UNSTRUCTURED GRIDS ON NURBS SURFACES

Jamshid Samareh-Abolhassani*

1 Abstract

A simple and efficient computational method is presented for unstructured surface grid generation. This method is built upon an advancing front technique combined with grid projection. The projection technique is based on a Newton-Raphson method. This combined approach has been successfully implemented for structured and unstructured grids. In this paper, the implementation for unstructured grid is discussed.

2 Introduction

The first step in obtaining a Computational Fluid Dynamics (CFD) solution is the creation of structured or unstructured grids. This step typically requires a considerable amount of time and effort on the part of a designer. The process of transforming an aerodynamic configuration from a Computer-Aided Design (CAD) model to a CFD surface grid is referred to as the surface-grid generation process. This process is often a formidable one for complex geometries such as realistic aircraft and spacecraft configurations. CAD systems typically represent the surfaces of aerodynamic vehicles with a set of parametric surfaces such as NonUniform Rational B-Splines (NURBS). Then, CFD surface grids are generated on these NURBS surfaces.

The surface grid can be generated either in a parameter space or on approximated/simplified NURBS surfaces. Generating grids in a parameter space is very common in structured grid generation. This approach has two serious restrictions. The first restriction is that the choice of surface parameterization affects the CFD surface grid. As shown in ref[1], a poor parameterization may cause the CFD surface-grid to be highly skewed. There are several ways to alleviate this problem which have been discussed in great detail in [1]. The second limitation is that a CFD surface grid can not be generated over several overlapping NURBS surfaces. This is the most serious restriction.

In the second method , the NURBS surfaces are approximated by a few smaller bi-linear patches. Then, the surface grids are generated on these bi-linear patches. This method is quite easy to implement, and it avoids the problems associated with surface parameterization. However, the resulting grids are close but not on the original NURBS surfaces. This problem can be alleviated by projecting the resulting grids onto the original NURBS surfaces. After the first projection, the projected grids may need to be smoothed and projected again. The projection techniques described here can be used for structured grid generation as well.

In this study, the NURBS surfaces are first approximated by a set of smaller bilinear patches. Then, an advancing front technique is used to generate grids

*Computer Scientist, Computer Sciences Corporation, Hampton, Virginia 23666.

on these patches. And, finally these grids are projected back onto the NURBS surfaces. In the following sections, the techniques for projecting a point on a curve and on a surface are described. Then, a variation of the advancing front technique is presented. Finally results are summarized.

3 Projecting on a Curve

Curves are generally expressed by parametric splines such as NURBS as,

$$\vec{R}(u) = \{x(u), y(u), z(u)\}^T, \; u \in [a, b].$$

Points $\vec{R}(a)$ and $\vec{R}(b)$ are the beginning and the ending points of the curve, respectively. The variable u is a parameter which has no geometric significance. However, as u increases, the point $\vec{R}(u)$ always moves from the beginning to the end of the curve (Fig. 1). This property is referred to as monotone parameterization which is essential for curve and surface representations.

The process of projecting a point, $\vec{r}$, on a curve, $\vec{R}(u)$, can be achieved by finding a parameter u such that the distance between the point, $\vec{r}$, and the curve, $\vec{R}(u)$, is minimum and u is constrained to $u \in [a, b]$. This is a constrained minimization problem which can be written in terms of distance, d, as

$$d^2(u) = f(u) = |\vec{R}(u) - \vec{r}|^2. \qquad (1)$$

For curves, the constrained minimization problem is relatively simple to solve. A Newton-Raphson method is used in this study. In order to find the minimum distance, Eq. 1 must be differentiated with respect to u and set equal to zero, as

$$\frac{\partial f(u)}{\partial u} = \frac{\partial \vec{R}(u)}{\partial u} \cdot \{\vec{R}(u) - \vec{r}\} = 0. \qquad (2)$$

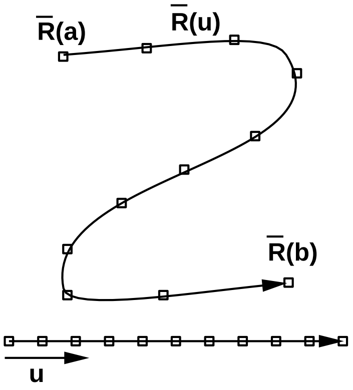

Figure 1: Parametric Curve

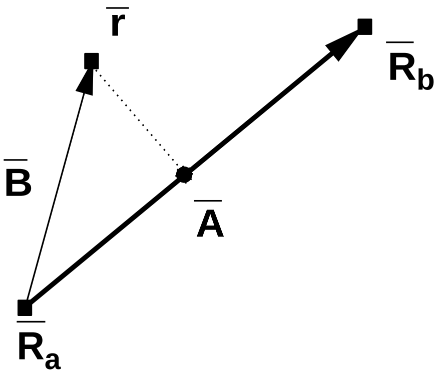

Figure 2: Projection on a Straight Line

Two test cases have been selected for this section: projecting a point on a straight line and on a NURBS curve.

In the first test case, a point is projected on a straight line as shown in Fig. 2. The line is expressed in terms of parameter u, as

$$\vec{R}(u) = (\vec{R}_b - \vec{R}_a)u + \vec{R}_a. \qquad (3)$$

The projected point, $\vec{A} = \vec{R}(u)$, can be determined by combining Eqs. 2-3,

$$u = \frac{(\vec{r} - \vec{R}_a) \cdot (\vec{R}_b - \vec{R}_a)}{(\vec{R}_b - \vec{R}_a) \cdot (\vec{R}_b - \vec{R}_a)}. \qquad (4)$$

In order to insure that the projected point is on the straight line segment, parameter u must be $\in [0, 1]$. Therefore, the parameter u (calculated from Eq. 4) must clipped as

$$u = \min\{u, 1\}, \qquad u = \max\{u, 0\}. \qquad (5)$$

The second test case is based on projecting on a NURBS curve. A NURBS curve can be expressed in terms of parameter u as,

$$\vec{R}(u) = \frac{\sum_{i=0}^{n} N_{i,p}(u) w_i \vec{P}_i}{\sum_{i=0}^{n} N_{i,p}(u) w_i} \qquad (6)$$

where $\vec{P}_i$ are control points (forming a control polygon), w_i are the weights, and $N_{i,p}$ are the p-th degree B-spline basis function defined on the non-periodic and nonuniform knot vector [2]. Combing Eqs. 2 and 6 yields a nonlinear equation which is solved by a Newton-Raphson method. Once the parameter u is determined, it has be be clipped as described by Eq. 5. This method requires an initial guess which can be obtained by sampling the curve, $\vec{R}(u)$, at several locations. This approach is very robust and efficient. It takes an average of five iterations to reduce the residual by five orders of magnitude. The projection process is shown in Fig. 3. Figure 4 shows th results of projecting points onto a NURBS curve. Th dashed-line shows the control polygon for the NURB curve. Weights for the first and last control points ar unity, and weights for the middle control points ar ten. The solid line shows the NURBS curve, the tri angle symbols show the points before the projection and the square symbols show the points after projec tion.

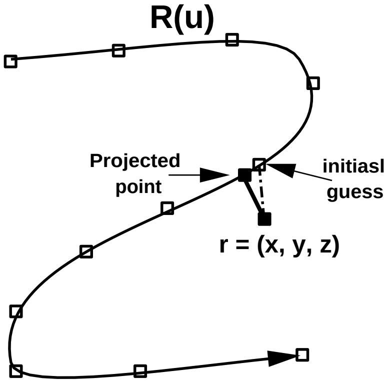

Figure 3: Projection Process for a Curve

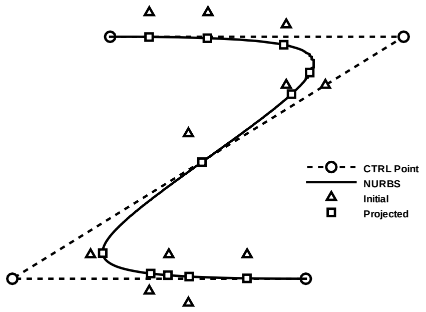

Figure 4: Projection on a NURBS Curve

4 Projecting on a Surface

Surfaces are generally approximated by parametric surfaces, e.g. NURBS, as,

$$\vec{R}(\vec{u}) = \{x(\vec{u}), y(\vec{u}), z(\vec{u})\}^T,$$

$$\vec{u} = \{u_1, u_2\}^T, \in [(a,b),(c,d)],$$

where $\vec{u}$ are the parameters which have no geometric significance. However, for a constant u_2, as u_1 increases, the point $\vec{R}(\vec{u})$ always moves from one side of the surface to the other side. The process of projecting a point, $\vec{r}$, on a surface, $\vec{R}(\vec{u})$, can be achieved by finding $\vec{u}$ such that distance, d, between the $\vec{r}$ and $\vec{R}(\vec{u})$ is minimum and $\vec{u}$ is constrained to be $\in [(a,b),(c,d)]$. Distance, d, can be written in terms parameter $\vec{u}$ as,

$$d^2(\vec{u}) = f(\vec{u}) = |\vec{R}(\vec{u}) - \vec{r}|^2. \tag{7}$$

In order to find the minimum distance, Eq. 7 must be minimized with respect to $\vec{u}$. This can be accomplished by setting the gradient of f, $\nabla f(\vec{u})$, equal to zero, as

$$\nabla f(\vec{u}) = G_i(\vec{u}) = \frac{\partial f(\vec{u})}{\partial u_i} = 0, \tag{8}$$

$$G_i(\vec{u}) = \frac{\partial \vec{R}(\vec{u})}{\partial u_i} \cdot \{\vec{R}(\vec{u}) - \vec{r}\}.$$

The above nonlinear system of equations must be solved for $\vec{u}$. Three test cases are discussed here: (1) projection on a three-dimensional triangle element, (2) projection on a bilinear patch, and (3) projection on a NURBS surface.

As shown in Fig. 6, a three-dimensional triangle element can be represented in terms of its parametric coordinates as

$$\vec{R}(\vec{u}) = \vec{R}_1 u_1 + \vec{R}_2 u_2 + \vec{R}_3 u_3,$$

$$u_1 + u_2 + u_3 = 1,$$

or

$$\vec{R}(\vec{u}) = (\vec{R}_1 - \vec{R}_3)u_1 + (\vec{R}_2 - \vec{R}_3)u_2 + \vec{R}_3. \tag{9}$$

Combining Eqs. 8-9 yields the following system of linear equations,

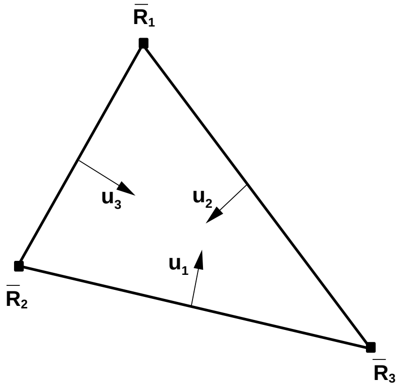

Figure 5: Parameter for a Triangular Element

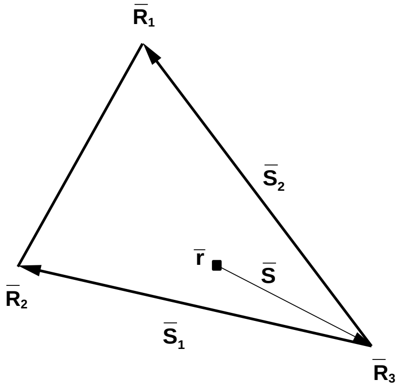

Figure 6: Projection on a Triangular Element

$$(\bar{S}_2 \cdot \bar{S}_2)u_1 + (\bar{S}_1 \cdot \bar{S}_2)u_2 = (\bar{S} \cdot \bar{S}_2)$$

$$(\bar{S}_2 \cdot \bar{S}_1)u_1 + (\bar{S}_1 \cdot \bar{S}_1)u_2 = (\bar{S} \cdot \bar{S}_1)$$

$$u_1 + u_2 + u_3 = 0.$$

For triangular elements, the above equations are solved for $\vec{u}$ (see Fig. 6 for definitions of S_1, S_2 and S). The projected point is inside the triangle if $0 \leq u_i \leq 1$ and $0 \leq u_1 + u_2 + u_3 \leq 1$, otherwise it is outside. If the projected point is outside of the triangular element, the parameters, u_i, must be clipped as described by Eq. 5.

The second example is for a bilinear surface, where the surface is approximated by a set of structured points. A bilinear surface can be decomposed into a

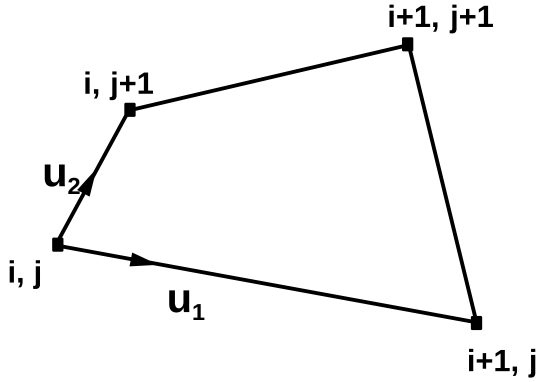

Figure 7: Parameter Space for a Bi-Linear Patch

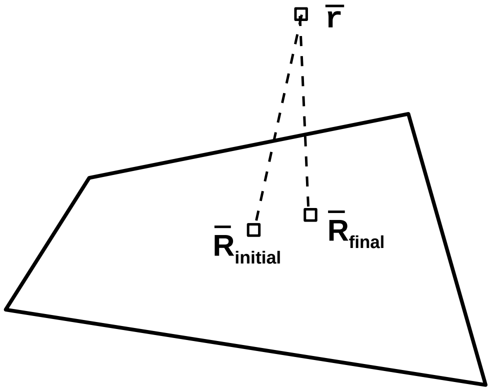

Figure 8: Projection Process for a surface

set of bilinear patches, and each patch, $\vec{R}(u_1, u_2)$, is approximated in terms of its parameters (see Fig. 7) as

$$\begin{aligned} \vec{R}(u_1, u_2) &= \\ (1-u_1)(1-u_2)\vec{R}_{i,j} &+ (1-u_1)u_2\vec{R}_{i,j+1} + \\ u_1(1-u_2)\vec{R}_{i+1,j} &+ u_1u_2\vec{R}_{i+1,j+1}. \qquad (10) \end{aligned}$$

Combining Eqs. 8 and 10 yields the following system of nonlinear equations,

$$\frac{\partial\vec{R}}{\partial u_1}\cdot\{\vec{R}(\vec{u})-\vec{r}\} = 0, \qquad \frac{\partial\vec{R}}{\partial u_2}\cdot\{\vec{R}(\vec{u})-\vec{r}\} = 0, \quad (11)$$

where,

$$\frac{\partial\vec{R}}{\partial u_1} = (1-u_2)(\vec{R}_{i+1,j}-\vec{R}_{i,j})+u_2(\vec{R}_{i+1,j+1}-\vec{R}_{i,j+1}),$$

$$\frac{\partial\vec{R}}{\partial u_2} = (1-u_1)(\vec{R}_{i,j+1}-\vec{R}_{i,j})+u_1(\vec{R}_{i+1,j+1}-\vec{R}_{i+1,j}).$$

Equation 11 is solved by Newton-Raphson method. It takes an average of 5 iterations to converge. The method requires an initial guess which is found by sampling the surface at various locations (see Fig. 8).

The third test case is projecting on a NURBS surface. A NURBS surface can be expressed as

$$\vec{R}(\vec{u}) = \frac{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u_1)N_{j,q}(u_2)w_i\vec{P}_{i,j}}{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u_1)N_{j,q}(u_2)w_i}. \quad (12)$$

Combining Eqs. 8 and 12 yields a system of nonlinear equations which is solved by the Newton-Raphson method.

5 Advancing Front Technique

Once the NURBS surfaces are decomposed into smaller patches (3/4-sided), the standard advancing front technique is used to generate the interior triangulation [3]. This triangulation is not on the actual NURBS surfaces, but it is close. Then, the resulting triangulation is projected on the actual NURBS surfaces as described in the previous sections. The VGRID system [3] has been used to generate the initial surface triangulation.

The spacing interpolation in the VGRID system is based on a structured background grid [4]. This technique simplifies the specifications of grid density by introducing nodal and linear sources. The contributions from nodal sources are inversely proportional to the square of the distance. A sample triangulation based on a nodal source is shown in Fig. 9. This is very similar to the Shepard method of interpolation [5]-[6]. However, the contributions from the linear sources are modeled similar to the diffusion which is not consistent with the nodal sources. The linear source integral-formulation in [4] becomes singular near the ends of the elements. Consequently, the spacing is more concentrated near the middle of the linear source which results in an asymmetric spacing distribution as shown in Fig. 10-11. In this study, the contributions from the linear sources are modified to resemble the nodal sources. In order to interpolate spacing for a point, $\vec{P}$, a point must be found on the linear source that is the closest. This can be accomplished by projecting the interpolation point, $\vec{P}$, onto the linear source as shown in Fig. 12. Equation 4 can be used to compute the projected point, $\bar{a}$. The associated spacing at point $\bar{a}$ can be calculated as

$$S_a = (S_2 - S_1)u + S_1$$

where S_1 and S_2 are the linear source spacings, and u is the unidirectional parameter associated with point $\bar{a}$. For point, $\vec{P}$, the contribution from the linear source can be treated as a nodal source at location at $\bar{a}$ with spacing S_a. Grids based on this formulation are shown in 13-14. This method can be easily extended to curve and surface sources.

6 Results and Discussions

Two test cases are projected in this study. The first case is a projection onto bilinear surfaces. Figure 15 shows the results of projecting an unstructured grid on bilinear surfaces. Figure. 16 shows the surface contours before (solid-line) and after (dash-line) projection. Figure 17 shows the projected surface grid for an X-15 configuration. This grid has been projected onto ten NURBS surfaces. Figure 18 shows the surface contours before (solid-line) and after (dash-line) projection. In both examples the projected grids were not distorted.

Using the projection technique described above, it is possible to generate structured and unstructured grids on CAD surfaces. The CAD surfaces could be overlapping as well.

References

[1] Samareh-Abolhassani, Jamshid, and Stewart, John E., "*Surface Grid Generation in a Parameter Space,* " AIAA-92-2717, 1992.

[2] Tiller, Wayne, "*Theory and Implementation of Non-Uniform Rational B-Spline Curves and Surfaces: class notes for NASA Langley,*" August 1991.

[3] Parikh, P. Pirzadeh, S., and Löhner, T., "*A Package for 3-D Unstructured Grid Generation Finite-Element Flow Solution and Flow Field Visualization,*" NASA CR-182090, September 1990.

[4] Pirzadeh, S.,"*Structured Background Grids for Generation of Unstructured Grids by Advancing Front Method,* " AIAA-91-3233, 1991.

[5] Shepard, D.,"*A Two Dimensional Interpolation Function for Irregular Spaced Data,* Proc. ACM Nat. Conf., pp. 517-524, 1965.

[6] Barnhill, R. E., "*Representation and Approximation of Surfaces,*" in: J. R. Rice, ed., Mathematical Software III, Academic Press, New York, 1977.

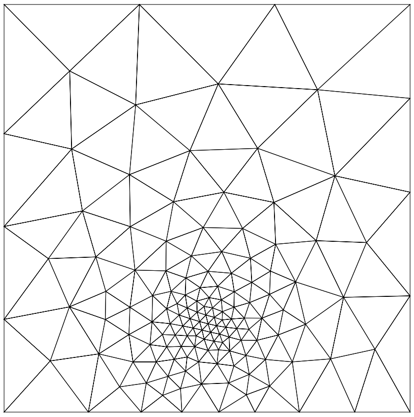

Figure 9: Grid Based on One Nodal Source

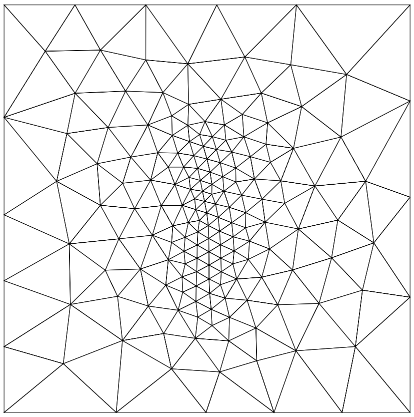

Figure 10: Grid Based on Old Linear Source

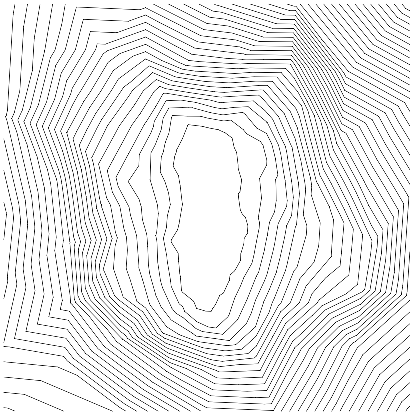

Figure 11: Area Contours for Old Linear Source

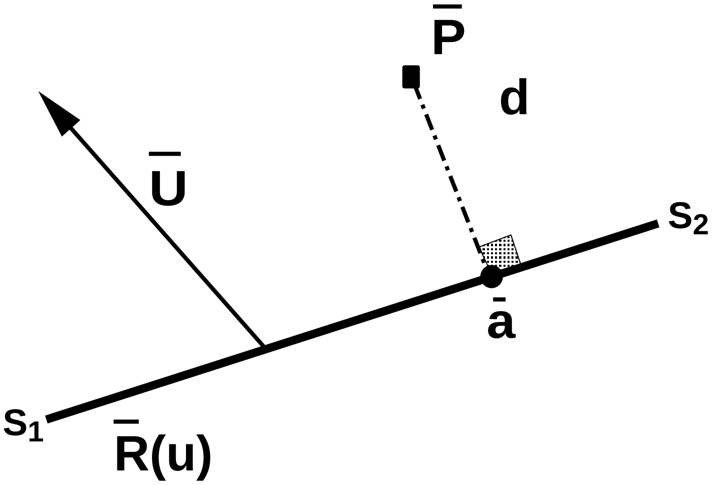

Figure 12: Linear Source

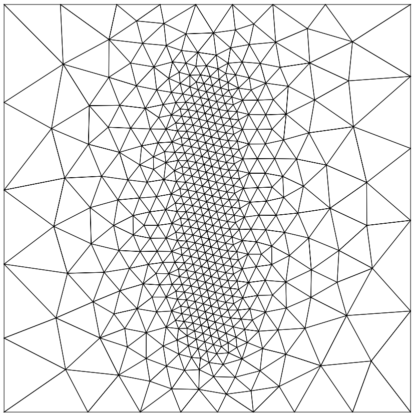

Figure 13: Grid Based on New linear Source

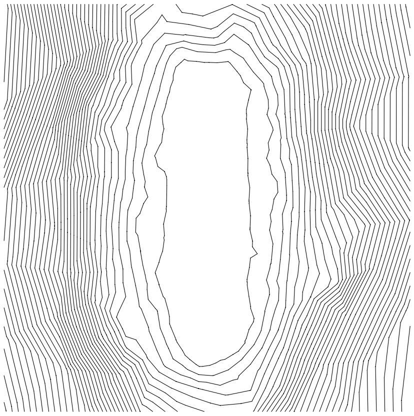

Figure 14: Area Contours for New Linear Source

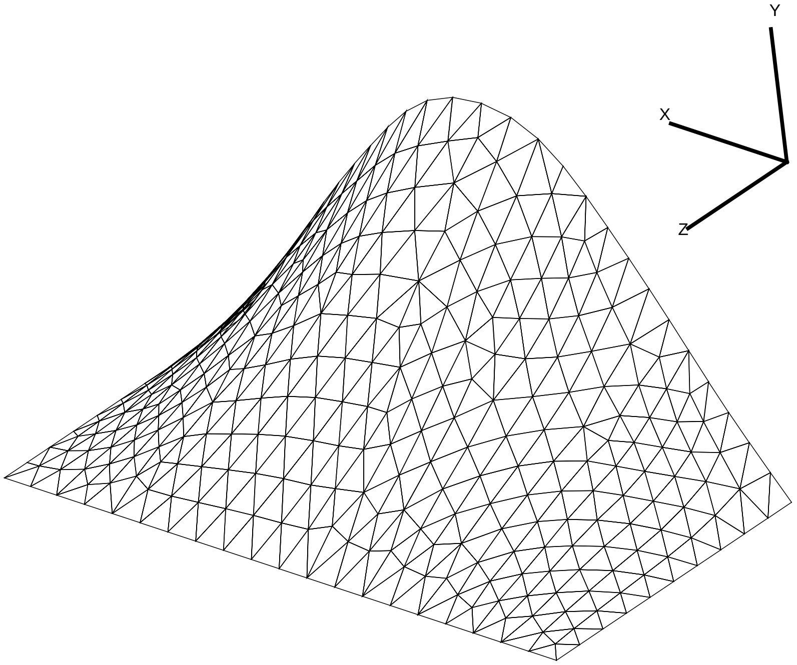

Figure 15: Projection for a Bilinear Surface

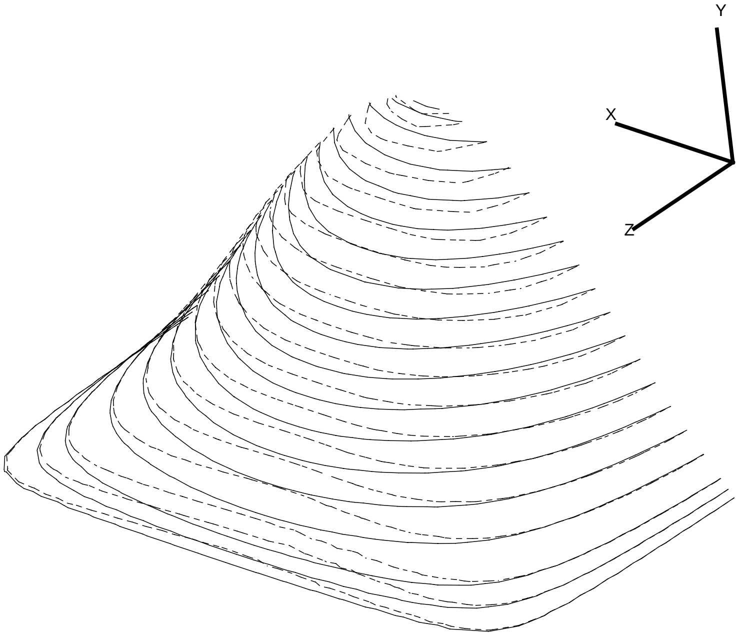

Figure 16: Projection for a Bilinear Patch

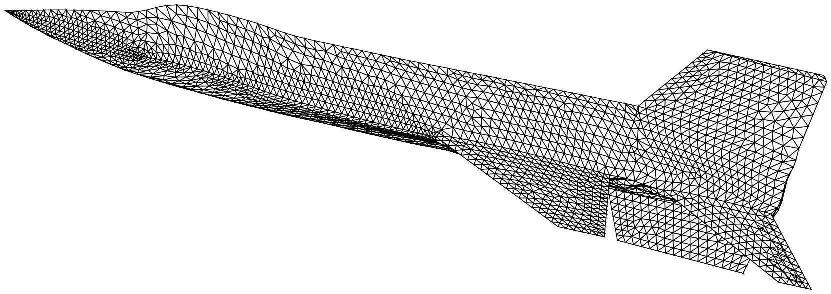

Figure 17: Triangulation for an X-15

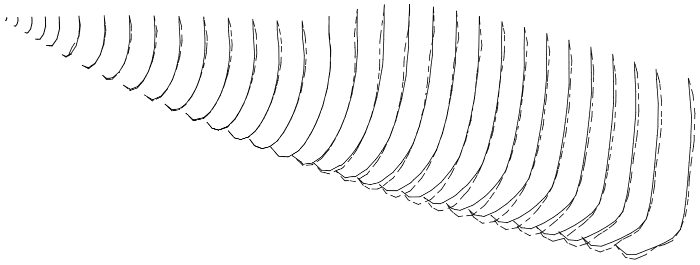

Figure 18: Surface Contours of an X-15