LA-UR-20-27150
Approved for public release; distribution is unlimited.

Title: MCNP® Code Version 6.3.0 Unstructured-mesh Quality Metrics & Assessment

Author(s): Kulesza, Joel A.

Intended for: Report

Issued: 2022-09-16 (rev.1)

Los Alamos
NATIONAL LABORATORY

LA-UR-20-27150

MCNP®

Code Version 6.3.0

Unstructured-mesh Quality Metrics & Assessment

Document compiled from `git` hash `a4c869d3e5` on September 8, 2022.

Los Alamos
NATIONAL LABORATORY

MCNP® Code Version 6.3.0 Unstructured-mesh Quality Metrics & Assessment

LA-UR-20-27150

September 8, 2022

Los Alamos National Laboratory

Joel A. Kulesza[1]

[1] Monte Carlo Codes Group (XCP-3)

Contents

1 Introduction

This report describes a variety of metrics calculated by the MCNP code for a user to assess the quality of an input unstructured mesh (UM). These UM quality metrics are calculated automatically, but the user has the opportunity to opt-out of doing so. At present, only linear tetrahedral elements and linear hexahedral elements have extensive quality metrics calculated and reported. The remaining elements have only the elemental Jacobian matrix determinant calculated and reported consistent with the `um_pre_op` utility [1, 2]. However, this negates the need to separately run the `um_pre_op` element checker.

The quality metrics assessed and reported by the MCNP code for linear tetrahedral and hexahedral elements are a subset of those that have historically been used in the finite-element analysis (FEA) community to identify mesh adequate for the deterministic structural, thermal, mechanical, etc. solvers applied to the mesh. For example, the Verdict library [3] can be used to perform UM quality assessment. Only a subset of metrics that have recommended ranges are reported by the MCNP code. These metrics may not be directly applicable to Monte Carlo particle transport analyses. Regardless, they do provide a quantitative measure of the shapes and features of individual mesh elements to assist a user in determining mesh quality.

Moreover, the MCNP code cannot predict which metrics may be of particular interest to a user and/or which may predict undesirable calculation behavior. Thus, the code will not halt if it identifies mesh elements that have quality metrics outside of the recommended ranges. Instead, the code will issue a warning and provide statistical information on the distribution of metrics observed. As such, the user must assess the metrics provided and take any remedial or follow-up action, as necessary.

This report is organized as follows: Sections 2 and 3 derive and describe the mesh quality metrics for linear tetrahedral and hexahedral elements, respectively. These derivations are reproduced here rather than referring to another document (e.g., [3]) to ensure that self-consistent notation is used and to be explicit in the derivations. Because no new metrics are assessed for other types of elements, [1, 2] remain appropriate references and the discussion therein is not repeated here. Next, Section 4 gives the result of several tests used to verify that the MCNP implementations of the formulae derived in Sections 2 and 3 are correct. Section 5 provides possible techniques to further interrogate UM quality with the ParaView software [4].

Note that this work uses zero-based indexing to provide as direct a comparison to [3] as possible. Also, note that this report does not describe how to control or interpret UM quality output. For that information, see [2]. Finally, many of the calculations herein include division where division by zero may occur. Such issues are not addressed in this document further, but the implementation within the MCNP code ensures that if division by a suitably small number occurs (e.g., $< 10^{-30}$), then the associated element quality for that particular metric is set to a high number, which indicates a poor quality element (e.g., 10^{30}).

2 Linear Tetrahedra Quality Metrics

Before addressing the metrics, some frequently used terms are defined based on the tetrahedron shown in Fig. 1, which gives vertex and edge (i.e., connectivity) information. These terms follow.

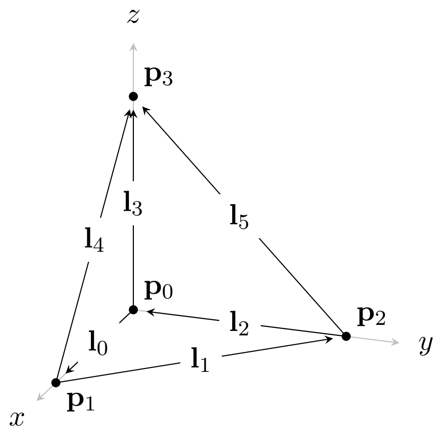


Figure 1: Linear Tetrahedron Overview

The edge vectors of the tetrahedron are given as

$$\mathbf{l}_0 = \mathbf{p}_1 - \mathbf{p}_0, \tag{1}$$

$$\mathbf{l}_1 = \mathbf{p}_2 - \mathbf{p}_1, \tag{2}$$

$$\mathbf{l}_2 = \mathbf{p}_0 - \mathbf{p}_2, \tag{3}$$

$$\mathbf{l}_3 = \mathbf{p}_3 - \mathbf{p}_0, \tag{4}$$

$$\mathbf{l}_4 = \mathbf{p}_3 - \mathbf{p}_1, \tag{5}$$

$$\mathbf{l}_5 = \mathbf{p}_3 - \mathbf{p}_2, \tag{6}$$

which have corresponding lengths

$$\ell_i = \|\mathbf{l}_i\|, \tag{7}$$

where $\|\ldots\|$ indicates the Euclidean norm unless otherwise noted. The volume of the tetrahedron is

$$V = \frac{(\mathbf{l}_2 \times \mathbf{l}_0) \cdot \mathbf{l}_3}{6}, \tag{8}$$

where "$\times$" indicates a cross (outer) product and "$\cdot$" indicates a dot (inner) product. The total surface area of the tetrahedron can be calculated as

$$A = \frac{1}{2}(\|\mathbf{l}_2 \times \mathbf{l}_0\| + \|\mathbf{l}_3 \times \mathbf{l}_0\| + \|\mathbf{l}_4 \times \mathbf{l}_1\| + \|\mathbf{l}_3 \times \mathbf{l}_2\|). \tag{9}$$

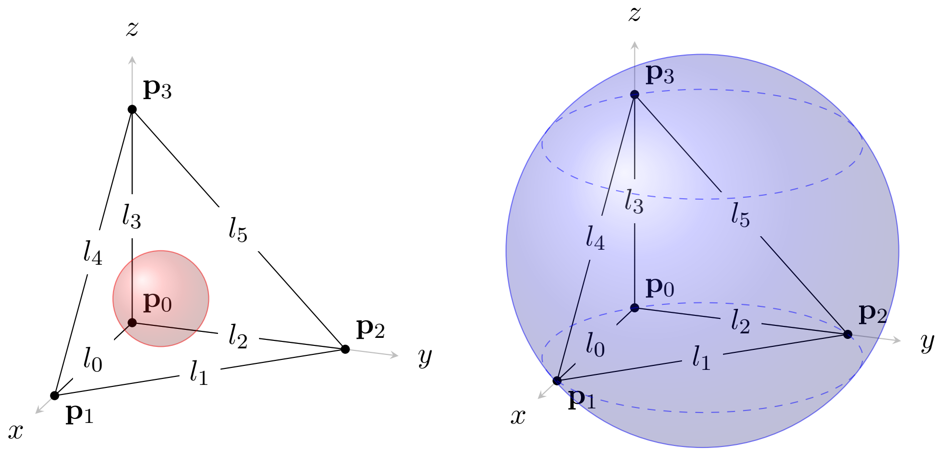
The "inradius," which is the radius of the largest sphere that can be inscribed within the tetrahedron, is

$$r = \frac{3V}{A} \tag{10}$$

and the "circumradius," which is the radius of the smallest sphere that inscribes the tetrahedron, is

$$R = \frac{\left\| \|\mathbf{l}_3\|^2 (\mathbf{l}_2 \times \mathbf{l}_0) + \|\mathbf{l}_2\|^2 (\mathbf{l}_3 \times \mathbf{l}_0) + \|\mathbf{l}_0\|^2 (\mathbf{l}_3 \times \mathbf{l}_2) \right\|}{12V}. \tag{11}$$

Two spheres demonstrating the inradius and circumradius (i.e., the insphere and circumsphere) are shown in Fig. 2. The incenter of the insphere is



(a) Insphere (b) Circumsphere

Figure 2: Spheres Characteristic of a First-order Tetrahedron

$$\mathbf{c}_r = \frac{\|f(\mathbf{p}_0,\mathbf{p}_1,\mathbf{p}_2)\|\mathbf{p}_3 + \|f(\mathbf{p}_0,\mathbf{p}_1,\mathbf{p}_3)\|\mathbf{p}_2 + \|f(\mathbf{p}_0,\mathbf{p}_2,\mathbf{p}_3)\|\mathbf{p}_1 + \|f(\mathbf{p}_1,\mathbf{p}_2,\mathbf{p}_3)\|\mathbf{p}_0}{\|f(\mathbf{p}_0,\mathbf{p}_1,\mathbf{p}_2)\| + \|f(\mathbf{p}_0,\mathbf{p}_1,\mathbf{p}_3)\| + \|f(\mathbf{p}_0,\mathbf{p}_2,\mathbf{p}_3)\| + \|f(\mathbf{p}_1,\mathbf{p}_2,\mathbf{p}_3)\|}, \tag{12}$$

where

$$f(\mathbf{p}_i,\mathbf{p}_j,\mathbf{p}_k) = (\mathbf{p}_j - \mathbf{p}_i) \times (\mathbf{p}_k - \mathbf{p}_i). \tag{13}$$

The incenter of the circumsphere, $\mathbf{c}_R(c_{R,x}, c_{R,y}, c_{R,z})$, is obtained by solving the equation

$$\det\left(\begin{bmatrix} c_{R,x}^2 + c_{R,y}^2 + c_{R,z}^2 & c_{R,x} & c_{R,y} & c_{R,z} & 1 \\ p_{0,x}^2 + p_{0,y}^2 + p_{0,z}^2 & p_{0,x} & p_{0,y} & p_{0,z} & 1 \\ p_{1,x}^2 + p_{1,y}^2 + p_{1,z}^2 & p_{1,x} & p_{1,y} & p_{1,z} & 1 \\ p_{2,x}^2 + p_{2,y}^2 + p_{2,z}^2 & p_{2,x} & p_{2,y} & p_{2,z} & 1 \\ p_{3,x}^2 + p_{3,y}^2 + p_{3,z}^2 & p_{3,x} & p_{3,y} & p_{3,z} & 1 \end{bmatrix}\right) = 0, \tag{14}$$

where $p_{i,x}$ is the x coordinate of point $\mathbf{p}_i$, $p_{i,y}$ is the y coordinate of point $\mathbf{p}_i$, and $p_{i,z}$ is the z coordinate of point $\mathbf{p}_i$ and $\det(\cdots)$ indicates the matrix determinant.

One approach to solving Eq. (14) is to decompose the 5×5 matrix into a series of 4×4 matrices and to then solve for the determinant of each,

$$D_\alpha = \det\left(\begin{bmatrix} p_{0,x} & p_{0,y} & p_{0,z} & 1 \\ p_{1,x} & p_{1,y} & p_{1,z} & 1 \\ p_{2,x} & p_{2,y} & p_{2,z} & 1 \\ p_{3,x} & p_{3,y} & p_{3,z} & 1 \end{bmatrix}\right) \tag{15}$$

$$D_\gamma = \det\left(\begin{bmatrix} p_{0,x}^2 + p_{0,y}^2 + p_{0,z}^2 & p_{0,x} & p_{0,y} & p_{0,z} \\ p_{1,x}^2 + p_{1,y}^2 + p_{1,z}^2 & p_{1,x} & p_{1,y} & p_{1,z} \\ p_{2,x}^2 + p_{2,y}^2 + p_{2,z}^2 & p_{2,x} & p_{2,y} & p_{2,z} \\ p_{3,x}^2 + p_{3,y}^2 + p_{3,z}^2 & p_{3,x} & p_{3,y} & p_{3,z} \end{bmatrix}\right) \tag{16}$$

$$D_x = \det\left(\begin{bmatrix} p_{0,x}^2 + p_{0,y}^2 + p_{0,z}^2 & p_{0,y} & p_{0,z} & 1 \\ p_{1,x}^2 + p_{1,y}^2 + p_{1,z}^2 & p_{1,y} & p_{1,z} & 1 \\ p_{2,x}^2 + p_{2,y}^2 + p_{2,z}^2 & p_{2,y} & p_{2,z} & 1 \\ p_{3,x}^2 + p_{3,y}^2 + p_{3,z}^2 & p_{3,y} & p_{3,z} & 1 \end{bmatrix}\right) \tag{17}$$

$$D_y = -\det\left(\begin{bmatrix} p_{0,x}^2 + p_{0,y}^2 + p_{0,z}^2 & p_{0,x} & p_{0,z} & 1 \\ p_{1,x}^2 + p_{1,y}^2 + p_{1,z}^2 & p_{1,x} & p_{1,z} & 1 \\ p_{2,x}^2 + p_{2,y}^2 + p_{2,z}^2 & p_{2,x} & p_{2,z} & 1 \\ p_{3,x}^2 + p_{3,y}^2 + p_{3,z}^2 & p_{3,x} & p_{3,z} & 1 \end{bmatrix}\right) \tag{18}$$

$$D_z = \det\left(\begin{bmatrix} p_{0,x}^2 + p_{0,y}^2 + p_{0,z}^2 & p_{0,x} & p_{0,y} & 1 \\ p_{1,x}^2 + p_{1,y}^2 + p_{1,z}^2 & p_{1,x} & p_{1,y} & 1 \\ p_{2,x}^2 + p_{2,y}^2 + p_{2,z}^2 & p_{2,x} & p_{2,y} & 1 \\ p_{3,x}^2 + p_{3,y}^2 + p_{3,z}^2 & p_{3,x} & p_{3,y} & 1 \end{bmatrix}\right), \tag{19}$$

to then compute

$$\mathbf{c}_{R,x} = \frac{D_x}{2D_\alpha} \tag{20}$$

$$\mathbf{c}_{R,y} = \frac{D_y}{2D_\alpha} \tag{21}$$

$$\mathbf{c}_{R,z} = \frac{D_z}{2D_\alpha}, \tag{22}$$

while noting that

$$R = \frac{\sqrt{D_x^2 + D_y^2 + D_z^2 + 4D_\alpha D_\gamma}}{2|D_\alpha|}. \tag{23}$$

2.1 Aspect Ratio, Frobenius

The Frobenius aspect ratio is computed as

$$q = \frac{\frac{3}{2}[(\mathbf{l}_0 \cdot \mathbf{l}_0) + (\mathbf{l}_2 \cdot \mathbf{l}_2) + (\mathbf{l}_3 \cdot \mathbf{l}_3)] - [(\mathbf{l}_0 \cdot -\mathbf{l}_2) + (\mathbf{l}_0 \cdot \mathbf{l}_3) + (-\mathbf{l}_2 \cdot \mathbf{l}_3)]}{3\left\{2[\mathbf{l}_0 \cdot (-\mathbf{l}_2 \times \mathbf{l}_3)]^2\right\}^{1/3}}, \tag{24}$$

where this formulation is consistent with the implementation in the Verdict library.

Normalizations are included such that the Frobenius aspect ratio is one for a regular tetrahedron. The recommended range of values is 1–1.3 [3].

2.2 Aspect Ratio, Gamma

The gamma aspect ratio computes the root-mean-square edge length to volume. The root-mean-square edge length is calculated as

$$l_{\mathrm{rms}} = \sqrt{\frac{\sum_{i=0}^{5} \|\mathbf{l}_i\|^2}{6}}. \tag{25}$$

The quality metric is then

$$q = \frac{l_{\mathrm{rms}}^3 \sqrt{2}}{12|V|} \tag{26}$$

where the constants arise from normalizing the metric to one for an equilateral tetrahedron.

The recommended range of values is 1–3 [3].

2.3 Aspect Ratio

The aspect ratio for a tetrahedron gives the maximum edge length versus the inradius and is calculated as

$$q = \frac{\max_{i=0,\ldots,5}(\ell_i)}{2\sqrt{6}r} \tag{27}$$

and has a recommended range of 1–3 [3].

2.4 Condition

The condition of a tetrahedron is calculated by first defining vectors

$$\mathbf{v}_0 = \mathbf{l}_0, \tag{28}$$

$$\mathbf{v}_1 = \frac{-2\mathbf{l}_2 - \mathbf{l}_0}{\sqrt{3}}, \tag{29}$$

$$\mathbf{v}_2 = \frac{3\mathbf{l}_3 + \mathbf{l}_2 - \mathbf{l}_0}{\sqrt{6}}, \tag{30}$$

scalars

$$t_0 = \mathbf{v}_0 \cdot \mathbf{v}_0 + \mathbf{v}_1 \cdot \mathbf{v}_1 + \mathbf{v}_2 \cdot \mathbf{v}_2, \tag{31}$$

$$t_1 = (\mathbf{v}_0 \times \mathbf{v}_1) \cdot (\mathbf{v}_0 \times \mathbf{v}_1) + (\mathbf{v}_1 \times \mathbf{v}_2) \cdot (\mathbf{v}_1 \times \mathbf{v}_2) + (\mathbf{v}_0 \times \mathbf{v}_2) \cdot (\mathbf{v}_0 \times \mathbf{v}_2), \tag{32}$$

and computing the determinant

$$v_{\mathrm{det}} = \det(\mathbf{v}_0 \cdot (\mathbf{v}_1 \times \mathbf{v}_2)). \tag{33}$$

In turn, the condition number is calculated as

$$q = \frac{\sqrt{t_0 t_1}}{3 v_{\mathrm{det}}} \tag{34}$$

and has a recommended range of 1–3 [3].

2.5 Edge Ratio

The edge ratio is computed as

$$q = \frac{\max_{i=0,\ldots,5}(\ell_i)}{\min_{i=0,\ldots,5}(\ell_i)} \tag{35}$$

and has a recommended range of 1–3 [3].

2.6 Jacobian

The Jacobian[1] for an element is computed as

$$q = \mathbf{l}_3 \cdot (\mathbf{l}_2 \times \mathbf{l}_0). \tag{36}$$

While there is no recommended range for the Jacobian of an element, a positive Jacobian indicates an element with non-zero volume that does not exhibit twist. Because a positive volume is required for meaningful track-length edit calculations, a warning is issued if a non-positive Jacobian is detected.

Because the Jacobian is directly proportional to volume, elemental volume is not assessed.

2.7 Minimum Dihedral Angle

To calculate the minimum dihedral angle between two faces on a tetrahedron, one must first compute the unit normal vectors for each of the four faces as

$$\mathbf{n}_0 = \frac{\mathbf{l}_0 \times \mathbf{l}_1}{\|\mathbf{l}_0 \times \mathbf{l}_1\|}, \tag{37}$$

$$\mathbf{n}_1 = \frac{\mathbf{l}_0 \times \mathbf{l}_3}{\|\mathbf{l}_0 \times \mathbf{l}_3\|}, \tag{38}$$

$$\mathbf{n}_2 = \frac{\mathbf{l}_3 \times \mathbf{l}_5}{\|\mathbf{l}_3 \times \mathbf{l}_5\|}, \tag{39}$$

$$\mathbf{n}_3 = \frac{\mathbf{l}_1 \times \mathbf{l}_5}{\|\mathbf{l}_1 \times \mathbf{l}_5\|}. \tag{40}$$

The dihedral along each of the six edges of the tetrahedron is then

$$d_0 = \arccos(\mathbf{n}_0 \cdot \mathbf{n}_1), \tag{41}$$

$$d_1 = \arccos(\mathbf{n}_0 \cdot \mathbf{n}_2), \tag{42}$$

$$d_2 = \arccos(\mathbf{n}_0 \cdot \mathbf{n}_3), \tag{43}$$

$$d_3 = \arccos(\mathbf{n}_1 \cdot \mathbf{n}_2), \tag{44}$$

$$d_4 = \arccos(\mathbf{n}_1 \cdot \mathbf{n}_3), \tag{45}$$

$$d_5 = \arccos(\mathbf{n}_2 \cdot \mathbf{n}_3), \tag{46}$$

so the minimum dihedral angle, in degrees, is

$$q = \frac{180 \min_{i=0,\ldots,5}(d_i)}{\pi} \tag{47}$$

and has a recommended range of $40°$–$180 \arccos(1/3)/\pi$ ($\approx 70.5287793655°$) [3].

2.8 Radius Ratio (also known as Aspect Ratio, Beta)

The Radius Ratio (historically called the Aspect Ratio Beta in prior versions of [3] and in the ParaView [4] Mesh Quality filter) is the normalized ratio of the inscribed sphere to the radius of the circumsphere. The factor of 1/3 is incorporated so an equilateral tetrahedron has a quality of 1.

The Radius Ratio is calculated as

$$q = \frac{R}{3r}. \tag{48}$$

and has a recommended range of 1–3 [3].

[1] Throughout this document, [3], and in the broader mathematics community, there is room for confusion when the unqualified term "Jacobian" is used. The term Jacobian can refer to either a matrix or a scalar value, where the scalar value is the determinant of that matrix. Throughout this work, the term Jacobian will always refer to the determinant of the associated Jacobian matrix.

Table 1: Summary of First-order Tetrahedron Quality Metrics

Metric Name	Recommended Minimum	Recommended Maximum
Aspect Ratio, Frobenius	1.0	1.3
Aspect Ratio, Gamma	1.0	3.0
Aspect Ratio	1.0	3.0
Condition	1.0	3.0
Edge Ratio	1.0	3.0
Jacobian	ϵ	∞
Minimum Dihedral Angle	40	$180 \arccos(1/3)/\pi$
Radius Ratio (Aspect Ratio, Beta)	1.0	3.0
Scaled Jacobian	0.5	$\sqrt{2}/2$
Shape	0.3	1.0

ϵ is a minimum positive value, which can be the result of a Fortran call to `tiny(0.0d0)` $\approx 10^{-308}$.
∞ is the largest value, which can be the result of a Fortran call to `huge(0.0d0)` $\approx 10^{308}$.

2.9 Scaled Jacobian

The scaled Jacobian for an element is calculated as

$$q = \frac{J\sqrt{2}}{\max(\{J, \|\mathbf{l}_0\|\|\mathbf{l}_2\|\|\mathbf{l}_3\|, \|\mathbf{l}_0\|\|\mathbf{l}_1\|\|\mathbf{l}_4\|, \|\mathbf{l}_1\|\|\mathbf{l}_2\|\|\mathbf{l}_5\|, \|\mathbf{l}_3\|\|\mathbf{l}_4\|\|\mathbf{l}_5\|\})}, \tag{49}$$

where J is the Jacobian calculated in Section 2.6. The scaled Jacobian has a recommend range of $1/2$–$\sqrt{2}/2$ [3].

2.10 Shape

The shape of an element is calculated as

$$q = \frac{3\left(J\sqrt{2}\right)^{2/3}}{\frac{3}{2}[(\mathbf{l}_0 \cdot \mathbf{l}_0) + (\mathbf{l}_2 \cdot \mathbf{l}_2) + (\mathbf{l}_3 \cdot \mathbf{l}_3)] - [(\mathbf{l}_0 \cdot -\mathbf{l}_2) + (\mathbf{l}_0 \cdot \mathbf{l}_3) + (-\mathbf{l}_2 \cdot \mathbf{l}_3)]}, \tag{50}$$

where J is the Jacobian calculated in Section 2.6 and the denominator is the same as the numerator in Section 2.1. If $J < 0$, then the resulting quality is set to zero ($q = 0$), which is similar to the Verdict implementation. The shape has a recommend range of 0.3–1.0 [3].

2.11 Summary and Example Output

To summarize, the metrics and associated recommend ranges are given in Table 1 and an example of the output is given in Listing 1.

3 Linear Hexahedra Quality Metrics

Before addressing the metrics, some frequently used terms are defined based on the hexahedron shown in Fig. 3, which gives vertex and edge (i.e., connectivity) information. These terms follow.

Listing 1: Example First-order Tetrahedron Quality Metric Output

```
First-order Tetrahedron Quality Metrics
----------------------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- ----------
Metric                   Rec. Min.  Rec. Max.       Mean  Std. Dev. Calc. Min. Calc. Max.  # Outside  # Checked  % Outside
----------------------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- ----------
Aspect Frobenius         1.000E+00  1.300E+00  1.163E+00  6.751E-02  1.028E+00  1.345E+00          6        200      3.000
Aspect Gamma             1.000E+00  3.000E+00  1.255E+00  1.097E-01  1.043E+00  1.560E+00          0        200      0.000
Aspect Ratio             1.000E+00  3.000E+00  1.409E+00  1.555E-01  1.101E+00  1.875E+00          0        200      0.000
Condition                1.000E+00  3.000E+00  1.167E+00  7.441E-02  1.030E+00  1.385E+00          0        200      0.000
Edge Ratio               1.000E+00  3.000E+00  1.544E+00  1.141E-01  1.249E+00  1.739E+00          0        200      0.000
Jacobian                 1.000E-30  1.000E+30  3.499E-01  9.755E-02  1.906E-01  5.719E-01          0        200      0.000
Min. Dihedral Ang.       4.000E+01  7.053E+01  5.242E+01  6.730E+00  3.914E+01  7.227E+01          4        200      2.000
Radius Ratio             1.000E+00  3.000E+00  1.214E+00  1.114E-01  1.037E+00  1.610E+00          0        200      0.000
Scaled Jacobian          5.000E-01  7.071E-01  6.319E-01  8.458E-02  4.760E-01  8.573E-01         56        200     28.000
Shape                    3.000E-01  1.000E+00  8.630E-01  4.960E-02  7.434E-01  9.724E-01          0        200      0.000
----------------------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- ----------
```

Listing 2: Example First-order Hexahedron Quality Metric Output

```
First-order Hexahedron Quality Metrics
----------------------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- ----------
Metric                   Rec. Min.  Rec. Max.       Mean  Std. Dev. Calc. Min. Calc. Max.  # Outside  # Checked  % Outside
----------------------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- ----------
Diagonal                 6.500E-01  1.000E+00  6.272E-01  1.245E-01  3.265E-01  9.421E-01        454        800     56.750
Jacobian                 1.000E-30  1.000E+30  3.240E-03  9.033E-04  1.764E-03  5.296E-03          0        800      0.000
Maximum Edge Ratio       1.000E+00  1.300E+00  1.412E+00  2.042E-01  1.011E+00  1.914E+00        518        800     64.750
Oddy                     0.000E+00  5.000E-01  7.909E+00  3.503E+00  2.269E+00  1.901E+01        800        800    100.000
Scaled Jacobian          5.000E-01  1.000E+00  5.455E-01  8.741E-02  3.366E-01  7.199E-01        228        800     28.500
Shear                    3.000E-01  1.000E+00  5.455E-01  8.741E-02  3.366E-01  7.199E-01          0        800      0.000
Skew                     0.000E+00  5.000E-01  3.981E-01  1.311E-01  3.355E-02  6.971E-01        172        800     21.500
Stretch                  2.500E-01  1.000E+00  4.636E-01  6.928E-02  3.110E-01  6.480E-01          0        800      0.000
Taper                    0.000E+00  5.000E-01  4.132E-01  4.785E-02  3.096E-01  5.552E-01         76        800      9.500
----------------------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- ----------
```

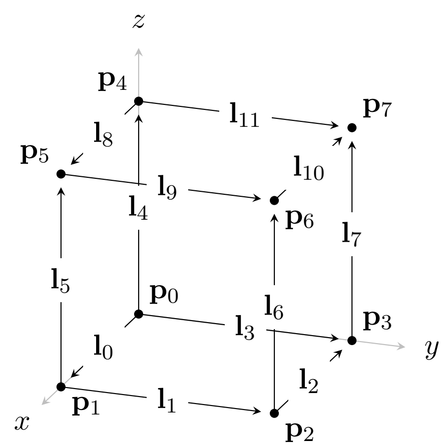


Figure 3: Linear Hexahedron Overview

The edge vectors of the tetrahedron are given as

$$\mathbf{l}_0 = \mathbf{p}_1 - \mathbf{p}_0, \tag{51}$$
$$\mathbf{l}_1 = \mathbf{p}_2 - \mathbf{p}_1, \tag{52}$$
$$\mathbf{l}_2 = \mathbf{p}_3 - \mathbf{p}_2, \tag{53}$$
$$\mathbf{l}_3 = \mathbf{p}_3 - \mathbf{p}_0, \tag{54}$$
$$\mathbf{l}_4 = \mathbf{p}_4 - \mathbf{p}_0, \tag{55}$$
$$\mathbf{l}_5 = \mathbf{p}_5 - \mathbf{p}_1, \tag{56}$$
$$\mathbf{l}_6 = \mathbf{p}_6 - \mathbf{p}_2, \tag{57}$$
$$\mathbf{l}_7 = \mathbf{p}_7 - \mathbf{p}_3, \tag{58}$$
$$\mathbf{l}_8 = \mathbf{p}_5 - \mathbf{p}_4, \tag{59}$$
$$\mathbf{l}_9 = \mathbf{p}_6 - \mathbf{p}_5, \tag{60}$$
$$\mathbf{l}_{10} = \mathbf{p}_7 - \mathbf{p}_6, \tag{61}$$
$$\mathbf{l}_{11} = \mathbf{p}_7 - \mathbf{p}_4, \tag{62}$$

which have corresponding lengths

$$\ell_i = \|\mathbf{l}_i\|. \tag{63}$$

The four diagonal vectors for each hexahedron are

$$\mathbf{d}_0 = \mathbf{p}_6 - \mathbf{p}_0, \tag{64}$$
$$\mathbf{d}_1 = \mathbf{p}_7 - \mathbf{p}_1, \tag{65}$$
$$\mathbf{d}_2 = \mathbf{p}_4 - \mathbf{p}_2, \tag{66}$$
$$\mathbf{d}_3 = \mathbf{p}_5 - \mathbf{p}_3. \tag{67}$$

The principal axes of each hexahedron are

$$\mathbf{x}_0 = (\mathbf{p}_1 - \mathbf{p}_0) + (\mathbf{p}_2 - \mathbf{p}_3) + (\mathbf{p}_5 - \mathbf{p}_4) + (\mathbf{p}_6 - \mathbf{p}_7), \tag{68}$$
$$\mathbf{x}_1 = (\mathbf{p}_3 - \mathbf{p}_0) + (\mathbf{p}_2 - \mathbf{p}_1) + (\mathbf{p}_7 - \mathbf{p}_4) + (\mathbf{p}_6 - \mathbf{p}_5), \tag{69}$$
$$\mathbf{x}_2 = (\mathbf{p}_4 - \mathbf{p}_0) + (\mathbf{p}_5 - \mathbf{p}_1) + (\mathbf{p}_6 - \mathbf{p}_2) + (\mathbf{p}_7 - \mathbf{p}_3). \tag{70}$$

The cross derivatives of each hexahedron are

$$\mathbf{x}_{01} = \mathbf{x}_{10} = (\mathbf{p}_2 - \mathbf{p}_3) + (\mathbf{p}_1 - \mathbf{p}_0) + (\mathbf{p}_6 - \mathbf{p}_7) + (\mathbf{p}_5 - \mathbf{p}_4), \tag{71}$$
$$\mathbf{x}_{02} = \mathbf{x}_{20} = (\mathbf{p}_5 - \mathbf{p}_1) + (\mathbf{p}_4 - \mathbf{p}_0) + (\mathbf{p}_6 - \mathbf{p}_2) + (\mathbf{p}_7 - \mathbf{p}_3), \tag{72}$$
$$\mathbf{x}_{12} = \mathbf{x}_{21} = (\mathbf{p}_7 - \mathbf{p}_4) + (\mathbf{p}_3 - \mathbf{p}_0) + (\mathbf{p}_6 - \mathbf{p}_5) + (\mathbf{p}_2 - \mathbf{p}_1). \tag{73}$$

A 3×3 Jacobian matrix is defined for each of the vertices and the center of the hexahedron as

$$A_0 = \begin{bmatrix} \mathbf{l}_0 & \mathbf{l}_3 & \mathbf{l}_4 \end{bmatrix}, \tag{74}$$
$$A_1 = \begin{bmatrix} \mathbf{l}_1 & -\mathbf{l}_0 & \mathbf{l}_5 \end{bmatrix}, \tag{75}$$
$$A_2 = \begin{bmatrix} \mathbf{l}_2 & -\mathbf{l}_1 & \mathbf{l}_6 \end{bmatrix}, \tag{76}$$
$$A_3 = \begin{bmatrix} -\mathbf{l}_3 & -\mathbf{l}_2 & \mathbf{l}_7 \end{bmatrix}, \tag{77}$$
$$A_4 = \begin{bmatrix} \mathbf{l}_{11} & \mathbf{l}_8 & -\mathbf{l}_4 \end{bmatrix}, \tag{78}$$
$$A_5 = \begin{bmatrix} -\mathbf{l}_8 & \mathbf{l}_9 & -\mathbf{l}_5 \end{bmatrix}, \tag{79}$$
$$A_6 = \begin{bmatrix} -\mathbf{l}_9 & \mathbf{l}_{10} & -\mathbf{l}_6 \end{bmatrix}, \tag{80}$$
$$A_7 = \begin{bmatrix} -\mathbf{l}_{10} & -\mathbf{l}_{11} & -\mathbf{l}_7 \end{bmatrix}, \tag{81}$$
$$A_8 = \begin{bmatrix} \mathbf{x}_0 & \mathbf{x}_1 & \mathbf{x}_2 \end{bmatrix}, \tag{82}$$

where each vector component is a column in the matrix. Using the column vectors $\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2$ in matrix A, we can introduce the notation

$$\|A\| \equiv \|\mathbf{a}_0\|^2 + \|\mathbf{a}_1\|^2 + \|\mathbf{a}_2\|^2 \tag{83}$$

and

$$\|\mathrm{adj}(A)\|^2 \equiv \|\mathbf{a}_0 \times \mathbf{a}_1\|^2 + \|\mathbf{a}_1 \times \mathbf{a}_2\|^2 + \|\mathbf{a}_2 \times \mathbf{a}_0\|^2. \tag{84}$$

Furthermore, the determinant of A is

$$\det(A) \equiv \mathbf{a}_0 \cdot (\mathbf{a}_1 \times \mathbf{a}_2) \tag{85}$$

and normalized A matrices can be calculated and represented as

$$B = \begin{bmatrix} \dfrac{\mathbf{a}_0}{\|\mathbf{a}_0\|} & \dfrac{\mathbf{a}_1}{\|\mathbf{a}_1\|} & \dfrac{\mathbf{a}_2}{\|\mathbf{a}_2\|} \end{bmatrix} \tag{86}$$
$$= \begin{bmatrix} \mathbf{b}_0 & \mathbf{b}_1 & \mathbf{b}_2 \end{bmatrix}, \tag{87}$$

which have determinant

$$\det(B) \equiv \mathbf{b}_0 \cdot (\mathbf{b}_1 \times \mathbf{b}_2). \tag{88}$$

3.1 Diagonal

The diagonal is calculated as

$$q = \frac{\min_{i=0,\ldots,3}\left(\dfrac{\mathbf{d}_i}{\|\mathbf{d}_i\|}\right)}{\max_{i=0,\ldots,3}\left(\dfrac{\mathbf{d}_i}{\|\mathbf{d}_i\|}\right)} \tag{89}$$

and has a recommended range of 0.65–1 [3].

3.2 Jacobian

The Jacobian for an element is calculated as

$$q = \min\left(\min_{i=0,\ldots,7}(\det(A_i)), \frac{\det(A_8)}{64}\right). \tag{90}$$

While there is no recommended range for the Jacobian of an element, a positive Jacobian indicates an element with non-zero volume that does not exhibit twist. Because a positive volume is required for meaningful track-length edit calculations, a warning is issued if a non-positive Jacobian is detected.

3.3 Maximum Edge Ratio

The diagonal is calculated as

$$q = \max\left(\max\left(\frac{\|\mathbf{x}_0\|}{\|\mathbf{x}_1\|}, \frac{\|\mathbf{x}_1\|}{\|\mathbf{x}_0\|}\right), \max\left(\frac{\|\mathbf{x}_0\|}{\|\mathbf{x}_2\|}, \frac{\|\mathbf{x}_2\|}{\|\mathbf{x}_0\|}\right), \max\left(\frac{\|\mathbf{x}_1\|}{\|\mathbf{x}_2\|}, \frac{\|\mathbf{x}_2\|}{\|\mathbf{x}_1\|}\right)\right) \tag{91}$$

and has a recommended range of 1–1.3 [3].

3.4 Oddy

Begin by defining the Oddy of a given Jacobian matrix as

$$O(A_i) = \frac{\|A_i^\intercal A_i\|^2 - \frac{1}{3}\|A_i\|^4}{\det(A_i)^{4/3}}, \tag{92}$$

where "$\intercal$" indicates matrix transpose and recalling the definition given in Eq. (83). Next, the maximum Oddy is calculated as

$$q = \max_{i=0,\ldots,8}(O(A_i)), \tag{93}$$

which can be interpreted as the maximum deviation of the metric tensor $A_i^\intercal A_i$ from the identity matrix evaluated at the corners of the hexahedron and at its center. The Oddy metric has a recommended range of 0–0.5 [3].

3.5 Scaled Jacobian

The scaled Jacobian is calculated from the scaled Jacobian matrices as

$$q = \min_{i=0,\ldots,8}(\det(B_i)) \tag{94}$$

and has a recommended range of 0.5–1 [3].

3.6 Shear

The shear is calculated from the scaled Jacobian matrices as

$$q = \min_{i=0,\ldots,7}(\det(B_i)) \tag{95}$$

and has a recommended range of 0.3–1 [3]. Note that the shear and scaled Jacobian calculations are identical; however, the shear calculation excludes the matrix at the center of the element calculated from the element's principal component axes. This distinction is not made in [3], where there is a typo [5] indicating that the shear and scaled Jacobian calculations are identical but have different recommended ranges.

3.7 Skew

To compute the skew, first calculate normalized principal axes

$$\mathbf{y}_0 = \frac{\mathbf{x}_0}{\|\mathbf{x}_0\|}, \tag{96}$$

$$\mathbf{y}_1 = \frac{\mathbf{x}_1}{\|\mathbf{x}_1\|}, \tag{97}$$

$$\mathbf{y}_2 = \frac{\mathbf{x}_2}{\|\mathbf{x}_2\|}, \tag{98}$$

to calculate individual skew values as

$$s_0 = |\mathbf{y}_0 \cdot \mathbf{y}_1|, \tag{99}$$

$$s_1 = |\mathbf{y}_0 \cdot \mathbf{y}_2|, \tag{100}$$

$$s_2 = |\mathbf{y}_1 \cdot \mathbf{y}_2|, \tag{101}$$

where $|\ldots|$ indicates absolute value. The skew is then the maximum of the individual values,

$$q = \max_{i=0,1,2}(s_i) \tag{102}$$

and has a recommended range of 0–0.5 [3].

3.8 Stretch

The stretch is calculated as

$$q = \sqrt{3}\frac{\min_{i=0,\ldots,11}\left(\frac{\mathbf{l}_i}{\|\mathbf{l}_i\|}\right)}{\max_{i=0,\ldots,3}\left(\frac{\mathbf{d}_i}{\|\mathbf{d}_i\|}\right)} \tag{103}$$

and has a recommended range of 0.25–1 [3].

3.9 Taper

The taper is calculated as

$$q = \max\left(\frac{\|\mathbf{x}_{01}\|}{\min(\|\mathbf{x}_0\|, \|\mathbf{x}_1\|)}, \frac{\|\mathbf{x}_{02}\|}{\min(\|\mathbf{x}_0\|, \|\mathbf{x}_2\|)}, \frac{\|\mathbf{x}_{12}\|}{\min(\|\mathbf{x}_1\|, \|\mathbf{x}_2\|)}\right) \tag{104}$$

and has a recommended range of 0–0.5 [3].

3.10 Summary and Example Output

To summarize, the metrics and associated recommend ranges are given in Table 2 and an example of the output is given in Listing 2.

4 Verification Testing

This section compares the results of the aforementioned metrics calculated during an MCNP calculation versus those calculated using ParaView [4] on the resulting UM XDMF [6] output file.

Table 2: Summary of First-order Hexahedron Quality Metrics

Metric Name	Recommended Minimum	Recommended Maximum
Diagonal	0.65	1.0
Jacobian	ϵ	∞
Maximum Edge Ratio	1.0	1.3
Oddy	0.0	0.5
Scaled Jacobian	0.5	1.0
Shear	0.3	1.0
Skew	0.0	0.5
Stretch	0.25	1.0
Taper	0.0	0.5

ϵ is a minimum positive value, which can be the result of a Fortran call to `tiny(0.0d0)` $\approx 10^{-308}$.
∞ is the largest real value, which can be the result of a Fortran call to `huge(0.0d0)` $\approx 10^{308}$.

4.1 Linear Tetrahedron Test Case

This test case consists of a pair of identical cubes with an irregular mesh of 100 linear tetrahedra each. The mesh is shown in Fig. 4. The cubes are created and meshed using the gmsh software [7].

The aforementioned mesh quality metrics calculated by the MCNP code are listed in Listing 1. The same metrics calculated by ParaView using the script given in Listing 5 are given in Listing 3. A comparison of Listings 1 and 3 shows identical mean, standard-deviation, calculated-minimum, and calculated-maximum values for each metric. This suggests that each MCNP metric is being calculated consistently with ParaView (and Verdict).

4.2 Linear Hexahedron Test Case

This test case consists of a pair of identical cubes with an irregular mesh of 400 linear hexahedra each. The mesh is shown in Fig. 5. The cubes are created and meshed using the gmsh software [7].

The aforementioned mesh quality metrics calculated by the MCNP code are listed in Listing 2. The same metrics calculated by ParaView using the script given in Listing 5 are given in Listing 4. A comparison of Listings 2 and 4 shows identical mean, standard-deviation, calculated-minimum, and calculated-maximum values for each metric. This suggests that each MCNP metric is being calculated consistently with ParaView (and Verdict).

5 Additional Mesh Quality Assessment Techniques

This section provides techniques to assess overall UM quality and how to find specific UM elements of interest using the ParaView software [4]. The techniques described herein are:

- How to apply a Mesh Quality Filter,
- How to obtain Mesh Quality Statistics, and
- How to segregate mesh elements based on their quality and query the associated element and its nodes.

This discussion assumes that a model of interest has already been loaded into the ParaView pipeline. For this discussion, the Stanford bunny [8] with embedded lettering [9] will be used, which consists of only linear tetrahedra. An overview of the mesh wireframe is shown in Fig. 6.

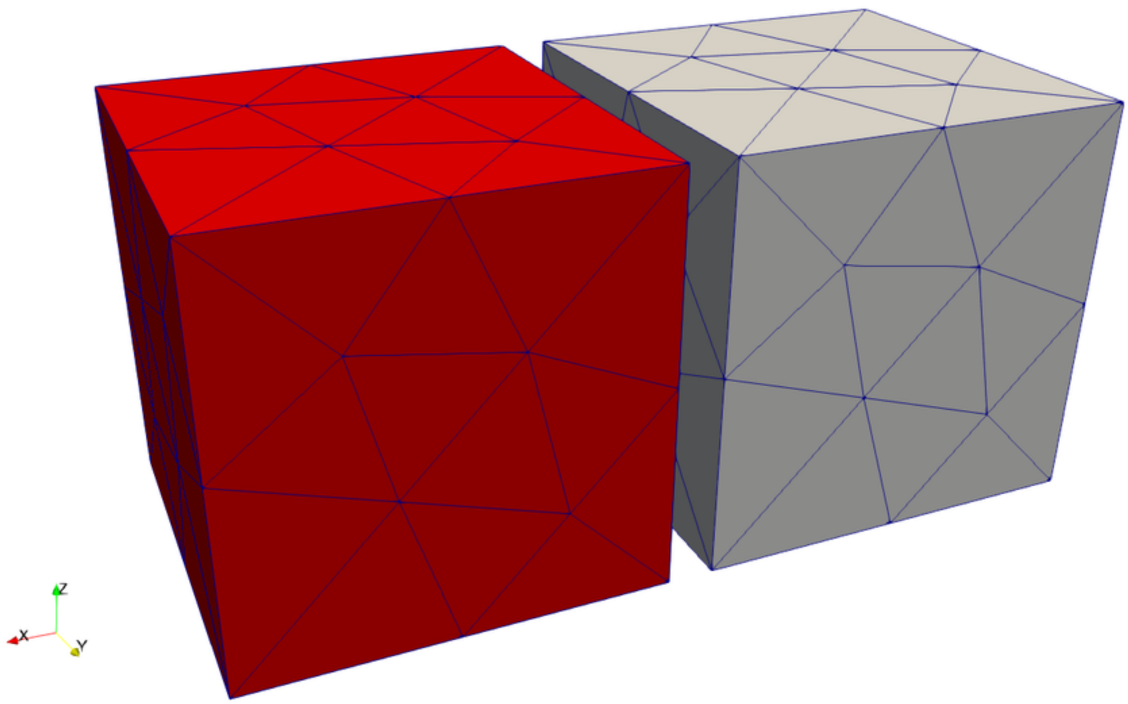


Figure 4: Linear Tetrahedron Test Case Mesh Wireframe

Listing 3: ParaView-calculated First-order Tetrahedron Quality Metrics

```
--------------------------------------------------------------------------------
Linear Tetrahedron Metrics
Metric                         Mean  Std. Dev. Calc. Min. Calc. Max.
Aspect Frobenius          1.163e+00  6.751e-02  1.028e+00  1.345e+00
Aspect Gamma              1.255e+00  1.097e-01  1.043e+00  1.560e+00
Aspect Ratio              1.409e+00  1.555e-01  1.101e+00  1.875e+00
Condition                 1.167e+00  7.441e-02  1.030e+00  1.385e+00
Edge Ratio                1.544e+00  1.141e-01  1.249e+00  1.739e+00
Jacobian                  3.499e-01  9.755e-02  1.906e-01  5.719e-01
Minimum Dihedral Angle    5.242e+01  6.730e+00  3.914e+01  7.227e+01
Aspect Beta               1.214e+00  1.114e-01  1.037e+00  1.610e+00
Scaled Jacobian           6.319e-01  8.458e-02  4.760e-01  8.573e-01
Shape                     8.630e-01  4.960e-02  7.434e-01  9.724e-01
--------------------------------------------------------------------------------
```

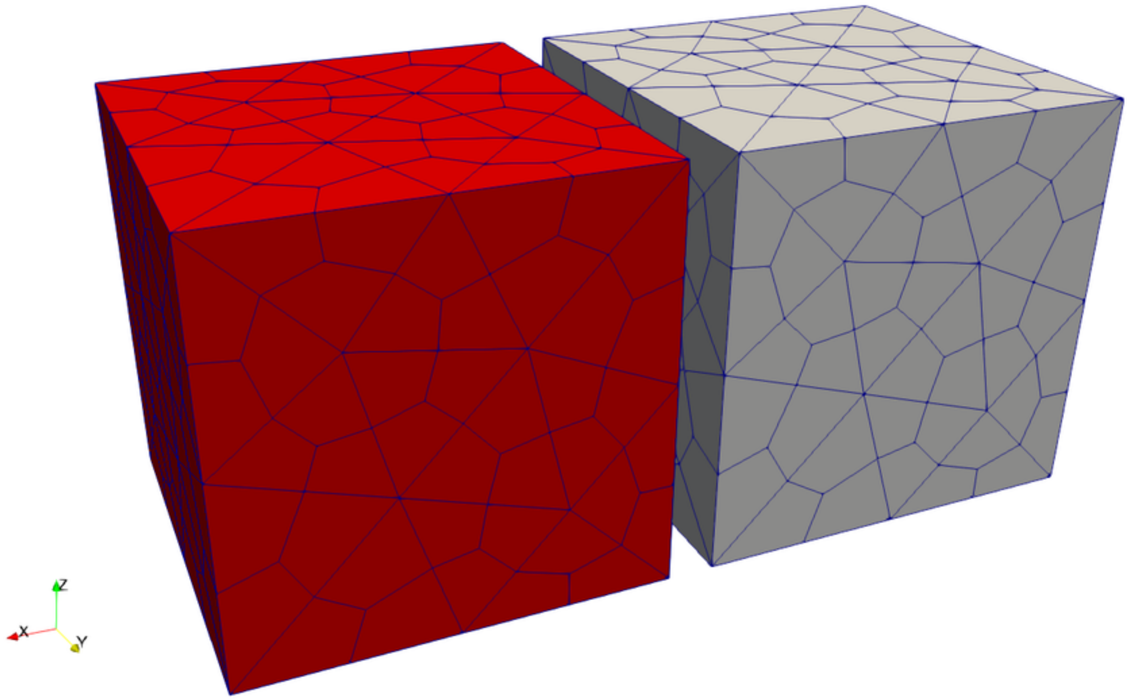


Figure 5: Linear Hexahedron Test Case Mesh Wireframe

Listing 4: ParaView-calculated First-order Hexahedron Quality Metrics

```
--------------------------------------------------------------------------------
Linear Hexahedron Metrics
Metric                              Mean  Std. Dev. Calc. Min. Calc. Max.
Diagonal                       6.272e-01  1.245e-01  3.265e-01  9.421e-01
Jacobian                       3.240e-03  9.033e-04  1.764e-03  5.296e-03
Maximum Edge Ratio             1.412e+00  2.042e-01  1.011e+00  1.914e+00
Oddy                           7.909e+00  3.503e+00  2.269e+00  1.901e+01
Scaled Jacobian                5.455e-01  8.741e-02  3.366e-01  7.199e-01
Shear                          5.455e-01  8.741e-02  3.366e-01  7.199e-01
Skew                           3.981e-01  1.311e-01  3.355e-02  6.971e-01
Stretch                        4.636e-01  6.928e-02  3.110e-01  6.480e-01
Taper                          4.132e-01  4.785e-02  3.096e-01  5.552e-01
--------------------------------------------------------------------------------
```

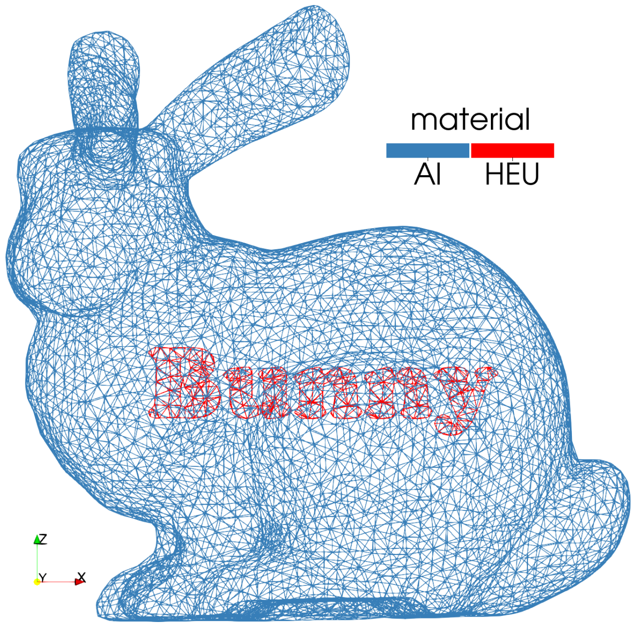


Figure 6: Stanford Bunny Mesh Wireframe Overview

5.1 Applying a Mesh Quality Filter

Before calculating mesh-quality metrics, it is important to consider whether metrics will be sought for the entire mesh or select blocks (if multiple blocks exist). For example, in the model shown in Fig. 6, there are six blocks: one block for the bunny and one block for each of the letters. If statistics are sought for the entire mesh, the blocks should be merged with the "Merge Blocks" filter.

For this example, we proceed by merging the blocks. The result of doing so is similar to what one would observe loading a VTU file [10] produced from an MCNP EEOUT file with the utility described in [11].

Once the blocks are merged, one can apply the Mesh Quality filter from the ParaView Filters menu (either by searching for it or selecting it from the Alphabetical submenu). Once selected, the Properties pane can be used to select from a variety of triangle, quadrilateral, tetrahedron, and hexahedron quality measures. Each selected measure will be applied to all applicable mesh components.

For this discussion, select Radius Ratio from the Tet Quality Measure drop-down menu and click the Apply button. The result should look similar to Fig. 7 (with log-scaling applied to the color bar).

5.2 Quantitatively Evaluating Mesh Quality

With the Mesh Quality filter applied, one may want to qualitatively assess the mesh. To do this, the Descriptive Statistics filter can be used. Select the Mesh Quality object in the Pipeline Browser and apply the Descriptive Statistics filter from the Filters menu (either via search or from the Statistics submenu). Next, optionally de-select all Variables of Interest except the Quality entry and click the Apply button.

As a result of clicking Apply, a spreadsheet view should appear adjacent to the RenderView with minimum, mean, maximum quality reported as well as second, third, and fourth moments of the distribution of mesh qualities to compute values such as standard deviation, skew, and kurtosis. The

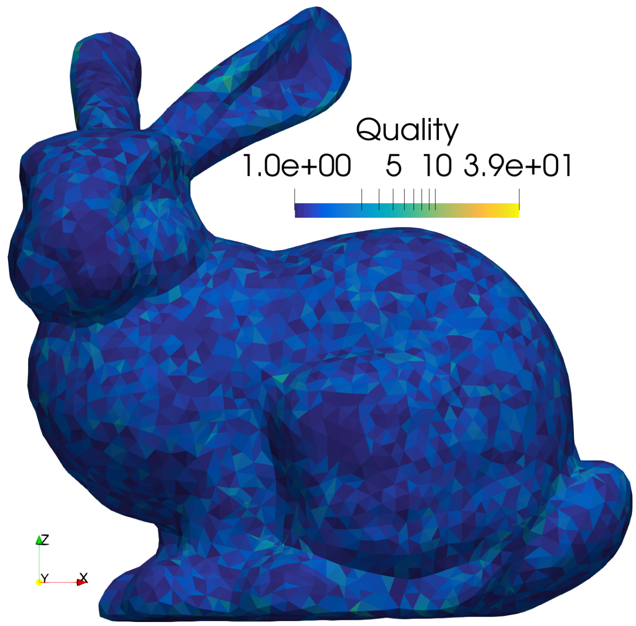


Figure 7: Stanford Bunny Shaded by Radius Ratio, Log-scaled

ParaView interface should look similar to Fig. 8.

5.3 Segregating Mesh Elements of Interest

The statistical information provided in Fig. 8 suggests some mesh-quality outlier values. Understanding where these mesh elements are may allow additional work to be done to refine the mesh to improve the quality of these elements. The most-direct way of isolating the poor-quality elements is with the Find Data dialog available in the Edit menu.

To begin, click Find Data in the Edit menu. Then, ensure that one is selecting Cells from the Mesh Quality entry in the Pipeline Browser and then set the selection criteria to act on Quality and to select all cells with quality greater than or equal to a threshold of interest (in this example, 20 is used). Then, click the Run Selection button. The result should appear as in Fig. 9. A listing of the elements (cells) with quality that meet the aforementioned criterion is listed. These cells can then be segregated by clicking the Extract Selection button, which will create a new Pipeline object. Clicking the Apply button for the new Pipeline object will show only the elements that matched the search criterion.

To view those elements in context, one can re-enable other Pipeline objects, such as the original model. Adjusting line width and/or opacity can help the extract elements stand out. For example, one might display the extract elements as shown in Fig. 10.

Finally, one can further interrogate the model using the mouse to get identifying information about the elements of interest. For example, the Hover Cells On () and/or Hover Points On () buttons will display mesh information as shown in Fig. 11.

Acknowledgements

The author thanks Jerawan C. Armstrong for performing the review of this work and documentation.

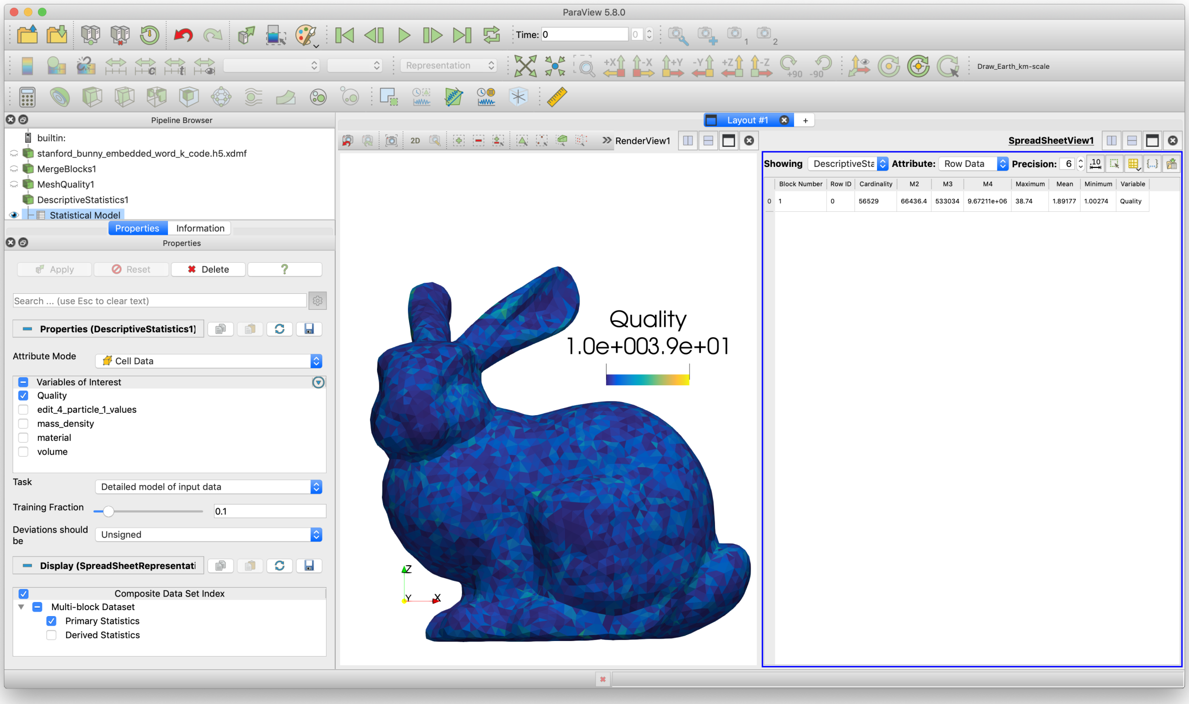


Figure 8: ParaView Interface Example with Mesh Quality Descriptive Statistics Shown

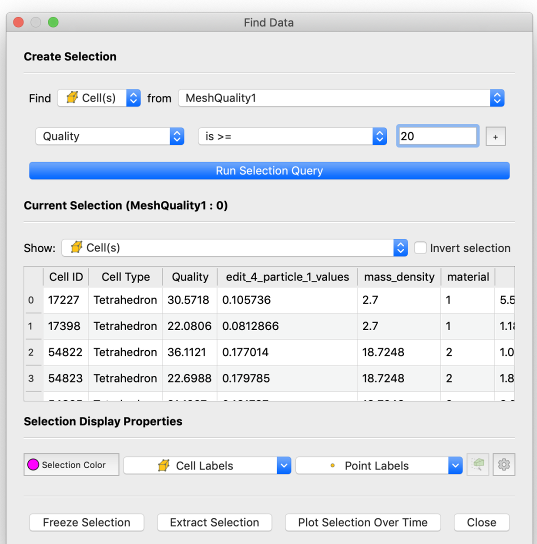
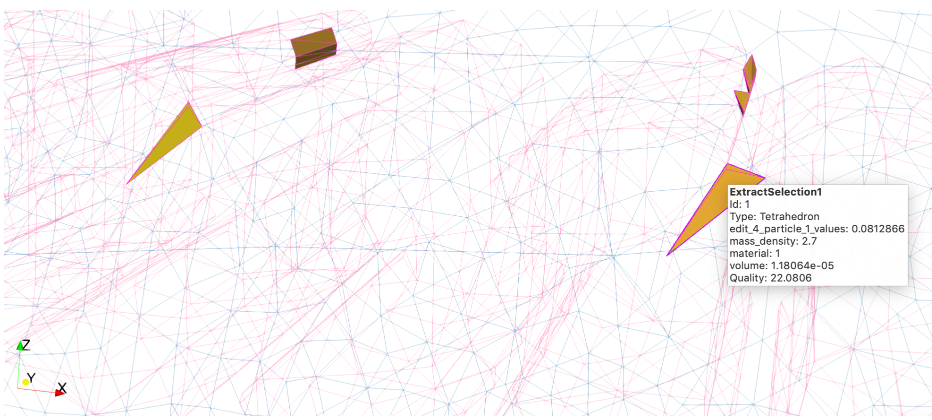
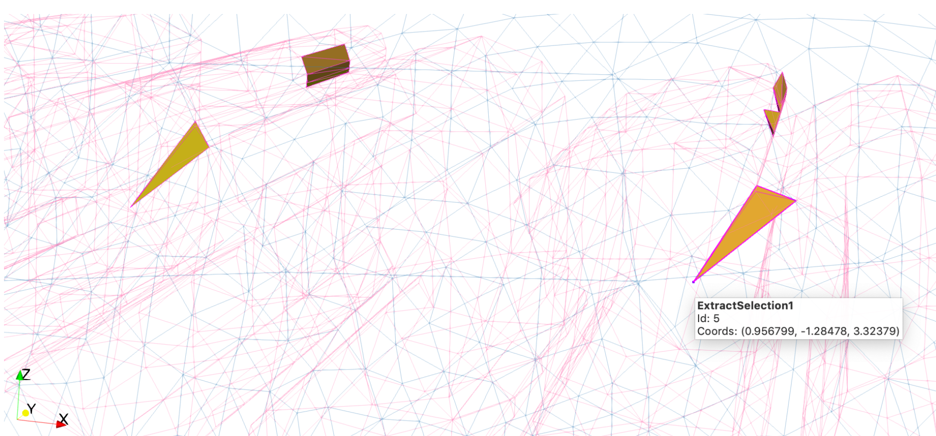


Figure 9: Example Find Data Dialog to Segregate Elements on Mesh Quality

Figure 10: Extracted Poor-quality Elements with Model Wireframe



(a) Hover Cells



(b) Hover Points

Figure 11: Interactive Interrogation of Mesh Information Example

References

[Citing pages are listed after each reference.]

1. R. L. Martz and D. L. Crane, "The MCNP6 Book on Unstructured Mesh Geometry: Foundations," Los Alamos National Laboratory, Los Alamos, NM, USA, Tech. Rep. LA-UR-12-25478, Nov. 2012. [Page 2]

2. R. L. Martz, "The MCNP6 Book On Unstructured Mesh Geometry: User's Guide for MCNP 6.2.1," Los Alamos National Laboratory, Los Alamos, NM, USA, Tech. Rep. LA-UR-18-27630, Aug. 2018. URL: http://permalink.lanl.gov/object/tr?what=info:lanl-repo/lareport/LA-UR-18-27630 [Page 2]

3. C. J. Stimpson, C. D. Ernst, P. K. and. P. P. Pébay, and D. Thompson, "The Verdict Geometric Quality Library," Sandia National Laboratories, Albuquerque, NM, USA, Tech. Rep. SAND2007-1751, Mar. 2007. DOI: 10.2172/901967 [Pages 2, 5, 6, 7, 8, 11, 12, and 13]

4. U. Ayachit, *The ParaView Guide*, community ed., L. Avila, K. Osterdahl, S. McKenzie, and S. Jordan, Eds. Kitware Inc., Jun. 2018. URL: https://www.paraview.org/paraview-guide/ [Pages 2, 7, 13, and 14]

5. C. J. Stimpson, "Verdict Library Inquiry," Private Email Communication, May 2020. [Page 12]

6. "XDMF Model and Format," Website, Apr. 2020. URL: http://xdmf.org/index.php/XDMF_Model_and_Format [Page 13]

7. C. Geuzaine and J.-F. Remacle, "Gmsh: A 3-D Finite Element Mesh Generator with Built-in Pre- and Post-processing Facilities," *International Journal for Numerical Methods in Engineering*, vol. 79, no. 11, pp. 1309–1331, 2009. DOI: 10.1002/nme.2579 [Page 14]

8. Stanford Computer Graphics Laboratory, "The Stanford 3D Scanning Repository," Aug. 2017. URL: http://graphics.stanford.edu/data/3Dscanrep/#bunny [Page 14]

9. J. A. Kulesza, "Recent Work on HDF5 and XDMF-based MCNP Unstructured Mesh Output Files," in *Proceedings of 14th International Conference on Radiation Shielding (ICRS14) and 21st Topical Meeting of the Radiation Protection and Shielding Division (RPSD2020)*, Seattle, WA, USA; September 13–17, 2020. [Page 14]

10. L. S. Avila, U. Ayachit, S. Barré, J. Baumes, F. Bertel, R. Blue, D. Cole, D. DeMarle, B. Geveci, W. A. Hoffman, B. King, K. Krishnan, C. C. Law, K. M. Martin, W. McLendon, P. Pebay, N. Russell, W. J. Schroeder, T. Shead, J. Shepherd, A. Wilson, and B. Wylie, *The VTK User's Guide*, 11th ed. Kitware, Inc., 2010. URL: http://www.vtk.org/vtk-users-guide/ [Page 17]

11. J. A. Kulesza and T. C. McClanahan, "A Python Script to Convert MCNP Unstructured Mesh Elemental Edit Output Files to XML-based VTK Files," Los Alamos National Laboratory, Los Alamos, NM, USA, Tech. Rep. LA-UR-19-20291, rev. 2, Nov. 2019. DOI: 10.2172/1574726 [Page 17]

A ParaView Quality Extraction Script

The script shown in Listing 5 can be executed in batch mode through ParaView's `pvpython` interpreter to compute statistics comparable to those printed in the new MCNP mesh quality output. This script is intended to provide a convenience when verifying correctness of the MCNP-calculated values.

Listing 5: ParaView Quality Tabulation Script

```python
#!/Users/jkulesza/Applications/ParaView-5.8.0.app/Contents/bin/pvpython

import sys

assert len(sys.argv) == 2, "XDMF File not specified."

# trace generated using paraview version 5.8.0
from paraview.simple import *

paraview.simple._DisableFirstRenderCameraReset()

import numpy as np

# Load data set.
xdmf = XDMFReader(FileNames=[sys.argv[1]])

# Create render view (necessary for clean exit).
renderView1 = GetActiveViewOrCreate("RenderView")

# Merge all blocks to get statistics over the entire model.
mergeBlocks = MergeBlocks(Input=xdmf)

# Apply mesh quality filter to mesh.
meshQuality = MeshQuality(Input=mergeBlocks)

# Iterate through mesh quality metrics and print statistics.
def print_metrics(metric_type, metrics, data):
    print(80 * "-")
    print(metric_type + " Metrics")
    print(
        "{:<24s} {:>10s} {:>10s} {:10s} {:>10s}".format(
            "Metric", "Mean", "Std. Dev.", "Calc. Min.", "Calc. Max."
        )
    )
    for q_metric in metrics:
        if metric_type == "Linear Tetrahedron":
            data.TetQualityMeasure = q_metric
        elif metric_type == "Linear Hexahedron":
            data.HexQualityMeasure = q_metric
        else:
            raise ValueError("Unknown mesh quality metric type.")
        q = np.array(
            paraview.servermanager.Fetch(data).GetCellData().GetArray("Quality")
        )
        print(
```

```
            "{:<24s} {:10.3e} {:10.3e} {:10.3e} {:10.3e}".format(
                q_metric, q.mean(), np.std(q), np.min(q), np.max(q)
            )
        )
    print(80 * "-")
    return


# Define metrics to query.
q_metrics_lintet = [
    'Aspect Frobenius',
    'Aspect Gamma',
    'Aspect Ratio',
    'Condition',
    'Edge Ratio',
    'Jacobian',
    'Minimum Dihedral Angle',
    'Aspect Beta',
    'Scaled Jacobian',
    'Shape'
]

q_metrics_linhex = [
    "Diagonal",
    "Jacobian",
    "Maximum Edge Ratio",
    "Oddy",
    "Scaled Jacobian",
    "Shear",
    "Skew",
    "Stretch",
    "Taper",
]

# Execute query and print metrics.
print_metrics("Linear Hexahedron", q_metrics_linhex, meshQuality)
print_metrics("Linear Tetrahedron", q_metrics_lintet, meshQuality)
```