Chapter 5
Algebraic Grid Generation

5.1 Introduction

The algebraic grid generation approach relies chiefly on an explicit construction of coordinate transformations through the formulas of transfinite interpolation. Of central importance in the method are blending functions (univariate quantities, each depending on one chosen coordinate only). These provide matching of the grid distribution on, and grid directions from, the boundaries and specified interior surfaces of an arbitrary domain. Direct control of the essential properties of the coordinate transformations in the vicinity of the boundaries and interior surfaces is carried out by the specification of the out-of-surface-direction derivatives and blending functions.

The purpose of this chapter is to describe common techniques of algebraic grid generation.

Nearly all of the formulas of transfinite interpolation include both repeated indices over which a summation is carried out and one repeated index, usually i, that is fixed. Therefore, in this chapter, we do not use the convention of summation of repeated indices, but instead use the common notation $\sum$ to indicate summation.

5.2 Transfinite Interpolation

This section describes some general three-dimensional formulas of transfinite interpolation which are used to define algebraic coordinate transformations from a standard three-dimensional cube Ξ^3 with Cartesian coordinates ξ^i, $i = 1, 2, 3$, onto a physical domain X^3 with Cartesian coordinates x^i, $i = 1, 2, 3$. The formulation of the three–dimensional interpolation is based on a particular operation of Boolean summation over unidirectional interpolations. So, first, the general formulas of unidirectional interpolation are reviewed.

V.D. Liseikin, *Grid Generation Methods*, Scientific Computation,
DOI 10.1007/978-3-319-57846-0_5

5.2.1 Unidirectional Interpolation

General Formulas

For the unit cube Ξ^3, let there be chosen one coordinate direction ξ^i and some sections of the cube orthogonal to this direction, defined by the planes $\xi^i = \xi^i_l$, $l = 1, \ldots, L_i$. Furthermore, on each section $\xi^i = \xi^i_l$, let there be given the values of some vector-valued function $\boldsymbol{r}(\boldsymbol{\xi})$, $\boldsymbol{\xi} = (\xi^1, \xi^2, \xi^3)$, and of its derivatives with respect to ξ^i up to order P^i_l. Then, the unidirectional interpolation of the function $\boldsymbol{r}(\boldsymbol{\xi})$ is a vector-valued function $\boldsymbol{P}^i[\boldsymbol{r}](\boldsymbol{\xi})$ from Ξ^3 into R^3 defined by the formula

$$\boldsymbol{P}_i[\boldsymbol{r}](\boldsymbol{\xi}) = \sum_{l=1}^{L^i} \sum_{n=0}^{P^i_l} \alpha^i_{l,n}(\xi^i) \frac{\partial^n}{(\partial \xi^i)^n} \boldsymbol{r}(\boldsymbol{\xi}|_{\xi^i = \xi^i_l}) \ . \tag{5.1}$$

Here, the smooth scalar functions $\alpha^i_{l,n}(\xi^i)$, depending on one independent variable ξ^i, are subject to the following restrictions:

$$\frac{\mathrm{d}^m}{(\mathrm{d}\xi^i)^m} \alpha^i_{l,n}(\xi^i_k) = \delta^l_k \delta^n_m \ , \qquad l, k = 1, \ldots, L^i \ , \qquad m, n = 0, 1, \ldots P^i_l \ , \tag{5.2}$$

where δ^j_i is the Kronecker delta function, i.e. $\delta^j_i = \begin{cases} 1, & i = j, \\ 0, & i \neq j. \end{cases}$

The expression $(\boldsymbol{\xi}|_{\xi^i = \xi^i_l})$ in (5.1) designates a point that is a projection of $\boldsymbol{\xi} = (\xi^1, \xi^2, \xi^3)$ on the section $\xi^i = \xi^i_l$, i.e. the ith coordinate ξ^i of $\boldsymbol{\xi}$ is fixed and equal to ξ^i_l; for example,

$$(\boldsymbol{\xi}|_{\xi^1 = \xi^1_l}) = (\xi^1_l, \xi^2, \xi^3) \ .$$

It is also assumed in (5.1) and below that the operator for the zero-order derivative is the identity operator, i.e.

$$\frac{\partial^0}{(\partial \xi^i)^0} f(\boldsymbol{\xi}) = f(\boldsymbol{\xi}) \ , \qquad \frac{\mathrm{d}^0}{(\mathrm{d}\xi^i)^0} g(\xi^i) = g(\xi^i) \ .$$

The coefficients $\alpha^i_{l,n}(\xi^i)$ in (5.1) are referred to as the blending functions. They serve to propagate the values of the vector-valued function $\boldsymbol{r}(\boldsymbol{\xi})$ from the specified sections of the cube Ξ^3 into its interior. It is easily shown that the conditions (5.2) imposed on the blending functions $\alpha^i_{l,n}(\xi^i)$ provide matching at the sections $\xi^i = \xi^i_l$ of the values of the function $\boldsymbol{P}_i[\boldsymbol{r}](\boldsymbol{\xi})$ and $\boldsymbol{r}(\boldsymbol{\xi})$, as well as the values of their derivatives with respect to ξ^i, namely,

$$\frac{\partial^n \boldsymbol{P}_i[\boldsymbol{r}]}{(\partial \boldsymbol{\xi}^i)^n}(\boldsymbol{\xi}|_{\xi^i = \xi^i_l}) = \frac{\partial^n \boldsymbol{r}}{(\partial \xi^i)^n}(\boldsymbol{\xi}|_{\xi^i = \xi^i_l}) \ , \qquad n = 0, \ldots, P^i_l \ .$$

Two-Boundary Interpolation

A very important interpolation for grid generation applications is the one which matches the values of the vector-valued function $\boldsymbol{r}(\boldsymbol{\xi})$ and of its derivatives exclusively at the boundary planes of the cube Ξ^3. In this case, $L^i = 2$, $\xi^i_1 = 0$, and $\xi^i_2 = 1$, and the relations (5.1) and (5.2) have the form

$$\begin{aligned}\boldsymbol{P}_i[\boldsymbol{r}](\boldsymbol{\xi}) = &\sum_{n=0}^{P^i_1} \alpha^1_{1,n}(\xi^i)\frac{\partial^n}{(\partial\xi^i)^n}\boldsymbol{r}(\boldsymbol{\xi}|_{\xi^i=0}) \\ &+ \sum_{n=0}^{P^i_2} \alpha^i_{2,n}(\xi^i)\frac{\partial^n}{(\partial\xi^i)^n}\boldsymbol{r}(\boldsymbol{\xi}|_{\xi^i=1}) \ ,\end{aligned} \tag{5.3}$$

$$\frac{\mathrm{d}^m}{(\mathrm{d}\xi^i)^m}\alpha^i_{l,n}(\xi^i_k) = \delta^l_k\delta^n_m \ , \qquad l, k = 1, 2 \ , \qquad m, n = 0, 1, \dots P^i_l \ . \tag{5.4}$$

The interpolation described by (5.3) is referred to as the two-boundary interpolation.

5.2.2 *Tensor Product*

The composition of two unidirectional interpolations $\boldsymbol{P}_i[\boldsymbol{r}](\boldsymbol{\xi})$ and $\boldsymbol{P}_j[\boldsymbol{r}](\boldsymbol{\xi})$ of $\boldsymbol{r}(\boldsymbol{\xi})$ in the directions ξ^i and ξ^j, respectively, is called their tensor product. This operation is denoted by $\boldsymbol{P}_i[r] \otimes \boldsymbol{P}_j[r](\xi)$ and, in accordance with (5.1), we obtain

$$\begin{aligned}\boldsymbol{P}_i[\boldsymbol{r}] \otimes \boldsymbol{P}_j[\boldsymbol{r}](\boldsymbol{\xi}) &= \boldsymbol{P}_i\big[\boldsymbol{P}_j[\boldsymbol{r}]\big](\boldsymbol{\xi}) \\ &= \sum_{l=1}^{L^i}\sum_{n=0}^{P^i_l} \alpha^i_{l,n}(\xi^i)\frac{\partial^n \boldsymbol{P}_j[\boldsymbol{r}]}{(\partial\xi^i)^n}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l}) \\ &= \sum_{k=1}^{L^j}\sum_{m=0}^{P^j_k}\sum_{l=1}^{L^i}\sum_{n=0}^{P^i_l} \alpha^i_{l,n}(\xi^i)\alpha^j_{k,m}(\xi^j)\frac{\partial^{n+m}\boldsymbol{r}}{(\partial\xi^i)^n(\partial\xi^j)^m}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l,\xi^j=\xi^j_k}) \ .\end{aligned} \tag{5.5}$$

Here, by the notation $(\boldsymbol{\xi}|_{\xi^i=\xi^i_l,\xi^j=\xi^j_k})$, we mean the point which is the projection of $\boldsymbol{\xi}$ on the intersection of the planes $\xi^i = \xi^i_l$ and $\xi^j = \xi^j_k$, e.g.

$$(\boldsymbol{\xi}|_{\xi^1=\xi^1_l,\xi^3=\xi^3_k}) = (\xi^1_l, \xi^2, \xi^3_k) \ .$$

Equation (5.5) shows clearly that the tensor product is a commutative operation, i.e.

$$\boldsymbol{P}_i[\boldsymbol{r}] \otimes \boldsymbol{P}_j[\boldsymbol{r}] = \boldsymbol{P}_j[\boldsymbol{r}] \otimes \boldsymbol{P}_i[\boldsymbol{r}] \ .$$

Using the relations (5.1), (5.2), and (5.5), we obtain

$$\begin{aligned}
&\frac{\partial}{\partial \xi^i} \boldsymbol{P}_i[\boldsymbol{r}] \otimes \boldsymbol{P}_j[\boldsymbol{r}](\boldsymbol{\xi}|_{\xi^i=\xi^i_s,\xi^j=\xi^j_t}) \\
&= \sum_{m=1}^{L^j} \sum_{k=0}^{P^j_k} \sum_{l=1}^{L^i} \sum_{p=0}^{P^i_l} \frac{\mathrm{d}}{\mathrm{d}_{\xi^i}} \alpha^i_{m,k}(\xi^i_s) \alpha^j_{l,p}(\xi^j_t) \frac{\partial^{k+p} \boldsymbol{r}}{(\partial \xi^i)^k (\partial \xi^j)^p} (\boldsymbol{\xi}|_{\xi^i=\xi^i_m,\xi^j=\xi^j_l}) \\
&= \frac{\partial \boldsymbol{r}}{\partial \xi^i} (\boldsymbol{\xi}|_{\xi^i=\xi^i_s,\xi^j=\xi^j_t}) \ .
\end{aligned}$$

Analogously,

$$\frac{\partial^{k+p}}{(\partial \xi^i)^k (\partial \xi^j)^p} (\boldsymbol{P}_i[\boldsymbol{r}] \otimes \boldsymbol{P}_j[\boldsymbol{r}])(\boldsymbol{\xi}|_{\xi^i=\xi^i_s,\xi^j=\xi^j_t}) = \frac{\partial^{k+p}}{(\partial \xi^i)^k (\partial \xi^j)^p} \boldsymbol{r}(\boldsymbol{\xi}|_{\xi^i=\xi^i_s,\xi^j=\xi^j_t}) \ .$$

Thus, the derivatives of the tensor product $\boldsymbol{P}_i[\boldsymbol{r}] \otimes \boldsymbol{P}_j[\boldsymbol{r}]$ with respect to ξ^i and ξ^j match the derivatives of the function $\boldsymbol{r}(\boldsymbol{\xi})$ at the intersections of the planes $\xi^i = \xi^i_s$ and $\xi^j = \xi^j_t$.

5.2.3 Boolean Summation

Bidirectional Interpolation

The bidirectional interpolation matching the values of the function $\boldsymbol{r}(\boldsymbol{\xi})$ and of its derivatives at the sections in the directions ξ^i and ξ^j is defined through the Boolean summation $\oplus$:

$$\boldsymbol{P}_i[\boldsymbol{r}] \oplus \boldsymbol{P}_j[\boldsymbol{r}](\boldsymbol{\xi}) = \boldsymbol{P}_i[\boldsymbol{r}](\boldsymbol{\xi}) + \boldsymbol{P}_j[\boldsymbol{r}](\boldsymbol{\xi}) - \boldsymbol{P}_i[\boldsymbol{r}] \otimes \boldsymbol{P}_j[\boldsymbol{r}](\boldsymbol{\xi}) \ . \tag{5.6}$$

Using (5.1) and (5.5), we obtain

$$\begin{aligned}
\boldsymbol{P}_i[\boldsymbol{r}] \oplus \boldsymbol{P}_j[\boldsymbol{r}](\boldsymbol{\xi}) = &\sum_{l=1}^{L^i} \sum_{n=0}^{P^i_l} \alpha^i_{l,n}(\xi^i) \frac{\partial^n \boldsymbol{r}}{(\partial \xi^i)^n} (\boldsymbol{\xi}|_{\xi^i=\xi^i_l}) \\
&+ \sum_{k=1}^{L^j} \sum_{m=0}^{P^j_k} \alpha^j_{k,m}(\xi^i) \frac{\partial^m \boldsymbol{r}}{(\partial \xi^j)^m} (\boldsymbol{\xi}|_{\xi^j=\xi^j_k}) \\
- \sum_{k=1}^{L^j} \sum_{m=0}^{P^j_k} \sum_{l=1}^{L^i} \sum_{n=0}^{P^i_l} &\alpha^i_{l,n}(\xi^i) \alpha^j_{k,m}(\xi^j) \frac{\partial^{n+m} \boldsymbol{r}}{(\partial \xi^i)^n (\partial \xi^j)^m} (\boldsymbol{\xi}|_{\xi^i=\xi^i_l,\xi^j=\xi^j_k}) \ .
\end{aligned} \tag{5.7}$$

Taking into account the relation

$$\boldsymbol{P}_j[\boldsymbol{r}] - \boldsymbol{P}_i[\boldsymbol{r}] \otimes \boldsymbol{P}_j[\boldsymbol{r}] = \boldsymbol{P}_j\big[\boldsymbol{r} - \boldsymbol{P}_i[\boldsymbol{r}]\big] \tag{5.8}$$

we obtain the result that the formulas (5.6) and (5.7) for the Boolean summation can be written as the ordinary sum of two unidirectional interpolants $\boldsymbol{P}_i[\boldsymbol{r}]$ and $\boldsymbol{P}_j\big[\boldsymbol{r} - \boldsymbol{P}_i[\boldsymbol{r}]\big]$. Thus, using (5.1), we obtain

$$\boldsymbol{P}_i[\boldsymbol{r}] \oplus \boldsymbol{P}_j[\boldsymbol{r}](\boldsymbol{\xi}) = \boldsymbol{P}_j[\boldsymbol{r}](\boldsymbol{\xi}) + \sum_{l=1}^{L^i} \sum_{n=0}^{P_l^i} \alpha_{l,n}^i(\xi^i) \frac{\partial^n (\boldsymbol{r} - \boldsymbol{P}_j[\boldsymbol{r}])}{(\partial \xi^i)^n}(\boldsymbol{\xi}|_{\xi^i = \xi_l^i}) \,. \tag{5.9}$$

From (5.7), it is evident that

$$\boldsymbol{P}_i[\boldsymbol{r}] \oplus \boldsymbol{P}_j[\boldsymbol{r}] = \boldsymbol{P}_j[\boldsymbol{r}] \oplus \boldsymbol{P}_i[\boldsymbol{r}] \,,$$

so the indices i and j in (5.7), (5.9) can be interchanged.

The Boolean summation (5.6) matches $\boldsymbol{r}(\boldsymbol{\xi})$ and its derivatives at all sections $\xi^i = \xi_k^i$ and $\xi^j = \xi_l^j$, i.e.

$$\frac{\partial^{k+p}}{(\partial \xi^i)^k (\partial \xi^j)^p} (\boldsymbol{P}_i[\boldsymbol{r}] \otimes \boldsymbol{P}_j[\boldsymbol{r}])(\boldsymbol{\xi}|_{\xi^t = \xi_l^t}) = \frac{\partial^{k+p}}{(\partial \xi^i)^k (\partial \xi^j)^p} \boldsymbol{r}(\boldsymbol{\xi}|_{\xi^t = \xi_l^t}) \,,$$

where either $t = i$ or $t = j$.

Three-Dimensional Interpolation

A multidirectional interpolation $\boldsymbol{P}[\boldsymbol{r}](\boldsymbol{\xi})$ of $\boldsymbol{r}(\boldsymbol{\xi})$, which matches the values of the function $\boldsymbol{r}(\boldsymbol{\xi})$ and of its derivatives at the sections $\xi^i = \xi_l^i, l = 1, \dots, L_i$, in all directions ξ^i, $i = 1, 2, 3$, is defined through the Boolean summation of all unidirectional interpolations $\boldsymbol{P}_i[\boldsymbol{r}]$, $i = 1.2, 3$:

$$\boldsymbol{P}[\boldsymbol{r}] = \boldsymbol{P}_1[\boldsymbol{r}] \oplus \boldsymbol{P}_2[\boldsymbol{r}] \oplus \boldsymbol{P}_3[\boldsymbol{r}] \,. \tag{5.10}$$

Taking into account (5.6), we obtain

$$\begin{aligned} \boldsymbol{P}[\boldsymbol{r}] = {} & \boldsymbol{P}_1[\boldsymbol{r}] + \boldsymbol{P}_2[\boldsymbol{r}] + \boldsymbol{P}_3[\boldsymbol{r}] \\ & - \boldsymbol{P}_1[\boldsymbol{r}] \otimes \boldsymbol{P}_2[\boldsymbol{r}] - \boldsymbol{P}_1[\boldsymbol{r}] \otimes \boldsymbol{P}_3[\boldsymbol{r}] - \boldsymbol{P}_2[\boldsymbol{r}] \otimes \boldsymbol{P}_3[\boldsymbol{r}] \\ & + \boldsymbol{P}_1[\boldsymbol{r}] \otimes \boldsymbol{P}_2[\boldsymbol{r}] \otimes \boldsymbol{P}_3[\boldsymbol{r}] \,. \end{aligned} \tag{5.11}$$

Recursive Form of Transfinite Interpolation

Using the relation (5.8), we can easily show that (5.11) is also equal to the following equation:

$$\boldsymbol{P}[\boldsymbol{r}] = \boldsymbol{P}_1[\boldsymbol{r}] + \boldsymbol{P}_2\big[\boldsymbol{r} - \boldsymbol{P}_1[\boldsymbol{r}]\big] + \boldsymbol{P}_3\Big[\boldsymbol{r} - \boldsymbol{P}_1[\boldsymbol{r}] - \boldsymbol{P}_2\big[\boldsymbol{r} - \boldsymbol{P}_1[\boldsymbol{r}]\big]\Big] \,. \tag{5.12}$$

This represents the formula (5.10) for multidirectional interpolation as the sum of the three unidirectional interpolations $\boldsymbol{P}_1[\boldsymbol{r}]$, $\boldsymbol{P}_2\big[\boldsymbol{r}-\boldsymbol{P}_1[\boldsymbol{r}]\big]$, and $\boldsymbol{P}_3\Big[\boldsymbol{r}-\boldsymbol{P}_1[\boldsymbol{r}]-\boldsymbol{P}_2\big[\boldsymbol{r}-\boldsymbol{P}_1[\boldsymbol{r}]\big]\Big]$. Therefore, the expression (5.12) for $\boldsymbol{P}[\boldsymbol{r}]$ gives a recursive form of the interpolation (5.10) through a sequence of the unidirectional interpolations (5.1):

$$\begin{aligned}
\boldsymbol{F}_1[\boldsymbol{r}] &= \boldsymbol{P}_1[\boldsymbol{r}] \ , \\
\boldsymbol{F}_2[\boldsymbol{r}] &= \boldsymbol{F}_1[\boldsymbol{r}] + \boldsymbol{P}_2\big[\boldsymbol{r}-\boldsymbol{F}_1[\boldsymbol{r}]\big] \ , \\
\boldsymbol{P}[\boldsymbol{r}] &= \boldsymbol{F}_2[\boldsymbol{r}] + \boldsymbol{P}_3\big[\boldsymbol{r}-\boldsymbol{F}_2[\boldsymbol{r}]\big]
\end{aligned}$$

which is usually applied in constructing algebraic coordinate transformations. Using (5.1), we obtain

$$\begin{aligned}
\boldsymbol{F}_1[\boldsymbol{r}](\boldsymbol{\xi}) &= \sum_{l=1}^{L^1}\sum_{n=0}^{P_l^1} \alpha_{l,n}^1(\xi^1)\frac{\partial^n \boldsymbol{r}}{(\partial\xi^1)^n}(\xi_l^1,\xi^2,\xi^3) \ , \\
\boldsymbol{F}_2[\boldsymbol{r}](\boldsymbol{\xi}) &= \boldsymbol{F}_1[\boldsymbol{r}](\boldsymbol{\xi}) + \sum_{l=1}^{L^2}\sum_{n=0}^{P_l^2} \alpha_{l,n}^2(\xi^2)\frac{\partial^n (\boldsymbol{r}-\boldsymbol{F}_1[\boldsymbol{r}])}{(\partial\xi^2)^n}(\xi^1,\xi_l^2,\xi^3) \ , \\
\boldsymbol{P}[\boldsymbol{r}](\boldsymbol{\xi}) &= \boldsymbol{F}_2[\boldsymbol{r}](\boldsymbol{\xi}) + \sum_{l=1}^{L^3}\sum_{n=0}^{P_l^3} \alpha_{l,n}^3(\xi^3)\frac{\partial^n (\boldsymbol{r}-\boldsymbol{F}_2[\boldsymbol{r}])}{(\partial\xi^3)^n}(\xi^1,\xi^2,\xi_l^3) \ .
\end{aligned} \tag{5.13}$$

It is easy to see, taking advantage of (5.2), that the multiple summation matches the function $\boldsymbol{r}(\boldsymbol{\xi})$ and its derivatives with respect to ξ^1, ξ^2, and ξ^3 on all sections $\xi^i=\xi_l^i$, $i=1,2,3$, of the cube Ξ^3.

Outer Boundary Interpolation

Equation (5.13) shows that the outer boundary interpolation based on the two-boundary unidirectional interpolations described by (5.4) has the following form:

$$\begin{aligned}
\boldsymbol{F}_1[\boldsymbol{r}](\boldsymbol{\xi}) &= \sum_{n=0}^{P_1^1} \alpha_{1,n}^1(\xi^1)\frac{\partial^n \boldsymbol{r}}{(\partial\xi^1)^n}(0,\xi^2,\xi^3) \\
&\quad + \sum_{n=0}^{P_2^1} \alpha_{2,n}^1(\xi^1)\frac{\partial^n \boldsymbol{r}}{(\partial\xi^1)^n}(1,\xi^2,\xi^3) \ , \\
\boldsymbol{F}_2[\boldsymbol{r}](\boldsymbol{\xi}) &= \boldsymbol{F}_1[\boldsymbol{r}](\boldsymbol{\xi}) + \sum_{n=0}^{P_1^2} \alpha_{1,n}^2(\xi^2)\frac{\partial^n (\boldsymbol{r}-\boldsymbol{F}_1[\boldsymbol{r}])}{(\partial\xi^2)^n}(\xi^1,0,\xi^3) \\
&\quad + \sum_{n=0}^{P_2^2} \alpha_{2,n}^2(\xi^2)\frac{\partial^n (\boldsymbol{r}-\boldsymbol{F}_1[\boldsymbol{r}])}{(\partial\xi^2)^n}(\xi^1,1,\xi^3) \ ,
\end{aligned}$$

$$\boldsymbol{P}[\boldsymbol{r}](\boldsymbol{\xi}) = \boldsymbol{F}_2[\boldsymbol{r}](\boldsymbol{\xi}) + \sum_{n=0}^{P_1^3} \alpha_{1,n}^3(\xi^3) \frac{\partial^n (\boldsymbol{r} - \boldsymbol{F}_2[\boldsymbol{r}])}{(\partial \xi^3)^n}(\xi^1, \xi^2, 0)$$
$$+ \sum_{n=0}^{P_2^3} \alpha_{2,n}^3(\xi^3) \frac{\partial^n (\boldsymbol{r} - \boldsymbol{F}_2[\boldsymbol{r}])}{(\partial \xi^3)^n}(\xi^1, \xi^2, 1) \ . \tag{5.14}$$

Two-Dimensional Interpolation

The formulas for two-dimensional transfinite interpolation of a two-dimensional vector-valued function $\boldsymbol{x}(\boldsymbol{\xi}) : \Xi^2 \to X^2$ are obtained from (5.13) and (5.14) by assuming $\boldsymbol{F}_2(\boldsymbol{r}) = \boldsymbol{P}(\boldsymbol{r})$, $\alpha_{l,k}^3 = 0$, and omitting ξ^3. For example, we obtain, from (5.13), the following formula for two-dimensional transfinite interpolation:

$$\boldsymbol{F}_1[\boldsymbol{r}](\xi^1, \xi^2) = \sum_{l=1}^{L^1} \sum_{k=0}^{P_l^1} \alpha_k^1(\xi^1) \frac{\partial^k \boldsymbol{r}}{(\partial \xi^1)^k}(\xi_l^1, \xi^2) \ ,$$
$$\boldsymbol{P}[\boldsymbol{r}](\xi^1, \xi^2) = \boldsymbol{F}_1[\boldsymbol{r}](\xi^1, \xi^2)$$
$$+ \sum_{l=1}^{L^2} \sum_{m=0}^{P_l^2} \alpha_{l,m}^2(\xi^2) \frac{\partial^m (\boldsymbol{r} - \boldsymbol{F}_1[\boldsymbol{r}])}{(\partial \xi^2)^m}(\xi^1, \xi_l^2) \ . \tag{5.15}$$

5.3 Algebraic Coordinate Transformations

This section sets out the definitions of the algebraic coordinate transformations appropriate for the generation of coordinate grids through the formulas of transfinite interpolation.

5.3.1 Formulation of Algebraic Coordinate Transformation

The formulas of transfinite interpolation described above give clear guidance on how to define an algebraic coordinate transformation

$$\boldsymbol{x}(\boldsymbol{\xi}) : \Xi^3 \to X^3 \ , \qquad \boldsymbol{x}(\boldsymbol{\xi}) = (x^1(\boldsymbol{\xi}), x^2(\boldsymbol{\xi}) \ , x^3(\boldsymbol{\xi})) \ , \qquad \boldsymbol{\xi} = (\xi^1, \xi^2, \xi^3)$$

from the cube Ξ^3 onto a domain $X^3 \subset R^3$ which matches, at the boundary and some chosen intermediate coordinate planes of the cube, the prescribed values and the specified derivatives of $\boldsymbol{x}(\boldsymbol{\xi})$ along the coordinate directions emerging from the coordinate surfaces (Fig. 5.1).

Let there be chosen, in each direction ξ^i, some coordinate planes $\xi^i = \xi_l^i$, $l = 1, \ldots, L^i$, of the cube Ξ^3, including two opposite boundary planes $\xi^i = \xi_1^i = 0$,

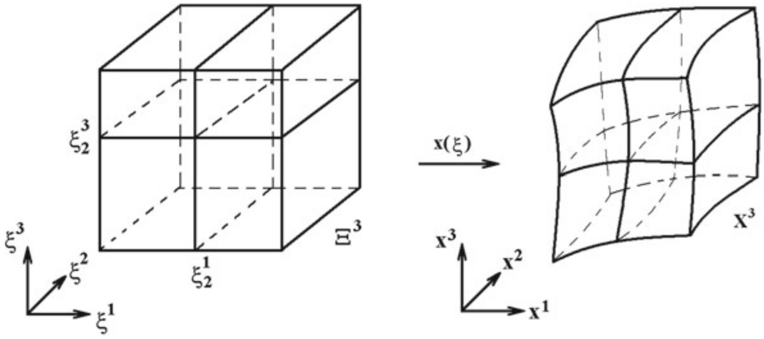

Fig. 5.1 Coordinate transformation

$\xi^i = \xi^i_{L^i} = 1$. Furthermore, let there be specified, at each section $\xi^i = \xi^i_l$, a smooth three-dimensional vector-valued function denoted by $\boldsymbol{A}^i_{l,0}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l})$, which is assumed to represent the values of the function $\boldsymbol{x}(\boldsymbol{\xi})$ being constructed at the points of this section. Also, let there be specified, at this section, three-dimensional vector-valued functions denoted by $\boldsymbol{A}^i_{l,n}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l})$ which represent derivatives with respect to ξ^i of the function $\boldsymbol{x}(\boldsymbol{\xi})$ on the respective sections $\xi^i = \xi^i_l$. Thus, it is assumed that

$$\boldsymbol{A}^i_{l,0}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l}) = \frac{\partial^0}{(\partial\xi^i)^0}\boldsymbol{x}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l}) = \boldsymbol{x}(\boldsymbol{\xi}|_{\xi^l=\xi^i_l}) , \qquad l = 1, \dots, L_i ,$$
$$\boldsymbol{A}^i_{l,n}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l}) = \frac{\partial^n}{(\partial\xi^i)^n}\boldsymbol{x}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l}) , \qquad n = 1, \dots, P^i_l .$$

Since

$$\frac{\partial^m}{(\partial\xi^j)^m}\Big(\frac{\partial^n \boldsymbol{x}}{(\partial\xi^i)^n}\Big) = \frac{\partial^n}{(\partial\xi^j)^n}\Big(\frac{\partial^m \boldsymbol{x}}{(\partial\xi^j)^m}\Big),$$

we find that the vector functions $\boldsymbol{A}^i_{l,n}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l})$ and $\boldsymbol{A}^j_{k,m}(\boldsymbol{\xi}|_{\xi^j=\xi^j_k})$ specifying the corresponding derivatives on the planes $\xi^i = \xi^i_l$ and $\xi^j = \xi^j_k$, respectively, must be compatible at the intersection of these planes, i.e.

$$\frac{\partial^m}{(\partial\xi^j)^m}\boldsymbol{A}^i_{l,n}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l,\xi^j=\xi^j_k}) = \frac{\partial^n}{(\partial\xi^j)^n}\boldsymbol{A}^j_{k,m}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l,\xi^j=\xi^j_k}) ,$$
$$n = 0, \dots, P^i_l , \qquad m = 0, \dots, P^j_k . \tag{5.16}$$

When the vector-valued functions $\boldsymbol{A}^i_{l,k}$ satisfying (5.16) are specified, the transformation $\boldsymbol{x}(\boldsymbol{\xi})$ is obtained by substituting the functions $\boldsymbol{A}^i_{l,0}$ and $\boldsymbol{A}^i_{l,n}$ for the values of $\boldsymbol{r}(\boldsymbol{\xi})$ and of its derivatives $\partial^n \boldsymbol{r}/\partial(\xi^i)^n(\boldsymbol{\xi}|_{\xi^i=\xi^i_l})$, respectively, in the above formulas for transfinite interpolation. Hence, the transformation based on the unidirectional interpolation given by (5.1) has the form

$$\boldsymbol{P}_i(\boldsymbol{\xi}) = \sum_{l=1}^{L^i} \sum_{n=0}^{P_l^i} \alpha_{l,n}^i(\xi^i) \boldsymbol{A}_{l,n}^i(\boldsymbol{\xi}|_{\xi^i=\xi_l^i}) \ . \tag{5.17}$$

This mapping matches the values of $\boldsymbol{A}_{l,n}^i$ only at the coordinate planes $\xi^i = \xi_l^i$ crossing the chosen coordinate ξ^i.

The formula (5.5) for the tensor product $\otimes$ of the two mappings $\boldsymbol{P}_i(\boldsymbol{\xi})$ and $\boldsymbol{P}_j(\boldsymbol{\xi})$ obtained from (5.17) then gives the transformation

$$\begin{aligned} &\boldsymbol{P}_i \otimes \boldsymbol{P}_j(\boldsymbol{\xi}) \\ &= \sum_{k=1}^{L^j} \sum_{m=0}^{P_k^j} \sum_{l=1}^{L^i} \sum_{n=0}^{P_l^i} \alpha_{l,n}^i(\xi^i) \alpha_{k,m}^j(\xi^j) \frac{\partial^n}{(\partial \xi^i)^n} \boldsymbol{A}_{k,m}^j(\boldsymbol{\xi}|_{\xi^i=\xi_l^i, \xi^j=\xi_k^j}) \ , \end{aligned} \tag{5.18}$$

which matches the values of $\boldsymbol{A}_{l,n}^i$ and $\boldsymbol{A}_{k,m}^j$ at the intersection of the planes $\xi^i = \xi_l^i$ and $\xi^j = \xi_k^j$. According to the consistency conditions (5.16), the operation of the tensor product is commutative, i.e.

$$\boldsymbol{P}_i \otimes \boldsymbol{P}_j(\boldsymbol{\xi}) = \boldsymbol{P}_j \otimes \boldsymbol{P}_i(\boldsymbol{\xi}) \ ,$$

which is indispensable for an appropriate definition of the coordinate transformation $\boldsymbol{x}(\boldsymbol{\xi})$.

5.3.2 General Algebraic Transformations

The general formula for the three-dimensional coordinate transformation $\boldsymbol{x}(\boldsymbol{\xi})$ that provides a matching with $\boldsymbol{A}_{l,n}^i$ in all directions and at all chosen coordinate planes $\xi^i = \xi_l^i$ is given by the replacement of the values of the function $\boldsymbol{r}(\boldsymbol{\xi})$ and of its derivatives in the recursive formula (5.13) by the functions $\boldsymbol{A}_{l,n}^i$. Thus, we obtain

$$\begin{aligned} \boldsymbol{F}_1(\boldsymbol{\xi}) &= \sum_{l=1}^{L^1} \sum_{n=0}^{P_l^1} \alpha_{l,n}^1(\xi^1) \boldsymbol{A}_{l,n}^1(\xi_l^1, \xi^2, \xi^3) \ , \\ \boldsymbol{F}_2(\boldsymbol{\xi}) &= \boldsymbol{F}_1(\xi) + \sum_{l=1}^{L^2} \sum_{n=0}^{P_l^2} \alpha_{l,n}^2(\xi^2) \Big(\boldsymbol{A}_{l,n}^2 - \frac{\partial^n \boldsymbol{F}_1}{(\partial \xi^2)^n} \Big)(\xi^1, \xi_l^2, \xi^3) \ , \\ \boldsymbol{x}(\boldsymbol{\xi}) &= \boldsymbol{F}_2(\boldsymbol{\xi}) + \sum_{l=1}^{L^3} \sum_{n=0}^{P_l^3} \alpha_{l,n}^3(\xi^3) \Big(\boldsymbol{A}_{l,n}^3 - \frac{\partial^n \boldsymbol{F}_2}{(\partial \xi^3)^n} \Big)(\xi^1, \xi^2, \xi_l^3) \ . \end{aligned} \tag{5.19}$$

As the specified functions $\boldsymbol{A}^i_{l,n}$ are consistent on the intersections of the planes $\xi^i = \xi^i_l$ and $\xi^j = \xi^j_k$ and, therefore, the tensor product of the transformations $\boldsymbol{P}_i(\boldsymbol{\xi})$ and $\boldsymbol{P}_j(\boldsymbol{\xi})$ is commutative, the result (5.19) is independent of the specific ordering of the successive interpolation directions ξ^i.

The formula for the two-dimensional algebraic coordinate transformation is obtained in a corresponding way from (5.15):

$$\boldsymbol{F}_1(\boldsymbol{\xi}) = \sum_{l=1}^{L^1}\sum_{n=0}^{P^1_l} \alpha^1_{l,n}(\xi^1)\boldsymbol{A}^1_{l,n}(\xi^1_l, \xi^2) \; ,$$
$$\boldsymbol{x}(\boldsymbol{\xi}) = \boldsymbol{F}_1(\boldsymbol{\xi}) + \sum_{l=1}^{L^2}\sum_{n=0}^{P^2_l} \alpha^2_{l,n}\Big(\boldsymbol{A}^2_{l,n} - \frac{\partial^n \boldsymbol{F}_1}{(\partial \xi^2)^n}\Big)(\xi^1, \xi^2_l) \; , \tag{5.20}$$

where $\boldsymbol{A}^i_{l,n}$ are two-dimensional vector-valued functions representing $\boldsymbol{x}(\boldsymbol{\xi})$ for $n = 0$ and its derivatives for $P^i_l \geq n > 0$ at the sections

$$\xi^i = \xi^i_l \; , \qquad i = 1, 2 \; , \qquad l = 1, \ldots, L^i \; .$$

These functions must satisfy the relations (5.16) at the points $(\xi^1_l, \; \xi^2_m)$, $l = 1, \ldots, L^1, \; m = 1, \ldots, L^2$.

The vector-valued function $\boldsymbol{x}(\boldsymbol{\xi})$ defined by (5.19) maps the unit cube Ξ^3 onto the physical region X^3 bounded by the six coordinate surfaces specified by the parametrizations $\boldsymbol{A}^i_{1,0}(\boldsymbol{\xi}|_{\xi^i=0})$ and $\boldsymbol{A}^i_{L^i,0}(\boldsymbol{\xi}|_{\xi^i=1})$, $i = 1, 2, 3$, from the respective boundary intervals of Ξ^3. The introduction of the intermediate planes $\xi^i = \xi^i_l$, $0 < \xi^i_l < 1$, into the formulas of transfinite interpolation allows one to control the grid distribution and grid spacing in the vicinity of some selected interior surfaces of the domain X^3. A similar result is achieved by joining, at the selected boundary surfaces, a series of transformations $\boldsymbol{x}(\boldsymbol{\xi})$ constructed using the above described outer boundary interpolation equation (5.14):

$$\boldsymbol{F}_1(\boldsymbol{\xi}) = \sum_{n=0}^{P^1_1} \alpha^1_{1,n}(\xi^1)\boldsymbol{A}^1_{1,n}(0, \xi^2, \xi^3)$$
$$+ \sum_{n=0}^{P^1_2} \alpha^1_{2,n}(\xi^1)\boldsymbol{A}^1_{2,n}(1, \xi^2, \xi^3) \; ,$$
$$\boldsymbol{F}_2(\xi) = \boldsymbol{F}_1(\xi) + \sum_{n=0}^{P^2_1} \alpha^2_{1,n}(\xi^2)\Big(\boldsymbol{A}^2_{1,n} - \frac{\partial^n \boldsymbol{F}_1}{(\partial \xi^2)^n}(\xi^1, 0, \xi^3)\Big)$$
$$+ \sum_{n=0}^{P^2_2} \alpha^2_{2,n}(\xi^2)\Big(\boldsymbol{A}^2_{2,n} - \frac{\partial^n \boldsymbol{F}_1}{(\partial \xi^2)^n}\Big)(\xi^1, 1, \xi^3) \; ,$$

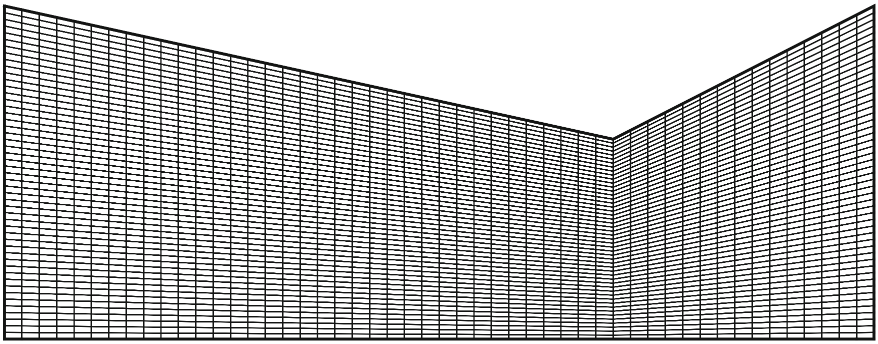

Fig. 5.2 Two-dimensional nonsmooth grid generated by means of transfinite interpolation

$$\boldsymbol{x}(\boldsymbol{\xi}) = \boldsymbol{F}_2(\xi) + \sum_{n=0}^{P_1^3} \alpha_{1,n}^3(\xi^3)\Big(\boldsymbol{A}_{1,n}^3 - \frac{\partial^n \boldsymbol{F}_2}{(\partial\xi^3)^n}\Big)(\xi^1, \xi^2, 0)$$
$$+ \sum_{n=0}^{P_2^3} \alpha_{2,n}^3(\xi^3)\Big(\boldsymbol{A}_{2,n}^3 - \frac{\partial^n \boldsymbol{F}_2}{(\partial\xi^3)^n}\Big)(\xi^1, \xi^2, 1)\ . \quad (5.21)$$

This boundary interpolation transformation $\boldsymbol{x}(\boldsymbol{\xi})$ is widely applied to generate grids in regions around bodies. These domains cannot be successfully gridded by one global mapping $\boldsymbol{x}(\boldsymbol{\xi})$ from the unit cube Ξ^3 because of the inevitable singularities pertinent to such global maps. An approach based on the matching of a series of boundary-interpolated transformations is thus preferable. It only requires the consistent specification of the parametrizations and coordinate directions at the corresponding boundary surfaces.

Equations (5.18)–(5.21) use the same set of blending functions $\alpha_{l,n}^i(\xi^i)$ to define each component $x^i(\xi)$ of the transformation $\boldsymbol{x}(\boldsymbol{\xi})$. These formulas can be generalized by introducing an individual set of blending functions $\alpha_{l,n}^i(\xi^i)$ for the definition of each component $x^i(\boldsymbol{\xi})$ of the map $\boldsymbol{x}(\boldsymbol{\xi})$ being built. Such a generalization gives broader opportunities to define appropriate algebraic coordinate transformations $\boldsymbol{x}(\boldsymbol{\xi})$ and, therefore, to generate grids more successfully.

One of the drawbacks of the method of transfinite interpolation for generating structured grids is that it carries boundary-sharp bends inside a domain (Fig. 5.2).

5.4 Lagrange and Hermite Interpolations

The recursive formula (5.19) represents a general form of transfinite interpolation which includes the prescribed values of the constructed coordinate transformation $\boldsymbol{x}(\boldsymbol{\xi})$ and of its derivatives up to order P_l^i at the sections $\xi^i = \xi_l^i$ of the cube Ξ^3.

However, most grid generation codes require, as a rule, only specification of the values of the function $\boldsymbol{x}(\boldsymbol{\xi})$ being sought and sometimes, in addition, the values of its first derivatives at the selected sections. Such sorts of algebraic coordinate transformation are described in this section.

5.4.1 Coordinate Transformations Based on Lagrange Interpolation

A Lagrange interpolation matches only the values of the function $\boldsymbol{r}(\boldsymbol{\xi})$ at some prescribed sections $\xi^i = \xi^i_l,\ l = 1, \dots, L^i$, of the cube Ξ^3. So, in accordance with (5.1), the unidirectional Lagrange interpolation has the following form:

$$\boldsymbol{P}_i[\boldsymbol{r}](\boldsymbol{\xi}) = \sum_{l=1}^{L^i} \alpha^i_l(\xi^i)\boldsymbol{r}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l}) \ .$$

The blending function $\alpha^i_l(\xi^i)$ in this equation corresponds to $\alpha^i_{l,0}(\xi^i)$ in the formula (5.1). Taking into account (5.2), the blending functions $\alpha^i_l(\xi^i),\quad l = 1, \dots, L^i$, depending on one independent variable ξ^i, must be subject to the following restrictions:

$$\alpha^i_l(\xi^i_k) = \delta^l_k \ , \qquad l, k = 1, \dots, L^i \ . \tag{5.22}$$

These restrictions imply that the blending function α^i_l for a fixed l equals 1 at the point $\xi^i = \xi^i_l$ and equals zero at all other points $\xi^i_m,\ m \neq l$. The formula for the construction of a three-dimensional coordinate mapping $\boldsymbol{x}(\boldsymbol{\xi})$ based on the Lagrangian interpolation is obtained from (5.19) as

$$\begin{aligned}
\boldsymbol{F}_1(\boldsymbol{\xi}) &= \sum_{l=1}^{L^1} \alpha^1_l(\xi^1)\boldsymbol{A}^1_l(\boldsymbol{\xi}|_{\xi^1=\xi^1_l}) \ , \\
\boldsymbol{F}_2(\boldsymbol{\xi}) &= \boldsymbol{F}_1(\boldsymbol{\xi}) + \sum_{l=1}^{L^2} \alpha^2_l(\xi^2)\Big(\boldsymbol{A}^2_l - \boldsymbol{F}_1\Big)(\boldsymbol{\xi}|_{\xi^2=\xi^2_l}) \ , \\
\boldsymbol{x}(\boldsymbol{\xi}) &= \boldsymbol{F}_2(\boldsymbol{\xi}) + \sum_{l=1}^{L^3} \alpha^3_l(\xi^3)\Big(\boldsymbol{A}^3_l - \boldsymbol{F}_2\Big)(\boldsymbol{\xi}|_{\xi^3=\xi^3_l}) \ ,
\end{aligned} \tag{5.23}$$

where the blending functions $\alpha^i_l(\xi^i)$ satisfy (5.22), and the functions $\boldsymbol{A}^i_l(\xi|_{\xi^i=\xi^i_l})$ corresponding to $\boldsymbol{A}^i_{l,0}$ in (5.22) specify the values of the mapping $\boldsymbol{x}(\boldsymbol{\xi})$ being sought. In accordance with (5.16), the specified functions $\boldsymbol{A}^i_l$ must coincide at the intersection of their respective coordinate planes $\xi^i = \xi^i_l$, i.e.

$$\boldsymbol{A}^i_l(\boldsymbol{\xi}|_{\xi^i=\xi^i_l,\xi^j=\xi^j_k}) = \boldsymbol{A}^j_k(\boldsymbol{\xi}|_{\xi^i=\xi^i_l,\xi^j=\xi^j_k}) \ .$$

When $L^1 = L^2 = L^3 = 2$, i.e. the prescribed interior sections are absent, then the ith component of the transformation (5.23) has the following form:

$$\begin{aligned} F^i_1(\xi^1,\xi^2,\xi^3) &= \alpha^1_1(\xi^1)\psi^i(0,\xi^2,\xi^3) + \alpha^1_2(\xi^1)\psi^i(1,\xi^2,\xi^3), \\ F^i_2(\xi^1,\xi^2,\xi^3) &= F^i_1(\xi^1,\xi^2,\xi^3) + \alpha^2_1(\xi^2)[\psi^i(\xi^1,0,\xi^3) - F^i_1(\xi^1,0,\xi^3)] + \\ &\quad + \alpha^2_2(\xi^2)[\psi^i(\xi^1,1,\xi^3) - F^i_1(\xi^1,1,\xi^3)], \\ x^i(\xi^1,\xi^2,\xi^3) &= F^i_2(\xi^1,\xi^2,\xi^3) + \alpha^3_1(\xi^3)[\psi^i(\xi^1,\xi^2,0) - F^i_2(\xi^1,\xi^2,0)] + \\ &\quad + \alpha^3_2(\xi^3)[\psi^i(\xi^1,\xi^2,1) - F^i_2(\xi^1,\xi^2,1)], \quad i = 1,2,3, \end{aligned} \tag{5.24}$$

where the function $\psi^i(\xi^1,\xi^2,\xi^3)$, $i = 1,2,3$ is the ith component of a specified boundary transformation $\boldsymbol{\psi}(\boldsymbol{\xi}) : \partial\Xi^3 \to \partial X^3$; $\alpha^i_l(t) : [0,1] \to \mathbb{R}$, $i = 1,2,3$, $l = 1,2$ are scalar blending functions subject to the following restrictions:

$$\alpha^i_1(0) = 1, \quad \alpha^i_1(1) = 0, \quad \alpha^i_2(0) = 0, \quad \alpha^i_2(1) = 1, \quad i = 1,2,3. \tag{5.25}$$

In particular, when $\alpha^i_1(t) = 1 - t$, $\alpha^i_2(t) = t$, $i = 1,2,3$, we obtain the simplest formula of transfinite interpolation in a vector form

$$\begin{aligned} \boldsymbol{F}_1(\boldsymbol{\xi}) &= (1-\xi^1)\boldsymbol{\psi}(0,\xi^2,\xi^3) + \xi^1\boldsymbol{\psi}(1,\xi^2,\xi^3), \\ \boldsymbol{F}_2(\boldsymbol{\xi}) &= \boldsymbol{F}_1(\boldsymbol{\xi}) + (1-\xi^2)[\boldsymbol{\psi}(\xi^1,0,\xi^3) - \boldsymbol{F}_1(\xi^1,0,\xi^3)] \\ &\quad + \xi^2[\boldsymbol{\psi}(\xi^1,1,\xi^3) - \boldsymbol{F}_1(\xi^1,1,\xi^3)], \\ \boldsymbol{x}(\boldsymbol{\xi}) &= \boldsymbol{F}_2(\boldsymbol{\xi}) + (1-\xi^3)[\boldsymbol{\psi}(\xi^1,\xi^2,0) - \boldsymbol{F}_2(\xi^1,\xi^2,0)] \\ &\quad + \xi^3[\boldsymbol{\psi}(\xi^1,\xi^2,1) - \boldsymbol{F}_2(\xi^1,\xi^2,1)]. \end{aligned} \tag{5.26}$$

The conditions (5.25) provide the identity

$$\boldsymbol{x}(\boldsymbol{\xi})|_{\partial\Xi^3} = \boldsymbol{\psi}(\boldsymbol{\xi})|_{\partial\Xi^3} \tag{5.27}$$

for the transformation $\boldsymbol{x}(\boldsymbol{\xi})$ obtained by (5.24).

Now we consider some examples of the blending functions used in Lagrange interpolations.

Lagrange Polynomials

The best-known blending functions $\alpha^i_l(\xi^i)$ satisfying (5.22) are defined as Lagrange polynomials applied to the points $\xi^i_1,\ldots,\xi^i_{L^i}$:

$$\alpha^i_l(\xi^i) = \prod_{j=1}^{L^i} \frac{\xi^i - \xi^i_j}{\xi^i_l - \xi^i_j} \ , \qquad j \neq l \ . \tag{5.28}$$

For example, when $L^i = 2$, then, from (5.28),

$$\alpha_1^i(\xi^i) = \frac{\xi^i - \xi_2^i}{\xi_1^i - \xi_2^i}\ , \qquad \alpha_2^i(\xi^i) = \frac{\xi^i - \xi_1^i}{\xi_2^i - \xi_1^i} = 1 - \alpha_1^i(\xi^i)\ . \tag{5.29}$$

Therefore, for the boundary interpolation, i.e. when $\xi_1^i = 0$, $\xi_2^i = 1$, we obtain

$$\alpha_1^i(\xi^i) = 1 - \xi^i\ , \qquad \alpha_2^i(\xi^i) = \xi^i\ . \tag{5.30}$$

Spline Functions

The Lagrange polynomials become polynomials of a high-order when a large number of intermediate sections $\xi^i = \xi_l^i$ is applied to control the grid distribution in the interior of the domain X^3. These polynomials of high order may cause oscillations. One way to overcome this drawback is to use splines as blending functions $\alpha_l^i(\xi^i)$. The splines are defined as polynomials of low-order between each of the specified points $\xi^i = \xi_{L^i}^i$, with continuity of some derivatives at the interior points.

Piecewise-continuous splines satisfying (5.22) can be derived by means of linear polynomials. The simplest pattern of such blending functions in the form of splines consists of piecewise linear functions:

$$\alpha_l(\xi^i) = \begin{cases} 0\ , & \xi^i \le \xi_{l-1}^i, \\ \dfrac{\xi^i - \xi_{l-1}^i}{\xi_l^i - \xi_{l-1}^i}\ , & \xi_{l-1}^i \le \xi^i \le \xi_l^i, \\ \dfrac{\xi_{l+1}^i - \xi^i}{\xi_{l+1}^i - \xi_l^i}\ , & \xi_l^i \le \xi^i \le \xi_{l+1}^i, \\ 0\ , & \xi^i \ge \xi_{l+1}^i \end{cases}$$

However, the use of these blending functions results in a nonsmooth point distribution, since they themselves are not smooth.

Continuity of the first derivative of a spline blending function can be achieved with polynomials of the third-order, regardless of the number of interior sections.

Construction Based on General Functions

The application of polynomials in the Lagrange interpolation gives only a poor opportunity to control the grid spacing near the selected boundary and interior surfaces. In this subsection, we describe a general approach, originally, proposed by Liseikin (1999), to constructing the blending functions $\alpha_l^i(\xi^i)$ through the use of a wide range of basic functions, which provides a real opportunity to control the grid point distribution.

The formulation of the blending functions on the interval $0 \le \xi^i \le 1$, with L^i specified points,

$$0 = \xi_1^i < \cdots < \xi_{L^i}^i = 1\ ,$$

requires only the specification of some univariate smooth positive function

$$\phi(x) : [0, \infty) \to [0, \infty)\ ,$$

satisfying the restrictions $\phi(0) = 0$, $\phi(1) = 1$. This function can be used as a basic element to derive the blending functions satisfying (5.22) through the following standard procedure.

First, we define two series of functions

$$\phi_l^f(\xi^i) \quad \text{and} \quad \phi_l^b(\xi^i) \ , \qquad l = 1, \ldots, L^i \ .$$

The functions $\phi_l^f(\xi^i)$ are defined for $l = 1$ by

$$\phi_1^f(\xi^i) = \phi(1 - \xi^i) \ , \qquad 0 \le \xi \le 1 \ ,$$

and for $1 < l \le L^i$ by

$$\phi_l^f(\xi^i) = \begin{cases} 0 \ , & 0 \le \xi^i \le \xi_{l-1}^i \ , \\ \phi\left(\dfrac{\xi^i - \xi_{l-1}^i}{\xi_l^i - \xi_{l-1}^i}\right) \ , & \xi_{l-1}^i \le \xi^i \le 1 \ . \end{cases}$$

The functions $\phi_l^b(\xi^i)$ are determined similarly:

$$\phi_{L^i}^b(\xi^i) = \phi(\xi^i)$$

and for $1 \le l < L^i$,

$$\phi_l^b(\xi^i) = \begin{cases} 0 \ , & 1 \ge \xi^i \ge \xi_{l+1}^i \ , \\ \phi\left(\dfrac{\xi_{l+1}^i - \xi^i}{\xi_{l+1}^i - \xi_l^i}\right) \ , & 0 \le \xi^i \le \xi_{l+1}^i \ . \end{cases}$$

Using the functions $\phi_l^f(\xi^i)$ and $\phi_l^b(\xi^i)$, the blending coefficients $\alpha_l^i(\xi^i)$ satisfying (5.22) are defined by

$$\alpha_l^i(\xi^i) = \phi_l^f(\xi^i)\phi_l^b(\xi^i) \ , \qquad l = 1, \ldots, L^i \ . \tag{5.31}$$

Each of these blending functions vanishes outside some interval, and thus it affects the interpolation function only locally (Fig. 5.3).

Note that this procedure for constructing blending functions for the Lagrange interpolations will yield splines if the original function ϕ is a polynomial. This construction may be extended by using various original functions for the terms ϕ_l^f and ϕ_l^b in (5.31).

The simplest example of the basic function is $\phi(x) = x$. However, this function generates nonsmooth blending coefficients $\alpha_l^i(\xi^i)$ at the points ξ_{l-1}^i and ξ_{l+1}^i, since $\alpha_l^i(\xi^i) \equiv 0$ outside the interval $(\xi_{l-1}^i, \ \xi_{l+1}^i)$. If the derivative of $\phi(x)$ at the point $x = 0$ is zero, then the blending functions derived by the procedure described are smooth. One example of such a function is $\phi(x) = x^2$. It can readily be shown that in this case, the blending functions α_l^i are of the class $C^1[0, 1]$.

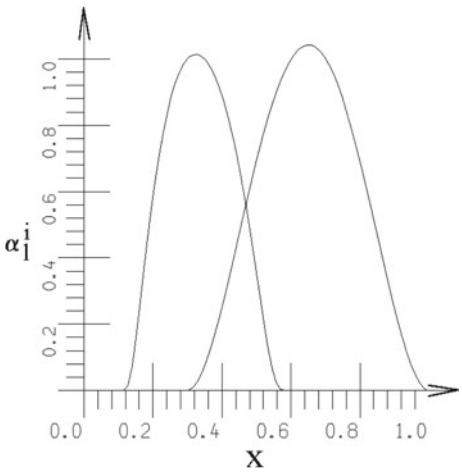

Fig. 5.3 Smooth blending functions

Continuity of the higher-order derivatives of the blending functions (5.31) is obtained when the basic function $\phi(x)$ satisfies the condition $\phi^{(k)}(0) = 0,\ k > 1$, in particular, if $\phi(x) = x^{k+1}$. The function $\phi(x) = \varphi(x)$, where

$$\varphi(x) = \begin{cases} 0\,, & x = 0\,, \\ a^{1-1/x}\,, & x > 0\,, \end{cases}$$

with $a > 1$, generates an infinitely differentiable blending function $\alpha_l^i(\xi^i)$ on the interval $[0, 1]$. Figure 5.3 demonstrates the blending functions constructed for $\phi(x) = \varphi(x)$ (left) and $\phi(x) = x^2$ (right).

Relations Between Blending Functions

Now we point out some relations between blending functions which can be useful for their construction. If the functions $\alpha_l^i(\xi^i)$ are blending functions for Lagrangian interpolation, namely, they are subject to the restrictions (5.22), then the functions $\beta_l^i(\xi^i)$ defined below satisfy the condition (5.22) as well:

$$\begin{aligned}
&(1)\quad \beta_l^i(\xi^i) = \alpha_l^i(\xi^i) f(\xi) \quad \text{if} \quad f(\xi_l^i) = 1\,, \\
&(2)\quad \beta_l^i(\xi^i) = \alpha_l^i[f(\xi^i)] \quad \text{if} \quad f(\xi_l^i) = \xi_l^i\,, \\
&(3)\quad \beta_l^i(\xi^i) = f[\alpha_l^i(\xi^i)] \quad \text{if} \quad f(0) = 0,\ f(1) = 1\,, \\
&(4)\quad \beta_l^i(\xi^i) = \alpha_l^i(\xi^i) + f(\xi^i) \quad \text{if} \quad f(\xi_l^i) = 0\,, \\
&(5)\quad \beta_l^i(\xi^i) = 0.5[\alpha_l^i(\xi^i) + \gamma_l^i(\xi^i)] \quad \text{if} \quad \gamma_l^i(\xi) \text{ satisfies (5.22)}\,.
\end{aligned} \tag{5.32}$$

5.4.2 Transformations Based on Hermite Interpolation

Hermite interpolation matches the values of both the function $\boldsymbol{r}(\boldsymbol{\xi})$, and its first derivatives $\partial \boldsymbol{r}/\partial \xi^i(\boldsymbol{\xi}|_{\xi^i=\xi^i_l})$ at each section $\xi^i = \xi^i_l$, $l = 1, \dots, L^i$, and therefore, the unidirectional interpolation (5.1) takes the following form:

$$\boldsymbol{P}_i[\boldsymbol{r}](\boldsymbol{\xi}) = \sum_{l=1}^{L^i} \Big(\alpha^i_{l,0}(\xi^i)\boldsymbol{r}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l}) + \alpha^i_{l,1}(\xi^i)\frac{\partial \boldsymbol{r}}{\partial \xi^i}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l}) \Big) . \tag{5.33}$$

The formula (5.19) in the case of a Hermite coordinate mapping $\boldsymbol{x}(\boldsymbol{\xi})$ which matches the specified values of $\boldsymbol{x}(\boldsymbol{\xi})$, denoted by $\boldsymbol{A}^i_{l,0}$, and of its first derivatives, denoted by $\boldsymbol{A}^i_{l,1}$, at all sections $\xi^i = \xi^i_l$, $l = 1, \dots, L^i$, and in all directions ξ^i, $i = 1, 2, 3$, is thus reduced to

$$\begin{aligned}
\boldsymbol{F}_1(\boldsymbol{\xi}) &= \sum_{l=1}^{L^1} \Big(\alpha^1_{l,0}(\xi^1)\boldsymbol{A}^1_{l,0}(\xi^1_l, \xi^2, \xi^3) + \alpha^1_{l,1}(\xi^1)\boldsymbol{A}^1_{l,1}(\xi^1_l, \xi^2, \xi^3) \Big) , \\
\boldsymbol{F}_2(\boldsymbol{\xi}) &= \boldsymbol{F}_1(\boldsymbol{\xi}) + \sum_{l=1}^{L^2} \Big(\alpha^2_{l,0}(\xi^2)(\boldsymbol{A}^2_{l,0} - \boldsymbol{F}_1)(\xi^1, \xi^2_l, \xi^3) \\
&\quad + \alpha^2_{l,1}(\xi^2)(\boldsymbol{A}^2_{l,1} - \frac{\partial \boldsymbol{F}_1}{\partial \xi^2})(\xi^1, \xi^2_l, \xi^3) \Big) , \\
\boldsymbol{x}(\boldsymbol{\xi}) &= \boldsymbol{F}_2(\boldsymbol{\xi}) + \sum_{l=1}^{L^3} \Big(\alpha^3_{l,0}(\xi^3)\big(\boldsymbol{A}^3_{l,0} - \boldsymbol{F}_2\big)(\xi^1, \xi^2, \xi^3_l) \\
&\quad + \alpha^3_{l,1}(\xi^3)\big(\boldsymbol{A}^3_{l,1} - \frac{\partial \boldsymbol{F}_2}{\partial \xi^3}\big)(\xi^1, \xi^2, \xi^3_l) \Big) ,
\end{aligned} \tag{5.34}$$

where, in accordance with (5.2), the blending functions $\alpha^i_{l,0}$, $\alpha^i_{l,1}$ satisfy the conditions

$$\begin{aligned}
\alpha^i_{l,0}(\xi^i_k) &= \delta^l_k , & \alpha^i_{l,1}(\xi^i_k) &= 0 , \\
\frac{\mathrm{d}}{\mathrm{d}\xi^i}\alpha^i_{l,1}(\xi^i_k) &= \delta^l_k , & \frac{\mathrm{d}}{\mathrm{d}\xi^i}\alpha^i_{l,0}(\xi^i_k) &= 0 , \\
l, k &= 1, \dots, L^i , & i &= 1, 2, 3 ,
\end{aligned} \tag{5.35}$$

and the vector-valued functions $\boldsymbol{A}^i_{l,n}(\xi|_{\xi^i=\xi^i_l})$ satisfy the consistency conditions (5.16):

$$\begin{aligned}
\boldsymbol{A}^i_{l,0}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l,\xi^j=\xi^j_k}) &= \boldsymbol{A}^j_{k,0}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l,\xi^j=\xi^j_k}) , \\
\frac{\partial}{\partial \xi^j}\boldsymbol{A}^i_{l,0}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l,\xi^j=\xi^j_k}) &= \boldsymbol{A}^j_{k,1}(\boldsymbol{\xi}|_{\xi^i=\xi^i_l,\xi^j=\xi^j_k}) .
\end{aligned} \tag{5.36}$$

Construction of Blending Functions

The blending functions $\alpha^i_{l,m}(\xi^i), m = 0, 1$, for Hermite interpolations can be obtained from the smooth blending functions defined for Lagrange interpolations. Namely, let $\alpha^i_l(\xi^i)$, $l = 1, \ldots, L^i$, be some smooth scalar functions meeting the conditions (5.22). The functions $\alpha^i_{l,m}$, $m = 0, 1$, determined by the relations

$$\alpha^i_{l,0} = \Big(1 - 2(\xi^i - \xi^i_l)\frac{d\alpha^i_l}{d\xi^i}(\xi_i)\Big)[\alpha^i_l(\xi^i)]^2 ,$$
$$\alpha^i_{l,1} = (\xi^i - \xi^i_l)[\alpha^i_l(\xi^i)]^2 , \tag{5.37}$$

then satisfy (5.35) and, therefore, are the blending functions for the Hermite interpolations. For example, if $L^i = 2$ and the Lagrangian blending functions are defined through (5.29), then, from (5.37),

$$\alpha^i_{1,0}(\xi^i) = \Big(1 - 2\frac{\xi^i - \xi^i_1}{\xi^i_1 - \xi^i_2}\Big)\Big(\frac{\xi^i - \xi^i_2}{\xi^i_1 - \xi^i_2}\Big)^2 ,$$
$$\alpha^i_{2,0}(\xi^i) = \Big(1 - 2\frac{\xi^i - \xi^i_2}{\xi^i_2 - \xi^i_1}\Big)\Big(\frac{\xi^i - \xi^i_1}{\xi^i_2 - \xi^i_1}\Big)^2 ,$$
$$\alpha^i_{1,1}(\xi^i) = (\xi^i - \xi^i_1)\Big(\frac{\xi^i - \xi^i_2}{\xi^i_1 - \xi^i_2}\Big)^2 ,$$
$$\alpha^i_{2,1}(\xi^i) = (\xi^i - \xi^i_2)\Big(\frac{\xi^i - \xi^i_1}{\xi^i_2 - \xi^i_1}\Big)^2 . \tag{5.38}$$

So, if $\xi^i_1 = 0$, $\xi^i_2 = 1$, then, from these relations,

$$\alpha^i_{1,0}(\xi^i) = (1 + 2\xi^i)(\xi^i - 1)^2 ,$$
$$\alpha^i_{2,0}(\xi^i) = (3 - 2\xi^i)(\xi^i)^2 = 1 - \alpha^i_{1,0}(\xi^i) ,$$
$$\alpha^i_{1,1}(\xi^i) = \xi^i(1 - \xi^i)^2 ,$$
$$\alpha^i_{2,1}(\xi^i) = (\xi^i - 1)(\xi^i)^2 . \tag{5.39}$$

If the blending functions for Lagrange interpolation satisfy the condition

$$\frac{d\alpha^i_l}{d\xi^i}(\xi^i) \equiv 0 , \quad \text{if} \quad \xi^i \geq \xi^i_{l+1} \quad \text{and} \quad \xi^i \leq \xi^i_{l-1} , \tag{5.40}$$

then the blending functions $\alpha^i_{l,n}(\xi^i)$ for the Hermite interpolation can be derived from $\alpha^i_l(\xi^i)$ by the relations

$$\alpha^i_{l,0}(\xi^i) = \Big(1 + (\xi^i - \xi^i_l)\frac{d\alpha^i_l}{d\xi^i}(\xi^i_l)\Big)\alpha^i_l(\xi^i) ,$$
$$\alpha^i_{l,1}(\xi^i) = (\xi^i - \xi^i_l)\alpha^i_l(\xi^i) . \tag{5.41}$$

It is readily shown that the blending functions $\alpha^i_{l,n}(\xi^i),\ n = 0, 1$, satisfy the restriction (5.35). Note that the approach described above for the general construction of the blending functions for Lagrange interpolation yields the smooth blending functions $\alpha^i_l(\xi^i),\ l = 1, \ldots, L^i$, in the form (5.31), which, in addition to (5.22), are also subject to (5.40).

Deficient Form of Hermite Interpolation

Often, it is not reasonable to specify the values of the first derivative with respect to ξ^i of the sought coordinate transformation $\boldsymbol{x}(\boldsymbol{\xi})$ at all sections $\xi^i = \xi^i_l, l = 1, \ldots, L^i$, but only at some selected ones. By omitting the corresponding terms

$$\alpha^1_{l,1}(\xi^1)\boldsymbol{A}^1_{l,1}(\boldsymbol{\xi}|_{\xi^1=\xi^1_l})$$

and/or

$$\alpha^i_{l,1}(\xi^i)(\boldsymbol{A}^i_{l,1} - \frac{\partial \boldsymbol{F}_{i-1}}{\partial \xi^i})(\boldsymbol{\xi}|_{\xi^i=\xi^i_l}) \ , \quad i = 2, 3 \ ,$$

in (5.34), a deficient form of Hermite interpolation is obtained which matches the values of the first derivatives at the selected sections only. For example, the outer boundary interpolation which contains the outer boundary specifications on all boundaries but the outward derivative with respect to ξ^1 on the boundary $\xi^1 = 0$ only has, in accordance with (5.34), the form

$$\begin{aligned}
\boldsymbol{F}_1(\boldsymbol{\xi}) &= \alpha^1_{1,0}(\xi^1)\boldsymbol{A}^1_{1,0}(0, \xi^2, \xi^3) + \alpha^1_{2,0}(\xi^1)\boldsymbol{A}^1_{2,0}(1, \xi^2, \xi^3) \\
&\quad + \alpha^1_{1,1}(\xi^1)\boldsymbol{A}^1_{1,1}(0, \xi^2, \xi^3) \ , \\
\boldsymbol{F}_2(\boldsymbol{\xi}) &= \boldsymbol{F}_1(\boldsymbol{\xi}) + \alpha^2_{1,0}(\xi^2)(\boldsymbol{A}^2_{1,0} - \boldsymbol{F}_1)(\xi^1, 0, \xi^3) \\
&\quad + \alpha^2_{2,0}(\xi^2)(\boldsymbol{A}^2_{2,0} - \boldsymbol{F}_1)(\xi^1, 1, \xi^3) \ , \\
\boldsymbol{x}(\boldsymbol{\xi}) &= \boldsymbol{F}_2(\boldsymbol{\xi}) + \alpha^3_{1,0}(\xi^3)(\boldsymbol{A}^3_{1,0} - \boldsymbol{F}_2)(\xi^1, \xi^2, 0) \\
&\quad + \alpha^3_{2,0}(\xi^3)(\boldsymbol{A}^3_{2,0} - \boldsymbol{F}_2)(\xi^1, \xi^2, 1) \ .
\end{aligned} \tag{5.42}$$

Specification of Normal Directions

In the outer boundary interpolation technique, the outward derivatives $\boldsymbol{A}^i_{1,1}(\xi|_{\xi^i=0}),\ \boldsymbol{A}^i_{2,1}(\xi|_{\xi^i=1})$ along the lines emerging from the boundary surfaces are usually required to be performed as normals to the corresponding boundary surfaces in order to generate orthogonal grids near the boundaries. The boundary surfaces are parametrized by the specified boundary transformations $\boldsymbol{A}^i_{1,0}(\xi|_{\xi^i=0})$ and $\boldsymbol{A}^i_{2,0}(\xi|_{\xi^i=1})$, respectively. Therefore, these normals can be computed from the cross product of the vectors tangential to the boundary surfaces. For example, the ξ^1 coordinate direction $\boldsymbol{A}^1_{l,1}(\xi^1_l, \xi^2, \xi^3)$ can be specified as

$$\boldsymbol{A}^1_{1,1}(0, \xi^2, \xi^3) = g(\xi^2, \xi^3)\Big(\frac{\partial}{\partial \xi^2}\boldsymbol{A}^1_{1,0}(0, \xi^2, \xi^3) \times \frac{\partial}{\partial \xi^3}\boldsymbol{A}^1_{1,0}(0, \xi^2, \xi^3)\Big) \ ,$$

where $g(\xi^2, \xi^3)$ is a scalar function that can be used to control the spacing of the grid lines emerging from the boundary surface represented by the parametrization $\boldsymbol{A}^1_{1,0}(0, \xi^2, \xi^3)$. Such a specification of the first derivatives can be chosen on the interior sections as well.

Parametrization of Cells

Formulas of transfinite interpolation can also be used for definition of the shape of a grid cell and the parametrization of this cell by values of the coordinates of its vertices. For this purpose, a transformation of the cube Ξ^3 into X^3 is written out in such a way that the vertices of the cube are transformed on the vertices of the cell, while the edges of the cube are transformed by the formula of the one–dimensional transformation. After this, the faces of the cube are transformed on the faces of the cell with the formula of the two–dimensional transfinite interpolation, and finally the interior of the cube Ξ^3 is mapped on the interior of the cell through the formula of the three–dimensional transfinite interpolation. The formula of such a transformation has the following form:

$$\begin{aligned}\boldsymbol{x}(\boldsymbol{\xi}) &= (1-\xi^1)(1-\xi^2)(1-\xi^3)\boldsymbol{x}_{000} + (1-\xi^1)(1-\xi^2)\xi^3\boldsymbol{x}_{001} + \\ &+ (1-\xi^1)\xi^2(1-\xi^3)\boldsymbol{x}_{010} + (1-\xi^1)\xi^2\xi^3\boldsymbol{x}_{011} + \xi^1(1-\xi^2)(1-\xi^3)\boldsymbol{x}_{100} + \\ &+ \xi^1(1-\xi^2)\xi^3\boldsymbol{x}_{101} + \xi^2\xi^2(1-\xi^3)\boldsymbol{x}_{110} + \xi^1\xi^2\xi^3\boldsymbol{x}_{111},\end{aligned}$$

where $\boldsymbol{x}_{i_1 i_2 i_3} = (x^1(i_1, i_2, i_3), x^2(i_1, i_2, i_3), x^3(i_1, i_2, i_3)), i_1, i_2, i_3 = 0, 1,$ are the vertices of the cell. The edges of this cell are the straight lines connecting its vertices, while its faces are surfaces of second order.

5.5 Control Techniques

Commonly, all algebraic schemes are computationally efficient but require a significant amount of user interaction and control techniques to define workable meshes. This section delineates some control approaches applied to algebraic grid generation.

The spacing between the grid points and the skewness of the grid cells in the physical domain is controlled in the algebraic method, primarily by the blending functions $\alpha^i_{l,n}(\xi^i)$, by the representations of the boundary and intermediate surfaces $\boldsymbol{A}^i_{l,0}(\xi|_{\xi^i=\xi^i_l})$, and by the values of the first derivatives $\boldsymbol{A}^i_{l,1}(\xi|_{\xi^i=\xi^i_l})$ in the interpolation equations.

As was stated in Chap. 4, an effective approach which significantly simplifies the control of grid generation relies on the introduction of an intermediate control domain between the computational and the physical regions. The control domain is a unit cube Q^3 with the Cartesian coordinates $q^i,\ i = 1, 2, 3$. In this approach, the coordinate transformation $\boldsymbol{x}(\boldsymbol{\xi})$ from the unit cube Ξ^3 onto the physical region X^3 is defined as a composition of two transformations: $\boldsymbol{q}(\boldsymbol{\xi})$ from Ξ^3 onto Q^3 and $g(\boldsymbol{q}),\ \boldsymbol{q} = (q^1, q^2, q^3)$, from Q^3 onto X^3, that is,

$$\boldsymbol{x}(\boldsymbol{\xi}) = g[\boldsymbol{q}](\boldsymbol{\xi}) : \Xi^3 \to X^3 \ .$$

The functions $g(\boldsymbol{q})$ and $\boldsymbol{q}(\boldsymbol{\xi})$ can be constructed through the formulas of transfinite interpolation or by other techniques. As both the computational domain Ξ^3 and the intermediate domain Q^3 are the standard unit cubes, the formulas of transfinite interpolation for the generation of the intermediate transformations $\boldsymbol{q}(\boldsymbol{\xi})$ are somewhat simpler than the original expressions. In these formulas, it can be assumed, without any loss of generality, that the boundary planes $\xi^i = 0$ and $\xi^i = 1$ for each $i = 1, 2, 3$ are transformed by the function $\boldsymbol{q}(\boldsymbol{\xi})$ onto the boundary planes $q^i = 0$ and $q^i = 1$, respectively, so that

$$\begin{aligned}
\boldsymbol{q}(\boldsymbol{\xi}|_{\xi^1=0}) &= [0, q^2(0, \xi^2, \xi^3), q^3(0, \xi^2, \xi^3)] \ ,\\
\boldsymbol{q}(\boldsymbol{\xi}|_{\xi^1=1}) &= [1, q^2(1, \xi^2, \xi^3), q^3(1, \xi^2, \xi^3)] \ .
\end{aligned}$$

Therefore, the first component $q^1(\xi)$ of the Lagrangian boundary interpolation for the intermediate mapping $\boldsymbol{q}(\boldsymbol{\xi})$ has the form

$$\begin{aligned}
F_1(\boldsymbol{\xi}) &= \alpha_2^1(\xi^1) \ ,\\
F_2(\boldsymbol{\xi}) &= F_1(\boldsymbol{\xi}) + \alpha_1^2(\xi^2)\Big(u^1(\xi^1, 0, \xi^3) - \alpha_2^1(\xi^1)\Big)\\
&\quad + \alpha_2^2(\xi^2)\ \Big(u^1(\xi^1, 1, \xi^3 - \alpha_2^1(\xi^1)\Big) \ ,\\
q^1(\boldsymbol{\xi}) &= F_2(\boldsymbol{\xi}) + \alpha_1^3(\xi^3)\Big(u^1(\xi^1, \xi^2, 0) - F_2(\xi^1, \xi^2, 0)\Big)\\
&\quad + \alpha_2^3(\xi^3)\Big(u^1(\xi^1, \xi^2, 1) - F_2(\xi^1, \xi^2, 1)\Big) \ . \qquad (5.43)
\end{aligned}$$

Analogous equations can be defined for the other components of the intermediate transformation $\boldsymbol{q}(\boldsymbol{\xi})$.

The functions based on the reference univariate transformations $x_{i,\mathrm{c}}(\varphi, \epsilon)$ and $x_{i,\mathrm{s}}(\varphi, \epsilon)$ introduced in Chap. 4 can be used very successfully as blending functions to construct intermediate transformations by Lagrange and Hermite interpolations in the two-boundary technique. In the case of Lagrange interpolation, the blending function $\alpha^i_{1,0}(\xi^i)$ satisfies the conditions $\alpha^i_{1,0}(0) = 1,\ \alpha^i_{1,0}(1) = 0$. Therefore, any monotonically decreasing function derived by applying the procedures described in Sect. 4.4 to the reference univariate functions can be used as the blending function $\alpha^i_{1,0}(\xi^i)$. Analogously, the blending function $\alpha^i_{2,0}(\xi^i)$ can be represented by any monotonically increasing mapping based on one of the standard local contraction functions $x_i(\varphi, \epsilon)$. The blending functions $\alpha^i_{i,1}(\xi^i)$ for Hermite interpolations can also use these standard transformations through applying the operation described by (5.37) to the blending functions $\alpha^i_{1,0}(\xi^i)$. By choosing the proper functions, one has an opportunity to construct intermediate transformations that provide adequate grid clustering in the zones where it is necessary.

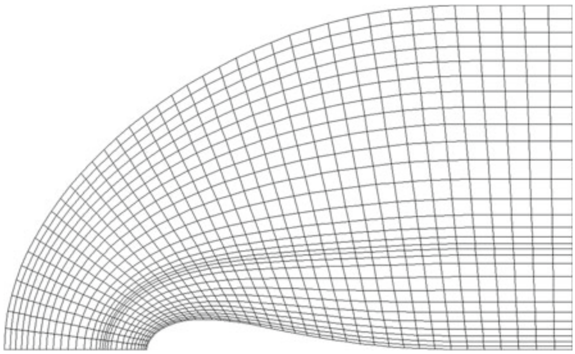

Fig. 5.4 Quadrangular adaptive grid

One example of a two-dimensional adaptive quadrangular grid, with the intermediate grid generated in such a manner through the basic stretching functions, is presented in Fig. 5.4.

5.6 Transfinite Interpolation from Triangles and Tetrahedrons

The formulas of transfinite interpolation define a coordinate transformation from the unit cube Ξ^3 (the square Ξ^2 in two dimensions and line Ξ^1 in one dimension) onto a physical domain X^3 (or X^2 or X^1). The application of this interpolation may lead to singularities of the type pertaining to polar transformations when any boundary segment of the physical domain, corresponding to a boundary segment of the computational domain, is contracted into a point. An example is when the boundary of a physical two-dimensional domain X^2 is composed of three smooth segments, as shown in Fig. 5.5. One way to treat such regions is to use coordinate transformations from triangular computational domains in two dimensions and tetrahedral domains in three dimensions. It can be seen that the transfinite interpolation approach can be modified to generate triangular or tetrahedral grids by mapping a standard triangular or tetrahedral domain, respectively. The formulation of a transfinite interpolation to obtain these transformations from the standard unit tetrahedron (triangle in two dimensions) is based on the composition of an operation of scaling (stretching) the coordinates to deform the tetrahedron into the unit cube Ξ^3 and an algebraic transformation constructed by the equations given above.

This procedure is readily clarified in two dimensions by the scheme depicted in Fig. 5.5. Suppose that the boundary segments AB, BC, and CD of the unit triangle T^2 are mapped onto the corresponding boundary segments AB, BC, and CD of the domain X^2. Then, in this procedure, the standard triangle T^2 with a uniform triangular grid is expanded to a square by a deformation $\boldsymbol{\xi}(\boldsymbol{t})$ uniformly stretching each horizontal line of the triangle to make it a rectangle, and afterwards, the rectangle

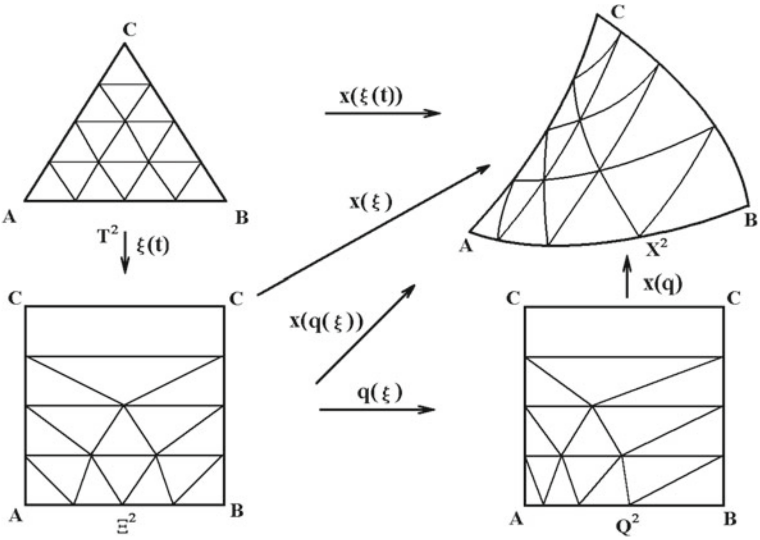

Fig. 5.5 Scheme for gridding triangular curvilinear domains with triangles

is uniformly stretched in the vertical direction to make it the unit square Ξ^2, as shown in Fig. 5.5. This operation is the inverse of the contraction $\boldsymbol{t}(\boldsymbol{\xi})$ of the square along the horizontal and vertical lines to transform it into the triangle. As a result, we obtain a square Ξ^2 with triangular cells on all horizontal levels except the top one. The number of these cells in each horizontal band reduces from the lower levels to the upper ones. The top level consists of one rectangular cell. With this deformation of T^2, the transformation between the boundaries of T^2 and X^2 generates the transformation

$$\boldsymbol{x}(\boldsymbol{\xi}) : \partial\Xi^2 \to \partial X^2 ,$$

which is the composition of $\boldsymbol{t}(\boldsymbol{\xi})$ and the assumed mapping of the boundary of T^2 onto the boundary of X^2. This boundary transformation maps the top segment of Ξ^2 onto the point C in X^2. Now, applying the formulas of transfinite interpolation to a square Ξ^2 with such grid cells, and the specified boundary transformation, one generates the algebraic transformation

$$\boldsymbol{x}(\boldsymbol{\xi}) : \Xi^2 \to X^2$$

and consequently

$$\boldsymbol{x}[\boldsymbol{\xi}(\boldsymbol{t})] : T^2 \to X^2$$

from the triangle to the physical region X^2 with the prescribed values of the transformation at the boundary segments of the triangle. Note that the composition $\boldsymbol{x}[\boldsymbol{\xi}(\boldsymbol{t})]$

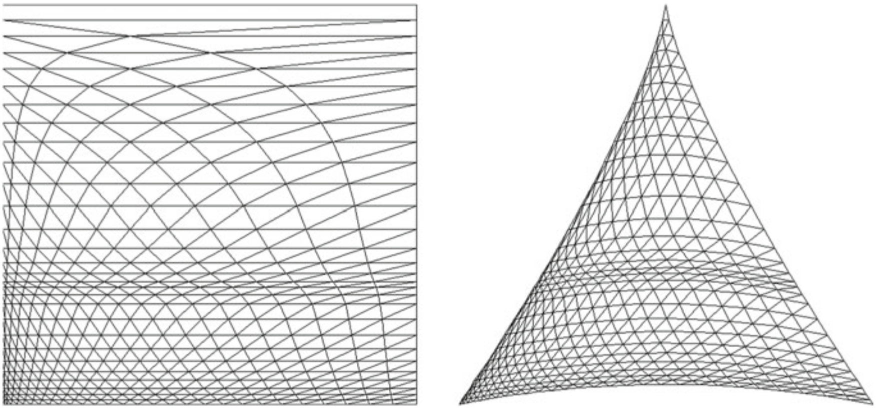

Fig. 5.6 Example of an adaptive algebraic triangular grid (*right*) and the corresponding grid on the intermediate domain (*left*) generated by the algebraic method

is continuous as the upper segment of Ξ^2 is transformed by $\boldsymbol{x}(\boldsymbol{\xi})$ onto one point C in X^2.

In fact, such a triangular grid in the physical domain can be generated directly by mapping the nonuniform grid constructed in the unit square Ξ^2 as described above onto X^2 with a standard algebraic coordinate transformation defined by transfinite interpolation.

The generation of grids by this approach is very well justified for regions shaped like curvilinear triangles, i.e. their boundaries are composed of three smooth curves intersecting at angles θ less than π. By dividing an arbitrary domain into triangular curvilinear domains, one can generate a composite triangular grid in the entire domain through the procedure described above.

An analogous procedure using transfinite interpolation is readily formulated for generating tetrahedral grids in regions with shapes similar to that of a tetrahedron.

The approach for generating triangular or tetrahedral meshes described above can be extended to include grid adaptation by adding to the scheme an intermediate domain and intermediate transformation $\boldsymbol{q}(\boldsymbol{\xi})$, as illustrated in Fig. 5.6, and special blending functions, as in the case of generating hexahedral (or quadrilateral) grids. Here, an adaptive triangular grid is generated through the composition of the transformations $\boldsymbol{q}(\boldsymbol{\xi})$ and $\boldsymbol{x}(\boldsymbol{q})$, where $\boldsymbol{q}(\boldsymbol{\xi})$ is an intermediate mapping providing grid adaptation and $\boldsymbol{x}(\boldsymbol{q})$ is an algebraic transformation.

Note that the procedure described above for generating triangular grids (tetrahedral and prismatic ones in three dimensions) can be realized analogously in other techniques based on coordinate transformations from the unit cube.

5.7 Drag and Sweeping Methods

Through the methods described above, the interior grid nodes are obtained by interpolating the grid points from all boundary segments. For domains of relatively simple geometries, the grid nodes may be obtained by dragging or sweeping the grid nodes of one boundary section. Suppose these nodes are $\boldsymbol{x}_i$, $i = 1, \ldots, N$; then, both the interior nodes and another boundary grid nodes $\boldsymbol{x}_i^j$, $i = 1, \ldots, N,\ j = 1, \ldots, M$, may be defined by the formula

$$\boldsymbol{x}_i^j = \boldsymbol{x}_i^{j-1} + \boldsymbol{v}_i^j, \quad \boldsymbol{x}_i^0 = \boldsymbol{x}_i,$$

with specified incremental vectors $\boldsymbol{v}_i^j$. If the vectors $\boldsymbol{v}_i^j$ are constant, then this algorithm is referred to as a drag method. However, the vectors $\boldsymbol{v}_i^j$ may be different, in which case the approach is referred to as a sweeping method. These methods were developed at the early stage of mesh generation by Park and Washam (1979).

5.8 Comments

The standard formulas of multivariate transfinite interpolation using Boolean operations were described by Gordon (1969, 1971), although a two-dimensional interpolation formula with the simplest blending functions for the construction of the boundaries of hexahedral patches from CAD data was proposed by Coons (1967) and Ahuja and Coons (1968). The construction of coordinate transformations through the formulas of transfinite interpolation was formulated by Gordon and Hall (1973) and Gordon and Thiel (1982). The Hermite interpolation was presented by Smith (1982).

The multisurface method was described by Eiseman (1980) and was, in its original form, a univariate formula for grid generation based on the specification of two boundary surfaces and an arbitrary number of interior control surfaces. The blending functions were implicitly derived from global and/or local interpolants which result from an expression for the tangential derivative spanning between the exterior boundary surfaces. The multisurface transformation can be described in the context of transfinite interpolation.

A two-boundary technique was introduced by Smith (1981). It is based on the description of two opposite boundary surfaces, tangential derivatives on the boundary surfaces which are used to compute normal derivatives, and Hermite cubic blending functions.

The construction of some special blending functions aimed at grid clustering at boundaries was performed by Eriksson (1982) and Smith and Eriksson (1987). A detailed description of various forms of blending functions with the help of splines was presented in a monograph by Thompson et al. (1985).

The procedures described above for generating smooth blending functions and algebraic triangulations were developed by Liseikin (1999).

References

Ahuja, D. V., & Coons, S. A. (1968). Geometry for construction and display. *IBM Systems Journal*, *7*, 188–205.

Coons, S. A. (1967). Surfaces for computer aided design of space forms. Project MAC, Design Division Department of Mechanical Engineering, MII, 1964; Revised to MAC-TR-41, June, 1967.

Eiseman, P. R. (1980). Geometric methods in computational fluid dynamics. ICASE Report 80-11 and Von Karman Institute for Fluid Dynamics Lecture Series Notes.

Eriksson, L. E. (1982). Generation of boundary-conforming grids around wing-body configurations using transfinite interpolation. *AIAA Journal*, *20*, 1313–1320.

Gordon, W. J. (1969). Distributive lattices and the approximation of multivatiate functions. In I. J. Shwenberg (Ed.), *Symposium on approximation with special emphasis on spline functions* (pp. 223–277). Madison: Academic Press.

Gordon, W. J. (1971). Blending-function methods of bivariate and multivariate interpolation and approximation. *SIAM Journal on Numerical Analysis*, *8*, 158–177.

Gordon, W. J., & Hall, C. A. (1973). Construction of curvilinear coordinate systems and applications to mesh generation. *International Journal for Numerical Methods in Engineering*, *7*, 461–477.

Gordon, W. J., & Thiel, L. C. (1982). Transfinite mappings and their application to grid generation. In J. F. Thompson (Ed.), *Numerical grid generation* (pp. 171–192). New York: North-Holland.

Liseikin, V. D. (1999). *Grid generation methods*. Berlin: Springer.

Park, S., & Washam, C. J. (1979). Drag method as a finite element mesh generation sheme. *Composite Structures*, *10*(1–2), 343–346.

Smith, R. E. (1981). Two-boundary grid generation for the solution of the three-dimensional Navier–Stokes equations. NASA TM-83123.

Smith, R. E. (1982). Algebraic grid generation. In J. F. Thompson (Ed.), *Numerical grid generation* (pp. 137–170). New York: North-Holland.

Smith, R. E., & Eriksson, L. E. (1987). Algebraic grid generation. *Computer Methods in Applied Mechanics and Engineering*, *64*, 285–300.

Thompson, J. F., Warsi, Z. U. A., & Mastin, C. W. (1985). *Numerical grid generation foundations and applications*. New York: North-Holland.