
A Direct Procedure for Interpolation on a Structured Curvilinear Two-Dimensional Grid

David W. Zingg, University of Toronto, Downsview, Ontario, Canada
Maurice Yarrow, Sterling Software, Ames Research Center, Moffett Field, California

July 1989



National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

Summary

A direct procedure is presented for locally bicubic interpolation on a structured, curvilinear, two-dimensional grid. The physical (Cartesian) space is transformed to a computational space in which the grid is uniform and rectangular by a generalized curvilinear coordinate transformation. Required partial derivative information is obtained by finite differences in the computational space. The partial derivatives in physical space are determined by repeated application of the chain rule for partial differentiation. A bilinear transformation is used to analytically transform the individual quadrilateral cells in physical space into unit squares. The interpolation is performed within each unit square using a piecewise bicubic spline.

1. Introduction

Problems in computational fluid dynamics (CFD) are often solved using a numerically generated grid which is structured but nonuniform, consisting of quadrilaterals in two dimensions and hexahedra in three dimensions. The grid is structured in that each cell contains an implied set of coordinate directions. A uniform rectangular grid in computational space is obtained from the physical space grid by a generalized curvilinear coordinate transformation (ref. 1). This facilitates the application of finite differences and finite volume methods.

In many applications of CFD, such as multiblock algorithms using overlapping grids, surface definition, and graphical representation of flow fields, data which are known at grid points must be interpolated at intermediate points. In the overlapping grid technique described by Benek et al. (ref. 2), interpolation is accomplished using an iterative technique. The purpose of the present paper is to present an alternative approach which involves no iteration.

The interpolation procedure presented uses a piecewise bicubic interpolant rather than the simpler bilinear interpolant. The bilinear interpolant has the often desirable property that it introduces no new extrema. However, the bicubic interpolant is more accurate and produces a smooth interpolant. Furthermore, bicubic interpolation is capable of preserving derivative information generated by a finite-difference flow solver. Important considerations regarding monotonicity and conservative properties of the interpolant are not addressed in this paper. Examples of alternative techniques for bivariate interpolation on a rectangular grid are given by Carlson and Fritsch (ref. 3) and Schiess (ref. 4).

The generalized curvilinear coordinate transformation is given in Section 2, and the iterative interpolation technique of Benek et al. is reviewed in Section 3. The direct interpolation procedure is presented in Section 4. The Appendix contains the formulas for bicubic spline interpolation on a unit square.

2. Generalized Curvilinear Coordinate Transformation

The generalized curvilinear coordinate transformation maps a nonuniform, nonrectangular grid in physical (Cartesian) space to a uniform, rectangular computational grid. The Cartesian coordinates (x, y) are mapped to the curvilinear coordinates (ξ, η) by the following general transformation:

$$\begin{aligned}\xi &= \xi(x, y) \\ \eta &= \eta(x, y),\end{aligned}\tag{1}$$

where we consider two space dimensions only and ignore time dependence for simplicity. The mapping is chosen such that the resulting grid in computational space is uniform, rectangular, and of unit length, i.e.,

$$\Delta\xi = \Delta\eta = 1.\tag{2}$$

Hence, finite-difference representations of ∂_ξ and ∂_η are easily formulated. The mapping is one-to-one except at topological singularities or cuts, where one physical point may be mapped to many computational points.

The chain rule for partial differentiation gives, in matrix form,

$$\begin{bmatrix} \partial_x \\ \partial_y \end{bmatrix} = \begin{bmatrix} \xi_x & \eta_x \\ \xi_y & \eta_y \end{bmatrix} \begin{bmatrix} \partial_\xi \\ \partial_\eta \end{bmatrix}.\tag{3}$$

The coordinate transformation is generally not known analytically and hence must be determined numerically. When the roles of the coordinate systems in equation (3) are reversed, the chain rule gives

$$\begin{bmatrix} \partial_\xi \\ \partial_\eta \end{bmatrix} = \begin{bmatrix} x_\xi & y_\xi \\ x_\eta & y_\eta \end{bmatrix} \begin{bmatrix} \partial_x \\ \partial_y \end{bmatrix}.\tag{4}$$

Therefore, we must have

$$\begin{bmatrix} \xi_x & \eta_x \\ \xi_y & \eta_y \end{bmatrix} = \begin{bmatrix} x_\xi & y_\xi \\ x_\eta & y_\eta \end{bmatrix}^{-1}\tag{5}$$

$$= J \begin{bmatrix} y_\eta & -y_\xi \\ -x_\eta & x_\xi \end{bmatrix},\tag{6}$$

where

$$J^{-1} = x_\xi y_\eta - x_\eta y_\xi.$$

All of the terms involving ∂_ξ and ∂_η are evaluated as finite differences. Equation (6) is the matrix form of the metric relations.

3. Iterative Interpolation Procedure

The iterative scheme developed by Benek et al. (ref. 2) is applicable to three-dimensional generalized curvilinear coordinates. Trilinear interpolation is used. Yarrow and Mehta (report to be published as a NASA Technical Memorandum) utilize tricubic interpolants within the same iterative scheme. In this section, the iterative procedure is presented for a two-dimensional space.

Since the grid is uniform and rectangular in computational space, the data can be readily interpolated as a function of the generalized coordinates by conventional multivariate interpolation techniques. The interpolant for the function f can be written as

$$f \simeq F(\xi, \eta). \quad (7)$$

The iterative procedure is used to determine the generalized coordinates (ξ^*, η^*) which correspond to the physical point (x^*, y^*) at which the interpolation is to be performed. Since each point in computational space is mapped to only one physical point, we can form the following interpolants for the Cartesian coordinates:

$$\begin{aligned} x &\simeq X(\xi, \eta) \\ y &\simeq Y(\xi, \eta). \end{aligned} \quad (8)$$

For a two-dimensional space, bivariate Newton-Raphson iteration is used to solve for (ξ^*, η^*) such that

$$\begin{bmatrix} x^* \\ y^* \end{bmatrix} = \begin{bmatrix} X(\xi^*, \eta^*) \\ Y(\xi^*, \eta^*) \end{bmatrix}. \quad (9)$$

The value of the function f at the point (x^*, y^*) is then determined from

$$f^* = F(\xi^*, \eta^*). \quad (10)$$

4. Direct Interpolation Procedure

The direct interpolation procedure proceeds in three steps. First, required partial derivative information in physical space is determined using finite-difference approximations to the partial derivatives in computational space. The individual cells are then analytically transformed into unit squares by a bilinear transformation. Finally, a bicubic spline interpolant is determined within each unit square.

For a function f known at the grid nodes, we require $f_x, f_y, f_{xx}, f_{xy}, f_{yy}$. Using finite differences in computational space, we can determine $f_\xi, f_\eta, f_{\xi\xi}, f_{\xi\eta}, f_{\eta\eta}$. Applying the chain rule to equation (3) yields

$$\begin{aligned}
\partial_{xx} &= \xi_{xx}\partial_\xi + \eta_{xx}\partial_\eta + \xi_x^2\partial_{\xi\xi} + 2\xi_x\eta_x\partial_{\xi\eta} + \eta_x^2\partial_{\eta\eta} \\
\partial_{xy} &= \xi_{xy}\partial_\xi + \eta_{xy}\partial_\eta + \xi_x\xi_y\partial_{\xi\xi} + (\xi_x\eta_y + \eta_x\xi_y)\partial_{\xi\eta} + \eta_x\eta_y\partial_{\eta\eta} \\
\partial_{yy} &= \xi_{yy}\partial_\xi + \eta_{yy}\partial_\eta + \xi_y^2\partial_{\xi\xi} + 2\xi_y\eta_y\partial_{\xi\eta} + \eta_y^2\partial_{\eta\eta}.
\end{aligned} \tag{11}$$

Equations (3) and (11) may be assembled as

$$\vec{\partial}_{xy} = B\vec{\partial}_{\xi\eta}, \tag{12}$$

where

$$B = \begin{bmatrix} \xi_x & \eta_x & 0 & 0 & 0 \\ \xi_y & \eta_y & 0 & 0 & 0 \\ \xi_{xx} & \eta_{xx} & \xi_x^2 & 2\xi_x\eta_x & \eta_x^2 \\ \xi_{xy} & \eta_{xy} & \xi_x\xi_y & \xi_x\eta_y + \eta_x\xi_y & \eta_x\eta_y \\ \xi_{yy} & \eta_{yy} & \xi_y^2 & 2\xi_y\eta_y & \eta_y^2 \end{bmatrix},$$

$$\vec{\partial}_{xy} = \begin{bmatrix} \partial_x \\ \partial_y \\ \partial_{xx} \\ \partial_{xy} \\ \partial_{yy} \end{bmatrix}, \quad \text{and} \quad \vec{\partial}_{\xi\eta} = \begin{bmatrix} \partial_\xi \\ \partial_\eta \\ \partial_{\xi\xi} \\ \partial_{\xi\eta} \\ \partial_{\eta\eta} \end{bmatrix}.$$

Similarly,

$$\vec{\partial}_{\xi\eta} = C\vec{\partial}_{xy}, \tag{13}$$

where

$$C = \begin{bmatrix} x_\xi & y_\xi & 0 & 0 & 0 \\ x_\eta & y_\eta & 0 & 0 & 0 \\ x_{\xi\xi} & y_{\xi\xi} & x_\xi^2 & 2x_\xi y_\xi & y_\xi^2 \\ x_{\xi\eta} & y_{\xi\eta} & x_\xi x_\eta & x_\xi y_\eta + y_\xi x_\eta & y_\xi y_\eta \\ x_{\eta\eta} & y_{\eta\eta} & x_\eta^2 & 2x_\eta y_\eta & y_\eta^2 \end{bmatrix}.$$

Therefore $B = C^{-1}$.

In block matrix form, B can be written as

$$B = \begin{bmatrix} B_1 & 0 \\ B_2 & B_3 \end{bmatrix}, \tag{14}$$

where

$$B_1 = \begin{bmatrix} \xi_x & \eta_x \\ \xi_y & \eta_y \end{bmatrix}, \quad B_2 = \begin{bmatrix} \xi_{xx} & \eta_{xx} \\ \xi_{xy} & \eta_{xy} \\ \xi_{yy} & \eta_{yy} \end{bmatrix}, \quad \text{and} \quad B_3 = \begin{bmatrix} \xi_x^2 & 2\xi_x\eta_x & \eta_x^2 \\ \xi_x\xi_y & \xi_x\eta_y + \eta_x\xi_y & \eta_x\eta_y \\ \xi_y^2 & 2\xi_y\eta_y & \eta_y^2 \end{bmatrix}.$$

Similarly,

$$C = \begin{bmatrix} C_1 & 0 \\ C_2 & C_3 \end{bmatrix}, \tag{15}$$

where

$$C_1 = \begin{bmatrix} x_\xi & y_\xi \\ x_\eta & y_\eta \end{bmatrix}, \quad C_2 = \begin{bmatrix} x_{\xi\xi} & y_{\xi\xi} \\ x_{\xi\eta} & y_{\xi\eta} \\ x_{\eta\eta} & y_{\eta\eta} \end{bmatrix}, \quad \text{and} \quad C_3 = \begin{bmatrix} x_\xi^2 & 2x_\xi y_\xi & y_\xi^2 \\ x_\xi x_\eta & x_\xi y_\eta + y_\xi x_\eta & y_\xi y_\eta \\ x_\eta^2 & 2x_\eta y_\eta & y_\eta^2 \end{bmatrix}.$$

Now $B = C^{-1}$ gives

$$B_1 = C_1^{-1} \quad (\text{the metric relations}), \quad (16)$$

$$B_3 = C_3^{-1}, \quad (17)$$

$$\text{and} \quad B_2 = -C_3^{-1} C_2 C_1^{-1}. \quad (18)$$

Substituting equations (16) and (17) into equation (18) gives

$$B_2 = -B_3 C_2 B_1. \quad (19)$$

Finally, use of the metric relations gives

$$\begin{aligned} \xi_{xx} &= -\frac{1}{J^3} [y_\eta^3 x_{\xi\xi} - 2y_\eta^2 y_\xi x_{\xi\eta} + y_\eta y_\xi^2 x_{\eta\eta} - x_\eta y_\eta^2 y_{\xi\xi} + 2x_\eta y_\eta y_\xi y_{\xi\eta} - x_\xi y_\xi^2 y_{\eta\eta}] \\ \eta_{xx} &= -\frac{1}{J^3} [-y_\xi y_\eta^2 x_{\xi\xi} + 2y_\xi^2 y_\eta - y_\xi^3 x_{\eta\eta} + x_\xi y_\eta^2 y_{\xi\xi} - 2x_\xi y_\eta y_\xi y_{\xi\eta} + x_\xi y_\xi^2 y_{\eta\eta}] \\ \xi_{xy} &= -\frac{1}{J^3} [-y_\eta^2 x_\eta x_{\xi\xi} + y_\eta (y_\eta x_\xi + x_\eta y_\xi) x_{\xi\eta} - y_\eta y_\xi x_\xi x_{\eta\eta} + x_\eta^2 y_\eta y_{\xi\xi} \\ &\quad - x_\eta (y_\eta x_\xi + x_\eta y_\xi) y_{\xi\eta} + x_\eta y_\xi x_\xi y_{\eta\eta}] \\ \eta_{xy} &= -\frac{1}{J^3} [y_\xi y_\eta x_\eta x_{\xi\xi} - y_\xi (y_\eta x_\xi + x_\eta y_\xi) x_{\xi\eta} + y_\xi^2 x_\xi x_{\eta\eta} - x_\xi y_\eta x_\eta y_{\xi\xi} \\ &\quad + x_\xi (y_\eta x_\xi + x_\eta y_\xi) y_{\xi\eta} - x_\xi^3 y_\xi y_{\eta\eta}] \\ \xi_{yy} &= -\frac{1}{J^3} [y_\eta x_\eta^2 x_{\xi\xi} - 2y_\eta x_\eta x_\xi x_{\xi\eta} + y_\eta x_\xi^2 x_{\eta\eta} - x_\eta^3 y_{\xi\xi} + 2x_\eta^2 x_\xi y_{\xi\eta} - x_\eta x_\xi^2 y_{\eta\eta}] \\ \eta_{yy} &= -\frac{1}{J^3} [-y_\xi x_\eta^2 x_{\xi\xi} + 2y_\xi x_\eta x_\xi x_{\xi\eta} - y_\xi x_\xi^2 x_{\eta\eta} + x_\xi x_\eta^2 y_{\xi\xi} - 2x_\eta x_\xi^2 y_{\xi\eta} + x_\xi^3 y_{\eta\eta}]. \end{aligned} \quad (20)$$

Therefore, all of the elements of matrix B can be expressed in terms of the elements of matrix C . As a result, $\vec{\partial}_{xy} f$ can be determined given $\vec{\partial}_{\xi\eta} f$. The elements of C and $\vec{\partial}_{\xi\eta} f$ can be approximated using finite differences. An alternative procedure is to calculate f_x and f_y given f_ξ and f_η from equation (3) and then to use finite differences to determine $(f_x)_\xi, (f_y)_\xi, (f_x)_\eta, (f_y)_\eta$. The required values of f_{xx}, f_{xy} , and f_{yy} can then be calculated using equation (3) again. The first method involves more computer storage but it also involves a smaller stencil.

Given $f_x, f_y, f_{xx}, f_{xy}, f_{yy}$ at the corners of a given cell, the problem remains to interpolate f on an arbitrary quadrilateral. We now consider the new two-dimensional space given by (p, q) shown in figure 1. The quadrilateral in physical space is related to a unit

square in (p, q) space by a bilinear mapping as follows. (Without loss of generality, we will assume the lower-left-hand corner to be at $(0, 0)$.)

$$x = ap + bq + cpq \quad (21a)$$

$$y = dp + eq + fpq \quad (21b)$$

Note that a different mapping is used for each quadrilateral, or cell, in the grid. This contrasts with the generalized curvilinear coordinate transformation in which a single smooth numerically generated mapping is applied to the entire grid. Consequently, analytical mappings can be utilized.

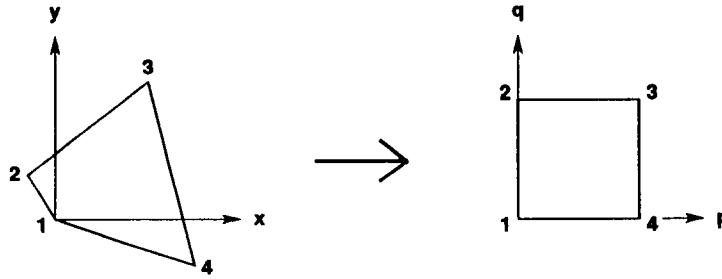


Figure 1. - Two-dimensional space given by (p, q) .

The coefficients of the bilinear mapping are given by

$$\begin{aligned} a &= x_4, & b &= x_2, & c &= x_3 - x_4 - x_2, \\ d &= y_4, & e &= y_2, & f &= y_3 - y_4 - y_2. \end{aligned} \quad (22)$$

The required derivatives in (p, q) space are given (as in equation (12)) by

$$\begin{aligned} \partial_p &= x_p \partial_x + y_p \partial_y \\ \partial_q &= x_q \partial_x + y_q \partial_y \\ \partial_{pq} &= x_{pq} \partial_x + y_{pq} \partial_y + x_p x_q \partial_{xx} + (x_p y_q + x_q y_p) \partial_{xy} + y_p y_q \partial_{yy}, \end{aligned} \quad (23)$$

where

$$x_p = a + cq, \quad y_p = d + fq, \quad x_q = b + cp, \quad y_q = e + fp, \quad x_{pq} = c, \quad y_{pq} = f,$$

and we have used $x_{pp} = x_{qq} = y_{pp} = y_{qq} = 0$.

Given a location in physical space (x_0, y_0) , we must find the corresponding (p_0, q_0) . From equation (21a), we have

$$q_0 = \frac{x_0 - ap_0}{b + cp_0}. \quad (24)$$

Substituting this into equation (21b) as follows:

$$y_0 = dp_0 + (e + fp_0)\left(\frac{x_0 - ap_0}{b + cp_0}\right) \quad (25)$$

gives the following quadratic for p_0

$$p_0^2(cd - af) + p_0(-cy_0 + bd + x_0f - ae) + (-y_0b + ex_0) = 0. \quad (26)$$

The solution of equation (26) gives two values of p_0 . The corresponding values of q_0 are determined from equation (24). One location (p_0, q_0) will not lie within the unit square and hence can be discarded.

Using $f_x, f_y, f_{xx}, f_{xy}, f_{yy}$ as calculated from equation (12), we can determine f_p, f_q , and f_{pq} from equation (23) at each corner of the unit square in (p, q) space. We then use the standard formula for bicubic interpolation within a unit square (see Appendix). The bicubic is finally evaluated at (p_0, q_0) , as calculated from equations (26) and (24), respectively.

5. Conclusions

A procedure has been presented for interpolation on a structured, curvilinear, two-dimensional grid. In contrast to existing methods (ref. 2), the procedure avoids the need for iteration. A piecewise bicubic spline is used, leading to a smooth and accurate interpolant which is capable of preserving derivative information generated by a flow solver based on finite differences. Potential applications include multiblock algorithms using overlapping grids in the solution of problems in computational fluid dynamics, surface definition, and graphical representation of data. Extension to three dimensions is straightforward but nontrivial.

Appendix

Locally Bicubic Spline Interpolation on a Unit Square

In this appendix, we give a piecewise bicubic spline formulation for interpolation within a unit square. De Boor (ref. 5) presents a technique for determining the required partial derivatives at the corners of each cell in a rectangular grid such that C^2 continuity is obtained for the domain. In the formulation used here, the required partial derivatives at the corners of the unit square are given by centered difference formulas, leading to C^1 continuity, analogous to the univariate cubic Bessel spline given by de Boor (ref. 6). This formulation has the advantage of producing a local interpolant.

The bicubic spline interpolant on a unit (p, q) square can be written as

$$f(p, q) \simeq \sum_{m,n=0}^3 \gamma_{mn} p^m q^n, \quad 0 \leq p, q \leq 1, \quad (A-1)$$

where

$$[\gamma_{mn}] = AKA^t,$$

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -3 & -2 & 3 & -1 \\ 2 & 1 & -2 & 1 \end{bmatrix},$$

and

$$K = \begin{bmatrix} f(0,0) & f_q(0,0) & f(0,1) & f_q(0,1) \\ f_p(0,0) & f_{pq}(0,0) & f_p(0,1) & f_{pq}(0,1) \\ f(1,0) & f_q(1,0) & f(1,1) & f_q(1,1) \\ f_p(1,0) & f_{pq}(1,0) & f_p(1,1) & f_{pq}(1,1) \end{bmatrix}.$$

References

1. Pulliam, T. H.: Efficient Solution Methods for the Navier-Stokes Equations. Lecture Notes for the Von Karman Institute for Fluid Dynamics Lecture Series: Numerical Techniques for Viscous Flow Computation in Turbomachinery Bladings, Brussels, Belgium, Jan. 1986.
2. Benek, J. A.; Buning, P. G.; and Steger, J. L.: A 3-D Chimera Grid Embedding Technique. AIAA Paper 85-1523, June 1985.
3. Carlson, R. E.; and Fritsch, F. N.: Monotone Piecewise Bicubic Interpolation. SIAM J. Numer. Anal., vol. 22, no. 2, April 1985.
4. Schiess, J. R.: Two Algorithms for Rational Spline Interpolation of Surfaces. NASA TP 2536, 1986.
5. De Boor, C.: Bicubic Spline Interpolation. J. Math. and Phys., vol. XLI, no. 3, Sept. 1962.
6. De Boor, C.: A Practical Guide to Splines. Springer-Verlag, New York, 1978.



Report Documentation Page

| | | | | | |
|---|--|--|---|--|------------------|
| 1. Report No. NASA TM-102213 | | 2. Government Accession No. | | 3. Recipient's Catalog No. | |
| 4. Title and Subtitle A Direct Procedure for Interpolation on a Structured Curvilinear Two-Dimensional Grid | | | | 5. Report Date July 1989 | |
| | | | | 6. Performing Organization Code | |
| 7. Author(s) David W. Zingg* and Maurice Yarrow** | | | | 8. Performing Organization Report No. A-89210 | |
| | | | | 10. Work Unit No. 505-60 | |
| 9. Performing Organization Name and Address Ames Research Center Moffett Field, CA 94035 | | | | 11. Contract or Grant No. | |
| | | | | 13. Type of Report and Period Covered Technical Memorandum | |
| 12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546-0001 | | | | 14. Sponsoring Agency Code | |
| | | | | 15. Supplementary Notes Point of Contact: Harvard Lomax, Ames Research Center, MS 202A-1, Moffett Field, CA 94035 (415) 694-5124 or FTS 464-5124 *University of Toronto, Downsview, Ontario, Canada M3H 5T6 **Sterling Software, Ames Research Center, Moffett Field, CA 94035 | |
| 16. Abstract <p>A direct procedure is presented for locally bicubic interpolation on a structured, curvilinear, two-dimensional grid. The physical (Cartesian) space is transformed to a computational space in which the grid is uniform and rectangular by a generalized curvilinear coordinate transformation. Required partial derivative information is obtained by finite differences in the computational space. The partial derivatives in physical space are determined by repeated application of the chain rule for partial differentiation. A bilinear transformation is used to analytically transform the individual quadrilateral cells in physical space into unit squares. The interpolation is performed within each unit square using a piecewise bicubic spline.</p> | | | | | |
| 17. Key Words (Suggested by Author(s)) Interpolation Bicubic interpolation Generalized curvilinear coordinates Structured grids | | | 18. Distribution Statement Unclassified - Unlimited Subject Category - 64 | | |
| 19. Security Classif. (of this report) Unclassified | | 20. Security Classif. (of this page) Unclassified | | 21. No. of Pages 10 | 22. Price A02 |