Nationaal Lucht- en Ruimtevaartlaboratorium
National Aerospace Laboratory NLR

NLR TP 96735

Elleptic generation systems

S.P. Spekreijse

DOCUMENT CONTROL SHEET

	ORIGINATOR'S REF. **NLR TP 96735 U**		SECURITY CLASS. **Unclassified**

ORIGINATOR
National Aerospace Laboratory NLR, Amsterdam, The Netherlands

TITLE
Elleptic generation systems.

PUBLISHED
as a chapter with the same title in the Handbook of Grid Generation, eds. J.F. Thompson, N.P. Weatherill and B.K. Soni. The Handbook will be published by CRC PRESS, INC. The expected publication date is late 1997.

AUTHORS **S.P. Spekreijse**	DATE **961202**	pp **54**	ref **35**

DESCRIPTORS
Algorithms
Boundary conditions
Computational fluid dynamics
Elliptic differential equations
Grid generation (mathematics)
Harmonic functions
Laplace-Beltrami equation
Minimal surface
Multiblock grids
Orthogonality
Poisson equation

ABSTRACT

The grid generation systems of elliptic quasi-linear second-order partial differential equations are the familiar so-called Poission systems with control functions to be specified. In this chapter, a Poisson system is considered as a system of partial differential equations which the composition of a grid control map and the inverse of a harmonic map has to obey. The control functions in the Poisson system are then completely defined by the grid control map. Boundary conforming grids in physical space are computed by solving the Poisson system with control functions specified by a grid control map.
One of the main advantages of this approach is that the method is non-iterative. If an appropriate grid control map has been constructed then the corresponding grid control functions of the Poisson system are computed and their values remain unchanged during the solution of the Poisson system. Another advantage is that the construction of an appropriate grid control map can be considered as a numerical implementation of the constructive proof for the existence of the desired grid in physical space. If the grid control map is one-to-one then the composition of the grid controlmap and the inverse of the harmonic maps exist so that the solution of the Poisson system is well defined.
In two dimensions, boundary orthogonality is obtained by applying Dirichlet-Neumann boundary conditions for the harmonic map. In that case, the harmonic map is quasi-conformal. This property shows the relation with orthogonal grid generation.
The use of harmonic maps and grid control maps for surface grid generation is also shortly described. The two-dimensional Poisson systems can be directly extended to surface grid generation on minimal surfaces (soap films). The extension to volume grid generation is also given.
The construction of appropriate grid control maps such that the corresponding grid in physical space has desired properties is the main issue of this chapter. The chosen examples mainly concern simple well-defined geometries so that the reader is able to recompute the grids. However, the in this chapter presented elliptic grid generation methods have been implemented in ENGRID, NLR's multi-block grid generation code [20, 21, 22], and are nowadays used on a routinely basis to construct Euler or Navier-Stokes grids in blocks and block-faces with complex geometrical shapes.
The construction of appropriate grid control maps for 3
D domains is less far developed than for 2D domains and surfaces. Further investigation is expected in this direction.

Elliptic Generation Systems

S.P. Spekreijse,National Aerospace Laboratory NLR,The Netherlands

1 Introduction

Since the pioneering work of Thompson on elliptic grid generation it is known that systems of elliptic second-order partial differential equations produce the best possible grids in the sense of smoothness and grid point distribution. The grid generation systems of elliptic quasi-linear second-order partial differential equations are so-called Poisson systems with control functions to be specified. The secret of each "good" elliptic grid is the method to compute the control functions [3].

Originally Thompson and Warsi introduced the Poisson systems by considering a curvilinear coordinate system which satisfies a system of Laplace equations and which is transformed to another coordinate system [1, 2]. Then this new coordinate system satisfies a system of Poisson equations with control functions completely specified by the transformation between the two coordinate systems. However Thompson did not advocate to use this approach for grid generation. Instead he proposed to use the Poisson system with control functions specified directly rather than through a transformation [1]. Since then the general approach is to compute the control functions at the boundary and to interpolate them from the boundaries into the field [7, 8]. The standard approach used to achieve grid orthogonality and specified cell height on boundaries has been the iterative adjustment of the control functions in the Poisson systems, first introduced by Sorenson of NASA Ames in the GRAPE code in the 80s [5]. Various modifications of this basic concept have been introduced in several codes, and the general approach is now common [6, 7, 8]. Although successful, it appears that the method is not easy to apply in practice [4]. Even today, new modifications are proposed to improve the grid quality and to overcome numerical difficulties in solving the Poisson grid generation equations [6, 9, 16].

In this chapter we describe a useful alternative approach to specify the control functions. It is based on Thompson's and Warsi's original idea to define the control functions by a transformation. The transformation, which we call a grid control map, is a differentiable one-to-one mapping from computational space to parameter space. The independent variables of the parameter space are harmonic functions in physical space. The map from physical space to parameter space is called the harmonic map. The composition of the grid control map and the inverse of the harmonic map obeys the familiar Poisson systems with control functions completely defined by the grid control map. The construction of appropriate grid control maps such that the corresponding grid in physical space has desired properties is the main issue of this chapter.

One of the main advantages of this approach is that the method is non-iterative. If an appropriate grid control map has been constructed then the corresponding grid control functions of the Poisson system are computed and their values remain unchanged during the solution of the Poisson system. Picard iteration appears to be a simple and robust method to solve the Poisson system with fixed control functions.

Another advantage is that the construction of an appropriate grid control map can be considered as a numerical implementation of the constructive proof for the existence of the desired grid in physical space. If the grid control map is one-to-one then the composition of the grid control map and the inverse of the harmonic maps exist so that the solution of the Poisson system is well-defined.

This chapter is organized as follows. Section 2 concerns the two-dimensional case. Although published earlier [19], the 2D Poisson system together with the expressions to compute the control functions from the grid control map are given for completeness. The solution of the Poisson system by Picard iteration is shortly described. Section 2.3 describes methods to construct appropriate grid control maps. Boundary orthogonality is obtained by applying Dirichlet-Neumann boundary conditions for the harmonic map and by applying cubic Hermite interpolation in parameter space. In that case, the harmonic map is quasi-conformal. This observation leads to the construction of appropriate grid control maps such that the solution of the Poisson system generates an orthogonal grid in physical space with boundary grid points fixed on two adjacent edges but moved along the other two opposite edges. This result is similar to that reported by Kang and Leal [13], although they used the Ryskin-Leal grid generation equations [15] instead of the Poisson grid generation equations. Section 2.4 shows generated grids in physical space for well-defined geometries so that the reader is able to recompute the grids (by the methods presented in this chapter or by his/her own favourite methods for comparison). The corresponding constructed grid control maps are shown as grids in parameter space.

In Section 3 is shortly described how the same methods to construct appropriate grid control maps for two-dimensional grids can also be used for grid generation on surfaces in 3D physical space. It is shown that surface grid generation on minimal surfaces (soap films) is in fact the same as 2D grid generation. Conceptually, the same methods can also be used for parametrically defined surfaces although the numerical implementation is completely different.

The extension to volume grid generation is described in Section 4. The construction of appropriate grid control maps for 3D domains is less well developed than for 2D domains. However, a method to construct a grid control map has been proposed which works surprisingly well for many applications.

The now-standard procedure in multi-block structured grid generation codes is to first generate surface grids on block faces, both boundary and interior block interfaces, from grid point distributions placed on the face edges by distribution functions. Then volume grids are generated within the blocks. For this reason, the elliptic grid generation methods described in this chapter assume fixed position of the prescribed boundary grid points.

2 Two-dimensional grid generation

2.1 Harmonic maps, grid control maps and Poisson systems

Consider a simply connected bounded domain $\mathcal{D}$ in two-dimensional space with Cartesian coordinates $\vec{x} = (x, y)^T$. Suppose that $\mathcal{D}$ is bounded by four edges E_1, E_2, E_3, E_4. Let (E_1, E_2) and (E_3, E_4) be the two pairs of opposite edges as shown in Fig.1.

A harmonic map is defined as a differentiable one-to-one map from $\mathcal{D}$ onto a unit square such that

1. the boundary of $\mathcal{D}$ is mapped onto the boundary of the unit square,
2. the vertices of $\mathcal{D}$ are mapped, in the proper sequence, onto the corners of the unit square,
3. the two components of the map are harmonic functions in the interior of $\mathcal{D}$.

Let $\vec{s} : \mathcal{D} \mapsto \mathcal{P}$ be a harmonic map where the parameter space $\mathcal{P}$ is the unit square in a two-dimensional space with Cartesian coordinates $\vec{s} = (s, t)^T$. Assume that

- $s \equiv 0$ at edge E_1 and $s \equiv 1$ at edge E_2,
- $t \equiv 0$ at edge E_3 and $t \equiv 1$ at edge E_4.

The problem of generating an appropriate grid in the physical domain $\mathcal{D}$ can be effectively reduced to a simpler problem of generating an appropriate grid in the parameter space $\mathcal{P}$, which can after that be mapped into $\mathcal{D}$, by using the inverse of the harmonic map $\vec{x} : \mathcal{P} \mapsto \mathcal{D}$.

Define the computational space $\mathcal{C}$ as the unit square in a two-dimensional space with Cartesian coordinates $\vec{\xi} = (\xi, \eta)^T$. A grid control map $\vec{s} : \mathcal{C} \mapsto \mathcal{P}$ is defined as a differentiable one-to-one map from $\mathcal{C}$ onto $\mathcal{P}$ and maps a uniform grid in $\mathcal{C}$ to a non-uniform (in general) grid in $\mathcal{P}$. Assume that

- $s(0, \eta) \equiv 0$ and $s(1, \eta) \equiv 1$,
- $t(\xi, 0) \equiv 0$ and $t(\xi, 1) \equiv 1$.

Then the computational coordinates also fulfill

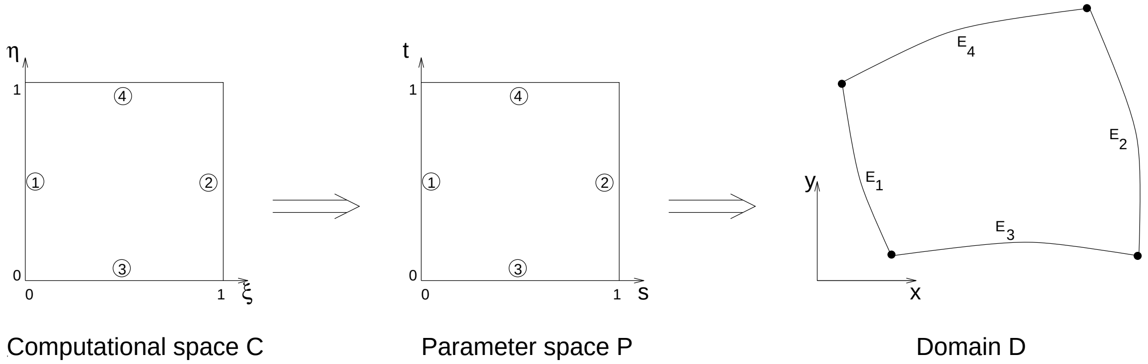


Figure 1: Composite map from computational (ξ, η) space to a domain $\mathcal{D}$ in Cartesian (x, y) space.

- $\xi \equiv 0$ at edge E_1 and $\xi \equiv 1$ at edge E_2,
- $\eta \equiv 0$ at edge E_3 and $\eta \equiv 1$ at edge E_4.

The composition of a grid control map $\vec{s} : \mathcal{C} \mapsto \mathcal{P}$ and the inverse of the harmonic map $\vec{x} : \mathcal{P} \mapsto \mathcal{D}$ define a map $\vec{x} : \mathcal{C} \mapsto \mathcal{D}$ which transforms a uniform grid in $\mathcal{C}$ to a non-uniform (in general) grid in $\mathcal{D}$. The composite map obeys a quasi-linear system of elliptic partial differential equations, known as the Poisson grid generation equations, with control functions completely defined by the grid control map. The secret of each "good" elliptic grid generation method is the method of computing appropriate control functions, which is thus equivalent to constructing appropriate grid control maps.

We will now derive the quasi-linear system of elliptic partial differential equations which the composite mapping $\vec{x} = \vec{x}(\vec{s}(\vec{\xi}))$ has to fulfill. Suppose that the harmonic map and the grid control map are defined so that the composite map exists. Introduce the two covariant base vectors

$$\vec{a}_1 = \frac{\partial \vec{x}}{\partial \xi} = \vec{x}_\xi \ , \ \vec{a}_2 = \frac{\partial \vec{x}}{\partial \eta} = \vec{x}_\eta, \tag{1}$$

and define the covariant metric tensor components as the inner product of the covariant base vectors

$$a_{ij} = (\vec{a}_i, \vec{a}_j) \ , \ i = \{1, 2\}, j = \{1, 2\}. \tag{2}$$

The two contravariant base vectors $\vec{a}^1 = \nabla\xi = (\xi_x, \xi_y)^T$ and $\vec{a}^2 = \nabla\eta = (\eta_x, \eta_y)^T$ obey

$$(\vec{a}^i, \vec{a}_j) = \delta^i_j \ , \ i = \{1, 2\}, j = \{1, 2\}, \tag{3}$$

with δ^i_j the Kronecker symbol. Define the contravariant metric tensor components

$$a^{ij} = (\vec{a}^i, \vec{a}^j) \ , \ i = \{1, 2\}, j = \{1, 2\}, \tag{4}$$

so that

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{pmatrix} \begin{pmatrix} a^{11} & a^{12} \\ a^{12} & a^{22} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \tag{5}$$

and

$$\begin{aligned} \vec{a}^1 &= a^{11}\vec{a}_1 + a^{12}\vec{a}_2 \ , \ \vec{a}^2 = a^{12}\vec{a}_1 + a^{22}\vec{a}_2, \\ \vec{a}_1 &= a_{11}\vec{a}^1 + a_{12}\vec{a}^2 \ , \ \vec{a}_2 = a_{12}\vec{a}^1 + a_{22}\vec{a}^2. \end{aligned} \tag{6}$$

Introduce the determinant J^2 of the covariant metric tensor: $J^2 = a_{11}a_{22} - a_{12}^2$.

Now consider an arbitrary function $\phi = \phi(\xi, \eta)$. Then ϕ is also defined in domain $\mathcal{D}$, and the Laplacian of ϕ is expressed as

$$\triangle\phi = \phi_{xx} + \phi_{yy} = \frac{1}{J}\left\{ \left(Ja^{11}\phi_\xi + Ja^{12}\phi_\eta\right)_\xi + \left(Ja^{12}\phi_\xi + Ja^{22}\phi_\eta\right)_\eta \right\}, \tag{7}$$

which may be found in the Appendix of this Handbook and in every textbook on Tensor Analysis and Differential Geometry (for example see [23]). Take as special cases respectively $\phi \equiv \xi$ and $\phi \equiv \eta$. Then Eq.(7) yields

$$\triangle\xi = \frac{1}{J}\left\{ \left(Ja^{11}\right)_\xi + \left(Ja^{12}\right)_\eta \right\} \ , \ \triangle\eta = \frac{1}{J}\left\{ \left(Ja^{12}\right)_\xi + \left(Ja^{22}\right)_\eta \right\}. \tag{8}$$

Thus the Laplacian of ϕ can also be expressed as

$$\triangle\phi = a^{11}\phi_{\xi\xi} + 2a^{12}\phi_{\xi\eta} + a^{22}\phi_{\eta\eta} + \triangle\xi\phi_\xi + \triangle\eta\phi_\eta. \tag{9}$$

Substitution of respectively $\phi \equiv s$ and $\phi \equiv t$ in this equation yields

$$\triangle s = a^{11}s_{\xi\xi} + 2a^{12}s_{\xi\eta} + a^{22}s_{\eta\eta} + \triangle\xi s_\xi + \triangle\eta s_\eta, \tag{10}$$

$$\triangle t = a^{11}t_{\xi\xi} + 2a^{12}t_{\xi\eta} + a^{22}t_{\eta\eta} + \triangle\xi t_\xi + \triangle\eta t_\eta. \tag{11}$$

Using these equations and the property that s and t are harmonic in domain $\mathcal{D}$, thus $\triangle s = 0$ and $\triangle t = 0$, we find the following expressions for the Laplacian of ξ and η:

$$\begin{pmatrix} \triangle\xi \\ \triangle\eta \end{pmatrix} = a^{11}\vec{P}_{11} + 2a^{12}\vec{P}_{12} + a^{22}\vec{P}_{22}, \tag{12}$$

where

$$\vec{P}_{11} = -T^{-1}\begin{pmatrix} s_{\xi\xi} \\ t_{\xi\xi} \end{pmatrix} \ , \ \vec{P}_{12} = -T^{-1}\begin{pmatrix} s_{\xi\eta} \\ t_{\xi\eta} \end{pmatrix} \ , \ \vec{P}_{22} = -T^{-1}\begin{pmatrix} s_{\eta\eta} \\ t_{\eta\eta} \end{pmatrix}, \tag{13}$$

and the matrix T is defined as

$$T = \begin{pmatrix} s_\xi & s_\eta \\ t_\xi & t_\eta \end{pmatrix} . \tag{14}$$

The six coefficients of the vectors $\vec{P}_{11} = (P^1_{11}, P^2_{11})^T$, $\vec{P}_{12} = (P^1_{12}, P^2_{12})^T$ and $\vec{P}_{22} = (P^1_{22}, P^2_{22})^T$ are the so called control functions. The six control functions are completely defined and easily computed for a given grid control map $\vec{s} = \vec{s}(\vec{\xi})$. Different and less useful expressions of these control functions can also be found in [1, 2].

Finally, substitution of $\phi \equiv \vec{x}$ in Eq.(9) yields

$$\triangle\vec{x} = a^{11}\vec{x}_{\xi\xi} + 2a^{12}\vec{x}_{\xi\eta} + a^{22}\vec{x}_{\eta\eta} + \triangle\xi\vec{x}_\xi + \triangle\eta\vec{x}_\eta \ . \tag{15}$$

Substituting Eq.(12) into this equation and using the fact that $\triangle\vec{x} \equiv 0$, we arrive at the familiar Poisson grid generation system:

$$\begin{aligned} a^{11}\vec{x}_{\xi\xi} + 2a^{12}\vec{x}_{\xi\eta} + a^{22}\vec{x}_{\eta\eta} \quad &+ \quad \left(a^{11}P^1_{11} + 2a^{12}P^1_{12} + a^{22}P^1_{22}\right)\vec{x}_\xi \\ &+ \quad \left(a^{11}P^2_{11} + 2a^{12}P^2_{12} + a^{22}P^2_{22}\right)\vec{x}_\eta = 0. \end{aligned} \tag{16}$$

Using Eqs.(2),(5) we find the following well-known expressions for the contravariant metric tensor components:

$$J^2a^{11} = a_{22} = (\vec{x}_\eta, \vec{x}_\eta) \ , \ J^2a^{12} = -a_{12} = -(\vec{x}_\xi, \vec{x}_\eta) \ , \ J^2a^{22} = a_{11} = (\vec{x}_\xi, \vec{x}_\xi). \tag{17}$$

Thus the Poisson grid generation system defined by Eq.(16) can be simplified by multiplication with J^2. Then we obtain:

$$\begin{aligned} a_{22}\vec{x}_{\xi\xi} - 2a_{12}\vec{x}_{\xi\eta} + a_{11}\vec{x}_{\eta\eta} \quad &+ \quad \left(a_{22}P^1_{11} - 2a_{12}P^1_{12} + a_{11}P^1_{22}\right)\vec{x}_\xi \\ &+ \quad \left(a_{22}P^2_{11} - 2a_{12}P^2_{12} + a_{11}P^2_{22}\right)\vec{x}_\eta = 0. \end{aligned} \tag{18}$$

This equation, together with the expressions for the control functions P^k_{ij} given by Eq.(13), is the two-dimensional grid generation system. For a given grid control map, so that the six control functions in Eq.(18) are given functions of ξ and η, boundary conforming grids in the interior of domain $\mathcal{D}$ are computed by solving this quasi-linear system of elliptic partial differential equations with prescribed boundary grid points as Dirichlet boundary conditions. The discretization and solution method of this Poisson system is discussed in the next section. The construction of appropriate grid control maps such that the corresponding grid in physical space has desired properties is discussed in the remaining sections.

2.2 Discretization and solution method

Consider a uniform rectangular grid of $(N+1)\times(M+1)$ points in computational space $\mathcal{C}$ defined as

$$\xi_{i,j}=\xi_i=i/N\ ,\ \eta_{i,j}=\eta_j=j/M\ ,\ i=0\dots N,\ j=0\dots M. \tag{19}$$

Assume that $\vec{x}_{i,j}$ is prescribed on the boundary of this grid and consider the computation of $\vec{x}_{i,j}$ in the interior of the computational grid based on the solution of the Poisson system defined by Eq.(18).

Assume that a grid control map $\vec{s}:\mathcal{C}\mapsto\mathcal{P}$ has been constructed. Thus the values $s_{i,j}$ and $t_{i,j}$ are known at each grid point. At each interior grid point $(i,j)\in(1\dots N-1,1\dots M-1)$, the six control functions P^1_{11},P^2_{11},P^1_{12}, P^2_{12},P^1_{22},P^2_{22} defined by Eq.(13) are now easily computed using central differences for the discretization of $s_{\xi\xi}$,$s_{\xi\eta}$,$s_{\eta\eta}$,s_ξ,s_η and $t_{\xi\xi}$,$t_{\xi\eta}$,$t_{\eta\eta}$,t_ξ,t_η.

The iterative solution process of the nonlinear elliptic Poisson grid generation system defined by Eq.(18) can be simply obtained by Picard iteration. Rewrite the Poisson system as

$$P\vec{x}_{\xi\xi}-2Q\vec{x}_{\xi\eta}+R\vec{x}_{\eta\eta}+S\vec{x}_\xi+T\vec{x}_\eta=0 \tag{20}$$

with

$$\begin{aligned}P&=(\vec{x}_\eta,\vec{x}_\eta)\ ,\ Q=(\vec{x}_\xi,\vec{x}_\eta)\ ,\ R=(\vec{x}_\xi,\vec{x}_\xi)\ ,\\ S&=PP^1_{11}-2QP^1_{12}+RP^1_{22},\\ T&=PP^2_{11}-2QP^2_{12}+RP^2_{22}.\end{aligned} \tag{21}$$

The iterative solution by Picard iteration can be written as

$$P^{k-1}\vec{x}^k_{\xi\xi}-2Q^{k-1}\vec{x}^k_{\xi\eta}+R^{k-1}\vec{x}^k_{\eta\eta}+S^{k-1}\vec{x}^k_\xi+T^{k-1}\vec{x}^k_\eta=0 \tag{22}$$

where k is the Picard index and

$$\begin{aligned}P^{k-1}&=(\vec{x}^{k-1}_\eta,\vec{x}^{k-1}_\eta)\ ,\ Q^{k-1}=(\vec{x}^{k-1}_\xi,\vec{x}^{k-1}_\eta)\ ,\ R^{k-1}=(\vec{x}^{k-1}_\xi,\vec{x}^{k-1}_\xi)\ ,\\ S^{k-1}&=P^{k-1}P^1_{11}-2Q^{k-1}P^1_{12}+R^{k-1}P^1_{22},\\ T^{k-1}&=P^{k-1}P^2_{11}-2Q^{k-1}P^2_{12}+R^{k-1}P^2_{22}.\end{aligned} \tag{23}$$

Thus a current approximate solution

$$\vec{x}^{k-1}=\left\{\vec{x}^{k-1}_{ij},i=0\dots N,j=0\dots M\right\} \tag{24}$$

is improved by the following steps:

- Compute at interior grid points the coefficients P^{k-1},Q^{k-1},R^{k-1},S^{k-1},T^{k-1} by applying central differences for the discretization of $\vec{x}_{\xi}^{k-1}$ and $\vec{x}_{\eta}^{k-1}$. Note that the six control functions remain unchanged during the iterative procedure.

- Discretize at interior grid points $\vec{x}_{\xi\xi}^{k}$, $\vec{x}_{\xi\eta}^{k}$, $\vec{x}_{\eta\eta}^{k}$, $\vec{x}_{\xi}^{k}$,$\vec{x}_{\eta}^{k}$ using central differences.

- After the discretization of $\vec{x}_{\xi\xi}^{k}$, $\vec{x}_{\xi\eta}^{k}$, $\vec{x}_{\eta\eta}^{k}$, $\vec{x}_{\xi}^{k}$, $\vec{x}_{\eta}^{k}$ we arrive at a linear system of equations for the unknowns $\vec{x}_{i,j}^{k}, i = 1 \ldots N-1, j = 1 \ldots M-1$. At each interior grid point we have a nine-point stencil. Boundary grid points are prescribed and remain unchanged.

  This linear system can be solved by a black-box multigrid solver. Such a multigrid solver is called twice to compute the two components $x_{i,j}^{k}$ and $y_{i,j}^{k}$ of $\vec{x}_{i,j}^{k}$. The solution of the linear system provides a better approximate solution $\vec{x}^{k}$.

The following algorithm describes the computation of an interior grid in domain $\mathcal{D}$ with prescribed boundary grid points and a given grid control map.

Algorithm 1. Grid generation.

1. Compute the six control functions from the grid control map.

2. Compute an initial grid in the interior of domain $\mathcal{D}$ by a simple algebraic grid generation method. The quality of the initial grid is unimportant and severe grid folding is allowed. The initial grid is used as starting solution for the Picard iteration process. The final grid will be independent of the initial grid.

3. Solve the quasi-linear Poisson grid generation equations iteratively by Picard iteration. The fixed position of the boundary grid points define Dirichlet boundary conditions. In general, a sufficiently converged grid is obtained in about 10 Picard iterations. The residual is then typically decreased by a factor 1000.

2.3 Construction of grid control maps

2.3.1 Laplace grids

The most simple grid control map is the identity map $\vec{s} = \vec{\xi}$. The six control functions are identical zero and the Poisson grid generation system defined by Eq.(18) simplifies to $a_{22}\vec{x}_{\xi\xi} - 2a_{12}\vec{x}_{\xi\eta} + a_{11}\vec{x}_{\eta\eta} = 0$ which is equivalent with $\triangle\xi = 0$ and $\triangle\eta = 0$ according to Eq.(12). Grids based on this equation are the so-called Laplace (or Harmonic) grids which were first introduced by Winslow [18]. The inherent smoothness of the Laplace operator makes the grid evenly spaced in the interior. Therefore, the quality of a Laplace grid will only be acceptable as long as the boundary grid points are evenly spaced along the edges.

This is illustrated in Fig.5 and Fig.6 where a region about a NACA0012 airfoil is subdivided into four domains. The domains have common edges and more or less evenly spaced boundary grid points are prescribed. Fig.6 shows Laplace grids in each domain. The result is not bad for this O-type Euler mesh. Only smooth grids are required for the solution of the Euler equations for non-viscous flow, where strong gradients near boundaries do not occur. Laplace grids provide no control about the angle distribution between internal grid lines and the boundary. This causes slope discontinuity of the grid lines across internal domain boundaries, as shown in Fig.6.

The situation is completely different for Navier-Stokes type of meshes where the grid must contain a boundary layer grid. Highly stretched grids are required for solutions of the Navier-Stokes equations for viscous flow, where large gradients occur near boundaries. Fig.9 shows a region about a RAE2822 airfoil also subdivided into four domains. The boundary grid point distribution is highly dense near the leading and trailing edge of the airfoil. Fig.10 shows the Laplace grids in the four domains. These grids are unacceptable because the inherent smoothness of the Laplace operator causes evenly spaced grids so that the interior grid contains no boundary layer at all. Therefore, Laplace grids are in general unusable in most practice.

2.3.2 Arc length based grids

Consider domain $\mathcal{D}$ as shown in Fig.1. Assume that the boundary grid points are prescribed at the four edges of $\mathcal{D}$. A boundary conforming grid in the interior of domain $\mathcal{D}$ with an interior grid point distribution which is a good reflection of the prescribed boundary grid point distribution can be obtained by constructing a grid control map based on normalized arc length. In order to construct such a grid control we define

- $s \equiv 0$ at edge E_1 and $s \equiv 1$ at edge E_2,
- s is the normalized arc length along edges E_3 and E_4,
- $t \equiv 0$ at edge E_3 and $t \equiv 1$ at edge E_4,
- t is the normalized arc length along edges E_1 and E_2.

For example this means that along edge E_3 we define $s(u) = \int_0^u \parallel \vec{x}_u \parallel du / \int_0^1 \parallel \vec{x}_u \parallel du$ where $\vec{x} : u \in [0,1] \mapsto (x,y) \in \mathcal{R}^2$ is a parametrization of edge E_3 in the right direction. Thus $\vec{s} : \partial\mathcal{D} \mapsto \partial\mathcal{P}$

is defined by these requirements. The two Laplace equations $\triangle s = 0$ and $\triangle t = 0$, together with the above specified Dirichlet boundary conditions, define the harmonic map $\vec{s} : \mathcal{D} \mapsto \mathcal{P}$. Note that this map only depends on the shape of domain $\mathcal{D}$ and is independent of the prescribed boundary grid point distribution.

The boundary grid points are prescribed at the four edges of $\mathcal{D}$. Thus $\vec{x} : \partial\mathcal{C} \mapsto \partial\mathcal{D}$ is prescribed. Because $\vec{x} : \partial\mathcal{C} \mapsto \partial\mathcal{D}$ is prescribed and $\vec{s} : \partial\mathcal{D} \mapsto \partial\mathcal{P}$ is defined as described above, it follows that $\vec{s} : \partial\mathcal{C} \mapsto \partial\mathcal{P}$ is also defined.

From the preceding requirements it follows that

$$s(0,\eta) = 0 \ , \ s(1,\eta) = 1 \ , \ s(\xi,0) = s^a_{E_3}(\xi) \ , \ s(\xi,1) = s^a_{E_4}(\xi), \tag{25}$$

where the functions $s^a_{E_3}, s^a_{E_4}$ are monotonically increasing, and

$$t(\xi,0) = 0 \ , \ t(\xi,1) = 1 \ , \ t(0,\eta) = t^a_{E_1}(\eta) \ , \ t(1,\eta) = t^a_{E_2}(\eta), \tag{26}$$

where the functions $t^a_{E_1}, t^a_{E_2}$ are also monotonically increasing. The superscript a is used to indicate that these functions measure the normalized arc length at the boundary grid points.

The grid control map $\vec{s} : \mathcal{C} \mapsto \mathcal{P}$ is now defined by the following two algebraic equations:

$$s = s^a_{E_3}(\xi)(1-t) + s^a_{E_4}(\xi)t, \tag{27}$$

$$t = t^a_{E_1}(\eta)(1-s) + t^a_{E_2}(\eta)s. \tag{28}$$

Eq.(27) implies that a coordinate line $\xi =$ constant is mapped to the parameter space $\mathcal{P}$ as a straight line: s is a linear function of t, and Eq.(28) implies that a grid line $\eta =$ constant is also mapped to $\mathcal{P}$ as a straight line: t is a linear function of s. For given values of ξ and η, the corresponding s and t values are found as the intersection point of the two straight lines. It can be easily verified that the grid control map is a differentiable and one-to-one because of the positiveness of the Jacobian: $s_\xi t_\eta - s_\eta t_\xi > 0$.

The discrete computation of the grid control map is straightforward. For a grid of $(N+1)\times(M+1)$ points, the distance between succeeding grid points at the boundary are computed as

$$\bar{d}_{0,j} = \parallel \vec{x}_{0,j} - \vec{x}_{0,j-1} \parallel \quad , \quad \bar{d}_{N,j} = \parallel \vec{x}_{N,j} - \vec{x}_{N,j-1} \parallel \ , \ j = 1 \ldots M, \tag{29}$$

$$\bar{d}_{i,0} = \parallel \vec{x}_{i,0} - \vec{x}_{i-1,0} \parallel \quad , \quad \bar{d}_{i,M} = \parallel \vec{x}_{i,M} - \vec{x}_{i-1,M} \parallel \ , \ i = 1 \ldots N. \tag{30}$$

Define the length of edges E_1, E_2, E_3, E_4 by

$$L_{E_1} = \sum_{j=1}^{M} \bar{d}_{0,j} \ , \ L_{E_2} = \sum_{j=1}^{M} \bar{d}_{N,j} \ , \ L_{E_3} = \sum_{i=1}^{N} \bar{d}_{i,0} \ , \ L_{E_4} = \sum_{i=1}^{N} \bar{d}_{i,M}, \tag{31}$$

and the normalized distances as

$$d_{0,j} = \bar{d}_{0,j}/L_{E_1} \quad , \quad d_{N,j} = \bar{d}_{N,j}/L_{E_2} \ , \ j = 1 \ldots M, \tag{32}$$

$$d_{i,0} = \bar{d}_{i,0}/L_{E_3} \quad , \quad d_{i,M} = \bar{d}_{i,M}/L_{E_4} \ , \ i = 1 \ldots N. \tag{33}$$

The discrete components $s_{i,j}$ and $t_{i,j}$ of the grid control map are computed at the boundary by

$$s_{0,j} \ = \ 0 \ , \ s_{N,j} = 1 \ , \ j = 0 \ldots M, \tag{34}$$

$$t_{i,0} \ = \ 0 \ , \ t_{i,M} = 1 \ , \ i = 0 \ldots N, \tag{35}$$

and

$$s_{i,0} \ = \ s_{i-1,0} + d_{i,0} \ , \ s_{i,M} = s_{i-1,M} + d_{i,M} \ , \ i = 1 \ldots N, \tag{36}$$

$$t_{0,j} \ = \ t_{0,j-1} + d_{0,j} \ , \ t_{N,j} = t_{N,j-1} + d_{N,j} \ , \ j = 1 \ldots M. \tag{37}$$

The interior values are defined according to Eqs.(27),(28) and are thus found by solving simultaneously the two linear algebraic equations:

$$s_{i,j} \ = \ s_{i,0}(1 - t_{i,j}) + s_{i,M}t_{i,j}, \tag{38}$$

$$t_{i,j} \ = \ t_{0,j}(1 - s_{i,j}) + t_{N,j}s_{i,j}, \tag{39}$$

for each pair $(i, j) \in (1 \ldots N - 1, 1 \ldots M - 1)$.

The next algorithm summarizes the computation of arc length based grid in the interior of $\mathcal{D}$.

Algorithm 2. Arc length based grids.

1. Compute the four edge functions $t^a_{E_1}$,$t^a_{E_2}$,$s^a_{E_3}$ and $s^a_{E_4}$ from the boundary grid point distribution.

2. Compute the grid control map according to Eqs.(27),(28).

3. Compute the corresponding interior grid in $\mathcal{D}$ as described in **Algorithm 1**.

Illustrations of boundary conforming grids obtained with this grid control map are shown in Fig.7 and Fig.11. As opposed to Laplace grids, the interior grid point distribution is always a good reflection of the prescribed boundary grid point distribution. Grid folding hardly ever occurs because both the grid control map and the harmonic map are one-to-one. When grid folding occurs then it must be caused by discretization errors [10]. Hence, grid folding will always disappear when the grid is sufficiently refined.

A shortcoming of this grid control map is that there is no control about the angle distribution between interior grid lines and the boundary edges of the domain. It is often desired that the interior grid lines are orthogonal at the boundary edges. For example, viscous flow simulations often require orthogonality of the grid in a boundary layer. This can be achieved with a grid control map as constructed below.

2.3.3 Grid orthogonality at the boundary

Consider domain $\mathcal{D}$ with prescribed boundary grid points. Suppose that it is desired to generate a boundary conforming grid in the interior of $\mathcal{D}$ which is orthogonal at all four edges of domain $\mathcal{D}$. This can be achieved by imposing Dirichlet-Neumann boundary conditions for the harmonic map:

- $s \equiv 0$ at edge E_1 and $s \equiv 1$ at edge E_2,
- $\frac{\partial s}{\partial n} = 0$ along edges E_3 and E_4, where n is the outward normal direction,
- $t \equiv 0$ at edge E_3 and $t \equiv 1$ at edge E_4,
- $\frac{\partial t}{\partial n} = 0$ along edges E_1 and E_2, where n is the outward normal direction.

The two Laplace equations $\triangle s = 0$ and $\triangle t = 0$, together with the above specified boundary conditions, define the harmonic map $\vec{s} : \mathcal{D} \mapsto \mathcal{P}$. Again this map only depends on the shape of domain $\mathcal{D}$ and is independent of the prescribed boundary grid point distribution.

The Neumann boundary conditions $\frac{\partial s}{\partial n} = 0$ along edges E_3 and E_4 imply that a parameter line $s =$ constant in $\mathcal{P}$ will be mapped into domain $\mathcal{D}$ by the inverse of the harmonic map as a curve which is orthogonal at those edges. Similarly, a parameter line $t =$ constant in $\mathcal{P}$ will be mapped as a curve in $\mathcal{D}$ which is orthogonal at edge E_1 and edge E_2. These properties can be used to construct a grid control map such that the interior grid in $\mathcal{D}$ will be orthogonal at the boundary.

The boundary grid points are prescribed at the four edges of $\mathcal{D}$. Thus $\vec{x} : \partial\mathcal{C} \mapsto \partial\mathcal{D}$ is prescribed. Because $\vec{x} : \partial\mathcal{C} \mapsto \partial\mathcal{D}$ is prescribed and $\vec{s} : \partial\mathcal{D} \mapsto \partial\mathcal{P}$ is also defined , it follows that $\vec{s} : \partial\mathcal{C} \mapsto \partial\mathcal{P}$ is also defined.

From the preceding requirements it follows that

$$s(0,\eta) = 0 \ , \ s(1,\eta) = 1 \ , \ s(\xi,0) = s^o_{E_3}(\xi) \ , \ s(\xi,1) = s^o_{E_4}(\xi), \tag{40}$$

where the functions $s^o_{E_3}, s^o_{E_4}$ are monotonically increasing, and

$$t(\xi,0) = 0 \ , \ t(\xi,1) = 1 \ , \ t(0,\eta) = t^o_{E_1}(\eta) \ , \ t(1,\eta) = t^o_{E_2}(\eta), \tag{41}$$

where the functions $t^o_{E_1}, t^o_{E_2}$ are also monotonically increasing. The superscript o is used to indicate that these functions are constructed in a way to obtain grid orthogonality at the boundary.

The grid control map $\vec{s} : \mathcal{C} \mapsto \mathcal{P}$ is now defined by:

$$s = s^o_{E_3}(\xi) H_0(t) + s^o_{E_4}(\xi) H_1(t), \tag{42}$$

$$t = t^o_{E_1}(\eta) H_0(s) + t^o_{E_2}(\eta) H_1(s). \tag{43}$$

where H_0 and H_1 are cubic Hermite interpolation functions defined as

$$H_0(s) = (1+2s)(1-s)^2, H_1(s) = (3-2s)s^2, 0 \leq s \leq 1. \tag{44}$$

Note that $H_0(0) = 1$, $H_0'(0) = 0$, $H_0(1) = 0$, $H_0'(1) = 0$ and $H_1(0) = 0$, $H_1'(0) = 0$, $H_1(1) = 1$, $H_1'(1) = 0$. It follows from Eq.(42) that a coordinate line $\xi =$ constant in $\mathcal{C}$ is mapped to parameter space $\mathcal{P}$ as a cubic curve (with t as dependent variable) which is orthogonal at both edge E_3 and edge E_4 in $\mathcal{P}$. Such a curve in parameter space $\mathcal{P}$ will thus be mapped by the inverse of the harmonic map $\vec{x} : \mathcal{P} \mapsto \mathcal{D}$ as a curve which is orthogonal at both edge E_3 and edge E_4 in $\mathcal{D}$. Similar observations can be made for coordinate lines $\eta =$ constant. Thus the grid will be orthogonal at all four edges in domain $\mathcal{D}$.

Grid orthogonality at boundaries may introduce grid folding. Fortunately, grid folding will not easily arise. From Eq.(42) it follows that two different coordinate lines $\xi = \xi_1$,$\xi = \xi_2$, $\xi_1 \neq \xi_2$, are mapped to parameter space $\mathcal{P}$ as two disjunct cubic curves which are orthogonal at both edge E_3 and edge E_4 in $\mathcal{P}$. This is due to the fact that $s^o_{E_3}(\xi)$ and $s^o_{E_4}(\xi)$ are monotonically increasing functions. The same holds for different coordinate lines $\eta = \eta_1$,$\eta = \eta_2$, $\eta_1 \neq \eta_2$. For given values of ξ and η, the corresponding s and t values are found as intersection point of two cubic curves. However, such two cubic curves may have more than one intersection point. In that case grid folding will occur. However, in practice we hardly ever encounter grid folding due to orthogonalization of the grid at the boundary.

We have described a method to obtain an orthogonal grid at all four edges of domain $\mathcal{D}$. In practice, orthogonality of the grid is often only desired at less than four edges. Suppose for example that it is only desired to have an orthogonal grid at edge E_3. Then take $t_{E_1}(\eta) = t^a_{E_1}(\eta)$, $t_{E_2}(\eta) = t^a_{E_2}(\eta)$, $s_{E_4}(\xi) = s^a_{E_4}(\xi)$ and $s_{E_3}(\xi) = s^o_{E_3}(\xi)$. Furthermore, the grid control map $\vec{s} : \mathcal{C} \mapsto \mathcal{P}$ is such that a coordinate line $\eta =$ constant is mapped to $\mathcal{P}$ as a straight line and a coordinate line $\xi =$ constant is mapped to $\mathcal{P}$ as a parabolic curve (with t as dependent variable) which is only orthogonal at edge E_3 in $\mathcal{P}$. For given values of ξ and η, the corresponding s and t values are then found as intersection point of a straight line and a parabolic curve.

The discrete computation of the grid control map is more complicated when grid orthogonality is required. We have seen that for a grid control map based on normalized arc length, the functions $t^a_{E_1}$,$t^a_{E_2}$,$s^a_{E_3}$ and $s^a_{E_4}$ can be directly computed from the prescribed boundary grid points only. However, when grid orthogonality is required, the functions $t^o_{E_1}$,$t^o_{E_2}$,$s^o_{E_3}$ and $s^o_{E_4}$ can only be found by solving the Laplace equations $\triangle s = 0$ and $\triangle t = 0$ supplied with the above mentioned Dirichlet-Neumann boundary conditions. The solution of the Laplace equations $\triangle s = 0$ and $\triangle t = 0$ supplied with the boundary conditions requires an initial folding-free grid in the interior of domain $\mathcal{D}$. Therefore, an orthogonal grid at the boundary is in general obtained in three steps:

Algorithm 3. Grid orthogonality at boundary.

1. Compute an initial boundary conforming grid in the interior of $\mathcal{D}$ without grid folding. Such a grid can be computed using the grid control map based on normalized arc length as described in **Algorithm 2**.

2. Solve on this mesh $\triangle s = 0$ and $\triangle t = 0$ supplied with the above specified Dirichlet-Neumann boundary conditions. A solution method is described in [19]. The solution at the boundary defines the edge functions $t^o_{E_1}$,$t^o_{E_2}$,$s^o_{E_3}$ and $s^o_{E_4}$.

3. Compute the grid control map according to Eqs.(42),(43).

4. Compute the corresponding interior grid in $\mathcal{D}$ as described in **Algorithm 1**.

Illustrations of boundary conforming grids obtained with this grid control map are shown in Fig.8 and Fig.12. The common interior boundary edges of the four domains can hardly be recognized anymore because of the excellent grid orthogonality at these edges. The grid spacing of the interior grid is also good in both cases.

In the next section we will prove that the harmonic map $\vec{s} : \mathcal{D} \mapsto \mathcal{P}$ supplied with Dirichlet-Neumann boundary conditions is quasi-conformal. This observation leads to the construction of appropriate grid control maps such that the corresponding grid is orthogonal, not only at the boundary but also in the interior of $\mathcal{D}$.

2.3.4 Orthogonal grids

There is a famous theorem in conformal mapping theory which states that each simply connected domain $\mathcal{D}$ can be mapped conformally to a rectangle $\mathcal{R}$ in such a way that the vertices of domain $\mathcal{D}$ are mapped, in the proper sequence onto the corners of the rectangle [11, 27]. The ratio of the length of two adjacent sides of the rectangle is called the conformal module M which is a characteristic and fundamental property of each domain.

Let $\vec{u} : \mathcal{D} \mapsto \mathcal{R}$ be the conformal map where $\mathcal{R}$ is the rectangle $[0,1] \times [0,M]$ in a two-dimensional space with Cartesian coordinates $\vec{u} = (u,v)^T$. The components of the conformal map obey the Cauchy-Riemann relations:

$$\begin{pmatrix} u_x \\ u_y \end{pmatrix} = \begin{pmatrix} v_y \\ -v_x \end{pmatrix} \tag{45}$$

Hence $\triangle u = 0$ and $\triangle v = 0$ in the interior of domain $\mathcal{D}$. Furthermore, we may assume that the map $\vec{u} : \mathcal{D} \mapsto \mathcal{R}$ obeys

- $u \equiv 0$ at edge E_1 and $u \equiv 1$ at edge E_2,
- $v \equiv 0$ at edge E_3 and $v \equiv M$ at edge E_4.

From these boundary conditions and using the Cauchy-Riemann relations we can also conclude that

- $\frac{\partial u}{\partial n} = 0$ along edges E_3 and E_4, where n is the outward normal direction,
- $\frac{\partial v}{\partial n} = 0$ along edges E_1 and E_2, where n is the outward normal direction.

Thus the conformal map $\vec{u} : \mathcal{D} \mapsto \mathcal{R}$ is harmonic and obeys the same set of Dirichlet-Neumann boundary conditions as the harmonic map $\vec{s} : \mathcal{D} \mapsto \mathcal{P}$. Therefore the two maps are related to each other according to

$$s = u \;,\; t = \frac{v}{M} \tag{46}$$

This means that the harmonic map is quasi-conformal and obeys

$$\begin{pmatrix} s_x \\ s_y \end{pmatrix} = M \begin{pmatrix} t_y \\ -t_x \end{pmatrix} \tag{47}$$

Thus the two contravariant vectors are orthogonal but have different lengths. It is not difficult to show, using the relations between covariant and contravariant vectors given by Eq.(6), that the covariant vectors fulfill

$$\begin{pmatrix} x_s \\ y_s \end{pmatrix} = \frac{1}{M} \begin{pmatrix} y_t \\ -x_t \end{pmatrix} \tag{48}$$

so that the inverse mapping obeys

$$M^2 \vec{x}_{ss} + \vec{x}_{tt} = 0 \tag{49}$$

which is the well-known partial differential equation for quasi-conformal maps [4],page 96. It can also be easily verified that the conformal module can be computed from

$$M = \int_{E_2} \frac{\partial s}{\partial n} d\sigma \tag{50}$$

where n is the outward normal direction and σ a line element along edge E_2 in $\mathcal{D}$ [11].

Conformal maps are angle preserving. The inverse of the conformal map $\vec{u} : \mathcal{D} \mapsto \mathcal{R}$ is also conformal and maps an orthogonal grid in the rectangle $\mathcal{R}$ to an orthogonal grid in $\mathcal{D}$. Therefore, an algorithm to compute an orthogonal grid in the interior of $\mathcal{D}$ with a prescribed boundary grid point distribution at all four edges may consist of the following steps

1. Compute an initial boundary conforming grid in the interior of $\mathcal{D}$ without grid folding. This can be achieved using the grid control map based on normalized arc length.

2. Solve on this mesh $\triangle s = 0$ and $\triangle t = 0$ supplied with Dirichlet-Neumann boundary conditions. Compute the edge functions $t^o_{E_1}$,$t^o_{E_2}$,$s^o_{E_3}$ and $s^o_{E_4}$ and the conformal module M according to Eq.(50).

3. Map the edge functions in $\mathcal{P}$ to the rectangle $\mathcal{R}$, using Eq.(46), and compute an orthogonal boundary conforming grid in $\mathcal{R}$.

4. Map the orthogonal grid in $\mathcal{R}$ to $\mathcal{P}$, again using Eq.(46). This grid in $\mathcal{P}$ defines a grid control map that will create an orthogonal grid in the interior of $\mathcal{D}$.

Thus, a difficult problem of generating an orthogonal grid in a domain $\mathcal{D}$ can be effectively reduced to a simpler problem of generating an orthogonal grid in the rectangle $\mathcal{R}$. Unfortunately, there is no simple algorithm available to generate an orthogonal grid in the interior of a rectangle with prescribed boundary grid points at all four sides. The question of an existence proof for this problem still remains unanswered [14]. Numerical experiments indicate that even for a rectangle it is probably not possible to generate an orthogonal grid for all kinds of boundary grid point distributions [12].

However, if the boundary grid points have fixed positions on two adjacent edges of domain $\mathcal{D}$ but are allowed to move along the boundary of the other two edges, then a simple algorithm does exist to generate an orthogonal grid in $\mathcal{D}$. This result is similar to that reported by Kang and Leal [13], although they used the Ryskin-Leal grid generation equations [15] instead of the Poisson grid generation equations. For example, suppose that the boundary grid points are fixed at edges E_1 and E_3 and are allowed to move along edges E_2 and E_4. Then the algorithm becomes

Algorithm 4. Grid orthogonality.

1. Compute an initial boundary conforming grid in the interior of $\mathcal{D}$ without grid folding. Such a grid can be computed using the grid control map based on normalized arc length as described in **Algorithm 2**.

2. Solve on this mesh $\triangle s = 0$ and $\triangle t = 0$ supplied with Dirichlet-Neumann boundary conditions and compute the edge functions $t^o_{E_1}$,$t^o_{E_2}$,$s^o_{E_3}$ and $s^o_{E_4}$.

3. The initial position of the boundary grid points at edge E_2 corresponds with the edge function $t^o_{E_2}$. Move the boundary grid points along edge E_2 in such a way that there new position corresponds

with $t^o_{E_1}$. This is simply a matter of interpolation. The points along edge E_4 should be moved such that there new position corresponds with $s^o_{E_3}$.

4. Define the grid control map as $s(\xi,\eta) = s^o_{E_3}(\xi)$ and $t(\xi,\eta) = t^o_{E_1}(\eta)$.

5. Compute the corresponding orthogonal grid in $\mathcal{D}$ as described in **Algorithm 1**.

The grid in parameter space $\mathcal{P}$ is a simple non-uniform rectangular mesh. Such a mesh also corresponds to a non-uniform rectangular grid in the rectangle $\mathcal{R}$ so that the corresponding grid in $\mathcal{D}$ will be indeed orthogonal.

An illustration of this algorithm is shown in Fig.13 which consists of two grids in a channel with a circular arc. The lower part shows a grid obtained with **Algorithm 3**. The grid points are prescribed and there position is fixed while grid orthogonality is obtained at all four edges. The upper part shows an orthogonal grid obtained by **Algorithm 4**. The figure clearly demonstrates how the boundary grid points have to move in order to obtain an orthogonal grid.

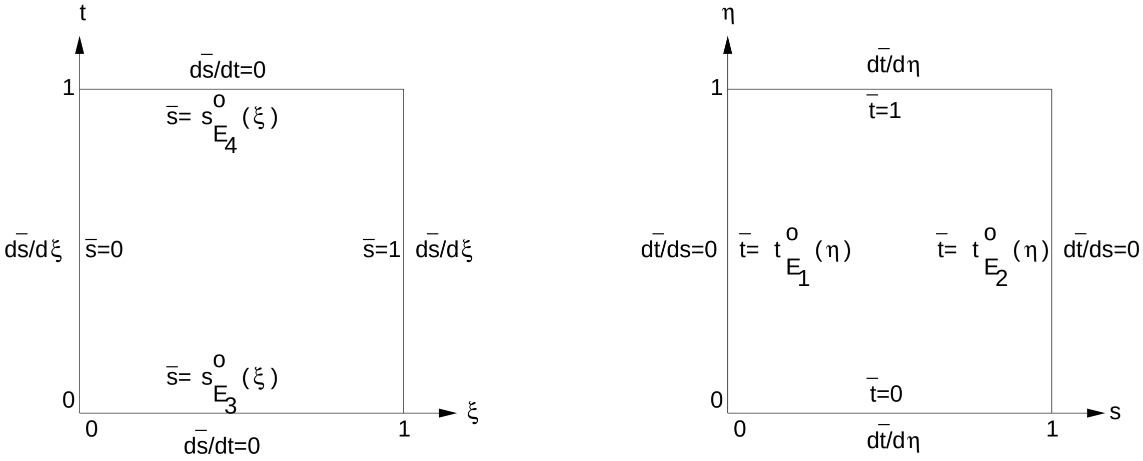
2.3.5 Complete grid control at the boundary

In Section 2.3.3 we have described the construction of a grid control map such that grid orthogonality is obtained at the boundary of $\mathcal{D}$. However, the method provides no precise control of the height of the first grid cells along the boundary. In general, the cell height distributions of the first grid cell along the boundary in $\mathcal{D}$ is fairly good as illustrated in Fig.8 and Fig.12. However, there are applications, specially in grid boundary layers for viscous flows, where not only grid orthogonality but also grid spacing should be precisely controlled. For example, it may be required that the first grid cell height is constant in the complete grid boundary layer, in spite of convex or concave parts of the boundary shape.

In order to have precise control about both grid orthogonality and grid cell height we have to consider more general grid control maps. Both the grid control map based on normalized arc length, defined by Eqs.(27),(28), and the one based on Dirichlet-Neumann boundary conditions, defined by Eqs.(42),(43), have the form

$$s = \overline{s}(\xi, t)\ ,\ t = \overline{t}(s, \eta) \tag{51}$$

Grid control maps of this type have the advantage that the two families of grid lines are independent: a grid line $\xi =$ constant in $\mathcal{C}$ is mapped to parameter space $\mathcal{P}$ as a curve defined by $s = \overline{s}(\xi, t)$ which will be mapped by the inverse of a harmonic map to a curve in domain $\mathcal{D}$. For given values of ξ and η, the corresponding grid point in $\mathcal{P}$ is found as the intersection point of the two curves $s = \overline{s}(\xi, t)$, $t = \overline{t}(s, \eta)$.



Boundary conditions for grid control function $s=\bar{s}(\xi,t)$

Boundary conditions for grid control function $t=\bar{t}(s,\eta)$

Figure 2: Boundary conditions for both control of orthogonality and first grid cell height.

When the boundary grid point distribution is changed in one set of opposite edges and remains unchanged in the other set, then one family of grid lines remains unchanged in both $\mathcal{P}$ and $\mathcal{D}$.

Suppose that grid orthogonality and first-cell height specification are required at all four edges. Then the boundary conditions for the grid control map defined by Eq.(51) are shown in Fig.2. The boundary condition $\partial\overline{s}/\partial t = 0$ at E_3 and E_4 in (ξ, t)-space is needed for grid orthogonality at E_3 and E_4 in $\mathcal{D}$. The values of $\partial\overline{s}/\partial\xi$ at E_1 and E_2 in (ξ, t)-space control the cell height of the first grid cells at E_1 and E_2 in $\mathcal{D}$. Similarly, the boundary condition $\partial\overline{t}/\partial s = 0$ at E_1 and E_2 in (s, η)-space is needed for grid orthogonality at E_1 and E_2 in $\mathcal{D}$. The values of $\partial\overline{t}/\partial\eta$ at E_3 and E_4 in (s, η)-space control the cell height of the first grid cells at E_3 and E_4 in $\mathcal{D}$.

The algorithm for complete control of both grid orthogonality and cell height along the four edges becomes

Algorithm 5. Complete grid control at boundary.

1. Use **Algorithm 3** to compute an initial boundary conforming grid in the interior of $\mathcal{D}$ which is orthogonal at the boundary. The corresponding grid control map is based on Eqs.(42),(43).

2. Compute $\partial\overline{s}/\partial\xi$ at E_1 and E_2 in (ξ, t)-space from Eq.(42). Compute $\partial\overline{t}/\partial\eta$ at E_3 and E_4 in (s, η)-space from Eq.(43). Adapt $\partial\overline{s}/\partial\xi$ and $\partial\overline{t}/\partial\eta$ such that the grid in domain $\mathcal{D}$ gets the desired grid cell height distribution along the corresponding edges. Note that the harmonic map and its inverse only depend on the shape of domain $\mathcal{D}$. Therefore it is possible to compute how a change, in for example $\partial\overline{s}/\partial\xi$ at E_1 in (ξ, t)-space, will change the cell height along edge E_1 in $\mathcal{D}$.

3. Compute $s = \overline{s}(\xi, t)$ in (ξ, t) space such all boundary conditions are satisfied. Also compute $t = \overline{t}(s, \eta)$ in (s, η) space such that all boundary conditions are satisfied. Compute the corresponding grid control map $\vec{s} : \mathcal{C} \mapsto \mathcal{P}$ for given values of ξ and η. The corresponding grid point in $\mathcal{P}$ is found as the intersection point of the two curves $s = \overline{s}(\xi, t)$, $t = \overline{t}(s, \eta)$.

4. Compute the corresponding interior grid in $\mathcal{D}$ as described in **Algorithm 1**.

The question remains how to compute $s = \overline{s}(\xi, t)$ and $t = \overline{t}(s, \eta)$ such that all boundary conditions are fulfilled. The boundary data $\overline{s}(0, t)$, $\overline{s}(1, t)$,$\overline{s}(\xi, 0)$,$\overline{s}(\xi, 1)$ and $\partial\overline{s}/\partial\xi(0, t)$, $\partial\overline{s}/\partial\xi(1, t)$, $\partial\overline{s}/\partial t(\xi, 0)$, $\partial\overline{s}/\partial t(\xi, 1)$, can be interpolated by using a bicubically blended Coon's patch [25, 26]. However, the use of such an algebraic interpolation method has a severe shortcoming because twist vectors have to be specified at the four corners.In general, the tangent boundary conditions $\partial\overline{s}/\partial\xi$, $\partial\overline{s}/\partial t$, are conflicting at a corner when the two edges of domain $\mathcal{D}$ are not orthogonal at the corresponding vertex. In that case the twist vector is not well-defined at the corner. Because of the conflicting tangent boundary conditions at the corners, we prefer to apply an elliptic partial differential equation to interpolate the boundary data. A fourth-order elliptic operator is needed to satisfy all boundary conditions. Therefore the biharmonic equations

$$\triangle\triangle\overline{s} = 0, \tag{52}$$

where $\triangle = \partial^2/\partial\xi^2 + \partial^2/\partial t^2$, and

$$\triangle\triangle\overline{t} = 0, \tag{53}$$

where $\triangle = \partial^2/\partial s^2 + \partial^2/\partial\eta^2$ is a proper choice. The advantage of the use of the biharmonic equation to interpolate the boundary data is that the solution is always a smooth function even when the tangent boundary conditions are conflicting at the corners. A disadvantage is that the biharmonic operator does not fulfill a maximum principle. When there is a grid boundary layer along for example edge E_1 in $\mathcal{D}$ then the monotonic boundary functions $s^o_{E_3}(\xi)$ and $s^o_{E_4}(\xi)$ have very small values in a large part of the interval $0 \ll \xi \ll 1$. In that case, the solution of the biharmonic equation may have small negative values in the interior, which is of course unacceptable. This problem is solved by applying a change in variables. In fact we solve $\triangle\triangle f = 0$ where $f : \overline{s} \in [0, 1] \mapsto [0, 1]$ is a monotonic function which maps a unit interval onto a unit interval. The boundary conditions for $\overline{s}$ are transferred to corresponding boundary conditions for f. After solving $\triangle\triangle f = 0$, we find f values at interior grid points and the corresponding $\overline{s}$ values are found using f^{-1}. In practice, we define $f : \overline{s} \in [0, 1] \mapsto [0, 1]$ such that $f(\frac{1}{2}(s^o_{E_3}(\xi) + s^o_{E_4}(\xi))) \equiv \xi$. A similar change in variable is used for the grid control function $t = \overline{t}(s, \eta)$.

The biharmonic equations are solved by the black-box biharmonic solver BIHAR [28] which is available on the electronic mathematical NETLIB library.

Algorithm 5 describes complete boundary control for both grid orthogonality and grid spacing. It is also possible to have only grid spacing control without boundary grid orthogonality. In that case, **Algorithm 2** must be used instead of **Algorithm 3** in the first step of **Algorithm 5**. An illustration of the result of grid spacing control is shown in Fig.14 through Fig.17. The same test case was also used by Eiseman [17]. The upper side of the domain is convex, the lower side is concave. The boundary grid points are prescribed and evenly distributed. Fig.14 shows a Laplace grid with the typical behaviour near the convex and concave parts of the boundary. Fig.15 shows the grid with mesh spacing control at the upper and lower side. Clearly, the cell height becomes constant at both the convex and concave sides. Fig.16 shows the grid with only grid orthogonality at the convex and concave sides and Fig.17 shows the grid with combined control of both mesh spacing and grid orthogonality at the convex and concave sides.

2.4 Best practices

In this section we show how the previous discussed algorithms work in practice. The chosen examples mainly concern simple well-defined geometries so that the reader is able to recompute the generated grids. In all cases, the boundary grid points are pre-defined and their location is fixed.

Example 1. Triangular domain.

This example illustrates **Algorithm 3** to obtain grid orthogonality at the boundary. Fig.19 shows the grid obtained with **Algorithm 2**. The corresponding grid control map, based on Eqs.(27),(28), is shown in Fig.18 as a grid in parameter space $\mathcal{P}$. Notice that the grid lines are straight in $\mathcal{P}$. Fig.21 shows the grid in parameter space obtained by solving $\triangle s = 0$ and $\triangle t = 0$ on the grid shown in Fig.19 supplied with Neumann boundary conditions on the two bottom edges of the triangle. It should be noticed that although this grid control map is completely different from the grid control map shown in Fig.18, the corresponding grid in the interior of the triangle will still be the same. Fig.22 shows the new grid control map based on Eqs.(42),(43). Thus the position of the boundary grid points is the same in both Fig.22 and Fig.21. Notice that the grid is orthogonal at the left and bottom edge of $\mathcal{P}$. These two edges in $\mathcal{P}$ correspond with the two bottom edges of the triangle. The corresponding grid is shown in Fig.23. The grid is clearly orthogonal at the two bottom edges of the triangle. Fig.24 shows the nice behaviour of the grid near the O-type singularity.

Example 2. Circular domain.

This example illustrates **Algorithm 5** for complete grid control at the boundary. The prescribed boundary grid points are evenly spaced as shown in Fig.26. The grid in parameter space $\mathcal{P}$, based on Eqs.(27),(28), is shown in Fig.25 and is thus uniform so that the corresponding grid in Fig.26 is a Laplace grid. Fig.27 shows the grid in parameter space obtained by solving $\triangle s = 0$ and $\triangle t = 0$ supplied with Neumann boundary conditions at all four sides. Fig.28 shows the new grid control map based on Eqs.(42),(43). This grid in parameter space is no longer uniform but remains rectangular because of the symmetry in both geometry and boundary grid. The corresponding grid in physical space, shown in Fig.29, is thus orthogonal as explained in Section 2.3.4. Notice the bad mesh spacing along the boundary of this orthogonal grid. The adapted grid in parameter space to obtain also a good mesh spacing is shown in Fig.30. This adapted grid is obtained by the method described in Section 2.3.5. Fig.31 shows the corresponding grid in physical space and demonstrates the successful combination of boundary grid orthogonality and good mesh spacing.

Example 3. Domain bounded by semi-circles on the four sides of the unit square.

This geometry is also used by Duraiswami [11] and Eça [12]. The prescribed boundary grid points are no longer evenly spaced but dense near the four corners of the domain. Fig.32 shows the grid in parameter space based on Eqs.(27),(28). Fig.33 shows the corresponding grid in physical space. Fig.34 shows the grid in parameter space obtained by solving $\triangle s = 0$ and $\triangle t = 0$ supplied with Neumann boundary conditions at all four sides. Fig.35 shows the new grid control map based on Eqs.(42),(43). This grid in parameter space is rectangular because of the symmetry in both geometry and boundary grid. The corresponding grid in physical space, shown in Fig.36, is thus orthogonal as explained in Section 2.3.4. The adapted grid in parameter space to obtain also a good mesh spacing is shown in Fig.37 and Fig.38 shows the result in physical space.

Example 4. Degenerated domains.

Two degenerated domains are considered: a lune bounded by the curves $y = x(1-x)$ and $y = -x(1-x^2)$ and a trilateral. The lune has two degenerated edges, the trilateral only one. Both geometries are also used by Duraiswami [11] and Eça [12].

In case of the lune, an evenly spaced boundary grid point distribution is used so that the grid in parameter space based on Eqs.(27),(28) is uniform and the corresponding grid in physical space is harmonic. See Fig.39 and Fig.40. Fig.41 shows the grid in parameter space obtained by solving $\triangle s = 0$ and $\triangle t = 0$ supplied with Neumann boundary conditions at the two non-degenerated edges. Notice the large change in the position of the boundary grid points in parameter space compared to the initial uniform grid. Fig.42 shows the new grid control map based on Eqs.(42),(43). This grid in parameter space is almost rectangular. The corresponding grid in physical space, shown in Fig.43, is therefore almost orthogonal.

For the trilateral, we only show the final grid in parameter space, obtained by **Algorithm 5**, and the corresponding grid in physical space. See Fig.44 and Fig.45.

Example 5. Navier-Stokes grid around a complex artificial boundary.

This example is used to demonstrate the robustness of the proposed algorithms. Fig.46 shows the grid in parameter space based on Eqs.(27),(28) and Fig.47 shows the corresponding C-type Navier-Stokes grid in physical space. Fig.49 shows the grid in parameter space obtained by solving $\triangle s = 0$ and $\triangle t = 0$ with Neumann boundary conditions at the lower boundary of the domain (three edges). Fig.50 shows the new grid in parameter space based on Eqs.(42),(43). The grid is orthogonal at the left,right and lower side of the parameter space. The corresponding grid in physical space is shown in Fig.51 and Fig.52.

3 Surface grid generation

The concepts of harmonic maps and grid control maps as used for grid generation in 2D domains can also be used for grid generation on surfaces in 3D.

Consider a surface $\mathcal{S}$ bounded by four edges E_1, E_2, E_3, E_4. Let (E_1, E_2) and (E_3, E_4) be the two pairs of opposite edges as shown in Fig.3.

A harmonic map is defined as a differentiable one-to-one map from $\mathcal{S}$ onto a unit square such that

1. the boundary of $\mathcal{S}$ is mapped onto the boundary of the unit square,
2. the vertices of $\mathcal{S}$ are mapped, in the proper sequence, onto the corners of the unit square,
3. the two components of the map are harmonic functions on $\mathcal{S}$. This means that the two components obey the Laplace-Beltrami equations for surfaces (see Section 5, Part II of the Appendix of this Handbook).

Let $\vec{s} : \mathcal{S} \mapsto \mathcal{P}$ be a harmonic map where the parameter space $\mathcal{P}$ is the unit square in a two-dimensional space with Cartesian coordinates $\vec{s} = (s,t)^T$. Thus $\triangle s = 0$ and $\triangle t = 0$ where $\triangle$ is the Laplace-Beltrami operator for surfaces [23].

The problem of generating an appropriate grid on surface $\mathcal{S}$ can be effectively reduced to a simpler problem of generating an appropriate grid in the parameter space $\mathcal{P}$, which can after that be mapped on $\mathcal{S}$, by using the inverse of the harmonic map $\vec{x} : \mathcal{P} \mapsto \mathcal{S}$.

Define the computational space $\mathcal{C}$ as the unit square in a two-dimensional space with Cartesian coordinates $\vec{\xi} = (\xi, \eta)^T$. A grid control map $\vec{s} : \mathcal{C} \mapsto \mathcal{P}$ is defined as a differentiable one-to-one map from $\mathcal{C}$ onto $\mathcal{P}$ and maps a uniform grid in $\mathcal{C}$ to a , in general, non-uniform grid in $\mathcal{P}$.

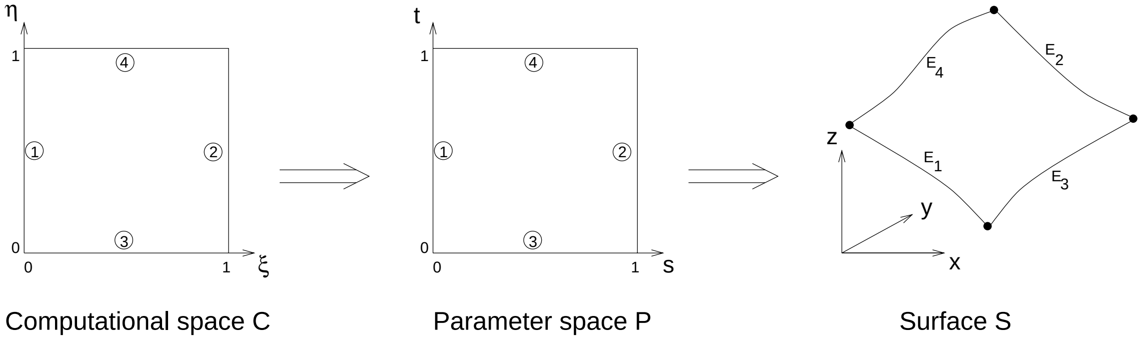


Figure 3: Composite map from computational (ξ, η) space to a surface $\mathcal{S}$ in Cartesian (x, y, z) space.

The composition of a grid control map $\vec{s} : \mathcal{C} \mapsto \mathcal{P}$ and the inverse of the harmonic map $\vec{x} : \mathcal{P} \mapsto \mathcal{S}$ defines a map $\vec{x} : \mathcal{C} \mapsto \mathcal{S}$ which transforms a uniform grid in $\mathcal{C}$ to a, in general, non-uniform grid on surface $\mathcal{S}$. The same ideas as used for 2D domains can be applied to construct appropriate grid control maps such that the corresponding surface grid has desired properties.

For example, assume that the boundary grid points are prescribed on surface $\mathcal{S}$ and suppose that it is desired to construct a boundary conforming grid on $\mathcal{S}$ which is orthogonal at all four edges. Then the same Neumann boundary conditions as used in Section 2.3.3. must be used to define the harmonic map. Furthermore, the grid control map must be defined by Eqs.(42),(43). Then the composite map defines a boundary conforming grid on $\mathcal{S}$ which is orthogonal at all four edges.

However, the numerical implementation of these ideas is different from the 2D case because the composite map no longer fulfills a simple Poisson system as defined by Eq.(18). There is an exception, namely when $\mathcal{S}$ is a minimal surface. A minimal surface has zero mean curvature, and its shape is a soap film bounded by its four edges. There is a famous theorem in Differential Geometry which states that the Laplace-Beltrami operator applied on the position vector of an arbitrary surface $\mathcal{S}$ obeys

$$\triangle \vec{x} = 2H\vec{n}, \tag{54}$$

where $\vec{n}$ is the unit vector normal to the surface and H is the mean curvature. (see Section 5, Part II of the Appendix of this Handbook or Dierkes et. al. [24] , Theorem 1, page 71). The requirement of zero mean curvature implies

$$\triangle \vec{x} = 0. \tag{55}$$

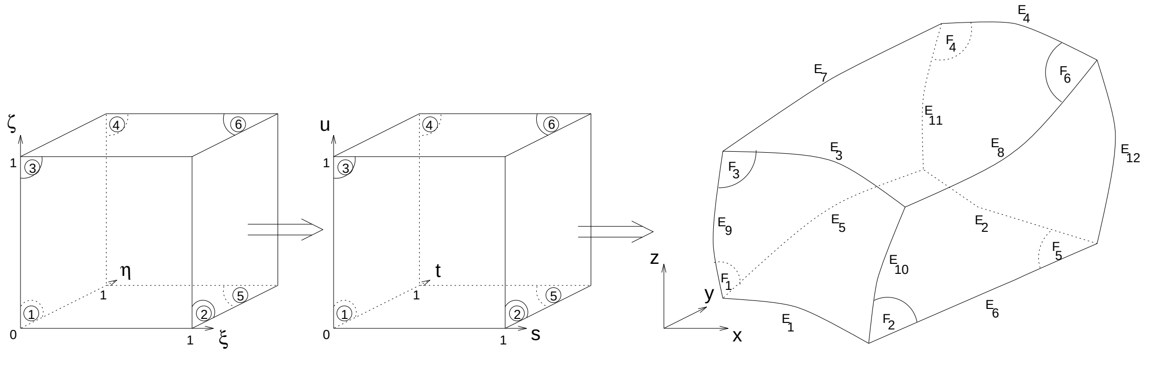
Thus for minimal surfaces we also have $\triangle s = 0$, $\triangle t = 0$ and $\triangle \vec{x} = 0$. Following the same derivation as in Section 2.1 for 2D domains, we find that the composite map obeys the same Poisson system given by Eq.(18) (for more details see [19]). Thus an interior grid point distribution on a minimal surface is found by solving Eq.(18) with the prescribed boundary grid points as Dirichlet boundary conditions. The only difference compared to the two-dimensional case is that now $\vec{x} = (x, y, z)^T$ instead of $\vec{x} = (x, y)^T$. The same ideas to construct appropriate grid control maps and their corresponding grids in 2D domains can also be directly applied to minimal surfaces. In fact, all previous discussed 2D examples are generated as minimal surface grids where the four boundary edges are lying in a plane in three-dimensional space.

Examples of characteristic minimal surface grids are shown in Fig.53 through Fig.57. Fig.53 is a so-called square Scherk surface [24]. Fig.54 shows what happens when the boundary edges of the Scherk surface are replaced by semi-circular arcs. Fig.55 and Fig.56 show the change in the shape of the minimal surface when these semi-circular arcs are bend together. Boundary orthogonality is imposed at all four sides for all these three cases. Because of the symmetry in both geometry and boundary grid point distribution, the generated surface grids are not only orthogonal at the boundary but also in the interior. Finally, Fig.57 is Schwarz's P-surface [24], which is in fact constructed as a collection of connected minimal surfaces.

In general, surface $\mathcal{S}$ is not a minimal surface but a parametrically defined surface with a prescribed geometrical shape given by a map $\vec{x} : \mathcal{Q} \mapsto \mathcal{S}$ where $\mathcal{Q}$ is some parameter space defined as a unit square in 2D. In order to construct, for example, a boundary conforming grid on $\mathcal{S}$ which is orthogonal at all four edges, we solve on an initial surface grid on $\mathcal{S}$ the Laplace-Beltrami equations with the same Neumann boundary conditions as used in Section 2.3.3. The solution can be written as a map $\vec{s} : \mathcal{Q} \mapsto \mathcal{P}$. The appropriate grid control map, defined by Eqs.(42),(43), defines a non-uniform grid in $\mathcal{P}$. The corresponding grid in $\mathcal{Q}$ can then be found by using the inverse map $\vec{s}^{-1} : \mathcal{P} \mapsto \mathcal{Q}$. This is done numerically in a way described in [19]. Once the corresponding grid in $\mathcal{Q}$ is found, then the corresponding surface grid on $\mathcal{S}$ is computed using the parametrization $\vec{x} : \mathcal{Q} \mapsto \mathcal{S}$. This new surface grid on $\mathcal{S}$ differs from the initial surface grid $\mathcal{S}$. The complete process should be repeated until the surface grid on $\mathcal{S}$ (and the corresponding grids in parameter space $\mathcal{P}$ and $\mathcal{Q}$) do not change anymore. In practice, only a few (2-5) iterations appear to be sufficient. After convergence, the final surface grid will not only be orthogonal at the boundary but is also independent of the parametrization and only depends on the shape of the surface and the position of the boundary grid points.

4 Volume grid generation

Consider a simply connected bounded domain $\mathcal{D}$ in three-dimensional space with Cartesian coordinates $\vec{x} = (x, y, z)^T$. Suppose that $\mathcal{D}$ is bounded by six faces $F_1, F_2, F_3, F_4, F_5, F_6$. Let (F_1, F_2) , (F_3, F_4) and



Computational space C — Parameter space P — Domain D

Figure 4: Composite mapping from computational (ξ, η, ζ) space to a domain $\mathcal{D}$ in Cartesian (x, y, z) space.

(F_5, F_6) be the three pairs of opposite faces. Furthermore, consider the twelve edges $\{E_i, i = 1 \ldots 12\}$ and assume that these edges are related to the six faces as shown in Fig.4

In 3D, a harmonic map is defined as a differentiable one-to-one map from $\mathcal{D}$ onto a unit cube such that

1. the boundary of $\mathcal{D}$ is mapped onto the boundary of the unit cube,

2. the vertices,edges and faces of $\mathcal{D}$ are mapped onto the corresponding vertices,edges and faces of the unit cube,

3. the three components of the map are harmonic functions in the interior of $\mathcal{D}$.

Let $\vec{s} : \mathcal{D} \mapsto \mathcal{P}$ be a harmonic map where the parameter space $\mathcal{P}$ is the unit cube in a three-dimensional space with Cartesian coordinates $\vec{s} = (s, t, u)^T$. Inside $\mathcal{D}$ the components obey

$$\triangle s = s_{xx} + s_{yy} + s_{zz} = 0 \ , \ \triangle t = t_{xx} + t_{yy} + t_{zz} = 0 \ , \ \triangle u = u_{xx} + u_{yy} + u_{zz} = 0 \ . \qquad (56)$$

Define the computational space $\mathcal{C}$ as the unit cube in a three-dimensional space with Cartesian coordinates $\vec{\xi} = (\xi, \eta, \zeta)^T$. A grid control map $\vec{s} : \mathcal{C} \mapsto \mathcal{P}$ is defined as a differentiable one-to-one map from $\mathcal{C}$ onto $\mathcal{P}$ and maps a uniform grid in $\mathcal{C}$ to a , in general, non-uniform grid in $\mathcal{P}$.

The composition of a grid control map $\vec{s} : \mathcal{C} \mapsto \mathcal{P}$ and the inverse of the harmonic map $\vec{x} : \mathcal{P} \mapsto \mathcal{D}$ defines a map $\vec{x} : \mathcal{C} \mapsto \mathcal{D}$ which transforms a uniform grid in $\mathcal{C}$ to a, in general, non-uniform grid in $\mathcal{D}$. As in 2D, the composite map obeys a quasi-linear system of elliptic partial differential equations, known as the Poisson grid generation equations, with control functions completely defined by the grid control map.

The derivation of the Poisson grid generation equations can be done along the same lines as for the 2D case. Suppose that the harmonic map and grid control map are defined so that the composite map exists. Introduce the three covariant base vectors

$$\vec{a}_1 = \vec{x}_\xi \ , \ \vec{a}_2 = \vec{x}_\eta \ , \ \vec{a}_3 = \vec{x}_\zeta . \tag{57}$$

and the covariant metric tensor components

$$a_{ij} = (\vec{a}_i, \vec{a}_j) \ , \ i = \{1,2,3\}, j = \{1,2,3\}. \tag{58}$$

The three contravariant base vectors $\vec{a}^1 = \nabla\xi = (\xi_x, \xi_y, \xi_z)^T$, $\vec{a}^2 = \nabla\eta = (\eta_x, \eta_y, \eta_z)^T$ and $\vec{a}^3 = \nabla\zeta = (\zeta_x, \zeta_y, \zeta_z)^T$ obey

$$(\vec{a}^i, \vec{a}_j) = \delta^i_j \ , \ i = \{1,2,3\}, j = \{1,2,3\}. \tag{59}$$

Define the contravariant metric tensor components

$$a^{ij} = (\vec{a}^i, \vec{a}^j) \ , \ i = \{1,2,3\}, j = \{1,2,3\}, \tag{60}$$

so that

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{12} & a_{22} & a_{23} \\ a_{13} & a_{23} & a_{33} \end{pmatrix} \begin{pmatrix} a^{11} & a^{12} & a^{13} \\ a^{12} & a^{22} & a^{23} \\ a^{13} & a^{23} & a^{33} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{61}$$

Define J^2 as the determinant of the covariant metric tensor.

Consider an arbitrary function $\phi = \phi(\xi, \eta, \zeta)$. Then ϕ is also defined in domain $\mathcal{D}$ and the Laplacian of ϕ can be expressed as

$$\begin{aligned} \triangle\phi = \frac{1}{J}\Big\{ \Big(Ja^{11}\phi_\xi + Ja^{12}\phi_\eta + Ja^{13}\phi_\zeta\Big)_\xi + \Big(Ja^{12}\phi_\xi + Ja^{22}\phi_\eta + Ja^{23}\phi_\zeta\Big)_\eta \\ + \Big(Ja^{13}\phi_\xi + Ja^{23}\phi_\eta + Ja^{33}\phi_\zeta\Big)_\zeta \Big\}. \end{aligned} \tag{62}$$

As in the two-dimensional case, substitution of $\phi \equiv \xi$, $\phi \equiv \eta$ and $\phi \equiv \zeta$ into this equation yields expressions for $\triangle\xi$, $\triangle\eta$ and $\triangle\zeta$. Combining these expressions with Eq.(62) gives

$$\triangle\phi = a^{11}\phi_{\xi\xi} + 2a^{12}\phi_{\xi\eta} + 2a^{13}\phi_{\xi\zeta} + a^{22}\phi_{\eta\eta} + 2a^{23}\phi_{\eta\zeta} + a^{33}\phi_{\zeta\zeta} + \triangle\xi\phi_\xi + \triangle\eta\phi_\eta + \triangle\zeta\phi_\zeta \tag{63}$$

Substitute $\phi = (s, t, u)^T$ in Eq.(63) and use the property that s, t and u are harmonic in domain $\mathcal{D}$, i.e. $\triangle s = 0$, $\triangle t = 0$ and $\triangle u = 0$. Then the following expressions for the Laplacian of ξ , η and ζ are found:

$$\begin{pmatrix} \triangle\xi \\ \triangle\eta \\ \triangle\zeta \end{pmatrix} = a^{11}\vec{P}_{11} + 2a^{12}\vec{P}_{12} + 2a^{13}\vec{P}_{13} + a^{22}\vec{P}_{22} + 2a^{23}\vec{P}_{23} + a^{33}\vec{P}_{33}, \tag{64}$$

where

$$\begin{aligned}
\vec{P}_{11} &= -T^{-1}\begin{pmatrix} s_{\xi\xi} \\ t_{\xi\xi} \\ u_{\xi\xi} \end{pmatrix} \; , \; \vec{P}_{12} = -T^{-1}\begin{pmatrix} s_{\xi\eta} \\ t_{\xi\eta} \\ u_{\xi\eta} \end{pmatrix} \; , \; \vec{P}_{13} = -T^{-1}\begin{pmatrix} s_{\xi\zeta} \\ t_{\xi\zeta} \\ u_{\xi\zeta} \end{pmatrix} \; , \\
\vec{P}_{22} &= -T^{-1}\begin{pmatrix} s_{\eta\eta} \\ t_{\eta\eta} \\ u_{\eta\eta} \end{pmatrix} \; , \; \vec{P}_{23} = -T^{-1}\begin{pmatrix} s_{\eta\zeta} \\ t_{\eta\zeta} \\ u_{\eta\zeta} \end{pmatrix} \; , \; \vec{P}_{33} = -T^{-1}\begin{pmatrix} s_{\zeta\zeta} \\ t_{\zeta\zeta} \\ u_{\zeta\zeta} \end{pmatrix} ,
\end{aligned} \tag{65}$$

and the matrix T is defined as

$$T = \begin{pmatrix} s_\xi & s_\eta & s_\zeta \\ t_\xi & t_\eta & t_\zeta \\ u_\xi & u_\eta & u_\zeta \end{pmatrix} . \tag{66}$$

The 18 coefficients of the six vectors $\vec{P}_{11}$,$\vec{P}_{12}$,$\vec{P}_{13}$, $\vec{P}_{22}$,$\vec{P}_{23}$,$\vec{P}_{33}$ are so called control functions. Thus the 18 control functions are completely defined and easily computed for a given grid control map $\vec{s} = \vec{s}(\vec{\xi})$.

Finally, substitution of $\phi \equiv \vec{x}$ in Eq.(63) and using the fact that $\triangle\vec{x} \equiv 0$ we arrive at the following equation

$$a^{11}\vec{x}_{\xi\xi} + 2a^{12}\vec{x}_{\xi\eta} + 2a^{13}\vec{x}_{\xi\zeta} + a^{22}\vec{x}_{\eta\eta} + 2a^{23}\vec{x}_{\eta\zeta} + a^{33}\vec{x}_{\zeta\zeta} + \triangle\xi\vec{x}_\xi + \triangle\eta\vec{x}_\eta + \triangle\zeta\vec{x}_\zeta = 0. \tag{67}$$

The final form of the Poisson grid generation system can now be derived from this equation by substitution of Eq.(64),by multiplication with J^2, and by expressing the contravariant tensor components in the covariant tensor components according to Eq.(61). The result can be written as:

$$\begin{aligned}
& \alpha^{11}\vec{x}_{\xi\xi} + 2\alpha^{12}\vec{x}_{\xi\eta} + 2\alpha^{13}\vec{x}_{\xi\zeta} + \alpha^{22}\vec{x}_{\eta\eta} + 2\alpha^{23}\vec{x}_{\eta\zeta} + \alpha^{33}\vec{x}_{\zeta\zeta} \\
+ & \left(\alpha^{11}P^1_{11} + 2\alpha^{12}P^1_{12} + 2\alpha^{13}P^1_{13} + \alpha^{22}P^1_{22} + 2\alpha^{23}P^1_{23} + \alpha^{33}P^1_{33}\right)\vec{x}_\xi \\
+ & \left(\alpha^{11}P^2_{11} + 2\alpha^{12}P^2_{12} + 2\alpha^{13}P^2_{13} + \alpha^{22}P^2_{22} + 2\alpha^{23}P^2_{23} + \alpha^{33}P^2_{33}\right)\vec{x}_\eta \\
+ & \left(\alpha^{11}P^3_{11} + 2\alpha^{12}P^3_{12} + 2\alpha^{13}P^3_{13} + \alpha^{22}P^3_{22} + 2\alpha^{23}P^3_{23} + \alpha^{33}P^3_{33}\right)\vec{x}_\zeta = 0,
\end{aligned} \tag{68}$$

with

$$\begin{aligned}
\alpha^{11} &= a_{22}a_{33} - a_{23}^2 \; , \; \alpha^{12} = a_{13}a_{23} - a_{12}a_{33} \; , \; \alpha^{13} = a_{12}a_{23} - a_{13}a_{22} \; , \\
\alpha^{22} &= a_{11}a_{33} - a_{13}^2 \; , \; \alpha^{23} = a_{13}a_{12} - a_{11}a_{23} \; , \; \alpha^{33} = a_{11}a_{22} - a_{12}^2,
\end{aligned} \tag{69}$$

and

$$\begin{aligned}
a_{11} &= (\vec{x}_\xi, \vec{x}_\xi) \; , \; a_{12} = (\vec{x}_\xi, \vec{x}_\eta) \; , \; a_{13} = (\vec{x}_\xi, \vec{x}_\zeta) \; , \\
a_{22} &= (\vec{x}_\eta, \vec{x}_\eta) \; , \; a_{23} = (\vec{x}_\eta, \vec{x}_\zeta) \; , \; a_{33} = (\vec{x}_\zeta, \vec{x}_\zeta) \; .
\end{aligned} \tag{70}$$

This equation, together with the expressions for the control functions P_{ij}^k given by Eq.(65), forms the 3D grid generation system. For a given grid control map, so that the 18 control functions in Eq.(68) are given functions of (ξ, η, ζ), boundary conforming grids in the interior of domain $\mathcal{D}$ are computed by solving this quasi-linear system of elliptic partial differential equations with prescribed boundary grid points as Dirichlet boundary conditions.

The construction of appropriate grid control maps for 3D domains is less well developed than for 2D domains. In [19], a grid control map has been proposed which works surprisingly well for many applications. The grid control map is the 3D extension of the 2D grid control map defined by Eqs.(27),(28). The map $\vec{s} : \mathcal{C} \mapsto \mathcal{P}$ is defined by

$$s = s_{E_1}(\xi)(1-t)(1-u) + s_{E_2}(\xi)t(1-u) + s_{E_3}(\xi)(1-t)u + s_{E_4}(\xi)tu, \tag{71}$$

$$t = t_{E_5}(\eta)(1-s)(1-u) + t_{E_6}(\eta)s(1-u) + t_{E_7}(\eta)(1-s)u + t_{E_8}(\eta)su, \tag{72}$$

$$u = u_{E_9}(\zeta)(1-s)(1-t) + u_{E_{10}}(\zeta)s(1-t) + u_{E_{11}}(\zeta)(1-s)t + u_{E_{12}}(\zeta)st. \tag{73}$$

where the twelve edge functions $s_{E_1}, \ldots, u_{E_{12}}$ measure the normalized arc length along the corresponding twelve edges of domain $\mathcal{D}$ (see Fig.4).

Eq.(71) implies that a grid plane $\xi =$ constant is mapped to the parameter space $\mathcal{P}$ as a bilinear surface: s is a bilinear function of t and u. Similarly, Eq.(72) and Eq.(73) imply that grid planes $\eta =$ constant and $\zeta =$ constant are also mapped to the parameter space $\mathcal{P}$ as bilinear surfaces. For a given computational coordinate (ξ, η, ζ) the corresponding (s, t, u) value is found as the intersection point of three bilinear surfaces. Newton iteration is used to compute the intersection points. It can be easily verified that two bilinear surfaces corresponding to two different ξ-values will never intersect in parameter space $\mathcal{P}$. The same is true for two different η or ζ values. This observation indicates that the grid control map is a differentiable one-to-one mapping.

An illustration of a volume grid computed by solving Eq.(68), with the grid control map defined by Eqs.(71),(72),(73), is shown in Fig.58 through Fig.61. The domain is a semi-torus. The prescribed boundary grid points on the surface of the semi-torus are shown in Fig.58. Fig.59 shows the surface grid on the two exterior circular grid planes. Fig.60 shows the computed interior grid depicted on some internal circular planes. Fig.61 shows the computed interior grid on the circular plane exactly halfway inside the torus. The mesh spacing of the interior grid is excellent despite the concave boundary. The angles between the interior grid lines and the boundary surface are reasonable but no longer orthogonal. This is not surprising because the grid control map provides no control about the angle distribution between interior grid lines and the boundary of the domain.

5 Research issues and summary

The grid generation systems of elliptic quasi-linear second-order partial differential equations are the familiar so-called Poisson systems with control functions to be specified. In this chapter, a Poisson system is considered as a system of partial differential equations which the composition of a grid control map and the inverse of a harmonic map has to obey. The control functions in the Poisson system are then completely defined by the grid control map. Boundary conforming grids in physical space are computed by solving the Poisson system with control functions specified by a grid control map.

One of the main advantages of this approach is that the method is non-iterative. If an appropriate grid control map has been constructed then the corresponding grid control functions of the Poisson system are computed and their values remain unchanged during the solution of the Poisson system. Another advantage is that the construction of an appropriate grid control map can be considered as a numerical implementation of the constructive proof for the existence of the desired grid in physical space. If the grid control map is one-to-one then the composition of the grid control map and the inverse of the harmonic maps exist so that the solution of the Poisson system is well-defined.

In two dimensions, boundary orthogonality is obtained by applying Dirichlet-Neumann boundary conditions for the harmonic map. In that case, the harmonic map is quasi-conformal. This property shows the relation with orthogonal grid generation.

The use of harmonic maps and grid control maps for surface grid generation is also shortly described. The two-dimensional Poisson systems can be directly extended to surface grid generation on minimal surfaces (soap films). The extension to volume grid generation is also given.

The construction of appropriate grid control maps such that the corresponding grid in physical space has desired properties is the main issue of this chapter. The chosen examples concern mainly simple well-defined geometries so that the reader is able to recompute the grids. However, the elliptic grid generation methods described in this chapter have been implemented in ENGRID, NLR's multi-block grid generation code [20, 21, 22], and are nowadays used on a routinely basis to construct Euler or Navier-Stokes grids in blocks and block-faces with complex geometrical shapes.

The construction of appropriate grid control maps for 3D domains is less well developed than for 2D domains and surfaces. Further investigation is expected in this direction.

References

[1] Thompson, J.F., Warsi, Z.U.A., and Mastin, C.W. 1985. *Numerical Grid Generation: Foundations and Applications*. Elsevier, New York.

[2] Warsi, Z.U.A. 1982. Basic Differential Models for Coordinate Generation, In *Proceedings Numerical Grid Generation*. ed. J.F. Thompson , p. 41-77. North-Holland,Amsterdam.

[3] Sonar, T. 1989. *Grid Generation Using Elliptic Partial Differential Equations*. DFVLR-Forschungsbericht 89-15.

[4] Knupp, P., Steinberg, S. 1993. *Fundamentals of Grid Generation*. CRC Press, Boca Raton.

[5] Sorenson, R.L., and Steger, J.L. 1980. Numerical Generation of Two-Dimensional Grids by use of Poisson Equations with Grid Control. In *Numerical Grid Generation Techniques*. ed. R.E. Smith, NASA-CP-2166. p. 449-461.

[6] Sorenson, R.L., Alter, S.J. 1995. 3DGRAPE/AL: The Ames/Langley Technology Update. In *Surface Modeling, Grid Generation, and Related Issues in Computational Fluid Dynamic (CFD) Solutions*. ed. Y.K. Choo, NASA-CP-3291. p.447-462.

[7] Chawner, J.R., Steinbrenner, J.P. 1995. Automatic Structured Grid Generation using GRIDGEN In *Surface Modeling, Grid Generation, and Related Issues in Computational Fluid Dynamic (CFD) Solutions*. ed. Y.K. Choo, NASA-CP-3291. p.463-476.

[8] Thompson, J.F., 1988. A Composite Grid Generation Code for General 3D Regions–the EAGLE Code. *AIAA Journal*, 26,Vol.3,p.271.

[9] Mastin, C. Wayne. 1995. Multilevel Elliptic Smoothing of Large Three-Dimensional Grids. In *Surface Modeling, Grid Generation, and Related Issues in Computational Fluid Dynamic (CFD) Solutions*. ed. Y.K. Choo, NASA-CP-3291. p.689-696.

[10] Roache, P.J., and Steinberg, S. 1985. A new Approach to Grid Generation Using a Variational Formulation. In *AIAA 7th Computational Fluid Dynamics Conference*. AIAA paper 85-1527,pp. 360-370.

[11] Duraiswami, R., Prosperetti, A. 1992. Orthogonal Mapping in two dimensions. *J.Comput.Phys.* 98:254:268.

[12] Eça, L. 1996. 2D Orthogonal Grid Generation with Boundary Point Distribution Control. *J.Comput.Phys.* 125:440-453.

[13] Kang, I.S., Leal, L.G. 1992. Orthogonal Grid Generation in a 2D Domain via the Boundary Integral Technique. *J.Comput.Phys.* 102:78-87.

[14] Oh, H.J., Kang, I.S. 1994. A Non-iterative Scheme for Orthogonal Grid generation with Control Function and Specified Boundary Correspondence on Three Sides. *J.Comput.Phys.* 112:138-148.

[15] Ryskin, G,. and Leal, L.G. 1983. Orthogonal Mapping. *J.Comput.Phys.* 50:71-100.

[16] Hsu, K., and Lee, S.L. 1991. A Numerical Technique for Two-Dimensional Grid Generation with Grid Control at All of the Boundaries. *J.Comput.Phys.* 96:451-469.

[17] Takahashi, S. and Eiseman P.R. 1994. Adaptive Grid Movement with respect to Boundary Curvature In *Numerical Grid Generation in Computational Fluid Dynamics and Related Fields, 4th International Grid Conference*. ed. N.P. Weatherill et. al. p. 563. Pineridge Press Limited, Swansea Wales (UK).

[18] Winslow, A. 1967. Numerical Solution of the Quasilinear Poisson Equations in a Nonuniform Triangle Mesh. *J.Comput.Phys* 2:149-172.

[19] Spekreijse, S.P. 1995. Elliptic Grid Generation Based on Laplace Equations and Algebraic Transformations. *J.Comput.Phys.* 118:38-61.

[20] Spekreijse, S.P., and Boerstoel, J.W. 1996. Multiblock Grid Generation. Part 1: Elliptic Grid Generation Methods for Structured Grids. In *Computational Fluid Dynamics*. VKI-Lecture-Series 1996-06. ed. H. Deconinck, p.1-39. Von Karman Institute for Fluid Dynamics.

[21] Spekreijse, S.P., and Boerstoel, J.W. 1996. Multiblock Grid Generation. Part 2: Multiblock Aspects. In *Computational Fluid Dynamics*. VKI-Lecture-Series 1996-06. ed. H. Deconinck, p.1-48. Von Karman Institute for Fluid Dynamics.

[22] Boerstoel, J.W., Kassies, A., Kok, J.C., Spekreijse, S.P. 1996. ENFLOW, A Full-Functionality System of CFD Codes for Industrial Euler/Navier-Stokes Flow Computations. In *Proceedings 2nd International Symposium on Aeronautical Science and Technology*, IASTTI’96, Jakarta,Indonesia.

[23] Kreyszig, E. 1991. *Differential Geometry*, Dover Publications, Inc. New York.

[24] Dierkes, U., Hildebrandt, S., Kuster, A., and Wohlrab, O. 1991. *Minimal Surfaces I*. Grundlehren der Mathematischen Wissenschaften 295, Springer Verlag, Berlin.

[25] Farin, G. 1990. *Curves and Surfaces for Computer Aided Geometric Design, A practical guide*.Academic Press, San Diego.

[26] Yamaguchi, F. 1988. *Curves and Surfaces in Computer Aided Geometric Design*. Springer Verlag, Berlin.

[27] Henrici, P. 1986. *Applied and Computational Complex Analysis, Vol. III*. Wiley, New York.

[28] Bjorstad, P.E. 1980. *Numerical Solution of the Biharmonic Equation.* Ph.D.thesis,Stanford University.

[29] Arcilla, A.S. (ed) et. al. 1991. *Numerical Grid Generation in Computational Fluid Dynamics and Related Fields*, Proceedings of the 3th International Conference, Barcelona, Spain, North-Holland.

[30] Weatherill, N.P. (ed) et. al. 1994. *Numerical Grid Generation in Computational Fluid Dynamics and Related Fields*, Proceedings of the 4th International Conference, Swansea, Wales, Pineridge Press.

[31] Soni, B.K. (ed) et. al. 1996. *Numerical Grid Generation in Computational Field Simulations*, Proceedings of the 5th International Conference, Mississippi State University,NSF Engineering Research Center for Computational Field Simulation.

[32] Weatherill, N.P.(ed) 1990. *Numerical Grid Generation,* VKI-Lecture-Series 1990-06, Von Karman Institute for Fluid Dynamics.

[33] Weatherill, N.P.(ed) 1994. *Grid Generation,* VKI-Lecture-Series 1994-02, Von Karman Institute for Fluid Dynamics.

[34] Smith, R.E. (ed) 1992. *Software Systems for Surface Modeling and Grid Generation*, NASA-CP-3143, Proceedings of a workshop held at NASA Langley, Hampton, VA.

[35] Choo, Y. K. (ed) 1995. *Surface Modeling, Grid Generation, and Related Issues in Computational Fluid Dynamic (CFD) Solutions*, NASA-CP-3291, Proceedings of a workshop held at NASA Lewis, Cleveland, Ohio.

6 Further information

The book of Thompson,Warsi and Mastin [1] is still the best introduction to elliptic grid generation systems. Also the book of Knupp and Steinberg [4] is a valuable source about the fundamentals of structured grid generation and related topics, like Tensor Analysis and Differential Geometry. The book of Kreyszig [23] and Dierkes et.al. [24] are excellent textbooks about Differential Geometry and Tensor Analysis.

The proceedings of the grid generation conferences [29, 30, 31], the VKI lecture series about grid generation [32, 33], and the NASA conference publications [34, 35] contain a lot of useful information about the application of elliptic grid generation systems, often embedded in multi-block grid generation systems.

The Journal of Computational Physics provides many good more or less fundamental articles about elliptic grid generation systems.

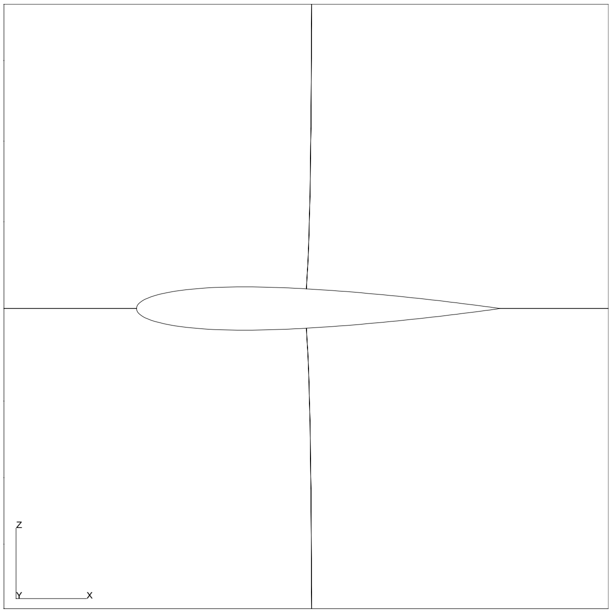


Figure 5: Domain boundaries near NACA0012 airfoil. The location of the grid points on the domain boundaries are prescribed and fixed.

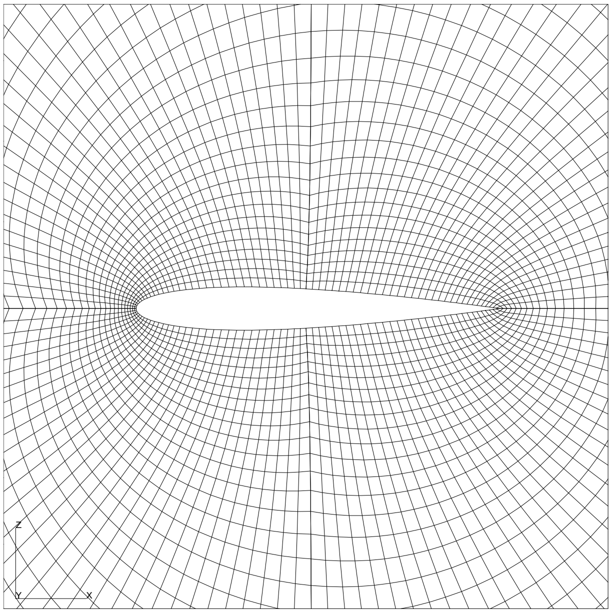


Figure 6: Laplace grid. Grid control map is the identity map.

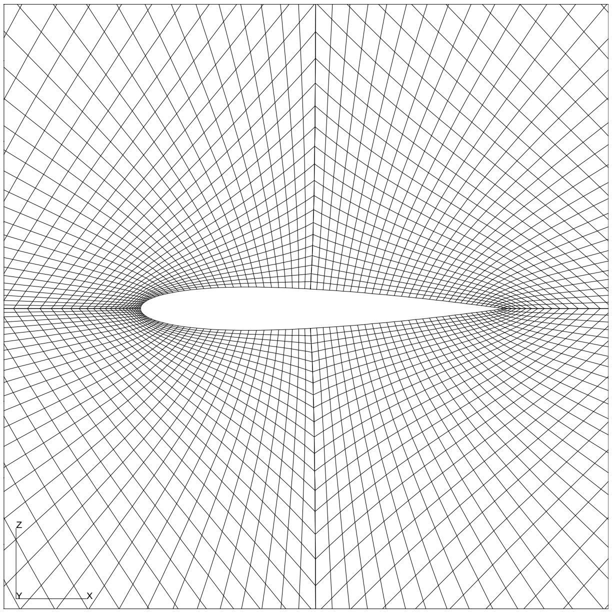


Figure 7: Arc length based grid.

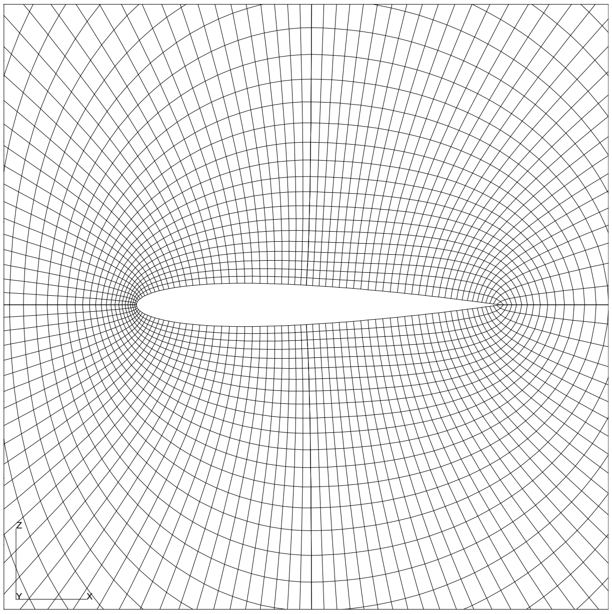


Figure 8: Grid with boundary orthogonality. Boundary orthogonality makes the grid smooth across internal domain boundaries.

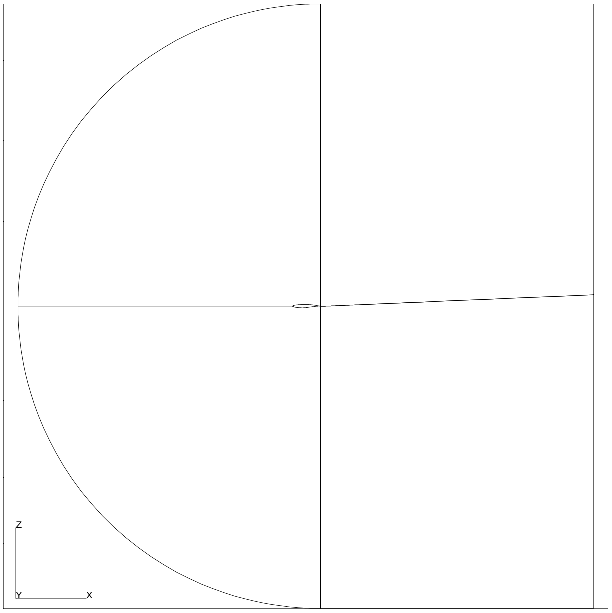


Figure 9: Region about RAE2822 airfoil subdivided into four domains.

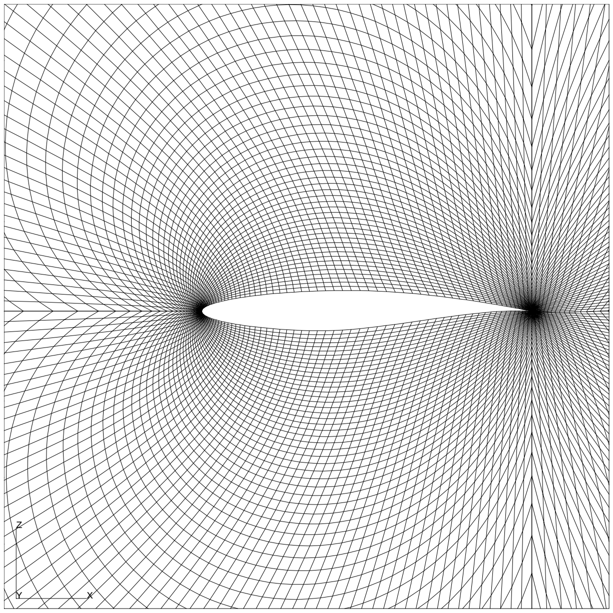


Figure 10: Laplace grid near airfoil. Grid control map is the identity map.

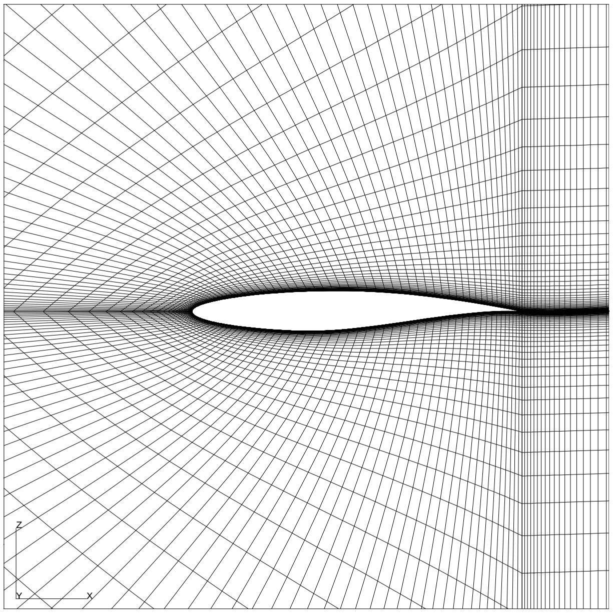


Figure 11: Arc length based grid.

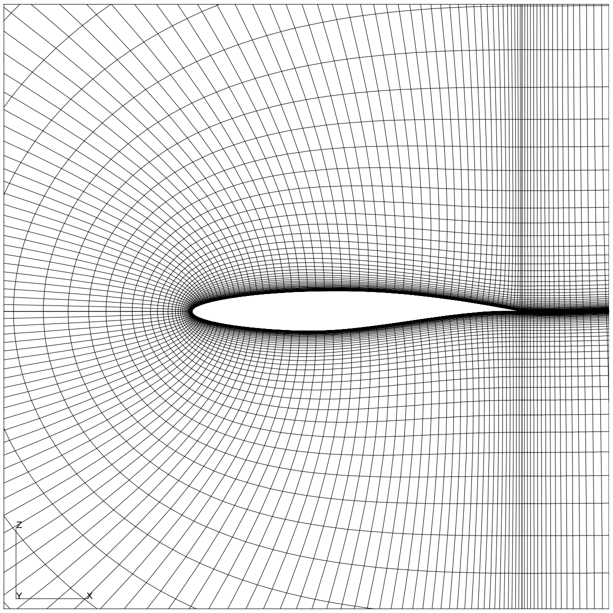


Figure 12: Grid with boundary orthogonality.

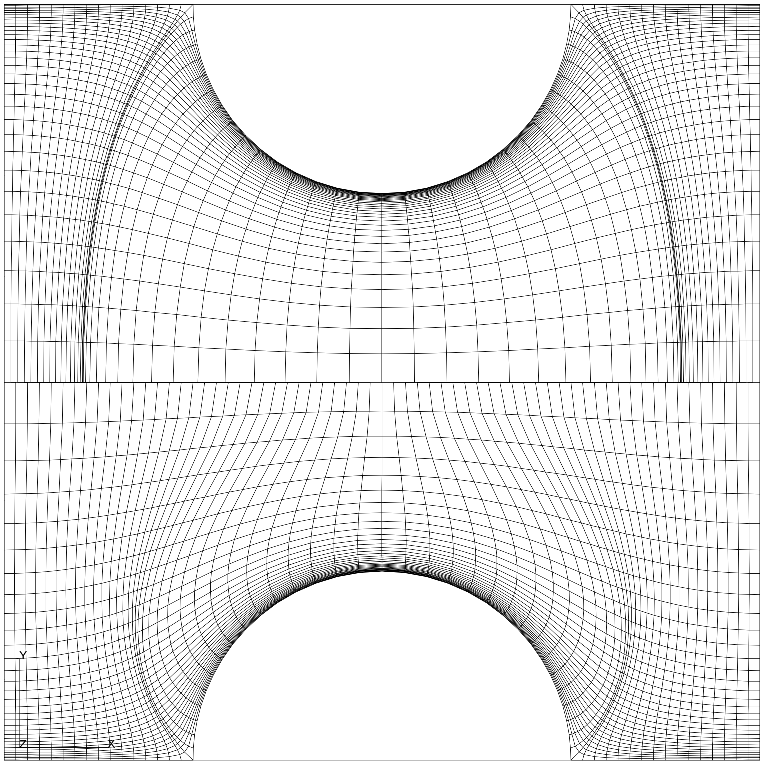


Figure 13: Orthogonal grid generation by boundary grid point movement along an edge. The grid in the lower part is only orthogonal at the boundary. The grid in the upper part is also orthogonal in the interior.

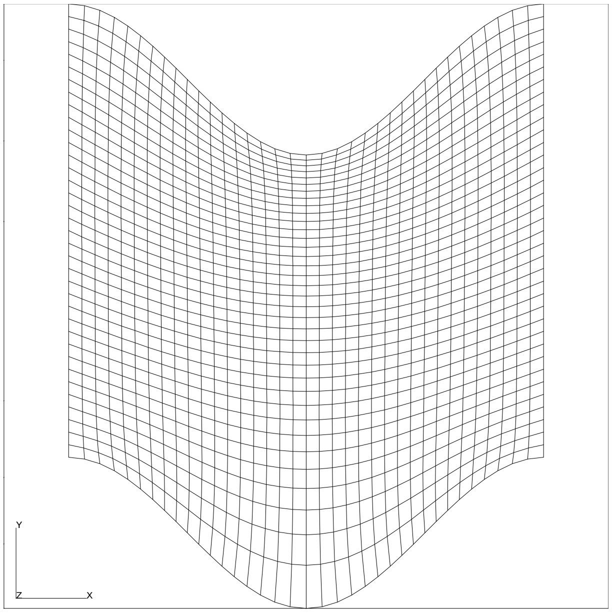


Figure 14: Laplace grid.

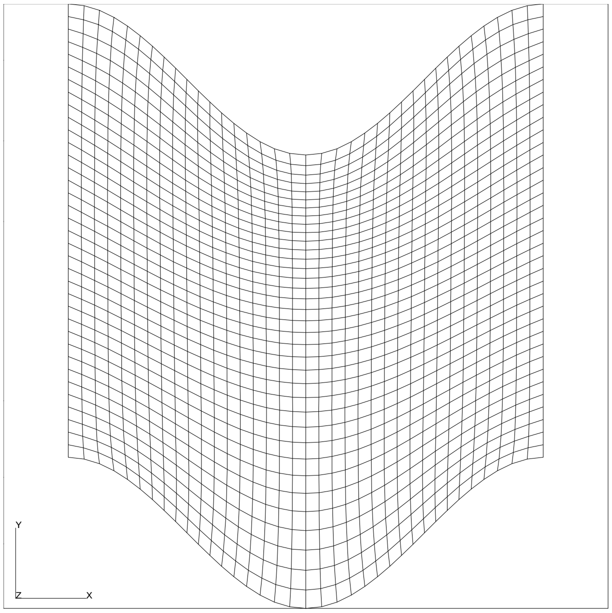


Figure 15: Grid with cell height control at upper and lower side.

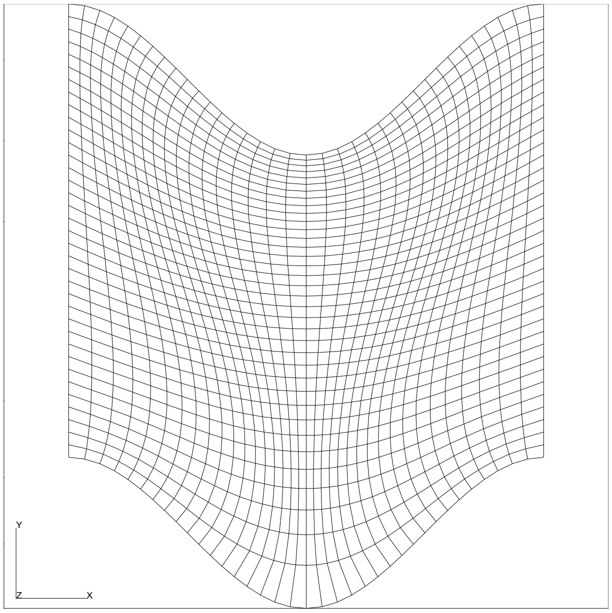


Figure 16: Grid with boundary orthogonality at upper and lower side.

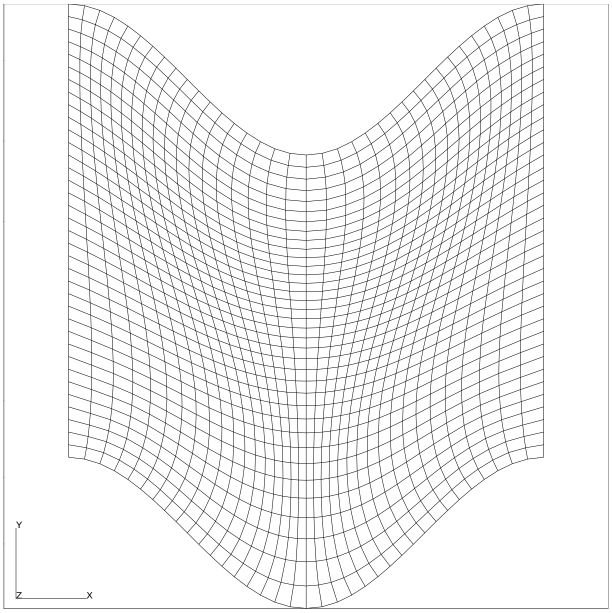
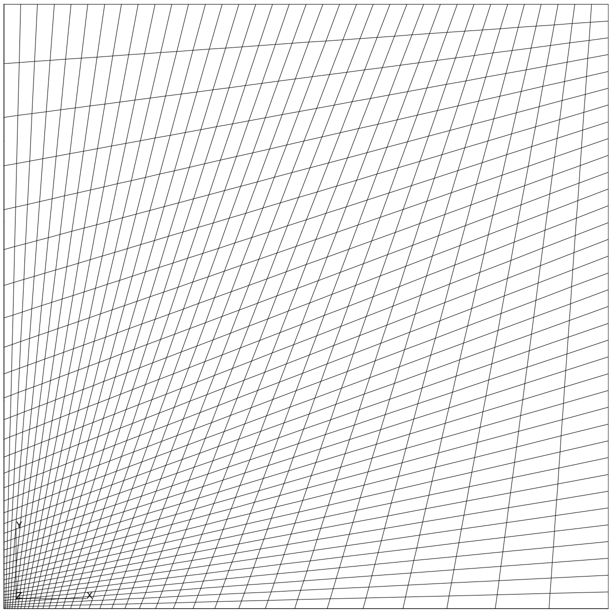


Figure 17: Grid with both cell height control and boundary orthogonality at upper and lower side.



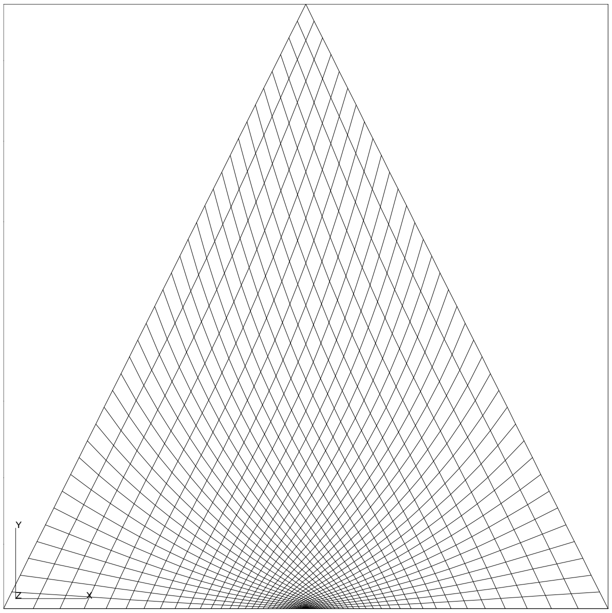
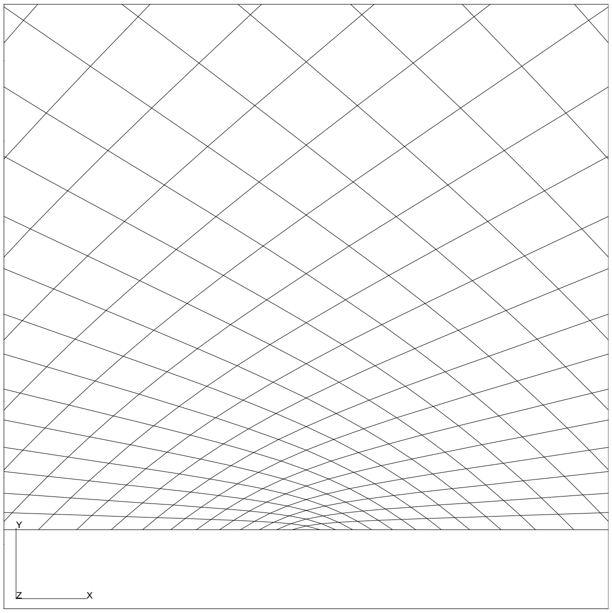


Figure 18: Initial grid in parameter space based on normalized arc length.

Figure 19: Corresponding grid in physical space.



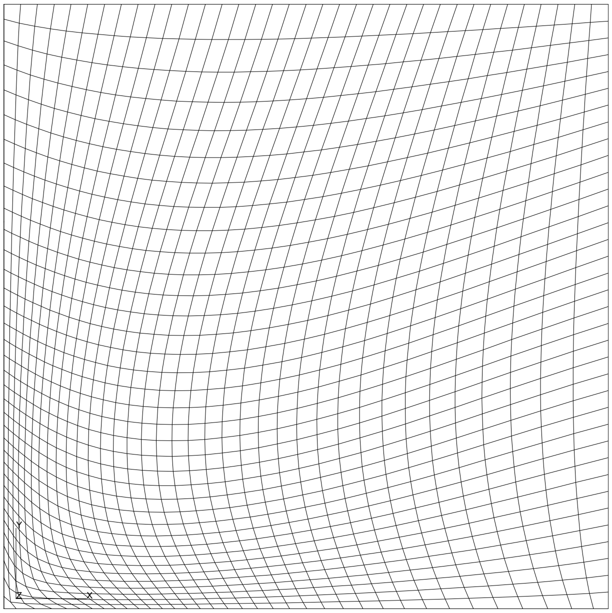


Figure 20: Blow up near O-type singularity.

Figure 21: Grid in parameter space obtained by solving Laplace equations with Neumann boundary conditions at the two bottom edges of the triangle.

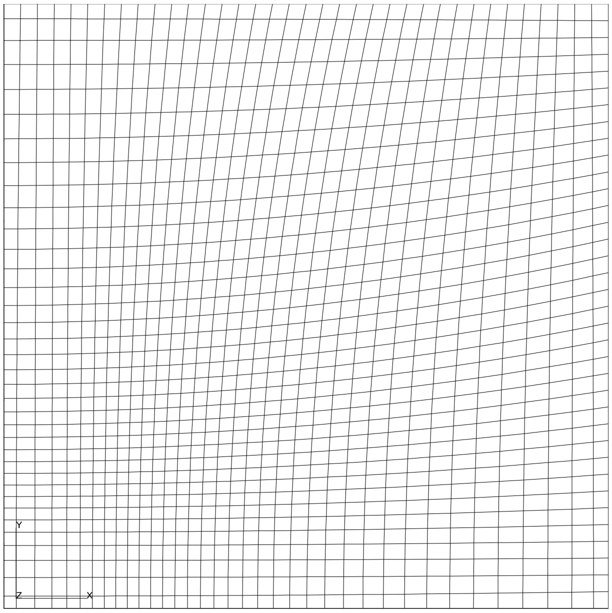


Figure 22: New grid in parameter space for boundary orthogonality. Position of boundary grid points are the same as in Fig.21.

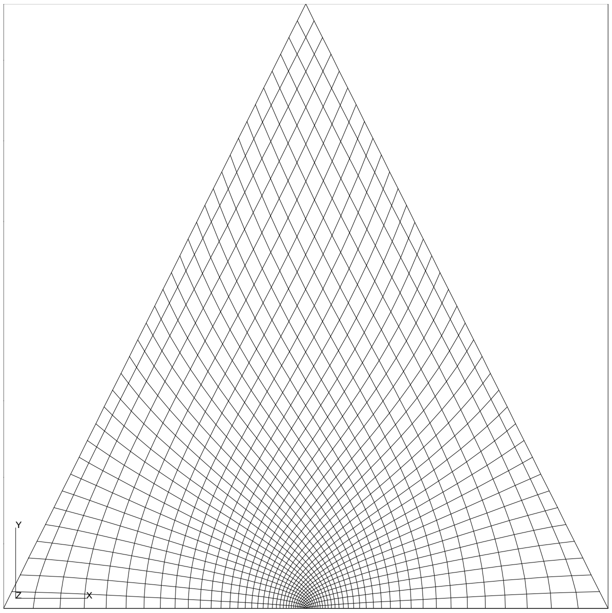


Figure 23: Corresponding grid in physical space.

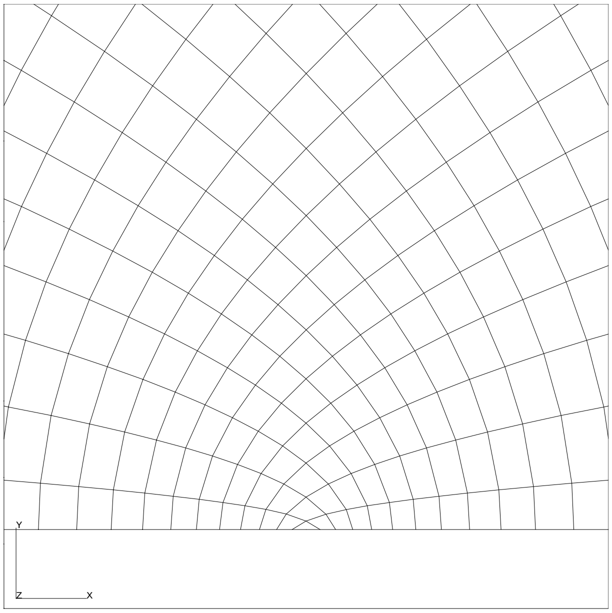


Figure 24: Blow up near O-type singularity.

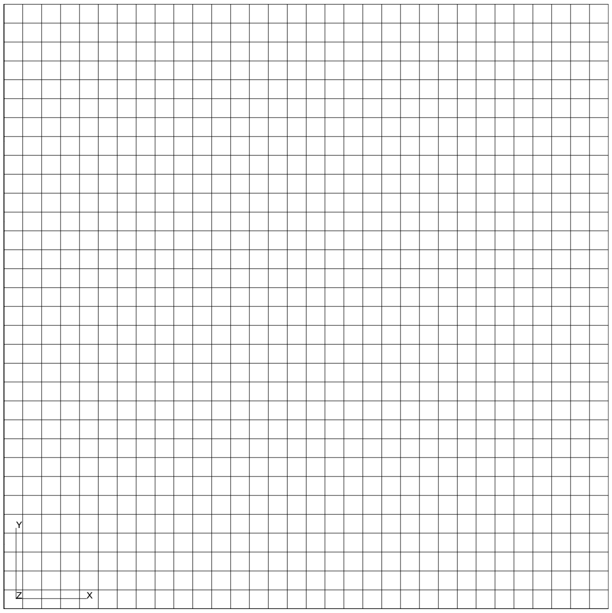


Figure 25: Initial uniform grid in parameter space based on normalized arc length.

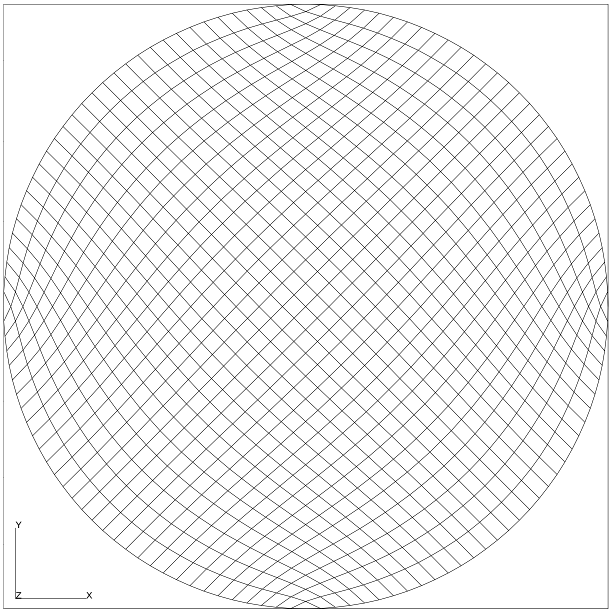


Figure 26: Corresponding Laplace grid in physical space.

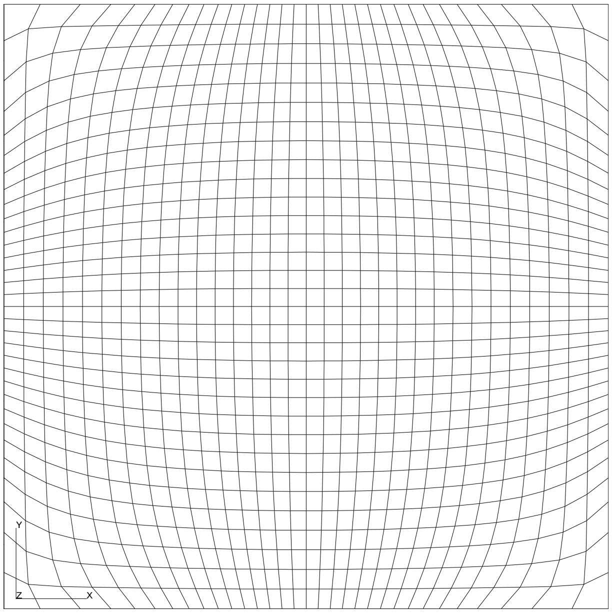


Figure 27: Grid in parameter space obtained by solving the Laplace equations with Neumann boundary conditions at all four sides.

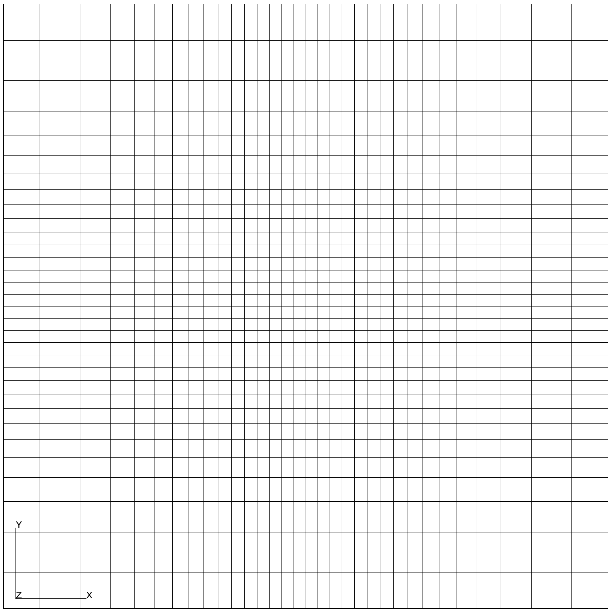


Figure 28: New grid in parameter space for boundary orthogonality at all four sides. Position of boundary points are the same as in Fig.27.

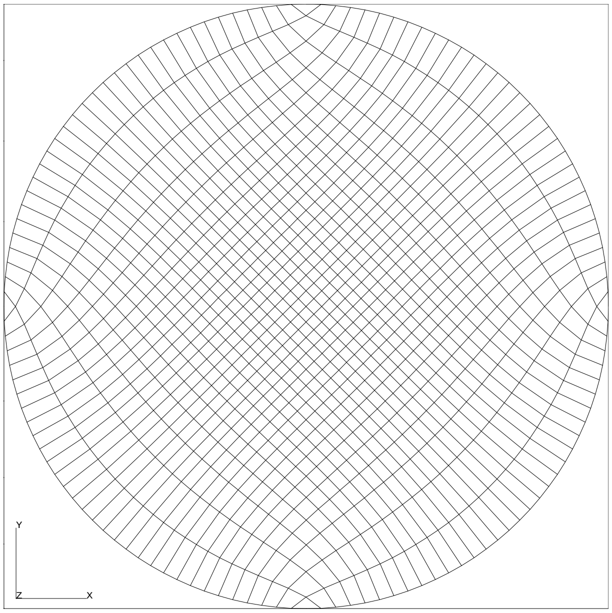


Figure 29: Corresponding grid in physical space. Interior grid is also orthogonal.

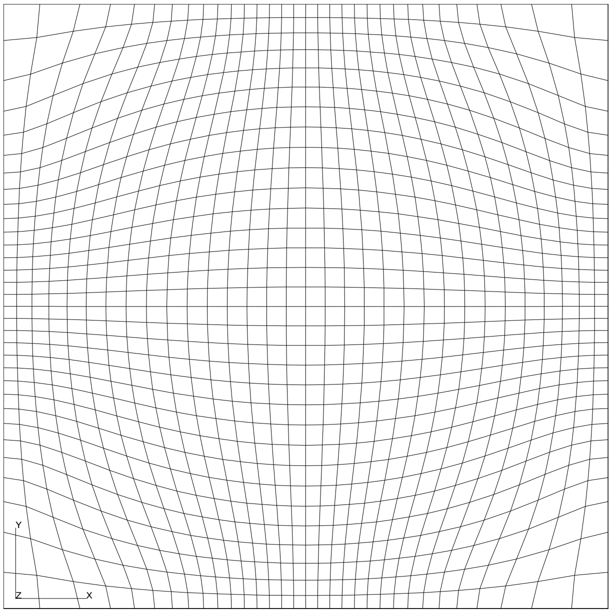


Figure 30: Adapted grid in parameter space for complete boundary control.

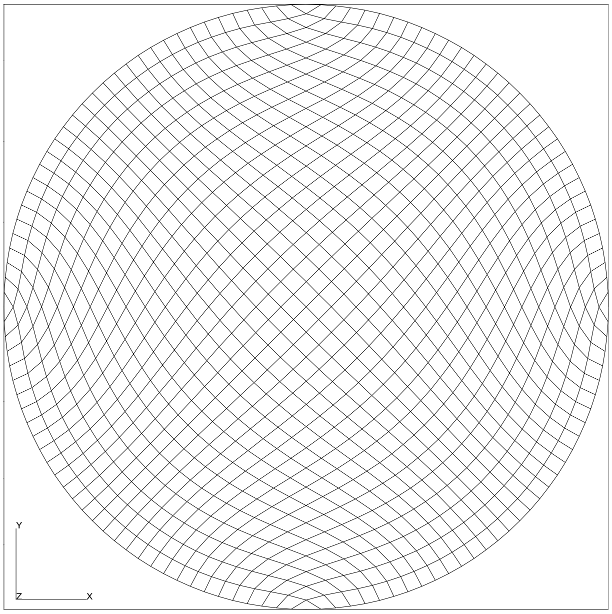


Figure 31: Corresponding grid in physical space.

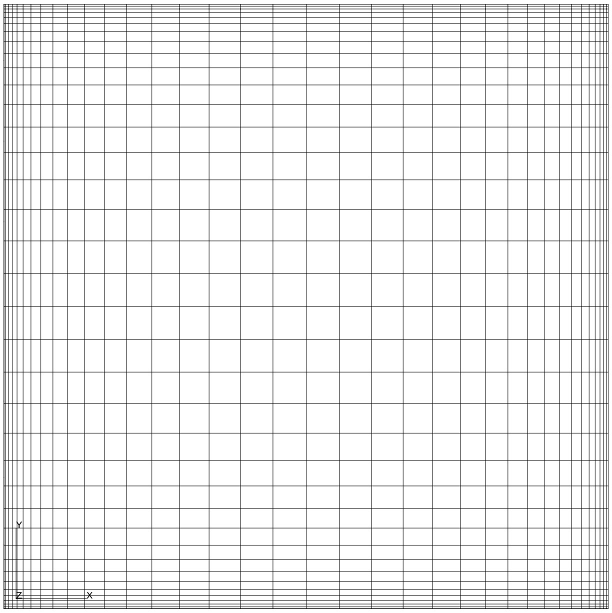


Figure 32: Initial grid in parameter space based on normalized arc length.

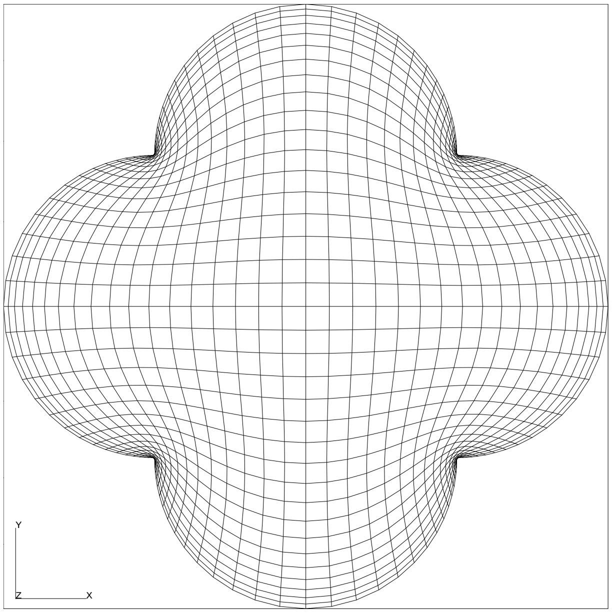


Figure 33: Corresponding grid in physical space.

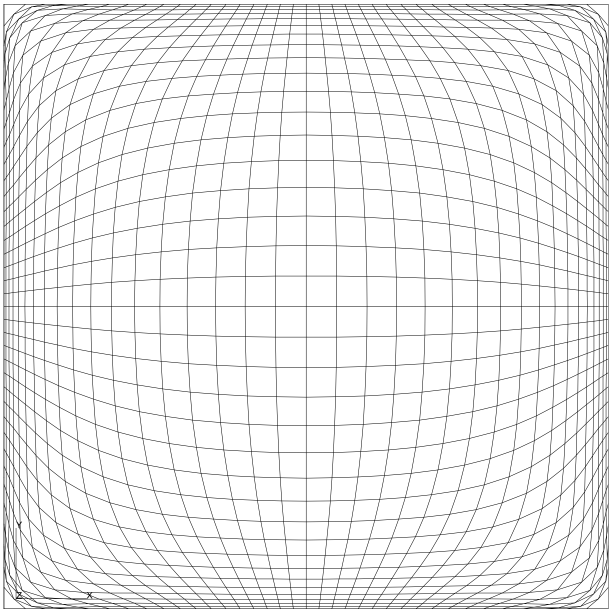


Figure 34: Grid in parameter space obtained by solving the Laplace equations with Neumann boundary conditions at all four sides.

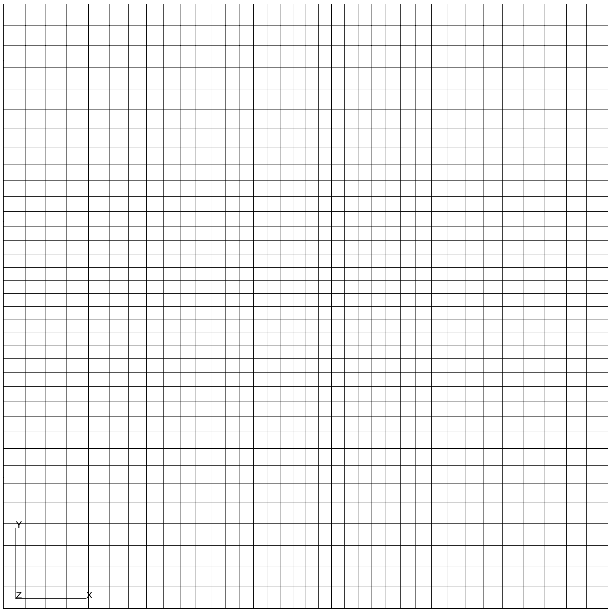


Figure 35: New grid in parameter space for boundary orthogonality. Position of boundary grid points are the same as in Fig.34.

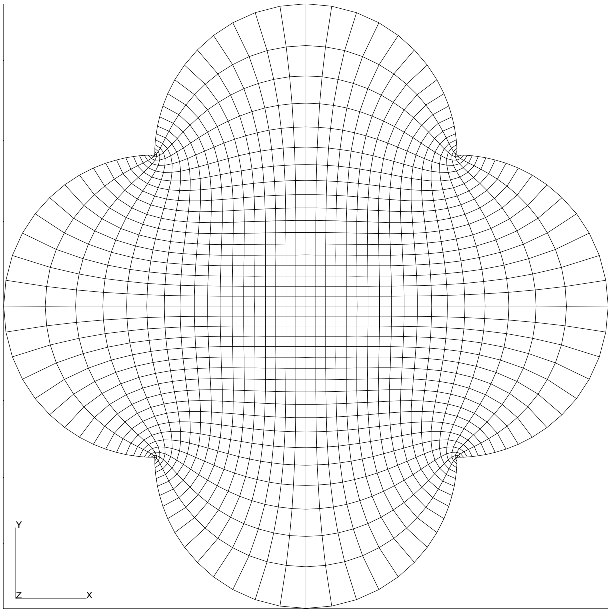


Figure 36: Corresponding grid in physical space. Interior grid is also orthogonal.

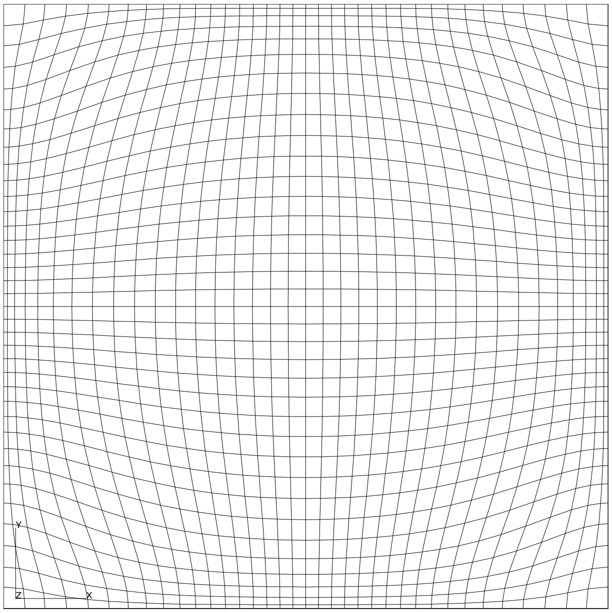


Figure 37: Adapted grid in parameter space for complete boundary control.

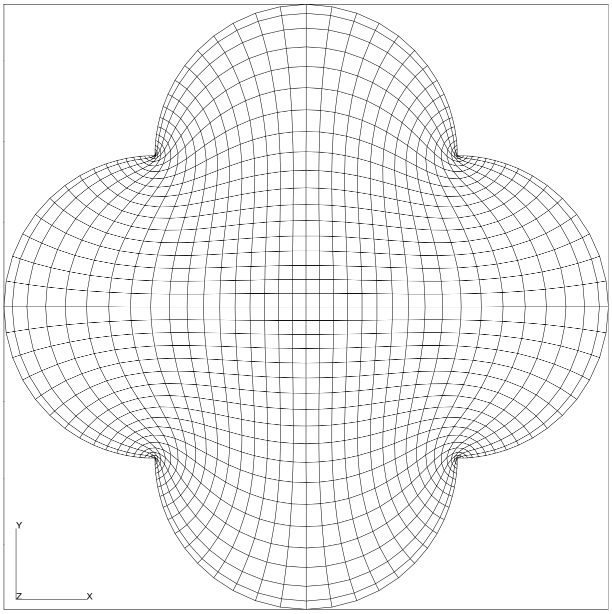


Figure 38: Corresponding grid in physical space.

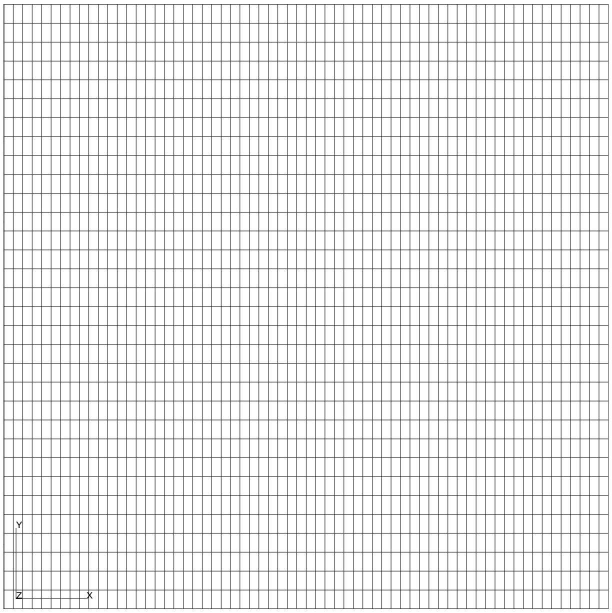


Figure 39: Initial uniform grid in parameter space based on normalized arc length.

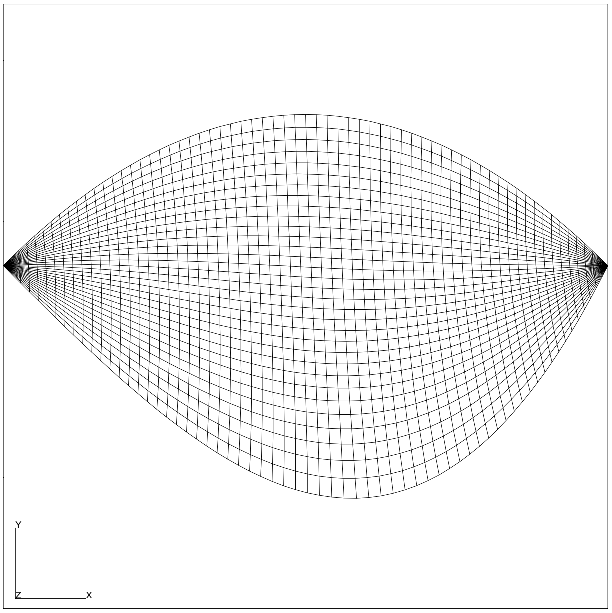


Figure 40: Corresponding Laplace grid in physical space.

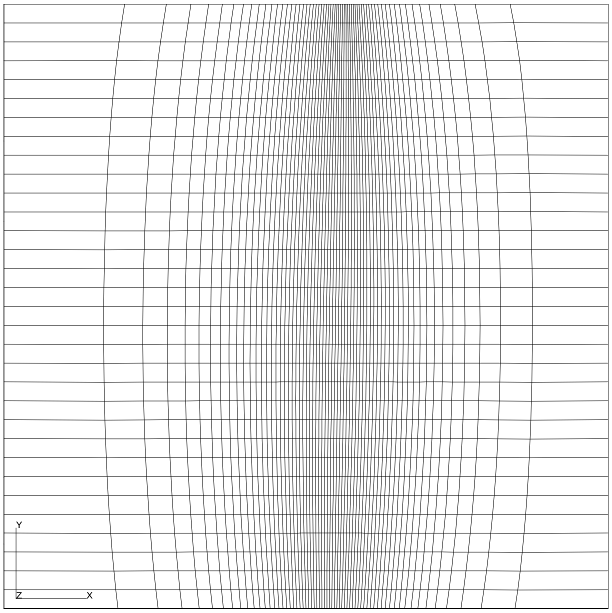


Figure 41: Grid in parameter space obtained by solving Laplace equations with Neumann boundary conditions at the two non-degenerated edges.

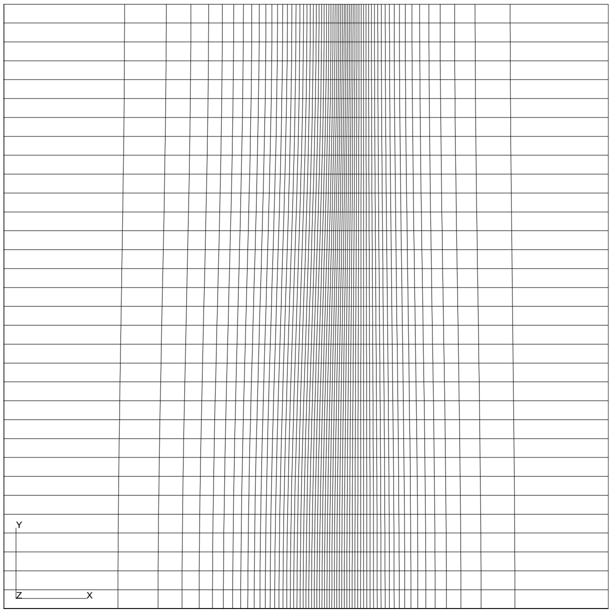


Figure 42: New grid in parameter space for boundary orthogonality at the two non-degenerated edges. Position of boundary grid points are the same as in Fig.41.

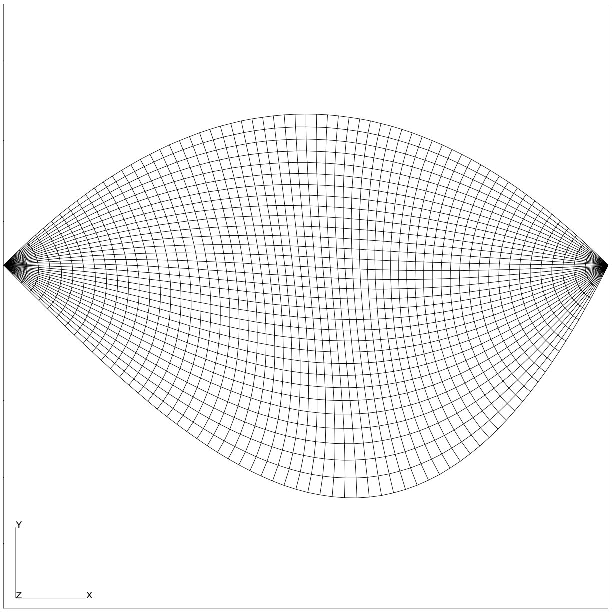
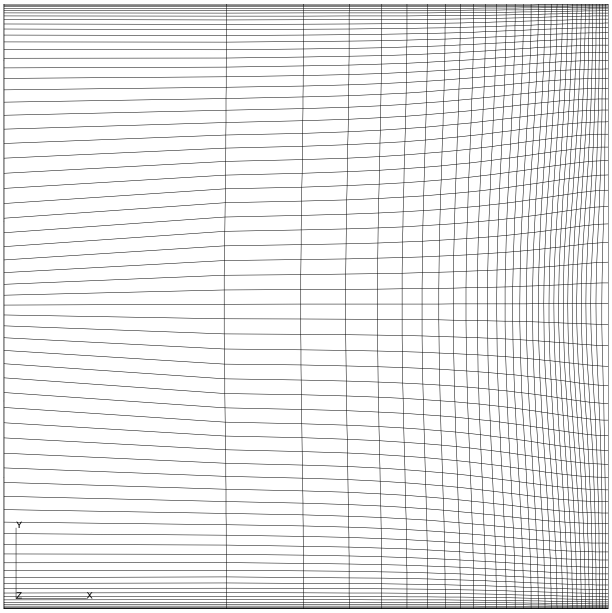


Figure 43: Corresponding grid in physical space.



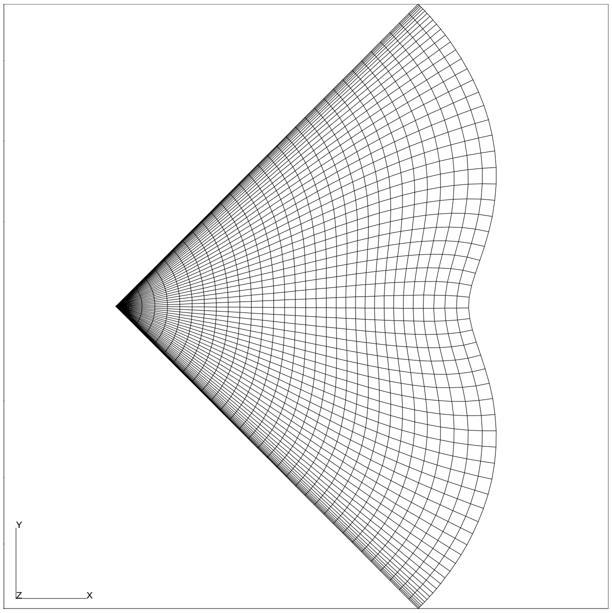


Figure 44: Constructed grid in parameter space for both grid orthogonality and mesh spacing control at the boundary of a trilateral.

Figure 45: Corresponding grid in trilateral.

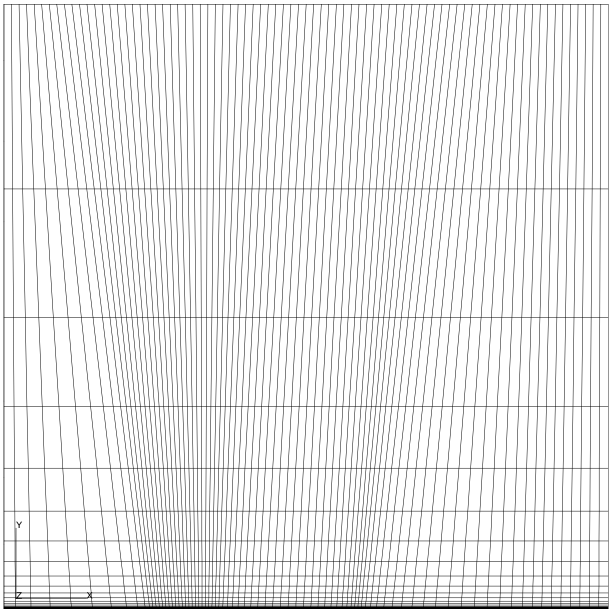


Figure 46: Initial grid in parameter space map based on normalized arc length.

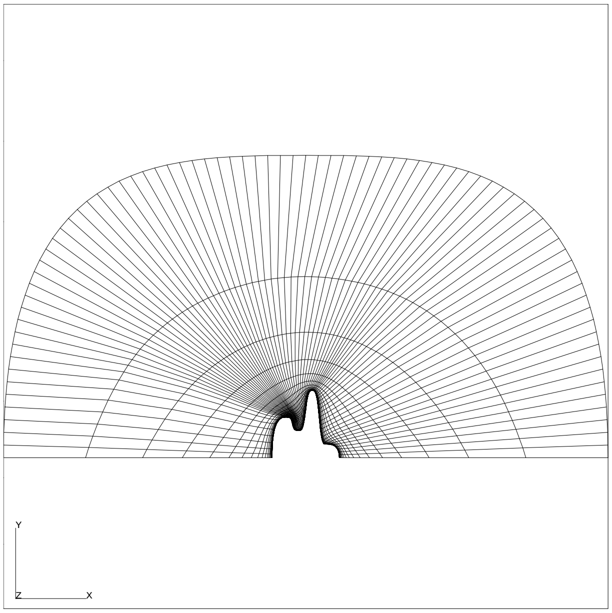


Figure 47: Corresponding grid in physical space.

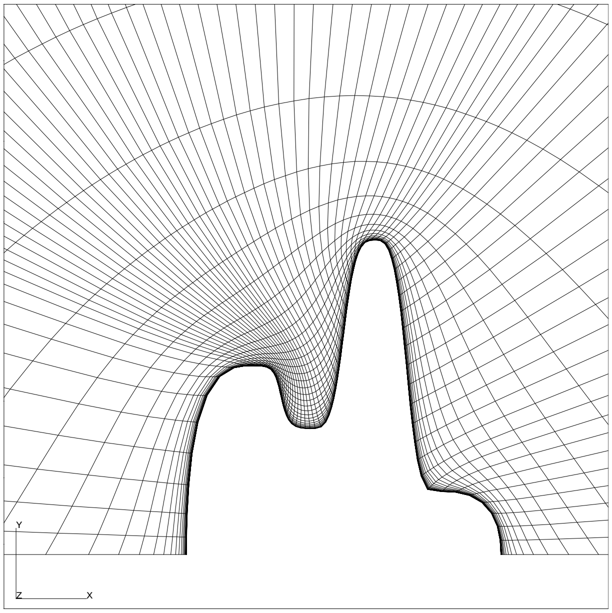


Figure 48: Blow up.

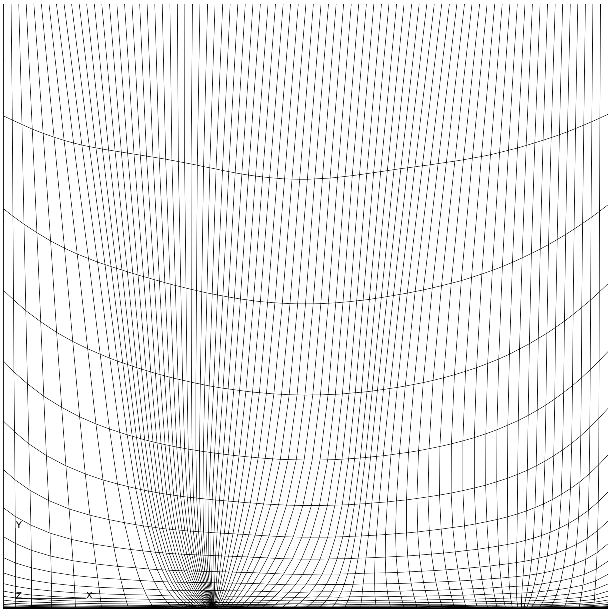
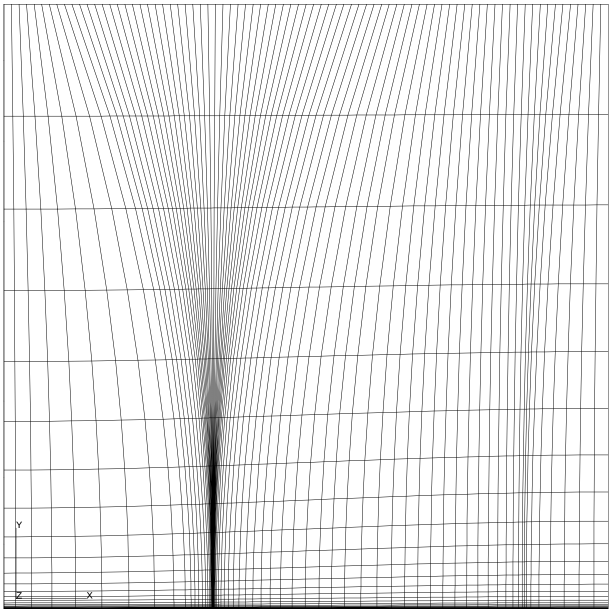


Figure 49: Solution of Laplace equations with Neumann boundary conditions at the three bottom edges of the domain.



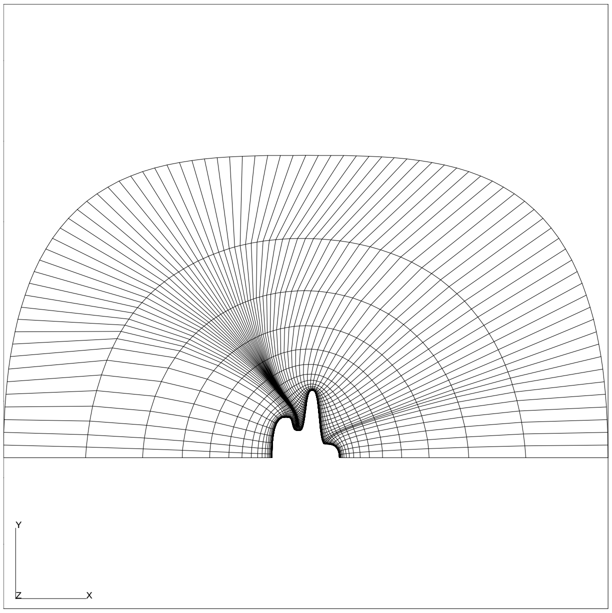


Figure 50: New grid in parameter space for boundary orthogonality at the three bottom edges of the domain. Position of the boundary grid points is the same as in Fig.49.

Figure 51: Corresponding grid in physical space.

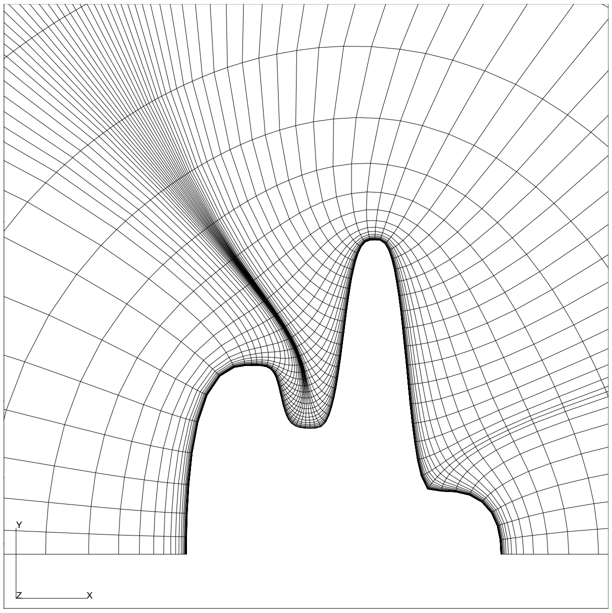


Figure 52: Blow up.

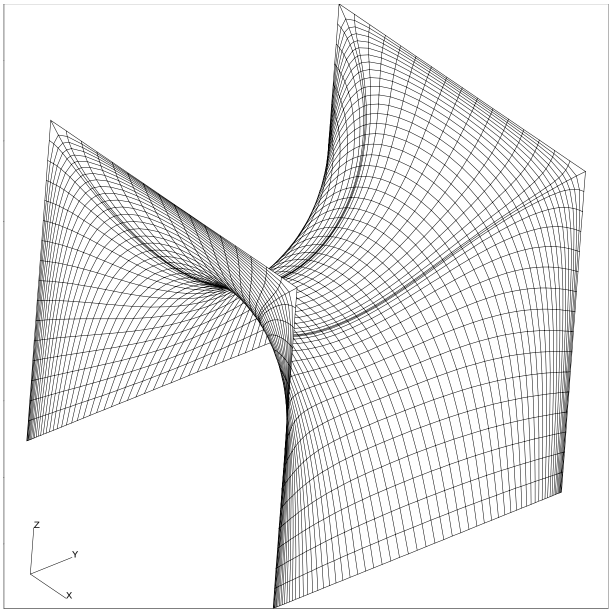


Figure 53: Minimal surface grid (Scherk surface). Surface grid is orthogonal.

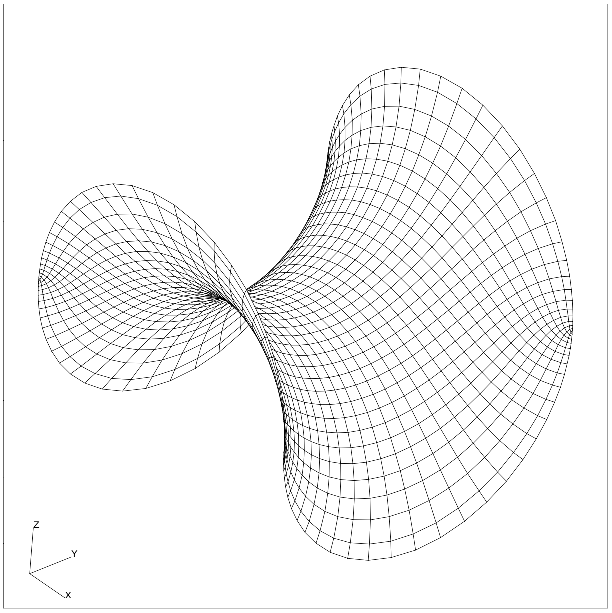


Figure 54: Minimal surface grid bounded by four orthogonal circular arcs. Surface grid is orthogonal.

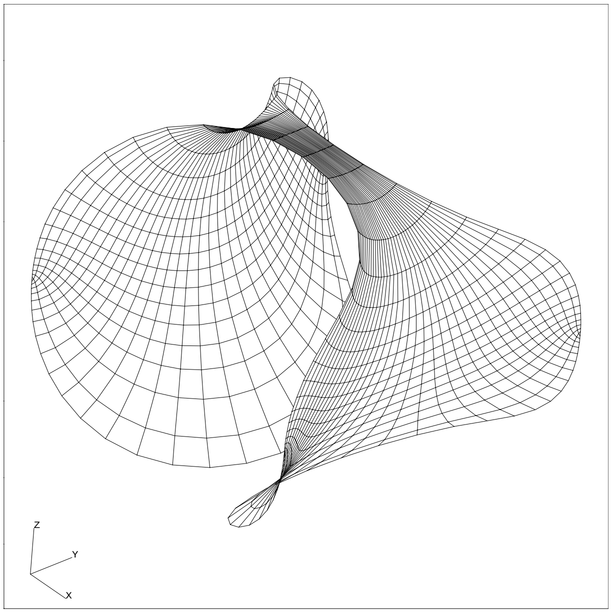


Figure 55: Change in shape by bending opposite circular arcs together.

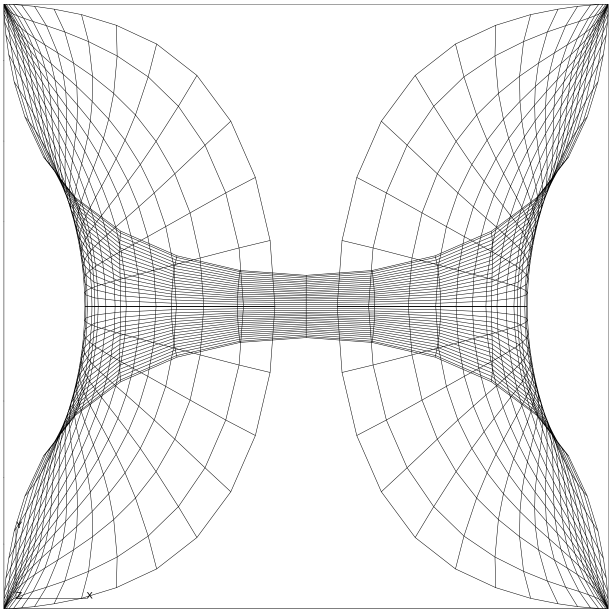


Figure 56: Projection on $xy-$ plane.

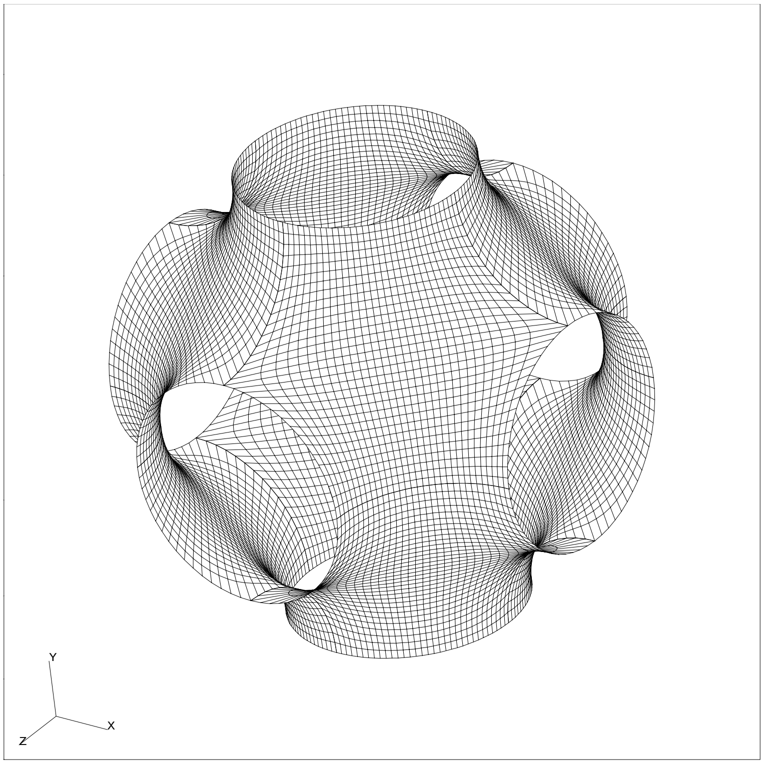


Figure 57: Schwarz’s P-minimalsurface.

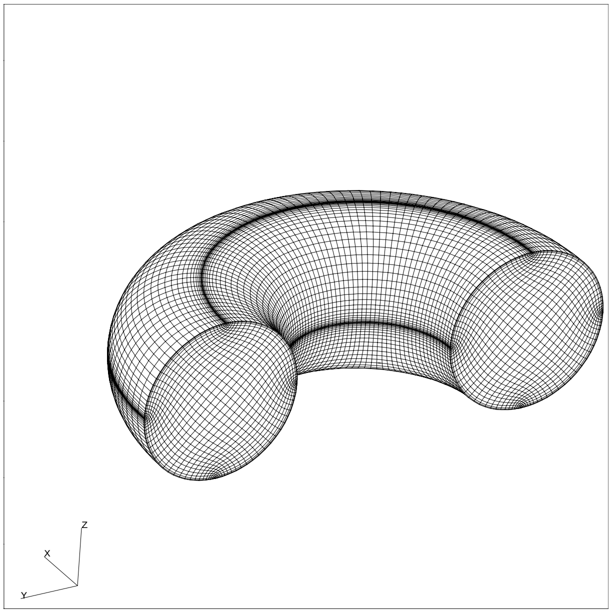


Figure 58: Boundary surface grid of a semi-torus

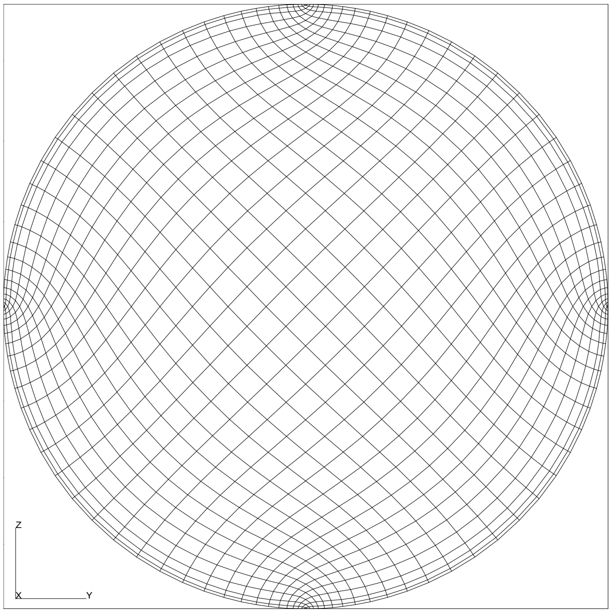


Figure 59: Surface grid on the two exterior circular planes.

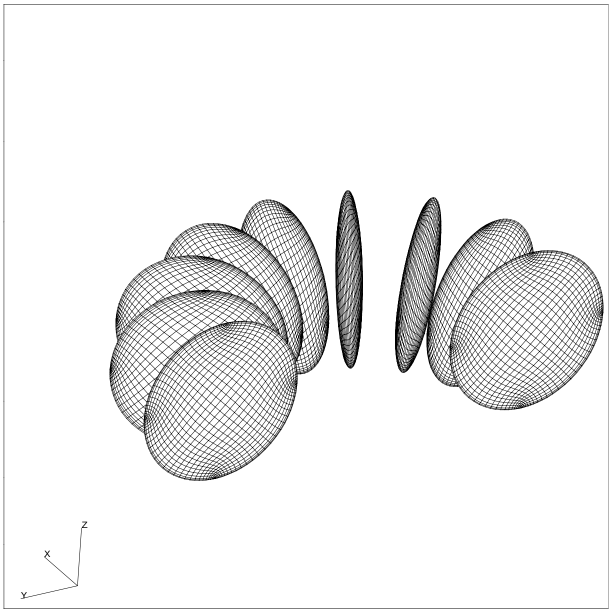


Figure 60: Interior grid planes inside the torus.

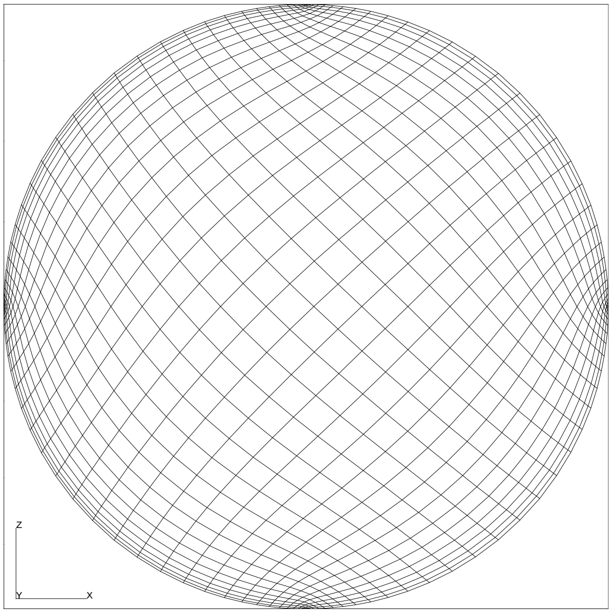


Figure 61: Interior grid inside the torus on a circular plane halfway between the two exterior circular planes.