REMARKS ON GENERATION OF THE ORTHOGONAL STRUCTURED GRIDS

S.I. Martynenko   Martynenko@ciam.ru
                                      Martynenko@icp.ac.ru

**Central Institute of Aviation Motors,
Moscow, Russian Federation
Institute of Problems of Chemical Physics, Russian Academy of Sciences,
Chernogolovka, Moscow Region, Russian Federation**

Abstract

Grid generation techniques have contributed significantly toward the application of mathematical modeling in large-scale engineering problems. The structured grids have the advantage that very robust and parallel computational algorithms have been proposed for solving (initial-)boundary value problems. Orthogonal grids make it possible to simplify an approximation of the differential equations and to increase computation accuracy. Opportunity of the orthogonal structured grid generation for solving two- and three-dimensional (initial-)boundary value problems is analyzed in the article in assumption that isolines or isosurfaces of d (=2,3) functions form this grid. Condition of the isolines/isosurfaces orthogonality is used for formulation of the boundary value problems, the solutions of which will be form the orthogonal grid. A differential substitution is proposed to formulate the boundary value problems directly from the orthogonality condition of the grid. The substitution leads to the general partial differential equations with undetermined coefficients. In the two-dimensional case, it is shown that the orthogonal grid generation is equivalent to the solution of partial differential equations of either elliptic or hyperbolic type. In three-dimensional domains, an orthogonal grid can be generated only in special cases. The obtained results are useful for mathematical modeling of the complex physicochemical processes in the technical devices

Keywords

Introduction. The rapid development of computers and mathematical modeling methods had a strong influence on mechanical engineering: a significant amount of experimental research in the new technology development can be replaced by mathematical modeling results. This significantly reduces the time and cost of research and development work. However, in the early 1990s, it became clear that the greatest effect from the use of mathematical modeling methods in mechanical engineering was achieved using specialized software built on the "black box" principle. The user of such software is busy only with setting the task (i.e., setting the geometry region, initial and boundary conditions, choosing materials and equations describing processes, etc.) analyzing the results, and computational experiment details are not available to him. Software packages, arranged according to the "black box" principle, are a powerful tool in the hands of an engineer that will allow him to concentrate on optimizing work processes occurring in developed technical devices, rather than wasting time writing and debugging computer programs.

All the problems associated with the creation of modern mathematical modeling software can be divided into three groups: 1) "physical"; 2) "mathematical"; 3) "computer". "Physical" problems are associated with the mathematical description difficulty of complex physicochemical processes, such as hydrodynamics and heat and mass transfer processes in multiphase reactive media, turbulent transfer, etc. "Computer" problems arise due to compatibility difficulties of various and quickly updated software and hardware, and "mathematical" problems are associated with the formalizing complexity of computational experiment main stages: building a computational grid, approximating the fundamental (non-)linear (integro-)differential equations and the effective solution of ill-conditioned systems of high order linear algebraic equations on a sequential or parallel computer. The complexity of "physical" problems is due to the diversity of the processes being modeled and their mathematical description depth, the "mathematical" problems are the result of insufficient knowledge of the underlying (non-)linear (integro-)differential equations and insufficiently developed methods for the numerical solution of mathematical physics equations.

Currently, several software packages, such as *STAR-CD*, *CFX*, *FLUENT*, *PHOENICS*, *SIGMAFLOW* and mesh generators, such as *ICEM CFD*, *Gambit*, *NETGEN*, etc., are developed and widely used for engineering calculations. The listed programs are still very far from perfect, although they can significantly reduce the mathematical modeling disadvantages.

Computational grid characteristics strongly influence the computation time and the obtained solution accuracy. On the one hand, in a region with a comp-

lex geometry, it is easier to construct an unstructured grid, but it is more difficult to solve grid equations (in modern packages algebraic multigrid methods are most often used [1], which do not use information about the grid). On the other hand, it is more difficult to construct a regular grid in the same area, but it is easier to solve grid equations (for example, to apply geometric multigrid methods [1] that use grid information). In modern software packages, it is often the grid type that determines the choice of an iterative method for solving (linearized) grid equations obtained as a result of approximation of the original (non)linear (integro-)differential equations of a mathematical model.

Previously it was assumed that the Seidel method is completely unsuitable for numerical solution of boundary value problems, since it requires performing $O(N^2)$ arithmetic operations in the two-dimensional case, where N is the number of unknowns vector components of the resulting linear algebraic equations system (for a single equation, N is the number of grid nodes) [2]. The use of upper relaxation method makes it possible to reduce arithmetic operations cost to $O(N^{3/2})$ arithmetic operations [2], however, the problem of finding the relaxation parameter optimal value immediately arises [3]. The complexity of classical multigrid methods is even lower. These methods cannot be represented as a unified algorithm, but with optimal adaptation of their problem-dependent components, have a minimal (unimprovable) complexity $O(N)$ of arithmetic operations. In other words, the lower the computational algorithm complexity, the lower the computation formalization level and the more difficult it is to use such an algorithm in software built on the "black box" principle. It is obvious that the fundamental idea of the first multigrid algorithm author, R.P. Fedorenko, described in [4], has a powerful potential, but the classical multigrid methods in their present form are difficult to use in "black box" software, due to problems associated with optimizing their problem-dependent components. In addition, the classical multigrid methods do not allow to effectively parallelizing smoothing iterations, especially on coarse grids [1].

In the early 1990s, it was done the attempt to realize a different form of R.P. Fedorenko fundamental idea, for this purpose, the basic multigrid principle was used in the single-mesh algorithm [5–7]. The developed iterative method for the numerical solution of boundary value problems is called *robust multigrid technique* (RMT). It was shown that if the computational grid is structured (that is, regular, but generating a special sequence of subgrids — a multigrid structure), the complexity of the Seidel method can be reduced to $O(N \lg N)$ arithmetic operations without involving problem-dependent components. In addition, it was shown in [5, 6, 8], that the maximum parallelization efficiency

(as compared to an unimproved sequential algorithm, which has the complexity of $O(N)$ arithmetic operations) will be independent of the number of processors used. Thus, RMT has a minimum of problem-dependent components, such as the Seidel single-grid method (criterion for stopping iterations, the lower relaxation parameter for nonlinear problems and the ordering of unknowns for anisotropic problems), and close to the optimal $O(N \lg N)$ arithmetic operations comparable to the classical multigrid methods. The most serious limitation for RMT is the structured computational grid requirement.

Currently, there are many ways to build grids, but all known grid generation algorithms have a low level of formalization and parallelism, and, as a rule, require expert evaluation of the built mesh quality [9, 10]. Often the grid should be changed in the process of solving mathematical model equations, adapting to the features of the solution and (or) changing the region. The development of a robust algorithm for constructing adaptive grids in multiply connected areas is one of the key problems of the mathematical apparatus for modern software tools designed to solve engineering problems.

In terms of practical application, the key problem is the development of a robust structured grids generator in areas with complex geometry, which can later be used to numerically solve boundary and initial-boundary problems with the help of RMT. Of particular theoretical and practical interest are orthogonal grids, which make it possible to significantly simplify the approximation of the original (non)linear (integro-)differential equations of a mathematical model.

The *purpose* of this work is to study the possibility of constructing orthogonal computational grids in two- and three-dimensional areas.

Governing equations for orthogonal computational grids. Consider the three-dimensional case and assume that the computational grid is formed by isosurfaces $U(x, y, z) = \text{const}$, $V(x, y, z) = \text{const}$ and $W(x, y, z) = \text{const}$, where U, V and W are some functions. Let (x_0, y_0, z_0) be some intersection point of three isosurfaces. We construct tangent planes to isosurfaces $U(x, y, z) = \text{const}$, $V(x, y, z) = \text{const}$ and $W(x, y, z) = \text{const}$, at a point (x_0, y_0, z_0) and require that these planes be mutually perpendicular. Due to the fact that the point (x_0, y_0, z_0) is chosen arbitrarily, the condition of mutual perpendicularity of the tangent planes will take the form

$$U'_x V'_x + U'_y V'_y + U'_z V'_z = 0; \tag{1a}$$

$$V'_x W'_x + V'_y W'_y + V'_z W'_z = 0; \tag{1b}$$

$$W'_x U'_x + W'_y U'_y + W'_z U'_z = 0. \tag{1c}$$

A computational grid, formed by isosurfaces $U(x, y, z) = \text{const}$, $V(x, y, z) = \text{const}$ and $W(x, y, z) = \text{const}$, is called orthogonal if the functions U, V and W satisfy the orthogonality condition (1a)–(1c). The task is to build such a system of differential equations, the solution of which are functions U, V and W satisfying the orthogonality condition (1a)–(1c). Then the generation of an orthogonal grid in a certain region will be reduced to the solution of this system and the construction of isosurfaces $U(x, y, z) = \text{const}$, $V(x, y, z) = \text{const}$ and $W(x, y, z) = \text{const}$.

We represent functions U, V and W in the form

$$
\begin{aligned}
U = a_{11}\Psi'_x + a_{12}\Psi'_y + a_{13}\Psi'_z + b_{11}\Phi'_x + b_{12}\Phi'_y + \\
+ b_{13}\Phi'_z + c_{11}\Lambda'_x + c_{12}\Lambda'_y + c_{13}\Lambda'_z;
\end{aligned}
\tag{2a}
$$

$$
\begin{aligned}
V = a_{21}\Psi'_x + a_{22}\Psi'_y + a_{23}\Psi'_z + b_{21}\Phi'_x + b_{22}\Phi'_y + \\
+ b_{23}\Phi'_z + c_{21}\Lambda'_x + c_{22}\Lambda'_y + c_{23}\Lambda'_z;
\end{aligned}
\tag{2b}
$$

$$
\begin{aligned}
W = a_{31}\Psi'_x + a_{32}\Psi'_y + a_{33}\Psi'_z + b_{31}\Phi'_x + b_{32}\Phi'_y + \\
+ b_{33}\Phi'_z + c_{31}\Lambda'_x + c_{32}\Lambda'_y + c_{33}\Lambda'_z,
\end{aligned}
\tag{2c}
$$

where $a_{ij}, b_{ij}, c_{ij}, i, j = 1, 2, 3$, are some coefficients; Ψ, Φ and Λ are some functions. For convenience of transformations, we introduce the notation

$$
\begin{aligned}
x = \varsigma_1, \quad y = \varsigma_2, \quad z = \varsigma_3, \\
U = \Theta_1, \quad V = \Theta_2, \quad W = \Theta_3.
\end{aligned}
$$

Then equations (2a)–(2c) take the form

$$
\Theta_i = \sum_{n=1}^{3} \left(a_{in}\Psi'_{\varsigma_n} + b_{in}\Phi'_{\varsigma_n} + c_{in}\Lambda'_{\varsigma_n} \right), \quad i = 1, 2, 3,
\tag{3}
$$

i.e., values $i = 1, 2, 3$ correspond equations (2a), (2b) and (2c). Differentiation of (3) leads to

$$
\left(\Theta_i \right)'_{\varsigma_k} = \sum_{n=1}^{3} \left(a_{in}\Psi''_{\varsigma_n \varsigma_k} + b_{in}\Phi''_{\varsigma_n \varsigma_k} + c_{in}\Lambda''_{\varsigma_n \varsigma_k} \right), \quad i, k = 1, 2, 3;
$$

$$
\left(\Theta_j \right)'_{\varsigma_k} = \sum_{l=1}^{3} \left(a_{jl}\Psi''_{\varsigma_l \varsigma_k} + b_{jl}\Phi''_{\varsigma_l \varsigma_k} + c_{jl}\Lambda''_{\varsigma_l \varsigma_k} \right), \quad j, k = 1, 2, 3.
$$

Taking into account the accepted notation, the condition of the grid orthogonality (1) takes the form

$$0 = \sum_{k=1}^{3} \left(\Theta_i\right)'_{\varsigma k} \left(\Theta_j\right)'_{\varsigma k} = \sum_{k,n,l=1}^{3} \left(a_{in}a_{jl}\Psi''_{\varsigma n \varsigma k}\Psi''_{\varsigma l \varsigma k} + a_{in}b_{jl}\Psi''_{\varsigma n \varsigma k}\Phi''_{\varsigma l \varsigma k} + \right.$$
$$+ \, a_{in}c_{jl}\Psi''_{\varsigma n \varsigma k}\Lambda''_{\varsigma l \varsigma k} + a_{jl}b_{in}\Phi''_{\varsigma n \varsigma k}\Psi''_{\varsigma l \varsigma k} + b_{in}b_{jl}\Phi''_{\varsigma n \varsigma k}\Phi''_{\varsigma l \varsigma k} +$$
$$+ \, b_{in}c_{jl}\Phi''_{\varsigma n \varsigma k}\Lambda''_{\varsigma l \varsigma k} + a_{jl}c_{in}\Lambda''_{\varsigma n \varsigma k} + \Psi''_{\varsigma l \varsigma k} +$$
$$\left. + \, b_{jl}c_{in}\Lambda''_{\varsigma n \varsigma k}\Phi''_{\varsigma l \varsigma k} + c_{in}c_{jl}\Lambda''_{\varsigma n \varsigma k}\Lambda''_{\varsigma l \varsigma k}\right), \tag{4}$$

where $i=1$, $j=2$ correspond to (1a), $i=2$, $j=3$ correspond to (1b) and $i=3$, $j=1$ correspond to (1c). In view of the identity

$$\sum_{n,l=1}^{3} a_{in}b_{jl}\Psi''_{\varsigma n \varsigma k}\Phi''_{\varsigma l \varsigma k} + \sum_{n,l=1}^{3} a_{jl}b_{in}\Phi''_{\varsigma n \varsigma k}\Psi''_{\varsigma l \varsigma k} =$$
$$= \sum_{n,l=1}^{3} \left(a_{in}b_{jl} + a_{jn}b_{il}\right)\Psi''_{\varsigma n \varsigma k}\Phi''_{\varsigma l \varsigma k}$$

individual terms in (4) can be combined, then

$$0 = \sum_{k,n,l=1}^{3} \left(a_{in}a_{jl}\Psi''_{\varsigma n \varsigma k}\Psi''_{\varsigma l \varsigma k} + \left(a_{in}b_{jl} + a_{jn}b_{il}\right)\Psi''_{\varsigma n \varsigma k}\Phi''_{\varsigma l \varsigma k} + \right.$$
$$+ \left(a_{in}c_{jl} + a_{jn}c_{il}\right)\Psi''_{\varsigma n \varsigma k}\Lambda''_{\varsigma l \varsigma k} + b_{in}b_{jl}\Phi''_{\varsigma n \varsigma k}\Phi''_{\varsigma l \varsigma k} +$$
$$\left. + \left(b_{in}c_{jl} + b_{jn}c_{il}\right)\Phi''_{\varsigma n \varsigma k}\Lambda''_{\varsigma l \varsigma k} + c_{in}c_{jl}\Lambda''_{\varsigma n \varsigma k}\Lambda''_{\varsigma l \varsigma k}\right). \tag{5}$$

Now we transform the "diagonal" terms in (5):

$$\sum_{k,n,l=1}^{3} a_{in}a_{jl}\Psi''_{\varsigma n \varsigma k}\Psi''_{\varsigma l \varsigma k} = \sum_{k,m=1}^{3} a_{im}a_{jm}\left(\Psi''_{\varsigma m \varsigma k}\right)^2 +$$
$$+ \sum_{k=1}^{3}\sum_{n=1}^{2}\sum_{l=n+1}^{3} \left(a_{in}a_{jl} + a_{il}a_{jn}\right)\Psi''_{\varsigma n \varsigma k}\Psi''_{\varsigma l \varsigma k}; \tag{6a}$$

$$\sum_{k,n,l=1}^{3} b_{in}b_{jl}\Phi''_{\varsigma n \varsigma k}\Phi''_{\varsigma l \varsigma k} = \sum_{k,m=1}^{3} b_{im}b_{jm}\left(\Phi''_{\varsigma m \varsigma k}\right)^2 +$$
$$+ \sum_{k=1}^{3}\sum_{n=1}^{2}\sum_{l=n+1}^{3} \left(b_{in}b_{jl} + b_{il}b_{jn}\right)\Phi''_{\varsigma n \varsigma k}\Phi''_{\varsigma l \varsigma k}; \tag{6b}$$

$$\sum_{k,n,l=1}^{3} c_{in}c_{jl}\Lambda''_{\varsigma n \varsigma k}\Lambda''_{\varsigma l \varsigma k} = \sum_{k,m=1}^{3} c_{im}c_{jm}\left(\Lambda''_{\varsigma m \varsigma k}\right)^2 +$$
$$+ \sum_{k=1}^{3}\sum_{n=1}^{2}\sum_{l=n+1}^{3} \left(c_{in}c_{jl} + c_{il}c_{jn}\right)\Lambda''_{\varsigma n \varsigma k}\Lambda''_{\varsigma l \varsigma k}. \tag{6c}$$

We note that the first terms in the right-hand side of (6a)–(6c) can be converted as follows:

$$\sum_{k,m=1}^{3} a_{im}a_{jm}\left(\Psi''_{\varsigma_m\varsigma_k}\right)^2 = \sum_{k=1}^{3} a_{ik}a_{jk}\left(\Psi''_{\varsigma_k\varsigma_k}\right)^2 +$$
$$+ \sum_{n=1}^{2}\sum_{l=n+1}^{3}\left(a_{in}a_{jn}+a_{il}a_{jl}\right)\left(\Psi''_{\varsigma_l\varsigma_k}\right)^2; \tag{7a}$$

$$\sum_{k,m=1}^{3} b_{im}b_{jm}\left(\Phi''_{\varsigma_m\varsigma_k}\right)^2 = \sum_{k=1}^{3} b_{ik}b_{jk}\left(\Phi''_{\varsigma_k\varsigma_k}\right)^2 +$$
$$+ \sum_{n=1}^{2}\sum_{l=n+1}^{3}\left(b_{in}b_{jn}+b_{il}b_{jl}\right)\left(\Phi''_{\varsigma_l\varsigma_k}\right)^2; \tag{7b}$$

$$\sum_{k,m=1}^{3} c_{im}c_{jm}\left(\Lambda''_{\varsigma_m\varsigma_k}\right)^2 = \sum_{k=1}^{3} c_{ik}c_{jk}\left(\Lambda''_{\varsigma_k\varsigma_k}\right)^2 +$$
$$+ \sum_{n=1}^{2}\sum_{l=n+1}^{3}\left(c_{in}c_{jn}+c_{il}c_{jl}\right)\left(\Lambda''_{\varsigma_l\varsigma_k}\right)^2. \tag{7c}$$

Substituting (7a)–(7c) into (6a)–(6c), and (6a)–(6c) into (5), we obtain the final equation

$$0 = \sum_{k=1}^{3} a_{ik}a_{jk}\left(\Psi''_{\varsigma_k\varsigma_k}\right)^2 + \sum_{n=1}^{2}\sum_{l=n+1}^{3}\left(a_{in}a_{jn}+a_{il}a_{jl}\right)\left(\Psi''_{\varsigma_l\varsigma_n}\right)^2 +$$
$$+ \sum_{k=1}^{3}\sum_{n=1}^{2}\sum_{l=n+1}^{3}\left(a_{in}a_{jl}+a_{il}a_{jn}\right)\Psi''_{\varsigma_n\varsigma_k}\Psi''_{\varsigma_l\varsigma_k} +$$
$$+ \sum_{k=1}^{3} b_{ik}b_{jk}\left(\Phi''_{\varsigma_k\varsigma_k}\right)^2 + \sum_{n=1}^{2}\sum_{l=n+1}^{3}\left(b_{in}b_{jn}+b_{il}b_{jl}\right)\left(\Phi''_{\varsigma_l\varsigma_n}\right)^2 +$$
$$+ \sum_{k=1}^{3}\sum_{n=1}^{2}\sum_{l=n+1}^{3}\left(b_{in}b_{jl}+b_{il}b_{jn}\right)\Phi''_{\varsigma_n\varsigma_k}\Phi''_{\varsigma_l\varsigma_k} + \sum_{k=1}^{3} c_{ik}c_{jk}\left(\Lambda''_{\varsigma_k\varsigma_k}\right)^2 +$$
$$+ \sum_{n=1}^{2}\sum_{l=n+1}^{3}\left(c_{in}c_{jn}+c_{il}c_{jl}\right)\left(\Lambda''_{\varsigma_l\varsigma_n}\right)^2 + \sum_{k=1}^{3}\sum_{n=1}^{2}\sum_{l=n+1}^{3}\left(c_{in}c_{jl}+c_{il}c_{jn}\right)\Lambda''_{\varsigma_n\varsigma_k}\Lambda''_{\varsigma_l\varsigma_k} +$$
$$+ \sum_{k,n,l=1}^{3}\left(\left(a_{in}b_{jl}+a_{jn}b_{il}\right)\Psi''_{\varsigma_n\varsigma_k}\Phi''_{\varsigma_l\varsigma_k} + \left(a_{in}c_{jl}+a_{jn}c_{il}\right)\Psi''_{\varsigma_n\varsigma_k}\Lambda''_{\varsigma_l\varsigma_k} +\right.$$
$$\left. + \left(b_{in}c_{jl}+b_{jn}c_{il}\right)\Phi''_{\varsigma_n\varsigma_k}\Lambda''_{\varsigma_l\varsigma_k}\right). \tag{8}$$

Recall that expression (8) contains 27 indefinite coefficients by choosing which properly this equation can be significantly simplified. Next, we consider the two- and three-dimensional cases.

2D orthogonal grids. The two-dimensional case matches the set $i = 1$, $j = 2$, $\Phi = \Lambda = 0$ and $(\cdot)'_z = 0$. Then (8) is reduced to

$$a_{11}a_{21}\left(\Psi''_{xx}\right)^2 + a_{12}a_{22}\left(\Psi''_{yy}\right)^2 + (a_{11}a_{21} + a_{12}a_{22})\left(\Psi''_{xy}\right)^2 +$$
$$+ (a_{11}a_{22} + a_{12}a_{21})\Psi''_{xx}\Psi''_{xy} + (a_{11}a_{22} + a_{12}a_{21})\Psi''_{xy}\Psi''_{yy} = 0.$$

We assume $a_{11} = a_{12} = a_{21} = -a_{22} = 1$, we arrive at the following statement.

Statement 1. *Let the function* $\Psi(x, y)$ *satisfies equality*

$$\left(\Psi''_{xx}\right)^2 = \left(\Psi''_{yy}\right)^2. \tag{9}$$

Then the isolines of the functions U *and* V, *defined as*

$$U = \Psi'_x + \Psi'_y; \tag{10}$$

$$V = \Psi'_x - \Psi'_y, \tag{11}$$

form an orthogonal grid.

As $0 = \left(\Psi''_{xx}\right)^2 - \left(\Psi''_{yy}\right)^2 = \left(\Psi''_{xx} + \Psi''_{yy}\right)\left(\Psi''_{xx} - \Psi''_{yy}\right)$, two special cases are possible.

Statement 2. *Let the function* $\Psi(x, y)$ *satisfy the Laplace equation*

$$\Psi''_{xx} + \Psi''_{yy} = 0. \tag{12}$$

Then the isolines of the functions U *and* V *defined as (10) and (11) form an orthogonal grid, and these functions satisfy the Laplace equation and the Cauchy — Riemann conditions.*

◄ From (10) and (11) follows

$$U + V = 2\Psi'_x; \tag{13}$$

$$U - V = 2\Psi'_y. \tag{14}$$

Differentiating (13) by x and differentiating (14) by y then summing them up considering (12), we get

$$U'_x + V'_x + U'_y - V'_y = 2\left(\Psi''_{xx} + \Psi''_{yy}\right) = 0. \tag{15}$$

Likewise differentiating (13) by y and differentiating (14) by x and taking into account the theorem on independence of a mixed derivative of the differentiation order, we obtain

$$U'_y + V'_y = U'_x - V'_x. \tag{16}$$

Summation and subtraction (15) and (16) leads to the Cauchy — Riemann conditions $U'_x - V'_y = 0$ and $U'_y + V'_x = 0,$ which lead to the Laplace equations $U''_{xx} + U''_{yy} = 0$ and $V''_{xx} + V''_{yy} = 0.$ In this case, a general view of the functions U and V is $U(x,y) = f_1(x+iy) + f_2(x-iy)$ and $V(x,y) = g_1(x+iy) + g_2(x-iy).$ ▶

Statement 3. *Let a function* $\Psi(x,y)$ *satisfy an equation of hyperbolic type*

$$\Psi''_{xx} - \Psi''_{yy} = 0. \tag{17}$$

Then the isolines of the functions U and V, defined as (10) and (11) form an orthogonal grid, and the functions satisfy an equation of hyperbolic type.

◀ Analogically the differentiating (13) and (14) based on (17) leads to $U'_x - U'_y = 0$ and $V'_x + V'_y = 0,$ which implies that $U''_{xx} - U''_{yy} = 0$ and $V''_{xx} - V''_{yy} = 0.$ In this case, a general view of the functions U and V is $U(x,y) = f_1(x+y) + f_2(x-y)$ and $V(x,y) = g_1(x+y) + g_2(x-y).$ ▶

In the two-dimensional case, orthogonal grids are most often constructed using the Laplace equations, and the grid is thickened by the appropriate construction of contour lines.

3D orthogonal grids. Let us return to (8) and analyze the possibility of orthogonal grids constructing in the three-dimensional case. At first, we require that $a_{ik}a_{jk} \neq 0$ at $a_{in}a_{jn} + a_{il}a_{jl} = 0.$ We have previously considered two-dimensional grids as a special case of three-dimensional grids. Now three-dimensional grids will be considered as a generalization of two-dimensional grids, i.e., we assume that some of the coefficients are already known from the analysis of the two-dimensional case, in particular $a_{11} = a_{12} = a_{21} = -a_{22} = 1.$ Taking into account the symmetry of the transformations, expressions (2a)–(2c) take the form

$$U = \Psi'_x + \Psi'_y + a_{13}\Psi'_z + b_{11}\Phi'_x + b_{12}\Phi'_y + b_{13}\Phi'_z + \Lambda'_x + c_{12}\Lambda'_y + \Lambda'_z;$$
$$V = \Psi'_x - \Psi'_y + a_{23}\Psi'_z + b_{21}\Phi'_x + \Phi'_y + \Phi'_z + c_{21}\Lambda'_x + c_{22}\Lambda'_y + c_{23}\Lambda'_z;$$
$$W = a_{31}\Psi'_x + a_{32}\Psi'_y + a_{33}\Psi'_z + b_{31}\Phi'_x + \Phi'_y - \Phi'_z + \Lambda'_x + c_{32}\Lambda'_y - \Lambda'_z.$$

Condition $a_{in}a_{jn} + a_{il}a_{jl} = 0$ for $i = 1, j = 2$ (i.e., equation (1a)) takes the form

$$a_{11}a_{21} + a_{12}a_{22} = 0, \quad n = 1, \quad l = 2;$$
$$a_{11}a_{21} + a_{13}a_{23} = 0, \quad n = 1, \quad l = 3;$$
$$a_{12}a_{22} + a_{13}a_{23} = 0, \quad n = 2, \quad l = 3.$$

If $a_{11} = a_{12} = a_{21} = -a_{22} = 1$, then the second and third equations are incompatible. It follows that the grid will be orthogonal only when $\Psi'_z = 0$. Further, it follows from (1b) that $a_{31} = a_{32} = 0$, i.e., the equations describing a three-dimensional orthogonal grid are

$$U(x, y) = \Psi'_x + \Psi'_y;$$
$$V(x, y) = \Psi'_x - \Psi'_y;$$
$$W(z) = \varphi(z).$$

Thus, in the three-dimensional case, an orthogonal grid can be constructed only in special cases:

1) if a three-dimensional region is formed by the movement of a certain surface along a guideline $\varphi(z)$ with preservation of the condition $\Psi'_z = 0$. We can give the following example: orthogonal grids in the simplest bodies — parallelepiped, cylinder, etc.;

2) if the three-dimensional region is axisymmetric, that is, it is formed by rotating the two-dimensional region.

Conclusion. It is shown that there are two methods for constructing orthogonal grids (based on partial differential equations of elliptic and hyperbolic type or in an equivalent formulation related to minimizing the functional) in two-dimensional domains and two particular cases (displacement and rotation of an orthogonal two-dimensional mesh) in three-dimensional domains. Equation (8) can also be used to construct structured grids that are close to orthogonal.

Translated by V. Shumaev

REFERENCES

[1] Trottenberg U., Oosterlee C.W., Schüller A. Multigrid. Academic Press, 2001.

[2] Marchuk G.I. Metody vychislitel'noy matematiki [Computational mathematics methods]. Moscow, Nauka Publ., 1989.

[3] Hageman L.A., Young D.M. Applied iterative methods. Academic Press, 1981.

[4] Fedorenko R.P. A relaxation method for solving elliptic difference equations. *USSR Comput. Math. & Math. Phys.*, 1962, vol. 1, iss. 4, pp. 1092–1096.
DOI: 10.1016/0041-5553(62)90031-9

[5] Martynenko S.I. Mnogosetochnaya tekhnologiya: teoriya i prilozheniya [Multigrid technology: theory and applications]. Moscow, Fizmatlit Publ., 2015.

[6] Martynenko S.I. Robust multigrid technique for Black-Box software. DeGruyter, Berlin, 2017.

[7] Martynenko S.I. Robust multigrid technique for solving partial differential equations on structured grids. *Vychislitel'nye metody i programmirovanie* [Numerical Methods and Programming], 2000, vol. 1, pp. 82–103 (in Russ.).

[8] Martynenko S.I., Volokhov V.M., Yanovskiy L.S. Parallel geometric multigrid. *Int. J. Comp. Sci. Math.*, 2016, vol. 7, no. 4, pp. 293–300. DOI: 10.1504/IJCSM.2016.078741

[9] Thompson J.F., Soni B.K., Weatherill N.P. Handbook of grid generation. CRC Press, 1998.

[10] Liseikin V.D. Grid generation methods. Springer, 1999.

[11] Liseykin V.D., Likhanova Yu.V., Shokin Yu.I. Raznostnye setki i koordinatnye preobrazovaniya dlya chislennogo resheniya singulyarno vozmushchennykh zadach [Difference grids and coordinate transforms for numerical solution of singularly perturbed problems]. Novosibirsk, Nauka Publ., 2007.

Martynenko S.I. — Dr. Sc. (Phys.-Math.), Researcher, Department of Special Engines and Chemmotology, Central Institute of Aviation Motors (Aviamotornaya ul. 2, Moscow, 111116 Russian Federation), Senior Researcher, Department of Combustion and Explosion, Institute of Problems of Chemical Physics, Russian Academy of Sciences (Academician Semenov prospekt 1, Chernogolovka, Moscow Region, 142432 Russian Federation).