiber is fixed in time, so taking rates gives

$$\dot{\underset{\sim}{f}} = \dot{\underset{\approx}{F}} \bullet \underset{\sim}{f}_o \tag{12.46}$$

From continuum mechanics, we know that $\dot{\underset{\approx}{F}} = \underset{\approx}{L} \bullet \underset{\approx}{F}$, where $\underset{\approx}{L}$ is the symmetric part of the velocity gradient. Therefore, the above equation becomes

$$\dot{\underset{\sim}{f}} = \underset{\approx}{L} \bullet \underset{\sim}{f} \tag{12.47}$$

Decomposing $\underset{\approx}{L}$ into its symmetric part $\underset{\approx}{D}$ plus its skew-symmetric part $\underset{\approx}{W}$ gives

$$\dot{\underset{\sim}{f}} = \underset{\approx}{D} \bullet \underset{\sim}{f} + \underset{\approx}{W} \bullet \underset{\sim}{f} \tag{12.48}$$

This relationship holds for *any* material fiber $\underset{\sim}{f}$. We wish to determine what part of $\dot{\underset{\sim}{f}}$ is caused by change in length and what part is caused by change in orientation. Towards this end, we write

$$\underset{\sim}{f} = f \underset{\sim}{n} \qquad \text{where} \tag{12.49}$$

where f is the magnitude of $\underset{\sim}{f}$ defined by

$$f \equiv \sqrt{\underset{\sim}{f} \bullet \underset{\sim}{f}}, \tag{12.50}$$

and $\underset{\sim}{n}$ is a unit vector in the direction of $\underset{\sim}{f}$. In other words, $\underset{\sim}{n} = \underset{\sim}{f}/f$ and

$$\underset{\sim}{n} \bullet \underset{\sim}{n} = 1 \tag{12.51}$$

Taking rates of the above three equations, we note that

$$\dot{\underset{\sim}{f}} = \dot{f} \underset{\sim}{n} + f \dot{\underset{\sim}{n}} \tag{12.52}$$

$$\dot{f} = \frac{\underset{\sim}{f} \bullet \dot{\underset{\sim}{f}}}{f} \tag{12.53}$$

$$\underset{\sim}{n} \bullet \dot{\underset{\sim}{n}} = 0 \tag{12.54}$$

Hence, Eq. (12.48) may be written

$$\dot{f} \underset{\sim}{n} + f \dot{\underset{\sim}{n}} = f \underset{\approx}{D} \bullet \underset{\sim}{n} + f \underset{\approx}{W} \bullet \underset{\sim}{n} \tag{12.55}$$

Dotting both sides by $\underset{\sim}{n}$ gives

$$\boxed{\dot{f} = f D_{nn}}, \text{ where } D_{nn} \equiv \underset{\sim}{n} \bullet \underset{\approx}{D} \bullet \underset{\sim}{n} \tag{12.56}$$

Here, we have noted that $\underset{\sim}{n} \bullet \underset{\approx}{W} \bullet \underset{\sim}{n} = 0$ because $\underset{\approx}{W}$ is skew symmetric. Substituting Eq. (12.56) back into Eq. (12.55) and dividing by f gives

$$\boxed{\dot{\underset{\sim}{n}} = \underset{\approx}{D} \bullet \underset{\sim}{n} - D_{nn}\underset{\sim}{n} + \underset{\approx}{W} \bullet \underset{\sim}{n}} \tag{12.57}$$

The above boxed equations hold for *any* material fiber $\underset{\sim}{f}$ regardless of its instantaneous orientation.

If the material fiber *happens* to instantaneously coincide with a principal direction of $\underset{\approx}{D}$ then *at this instant*, $\underset{\approx}{D} \bullet \underset{\sim}{n} = \lambda \underset{\sim}{n}$ and therefore $D_{nn} = \underset{\sim}{n} \bullet \underset{\approx}{D} \bullet \underset{\sim}{n} = \lambda$. Consequently, *at this instant in time,* the first two terms in Eq. (12.57) cancel leaving only $\dot{\underset{\sim}{n}} = \underset{\approx}{W} \bullet \underset{\sim}{n}$, which proves the theorem that the vorticity governs the angular velocity of material fibers at the moment when they are coincident with a principal direction of $\underset{\approx}{D}$. This special theorem seems of little practical use since the same material fiber will *not* be rotating according to the vorticity an instant later. We have included the theorem here only for the purpose of disproving its common mis-interpretation.

Rates of *sequential* rotations

Consider three rotations, $\underset{\approx}{R}_1$ about the X-axis, $\underset{\approx}{R}_2$ about the Y-axis, and $\underset{\approx}{R}_3$ about the Z-axis. If these rotations are applied sequentially (i.e., the first followed by the second and then the third), we know that the total rotation is given by

$$\underset{\approx}{R} = \underset{\approx}{R}_3 \bullet \underset{\approx}{R}_2 \bullet \underset{\approx}{R}_1 \tag{12.58}$$

Conversely, we also know that for any rotation $\underset{\approx}{R}$, there exist three rotations $\{\underset{\approx}{R}_1, \underset{\approx}{R}_2, \text{and } \underset{\approx}{R}_3\}$ about the laboratory axes such that Eq. (12.58) holds.

If the three axis rotations are known for all time, how is the spin tensor ($\underset{\approx}{\Omega} \equiv \dot{\underset{\approx}{R}} \bullet \underset{\approx}{R}^T$) related to the individual spin tensors for each of the axis rotations?

The time rate of Eq. (12.58) gives

$$\dot{\underset{\approx}{R}} = \dot{\underset{\approx}{R}}_3 \bullet \underset{\approx}{R}_2 \bullet \underset{\approx}{R}_1 + \underset{\approx}{R}_3 \bullet \dot{\underset{\approx}{R}}_2 \bullet \underset{\approx}{R}_1 + \underset{\approx}{R}_3 \bullet \underset{\approx}{R}_2 \bullet \dot{\underset{\approx}{R}}_1 \tag{12.59}$$

Substituting Eq. (12.5) for $\dot{\underset{\approx}{R}}$ and making analogous substitutions for $\dot{\underset{\approx}{R}}_k$ shows that

$$\underset{\approx}{\Omega} = \underset{\approx}{\Omega}_3 + \underset{\approx}{R}_3 \bullet \underset{\approx}{\Omega}_2 \bullet \underset{\approx}{R}_3^{\mathrm{T}} + \underset{\approx}{R}_3 \bullet \underset{\approx}{R}_2 \bullet \underset{\approx}{\Omega}_1 \bullet \underset{\approx}{R}_2^{\mathrm{T}} \bullet \underset{\approx}{R}_1^{\mathrm{T}} \tag{12.60}$$

Thus, in this formulation, we can see that the spin tensor also depends on the order of application of the sequentially applied rotation tensors. In the limit of *infinitesimally small* rotations (but not necessarily small rotation rates), each of the $\underset{\approx}{R}_k$ are approximately equal to the identity tensor and we obtain the result that

$$\underset{\approx}{\Omega} = \underset{\approx}{\Omega}_3 + \underset{\approx}{\Omega}_2 + \underset{\approx}{\Omega}_1 \tag{12.61}$$

Thus, for infinitesimal sequential rotations,

$$\underset{\sim}{\omega} = \underset{\sim}{\omega}_3 + \underset{\sim}{\omega}_2 + \underset{\sim}{\omega}_1 \tag{12.62}$$

For the *special case* that the rotation axis of each $\underset{\approx}{\boldsymbol{R}}_k$ is constant, we have

$$\underset{\sim}{\omega} = \dot{\alpha}_1 \underset{\sim}{\boldsymbol{a}}_1 + \dot{\alpha}_2 \underset{\sim}{\boldsymbol{a}}_2 + \dot{\alpha}_3 \underset{\sim}{\boldsymbol{a}}_3 \tag{12.63}$$

Even for this special case where each rotation axis is constant, the rotation axis of the *total* rotation is not necessarily constant because it may be varied by changing the individual rotation rates $\{\dot{\alpha}_1, \dot{\alpha}_2, \text{ and } \dot{\alpha}_3\}$. This result motivates the study of *simultaneously* applied rotations.

Rates of *simultaneous* rotations

Consider the rotation illustrated in Fig. 7.3. By visual inspection, the rotation axis passes through the point (1,1,1), and the rotation angle is $120°$. One could imagine directly rotating the body about the axis so that rotations would be simultaneously occurring about all axes. For a rotation that is applied directly from zero angle to final angle without changing the axis of rotation, the angular velocity vector is

$$\underset{\sim}{\omega} = \dot{\alpha} \underset{\sim}{\boldsymbol{a}} \tag{12.64}$$

writing the axis as $\underset{\sim}{\boldsymbol{a}} = a_1 \underset{\sim}{\boldsymbol{e}}_1 + a_2 \underset{\sim}{\boldsymbol{e}}_2 + a_3 \underset{\sim}{\boldsymbol{e}}_3$, this equation becomes

$$\underset{\sim}{\omega} = \dot{\alpha}_1 \underset{\sim}{\boldsymbol{e}}_1 + \dot{\alpha}_2 \underset{\sim}{\boldsymbol{e}}_2 + \dot{\alpha}_3 \underset{\sim}{\boldsymbol{e}}_3 \tag{12.65}$$

where

$$\dot{\alpha}_k \equiv \dot{\alpha} a_k \tag{12.66}$$

Once the rotation is complete, the final rotation tensor is given by the Euler-Rodrigues formula, Eq. (3.14). In view of Eq. (12.65), this final rotated configuration can be visualized as the simultaneous application of three rotations about the three coordinate axes with the respective rotation angles given by

$$\alpha_k \equiv \alpha a_k \,. \tag{12.67}$$

Thus, for the rotation in Fig. 7.3, the simultaneously applied angles of rotation about each coordinate axis are $120°/\sqrt{3} = 69.28°$.

Conversely, given the rotation angles $\{\alpha_1, \alpha_2, \alpha_3\}$, the total rotation vector is given by

$$\alpha \underset{\sim}{\boldsymbol{a}} = \alpha_1 \underset{\sim}{\boldsymbol{e}}_1 + \alpha_2 \underset{\sim}{\boldsymbol{e}}_2 + \alpha_3 \underset{\sim}{\boldsymbol{e}}_3 \tag{12.68}$$

Importantly, note that α_k is not equivalent to $\alpha_k + 2\pi$. In other words, it is not admissible to add an integral multiple of 2π to a coordinate rotation angle. Referring to Eq. (12.66), the individual coordinate angles must always be less than or equal to the total rotation angle. Furthermore, it is important to realize that

$$\dot{\alpha}_k \neq \omega_k \tag{12.69}$$

Equality holds only if the total rotation axis never changes with time.

In practice, angular velocity generally changes in both magnitude and orientation. Hence rotations are not generally achieved by rotations about a constant axis. Nevertheless, given any rotation tensor $\boldsymbol{R}$, there always exist a rotation axis and angle regardless of whether the rotation was actually achieved directly through the axis and angle of rotation. There are corresponding coordinate rotation angles $\{\alpha_1, \alpha_2, \alpha_3\}$.

For simultaneous rotations about the coordinate axes, the relationship

$$\boldsymbol{\omega} = \omega_1 \boldsymbol{e}_1 + \omega_2 \boldsymbol{e}_2 + \omega_3 \boldsymbol{e}_3 \tag{12.70}$$

always holds, but because the magnitudes of the angular velocities may vary, the total axis of rotation generally varies with time and is not coincident with the direction of the angular velocity vector.

To find the total rotation axis, the total rotation tensor is first found by solving Eq (12.4) as a differential equation. Once the total rotation is found, the angle and axis of rotation may be determined via Eqs. (3.86) and (3.83). Hence, the axis and angle of rotation are always well-defined, and they may be decomposed into coordinate rotations, but they are related to the angular velocity in a highly-nonlinear way.

The principal point of this section is to show that it *is* possible to define rotation vectors, but they are of limited use because their time rates are not related to the angular velocity in a straightforward way.

13. Random Rotations

For problems at the mesoscopic scale where, for example, crystal anisotropy is modeled, simulations often require a means of randomly assigning a rotation of the crystal from its unit cell reference orientation to the actual orientation [10]. For a macroscopically isotropic material of uniform grain size, a uniformly-random distribution of grain orientations is desired. This section discusses how to achieve such a distribution.

Statistical notation

We will use the notation $p_\gamma(\ )$ to denote the distribution function for any continuously random scalar variable γ. The distribution function is defined such that the probability that $a < \gamma < b$ is given by

$$\int_a^b p_\gamma(\gamma')\,d\gamma' \tag{13.1}$$

The corresponding cumulative distribution is denoted $g_\gamma(\ )$ and is defined

$$g_\gamma(s) = \int_{-\infty}^{s} p_\gamma(\gamma')\,d\gamma' \tag{13.2}$$

Thus, $g_\gamma(s)$ is the probability that $\gamma < s$.

We will let $p_{\nu,\eta}(\ ,\ )$ denote joint distributions for two variables ν and η. The joint distribution is defined such that the probability that $\nu_1 < \nu < \nu_1$ and $\eta_1 < \eta < \eta_1$ is

$$\int_{\nu_1}^{\nu_2}\int_{\eta_1}^{\eta_2} p_{\nu,\eta}(\nu',\eta')\,d\nu'\,d\eta' \tag{13.3}$$

Finally, if $\underset{\sim}{x}$ is a random vector in 3D space, then the distribution function $p_{\underset{\sim}{x}}(\ )$ is defined such that the probability that $\underset{\sim}{x}$ lies in a region of space B is

$$\int_B p_{\underset{\sim}{x}}(\underset{\sim}{x}')\,dV\,, \tag{13.4}$$

where dV is the volume element. Any vector $\underset{\sim}{x}$ can be alternatively expressed in terms of, say, spherical coordinates r, θ, ϕ. Thus, there exists a joint distribution function $p_{r,\theta,\phi}(\ )$. Importantly,

$$p_{\underset{\sim}{x}}(\underset{\sim}{x}) \neq p_{r,\theta,\phi}(r,\theta,\phi) \text{ in general.} \tag{13.5}$$

Uniformly random unit vectors — the theory

Before considering the case of a random rotation, first consider how to randomly select a unit vector such that no unit vector orientation is more likely than any other. Unit vectors may be regarded as points on the unit sphere. Therefore a unit vector $\underset{\sim}{\boldsymbol{n}}$ is completely specified by the spherical coordinates θ and ϕ (not to be confused with Euler angles of the same symbols):

$$\underset{\sim}{\boldsymbol{n}} = \sin\theta\cos\phi\,\underset{\sim}{\boldsymbol{e}}_1 + \sin\theta\sin\phi\,\underset{\sim}{\boldsymbol{e}}_2 + \cos\theta\,\underset{\sim}{\boldsymbol{e}}_3 \tag{13.6}$$

We want any point on the unit sphere to be equally likely — we seek a *uniform* distribution. Hence, the distribution function $p_{\underset{\sim}{\boldsymbol{n}}}(\underset{\sim}{\boldsymbol{n}})$ must equal the inverse of the *area* of the unit sphere: $p_{\underset{\sim}{\boldsymbol{n}}}(\underset{\sim}{\boldsymbol{n}}) = 1/4\pi$.

The expected value of a uniformly random unit vector is zero. The expected value of a randomly generated unit vector $\underset{\sim}{\boldsymbol{n}}$ is defined by

$$\overline{\underset{\sim}{\boldsymbol{n}}} \equiv \int_{A^{\text{sphere}}} \underset{\sim}{\boldsymbol{n}}\, p(\underset{\sim}{\boldsymbol{n}})\, dA \quad , \tag{13.7}$$

where $p(\underset{\sim}{\boldsymbol{n}})$ is the distribution function. For a uniform distribution, we have

$$p(\underset{\sim}{\boldsymbol{n}}) = \frac{1}{A^{\text{sphere}}} = \frac{1}{4\pi} \tag{13.8}$$

so

$$\overline{\underset{\sim}{\boldsymbol{n}}} \equiv \frac{1}{4\pi}\int_{A^{\text{sphere}}} \underset{\sim}{\boldsymbol{n}}\, dA \tag{13.9}$$

Here is a neat trick for evaluating this integral: just apply the divergence theorem to convert this area integral into a volume integral. The integrand becomes the gradient of 1, which is zero. Thus

$$\overline{\underset{\sim}{\boldsymbol{n}}} \equiv \underset{\sim}{\boldsymbol{0}}$$

The distributions of coordinates for uniformly random unit vectors.

Recall that the distribution density of a uniformly random unit vector $\underset{\sim}{\boldsymbol{n}}$ is simply $p_{\underset{\sim}{\boldsymbol{n}}}(\underset{\sim}{\boldsymbol{n}}) = 1/(4\pi)$. Now we seek to convert this distribution function into the individual distribution functions for the two spherical coordinates defining the unit vector as a point on a unit sphere.

Let Ω represent a patch of area on the unit sphere. Then the probability that the unit normal will lie within that patch is

$$\int_{\Omega} p_{\underset{\sim}{\boldsymbol{n}}}(\underset{\sim}{\boldsymbol{n}})\, dA = \frac{1}{4\pi}\int_{\Omega} dA \tag{13.10}$$

For numerical implementation, we need to convert the distribution $p_{\underset{\sim}{n}}(\underset{\sim}{n})$ into a joint distribution for θ and ϕ. The area element in Eq. (13.10) is $dA = \sin\theta \; d\theta \; d\phi$. Thus, the probability that a unit normal corresponds to a point on the unit sphere for which $\theta_1 < \theta < \theta_2$ and $\phi_1 < \phi < \phi_2$ is

$$\frac{1}{4\pi}\int_{\phi_1}^{\phi_2}\int_{\theta_1}^{\theta_2} \sin\theta \; d\theta \; d\phi \tag{13.11}$$

The distribution $p_\theta(\theta)$ for θ is determined by integrating $p_{\underset{\sim}{n}}(\underset{\sim}{n})$ over all possible values of ϕ (*i.e.*, for ϕ=0 to 2π). The function $p_\theta(\theta)$ is defined such that the probability that θ will lie in an interval between θ_1 and θ_2 is

$$\int_{\theta_1}^{\theta_2} p_\theta(\theta) \; d\theta = \frac{1}{4\pi}\int_0^{2\pi}\int_{\theta_1}^{\theta_2} \sin\theta \; d\theta \; d\phi = \frac{1}{2}\int_{\theta_1}^{\theta_2} \sin\theta \; d\theta \tag{13.12}$$

Similarly, noting that θ ranges from 0 to π, the distribution $p_\phi(\phi)$ for ϕ is defined so that the probability that a unit normal corresponds to $\phi_1 < \phi < \phi_2$ is

$$\int_{\phi_1}^{\phi_2} p_\phi(\phi) \; d\phi = \frac{1}{4\pi}\int_{\phi_{11}}^{\phi_2}\int_0^{\pi_2} \sin\theta \; d\theta \; d\phi = \frac{1}{2\pi}\int_{\phi_{11}}^{\phi_2} d\phi \tag{13.13}$$

The last two results show that

$$\boxed{p_\theta(\theta) = \frac{1}{2}\sin\theta} \qquad \text{and} \qquad \boxed{p_\phi(\phi) = \frac{1}{2\pi}} \tag{13.14}$$

we note that

$$p_{\underset{\sim}{n}}(\underset{\sim}{n}) \, dA = p_\theta(\theta) q_\phi(\phi) \, d\theta \, d\phi, \tag{13.15}$$

which verifies that θ and ϕ are *independent* random variables. The key results of this section are the two boxed equations. The distribution is uniform over the unit sphere, but the distribution for θ is *nonuniform*. The function is a sine, which expresses the fact that lines of constant θ on the unit sphere are longer near the equator than near the poles. Hence, a uniformly distributed population of points on the unit sphere will have more points near the equator than near the poles.

Uniformly random unit vectors — formalized implementation

Now we have the distributions $p(\theta)$ and $q(\phi)$ that correspond to a uniform distribution on the unit sphere, but we need a procedure for *sampling* a point on the unit sphere. Pseudo random number generators conventionally provide a random number uniformly distributed in the interval [0,1), but we need a sample that is governed by a different nonuniform distribution. This section summarizes the standard theory for converting a random number on [0,1) to a sample from any desired nonuniform distribution.

Requisite probability theory. Consider any nonuniform distribution $p(x)$ on the interval $[a,b)$. We seek a mapping function $x = f(r)$ such that x will have the desired $p(x)$ distribution whenever r has a uniform distribution on [0,1). We will further require the function f to have a correspondence such that $f(0) = a$ and $f(1) = b$. Let's first define the **cumulative distribution**:

$$\boxed{g_x(s) \equiv \int_{-\infty}^{s} p_x(x')\ dx'}. \tag{13.16}$$

The density function $p_x(\ )$ is nonnegative, so the cumulative distribution function $g_x(s)$ must be a nondecreasing function of s, and therefore it has an inverse except possibly at discrete points. In other words, g_x^{-1} may permissibly contain discrete bounded jumps.

The distribution function p is defined such that the probability that x will lie between x_1 and x_2 is

$$\int_{x_1}^{x_2} p_x(x')\ dx' = \int_{f(r_1)}^{f(r_2)} p_x(x')\ dx' = g_x(f(r_2)) - g_x(f(r_1)) \tag{13.17}$$

Since r is uniformly distributed on the interval [0,1), the probability that $r_1 < r < r_2$ is

$$\int_{r_1}^{r_2} dr = r_2 - r_1 \tag{13.18}$$

Because we are seeking an invertible mapping function, these last two probabilities must be equal:

$$g_x(f(r_2)) - g_x(f(r_1)) = r_2 - r_1 \tag{13.19}$$

This must hold for any values of r_1 and r_2. Let's take r_1=0 and r_2=r. Recall that we have required the point r=0 to map to the point x=a Therefore $f(0) = a$, and Eq. (13.19) implies

$$g_x(f(r)) - g_x(a) = r \tag{13.20}$$

Recalling that $p(x)=0$ for all $x<a$, we note that $g_x(a)=0$, and the above equation becomes

$$g_x(f(r)) = r, \tag{13.21}$$

or

$$\boxed{f(r) = g_x^{-1}(r)} \tag{13.22}$$

In summary, if r has a uniform distribution on [0,1), then $f(r)$ will have the distribution $p_x(x)$ on the interval [a,b).

Application to uniformly random unit normal. Consider the distribution of Eq. (13.14):

$$p_\theta(\theta) = \frac{1}{2}\sin\theta \tag{13.23}$$

This distribution is understood to be zero outside the interval $0<\theta<\pi$. Applying Eq. (13.16),

$$g_\theta(s) \equiv \int_0^s \frac{1}{2}\sin\theta \; d\theta = -\frac{1}{2}[\cos(s)-1] \tag{13.24}$$

Putting $g_\theta(s)=r$ and solving for s gives

$$g_\theta^{-1}(r) = \text{ArcCos}[1-2r] \tag{13.25}$$

Thus, applying Eq. (13.22), the mapping function is

$$f(r) = \text{ArcCos}[1-2r] \tag{13.26}$$

In the numerical implementation, this equation is applied by calling a pseudo random number generator to obtain a sample number r that is uniformly distributed on the interval [0,1). Eq. (13.26) converts r into the sample angle θ which is distributed on $[0,\pi)$ according to Eq. (13.14):

$$\boxed{\theta = \text{ArcCos}[1-2r]} \tag{13.27}$$

A similar analysis for ϕ gives

$$\boxed{\phi = 2\pi R}, \tag{13.28}$$

where R is another (pseudo) random number on the interval [0,1). Once the sample realizations of θ and ϕ are known, the uniformly random unit normal is constructed by applying Eq. (13.6):

$$\boxed{\underset{\sim}{\boldsymbol{n}} = \sin\theta\cos\phi\,\underset{\sim}{\boldsymbol{e}}_1 + \sin\theta\sin\phi\,\underset{\sim}{\boldsymbol{e}}_2 + \cos\theta\,\underset{\sim}{\boldsymbol{e}}_3} \tag{13.29}$$

These boxed equations are implemented in **Listing 6 (Generating a uniformly random unit normal) on page A-7**.

Uniformly random unit vectors — faster implementation

The techniques outlined in the preceding section followed classical methods of probability theory. Specifically, a uniformly random unit vector was generated by first sampling two random numbers R and r, each uniformly distributed on the interval [0,1). Then Eqs. (13.27) through (13.29) were applied to construct the uniformly random unit vector. Unfortunately, numerical implementations of that procedure requires calling trigonometric intrinsic functions, which can be expensive.

In this section we describe a much more straightforward way to generate a uniformly random unit vector, and this method is generally faster than the method of the preceding section so long as the computer's pseudo random number generator is well-optimized. The idea behind the alternative algorithm is to first generate a (non-unit) vector $\underset{\sim}{r}$ using a uniform distribution inside the unit sphere. Then this vector is simply normalized to a unit vector $\underset{\sim}{\boldsymbol{n}}$ by applying the formula

$$\underset{\sim}{\boldsymbol{n}} = \frac{\underset{\sim}{\boldsymbol{r}}}{\sqrt{\underset{\sim}{\boldsymbol{r}} \bullet \underset{\sim}{\boldsymbol{r}}}} \tag{13.30}$$

Generating the vector $\underset{\sim}{\boldsymbol{r}}$ from a uniformly distributed sample space inside the unit sphere is algorithmically trivial. First, three pseudo random numbers, $\{r_1, r_2, r_3\}$ on the interval [0,1) are generated. Then, a scalar $\mu = r_1^2 + r_2^2 + r_3^2$ is computed. If $\mu > 1$, the three random numbers must be discarded and a new set of three random numbers is generated. This process continues until a set of three random numbers is generated for which $\mu \le 1$, in which case, the unit vector $\underset{\sim}{\boldsymbol{n}}$ is then given by Eq. (13.30).

Uniformly random unit vectors —The visualization

Visualizing unit vectors (stereographic projection). A unit vector $\underset{\sim}{\boldsymbol{n}}$ may be regarded as a point on the unit sphere. For visualization, points on the unit sphere must be mapped to the plane (this sheet of paper). Any such mapping is called a "stereographic projection." The most common projection is what you would get if you were to take a photograph of a sphere. The surface of the sphere maps to the planar surface of the photo. Two photos of the sphere (showing the front and the rear) would be needed for this type of stereographic projection. There are numerous alternative ways to define the projection, and the best choice depends on the application. We will see that an "area preserving" projection is the most natural choice for visualizing statistical distributions (or any other quantity defined "per-area" on the sphere).

Before discussing the "sphere-plane" problem, first consider the simpler circle-line problem of visualizing points on a simple circle defined by the equation $x^2 + y^2 = R^2$. This circle is a one-dimensional entity existing in two dimensions. As sketched in Fig. 13.1, we can "snip" the circle at the point $(0, -1)$ and then "unwrap" or "flatten" it into a line. This is an easy mapping to visualize because both the circle and the line are Euclidean spaces (both the circle and the line can be formed by a strip of paper).

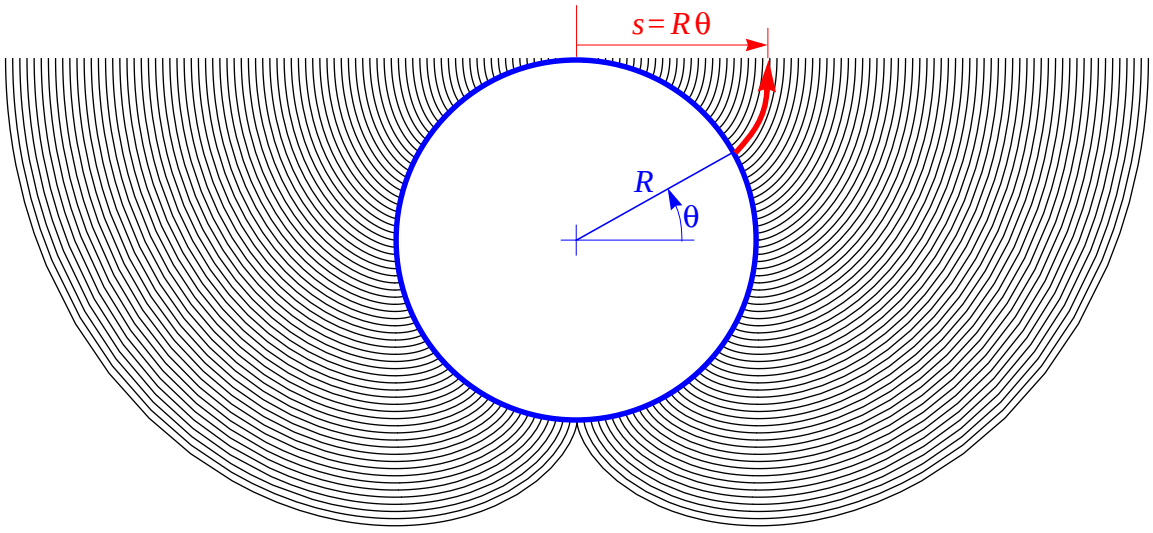


Figure 13.1. Length-preserving mapping of a circle to a one-dimensional straight line. The polar coordinate θ maps to a linear coordinate s, which is just the arc length. Note that the lower hemisphere maps to a discontinuous region defined by $|s| > R\pi/2$. The mapping is one-to-one everywhere except that the point $(0, -1)$ maps to both $s=1$ and $s=-1$.

In crystallography, a different kind of stereographic projection (shown in Fig. 13.2) is commonly used. The disadvantage of this kind of projection is that arc length is no preserved. Thus, if a classical stereographic projection is used, a distribution that is uniform on the circle will map to a distribution that is non-uniform on the line.

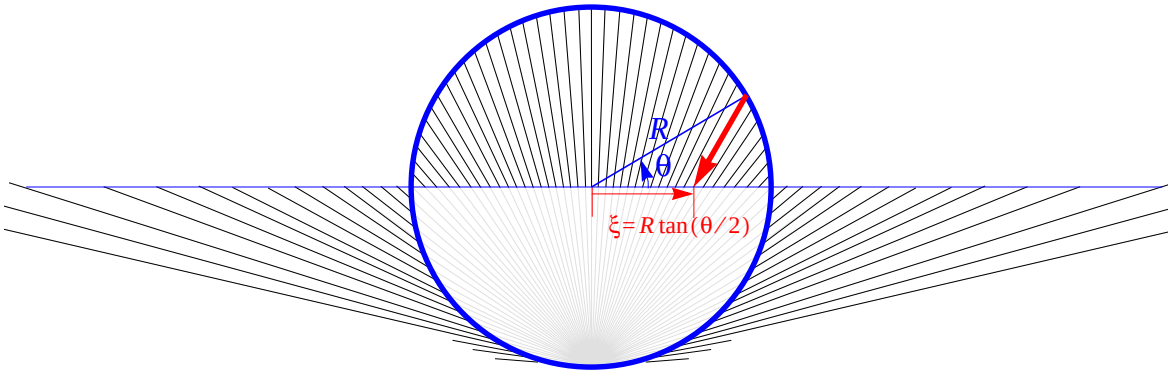


Figure 13.2. Classical stereographic mapping of a circle to a one-dimensional straight line. The polar coordinate θ maps to a linear coordinate ξ, which is defined by where a line from the south pole to the point intersects the equator plane. Note that the upper semicircle maps to points for which $\xi < R$ and points on the lower semicircle map to points for which $\xi > R$.

Now consider the much harder sphere-plane problem. Mapping the surface of a *sphere* to the plane defined by this sheet of paper must involve area distortion since the surface of a sphere is non-Euclidean. In other words, an area element on the sphere will map to an area element on the plane, but the two elements will generally have different aspect ratios. Imagine introducing a tiny hole at the "south pole" of a racquetball. In order to bend the racquetball into the plane, the area elements that were originally near the south pole would have to be significantly distorted. Unlike the circle-line mapping of Fig. 13.1 (which involved zero change in arc length), the sphere-plane mapping necessarily involves aspect ratio distortion. However, as we soon show, it is possible to define a stereographic projection such that the magnitude of the area is preserved despite the fact that the *shape* of the area element necessarily changes.

The left side of Fig. 13.3 shows the conventional (photographic) rendering of the sphere as seen when looking down the $\{1, 1, 1\}$ axis passing through the first octant. In such an illustration, the sphere "looks like a sphere," which is aesthetically appealing in most situations. However, when illustrating statistical distributions on a sphere, it is more useful to employ an *area-preserving* mapping of the sphere to the two dimensional plane. In this way, a

probability function defined on the surface of the sphere will have the same value when mapped to the 2D plane. With equal-area mapping, a uniform distribution on the sphere will map to a uniform distribution on the plane — any clustering in the 2D map will correspond to real clusters on the sphere.

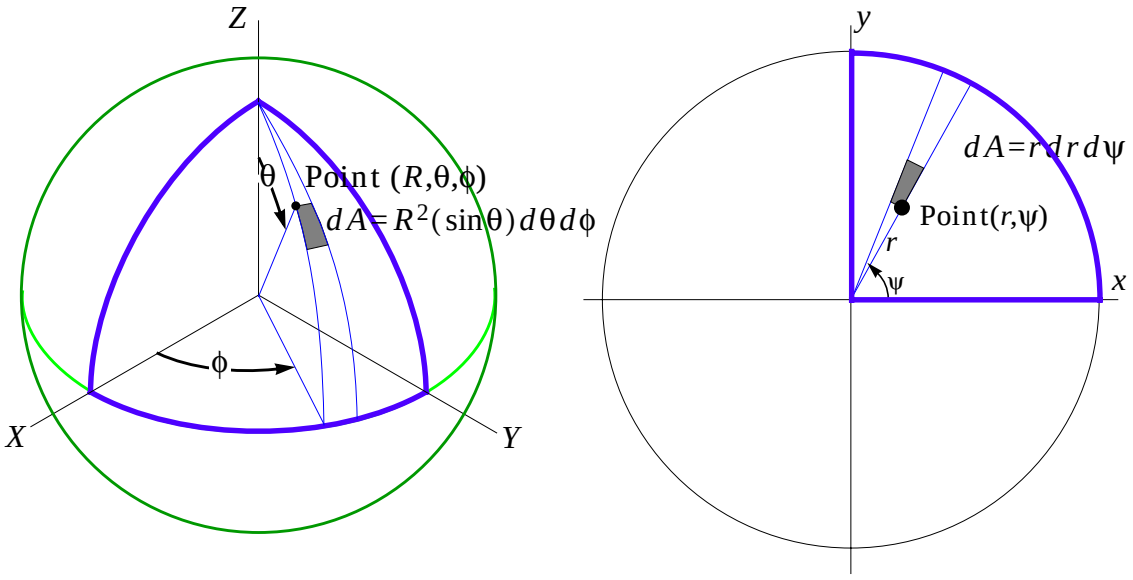


Figure 13.3. Mapping points on a sphere of radius R to a two-dimensional disk. The 2D disk shown on the right-hand side of the above figure may be regarded as a distorted view of the sphere as seen looking down the Z-axis.

We are ultimately interested in mapping a *unit* sphere to the plane, but let's derive the equations for a sphere of radius R, and then later set $R=1$. Equating the area elements shown in Fig. 13.3 gives

$$R^2(\sin\theta)d\theta d\phi = r dr d\psi \tag{13.31}$$

For our mapping, we *choose* $\psi = \phi$, so that Eq. (13.31) becomes an ordinary differential equation,

$$R^2(\sin\theta)d\theta = r dr \ , \tag{13.32}$$

which is easily solved (subject to $r=0$ when $\theta=0$) to give r as a function of θ. Specifically,

Equal area mapping: If a point on a sphere has spherical coordinates $\{R, \theta, \phi\}$, then the (unit Jacobian) mapped point on the 2D disk has *polar* coordinates

$$\boxed{\psi = \phi} \text{ and } \boxed{r \equiv R\sqrt{2(1-\cos\theta)}} \tag{13.33}$$

As shown in Fig. 13.4, a sample point on the equator of the sphere (*i.e.*, a point for which $\theta=\pi/2$) will map to a radius $r=\sqrt{2}R$. All points in the upper hemisphere have mapped radial coordinates smaller than $\sqrt{2}R$. As seen by the graph in Fig. 13.4, points in the upper hemisphere have a mapped

radial coordinate r that is *roughly* equal to the spherical arc length $R\theta$. The equal area mapping is well-defined for the larger values of θ in the lower hemisphere (shown in grey in Fig. 13.4). The south pole ($\theta=\pi$) maps to the outer boundary of the disk where $r=2R$. To preserve area, points mapped from the lower hemisphere require a greater amount of distortion (*i.e.*, Δr is significantly different from $R\Delta\theta$).

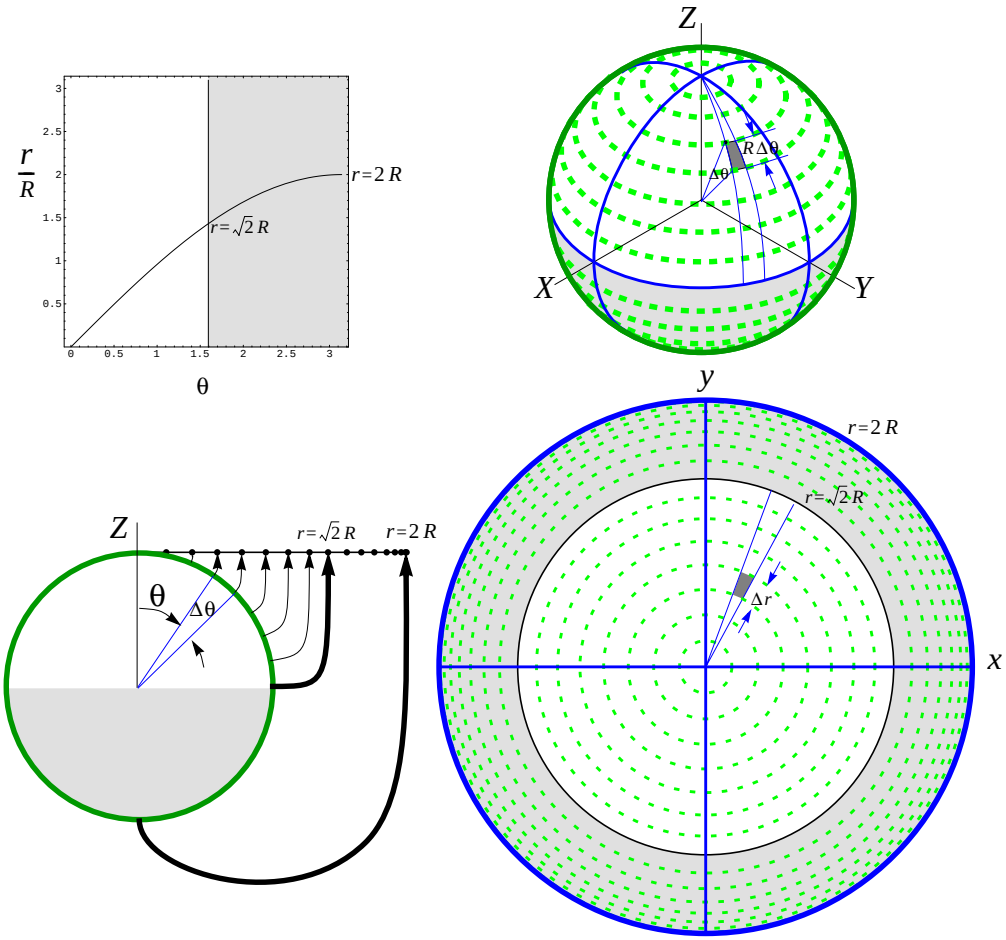


Figure 13.4. Equal area mapping. The upper-left graph of r vs. θ shows that $r \approx R\theta$ for points in the upper-hemisphere. The lower-left plot shows a side-view of the sphere to illustrate how the θ coordinate maps to the disk radial coordinate r. The upper-right sketch shows a conventional "3D" rendering of the sphere as seen down the $\{1,1,1\}$ axis. The lower-right plot is the equal area mapping of the sphere as seen down the Z-axis. The relative sizes of the plots are shown to correct scale (i.e, the disk has twice the radius of the sphere.

Equal area mapping applied to uniformly distributed unit vectors. Figure 13.5*(a)* shows a plot of uniformly random unit vectors displayed using the equal area stereographic projection. Because the mapping preserves the relative magnitudes of area elements, a uniform distribution of points on the sphere corresponds to a uniform distribution of points on the stereographic projection disk.

By contrast, Fig. 13.5*(b)* shows that merely normalizing a vector sampled uniformly from the unit *cube* results in a *nonuniform* distribution of normalized vectors on the sphere — the clustering reflects over-sampling the corners of the cube.

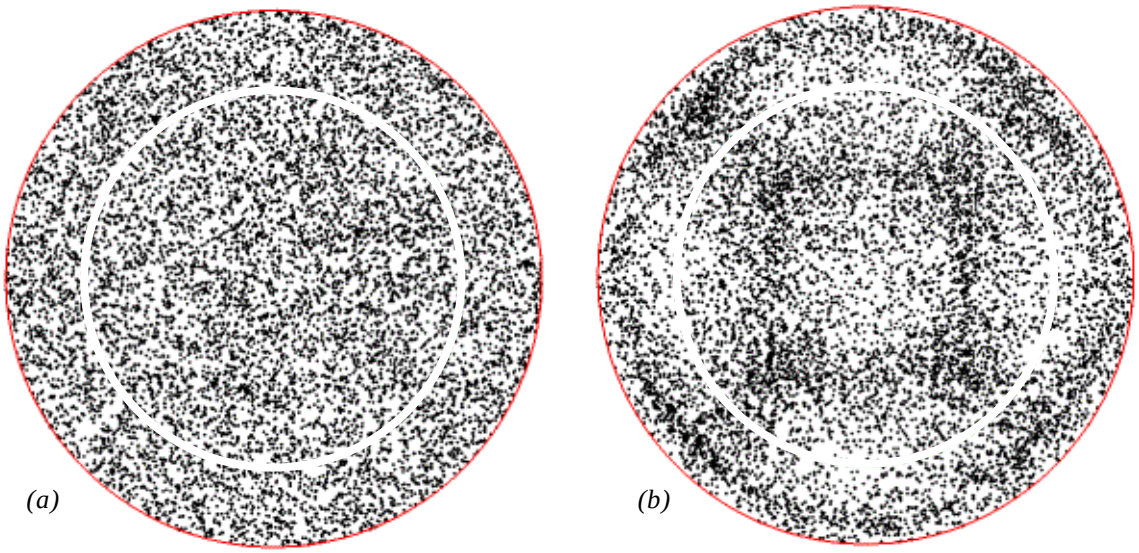


Figure 13.5. Stereographic projections of random unit vectors. (a) a *uniformly* random unit vector distribution found by normalizing a uniform sampling of points inside the sphere. (b) a *non-uniform* set of unit vectors resulting from normalizing a uniform sampling of points inside a unit *cube.*

Uniformly random rotations

In this section, we discuss several methods for generating a uniformly random rotation. First we need a definition of uniformity. Let $\underset{\sim}{N}$ denote an arbitrary unit **reference vector**. The **sample vector** $\underset{\sim}{n}$ generated from a random rotation $\underset{\approx}{R}$ is defined

$$\underset{\sim}{n} \equiv \underset{\approx}{R} \bullet \underset{\sim}{N} \tag{13.34}$$

The rotation tensor $\underset{\approx}{R}$ is uniformly distributed if and only if $\underset{\sim}{n}$ is uniformly distributed for any reference direction $\underset{\sim}{N}$. Equivalently [3], a rotation tensor $\underset{\approx}{R}$ is uniformly distributed if and only if $\underset{\approx}{Q} \bullet \underset{\approx}{R}$ and $\underset{\approx}{R} \bullet \underset{\approx}{Q}$ have the same distribution as $\underset{\approx}{R}$ for any fixed rotation $\underset{\approx}{Q}$.

In upcoming sections, we first demonstrate that a uniform rotation can *not* be obtained by using a uniform angle and axis of rotation. We will find that uniform rotations require the distribution for the angle to be biased toward $180°$.

Next we show that uniform Euler angles does not give a uniform rotation either. To obtain a uniform rotation using Euler angles, the distribution for the second Euler angle must be biased toward $90°$.

We will present the computationally simplest algorithm we've found for generating a uniform rotation. Namely, we generate two uniformly random unit vectors, use Gram-Schmidt technique to orthonormalize them. A third unit vector is formed by a right-hand cross product of the orthonormal pair, and the rotation tensor is constructed by placing the three orthonormalized vectors into columns of the rotation tensor. The principal unsavory feature of this algorithm is that we have yet to find a *proof* that it results in a uniform rotation. As engineers, however, we have tested this algorithm by generating tens of thousands of instances rotations with this method, and we have verified by computing the resulting axis and angle that the distribution does indeed agree exactly with the known distribution required to obtain a uniform rotation.

After giving the simple algorithm, we present some other more computationally awkward — but rigorously proved — algorithms for computing uniform rotations.

Finally, we finish up this section by showing how to average a tensor over all possible uniform orientations.

Inadequacy of using uniformly random angle and axis. A uniformly random rotation tensor must have an expected value of $\underset{\approx}{\boldsymbol{0}}$ so that the rotated sample vector $\underset{\sim}{\boldsymbol{n}}$ is equally likely to point in any direction. As discussed by Shoemake [3], one would *think* that a uniformly random rotation axis with a uniformly random rotation angle would also correspond to a uniformly random rotation tensor (computed via Eq. 3.14). However, as shown in the numerical experiment of Fig. 13.6, such a scheme make the rotated sample vectors cluster about their original orientation, and the rotation is therefore not uniform. As a matter of fact, the expected value of the rotation turns out to be $\underset{\approx}{\boldsymbol{I}}/6\pi$, which is not zero as required for a uniform rotation.

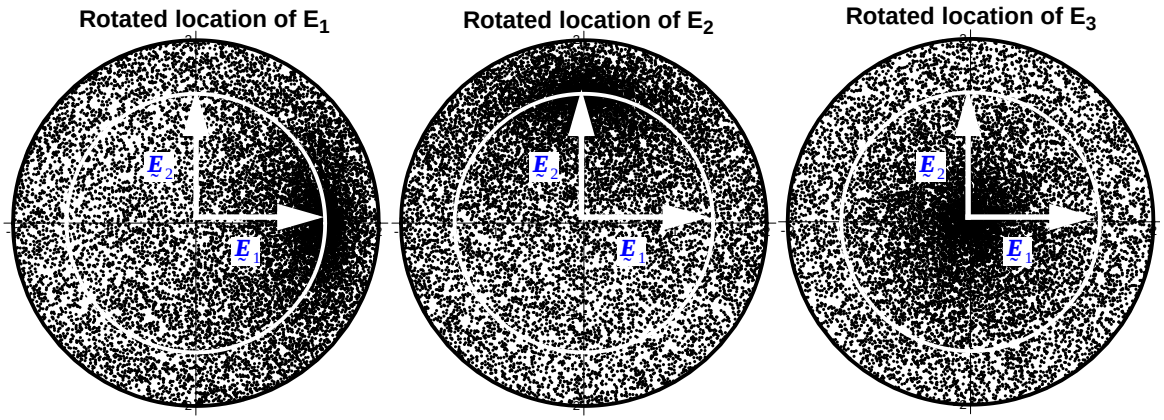


Figure 13.6. Uniform axis and angle do not lead to uniform rotation. Roughly 10^4 random rotations were generated using uniform axis and angle. For each sample rotation, the rotated base vector, $\underset{\approx}{\boldsymbol{R}} \bullet \underset{\sim}{\boldsymbol{E}}_k$, was computed and (regarding it as a point on the unit sphere) mapped to the plane via equal-area mapping. The white circle demarks the boundary between the upper and lower hemispheres [See Fig. 13.4]. The above dot plots show that the rotated positions of the three base vectors tend to cluster about their original orientations.

To create Fig. 13.6, we generated roughly 10^4 random rotations by taking the axis $\underset{\sim}{\boldsymbol{a}}$ and angle α to be uniform. The rotation tensor itself was then constructed by applying the Euler-Rodrigues formula of Eq. (3.14):

$$\underset{\approx}{\boldsymbol{R}} = \cos\alpha\ (\underset{\approx}{\boldsymbol{I}} - \underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}}) + \underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}} + \sin\alpha\ \underset{\approx}{\boldsymbol{A}} \tag{13.35}$$

The sample points plotted in Fig. 13.6 show the rotated locations $\underset{\approx}{\boldsymbol{R}} \bullet \underset{\sim}{\boldsymbol{E}}_1$, $\underset{\approx}{\boldsymbol{R}} \bullet \underset{\sim}{\boldsymbol{E}}_2$, and $\underset{\approx}{\boldsymbol{R}} \bullet \underset{\sim}{\boldsymbol{E}}_3$, respectively as seen using equal area projection of the sphere when viewed down the $\underset{\sim}{\boldsymbol{E}}_3$ base vector (see Eq. 13.33). In all cases, the rotated location of $\underset{\sim}{\boldsymbol{E}}_k$ tends to cluster about $\underset{\sim}{\boldsymbol{E}}_k$ itself!

To better understand why a uniform rotation must not have a uniform rotation angle, consider any reference vector $\underset{\sim}{N}$. If the rotation axis is uniformly distributed, then a significant fraction* of the rotation tensors will have a rotation axis that forms an angle with $\underset{\sim}{N}$ of less than 45°. As indicated in Fig. 13.7, these "polar" axes will *always* produce a sample vector $\underset{\approx}{R} \bullet \underset{\sim}{N}$ that lies in the upper hemisphere closest to $\underset{\sim}{N}$. These polar axes are incapable of producing rotated sample points in the lower hemisphere.

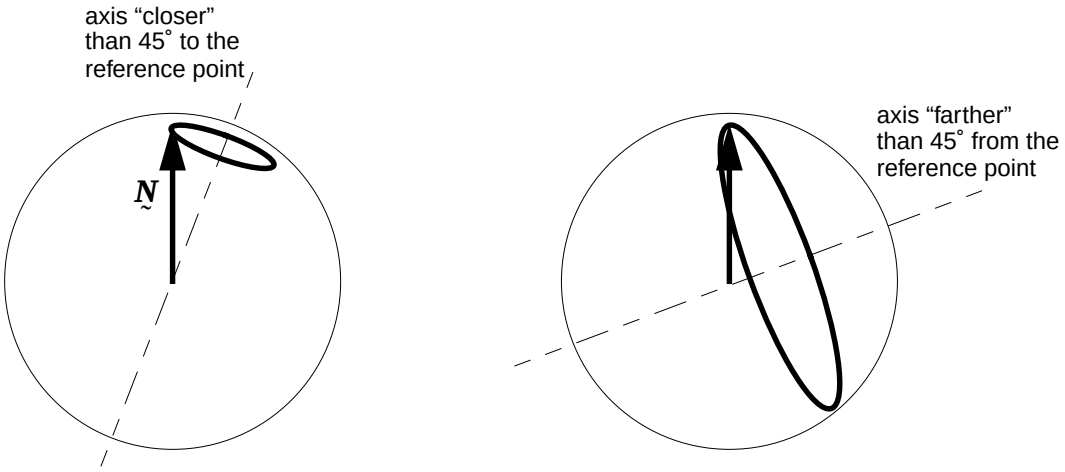


Figure 13.7. Qualitative reason why a uniform rotation must have a non-uniform rotation angle. Suppose that the rotation axis forms an angle smaller than 45° with the reference vector $\underset{\sim}{N}$. Then regardless of the distribution for the rotation angle, the rotated sample vector $\underset{\approx}{R} \bullet \underset{\sim}{N}$ will lie in the upper hemisphere closest to $\underset{\sim}{N}$. Only axes farther than 45° away from the reference vector even have the capability of producing sample points in the lower hemisphere. Consequently, the rotation angle must be biased towards 180° to allow the equatorial axes to generate enough sample points in the lower hemisphere to balance those in the upper hemisphere.

Only "equatorial" rotation axes that lie farther than 45° from $\underset{\sim}{N}$ can *possibly* generate points in the lower hemisphere. Biasing the rotation angle α toward 180° allows the equatorial axes to produce a compensating number of sample points in the lower hemisphere. We will later show that the rotation angle α has a probability density function given by

$$p_\alpha(\alpha) = \frac{1}{\pi}\sin^2\frac{\alpha}{2} \text{ for rotation angles ranging from } -\pi \text{ to } \pi.$$
$$p_\alpha(\alpha) = \frac{2}{\pi}\sin^2\frac{\alpha}{2} \text{ for rotation angles ranging from } 0 \text{ to } \pi. \qquad (13.36)$$

which is minimum at 0° and maximum at 180°.

* roughly 21% ($=1-1/\sqrt{2}$)

Inadequacy of using uniform Euler angles. As mentioned in the preceding section, a uniformly random rotation tensor must have an expected value of zero. This is a necessary, but not sufficient condition for uniformity. Suppose that you generate three uniformly random Euler angles $\{\varphi, \vartheta, \psi\}$ and then construct the rotation tensor by applying Eq. (7.4),

$$[R_{ij}] = \begin{bmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\vartheta & -\sin\vartheta \\ 0 & \sin\vartheta & \cos\vartheta \end{bmatrix} \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{13.37}$$

We claim that the resulting rotation tensor will not be *uniformly* random.

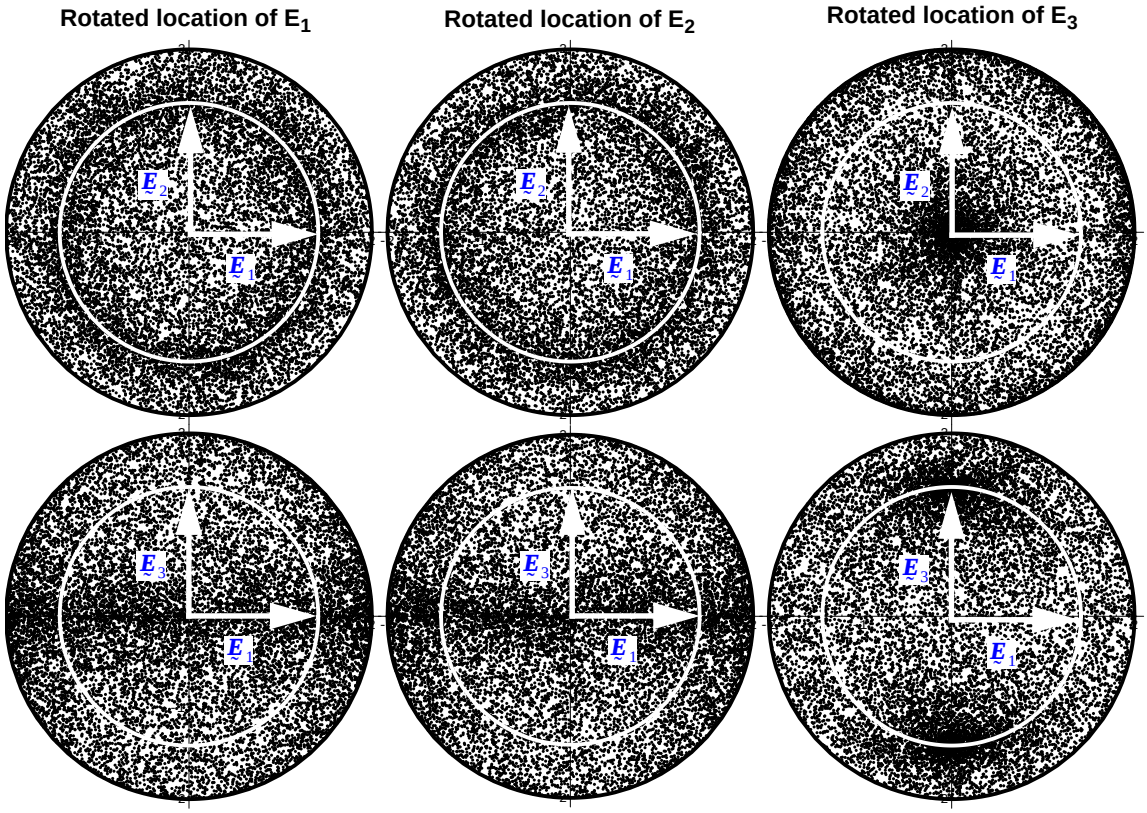


Figure 13.8. Uniform Euler angles produce non-uniform rotations. The top three views show equal area projections of the rotated vectors $\underset{\approx}{\boldsymbol{R}} \cdot \underset{\sim}{\boldsymbol{E}}_1$, $\underset{\approx}{\boldsymbol{R}} \cdot \underset{\sim}{\boldsymbol{E}}_2$, and $\underset{\approx}{\boldsymbol{R}} \cdot \underset{\sim}{\boldsymbol{E}}_3$ respectively as seen when viewing the sphere down the $\underset{\sim}{\boldsymbol{E}}_3$ axis. The bottom three views show the same rotated base vectors as seen when viewing down the $\underset{\sim}{\boldsymbol{E}}_2$ axis.

A rotation is deemed uniformly random if the distribution for $\underset{\approx}{\boldsymbol{R}} \cdot \underset{\sim}{\boldsymbol{N}}$ is uniform for any unit vector $\underset{\sim}{\boldsymbol{N}}$. The sample points plotted in Fig. 13.8 show the rotated locations $\underset{\approx}{\boldsymbol{R}} \cdot \underset{\sim}{\boldsymbol{E}}_1$, $\underset{\approx}{\boldsymbol{R}} \cdot \underset{\sim}{\boldsymbol{E}}_2$, and $\underset{\approx}{\boldsymbol{R}} \cdot \underset{\sim}{\boldsymbol{E}}_3$ corresponding to 10^4 sample points generated using uniform Euler angles. In this case, the expected value

of the rotated vectors (and therefore the expected value of the rotation tensor) are zero. This is a necessary, but not sufficient condition for uniformity. The uniform-Euler-angle scheme still produces clustering. The two upper-left plots in Fig. 13.8 show that the rotated locations of $\underset{\sim}{\boldsymbol{E}}_1$ and $\underset{\sim}{\boldsymbol{E}}_2$ could visually *appear* to be uniform if only the points corresponding to the upper hemisphere were plotted. Only seemingly minor clustering at the equator would be detectable.* The clustering is more clearly visible when viewing the unit sphere down the $\underset{\sim}{\boldsymbol{E}}_2$ axis (lower-left two plots in Fig. 13.8). The rotated vector $\underset{\approx}{\boldsymbol{R}} \bullet \underset{\sim}{\boldsymbol{E}}_3$ tends to cluster about $\pm\underset{\sim}{\boldsymbol{E}}_3$.

Without proof, we claim that the *correct* distributions for the Euler angles needed to produce a truly uniform rotation are

$$\begin{aligned} p_\varphi(\varphi) &= \frac{1}{2\pi} \text{ on } -\pi < \varphi < \pi \\ p_\vartheta(\vartheta) &= \frac{1}{2}\sin\vartheta \text{ on } 0 < \vartheta < \pi \\ p_\psi(\psi) &= \frac{1}{2\pi} \text{ on } -\pi < \psi < \pi \end{aligned} \tag{13.38}$$

Note that the distributions for φ and ψ are uniform, but the distribution for ϑ must be biased toward values of $90°$.

* This clustering could be easily missed if any other (non-area-preserving) form of stereographic projection were used.

An easy algorithm for generating a uniformly random rotation.

Any rotation tensor $\underset{\approx}{\boldsymbol{R}}$ can be defined by how it transforms the three laboratory base vectors, $\{\underset{\sim}{\boldsymbol{E}}_1, \underset{\sim}{\boldsymbol{E}}_2, \underset{\sim}{\boldsymbol{E}}_3\}$. Upon a rigid rotation, this triad becomes $\{\underset{\sim}{\boldsymbol{e}}_1, \underset{\sim}{\boldsymbol{e}}_2, \underset{\sim}{\boldsymbol{e}}_3\}$. Then the rotation tensor is $\underset{\approx}{\boldsymbol{R}} = \underset{\sim}{\boldsymbol{e}}_1\underset{\sim}{\boldsymbol{E}}_1 + \underset{\sim}{\boldsymbol{e}}_2\underset{\sim}{\boldsymbol{E}}_2 + \underset{\sim}{\boldsymbol{e}}_3\underset{\sim}{\boldsymbol{E}}_3$, meaning that the columns of the component matrix are the rotated base vectors:

$$[\underset{\approx}{\boldsymbol{R}}] = [\{\underset{\sim}{\boldsymbol{e}}_1\}\{\underset{\sim}{\boldsymbol{e}}_2\}\{\underset{\sim}{\boldsymbol{e}}_3\}] \tag{13.39}$$

To generate a uniformly random rotation tensor $\underset{\approx}{\boldsymbol{R}}$, we set $\underset{\sim}{\boldsymbol{e}}_1$ to be a uniformly random unit vector [constructed via Eqs. (13.27), (13.28), and (13.29)]. We then construct a second uniformly random vector $\underset{\sim}{\boldsymbol{m}}$ and subtract away its part in the direction of $\underset{\sim}{\boldsymbol{e}}_1$. In other words, we compute $\underset{\sim}{\boldsymbol{e}}_2$ as follows:

$$\underset{\sim}{\boldsymbol{e}}_2 = \underset{\sim}{\mu}/\|\underset{\sim}{\mu}\|, \quad \text{where} \quad \underset{\sim}{\mu} = \underset{\sim}{\boldsymbol{m}} - (\underset{\sim}{\boldsymbol{m}} \bullet \underset{\sim}{\boldsymbol{e}}_1)\underset{\sim}{\boldsymbol{e}}_1 \tag{13.40}$$

The third rotated unit vector is given by the right-hand cross product:

$$\underset{\sim}{\boldsymbol{e}}_3 = \underset{\sim}{\boldsymbol{e}}_1 \times \underset{\sim}{\boldsymbol{e}}_2 \tag{13.41}$$

With the three rotated base vectors known, the uniformly random rotation $\underset{\approx}{\boldsymbol{R}}$ is given by Eq. (13.39). The numerical implementation of this technique is given in ***Listing 7 (Generating a uniformly random rigid rotation.) on page A-8***. Numerical experimentation (below) indicates that the above algorithm does yield a uniformly random rotation, but we still seek a *proof*.

Numerical verification. To check the above scheme, we wrote a test code to generate 10^4 instances of random rotations. Figure 13.9 shows dot plots of the uniformly random locations of the $\underset{\approx}{\boldsymbol{R}} \bullet \underset{\sim}{\boldsymbol{E}}_k$ vectors.

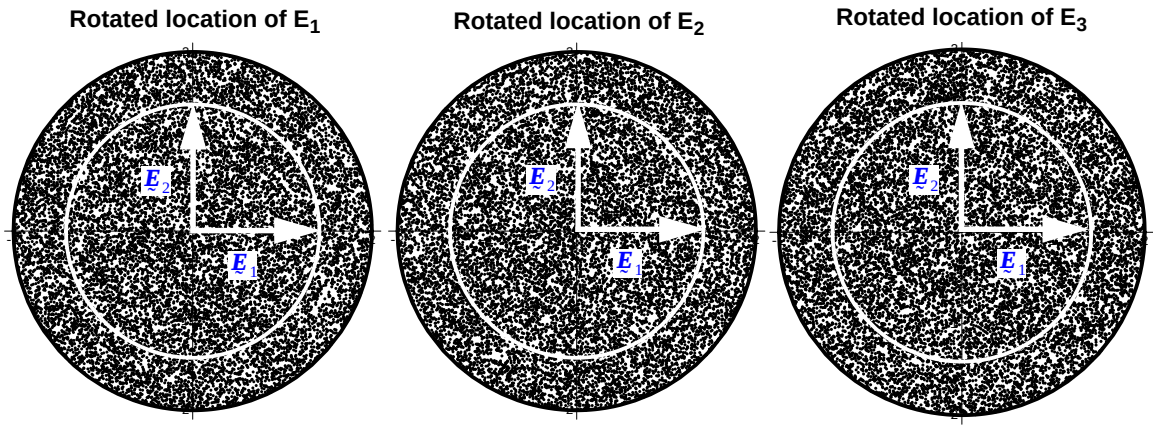


Figure 13.9. Uniformly random locations of the three rotated base vectors. Each dot represents one sample point on the unit sphere, projected to the circle via equal area mapping (Hence, because the dots appear to be distributed without bias in this picture, then they would also appear evenly distributed on the unit sphere.)

Fig. 13.9 shows an equal area projection of the unit sphere, as defined on page 13.33. This mapping ensures that a uniform distribution on the sphere will correspond to a uniform distribution of dots in our 2D visualization. Figure 13.9 may be *qualitatively* interpreted as views of the sphere looking down the x_3 axis. On all three plots, the $\underset{\sim}{\boldsymbol{E}}_1$ direction points to the right, the $\underset{\sim}{\boldsymbol{E}}_2$ direction points up, and the $\underset{\sim}{\boldsymbol{E}}_3$ direction points out of the page. The three dot plots show the rotated vectors $\underset{\sim}{\boldsymbol{e}}_1$, $\underset{\sim}{\boldsymbol{e}}_2$, and $\underset{\sim}{\boldsymbol{e}}_3$, respectively as points on the unit sphere. The fact that all three dot plots show a uniform distribution of points heuristically demonstrates that the algorithm does indeed produce a uniform rotation.

Given that the rotation tensor is generated by the uniform distribution algorithm on page 105, we should be able to *infer* the distributions for the rotation axis and angle. Using the algorithm on page 105 to generate 10^4 uniformly random rotations (as in Fig. 13.9), we called the routine DC2AA (on page 3). In this way, we "experimentally" measured the distributions for axis and angle that are shown in Figure 13.10.

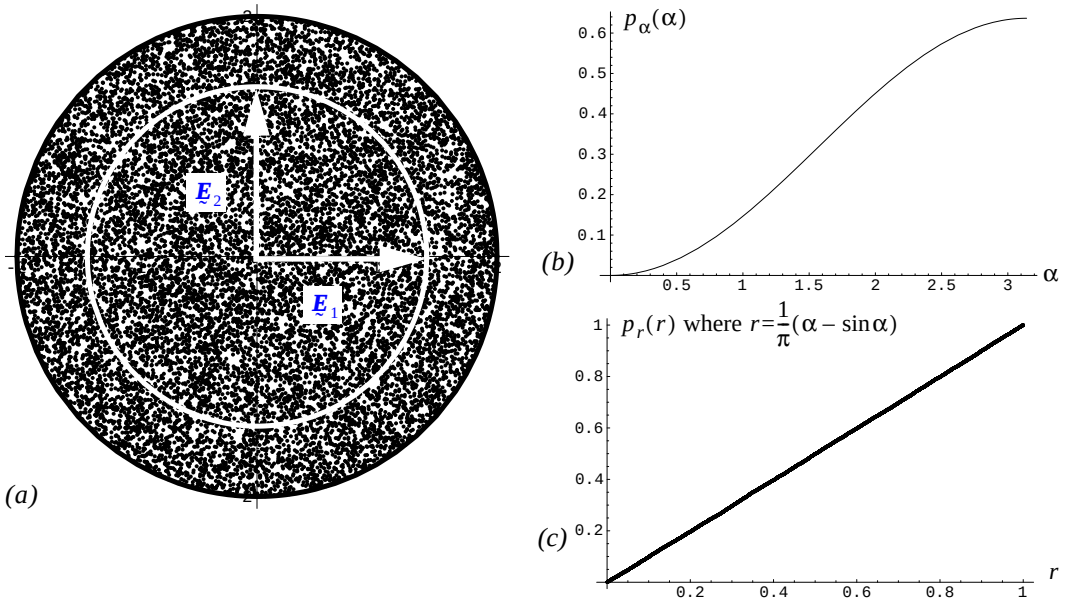


Figure 13.10. Numerical measurement of rotation axis and angle. The left side *(a)* shows a dot plot of the rotation axis. The graph *(b)* is the cumulative distribution for the rotation angle. The graph *(c)* is the cumulative distribution for the variable $r = (\alpha - \sin\alpha)/\pi$.

Figure 13.10 suggests that the rotation axis is distributed uniformly. The rotation angle α clearly is not uniform. As previously explained, the rotation angle must be biased toward values of 180°. We later prove that a uniform rotation must have a distribution for the rotation angle given by

$$p_\alpha(\alpha) = \frac{1}{\pi}\sin^2\frac{\alpha}{2} \text{ for } \alpha \in [-\pi, \pi], \quad (13.42)$$

which is plotted in Fig. 13.11. Note that the rotation angle is weighted toward ±180°, as expected.

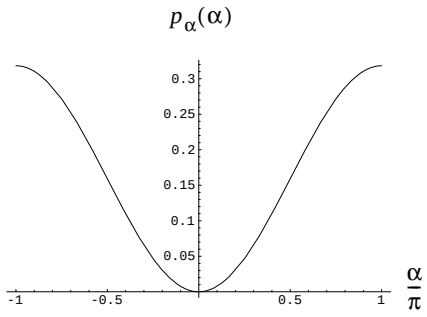


Figure 13.11. Distribution for the angle of rotation.

Without loss in generality, the rotation angle may be alternatively taken to be always positive (as is presumed in the routine DC2AA), in which case the distribution is

$$p_\alpha(\alpha) = \frac{2}{\pi}\sin^2\frac{\alpha}{2} \text{ for } \alpha \in [0, \pi] \quad (13.43)$$

This means that the *cumulative distribution function* must be given by

$$g_\alpha(\alpha) = \int_0^\alpha p_\alpha(\alpha')\,d\alpha' = \frac{1}{\pi}(\alpha - \sin\alpha) \quad (13.44)$$

Consequently, the quantity $(\alpha - \sin\alpha)/\pi$ must be uniformly distributed on the interval from 0 to 1, which is verified in the numerical experiment of Fig. 13.10. This numerical observation further confirms (without proof) that the algorithm on page 105 does indeed correspond to a uniform rotation.

Since we know that

$$r = (\alpha - \sin\alpha)/\pi \quad (13.45)$$

is uniformly distributed on the interval from 0 to 1, then we know we can generate a representative sampling of the rotation angle by generating a number r that is uniformly random on the interval from 0 to 1, and then the rotation angle α would then be obtained by solving Eq. (13.45) for α.

An alternative algorithm for generating a uniformly random rotation.

The above method for generating a uniformly random rotation simply required applying Gram-Schmidt orthogonalization to two uniformly random unit vectors, with the third unit vector being given by a cross product. The resulting right-handed orthogonal set of unit vectors was then assembled to form columns of a 3×3 rotation matrix. Algorithmically this is a trivial procedure so long as a function is already available to generate a uniformly random unit vector. The only (minor) numerical issue is that the second vector must be re-generated if it is parallel (or nearly parallel) to the first vector.

An alternative algorithm begins in the same manner by generating a uniformly random unit vector $\underset{\sim}{\boldsymbol{n}}_1$. If the k^{th} laboratory component of $\underset{\sim}{\boldsymbol{n}}_1$ is the smallest (or tied for smallest), then we know that the lab base vector $\underset{\sim}{\boldsymbol{E}}_k$ will be linearly independent of $\underset{\sim}{\boldsymbol{n}}_1$ (i.e, $\underset{\sim}{\boldsymbol{n}}_1 \bullet \underset{\sim}{\boldsymbol{E}}_k \neq 0$). Consequently, we can construct a vector $\underset{\sim}{\boldsymbol{m}}$ that is perpendicular to $\underset{\sim}{\boldsymbol{n}}_1$ by the formula

$$\underset{\sim}{\boldsymbol{m}} = \frac{\underset{\sim}{\boldsymbol{E}}_k - \underset{\sim}{\boldsymbol{n}}_1(\underset{\sim}{\boldsymbol{E}}_k \bullet \underset{\sim}{\boldsymbol{n}}_1)}{\left\|\underset{\sim}{\boldsymbol{E}}_k - \underset{\sim}{\boldsymbol{n}}_1(\underset{\sim}{\boldsymbol{E}}_k \bullet \underset{\sim}{\boldsymbol{n}}_1)\right\|} \tag{13.46}$$

A third unit vector can be created by

$$\underset{\sim}{\boldsymbol{t}} = \underset{\sim}{\boldsymbol{n}}_1 \times \underset{\sim}{\boldsymbol{m}} \tag{13.47}$$

The two unit vectors $\underset{\sim}{\boldsymbol{m}}$ and $\underset{\sim}{\boldsymbol{t}}$ form a plane perpendicular to $\underset{\sim}{\boldsymbol{n}}_1$. If R is a random number on the interval [0,1), then the vector

$$\underset{\sim}{\boldsymbol{n}}_2 = \cos(2\pi R)\underset{\sim}{\boldsymbol{m}} + \sin(2\pi R)\underset{\sim}{\boldsymbol{t}} \tag{13.48}$$

is uniformly random in the plane perpendicular to $\underset{\sim}{\boldsymbol{n}}_1$. The uniformly random rotation tensor would then be constructed by

$$[\underset{\approx}{\boldsymbol{R}}] = [\{\underset{\sim}{\boldsymbol{n}}_1\}\{\underset{\sim}{\boldsymbol{n}}_2\}\{\underset{\sim}{\boldsymbol{n}}_3\}], \text{ where } \underset{\sim}{\boldsymbol{n}}_3 = \underset{\sim}{\boldsymbol{n}}_1 \times \underset{\sim}{\boldsymbol{n}}_2 \tag{13.49}$$

We think that the previous algorithm was much simpler, both conceptually and numerically.

Shoemake's algorithm for uniformly random rotations.

Recall that a rotation tensor may always be expressed in terms of a point (u_o, u_1, u_2, u_3) lying on the surface of a sphere in four-dimensional space:

$$\underset{\approx}{\boldsymbol{R}} = (2u_o^2 - 1)\underset{\approx}{\boldsymbol{I}} + 2\underset{\sim}{\boldsymbol{u}}\,\underset{\sim}{\boldsymbol{u}} - 2u_o \underset{\approx}{\boldsymbol{\varepsilon}} \bullet \underset{\sim}{\boldsymbol{u}}$$

$$\text{where } \underset{\sim}{\boldsymbol{u}} \equiv u_1 \underset{\sim}{\boldsymbol{E}}_1 + u_2 \underset{\sim}{\boldsymbol{E}}_2 + u_o \underset{\sim}{\boldsymbol{E}}_3 \tag{13.50}$$

The surface area of the 4D unit hypersphere is $2\pi^2$. Therefore, for a uniform distribution on the 4D hypersphere, the joint distribution function must be defined such that

$$p_{\vec{u}}(\vec{u}) = \frac{1}{2\pi^2} \tag{13.51}$$

The symbol $\vec{u}$ denotes the four-dimensional vector $\{u_o, u_1, u_2, u_3\}$. By virtue of Eq. (5.2), we may introduce 4-dimensional spherical coordinates $\{\beta, \theta, \phi\}$ such that

$$\begin{aligned} u_o &= \cos\beta \\ u_1 &= \sin\beta(\sin\theta\cos\phi) \\ u_2 &= \sin\beta(\sin\theta\sin\phi) \\ u_3 &= \sin\beta(\cos\theta) \end{aligned} \tag{13.52}$$

The area element on the 4D unit hypersphere is

$$dA = \sin\theta(\sin^2\beta)\,d\beta\,d\theta\,d\phi \tag{13.53}$$

Noting that β and θ range from 0 to π and ϕ ranges from 0 to 2π, the surface area of the entire 4D unit hypersphere is $2\pi^2$. For a uniform distribution on the 4D hypersphere, the joint distribution function for the spherical coordinates must be defined such that

$$p_{\vec{u}}(\vec{u})\,dA = p_{\beta,\theta,\phi}(\beta, \theta, \phi)\,d\beta\,d\theta\,d\phi \tag{13.54}$$

Hence, substituting Eqs. (13.51) and (13.53) into Eq. (13.54),

$$p_{\beta,\theta,\phi}(\beta, \theta, \phi) = \frac{\sin\theta(\sin^2\beta)}{2\pi^2} \tag{13.55}$$

By symmetry, the joint distribution should be separable. Therefore

$$p_{\beta,\theta,\phi}(\beta, \theta, \phi) = p_\beta(\beta)\,p_\theta(\theta)\,p_\phi(\phi) \tag{13.56}$$

Comparing the above two equations, we conclude

$$p_\beta(\beta) = \frac{2}{\pi}\sin^2(\beta) \tag{13.57}$$

$$p_\theta(\theta) = \frac{1}{2}\sin\theta \tag{13.58}$$

$$p_\phi(\phi) = \frac{1}{2\pi} \tag{13.59}$$

We have assigned the coefficients so that the integrals of each of the distribution functions over the full range of their arguments will equal unity, as required for any distribution function.

Recalling that the rotation angle α equals 2β, the density function for the rotation angle α must be given by

$$p_\alpha(\alpha) = \frac{1}{2} p_\beta\left(\frac{\alpha}{2}\right) = \frac{1}{\pi}\sin^2\left(\frac{\alpha}{2}\right) \tag{13.60}$$

The above distribution corresponds to

$$-\pi \le \alpha \le \pi \tag{13.61}$$

Without loss in generality it is always possible (by changing the direction of the rotation axis) to presume that the rotation angle satisfies $\sin\alpha \ge 0$, in which case,

$$0 \le \alpha \le \pi, \text{ and } p_\alpha(\alpha) = \frac{2}{\pi}\sin^2\left(\frac{\alpha}{2}\right) \tag{13.62}$$

In this case, a rotation is described by a unit quaternion lying on a spherical *hemisphere* in four dimensional space (so the factor of $2\pi^2$ in Eq. 13.55 would become π^2 and all of the subsequent arguments would go through unchanged to ultimately lead to Eq. 13.62).

Shoemake's quaternion algorithm for directly computing a uniform rotation. Shoemake claims that the following algorithm will result in a uniform rotation:

STEP 1. Generate three independent random numbers $\{X_0, X_1, X_2\}$ that are uniformly distributed between 0 and 1.

STEP 2. Compute two uniformly distributed angles, $\gamma_1 = 2\pi X_1$ and $\gamma_2 = 2\pi X_2$, and their sines and cosines, s_1, c_1, s_2, c_2.

STEP 3. Compute $r_1 = \sqrt{1 - X_o}$ and $r_2 = \sqrt{X_o}$.

STEP 4. Compute the unit quaternion with components

$$\begin{aligned} u_o &= c_2 r_2, \\ u_1 &= s_1 r_1, \quad u_2 = c_1 r_1, \quad u_3 = s_2 r_2 \end{aligned} \tag{13.63}$$

STEP 5. Compute the rotation by applying Eq. (13.50):

$$\begin{aligned} &\boldsymbol{R} = (2u_o^2 - 1)\boldsymbol{I} + 2\boldsymbol{u}\boldsymbol{u} - 2u_o\boldsymbol{\varepsilon}\bullet\boldsymbol{u} \\ &\text{where } \boldsymbol{u} \equiv u_1\boldsymbol{E}_1 + u_2\boldsymbol{E}_2 + u_3\boldsymbol{E}_3 \end{aligned} \tag{13.64}$$

Shoemake's algorithm has been verified by us and is given in the routine called "RNDC" in ***Listing 7 (Generating a uniformly random rigid rotation.) on page A-8***.

Numerically generating a rotation angle. The function needed to convert from a pseudo-random number on [0,1) to a rotation angle with distribution of Eq. (13.43) is determined by applying Eq. (13.16):

$$g(\alpha) \equiv \int_{-\infty}^{\alpha} p_\alpha(\alpha')\ d\alpha' = \frac{\alpha - \sin\alpha}{\pi} \tag{13.65}$$

Setting $g(\alpha) = r$ gives

$$\boxed{\alpha - \sin\alpha = \pi r} \tag{13.66}$$

If r is uniformly distributed on the interval from 0 to 1, then α is distributed according from 0 to π according to Eq. (13.43), and vice versa. Fig. 13.10*(b)* verifies the converse statement. Specifically, we generated 10^4 random rotations, and for each rotation, we used the function on page 3 to compute the rotation angle α. Fig. 13.10*(b)* shows that the cumulative distribution for $r = (\alpha - \sin\alpha)/\pi$ is linear, indicating that r is uniform.

Unfortunately, Eq. (13.66) cannot be solved to give α as an analytical function of r, so we would need to use a numerical root solver to obtain the sample value of α. **Listing 6 (Generating a uniformly random unit normal) on page A-7** could be used to generate a uniformly random rotation axis $\underset{\sim}{\boldsymbol{a}}$. The corresponding rotation tensor can be constructed by using the Euler-Rodrigues formula from Eq. (3.14):

$$\underset{\approx}{\boldsymbol{R}} = \cos\alpha\ (\underset{\approx}{\boldsymbol{I}} - \underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}}) + \underset{\sim}{\boldsymbol{a}}\,\underset{\sim}{\boldsymbol{a}} - \sin\alpha\ \underset{\approx}{\boldsymbol{\varepsilon}} \bullet \underset{\sim}{\boldsymbol{a}} \tag{13.67}$$

This random rotation will be uniformly distributed in the sense that any reference point $\underset{\sim}{\boldsymbol{N}}$ will map to a uniformly distributed sample point $\underset{\approx}{\boldsymbol{R}} \bullet \underset{\sim}{\boldsymbol{N}}$. The above scheme for computing the random rotation is equivalent to — but less efficient than — to the method described on page 100. The advantage of this formulation is that the analytical expressions for the density functions are better suited for computing expected values.

The expected value of a uniformly random rotation is zero.

We can write the rotation axis in terms of spherical coordinates as

$$\underset{\sim}{a} = \sin\theta(\cos\phi\,\underset{\sim}{e}_1 + \sin\phi\,\underset{\sim}{e}_2) + \cos\theta\,\underset{\sim}{e}_3 \tag{13.68}$$

If the rotation axis is uniformly distributed, then the distribution functions for θ and ϕ are

$$p_\theta(\theta) = \tfrac{1}{2}\sin\theta \quad \text{and} \quad p_\phi(\phi) = \frac{1}{2\pi} \tag{13.69}$$

Let $\underset{\sim}{N}$ be any unit vector. We wish to determine whether the distribution for $\underset{\approx}{R}\bullet\underset{\sim}{N}$ is uniform. We can always set up a basis such that $\underset{\sim}{e}_3 = \underset{\sim}{N}$. Then

$$\begin{aligned}\underset{\approx}{R}\bullet\underset{\sim}{N} &= \cos\alpha(\underset{\sim}{e}_3 - \underset{\sim}{a}\cos\theta) + \underset{\sim}{a}\cos\theta + \sin\alpha(\sin\theta(-\cos\phi\,\underset{\sim}{e}_2 + \sin\phi\,\underset{\sim}{e}_1))\\ &= \underset{\sim}{e}_1[(\sin\theta)(\cos\phi)(\cos\theta)(1-\cos\alpha) + (\sin\alpha)(\sin\theta)(\sin\phi)]\\ &+ \underset{\sim}{e}_2[(\sin\theta)(\sin\phi)(\cos\theta)(1-\cos\alpha) - (\sin\alpha)(\sin\theta)(\cos\phi)]\\ &+ \underset{\sim}{e}_3[\cos^2\theta + (\cos\alpha)\sin^2\theta]\end{aligned} \tag{13.70}$$

Let's verify that the distributions of Eqs. (13.42) and (13.69) correspond to a zero expected value of $\underset{\approx}{R}\bullet\underset{\sim}{e}_3$:

$$E[\underset{\approx}{R}\bullet\underset{\sim}{N}] = \int_{-\pi}^{\pi}\int_{0}^{\pi}\int_{0}^{(2\pi)} (\underset{\approx}{R}\bullet\underset{\sim}{e}_3)(p_\phi(\phi)(p_\theta(\theta)p_\alpha(\alpha)))d\phi\;\,d\theta\;\,d\phi \tag{13.71}$$

Performing this integral does indeed show that the distribution has a zero expected value. This is a necessary (not sufficient*) condition for $\underset{\approx}{R}\bullet\underset{\sim}{N}$ to be uniformly distributed on the unit sphere. To prove that $\underset{\approx}{R}\bullet\underset{\sim}{N}$ is uniformly distributed, we write

$$\underset{\approx}{R}\bullet\underset{\sim}{N} = n_1\underset{\sim}{e}_1 + n_2\underset{\sim}{e}_2 + n_3\underset{\sim}{e}_3 \tag{13.72}$$

The distributions of Eqs. (13.42) and (13.69) correspond to a uniform distribution for the rotation if and only if the coordinates (n_1, n_2, n_3) are uniformly distributed on the unit sphere. Thus

$$p_{\underset{\sim}{n}}(\underset{\sim}{n}) = \frac{1}{4\pi}, \tag{13.73}$$

which is indeed the distribution implied by Eqs. (13.42) and (13.69).

* Uniform Euler angles will produce a non-uniform distribution of $\underset{\approx}{R}\bullet\underset{\sim}{N}$ that nevertheless has a zero expected value.

The connection between the isotropic part of a tensor and the expected value of that tensor over the set of uniform superimposed rotations. Let $\underset{\approx}{\boldsymbol{S}}$ be some known tensor and let $\overline{\underset{\approx}{\boldsymbol{S}}}$ denote a second tensor that is generated by superimposing a rotation on the first tensor:

$$\overline{\underset{\approx}{\boldsymbol{S}}} = \underset{\approx}{\boldsymbol{R}}^T \bullet \underset{\approx}{\boldsymbol{S}} \bullet \underset{\approx}{\boldsymbol{R}} \tag{13.74}$$

If $\underset{\approx}{\boldsymbol{R}}$ is uniformly random, then (without proof) the expected value of $\overline{\underset{\approx}{\boldsymbol{S}}}$ is

$$E[\overline{\underset{\approx}{\boldsymbol{S}}}] = \frac{1}{3}(\mathrm{tr}\underset{\approx}{\boldsymbol{S}})\underset{\approx}{\boldsymbol{I}} \tag{13.75}$$

In other words, the expected value of a tensor over all possible uniform rotations of that tensor is just the *isotropic part* of that tensor. This result also holds for higher-order tensors as well. In general, finding the isotropic part of a tensor is the same as projecting that tensor to the space of isotropic tensors. For example, a second-order tensor $\underset{\approx}{\boldsymbol{A}}$ is isotropic if and only if

$$\underset{\approx}{\boldsymbol{R}}^T \bullet \underset{\approx}{\boldsymbol{A}} \bullet \underset{\approx}{\boldsymbol{R}} = \underset{\approx}{\boldsymbol{A}} \quad \text{or} \quad R_{pi}R_{qj}A_{pq} = A_{ij} \text{ for all rotations } \underset{\approx}{\boldsymbol{R}}. \tag{13.76}$$

Writing this definition in component form eventually reveals that any isotropic tensor (in 3D space) must be simply proportional to the identity tensor $\underset{\approx}{\boldsymbol{I}}$. Therefore, the identity tensor must be a *basis* for the space of isotropic second-order tensors referenced to 3D physical space. By "referenced to 3D physical space" we simply mean that the tensor is expressible in terms of a 3×3 matrix. Because all isotropic 3×3 matrices are expressible as some scalar times one matrix (the identity matrix), the space of isotropic second-order tensors referenced to 3D physical space must be one-dimensional. A 3D second-order tensor is projected to its one-dimensional isotropic space by applying the following projection operation:

$$\mathrm{iso}\underset{\approx}{\boldsymbol{S}} = (\underset{\approx}{\boldsymbol{S}}*\hat{\underset{\approx}{\boldsymbol{I}}})\hat{\underset{\approx}{\boldsymbol{I}}} \quad \text{for 3D } 2^{\text{nd}}\text{-order tensors} \tag{13.77}$$

Here, the * denotes the inner product, which for second-order tensors is defined $\underset{\approx}{\boldsymbol{A}}*\underset{\approx}{\boldsymbol{B}} = A_{ij}B_{ij}$. The "hat" over the identity tensor denotes normalization of the identity tensor. Specifically:

$$\hat{\underset{\approx}{\boldsymbol{I}}} = \frac{\underset{\approx}{\boldsymbol{I}}}{\sqrt{\underset{\approx}{\boldsymbol{I}}*\underset{\approx}{\boldsymbol{I}}}} = \frac{\underset{\approx}{\boldsymbol{I}}}{\sqrt{3}} \tag{13.78}$$

Noting that $\underset{\approx}{\boldsymbol{S}}*\underset{\approx}{\boldsymbol{I}}$ is the same thing as $\mathrm{tr}\underset{\approx}{\boldsymbol{S}}$, Eq. (13.77), becomes the traditional expression for the isotropic part of a 3D second-order tensor:

$$\mathrm{iso}\underset{\approx}{\boldsymbol{S}} = \tfrac{1}{3}(\mathrm{tr}\underset{\approx}{\boldsymbol{S}})\underset{\approx}{\boldsymbol{I}} \quad \text{for 3D } 2^{\text{nd}}\text{-order tensors} \tag{13.79}$$

We have emphasized that this result holds for second-order tensors referenced to *three-dimensional* physical space. To see why this is important, consider *two-dimensional* space. A 2D second-order tensor is expressible in terms of a 2×2 matrix and the tensor is isotropic if and only if

$$\begin{bmatrix}\cos\alpha & \sin\alpha\\ -\sin\alpha & \cos\alpha\end{bmatrix}\begin{bmatrix}A_{11} & A_{12}\\ A_{21} & A_{22}\end{bmatrix}\begin{bmatrix}\cos\alpha & -\sin\alpha\\ \sin\alpha & \cos\alpha\end{bmatrix} = \begin{bmatrix}A_{11} & A_{12}\\ A_{21} & A_{22}\end{bmatrix} \text{ for all } \alpha \tag{13.80}$$

Analyzing this definition reveals that the only restrictions that need to be placed on the components of $[A]$ are

$$A_{11} = A_{22} \text{ and } A_{12} = -A_{21} \tag{13.81}$$

Therefore, any isotropic tensor in *two-dimensions* is expressible in the form

$$\begin{bmatrix}\gamma & \beta\\ -\beta & \gamma\end{bmatrix} \text{ for some scalars } \gamma \text{ and } \beta. \tag{13.82}$$

In other words, any isotropic tensor in *two-dimensions* can always be written as a linear combination of the 2D identity tensor $\underset{\approx}{\boldsymbol{I}}$ and the 2D alternating tensor $\underset{\approx}{\varepsilon}$ (defined such that ε_{ij} is zero if $i=j$, $+1$ if $ij = 12$, and -1 if $ij = 21$). Thus, the space of isotropic tensors in *two-dimensions* is two dimensional. The matrix expressions for the two base tensors are

$$[\underset{\approx}{\boldsymbol{I}}] = \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} \text{ and } [\underset{\approx}{\varepsilon}] = \begin{bmatrix}0 & 1\\ -1 & 0\end{bmatrix} \tag{13.83}$$

This basis is already orthogonal. The normalized basis is

$$\hat{\underset{\approx}{\boldsymbol{I}}} = \frac{\underset{\approx}{\boldsymbol{I}}}{\sqrt{\underset{\approx}{\boldsymbol{I}}*\underset{\approx}{\boldsymbol{I}}}} = \frac{\underset{\approx}{\boldsymbol{I}}}{\sqrt{2}} \quad \text{and} \quad \hat{\underset{\approx}{\varepsilon}} = \frac{\underset{\approx}{\varepsilon}}{\sqrt{\underset{\approx}{\varepsilon}*\underset{\approx}{\varepsilon}}} = \frac{\underset{\approx}{\varepsilon}}{\sqrt{2}} \tag{13.84}$$

Here, the "*" operator is the 2D inner product, defined by $\underset{\approx}{\boldsymbol{A}}*\underset{\approx}{\boldsymbol{B}} = A_{ij}B_{ij}$ where repeated indices are summed from 1 to 2.

The projection of a 2D second-order tensor $\underset{\approx}{\boldsymbol{S}}$ to its isotropic part is obtained by the projection operation

$$\text{iso}\underset{\approx}{\boldsymbol{S}} = (\underset{\approx}{\boldsymbol{S}}*\hat{\underset{\approx}{\boldsymbol{I}}})\hat{\underset{\approx}{\boldsymbol{I}}} + (\underset{\approx}{\boldsymbol{S}}*\hat{\underset{\approx}{\varepsilon}})\hat{\underset{\approx}{\varepsilon}} \quad \text{for 2D } 2^{\text{nd}}\text{-order tensors} \tag{13.85}$$

Expanding this out in component form gives

$$[\text{iso}\underset{\approx}{\boldsymbol{S}}] = \begin{bmatrix}\frac{1}{2}(S_{11}+S_{22}) & \frac{1}{2}(S_{12}-S_{21})\\ \frac{1}{2}(S_{21}-S_{12}) & \frac{1}{2}(S_{11}+S_{22})\end{bmatrix} \quad \text{for 2D } 2^{\text{nd}}\text{-order tensors} \tag{13.86}$$

This formula is significantly different from the corresponding operation for 3D tensors!

The techniques described so far for second-order tensors generalize similarly to higher-order tensors. For example, a 3D third-order tensor $\underset{\approx}{\boldsymbol{A}}$ is isotropic if and only if

$$R_{pi}R_{qj}R_{sk}A_{pqs} = A_{ijk} \text{ for all rotations } \underset{\approx}{\boldsymbol{R}}. \tag{13.87}$$

It turns out that this condition can be satisfied if and only if $\underset{\approx}{\boldsymbol{A}}$ is proportional to the third-order alternating tensor. Therefore, the isotropic part of a third-order tensor is obtained by the projection operation

$$\text{iso}\underset{\approx}{\zeta} = \left(\underset{\approx}{\zeta} * \hat{\underset{\approx}{\varepsilon}}\right)\hat{\underset{\approx}{\varepsilon}}, \tag{13.88}$$

where "*" is now the third-order inner product ($\underset{\approx}{\boldsymbol{A}} * \underset{\approx}{\boldsymbol{B}} = A_{ijk}B_{ijk}$) and

$$\hat{\underset{\approx}{\varepsilon}} = \frac{\underset{\approx}{\varepsilon}}{\sqrt{\underset{\approx}{\varepsilon} * \underset{\approx}{\varepsilon}}} = \frac{\underset{\approx}{\varepsilon}}{\sqrt{6}} \tag{13.89}$$

As before, the isotropic part of a third order tensor is identical to the expected value of that tensor over the set of uniform superimposed rotations. Thus, the expected value of a 3D third-order tensor over a uniformly random rotation is given by

$$E\left[\bar{\underset{\approx}{\zeta}}\right] = \text{iso}\underset{\approx}{\zeta} = \frac{1}{6}(\zeta_{mnp}\varepsilon_{mnp})\varepsilon_{ijk}\underset{\approx}{\boldsymbol{e}}_i\underset{\approx}{\boldsymbol{e}}_j\underset{\approx}{\boldsymbol{e}}_k \tag{13.90}$$

Continuing in this same vein, the expected value of a fourth-order tensor over uniform rotations is given by the isotropic part of the tensor:

$$E\left[\bar{\underset{\approx}{\underset{\approx}{C}}}\right] = \text{iso}\underset{\approx}{\underset{\approx}{C}} \tag{13.91}$$

Again, to compute the isotropic part, we must project the tensor to the space of isotropic fourth-order tensors. It turns out that an isotropic 3D fourth-order tensor is always expressible as a linear combination of three isotropic base tensors having $ijkl$ components given by

$$\delta_{ij}\delta_{kl},\ \delta_{ik}\delta_{jl},\ \text{and}\ \delta_{il}\delta_{jk} \tag{13.92}$$

These three tensors form a basis for isotropic 3D fourth-order tensors. However, the basis is not orthonormalized, as it must be if we wish to define the projection operation is the usual way. An alternative orthogonal set of base-tensors for the space of isotropic 4th-order tensors is

$$\begin{aligned} P^{\mathsf{I}}_{ijkl} &= \frac{1}{3}\delta_{ij}\delta_{kl} \\ P^{\mathsf{SD}}_{ijkl} &= \frac{1}{2}(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}) - \frac{1}{3}\delta_{ij}\delta_{kl} \\ P^{\mathsf{A}}_{ijkl} &= \frac{1}{2}(\delta_{ik}\delta_{jl} - \delta_{il}\delta_{jk}) \end{aligned} \tag{13.93}$$

The tensors in this alternative basis are projectors themselves. When $P^{\mathtt{I}}_{ijkl}$ acts on a second-order tensor, the result is the isotropic part of that tensor. When $P^{\mathtt{SD}}_{ijkl}$ acts on a second-order tensor, the result is the symmetric-deviatoric part of that tensor. When $P^{\mathtt{A}}_{ijkl}$ acts on a tensor, the result is the anti-symmetric part of that tensor.

We note that

$$\begin{aligned} P^{\mathtt{I}}_{ijkl}P^{\mathtt{I}}_{ijkl} &= 1 \\ P^{\mathtt{SD}}_{ijkl}P^{\mathtt{SD}}_{ijkl} &= 5 \\ P^{\mathtt{A}}_{ijkl}P^{\mathtt{A}}_{ijkl} &= 3 \end{aligned} \tag{13.94}$$

Therefore, the isotropic part of a fourth-order tensor C_{ijkl} is obtained by

$$\left(\mathrm{iso}\underset{\approx}{C}\right)_{ijkl} = (C_{pqrs}P^{\mathtt{I}}_{pqrs})P^{\mathtt{I}}_{ijkl} + \frac{(C_{pqrs}P^{\mathtt{SD}}_{pqrs})P^{\mathtt{SD}}_{ijkl}}{5} + \frac{(C_{pqrs}P^{\mathtt{A}}_{pqrs})P^{\mathtt{A}}_{ijkl}}{3} \tag{13.95}$$

14. SCALARS and INVARIANTS

An invariant is something that remains unchanged upon a change of basis. The components of a vector are NOT invariant because those components change when we change the basis. However, the *sum* of components times base vectors will remain unchanged upon a change of basis. Thus, in this sense, vectors such as velocity and electric field are invariant -- these physical quantities themselves do not change simply because we change our own orientation from which we view them. If the invariant is a *single number* (such as temperature or mass density), then it is usually called a *scalar*. The term *scalar invariant* is typically used to denote a single number invariant that is obtained by some operation applied to a directional quantity. For example, the magnitude of a vector is a scalar invariant of the vector. The trace of a tensor is a scalar invariant of that tensor. A scalar invariant is a real-valued function of the components of a vector or tensor that will give the same result regardless of what basis is used. For example, the trace of a tensor $\underset{\approx}{\boldsymbol{A}}$ is typically defined

$$\sum_{i=1}^{3} A_{ii} \tag{14.1}$$

To prove that this is a scalar invariant, we must prove that it gives the same result regardless of the basis used.

Rebecca: provide this proof

In Chapter 2, we discussed the distinction between

1. leaving physical quantities alone while we reorient *ourselves*.
2. leaving our orientation fixed, while we reorient the physical plane.

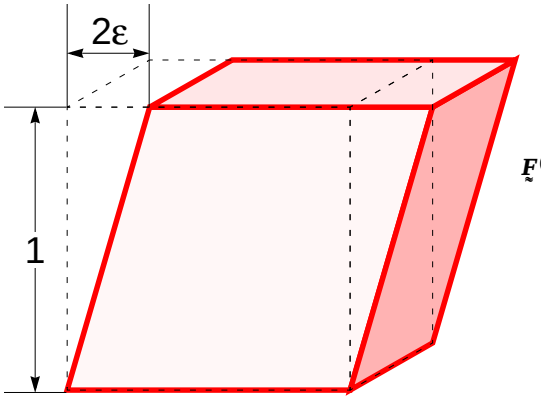
For the first situation, all scalar invariants will be the same before and after we reorient ourselves. For the second situation, scalar invariants of *most* of the physical quantities (e.g., their magnitudes) will remain unchanged, but some of the scalar invariants might actually change. A good example, is the trace of the rotation tensor. If we have a rotation tensor and take its trace in two different orthonormal bases, the result will be the same. However, if we have a rotation tensor in one physical system, and we create a second physical system by adding additional rotation to the first one, then the rotation tensor for the second system will be different from that of the first. In particular, if both systems share the same rotation axis, then the rotation angle for the second system will be larger than the rotation angle for the first system by an amount equal to our superimposed rotation.

15. The principle of material frame indifference (PMFI)

What is a "superimposed rotation"?

Suppose that you have worked extremely hard to derive the mapping function, stretch, rotation, and other kinematic properties for a particular motion such as uniaxial strain or simple shear. Suppose that you also applied material models so that you now know all the applied stresses needed to obtain this particular deformed state. Now suppose that you were *supposed* to have derived the results for a deformation identical to the deformation that you analyzed, except rotated. Is there any way to easily convert your results, or do you have to start over from scratch?

The starting deformation for which you have already suffered through obtaining the kinematic and mechanics solution is called the "fiducial" deformation. In physics, the word "fiducial" means "regarded or employed as a standard of reference for measurement or calculation." When analyzing the fiducial deformation, the standard of reference is the undeformed state. Now we ask: what if we seek to analyze a new deformation where the reference configuration is still the undeformed reference, but the deformation is now given by the fiducial deformation subjected to an additional rotation?



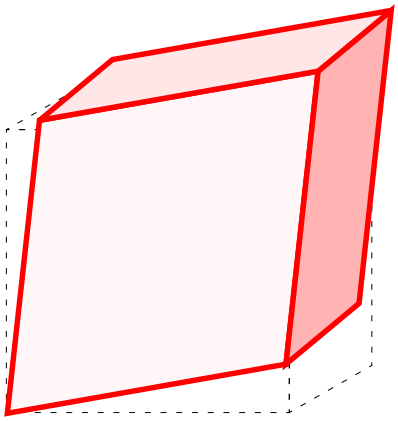
"fiducial" deformation: SIMPLE SHEAR

$$\underset{\sim}{\boldsymbol{F}}^{(\text{fiducial})} = \begin{bmatrix} 1 & 2\varepsilon & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \underset{\sim}{\boldsymbol{R}}^{(\text{fiducial})} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

where $\theta = \tan^{-1}\varepsilon$

$$\underset{\sim}{\boldsymbol{U}}^{(\text{fiducial})} = (\underset{\sim}{\boldsymbol{R}}^{(\text{fiducial})})^T \bullet \underset{\sim}{\boldsymbol{F}}^{(\text{fiducial})}$$

$$\underset{\sim}{\boldsymbol{V}}^{(\text{fiducial})} = \underset{\sim}{\boldsymbol{F}}^{(\text{fiducial})} \bullet (\underset{\sim}{\boldsymbol{R}}^{(\text{fiducial})})^T$$



"star" deformation: SIMPLE SHEAR with superimposed rotation Q with axis parallel to 3-direction

$$\underset{\sim}{\boldsymbol{F}}^{(\text{star})} = \underset{\sim}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{F}}^{(\text{fiducial})}$$

$$\underset{\sim}{\boldsymbol{R}}^{(\text{star})} = \underset{\sim}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{R}}^{(\text{fiducial})}$$

$$\underset{\sim}{\boldsymbol{U}}^{(\text{star})} = \underset{\sim}{\boldsymbol{U}}^{(\text{fiducial})}$$

$$\underset{\sim}{\boldsymbol{V}}^{(\text{star})} = \underset{\sim}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{V}}^{(\text{fiducial})} \bullet \underset{\sim}{\boldsymbol{Q}}^T$$

This additional rotation that is applied to the fiducial deformation is called a superimposed rotation. As indicated in the above figure, the results from the fiducial analysis can be immediately modified to permit the superimposed rotation. Specifically, the "star" deformation that results from superimposing the rotation $\underset{\approx}{\boldsymbol{Q}}$ on top of the fiducial deformation $\underset{\approx}{\boldsymbol{F}}^{(\text{fiducial})}$ is immediately seen to be $\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{F}}^{(\text{fiducial})}$. Likewise, the rotation tensor must be obtained by applying the superimposed rotation $\underset{\approx}{\boldsymbol{Q}}$ on top of the fiducial rotation $\underset{\approx}{\boldsymbol{R}}^{(\text{fiducial})}$ so that the rotation resulting from these two sequentially applied deformations must be given by $\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{R}}^{(\text{fiducial})}$.

Things get confusing when we look at the stretch tensors. Recall that the *right* stretch $\underset{\approx}{\boldsymbol{U}}$ is the stretch applied in the undeformed reference configuration. Now recall that a "superimposed rotation" corresponds to an alteration of the fiducial physical system by adding additional rotation. The key is that both the fiducial system and the rotated system have the same initial states. Thus, for example, if $\underset{\sim}{\boldsymbol{N}}$ is the *initial* orientation of some material fiber, then that initial orientation is the same for both the fiducial system and the one that includes superimposed rotation.

$$\underset{\sim}{\boldsymbol{N}}^{(\text{star})} = \underset{\sim}{\boldsymbol{N}}^{(\text{fiducial})} \tag{15.1}$$

The right stretch tensor is similar. Recall that the polar decomposition $\underset{\approx}{\boldsymbol{F}} = \underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\boldsymbol{U}}$ can be interpreted physically as a pure stretch $\underset{\approx}{\boldsymbol{U}}$ *followed by* a pure rotation $\underset{\approx}{\boldsymbol{R}}$. Since the superimposed rotation is applied *after* the initial stretch, we see that the initial stretch must be the same for both the "star" and "fiducial" systems:

$$\underset{\approx}{\boldsymbol{U}}^{(\text{star})} = \underset{\approx}{\boldsymbol{U}}^{(\text{fiducial})} \tag{15.2}$$

By contrast, the right stretch tensor $\underset{\approx}{\boldsymbol{V}}$ in the polar decomposition $\underset{\approx}{\boldsymbol{F}} = \underset{\approx}{\boldsymbol{V}} \bullet \underset{\approx}{\boldsymbol{R}}$ is interpreted physically as a stretch that takes place *after* the rotation is applied. For the two systems, we have

$$\underset{\approx}{\boldsymbol{F}}^{(\text{star})} = \underset{\approx}{\boldsymbol{V}}^{(\text{star})} \bullet \underset{\approx}{\boldsymbol{R}}^{(\text{star})} \tag{15.3}$$

and

$$\underset{\approx}{\boldsymbol{F}}^{(\text{fiducial})} = \underset{\approx}{\boldsymbol{V}}^{(\text{fiducial})} \bullet \underset{\approx}{\boldsymbol{R}}^{(\text{fiducial})} \tag{15.4}$$

Recalling that $\underset{\approx}{\boldsymbol{R}}^{(\text{star})} = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{R}}^{(\text{fiducial})}$ and $\underset{\approx}{\boldsymbol{F}}^{(\text{star})} = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{F}}^{(\text{fiducial})}$, we can rewrite Eq. (15.3) as

$$\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{F}}^{(\text{fiducial})} = \underset{\approx}{\boldsymbol{V}}^{(\text{star})} \bullet \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{R}}^{(\text{fiducial})} \tag{15.5}$$

Thus putting Eq. (15.4) into (15.5) and solving for $\underset{\approx}{\boldsymbol{V}}^{(\text{star})}$ gives

$$\underset{\approx}{\boldsymbol{V}}^{(\text{star})} = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{V}}^{(\text{fiducial})} \bullet \underset{\approx}{\boldsymbol{Q}}^{T} \tag{15.6}$$

Thus, the right stretch tensor transforms under a superimposed rigid rotation in a manner that looks quite similar to the component transformation formula associated with a basis change (see, for example, Eq. 1.32). The fact that some tensors transform in a manner like a basis transform while others don't is a key feature that distinguishes superimposed rotation from simple reorientation of the observer. Some books speak of the principle of material frame indifference by speaking of an orthogonal change of frame. It's important to realize that the change of frame is applied *after* the physical state has been estabished. The way to think about this viewpoint is that two observers are initially identically oriented, but they adopt different orientations as the physical system undergoes its process. Then both observers record *identical* component matrices for all of the *initial* vectors and tensors, while they observe relatively rotated components for the current — or *spatial* — vectors and tensors.

"Reference" and "Objective/spatial" tensors

A **reference** tensor (including zeroth and first order tensors — i.e., scalars and vectors) is one that is unchanged by a superimposed rotation. Examples include, as discussed above, the initial orientation of a vector, the right stretch tensor, etc. That is,

$$\text{If } s^{(\text{star})} = s^{(\text{fiducial})}\text{, then the scalar } s \text{ is a reference scalar} \tag{15.7}$$

$$\text{If } \underset{\sim}{\boldsymbol{v}}^{(\text{star})} = \underset{\sim}{\boldsymbol{v}}^{(\text{fiducial})}\text{, then the vector } \underset{\sim}{\boldsymbol{v}} \text{ is a reference vector} \tag{15.8}$$

$$\text{If } \underset{\approx}{\boldsymbol{T}}^{(\text{star})} = \underset{\approx}{\boldsymbol{T}}^{(\text{fiducial})}\text{, then the tensor } \underset{\approx}{\boldsymbol{T}} \text{ is a reference tensor} \tag{15.9}$$

and so on.

An *objective* or *spatial* tensor is one whose transformation under a superimposed rotation has the same form as the transformation under an orthogonal basis rotation.

$$\text{If } s^{(\text{star})} = s^{(\text{fiducial})}\text{, then the scalar } s \text{ is a spatial (objective) scalar} \tag{15.10}$$

$$\text{If } \underset{\sim}{\boldsymbol{v}}^{(\text{star})} = \underset{\sim}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{v}}^{(\text{fiducial})}\text{, then the vector } \underset{\sim}{\boldsymbol{v}} \text{ is a spatial (objective) vector} \tag{15.11}$$

$$\text{If } \underset{\approx}{\boldsymbol{T}}^{(\text{star})} = \underset{\sim}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{T}}^{(\text{fiducial})} \bullet \underset{\sim}{\boldsymbol{Q}}^{T}\text{, then the tensor } \underset{\approx}{\boldsymbol{T}} \text{ is a spatial (objective) tensor} \tag{15.12}$$

First note that, if a scalar is objective, then it is also a reference scalar. The vast majority of scalars are objective, but (as discussed below) some are not.

Note that Eq. (15.12) can be written in component form as

$$\text{If } T_{ij}^{(\text{star})} = Q_{im} Q_{jn} T_{mn}^{(\text{fiducial})}\text{, then the tensor } \underset{\approx}{\boldsymbol{T}} \text{ is a spatial (objective) tensor} \tag{15.13}$$

For higher-order tensors, it's easiest to define the objectivity (spatial tensor) property using a notation like this. Consider, for example, a third order tensor $\underset{\approx}{\xi}$:

If $\xi_{ijk}^{(\text{star})} = Q_{im}Q_{jn}Q_{kp}\xi_{mnp}^{(\text{fiducial})}$, then the tensor $\underset{\approx}{\xi}$ is a spatial (objective) tensor. (15.14)

An alternative way to think about an objective tensor is that the "star" tensor is the same as the "fiducial" tensor if all of the base vectors used in the basis expansion of the tensor are changed from $\underset{\sim}{\boldsymbol{E}}_i$ to $\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{E}}_i$. In other words,

If $\underset{\approx}{\xi}^{(\text{star})} = \xi_{ijk}^{(\text{fiducial})}(\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{E}}_i)(\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{E}}_i)(\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{E}}_i)$, then $\underset{\approx}{\xi}$ is a spatial (objective) tensor (15.15)

This result suggests that it might be useful to employ two different orthonormal bases when working physical problems.

Reference tensors are most naturally expressed in terms of the $\underset{\sim}{\boldsymbol{E}}_i$ basis (15.16)

Objective/spatial tensors are most naturally expressed in terms of a basis $\underset{\sim}{\boldsymbol{e}}_i \equiv \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{E}}_i$ (15.17)

True or False: scalars are unaffected by a superimposed rigid rotation.

The answer is "false." Understanding the reason for this answer is an important step towards understanding why the principle of material frame indifference (PMFI) is a concept distinct from the rules governing orthogonal coordinate transformations. As discussed in Section 14, a scalar is something whose value is unchanged upon a basis rotation. Most physical scalars are objective. However, not all scalars will be unchanged upon a superimposed rotation. A good counterexample is the trace of the rotation tensor discussed in Section 14 -- that quantity is a scalar invariant, but it is neither a reference nor objective scalar.

As a rule of thumb, reference tensors are quantities whose definition is closely coupled to the *initial* state (which, recall, is the same for both the "fiducial" and "star" states). Objective/spatial tensors are quantities whose definition is most physically meaningful in the *current* state. Some quantities, such as the deformation gradient tensor or the trace of the rotation, are neither reference nor spatial — these entities typically carry information that couples the initial state to the current state.

The natural basis for reference tensors is the $\underset{\sim}{\boldsymbol{E}}_i\underset{\sim}{\boldsymbol{E}}_i$ basis, while the natural basis for spatial tensors is the $\underset{\sim}{\boldsymbol{e}}_i\underset{\sim}{\boldsymbol{e}}_i$ basis. For tensors that are neither reference nor spatial, the natural basis is often a "two-point" or mixed basis. The natural basis for the deformation gradient is $\underset{\sim}{\boldsymbol{e}}_i\underset{\sim}{\boldsymbol{E}}_j$, whereas the natural basis for the *transpose* of the deformation gradient is $\underset{\sim}{\boldsymbol{E}}_j\underset{\sim}{\boldsymbol{e}}_i$. The concept of a natural basis is useful for double-checking analytical work because any operation that involves the dot product of $\underset{\sim}{\boldsymbol{e}}_i$ with $\underset{\sim}{\boldsymbol{E}}_j$ is so rare that it serves as a "red flag" that an error might exist. If, for example, your analysis results in the appearance of $\underset{\approx}{\boldsymbol{F}} \bullet \underset{\approx}{\boldsymbol{F}}$, you should be worried because, in terms of the natural basis,

$$\underset{\approx}{\boldsymbol{F}} \bullet \underset{\approx}{\boldsymbol{F}} = (F_{ij}\underset{\sim}{\boldsymbol{e}}_i\underset{\sim}{\boldsymbol{E}}_j) \bullet (F_{mn}\underset{\sim}{\boldsymbol{e}}_m\underset{\sim}{\boldsymbol{E}}_n) \tag{15.18}$$

Expanding this out involves the dot product $\underset{\sim}{\boldsymbol{E}}_j \bullet \underset{\sim}{\boldsymbol{e}}_m$, which involves base vectors from two different systems. This suggests that an error must have creeped into your work. You might have intended to write, say, $\underset{\approx}{\boldsymbol{F}}^T \bullet \underset{\approx}{\boldsymbol{F}}$ or $\underset{\approx}{\boldsymbol{F}} \bullet \underset{\approx}{\boldsymbol{F}}^T$, both of which involve dot products of *the same class of base vector.* These are subtle concepts that a practicing engineer can get by without ever knowing — they just serve as another item in one's bag of tricks that is sometimes useful for spotting errors.

Prelude to PMFI: A philosophical diversion

The study of physics often involves intuitively simple concepts stated in a very obtuse — but ultimately more useful — mathematical form. Switching from imprecise English (or other language) to mathematics is essential for any advanced analysis. Unfortunately, however, the original childishly simple obviousness of the original concept often becomes lost in the process. Consider, for example, the mathematically precise statement of conservation of mass:

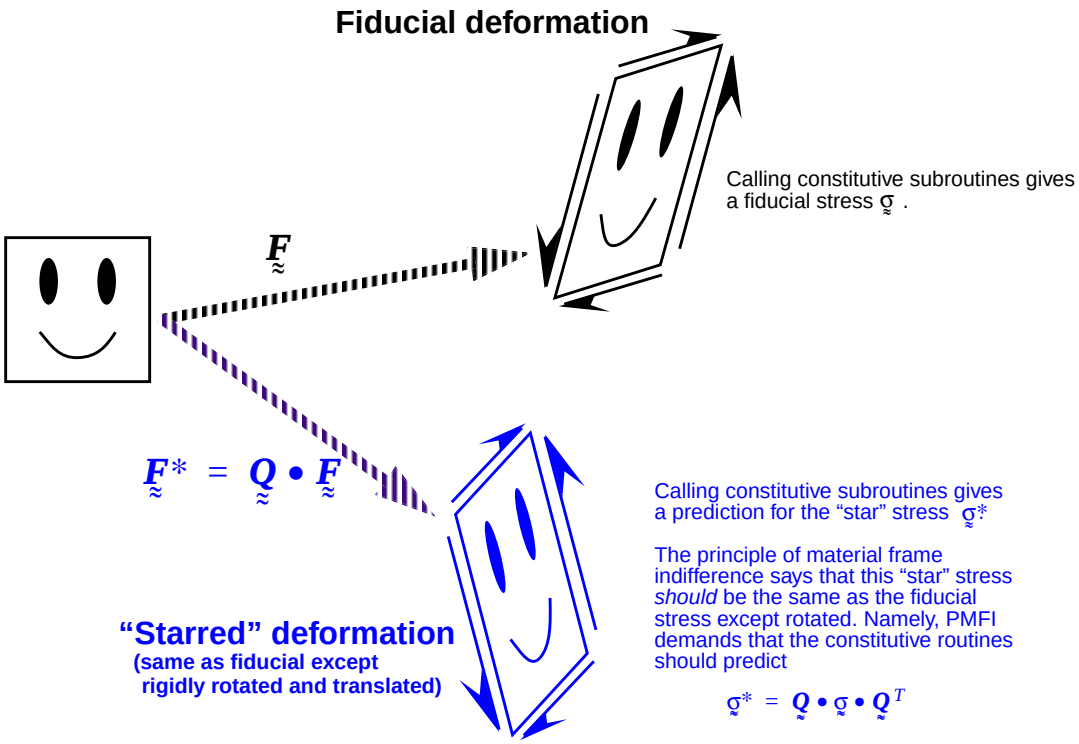
$$\dot{\rho} + \rho\nabla \cdot \underset{\sim}{\boldsymbol{v}} = 0, \tag{15.19}$$

where ρ is the mass density, $\nabla \cdot (\ )$ is the divergence operator, and $\underset{\sim}{\boldsymbol{v}}$ is the material velocity. It's easy to forget that this stilted equation is just an obtuse way of expressing the intuitive concept that **"What comes in must go out — or stay there."**

Intuitively simple concepts often become quite obtuse when cast in mathematical form. The mathematical form is nonetheless essential to ensure adherence to the principle. Material frame indifference is no exception. Its precise statement is somewhat daunting, but its basic meaning is really quite simple. In the subsequent sections, we are going to discuss several motivational examples to introduce the basic idea of PMFI.

PMFI: a sloppy introduction

The principle of material frame indifference imposes a physically intuitive constraint on material constitutive laws. If you apply the constitutive law for some particular deformation, which we shall call the "fiducial" deformation, then you get a certain prediction for the stress tensor $\underset{\approx}{\sigma}$. Suppose that you compute the prediction of the constitutive law for a *different* "starred" deformation that is identical to the fiducial deformation except that it additionally has rigid rotation and translation. Intuitively, you would expect that the stress for the starred deformation should be exactly the same as the stress for the fiducial deformation except rotated along with the material. The principle of material frame indifference (PMFI) is the obtuse and often confusing mathematical expression of this intuitively simple physical concept. We will later see that models violating PMFI can usually be corrected by simply changing the model's independent variables.



Translational frame invariance

In this section, we are going to show a simple and obviously flawed model that is *not* frame indifferent. The idea here is to pretend that you have programmed your model into a dumb computer that is capable of *only* applying the model literally — the computer does not assess validity of the model. Any rational human being would have no problem discerning the *intended* meaning of the flawed model. The point of this section is that the *computer* (which always applies the model exactly as you program it) will get the wrong result.

Before looking at material constitutive equations, lets consider a model for a linear spring: The force f in the spring equals a spring constant k times change in length δ of spring.

$$f = k\delta \tag{15.20}$$

Suppose that we wish to generalize this model so that it applies in vector form. Then we might introduce the following variables:

$$\begin{aligned} \underset{\sim}{\boldsymbol{x}} &= \text{current location of the tip of the spring.} \\ \underset{\sim}{\boldsymbol{x}}_o &= \text{original (unstressed) location of the tip of the spring.} \end{aligned} \tag{15.21}$$

An undergraduate physics student might be tempted to propose

$$\underset{\sim}{\boldsymbol{f}} = k(\underset{\sim}{\boldsymbol{x}} - \underset{\sim}{\boldsymbol{x}}_o) \qquad \leftarrow \text{(violates PMFI)} \tag{15.22}$$

This model violates invariance under rigid translation because *in its literal sense*, it does not account for the possibility that the *tail* of the spring might move. If we were to implement Eq. (15.22) on a dumb computer, the resulting program would wrongly predict a *larger* spring force for the star deformation in Fig. 15.1 than for the fiducial deformation, even though both deformations involve the same amount of spring stretch.

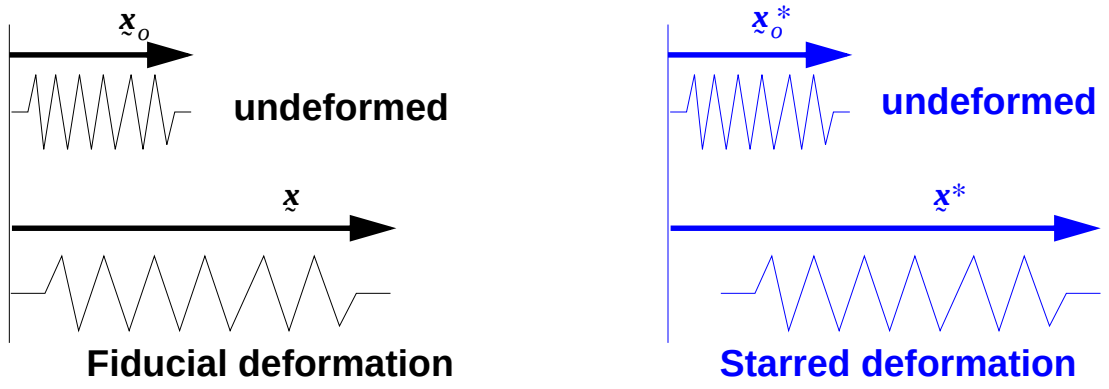


Figure 15.1. Two deformations involving the same spring stretch for which the faulty model wrongly predicts different spring forces. On the left is a deformation in which the tail of the spring does not move. On the right is a deformation that is identical to the fiducial deformation except that the spring is also rigidly translated. The model of Eq. (15.22) predicts the wrong answer because it does not include the position of the *tail*

As human beings, we know *intuitively* that the deformation on the right side of Fig. 15.1 should involve the same spring force since it involves the same amount of spring elongation. However, a computer program using Eq. (15.1) has no intuition — it just gives us the wrong answer.* For (Galilean) invariance under translation, our spring model must have the property that it will predict exactly the same force whenever we rigidly translate the spring in space. This is an intuitively simple concept. Its mathematical expression, however, is less obvious. The idea here is that we compare two deformations:

> **Fiducial deformation**: stretch the spring by some known amount.
> **Starred deformation**: stretch the spring by the same amount and also translate it by an amount $\boldsymbol{c}$.

Both scenarios involve the same starting location, so $\boldsymbol{x}_o$ is the same in both cases. Let $\boldsymbol{x}$ denote the deformed location of the tip for the fiducial deformation. Let $\boldsymbol{x}^*$ denote the deformed location of the tip for the starred deformation. Denoting the amount of translation by $\boldsymbol{c}$, we note that the location of the spring tip for the starred deformation is given by

$$\boldsymbol{x}^* = \boldsymbol{x} + \boldsymbol{c} \tag{15.23}$$

Both scenarios involve the same *starting* configuration, so $\boldsymbol{x}_o$ for the starred deformation is the same as $\boldsymbol{x}_o$ for the fiducial deformation. It is common practice to denote all variables used in the starred deformation variable by an asterisk. Hence,

$$\boldsymbol{x}^*_o = \boldsymbol{x}_o \tag{15.24}$$

The principle of invariance under translation says that both scenarios are *supposed to involve the same spring force.* Thus, we require that our spring model should predict $\boldsymbol{f}$ to be the same for both the starred and fiducial deformations:

$$\boldsymbol{f}^*_{\text{desired}} = \boldsymbol{f} \tag{15.25}$$

We demand that this condition should hold no matter how much translation we use. To demonstrate that Eq. (15.22) violates translational frame invariance, we simply apply it in both scenarios:

For the fiducial deformation, $\boldsymbol{f} = k(\boldsymbol{x} - \boldsymbol{x}_o)$ (15.26)

For starred deformation, the spring model predicts

$$\boldsymbol{f}^*_{\text{predicted}} = k(\boldsymbol{x}^* - \boldsymbol{x}^*_o) \tag{15.27}$$

To check whether or not this prediction for $\boldsymbol{f}^*$ agrees with the desired result given in Eq. (15.25), we substitute Eqs. (15.23) and (15.24) into Eq. (15.27) to obtain

* a trillion times per second!

$$\underset{\sim}{f}^*_{\text{predicted}} = k(\underset{\sim}{x} + \underset{\sim}{c} - \underset{\sim}{x}_o) = \underset{\sim}{f}^*_{\text{desired}} + k\underset{\sim}{c} \tag{15.28}$$

We want our spring model to give the correct answer for both the fiducial and starred deformations. However, the only way that $\underset{\sim}{f}^*_{\text{predicted}}$ will equal $\underset{\sim}{f}^*_{\text{desired}}$ is if the translation $\underset{\sim}{c}$ is zero. Thus, our undergraduate spring model of Eq. (15.22) should be applied only under the restriction of zero spring translation. For general motions, it violates the principle of translational frame indifference.

The above discussion captures the flavor of invariance derivations. We have selected a simple and obviously wrong spring equation to permit you to better understand the precise (but stilted and often non-intuitive) *proof* that the model is wrong. Flaws in real-life models are not always so intuitively apparent, and the abstract mathematical analysis might be the only sensible way to verify translational frame indifference.

To summarize, analysis of translational frame indifference goes as follows:

- Use kinematics and/or careful application of definitions to deduce how the independent variables in the translated "starred" deformation must be related to those of the fiducial deformation. For our spring model, for example, we asserted that $\underset{\sim}{x}^* = \underset{\sim}{x} + \underset{\sim}{c}$. Keep in mind: superimposing translation affects only the deformed state — we do *not* impose a translation on the initial state. That's why we stated that $\underset{\sim}{x}^*_o = \underset{\sim}{x}_o$ to reflect the fact that both deformations started from the *same* initial condition. The fact that the fiducial and starred deformations have the same initial state is the key difference that distinguishes frame invariance from coordinate invariance.* If the model is written in rate form, then the kinematical condition that relates the starred velocity to the fiducial velocity would be $\underset{\sim}{v}^* = \underset{\sim}{v} + \dot{\underset{\sim}{c}}$.
- Apply the principle of material frame indifference to decide how the *dependent* variables for the starred deformation should be related to those of the fiducial deformation. For our spring model, we asserted the *physical* condition that the spring force should be unchanged by rigid translation ($\underset{\sim}{f}^* = \underset{\sim}{f}$).
- Apply the material constitutive model for both the fiducial and starred deformations. The model is translationally invariant if consistency of the fiducial and starred model predictions requires no restrictions on the frame translation vector $\underset{\sim}{c}$. For our undergraduate spring model, we

* The principle of translational coordinate invariance says that we should be able to translate the origin to a new location without affecting the prediction of the model. If the origin were moved by an amount $\underset{\sim}{C}$, then, $\underset{\sim}{x}_o$ would become $\underset{\sim}{x}_o' = \underset{\sim}{x}_o - \underset{\sim}{C}$ and $\underset{\sim}{x}$ would become $\underset{\sim}{x}' = \underset{\sim}{x} - \underset{\sim}{C}$. Consequently, $\underset{\sim}{x}' - \underset{\sim}{x}_o' = \underset{\sim}{x} - \underset{\sim}{x}_o$, thereby demonstrating that the model would give the same answer for spring force even if the origin were to be moved. Thus, our spring model *does* satisfy translational coordinate invariance even though it violates translational frame invariance.

proved that the fiducial and starred spring forces were consistent only if $\underset{\sim}{\boldsymbol{c}}=\underset{\sim}{\boldsymbol{0}}$, which therefore proved that the model was not translationally frame invariant. To be translationally invariant, the fiducial force needed to be equal to the starred force *regardless of the value of* $\underset{\sim}{\boldsymbol{c}}$.

Again, keep in mind that we knew *intuitively* that something was wrong with Eq. (15.22) just by considering Fig. 15.1. The hard part was *proving it mathematically*! Frame indifference analysis is really all about enforcing what we intuitively know should be true. The problem is that our dumb computers only do what we tell them to do — they lack intuition, so we must be very careful when developing our numerical models.

Rotational invariance.

To satisfy *translation* invariance, a masters-level student might propose a new improved version of Eq. (15.22). Namely,

$$\underset{\sim}{f} = k(\underset{\sim}{L} - \underset{\sim}{L}_o) \qquad \leftarrow \text{better, but still bad} \tag{15.29}$$

Here, $\underset{\sim}{L}$ is the vector pointing from the tail of the spring to the tip of the spring. Since this vector represents the difference between two points on the spring, it will be invariant under a superimposed translation. Thus, under translation, $\underset{\sim}{L}^*=\underset{\sim}{L}$. We again have the restriction that $\underset{\sim}{f}^*=\underset{\sim}{f}$ and $\underset{\sim}{L}_o^* = \underset{\sim}{L}_o$. If Eq. (15.29) is assumed true, then the equation $\underset{\sim}{f}^* = k(\underset{\sim}{L}^* - \underset{\sim}{L}^*_o)$ is also true without having to impose any assumptions about the translation vector $\underset{\sim}{c}$. Consequently, Eq. (15.29) is *translationally* frame invariant.

Unfortunately, the spring model of Eq. (15.29) still has problems because it is not *rotationally* frame invariant. To understand the issue, consider Fig. 15.2 where we show the same amount of spring elongation, but a different amount of spring rotation. For convenience, we artificially selected a rotation that would make the vector $\underset{\sim}{L} - \underset{\sim}{L}_o$ point straight down. If we were to blindly program Eq. (15.29) into a dumb computer, then it would predict that the force in the spring is oriented vertically. As human beings, we know *intuitively* that this is an absurd result — the force in the spring should point along the axis of the spring for *both* the fiducial and starred deformations.

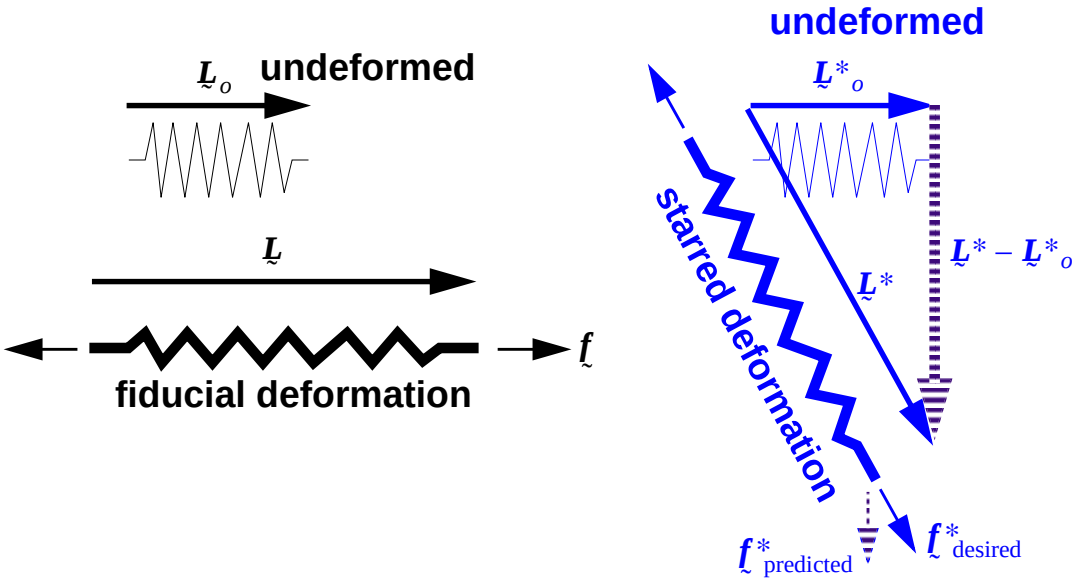


Figure 15.2. Illustration of a rotationally faulty model. On the left is a deformation in which the orientation of the spring remains fixed. On the right is a deformation in which the spring experiences the *same elongation* but it is also rotated. To be rotationally consistent with the fiducial deformation, the starred force vector $\underset{\sim}{f}^*$ needs to be the same as the fiducial $\underset{\sim}{f}$ except rotated with the spring. For the picture on the right, we contrived the amount of rotation so that the vector $\underset{\sim}{L} - \underset{\sim}{L}_o$ would point straight down. That means our faulty masters-level spring equation predicts a spring force that is vertical instead of lined up with the spring the way it should be!

Following the same sort of logic as we used for translational invariance, we will now prove *mathematically* that Eq. (15.29) is not rotationally invariant. The idea is to again to check whether or not the predictions of the model for the starred rotated deformation are physically consistent with the prediction in the fiducial deformation. Again consider two scenarios:

fiducial deformation: stretch the spring without rotation

starred deformation: apply the same deformation as the fiducial deformation, but also rigidly rotate the spring.

Both scenarios involve the same starting length vector. Thus,

$$\underset{\sim}{\boldsymbol{L}}_o^* = \underset{\sim}{\boldsymbol{L}}_o \tag{15.30}$$

Let $\underset{\sim}{\boldsymbol{L}}$ denote the deformed length vector for the fiducial deformation. Let $\underset{\sim}{\boldsymbol{L}}^*$ denote the deformed length vector for the starred deformation. Let $\underset{\approx}{\boldsymbol{Q}}$ denote the rotation tensor. Then simple kinematics reveals that

$$\underset{\sim}{\boldsymbol{L}}^* = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{L}} \tag{15.31}$$

The principle of invariance under rotation imposes the intuitively "obvious" requirement that the spring force predicted for the starred deformation should be the same as the fiducial force except rotated with the material. Stated differently, the force in the spring should form the same angle *relative* to the spring for both scenarios. Some authors would state this property by saying that, under rigid rotation, the force "co-rotates" with the spring. Mathematically, this requirement is expressed by asserting that

$$\underset{\sim}{\boldsymbol{f}}^*_{\text{desired}} = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{f}} \tag{15.32}$$

To demonstrate that Eq. (15.29) violates rotational frame invariance, we will apply it in both scenarios and demonstrate that $\underset{\sim}{\boldsymbol{f}}^*_{\text{predicted}} \neq \underset{\sim}{\boldsymbol{f}}^*_{\text{desired}}$. The spring model says that $\underset{\sim}{\boldsymbol{f}} = k(\underset{\sim}{\boldsymbol{L}} - \underset{\sim}{\boldsymbol{L}}_o)$ and therefore Eq. (15.32) becomes

$$\underset{\sim}{\boldsymbol{f}}^*_{\text{desired}} = k(\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{L}} - \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{L}}_o) \tag{15.33}$$

For the starred (rotated) deformation, the spring model predicts $\underset{\sim}{\boldsymbol{f}}^*_{\text{predicted}} = k(\underset{\sim}{\boldsymbol{L}}^* - \underset{\sim}{\boldsymbol{L}}_o^*)$. Hence, substituting Eqs. (15.30) and (15.31) gives

$$\text{For the starred deformation, } \underset{\sim}{\boldsymbol{f}}^*_{\text{predicted}} = k(\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{L}} - \underset{\sim}{\boldsymbol{L}}_o) \tag{15.34}$$

To satisfy rotational frame indifference, $\underset{\sim}{\boldsymbol{f}}^*_{\text{predicted}}$ must be the same as $\underset{\sim}{\boldsymbol{f}}^*_{\text{desired}}$. However, subtracting Eqs. (15.33) from (15.34) gives

$$\underset{\sim}{\boldsymbol{f}}^*_{\text{predicted}} - \underset{\sim}{\boldsymbol{f}}^*_{\text{desired}} = k(\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{L}}_o - \underset{\sim}{\boldsymbol{L}}_o) \tag{15.35}$$

The only way that the above result will equal zero (as required for rotational frame invariance) is if the rotation tensor equals the identity tensor. In other words, there must be no rotation. We therefore conclude that the our masters-level spring model violates rotational frame invariance. We desire our spring model to give the correct answer for both the fiducial and the starred scenarios, so we now look into a Ph.D. modification of the spring model.

Here's a model that satisfies the principle of material frame invariance.

To satisfy both translational and rotational invariance, a Ph.D-level student might propose a more improved version of Eq. (15.29). Namely,

$$\underset{\sim}{\boldsymbol{f}} = k\delta\underset{\sim}{\boldsymbol{n}},$$
$$\text{where } \delta \equiv \|\underset{\sim}{\boldsymbol{L}}\| - \|\underset{\sim}{\boldsymbol{L}}_o\|, \text{ and } \underset{\sim}{\boldsymbol{n}} = \frac{\underset{\sim}{\boldsymbol{L}}}{\|\underset{\sim}{\boldsymbol{L}}\|} \tag{15.36}$$

To prove that this equation is indeed both translationally and rotationally invariant, we again consider how the prediction would change if we were to consider a "starred" deformation that superimposes a rigid translation and rotation on the deforming material. Keep in mind that the only difference between the fiducial and starred deformations is a rigid motion. Therefore, lengths are the same in both cases. Hence, $\|\underset{\sim}{\boldsymbol{L}}^*\| = \|\underset{\sim}{\boldsymbol{L}}\|$. Of course, as before, the initial state is the same for both cases, so $\|\underset{\sim}{\boldsymbol{L}}_o{}^*\| = \|\underset{\sim}{\boldsymbol{L}}_o\|$. Consequently,

$$\delta^* = \delta. \tag{15.37}$$

Under a rigid motion, the "starred" length vector $\underset{\sim}{\boldsymbol{L}}^*$ is given by $\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{L}}$. Thus, the "starred" unit vector is related to the un-starred unit vector by a simple rotation

$$\underset{\sim}{\boldsymbol{n}}^* = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{n}} \tag{15.38}$$

As before, we require that the force for the starred deformation should be related to the fiducial force by

$$\underset{\sim}{\boldsymbol{f}}^*_{\text{desired}} = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{f}} = k\delta\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{n}} \tag{15.39}$$

If we apply the improved spring model of Eq. (15.36) to the starred deformation, we obtain

$$\underset{\sim}{\boldsymbol{f}}^*_{\text{predicted}} = k\delta^*\underset{\sim}{\boldsymbol{n}}^* \tag{15.40}$$

In light of Eqs. (15.37) and (15.38), we conclude that this new spring model is frame indifferent because we finally have consistent results. Namely,

$$\underset{\sim}{\boldsymbol{f}}^*_{\text{predicted}} = \underset{\sim}{\boldsymbol{f}}^*_{\text{desired}} \tag{15.41}$$

Is frame indifference all you need to make a model good for large deformations?

A point of caution is in order here. For some reason, many people think that satisfaction of frame indifference guarantees good performance of a model under large deformations. This is false. Frame indifference merely demands that the predictions for the fiducial and starred deformations should be *consistent* with each other. There is no guarantee that the fiducial prediction is any good in the first place!

A practicing engineer might well reject our Ph.D student's beautifully frame indifferent spring model if the spring being modelled is actually nonlinear. Frame indifferent models are everywhere, but ones that accurately describe a material's nonlinear large distortion response are rare indeed.

The principle of material frame indifference in general

When you want to test a material constitutive equation for frame indifference, you must first note what variables are involved in the constitutive equation. You must use physical reasoning to assert how those quantities are *supposed to* change upon a superimposed rigid motion. Then you must demonstrate that the constitutive model gives consistent results with and without superimposed motion. This self-consistency must be achieved without imposing *any* conditions on the superimposed rigid motion or on the *rate* of the rigid motion. We have illustrated these concepts in the simple setting of a spring model where the nature of the problems was quite obvious. When getting into real material constitutive laws, the issues are not so obvious and one must then rely exclusively on the mathematics.

Objectivity transformation tables. By now, you have likely realized that testing for frame indifference requires that you know kinematics well enough to determine how a variable will change from its fiducial value to its starred value in response to a superimposed rigid motion. Such changes are called **objectivity transformations**, and it is a good idea for you to simply maintain an ongoing table of such relationships.

For example, under general rotation and translation, the position vector $\underset{\sim}{x}^*$ in the starred configuration is related to the position vector $\underset{\sim}{x}$ in the fiducial configuration by

$$\boxed{\underset{\sim}{x}^* = \underset{\approx}{Q} \bullet \underset{\sim}{x} + \underset{\sim}{c}}, \tag{15.42}$$

where $\underset{\approx}{Q}$ is the rigid rotation and $\underset{\sim}{c}$ is the rigid translation.

When we discussed the spring example, recall that the *initial* spring vector $\underset{\sim}{L}_o$ was the same for both the starred and fiducial deformations. They both started from the same initial configuration — only the starred deformation suffered additional rotation and translation. This is one example of the fact

that some vectors will *not* change at all under rigid motion. For continuous motions, the initial position vector $\underset{\sim}{X}$ is the same for both the fiducial and starred deformations. Thus, since both the fiducial and starred deformations begin from the same initial configuration, we conclude that

$$\boxed{\underset{\sim}{X}^* = \underset{\sim}{X}} \tag{15.43}$$

The deformation gradient for the fiducial deformation is defined as usual by

$$\underset{\approx}{F} = \frac{\partial \underset{\sim}{x}}{\partial \underset{\sim}{X}} \tag{15.44}$$

Likewise, the deformation gradient for the starred deformation is defined by

$$\underset{\approx}{F}^* = \frac{\partial \underset{\sim}{x}^*}{\partial \underset{\sim}{X}^*} \tag{15.45}$$

Applying the chain rule,

$$\underset{\approx}{F}^* = \frac{\partial \underset{\sim}{x}^*}{\partial \underset{\sim}{x}} \bullet \frac{\partial \underset{\sim}{x}}{\partial \underset{\sim}{X}} \bullet \frac{\partial \underset{\sim}{X}}{\partial \underset{\sim}{X}^*} \tag{15.46}$$

From Eq. (15.42), we know that

$$\frac{\partial \underset{\sim}{x}^*}{\partial \underset{\sim}{x}} = \frac{\partial (\underset{\approx}{Q} \bullet \underset{\sim}{x} + \underset{\sim}{c})}{\partial \underset{\sim}{x}} = \underset{\approx}{Q}, \tag{15.47}$$

where we have used the fact that $\underset{\approx}{Q}$ and $\underset{\sim}{c}$ are rigid rotation and translation vectors and they consequently are the same throughout space — they do not vary with $\underset{\sim}{x}$.

From Eq. (15.43), we know that

$$\frac{\partial \underset{\sim}{X}}{\partial \underset{\sim}{X}^*} = \underset{\approx}{I} \tag{15.48}$$

Thus, substituting Eqs. (15.44), (15.47) and (15.48) into Eq. (15.46) gives

$$\boxed{\underset{\approx}{F}^* = \underset{\approx}{Q} \bullet \underset{\approx}{F}} \tag{15.49}$$

This unsurprising result says that the starred deformation gradient is obtained by operating on the fiducial deformation gradient by the rotation tensor.

The right Cauchy-Green tensor is defined for the fiducial deformation by

$$\underset{\approx}{C} = \underset{\approx}{F}^T \bullet \underset{\approx}{F} \tag{15.50}$$

The right Cauchy-Green tensor for the starred deformation is

$$\underset{\approx}{C}^* = \underset{\approx}{F}^{*T} \bullet \underset{\approx}{F}^* \tag{15.51}$$

Substituting Eq. (15.49) into (15.51) and recalling that $\underset{\approx}{\boldsymbol{Q}}$ is orthogonal reveals that

$$\underset{\approx}{\boldsymbol{C}}^* = \underset{\approx}{\boldsymbol{C}} \tag{15.52}$$

The fiducial and starred deformations have the same stretches before rotation is applied. Hence, it should not be surprising that the fiducial and starred deformations have the same right Cauchy-Green tensors because this tensor is a measure of the material stretching *before* any material rotation is applied. By contrast, consider the left Cauchy-Green tensor:

$$\underset{\approx}{\boldsymbol{B}} = \underset{\approx}{\boldsymbol{F}} \bullet \underset{\approx}{\boldsymbol{F}}^T \tag{15.53}$$

For the starred deformation, we have

$$\underset{\approx}{\boldsymbol{B}}^* = \underset{\approx}{\boldsymbol{F}}^* \bullet \underset{\approx}{\boldsymbol{F}}^{*T} \tag{15.54}$$

Substituting Eq. (15.49) into (15.54) reveals that

$$\boxed{\underset{\approx}{\boldsymbol{B}}^* = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{B}} \bullet \underset{\approx}{\boldsymbol{Q}}^T} \tag{15.55}$$

The boxed equations within this section required a fair amount of effort to derive, and these results are often needed when performing frame indifference calculations. Therefore, it is highly recommended that you create and continually expand your own table of so-called "objectivity transformations" that show how a quantity in the starred configuration is related to the fiducial deformation. One such table is available in the nomenclature list at

http://www.me.unm.edu/~rmbrann/gobag.html.

A key lesson here is that the starred quantities change in different ways depending on their physical definitions. Some tensors don't change at all, while others "co-rotate" with the superimposed rotation. The fact that tensors change in different ways depending on their physical definitions is a key feature that distinguishes objectivity transformations from ordinary basis transformations. For a basis transformation, tensors themselves do not change, and their matrix of tensor components will transform in exactly the same way regardless of the physical meaning of the tensor. For an objectivity transformation, the physical meaning of the tensor is important.

Reference and spatial tensors. Referring to Eq. (15.52), note that the right stretch tensor is the *same* in both the fiducial and starred configurations:

$$\underset{\approx}{\boldsymbol{U}}^* = \underset{\approx}{\boldsymbol{U}} \tag{15.56}$$

by contrast, referring to Eq. (15.55), the left stretch changes according to

$$\underset{\approx}{\boldsymbol{V}}^* = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{V}} \bullet \underset{\approx}{\boldsymbol{Q}}^T \tag{15.57}$$

Any tensor such as $\underset{\approx}{\boldsymbol{U}}$ or $\underset{\approx}{\boldsymbol{C}}$ that is the same for both the fiducial and starred configurations is referred to as a **reference tensor**. Any tensor $\underset{\approx}{\boldsymbol{A}}$ that transforms according to

$$\underset{\approx}{\boldsymbol{A}}^* = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{A}} \bullet \underset{\approx}{\boldsymbol{Q}}^T \tag{15.58}$$

is called a **spatial tensor**. The term **"objective tensor"** is also used. Thus, for example, $\underset{\approx}{\boldsymbol{B}}$ and $\underset{\approx}{\boldsymbol{V}}$ are spatial tensors. If B_{ij} are the components of the fiducial $\underset{\approx}{\boldsymbol{B}}$ tensor with respect to the laboratory basis $\{\underset{\sim}{\boldsymbol{E}}_1, \underset{\sim}{\boldsymbol{E}}_2, \underset{\sim}{\boldsymbol{E}}_3\}$, then the starred $\underset{\approx}{\boldsymbol{B}}^*$ tensor has the *same* components except referenced to a rotated basis, $\underset{\sim}{\boldsymbol{e}}_k = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{E}}_k$. In this sense, a spatial tensor is one that "co-rotates" under a superimposed rotation. By contrast, a reference tensor remains unchanged.

Many tensors are neither reference nor spatial tensors. For example, the transformation result of Eq. (15.49) does not match the structure of either Eq. (15.56) or (15.57). Consequently, the deformation gradient is neither spatial nor referential.

A vector (such as the initial position vector $\underset{\sim}{\boldsymbol{X}}$) that is unaffected by superimposed rotation is called a **reference vector**. On the other hand, a vector $\underset{\sim}{\boldsymbol{z}}$ is **spatial** if it transforms according to

$$\underset{\sim}{\boldsymbol{z}}^* = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{z}} \tag{15.59}$$

Consider a vector $\Delta\underset{\sim}{\boldsymbol{x}}$ that connects two material points:

$$\Delta\underset{\sim}{\boldsymbol{x}} = \underset{\sim}{\boldsymbol{x}}_2 - \underset{\sim}{\boldsymbol{x}}_1 \tag{15.60}$$

In the starred configuration, this material vector becomes

$$(\Delta\underset{\sim}{\boldsymbol{x}})^* = \underset{\sim}{\boldsymbol{x}}^*_2 - \underset{\sim}{\boldsymbol{x}}^*_1 \tag{15.61}$$

or, using Eq. (15.42),

$$(\Delta\underset{\sim}{\boldsymbol{x}})^* = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{x}}_2 + \underset{\sim}{\boldsymbol{c}} - (\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{x}}_1 + \underset{\sim}{\boldsymbol{c}}) \tag{15.62}$$

Using Eq. (15.60), we conclude that

$$\boxed{(\Delta\underset{\sim}{\boldsymbol{x}})^* = \underset{\approx}{\boldsymbol{Q}} \bullet \Delta\underset{\sim}{\boldsymbol{x}}} \tag{15.63}$$

Thus, this sort of material fiber is a *spatial* vector — it co-rotates with the material.

Consider displacement $\underset{\sim}{\boldsymbol{u}}$, which is defined by

$$\underset{\sim}{\boldsymbol{u}} = \underset{\sim}{\boldsymbol{x}} - \underset{\sim}{\boldsymbol{X}} \tag{15.64}$$

For the starred deformation,

$$\boxed{\underset{\sim}{\boldsymbol{u}}^* = \underset{\sim}{\boldsymbol{x}}^* - \underset{\sim}{\boldsymbol{X}}^* = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{x}} + \underset{\sim}{\boldsymbol{c}} - \underset{\sim}{\boldsymbol{X}} = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{u}} + \underset{\sim}{\boldsymbol{c}} + (\underset{\approx}{\boldsymbol{Q}} - \underset{\approx}{\boldsymbol{I}}) \bullet \underset{\sim}{\boldsymbol{X}}} \tag{15.65}$$

Thus, displacement is *neither* a reference vector nor a spatial vector. Again, we want to emphasize that relationships like the boxed equations above are tedious to derive. You should add objectivity transformations to your own personal collection for future use during frame indifference analyses.

You can also define reference and spatial *scalars*. Many researchers mistakenly jump to the conclusion that scalars are unaffected by superimposed rotations. This is false, as we now will show. Consider the polar rotation tensor

$$\underset{\approx}{\boldsymbol{R}} = \underset{\approx}{\boldsymbol{F}} \bullet \underset{\approx}{\boldsymbol{U}}^{-1} \tag{15.66}$$

In the starred configuration,

$$\underset{\approx}{\boldsymbol{R}}^{*} = \underset{\approx}{\boldsymbol{F}}^{*} \bullet \underset{\approx}{\boldsymbol{U}}^{*-1} \tag{15.67}$$

Using Eqs. (15.49) and (15.56),

$$\underset{\approx}{\boldsymbol{R}}^{*} = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{F}} \bullet \underset{\approx}{\boldsymbol{U}}^{-1}. \tag{15.68}$$

Thus, substituting Eq. (15.66) gives

$$\boxed{\underset{\approx}{\boldsymbol{R}}^{*} = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{R}}} \tag{15.69}$$

which shows that $\underset{\approx}{\boldsymbol{R}}$ is neither spatial nor objective. Now let's consider the rotation angle α associated with $\underset{\approx}{\boldsymbol{R}}$. Based on the Euler-Rodrigues formula, the cosine of the rotation angle is given by

$$\cos\alpha = \frac{1}{2}(\text{trace}\,\underset{\approx}{\boldsymbol{R}} - 1) \tag{15.70}$$

Under superimposed rotation,

$$\cos\alpha^{*} = \frac{1}{2}(\text{trace}\,\underset{\approx}{\boldsymbol{R}}^{*} - 1) = \frac{1}{2}(\text{trace}(\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{R}}) - 1) \tag{15.71}$$

From this, we conclude that

$$\alpha^{*} \neq \alpha \tag{15.72}$$

This proves that not all scalars are unaffected by superimposed rigid rotation. As with tensors and vectors, the physical definition of the scalar must be considered in deciding objectivity.

Stress and strain measures. The most common choice for a stress measure is the Cauchy stress $\underset{\approx}{\boldsymbol{\sigma}}$, which is defined such that the force on an area element $\underset{\sim}{\boldsymbol{n}}\,dA$ is given by $\underset{\approx}{\boldsymbol{\sigma}} \bullet \underset{\sim}{\boldsymbol{n}}\,dA$. In this definition, the normal to the patch of area is a spatial vector, so it satisfies

$$\boxed{\underset{\sim}{\boldsymbol{n}}^{*} = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{n}}} \tag{15.73}$$

The principle of material frame indifference says that, upon a rigid rotation, the traction (force per unit area) should co-rotate with the material. Consequently, the Cauchy stress must be a spatial tensor:

$$\underset{\approx}{\sigma}^* = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\sigma} \bullet \underset{\approx}{\boldsymbol{Q}}^T \tag{15.74}$$

The Cauchy stress measure is not the only stress measure used in the literature. Many researchers employ the **second Piola-Kirchhoff stress** (PK2) tensor, defined by

$$\underset{\approx}{\bar{\boldsymbol{s}}} = J \underset{\approx}{\boldsymbol{F}}^{-1} \bullet \underset{\approx}{\sigma} \bullet \underset{\approx}{\boldsymbol{F}}^{-T}, \tag{15.75}$$

where J is the Jacobian of the deformation,

$$J = \det \underset{\approx}{\boldsymbol{F}} \tag{15.76}$$

Upon a superimposed rotation,

$$J^* = \det \underset{\approx}{\boldsymbol{F}}^* = \det(\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{F}}) = \det(\underset{\approx}{\boldsymbol{Q}}) \det(\underset{\approx}{\boldsymbol{F}}) = J \tag{15.77}$$

Also

$$\underset{\approx}{\bar{\boldsymbol{s}}} = J^* \underset{\approx}{\boldsymbol{F}}^{*-1} \bullet \underset{\approx}{\sigma}^* \bullet \underset{\approx}{\boldsymbol{F}}^{*-T} \tag{15.78}$$

Substituting the objectivity transformation relations for J, $\underset{\approx}{\boldsymbol{F}}$ and $\underset{\approx}{\sigma}$ reveals that

$$\underset{\approx}{\bar{\boldsymbol{s}}}^* = \underset{\approx}{\bar{\boldsymbol{s}}} \tag{15.79}$$

Thus, *the second Piola Kirchhoff (PK2) stress is a reference stress tensor* — it is unaffected by a superimposed rigid motion.

The Cauchy stress $\underset{\approx}{\sigma}$ is said to be "conjugate" to the symmetric part of the velocity gradient $\underset{\approx}{\boldsymbol{D}}$ because the stress power per unit mass is given by

$$P_s = \frac{1}{\rho} \underset{\approx}{\sigma} : \underset{\approx}{\boldsymbol{D}} \tag{15.80}$$

The symmetric part of the velocity gradient can be shown to be a spatial tensor; i.e.,

$$\underset{\approx}{\boldsymbol{D}}^* = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{D}} \bullet \underset{\approx}{\boldsymbol{Q}}^T \tag{15.81}$$

The mass density ρ is a reference scalar (i.e., it is unchanged by superimposed rotation). Note that this does *not* mean that the mass density ρ is equal to its initial value ρ_o — it only means that it is unaffected by superimposed rigid rotation. P_s is itself a reference scalar. By using the definition of the second Piola-Kirchhoff (PK2) stress, it can be shown that there exists a strain $\underset{\approx}{\bar{\varepsilon}}$ that is conjugate to the PK2 stress in the sense that

$$P_s = \frac{1}{\rho} \underset{\approx}{\bar{\boldsymbol{s}}} : \dot{\underset{\approx}{\bar{\varepsilon}}} \tag{15.82}$$

Specifically, this special strain — called the Lagrange strain — is defined

$$\bar{\underset{\approx}{\varepsilon}} = \frac{1}{2}(\underset{\approx}{\boldsymbol{C}} - \underset{\approx}{\boldsymbol{I}}) \tag{15.83}$$

Recalling that $\underset{\approx}{\boldsymbol{C}}$ is a reference tensor, we immediately see that $\bar{\underset{\approx}{\varepsilon}}$ is also a reference tensor. That is,

$$\bar{\underset{\approx}{\varepsilon}}^* = \bar{\underset{\approx}{\varepsilon}} \tag{15.84}$$

Many researchers elect to phrase their constitutive equations in terms of the Lagrange strain and the PK2 stress because they are both reference tensors, and the principle of material frame indifference is therefore automatically satisfied. The principle of material frame indifference has its most dramatic impact on constitutive models that use spatial tensors.

Example: elasticity. In its most general sense, a material is usually said to be elastic if the stress in the material depends only on the deformation of the material. This definition is actually a sloppy because it might violates frame indifference when taken literally. Specifically, let $\underset{\approx}{\sigma}$ denote Cauchy stress and let $\underset{\approx}{\boldsymbol{F}}$ denote the deformation gradient tensor. According to the above sloppy definition of elasticity, the constitutive law may be written

$$\underset{\approx}{\sigma} = f(\underset{\approx}{\boldsymbol{F}}) \tag{15.85}$$

where f is the constitutive function. Physically, we know that application of a superimposed rigid motion *should* change the stress to

$$\underset{\approx}{\sigma}^* = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\sigma} \bullet \underset{\approx}{\boldsymbol{Q}}^T \tag{15.86}$$

Furthermore, when we have a deformation $\underset{\approx}{\boldsymbol{F}}$ and we apply a superimposed rotation $\underset{\approx}{\boldsymbol{Q}}$, then the new deformation gradient is

$$\underset{\approx}{\boldsymbol{F}}^* = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{F}} \tag{15.87}$$

When applied under a superimposed rigid motion, the constitutive law of Eq. (15.85) gives

$$\underset{\approx}{\sigma}^* = f(\underset{\approx}{\boldsymbol{F}}^*) \tag{15.88}$$

or

$$\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\sigma} \bullet \underset{\approx}{\boldsymbol{Q}}^T = f(\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{F}}) \tag{15.89}$$

The principle of material frame indifference demands that both Eq. (15.85) and (15.89) must be true. Thus, we may substitute Eq. (15.85) into (15.89) to obtain

$$\underset{\approx}{\boldsymbol{Q}} \bullet f(\underset{\approx}{\boldsymbol{F}}) \bullet \underset{\approx}{\boldsymbol{Q}}^T = f(\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{F}}),$$

which must hold for all rotations $\underset{\approx}{\boldsymbol{Q}}$ and deformations $\underset{\approx}{\boldsymbol{F}}$ (15.90)

The above equation represents a restriction on admissible forms for the function f. Any proposed elastic constitutive equation must satisfy the above restriction.

Consider the following special case: many authors assume that stress can be written as a function of spatial strain:

$$\underset{\approx}{\sigma} = g(\underset{\approx}{\varepsilon}) \tag{15.91}$$

The term "spatial" is used to describe any tensor that transforms according to

$$\underset{\approx}{\boldsymbol{A}}^* = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\boldsymbol{A}} \bullet \underset{\approx}{\boldsymbol{Q}}^T \tag{15.92}$$

upon a superimposed rigid rotation $\underset{\approx}{\boldsymbol{Q}}$. Referring to Eq. (15.86), we note that Cauchy stress is a spatial tensor. Eq. (15.87) shows that the deformation gradient is *not* a spatial tensor. If our strain measure is spatial, then we know that

$$\underset{\approx}{\varepsilon}^* = \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\varepsilon} \bullet \underset{\approx}{\boldsymbol{Q}}^T \tag{15.93}$$

Eq. (15.91) satisfies frame indifference only if

$$\underset{\approx}{\sigma}^* = g(\underset{\approx}{\varepsilon}^*) \tag{15.94}$$

In other words,

$$\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\sigma} \bullet \underset{\approx}{\boldsymbol{Q}}^T = g(\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\varepsilon} \bullet \underset{\approx}{\boldsymbol{Q}}^T) \tag{15.95}$$

To be consistent with Eq. (15.91), this implies that

$$\underset{\approx}{\boldsymbol{Q}} \bullet g(\underset{\approx}{\varepsilon}) \bullet \underset{\approx}{\boldsymbol{Q}}^T = g(\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\varepsilon} \bullet \underset{\approx}{\boldsymbol{Q}}^T)$$

which must hold for all rotations $\underset{\approx}{\boldsymbol{Q}}$ and spatial strains $\underset{\approx}{\varepsilon}$ (15.96)

The above equation imposes a very restrictive admissibility constraint on the constitutive function g. In particular, the above restriction can be satisfied only if the material is isotropic! To see why this is so, let's suppose to the contrary that the material is anisotropic such that it is very stiff in the 1-direction and very compliant in the 2-direction. Now consider a fiducial strain $\underset{\approx}{\varepsilon}$ that is uniaxial in the 1-direction. This will produce a very large stress in the 1-direction because the material is stiff in that direction. If we were to apply the same magnitude of strain in the 2-direction, we would not expect as large of a stress. However, the constraint of Eq. (15.96) does not permit this desired behavior. If $\underset{\approx}{\boldsymbol{Q}}$ represents a rotation of 90 degrees about the 3-direction, then $\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\varepsilon} \bullet \underset{\approx}{\boldsymbol{Q}}^T$ is the same as our fiducial strain except that it is now applied in the 2-direction. Note that $g(\underset{\approx}{\varepsilon})$ represents the stress for our fiducial strain and $g(\underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\varepsilon} \bullet \underset{\approx}{\boldsymbol{Q}}^T)$ represents the stress for our rotated strain. Equation (15.96) says that the stress for the rotated strain should be identical to the stress for the fiducial strain except that it must be rotated by 90 degrees. Consequently, Eq. (15.96) does not permit the magnitude of stress to be smaller for strains pointing in the 2-direction.

If your material is isotropic, then PMFI permits you to use a constitutive model of the form in Eq. (15.91). However, if you are modelling an anisotropic material, then PMFI tells you that your constitutive equation cannot be expressed in the form in Eq. (15.91). How do we use this information? The proper conclusion is that, for anisotropic media, a spatial constitutive equation must depend on *more* than just the spatial strain. If, for example, your material is transversely isotropic, then PMFI will permit you to propose a constitutive model of the form

$$\underset{\approx}{\sigma} = g(\underset{\approx}{\varepsilon}, \underset{\sim}{n}), \tag{15.97}$$

where $\underset{\sim}{n}$ is the axis of transverse symmetry. By including the material direction as an independent variable in the constitutive model, you will be able to assert that $\underset{\sim}{n}^* = \boldsymbol{Q} \bullet \underset{\sim}{n}$, which in turn permits the anisotropic stiffnesses to rotate with the material upon superimposed deformation.

PMFI in rate forms of the constitutive equations

For the purpose of illustration, let's revisit a constitutive law of the following form

$$\underset{\approx}{\sigma} = g(\underset{\approx}{\varepsilon}) \tag{15.98}$$

Taking the time rate of both sides gives

$$\dot{\sigma}_{ij} = E_{ijkl}\dot{\varepsilon}_{kl}, \text{ where } E_{ijkl} \equiv \frac{\partial g(\underset{\approx}{\varepsilon})}{\partial \varepsilon_{kl}} \tag{15.99}$$

For satisfying PMFI, we have already asserted that the material must be isotropic. Hence, the stiffness E_{ijkl} must be isotropic. As long as this is true, the constitutive equation defined by Eq. (15.99) will satisfy frame indifference because it was derived by differentiating a frame indifferent equation.

For elasticity problems, it is common (though we believe inadvisable) to employ a similar-looking but fundamentally *different* constitutive equation

$$\dot{\underset{\approx}{\sigma}} = g(\underset{\approx}{\boldsymbol{D}}) \tag{15.100}$$

where the strain rate has been replaced by the stretching tensor $\underset{\approx}{\boldsymbol{D}}$, which is just the symmetric part of the velocity gradient:

$$\underset{\approx}{\boldsymbol{D}} = \tfrac{1}{2}(\underset{\approx}{\boldsymbol{L}} + \underset{\approx}{\boldsymbol{L}}^T), \text{ where } L_{ij} = \frac{\partial v_i}{\partial x_j} \tag{15.101}$$

We will show that Eq. (15.100) violates PMFI, so it is common practice to propose modifying it as follows:

$$\overset{\circ}{\underset{\approx}{\sigma}} = g(\underset{\approx}{\boldsymbol{D}}), \tag{15.102}$$

where $\overset{\circ}{\underset{\sim}{\sigma}}$ denotes a special "co-rotational" rate that *effectively eliminates the part of the stress rate caused by rotation* as discussed below

A fundamental theorem from continuum mechanics states that the velocity gradient tensor $\underset{\sim}{\boldsymbol{L}}$ is related to the deformation gradient tensor $\underset{\sim}{\boldsymbol{F}}$ by

$$\underset{\sim}{\boldsymbol{L}} = \dot{\underset{\sim}{\boldsymbol{F}}} \bullet \underset{\sim}{\boldsymbol{F}}^{-1} \tag{15.103}$$

Under a superimposed rotation $\underset{\sim}{\boldsymbol{Q}}$, recall that

$$\underset{\sim}{\boldsymbol{F}}^{*} = \underset{\sim}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{F}} \tag{15.104}$$

The starred velocity gradient is

$$\underset{\sim}{\boldsymbol{L}}^{*} = \dot{\underset{\sim}{\boldsymbol{F}}}^{*} \bullet \underset{\sim}{\boldsymbol{F}}^{*-1} = (\dot{\underset{\sim}{\boldsymbol{Q}}} \bullet \underset{\sim}{\boldsymbol{F}} + \underset{\sim}{\boldsymbol{Q}} \bullet \dot{\underset{\sim}{\boldsymbol{F}}}) \bullet (\underset{\sim}{\boldsymbol{F}}^{-1} \bullet \underset{\sim}{\boldsymbol{Q}}^{T}) \tag{15.105}$$

For convenience, we introduce an angular rate of rotation tensor $\underset{\sim}{\Upsilon}$ associated with the $\dot{\underset{\sim}{\boldsymbol{Q}}}$ rate of superimposed rotation. Specifically we define

$$\underset{\sim}{\Upsilon} = \dot{\underset{\sim}{\boldsymbol{Q}}} \bullet \underset{\sim}{\boldsymbol{Q}}^{T} \tag{15.106}$$

Because $\underset{\sim}{\boldsymbol{Q}}$ is a rotation, the tensor $\underset{\sim}{\Upsilon}$ is skew-symmetric:

$$\underset{\sim}{\Upsilon}^{T} = -\underset{\sim}{\Upsilon} \tag{15.107}$$

Using this tensor, we find that the starred velocity gradient of Eq. (15.105) simplifies to

$$\underset{\sim}{\boldsymbol{L}}^{*} = \underset{\sim}{\Upsilon} + \underset{\sim}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{L}} \bullet \underset{\sim}{\boldsymbol{Q}}^{T} \tag{15.108}$$

The presence of $\underset{\sim}{\Upsilon}$ means that the velocity gradient is not a spatial tensor. However, since $\underset{\sim}{\Upsilon}$ is skew-symmetric, we note that the stretching tensor $\underset{\sim}{\boldsymbol{D}} = \text{sym}\underset{\sim}{\boldsymbol{L}}$ is a spatial tensor:

$$\underset{\sim}{\boldsymbol{D}}^{*} = \underset{\sim}{\boldsymbol{Q}} \bullet \underset{\sim}{\boldsymbol{D}} \bullet \underset{\sim}{\boldsymbol{Q}}^{T} \tag{15.109}$$

Recall that the Cauchy stress is a spatial tensor:

$$\underset{\sim}{\sigma}^{*} = \underset{\sim}{\boldsymbol{Q}} \bullet \underset{\sim}{\sigma} \bullet \underset{\sim}{\boldsymbol{Q}}^{T} \tag{15.110}$$

We are now going to prove that this implies that the *rate* of the Cauchy stress is *not* a spatial tensor. The time rate of the starred stress is found by differentiating Eq. (15.110) to give

$$\dot{\underset{\sim}{\sigma}}^{*} = \underset{\sim}{\boldsymbol{Q}} \bullet \dot{\underset{\sim}{\sigma}} \bullet \underset{\sim}{\boldsymbol{Q}}^{T} + \dot{\underset{\sim}{\boldsymbol{Q}}} \bullet \underset{\sim}{\sigma} \bullet \underset{\sim}{\boldsymbol{Q}}^{T} + \underset{\sim}{\boldsymbol{Q}} \bullet \underset{\sim}{\sigma} \bullet \dot{\underset{\sim}{\boldsymbol{Q}}}^{T} \tag{15.111}$$

With the substitution $\dot{\underset{\sim}{\boldsymbol{Q}}} = \underset{\sim}{\Upsilon} \bullet \underset{\sim}{\boldsymbol{Q}}$, this becomes

$$\dot{\underset{\sim}{\sigma}}^{*} = \underset{\sim}{\boldsymbol{Q}} \bullet \dot{\underset{\sim}{\sigma}} \bullet \underset{\sim}{\boldsymbol{Q}}^{T} + \underset{\sim}{\Upsilon} \bullet \underset{\sim}{\boldsymbol{Q}} \bullet \underset{\sim}{\sigma} \bullet \underset{\sim}{\boldsymbol{Q}}^{T} + \underset{\sim}{\boldsymbol{Q}} \bullet \underset{\sim}{\sigma} \bullet \underset{\sim}{\boldsymbol{Q}}^{T} \bullet \underset{\sim}{\Upsilon}^{T} \tag{15.112}$$

Using Eq. (15.111) and recalling that the angular rotation tensor is skew-symmetric, we find that the starred stress *rate* is related to the fiducial stress rate in the following complicated way:

$$\dot{\underset{\sim}{\sigma}}^* = \underset{\sim}{\boldsymbol{Q}} \bullet \dot{\underset{\sim}{\sigma}} \bullet \underset{\sim}{\boldsymbol{Q}}^T + \underset{\sim}{\Upsilon} \bullet \underset{\sim}{\sigma}^* - \underset{\sim}{\sigma}^* \bullet \underset{\sim}{\Upsilon} \tag{15.113}$$

Note that

$$\dot{\underset{\sim}{\sigma}}^* \neq \underset{\sim}{\boldsymbol{Q}} \bullet \dot{\underset{\sim}{\sigma}} \bullet \underset{\sim}{\boldsymbol{Q}}^T \tag{15.114}$$

Therefore the Cauchy stress rate is *not* spatial. This means that the proposed constitutive equation in Eq. (15.100) violates PMFI.

Co-rotational rates (convected, Jaumann, Polar)

To rectify the problem with Eq. (15.100), it is conventional to propose an alternative constitutive equation of the form

$$\overset{\circ}{\underset{\sim}{\sigma}} = g(\underset{\sim}{\boldsymbol{D}}), \tag{15.115}$$

where $\overset{\circ}{\underset{\sim}{\sigma}}$ denotes a so-called "co-rotational" rate defined by

$$\overset{\circ}{\underset{\sim}{\sigma}} = \dot{\underset{\sim}{\sigma}} - \underset{\sim}{\Lambda} \bullet \underset{\sim}{\sigma} - \underset{\sim}{\sigma} \bullet \underset{\sim}{\Lambda}^T \tag{15.116}$$

In this equation, the tensor $\underset{\sim}{\Lambda}$ may be selected to be any convenient physical quantity so long as it transforms under superimposed rotation so that

$$\underset{\sim}{\Lambda}^* = \underset{\sim}{\Upsilon} + \underset{\sim}{\boldsymbol{Q}} \bullet \underset{\sim}{\Lambda} \bullet \underset{\sim}{\boldsymbol{Q}}^T \tag{15.117}$$

Referring to Eq. (15.108), one possible choice for $\underset{\sim}{\Lambda}$ is

$$\underset{\sim}{\Lambda} = \underset{\sim}{\boldsymbol{L}} \qquad \leftarrow \text{ this corresponds to the "convected" rate} \tag{15.118}$$

This is not the only choice. We could alternatively take the co-rotational tensor to be the vorticity tensor:

$$\underset{\sim}{\Lambda} = \underset{\sim}{\boldsymbol{W}} = \frac{1}{2}(\underset{\sim}{\boldsymbol{L}} - \underset{\sim}{\boldsymbol{L}}^T) \qquad \leftarrow \text{ this corresponds to the "Jaumann" rate} \tag{15.119}$$

We could select the co-rotational tensor to equal the polar spin

$$\underset{\sim}{\Lambda} = \underset{\sim}{\Omega} = \dot{\underset{\sim}{\boldsymbol{R}}} \bullet \underset{\sim}{\boldsymbol{R}}^T \qquad \leftarrow \text{ this corresponds to the "Polar" rate} \tag{15.120}$$

Regardless of the choice for the co-rotational tensor, so long as Eq. (15.116) holds, you can demonstrate that

$$\overset{\circ}{\underset{\sim}{\sigma}}^* = \underset{\sim}{\boldsymbol{Q}} \bullet \overset{\circ}{\underset{\sim}{\sigma}} \bullet \underset{\sim}{\boldsymbol{Q}}^T \tag{15.121}$$

In other words, the co-rotational rate is a spatial tensor. Consequently, Eq. (15.115) satisfies PMFI so long as the constitutive function g is isotropic.

Keep in mind that the choice for the co-rotation tensor (i.e., convected, Jaumann, polar, or something else) is not unique. As long as your choice satisfies Eq. (15.117), then your constitutive model will be frame invariant. However, *other* physical arguments *besides* invariance may lead you to prefer one co-rotation tensor over another. For example, Dienes [8] demonstrated quite elegantly that the Jaumann rate predicts physically anomalous oscillatory stress when used with constant isotropic elastic moduli in simple shear, so he recommended the use of the Polar rate. However, Brannon [11] demonstrated that the polar rate is incapable of adequately describing the distortion of directions of material anisotropy, so she cautiously recommended the convected rate (except cast as a Lie derivative — see page 143).

Lie Derivatives and reference configurations

Let $\underset{\sim}{\underset{\sim}{G}}$ denote any tensor that transforms under a rigid superimposed rotation $\underset{\sim}{\underset{\sim}{Q}}$ according to

$$\underset{\sim}{\underset{\sim}{G}}^{*} = \underset{\sim}{\underset{\sim}{Q}} \bullet \underset{\sim}{\underset{\sim}{G}} \tag{15.122}$$

Two particular choices for the $\underset{\sim}{\underset{\sim}{G}}$ tensor are quite common:

- **convected:** $\underset{\sim}{\underset{\sim}{G}} = \underset{\sim}{\underset{\sim}{F}}$, where $\underset{\sim}{\underset{\sim}{F}}$ is the deformation gradient tensor.
- **polar:** $\underset{\sim}{\underset{\sim}{G}} = \underset{\sim}{\underset{\sim}{R}}$, where $\underset{\sim}{\underset{\sim}{R}}$ is the polar rotation tensor.

Note that both of these choices transform according to Eq. (15.122), as required.

Consider any spatial tensor $\underset{\sim}{\underset{\sim}{A}}$. Define an "overbar" operation as

$$\overline{\underset{\sim}{\underset{\sim}{A}}} = \underset{\sim}{\underset{\sim}{G}}^{-1} \bullet \underset{\sim}{\underset{\sim}{A}} \bullet \underset{\sim}{\underset{\sim}{G}}^{-T} \tag{15.123}$$

To understand the reason for the transpose, note that the tensor $\underset{\sim}{\underset{\sim}{A}}$ can be written in basis notation as

$$\underset{\sim}{\underset{\sim}{A}} = A^{ij} \underset{\sim}{g}_i \underset{\sim}{g}_j \tag{15.124}$$

where two vectors written side-by-side are multiplied dyadically, and the base vectors need not be orthonormal. Operating from the left by $\underset{\sim}{\underset{\sim}{G}}^{-1}$ and from the right by $\underset{\sim}{\underset{\sim}{G}}^{-T}$ gives

$$\overline{\underset{\sim}{\underset{\sim}{A}}} = A^{ij} (\underset{\sim}{\underset{\sim}{G}}^{-1} \bullet \underset{\sim}{g}_i)(\underset{\sim}{g}_j \bullet \underset{\sim}{\underset{\sim}{G}}^{-T}) = A^{ij} (\underset{\sim}{\underset{\sim}{G}}^{-1} \bullet \underset{\sim}{g}_i)(\underset{\sim}{\underset{\sim}{G}}^{-1} \bullet \underset{\sim}{g}_j) \tag{15.125}$$

In writing the last form, we have noted that $\underset{\sim}{g}_j \bullet \underset{\sim}{\underset{\sim}{G}}^{-T}$ is the same thing as $\underset{\sim}{\underset{\sim}{G}}^{-1} \bullet \underset{\sim}{g}_j$. We introduce a set of "helper" vectors defined

$$\overline{\underset{\sim}{g}}_i \equiv \underset{\sim}{\underset{\sim}{G}}^{-1} \bullet \underset{\sim}{g}_i, \tag{15.126}$$

then Eq. (15.125) becomes

$$\overline{\underset{\sim}{\underset{\sim}{A}}} = A^{ij} \overline{\underset{\sim}{g}}_i \overline{\underset{\sim}{g}}_j \tag{15.127}$$

This result shows that the components of $\overline{\underset{\sim}{\underset{\sim}{A}}}$ with respect to the $\overline{\underset{\sim}{g}}_k$ basis are identical to the components of $\underset{\sim}{\underset{\sim}{A}}$ with respect to the $\underset{\sim}{g}_k$ basis. In a very loose sense, $\overline{\underset{\sim}{\underset{\sim}{A}}}$ is an "undistorted" version of $\underset{\sim}{\underset{\sim}{A}}$. For example, when $\underset{\sim}{\underset{\sim}{G}} = \underset{\sim}{\underset{\sim}{F}}$, the undistortion operation $\underset{\sim}{\underset{\sim}{G}}^{-1}$ takes away material distortion. When the $\underset{\sim}{\underset{\sim}{G}}$ tensor is a proper rotation, then both the $\overline{\underset{\sim}{g}}_k$ and $\underset{\sim}{g}_k$ bases may be selected to be orthonormal. In this case, $\overline{\underset{\sim}{\underset{\sim}{A}}}$ is an *un-rotated* version of $\underset{\sim}{\underset{\sim}{A}}$. In this case, the operation $\underset{\sim}{\underset{\sim}{G}}^{-1}$ takes away material rotation.

Recall that we assumed that $\underset{\sim}{\underset{\sim}{A}}$ is a spatial tensor. Hence, it transforms according to

$$\underset{\sim}{\underset{\sim}{A}}^{*} = \underset{\sim}{\underset{\sim}{Q}} \bullet \underset{\sim}{\underset{\sim}{A}} \bullet \underset{\sim}{\underset{\sim}{Q}}^{T} \tag{15.128}$$

In light of the assumption of Eq. (15.122), "barred" tensor $\overline{\underset{\approx}{\boldsymbol{A}}}$ is easily verified to be a "reference" tensor, meaning that its value is unaffected by a superimposed rigid rotation:

$$\overline{\underset{\approx}{\boldsymbol{A}}}^{*} = \overline{\underset{\approx}{\boldsymbol{A}}} \tag{15.129}$$

To describe the rate of the distortion tensor $\underset{\approx}{\boldsymbol{G}}$, we will define

$$\underset{\approx}{\Lambda} \equiv \dot{\underset{\approx}{\boldsymbol{G}}} \bullet \underset{\approx}{\boldsymbol{G}}^{-1} \tag{15.130}$$

By using our assumption of Eq. (15.122), it is straightforward to prove that

$$\underset{\approx}{\Lambda}^{*} = \underset{\approx}{\Upsilon} + \underset{\approx}{\boldsymbol{Q}} \bullet \underset{\approx}{\Lambda} \bullet \underset{\approx}{\boldsymbol{Q}}^{T} \tag{15.131}$$

We wish to take rates of $\overline{\underset{\approx}{\boldsymbol{A}}}$. To do this, we need the following helper identity for the rate of an inverse:*

$$\frac{d}{dt}\underset{\approx}{\boldsymbol{G}}^{-1} = -\underset{\approx}{\boldsymbol{G}}^{-1} \bullet \dot{\underset{\approx}{\boldsymbol{G}}} \bullet \underset{\approx}{\boldsymbol{G}}^{-1} \tag{15.132}$$

Using this, you can show that

$$\dot{\overline{\underset{\approx}{\boldsymbol{A}}}} = \underset{\approx}{\boldsymbol{G}}^{-1} \bullet \overset{\circ}{\underset{\approx}{\boldsymbol{A}}} \bullet \underset{\approx}{\boldsymbol{G}}^{-T}, \tag{15.133a}$$

$$\text{where } \overset{\circ}{\underset{\approx}{\boldsymbol{A}}} = \dot{\underset{\approx}{\boldsymbol{A}}} - \underset{\approx}{\Lambda} \bullet \underset{\approx}{\boldsymbol{A}} - \underset{\approx}{\boldsymbol{A}} \bullet \underset{\approx}{\Lambda}^{T} \tag{15.133b}$$

To interpret this equation physically, first lets invert Eq. (15.126) to obtain the spatial $\underset{\sim}{\boldsymbol{g}}_i$ basis in terms of the reference $\overline{\underset{\sim}{\boldsymbol{g}}}_i$ basis:

$$\underset{\sim}{\boldsymbol{g}}_i \equiv \underset{\approx}{\boldsymbol{G}} \bullet \overline{\underset{\sim}{\boldsymbol{g}}}_i, \tag{15.134}$$

If the reference basis $\overline{\underset{\sim}{\boldsymbol{g}}}_i$ is constant, then taking rates gives

$$\dot{\underset{\sim}{\boldsymbol{g}}}_i \equiv \dot{\underset{\approx}{\boldsymbol{G}}} \bullet \overline{\underset{\sim}{\boldsymbol{g}}}_i = \underset{\approx}{\Lambda} \bullet \underset{\approx}{\boldsymbol{G}} \bullet \overline{\underset{\sim}{\boldsymbol{g}}}_i = \underset{\approx}{\Lambda} \bullet \underset{\sim}{\boldsymbol{g}}_i \tag{15.135}$$

Thus, the convected derivative $\overset{\circ}{\underset{\approx}{\boldsymbol{A}}}$ in Eq. (15.133b) is seen to equal the ordinary derivative $\dot{\underset{\approx}{\boldsymbol{A}}}$ minus those parts of the ordinary derivative that arise from rates of the $\underset{\sim}{\boldsymbol{g}}_k$ base vectors. Recalling that we are regarding the $\underset{\approx}{\boldsymbol{G}}$ as a distortion (or rotation if it is orthogonal), we interpret the convected derivative to be the part of the rate that is *not* caused by the distortion (or rotation). From a modelling perspective, this would be the part of the rate caused by material constitutive response, not just by material reorientation.

Importantly, note that Eq. (15.133a) may be written

$$\dot{\overline{\underset{\approx}{\boldsymbol{A}}}} = \overline{\overset{\circ}{\underset{\approx}{\boldsymbol{A}}}} \tag{15.136}$$

* Note the similarity of this identity with the scalar equation, $\frac{d}{dt}\left(\frac{1}{x}\right) = -\dot{x}/x^2 = -(x^{-1})\dot{x}(x^{-1})$.

This equation says that the *ordinary* rate of the bared tensor is the same as the bar operation acting on the co-rotational rate. Whenever there exists a tensor $\underset{\approx}{\boldsymbol{G}}$ such that the co-rotation tensor $\underset{\approx}{\Lambda}$ is expressible in the form of Eq. (15.130), then the co-rotational rate is called a "**Lie derivative**" and you will usually see it written in books in the form

$$\overset{\circ}{\underset{\approx}{\boldsymbol{A}}} = \underset{\approx}{\boldsymbol{G}} \bullet \frac{d}{dt}(\underset{\approx}{\boldsymbol{G}}^{-1} \bullet \underset{\approx}{\boldsymbol{A}} \bullet \underset{\approx}{\boldsymbol{G}}^{-T}) \bullet \underset{\approx}{\boldsymbol{G}}^{T} \tag{15.137}$$

Alternative to Lie Derivatives in Constitutive models. For a mathematically limber reader, the preceding discussion of Lie derivatives probably did shed some light on the physical meaning of a convected derivative. However, the use of such strange time derivatives is foreign to many readers. Furthermore, using convected derivatives in numerical computations can be quite cumbersome and sometimes inefficient. Consider, for example, the linear version of Eq. (15.115):

$$\overset{\circ}{\underset{\approx}{\sigma}} = \underset{\approx\approx}{\boldsymbol{E}} : \underset{\approx}{\boldsymbol{D}}, \qquad (\text{in indicial form, } \overset{\circ}{\sigma}_{ij} = E_{ijkl} D_{kl}) \tag{15.138}$$

where $\underset{\approx\approx}{\boldsymbol{E}}$ is the fourth-order tangent stiffness tensor. In order for this equation to be generalized to anisotropic materials, the dependence on $\underset{\approx}{\boldsymbol{D}}$ in Eq. (15.115) must be appended to also include dependence on material orientation. In other words, upon a superimposed rigid rotation, not must $\underset{\approx}{\boldsymbol{D}}$ change, the material stiffness tensor must also change. Specifically, to satisfy PMFI, we must have

$$\overset{\circ}{\underset{\approx}{\sigma}}{}^{*} = \underset{\approx\approx}{\boldsymbol{E}}^{*} : \underset{\approx}{\boldsymbol{D}}^{*} \tag{15.139}$$

Recalling Eq. (15.121) and (15.109), both $\overset{\circ}{\underset{\approx}{\sigma}}$ and $\underset{\approx}{\boldsymbol{D}}$ are spatial tensors. Thus, to satisfy PMFI, the stiffness tensor must transform under superimposed rotation such that

$$E^{*}_{ijkl} = Q_{ip} Q_{jq} Q_{kr} Q_{ls} E_{pqrs} \tag{15.140}$$

In other words, the stiffness components must be referenced to a basis that rotates with the material. Computing the rotated stiffnesses can be quite expensive. An alternative and equivalent technique is to leave the stiffnesses unchanged and instead *unrotate* the stress and strain rate. Then, once the constitutive model is applied in this unrotated frame, the final result may be rotated back to the spatial frame. This qualitative proposal is derived in mathematical detail below in terms of our generalized distortion/rotation tensor $\underset{\approx}{\boldsymbol{G}}$ and its associated "bar" operation defined in Eq. (15.123).

Barring both sides of Eq. (15.115) and applying Eq. (15.136) gives

$$\dot{\underset{\approx}{\bar{\sigma}}} = \overline{g(\underset{\approx}{\boldsymbol{D}})}, \text{ or in the linear case, } \dot{\underset{\approx}{\bar{\sigma}}} = \overline{\underset{\approx\approx}{\boldsymbol{E}} : \underset{\approx}{\boldsymbol{D}}} \tag{15.141}$$

We can always define a new constitutive function $\bar{g}$ that depends implicitly on $\underset{\approx}{\boldsymbol{G}}$ such that

$$\overline{g(\underset{\approx}{\boldsymbol{D}})} = \bar{g}(\overline{\underset{\approx}{\boldsymbol{D}}}) \tag{15.142}$$

Hence, we may write the constitutive equation as

$$\dot{\overline{\underset{\approx}{\sigma}}} = \bar{g}(\overline{\underset{\approx}{\boldsymbol{D}}}) \tag{15.143}$$

In the linear case, applying the definition of the "bar" operation to Eq. (15.141) shows that

$$\dot{\overline{\underset{\approx}{\sigma}}} = \overline{\underset{\approx\approx}{\boldsymbol{E}}} : \overline{\underset{\approx}{\boldsymbol{D}}}, \text{ where } \overline{E}_{ijkl} = G^{-1}_{ip} G^{-1}_{iq} G^{-1}_{ir} G^{-1}_{is} E_{pqrs} \tag{15.144}$$

Here we have the *ordinary* rate of an "undistorted/unrotated" stress being a function of the "undistorted/unrotated" stretching. This formulation has several advantages:

- The stress rate is an ordinary rate, not a convected rate, and it is therefore easier to integrate over time.
- After a numerical constitutive model has been developed to give good answers for zero-rotation problems, it can be immediately generalized to permit rotations by simply having the host code perform the "bar" operation on all of the arguments to the constitutive model before calling the model. Upon return, the host code then "unbars" the results.
- When expressed in terms of the barred reference configuration, the constitutive Eq. (15.143) is not restricted to isotropy. For the linear case, the "bar" stiffness is often regarded as an unchanging fourth-order tensor, and the fourth-order transformation operation therefore becomes unnecessary when working exclusively in the "barred" space.

The two most common choices for the "bar" transformation tensor $\underset{\approx}{\boldsymbol{G}}$ are

- $\underset{\approx}{\boldsymbol{G}} = \underset{\approx}{\boldsymbol{R}}$ from the polar decomposition, in which case, using the Lie derivative in the "unrotated" configuration is roughly equivalent to using polar rates in a spatial formulation.
- $\underset{\approx}{\boldsymbol{G}} = \underset{\approx}{\boldsymbol{F}}$ from the deformation gradient, in which case, using the Lie derivative in the "un-convected" configuration is equivalent to using convected coordinates. A key disadvantage is that this formulation may require special attention to material nonlinearity to avoid instabilities in compression.

When you consider thermodynamics, the Cauchy stress measure is typically abandoned in favor of the Kirchhoff stress $\underset{\approx}{\tau}$, defined to equal the Cauchy stress times the Jacobian of the deformation. So the thermodynamically preferable form for the constitutive law is

$$\dot{\overline{\underset{\approx}{\tau}}} = \bar{h}(\overline{\underset{\approx}{\boldsymbol{D}}}) \tag{15.145}$$

If we choose $\underset{\approx}{\boldsymbol{G}} = \underset{\approx}{\boldsymbol{F}}$, then $\bar{\underset{\approx}{\tau}}$ equals the second Piola Kirchhoff (PK2) stress and $\underset{\approx}{\overline{\boldsymbol{D}}}$ equals the ordinary rate of Lagrange strain. If we choose $\underset{\approx}{\boldsymbol{G}} = \underset{\approx}{\boldsymbol{R}}$, then $\bar{\underset{\approx}{\tau}}$ is the *unrotated* Kirchhoff stress and $\underset{\approx}{\overline{\boldsymbol{D}}}$ is the unrotated symmetric part of the velocity gradient.*

Frame indifference is only an essential (not final) step

It's important to keep in mind that frame indifference is only a *necessary* property that must hold for any general constitutive model. Satisfying PMFI is not *sufficient* to ensure that the constitutive model will perform well under large material distortions. Different constitutive laws can predict widely varying — and sometimes absurd — results even if they satisfy PMFI.

Recall from Eq. (15.116) that a co-rotational stress rate is generally expressible in the following form:

$$\overset{\circ}{\underset{\approx}{\sigma}} = \dot{\underset{\approx}{\sigma}} - \underset{\approx}{\Lambda} \bullet \underset{\approx}{\sigma} - \underset{\approx}{\sigma} \bullet \underset{\approx}{\Lambda}^T \tag{15.146}$$

For the polar rate, the tensor $\underset{\approx}{\Lambda}$ is equal to the skew-symmetric polar spin $\underset{\approx}{\Omega}$. For the Jaumann rate, $\underset{\approx}{\Lambda}$ is the vorticity $\underset{\approx}{\boldsymbol{W}}$. It is straightforward (but not trivial) to demonstrate that both rates are equivalent whenever material distortion is small. By this, we mean that the two rates are nearly equivalent whenever the material has not significantly changed *shape* even if large *size* and/or *orientation* changes have occurred. Mathematically, a motion involves small distortions if the principal stretches are all nearly equal to each other. For problems with small material distortion, all co-rotational rates predict approximately the same answer, so it makes sense to use the one that is computationally least expensive. For problems with large material distortion, however, special care must be taken to handle the material nonlinearity.

To demonstrate that satisfaction of PMFI is not enough to obtain sensible constitutive model predictions, Dienes [8] considered the prediction under simple shear of simple isotropic linear elastic constitutive law

$$\overset{\circ}{\underset{\approx}{\sigma}} = \underset{\approx\approx}{\boldsymbol{E}} : \underset{\approx}{\boldsymbol{D}}, \text{ where } \underset{\approx\approx}{\boldsymbol{E}} \text{ is isotropic and constant.} \tag{15.147}$$

Regardless of which objective rate is used, the above equation satisfies PMFI. However, Dienes demonstrated that the *Jaumann* rate predicts anomalous oscillatory shear stresses whereas the Polar rate predicts intuitively more appealing monotonically increasing shear stress. The Jaumann rate performs so poorly because, for simple shear, the vorticity $\underset{\approx}{\boldsymbol{W}}$ is constant throughout time, so the Jaumann rate basically assumes that a material element tumbles and tumble throughout time. Simply drawing a single material element as it deforms throughout time demonstrates that this thinking is flawed. Fibers

* For general motions, there does *not* exist any strain tensor $\underset{\approx}{\xi}$ such that $\underset{\approx}{\boldsymbol{D}} = \dot{\underset{\approx}{\xi}}$. I don't know if a similar statement holds in the unrotated configuration.

that were originally perpendicular to the shear plane eventually rotate approximately 90° into the shear plane. Fibers in the shear plane never rotate at all. Unlike the vorticity, the polar spin $\underset{\approx}{\Omega}$ captures this effect — it equals the vorticity at first, but it approaches zero as time increases. As a rule, all of the objective rates are equivalent whenever material distortion is small. For small distortions, the material does not significantly change shape, but may permissibly change size and orientation.

Dienes definitely demonstrated that the Jaumann rate was "bad" in the this context of simple shear with isotropic linear elasticity. However, the fact that the polar rate was "less bad" does not mean that the polar rate was *correct*. Brannon [11] demonstrated that even the polar rate gives incorrect predictions when applied to distortion of a fiber-dominant composite material. The heart of the issue is not which rate you use. The key to good constitutive modelling is to handle material nonlinearity properly. The natural rate is the one that makes the nonlinear constitutive model as simple in structure as possible.

When implementing large distortion capability into a code, it is essential to permit the constitutive equations to retain control of the nonlinear material response. Minimally, all material models must satisfy PMFI. One way to satisfy this requirement is for the host code to always work in the unrotated reference configuration. Beyond that, the treatment of large material distortion should be retained in the realm of the constitutive model. We do not believe that there is one magical strain measure or fabulous co-rotational rate that will serve all materials well — that's why the treatment of large material distortion must remain the responsibility of the material model. The host code should be responsible for PMFI by always working in an unrotated configuration.

16. Rigid Body Mechanics

The mechanics of rigid bodies can be found in any good dynamics textbook [e.g., References 4, 12]. However, very few textbooks present the theory in a form that is optimized for readers who are already familiar with advanced tensor analysis. Consequently, such books often provide derivations in very awkward component forms. Here we present the theory for readers who already know tensor analysis as it applies to continuum mechanics.

A body is considered "rigid" if the distance between any two points in the body remains unchanged. This, of course, implies that the angles between any two lines in the body also remains unchanged.

Let O_o denote a fixed origin. Let O denote a translating origin that differs from the fixed origin by a vector $\underset{\sim}{\boldsymbol{k}}$. As sketched below, let $\underset{\sim}{\boldsymbol{X}}$ be the *initial* position of a particle with respect to O_o. Let $\underset{\sim}{\boldsymbol{x}}$ be the *current* position of the same particle with respect to the fixed origin O_o. Finally, let $\underset{\sim}{\boldsymbol{r}}$ denote the current position of the particle with respect to the moving origin O.

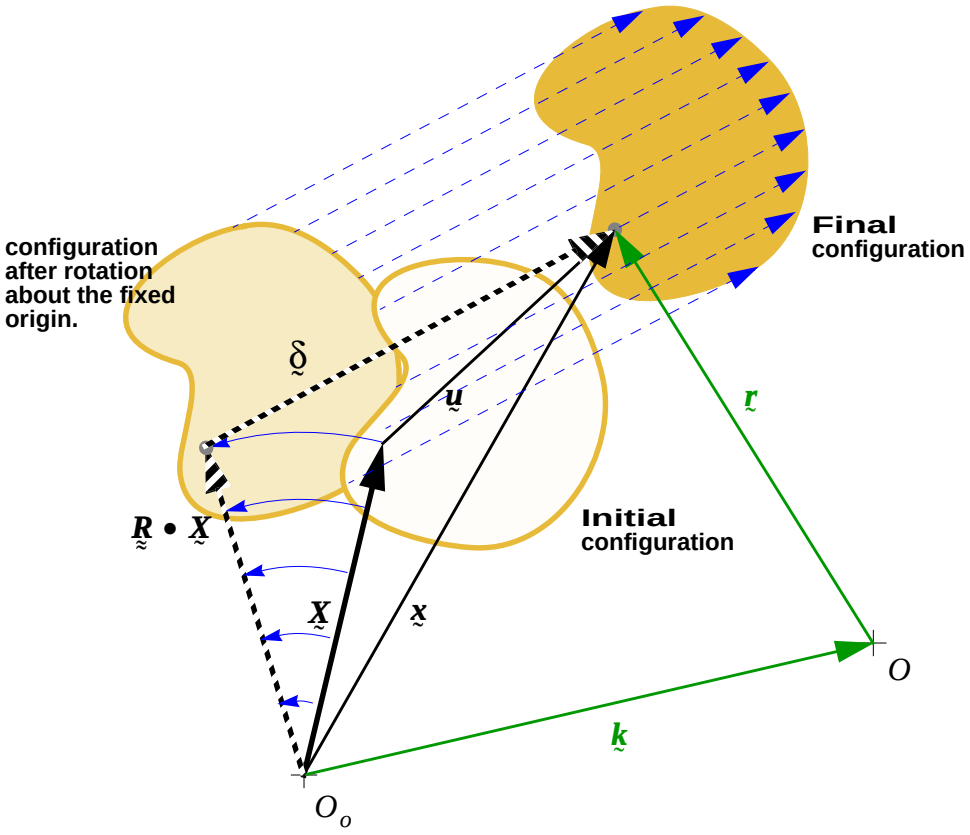


Figure 16.1. Identifying points within a rigid body. The position vector $\underset{\sim}{\boldsymbol{X}}$ identifies the *initial* location of the grey particle. The position vectors $\underset{\sim}{\boldsymbol{x}}$ and $\underset{\sim}{\boldsymbol{r}}$ both point the *current* location of the same particle.

The position vectors $\underset{\sim}{\boldsymbol{X}}$ and $\underset{\sim}{\boldsymbol{x}}$ are defined to point from the *fixed* origin O_o to the particle in question. Consequently, $\underset{\sim}{\boldsymbol{X}}$ and $\underset{\sim}{\boldsymbol{x}}$ are both origin-dependent vectors — they will change if a different fixed origin is used. By contrast, the displacement vector $\underset{\sim}{\boldsymbol{u}}$ is defined to point from the initial location of a particle to the current location of the same particle. Consequently, the displacement vector is unaffected by the choice of origin. Any vector with this property is called a *free vector*.

Fig. 16.1 shows how a rigid motion may be regarded as a rotation $\underset{\approx}{\boldsymbol{R}}$ about the fixed origin followed by a rigid translation $\underset{\sim}{\delta}$. This conclusion follows directly from the requirement that the distances between any two points must remain fixed. Fig. 16.2 demonstrates that the translation vector $\underset{\sim}{\delta}$ depends on the choice of fixed origin. Therefore, much like position vectors, the translation vector is an origin-dependent vector, not a free vector. By contrast, the rotation tensor $\underset{\approx}{\boldsymbol{R}}$ is the same regardless of the choice of origin. Thus, $\underset{\approx}{\boldsymbol{R}}$ is a free tensor.

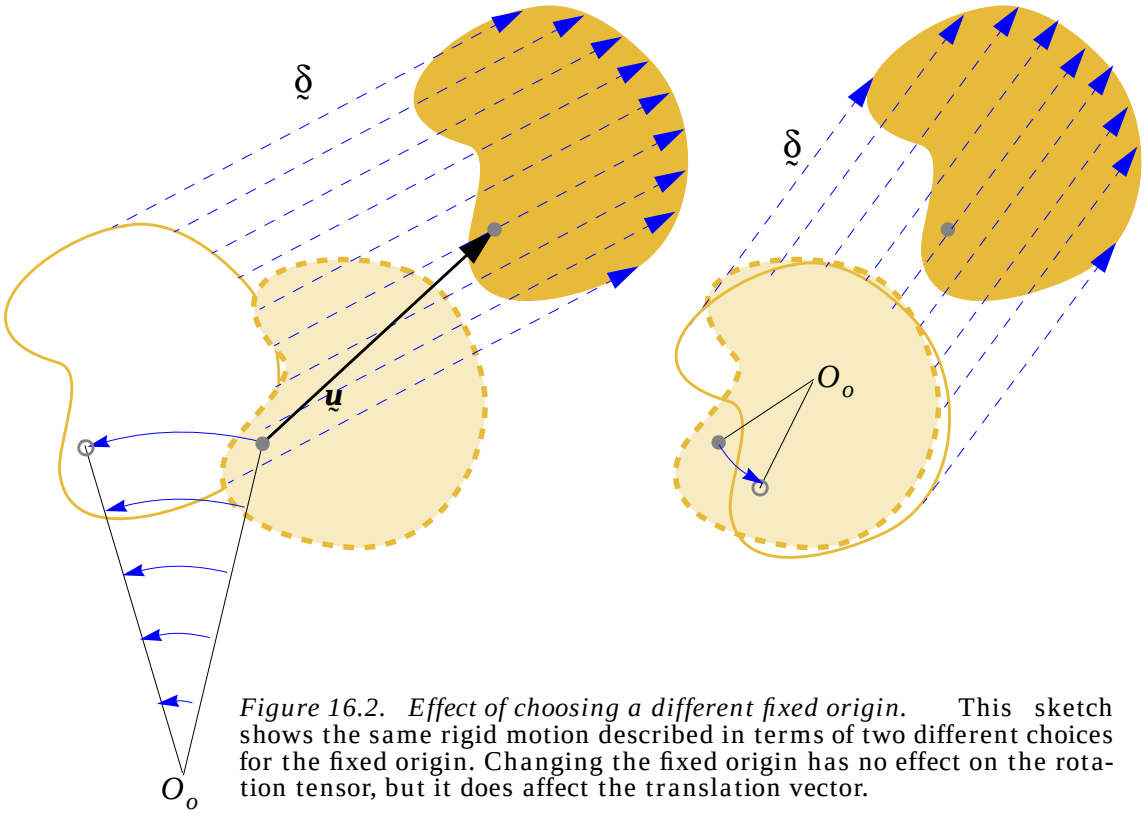


Figure 16.2. Effect of choosing a different fixed origin. This sketch shows the same rigid motion described in terms of two different choices for the fixed origin. Changing the fixed origin has no effect on the rotation tensor, but it does affect the translation vector.

A simple description of rigid motion

Under the assumption of rigid motion, the particle's current position $\underset{\sim}{x}$ must be expressible in the form of a rotation plus a translation

$$\underset{\sim}{x} = \underset{\approx}{R} \bullet \underset{\sim}{X} + \underset{\sim}{\delta} \tag{16.1}$$

where $\underset{\approx}{R}$ is a rotation tensor and $\underset{\sim}{\delta}$ is a translation vector, both of which depend on time but not position. In short hand, we indicate this dependence as

$$\underset{\approx}{R} = \underset{\approx}{R}(t) \tag{16.2}$$

$$\underset{\sim}{\delta} = \underset{\sim}{\delta}(t) \tag{16.3}$$

As mentioned earlier, one disadvantage of this description of rotation is that it contains the origin-dependent (and therefore physically non-intuitive) displacement vector $\underset{\sim}{\delta}$.

A relative description of rigid motion

For rigid motion problems, there is often a special point in the body for which the complete motion is known (or there is a point for which the complete motion can be relatively easily computed). This special point might be the location of a hinge on a rigid robotic arm. Alternatively, the special point might be the center of mass of the body. We will later show that the position of the center of mass is governed by the linear momentum equation, which depends only on the net force on the rigid body. Consequently, *the position of the body's center of mass is relatively easy to track over time*, making it a good choice for use as a reference point.

A real physical body occupies only a finite amount of space. We will speak of the **virtual body** as the extension of the rigid body to include all points in space. Suppose, for example, that the real rigid body is a spherical shell. The centroid of this body lies at the sphere center, which is not part of the physical body, but it is part of the virtual body. Points in the infinite virtual body move according to Eq. (16.1). Namely

$$\underset{\sim}{x} = \underset{\approx}{R} \bullet \underset{\sim}{X} + \underset{\sim}{\delta} \tag{16.4}$$

Let $\underset{\sim}{X}^*$ denote some *particular* convenient point in the virtual body. Let $\underset{\sim}{x}^*$ denote the current position of the reference point. Later on, it will become evident that a very good choice for the reference particle is the body's center of mass, $\underset{\sim}{x}^c$. Another common choice for the reference particle is any particle (such as the hinge point on a rigid robotic arm) for which the position vector of the reference point is known for all time.

Equation (16.4) holds for *all* points on the rigid body, so it must also hold for the reference point:

$$\underset{\sim}{x}^* = \underset{\approx}{R} \bullet \underset{\sim}{X}^* + \underset{\sim}{\delta} \tag{16.5}$$

Subtracting Eq. (16.5) from (16.1) gives

$$\underset{\sim}{x} - \underset{\sim}{x}^* = \underset{\approx}{R} \bullet (\underset{\sim}{X} - \underset{\sim}{X}^*) \tag{16.6}$$

In this manner, we have separated a rigid motion into two distinct components: The motion of a single reference particle $\underset{\sim}{X}^*$ and motion of all other particles *relative to the reference particle*. In all of the following discussions, we will presume that the reference particle is *Lagrangian* — in other words, it always corresponds to the same physical point on the body so that the time rate of $\underset{\sim}{X}^*$ will be zero.

Velocity and angular velocity for rigid motion

Taking the reference point to be the center of mass, recall that rigid motion may be described by

$$\underset{\sim}{x} = \underset{\sim}{x}_c + \underset{\approx}{R} \bullet (\underset{\sim}{X} - \underset{\sim}{X}_c) \tag{16.7}$$

in which

$$\underset{\approx}{R} = \underset{\approx}{R}(t) \text{ and } \underset{\sim}{x}_c = \underset{\sim}{x}_c(t) \tag{16.8}$$

are two known functions of time. In other words, the motion of a rigid body is completely described by specifying the position of the center of mass $\underset{\sim}{x}_c$ as a function of time and the rotation tensor $\underset{\approx}{R}$ as a function of time. For problems in mechanics, the set of forces acting on the body are known, and the resulting motion $\{\underset{\sim}{x}_c(t), \underset{\approx}{R}(t)\}$ is desired.

Importantly, the origin from which $\underset{\sim}{x}$ is measured is fixed, so the material velocity is defined by

$$\dot{\underset{\sim}{x}} \equiv \left(\frac{\partial \underset{\sim}{x}}{\partial t}\right)_{\underset{\sim}{X}} \tag{16.9}$$

or

$$\dot{\underset{\sim}{x}} = \dot{\underset{\sim}{x}}_c + \dot{\underset{\approx}{R}} \bullet (\underset{\sim}{X} - \underset{\sim}{X}_c) \tag{16.10}$$

Noting that $\underset{\sim}{X} - \underset{\sim}{X}_c = \underset{\approx}{R}^T \bullet (\underset{\sim}{x} - \underset{\sim}{x}_c)$, this may be written as

$$\boxed{\dot{\underset{\sim}{x}} = \dot{\underset{\sim}{x}}_c + \underset{\approx}{\Omega} \bullet (\underset{\sim}{x} - \underset{\sim}{x}_c)}, \tag{16.11}$$

where

$$\boxed{\underset{\approx}{\Omega} \equiv \dot{\underset{\approx}{R}} \bullet \underset{\approx}{R}^T} \tag{16.12}$$

The angular velocity vector $\underset{\sim}{\omega}$ is defined to equal the axial vector associated with the spin tensor $\underset{\approx}{\Omega}$. In other words, it is defined such that

$$\boxed{\underset{\approx}{\Omega} \bullet \underset{\sim}{u} = \underset{\sim}{\omega} \times \underset{\sim}{u}} \text{ for any vector } \underset{\sim}{u} \tag{16.13}$$

In direct notation, this implies that the angular rotation *tensor* $\underset{\approx}{\Omega}$ is related to the angular rotation *vector* $\underset{\sim}{\omega}$ by

$$\underset{\approx}{\Omega} = -\underset{\approx\approx}{\varepsilon} \bullet \underset{\sim}{\omega} \tag{16.14}$$

In indicial notation, this is written

$$\Omega_{ij} = -\sum_{k=1}^{3} \varepsilon_{ijk}\omega_k \tag{16.15}$$

Thus, the component matrix of $\underset{\approx}{\Omega}$ can be constructed from the angular velocity vector

$$[\underset{\approx}{\Omega}] = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \tag{16.16}$$

Conversely, if the angular rotation tensor $\underset{\approx}{\Omega}$ is known, then the angular velocity vector $\underset{\sim}{\omega}$ can be computed by

$$\underset{\sim}{\omega} = -\frac{1}{2}\underset{\approx\approx}{\varepsilon} : \underset{\approx}{\Omega} \tag{16.17}$$

or, written out explicitly (recalling that $\underset{\approx}{\Omega}$ is skew symmetric),

$$\omega_1 = \Omega_{32} \tag{16.18}$$

$$\omega_2 = \Omega_{13} \tag{16.19}$$

$$\omega_3 = \Omega_{21} \tag{16.20}$$

As mentioned earlier, axial tensors are defined so that they are related to their axial vectors by the important identity

$$\boxed{\underset{\approx}{\Omega} \bullet \underset{\sim}{u} = \underset{\sim}{\omega} \times \underset{\sim}{u}} \text{ for any vector } \underset{\sim}{u} \tag{16.21}$$

Thus, applying this identity to Eq. (16.11) gives

$$\dot{\underset{\sim}{x}} = \dot{\underset{\sim}{x}}_c + \underset{\sim}{\omega} \times (\underset{\sim}{x} - \underset{\sim}{x}_c), \tag{16.22}$$

which is the form of the velocity cited in most elementary dynamics textbooks.

Time rate of a vector embedded in a rigid body

Define an embedded vector $\underset{\sim}{z}$ to be one that points from one distinct point $\underset{\sim}{x}^{\text{tail}}$ to a second point $\underset{\sim}{x}^{\text{tip}}$ embedded in a rigid body:

$$\underset{\sim}{z} \equiv \underset{\sim}{x}^{\text{tip}} - \underset{\sim}{x}^{\text{tail}} \tag{16.23}$$

The rate of this embedded vector is

$$\dot{\underset{\sim}{z}} \equiv \dot{\underset{\sim}{x}}^{\text{tip}} - \dot{\underset{\sim}{x}}^{\text{tail}} \tag{16.24}$$

Applying Eq. (16.11) shows that

$$\dot{\underset{\sim}{x}}^{\text{tip}} = \dot{\underset{\sim}{x}}_c + \underset{\approx}{\Omega} \bullet (\underset{\sim}{x}^{\text{tip}} - \underset{\sim}{x}_c), \tag{16.25}$$

$$\dot{\underset{\sim}{x}}^{\text{tail}} = \dot{\underset{\sim}{x}}_c + \underset{\approx}{\Omega} \bullet (\underset{\sim}{x}^{\text{tail}} - \underset{\sim}{x}_c) \tag{16.26}$$

Hence, Eq. (16.24) becomes

$$\dot{\underset{\sim}{z}} \equiv \underset{\approx}{\Omega} \bullet (\underset{\sim}{x}^{\text{tip}} - \underset{\sim}{x}^{\text{tail}}) \tag{16.27}$$

or

$$\boxed{\dot{\underset{\sim}{z}} \equiv \underset{\approx}{\Omega} \bullet \underset{\sim}{z}} \tag{16.28}$$

In terms of the angular velocity vector, this result is written

$$\boxed{\dot{\underset{\sim}{z}} = \underset{\sim}{\omega} \times \underset{\sim}{z}} \tag{16.29}$$

Acceleration for rigid motion

The second rate of Eq. (16.10) gives a very simple expression for the acceleration:

$$\boxed{\ddot{\underset{\sim}{x}} = \ddot{\underset{\sim}{x}}_c + \ddot{\underset{\approx}{R}} \bullet (\underset{\sim}{X} - \underset{\sim}{X}_c)}, \tag{16.30}$$

or, again noting that $\underset{\sim}{X} - \underset{\sim}{X}_c = \underset{\approx}{R}^T \bullet (\underset{\sim}{x} - \underset{\sim}{x}_c)$, the fully spatial expression for the acceleration is

$$\ddot{\underset{\sim}{x}} = \ddot{\underset{\sim}{x}}_c + \ddot{\underset{\approx}{R}} \bullet \underset{\approx}{R}^T \bullet (\underset{\sim}{x} - \underset{\sim}{x}_c) \tag{16.31}$$

Recall that

$$\dot{\underset{\approx}{R}} = \underset{\approx}{\Omega} \bullet \underset{\approx}{R}. \tag{16.32}$$

Therefore

$$\ddot{\underset{\approx}{R}} = \dot{\underset{\approx}{\Omega}} \bullet \underset{\approx}{R} + \underset{\approx}{\Omega} \bullet \dot{\underset{\approx}{R}} = (\dot{\underset{\approx}{\Omega}} + \underset{\approx}{\Omega} \bullet \underset{\approx}{\Omega}) \bullet \underset{\approx}{R} \tag{16.33}$$

Thus

$$\boxed{\ddot{\underset{\approx}{R}} \bullet \underset{\approx}{R}^T = \dot{\underset{\approx}{\Omega}} + \underset{\approx}{\Omega}^2} \tag{16.34}$$

and

$$\boxed{\ddot{\underset{\sim}{x}} = \ddot{\underset{\sim}{x}}_c + [\dot{\underset{\approx}{\Omega}} + \underset{\approx}{\Omega}^2] \bullet (\underset{\sim}{x} - \underset{\sim}{x}_c)}. \tag{16.35}$$

Applying Eq. (16.21) twice shows that

$$\boxed{\underset{\sim}{\Omega}^2 \bullet \underset{\sim}{u} = \underset{\sim}{\omega} \times (\underset{\sim}{\omega} \times \underset{\sim}{u})} \text{ for any vector } \underset{\sim}{u} \tag{16.36}$$

Note from Eq. (16.16) that

$$[\underset{\sim}{\Omega}^2] = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} = \begin{bmatrix} -(\omega_2^2 + \omega_3^2) & \omega_1\omega_2 & \omega_1\omega_3 \\ \omega_2\omega_1 & -(\omega_3^2 + \omega_1^2) & \omega_2\omega_3 \\ \omega_3\omega_1 & \omega_3\omega_2 & -(\omega_1^2 + \omega_2^2) \end{bmatrix} \tag{16.37}$$

or, rearranging this result,

$$[\underset{\sim}{\Omega}^2] = \begin{bmatrix} \omega_1^2 & \omega_1\omega_2 & \omega_1\omega_3 \\ \omega_2\omega_1 & \omega_2^2 & \omega_2\omega_3 \\ \omega_3\omega_1 & \omega_3\omega_2 & \omega_3^2 \end{bmatrix} - (\omega_1^2 + \omega_2^2 + \omega_3^2) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{16.38}$$

In direct notation,

$$\boxed{\underset{\sim}{\Omega}^2 = \underset{\sim}{\omega}\underset{\sim}{\omega} - (\underset{\sim}{\omega} \bullet \underset{\sim}{\omega})\underline{\underline{I}}} \tag{16.39}$$

Thus, a way of computing $\underset{\sim}{\Omega}^2 \bullet \underset{\sim}{u}$ that is an alternative to Eq. (16.36) is

$$\boxed{\underset{\sim}{\Omega}^2 \bullet \underset{\sim}{u} = \underset{\sim}{\omega}(\underset{\sim}{\omega} \bullet \underset{\sim}{u}) - (\underset{\sim}{\omega} \bullet \underset{\sim}{\omega})\underset{\sim}{u}} \text{ for any vector } \underset{\sim}{u} \tag{16.40}$$

Recall from Eq. (16.14) that $\underset{\sim}{\omega}$ is the axial vector associated with the spin tensor $\underset{\sim}{\Omega}$. The alternating tensor $\underline{\underline{\varepsilon}}$ is independent of time. Therefore the time rate of Eq. (16.14) gives

$$\dot{\underset{\sim}{\Omega}} = -\underline{\underline{\varepsilon}} \bullet \dot{\underset{\sim}{\omega}}, \tag{16.41}$$

which shows that the *angular acceleration vector* $\dot{\underset{\sim}{\omega}}$ is the axial vector associated with the rate of the spin tensor $\dot{\underset{\sim}{\Omega}}$. Therefore the angular acceleration vector and tensor satisfies the basic identity:

$$\dot{\underset{\sim}{\Omega}} \bullet \underset{\sim}{u} = \dot{\underset{\sim}{\omega}} \times \underset{\sim}{u} \text{ for any vector } \underset{\sim}{u} \tag{16.42}$$

We may apply Eqs. (16.42) and (16.40) to Eq. (16.35) to write the acceleration strictly in terms of the angular velocity vector:

$$\boxed{\ddot{\underset{\sim}{x}} = \ddot{\underset{\sim}{x}}_c + \dot{\underset{\sim}{\omega}} \times (\underset{\sim}{x} - \underset{\sim}{x}_c) + \underset{\sim}{\omega}[\underset{\sim}{\omega} \bullet (\underset{\sim}{x} - \underset{\sim}{x}_c)] - (\underset{\sim}{\omega} \bullet \underset{\sim}{\omega})(\underset{\sim}{x} - \underset{\sim}{x}_c)} \tag{16.43}$$

Alternatively, using Eq. (16.36) instead of (16.40),

$$\boxed{\ddot{\underset{\sim}{x}} = \ddot{\underset{\sim}{x}}_c + \dot{\underset{\sim}{\omega}} \times (\underset{\sim}{x} - \underset{\sim}{x}_c) + \underset{\sim}{\omega} \times [\underset{\sim}{\omega} \times (\underset{\sim}{x} - \underset{\sim}{x}_c)]} \tag{16.44}$$

This is the form of the acceleration typically cited in dynamics textbooks. Clearly, however, anyone with even a modicum of skill with tensor analysis would prefer the elegant simplicity of Eq. (16.30) or (16.35).

Important properties of a rigid body

One of the most profound discoveries in the study of rigid body dynamics is that we do not need to know the exact mass distribution of the body in order to completely determine its motion. Rather, it is sufficient to know the total mass M, the location of the center of mass $\underset{\sim}{x}_c$ and the value of a special second-order "inertia tensor" $\underset{\approx}{\mu}_c$ defined relative to the center of mass. These three quantities depend only on the body's geometrical shape and its spatial density distribution. They may be computed *a priori* at time zero and then easily updated through time if the rotation tensor $\underset{\approx}{\boldsymbol{R}}$ and the position of one point $\underset{\sim}{x}_p$ are known as functions of time.

Suppose that the size and shape of the rigid body B are known. Let $\rho = \rho(\underset{\sim}{x})$ be the mass density* of the rigid body at position $\underset{\sim}{x}$. We can define moments of the rigid body as follows:

Zeroth moment (the mass): $$M \equiv \int_B \rho\, dV = \int_{B_o} \rho_o\, dV_o \tag{16.45}$$

First moment (the CM): $$\underset{\sim}{x}_c \equiv \frac{1}{M}\int_B \underset{\sim}{x}\rho\, dV \tag{16.46}$$

Second moment about the CM: $$\underset{\approx}{y}_c \equiv \frac{1}{M}\int_B (\underset{\sim}{x}-\underset{\sim}{x}_c)(\underset{\sim}{x}-\underset{\sim}{x}_c)\rho\, dV \tag{16.47}$$

Note that the zeroth moment is just the mass of the body. The first moment is the *center-of-mass* (henceforth abbreviated CM). For now, the second moment may be regarded as an abstract mathematical entity having no intuitive meaning. Note that the second moment is defined using the position vector relative to the center of mass. The second moment about an arbitrary point $\underset{\sim}{x}_p$ is defined

Second moment about $\underset{\sim}{x}_p$: $$\underset{\approx}{y}_p \equiv \frac{1}{M}\int_B (\underset{\sim}{x}-\underset{\sim}{x}_p)(\underset{\sim}{x}-\underset{\sim}{x}_p)\rho\, dV \tag{16.48}$$

We will later show that the second moment about an arbitrary point $\underset{\sim}{x}_p$ can be computed from the second moment about the CM by the very simple formula:

$$\boxed{\underset{\approx}{y}_p \equiv \underset{\approx}{y}_c + \underset{\sim}{d}_{pc}\underset{\sim}{d}_{pc}}, \text{ where } \underset{\sim}{d}_{pc} \equiv \underset{\sim}{x}_p - \underset{\sim}{x}_c \tag{16.49}$$

Thus, if $\underset{\approx}{y}_c$ and $\underset{\sim}{x}_c$ are known, then the second moment about $\underset{\sim}{x}_p$ can be immediately computed without needing to perform any integrations. In other words, knowledge of $\underset{\approx}{y}_p$ does not provide any additional information, but its use might simplify certain equations.

* spatially varying, but constant (in a Lagrangian sense) with respect to time.

We will use a superscript "0" as an alternative notation for the value of a quantity at time zero. Thus, for example,

$$\underset{\sim}{x}^0 \equiv \underset{\sim}{x}|_{t=0} = \underset{\sim}{X} \tag{16.50}$$

$$\underset{\sim}{x}_c^0 \equiv \underset{\sim}{x}_c|_{t=0} = \frac{1}{M}\int_B \underset{\sim}{x}^0 \rho\, dV = \frac{1}{M}\int_B \underset{\sim}{X}\rho_o\, dV_o = \underset{\sim}{X}_c \tag{16.51}$$

$$\underset{\approx}{y}_c^0 \equiv \underset{\approx}{y}_c\Big|_{t=0} = \frac{1}{M}\int_B (\underset{\sim}{X} - \underset{\sim}{X}_c)(\underset{\sim}{X} - \underset{\sim}{X}_c)\rho_o\, dV_o \tag{16.52}$$

When analyzing the angular momentum equation, we will encounter the expression

$$M\underset{\approx}{\varepsilon}\!:\!(\underset{\approx}{y}_p \bullet \underset{\approx}{\Omega}^T) \tag{16.53}$$

where (recall) M is the total mass of the body, $\underset{\approx}{\varepsilon}$ is the alternating (permutation) tensor, $\underset{\approx}{y}_p$ is the second moment of the body about a point $\underset{\sim}{x}_p$, and $\underset{\approx}{\Omega}$ is the angular velocity tensor. The expression in Eq. (16.53) is linear with respect to $\underset{\approx}{\Omega}$ and it must therefore be linear with respect to the axial vector $\underset{\sim}{\omega}$. In other words, there must exist a tensor $\underset{\approx}{\mu}_p$ that is independent of $\underset{\sim}{\omega}$ such that

$$M\underset{\approx}{\varepsilon}\!:\!(\underset{\approx}{y}_p \bullet \underset{\approx}{\Omega}^T) = \underset{\approx}{\mu}_p \bullet \underset{\sim}{\omega} \tag{16.54}$$

Writing this expression in indicial form reveals that

$$\underset{\approx}{\mu}_p \equiv M[(\mathrm{tr}\,\underset{\approx}{y}_p)\underset{\approx}{I} - \underset{\approx}{y}_p] \tag{16.55}$$

The tensor $\underset{\approx}{\mu}_p$ is called the rotational moment of inertia for the body, and we will see that it plays a role in resisting rotational motion that is similar to the inertial role of mass in resisting translational motion.

Example: sphere. For a sphere of radius R, performing the integrals defining the second moment gives

$$\underset{\approx}{y}_p^{\text{sphere}} = \frac{1}{5}R^2\underset{\approx}{I} \tag{16.56}$$

and

$$\underset{\approx}{\mu}_p^{\text{sphere}} = \frac{2}{5}M R^2\underset{\approx}{I} \tag{16.57}$$

This result can be found in any dynamics textbook.

Example: ellipsoid. Suppose an ellipsoid is defined by three orthogonal vectors $\{\underset{\sim}{a}, \underset{\sim}{b}, \underset{\sim}{c}\}$ which equal the principal axes in both orientation and direction. We could perform the integrals as was done for the sphere, but the answer can be obtained more elegantly. We can construct a tensor $\underset{\approx}{G}$, whose columns contain the axis vectors, that maps a *unit* sphere to the ellipsoid such that

$$\underset{\sim}{x}^{\text{ellip}} = \underset{\sim}{x}_p^{\text{ellip}} + \underset{\approx}{G} \bullet (\underset{\sim}{X}^{\text{sphere}} - \underset{\sim}{X}_p^{\text{sphere}}) \tag{16.58}$$

then

$$\underset{\approx}{y}_p^{\text{ellipsoid}} = \frac{1}{5}\underset{\approx}{G} \bullet \underset{\approx}{y}_p^{\text{sphere}} \bullet \underset{\approx}{G}^T = \frac{1}{5}(\underset{\approx}{G} \bullet \underset{\approx}{G}^T) \tag{16.59}$$

and

$$\underset{\approx}{\mu}_p^{\text{ellipsoid}} = \frac{1}{5}M[(\underset{\approx}{G}:\underset{\approx}{G})\underset{\approx}{I} - \underset{\approx}{G} \bullet \underset{\approx}{G}^T] \tag{16.60}$$

The tensor $\underset{\approx}{G} \bullet \underset{\approx}{G}^T$ is easily constructed from the sum of axis dyads:

$$\underset{\approx}{G} \bullet \underset{\approx}{G}^T = \underset{\sim}{a}\underset{\sim}{a} + \underset{\sim}{b}\underset{\sim}{b} + \underset{\sim}{c}\underset{\sim}{c} \tag{16.61}$$

From which it follows that

$$\underset{\approx}{G}:\underset{\approx}{G} = \text{tr}(\underset{\approx}{G} \bullet \underset{\approx}{G}^T) = a^2 + b^2 + c^2 \tag{16.62}$$

The planar moment of inertia about $\underset{\sim}{a}$ is given by

$$\frac{\underset{\sim}{a} \bullet \underset{\approx}{\mu}_p^{\text{ellipsoid}} \bullet \underset{\sim}{a}}{a^2} = \frac{1}{5}M[b^2 + c^2] \tag{16.63}$$

The tensor $\underset{\approx}{\mu}_p$ is called the rotational moment of inertia tensor. Just as mass represents resistance of a body to translational acceleration, the inertia tensor quantifies resistance of a body to rotational acceleration.

Switching between the second moment and inertia tensor. The inertia tensor carries the same information as the second moment tensor. As a matter of fact, if $\underset{\approx}{\mu}_p$ is known, then the second moment may be immediately constructed by

$$\underset{\approx}{y}_p = \frac{1}{M}[\tfrac{1}{2}(\text{tr}\underset{\approx}{\mu}_p)\underset{\approx}{I} - \underset{\approx}{\mu}_p] \tag{16.64}$$

Center shift for the inertia tensor. Using (16.49), we note that $\underset{\approx}{\mu}_p$ can be computed from $\underset{\approx}{\mu}_c$ by

$$\boxed{\underset{\approx}{\mu}_p \equiv \underset{\approx}{\mu}_c + M[(\underset{\sim}{d}_{pc} \bullet \underset{\sim}{d}_{pc})\underset{\approx}{I} - \underset{\sim}{d}_{pc}\underset{\sim}{d}_{pc}]}, \text{ where } \underset{\sim}{d}_{pc} \equiv \underset{\sim}{x}_p - \underset{\sim}{x}_c \tag{16.65}$$

This result is a generalization of the parallel axis theorem cited in elementary dynamics textbooks. It is a generalization because we make no requirement of the existence of any parallel axes. Instead, the rotational moment of inertia with respect to an arbitrary point $\underset{\sim}{x}_p$ can be computed if the rotational moment of inertia about the CM is known. The traditional (specialized) parallel axis theorem is presented in Eq. (16.94)

Moment of inertia about a given axis. Given a plane of unit normal $\underset{\sim}{\boldsymbol{n}}$, the planar moment of inertia is defined by

$$\phi_{cn} \equiv \underset{\sim}{\boldsymbol{n}} \bullet \underset{\approx}{\mu}_c \bullet \underset{\sim}{\boldsymbol{n}} \tag{16.66}$$

or, in terms of the second moment,

$$\phi_{cn} = M[(\mathrm{tr}\,\underset{\approx}{\boldsymbol{y}}_c) - \underset{\sim}{\boldsymbol{n}} \bullet \underset{\approx}{\boldsymbol{y}}_c \bullet \underset{\sim}{\boldsymbol{n}}] \tag{16.67}$$

The planar moment of inertia ϕ_{cn} is a measure of how "difficult" it is to rotate the body about an axis parallel to $\underset{\sim}{\boldsymbol{n}}$ that passes through the center of mass.

Explicit matrices for the second moment and inertia tensor. To gain an intuitive feeling for the general structure of the various second moments suppose that we can set up a coordinate system with the origin at $\underset{\sim}{\boldsymbol{x}}_p$. Then we can write

$$\underset{\sim}{\boldsymbol{x}} - \underset{\sim}{\boldsymbol{x}}_p = x\underset{\sim}{\boldsymbol{e}}_x + y\underset{\sim}{\boldsymbol{e}}_y + z\underset{\sim}{\boldsymbol{e}}_z \tag{16.68}$$

The dyad $(\underset{\sim}{\boldsymbol{x}} - \underset{\sim}{\boldsymbol{x}}_p)(\underset{\sim}{\boldsymbol{x}} - \underset{\sim}{\boldsymbol{x}}_p)$ would then have a Cartesian component matrix given by

$$[(\underset{\sim}{\boldsymbol{x}} - \underset{\sim}{\boldsymbol{x}}_p)(\underset{\sim}{\boldsymbol{x}} - \underset{\sim}{\boldsymbol{x}}_p)] = \begin{bmatrix} x^2 & xy & xz \\ yx & y^2 & yz \\ zx & zy & z^2 \end{bmatrix} \tag{16.69}$$

Thus,

$$[\underset{\approx}{\boldsymbol{y}}_p] \equiv \frac{1}{M}\int_B \begin{bmatrix} x^2 & xy & xz \\ yx & y^2 & yz \\ zx & zy & z^2 \end{bmatrix} \rho\, dV \tag{16.70}$$

The moment of inertia tensor would then be given by

$$[\underset{\approx}{\mu}_p] = \int_B \left((x^2+y^2+z^2)\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} x^2 & xy & xz \\ yx & y^2 & yz \\ zx & zy & z^2 \end{bmatrix} \right) \rho\, dV \tag{16.71}$$

or

$$[\underset{\approx}{\mu}_p] = \int_B \begin{bmatrix} y^2+z^2 & -xy & -xz \\ -yx & z^2+x^2 & -yz \\ -zx & -zy & x^2+y^2 \end{bmatrix} \rho\, dV \tag{16.72}$$

The planar moment of inertia about the z-direction would then be

$$\phi_{cz} = \underset{\sim}{e}_z \bullet \underset{\approx}{\mu}_c \bullet \underset{\sim}{e}_z = \int_B (x^2 + y^2)\rho\, dV \tag{16.73}$$

Thus, ϕ_{cz} is the integral over the body of the distance from the z-axis, when the coordinate system is set up so that the z-axis passes through the CM. The off-diagonal components of the inertia tensor $\underset{\approx}{\mu}_c$ are called the "products" of inertia. Some books omit the negative sign in their definitions, which is fine so long as the negative is re-inserted whenever a change in coordinates is performed.

We will later see that the second moment plays a role in the conservation of angular momentum that is similar to the role played by the mass in the linear momentum equation. Mass is a measure of the body's "resistance" to changes in linear motion while the inertia tensor measures the body's resistance to changes in angular momentum.

Relationship between the current and initial moments. Conservation of mass guarantees that the zeroth moments are equal. That is,

$$\rho\, dV = \rho_o\, dV_o \tag{16.74}$$

Recall the definition of the center-of-mass:

$$\underset{\sim}{x}_c \equiv \frac{1}{M}\int_B \underset{\sim}{x}\rho\, dV \tag{16.75}$$

The second moment about the origin is defined

$$\underset{\approx}{y} \equiv \frac{1}{M}\int_B \underset{\sim}{x}\underset{\sim}{x}\rho\, dV \tag{16.76}$$

A rigid motion may be described in terms of the motion of the CM by

$$\underset{\sim}{x} = \underset{\sim}{x}_c + \underset{\approx}{R} \bullet (\underset{\sim}{X} - \underset{\sim}{X}_c) \tag{16.77}$$

Therefore

$$\begin{aligned}\underset{\sim}{x}\underset{\sim}{x} = {} & \underset{\sim}{x}_c\underset{\sim}{x}_c + \underset{\approx}{R} \bullet (\underset{\sim}{X}\underset{\sim}{x}_c - \underset{\sim}{X}_c\underset{\sim}{x}_c) + (\underset{\sim}{x}_c\underset{\sim}{X} - \underset{\sim}{x}_c\underset{\sim}{X}_c) \bullet \underset{\approx}{R}^T \\ & + \underset{\approx}{R} \bullet (\underset{\sim}{X}\underset{\sim}{X} - \underset{\sim}{X}_c\underset{\sim}{X} - \underset{\sim}{X}\underset{\sim}{X}_c + \underset{\sim}{X}_c\underset{\sim}{X}_c) \bullet \underset{\approx}{R}^T\end{aligned} \tag{16.78}$$

Substituting this into Eq. (16.76) and pulling out arguments that don't vary with position gives

$$\begin{aligned}\underset{\approx}{y} \equiv {} & \underset{\sim}{x}_c\underset{\sim}{x}_c + \underset{\approx}{R} \bullet (\underset{\sim}{X}_c\underset{\sim}{x}_c - \underset{\sim}{X}_c\underset{\sim}{x}_c) + (\underset{\sim}{x}_c\underset{\sim}{X}_c - \underset{\sim}{x}_c\underset{\sim}{X}_c) \bullet \underset{\approx}{R}^T \\ & + \underset{\approx}{R} \bullet (\underset{\approx}{y}^0 - \underset{\sim}{X}_c\underset{\sim}{X}_c - \underset{\sim}{X}_c\underset{\sim}{X}_c + \underset{\sim}{X}_c\underset{\sim}{X}_c) \bullet \underset{\approx}{R}^T\end{aligned} \tag{16.79}$$

Simplifying shows that

$$\underset{\approx}{y} \equiv \underset{\sim}{x}_c\underset{\sim}{x}_c + \underset{\approx}{R} \bullet (\underset{\approx}{y}^0 - \underset{\sim}{x}^0_c\underset{\sim}{x}^0_c) \bullet \underset{\approx}{R}^T \tag{16.80}$$

From the center shift theorem (derived below), we know that

$$\underset{\approx}{\boldsymbol{y}}_c^0 \equiv \underset{\approx}{\boldsymbol{y}}^0 - \underset{\sim}{\boldsymbol{x}}_c^0 \underset{\sim}{\boldsymbol{x}}_c^0 \tag{16.81}$$

Therefore

$$\boxed{\underset{\approx}{\boldsymbol{y}}_c \equiv \underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\boldsymbol{y}}_c^0 \bullet \underset{\approx}{\boldsymbol{R}}^T} \tag{16.82}$$

Now recall that

$$\underset{\approx}{\boldsymbol{y}}_p = \underset{\approx}{\boldsymbol{y}}_c + \underset{\sim}{\boldsymbol{d}}_{pc} \underset{\sim}{\boldsymbol{d}}_{pc} \tag{16.83}$$

And

$$\underset{\sim}{\boldsymbol{d}}_{pc} = \underset{\approx}{\boldsymbol{R}} \bullet \underset{\sim}{\boldsymbol{d}}_{pc}^0 \tag{16.84}$$

Therefore

$$\boxed{\underset{\approx}{\boldsymbol{y}}_p \equiv \underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\boldsymbol{y}}_p^0 \bullet \underset{\approx}{\boldsymbol{R}}^T} \tag{16.85}$$

In other words, the current second moment is related to the initial second moment by a simple rotation.

Moment center shift theorem. Recall the definition of the second moment about some point $\underset{\sim}{\boldsymbol{x}}_p$

$$\underset{\approx}{\boldsymbol{y}}_p \equiv \frac{1}{M}\int_B \underset{\sim}{\boldsymbol{d}}_{xp}\, \underset{\sim}{\boldsymbol{d}}_{xp} \rho\, dV \tag{16.86}$$

The goal of the moment center shift theorem is to find a formula that relates $\underset{\approx}{\boldsymbol{y}}_p$ to the (presumably already computed) second moment $\underset{\approx}{\boldsymbol{y}}_c$ that is centered about the CM. Written out explicitly, the integrand of Eq. (16.86) is

$$\underset{\sim}{\boldsymbol{d}}_{xp} \underset{\sim}{\boldsymbol{d}}_{xp} = (\underset{\sim}{\boldsymbol{x}} - \underset{\sim}{\boldsymbol{x}}_p)(\underset{\sim}{\boldsymbol{x}} - \underset{\sim}{\boldsymbol{x}}_p) = \underset{\sim}{\boldsymbol{x}}\underset{\sim}{\boldsymbol{x}} - \underset{\sim}{\boldsymbol{x}}\underset{\sim}{\boldsymbol{x}}_p - \underset{\sim}{\boldsymbol{x}}_p\underset{\sim}{\boldsymbol{x}} + \underset{\sim}{\boldsymbol{x}}_p\underset{\sim}{\boldsymbol{x}}_p \tag{16.87}$$

Noting that $\underset{\sim}{\boldsymbol{x}}_p$ is simply the location of a *particular point* and consequently does not vary with position, it may be pulled out of the integrals in Eq. (16.87) to give

$$\underset{\approx}{\boldsymbol{y}}_p \equiv \underset{\approx}{\boldsymbol{y}} - \underset{\sim}{\boldsymbol{x}}_c\underset{\sim}{\boldsymbol{x}}_p - \underset{\sim}{\boldsymbol{x}}_p\underset{\sim}{\boldsymbol{x}}_c + \underset{\sim}{\boldsymbol{x}}_p\underset{\sim}{\boldsymbol{x}}_p \tag{16.88}$$

If $\underset{\sim}{\boldsymbol{x}}_p$ coincides with $\underset{\sim}{\boldsymbol{x}}_c$, then Eq. (16.88) becomes

$$\underset{\approx}{\boldsymbol{y}}_c \equiv \underset{\approx}{\boldsymbol{y}} - \underset{\sim}{\boldsymbol{x}}_c\underset{\sim}{\boldsymbol{x}}_c \tag{16.89}$$

Thus, we may substitute $\underset{\approx}{\boldsymbol{y}} = \underset{\approx}{\boldsymbol{y}}_c + \underset{\sim}{\boldsymbol{x}}_c\underset{\sim}{\boldsymbol{x}}_c$ into Eq. (16.88) to obtain

$$\underset{\approx}{\boldsymbol{y}}_p \equiv \underset{\approx}{\boldsymbol{y}}_c + \underset{\sim}{\boldsymbol{d}}_{pc}\underset{\sim}{\boldsymbol{d}}_{pc}, \text{ where } \underset{\sim}{\boldsymbol{d}}_{pc} \equiv \underset{\sim}{\boldsymbol{x}}_p - \underset{\sim}{\boldsymbol{x}}_c \tag{16.90}$$

This result shows that $\underset{\approx}{\boldsymbol{y}}_p$ may be immediately constructed without having to perform any new integrals over the body so long as $\underset{\approx}{\boldsymbol{y}}_c$ and $\underset{\sim}{\boldsymbol{x}}_c$ are known. This is why the moments about the centroid are generally considered the "primitive" moments from which all other moments may be derived.

To remember Eq. (16.90), it helps to recognize that $\underset{\approx}{\boldsymbol{y}}_p$ is a measure of the "resistance" of a body to an angular acceleration. This resistance is the smallest when the axis of rotation passes through the CM. Thus, $\underset{\approx}{\boldsymbol{y}}_p$ must always be larger than $\underset{\approx}{\boldsymbol{y}}_c$. Hence, $\underset{\approx}{\boldsymbol{y}}_p$ is obtained from $\underset{\approx}{\boldsymbol{y}}_c$ by *adding* the dyad $\underset{\sim}{\boldsymbol{d}}_{pc}\underset{\sim}{\boldsymbol{d}}_{pc}$.

The parallel axis theorem. Recall the definition of the planar moment of inertia, which measures the resistance of a body to an angular acceleration about an axis parallel to $\underset{\sim}{\boldsymbol{n}}$ passing through the CM $\underset{\sim}{\boldsymbol{x}}_c$:

$$\phi_{cn} = M[(\mathrm{tr}\,\underset{\approx}{\boldsymbol{y}}_c) - \underset{\sim}{\boldsymbol{n}} \cdot \underset{\approx}{\boldsymbol{y}}_c \cdot \underset{\sim}{\boldsymbol{n}}] \tag{16.91}$$

We can similarly define the planar moment of inertia about a parallel axis that passes through a different point $\underset{\sim}{\boldsymbol{x}}_p$:

$$\phi_{pn} = M[(\mathrm{tr}\,\underset{\approx}{\boldsymbol{y}}_p) - \underset{\sim}{\boldsymbol{n}} \cdot \underset{\approx}{\boldsymbol{y}}_p \cdot \underset{\sim}{\boldsymbol{n}}] \tag{16.92}$$

We can substitute Eq. (16.90) into this equation to obtain

$$\phi_{pn} = M[(\mathrm{tr}[\underset{\approx}{\boldsymbol{y}}_c + \underset{\sim}{\boldsymbol{d}}_{pc}\underset{\sim}{\boldsymbol{d}}_{pc}]) - \underset{\sim}{\boldsymbol{n}} \cdot (\underset{\approx}{\boldsymbol{y}}_c + \underset{\sim}{\boldsymbol{d}}_{pc}\underset{\sim}{\boldsymbol{d}}_{pc})_p \cdot \underset{\sim}{\boldsymbol{n}}] \tag{16.93}$$

Rearranging gives the parallel axis theorem:

$$\phi_{pn} = \phi_{cn} + r_{pc}^2, \text{ where } r \equiv \sqrt{(\underset{\sim}{\boldsymbol{d}}_{pc} \cdot \underset{\sim}{\boldsymbol{d}}_{pc}) - (\underset{\sim}{\boldsymbol{d}}_{pc} \cdot \underset{\sim}{\boldsymbol{n}})^2} \tag{16.94}$$

Note that $\underset{\sim}{\boldsymbol{d}}_{pc} \cdot \underset{\sim}{\boldsymbol{d}}_{pc}$ is the square magnitude of $\underset{\sim}{\boldsymbol{d}}_{pc}$ and $\underset{\sim}{\boldsymbol{d}}_{pc} \cdot \underset{\sim}{\boldsymbol{n}}$ is the magnitude of the part of $\underset{\sim}{\boldsymbol{d}}_{pc}$ that is parallel to $\underset{\sim}{\boldsymbol{n}}$. Therefore, r is the perpendicular distance between the two parallel axes.

Linear momentum of a rigid body

The total momentum for a body B is defined by

$$\underset{\sim}{\boldsymbol{L}} = \int_B \underset{\sim}{\boldsymbol{v}}\, dm, \text{ where } \underset{\sim}{\boldsymbol{v}} = \dot{\underset{\sim}{\boldsymbol{x}}}, \text{ and } dm = \rho\, dV \tag{16.95}$$

Here, ρ is the mass density and dV is the volume element. We can substitute Eq. (16.10) into the above definition to obtain

$$\underset{\sim}{\boldsymbol{L}} = \int_B [\dot{\underset{\sim}{\boldsymbol{x}}}_c + \dot{\underset{\approx}{\boldsymbol{R}}} \cdot (\underset{\sim}{\boldsymbol{X}} - \underset{\sim}{\boldsymbol{X}}_c)]\rho\, dV \tag{16.96}$$

Now we recall that the rotation tensor $\underset{\approx}{\boldsymbol{R}}$ depends only on time, not position. Likewise, the velocity of the CM depends only on time, not position. Thus,

$$\underset{\sim}{\boldsymbol{L}} = M\dot{\underset{\sim}{\boldsymbol{x}}}_c + \dot{\underset{\approx}{\boldsymbol{R}}} \cdot \int_B \underset{\sim}{\boldsymbol{X}}\rho\, dV - \dot{\underset{\approx}{\boldsymbol{R}}} \cdot (M\underset{\sim}{\boldsymbol{X}}_c) \tag{16.97}$$

We recognize the remaining integral to equal M times the position of the centroid $\underset{\sim}{X}_c$. Therefore, the last two terms cancel leaving only

$$\underset{\sim}{L} = M\dot{\underset{\sim}{x}}_c \tag{16.98}$$

Thus, the angular spin of a rigid body has no effect on its linear momentum.

Angular momentum of a rigid body

The total angular momentum about the origin for a body B is defined by

$$\underset{\sim}{H} = \int_B \underset{\sim}{x} \times \underset{\sim}{v}\, dm \tag{16.99}$$

This may be written as

$$\underset{\sim}{H} = \underset{\approx}{\varepsilon} : \int_B \underset{\sim}{x}\underset{\sim}{v}\, dm \tag{16.100}$$

We can again substitute Eq. (16.11) into Eq. (16.100) to obtain

$$\begin{aligned}\int_B \underset{\sim}{x}\underset{\sim}{v}\, dm &= \int_B \underset{\sim}{x}[\dot{\underset{\sim}{x}}_c + \underset{\approx}{\Omega} \bullet (\underset{\sim}{x} - \underset{\sim}{x}_c)]\, dm \\ &= M\underset{\sim}{x}_c\dot{\underset{\sim}{x}}_c + \left(\int_B (\underset{\sim}{x}\underset{\sim}{x} - \underset{\sim}{x}\underset{\sim}{x}_c)\, dm\right) \bullet \underset{\approx}{\Omega}^T \\ &= M\underset{\sim}{x}_c\dot{\underset{\sim}{x}}_c + M(\underset{\approx}{y} - \underset{\sim}{x}_c\underset{\sim}{x}_c) \bullet \underset{\approx}{\Omega}^T \end{aligned} \tag{16.101}$$

Thus, noting that $\underset{\approx}{y} - \underset{\sim}{x}_c\underset{\sim}{x}_c = \underset{\approx}{y}_c$,

$$\int_B \underset{\sim}{x}\underset{\sim}{v}\, dm = M[\underset{\sim}{x}_c\dot{\underset{\sim}{x}}_c + \underset{\approx}{y}_c \bullet \underset{\approx}{\Omega}^T] \tag{16.102}$$

So the angular momentum becomes

$$\underset{\sim}{H} = M\underset{\approx}{\varepsilon} : (\underset{\sim}{x}_c\dot{\underset{\sim}{x}}_c + \underset{\approx}{y}_c \bullet \underset{\approx}{\Omega}^T) \tag{16.103}$$

The definition of the inertia tensor about any point $\underset{\sim}{x}_p$ is given in Eq. (16.54):

$$M\underset{\approx}{\varepsilon} : (\underset{\approx}{y}_p \bullet \underset{\approx}{\Omega}^T) = \underset{\approx}{\mu}_p \bullet \underset{\approx}{\omega} \tag{16.104}$$

With this, the angular momentum about the fixed laboratory origin becomes

$$\boxed{\underset{\sim}{H} = M\underset{\sim}{x}_c \times \dot{\underset{\sim}{x}}_c + \underset{\approx}{\mu}_c \bullet \underset{\sim}{\omega}} \tag{16.105}$$

Kinetic energy

The total kinetic energy for a body B is defined by

$$\dot{E} = \frac{1}{2}\int_B \underset{\sim}{v} \bullet \underset{\sim}{v}\, dm \tag{16.106}$$

or, using Eq. (__) for the velocity

$$\dot{E} = \tfrac{1}{2}\int_B [\dot{\underset{\sim}{x}}_c + \underset{\approx}{\Omega} \bullet (\underset{\sim}{x} - \underset{\sim}{x}_c)] \bullet [\dot{\underset{\sim}{x}}_c + \underset{\approx}{\Omega} \bullet (\underset{\sim}{x} - \underset{\sim}{x}_c)]\, dm$$

$$= \int_B \left(\frac{1}{2}(\dot{\underset{\sim}{x}}_c \bullet \dot{\underset{\sim}{x}}_c) + \dot{\underset{\sim}{x}}_c \bullet \underset{\approx}{\Omega} \bullet (\underset{\sim}{x} - \underset{\sim}{x}_c) + \frac{1}{2}\mathrm{tr}[\underset{\approx}{\Omega} \bullet (\underset{\sim}{x} - \underset{\sim}{x}_c)(\underset{\sim}{x} - \underset{\sim}{x}_c) \bullet \underset{\approx}{\Omega}^T] \right) dm$$

$$= \tfrac{1}{2} M v_c^2 + 0 + \frac{1}{2} M \mathrm{tr}[\underset{\approx}{\Omega} \bullet \underset{\approx}{\boldsymbol{y}}_c \bullet \underset{\approx}{\Omega}^T] \qquad (16.107)$$

It is straightforward to demonstrate that this can be simplified to

$$\boxed{\dot{E} = \tfrac{1}{2} M v_c^2 + \tfrac{1}{2} \underset{\sim}{\omega} \bullet \underset{\approx}{\mu}_c \bullet \underset{\sim}{\omega}} \text{, where } v_c^2 \equiv \dot{\underset{\sim}{x}}_c \bullet \dot{\underset{\sim}{x}}_c \qquad (16.108)$$

Note that the inertia tensor $\underset{\approx}{\mu}_c$ and the angular velocity $\underset{\sim}{\omega}$ play roles that are analogous to the roles played by the mass M and velocity v_c.

NEWTON'S EQUATION (balance of linear momentum)

The conservation of linear momentum states that the net force $\underset{\sim}{\boldsymbol{F}}$ acting on a rigid body must equal the rate of change of its linear momentum. That is

$$\underset{\sim}{\boldsymbol{F}} = \dot{\underset{\sim}{\boldsymbol{L}}} \tag{16.109}$$

Thus

$$\underset{\sim}{\boldsymbol{F}} = M\,\underset{\sim}{\boldsymbol{a}}_c, \text{ where } \underset{\sim}{\boldsymbol{a}}_c \equiv \ddot{\underset{\sim}{\boldsymbol{x}}}_c \tag{16.110}$$

In other words, the net force equals the total mass times the acceleration of the center of mass.

EULER'S EQUATION (balance of angular momentum)

The balance of angular momentum requires that the net torque (about the fixed Laboratory origin) must equal the rate of change of the angular momentum:

$$\underset{\sim}{\boldsymbol{T}} = \dot{\underset{\sim}{\boldsymbol{H}}} \tag{16.111}$$

Thus, taking rates of Eq. (16.105), gives

$$\underset{\sim}{\boldsymbol{T}} = M\,\underset{\sim}{\boldsymbol{x}}_c \times \ddot{\underset{\sim}{\boldsymbol{x}}}_c + \underset{\approx}{\mu}_c \bullet \dot{\underset{\sim}{\omega}} + \dot{\underset{\approx}{\mu}}_c \bullet \underset{\sim}{\omega} \tag{16.112}$$

Recall that

$$\underset{\approx}{\mu}_c = \underset{\approx}{\boldsymbol{R}} \bullet \underset{\approx}{\mu}^0_c \bullet \underset{\approx}{\boldsymbol{R}}^T \tag{16.113}$$

Taking rates (noting that $\underset{\approx}{\mu}^0_c$ does not vary with time and $\dot{\underset{\approx}{\boldsymbol{R}}} = \underset{\approx}{\Omega} \bullet \underset{\approx}{\boldsymbol{R}}$) gives

$$\dot{\underset{\approx}{\mu}}_c = \underset{\approx}{\Omega} \bullet \underset{\approx}{\mu}_c - \underset{\approx}{\mu}_c \bullet \underset{\approx}{\Omega} \tag{16.114}$$

Dotting from the right by $\underset{\sim}{\omega}$, noting that $\underset{\approx}{\Omega} \bullet \underset{\sim}{\omega} = \underset{\sim}{\boldsymbol{0}}$ permits Eq. (16.112) to be written

$$\underset{\sim}{\boldsymbol{T}} = M\,\underset{\sim}{\boldsymbol{x}}_c \times \ddot{\underset{\sim}{\boldsymbol{x}}}_c + \underset{\approx}{\mu}_c \bullet \dot{\underset{\sim}{\omega}} + \underset{\approx}{\Omega} \bullet \underset{\approx}{\mu}_c \bullet \underset{\sim}{\omega} \tag{16.115}$$

or

$$\boxed{\underset{\sim}{\boldsymbol{T}} = M\,\underset{\sim}{\boldsymbol{x}}_c \times \underset{\sim}{\boldsymbol{a}}_c + \underset{\approx}{\mu}_c \bullet \dot{\underset{\sim}{\omega}} + \underset{\sim}{\omega} \times (\underset{\approx}{\mu}_c \bullet \underset{\sim}{\omega})} \tag{16.116}$$

This result is the *direct notation* form of Euler's equations of motion. This expression may be applied using any convenient coordinate system

Many texts present a much specialized version of Euler's equation using an origin coinciding with the CM (so that $\underset{\sim}{\boldsymbol{x}}_c = \underset{\sim}{\boldsymbol{0}}$) and also using a basis aligned with the principal directions of the inertia tensor (so that the products of inertial are zero). In this special limiting case, the above equation can be written in component form as

$$T_1 = \mu_{11}\dot{\omega}_1 + \omega_3\omega_2(\mu_{33} - \mu_{22}) \tag{16.117}$$

$$T_2 = \mu_{22}\dot{\omega}_2 + \omega_1\omega_3(\mu_{11} - \mu_{33}) \tag{16.118}$$

$$T_3 = \mu_{33}\dot{\omega}_3 + \omega_2\omega_1(\mu_{22} - \mu_{11}), \tag{16.119}$$

which is the form cited most frequently in elementary dynamics books.

REFERENCES

[1]Goldstein, Herbert; Poole, Charles P. Jr; Safko, John L. (2002), **Classical Mechanics**, Columbia University. Addison Wesley.

[2]Argyris, J. (1982), *An excursion into large rotations.* Computer Methods in Applied Mechanics and Engineering **32**.

[3]Shoemake, K., *Uniform random rotations*, in: D. Kirk (Ed.), Graphics Gems III, Academic Press, London, 1992, pp. 124-132.

[4]Desloge, E.A., **Classical Mechanics**, V. 1&2, Florida State University, Wiley & Sons, New York, 1982.

[5]Wiley, C.R. and Barrett, L.C. (1982) **Advanced Engineering Mathematics** McGraw-Hill. {QA401.W9 1982, ISBN 0-070072188-2}

[6]Chadwick, P. (1999) **Continuum Mechanics: Concise Theory and Problems (second revised and enlarged version).** Dover. {QA808.C46, ISBN 0-486-40180-4 (pbk)}

[7]Malvern, L.E. (1969) **Introduction to the Mechanics of a Continuous Medium.** Prentice-Hall, Englewood Cliffs, New Jersey.

[8]Dienes, J.K., (1979) *On the analysis of Rotation and Stress Rate in Deforming Bodies.* Acta Mechanica 32, pp. 217-232.

[9]Eringen, A. Cemal. (1962), **Nonlinear Theory of Continuous Media**. McGraw-Hill.

[10]Schindler, W. (1997), *On the efficient simulation of a particular class of random rotations relevant to computer graphics.* J. Computational and Appl. Math. 81, pp. 107-114.

[11]Brannon, R.M., *Caveats concerning conjugate stress and strain measures for large distortion of anisotropic materials*, Acta Mechanica, v. 129(#1-2) pp. 107-116 1998.

[12]Shames, I.H. **Engineering Mechanics, VII Dynamics**, Prentice - Hall, Englewood Cliffs, New Jersey, 1980

[13]Lai, W. Michael; Rubin, David; Krempl, Erhard. **Continuum Mechanics, 3rd Ed.** Butterworth-Heinemann Ltd 1993. ISBN 0 7506 2894 4

[14]Brannon, R.M. http://www.me.unm.edu/~rmbrann/Tensors.pdf

APPENDIX A.
FORTRAN CODE LISTINGS

Listing 1: Testing whether a matrix is a rotation

```
 SUBROUTINE CHKROT(DC,IERR)
C     ********************************************************
C     This routine checks if a matrix DC is  a proper rotation.
C     To be orthogonal, the columns (or) rows of DC must form an
C     orthoNORMAL triad. That means that each column dotted with
C     itself must equal 1, and each column dotted with a different
C     column must equal 0. Furthermore, to be a proper rotation,
C     the columns of DC must form a RIGHT-HANDED orthonormal triad.
C     In other words, the determinant of DC must equal +1.
C     ********************************************************
C
C INPUT
C -----
C   DC: candidate rotation matrix
C
C OUTPUT
C ------
C  IERR = 0 if the matrix is a proper rotation
C
C       = -1 if the matrix is orthogonal but has a negative determinant
C
C       = ij if the dot product between column i and column j is wrong
C          For example, if column 3 is not normalized, then ierr=33.
C          If column 1 is not perpendicular to column 2, then ierr = 12.


      INCLUDE 'implicit.h'
C For double precison, the "implicit.h" include should contain one
C single line, depending on the desired precision. As appropriate,
C        IMPLICIT REAL(A-H,O-Z)             ! for single precision
C        IMPLICIT DOUBLE PRECISION(A-H,O-Z) ! for double precision


C ....parameters
C     real*8 pzero,pone
      PARAMETER (PZERO=0.0D0,PONE=0.1D1)
C ....passed
C     real*8 dc
      DIMENSION DC(3,3)
      INTEGER IERR
C
C ....local
      INTEGER I,J,K
C     real*8 dum
*************************************************************************
      IERR=0
C       The direction cosines are known
C       Check that the columns form an orthonormal set
C
        DO I=1,3
        DO J=1,3
          DUM=PZERO
          DO K=1,3
             DUM=DUM+DC(K,I)*DC(K,J)
          END DO
          IF(I.EQ.J)DUM=DUM-PONE
          IF(ABS(DUM).GT.0.001D0)THEN
             IERR=10*I+J
             RETURN
          END IF
        ENDDO
        ENDDO
C
```

```
C       Check if proper rotation by checking if det(dc)=+1
        DUM=DC(1,1)*DC(2,2)*DC(3,3)
     $     +DC(1,2)*DC(2,3)*DC(3,1)
     $     +DC(2,1)*DC(3,2)*DC(1,3)
     $     -DC(3,1)*DC(2,2)*DC(1,3)
     $     -DC(2,1)*DC(1,2)*DC(3,3)
     $     -DC(1,1)*DC(2,3)*DC(3,2)
        DUM=DUM-PONE
        IF(ABS(DUM).GT.0.001D0)THEN
           IERR=-1
           RETURN
        END IF
C
      IERR=0
      RETURN
      END
```

Listing 2: Converting axis and angle to direction cosines

```
c---.----1----.----2----.----3----.----4----.----5----.----6----.----7--
      SUBROUTINE AA2DC(axis,angle,dc,ierr)
C     This routine converts axis and angle of rotation to a direction
c     cosine matrix.
C
C input
C -----
C    AXIS: a unit vector in the direction of the axis of rotation
C    ANGLE: the angle of rotation (by right hand rule around AXIS)
C
C output
C -----
C    DC: the direction cosine matrix defined such that DC(i,j) equals
C        the dot product between the ith laboratory base vector and the
C        jth rotated base vector.
C
C  MODIFICATION HISTORY
C  980826:rmb:created routine
C
**************************************************************************
      INCLUDE 'implicit.h'
C.....parameters
c     real*8 pone
      parameter (pone=0.1d1)
c.....passed
      integer ierr
c     real*8 axis,angle,dc
      dimension axis(3),dc(3,3)
C.....function (functions instead of subroutines)
C.....external
c.....local (not saved)
c     real*8 c,s,omc
C.....local (saved)
C.....data
C.....statement functions
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      ierr=0

      c=cos(angle)
      s=sin(angle)
      omc=pone-c
      dc(1,1)=omc*axis(1)*axis(1)+c
      dc(2,2)=omc*axis(2)*axis(2)+c
      dc(3,3)=omc*axis(3)*axis(3)+c
      dc(1,2)=omc*axis(1)*axis(2)-s*axis(3)
      dc(2,3)=omc*axis(2)*axis(3)-s*axis(1)
      dc(3,1)=omc*axis(3)*axis(1)-s*axis(2)
      dc(2,1)=omc*axis(2)*axis(1)+s*axis(3)
      dc(3,2)=omc*axis(3)*axis(2)+s*axis(1)
      dc(1,3)=omc*axis(1)*axis(3)+s*axis(2)
```

```
      RETURN
      END
```

Listing 3: Converting direction cosines to axis and angle

The following listing uses c-preprocessor directives to select the method by which the task is performed. If METHOD1 is defined, then method 1 from page -36 is used. Otherwise, method 2 from page -40 is used.

```
#ifdef METHOD1
      SUBROUTINE DC2AA(DC,AXIS,ANGLE,ierr)
************************************************************************
      INCLUDE 'implicit.h'
c ....parameters
      parameter (pzero=0.0d0,pone=0.1d1,phalf=0.5d0)
      parameter (puny=0.1d-10)
c ....passed
      integer ierr
      dimension dc(3,3),axis(3)
c ....local
      integer i,j,k
      dimension scr(3,3)
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      ierr=0

c     The axis and angle are desired
      cth=(dc(1,1)+dc(2,2)+dc(3,3)-pone)*phalf
      sth=SQRT(pone-cth*cth)
      angle=acos(cth)
      if(abs(sth).gt.puny)then
          axis(1)=(dc(3,2)-dc(2,3))*phalf/sth
          axis(2)=(dc(1,3)-dc(3,1))*phalf/sth
          axis(3)=(dc(2,1)-dc(1,2))*phalf/sth
      elseif(cth.gt.pzero) then
c         The rotation tensor is the identity
c         and any axis will do.
          axis(1)=pone
          axis(2)=pzero
          axis(3)=pzero
      else
c         The rotation angle is exactly 180 degrees.
c         The rotation axis is therefore parallel to any nonzero
c         column of I+dc, where I is the identity matrix.
          do i=1,3
          do j=1,3
             scr(i,j)=dc(i,j)
          enddo
          enddo

          cth=pzero
          j=-12345
          do k=1,3
             scr(k,k)=scr(k,k)+pone
             sth=scr(1,k)**2+scr(2,k)**2+scr(3,k)**2
             if(sth.gt.cth)then
                cth=SQRT(sth)
                j=k
                scr(1,j)=scr(1,j)/cth
                scr(2,j)=scr(2,j)/cth
                scr(3,j)=scr(3,j)/cth
             endif
          enddo
          axis(1)=scr(1,j)
          axis(2)=scr(2,j)
```

```
          axis(3)=scr(3,j)
      endif
      return
      end

#else
      SUBROUTINE DC2AA(DC,AXIS,ANGLE,ierr)
************************************************************************
      INCLUDE 'implicit.h'
c ....parameters
      parameter (pzero=0.0d0,pone=0.1d1,ptwo=0.2d1,phalf=0.5d0)
c ....passed
      integer ierr
      dimension dc(3,3),axis(3)
c ....local
      integer i,j,k
      dimension scr(3,3)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      ierr=0

c     The axis and angle are desired
      dum=(dc(1,1)+dc(2,2)+dc(3,3)-pone)*phalf
      angle=acos(dum)
c
c     The rotation axis is  parallel to any nonzero
c     column of R+R^T+(1-trR)I, where I is the identity matrix.
c     and R^T is the tranpose of R. Here R is the rotation matrix DC.
      dum=-ptwo*dum
      do i=1,3
      do j=1,3
         scr(i,j)=dc(i,j)+dc(j,i)
      enddo
      scr(i,i)=scr(i,i)+dum
      enddo

      smag=pzero
      j=-12345
      do k=1,3
         dum=scr(1,k)**2+scr(2,k)**2+scr(3,k)**2
         if(dum.gt.smag)then
            smag=dum
            j=k
         endif
      enddo
      smag=sqrt(smag)
      axis(1)=sign(   scr(1,j)/smag,   dc(3,2)-dc(2,3)   )
      axis(2)=sign(   scr(2,j)/smag,   dc(1,3)-dc(3,1)   )
      axis(3)=sign(   scr(3,j)/smag,   dc(2,1)-dc(1,2)   )

      return
      end

#endif
```

Listing 4: Converting Euler angles to direction cosines.

```
c---.----1----.----2----.----3----.----4----.----5----.----6----.----7--
      SUBROUTINE EU2DC(EU,DC,ierr)
C
C INPUT
C -----
C   EU: The euler angles
C
C OUTPUT
c ------
c   DC: The direction cosine matrix.
c
**********************************************************************
C     PURPOSE: This routine ....
C
C  MODIFICATION HISTORY
C  mm/dd/yy:who:modification
C
      INCLUDE 'implicit.h'
C.....parameters
C.....common
c.....passed
      integer ierr
c     real*8 eu,dc
      dimension eu(3),dc(3,3)
C.....function (functions instead of subroutines)
C.....external
c.....local (not saved)
c     real*8 cphi,sphi,cth,sth,cpsi,spsi
C.....local (saved)
C.....data
C.....statement functions
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      ierr=0

      cphi=COS(eu(1))
      sphi=SIN(eu(1))
      cth =COS(eu(2))
      sth =SIN(eu(2))
      cpsi=COS(eu(3))
      spsi=SIN(eu(3))

      dc(1,1)=cphi*cpsi - cth*sphi*spsi
      dc(2,1)=cpsi*sphi + cphi*cth*spsi
      dc(3,1)=spsi*sth

      dc(1,2)=-(cpsi*cth*sphi) - cphi*spsi
      dc(2,2)=cphi*cpsi*cth - sphi*spsi
      dc(3,2)=cpsi*sth

      dc(1,3)=sphi*sth
      dc(2,3)=-(cphi*sth)
      dc(3,3)=cth

      RETURN
      END
```

Listing 5: Converting direction cosines to Euler angles.

```
C---.----1----.----2----.----3----.----4----.----5----.----6----.----7--
      SUBROUTINE DC2EU(DC,EULER,ierr)
c
c   This routine converts an orthogonal proper rotation matrix to euler
c   angles.
c
c INPUT
c -----
c  DC: direction cosine matrix for the rotation TENSOR
C      DC(i,j) is defined as the inner product between the ith lab
C      base vector with the jth rotated base vector.
c
c OUTPUT
c ------
C  EULER: euler angles {phi,theta,psi}, which describe the rotation
C     of the lab triad by the following procedure...First rotate
C     the triad an angle phi about its z-axis. Then rotate the new
C     triad an angle theta about its own new x-axis. Then rotate
C     the newer triad an angle psi about its own z-axis.
c  IERR: =0 if success, =1 otherwise
**********************************************************************
      INCLUDE 'implicit.h'
c ....parameters
c     real*8 pzero,pone,puny,small
      parameter (pzero=0.0d0,pone=0.1d1,small=0.1d-8,puny=0.1d-20)
c ....passed
      integer ierr
c     real*8 dc,euler
      dimension dc(3,3), euler(3)
c ....local
c     real*8 dum,dum11,dum12,dum21,dum22,cth,sthsq,theta,psi,phi
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      cth=dc(3,3)
      sthsq=pone-cth*cth
      if(sthsq.lt.puny)then
        theta=acos(cth)
        psi=pzero
        phi=atan2(dc(2,1),dc(1,1))
        return
      endif

      dum11=dc(1,1)
     $   +((dc(2,3)*dc(3,2) + dc(1,3)*dc(3,1)*dc(3,3))/sthsq)
      dum12=dc(1,2)
     $   -(dc(2,3)*dc(3,1) - dc(1,3)*dc(3,2)*dc(3,3))/sthsq
      dum21=dc(2,1)
     $   -(dc(1,3)*dc(3,2) - dc(2,3)*dc(3,1)*dc(3,3))/sthsq
      dum22=dc(2,2)
     $   +(dc(1,3)*dc(3,1) + dc(2,3)*dc(3,2)*dc(3,3))/sthsq

      dum=sqrt(dum11**2+dum22**2+dum12**2+dum21**2)
      if(dum.gt.small)then
c       The DC matrix is not proper orthogonal
        ierr=1
        return
      endif
      theta=acos(cth)
      phi=atan2(dc(1,3),-dc(2,3))
      psi=atan2(dc(3,1), dc(3,2))
      euler(1)=phi
      euler(2)=theta
      euler(3)=psi

      ierr=0
      return
      end
```

Listing 6: Generating a uniformly random unit normal

```
 SUBROUTINE RANDA(AXIS)
c     This routine outputs a random unit normal
c
c OUTPUT
c ------
C    AXIS: The random unit axis.
C
**********************************************************************
C
      INCLUDE 'implicit.h'
c ....parameters
c     real*8 pi
      parameter (pi=0.3141592653589793d1)
c     real*8 pone,ptwo
      parameter (pone=0.1d1,ptwo=0.2d1)
c ....passed
c     real*8 axis
      dimension axis(3)
c ....functions
c     real*8 rand
c ....local
c     real*8 phi, theta
c ....local (SAVED)
c     real*8 seed
c     save seed
c     data seed/0.0/
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      phi=ptwo*pi*drand(0)
      theta=ACOS(pone-ptwo*drand(0))
      axis(1)=sin(theta)*cos(phi)
      axis(2)=sin(theta)*sin(phi)
      axis(3)=cos(theta)
      return
      end
```

Listing 7: Generating a uniformly random rigid rotation.

```
C This file contains two subroutines, RANDC and RNDC, both of
C which compute a uniformly random rotation tensor. The only difference
C between them is the way that they do it.
C
C The routine randc computes the rotation by generating two
C uniformly random unit vectors, orthonormalizing them,
C and then generating the third vector by a right-hand cross product.
C
C The routine rndc computes the rotation via Shoemake's algorithm.
C
      SUBROUTINE RANDDC(R)
C This routine calls either RANDC or RNDC, depending on how the
C programmer desires the rotation to be generated.
      INCLUDE 'implicit.h'
C     real*8 r
      DIMENSION R(3,3)
      CALL RANDC(R)
      RETURN
      END
C
C
C
C
**********************************************************************
      SUBROUTINE RANDC(R)
C This routine computes a uniformly random rigid rotation tensor
C This routine calls randa to generate a uniformly random
C direction for the rotated e1 axis.  Then randa is called again
C to generate a second random axis. The rotated e2 is taken to
C equal the projection of the second random direction to the
C plane with normal equal to e1. Finally, the third direction is
C given by the right hand cross product of e1 x e2.
C
C OUTPUT
C ------
C  R: The direction cosine tensor components.  The ij component
C     of R equals the ith lab base vector dotted with the jth rotated
C     base vector.
**********************************************************************
      INCLUDE 'implicit.h'
C     real*8 small,r,dum
      PARAMETER (SMALL=.1D0)
      DIMENSION R(3,3)
      EXTERNAL RANDA
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL RANDA(R(1,1))
    7 CALL RANDA(R(1,2))
C     subtract off the part of e2 that is in the direction of e1
      DUM=R(1,1)*R(1,2)+R(2,1)*R(2,2)+R(3,1)*R(3,2)
      R(1,2)=R(1,2)-DUM*R(1,1)
      R(2,2)=R(2,2)-DUM*R(2,1)
      R(3,2)=R(3,2)-DUM*R(3,1)
      DUM=SQRT(R(1,2)**2+R(2,2)**2+R(3,2)**2)
C
C     make certain that the two vectors are independent.
C     If dum is small, then the second random vector is nearly
C     parallel to the first, and we need to sample again.
      IF(DUM.LT.SMALL)GO TO 7
C
      R(1,2)=R(1,2)/DUM
      R(2,2)=R(2,2)/DUM
      R(3,2)=R(3,2)/DUM
C
      R(1,3)=R(2,1)*R(3,2)-R(3,1)*R(2,2)
      R(2,3)=R(3,1)*R(1,2)-R(1,1)*R(3,2)
      R(3,3)=R(1,1)*R(2,2)-R(2,1)*R(1,2)
C
      RETURN
      END
C
```

```
C
**********************************************************************
      SUBROUTINE RNDC(R)
C This routine computes a uniformly random rigid rotation tensor
C using Shoemake's method.
C
C OUTPUT
C ------
C  R: The direction cosine tensor components.  The ij component
C     of R equals the ith lab base vector dotted with the jth rotated
C     base vector.
**********************************************************************
      INCLUDE 'implicit.h'
      PARAMETER (ONE=0.1D1,TWO=0.2D1,
     & PI=0.31415926535897932384626433832795028D1,
     & TWOPI=TWO*PI)
C     real*8 r
      DIMENSION R(3,3)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      X0=drand(0)
      Y1=TWOPI*drand(0)
      Y2=TWOPI*drand(0)
C
      R1=SQRT(ONE-X0)
      R2=SQRT(X0)
      U0=COS(Y2)*R2
      U1=SIN(Y1)*R1
      U2=COS(Y1)*R1
      U3=SIN(Y2)*R2
C
      COEFI = TWO*U0*U0-ONE
      COEFUU= TWO
      COEFE = TWO*U0
C
      R(1,1)=COEFI+COEFUU*U1*U1
      R(2,2)=COEFI+COEFUU*U2*U2
      R(3,3)=COEFI+COEFUU*U3*U3
C
      R(2,3)=COEFUU*U2*U3-COEFE*U1
      R(3,1)=COEFUU*U3*U1-COEFE*U2
      R(1,2)=COEFUU*U1*U2-COEFE*U3
C
      R(3,2)=COEFUU*U3*U2+COEFE*U1
      R(1,3)=COEFUU*U1*U3+COEFE*U2
      R(2,1)=COEFUU*U2*U1+COEFE*U3
C
      RETURN
      END
C
C
```

APPENDIX B.
Tensor and vector notation

This appendix outlines the essential vector and tensor concepts used in this report. A far more extensive tutorial on tensor analysis may be found in Reference [14]. Throughout this report, scalars are denoted in plain italics (s, r, t). Vectors are typeset with a single under-tilde ($\underset{\sim}{v}, \underset{\sim}{w}, \underset{\sim}{x}$). Second-order tensors are shown with two under-tildes ($\underset{\approx}{\sigma}, \underset{\approx}{S}, \underset{\approx}{T}$). Likewise, the order of higher-order tensors is indicated by the number of under-tildes.

Two vectors written side-by-side are multiplied dyadically. For example, $\underset{\sim}{a}\underset{\sim}{b}$ is a second-order tensor with ij components given by $a_i b_j$. Any second-order tensor $\underset{\approx}{T}$ may be expanded in terms of basis dyads as $\underset{\approx}{T} = T_{ij}\underset{\sim}{e}_i\underset{\sim}{e}_j$. Here (and throughout this report) all free indices range from 1 to 3. Furthermore, repeated indices imply summation from 1 to 3.

A single raised dot denotes the vector inner-product defined by

$$\underset{\sim}{u} \cdot \underset{\sim}{v} = u_1 v_1 + u_2 v_2 + u_3 v_3 = u_k v_k . \tag{B.1}$$

The single raised dot continues to denote the vector inner product even when acting between higher-order tensors. For example,

$$\underset{\approx}{A} \cdot \underset{\sim}{x} = A_{ij} x_j \ \underset{\sim}{e}_i \ . \tag{B.2}$$

Composition of two tensors is another example:

$$\underset{\approx}{A} \cdot \underset{\approx}{B} = A_{ik} B_{kj} \ \underset{\sim}{e}_i\underset{\sim}{e}_j . \tag{B.3}$$

The deviatoric part of a tensor is denoted by a "prime." Hence,

$$\underset{\approx}{A}' \equiv \underset{\approx}{A} - \tfrac{1}{3}(\mathrm{tr}\underset{\approx}{A})\underset{\approx}{I} , \tag{B.4}$$

where $\underset{\approx}{I}$ is the identity tensor and "tr" denotes the trace. Specifically,

$$\mathrm{tr}\underset{\approx}{A} \equiv A_{11} + A_{22} + A_{33} = A_{kk} . \tag{B.5}$$

The tensor inner product is denoted by "$:$" and is defined by

$$\underset{\approx}{A} : \underset{\approx}{B} = A_{ij} B_{ij} . \tag{B.6}$$

Note that

$$\underset{\approx}{A} : \underset{\approx}{B} = \underset{\approx}{B} : \underset{\approx}{A} . \tag{B.7}$$

The magnitude of a second-order tensor is defined

$$\| \underset{\approx}{A} \| \equiv \sqrt{\underset{\approx}{A} : \underset{\approx}{A}} . \tag{B.8}$$

The tensor inner product is allowed to operate between any two tensors of *at least* second order. For example, if $\underset{\approx}{\underset{\approx}{E}}$ is a fourth-order tensor, then

$$\underset{\approx}{\underset{\approx}{\boldsymbol{E}}}:\underset{\sim}{\boldsymbol{A}} = E_{ijkl}A_{kl}\ \underset{\sim}{\boldsymbol{e}}_i\underset{\sim}{\boldsymbol{e}}_j . \tag{B.9}$$

The remainder of this appendix provides some introductory remarks about the vector and tensor analysis, and may be skipped by readers familiar with elementary concepts in those fields of study.

Vectors

A first course in calculus usually vaguely defines an *engineering* vector as "something with length and direction," such as velocity, electric field, or displacement. The vague notion of a vector is then refined by showing how a vector can be described by a set of three components referenced to three base directions. A mathematician defines a vector more generally as a member of a set for which addition and scalar multiplication are *closed* and satisfy certain properties. By closed, we mean that adding two vectors gives a vector and multiplying a vector times a scalar gives a vector. This sounds simple enough, but proving closure is typically the most awkward and difficult step because it requires a clear discerning definition of membership in the proposed set. An ordinary engineering vector is an entity consisting of three numbers (components) and three non-parallel reference directions (basis). The components must change in a particular manner upon a change in basis. It is this "change-of-basis" rule that is used as the discerning definition of an engineering vector. When the three reference directions are mutually perpendicular, orthogonal matrices play a predominant role.

If $\{\underset{\sim}{\boldsymbol{E}}_1, \underset{\sim}{\boldsymbol{E}}_2, \underset{\sim}{\boldsymbol{E}}_3\}$ are base vectors then any engineering vector $\underset{\sim}{\boldsymbol{v}}$ may be written

$$\underset{\sim}{\boldsymbol{v}} = v_1\underset{\sim}{\boldsymbol{E}}_1 + v_2\underset{\sim}{\boldsymbol{E}}_2 + v_3\underset{\sim}{\boldsymbol{E}}_3 \tag{B.10}$$

where $\{v_1, v_2, v_3\}$ are the "components" of the vector $\underset{\sim}{\boldsymbol{v}}$ with respect to the $\{\underset{\sim}{\boldsymbol{E}}_1, \underset{\sim}{\boldsymbol{E}}_2, \underset{\sim}{\boldsymbol{E}}_3\}$ basis. If $\{\underset{\sim}{\boldsymbol{e}}_1, \underset{\sim}{\boldsymbol{e}}_2, \underset{\sim}{\boldsymbol{e}}_3\}$ is a different choice for the basis, the *same vector* may be written

$$\underset{\sim}{\boldsymbol{v}} = \tilde{v}_1\underset{\sim}{\boldsymbol{e}}_1 + \tilde{v}_2\underset{\sim}{\boldsymbol{e}}_2 + \tilde{v}_3\underset{\sim}{\boldsymbol{e}}_3 , \tag{B.11}$$

where $\{\tilde{v}_1, \tilde{v}_2, \tilde{v}_3\}$ are the components of the vector $\underset{\sim}{\boldsymbol{v}}$ with respect to the $\{\underset{\sim}{\boldsymbol{e}}_1, \underset{\sim}{\boldsymbol{e}}_2, \underset{\sim}{\boldsymbol{e}}_3\}$ basis. If the two bases are orthonormal, then the v_i components are related to the $\tilde{v}_j$ components in a particular way, as explained in the main text.

Tensors

A tensor is often described as a linear transformation that takes vectors to vectors. Consider, for example, a function f that transforms a vector into a new vector 3 times as long, but with the same orientation. In other words, consider

$$f(\underset{\sim}{v}) = 3\underset{\sim}{v} \tag{B.12}$$

This function is linear because (as can be easily verified)

$$\begin{aligned} &f(\alpha\underset{\sim}{v}) = \alpha f(\underset{\sim}{v}) \text{ for all scalars } \alpha \\ &\text{and } f(\underset{\sim}{v}+\underset{\sim}{w}) = f(\underset{\sim}{v})+f(\underset{\sim}{w}) \text{ for all vectors } \underset{\sim}{v} \text{ and } \underset{\sim}{w}. \end{aligned} \tag{B.13}$$

Thus, we now know that "f" is a tensor. However, f is just the transformation rule. When most people talk about "tensors," they are referring to something that is described by a 3×3 matrix of components that are referenced to a basis, similar to the way that vector components are referenced to a basis. Before discussing how to convert f into a 3×3 matrix of components, consider the following less trivial example of a linear operator that takes vectors to vectors: Suppose $\underset{\sim}{c}$ is a known constant vector. Then the vector cross product

$$f(\underset{\sim}{v}) = \underset{\sim}{c}\times\underset{\sim}{v} \tag{B.14}$$

is a vector-to-vector transformation is that is linear with respect to $\underset{\sim}{v}$ because it also satisfies the linearity conditions of Eq. (B.13). Thus, we now know that this "f" is a tensor. Here it is even less clear how to construct the 3×3 matrix associated with this tensor.

When the transformed vectors are ordinary engineering vectors a tensor "f" is describable by a 3×3 matrix referenced to a basis. To see why, let's look at the linearity property itself. Let $\{\underset{\sim}{E}_1, \underset{\sim}{E}_2, \underset{\sim}{E}_3\}$ denote some convenient orthonormal basis. Now let $\underset{\sim}{v}$ be some vector. Since the $\{\underset{\sim}{E}_1, \underset{\sim}{E}_2, \underset{\sim}{E}_3\}$ vectors form a basis, then we know that *there must exist* three components $\{v_1, v_2, v_3\}$ such that

$$\underset{\sim}{v} = v_1\underset{\sim}{E}_1 + v_2\underset{\sim}{E}_2 + v_3\underset{\sim}{E}_3 \tag{B.15}$$

Because the function f is linear, we can apply Eq. (B.13) to write

$$\begin{aligned} &f(\underset{\sim}{v}) = v_1\underset{\sim}{g}_1 + v_2\underset{\sim}{g}_2 + v_3\underset{\sim}{g}_3, \\ &\text{where } \underset{\sim}{g}_k = f(\underset{\sim}{E}_k) \end{aligned} \tag{B.16}$$

Being vectors themselves, each of the $\underset{\sim}{g}_k$ vectors must be expressible as a linear combination of the $\{\underset{\sim}{E}_1, \underset{\sim}{E}_2, \underset{\sim}{E}_3\}$ basis. In other words, we know there must exist components F_{jk} such that

$$\underset{\sim}{g}_k = F_{1k}\underset{\sim}{E}_1 + F_{2k}\underset{\sim}{E}_2 + F_{3k}\underset{\sim}{E}_3 \tag{B.17}$$

The index k takes values from 1 to 3, so $[F_{ij}]$ is a 3×3 matrix whose columns contain the components of the three $\underset{\sim}{\boldsymbol{g}}_k$ vectors with respect to the $\{\underset{\sim}{\boldsymbol{E}}_1, \underset{\sim}{\boldsymbol{E}}_2, \underset{\sim}{\boldsymbol{E}}_3\}$ basis. If $\underset{\sim}{\boldsymbol{w}} = f(\underset{\sim}{\boldsymbol{v}})$ and if $\underset{\sim}{\boldsymbol{w}} = w_1\underset{\sim}{\boldsymbol{E}}_1 + w_2\underset{\sim}{\boldsymbol{E}}_2 + w_3\underset{\sim}{\boldsymbol{E}}_3$, then putting (B.17) into (B.16) shows that

$$\begin{Bmatrix} w_1 \\ w_2 \\ w_3 \end{Bmatrix} = \begin{bmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{bmatrix} \begin{Bmatrix} v_1 \\ v_2 \\ v_3 \end{Bmatrix} \tag{B.18}$$

Thus, all of the information about the linear transformation function f is contained completely in the 3×3 $[F_{ij}]$ matrix. Once the $[F_{ij}]$ component matrix is known, computing $\underset{\sim}{\boldsymbol{w}}$ is simply a matter of matrix multiplication. In direct notation, Eq. (B.18) is written

$$\underset{\sim}{\boldsymbol{w}} = \underset{\approx}{\boldsymbol{F}} \bullet \underset{\sim}{\boldsymbol{v}}, \tag{B.19}$$

where

$$\underset{\approx}{\boldsymbol{F}} = F_{ij}\underset{\sim}{\boldsymbol{E}}_i\underset{\sim}{\boldsymbol{E}}_j \tag{B.20}$$

The nine possible $\underset{\sim}{\boldsymbol{E}}_i\underset{\sim}{\boldsymbol{E}}_j$ pairs are called the tensor basis dyads. The basis dyad $\underset{\sim}{\boldsymbol{E}}_i\underset{\sim}{\boldsymbol{E}}_j$ is a special tensor whose associated matrix has zeros everywhere except a 1 in the ij position. In practice, when someone speaks of a "tensor," they are usually referring to $\underset{\approx}{\boldsymbol{F}}$ instead of the original transformation rule f.

The component matrix for the tensor associated with Eq. (B.12) is

$$[\underset{\approx}{\boldsymbol{F}}] = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 3 \end{bmatrix} \tag{B.21}$$

The component matrix for the tensor associated with Eq. (B.14) is

$$[\underset{\approx}{\boldsymbol{F}}] = \begin{bmatrix} 0 & -c_3 & c_2 \\ c_3 & 0 & -c_1 \\ -c_2 & c_1 & 0 \end{bmatrix} \tag{B.22}$$

In general, given a tensor transformation rule $f(\underset{\sim}{\boldsymbol{v}})$ expressed in component form such that $w_i = f(v_1, v_2, v_3)$. The tensor component matrix may be found by

$$F_{ij} = \frac{\partial w_i}{\partial v_j} \tag{B.23}$$