POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Mesh Morphing for CFD Applications

Tesi di Laurea Magistrale in
Energy Engineering - Ingegneria Energia

Author: **Vishal Garg**

Student ID: 966001
Advisor: Prof. Fabio Inzoli
Co-advisors: Prof. Federico Piscaglia, Dr. Konstantinos Gkaragkounis (Optimad engineering Srl)
Academic Year: 2022-23

Abstract

In present work, a linear elasticity analogy based mesh morpher is developed for various CFD applications like Fluid Structure Interaction (FSI), aerodynamic shape optimization etc. The mesh is assumed to have the behaviour of an elasticity body which follows the law of linear elasticity. Due to the applicability of the law of linear elasticity on small deformations only, a multi-step approach is applied to handle large deformations. A variation of two different pre-existing relations for the computation of the elasticity coefficient is proposed and analyzed. A parametric study is conducted over the parameters introduced in the proposed relations to compute the elasticity coefficient or stiffness. The results obtained using the linear elasticity solver are then compared with available promising mesh morphing techniques like the Laplacian and the Laplacian quaternions techniques. The developed linear elasticity solver is tested on different kinds of meshes from simple to complex with different kinds of deformations which the related application might undergo. In the end, it is found in the results of the case studies that a good quality mesh well beyond the practically needed deformations can be obtained using the presently developed linear elasticity based mesh morpher.

Keywords: Linear elasticity analogy, Mesh morphing, Fluid Structure Interaction, Aerodynamic shape optimization, Elasticity coefficient, Laplacian quaternions

Contents

Abstract **i**

Contents **iii**

1 Introduction **1**

1.1 Literature survey . . . 2
1.1.1 Physical analogy based techniques . . . 2
1.1.2 Interpolation based techniques . . . 4
1.2 Objectives and scope . . . 8

2 Governing equations and Mathematical model **11**

2.1 Elasticity analogy and governing equations . . . 11
2.2 Explicit FVM mathematical model . . . 13
2.3 Boundary condition . . . 14
2.4 Design of the overall solver . . . 15
2.5 Computation of cell and face gradients . . . 16
2.6 Boundary corrections in cell and face gradients . . . 19
2.6.1 Correction in cell gradient stencils . . . 20
2.6.2 Correction in face gradient stencils . . . 21
2.7 Computation of the Laplacian term . . . 22
2.8 Convergence condition and control method . . . 23
2.9 Interpolation from nodes to face centers and cell centers to nodes . . . 24
2.9.1 Nodes to face center interpolation . . . 24
2.9.2 Cell center to nodes interpolation . . . 25
2.10 Multi-step algorithm . . . 26

3 Results: A cube of 64k cells **29**

3.1 Gradient computation at cell and face centers . . . 29
3.1.1 Cell gradients . . . 30
3.1.2 Face gradients . . . 33

3.2 Mesh morphing results: outward and inward deformations 36
3.3 Multi-step results and comparisons with single step results. 39
3.3.1 Outward deformation 39
3.3.2 Inward deformation 41
3.4 Conclusions 43

4 Parametric study **45**
4.1 Elasticity coefficient 45
4.1.1 Outward mesh deformation 46
4.1.2 Inward mesh deformation 51
4.2 Poisson's ratio 55
4.2.1 Outward mesh deformation 55
4.2.2 Inward mesh deformation 57
4.3 Conclusions 59

5 Comparisons with Laplacian and Laplacian quaternions techniques **61**
5.1 Comparisons for a single-step approach 61
5.1.1 Outward mesh deformation 62
5.1.2 Inward mesh deformation 64
5.2 Comparisons for a multi-step approach 65
5.2.1 Outward mesh deformation 66
5.2.2 Inward mesh deformation 68
5.3 Conclusions 70

6 Industrial cases **71**
6.1 Wing of an airplane 71
6.1.1 Bending 72
6.1.2 Twisting 77
6.1.3 Bending + Twisting 80
6.1.4 Cambering 82
6.1.5 Comparison with Laplacian quaternions technique 85
6.2 Aerodynamic shape optimization in a Car 86
6.3 Conclusions 89

7 Conclusions and future developments **91**

Bibliography **95**

A Appendix A **101**

List of Figures **103**

List of Tables **105**

List of Symbols **107**

List of Acronyms **109**

Acknowledgements **111**

1 | Introduction

The role of quality of dynamically updated 3D meshes is quite important in the design and analysis of many engineering applications where the boundary between the fluid and the solid can have motion. Some of the examples of these applications are as follows:

- Aerodynamic shape optimization: In various aerodynamic applications, the shape optimization takes many design iterations which makes mesh morphing a faster alternative [1].
- Moving boundary problem: When the boundary between the solid and the fluid has moving components [2].
- Fluid-structure interaction (FSI): When the fluid mesh boundary undergoes a deformation [3, 4]. It is mostly critical when the degree of physical coupling between the fluid and the structure is considerable. Ex.(in increasing order of physical coupling)- rigid bodies and blade deformations, vortex-induced vibration and aero-elastic flutter, biomedical membranes and highly deformable solids.
- Free surface flows

The design process of these dynamic CFD applications require new meshes in every design iteration. There exist mainly two kinds of methods to generate the new meshes for each design iteration. The first method is to regenerate the mesh after each iteration. Since the generation of mesh is a tedious task, it is very time expensive to recreate a good quality mesh after every design iteration, and therefore it is not the best method to carry on design iterations. Another shortcoming of re-meshing is in applications like aerodynamic optimization, where after each design step, the targeted field suffers from discretization errors [5]. The second method to create the new mesh for the new design iteration is to morph the existing mesh. This method is hardly as time consuming but the quality depends on the technique used to morph the existing mesh. Another benefit of mesh morphing is the ease to achieve high-order accuracy in time because of the unchanged mesh topology with respect to the time steps [6].

1.1. Literature survey

In literature, numerous research has been carried out to morph the volumetric meshes [6–13]. There exist mainly two kinds of techniques to morph a target mesh namely; 1. Physical analogy based techniques and, 2. Interpolation based techniques. Among the physical analogy based techniques, there are spring based, elasticity based and Laplacian based techniques, whereas among the interpolation based techniques, there exist the Transfinite interpolation (TFI), Inverse distance weighting (IDW), Radial basis functions (RBF), Delaunay graph, Quaternions based etc. techniques.

1.1.1. Physical analogy based techniques

As the title suggests, physical analogy based techniques are based on some physical principles which are generally governed by differential equations. The mesh morphing using these techniques require solution of the system of these differential equations which are some times partial differential equations. In general, physical analogy based techniques need grid connectivity information which makes them memory intensive. Following is a brief about the most commonly used physical analogy based techniques

- **Spring based analogy:** In the spring based analogy, an edge of the mesh is considered analogous to a tension spring. One of the oldest and well known work incorporating this technique is by Batina [7] where the spring constant is considered proportional to the inverse of the length of the edge. In this technique, the equilibrium length of the spring is assumed to be equal to the initial length of the edge in the mesh. The governing equation for this technique is derived by equating the total force at a node by 0 under the condition of static equilibrium. The following equation is obtained for the displacement vector $\vec{\delta}$ which is to be solver iteratively:

$$\vec{\delta}_i^{k+1} = \frac{\sum_{j=1}^{n_i} \alpha_{ij} \vec{\delta}_j^{k}}{\sum_{j=1}^{n_i} \alpha_{ij}} \tag{1.1}$$

  Here, α_{ij} is the stiffness of the edge between nodes i and j, and n_i is the number of neighbours of node i. In a work conducted by Blom [14], it was found for a 1D mesh undergoing a deformation that no collisions in the nodes take place. This although doesn't stand for a triangular mesh undergone a rotary deformation where the collision of nodes and therefore invalid mesh was observed. In order to solve the problems with the element inversion, various modified spring analogies were also published in the literature. Farhat [15] proposed a torsional spring analogy where

a torsional spring was placed between every set of adjacent edges in the triangular mesh. Another similar analogy called semi-torsional spring was proposed by [14]. In this analogy, in order to increase the stiffness of the mesh for large deformations, the spring stiffness was divided by the angles of the rest of the two edges of the triangle. This although converted the system of equations into a non-linear one, which is computationally expensive. Two more well known tension spring based analogies are the ortho-torsional spring analogy and the ball-vertex spring analogy, which were initially explained by [16] and [17] respectively.

- **Linear elasticity based analogy:** In this analogy, the mesh is considered analogous to a structure which follows the law of linear elasticity. The displacement field in the mesh is obtained by solving the set of linear elasticity equation 1.2 in the bulk mesh.

$$\nabla.\sigma = f \quad on\ \Omega, \tag{1.2}$$

The equation of linear elasticity depends on the material properties like elasticity coefficient and Poisson's ratio. While applying the linear elasticity analogy, these properties are defined such that they are functions of the mesh characteristics. In literature, various relations are suggested for these properties as a function of the mesh characteristics. In two of the most popular relations, the Poisson's ratio is kept uniform and constant within a range of -1 to 0.5 and the elasticity coefficient is assumed to be proportional to the inverse of the volume of the cell and inverse of the distance from nearest boundary respectively [6, 18, 19]. These two kinds of relations help in avoiding invalid elements close to the boundary. An alternative approach suggested in the literature is to set the elasticity coefficient to a constant and keep the Poisson's ratio such that $\frac{1}{(1-2\nu)}$ = aspect ratio of the cell [20, 21]. A similar alternative approach suggested is to keep the Poisson's ratio ν = constant and E = aspect ratio of the cell [22]. An adjoint based optimization procedure was proposed and implemented by Yang and Mavripilis [23] where an optimal distribution of E was produced. For this optimization, an objective function proportional to the cell volume was minimized by setting different E in all the cells. This optimization resulted in valid mesh for high deformations but turned out to be computationally expensive. In another approach by Hsu [24], the final deformation was achieved in two steps. Among these steps, the first one takes into account a constant $E = 1$ whereas the second step takes into account a varying E. Most of the studies mentioned above have taken into account a finite element method (GMRES most commonly) to discretize and solve the equations of linear elasticity [18, 19, 22, 24, 25].

- **Laplacian technique:** The solution of the Laplacian equation ($\nabla.(\nabla u) = 0$) corresponds to maximum/minimum principle. This solution also means that the displacements in the bulk mesh are going to be bounded by the boundary deformation. This prevents the internal mesh from crossing the boundary and keeps it valid. A modified version of the Laplacian ($\nabla.(\gamma^q \nabla u) = 0$) equation was also introduced with the introduction of diffusion coefficient (γ) and it's exponent (q). The main disadvantage of the normal Laplacian solution is that all the 3 components are solved independently and therefore and not coupled. As far as the modified Laplacian is concerned, another challenge is for the optimal results in a practical case, where the exponent q is not same for all the cells.

1.1.2. Interpolation based techniques

The interpolation based techniques are computationally effective and require lesser memory in general because of no need of connectivity information. Although, because of based on interpolations, these techniques do have interpolation errors associated.

Following is a brief of the most commonly used interpolation based techniques:

- **TFI and Algebraic damping technique:** TFI [8] and algebraic damping technique [9] are two of the most simple interpolation based techniques. In TFI, a generic node in the bulk mesh is deformed equal to the boundary times a scalar factor which depends on the distance between the boundary and the node. In algebraic damping technique, every node in the bulk mesh is assigned a boundary node and the displacement is computed using a distance function and a displacement function as shown in Eq. 1.3.

$$\vec{D}(x_i) = f(x_{bi}).\vec{D}(x_{bi}), \tag{1.3}$$

  The TFI technique is not suitable for the unstructured mesh because of high skewness and no prevention from element crossings. On the other hand, the algebraic damping technique can provide robust results even for high deformations with the use of a smoothing procedure [9].

- **IDW technique:** As the name suggests, for every bulk mesh node, a weight is assigned to all the boundary nodes inversely proportional to it's distance from the bulk mesh node to an exponent. As shown in Eq. 1.4, an average over these weights is computed to obtain the displacement at the bulk mesh node.

$$\vec{D}(x_i) = \frac{\sum_{k=1}^{nb} x_k w(r_k)}{\sum_{k=1}^{nb} w(r_k)} \tag{1.4}$$

In this technique, to improve the mesh orthogonality at the boundaries, the boundary rotation is treated in a separate manner [10]. This method is computationally very efficient because of no need to solve any matrices.

- **RBF technique:** RBF is one of the well developed tools for mesh morphing and for interpolating data in general [11, 26, 27]. This method produces high quality mesh for high deformations. The orthogonality at the boundary can be reasonably preserved using RBF. Like any other interpolation method, RBF does not need any mesh-connectivity data which makes it memory efficient. The RBF method takes into account a linear system of equations which is constructed only for the boundary mesh points and not for the bulk mesh points. This shows the need to solve the equations only at the boundary nodes making the RBF method computationally efficient as well. In RBF technique, an interpolation function S can be approximated by the sum of the product of displacement and basis function over the boundary nodes.

$$S(\vec{X}) = \sum_{j=1}^{n_b} \alpha_j \phi(\vec{X} - \vec{X_{b_j}}) + P(\vec{X}) \tag{1.5}$$

Here, $\vec{X_{b_j}}$ is the displacement vector of the boundary nodes which is known, P is a polynomial, n_b is the number of boundary nodes, and ϕ is the selected basis function over the Euclidean distance $||x||$. The coefficients α_j and polynomial P are obtained using the interpolation conditions. The basic implementation of RBF for 3-D problems is costly though. To improve the RBF technique for such cases, Rendal and Allen [28] proposed a data reduction algorithm called the greedy algorithm which was implemented with the RBF interpolation. With the inclusion of the greedy algorithm and further improvements in the RBF technique ([1, 29–31]), a considerable improvement in the computational efficiency and mesh quality was obtained.

- **Delaunay graph:** This method needs the creation of a Delaunay graph of the initial mesh. To create a Delaunay graph, only boundary nodes are required. All the interior nodes are assigned to different Delaunay elements that these nodes belong to. After deforming the Delaunay graph using the boundary condition, the displacement in the interior mesh can be interpolated [12]. This method needs mainly 4 steps; namely, Generation of Delaunay graph, Allocation of mesh points on the Delaunay graph, Deforming the Delaunay graph, and Relocating the interior mesh nodes ([32, 33]). The main disadvantage of the Delaunay graph method is that for complex meshes and high deformations, intersections between the Delaunay elements may occur and a regeneration of the Delaunay graph followed by relocation

of interior nodes is needed in such cases which is computationally costly.

- **Quaternion based technique:** A quaternion is a scalar with a direction and is composed of a real and three imaginary numbers ($Q = q_0 + q_1 i + q_2 j + q_3 k$). A quaternion is useful when translational and rotational information is required. A quaternion based mesh morphing technique was proposed by Samareh [13] where a three step approach to morph the bulk mesh was introduced. These steps were following: 1. Translation of the deformed nodes, undeformed nodes and neighbor nodes of the origin. 2. Rotate the undeformed nodes to make their normals align with the normals of deformed nodes using a quaternion as follows:

$$Q_1 = [cos(\alpha/2), n_u \times n_d sin(\alpha/2)] \tag{1.6}$$

where α is the angle between the two normals n_u and n_d. 3. Rotate the undeformed nodes about the deformed boundary normal vectors to minimize the angle between neighbour nodes using another quaternion as follows:

$$Q_2 = [cos(\theta/2), n_d sin(\theta/2)] \tag{1.7}$$

where θ denotes the average angles between the neighbour nodes. After all this, total rotation and translation can be computed as $Q = Q_1.Q_2$ and $T = X_d - QX_uQ^*$ respectively. Here, X_d and X_u are the deformed and undeformed positions respectively. The translation and rotation now can be propagated using a spring analogy technique or a Laplacian based technique as done in [34].

There have been various other methods in literature for the bulk mesh deformation. Some noticeable methods are as follows: a free form deformation (FFD) technique [35–37], a tree-code optimization based interpolation technique for polyhedral meshes by Luke et al. [38], a hybrid mesh deformation algorithm using anisotropic PDEs and multiobjective mesh optimization by Kim et al. [39], and a mesh deformation energy minimization based technique for FSI problems by Oh [40].

To compare the various mesh morphing techniques, the comparison table presented by Selim and Koomullil [3] is extended and shown in table 1.1. The comparison is based on the main advantages, disadvantages and order of complexity of these techniques. It can be observed in the comparison that among the physical analogy based techniques, the linear elasticity based analogy is promising but the main challenges are the computational cost and optimization of the physical properties E and ν. As far as the interpolation based techniques are concerned, the RBF with greedy algorithm seems to be the well developed

	Main advantage	Main disadvantage	Complexity	Note
Linear spring	Simple and easy to implement	Element crossing and overlapping	$O({n_e}^3)$	n_e is total number of edges
Torsional spring	Robust, preserves mesh quality	Computationally expensive	$O({n_e}^3 + {n_v}^3)$	n_v is the total number of vertices
Linear elasticity	Computationally feasible	Optimizing E and ν to avoid invalid element is difficult	$O(n_e log n_e)$	Uses FEM discretization with GMRES solver
Laplacian	Computationally efficient	Works for single frequency deformations only	$O(n_e)$	Partial differential equation based
TFI	Simple and efficient	Element crossing and overlapping	$O(n_v)$	Applicable only on structured meshes
Algebraic damping	Simple and efficient	Highly rigid deformation near boundaries and highly elastic elsewhere	$O(n_v)$	Smoothing procedure can be used to improve mesh quality
IDW	Simple and efficient	Basic IDW does not incorporate rotational deformations	$O(n_v)$	Rotational deformations can be handled separately for additional cost
Delaunay graph	Robust and computationally feasible	Delaunay Graph needs to be regenerated frequently	$O(n_D + {n_v}^{1/3})$	n_D = Delaunay graph vertices
RBF	Robust, preserves mesh quality	Computationally expensive	$O({n_b}^3)$	n_b = No. of boundary nodes
RBF with greedy algorithm	Robust and computationally efficient	No major disadvantage	$O({n_s}^3)$	$n_s < 5\% n_b$
Laplacian Quaternion	Computationally feasible	No major disadvantage	$10 \times O(RBF)$	Laplacian propagation on Quaternions
Tree-code optimization	Computationally feasible	Might suffer from robustness for low error bounds	$O(n_e log n_e)$	Complexity can be reduced by further optimization

Table 1.1: Comparison of various mesh morphing techniques (extension of [3]).

and best performing technique.

The linear elasticity analogy based mesh morphing technique has been widely studied in literature. One of the initial well accepted study on mesh morphing using linear elasticity analogy was presented by Stein et al. [41] where the capability of the technique to handle large deformations specifically for fluid-structure interaction (FSI) problems was demonstrated. A finite element formulation was used to discretize and solve the governing equations with a jacobian option to keep the stiffness of the smaller cells close to the boundary more than the stiffness of the bigger cells far from the boundary. After the demonstration of the linear elasticity analogy for mesh morphing by Stein et al. [41], there have been various further additions in the basic technique. Persson and Peraire [42] used the linear elasticity model to morph the curved mesh. In a study by Dwight [25], it was investigated that the solution of the linear elasticity equations does not provides good results in case of rigid body rotation and proposed a modification in the constitutive linear elasticity law. It was illustrated that the modified constitutive law gives much better results for the rigid body rotation cases. In a study by Barral et al. [43], for a transient FSI problem, the linear elasticity model was coupled with a changing connectivity moving mesh method. This study observed valid meshes for high deformations as well. The linear elasticity model was also used to morph the mesh in an application of electro-magnetic solver [44]. The objective of the study was to do the parametric study and optimization of high frequency devices with FEM. In FSI and moving boundary and interfaces (MBI), elasticity mesh morpher is used with the mesh-jacobian based stiffness (MJBS) to keep the mesh cells close to the boundary (small cells) stiffer with respect to the cells far from the boundary which are bigger in general. In a transient problem between time t_n and t_{n+1}, a deformation is executed in cycles which is path dependent which in turn includes accumulation of the mesh distortion. In a recent study by Tonon et al. [45], a method called as back-cycle-based mesh movement (BCBMM) is proposed to eliminate this accumulation based mesh distortion. To do this, the mesh motion is computed from the configuration in the first cycle.

1.2. Objectives and scope

From the existing literature of linear elasticity technique, it is evident that finding an optimum relation for the elasticity coefficient E and Poisson's ratio ν is still a challenge and hence required to be worked upon. In the field of aerodynamic shape optimization, the contribution of the linear elasticity technique is not explored yet. Moreover, for the FSI applications, complex deformations for complex geometries are not studied well in detail.

In literature, the results of the linear elasticity technique have not been compared with the results obtained using any other promising technique at any level of detail. Therefore, there exist the scope of not only finding the new relations for the elasticity coefficient and Poisson's ratio in the linear elasticity technique but also exploration of it's use in different CFD applications and comparisons of the results with the other different techniques.

Driven by the promising nature of the linear elasticity technique, the related applications and possible improvements, the present thesis focuses on the development of the linear elasticity analogy based mesh morphing technique. The objective of the present study is to develop a robust and accurate mathematical model and find a global relation for the physical properties E and ν. A special emphasis is given to the accuracy of mathematical model at the boundaries. A second order accuracy is ensured throughout the mesh. A parametric study is conducted to see the effect of different parameters in the relations of E and ν on the quality of the mesh. The effect of different kinds of surface deformation on the optimum parameters is analysed too. In present work, the benefit of using a linear elasticity at the boundary is also demonstrated. Another objective of the present study is to apply the elasticity morpher to different kinds of industrial applications and check the quality of the deformed mesh under different kinds of extreme deformations that the industrial application might undergo. The results obtained by the linear elasticity morpher are also compared with the results obtained by the existing Laplacian and the Laplacian quaternions techniques which were available to the author. Throughout the study, efforts are made to increase the accuracy of the various computation terms.

The present thesis document is structured in the following order: 2. Governing equations and the mathematical model 3. Results and discussion on the test case of a cube. 4.0 Parametric study for the E and ν relations. 5. Comparisons with the Laplacian and the Laplacian quaternions techniques 6. Testing of the linear elasticity analogy based technique on industrial cases 7. Conclusions and Discussion.

2 | Governing equations and Mathematical model

This chapter is dedicated to how the linear elasticity analogy described in Chapter 1 for the bulk mesh deformation can be elaborated by means of governing equations and how a mathematical model is developed to incorporate the linear elasticity equations to deform the bulk mesh. The overall design of the mesh morpher and various schemes used in the solver are described in this chapter.

2.1. Elasticity analogy and governing equations

An elastic solid subjected to body forces and surface traction undergoes deformation. In case of small deformations it is possible to govern the behaviour of the solid using the law of linear elasticity. For a small displacement $U = (u, v, w)$ the equation of linear elasticity can be written as follows:

$$\nabla.\sigma = f \quad on\ \Omega, \tag{2.1}$$

Here, f is the body force, σ is the stress tensor, and Ω is the computational domain. The stress tensor σ can be described in terms of the strain tensor ϵ using the constitutive relation as follows:

$$\sigma = \lambda Tr(\epsilon)I + 2\mu\epsilon \tag{2.2}$$

where λ and μ are the Lamé constants and Tr is the trace. The Lamé constants are functions of the material properties Young's modulus E and Poisson's ratio ν as follows:

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}, \qquad \mu = \frac{E}{2(1+\nu)} \tag{2.3}$$

The Young's modulus E shows the stiffness of the solid. A large E indicates that the solid is rigid, whereas a low E indicates more elastic nature. Poisson's ratio ν indicates the measure of deformation in lateral direction when the solid is undergone a deformation in the axial direction. The value of ν for physical materials can lie between $(-1, 0.5)$.

As per the linear kinematic law

$$\epsilon = \frac{1}{2}(\nabla U + \nabla U^T), \tag{2.4}$$

describes the change in length and orientation of an element in the solid. For rigid body rotations, an alternative relation for the Lamé constants is suggested in a study by Dwight [25] by making the stress on the elements equal to 0 instead of the ones derived from the elasticity equations. It is reported that a better mesh quality is obtained for not only the rigid body rotation cases but all the other applications in general. These alternative Lamé constants are shown hereby:

$$\lambda = -E, \qquad \mu = E, \tag{2.5}$$

In present work, these alternative Lamé constants are inspected along the ones obtained from the elasticity equations. As far as boundary conditions are concerned, one of the possibility over the boundary surface can be the Dirichlet boundary condition given by $U = U_b$ in $\partial\Omega$.

In literature there have been various ways of computing the elasticity Coefficient E and the Poisson's ratio ν as discussed in Sec. 1.1.1. Among the relations proposed, the relation of E inversely proportional to the volume of the cell is widely accepted and used [3], with ν kept as a constant value between $(-1.0, 0.5)$. Given the agreement in literature, in present work, a variant of this relation is proposed and studied. This new relation is shown as follows:

$$E_{Cell} = \frac{1}{(V_{Cell})^p}, \tag{2.6}$$

Where, E_{Cell} and V_{Cell} are the elasticity coefficient and volume of the concerned cell, and p is an exponent to the inverse of volume of the cell. Exponent p helps in providing the cells close to the boundary, which have smaller volume with respect to the cells far from the boundary, with considerably high stiffness which in turn helps these cells in preserving their shape. Another widely used relation for E keeping the ν constant is inverse of distance from the boundary. In present work, a variant of this relation is proposed and studied. This new relation is as follows:

$$E_{Cell} = \frac{1}{(d_{Cell})^q}, \tag{2.7}$$

Where, d_{Cell} is the distance of the concerned cell centroid from the nearest boundary point , and q is an exponent to the inverse of this distance. Just like p, exponent q helps

in providing the cells close to the boundary, which have smaller d_{Cell} with respect to the cells far from the boundary, with considerably high stiffness which in turn helps these cells in preserving their shape.

2.2. Explicit FVM mathematical model

In literature, the equation of linear elasticity Eq. 2.1 is discretized mostly using a finite element method and the related system of linear equation is solved using a GMRES method [18, 19, 22, 24, 25]. In present work, equation of linear elasticity is discretized using a finite volume method (FVM) and solved using a PETSc library. The main reasons for choosing the FVM are the simplicity of the method and existence of some useful developed modules. In present work, the FVM discretization is done based on field variables at the cell centers. Using Eqs. 2.2 and 2.4, Eq. 2.1 can be rewritten in the following Einstein form:

$$\frac{\partial}{\partial x_i}\left[\lambda\frac{\partial u_k}{\partial x_k}I_{ij}+\mu\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\right]=f_i \tag{2.8}$$

Equation 2.8 contains mainly three kinds of terms in the left hand side. These three terms can be described as following:

$$terma=\frac{\partial}{\partial x_i}\left(\lambda\frac{\partial u_k}{\partial x_k}I_{ij}\right);\quad termb1=\frac{\partial}{\partial x_i}\left(\mu\frac{\partial u_i}{\partial x_j}\right);\quad termb2=\frac{\partial}{\partial x_i}\left(\mu\frac{\partial u_j}{\partial x_i}\right) \tag{2.9}$$

These terms are then integrated with respect to volume and converted into the surface integrals using the Divergence theorem as follows:

$$Terma=\sum_f \lambda^f\frac{\partial u_k}{\partial x_k}\bigg|^f I_{ij}{n_i}^f\Delta S^f \tag{2.10}$$

$$Termb1=\sum_f \mu^f\left(\frac{\partial u_i}{\partial x_j}\right)^f {n_i}^f\Delta S^f \tag{2.11}$$

$$Termb2=\sum_f \mu^f\left(\frac{\partial u_j}{\partial x_i}\right)^f {n_i}^f\Delta S^f \tag{2.12}$$

Here, the summation shows the sum over all the faces of the considered finite volume or cell, ${n_i}^f$ indicates the i^{th} component of the normal vector to the face f, I_{ij} is the identity matrix, and ΔS^f is the area of the face f of the cell. In equation 2.10, $Terma$ contains the trace of the gradient which increases the order of coupling between the different components of Eq. 2.8. In first of the attempts, an explicit method is used to

solve Eq. 2.8 over the volumetric mesh. In the explicit method, $Termb2$ is kept in the LHS whereas the other terms $Terma$ and $Termb1$ are kept in the source term. It is to be noticed that the gradients at the faces are required to compute all the terms. It can be observed that $termb2$ in Eq. 2.9 is a Laplacian term which as shown in Eq. 2.12 depends only on the normal fluxes incoming from the all the faces of the considered cell.

2.3. Boundary condition

As mentioned in Sec. 2.2, the system of equation needs boundary conditions to be solved. This boundary condition in case of the equation of linear elasticity can be Dirichlet kind or the Neumann kind based on the boundary information provided. In case when the surface deformation over the whole boundary is known, a Dirichlet boundary condition shown in Eq. 2.13 can be used.

$$U = U_b, \ on \quad \partial\Omega \tag{2.13}$$

If the gradient of the field over a part of the boundary or the whole boundary is known, The Neumann boundary condition can be used as shown in Eq. 2.14

$$\frac{\partial U}{\partial x_i} = Constant, \ on \quad \partial\Omega \tag{2.14}$$

In symmetric CFD meshes, in general a symmetry plane is used to split the computational domain in half to reduce the complexity of the problem. For such cases, the displacements and therefore the Dirichlet boundary condition on the symmetry planes are not available and a slip condition is required to be applied. The slip condition for the symmetry plane can be described as follows:

$$U.\vec{n} = 0, \quad for \ \ U \ on \ symmetry \ plane \tag{2.15}$$

$$\frac{\partial U}{\partial x_i} - \frac{\partial U}{\partial x_i}.\vec{n} = 0, \quad for \ \ U \ on \ symmetry \ plane \tag{2.16}$$

Here, $\vec{n}$ is the normal unit vector to the symmetry plane. Eq. 2.15 shows the zero Dirichlet boundary condition for the normal component of the displacement on the symmetry plane and Eq. 2.16 shows the zero Neumann boundary condition for the components parallel to the symmetry plane on the symmetry plane. The combination of Eq. 2.15 and 2.16 shows the slip boundary condition for the symmetry plane.

2.4. Design of the overall solver

The required mesh morpher needs to execute different kinds of operations like the boundary deformation, the bulk mesh deformation and the generation of new volumetric mesh. Based on these operations, the overall solver can be described mainly in three parts as shown in Fig. 2.1. The first part of the overall solver uses a previously developed RBF or FFD based surface manipulator which takes the initial volumetric mesh and the deformation information as inputs and gives the initial geometry and the deformation field on the boundary nodes as the outputs. Internally, this part of the overall solver extracts the boundary mesh first and applies the RBF or the FFD techniques based on the deformation information. The output of the first part is the displacement field on the boundary of the volumetric mesh.

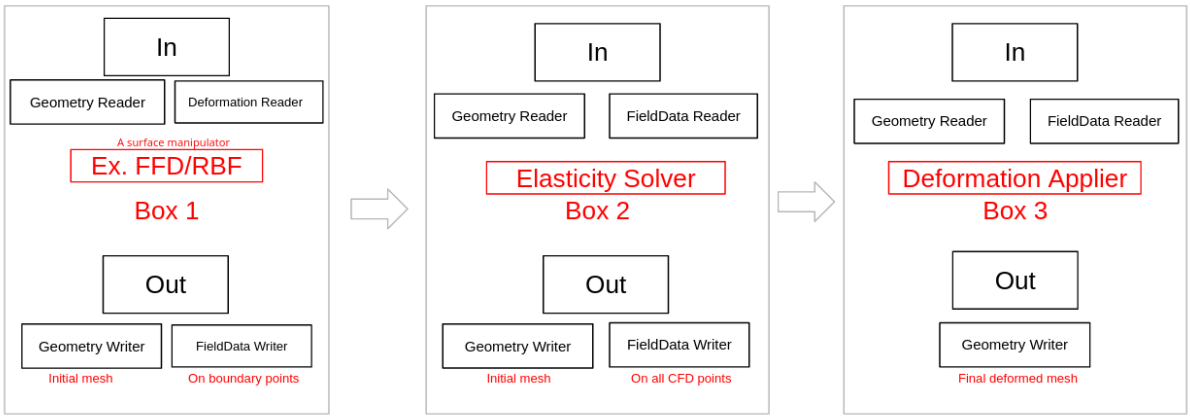


Figure 2.1: Design scheme of the overall solver.

The second part of the overall solver is the elasticity solver where the equation of linear elasticity mentioned in Sec. 2.1 are solved. This elasticity solver takes in the initial volumetric mesh and the output of the first part of the overall solver as inputs and generates the displacement field on the bulk volumetric mesh as the output. For the equations of linear elasticity in the elasticity solver, the output of the first part which is the displacement field on the boundary nodes works as the boundary condition, Dirichlet kind particularly.

The third and final part of the overall solver is the deformation applier as shown in Fig. 2.1. Deformation applier takes in the initial geometry and the displacement field on the bulk mesh as the inputs and generates the final deformed mesh. This is to be noted that in all the three parts, the connectivity information is required and passed as inputs as well. The second part of the overall solver which is the elasticity solver is developed

in the present thesis work. The first and third parts of the overall solver were developed already in a previous work and out of scope of the present work.

2.5. Computation of cell and face gradients

Considering the unstructured 3D domain of the target geometries, the computation of the gradients over cells and faces is done by forming stencils. In present work, a second order least square method is used to compute the cell and face gradients as described in Darwish and Moukalled [46]. A least square method provides a good flexibility in the order of accuracy and the used stencils. For the 2D representation of a generic 3D mesh shown in Fig. 2.2, the gradients on cell centers for a field ϕ according to the least square method can be written as follows:

$$\begin{bmatrix} \sum_{k=1}^{NB(C)} w_k \Delta x_k \Delta x_k & \sum_{k=1}^{NB(C)} w_k \Delta x_k \Delta y_k & \sum_{k=1}^{NB(C)} w_k \Delta x_k \Delta z_k \\ \sum_{k=1}^{NB(C)} w_k \Delta y_k \Delta x_k & \sum_{k=1}^{NB(C)} w_k \Delta y_k \Delta y_k & \sum_{k=1}^{NB(C)} w_k \Delta y_k \Delta z_k \\ \sum_{k=1}^{NB(C)} w_k \Delta z_k \Delta x_k & \sum_{k=1}^{NB(C)} w_k \Delta z_k \Delta y_k & \sum_{k=1}^{NB(C)} w_k \Delta z_k \Delta z_k \end{bmatrix} \begin{bmatrix} \left(\frac{\partial \phi}{\partial x}\right)_C \\ \left(\frac{\partial \phi}{\partial y}\right)_C \\ \left(\frac{\partial \phi}{\partial z}\right)_C \end{bmatrix} = \begin{bmatrix} \sum_{k=1}^{NB(C)} w_k \Delta x_k \Delta \phi_k \\ \sum_{k=1}^{NB(C)} w_k \Delta y_k \Delta \phi_k \\ \sum_{k=1}^{NB(C)} w_k \Delta z_k \Delta \phi_k \end{bmatrix} \tag{2.17}$$

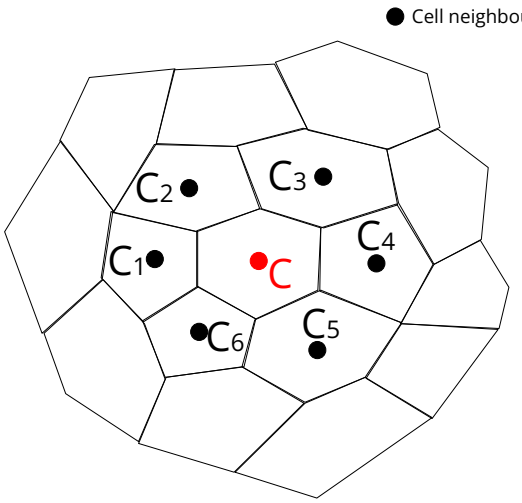


Figure 2.2: 2D representation of a generic geometry with the immediate neighbours.

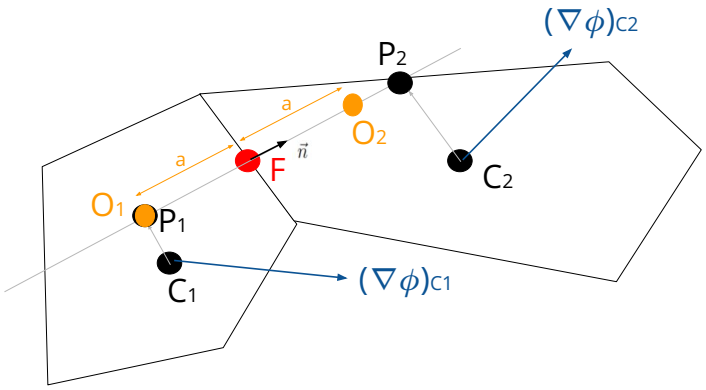
Eq. 2.17 can be used to compute the stencils for gradients at the cell centers given that the matrix in the LHS is not singular. The weight w_k used in Eq. 2.17 describes the properties of the stencil and dependence over the neighbor cells k. Generally, w can be described as a function of the distance between the neighbour and owner cell center as shown in Eq. 2.18.

$$w_k = \frac{1}{|r_{F_k} - r_C|^p} \tag{2.18}$$

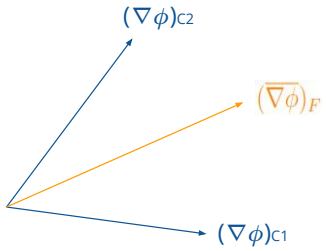
Here, p is an exponent which can be used to describe the dependence of the gradient on the neighbor cells. In the 2D representation as shown in Fig. 2.2, it is possible to show only the neighbours that are adjacent to the faces of the cell under consideration. In 3D geometries, there can be not only the neighbours which share a common face but also the neighbours which share only an edge. In present study, in order to increase the accuracy of the gradients, both the face and the edge adjacent neighbours are considered in the stencils.

In a similar way as described in Darwish and Moukalled [46], the gradients at the face centers are computed by interpolating from the cell centers to the face centers as shown in Fig. 3.4. The gradients at the face centers are provided with 2 different contributions; first, parallel to the face, and second, along the normal of the face.

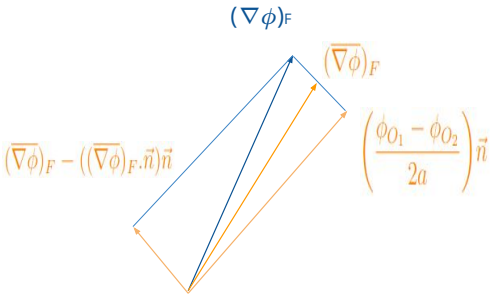
To compute these contributions, the averaging of the neighbour cells is not enough because it does not take into account the actual fields at these neighbour cells. In order to take the contribution of the fields at the neighbour cells, the normal component of the face gradient is computed by projecting the fields at the neighbouring cells (at C_1 and C_2) at the normal line (P_1 and P_2) passing through the face center F. Followed by taking the projections P_1 and P_2, the distance of these projections is computed from the face center F. The distance $a = min(P_1F, P_2F)$ is considered for the computation of the nor-



(a) Face gradient computation.



(b) Average of neighbour gradients.



(c) Components of face gradient.

Figure 2.3: Computation of the face gradients using the cell gradients.

mal component of the face gradient. The component of the face gradient along the normal of the face can be computed as shown in Eq. 2.19.

$$(\nabla\phi)_{F,\perp} = \left(\frac{\phi_{O_1} - \phi_{O_2}}{2a}\right)\vec{n} \tag{2.19}$$

where, the fields ϕ_{O_1} and ϕ_{O2} are computed by interpolating the fields from the cell center to points O_1 and O_2 respectively as follows:

$$\phi_{O_i} = \phi_{C_i} + (\nabla\phi_{C_i}).\overrightarrow{C_iO_i} \tag{2.20}$$

For the component of face gradient parallel to the face, an average of the gradients at the neighbor cells is computed and subtracted from the component of this average along the direction of normal to the face as shown in Eq. 2.21. The average of neighbor stencils (Fig. 2.3b) can be computed by taking a simple average as shown in Eq. 2.22.

$$(\nabla\phi)_{F,\parallel} = (\overline{\nabla\phi})_F - ((\overline{\nabla\phi})_F.\vec{n})\vec{n} \tag{2.21}$$

$$(\overline{\nabla\phi})_F = \frac{((\nabla\phi)_{C_1} + (\nabla\phi)_{C_2})}{2} \tag{2.22}$$

In this way, the face gradient stencils for the whole geometry can be computed using the cell gradient stencils. Though it is to be noticed at the boundary, that a boundary face always has only one neighbour cell (as shown in Fig. 2.4) which makes the interpolation technique explained by Eqs. 2.19- 2.21 inappropriate to use. For the boundary faces, the face gradients are defined as follows:

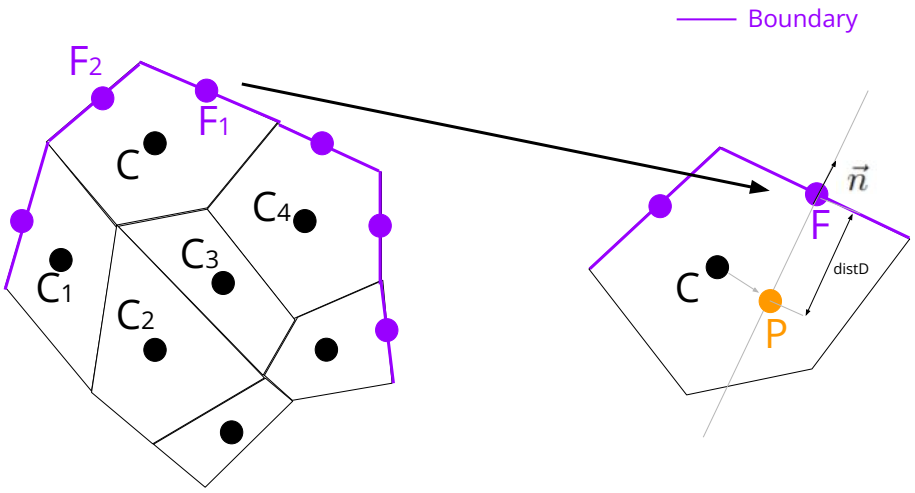


Figure 2.4: 2D representation of a generic mesh at the boundary.

$$(\nabla\phi)_{F,\parallel} = (\nabla\phi)_{C,\parallel} = (\nabla\phi)_C - ((\nabla\phi)_C.\vec{n})\vec{n} \tag{2.23}$$

$$(\nabla\phi)_{F,\perp} = \frac{\phi_P - \phi_F}{distD} \tag{2.24}$$

Here, ϕ_F denotes the field at the boundary face centroid F, ϕ_P denotes the field at a point P projected from the cell center C to the normal of the boundary face, and $distD$ represents the distance between the point of projection and the boundary face center. In Eq. 2.24, ϕ_P can be computed using the cell gradient as shown in Eq. 2.20.

2.6. Boundary corrections in cell and face gradients

It is to be noticed that the cell and face gradients formulation shown in Sec. 2.5 illustrate the gradients in the bulk mesh well but their accuracy at the boundary still needs to be checked. It can be observed in Eq. 2.17, that a cell gradient stencil is constructed using the neighbour cells. In case of a boundary cell, for example in Fig. 2.4, it can be

observed in Eq. 2.17 that the contribution due to the boundaries is not taken into account in computing the cell gradient. In such a case, the accuracy of the gradient becomes $1st$ order instead of desired $2nd$ order. For good quality mesh morphing, it is under utmost priority that the boundary mesh should be preserved, and hence it is needed for the boundary gradients to be of high order accuracy. In order to recover the second order accuracy of cell gradients at the boundary cells, it is required to consider the contribution of the boundary faces as well.

It is evident from Eq. 2.19- 2.21 that the face gradients are interpolated from the cell gradients to the faces. This indicates that the contribution of the boundary faces is required to be considered not only in cell but face gradient stencils as well.

2.6.1. Correction in cell gradient stencils

In order to consider the contribution of the boundary faces in the boundary cell gradient, an additional stencil is defined for all the cells in the geometry. This new face based stencil does not contain any element for the non-border cells. For the border cells, the elements of this second stencil are only the boundary faces associated with the concerned border cell. The contribution or weight of every boundary face is computed in exactly the same way as computed for neighbour cells as shown in Eqs. 2.17 and 2.18 with the use of centroid of the boundary face instead of neighbour cell centers. Therefore, the new overall stencil for the cells can be written as follows:

$$St_{Cell} = St_C + St_{bf} \tag{2.25}$$

Here, St_{Cell} represents the overall stencil for the cell gradient, St_C represents a stencil with the contributions from the neighbour cells and St_{bf} represents a stencil with the contributions from the boundary faces. Following equation needs to be used to compute the gradient at the cell center:

$$(\nabla\phi)_C = St_C.(\phi_N) + St_{bf}.(\phi_{bf}) \tag{2.26}$$

Here, ϕ_N is the set of fields at the neighbour cells, and ϕ_{bf} is the set of fields at the neighbour boundary faces. An example of the sets ϕ_N and ϕ_{bf} for a cell C can be seen in Fig 2.4 where ϕ_N is the set of fields on cell centers C_1-C_4 and ϕ_{bf} is the set of fields on boundary faces F_1-F_2 respectively.

2.6.2. Correction in face gradient stencils

As it can be seen from Eqs. 2.21- 2.22 and 2.23- 2.24, in the whole geometry, the face gradient stencils depend on the cell gradient stencils. The addition of the contributions from boundary faces in the cell gradient stencils as shown in Eq. 2.25 makes this contribution to be accounted in the face gradient stencils as well. As shown in Sec. 2.5, since there exist two kinds of face gradient stencils; one for the boundary faces in Eqs. 2.23- 2.24 and the other for the faces in internal mesh in Eqs. 2.19- 2.21, both of these kinds of stencils are needed to be corrected differently.

Case1: Boundary faces

For the boundary faces, Eqs. 2.23- 2.24 are modified in order to consider the contribution of boundary faces which is considered in Sec. 2.6.1 for the cell gradient stencils.

$$St_{Face,\parallel} = (St_C - (St_C.\vec{n})\vec{n}) + (St_{bf} - (St_{bf}.\vec{n})\vec{n}) \tag{2.27}$$

$$St_{Face,\perp} = \frac{(St_C.\overrightarrow{CP})\vec{n}}{distD} + \frac{((St_{bf}.\overrightarrow{CP}) - \phi_F)\vec{n}}{distD} \tag{2.28}$$

Eqs. 2.27 and 2.28 contain contributions from the neighbour cells and the contribution from the neighbour boundaries as well. A boundary face gradient can be computed using the following equation:

$$(\nabla\phi)_F = St_{Face,\parallel}.\{\phi_N, \phi_{bf}\} + St_{Face,\perp}.\{\phi_N, \phi_{bf}\} \tag{2.29}$$

Here, ϕ_N is the set of fields at the neighbor cells of the neighbor cell of the boundary face, and ϕ_{bf} is the set of fields at the boundary faces of the neighbour cell of the boundary face. An example of the sets ϕ_N and ϕ_{bf} for a Face F can be seen in Fig. 2.4 where the only neighbour of F is cell C. Hence, the set ϕ_N for face F is the neighbour cells of C which are $C_1 - C_4$, and set ϕ_{bf} is boundary neighbour faces of C which are $F_1 - F_2$.

Case2: Faces in internal mesh

For faces in the internal mesh, Eqs. 2.19- 2.21 are corrected by considering the corrected cell gradients. The corrected parallel component of the face gradient stencils is computed as follows:

$$St_{Face,\parallel} = (\overline{St_F} - (\overline{St_F}.\vec{n})\vec{n}) + (\overline{St_{bf,F}} - (\overline{St_{bf,F}}.\vec{n})\vec{n}) \tag{2.30}$$

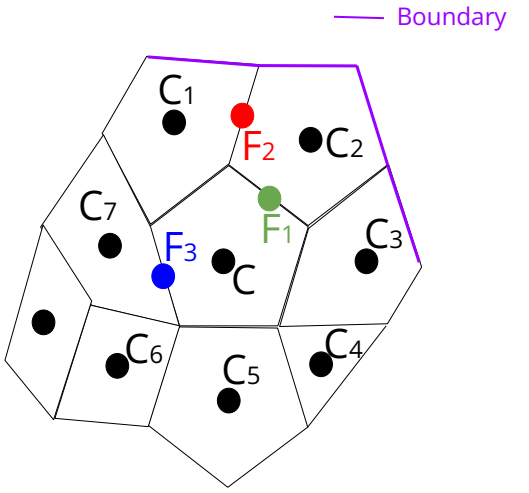


Figure 2.5: Bulk mesh near the boundary with different kinds of neighbours.

Here, $\overline{St_F} = \frac{St_{C1}+St_{C2}}{2}$, and $\overline{St_{bf,F}} = \frac{St_{bf,C1}+St_{bf,C2}}{2}$, where $St_{bf,C1}$ is the stencil containing the boundary face neighbours of the first neighbour cell and $St_{bf,C2}$ is the stencil containing the boundary face neighbours of second neighbour cell. The corrected normal component of the face gradient stencils is computed as follows:

$$St_{Face,\perp} = \frac{(St_{C1}.\overrightarrow{O_1C_1} - St_{C2}.\overrightarrow{O_2C_2})\vec{n}}{2.distD} + \frac{(St_{bf,C1}.\overrightarrow{O_1C_1} - St_{bf,C2}.\overrightarrow{O_2C_2})\vec{n}}{2.distD} \tag{2.31}$$

In this case, there can be different sub-cases as well. All the cell gradient stencils do not have the contributions from the boundary faces. Among the non-boundary faces, there exist faces which have either 1, 2 or no neighbour cells with boundary face contributions as shown in Fig. 2.5 by face centers F_1, F_2, and F_3 respectively. The Eqs. 2.30 and 2.31 are generic for all these kind of faces and applicable for all.

2.7. Computation of the Laplacian term

With the computation of the cell and face gradient stencils, $Termb2$ in Eq. 2.12 which is the only term in the LHS, can be computed in terms of a product of a new stencil $St_{Lapl,Cell}$ and the field at the elements of this stencil as follows:

$$Termb2 = St_{Lapl,Cell}.\Phi \tag{2.32}$$

$$St_{Lapl,Cell} = \sum_F \mu^F St_F.{n_i}^F \Delta S^F \tag{2.33}$$

Here, $St_{Lapl,Cell}$ is the LHS in the present explicit linear solver, whereas Φ is the unknown vector in the linear solver. It is to be noticed that in Eq. 2.12, the Lamé constant μ works as diffusivity, which is not a constant throughout the geometry but depends on the elasticity Coefficient E and Poisson's ratio ν which in turn depend on the mesh characteristics. In present work, μ^F at a face F is considered to be the average of the two neighbour cells of the face F. In case of boundary faces, μ at the only neighbour cell in the internal mesh is assigned to the boundary face.

It is to be noticed that at the border cells, Eq. 2.33 does not considers the contribution of the boundary faces. This contribution now is directly incorporated in the source term of the linear solver. This contribution in the source term is computed by the product of the $St_{Lapl,Cell}$ and field at the boundary faces alongside boundary face contribution from the St_F in Eq. 2.33. The other contribution for the source term are computed using Eqs. 2.10 and 2.11.

2.8. Convergence condition and control method

In present work, the residuals after every elasticity solver iteration are computed using a method similar to the one used in [46]. This method is shown hereby for a matrix system of equation $Ax = b$. Residuals r_i for the i^{th} component of the displacement can be defined as:

$$r_i = b_i - Ax_i \tag{2.34}$$

Applying residual scaling using the following normalization steps:

$$n_i = \sum(|Ax_i - A\overline{x_i}| + |b_i - A\overline{x_i}|) \tag{2.35}$$

Here, $\overline{x_i}$ stands for the average displacement of the i^{th} displacement component. The scaled residuals can be obtained as:

$$r_i = \frac{1}{n_i}\sum|b_i - Ax_i| \tag{2.36}$$

$$R = max(r_i) \tag{2.37}$$

In the present study, for the linear elasticity solution to converge, the maxima of the residuals, R, is kept under a tolerance of 1E-10. In order to converge the solution, a

relaxation factor is considered which is shown in the following equation:

$$x = ax + (1 - a)x_{old} \tag{2.38}$$

Here, a is the relaxation factor, x is the solution of the new elasticity solver iteration, and x_{old} is the solution of the previous or old iteration. In present work, different relaxation factors between 0 and 1 are used for different case studies and geometries.

In order to solve the linear system of equations, a PETSc library is used with a tolerance of 1E-12.

2.9. Interpolation from nodes to face centers and cell centers to nodes

In present work, there are mainly two kinds of interpolation required in two different steps, namely; nodes to face center interpolation, and cell center to nodes interpolation.

2.9.1. Nodes to face center interpolation

This kind of interpolation is required to connect the first part of the overall solver to the second part of the overall solver. The displacement data obtained from the RBF/FFD manipulator is available at the nodes but the elasticity solver needs the boundary condition at the face centers. The difference in data location creates the need of an interpolation from nodes to the face centers at the boundary of the volumetric mesh as shown in Fig. 2.6. The interpolation technique used in the present work for this purpose is shown as follows:

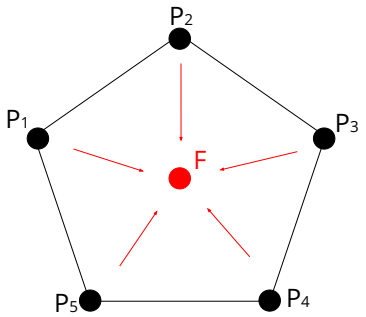


Figure 2.6: Interpolation from nodes to face centers.

$$X_F = \frac{\sum_{i=1}^{n} w_i.X_i}{\sum_{i=1}^{n} w_i}, \; where \quad w_i = \frac{1}{(d_{FP_i})^p} \tag{2.39}$$

Here, X_F is the interpolated field at the face center F, X_i is the field at the neighbour node i, n is the number of neighbour nodes of the face center F, and w_i is the weight function for the i^{th} neighbour of the face center which is equal to the inverse of the distance d_{FP_i} between this neighbour and the face center F to the power an exponent p.

2.9.2. Cell center to nodes interpolation

The elasticity solver in the present study takes into account a FVM method of distretization to solve the linear elasticity equations with fields on the cell centers. As a result of the this consideration, the output of the elasticity solver, which is the displacement of the bulk mesh, is also obtained on the cell centers. In order to define or update a geometry, the displacements are required to be on the node location and not on the cell centers. Therefore, there is a need of an interpolation from cell centers to the nodes in the volumetric mesh to provide the nodes with the intended displacements as shown in Fig. 2.7. In present work, this interpolation is done in the following way:

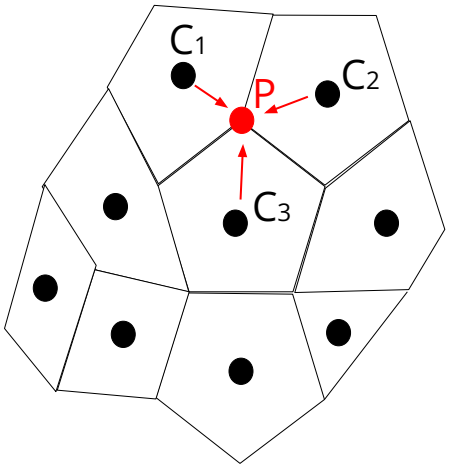


Figure 2.7: 2D representation of interpolation from cell centers to nodes in a volumetric mesh.

$$X_P = \frac{\sum_{i=1}^{n} w_i (X_{C_i} + \frac{\partial X}{\partial z_k}\big|_{C_i} . (\vec{d}_{PC_i}))}{\sum_{i=1}^{n} w_i}, \; where \quad w_i = \frac{1}{(V_i)^p} \tag{2.40}$$

Here, X_P is the interpolated field at the node P, X_{C_j} is the field at the cell center i, n is the number of neighbour cells of the node P, and w_i is the weight function for the i^{th} neighbour of the node P which is equal to the inverse of the volume V_i of the neighbour cell to the power an exponent p. Vector $\vec{d}_{PC_i}$ denotes the vector connecting point P to

the neighbor C. The method of interpolation shown in Eq. 2.40 requires the gradients at cell centers C_i which can be computed as described in Sec. 2.6.

2.10. Multi-step algorithm

As mentioned in Sec. 2.2, an explicit scheme is used to solve the system of Eq. 2.8. From the first part of the solver, we get the initial geometry and the boundary displacements as the boundary condition for elasticity solver as shown in Fig. 2.1. The algorithm incorporated in the present work in the elasticity solver is shown in Algorithm 2.1.

Algorithm 2.1 Solution algorithm for the elasticity solver with a single step.

1: Setup the solver: Initial geometry, boundary displacements on nodes, relaxation factor
2: Interpolate displacements from boundary nodes to boundary face centers
3: Computation of the cell and face gradient stencils
4: Computation of the Lamé constants
5: Computation of the Laplacian stencils
6: Initial guess for the displacements in the bulk mesh
7: Initialize the solver by passing the LHS
8: **while** $max(residuals) < Tolerance$ **do**
9: Compute the cell and face gradients
10: Compute the source terms due to $Terma$ and $Termb1$
11: Compute the source term due to the contribution from Laplacian term
12: Compute the overall source term
13: Solve the system of linear equations using PETSc component by component
14: Compute the residuals using the new and old solutions
15: **end while**
16: Interpolate the obtained displacements from cell centers to nodes
17: Calculate the error between the boundary displacement provided to elasticity solver and obtained from elasticity solver and propagate into the bulk mesh
18: Final displacements

The implementation of the elasticity solver as shown in Algorithm 2.1 has some pros and cons as well. In this algorithm, it can be noticed that all the deformation is executed directly in one step. This results in considering the cell and face gradient stencils computed using the initial undeformed geometry which can be a lot different than the final geometry. Also, it can be noticed that the Lamé constants which depend on the mesh characteristics,

Algorithm 2.2 Solution algorithm for the elasticity solver with multi-step.

Setup the solver: Initial geometry, boundary displacements on nodes, relaxation factor
Interpolate displacements from boundary nodes to boundary face centers
Computation of the cell and face gradient stencils
Computation of the Lamé constants
Computation of the Laplacian stencils
Initial guess for the displacements in the bulk mesh
Divide the boundary displacements by nsteps
Initialize the solver by passing the LHS
for $(istep = 0;\ istep < nsteps;\ istep++)$ **do**
  while $max(residuals) < Tolerance$ **do**
    Compute the cell and face gradients
    Compute the source terms due to $Terma$ and $Termb1$
    Compute the source term due to the contribution from Laplacian term
    Compute the overall source term
    Solve the linear system of equations using PETSc component by component
    Compute the residuals using the new and old solutions
  end while
  Interpolate the obtained displacements from cell centers to nodes
  Calculate the error between the boundary displacement provided and obtained from this step and propagate into the bulk mesh
  Update the displacements in the bulk mesh
  Apply the displacements on the bulk mesh and update the geometry
  if istep < nsteps-1 **then**
    Update the cell and face gradient stencils
    Update the Lamé constants
    Update the Laplacian gradient stencils
    Update the LHS in the solver
  end if
  Compute the total displacement upto this step
  Set the solution as the initial guess for the next step
end for
Restore the initial geometry
Restore the initial boundary condition
Final displacements

are computed before the while loop in step 7 which in turn indicates that the Lamé constants belongs to the initial geometry which can be very different from the final geometry. Similarly, the amplitude of the error propagated in step 16 using the Laplacian method also comes out a lot more considerable in one step. Therefore, because of these reasons, it is needed that an incremental deformation is provided to the mesh under consideration. An important benefit of using the incremental deformation approach is that the linear elasticity law stands for small deformations only and considering an incremental deformation enables the model to be applicable for high deformations as well.

By applying this multi-step approach, the approximations considered can be minimized. In present work, the algorithm for the implementation of this multistep deformation is described in Algorithm 2.2.

It can be observed in Algorithm 2.2 that the cell and face gradient stencils and the Laplacian stencil (which is LHS) are updated after every step in order to consider the newly available geometry. Similarly, Lamé constants are updated after every step. In the end of this algorithm, the initial Geometry is restored to apply the accumulated displacement in the bulk mesh which is done in the third part of the overall solver. The multistep is expected to give better results because of the aforementioned improvements but computationally expensive as well at the same time.

3 | Results: A cube of 64k cells

This chapter is dedicated to demonstrate the results obtained using the linear elasticity solver developed in the present work as shown in Chapter 2. First, the accuracy of the computed gradients using the stencils derived in Sections 2.5 and 2.6 for a test case is compared. Following the validity of the gradients, a bigger case of a non-uniform cube is considered to explain the results. In this chapter, the results are obtained for different kinds of deformations using single and multi-step approaches as explained in Sec. 2.10. In the final part of this chapter, the results of the multi-step approach are compared with the results of a single step approach. The software tool *ParaView* is used to visualize the results.

3.1. Gradient computation at cell and face centers

As shown in Eqs. 2.10- 2.12, the present linear elasticity solver requires the gradient of the field variable displacement at the interfaces of the cells. It is evident from Sec. 2.5 that the gradients at the interfaces are interpolated from the cell gradients. Therefore, the accuracy of the solution of the linear elasticity solver depends on the accuracy of the computed cell and face gradients both. In present work, a generic second order least square methodology to compute cell and face gradients is defined as shown in Sec. 2.5 which suffers from first order accuracy at the boundaries. The accuracy of second order at the boundary cells and faces is preserved by introducing corrections as derived in Sec. 2.6. In present section, the accuracy of cell and face gradients without and with the correction are demonstrated for a generic field function as shown in Eq. 3.1. This function $F(x, y, z)$ contains linear and non-linear terms to demonstrate the accuracy of gradients not only for different polynomials but for polynomials coupled in different dimensions as well.

$$F(x, y, z) = a.x + b.y^2 + c.z^3 + d \tag{3.1}$$

Here, a, b, c, and d are coefficients of the linear, quadratic, cubic and constant terms respectively. For the ease of understanding, a uniform cube with 64 cells and 240 faces is

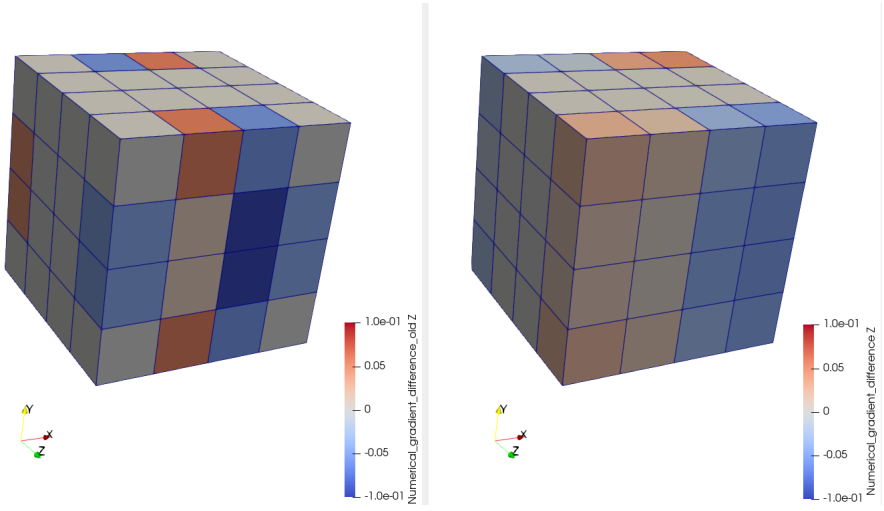
used in the present study. The span of the used cube is $(2, 2, 2)$ with all the cells having the same volume. The center of the cube is kept at $(0, 0, 0)$. A relative error Err_{rel_i} for component i of the computed gradient is calculated as follows:

$$Err_{rel_i} = \frac{(F_{i,Comp} - F_{i,Ana})}{F_{i,Ana}} \tag{3.2}$$

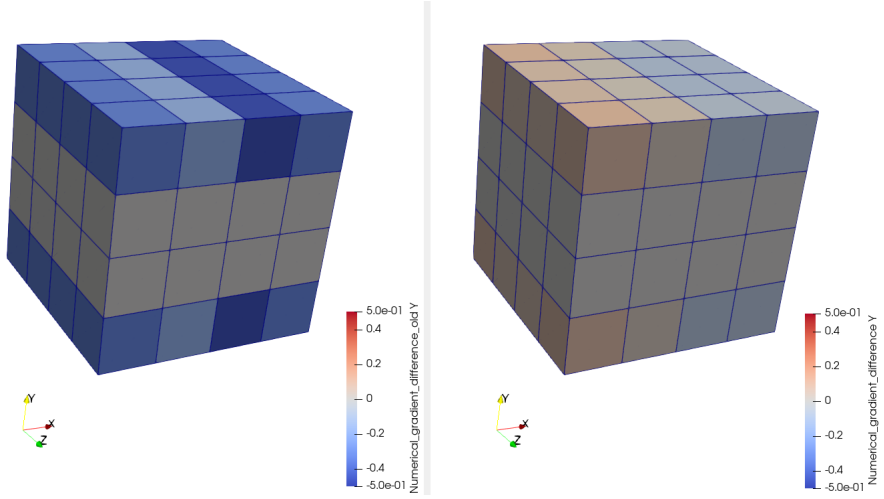
Here, $F_{i,Comp}$ and $F_{i,Ana}$ represent the computed and analytical gradients respectively for the i^{th} component.

3.1.1. Cell gradients

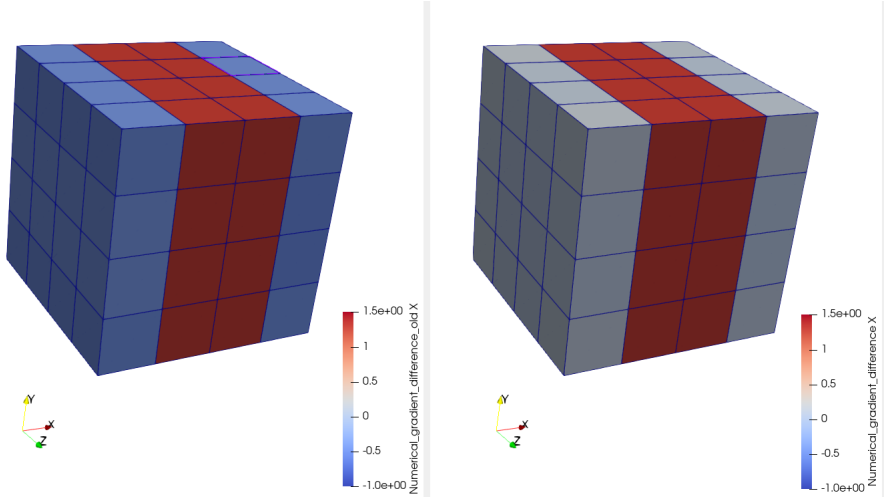
In this subsection, the relative error in the computed partial derivative of the function $F(x, y, z)$ over the cell centers is presented component by component using $ParaView$ tool as shown in Fig. 3.1. The upper and lower limits of the relative errors are kept same for the same components to compare the figures by the intensity of the colors. Among the color map, red color denotes a positive error, blue color denotes a negative error and grey color denotes an error approaching zero. In Fig. 3.1, the figures on left and right denote the gradients without and with the boundary face contributions derived in Sec. 2.6 respectively. The x component of the gradient shown in Fig. 3.1a shows that the inclusion of boundary face contributions as shown in Sec. 2.6 makes the red and blue colors in the cube lighter, which indicates that the relative error in the x gradient has reduced. Similarly, in Figs. 3.1b and 3.1c, it can be observed that the relative error in the y and z gradients respectively decreases by the introduction of the boundary face contributions.



(a) x component.



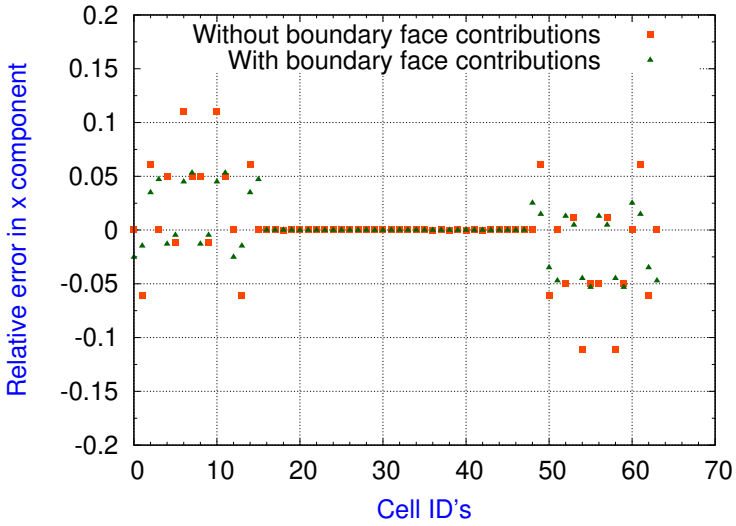
(b) y component.



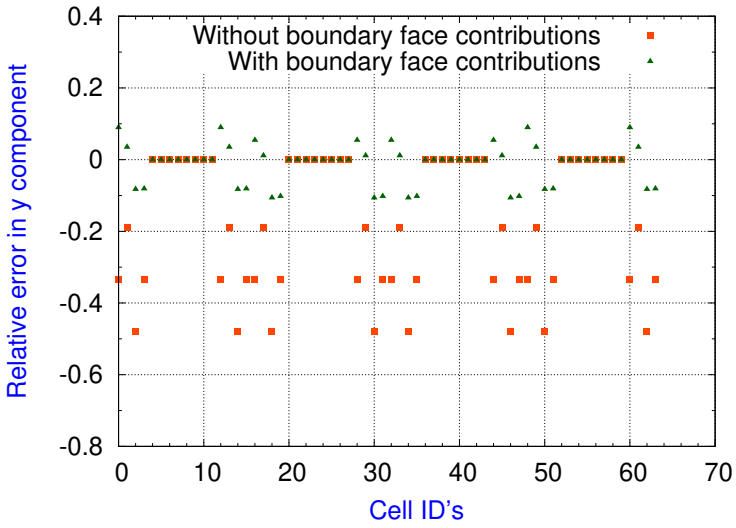
(c) z component.

Figure 3.1: Relative error in cell gradients without and with boundary face contributions.

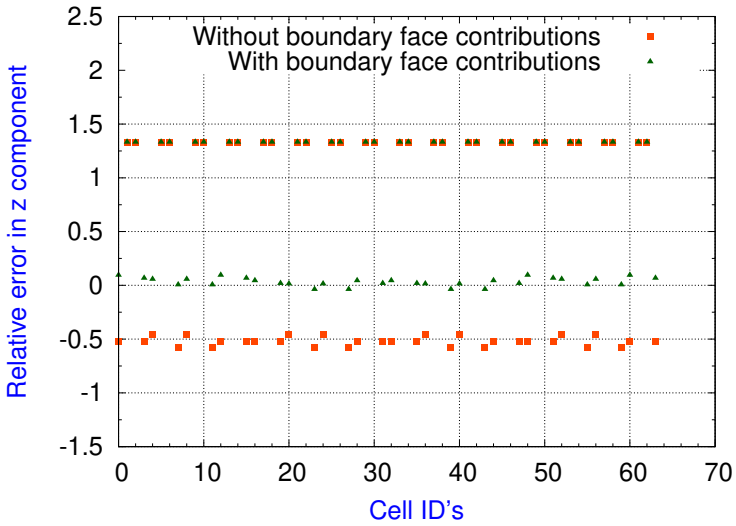
To get a more clear and precise estimation of these gradients, the results are plotted on relative error *vs* cell ID's plane as shown in Fig. 3.2 component by component. It is evident from the plots that the precision of the gradients improves by the introduction of the boundary face contributions. For the results including boundary face contributions, though the gradients improve by incorporating the corrections in the stencils, a careful evaluation of the relative errors indicates that for the x gradient component, which is the linear part of $F(x, y, z)$, the range of relative error stays $\pm 5\%$, whereas for y gradient component, which is the quadratic part of $F(x, y, z)$, the range increases to $\pm 10\%$, and for z gradient component, which is the cubic part of $F(x, y, z)$, the range of relative error further increases to $(0, 130)\%$. Since the cube being descretized in a very course manner,



(a) Gradient of linear part.



(b) Gradient of quadratic part.



(c) Gradient of cubic part.

Figure 3.2: Relative errors in cell gradient vs cell ID's.

errors upto ±10% are acceptable but the error of 130% for the cubic function needs to be studied further.

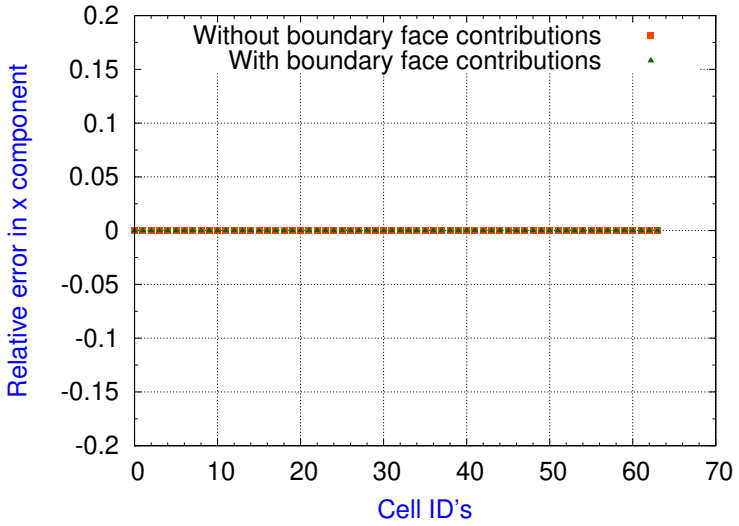
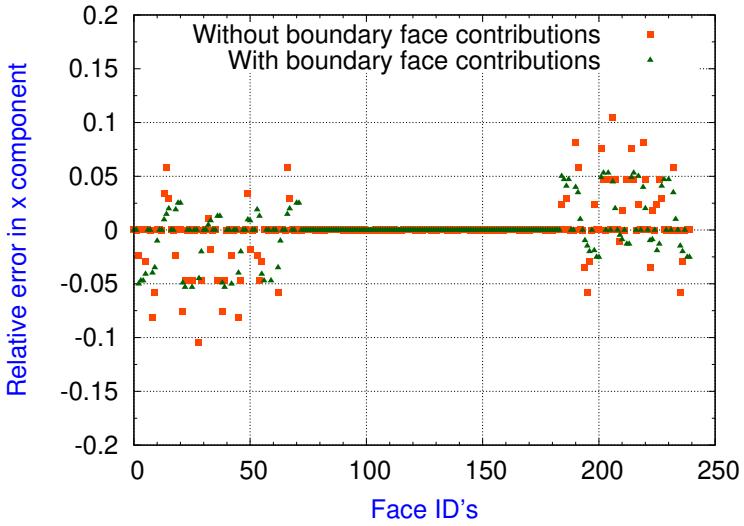


Figure 3.3: Gradient for a linear equation.

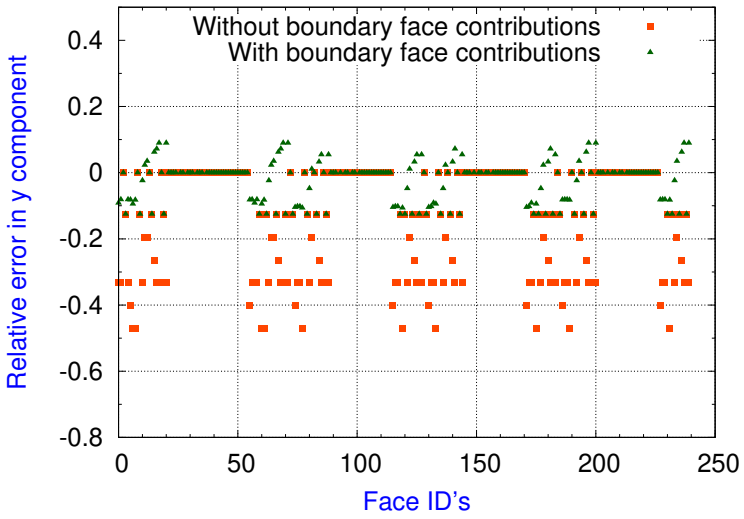
For a linear function, the gradients from the least square method are expected to be exact and not approximated even for a very course symmetric mesh but a non-zero relative error is found as shown in Fig. 3.2a. The main reason behind this is the presence of the other components in the function $F(x, y, z)$. In Fig. 3.3, the x gradient of $F(x, y, z)$ with coefficients b and c equal to 0 is shown without and with the contributions from the boundary faces. It is observed that the relative error in x gradient comes out to be approaching to 0 all over the cube for both without and with boundary face contributions.

3.1.2. Face gradients

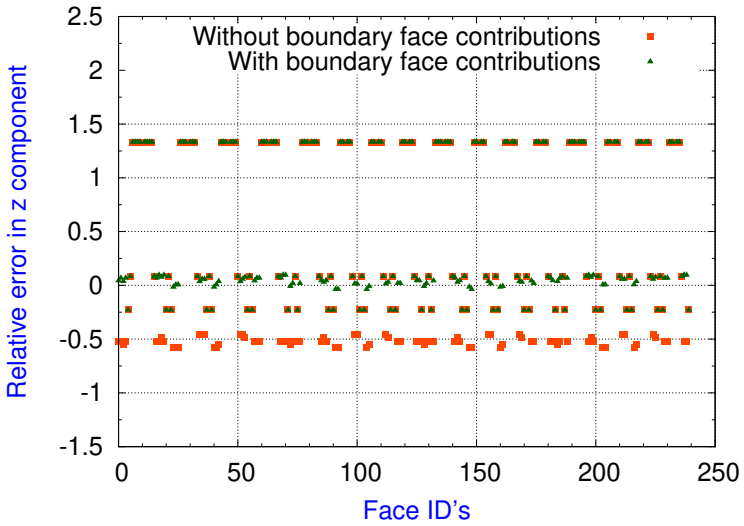
In this subsection, the computed partial derivative of the function $F(x, y, z)$ over the face centers is presented component by component as shown in Fig. 3.4. Since the *paraview* tool does not support the face ID field visualization, the results are shown on relative error *vs* face ID's planes. It can be observed from the results that the relative error decreases with the inclusion of the boundary face contributions for all the components. Almost like cell center gradients, for the case with boundary face contributions, the x and y components of the gradient show the relative error ranges of ±6% and ±12% respectively, whereas the z component of the gradient shows a relative error range of $(-25, 130)\%$. The range of relative error for the cubic part is again a considerable one and needs to be inspected.



(a) Gradient of linear part.



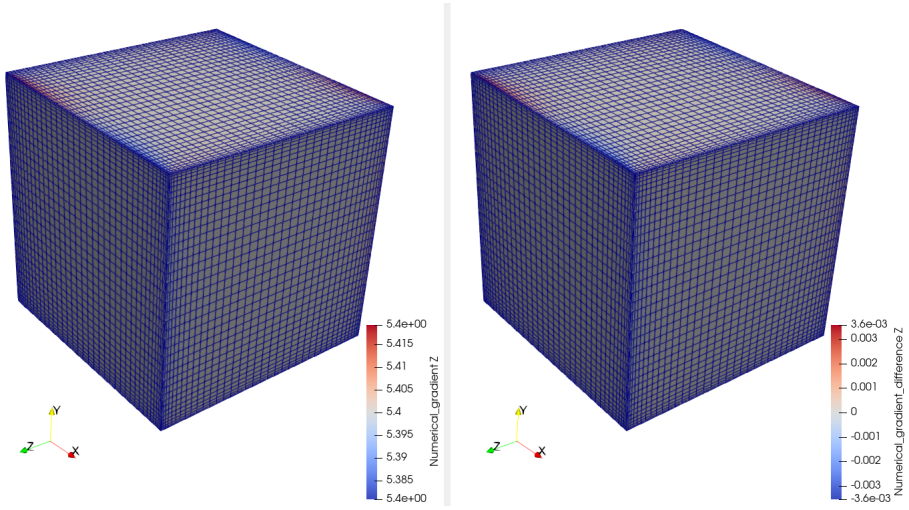
(b) Gradient of quadratic part.



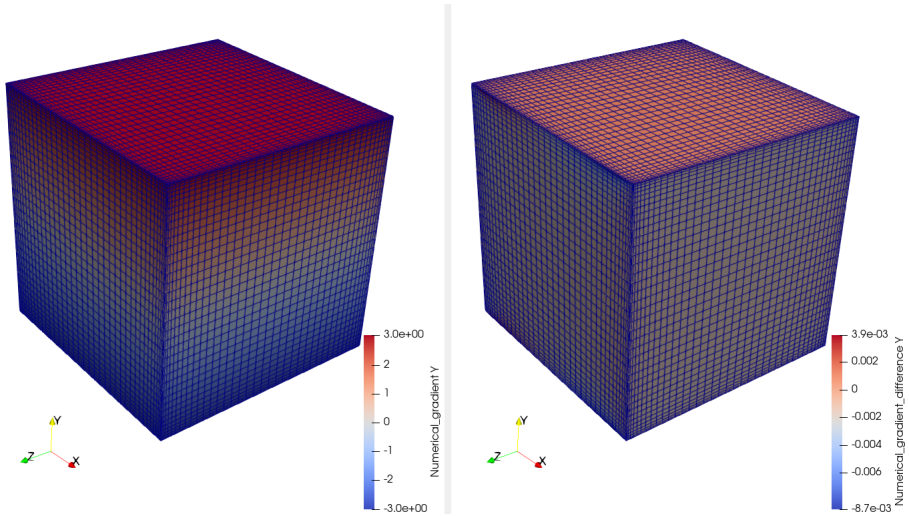
(c) Gradient of cubic part.

Figure 3.4: Face gradients without and with boundary face contributions.

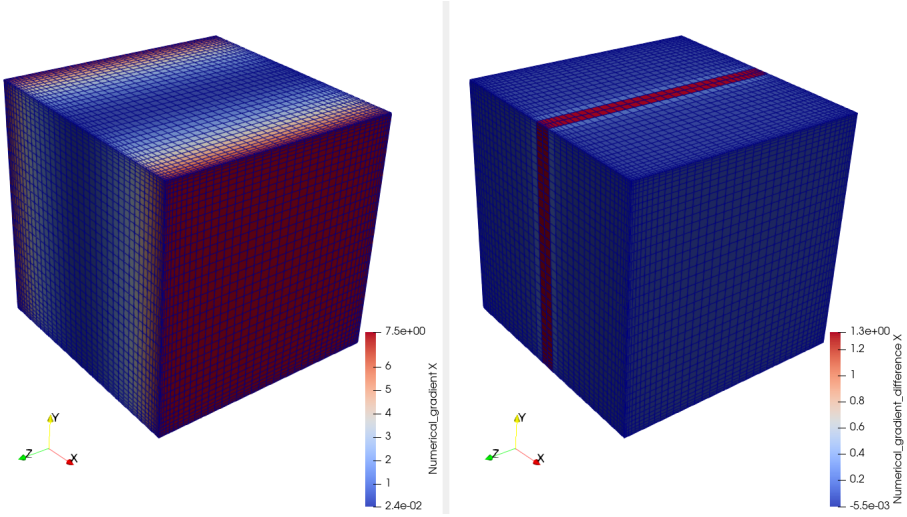
To demonstrate the computation of gradients over a more complex and fine mesh, and to inspect the high relative error in the gradient of the cubic part of $F(x, y, z)$, a non-uniform cube with $64k$ cells is considered. The cube is made in such a way that the cell thickness of the cells increases from the boundary to the bulk making the volume of all the cells non-uniform in the bulk. For the case with the inclusion of the boundary face contributions, Fig. 3.5 shows the computed cell gradients in the left and the relative errors in the computed cell gradients in the right respectively component by component. It can be observed in Figs. 3.5 that the range of relative errors in linear and quadratic parts of the function $F(x, y, z)$ drops to $\pm 0.36\%$ and $\pm 0.87\%$ respectively. Though a maxima of relative error 130% still exists in the gradient of the cubic part of the function $F(x, y, z)$ as shown in Fig. 3.5c.



(a) Gradient of linear part.



(b) Gradient of quadratic part.



(c) Gradient of cubic part.

Figure 3.5: Cell gradients and their relative errors with face contributions using Paraview.

It is to be noticed in the left figure of Fig. 3.5c that the magnitude of the gradient at the cells near the center of the cube where the x coordinate approaches 0 is very close to 0. This makes the order of the analytical gradient close to the order of the error which results in higher relative error (see Eq. 3.2) close to the center of the cube. Overall, since the magnitude of gradient for the cubic part of Eq. 3.1 is close to 0 near the coordinate 0 as shown in Fig. 3.5c, the impact of high relative error can be considered as negligible. Therefore, based on these results it can be said that the gradients in the present work are accurate enough to develop the elasticity solver and carry out the mesh morphing.

3.2. Mesh morphing results: outward and inward deformations

In present work, the results of the developed elasticity solver are demonstrated on a non-uniform cube with 64k cells by applying mainly two different and opposite kinds of boundary conditions; namely, outward and inward. In the outward kind of boundary condition, all the 6 faces of the cube are expanded by pivoting the center of the faces and pulling by a length 60% of the edge length of the cube. In the inward kind of boundary condition, the same faces are compressed by pivoting the centers of the faces and pushing by a length 25% of the edge length of the cube. To obtain the results in the present section, the elasticity coefficients are computed using an inverse of volume method as explained in Sec. 2.1 with exponent $p = 1$. In present work, because of the use of a

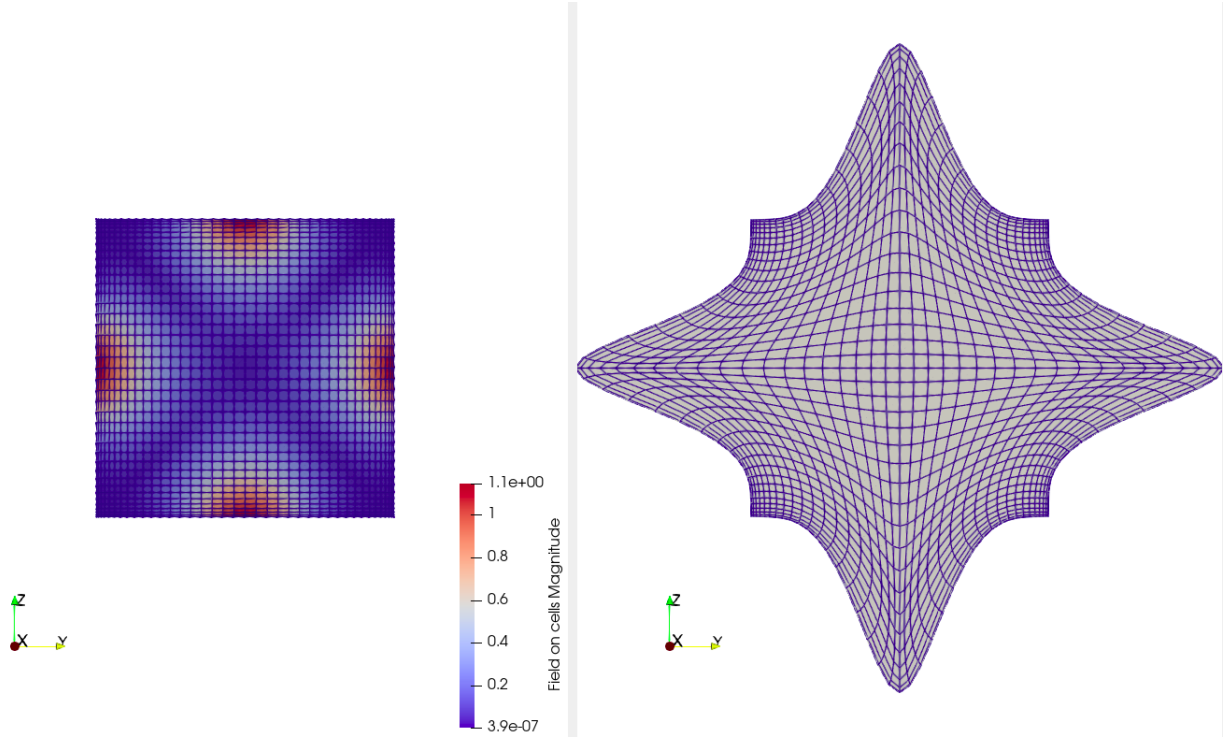


Figure 3.6: Mesh morphing for a cube with 64k cells: outward deformation.

FVM for discretization, the displacement field is computed at the cell centers and then interpolated to the nodes. The magnitude of the displacements obtained from the linear elasticity solver for an outward deformation at the cell centers are shown in the left of Fig 3.6. The displacement field at the cell centers increases from blue to red color. After interpolating the displacement field from the cell centers to the nodes and applying this displacement field on initial geometry (in the final part of the overall solver as shown in Sec. 2.4), the final deformed geometry is obtained as shown in right of Fig 3.6. Similarly, for the inward deformation, the displacements on the cell centers and the final deformed mesh are shown in Fig. 3.7. For the cases of outward and inward deformations, it is observed that the boundary cells of the cube show the least shape change and hence preserve their shapes. This happens because of more stiffness at the cells close to the boundaries due to the low cell volumes near the boundaries. Fig. 3.8 shows a section of the deformed cube for the outward and inward deformations. It is to be noticed from the results that the orthogonality of the mesh near the boundary is very close to 90 deg, which shows a good quality of the mesh near the boundary. It can be also observed that the deformed mesh in both the outward and inward deformations is not in straight line but curved lines which means that the displacements in different dimensions are dependent. The reason for this behaviour of the deformed mesh is the coupled behaviour of the linear elasticity equation defined by a trace as shown in Eq. 2.10.

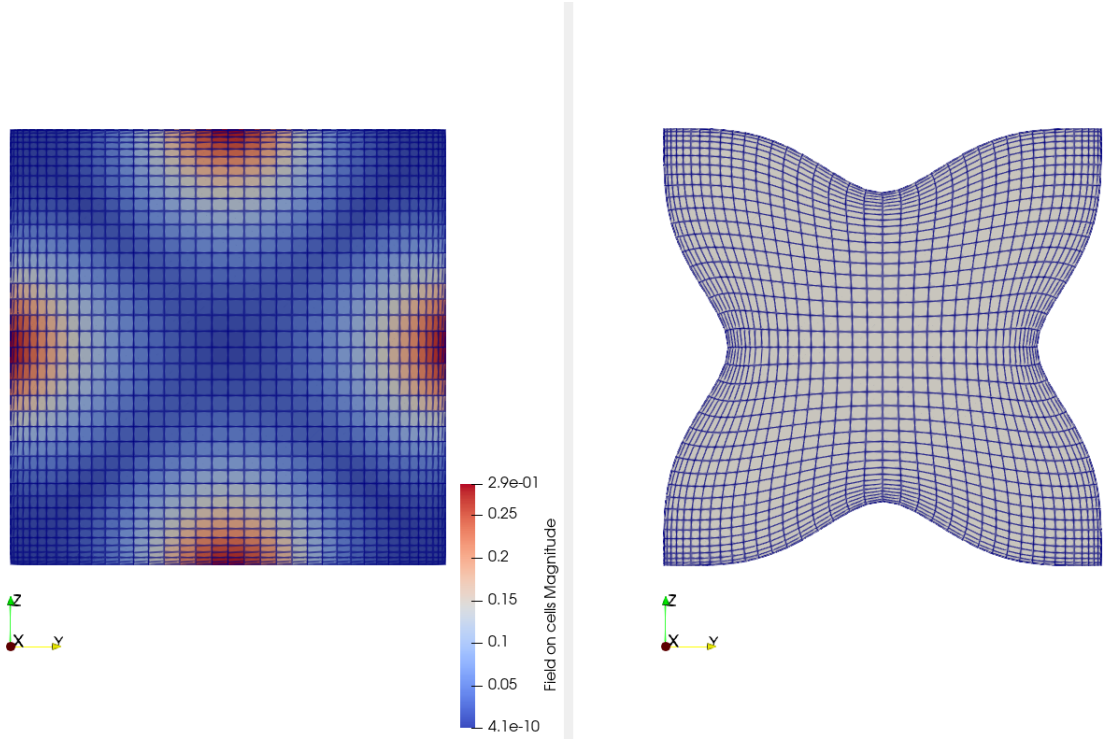
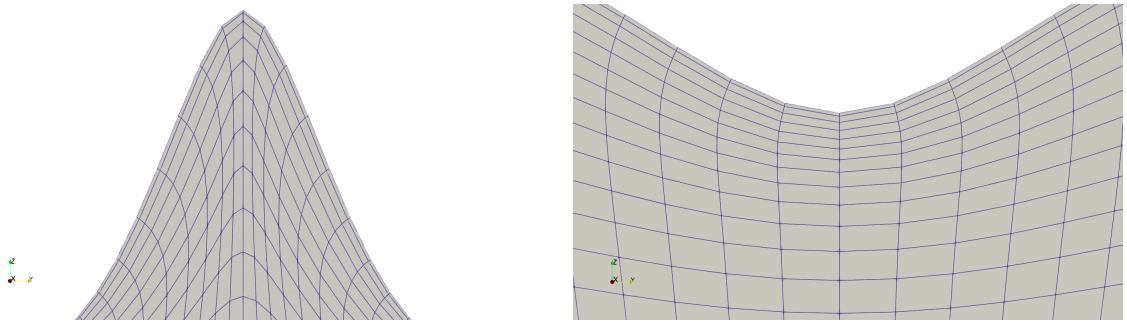


Figure 3.7: Mesh morphing for a cube with 64k cells: Negative deformation.



(a) Outward deformation.

(b) Inward deformation.

Figure 3.8: Zoomed figures of the morphed mesh close to the boundary.

A table containing various mesh quality parameters as defined in Appendix A for the outward and inward deformations is shown in Table 3.1. Positive minimum cell volumes and passed cell face validity show that both the cases generate valid meshes for the applied displacements. For a good quality mesh, the non-orthogonality at any interface should not be more than 70° which is satisfied by the results in Table 3.1. For the mesh morphing applications as discussed in Chapter 1, the quality of the mesh at the boundaries should be good. It is evident from Table 3.1 for the present case of a non-uniform cube, that the maximum non-orthogonality at the boundary comes out to be almost 17° and 3.5° for the

outward and inward deformations respectively which show a good mesh quality.

Deformation	**Max. non-orthogonality (deg.)**	**Max. boundary non-orthogonality (deg.)**	**Avg. non-orthogonality (deg.)**	**Max. skewness**	**Min. cell volume**	**Cell face validity**
Outward	66.13	17.11	21.76	0.83	1.25e-07	1
Inward	22.71	3.65	6.64	0.44	1.25e-07	1

Table 3.1: Mesh quality in outward, inward and no deformation cases.

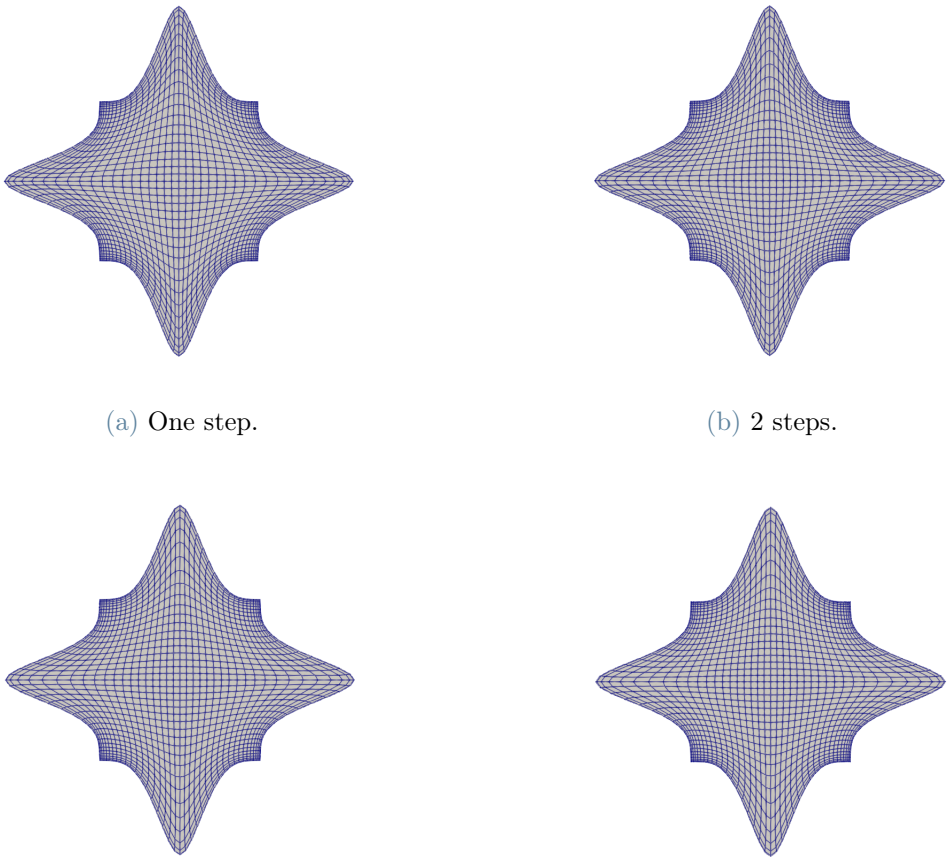
3.3. Multi-step results and comparisons with single step results.

Following the discussion in Sec. 2.10 regarding the merits of a multi-step approach, the present section describes the results for the case of a non-uniform cube mentioned in the previous section using a multi-step approach. For the current study, 4 different number of steps are chosen and compared for both the outward and inward deformation boundary conditions. The boundary conditions used in the present section are similar as used in the previous section. As per the discussion in Sec. 2.1, the Lamé constants are computed using an approach mentioned in Eq. 2.5 to consider the rigid body motion as well.

3.3.1. Outward deformation

For an outward deformation as mentioned in the previous section, the non-uniform meshed cube is simulated with 1, 2, 5 and 10 steps. The results for these 4 different cases are shown in Fig. 3.9. In the case of an outward deformation, the effect of multi-step is not so clear in the cross section images, and therefore, a mesh quality Table 3.2 is shown for details on the quality of the mesh with the number of steps.

It can be observed in Table 3.2 that the maximum non-orthogonality does not change much but the average non-orthogonality decreases considerably by increasing the number of steps. It can be also observed that the maximum non-orthogonality at the boundary decreases as well which is much needed improvement for mesh morphing applications. For all the cases with different number of steps, it is found that the minimum cell volume



(a) One step.

(b) 2 steps.

(c) 5 steps.

(d) 10 steps.

Figure 3.9: Outward deformation using the multi-step approach as shown in Sec. 2.10.

Steps	**Max. non-orthogonality (deg.)**	**Max. boundary non-orthogonality (deg.)**	**Avg. non-orthogonality (deg.)**	**Max. skewness**	**Min. cell volume**	**Cell face validity**
1	61.99	24.37	20.69	0.80	1.25e-07	1
2	61.63	22.79	19.17	0.83	1.25e-07	1
5	62.01	22.02	18.03	0.85	1.25e-07	1
10	61.95	21.65	17.58	0.86	1.25e-07	1

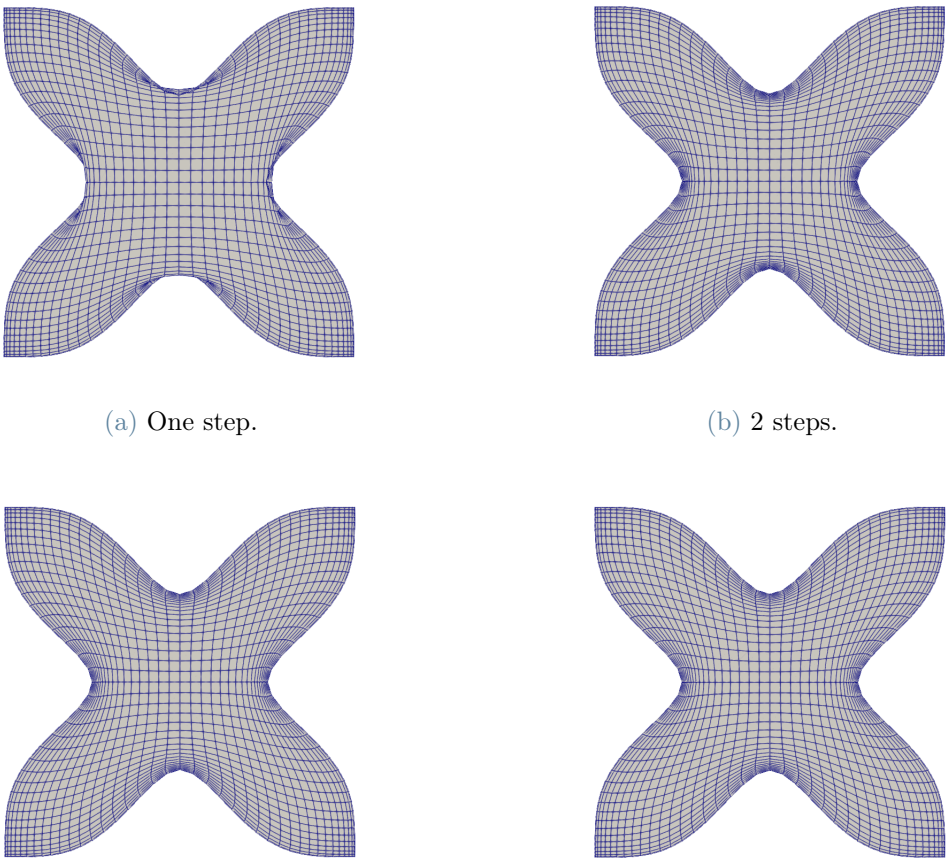
Table 3.2: Mesh quality in outward deformation with different number of steps.

is positive and all the cell face validity are passed which indicate that all the deformed

meshes remain valid. It is to be noticed that the skewness increases with the increase in number of steps but the extent of increase is not considerable with respect to the betterment in the non-orthogonality and a trade-off benefits the high number of steps.

3.3.2. Inward deformation

For an inward deformation, all the 6 faces of the cube are pushed by a length equal to 25% of the edge length of the cube with respect to 15% in the previous section to better show the impact of the multi-step approach. The simulations are done for 4 different number of steps: 1, 2, 5, and 10. The results for inward deformation for these 4 cases are shown in Fig. 3.10. Unlike the outward deformations in Fig. 3.9, the effect of using a multi-step approach is clearly visible in Fig. 3.10. For such a high inward deformation, a single step approach as shown in Fig. 3.10a fails to provide a valid mesh. It can be seen that the nodes at and near the boundary intersect with the nodes in the bulk mesh. For steps 2-10



(a) One step.

(b) 2 steps.

(c) 5 steps.

(d) 10 steps.

Figure 3.10: Inward deformation using the multi-step approach as shown in Sec. 2.10.

in Fig. 3.10, it can be observed that the congestion of the mesh near the boundary decreases with increase in number of steps, which indicates that the mesh near the boundary is preserving it's shape more for more number of steps. To look into further details, the mesh quality parameters are listed in Table 3.3 for a comparison between all the 4 cases.

Steps	**Max. non-orthogonality (deg.)**	**Max. boundary non-orthogonality (deg.)**	**Avg. non-orthogonality (deg.)**	**Max. skewness**	**Min. cell volume**	**Cell face validity**
1	180.0	3.99	13.25	6.7	-5.10e-05	0
2	37.98	4.83	12.57	0.73	1.25e-07	1
5	39.61	5.61	13.49	0.73	1.25e-07	1
10	40.3	5.99	13.82	0.73	1.25e-07	1

Table 3.3: Mesh quality in inward deformation with different number of steps.

For a single step approach, it is clear from Table 3.3 that the maximum mesh non-orthogonality of 180°, negative mesh volumes and a 0 cell face validity exist, which make the overall mesh invalid. For number of steps $2 - 10$ it is observed that the maximum and average non-orthogonality and maximum skewness increase slightly with the increase in number of steps. The reason behind this behaviour can be explained using Fig. 3.10 where it can be seen that the increase in number of steps decreases the congestion near the boundaries. Due to decrease in congestion near the boundaries, the mesh overall moves towards the center of the cube and because of the finite volume inside the cube, the average angle between the normal of a face and the vector connecting the two neighbour cell centers increases. The similar happens at the boundary, where the cells are more orthogonal when they are smaller and hence closer to each other but they become less orthogonal when the congestion reduces and cells try to preserve their original shape and hence farther from each other.

Overall, it can be said based on observations that a multi-step approach gives better results than a single-step approach in terms of mesh quality. Although the optimum number of steps can vary application to application. For applications with outward like deformations, clearly more number of steps help in providing better quality morphed mesh, but for inward like deformations, more number of steps helps in preserving the shape of the mesh close to the boundary but it hardly helps in enhancing the mesh quality parameters studied in the present work.

3.4. Conclusions

Based on the results presented in the present chapter, following conclusions can be listed:

- The implementation of the proposed corrections in the second order least-square method to include the contribution of boundary faces as described in Sec. 2.6 improves the quality of the cell and face gradients considerably.
- The linear elasticity analogy is able to morph the volumetric mesh for different kinds of deformations of opposite nature for a non-uniform cube of 64k cells.
- The linear elasticity analogy when applied using a multi-step approach produces considerably better results than a single step approach for all kinds of deformations considered in the present work. The mesh at the boundary can be more preserved when a multi-step approach is used.
- In general, the optimum number of steps in the multi-step approach is a trade off between the computational complexity and mesh quality. Although, it is found that more number of steps necessarily does not mean better mesh quality but depends on the kind of deformation as well.

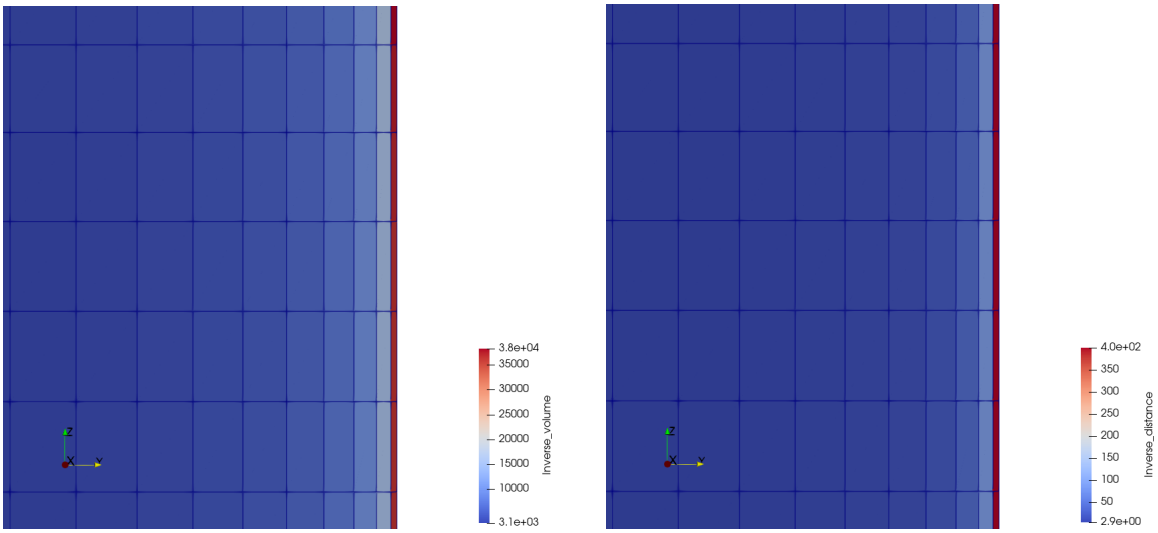
4 | Parametric study

One of the objective of the present work is to investigate the proposed relations for the elasticity coefficient E and Poisson's ration ν as described in Sec. 2.1. This chapter is dedicated to investigate these relations and present a parametric study for the parameters present in the proposed relations. As described in Sec. 2.1, the present parametric study is divided in two sections; namely for elasticity coefficient E and for Poisson's ratio ν. In both these sections, a non-uniform cube is simulated for both the outward and inward deformations as described in Sec. 3.3 of the previous chapter. A single-step approach is used to conduct all the simulations in the present chapter.

4.1. Elasticity coefficient

For the elasticity coefficient E, there are mainly 2 relations studied in the present work as discussed in Sec: 2.1; namely, inverse of volume and inverse of distance relations with exponents p and q respectively. In present work, for both the proposed relations, different simulations are done for these exponents p and q starting from 0 to 5. For the computation of Lamé constants, Eq. 2.5 which considers the rigid body motion as well is used , and therefore, the Poisson's ratio ν does not play a role in this section.

The distribution of the elasticity coefficient E over the cells near the boundary of the non-uniform cube for exponent $p = 1$ for the inverse of volume method and $q = 1$ for the inverse of distance method is shown in Fig. 4.1. It can be seen in Fig. 4.1a that that due to the small volumes, the cells near the boundary have a much higher elasticity coefficient with respect to the cells far from the boundary. It can be observed that the elasticity coefficient decreases gradually while moving away from the boundary. Similarly, in Fig. 4.1b, it can be observed that due to small distance from the nearest boundary, the cells near the boundary have a much higher elasticity coefficient with respect to the cells far from the boundary. With increasing distance from the boundary, the elasticity coefficient gradually decreases while moving away from the boundary. At the boundary, an increase in exponent p or q increases the elasticity coefficient further with respect to the cells far from boundary.



(a) Inverse of Volume relation ($p = 1$). (b) Inverse of Distance relation ($q = 1$).

Figure 4.1: Distribution of elasticity coefficient E near the boundary of non-uniform cube.

It is to be noticed in the present work, that the difference between the elasticity coefficient between the boundary and the bulk mesh plays the key role in defining the stiffness of the cells. It can be observed from Fig. 4.1a that with respect to the boundary cells, the elasticity coefficient drops almost by 50% in the cells adjacent to the boundary cells, whereas in Fig. 4.1b, this drop in elasticity coefficient is almost by 75%. Similarly, it can be observed in the range of elasticity coefficient in both of Figs. 4.1a and 4.1a that the ratio of maximum limit to the minimum limit in elasticity coefficient is almost 10 in the earlier whereas more than 100 in the later. This comparison overall shows that the difference in the elasticity coefficient E between the boundary and bulk mesh is much more in inverse of distance method than in inverse of volume method. This difference increases further by the increase of the exponents p and q.

4.1.1. Outward mesh deformation

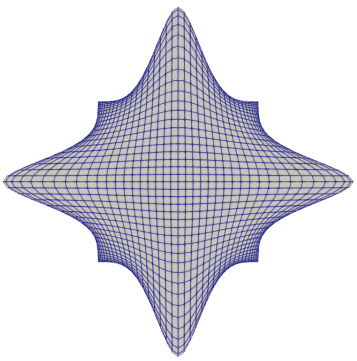
In this subsection, the results for outward deformation on the cube are shown with E computed using an inverse of the volume of the cell and inverse of distance of the cell center from the nearest boundary surface with exponents p and q respectively.

Inverse of volume method

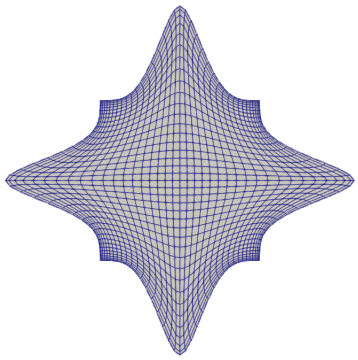
To demonstrate the results with the use of inverse of cell volume method, 6 different simulations have been carried out with the exponent p from $0 - 6$. The cross sections of the deformed cubes for exponent $p = 0 - 5$ passing through the center of the cube are

shown in Fig. 4.2. For exponent $p = 0$, the elasticity coefficient E becomes a constant throughout the domain irrespective of the volume of the cell. The exponent p starting from 1 to 5 shows that the lesser the volume, the more the elasticity coefficient and the more the stiffness. From Fig. 4.2a− 4.2f, it can be observed that the deformed meshes look quite different. To understand the quality of the mesh better, different mesh quality parameters are listed in Table 4.1 for exponents $0 - 5$.

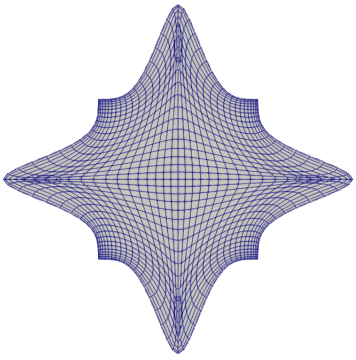
It can be observed from Table 4.1 that the maximum and average non-orthogonality decreases significantly by increasing the exponent from 0 to 1. In addition, a drastic improvement in the maximum non-orthogonality at the boundary is found which plays an important role for the mesh morphing applications. On the other hand, an increase in maximum skewness is observed. An increase in exponent p from 1 to 2 shows a valid mesh with positive volumes and passed cell face validity. It can be observed from Table 4.1, that the maximum and average non-orthogonality increases but the maximum non-orthogonality at the boundaries decreases drastically. The maximum skewness increase a little by changing exponent p from 1 to 2. The change in the behaviour of



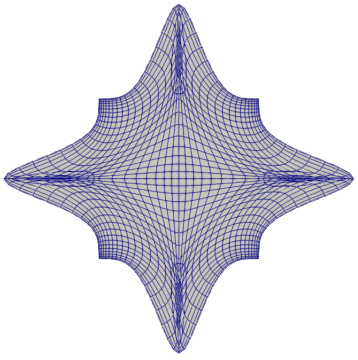
(a) p = 0.



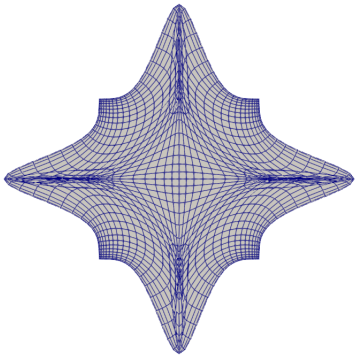
(b) p = 1.



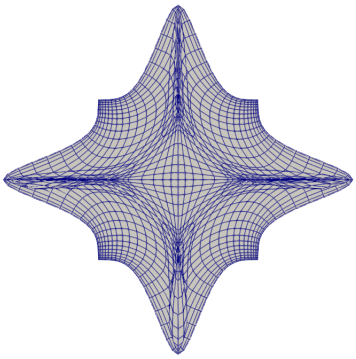
(c) p = 2.



(d) p = 3.



(e) p = 4.



(f) p = 5.

Figure 4.2: Effect of inverse of volume exponent p on the Outward deformation.

Exponent p	**Max. non-orthogonality (deg.)**	**Max. boundary non-orthogonality (deg.)**	**Avg. non-orthogonality (deg.)**	**Max. skewness**	**Min. cell volume**	**Cell face validity**
0	69.92	70.39	24.41	0.58	1.25e-07	1
1	61.90	24.37	20.69	0.80	1.25e-07	1
2	71.87	2.17	21.93	0.86	1.25e-07	1
3	180.0	2.33	24.39	6.31	-8.46e-06	0
4	180.0	1.81	27.14	14.47	-8.2e-05	0
5	180.0	1.41	29.4	108.32	-2.4e-04	0

Table 4.1: Mesh quality in outward deformation with different number of steps.

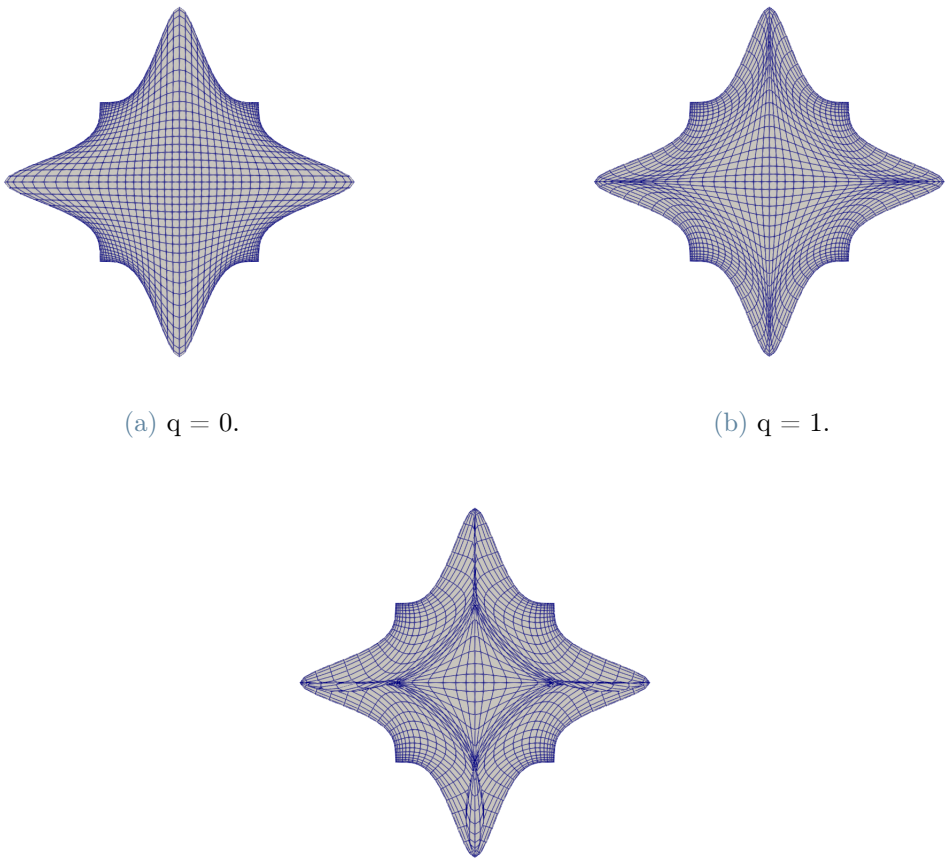
mesh by changing the exponent from 1 to 2 can be observed in Fig. 4.2b and 4.2c that the mesh at the boundary becomes much more orthogonal but gets compressed in the bulk resulting in more non-orthogonality in the bulk mesh.

A further increase in the exponent p from 2 to 3 results in an invalid mesh as the minimum cell volume becomes negative and the cell face validity fails. The maximum non-orthogonality becomes 180° which is evident from Fig. 4.2d. Increasing the exponent p further to 4 and 5 results in even worse mesh quality as can be clearly seen in Figs. 4.2e and 4.2f. It can be noticed from Table 4.1 that the minimum negative cell volume increases and the skewness increases unacceptably as well.

Therefore, based on the results, it can be said the exponent of the inverse of volume from 0 to 2 result in valid meshes but from $3-5$ result in invalid meshes. It is observed, that for a generic application, exponent $p = 1$ can be a good candidate but exponent $p = 2$ shows much better results at the boundaries. It is to be noticed that in case of exponent $p = 2$, the maximum non-orthogonality is found to be $> 70°$ which is not recommended as a good mesh characteristic and the user must be careful while using this exponent. Therefore, for mesh morphing applications, exponents 1 and 2 can be used based on the application.

Inverse of distance method

For the inverse of distance relation for the elasticity coefficient E, 3 different cases with the exponent q changing from 0 to 2 are simulated. The planes passing through the centers of the deformed cubes are shown in Fig. 4.3 for all the 3 cases. For exponent $q = 0$, the



(a) q = 0.

(b) q = 1.

(c) q = 2.

Figure 4.3: Effect of inverse of distance exponent q on the Outward deformation.

elasticity coefficient E becomes a constant throughout the cube mesh irrespective of the distance from the nearest boundary from the cell center. The increase in exponent q shows the increase in elasticity coefficient in the cells close to the boundary and therefore more stiffness for the cells close to the boundary. It can be observed in Fig. 4.3 that the mesh changes drastically by increasing the exponent q from 0 to 2. Various mesh quality parameters considered in the present work are listed in Table 4.2 for exponent q from $0 - 2$. It can be observed in Table 4.2, that by increasing the exponent q from 0 to 1, the maximum non-orthogonality increases by almost 5° and becomes $> 70°$ which is not good but the maximum boundary non-orthogonality reduces by over 65° which is a big improvement for the mesh near the boundary. This decrease in the maximum non-orthogonality

Exponent q	**Max. non-orthogonality (deg.)**	**Max. boundary non-orthogonality (deg.)**	**Avg. non-orthogonality (deg.)**	**Max. skewness**	**Min. cell volume**	**Cell face validity**
0	69.92	70.39	24.41	0.58	1.25e-07	1
1	74.99	4.43	23.48	0.85	1.25e-07	1
2	180.0	1.46	31.17	106.43	-1.93e-04	0

Table 4.2: Mesh quality in outward deformation with different number of steps.

can be clearly observed in Fig. 4.3b where it can be observed that the boundary mesh becomes almost orthogonal. It can be also observed in Table 4.2 that the average non-orthogonality decreases as well but the maximum skewness increases.

For an increase in the exponent q from 1 to 2, it can be observed that the minimum cell volume becomes negative and the cell face validity fails which indicate that the mesh has become invalid for exponent $q = 2$. Fig. 4.3c clearly reflects this bad quality of the mesh and the inverted cells. The boundary becomes further more orthogonal but compresses the internal mesh so much that the maximum non-orthogonality becomes 180° in the bulk resulting in a failed mesh.

Overall, based on the results, it can be said that the relation of E as the inverse of the distance between the centroid of the cell and the nearest boundary point works well only for the case with an exponent $q = 1$. Because of the existence of maximum non-orthogonality $> 70°$, the user must be careful while using this relation. In present work,

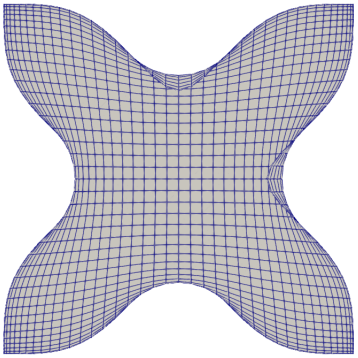
the simulations with $q > 2$ were carried out as well but the quality of the deformed mesh becomes unacceptably worse and therefore it is needless to show them here.

4.1.2. Inward mesh deformation

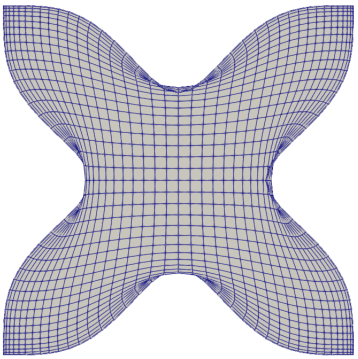
In this subsection, the results for inward mesh deformation for a non-uniform cube with E computed using the inverse of volume and inverse of distance relations are presented.

Inverse of volume method

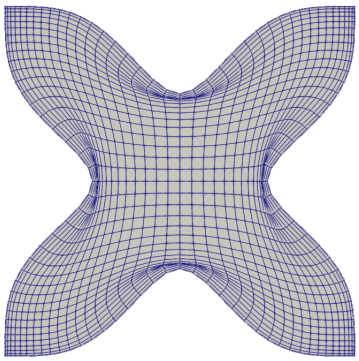
Just like the outward deformation, 6 different simulations have been carried out with the exponent p from $0-5$. For these 6 cases, the planes passing through the center of the corresponding deformed cubes are shown in Fig. 4.4. For $p = 0$, it can be seen in Fig. 4.4a that the mesh stays preserved around the center of the deformed cube but dramatically fails near the boundary because of the same stiffness of all the cell irrespective to their cell volumes. For an exponent $p = 1$, it can be seen in Fig. 4.4b, that the mesh quality near the boundary improves but the cells intersect and result in a failed mesh overall. A further increase in exponent $p > 1$ shows further improves the quality of the mesh at the boundary but contracts the mesh in the bulk as well. To understand the results better, the various mesh quality parameters are listed in Table 4.3 for exponent p from $0-5$. By increasing the exponent p from 0 to 1, it can be observed that the average non-orthogonality, maximum boundary non-orthogonality and skewness all decrease drastically, but due to negative mesh volume, 180° maximum non-orthogonality and failed cell face validity, the overall mesh fails. An increase in the exponent p from 1 to 2 decreases the maximum skewness to a practical value and decrease the average non-orthogonality with positive minimum cell volume and passed cell face validity, but the maximum non-orthogonality comes out to be 180 which is unacceptable.



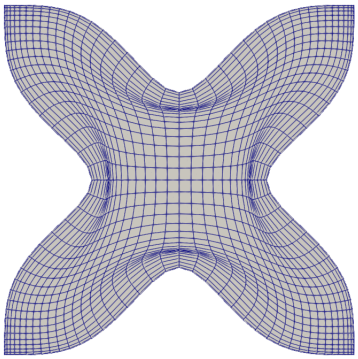
(a) p = 0.



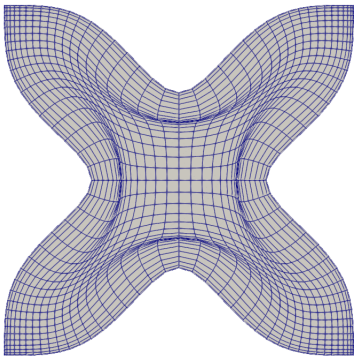
(b) p = 1.



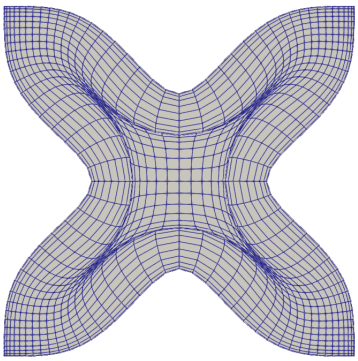
(c) p = 2.



(d) p = 3.



(e) p = 4.



(f) p = 5.

Figure 4.4: Effect of inverse of volume exponent p on the Inward deformation.

Exponent p	**Max. non-orthogonality (deg.)**	**Max. boundary non-orthogonality (deg.)**	**Avg. non-orthogonality (deg.)**	**Max. skewness**	**Min. cell volume**	**Cell face validity**
0	180.0	175.65	20.03	70.96	-9.38e-05	0
1	180.0	3.99	13.25	6.69	-5.1e-05	0
2	180.0	4.73	12.33	0.73	1.25e-07	1
3	53.46	4.88	14.06	0.73	1.25e-07	1
4	69.62	4.79	16.43	0.73	1.25e-07	1
5	180.0	4.73	19.57	18.60	-5.07e-05	0

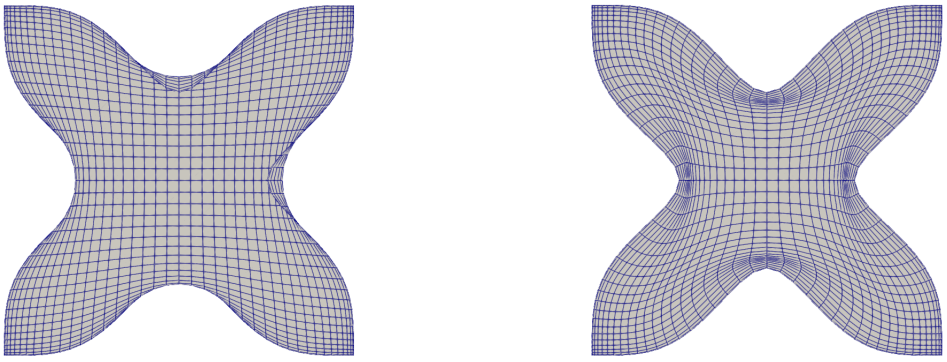
Table 4.3: Mesh quality in outward deformation with different number of steps.

A further increase in the exponent p to 3 makes the mesh valid for all the quality parameters as shown in Table 4.3. It can be observed in Fig. 4.4d that the mesh near the boundary preserve their shape very well but results in bit more non-orthogonality in the bulk mesh. An increase in exponent p from 3 to 4 makes the mesh near the boundary even more preserved, but as the internal mesh gets compressed even more, the maximum and average non-orthogonality increase as well. For an exponent $p = 5$, the internal mesh gets compressed so much that it starts to overlap and becomes invalid as shown in Fig. 4.4f. It can be observed that there exists negative minimum volume, 180° maximum non-orthogonality, an unacceptable skewness of 18.6039, and a failed cell face validity.

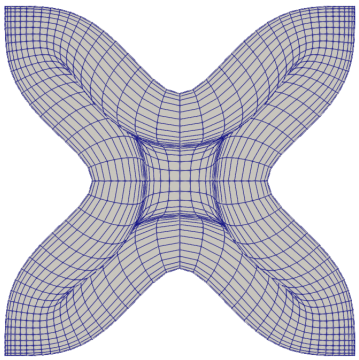
Therefore, for the present inward deformation case, the relation of elasticity coefficient with the inverse of volume comes out such as the exponents $0 - 2$ result in a failed mesh because of the boundary mesh being highly skewed and maximum non-orthogonality of 180° near the boundary whereas the exponent $p = 5$ fails because of crushing of the internal cells far from the boundary in order to preserve the mesh near the boundary. The exponents $p = 3, 4$ work well in preserving the mesh at the boundary and keeping the internal mesh valid.

Inverse of distance method

Like in previous Sec. 4.1.1, 3 different cases with 3 different exponents q are considered to discuss the effect of inverse of distance relation. For these 3 cases with exponent $q = 0 - 2$, the planes passing through the center of the deformed cube are demonstrated as shown in Fig. 4.5. Just like the case with $p = 0$ in the inverse of volume method, all cells have the a constant and same elasticity coefficient for $q = 0$ case and therefore the result is the same as well for this case. For $q = 1$, the elasticity coefficient becomes large near the boundaries providing the cells nearby with high stiffness. This high stiffness helps in



(a) q = 0. (b) q = 1.



(c) q = 2.

Figure 4.5: Effect of inverse of distance exponent q on the Inward deformation.

Exponent q	**Max. non-orthogonality (deg.)**	**Max. boundary non-orthogonality (deg.)**	**Avg. non-orthogonality (deg.)**	**Max. skewness**	**Min. cell volume**	**Cell face validity**
0	180.0	175.65	20.03	70.96	-9.38e-05	1
1	51.17	4.54	13.7	0.73	1.25e-07	1
2	180.0	4.74	24.08	90.80	-2.1e-04	0

Table 4.4: Mesh quality in outward deformation with different number of steps.

preserving the mesh near the boundary as shown in Fig. 4.5b. The table containing the mesh quality parameters for exponents $q = 0 - 2$ is shown in Table 4.4. A drastic decrease in all the mesh quality parameters is observed by increasing exponent q from 0 to 1 which makes the mesh valid for $q = 1$ case. An increase in exponent q from 1 to 2 improves the mesh quality near the boundaries a lot and the mesh near the boundary almost preserves their initial shape as shown in Fig. 4.5c, but the internal mesh gets crushed and results in 180° of maximum non-orthogonality, negative minimum cell volume and a failed cell face validity, which indicates that the mesh fails for $q = 2$ case.

Therefore, based on the results, it can be said for the present inward deformation, that for the inverse of distance relation, the exponent $q = 1$ works well at both the boundary and the internal mesh, but $q = 0$ results in failed mesh at the boundaries and good mesh far from boundary, whereas $q = 2$ results in good mesh near the boundary and failed mesh

far from the boundary.

To sum up this section, it can be said based on the results that for the inverse of distance relation, only the exponent $q = 1$ provides us with a valid deformed mesh for the given given outward and inward deformations. For the inverse of volume method, the exponents $p = 3, 4$ work best for the given outward deformation, whereas exponents $p = 0 - 2$ work for the given inward deformation. Based on the results, it can be said for the inverse of volume relation that the optimum exponent depends on the kind and amount of deformation provided to the initial mesh. In a general case, the user can use the exponents $p = 1 - 4$ for good quality deformed mesh.

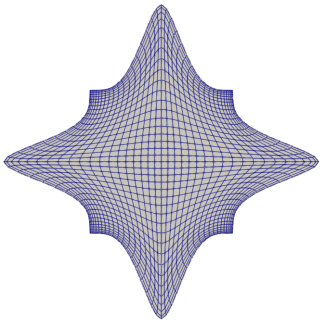
4.2. Poisson's ratio

In the basic linear elasticity model, the Lamé coefficients are computed using the equation derived from the linear elasticity law as shown in Eq. 2.3 which uses both the elasticity coefficient E and the Poisson's ratio ν. In the proposed relations for E and ν in the present work as described in Sec. 2.1, the Poisson's ratio ν is kept at a same and constant value throughout the mesh. The present section is dedicated to find out the optimum Poisson's ratio ν for both the outward and inward deformation cases for a non-uniform cube. As per Eq. 2.3, ν can be kept between -1.0 and 0.5. In present work, 7 different case studies are simulated with 7 different Poisson's ratios ν at a difference of almost 0.25 starting from -1.0.

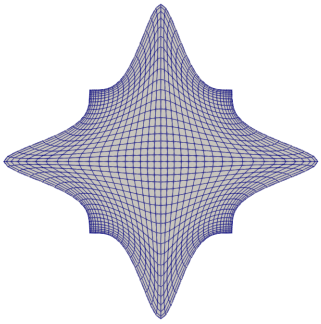
4.2.1. Outward mesh deformation

The non-uniform cube under consideration is undergone outward deformation for all the faces as described in the previous section. In present subsection, the elasticity coefficient relation inversely proportional to the volume of the cell with an exponent $p = 1$ is considered. The results with an outward deformation for the range of ν between $(-1.0, 0.5)$ are shown in Fig. 4.6. For the first and final simulation $\nu = -1.0$ and 0.5 would give a undefined solution and therefore a value of -0.99 and 0.49 are simulated. It is found out that the solution does not converge for 0.49 and an upper limit of $\nu = 0.30$ is obtained. From Fig. 4.6, it can be observed that the results do not change a lot. The only noticeable detail seems that the orthogonality at the boundary increases from Fig. 4.6a to 4.6g but the internal mesh gets compressed as well. To understand better the change in mesh, a table with various mesh quality parameters is shown in Table 4.5.

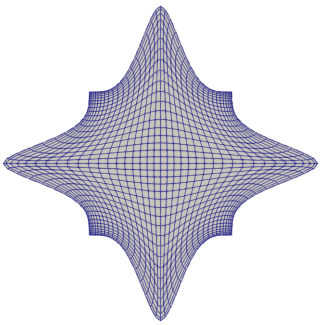
It can be observed in Table 4.5, that with the increase in Poisson's ratio ν, the maximum



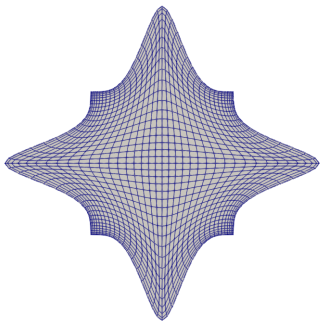
(a) $\nu = -0.99$.



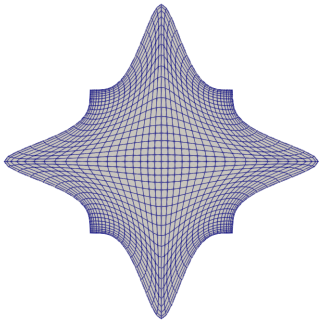
(b) $\nu = -0.75$.



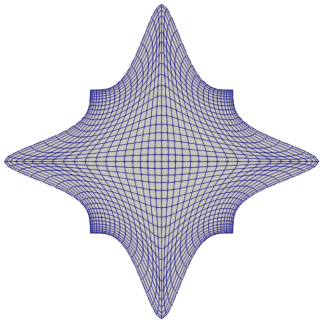
(c) $\nu = -0.5$.



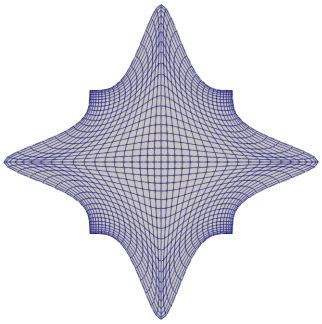
(d) $\nu = -0.25$.



(e) $\nu = 0.0$.



(f) $\nu = 0.25$.



(g) $\nu = 0.35$.

Figure 4.6: Effect of Poisson's ratio on the outward deformation.

ν	**Max. non-orthogonality (deg.)**	**Max. boundary non-orthogonality (deg.)**	**Avg. non-orthogonality (deg.)**	**Max. skewness**	**Min. cell volume**	**Cell face validity**
-0.99	62.55	22.18	20.98	0.81	1.25e-07	1
-0.75	62.74	21.84	21.03	0.81	1.25e-07	1
-0.5	63.08	21.35	21.1	0.81	1.25e-07	1
-0.25	63.66	20.62	21.2	0.82	1.25e-07	1
0	64.54	19.43	21.37	0.82	1.25e-07	1
0.25	66.13	17.11	21.76	0.83	1.25e-07	1
0.3	66.93	16.34	21.9	0.83	1.25e-07	1

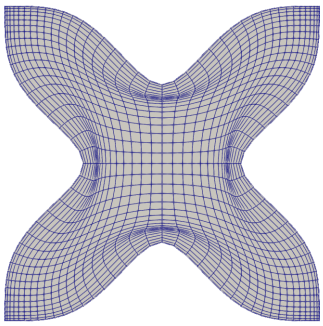
Table 4.5: Mesh quality in outward deformation with different number of steps.

non-orthogonality decreases at the boundary which is good for the mesh morphing applications, whereas the maximum non-orthogonality in the bulk mesh increases. The reason for maximum non-orthogonality to increase is again that to preserved the boundary mesh, the internal mesh gets compressed and suffers from more non-orthogonality. It is observed that the average non-orthogonality and the maximum skewness increase but not considerably. The minimum cell volume stays positive and the cell face validity passes which indicates that the mesh does not fail in any case.

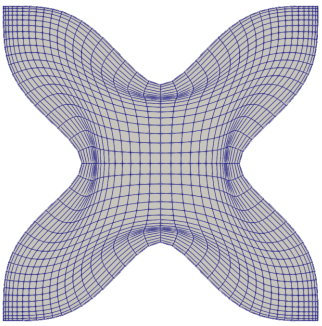
Overall, for better mesh quality at the boundaries which is needed in the mesh morphing applications, a positive Poisson's ratio ν can be used. This does suffers from bad mesh quality in the bulk mesh but a multi-step approach can help in optimizing the mesh quality further.

4.2.2. Inward mesh deformation

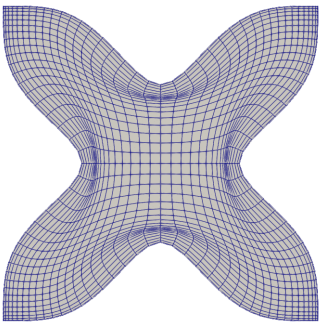
In this subsection, the non-uniform cube is undergone an inward deformation at all the faces. The elasticity coefficient is considered to be computed from the inverse of cell volume approach with an exponent $p = 3$. The results for an inward deformation for the range of ν between $(-1.0, 0.5)$ are shown in Fig. 4.7. For an inward deformation, the maximum ν for which a converged solution is obtained is found out to be 0.35. Therefore, 6 simulations for ν from $-0.99 - 0.25$ and one for 0.35 are conducted in the present study. It can be observed from Fig. 4.7a to 4.7g that the cells between the boundary and the



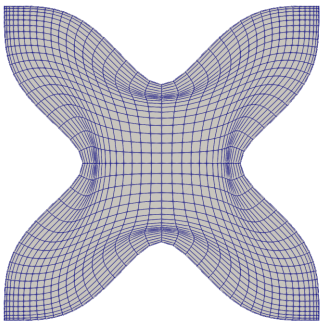
(a) $\nu = -0.99$.



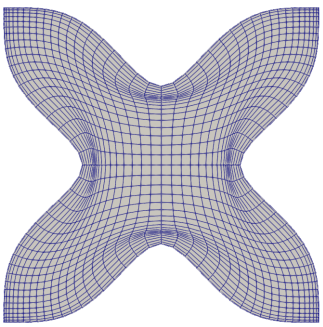
(b) $\nu = -0.75$.



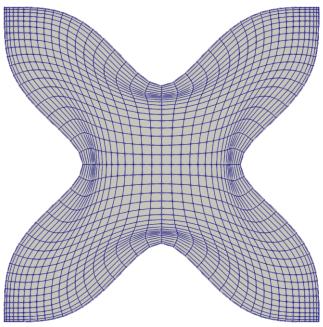
(c) $\nu = -0.5$.



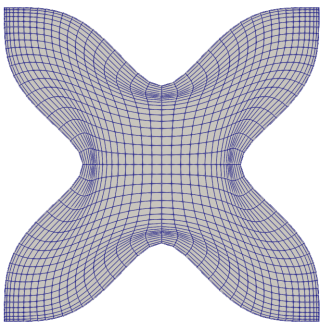
(d) $\nu = -0.25$.



(e) $\nu = 0.0$.



(f) $\nu = 0.25$.



(g) $\nu = 0.35$.

Figure 4.7: Effect of Poisson's ratio on the inward deformation.

center of the cube become less compressed. The details of various mesh quality parameters for all the ν are listed in Table 4.6.

For all the 7 cases with different ν, the minimum cell volume stays positive and cell face validity passes which indicates that all the deformed meshes are valid. It can observed in Table 4.6 that the maximum non-orthogonality in the mesh and the maximum non-orthogonality at the boundary decrease with increase in the ν whereas the average non-orthogonality and maximum skewness do not change considerably.

The results overall show that the mesh quality improves from negative to positive Poisson's

ν	**Max. non-orthogonality (deg.)**	**Max. boundary non-orthogonality (deg.)**	**Avg. non-orthogonality (deg.)**	**Max. skewness**	**Min. cell volume**	**Cell face validity**
-0.99	51.69	4.4	14.01	0.73	1.25e-07	1
-0.75	51.47	4.33	14.02	0.73	1.25e-07	1
-0.5	51.17	4.23	14.03	0.73	1.25e-07	1
-0.25	50.75	4.1	14.05	0.73	1.25e-07	1
0	50.36	3.89	14.12	0.73	1.25e-07	1
0.25	49.93	3.53	14.31	0.73	1.25e-07	1
0.35	49.86	3.42	14.39	0.73	1.25e-07	1

Table 4.6: Mesh quality in outward deformation with different number of steps.

ratio ν. Therefore, it can be said in general that a positive Poisson's ratio ($\nu > 0$) shows the optimum results from the point of view of good mesh quality.

Based on both the above subsections, it can be said that a positive Poisson's ratio ν gives better results than a negative one and is recommended to be used.

4.3. Conclusions

In present chapter, 2 different kinds of parametric studies are conducted; One on the Elasticity coefficient E, and the other on the Poisson's ratio ν. In the first kind of parametric study on elasticity coefficient E, 2 different relations to compute the elasticity coefficient are considered with two different parameters p and q respectively. In the second kind of parametric study, an optimum Poisson's ratio ν is found out by simulating

the linear elasticity solver for different ν values. An outward and an inward kinds of deformations are considered for the simulations with a single step approach. A summary of the conclusions based on the parametric study conducted in the present chapter can be listed as follows:

Elasticity coefficient E

- For any kind of deformation considered in this chapter, increasing the parameter p in the inverse volume relation or q in the inverse distance relation helps in preserving the mesh near the boundary but compresses the internal mesh far from the boundary resulting in clustering of cells.
- For an outward deformation, parameter $p = 0, 1, 2$ provides valid meshes in increasing order of mesh quality and parameter $q = 0, 1$ provides valid meshes with increasing order of mesh quality. When comparing the mesh quality tables, it is found that the inverse volume method with $p = 1, 2$ provides better results than inverse distance relation with $q = 1$.
- For an inward deformation, a valid mesh is generated using $p = 3, 4$ with decreasing mesh quality and $q = 1$. The deformed mesh fails for $p = 0, 1, 2$ and $q = 0$ because of clustering and inversion of cells near the boundary and fails at $p = 5$ and $q >= 2$ due to clustering and inversion of cells far from the boundary. When comparing the mesh quality tables, $p = 3$ and $q = 1$ provide comparable mesh quality.

It is to be noted that the present parametric study is conducted over high deformations. There might exist small deformations where all the relations are able to provide with a valid mesh. The optimum choice of the parameters p and q depends on the kind of deformation provided as well.

Poisson's ratio ν

- Despite the possible upper limit of the Poisson's ratio ν being 0.5, a converged solution is obtained only until a maximum of 0.3 for both the outward and the inward kinds of deformations.
- For both kinds of deformations, a Poisson's ratio $\nu > 0$ generates the mesh with optimum mesh quality.

5 | Comparisons with Laplacian and Laplacian quaternions techniques

There exist various techniques to morph a given volumetric unstructured mesh as discussed in Chapter 1. All the techniques have some pros and cons, and therefore, the user is required to choose the appropriate method according to the application and type of deformation. In present chapter, the results of mesh morphing using the linear elasticity technique developed in the present work are compared with the results obtained using other promising mesh morphing techniques as follows:

- Laplacian technique
- Laplacian quaternions technique
- Laplacian technique with narrow band width
- Laplacian quaternions technique with narrow band width

The first two of the aforementioned techniques are explained in Chapter 1. The concept of narrow band width is such that the mesh until a distance from the boundary is forced to be undeformed in order to preserve it. In present chapter, the elasticity coefficient is computed using the relation inversely proportional to the cell volume with an exponent $p = 1$. The Lamé constants are computed using the relation shown in Eq. 2.5. The present chapter is divided mainly in two sections with demonstrating the comparisons for single and multi-step approaches. Both of these sections contain the results for outward and inward kinds of deformations.

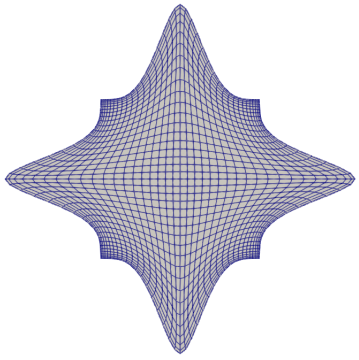
5.1. Comparisons for a single-step approach

For a single-step approach, the outward and inward deformations are used as boundary conditions as used in Sec. 4.1 of the previous chapter.

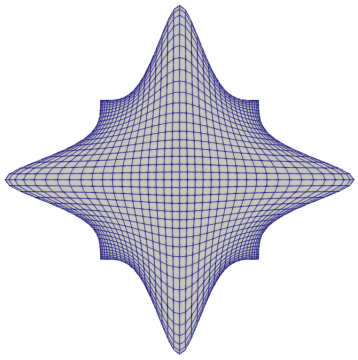
5.1.1. Outward mesh deformation

For an outward deformation, the planes of the non-uniform deformed cubes passing through the center and parallel to one of the initial faces are shown in Fig. 5.1 for the linear elasticity technique and the aforementioned techniques. It is observed for the techniques other than the linear elasticity solver that the mesh is hardly orthogonal at the boundaries of the deformed cube. For the Laplacian quaternions technique without and with the narrow band width, it can be observed in Figs. 5.2c and 5.2e that the mesh get accumulated near the boundary but does not deform far from the boundary. For better description of the mesh quality, the various mesh quality parameters are listed in Table 5.1.

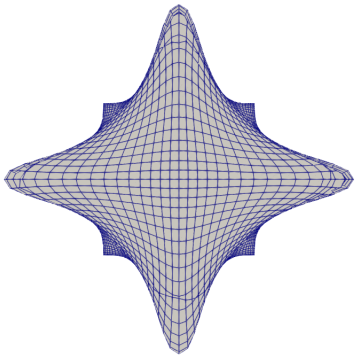
It is observed in Table 5.1, that for a single-step approach, with 180° of maximum non-orthogonality, large maximum non-orthogonality at the boundary, unacceptably large maximum skewness, negative minimum cell volume, and failed cell face validity, Laplacian



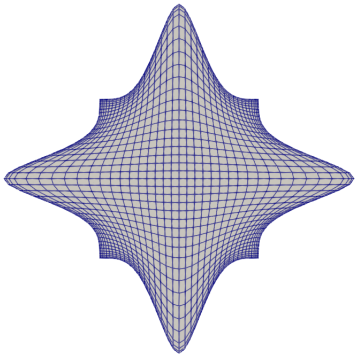
(a) Linear elasticity analogy.



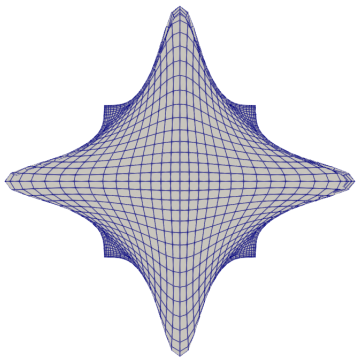
(b) Laplacian technique.



(c) Laplacian quaternions technique.



(d) Laplacian technique with Narrow band width.



(e) Laplacian quaternions technique with Narrow band width.

Figure 5.1: Outward deformation in the cube with a single step approach.

Technique	**Max. non-orthogonality (deg.)**	**Max. boundary non-orthogonality (deg.)**	**Avg. non-orthogonality (deg.)**	**Max. skewness**	**Min. cell volume**	**Cell face validity**
(a)	61.90	24.37	20.69	0.80	1.25e-07	1
(b)	67.97	68.57	24.95	0.77	1.25e-07	1
(c)	180.0	144.06	41.94	29.67	-4.3e-04	0
(d)	67.44	68.24	25.48	0.83	1.25e-07	1
(e)	180.0	84.92	38.28	83.36	-4.7e-04	0

Table 5.1: Mesh quality in outward deformation with a single step approach.

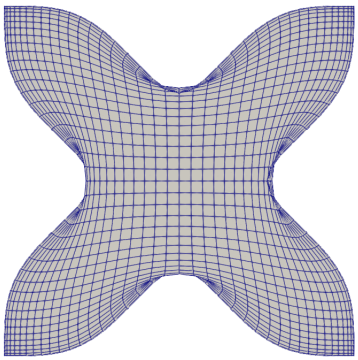
quaternion approach fails to to deliver a valid mesh with or without the narrow band width. On the other hand, Laplacian technique with and without the narrow band width delivers a valid mesh. In comparison to the linear elasticity technique, the Laplacian technique results in large non-orthogonality not only at the boundary but throughout the mesh as anticipated from Fig. 5.1.

Therefore, for an outward deformation with a single-step, it is found out that that the developed linear elasticity technique generates the best quality mesh among the techniques aforementioned.

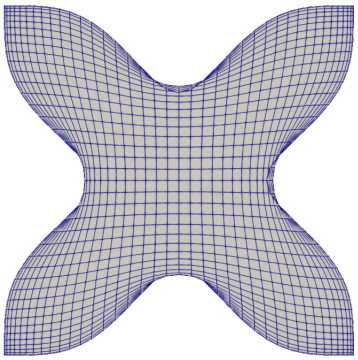
5.1.2. Inward mesh deformation

For an inward deformation on the non-uniform cube, the planes passing through the center of the cube and parallel to the original surface of the cube are shown in 5.2. For all the Figs. 5.2a- 5.2e, it can be noticed that all meshes show very bad quality at the boundary of the cube and result in overlapped and negative volume cells. The various mesh quality parameters are listed in Table 5.2 for the meshes shown in Fig. 5.2.

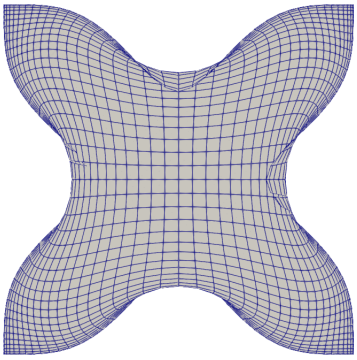
It can be observed for all the meshes that the maximum non-orthogonality stays unacceptable 180° with high maximum boundary non-orthogonality and negative minimum cell volumes, which make these meshes invalid. Though, it can be noticed among these meshes that the average non-orthogonality is lowest in case of linear elasticity technique than all the other techniques. Similarly, the maximum skewness turns out to be minimum using the linear elasticity technique.



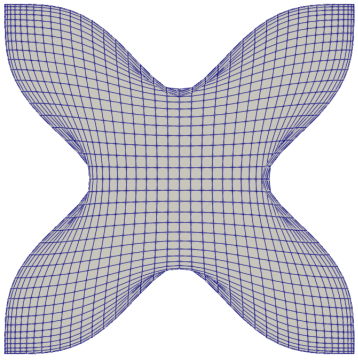
(a) Linear elasticity analogy.



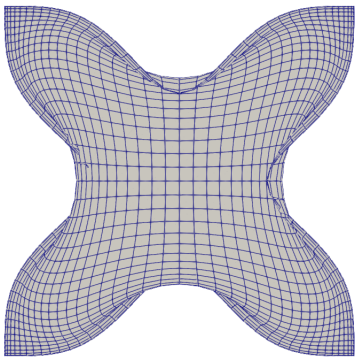
(b) Laplacian technique.



(c) Laplacian quaternions technique.



(d) Laplacian technique with Narrow band width.



(e) Laplacian quaternions technique with Narrow band width.

Figure 5.2: Inward deformation in the cube with a single step approach.

Technique	**Max. non-orthogonality (deg.)**	**Max. boundary non-orthogonality (deg.)**	**Avg. non-orthogonality (deg.)**	**Max. skewness**	**Min. cell volume**	**Cell face validity**
(a)	180.0	3.99	13.25	6.7	-5.1e-05	0
(b)	180.0	46.47	15.89	17.42	-4.6-05	0
(c)	180.0	104.16	18.02	232.34	-1.9-04	0
(d)	180.0	46.23	16.15	31.32	-7.4-05	0
(e)	180.0	3.44	16.53	111.43	-3.1e-04	0

Table 5.2: Mesh quality in inward deformation with a single step approach.

Therefore, based on the results for the given inward deformation, it can be said that the mesh fails in all the cases but the best quality among the failed meshes comes out to be for the case with the use of linear elasticity technique.

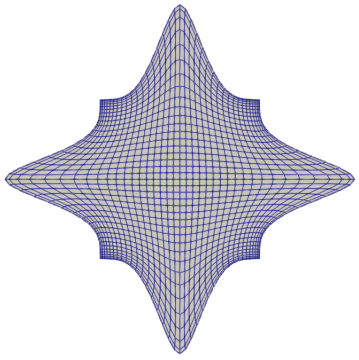
5.2. Comparisons for a multi-step approach

In this section, the simulations are done using a multi-step approach for the linear elasticity technique and all the other techniques mentioned above. A total 10 number of steps are used to simulate every case. An outward and an inward deformations are provided to the non-uniform cube for the comparisons.

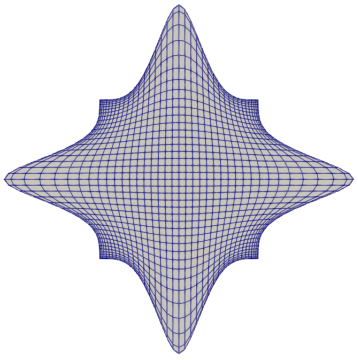
5.2.1. Outward mesh deformation

In this subsection, the non-uniform cube is provided with an outward deformation to all the faces of the cube as boundary condition. The results are shown in form of a cross section plane passing through the center of the cube as shown in Fig. 5.3. In order to support the analysis, the mesh quality parameters for all these techniques are listed in Table 5.3.

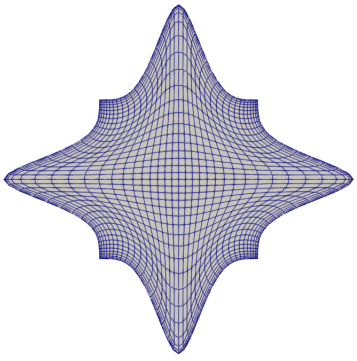
It is observed from Table 5.3 that the Laplacian technique with and without the narrow band width shows high maximum and average non-orthogonality over the bulk mesh. It is to be noticed that the non-orthogonality at the boundary for the Laplacian technique with and without the narrow width exhibit large values which are also evident from Fig. 5.3b and 5.3d. Laplacian quaternion technique, on the other hand, generates a good quality mesh with a low maximum non-orthogonality at the boundary. This non-orthogonality at the boundary for the Laplacian quaternion technique reduced further when a narrow



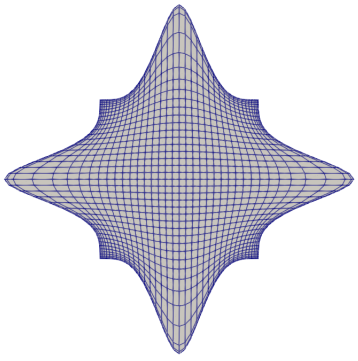
(a) Linear elasticity analogy.



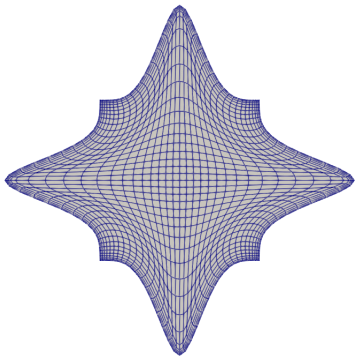
(b) Laplacian technique.



(c) Laplacian quaternions technique.



(d) Laplacian technique with Narrow band width.



(e) Laplacian quaternions technique with Narrow band width.

Figure 5.3: Outward deformation in the cube with a 10 step approach.

Technique	**Max. non-orthogonality (deg.)**	**Max. boundary non-orthogonality (deg.)**	**Avg. non-orthogonality (deg.)**	**Max. skewness**	**Min. cell volume**	**Cell face validity**
a	61.86	21.65	17.58	0.86	1.25e-07	1
b	67.55	68.38	21.17	0.76	1.25e-07	1
c	66.49	15.79	21.47	0.88	1.25e-07	1
d	68.7	68.24	21.20	0.83	1.25e-07	1
e	70.77	12.74	22.31	0.87	1.25e-07	1

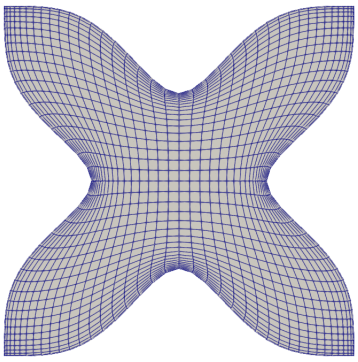
Table 5.3: Mesh quality in outward deformation with a 10 step approach.

band width is applied. For the linear elasticity solver developed in the present work, the maximum non-orthogonality obtained at the boundary is larger than in case of Laplacian quaternion with and without the narrow band width. As far as maximum and average non-orthogonality in the bulk mesh are concerned, the presently developed linear elasticity technique provides much better results. In case of Laplacian quaternion technique with narrow band width, despite very good maximum boundary non-orthogonality, the overall maximum non-orthogonality crosses 70° which is labelled as bad characteristic for a mesh. It is to be noticed that the maximum skewnwss is also lesser in case of linear elasticity techniques than the Laplacian quaternions techniques.

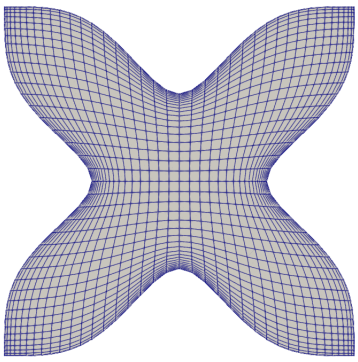
Therefore, based on the results and comparisons, it can be said that the linear elasticity solver generate a much better quality mesh compared to Laplacian techniques with and without the narrow band width, where it can be said upto some extent, that the Laplacian quaternions technique with narrow bandwidth show a mesh quality comparable with the linear elasticity model as used in the present section. Although, because of the possibility of using an exponent $p > 1$ or $q = 1$, it is possible to further enhance the quality of mesh at the boundary and the bulk for the linear elasticity solver.

5.2.2. Inward mesh deformation

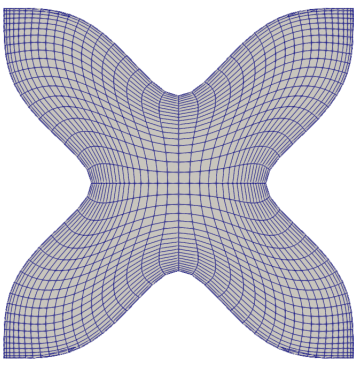
For an inward mesh deformation on a non-uniform cube, the cross-sectional view of the cube parallel to the initial surface of the cube is shown in Fig. 5.4. It is observed in the results that unlike the single-step case, all the meshes look without any overlapping and valid. Various mesh quality parameters for all the techniques using a 10 step approach are listed in Table 5.4.



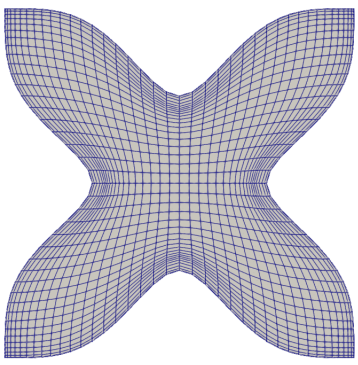
(a) Linear elasticity analogy.



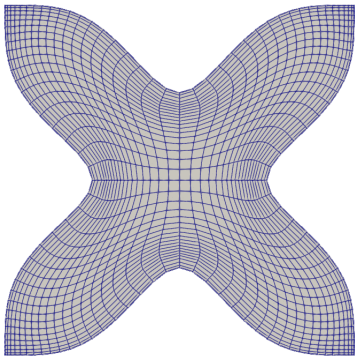
(b) Laplacian technique.



(c) Laplacian quaternions technique.



(d) Laplacian technique with Narrow band width.



(e) Laplacian quaternions technique with Narrow band width.

Figure 5.4: Inward deformation in the cube with a 10 step approach.

Technique	**Max. non-orthogonality (deg.)**	**Max. boundary non-orthogonality (deg.)**	**Avg. non-orthogonality (deg.)**	**Max. skewness**	**Min. cell volume**	**Cell face validity**
a	40.33	5.99	13.82	0.73	1.25e-07	1
b	45.91	46.66	15.11	0.74	1.25e-07	1
c	47.33	3.46	12.1	0.74	1.25e-07	1
d	46.24	46.23	16.36	0.75	1.25e-07	1
e	51.34	4.66	12.76	0.74	1.25e-07	1

Table 5.4: Mesh quality in inward deformation with a 10 step approach.

It is observed in Table 5.4 that all the techniques generate a mesh with positive minimum cell volume and passed cell face validity and hence a valid mesh. It is noticed that the maximum skewness remains almost same in all results from all the techniques. For the Laplacian technique with and without the narrow band width, it is found that the non-orthogonality at the boundary comes out to be very high with respect to the other methods. The maximum and average non-orthogonality in the bulk mesh is also found out to be the higher than rest of the techniques. For Laplacian quaternion techniques with and without the narrow band width, it is found that the maximum non-orthogonality at the boundary and the average non-orthogonality come out to be lesser than the linear elasticity technique, with the later being smaller in both the properties. On the other

hand, the maximum non-orthogonality for the linear elasticity method comes out to be considerably lesser than the Laplacian quaternion method with or without the narrow band width.

Therefore, based on the results and comparisons, it can be said that the Laplacian techniques with or without the narrow band width does not perform well when compared to the rest of the methods. For the Laplacian quaternion technique with and without the narrow band width, it can be said that it performs comparative to the linear elasticity solver as it is used. It is to be noticed that there is a possibility of an improvement in the quality of mesh near the boundary and the bulk by using $p > 1$ or $q = 1$.

5.3. Conclusions

In this section, the developed linear elasticity solver is compared with some of the promising existing mesh morphing techniques; namely, Laplacian and Laplacian quaternion techniques each with and without the narrow band width. The results of all the techniques are compared for a single as well as multi-step approaches. The results presented in the present chapter can be concluded as follows:

- For a single step approach, irrespective of the kind of deformation, the linear elasticity solver clearly provides results with much better mesh quality. The boundary mesh is preserved in a much better way in the solution of linear elasticity solver. The linear elasticity technique outperforms all the other techniques in all the mesh quality parameters.
- For a multi-step approach, the linear elasticity technique clearly outperforms the Laplacian technique for any kind of deformation with or without the narrow band width.
- For a multi-step approach, for an outward deformation, the linear elasticity technique provides better results in the bulk mesh than the Laplacian quaternion technique with or without the narrow band width, but the Laplacian quaternion techniques with and without the narrow band width provides with better results at the boundary.
- For a multi-step approach with an inward deformation, the mesh quality comes out to be comparable for the linear elasticity technique and the Laplacian quaternion technique with or without the narrow band width. The narrow band width improves the mesh quality for both the outward and inward deformations at the boundary but worsen the mesh quality in the bulk mesh.

6 | Industrial cases

Mesh morphing has various applications in the field of aerodynamic shape optimization, FSI, moving boundary problem etc. as discussed in Chapter 1. These wide field of applications provide the opportunity to apply the presently developed linear elasticity mesh morpher on the meshes undergoing such applications and test the quality of the output mesh. So far, the developed linear elasticity mesh morpher is tested only on a non-uniform cube with 64k cells. To test the robustness of the presently developed linear elasticity technique, two industrial cases from two different application fields are considered as follows:

- Wing of an airplane: FSI application
- Aerodynamic shape optimization in a car

6.1. Wing of an airplane

In present section, the industrial case of the wing of an airplane is considered. Unlike the non-uniform cube, where only the structured mesh with hexahedron kinds of cells was used, the wing contains a totally unstructured mesh with tetrahedron and hexahedron both kinds of cells. The mesh of the wing contains over 1.25 million cells including a symmetry plane. The mesh quality parameters for the undeformed non-uniform cube and the wing cases are listed in Table 6.1. It can be seen that the case of wing has a large

Mesh	**Max. non-orthogonality (deg.)**	**Max. boundary non-orthogonality (deg.)**	**Min. cell volume**	**Cell face validity**
Non-uniform cube	0	0	1.25e-07	1
Wing of an airplane	80.32	72.31	6.95e-11	1

Table 6.1: Mesh quality parameters for a non-uniform cube and a wing of an airplane.

maximum cell and maximum boundary non-orthogonality. Therefore, it can be said that the industrial case of a wing is good case to test the robustness of the presently developed linear elasticity mesh morpher.

For the wing of an airplane, one of the most critical application is to undergo the well known event of turbulence. During turbulence, the example of FSI can be clearly noticed. Under the turbulence, the wing can undergo difference kinds of deformations. In present work, mainly 4 kinds of deformations are considered. These deformations are as follows:

- Bending
- Twisting
- Bending + Twisting
- Cambering

Out of these deformations, only bending, twisting and bending + twisting deformations can be found during the event of turbulence. In present work, cambering deformation is also simulated to show an example of the aerodynamic shape optimization application for the wing. The boundary conditions used for all these kinds of simulations is the Dirichlet boundary condition. The Dirichlet boundary condition is obtained using the part 1 of the overall solver as described in Sec. 2.4. Unlike the non-uniform cube case where a RBF technique was used to get the surface deformations, a FFD technique is used to get the deformations for the surface of the wing. This surface deformation works as the Dirichlet boundary condition in the present chapter. In the FFD technique for surface deformations, a FFD box is made around the surface mesh to be morphed and a deformation in the FFD box leads to the deformation in the surface mesh. The FFD box in it's original shape can be seen alongside the wing in Fig. 6.1. For different kinds of deformations, this FFD box is provided with the respective deformation information and the surface deformation or the boundary conditions are generated.

6.1.1. Bending

In order to get the Dirichlet boundary condition for the bending case, the FFD box is rotated about the line where the wing and the symmetry plane intersect, i.e. the root of the wing. For the bending deformation, mainly two cases with two different kinds of bends are considered. In the first kind of bend, the FFD box is rotated linearly, whereas in the second kind of bend, the FFD box is rotated non-linearly as shown in Fig. 6.2. A multi-step approach is used for the present study with every step equal to 5° of bending.

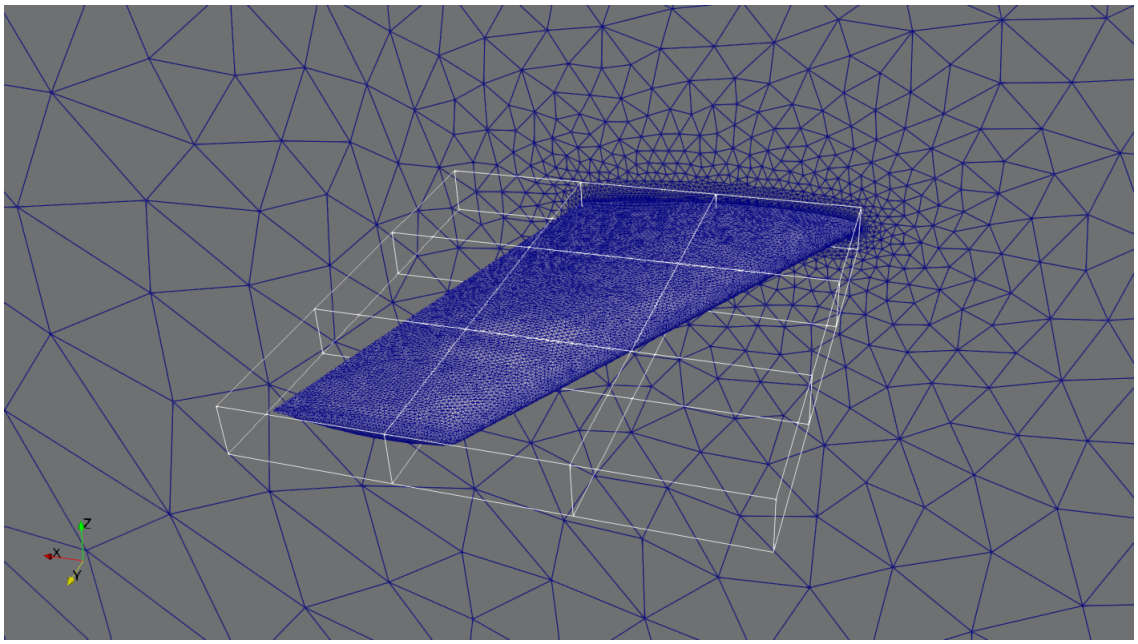
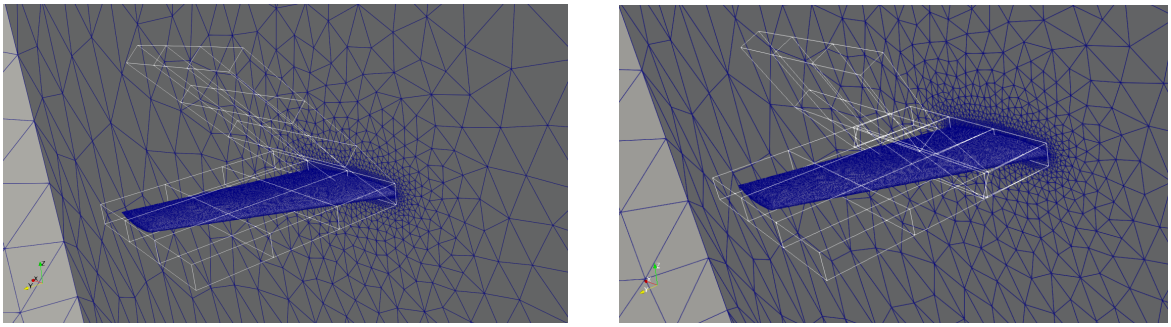


Figure 6.1: The wing of an airplane with the FFD box.



(a) Linear bend. (b) Non-linear bend.

Figure 6.2: FFD box for linear and non-linear bends.

In the linear bend as shown in Fig. 6.2a, the angle of bend is equal to the angle between the original and the new FFD box, whereas in the non-linear bend as shown in Fig. 6.2b, the angle of bend is equal to the angle between the line connecting the second and fourth points of the FFD box and the original FFD box as shown in Fig. 6.3.

The mesh quality parameters for the linear bend in the wing are listed in Table 6.2 and plotted in Fig. 6.4a for different angles of bending from 0° to 35°. It can be observed from the results that the maximum cell non-orthogonality decreases slightly until 30° and then increases rapidly, whereas the maximum boundary non-orthogonality increases rapidly with the increase in the angle of bend. The maximum cell non-orthogonality crosses the

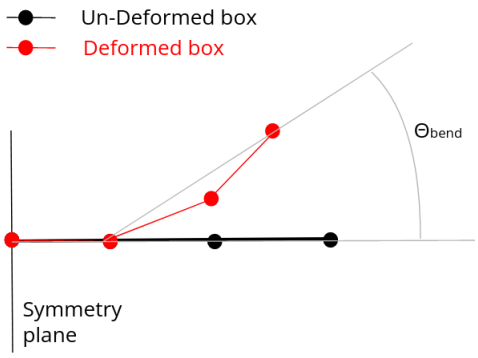


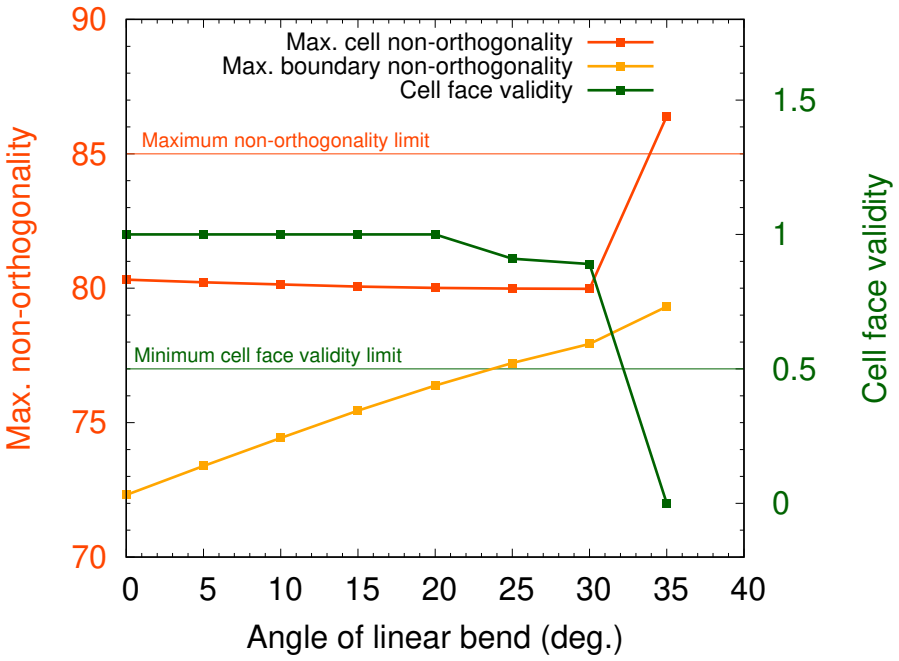
Figure 6.3: Angle of bend for a non-linear bend.

Angle of Bend (deg.)	Max. non-orthogonality (deg.)	Max. boundary non-orthogonality (deg.)	Min. cell volume	Cell face validity
0	80.32	72.31	6.95e-11	1.0
10	80.14	74.43	7.54e-11	1.0
20	80.01	76.38	6.78e-11	1.0
30	79.98	77.93	6.33e-11	0.89
35	86.4	79.32	5.9e-11	0.0

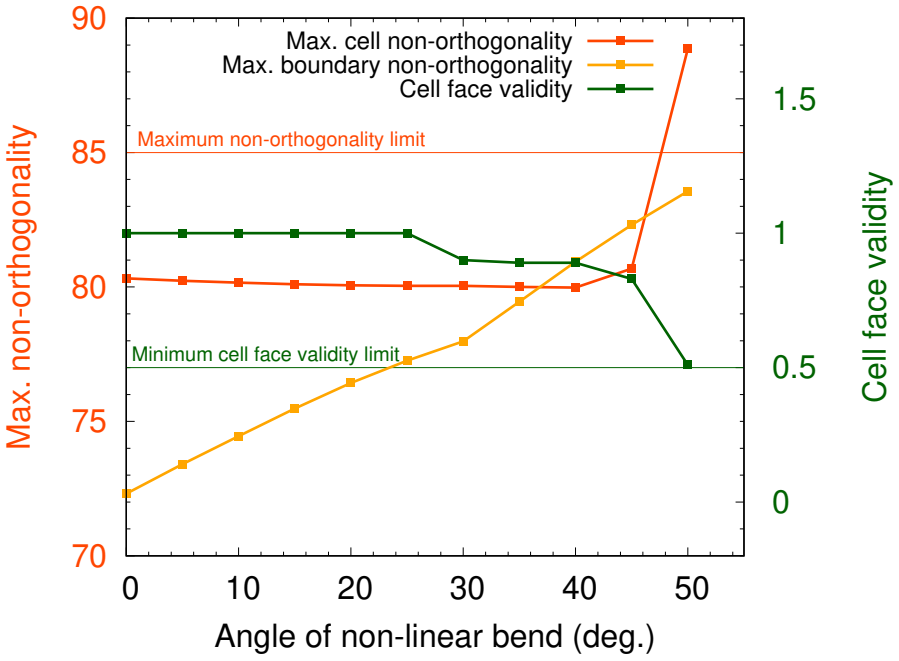
Table 6.2: Mesh quality parameters for a linear bending in the wing of an airplane.

limit of maximum non-orthogonality for a valid mesh which is set at 85° almost at 34° of bending. Moreover, it is observed that the cell face validity decreases slightly with the increase of angle of bend until almost 30° and then decreases further suddenly to zero by crossing the minimum cell face validity limit which is set at 0.5. Therefore, it is safe to say based on the simulation results that the mesh quality remains valid until almost 34° of a linear bend and becomes invalid with a further increase of angle of bend due to bad maximum non-orthogonality in the bulk mesh and inversion of some elements.

The mesh quality parameters for the non-linear bend in the wing are plotted in Fig. 6.4b and listed in Table 6.3 for different angles of bending from 0° to 50°. A similar behaviour that to the linear bend is observed in the non-linear bend. The maximum cell non-orthogonality decrease slightly with the increase in the angle of bend, where the maximum boundary non-orthogonality increases rapidly. The maximum cell non-orthogonality crosses the maximum non-orthogonality limit at almost 47° of bend. Therefore, based on the results, it can be said that a valid mesh can be obtained upto a non-linear bend of almost 47°. In the case of a non-linear bend, it is to be noticed that all the cell volumes



(a) Linear bend.



(b) Non-linear bend.

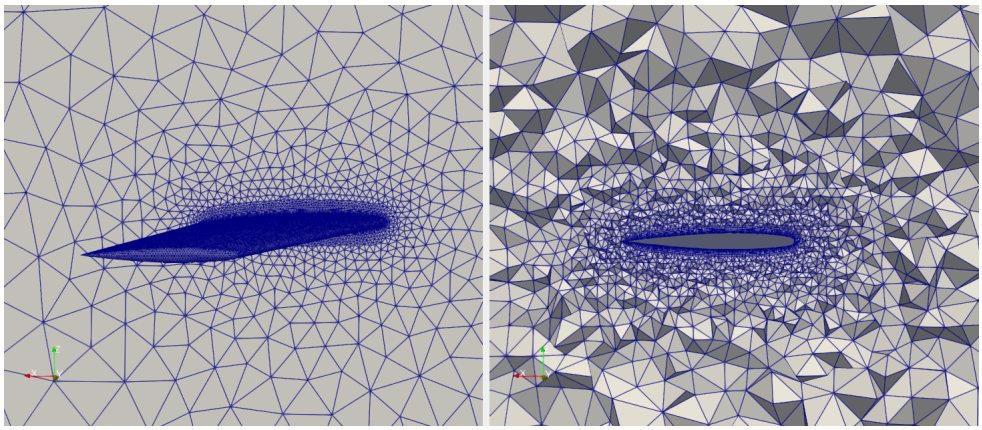
Figure 6.4: Mesh quality plot for bending in a wing.

stay positive and and no cell inversion occurs but the mesh becomes invalid because of the bad maximum non-orthogonality in the bulk mesh.

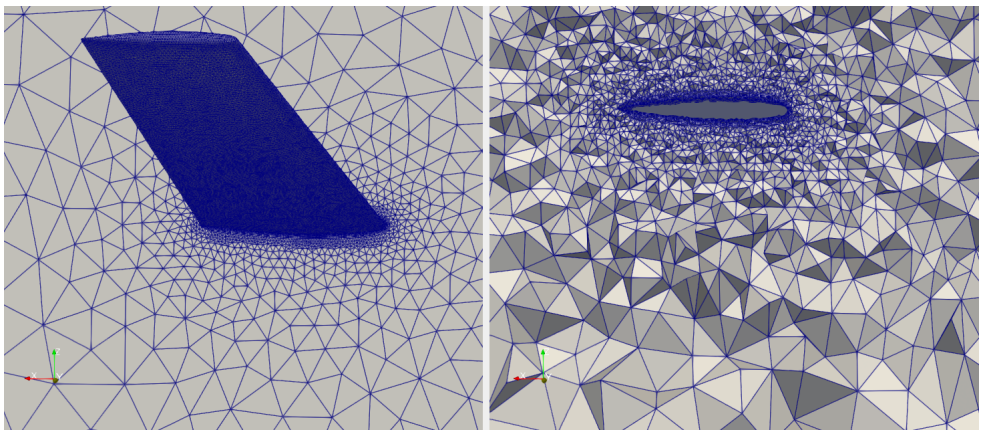
Angle of Bend (deg.)	Max. non-orthogonality (deg.)	Max. boundary non-orthogonality (deg.)	Min. cell volume	Cell face validity
0	80.32	72.31	6.95e-11	1
10	80.16	74.45	7.53e-11	1
20	80.06	76.43	6.76e-11	1
30	80.04	77.98	6.31e-11	0.9
40	79.98	80.94	5.5e-11	0.89
45	80.68	82.31	5.19e-11	0.83
50	88.86	83.56	4.96e-11	0.51

Table 6.3: Mesh quality parameters for a non-linear bending in the wing of an airplane.

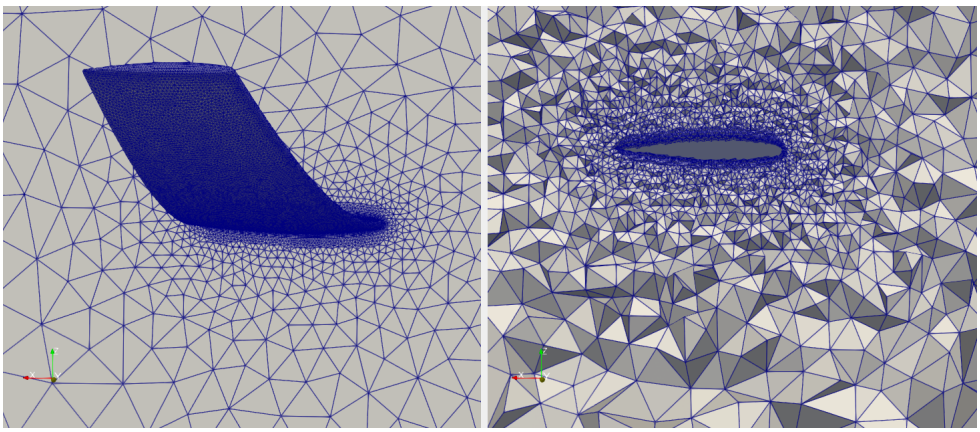
The original wing mesh with the maximum possible linear and non-linear bends of 30° and 45° respectively with a valid mesh are shown in Fig. 6.5. The left hand side figures



(a) Undeformed mesh.



(b) Linear bend of 30°.



(c) Non-linear bend of 45°.

Figure 6.5: Bending in a wing for linear and non-linear bends.

in Fig. 6.5 show the surface deformation obtained using the FFD technique and the right hand side figures show the deformed volumetric mesh at a plane parallel to the symmetry plane near the tip of the wing. It can be seen that due to the upward bends, the aerofoil shaped region of the wing moves up resulting in the mesh over the wing compressed whereas the mesh below the wing expanded. In both the linear and non-linear bends, it can be observed from the results that the mesh near the boundary of the wing remains almost the same due to small cell volume and high cell stiffness whereas the mesh far from the boundary deforms due to high cell volumes and low cell stiffness. This way, the shape of the cells near the surface of the wing preserves.

6.1.2. Twisting

In the twisting kind of deformation on the wing, the Dirichlet boundary condition is obtained by rotating the FFD box with respect to an axis normal to the symmetry plane and passing through the center of the FFD box as shown in Fig. 6.6. A multi-step approach is used for the twisting case with an increment of 5° in every step.

The mesh quality parameters for the twisting deformation are plotted in Fig. 6.7 and listed in Table 6.4 for various angles of twist from 0° to 50°. It is observed from the results that the maximum cell and boundary non-orthogonality increase with an increase in the angle of twist. The maximum boundary non-orthogonality crosses the limit of maximum non-orthogonality at an angle of twist almost equal to 47°. It is also observed that the cell face validity decreases with an increase in the angle of twist approaching the limit value of 0.5 at 47° of angle of twist. It is to be noticed from Fig. 6.7 that a decrease in maximum boundary non-orthogonality is observed for the angle of twist from 0° to 5°.

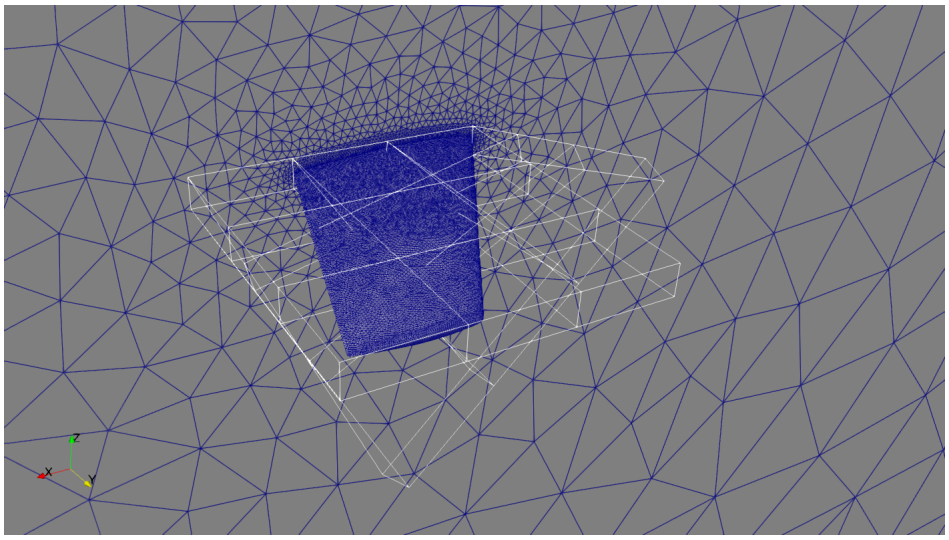


Figure 6.6: FFD box for a twist in the wing of an airplane.

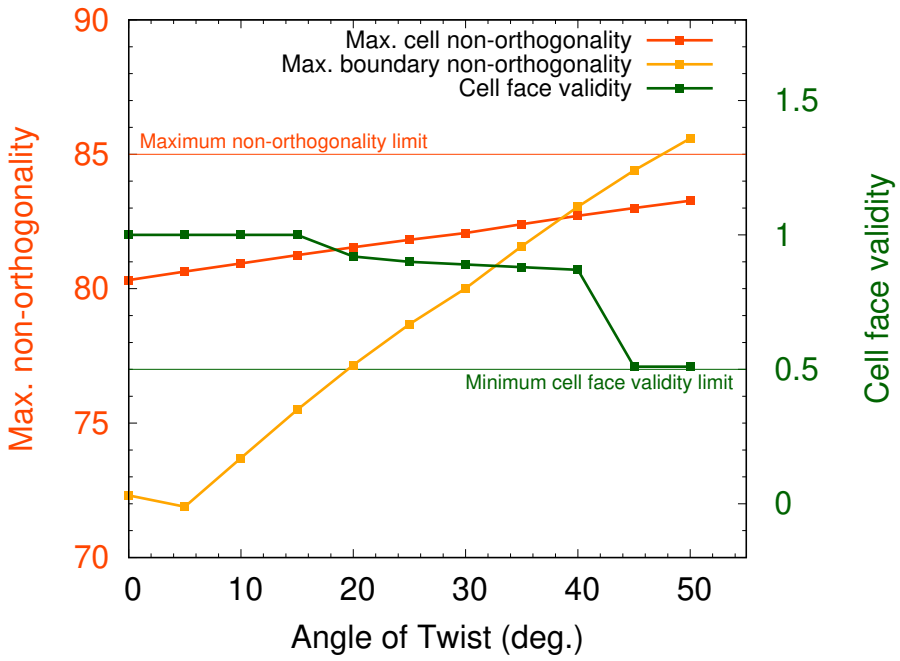


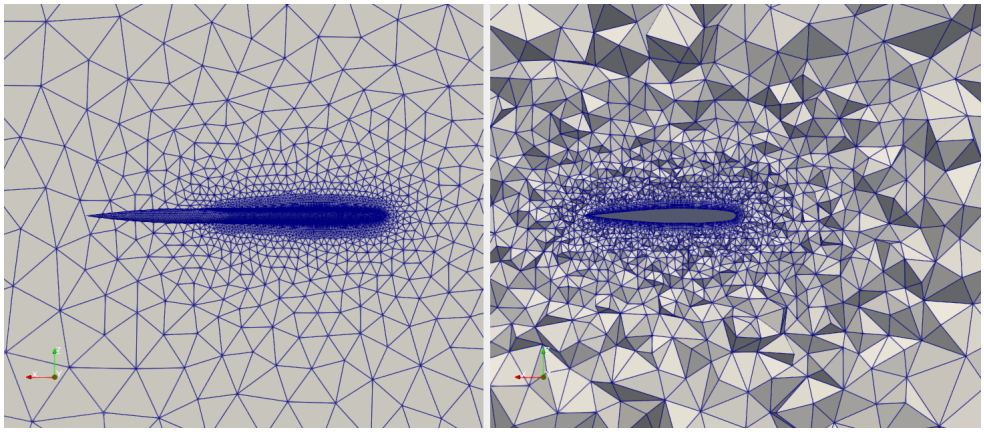
Figure 6.7: Mesh quality plot for a twist in the wing of an airplane.

The reason behind this behaviour can be the initial structure of the mesh which might get better at the boundary due to a twist until 5°. Therefore, it can be said based on the results shown in Fig. 6.7 and Table 6.4 that a valid mesh is obtained upto an angle of twist of 47°.

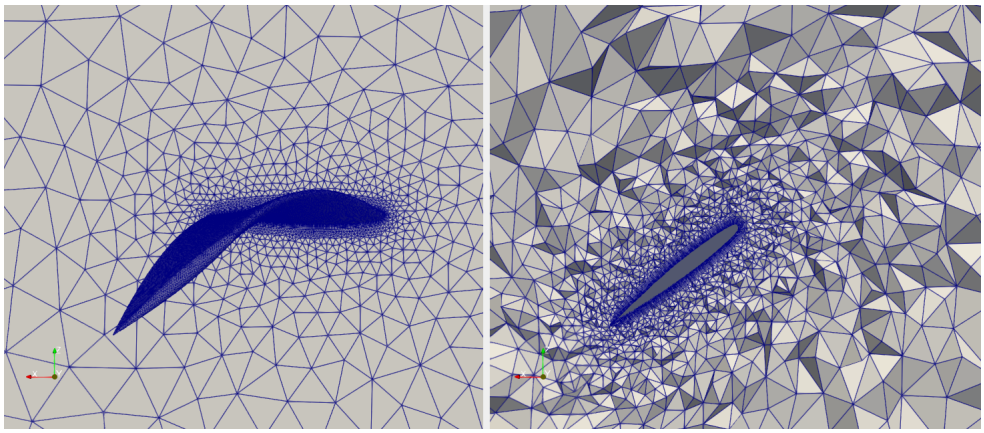
It is to be noticed that for the case with 50° angle of twist, unlike the bending case, the mesh fails because of the maximum boundary non-orthogonality crossing the limits inst-

Angle of twist (deg.)	Max. non-orthogonality (deg.)	Max. boundary non-orthogonality (deg.)	Min. cell volume	Cell face validity
0	80.32	72.31	6.95e-11	1.0
10	80.94	73.70	5.75e-11	1.0
20	81.54	77.16	4.66e-11	0.92
30	82.07	80.01	3.81e-11	0.89
40	82.71	83.05	2.7e-11	0.87
45	83.0	84.4	2.22e-11	0.51
50	83.27	85.59	1.78e-11	0.51

Table 6.4: Mesh quality parameters for twisting in the wing of an airplane.



(a) Undeformed mesh.



(b) Twist of 45°.

Figure 6.8: Twisting in a wing.

ead of maximum cell non-orthogonality, which indicates that the twisting of the wing deteriorates the quality of mesh more in the boundary mesh, whereas the bending of wing deteriorates the quality of mesh more in the bulk mesh. The original wing mesh alongside the mesh with a twist of 45° with a valid mesh are shown in Fig. 6.8. The left hand side figures show the surface deformation obtained using the FFD technique whereas the right hand side figures show the deformed volumetric mesh at a plane parallel to the symmetry plane near the tip of the wing. It can be observed from Fig. 6.8 that the mesh near the boundary of the wing stays the same even after a big twist of angle 45°.

6.1.3. Bending + Twisting

Another possible case study is the case with both the bending and twisting deformations together. The deformation in the FFD box for such a deformation is shown in Fig. 6.9. Because of more feasibility, the non-linear bending is chosen in the present section among the options. A multi-step approach with an increment of 5° of angle of bend and angle of twist both in every step is chosen for the present section.

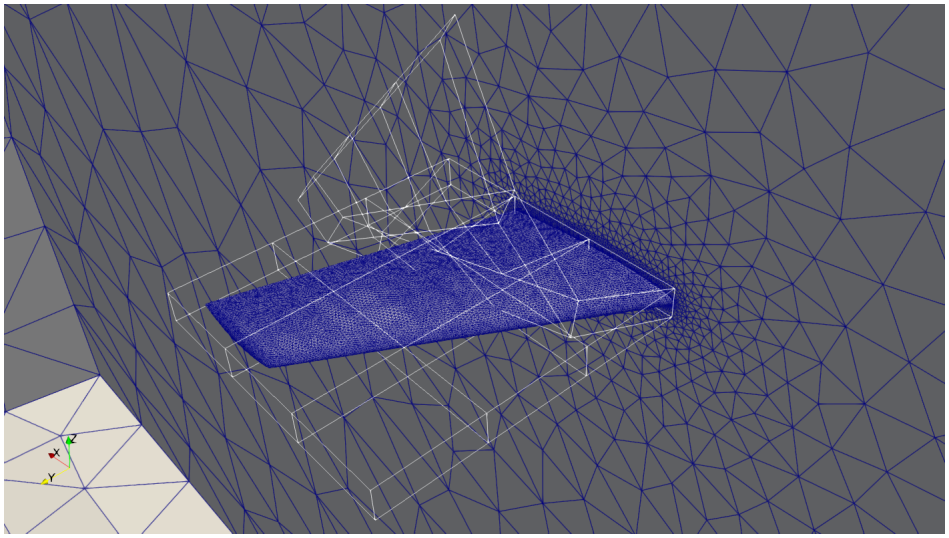


Figure 6.9: FFD box for a bend + twist in the wing of an airplane.

The mesh quality parameters are plotted in Fig. 6.10 and listed in Table 6.5 for different angles starting from 0° to 55°. It is observed from the results that the maximum cell and boundary non-orthogonality increase with an increase in the angle of bend and twist. The maximum cell non-orthogonality crosses the limit of maximum non-orthogonality at an angle of bend and twist equal to almost 51°. It is also observed that the cell face validity decreases with an increase in the angle of bend and twist. Therefore, it can be said that the original mesh of the wing can be deformed upto 51° bend and twist altogether with

a valid mesh.

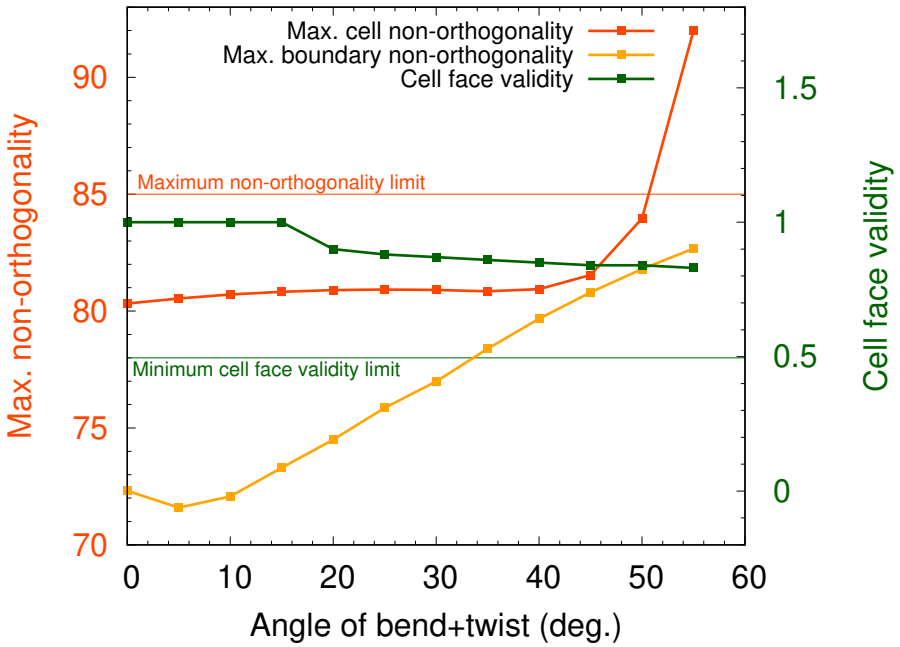
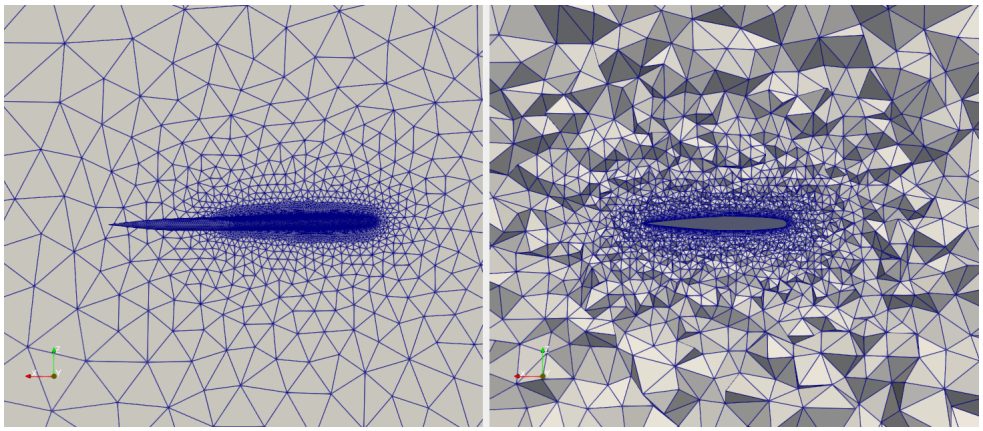


Figure 6.10: Mesh quality plot for a bend + twist in the wing of an airplane.

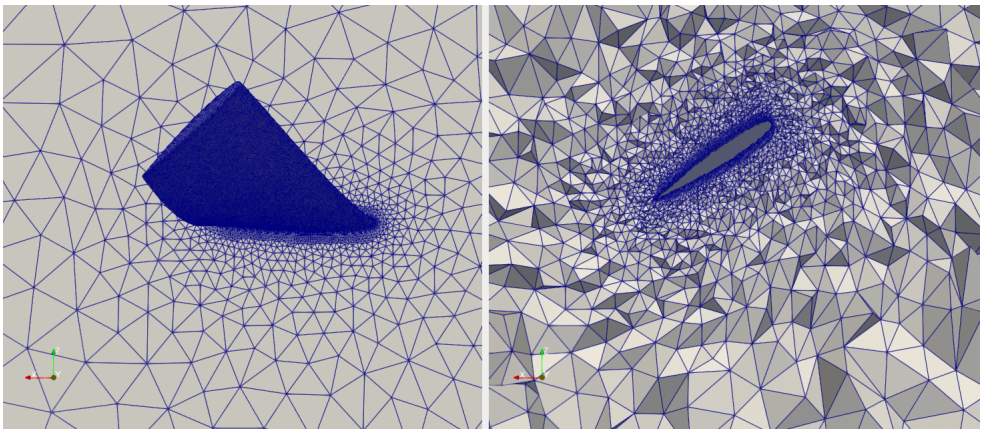
Angle of bend and twist (deg.)	**Max. non-orthogonality (deg.)**	**Max. boundary non-orthogonality (deg.)**	**Min. cell volume**	**Cell face validity**
0	80.3215	72.31	6.95e-11	1.0
10	80.707	72.07	6.55e-11	1.0
20	80.8959	74.50	6.31e-11	0.90
30	80.905	76.99	6.46e-11	0.87
40	80.9365	79.67	5.23e-11	0.85
50	83.9625	81.80	4.27e-11	0.84
55	91.99	82.67	3.91e-11	0.83

Table 6.5: Mesh quality parameters for bending + twisting in the wing of an airplane.

The original mesh alongside the mesh with a bend + twist of 50° is shown in Fig. 6.11. The left hand side figures show the surface deformation in the wing and the right hand side figures show a plane parallel to the symmetry plane close to the tip of the wing. It



(a) Undeformed mesh.



(b) Non-linear bend + twist of 50°.

Figure 6.11: Non-linear bending + twisting in a wing.

can be observed in right hand side of Fig. 6.11 that the aerofoil boundary moves up due to bending and rotates due to twisting. It can be observed from the results that the mesh near the boundary of the wing does not change much and moves as the wing surface moves keeping the cells near the boundary preserved despite large movement.

6.1.4. Cambering

During the aerodynamic shape optimization of the wing, the camber angle of the wing is one of the most influential parameter to determine the lift or drag forces on the wing. The design iterations for the design of the wing consist of the different camber angles throughout the span. In present subsection, the wing is given cambering kind of deformation where the wing is undergone an incremental camber angle starting from zero camber at the root of the wing. In the initial geometry of the wing, the camber angle is equal to zero throughout the span, i.e. a symmetric wing. A multi-step approach is used in the

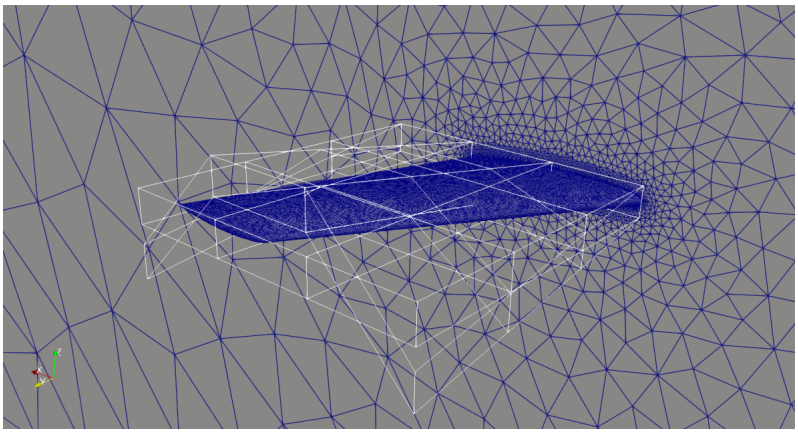


Figure 6.12: FFD box for cambering the wing of an airplane.

present subsection with a cambering angle at the tip from 0° to 60°. The FFD box for a cambering of a wing is shown in Fig. 6.12. The mesh quality parameters for all the multi-steps are plotted in Fig. 6.13 and listed in Table 6.6. It can be observed that the maximum cell and boundary non-orthogonality increases with an increase in the camber angle. The maximum cell non-orthogonality crosses the limit of 85° at a tip camber angle of almost 56°. The cell face validity decreases with an increase in the tip camber angle and crosses the limit of 0.5 at an tip camber angle of almost 55°. Therefore, based on the results, it can be said that the present developed elasticity solver can camber the wing until an angle of 55°.

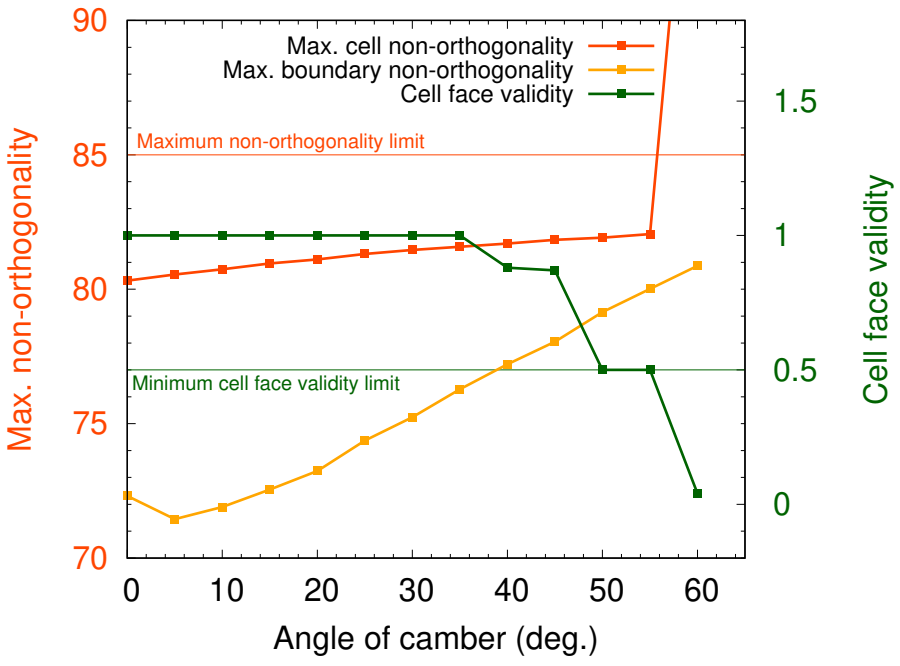
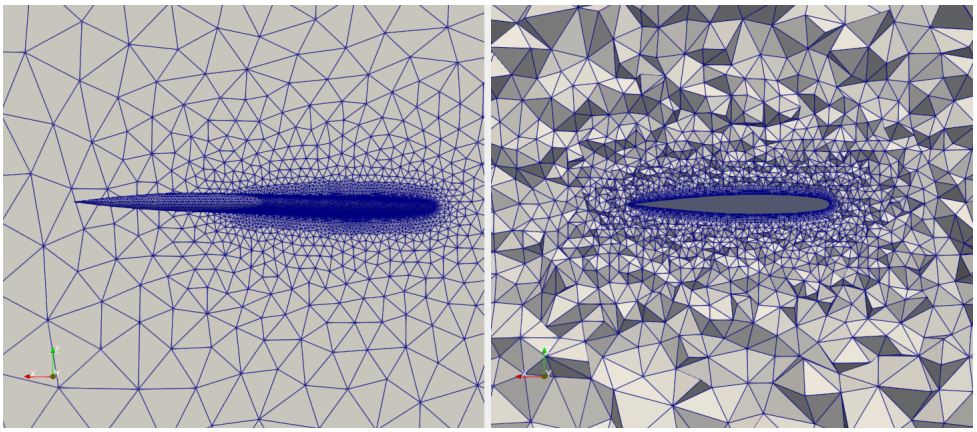


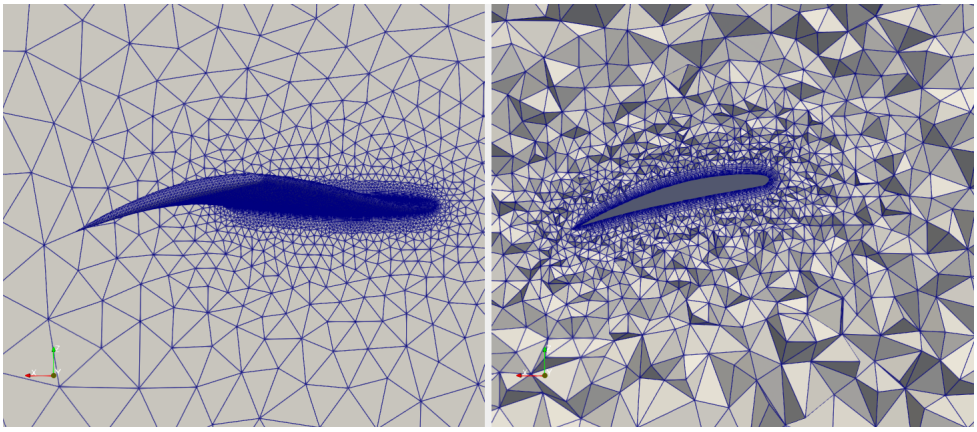
Figure 6.13: Mesh quality plot for cambering in the wing of an airplane.

Angle of camber (deg.)	Max. non-orthogonality (deg.)	Max. boundary non-orthogonality (deg.)	Min. cell volume	Cell face validity
0	80.32	72.31	6.95e-11	1.0
10	80.74	71.9	6.31e-11	1.0
20	81.11	73.24	5.83e-11	1.0
30	81.46	75.24	5.27e-11	1.0
40	81.7	77.21	4.71e-11	0.88
50	81.92	79.15	4.02e-11	0.50
55	82.05	80.01	3.12e-11	0.50
60	101.26	80.87	-4.9e-09	0.04

Table 6.6: Mesh quality parameters for cambering in the wing of an airplane.



(a) Undeformed mesh.



(b) Cambering of 55°.

Figure 6.14: Cambering in the wing of an airplane.

The original undeformed mesh and the deformed mesh with most possible tip camber angle with valid mesh are shown in Fig. 6.14. It can be seen in the results in the right hand side figures that the cells near the boundary of the wing at a plane parallel to the symmetry plane remain preserved and results in a valid mesh. These results confirm that the presently developed linear elasticity technique can be used for aerodynamic shape optimization applications.

6.1.5. Comparison with Laplacian quaternions technique

For the industrial case of the wing of an airplane, all the kinds of deformations performed in the present section are simulated using a Laplacian quaternion technique and compared with the results obtained for presently developed linear elasticity technique. A multi-step approach with 10 steps is used for both the techniques. The maximum deformations possible with a valid mesh using both the techniques are listed in Table 6.7.

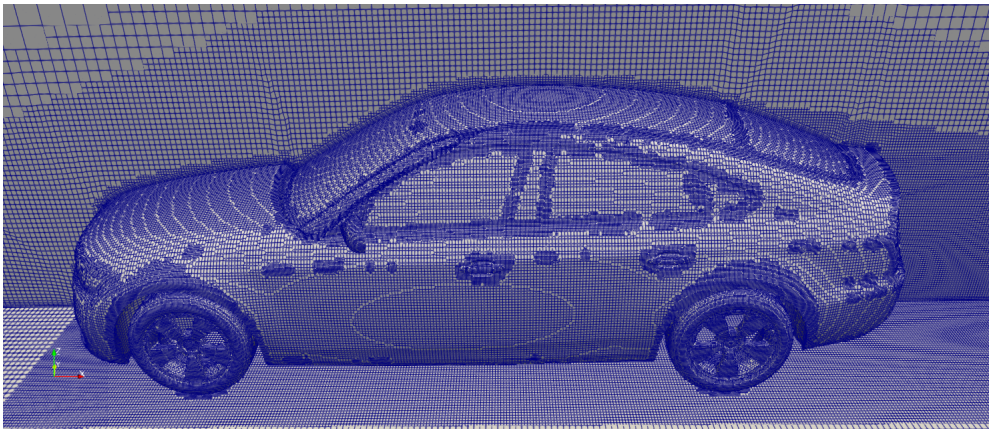
Kind of deformation	**Linear elasticity technique (deg.)**	**Laplacian quaternions technique (deg.)**
Non-linear bending	47	46
Twisting	47	48
Non-linear bending + twisting	51	42
Cambering	55	54

Table 6.7: Maximum deformation with a valid mesh using different techniques.

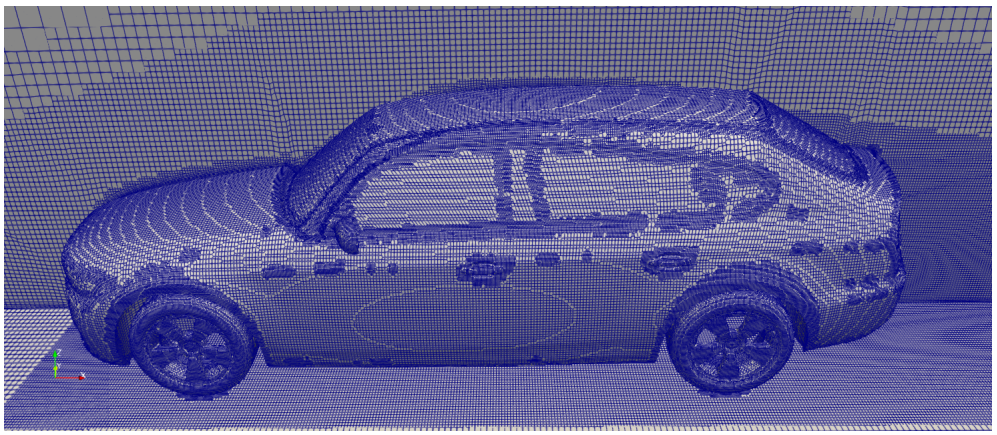
It can be observed that the Laplacian quaternions technique gives comparable results to the linear elasticity technique for the non-linear bending and twisting kinds of deformations. For the non-linear bending + twisting kind of deformation, the presently developed linear elasticity technique is able to provide valid mesh for 9 more degrees of deformation than the Laplacian quaternions technique. For the cambering deformation, the results seem to be comparable. Therefore, it can be said based on the results, that between the two schemes, the maximum possible deformation is comparable for simpler kinds of deformations like non-linear bending, twisting and cambering, whereas the presently developed linear elasticity technique outperforms the Laplacian quaternions technique for a relatively complex deformation like non-linear bending + twisting.

6.2. Aerodynamic shape optimization in a Car

Aerodynamics shape optimization is another important application of mesh morphing. In the automobile sector, the design phase contains numerous number of design iterations and it can be highly time and resource expensive to reconstruct the mesh for all the design iterations. Therefore, a robust mesh morphing tool developed in the present work can play an important role in speeding up the aerodynamic shape optimization. In present section, the industrial case of a car is considered as shown in Fig. 6.15a. The mesh for the car consists of almost 2 million cells with polyhedral shape. To obtain the boundary deformation as the boundary condition, a RBF parameterization technique is used. The deformation in the surface of the car considered in the present section is shown in Fig. 6.15b. For the surface deformation, 6 different kinds of deformations are provided at different parts of the car. These deformations are: front window angle change, rear window angle change, roof drop or change in roof angle, greenhouse angle change, front



(a) Undeformed surface mesh.



(b) Deformed surface mesh.

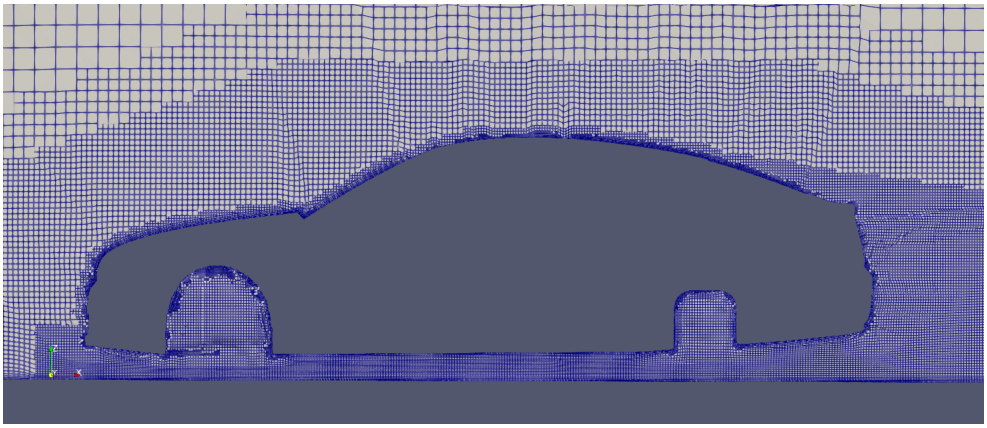
Figure 6.15: Surface deformation in the industrial case of a car.

bumper nose extrusion, and front bumper nose drop. All these deformations can be observed in Fig. 6.15.

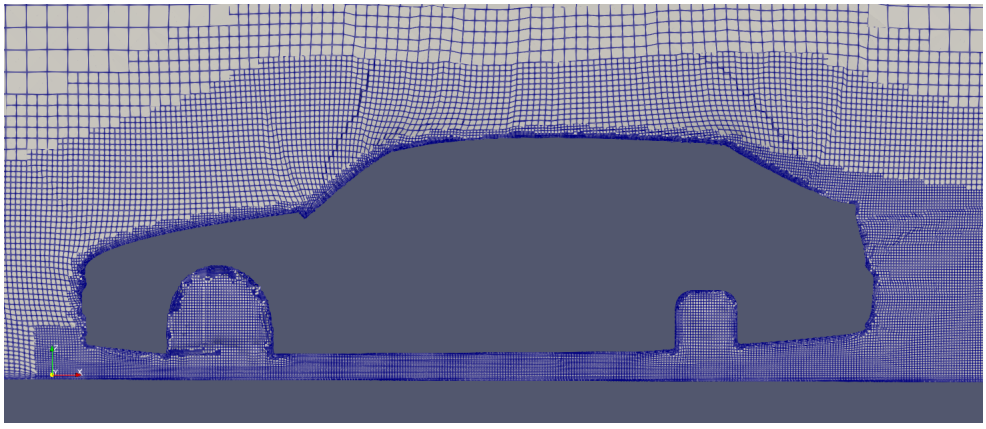
It is to be noticed that the Dirichlet boundary condition is available only at the car surface and the walls of the CFD domain, and therefore, a slip condition as described in Sec. 2.3 is applied at the symmetry plane.

In present section, a multi-step approach with 5 steps is used to propagate the deformation in the volumetric mesh of the car. The elasticity coefficient E for the present section is computed using the relation with the inverse of volume with an exponent $p = 1$. The volumetric mesh displacement is demonstrated by showing a plane parallel to the symmetry plane for the initial and final geometries as shown in Fig. 6.16. The mesh quality parameters for the initial geometry and the geometries after all the steps of the multi-step approach are listed in Table 6.8.

It can be observed from the results that from step 0 to step 3, the mesh quality changes



(a) Undeformed mesh.



(b) Deformed mesh.

Figure 6.16: Volumetric mesh at a plane parallel to the symmetry plane.

Step	Max. non-orthogonality (deg.)	Max. boundary non-orthogonality (deg.)	Min. cell volume	Cell face validity
0	76.99	103.26	1.43e-09	0.53
1	76.99	103.25	1.44e-09	0.53
2	76.99	103.24	1.44-09	0.53
3	77.87	103.23	1.45e-09	0.53
4	87.11	103.22	1.45e-09	0.53
5	110.64	103.2	1.46e-09	0.53
Laplacian quaternions	178.54	175.19	-8.41e-05	0

Table 6.8: Mesh quality parameters for the industrial case of a car.

very little and the mesh remains valid, whereas for step 4, a considerable increase in the maximum cell non-orthogonality is observed. It is found for step 4 that one face exists where the cell non-orthogonality crosses the 85° limit of maximum non-orthogonality. For the final step 5, it is observed that the maximum cell non-orthogonality increases further to 110°. It is found that four faces have non-orthogonality more than the limit of 85°. Despite the limit crossing non-orthogonality in few faces, it is found that the cell volume remains positive for all the cells and cell face validity remains more that the limit 0.5. The maximum non-orthogonality at the boundaries is found out to be almost 103° for all the steps because of some bad cells near the wheels of the car. Since a zero Dirichlet boundary condition is provided to the wheels, these bad quality cells do not affect the solution considerably.

The results for the present industrial case are compared also with the results obtained using a multi-step approach Laplacian quaternions technique as explained in Chapter 5. The mesh quality parameters after the final step of the Laplacian quaternions technique are listed in Table 6.8. It can be observed from Table 6.8 that negative volumes with failed cell face validity exist for the solution of the Laplacian quaternions technique. It is also observed that the maximum cell and boundary non-orthogonality come out to be unacceptably high for the Laplacian quaternions method. Therefore, based on the comparison, it can be said that the linear elasticity solver gives much better results for the present industrial case than the Laplacian quaternions technique which was giving comparable quality results in case of a non-uniform cube as shown in Chapter 5.

Therefore, based on the results, it can be said for the present industrial case that the linear

elasticity technique with multi-step approach works well and can be used for aerodynamic shape optimization. Although, for large deformations, the user must be careful about the high non-orthogonality at a few faces.

6.3. Conclusions

In this chapter, the applications of the presently developed linear elasticity mesh morpher are demonstrated by means of two different industrial cases. An industrial case of the wing of an airplane is considered to show the application in the field of FSI, whereas an industrial case of a car is used to show the application in the field of aerodynamic shape optimization. Both the industrial cases are undertaken different kinds of deformations and the following conclusions are made:

- For the possible deformations under a turbulence like situation, the wing can withstand a linear bending upto 34°, non-linear bending upto 47°, twisting upto 47°, cambering upto 55°, and bending + twisting of upto 51°.
- By increasing the angle of bend or twist more than the limit mentioned in above bullet, the mesh fails due to a high maximum cell non-orthogonality (>85°) for the bending and bending + twisting cases and due to a high maximum boundary non-orthogonality for the twisting case. The cambering deformation fails because of the max. non-orthogonality and cell face validity crossing the respective limits and a negative cell volume altogether.
- The maximum possible bend, twist or camber with a valid mesh with the use of the linear elasticity technique are of the order to that when using the Laplacian quaternions technique, whereas the linear elasticity solver outperforms the Laplacian quaternions technique for a relatively complex non-linear bending + twisting kind of deformation.
- The present linear elasticity based technique can morph the high deformation provided in the design of the car.
- For the car case, the presently developed linear elasticity technique provides with much better mesh compared to the Laplacian quaternions technique.
- For both the wing and the car industrial cases, it is found that the mesh near the boundary preserved their shape despite high deformations and makes the presently developed linear elasticity technique a robust tool to deform the meshes with high deformations thanks to the multi-step approach.

7 | Conclusions and future developments

In present thesis, a mesh morpher based on the linear elasticity analogy is developed to propagate the surface deformation in the volumetric mesh. A FVM is used to descritize the PDEs and a PETSc library is used to solve the linear system of equations. While computing the cell and face gradients using a $2nd$ order least square method, the missing boundary contributions in the original method are also considered in order to keep the $2nd$ order of accuracy throughout the domain including the boundary. The newly added boundary contributions are tested over a sample function for simple geometries and it is found that the added boundary contributions improve the accuracy of the cell and face gradients. Since the linear elasticity law is defined for small displacements, a multi-step step approach is developed where the solution geometry of the previous step is used as the initial geometry of the next step. To demonstrate the results, two different kinds of deformations, outward and inward are used for the case of a cube with 64k cells. For an outward deformation, it is found that the quality of the resulted mesh improves with an increase in the number of the steps in the multi-step approach, whereas in the inward deformation, it is found that a multi-step approach improves the quality of the resulted mesh considerably with respect to a single step approach but increasing the number of steps does not improve mesh quality further due to congestion in the internal mesh.

A variation of two of the pre-existing relations for the computation of the elasticity coefficient E is proposed and analyzed in the present work. It is observed that the optimum values of the parameters used in these relations depend on the kind of deformation and the complexity of the mesh. For a simple mesh like a non-uniform cube, the parameter p, which is the exponent of the inverse of volume relation, gives best results for an outward deformation when lying in a range $1-2$, whereas gives best results for an inward deformation when lying in a range $3-4$. On the other hand, for the complex meshes in the industrial cases, the wing of an airplane and the car, it is found that the best results are produced for the exponent $p = 1$. Similarly, the exponent q in case of inverse of distance relation gives best results when set equal to 1 for outward and inward both kinds

of deformation in a non-uniform cube. Therefore, it can be concluded that the proposed relations for the computation of the elasticity coefficient provides the user with a degree of freedom to choose a p or q depending on the kind of deformation and the complexity of geometry.

In present work, the results obtained using the presently developed linear elasticity technique are also compared with some of the promising techniques such as the Laplacian technique and the Laplacian quaternions technique with and without a narrow band width for single and multi-step approaches. For the initial geometry of a non-uniform cube, it is found that the linear elasticity technique clearly outperforms the Laplacian technique with and without the narrow band width for both the outward and inward kinds of deformations, whereas the Laplacian quaternions technique produces comparable results with the linear elasticity technique for an inward deformation and worse results than the linear elasticity technique for an outward deformation. For the complex initial geometry of a wing, simple kinds of deformations like linear and non-linear bending, twisting, and cambering, the linear elasticity technique can generate a valid mesh up to same order of angle of respective deformation as with Laplacian quaternions technique, whereas the linear elasticity technique outperforms the Laplacian quaternions technique for a relatively complex deformation like non-linear bending + twisting. For another industrial case of the car, it is observed for a complex boundary condition that the linear elasticity solver outperforms the Laplacian quaternions technique by a good margin.

The developed linear elasticity technique is tested for two different CFD applications, FSI and aerodynamic shape optimization, by means of two different industrial cases with much more complex and different kinds of meshes than the non-uniform cube. The first industrial case of the wing of an airplane contained tetrahedral and hexagonal mesh cells and tested over different kinds of deformations inspired by the turbulence event and aerodynamic shape optimization. Under the turbulence event, the wing of the airplane gives valid meshes until 47 degrees of twisting, 47 degrees of bending and 51 degrees of twisting + bending. As an application of aerodynamic shape optimization, the wing showed a valid mesh until a cambering of 55 degrees. It is observed in all kinds of deformations that the mesh near the boundary of the wing stays preserved due to small volume resulting in high stiffness caused by inverse of volume relation with exponent $p = 1$. The second industrial case of a car shows an application of aerodynamic shape optimization where the surface mesh of the car is morphed highly and this surface displacements are propagated in the volume mesh. This mesh contained polygonal kinds of mesh cells and is used for turbulent CFD. It is observed that the presently developed linear elasticity solver is able to solve the high deformation provided to the car without any negative cell

volume or cell inversion. The cells near the boundary of the car preserve their shape due to high stiffness caused by small mesh volume.

Therefore, based on the results of the non-uniform cube, the wing of an airplane and the car, it can be said that the presently developed multi-step approached linear elasticity solver provides with the best results compared to the Laplacian and Laplacian quaternions techniques with and without the narrow band width. This shows and affirms that the presently developed linear elasticity based mesh morpher can be used for the FSI, aerodynamic shape optimization and all the other kinds of CFD applications where mesh morphing is required.

It is to be noticed that the present work also sets up a platform for possible future research and opens up new possibilities in the field of mesh morphing for CFD applications. Some of these possibilities are as follows:

- Based on the present work, it can be said that the different kinds of meshes and deformations require different optimum exponents for the inverse of volume method and therefore, an interesting future scope of the present work is to distribute the exponent p over the cells of the mesh rather than a single value. The same stands for the inverse of distance method as well.
- Another possibility is to merge the linear elasticity technique and the Laplacian or Laplacian quaternions technique in such a way that the later can be used when it comes to the cells close to zero Dirichlet boundary condition and the earlier can be used otherwise. The main benefit of this merge of two different techniques will be decreases in complexity of the linear elasticity solver.
- The present method of implementation includes the use of FVM which includes interpolation errors at the end. If the equations are descretized using a FEM than the interpolation error can be avoided and there will not be any need of the propagation of the error in the whole mesh.

Bibliography

[1] Hong Fang, Chunye Gong, Caihui Yu, Changwan Min, Xing Zhang, Jie Liu, and Liquan Xiao. Efficient mesh deformation based on cartesian background mesh. *Computers & Mathematics with Applications*, 73(1):71–86, 2017.

[2] O Hassan, EJ Probert, and K Morgan. Unstructured mesh procedures for the simulation of three-dimensional transient compressible inviscid flows with moving boundary components. *International journal for numerical methods in fluids*, 27(1-4):41–55, 1998.

[3] MM Selim and RP Koomullil. Mesh deformation approaches–a survey. *Journal of Physical Mathematics*, 7(2):1–9, 2016.

[4] Mamadou T Bah, Prasanth B Nair, and Martin Browne. Mesh morphing for finite element analysis of implant positioning in cementless total hip replacements. *Medical engineering & physics*, 31(10):1235–1243, 2009.

[5] RP Dwight and J Brezillon. Effect of various approximations of the discrete adjoint on gradient-based optimisation. In *44th AIAA Aerospace Meeting*, pages 9–12, 2006.

[6] Zhi Yang and Dimitri Mavriplis. Unstructured dynamic meshes with higher-order time integration schemes for the unsteady navier-stokes equations. In *43rd AIAA aerospace sciences meeting and exhibit*, page 1222, 2005.

[7] John T Batina. Unsteady euler airfoil solutions using unstructured dynamic meshes. *AIAA journal*, 28(8):1381–1388, 1990.

[8] Joe F Thompson, Zahir UA Warsi, and C Wayne Mastin. *Numerical grid generation: foundations and applications*. Elsevier North-Holland, Inc., 1985.

[9] Yong Zhao and Ahmed Forhad. A general method for simulation of fluid flows with moving and compliant boundaries on unstructured grids. *Computer methods in applied mechanics and engineering*, 192(39-40):4439–4466, 2003.

[10] Jeroen Witteveen. Explicit and robust inverse distance weighting mesh deformation

for cfd. In *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, page 165, 2010.

[11] Aukje De Boer, Martijn S Van der Schoot, and Hester Bijl. Mesh deformation based on radial basis function interpolation. *Computers & structures*, 85(11-14):784–795, 2007.

[12] Xueqiang Liu, Ning Qin, and Hao Xia. Fast dynamic grid deformation based on delaunay graph mapping. *Journal of Computational Physics*, 211(2):405–423, 2006.

[13] Jamshid A Samareh. Application of quaternions for mesh deformation. Technical report, 2002.

[14] Frederic J Blom. Considerations on the spring analogy. *International journal for numerical methods in fluids*, 32(6):647–668, 2000.

[15] Charbel Farhat, Christoph Degand, Bruno Koobus, and Michel Lesoinne. Torsional springs for two-dimensional dynamic unstructured fluid meshes. *Computer methods in applied mechanics and engineering*, 163(1-4):231–245, 1998.

[16] George A Markou, Zacharias S Mouroutis, Dimos C Charmpis, and Manolis Papadrakakis. The ortho-semi-torsional (ost) spring analogy method for 3d mesh moving boundary problems. *Computer methods in applied mechanics and engineering*, 196(4-6):747–765, 2007.

[17] Carlo L Bottasso, Davide Detomi, and Roberto Serra. The ball-vertex method: a new simple spring analogy method for unstructured dynamic meshes. *Computer Methods in Applied Mechanics and Engineering*, 194(39-41):4244–4264, 2005.

[18] Robert Biedron and Elizabeth Lee-Rausch. Rotor airloads prediction using unstructured meshes and loose cfd/csd coupling. In *26th AIAA Applied Aerodynamics Conference*, page 7341, 2008.

[19] Andrew A Johnson and Tayfun E Tezduyar. Simulation of multiple spheres falling in a liquid-filled tube. *Computer Methods in Applied Mechanics and Engineering*, 134(3-4):351–373, 1996.

[20] Eric J Nielsen and W Kyle Anderson. Recent improvements in aerodynamic design optimization on unstructured meshes. *AIAA journal*, 40(6):1155–1163, 2002.

[21] Steve L Karman Jr, W Kyle Anderson, and Mandar Sahasrabudhe. Mesh generation using unstructured computational meshes and elliptic partial differential equation smoothing. *AIAA journal*, 44(6):1277–1286, 2006.

[22] Steve L Karman Jr. Unstructured viscous layer insertion using linear-elastic smoothing. *AIAA journal*, 45(1):168–180, 2007.

[23] Zhi Yang and Dimitri J Mavriplis. Mesh deformation strategy optimized by the adjoint method on unstructured meshes. *AIAA journal*, 45(12):2885–2896, 2007.

[24] Su-Yuen Hsu, Chau-Lyan Chang, and Jamshid Samareh. A simplified mesh deformation method using commercial structural analysis software. In *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, page 4409, 2004.

[25] Richard P Dwight. Robust mesh deformation using the linear elasticity equations. In *Computational fluid dynamics 2006*, pages 401–406. Springer, 2009.

[26] Stefan Jakobsson and Olivier Amoignon. Mesh deformation using radial basis functions for gradient-based aerodynamic shape optimization. *Computers & Fluids*, 36(6):1119–1136, 2007.

[27] Marco Evangelos Biancolini. Mesh morphing and smoothing by means of radial basis functions (rbf): a practical example using fluent and rbf morph. In *Handbook of research on computational science and engineering: theory and practice*, pages 347–380. IGI Global, 2012.

[28] Thomas CS Rendall and Christian B Allen. Efficient mesh motion using radial basis functions with data reduction algorithms. *Journal of Computational Physics*, 228(17):6231–6249, 2009.

[29] Chunhua Sheng and Christian B Allen. Efficient mesh deformation using radial basis functions on unstructured meshes. *AIAA journal*, 51(3):707–720, 2013.

[30] Mohamed M Selim and Roy Koomullil. Incremental matrix inversion approach for radial basis function mesh deformation. In *Proceedings of the Fifteenth Annual Early Career Technical Conference, The University of Alabama at Birmingham*, 2015.

[31] Andreas K Michler. Aircraft control surface deflection using rbf-based mesh deformation. *International Journal for Numerical Methods in Engineering*, 88(10):986–1007, 2011.

[32] Leonidas J Guibas, Donald E Knuth, and Micha Sharir. Randomized incremental construction of delaunay and voronoi diagrams. *Algorithmica*, 7(1):381–413, 1992.

[33] Luc Devroye, Ernst P Mücke, and Binhai Zhu. A note on point location in delaunay triangulations of random points. *Algorithmica*, 22(4):477–482, 1998.

[34] Daigo Maruyama, Didier Bailly, and Gérald Carrier. High-quality mesh deformation

using quaternions for orthogonality preservation. *AIAA journal*, 52(12):2712–2729, 2014.

[35] EM Papoutsis-Kiachagias, N Magoulas, J Mueller, C Othmer, and KC Giannakoglou. Noise reduction in car aerodynamics using a surrogate objective function and the continuous adjoint method with wall functions. *Computers & Fluids*, 122:223–232, 2015.

[36] Hugo Gagnon and David Zingg. Two-level free-form deformation for high-fidelity aerodynamic shape optimization. In *12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference and 14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, page 5447, 2012.

[37] Jason E Hicken and David W Zingg. Aerodynamic optimization algorithm with integrated geometry parameterization and mesh movement. *AIAA journal*, 48(2):400–413, 2010.

[38] Edward Luke, Eric Collins, and Eric Blades. A fast mesh deformation method using explicit interpolation. *Journal of Computational Physics*, 231(2):586–601, 2012.

[39] Jibum Kim, Brian J Miller, and Suzanne M Shontz. A hybrid mesh deformation algorithm using anisotropic pdes and multiobjective mesh optimization. *Computers & Mathematics with Applications*, 70(8):1830–1851, 2015.

[40] Sahuck Oh. A new mesh moving technique for a fluid–structure interaction problem using mesh deformation energy minimization. *International Journal of Computational Methods*, 18(01):1950039, 2021.

[41] Keith Stein, Tayfun Tezduyar, and Richard Benney. Mesh moving techniques for fluid-structure interactions with large displacements. *J. Appl. Mech.*, 70(1):58–63, 2003.

[42] Per-Olof Persson and Jaime Peraire. Curved mesh generation and mesh refinement using lagrangian solid mechanics. In *47th AIAA aerospace sciences meeting including the new horizons forum and aerospace exposition*, page 949, 2009.

[43] Nicolas Barral, Edward Luke, and Frédéric Alauzet. Two mesh deformation methods coupled with a changing-connectivity moving mesh method for cfd applications. *Procedia Engineering*, 82:213–227, 2014.

[44] Adam Lamecki. A mesh deformation technique based on solid mechanics for parametric analysis of high-frequency devices with 3-d fem. *IEEE Transactions on Microwave Theory and Techniques*, 64(11):3400–3408, 2016.

[45] Patrícia Tonon, Rodolfo André Kuche Sanches, Kenji Takizawa, and Tayfun E Tezduyar. A linear-elasticity-based mesh moving method with no cycle-to-cycle accumulated distortion. *Computational Mechanics*, 67(2):413–434, 2021.

[46] Marwan Darwish and Fadl Moukalled. *The finite volume method in computational fluid dynamics: an advanced introduction with OpenFOAM® and Matlab®*. Springer, 2021.

A | Appendix A

Mesh quality parameters

In present work, the considered mesh quality parameters can be defined as follows:

- **maximum cell non-orthogonality:** The maximum angle between the normal of a face and the vector connecting the centroids of the two neighbour cells of the face. More maximum non-orthogonality indicates non-orthogonal cells which are not good for a good quality mesh.
- **maximum boundary non-orthogonality:** The maximum angle between the normal of the boundary face and the vector connecting the centroid of the neighbour cell and the center of the boundary face.
- **Maximum skewness:** The maximum distance between the centroid of a face and the point where the line connecting the two neighbour cell centroids and the line along the face meet. The less the maximum skewness, the better the mesh quality.
- **Cell face validity:** In a cell, minimum ratio of the sum of the area of the faces for which the dot product between the normal to the face and the vector connecting the face and cell centroids is positive and sum of the area of all the faces of a cell. This mesh quality parameter is not equal to 1 especially when the mesh consists of pyramid shaped, polygonal or convex cells, and when the angle between the normal to the face and the vector connecting face and cell centres is more than or equal to 90°. In a case with the cell face validity < 0.5, it can be considered that the cell has been inverted resulting in a failed mesh.

Average cell non-orthogonality is the average of the cell and boundary non-orthogonality all over the geometry, and minimum cell volume, as the name suggests, is the volume of the smallest cell by volume.

List of Figures

2.1 Design scheme of the overall solver. 15
2.2 2D representation of a generic geometry with the immediate neighbours. . 16
2.3 Computation of the face gradients using the cell gradients. 18
2.4 2D representation of a generic mesh at the boundary. 19
2.5 Bulk mesh near the boundary with different kinds of neighbours. 22
2.6 Interpolation from nodes to face centers. 24
2.7 2D representation of interpolation from cell centers to nodes in a volumetric mesh. 25

3.1 Relative error in cell gradients without and with boundary face contributions. 31
3.2 Relative errors in cell gradient vs cell ID's. 32
3.3 Gradient for a linear equation. 33
3.4 Face gradients without and with boundary face contributions. 34
3.5 Cell gradients and their relative errors with face contributions using Paraview. 36
3.6 Mesh morphing for a cube with 64k cells: outward deformation. 37
3.7 Mesh morphing for a cube with 64k cells: Negative deformation. 38
3.8 Zoomed figures of the morphed mesh close to the boundary. 38
3.9 Outward deformation using the multi-step approach as shown in Sec. 2.10. 40
3.10 Inward deformation using the multi-step approach as shown in Sec. 2.10. . 41

4.1 Distribution of elasticity coefficient E near the boundary of non-uniform cube. 46
4.2 Effect of inverse of volume exponent p on the Outward deformation. 48
4.3 Effect of inverse of distance exponent q on the Outward deformation. . . . 49
4.4 Effect of inverse of volume exponent p on the Inward deformation. 52
4.5 Effect of inverse of distance exponent q on the Inward deformation. 54
4.6 Effect of Poisson's ratio on the outward deformation. 56
4.7 Effect of Poisson's ratio on the inward deformation. 58

5.1 Outward deformation in the cube with a single step approach. 63
5.2 Inward deformation in the cube with a single step approach. 65

5.3 Outward deformation in the cube with a 10 step approach. 67
5.4 Inward deformation in the cube with a 10 step approach. 69

6.1 The wing of an airplane with the FFD box. 73
6.2 FFD box for linear and non-linear bends. 73
6.3 Angle of bend for a non-linear bend. 74
6.4 Mesh quality plot for bending in a wing. 75
6.5 Bending in a wing for linear and non-linear bends. 77
6.6 FFD box for a twist in the wing of an airplane. 78
6.7 Mesh quality plot for a twist in the wing of an airplane. 78
6.8 Twisting in a wing. 79
6.9 FFD box for a bend + twist in the wing of an airplane. 80
6.10 Mesh quality plot for a bend + twist in the wing of an airplane. 81
6.11 Non-linear bending + twisting in a wing. 82
6.12 FFD box for cambering the wing of an airplane. 83
6.13 Mesh quality plot for cambering in the wing of an airplane. 83
6.14 Cambering in the wing of an airplane. 84
6.15 Surface deformation in the industrial case of a car. 86
6.16 Volumetric mesh at a plane parallel to the symmetry plane. 87

List of Tables

1.1 Comparison of various mesh morphing techniques (extension of [3]). 7

3.1 Mesh quality in outward, inward and no deformation cases. 39
3.2 Mesh quality in outward deformation with different number of steps. 40
3.3 Mesh quality in inward deformation with different number of steps. 42

4.1 Mesh quality in outward deformation with different number of steps. 48
4.2 Mesh quality in outward deformation with different number of steps. 50
4.3 Mesh quality in outward deformation with different number of steps. 52
4.4 Mesh quality in outward deformation with different number of steps. 54
4.5 Mesh quality in outward deformation with different number of steps. 57
4.6 Mesh quality in outward deformation with different number of steps. 59

5.1 Mesh quality in outward deformation with a single step approach. 63
5.2 Mesh quality in inward deformation with a single step approach. 65
5.3 Mesh quality in outward deformation with a 10 step approach. 67
5.4 Mesh quality in inward deformation with a 10 step approach. 69

6.1 Mesh quality parameters for a non-uniform cube and a wing of an airplane. 71
6.2 Mesh quality parameters for a linear bending in the wing of an airplane. . 74
6.3 Mesh quality parameters for a non-linear bending in the wing of an airplane. 76
6.4 Mesh quality parameters for twisting in the wing of an airplane. 79
6.5 Mesh quality parameters for bending + twisting in the wing of an airplane. 81
6.6 Mesh quality parameters for cambering in the wing of an airplane. 84
6.7 Maximum deformation with a valid mesh using different techniques. 85
6.8 Mesh quality parameters for the industrial case of a car. 88

List of Symbols

Variable	Description	SI unit
$\boldsymbol{\sigma}$	stress	N/m^2
$\boldsymbol{\epsilon}$	strain	
$\boldsymbol{f}$	force	N
$\boldsymbol{E}$	Elasticity coefficient	N/m^2
$\boldsymbol{\nu}$	Poisson's ratio	
$\boldsymbol{V}$	Volume	m^3
$\boldsymbol{d}$	distance	m
$\boldsymbol{u_i}$	displacement field	m
$\boldsymbol{\vec{n}}$	normal vector	
$\boldsymbol{S}$	Area	m^2
$\boldsymbol{St}$	Stencil	
$\boldsymbol{Err_{rel}}$	relative error	
$\boldsymbol{R}$	residual	

List of Acronyms

Acronym	Description
CFD	computational fluid dynamics
FVM	finite volume method
FEM	finite element method
FSI	fluid-structure interaction
TFI	transfinite interpolation
IDW	inverse distance weighting
RBF	radial basis function
FFD	free-form deformation
GMRES	generalized minimal residual method

Acknowledgements

I would like to take this opportunity to thank my supervisor **Konstantinos Gkaragkounis** (Senior engineer, Optimad Srl) for his constant support and valuable guidance throughout the internship period. Without his contribution and support, the realization of this project would not have been possible. I would like to thank my colleagues at Optimad Srl **Rocco Arpa** (Senior Engineering, Optimad Srl) and **Edoardo Lombardi** (Senior Engineer, Optimad Srl) for their feedback and support throughout the internship period and helping to improve the quality of the overall work. I would like to also thank **Haysam Telib** (CEO, Optimad Srl) for believing in me and his constant motivation and valuable feedback on the work. I would like to take this opportunity to address my gratitude towards my professor guide **Prof. Fabio Inzoli** (Politecnico di Milano) and **Prof. Federico Piscaglia** (Politecnico di Milano) for their valuable feedback and advise throughout the thesis work. Without their contribution the quality of the thesis work could not be what it is. I would be missing a lot if not thanked to the incredible colleagues **Federico Gallizio** (Owner, Optimad Srl) and **Marco Cisternino** (Senior Engineer, Optimad Srl) for all the technical help and being incredibly nice to me. Last but not the least, I would like to thank the authors of the papers that I have cited in this work and the internet for answering most of the questions I faced.