

Dedicated to the Memory of DR. Meng-Sing Liou (NASA Glenn Research Center: 1987-2017)

# Coordinate Transformations and High-Order Meshes with Mathematica

H. T. Huynh <sup>1</sup>

NASA Glenn Research Center

Cleveland, OH 44135, USA

Corresponding author: [huynh@grc.nasa.gov](mailto:huynh@grc.nasa.gov)

**Abstract:** The derivation of coordinate transformations and various metric quantities for the case of arbitrary high-order order meshes is carried out using the symbolic manipulation, three-dimensional plotting, and object rotating capabilities of Mathematica. The geometric description, relations among the metric quantities, and the metric identity are derived in a manner considerably simpler than the standard derivation. The resulting Mathematica program facilitates the visualization of these quantities and may enhance understanding and/or improve intuition on coordinate transformation.

*Keywords:* Coordinate transformation, high-order meshes, metric quantities, geometric description, visualization.

## 1 Introduction

Recently, there has been a significant international effort to develop high-order methods in the field of Computational Fluid Dynamics (CFD) with the goal of incorporating them into design tools for aerospace industries. Evidence of such efforts includes: the series of International Workshop on High-Order CFD methods I-V, the TILDA project (Towards Industrial LES/DNS in Aeronautics) supported by the European Commission, NASA RCA (Revolutionary Computational Aerosciences) subproject, as well as the numerous papers on high-order CFD methods presented at conferences worldwide. These methods require solid curved boundaries to be represented by high-order polynomials, resulting in high-order meshes. Generating such meshes is currently a focus at research institutions and commercial CFD and mesh-generation companies. In addition, standard plotting packages deal only with linear meshes. Plotting tools and visualization for high-order solutions remain to be developed.

For high-order as well as many second-order accurate methods, flow field calculations are carried out using coordinate transformations between reference and physical domains. The metric quantities, i.e., quantities associated with the derivatives of the transformation, are typically obtained algebraically. Their geometric descriptions and relations are not always easily grasped. These metric terms also satisfy a set of identities known as “free stream preservation”, a subject widely studied in the finite difference

---

<sup>1</sup> The author has no relations or financial ties with Mathematica or the Wolfram Company. He only uses the software package as a tool for his research.

community (Thomas and Lombard 1979, Thompson et al. 1982, 1985). For the case of high-order, free stream preservation was investigated in (Kopriva 2006) and (Abe et al. 2015).

In this paper, the derivation of coordinate transformations and metric quantities for arbitrary high-order meshes is carried out using the symbolic manipulation, three-dimensional plotting, and object rotating capabilities of Mathematica. The geometric description, relations among the metric quantities, and the metric identity are derived in a manner considerably simpler than the standard derivation (e.g., Thompson et al. 1985). The resulting Mathematica program facilitates the visualization of these quantities and may enhance understanding and/or improve intuition on coordinate transformation.

This paper is self-contained and organized as follows. Transformation, metric quantities, and their relations are presented in Section 2. The divergence formula in the reference frame and the fundamental metric identity are derived in Section 3. Conclusions can be found in Section 4. Finally, the Appendix contains the Mathematica program, which can be copied in pdf format and paste into a Mathematica notebook.

## 2 Transformation and Metric Quantities

### 2.1 Reference (Master) Domain

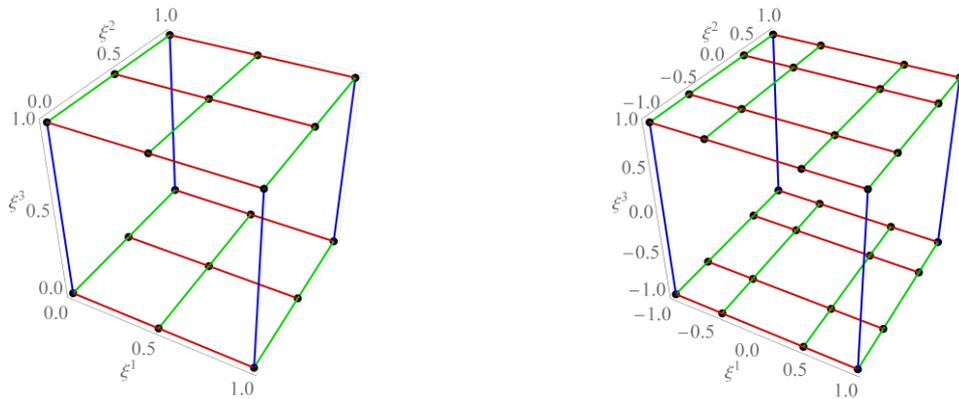
Consider the case of hexahedral meshes. For such a mesh, the reference or master domain is the unit or the bi-unit cube  $C$  as shown in Figure 2.1. The reference interval for the unit cube is  $I = [0, 1]$  and for the biunit cube,  $I = [-1, 1]$ . The coordinates in the reference domain are denoted interchangeably by (note the superscripts and the bold face letter for a vector)

$$\mathbf{\Xi} = (\xi^1, \xi^2, \xi^3) = (\xi, \eta, \zeta).$$

The top and bottom surfaces are mapped to curved surfaces in the physical domain. (The program can be easily modified to allow for high-order vertical sides; however, only high-order top and bottom surfaces are considered here for ease of observation and since most solid curved surfaces in applications can be accommodated by meshes of this type.)

Let  $M \geq 2$  be the number of points per line segment for the (high-order) top and bottom surfaces. Let the indices  $i, j, k$  correspond to the grid points in the  $\xi, \eta,$  and  $\zeta$  directions, respectively, such that  $1 \leq i, j \leq M$  and  $1 \leq k \leq 2$ . The grid points are represented by the dots in Figure 2.1. The case  $M = 2$  corresponds to the standard cube defined by 8 corner vertices.

Let  $\xi_i$  and  $\eta_j$ ,  $1 \leq i, j \leq M$ , be either the equidistant points including the two boundaries or the Lobatto points on  $I$ . Thus  $\xi_1 = 0$  for the unit cube whereas  $\xi_1 = -1$  for the biunit cube, and  $\xi_M = 1$  for both. A similar statement holds for  $\eta$ .



(a) Unit cube;  $M = 3$ ; 3-equidistant points per line segment on top and bottom surfaces.

(b) Bi-unit cube;  $M = 4$ ; 4-Lobatto points per line segment on top and bottom surfaces.

Figure 2.1: The cube  $C$  in the reference domain, (a) unit and (b) bi-unit. Red, green, and blue line segments are lines with varying  $\xi, \eta,$  and  $\zeta$ , respectively.

For ease of discussion, the reference interval is set to  $I = [-1, 1]$  in the remainder of this paper, i.e., we will only consider the case of the biunit cube; the unit cube case involves a straightforward scaling and appropriate  $\xi_1$ .

## 2.2 Physical Domain

Curved surfaces can be defined analytically or numerically as output of a Computer Aided Design (CAD) program. From such definitions, we can obtain the point data for the curved mesh.

Figure 2.2 shows (a) spherical surfaces and (b) perturbed spherical surfaces, both are analytical. From here on, it is assumed that the point data represented by the dots on the curved surfaces are given. They are called grid points.

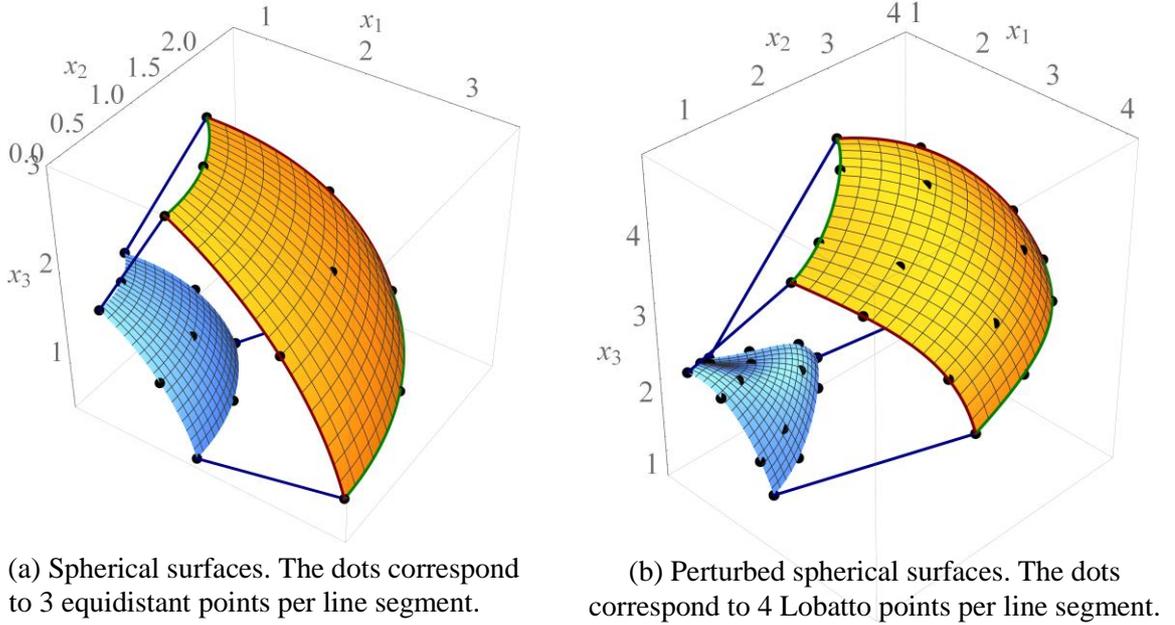


Figure 2.2: Physical domains (hexahedron with curved faces) and grid points.

## 2.3 Transformations

A point in the physical space is denoted interchangeably by (note the subscripts)

$$\mathbf{X} = (x_1, x_2, x_3) = (x, y, z).$$

Consider the points  $\xi_n$ ,  $1 \leq n \leq M$ , which are the Lobatto or equidistant points. For each  $n$ , let  $l_n$  be the corresponding Lagrange polynomial, i.e.,  $l_n$  is of degree  $M - 1$  taking on the value 1 at  $\xi_n$  and 0 at all other  $\xi_m$ ,

$$l_n(\xi) = \prod_{\substack{m=1 \\ m \neq n}}^M \frac{\xi - \xi_m}{\xi_n - \xi_m}. \quad (2.1)$$

Figure 2.3 shows the Lagrange polynomials on  $I = [-1, 1]$  for (a) eleven Lobatto points and (b) eleven equidistant points. For the general case of  $M$  Lobatto points, these polynomials have the peculiar property that for each  $n$ , the Lagrange polynomial  $l_n$  attains its maximum value of 1 on  $I$  at  $\xi_n$ . Note that the Lagrange polynomials defined by equidistant points do not have this property; their maximum take place near the boundary and grow with the number of points as can be seen in Fig. 2.3(b).

The  $\eta$  direction is similar to that of  $\xi$ . As for the  $\zeta$  direction, with the superscript (1) representing degree 1, let  $l_1^{(1)}$  and  $l_2^{(1)}$  be the linear Lagrange polynomials for the points  $\zeta_1 = -1$  and  $\zeta_2 = 1$ , respectively:

$$l_1^{(1)}(\zeta) = \frac{1 - \zeta}{2} \quad \text{and} \quad l_2^{(1)}(\zeta) = \frac{1 + \zeta}{2}. \quad (2.2)$$

If the four side faces are of the same order as the top and bottom faces, the Lagrange polynomials  $l_n^{(1)}$  above must be replaced by  $l_n$  of (2.1).

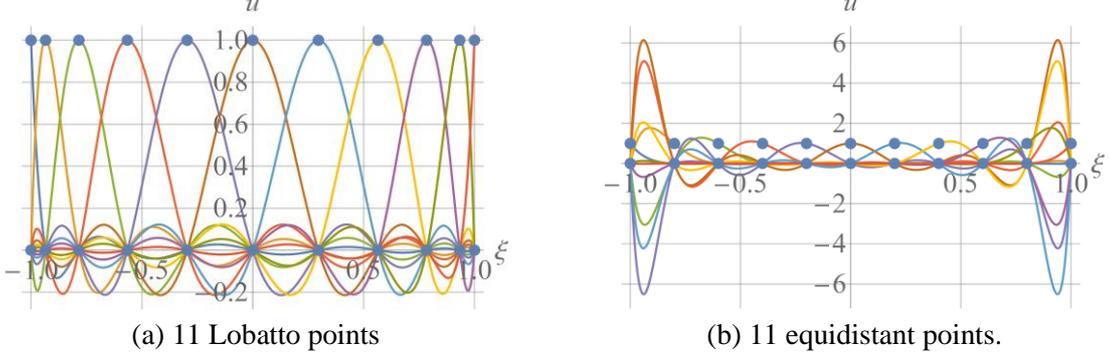


Figure 2.3: Lagrange polynomials.

Let the grid points  $\mathbf{X}_{i,j,k} = (x_i, y_j, z_k)$ ,  $1 \leq i, j \leq M$  and  $1 \leq k \leq 2$ , in the physical space be given; there are  $2M^2$  such points. A hexahedron with curved faces defined by polynomials is called a polynomial hexahedron here. The transformation from the biunit cube to the polynomial hexahedron is defined by:

$$\mathbf{X}(\mathbf{\Xi}) = \mathbf{X}(\xi, \eta, \zeta) = \sum_{i=1}^M \sum_{j=1}^M \sum_{k=1}^2 l_i(\xi) l_j(\eta) l_k^{(1)}(\zeta) \mathbf{X}_{i,j,k}. \quad (2.3)$$

The above transformation can easily be coded to arbitrary order with  $M$  as an input.

The inverse transformation from the polynomial hexahedron to the biunit cube is denoted by  $\mathbf{\Xi}(\mathbf{X})$ . As opposed to  $\mathbf{X}(\mathbf{\Xi})$  defined explicitly by (2.3),  $\mathbf{\Xi}(\mathbf{X})$  is generally not a polynomial function and is difficult to be expressed explicitly. Figure 2.4 depicts the transformation  $\mathbf{X}(\mathbf{\Xi})$  and its inverse  $\mathbf{\Xi}(\mathbf{X})$ .

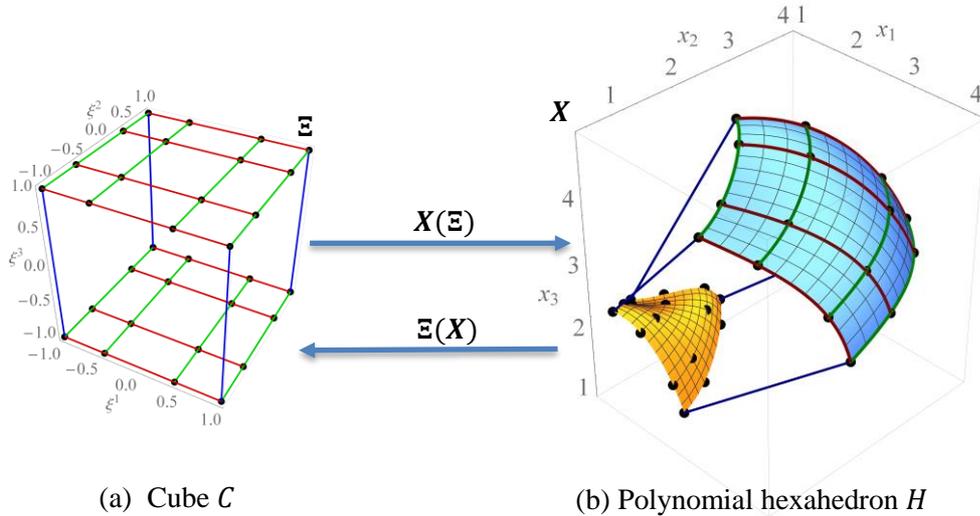


Figure 2.4: Transformation  $\mathbf{X}(\mathbf{\Xi})$  from reference to physical space and its inverse  $\mathbf{\Xi}(\mathbf{X})$ . Red, green, and blue curves or lines are coordinate curves with respectively varying  $\xi$ ,  $\eta$ , and  $\zeta$ .

The polynomial surfaces provide an accurate approximation to the smooth physical or analytical surfaces as can be seen in Figure 2.5; here, the purple and dark blue surfaces are those defined by the

polynomial approximations, and the yellow and light blue surfaces are analytical. Between the polynomial and analytical surfaces, only the one on top can be seen.

It is generally believed that Lobatto points provide higher accuracy than equidistant points for approximating curved surfaces. This belief could relate to the behavior of the maximum values of the Lagrange polynomials discussed in Fig. 2.3 or the fact that quadratures using Lobatto points are more accurate than those using equidistant points. The comparison of accuracy between these two sets of points for approximating curved geometries in CFD remains to be investigated.

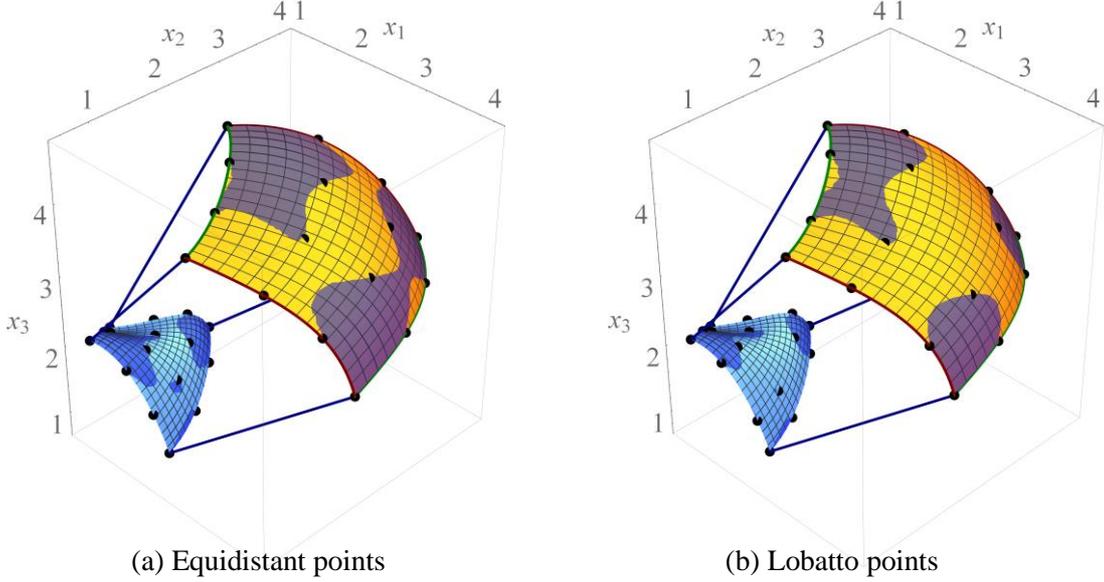


Figure 2.5: Physical (analytical) surfaces are in yellow and light blue, polynomial ones in purple and dark blue.

## 2.4 Coordinate Curves (or Curvilinear Coordinate Lines) and Surfaces

Below, we only deal with the polynomial hexahedron as defined in (2.3).

Let  $\Xi_0 = (\xi_0, \eta_0, \zeta_0)$  be a fixed point in the reference cube. In the physical space, set  $\mathbf{X}_0 = \mathbf{X}(\Xi_0)$ . A coordinate curve through  $\mathbf{X}_0$  is a curve along which only one of the reference coordinates varies.

Figure 2.6 shows (a)  $\Xi_0 = (\xi_0, \eta_0, \zeta_0) = (0.6, -0.8, 1)$  on the top surface of the reference cube represented by the red dot, and (b) the three coordinate curves in physical space through  $\mathbf{X}_0$ :

$$\mathbf{X}(\xi_0 + \tau, \eta_0, \zeta_0), \mathbf{X}(\xi_0, \eta_0 + \tau, \zeta_0), \text{ and } \mathbf{X}(\xi_0, \eta_0, \zeta_0 + \tau), \text{ where } \tau \text{ varies on } [-1, 1].$$

A surface through  $\mathbf{X}_0$  along which two of the coordinates vary and one remains constant is called a coordinate surface. It is referred to by the constant coordinate, e.g., the coordinate surface  $\xi = \xi_0$ .

## 2.5 Derivative of the Transformation and Covariant Base Vectors

With  $\mathbf{X}(\Xi)$  defined by (2.3), its derivative is a  $3 \times 3$  matrix called the Jacobian matrix:

$$\frac{d\mathbf{X}}{d\Xi} = \begin{pmatrix} \frac{\partial x_1}{\partial \xi^1} & \frac{\partial x_1}{\partial \xi^2} & \frac{\partial x_1}{\partial \xi^3} \\ \frac{\partial x_2}{\partial \xi^1} & \frac{\partial x_2}{\partial \xi^2} & \frac{\partial x_2}{\partial \xi^3} \\ \frac{\partial x_3}{\partial \xi^1} & \frac{\partial x_3}{\partial \xi^2} & \frac{\partial x_3}{\partial \xi^3} \end{pmatrix} = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} & \frac{\partial x}{\partial \zeta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} & \frac{\partial y}{\partial \zeta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} & \frac{\partial z}{\partial \zeta} \end{pmatrix}. \quad (2.4)$$

The Jacobian matrix  $\frac{d\mathbf{X}}{d\Xi}$  is sometimes called the Jacobian. However, here, the term Jacobian refers to the determinant of the Jacobian matrix, i.e.,  $\det\left(\frac{d\mathbf{X}}{d\Xi}\right)$ .

For  $1 \leq i \leq 3$ , the  $i$ -th column of  $\frac{d\mathbf{X}}{d\Xi}$  is denoted by  $\mathbf{a}_i$  and called a covariant base vector; that is,

$$\mathbf{a}_i = \frac{\partial \mathbf{X}}{\partial \xi^i}. \quad (2.5)$$

The above definition implies that  $\mathbf{a}_i$  is tangent to the  $\xi^i$ -th coordinate curve.

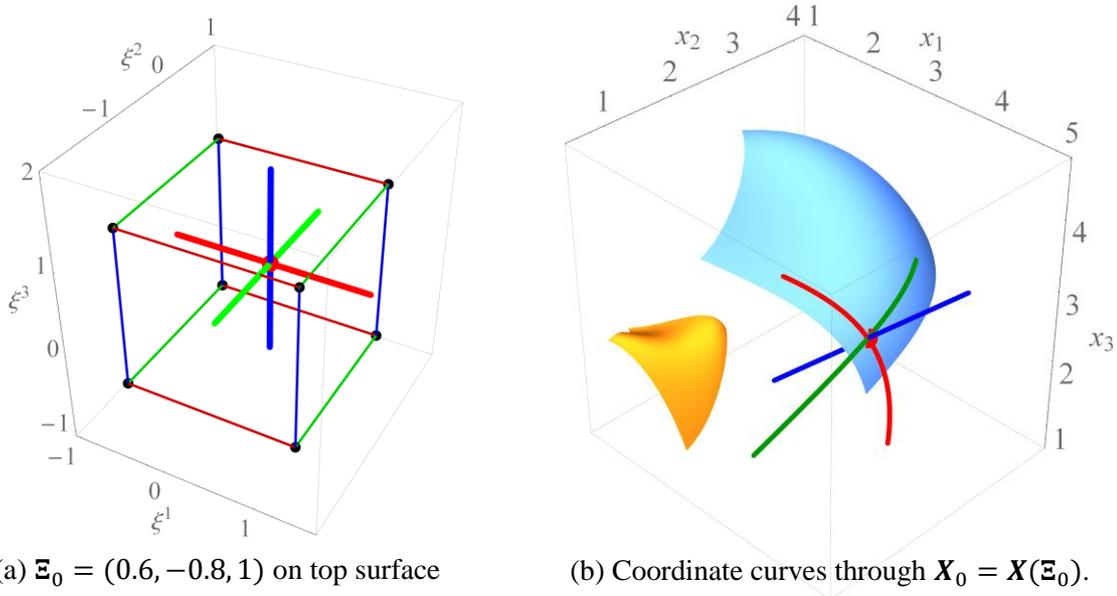


Figure 2.6: Coordinate curves through  $\mathbf{X}_0$ . Red, green, and blue curves correspond to lines with respectively varying  $\xi$ ,  $\eta$ , and  $\zeta$ . Also shown are the coordinate surfaces  $\zeta = -1$  (yellow) and  $\zeta = 1$  (blue).

Denote the basis for  $R^3$  by

$$\boldsymbol{\epsilon}^1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad \boldsymbol{\epsilon}^2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad \text{and} \quad \boldsymbol{\epsilon}^3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}. \quad (2.6)$$

The matrix  $\frac{d\mathbf{X}}{d\Xi}$  maps the vector space  $R^3$  in the reference frame to  $R^3$  in the physical frame. In particular, by (2.4) and the above definition,  $\frac{d\mathbf{X}}{d\Xi}$  maps  $\boldsymbol{\epsilon}^i$  of the reference space to the covariant base vector  $\mathbf{a}_i$  in the physical space:

$$\frac{d\mathbf{X}}{d\Xi} \boldsymbol{\epsilon}^i = \mathbf{a}_i. \quad (2.7)$$

Figure 2.7 shows the three covariant base vectors together with the coordinate curves at (a)  $\mathbf{X}_0$  corresponding to  $\Xi_0 = (0.6, -0.8, 1)$  and (b)  $\mathbf{X}_0$  corresponding to  $\Xi_0 = (-0.2, 0.4, 1)$ .

## 2.6 Jacobian $J$ .

The determinant of  $\frac{d\mathbf{X}}{d\Xi}$  is called the Jacobian and denoted by  $J$ . For simplicity of notation, set

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} = \frac{d\mathbf{X}}{d\Xi}. \quad (2.8)$$

Then the covariant base vectors are the columns  $\mathbf{a}_i = (a_{1i}, a_{2i}, a_{3i})^T$ ,  $1 \leq i \leq 3$ , where the superscript  $T$  denotes the transpose. By the definition of the determinant using the cofactor along the first column,

$$\det\left(\frac{d\mathbf{X}}{d\boldsymbol{\Xi}}\right) = a_{11}(a_{22}a_{33} - a_{23}a_{32}) + a_{21}(a_{32}a_{13} - a_{12}a_{33}) + a_{31}(a_{12}a_{23} - a_{22}a_{13}).$$

Recall the cross product formula:

$$\mathbf{a}_2 \times \mathbf{a}_3 = (a_{22}a_{33} - a_{23}a_{32}, a_{32}a_{13} - a_{12}a_{33}, a_{12}a_{23} - a_{22}a_{13}).$$

The above two equations imply

$$J = \mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3).$$

In fact, if  $(i, j, k)$  is cyclic, the same argument yields

$$J = \mathbf{a}_i \cdot (\mathbf{a}_j \times \mathbf{a}_k). \quad (2.9)$$

Thus,  $J$  is the (signed) volume of the parallelepiped defined by  $\mathbf{a}_i$ ,  $\mathbf{a}_j$ , and  $\mathbf{a}_k$ .

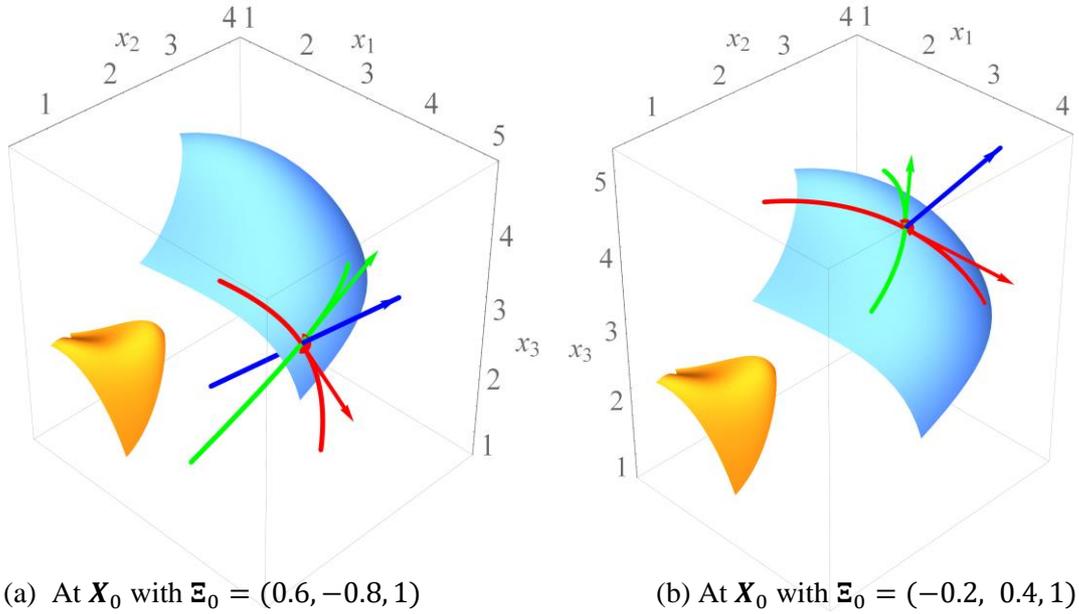


Figure 2.7: Covariant base vectors.

## 2.7 Contravariant Base Vectors

The inverse of  $\mathbf{X}(\boldsymbol{\Xi})$  is denoted by  $\boldsymbol{\Xi}(\mathbf{X})$ . It maps the polynomial hexahedron to the reference cube. Its derivative is

$$\frac{d\boldsymbol{\Xi}}{d\mathbf{X}} = \begin{pmatrix} \frac{\partial \xi^1}{\partial x_1} & \frac{\partial \xi^1}{\partial x_2} & \frac{\partial \xi^1}{\partial x_3} \\ \frac{\partial \xi^2}{\partial x_1} & \frac{\partial \xi^2}{\partial x_2} & \frac{\partial \xi^2}{\partial x_3} \\ \frac{\partial \xi^3}{\partial x_1} & \frac{\partial \xi^3}{\partial x_2} & \frac{\partial \xi^3}{\partial x_3} \end{pmatrix} = \begin{pmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \xi}{\partial y} & \frac{\partial \xi}{\partial z} \\ \frac{\partial \eta}{\partial x} & \frac{\partial \eta}{\partial y} & \frac{\partial \eta}{\partial z} \\ \frac{\partial \zeta}{\partial x} & \frac{\partial \zeta}{\partial y} & \frac{\partial \zeta}{\partial z} \end{pmatrix}. \quad (2.10)$$

How should we describe the ‘geometry’ of this matrix? The columns of  $\frac{d\mathbf{X}}{d\boldsymbol{\Xi}}$  are described in the previous subsection as vectors tangent to the coordinate curves in the physical frame; therefore, the

columns of  $\frac{d\Xi}{dX}$  can be described as vectors in the reference frame; however, such a description is not fruitful.

The relation between  $\frac{dX}{d\Xi}$  and  $\frac{d\Xi}{dX}$  is better demonstrated by considering, for the matrix  $\frac{d\Xi}{dX}$ , not its column but its row vectors in the *physical space* as discussed below.

Denote by  $I$  the  $3 \times 3$  identity matrix. Since  $\Xi(X)$  is the inverse of  $X(\Xi)$ ,  $\frac{d\Xi}{dX}$  is the inverse of  $\frac{dX}{d\Xi}$ :

$$\frac{d\Xi}{dX} \frac{dX}{d\Xi} = I. \quad (2.11)$$

The  $i$ -th row of  $\frac{d\Xi}{dX}$  is denoted by  $\mathbf{a}^i$  and called a contravariant base vector; that is, in the physical space, with  $\nabla$  denote the gradient operator  $(\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \frac{\partial}{\partial x_3})$  or  $(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})$ , by (2.10),

$$\mathbf{a}^i = \nabla \xi^i. \quad (2.12)$$

Since  $\mathbf{a}^i$  is the  $i$ -th row of  $\frac{d\Xi}{dX}$  and  $\mathbf{a}_j$  is the  $j$ -th column of  $\frac{dX}{d\Xi}$ , for  $1 \leq i, j \leq 3$ , Eq. (2.11) implies

$$\mathbf{a}^i \cdot \mathbf{a}_j = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}. \quad (2.13)$$

For cyclic  $(i, j, k)$ , (i.e., it equals the permutation  $(1, 2, 3)$ ,  $(2, 3, 1)$ , or  $(3, 1, 2)$ ), the above means  $\mathbf{a}^i \cdot \mathbf{a}_j = \mathbf{a}^i \cdot \mathbf{a}_k = 0$ . That is, the contravariant base vector  $\mathbf{a}^i$  is orthogonal to the two covariant base vectors  $\mathbf{a}_j$  and  $\mathbf{a}_k$ . As a result,  $\mathbf{a}^i$  is orthogonal to the coordinate surface defined by  $\xi^i = \text{constant}$ . This fact is also consistent with (2.12) above.

Since the vector  $\mathbf{a}_j \times \mathbf{a}_k$  is also orthogonal to  $\mathbf{a}_j$  and  $\mathbf{a}_k$ , we have

$$\mathbf{a}^i = c (\mathbf{a}_j \times \mathbf{a}_k) \quad (2.14)$$

for some constant  $c$ . Applying the dot product by  $\mathbf{a}_i$  on the right to both sides of the above equation,

$$\mathbf{a}^i \cdot \mathbf{a}_i = c (\mathbf{a}_j \times \mathbf{a}_k) \cdot \mathbf{a}_i = c [\mathbf{a}_i \cdot (\mathbf{a}_j \times \mathbf{a}_k)].$$

By (2.13),  $\mathbf{a}^i \cdot \mathbf{a}_i = 1$ . Therefore,

$$c [\mathbf{a}_i \cdot (\mathbf{a}_j \times \mathbf{a}_k)] = 1.$$

That is, by (2.9),

$$c = \frac{1}{J}. \quad (2.15)$$

Thus, (2.14) implies for cyclic  $(i, j, k)$ ,

$$\mathbf{a}^i = \frac{1}{J} (\mathbf{a}_j \times \mathbf{a}_k) = \frac{\mathbf{a}_j \times \mathbf{a}_k}{\mathbf{a}_i \cdot (\mathbf{a}_j \times \mathbf{a}_k)}, \quad (2.16)$$

or

$$J \mathbf{a}^i = \mathbf{a}_j \times \mathbf{a}_k. \quad (2.17)$$

Loosely put,  $\mathbf{a}^i$  is a sort of ‘inverse vector’ of  $\mathbf{a}_i$ , but since the inverse of a vector is not well-defined, the precise relation is given by (2.16).

As will be shown below, the contravariant vector  $J \mathbf{a}^i$  is employed more often than the contravariant base vector  $\mathbf{a}^i$ . In addition, in the case of a singularity such as collapsing one face of the hexahedron to a point,  $J$  becomes zero at the singularity and  $\mathbf{a}^i$  becomes undefined, whereas  $J \mathbf{a}^i$  is still well defined.

Since each components of the covariant base vectors  $\mathbf{a}_i = \frac{\partial X}{\partial \xi^i}$  is a polynomial, (2.17) implies that each component of  $J \mathbf{a}^i$  is also a polynomial. Each component of  $\mathbf{a}^i$ , however, is not a polynomial. As can be seen by (2.16), it is a rational function (ratio of two polynomials).

In linear algebra, the inverse of an arbitrary  $3 \times 3$  matrix is defined and calculated algebraically. The above discussion of covariant and contravariant base vectors provides the geometric relation between the columns of a  $3 \times 3$  matrix and the rows of its inverse.

By assuming that the reference frame is non-dimensional, the (components of the) covariant base vectors  $\mathbf{a}_i$  have dimension length, and the contravariant base vectors  $\mathbf{a}^i$  have dimension  $1/\text{length}$ . Since  $J$  has dimension volume, the contravariant vectors  $J\mathbf{a}^i$  have dimension area, a fact consistent with (2.17).

The contravariant vector  $J\mathbf{a}^i$  calculated via (2.17) is in cross product form. It can also be expressed in conservative or curl form; for more discussion on these forms, see (Thomas and Lombard 1979, Kopriva 2005).

Figure 2.8 shows the contravariant vector  $J\mathbf{a}^3$  (thick arrow) together with the coordinate curves and the covariant base vectors (thin arrows) at (a)  $\mathbf{X}_0$  corresponding to  $\mathbf{\Xi}_0 = (0.6, -0.8, 1)$  and (b)  $\mathbf{X}_0$  corresponding to  $\mathbf{\Xi}_0 = (-0.2, 0.4, 1)$ .

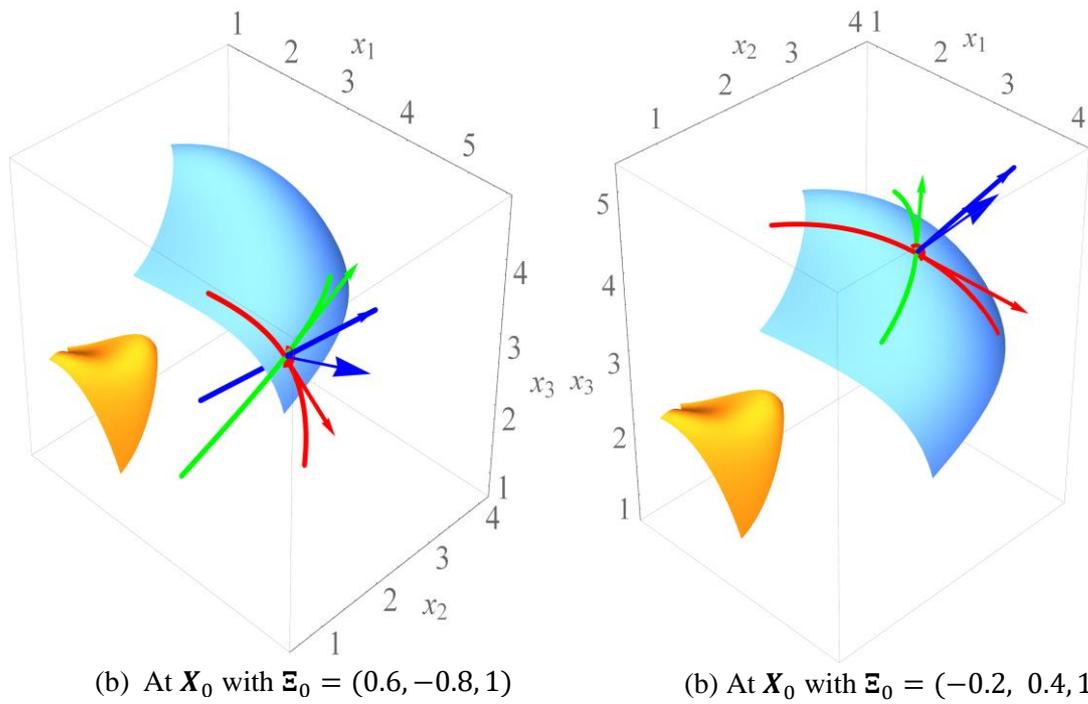


Figure 2.8: Contravariant vector  $J\mathbf{a}^3$  represented by thick arrows and the 3 covariant base vectors by thin arrows.

### 3 Divergence Theorem and Fundamental Metric Identity

#### 3.1 Divergence Theorem

The divergence theorem in physical space plays a key role in Computational Fluid Dynamics. The corresponding divergence operator in the reference frame involves the previously defined metrics as discussed below.

In physical space, let  $\mathbf{F} = (F_1, F_2, F_3)$  be a (smooth) vector field on a volume  $V$  bounded by a surface  $\partial V = S$  with outward unit normal  $\mathbf{n}$  (e.g.,  $\mathbf{F}$  is a velocity field). Following common practice, the vector

field  $\mathbf{F}(\mathbf{X}(\boldsymbol{\Xi}))$  with domain in the reference frame is abbreviated to  $\mathbf{F}$  when there is no ambiguity. This practice is employed in, e.g., the chain rule  $\frac{dy}{dx} = \frac{dy}{du} \frac{du}{dx}$ .

The divergence theorem in physical space takes the form

$$\int_V \nabla \cdot \mathbf{F} dV = \int_S \mathbf{F} \cdot \mathbf{n} ds. \quad (3.1)$$

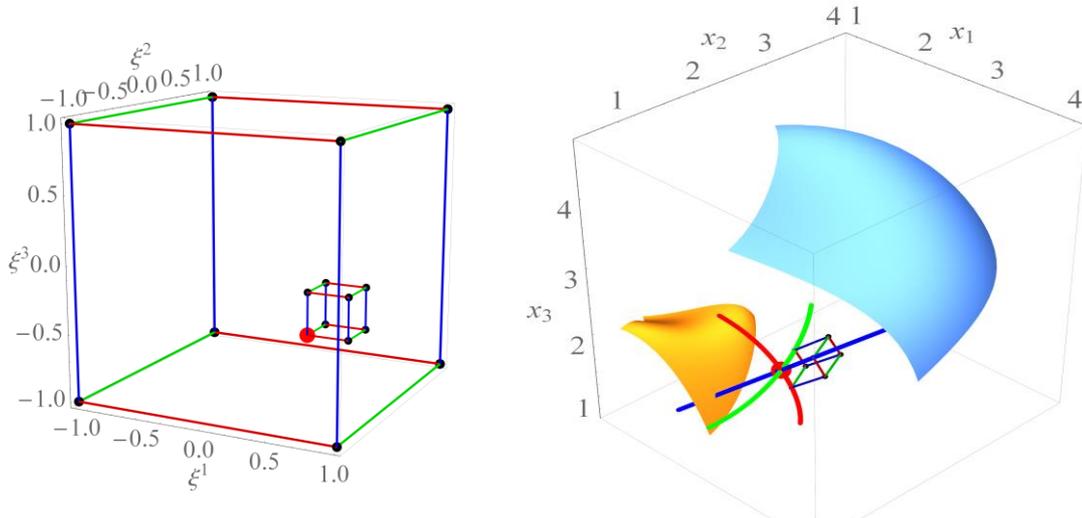
Next, we apply this theorem to a volume increment to derive formulas for the divergence in the reference frame.

At a fixed point  $\boldsymbol{\Xi}_0 = (\xi_0^1, \xi_0^2, \xi_0^3)$ , consider an increment,  $\Delta\boldsymbol{\Xi} = (\Delta\xi^1, \Delta\xi^2, \Delta\xi^3)$ . With  $\boldsymbol{\epsilon}^i$  defined in (2.6), denote

$$\Delta\boldsymbol{\Xi}^i = \Delta\xi^i \boldsymbol{\epsilon}^i, \quad (3.2)$$

e.g.,  $\Delta\boldsymbol{\Xi}^2 = (0, \Delta\xi^2, 0)$ .

Figure 3.1 shows an example of a volume increment in the reference and physical domains.



(a) Small cube at  $\boldsymbol{\Xi}_0$  in the reference frame      (b) Parallelepiped  $P$  at  $\mathbf{X}_0$  in the physical frame

Figure 3.1: Volume increments in the reference and physical domains

Along the  $\xi^i$ -coordinate curve, since  $\mathbf{a}_i = \frac{\partial \mathbf{X}}{\partial \xi^i}$ ,

$$\frac{\mathbf{X}(\boldsymbol{\Xi}_0 + \Delta\boldsymbol{\Xi}^i) - \mathbf{X}(\boldsymbol{\Xi}_0)}{\Delta\xi^i} \approx \mathbf{a}_i(\boldsymbol{\Xi}_0). \quad (3.3)$$

Thus,

$$\mathbf{X}(\boldsymbol{\Xi}_0 + \Delta\boldsymbol{\Xi}^i) \approx \mathbf{X}(\boldsymbol{\Xi}_0) + \Delta\xi^i \mathbf{a}_i(\boldsymbol{\Xi}_0). \quad (3.4)$$

Along the diagonal,  $\mathbf{X}(\boldsymbol{\Xi}_0 + \Delta\boldsymbol{\Xi}^i + \Delta\boldsymbol{\Xi}^j)$  can be approximated by

$$\begin{aligned} \mathbf{X}(\boldsymbol{\Xi}_0 + \Delta\boldsymbol{\Xi}^i + \Delta\boldsymbol{\Xi}^j) &\approx \mathbf{X}(\boldsymbol{\Xi}_0 + \Delta\boldsymbol{\Xi}^i) + \Delta\xi^j \mathbf{a}_j(\boldsymbol{\Xi}_0 + \Delta\boldsymbol{\Xi}^i) \\ &\approx \mathbf{X}(\boldsymbol{\Xi}_0) + \Delta\xi^i \mathbf{a}_i(\boldsymbol{\Xi}_0) + \Delta\xi^j \mathbf{a}_j(\boldsymbol{\Xi}_0) \end{aligned} \quad (3.5)$$

where, for the second approximation, the higher order terms  $O(\Delta\xi^i \Delta\xi^j)$  are omitted.

As a result of (3.4) and (3.5), for cyclic  $(i, j, k)$ , the square defined by the four corners  $\Xi_0$ ,  $\Xi_0 + \Delta\Xi^i$ ,  $\Xi_0 + \Delta\Xi^j$ , and  $\Xi_0 + \Delta\Xi^i + \Delta\Xi^j$  in the reference frame is mapped to a quadrilateral in the physical space with corners  $\mathbf{X}(\Xi_0)$ ,  $\mathbf{X}(\Xi_0 + \Delta\Xi^i)$ ,  $\mathbf{X}(\Xi_0 + \Delta\Xi^j)$ , and  $\mathbf{X}(\Xi_0 + \Delta\Xi^i + \Delta\Xi^j)$ . This quadrilateral can be approximated by the parallelogram defined by the corner  $\mathbf{X}(\Xi_0)$  and the two vectors  $\Delta\xi^i \mathbf{a}_i(\Xi_0)$  and  $\Delta\xi^j \mathbf{a}_j(\Xi_0)$  representing the two edges.

In a similar manner, the small cube defined by the point  $\Xi_0$  with the increments  $\Delta\Xi^i$  along the three directions is mapped to a hexahedron, which can be approximated by the parallelepiped  $P$  defined by the corner  $\mathbf{X}_0 = \mathbf{X}(\Xi_0)$  and the three vectors  $\Delta\xi^i \mathbf{a}_j(\Xi_0)$  or  $\Delta\xi^i \mathbf{a}_i$ ,  $1 \leq i \leq 3$ . Using (3.4) and 2.9), the volume of the parallelepiped  $P$  is

$$(\Delta\xi^1 \Delta\xi^2 \Delta\xi^3) [\mathbf{a}_i \cdot (\mathbf{a}_j \times \mathbf{a}_k)](\Xi_0) = J(\Xi_0) \Delta\xi^1 \Delta\xi^2 \Delta\xi^3. \quad (3.6)$$

In the physical frame, denote the two faces of  $P$  with constant  $\xi^i$  by  $\partial P_+^i$  and  $\partial P_-^i$  where the subscripts + and - represent respectively the larger and smaller values of  $\xi^i$ .

For cyclic  $(i, j, k)$ , the cross product  $(\Delta\xi^j \Delta\xi^k) \mathbf{a}_j \times \mathbf{a}_k$  provides the normal direction as well as the area for the corresponding parallelogram:

$$(\Delta\xi^j \Delta\xi^k) (\mathbf{a}_j \times \mathbf{a}_k)(\Xi_0) = \mathbf{n} \text{Area}(\partial P_-^i) \quad (3.7)$$

where  $\mathbf{n}$  is in the direction of increasing  $\xi^i$  for the rest of this subsection. Similarly,

$$(\Delta\xi^j \Delta\xi^k) (\mathbf{a}_j \times \mathbf{a}_k)(\Xi_0 + \Delta\Xi^i) = \mathbf{n} \text{Area}(\partial P_+^i). \quad (3.8)$$

Instead of the above,  $\mathbf{n} \text{Area}(\partial P_+^i)$  can be estimated by the left hand side of (3.7) as well. The result is the nonconservation form of the divergence formula as discussed below.

Applying the divergence theorem to the parallelepiped  $P$

$$\int_P \nabla \cdot \mathbf{F} dV = \int_{\partial P} \mathbf{F} \cdot \mathbf{n} ds. \quad (3.9)$$

For the above left hand side, since the volume of  $P$  is estimated by (3.6),

$$\int_P \nabla \cdot \mathbf{F} dV \approx J(\Xi_0) (\Delta\xi^i \Delta\xi^j \Delta\xi^k) (\nabla \cdot \mathbf{F})(\Xi_0). \quad (3.10)$$

Concerning the right hand side of (3.9), for the surfaces  $\partial P_+^i$  and  $\partial P_-^i$ , (3.7) and (3.8) can be employed. With cyclic  $(i, j, k)$ , the right hand side can be estimated by

$$\sum_{i=1}^3 \Delta\xi^j \Delta\xi^k \{ [\mathbf{F} \cdot (\mathbf{a}_j \times \mathbf{a}_k)](\Xi_0 + \Delta\Xi^i) - [\mathbf{F} \cdot (\mathbf{a}_j \times \mathbf{a}_k)](\Xi_0) \}. \quad (3.11)$$

The above can be written as

$$\sum_{i=1}^3 (\Delta\xi^i \Delta\xi^j \Delta\xi^k) \frac{[\mathbf{F} \cdot (\mathbf{a}_j \times \mathbf{a}_k)](\Xi_0 + \Delta\Xi^i) - [\mathbf{F} \cdot (\mathbf{a}_j \times \mathbf{a}_k)](\Xi_0)}{\Delta\xi^i}.$$

That is, since  $\Delta\xi^i \Delta\xi^j \Delta\xi^k = \Delta\xi^1 \Delta\xi^2 \Delta\xi^3$ ,

$$(\Delta\xi^i \Delta\xi^j \Delta\xi^k) \sum_{i=1}^3 \frac{[\mathbf{F} \cdot (\mathbf{a}_j \times \mathbf{a}_k)](\Xi_0 + \Delta\Xi^i) - [\mathbf{F} \cdot (\mathbf{a}_j \times \mathbf{a}_k)](\Xi_0)}{\Delta\xi^i}. \quad (3.12)$$

Using (3.10) and (3.12), as the small cube size shrinks to zero, (3.9) implies

$$J(\Xi_0) (\nabla \cdot \mathbf{F})(\Xi_0) = \sum_{i=1}^3 [\mathbf{F} \cdot (\mathbf{a}_j \times \mathbf{a}_k)]_{\xi^i}(\Xi_0) \quad (3.13)$$

Since  $\Xi_0$  is arbitrary on the reference cube,

$$\nabla \cdot \mathbf{F} = \frac{1}{J} \sum_{i=1}^3 [(\mathbf{a}_j \times \mathbf{a}_k) \cdot \mathbf{F}]_{\xi^i}, \quad (3.14)$$

Recall (2.17):  $\mathbf{a}_j \times \mathbf{a}_k = J\mathbf{a}^i$ . The above results in the ‘conservation form’ for the divergence in the reference frame:

$$J \nabla \cdot \mathbf{F} = \sum_{i=1}^3 (J\mathbf{a}^i \cdot \mathbf{F})_{\xi^i}. \quad (3.15)$$

If we use the left hand side of (3.7), namely  $(\Delta\xi^j \Delta\xi^k)(\mathbf{a}_j \times \mathbf{a}_k)(\Xi_0)$  to estimate  $\mathbf{n} \text{Area}(\partial P_+^i)$ , the result for the right hand side of (3.9) is the following instead of (3.11):

$$\sum_{i=1}^3 \Delta\xi^j \Delta\xi^k (\mathbf{a}_j \times \mathbf{a}_k)(\Xi_0) [\mathbf{F}(\Xi_0 + \Delta\Xi^i) - \mathbf{F}(\Xi_0)].$$

Therefore, in place of (3.15), we obtain the following ‘nonconservation form’ of the divergence:

$$\nabla \cdot \mathbf{F} = \sum_{i=1}^3 \mathbf{a}^i \cdot (\mathbf{F}_{\xi^i}) \quad (3.16)$$

The two forms (3.15) and (3.16) yield the same result analytically; however, they generally yield different results numerically. The conservation form (3.15) is often preferred especially in the presence of shocks.

### 3.2 Fundamental Metric Identity

By expanding (3.15),

$$\nabla \cdot \mathbf{F} = \frac{1}{J} \sum_{i=1}^3 (J\mathbf{a}^i)_{\xi^i} \cdot \mathbf{F} + \frac{1}{J} \sum_{i=1}^3 (J\mathbf{a}^i) \cdot (\mathbf{F}_{\xi^i}).$$

The above and (3.16) imply, for any vector field  $\mathbf{F}$ ,

$$\frac{1}{J} \sum_{i=1}^3 (J\mathbf{a}^i)_{\xi^i} \cdot \mathbf{F} = 0. \quad (3.17)$$

Since  $\mathbf{F}$  is arbitrary,

$$\sum_{i=1}^3 (J\mathbf{a}^i)_{\xi^i} = \mathbf{0}, \quad (3.18)$$

or

$$(J\mathbf{a}^1)_{\xi} + (J\mathbf{a}^2)_{\eta} + (J\mathbf{a}^3)_{\zeta} = \mathbf{0}. \quad (3.19)$$

Note that the above two equations are vector equations whereas (3.17) is a scalar one.

As discussed on p. 104 of (Thompson et al. 1985), (3.19) above is “a fundamental metric identity”. It is sometimes referred to as “metric identity” (e.g., Kopriva 2005) or “geometric conservation law” (Thomas and Lombard 1979). It is also called the ‘surface closure law’ representing the closed surface of the infinitesimal hexahedron approximated by the parallelepiped  $P$ .

The above proof shows that the metric identity holds analytically for any smooth transformation, polynomial ones in particular. Numerically, however, it may not. The reason is that formulas for numerical derivative are based on a certain number of points, which may not be enough to estimate the derivative of  $J\mathbf{a}^i = \mathbf{a}_j \times \mathbf{a}_k$  exactly. For more discussions, see, e.g., (Kopriva 2005, Abe et al. 2015).

The fact that (3.17) implies each component of (3.18) equals zero can also be shown by setting  $\mathbf{F} = \boldsymbol{\epsilon}^i$  in (3.17). Equivalently, if  $\mathbf{F} = (c_1, c_2, c_3)$  is a constant velocity field in physical space, then  $\nabla \cdot \mathbf{F}$  equals zero. For the calculation of  $\nabla \cdot \mathbf{F}$  in the reference frame using (3.15) to result in zero, (3.18) or (3.19) must hold. For this reason, in discretized form, if (3.19) holds, the discretization is said to have the property of “free stream preservation” in the sense that a constant velocity field remains constant with time.

## 4 Conclusions and Discussion

The derivation of coordinate transformations and various metric quantities for the case of arbitrary high-order meshes was carried out using the capabilities of symbolic manipulation, three-dimensional plotting, and object rotating of Mathematica. The geometric description, relations among the metric quantities, and the metric identity are derived in a manner considerably simpler than the standard derivation. The resulting Mathematica program facilitates the visualization of these quantities and may enhance understanding and/or improve intuition on coordinate transformation. Concerning additional research needed, the discretized metric identity for the case of high-order tetrahedral as well as hexahedral meshes, the high-order mesh generation, and the post processing of large data remain to be studied and developed.

## References

- [1] Y. Abe, T. Haga, T. Nonomura, and K. Fujii, On the freestream preservation of high-order conservative flux-reconstruction schemes, *J. Comput. Phys.* 281 (2015) 28-54.
- [2] D. Kopriva, Metric identities and the discontinuous spectral element method on curvilinear meshes, *J. Sci. Comput.* 26 (2006) 301-326.
- [3] P.D. Thomas and C.K. Lombard, Geometric conservation law and its application to flow computations on moving grids. *AIAA J.* 17(10) (1979) 1030-1037.
- [4] J.F. Thompson, Z.U.A. Warsi, and C.W. Mastin, Boundary-fitted coordinate systems for numerical solution of partial differential equations, a review. *J. Comput. Phys.* 47(1) (1982) 1-108.
- [5] J.F. Thompson, Z.U.A. Warsi, and C.W. Mastin, *Numerical Grid Generation: Foundations and Applications*, Elsevier Science Ltd, 1985.

## Appendix

(\*Coordinate Transformations and High-Order Meshes with Mathematica.

By H.T.Huynh, huynh@grc.nasa.gov, July 2018, NASA Glenn Research Center, Cleveland, Ohio, USA. This program can be copied in pdf form and paste into a Mathematica notebook. It can be broken into cells along the blank lines for ease of relating outputs with commands. \*)

(\* 1. Inputs

M: number of gridpoints in 1D including the two boundaries;

optLo  $\neq$  0 for Lobatto points, = 0 for equi-distant points;

optCube = 2 bi-unit cube, = 1 unit cube for the master or reference domain;

optPhys = 1 spherical surfaces in the physical domain, = 2 perturbed spherical surfaces, = 3 another pair of curved surfaces, = 4 collapsing the square at  $y = 1$  to its midpoint. HO: High-Order \*)

M = 4; optLo = 1; optCube = 2; optPhys = 2; If[optPhys==4, M = 2];

(\* 2. Reference domain; first, define Lobatto polynomial function on [-1,1] \*)

Lo[k\_,x\_]:=LegendreP[k,x]-LegendreP[k-2,x];

(\* This function gets the zeros of a polynomial and output as a table \*)

getZeros[f\_,x\_]:=Module[{deg,solu,tz}, deg=Exponent[f,x]; solu=Simplify[NSolve[f,x]];

tz=Table[Chop[x/.solu[[k]]],{k,1,deg}];

(\* Get  $\tau[k]$  on [0,1] for locations of 1D HO curve; also use to define physical surfaces \*)

If[optLo==0, (\*equi-distant else Lobatto points\*)

ts=Table[ $\tau[k]=N[(k-1)/(M-1)]$ ,{k,1,M}],

pts1D=getZeros[Lo[M,x],x];

ts1=(pts1D+1.)/2; ts=Table[ $\tau[k]=ts1[[k]]$ ,{k,1,M}]

];

```

(* on [0,1] or [-1,1] depending on optCube = 1 or 2*) IgnorePdfPageNumber;
If[optCube==1,txi=Table[xi[k]=\[Tau][k],{k,1,M}],txi=Table[xi[k]=-1+2*\[Tau][k],{k,1,M}]];
teta=Table[eta[k]=xi[k],{k,1,M}]; tzeta={zeta[1]=xi[1],zeta[2]=xi[M]};
(* Plot lines on reference cube that yield HO grid; tp_List: table of grid points; np1D: number of
points for HO curve in 1D; color1: color for \[Xi]^1-lines, default black; color2: \[Xi]^2-lines; color3,
\[Xi]^3-lines; thick: thickness of lines *)
plotLinesHOCube[tp_List,np1D_:Integer,{color1_:Black,color2_:Black,color3_:Black,
thick_:0.006}]:=
Module[{t1,t2,t3}, t1=Flatten[Table[Line[{tp[[1,j,k]],tp[[np1D,j,k]]},{j,1,np1D},{k,1,2}]];
t2=Flatten[Table[Line[{tp[[i,1,k]],tp[[i,np1D,k]]},{i,1,np1D},{k,1,2}]];
t3=Flatten[Table[Line[{tp[[i,j,1]],tp[[i,j,2]]},{i,1,np1D,np1D-1},{j,1,np1D,np1D-1}]];
Graphics3D[{color1,Thickness[thick],t1,color2,t2,color3,t3}];
(* Reference Domain; G for grid; high-order for bottom (k=1) and top (k=2) surfaces *)
t[CapitalXi]G=Table[[CapitalXi]G[i,j,k]={xi[i],eta[j],zeta[k]},{i,1,M},{j,1,M},{k,1,2}];
plotPtsRef=Graphics3D[{PointSize[0.025],Point[Flatten[t[CapitalXi]G,2]}];
plotLinesRef=plotLinesHOCube[t[CapitalXi]G,M,{Darker[Red,.2],Darker[Green,.2],Darker[Blue,.2]}];
];
(*so that we don't have to type this line many times when plot*)
pltOption1=Sequence[BaseStyle->{FontSize->20,FontFamily->"Times"},PlotRange->All,
Axes->True];
plotCubeRef=Show[plotPtsRef,plotLinesRef,Evaluate@pltOption1,
AxesLabel->{"\[Xi]^1",\[Xi]^2,\[Xi]^3}
(* Export for copying into paper in png*)
Export["plotCubeRef.png",plotCubeRef,ImageResolution->500];
(* table of points for linear mesh on reference domain *)
t[CapitalXi]G2=Table[[CapitalXi]G2[i,j,k]={zeta[i],zeta[j],zeta[k]},{i,1,2},{j,1,2},{k,1,2}];
plotPtsRef2=Graphics3D[{PointSize[0.025],Point[Flatten[t[CapitalXi]G2,2]}];
plotLinesRef2=plotLinesHOCube[t[CapitalXi]G2,2,
{Darker[Red,.2],Darker[Green,.2],Darker[Blue,.2]}];
plotCubeRef2=Show[plotPtsRef2,plotLinesRef2,Evaluate@pltOption1,
AxesLabel->{"\[Xi]^1",\[Xi]^2,\[Xi]^3}

(* 3a. Physical Domain. Define XT={xT,yT,zT} as functions of r \[Theta] and \[Phi] *)
Clear[r,rb,re,\[Theta],\[Theta]b,\[Theta]e,\[Phi],\[Phi]b,\[Phi]e];
Which[
optPhys==1, (*spherical surfaces*) rb=2.; re=3.5;
\[Theta]b=\[Pi]/6.; \[Theta]e=\[Theta]b+\[Pi]/4; \[Phi]b=0; \[Phi]e=\[Phi]b+\[Pi]/4;
XT={xT=r*Sin[\[Theta]]*Cos[\[Phi]],yT=r*Sin[\[Theta]]*Sin[\[Phi]],zT=r*Cos[\[Theta]}}];
XTb=(XT/.r->rb); XTe=(XT/.r->re),
optPhys==2, (* perturbed spherical surfaces *) rb=2.; re=4.;
\[Theta]b=\[Pi]/6.; \[Theta]e=\[Theta]b+\[Pi]/4; \[Phi]b=0; \[Phi]e=\[Phi]b+\[Pi]/4;
xT=r*Sin[\[Theta]]*Cos[\[Phi]]+.5(\[Theta]-\[Theta]b)^6 (\[Phi]-\[Phi]b)^7;
yT=r*Sin[\[Theta]]*Sin[\[Phi]]+xT/2;
zT=r*Cos[\[Theta]]+yT/2+.3*Sin[r]Sin[\[Pi] (\[Phi]-.5(\[Phi]b+\[Phi]e))/(\[Phi]e-\[Phi]b)]*
Sin[\[Pi] (\[Theta]-.5(\[Theta]b+\[Theta]e))/(\[Theta]e-\[Theta]b)];
xT=xT+.1*zT; XT={xT,yT,zT};
XTb=(XT/.r->rb); XTe=(XT/.r->re),
optPhys==3, (* another set of smooth surfaces *)
rb=0; re=.7; \[Theta]b=-.5; \[Theta]e=.5;
\[Phi]b=-.5; \[Phi]e=.5;
XTb={.5+.5\[Theta]-.1\[Phi],.6\[Phi]-.1\[Theta],rb-.1\[Theta]^7+.2\[Phi]^5-.2Sin[\[Theta]*\[Phi]}}];
XTe={.8+.6\[Theta]+.1\[Phi]+.1\[Phi]^6 \[Theta]^5,.2\[Theta]+.5\[Phi]-.1Sin[\[Theta]]Cos[\[Phi]],
re+.4(\[Theta]-.5)^4-.4\[Phi]^2-.4*Sin[\[Theta]*\[Phi]}}];
XT=(1-r)*XTb+r*XTe];

```

```
(* 3b Physical Domain.
k=1 for b or inner, results in k+(k-1)(M-2) = 1, and \[Tau][k] = \[Tau][1] = 0; k=2 for e or outer,
results in k+(k-1)(M-2) = M, and \[Tau][k] = \[Tau][M] = 1 *) IgnorePdfPageNumber;
If[ (* for curved surfaces*) optPhys<=3,
tXG = Table[XG[i,j,k]=
Chop[XT/.{r->(rb+(re-rb)\[Tau][k+(k-1)(M-2))],
\[Theta]->\[Theta]b+\[Tau][i]*(\[Theta]e-\[Theta]b),\[Phi]->\[Phi]b+\[Tau][j]*(\[Phi]e-\[Phi]b)},
{i,1,M},{j,1,M},{k,1,2}];
plotPtsPhys=Graphics3D[{PointSize[0.025],Point[Flatten[tXG,2]]}];
plotCv\[Xi]=ParametricPlot3D[{(XT/.{r->re,\[Phi]->\[Phi]b}),
(XT/.{r->re,\[Phi]->\[Phi]e})},
{\[Theta],[Theta]b,[Theta]e},PlotStyle->{Darker[Red,.2]}];
plotCv\[Eta]=ParametricPlot3D[{(XT/.{r->re,\[Theta]->\[Theta]b}),
(XT/.{r->re,\[Theta]->\[Theta]e})},
{\[Phi],[Phi]b,[Phi]e},PlotStyle->{Darker[Green,.2]}];
t1=Flatten[Table[Line[{XG[i,j,1],XG[i,j,2]}],{i,1,M,M-1},{j,1,M,M-1}]];
plot4VLines=Graphics3D[{Darker[Blue,.2],Thickness[0.006],t1}]; Clear[t1];
plotSurfsPhys=ParametricPlot3D[{(XT/.{r->re}),
(XT/.{r->rb})},{\[Theta],[Theta]b,[Theta]e},{\[Phi],[Phi]b,[Phi]e},Evaluate@pltOption1,
AxesLabel->{Subscript[x, 1],Subscript[x, 2],Subscript[x, 3]},ViewPoint->{3,-3,3}];
plotData=Show[plotSurfsPhys,plot4VLines,plotCv\[Xi],plotCv\[Eta],plotPtsPhys,
PlotLabel->"Physical Domain"]]
```

```
(* 3c. Collapsed cube *)
If[optPhys==4,
If[optCube==1, Do[If[i==1,XG[i,j,k]={i-1,j-1,k-1},XG[i,j,k]={i-1,0.5,0.5} ],
{i,1,2},{j,1,2},{k,1,2}],
(* otherwise optCube = 2 *)
Do[If[i==1,XG[i,j,k]={2i-3,2j-3,2k-3},
XG[i,j,k]={2i-3,0.,0.} ],
{i,1,2},{j,1,2},{k,1,2}]];
tXG=Table[XG[i,j,k],{i,1,2},{j,1,2},{k,1,2}];
plotPtsPhys=Graphics3D[{PointSize[0.025],Point[Flatten[tXG,2]]}];
plotLines=plotLinesHOCube[tXG,M,{Darker[Red,.2],Darker[Green,.2],Darker[Blue,.2],0.01}];
plotData=Show[plotPtsPhys,plotLines,Evaluate@pltOption1,
AxesLabel->{Subscript[x, 1],Subscript[x, 2],Subscript[x, 3]},PlotLabel->"Physical Domain"]]
```

```
(* Verify that spherical coordinate is orthogonal *)
If[optPhys==1,u1=D[XT,\[Theta]]; u2=D[XT,\[Phi]]; u3=D[XT,r];
Simplify[{u1.u2, u2.u3,u3.u1}]]
```

```
(* 4a. Transformation from reference domain to physical domain*)
tLa\[Xi]=Table[La\[Xi][n]=(!(\[Product]m=1),\[Product]n=1))
\[FractionBox[(\[Xi]-xi[m]),(xi[n]-xi[m])])!(\[Product]m=n+1),\[Product]n=1,M];
La\[Zeta][1]=(\[Zeta]-zeta[2])/(zeta[1]-zeta[2]);
La\[Zeta][2]=(\[Zeta]-zeta[1])/(zeta[2]-zeta[1]); t1=Flatten[Table[{xi[n],0},{xi[n],1}},{n,1,M}],1];
plotLagrange=Show[Plot[tLa\[Xi],{\[Xi],xi[1],xi[M]},Evaluate@pltOption1,
GridLines->Automatic,AxesLabel->{\[Xi],u}],
ListPlot[t1,PlotStyle->PointSize[0.025]],PlotLabel->"Lagrange Polys for Grid"]]
```

```
(* 4b. *) IgnorePdfPageNumber;
```

```
(* 4b. Basis functions for grid in 3D*) IgnorePdfPageNumber;
Table[bfG[i,j,k]=(La[Xi][i])(La[Xi][j]/.[Xi]->[Eta])(La[Zeta][k]),
{k,1,2},{j,1,M},{i,1,M}];
(*Mapping X from reference cube to physical domain *)
X=Chop[Expand[!(
\*UnderoverscriptBox[\(\[Sum]\), \(\i = 1\), \(\M)\](
\*UnderoverscriptBox[\(\[Sum]\), \(\j = 1\), \(\M)\](
\*UnderoverscriptBox[\(\[Sum]\), \(\k = 1\), \(\2)\]bfG[i, j, k]*XG[i, j, k]\)\)\)\)];

(* 4c. Plot polynomial surfaces and curves*)
Xb=Simplify[X/. [Zeta]->zeta[1]]; Xe=Simplify[X/. [Zeta]->zeta[2]];
plotbe=ParametricPlot3D[{Xb,Xe},{[Xi],xi[1],xi[M]},{[Eta],eta[1],eta[M]},
AxesLabel->{Subscript[x, 1],Subscript[x, 2],Subscript[x, 3]},Mesh->None,
Evaluate@pltOption1,ViewPoint->{3,-3,3}];
(* Grid curves *)
tXe1=Table[Xe/. [Eta]->eta[j],{j,1,M}];
plCvPoly[Xi]=ParametricPlot3D[tXe1,{[Xi],xi[1],xi[M]},
PlotStyle->{Darker[Red,.2],Thickness[0.008]}];
tXe2=Table[Xe/. [Xi]->xi[j],{j,1,M}];
plCvPoly[Eta]=ParametricPlot3D[tXe2,{[Eta],eta[1],eta[M]},
PlotStyle->{Darker[Green,.2],Thickness[0.008]}];
t1=Flatten[Table[Chop[Simplify[X/. {[Xi]->xi[i],[Eta]->eta[j]}]],{i,1,M,M-1},{j,1,M},1];
t2=Flatten[Table[Chop[Simplify[X/. {[Xi]->xi[i],[Eta]->eta[j]}]],{i,1,M},{j,1,M,M-1},1];
t3=Union[t1,t2];
plCvPoly[Zeta]=ParametricPlot3D[t3,{[Zeta],zeta[1],zeta[2]},
PlotStyle->{Darker[Blue,.5],Thickness[0.006]}];
t1=Flatten[Table[Chop[Simplify[X/. {[Xi]->xi[i],[Eta]->eta[j]}]],
{i,1,M,M-1},{j,1,M,M-1},1]; t2=Flatten[Table[Chop[Simplify[X/. {[Xi]->xi[i],[Eta]->eta[j]}]],
{i,1,M,M-1},{j,1,M,M-1},1];
t3=Union[t1,t2];
plCvPoly[Zeta]4=ParametricPlot3D[t3,{[Zeta],zeta[1],zeta[2]},
PlotStyle->{Darker[Blue,.5],Thickness[0.006]}];
plotPoly=Show[plotbe,plCvPoly[Xi],plCvPoly[Eta],plCvPoly[Zeta]4,
PlotLabel->"Polynomial Hexahedra"]

(* 4d. Plot polynomial surfaces and curves*)
plotbe1=ParametricPlot3D[{Xb,Xe},{[Xi],xi[1],xi[M]},{[Eta],eta[1],eta[M]},
PlotStyle->Directive[Blue,Opacity[0.5]],Lighting->Automatic,
Evaluate@pltOption1,AxesLabel->{Subscript[x, 1],Subscript[x, 2],Subscript[x, 3]}];
If[optPhys==4,plotSurfsPhys=plotbe1];
plotPolyPhys=Show[plotSurfsPhys,plotbe1,plotPtsPhys,
PlotLabel->"Phys. & Poly. Surfaces"]

(* 5a. Jacobian Matrix,Covariant Vectors, and J*Contravariant Vectors*)
\[\CapitalXi]={[Xi],[Eta],[Zeta]};
JM=dXd[\CapitalXi]=Table[dXd[\CapitalXi]e[i,j]=Chop[Simplify[D[X[[i]],[\CapitalXi][[j]]
]],{i,1,3},{j,1,3}];
(* we can also use the command Outer *) Clear[t1];
t1=Outer[D,X,\[\CapitalXi]]; Chop[Simplify[t1-dXd[\CapitalXi]]];

(* 5b. Covariant base vectors Subscript[a, i] as functions of ([Xi], [Eta], [Zeta]) *)
Clear[a]; dXd[\CapitalXi]T=Transpose[dXd[\CapitalXi]];
(* The i-th column of dXd[\CapitalXi], i.e., i-th row of dXd[\CapitalXi]T, is Subscript[a, i] *)
Table[Subscript[a, i]=Chop[Expand[dXd[\CapitalXi]T[[i]]],{i,1,3}];
```

(\*J\*Contravariant base vectors  $\text{Ja}^i = \text{Subscript}[\text{JaSup}, i]$  as functions of  $(\text{Xi}, \text{Eta}, \text{Zeta})$ \*)

IgnorePdfPageNumber;

crossPro[{a\_,b\_,c\_},{x\_,y\_,z\_}] := {b z-c y,c x-a z,a y-b x};

Subscript[JaSup, 1] = Chop[Expand[crossPro[Subscript[a, 2], Subscript[a, 3]]];

Subscript[JaSup, 2] = Chop[Expand[crossPro[Subscript[a, 3], Subscript[a, 1]]];

Subscript[JaSup, 3] = Chop[Expand[crossPro[Subscript[a, 1], Subscript[a, 2]]];

(\* 6a. Fixed point  $\text{CapitalXi}N$  in reference domain, N for numerical, red point \*)

(\*  $\text{CapitalXi}s$  is set relative to  $[0,1]$ \*)

case=1; (\* -2  $\leq$  case  $\leq$  3 \*)

Which[case==2,  $\text{CapitalXi}s = \{1,0,1\}$ , case==-

1,  $\text{CapitalXi}s = \{0,0,0\}$ , case==0,  $\text{CapitalXi}s = \{0,0,1\}$ ,

case==1,  $\text{CapitalXi}s = \{.7,.2,1\}$ , case==2,  $\text{CapitalXi}s = \{.4,.7,1\}$ , case==3,  $\text{CapitalXi}s = \{.7,.4,.3\}$ ];

(\*  $\text{CapitalXi}N$  is defined relative to either  $[0,1]$  or  $[-1,1]$  \*)

If[optCube==2,  $\text{CapitalXi}N = \{-1,-1,-1\} + 2\text{CapitalXi}s$ ,  $\text{CapitalXi}N = \text{CapitalXi}s$ ;

t0=Graphics3D[{Red,PointSize[0.04],Point[ $\text{CapitalXi}N$ ], Red, Thickness[0.015],

Line[{ $\text{CapitalXi}N + \{xi[1],0,0\}$ ,  $\text{CapitalXi}N + \{xi[M],0,0\}$ },Green,

Line[{ $\text{CapitalXi}N + \{0,eta[1],0\}$ ,  $\text{CapitalXi}N + \{0,eta[M],0\}$ },Blue,

Line[{ $\text{CapitalXi}N + \{0,0,zeta[1]\}$ ,  $\text{CapitalXi}N + \{0,0,zeta[2]\}$ }}];

plotCube[ $\text{CapitalXi}N$ ]=Show[plotCubeRef2,t0,PlotLabel->"Fixed Pt & Coord Lines",

ViewPoint->{3,-2.5,2}]

(\* 7a. Coordinate Lines in Physical Space\*)

$\text{XN} = \text{Chop}[\text{X} / . \{ \text{Xi} \rightarrow \text{CapitalXi}N[[1]], \text{Eta} \rightarrow \text{CapitalXi}N[[2]], \text{Zeta} \rightarrow \text{CapitalXi}N[[3]] \}$ ];

pt0=Graphics3D[{Red,PointSize[0.04],Point[XN]}; Clear[t];

curv[Xi]=X/.{Xi->CapitalXiN[[1]]+t, Eta->CapitalXiN[[2]], Zeta->CapitalXiN[[3]]};

curv[Eta]=X/.{Xi->CapitalXiN[[1]], Eta->CapitalXiN[[2]]+t, Zeta->CapitalXiN[[3]]};

curv[Zeta]=X/.{Xi->CapitalXiN[[1]], Eta->CapitalXiN[[2]], Zeta->CapitalXiN[[3]]+t};

pcurvs=ParametricPlot3D[{curv[Xi],curv[Eta],curv[Zeta]},{t,xi[1],xi[M]},

PlotStyle->{{Thickness[0.01],Red},{Thickness[0.01],Darker[Green,.2]},{Thickness[0.01],Blue}}];

plotCoordCurves=Show[plotbe,pt0,pcurvs,PlotRange->All,

PlotLabel->"Coordinate Curves"]

(\* 7b. Covariant base vectors are the columns of  $dXd[\text{CapitalXi}N]$ , N for numerical\*)

$dXd[\text{CapitalXi}N] = \text{Chop}[dXd[\text{CapitalXi}]/. \{ \text{Xi} \rightarrow \text{CapitalXi}N[[1]],$

$\text{Eta} \rightarrow \text{CapitalXi}N[[2]], \text{Zeta} \rightarrow \text{CapitalXi}N[[3]] \}$ ];

$dXd[\text{CapitalXi}N]^T = \text{Transpose}[dXd[\text{CapitalXi}N]$ ;

taN=Table[Subscript[aN, i]= $dXd[\text{CapitalXi}N]^T[[i]]$ , {i,1,3}];

Subscript[v, 1]=Graphics3D[{Red,Arrowheads[0.03],Thickness[.008],Arrow[{XN,XN+Subscript[aN,

1]}]}];

Subscript[v,

2]=Graphics3D[{Darker[Green,.2],Arrowheads[0.03],Thickness[.008],Arrow[{XN,XN+Subscript[aN,

2]}]}];

Subscript[v,

3]=Graphics3D[{Blue,Arrowheads[0.03],Thickness[.008],Arrow[{XN,XN+Subscript[aN, 3]}]}];

tCoVec={Subscript[v, 1],Subscript[v, 2],Subscript[v, 3]}];

plotCo=Show[plotCoordCurves,tCoVec,PlotLabel->"Covariant Vectors (Thin Arrows)",

PlotRange->All]

length[v\_List]:=Chop[Sqrt[v.v]];

tlaN=Table[Subscript[laN, k]=length[Subscript[aN, k]],{k,1,3}];

JDet=J=Det[dXd[ $\text{CapitalXi}N$ ];

```
(* 7c. J*Contravariant Base Vectors, Thick Arrows*) IgnorePdfPageNumber;
tJaNSup={
Subscript[JaNSup, 1]=Chop[crossPro[Subscript[aN, 2], Subscript[aN, 3]]],
Subscript[JaNSup, 2]=Chop[crossPro[Subscript[aN, 3], Subscript[aN, 1]]],
Subscript[JaNSup, 3]=Chop[crossPro[Subscript[aN, 1], Subscript[aN, 2]]];
Subscript[JvCon, 1]=Graphics3D[{Thickness[0.005],Red,Arrowheads[0.06],
Arrow[{XN,XN+Subscript[JaNSup, 1]}]}];
Subscript[JvCon, 2]=Graphics3D[{Thickness[0.005],Green,Arrowheads[0.06],
Arrow[{XN,XN+Subscript[JaNSup, 2]}]}];
Subscript[JvCon, 3]=Graphics3D[{Thickness[0.005],Blue,Arrowheads[0.06],
Arrow[{XN,XN+Subscript[JaNSup, 3]}]}];
(*Table of length of J*contravariant base vectors*)
tJaNSup=Table[Subscript[lJaNSup, k]=length[Subscript[JaNSup, k]],{k,1,3}];
plotJContra=Show[plotbe,pcurvs,pt0,Subscript[JvCon, 1],Subscript[JvCon, 2],Subscript[JvCon, 3],
PlotLabel->"J*Contr Vectors (Thick Arrows)",PlotRange->All]
(* 7d. One J*Contravariant Base Vectors, set n1 to either 1,2,or 3*)
{n1=3, n2=Mod[n1,3]+1,n3=Mod[n1+1,3]+1};
plotJContra=Show[plotbe,pcurvs,pt0,
Subscript[JvCon, n1],Subscript[v, 1],Subscript[v, 2],Subscript[v, 3],
PlotLabel->"1 J*Contr. & 3 Cov. Vectors",PlotRange->All]

(* 8a. Differential Volume increment, reference space*)
d[Xi]=.15*optCube;
d[CapitalXi]={Subscript[d[Xi]Sup, 1]=d[Xi],Subscript[d[Xi]Sup, 2]=d[Xi],
Subscript[d[Xi]Sup, 3]=d[Xi]};
t[CapitalXi]PSmCu=Table[CapitalXiPSmCu[i,j,k]=CapitalXiN+
Subscript[d[Xi]Sup, 1](i-1){1,0,0}+Subscript[d[Xi]Sup, 2](j-1){0,1,0}+
Subscript[d[Xi]Sup, 3](k-1){0,0,1},{i,1,2},{j,1,2},{k,1,2}];
plotPtsSmCu=Graphics3D[{PointSize[0.02],Point[Flatten[t[CapitalXi]PSmCu,2]],
Red,PointSize[0.04],Point[CapitalXiN]};
plotLinesSmCu=plotLinesHOCube[t[CapitalXi]PSmCu,2,{Darker[Red,0.2],
Darker[Green,0.2],Darker[Blue,0.2], 0.005}];
IncrVolRef=Show[plotCubeRef2,plotPtsSmCu,plotLinesSmCu,
BaseStyle->{FontSize->20,FontFamily->"Times"},Axes->True,
AxesLabel->{"[Xi]^1",[Xi]^2,[Xi]^3},ViewPoint->{2,-4,2}]

(* 8b. Differential Volume increment, physical space*)
tXPSmCu=Table[XPSmCu[i,j,k]=
Chop[X/.{[Xi]->CapitalXi]PSmCu[i,j,k][[1]],Eta->CapitalXi]PSmCu[i,j,k][[2]],
Zeta->CapitalXi]PSmCu[i,j,k][[3]]},{i,1,2},{j,1,2},{k,1,2}];
plotPtsPhysSmCu=Graphics3D[{PointSize[0.012],Point[Flatten[tXPSmCu,2]],PointSize[0.02],
Red,Point[XPSmCu[1,1,1]}];
plotLinesPhysSmCu=plotLinesHOCube[tXPSmCu,2,{Darker[Red,.2],Darker[Green,.2],
Darker[Blue,.2], 0.005}];
IncrVolPhys=Show[plotbe,pt0,pcurvs,plotPtsPhysSmCu,plotLinesPhysSmCu,
ViewPoint->{3,-3,3}]
(* end program *) IgnorePdfPageNumber;
```