ECCOMAS Congress 2016
VII European Congress on Computational Methods in Applied Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou, V. Plevris (eds.)
Crete Island, Greece, 5–10 June 2016

SPHERICITY: MESH OPTIMIZATION FOR ARBITRARY ELEMENT TOPOLOGY

Pavlos P. Alexias[1,2] **and Eugene De Villiers**[3]

[1]Engys S.R.L
Via del Follatoio 12, Trieste 34148, Italy
e-mail: p.alexias@engys.com

[2] NTUA - Parallel CFD & Optimization Unit (PCOpt)
Heroon Polytechniou 9, Zografou 15780, Greece
e-mail: p.alexias@engys.com

[3] Engys Ltd.
Studio 20, RVPB, John Archer Way, London, SW 18 3SX, U.K
e-mail: e.devilliers@engys.com

Keywords: Sphericity, Mesh Optimization, Constrained Optimization, Shape Optimization, BFGS.

Abstract. *In this paper, a new method for optimizing CFD meshes, based on the usage of a geometric quantity called Sphericity, is introduced. Sphericity is an element-wise property that provides a measure of the shapes deviation from a perfect sphere. When used as an objective in a gradient based framework, it results in a fast, scale invariant, optimization algorithm applicable to arbitrary finite volume element topologies. The methodology is heuristic in nature in that it does not consider known numerical quality metrics explicitly. Our study has however shown it to be exceptionally robust and effective in improving mesh quality, provided that grid topology does not introduce fundamentally concave cells.*

The main focus of this paper is the development of a robust optimization algorithm that can effectively handle complications arising from constraints imposed by surface interactions. Specifically, efficient heuristic and gradient based techniques, for handling difficult mesh topologies with acutely curved surfaces, feature edges and high aspect ratio layer cells are introduced. In combination with the base method, these measures ensure exceptional quality meshes even on industrial scale applications. As a final step, the method is integrated into a general purpose mesh motion framework, allowing the maintenance of high cell quality even during extreme deformation events. Examples and industrial applications in the context of adjoint based shape optimization are presented, illustrating the method's general effectiveness.

1 INTRODUCTION

In the field of computational solutions of partial differential equations (PDEs), the quality of the computational mesh plays a significant role in various aspects of the solution procedure. Important properties such as the rate of convergence of the solution, the fidelity of the results and the robustness of the algorithm are all impacted by the mesh. In this context, the development of an efficient and effective mesh quality optimization algorithm, can provide significant benefits, not just in the context of static meshes, but also for moving mesh problems where degradation of quality is often a severe constraint.

The quality of the mesh can be evaluated using element-wise quality metrics such as, non-inversion, cell skewness, non-orthogonality, cell determinant, volume deviation etc. The mesh optimization algorithm that is presented in this paper aims to calculate a new (optimum) set of mesh vertex positions inside the computational domain that improves these mesh quality metrics. The algorithm does not alter the mesh topology, i.e. no merging, swapping, insertion or subdivision of mesh elements take place. Attempting to directly optimize all important mesh quality metrics simultaneously would lead to an extremely costly and complex multi-objective optimization problem that would require careful calibration. In an attempt to circumvent these issues we introduce without further justification a geometric quantity termed sphericity as the main quality metric and use its maximization as the objective for the mesh optimization. Since sphericity is scale invariant and we only have to optimize for a single objective the process will be much more efficient and robust than would otherwise be the case. It is shown empirically that the maximization of sphericity inside the computational domain results a global improvement to all other quality metrics, provided certain conditions are met.

While this paper deals with the mesh optimization problem specifically for finite volume (FV) meshes, the method can be easily generalized for other areas of applications, such as Finite Elements and surface triangulation problems, with the same efficacy.

2 OPTIMIZATION ALGORITHM

2.1 Sphericity

Sphericity is an element-wise geometric quantity which will be used as the objective function on the mesh optimization problem. Sphericity defines how spherical a geometrical object is and is given by the formula

$$Sph = \frac{\left(\frac{6}{\pi}V_o\right)^{1/3}}{\left(\frac{1}{\pi}S_o\right)^{1/2}} \tag{1}$$

where V_o is the volume and S_o is the surface of the object. In other words sphericity is the ratio between the diameter of a sphere, that has the same volume with the object, and the diameter of a sphere that has the same surface area as the object (considering that for a sphere holds $S = \pi^{1/3}\,(6V)^{2/3}$). Given this the sphericity of a sphere is unity by definition and unity is the largest value that any three dimensional object can have. It therefore follows that maximizing this quantity will make elements more spherical. The most important property of spherical elements is isotropy. Highly anisotropic elements can be in general elements with high aspect ratio or/and skewness values that, in a context of computational solutions of PDEs, can affect the quality of the results or even the total convergence of the solution.

In the case of a finite volume mesh, each cell will have a sphericity value calculated as a function of the cell's surface and volume. The goal is to re-position the vertices of the mesh such that the cells will have the maximum sphericity. In order to do so it is necessary to calculate the derivative of sphericity w.r.t. the position of the points that constitute a cell.

2.2 Sphericity Differentiation

Assuming that a point P is surrounded by M cells, the total sensitivity derivative G_P of that point is the arithmetic mean of the sensitivities from all contributing cells

$$G_P = \frac{1}{M} \sum_{n=1}^{M} \frac{dSph_n}{dP} \tag{2}$$

For each cell that uses point P, the first derivative of sphericity with respect to the position of a point P in the cell is,

$$\frac{dSph_n}{dP} = \frac{\partial \frac{\left(\frac{6}{\pi} V_c\right)^{1/3}}{\left(\frac{1}{\pi} S_c\right)^{1/2}}}{\partial P} = -\frac{6^{1/3}\pi^{1/6} V_c(P)^{1/3} S_c'(P)}{2S_c(P)^{3/2}} + \frac{6^{1/3}\pi^{1/6} V_c'(P)}{3\sqrt{Sc(P)} V_c(P)^{2/3}} \tag{3}$$

where $S_c'(P)$ and $V_c'(P)$ are the partial derivatives of the cell surface and cell volume w.r.t. the position of the point P respectively. Taking into consideration that the cell surface can be written as the sum of the areas of the facets surfaces that constitute the cell, i.e.

$$S_c(P) = \sum_{k=1}^{K} S_f(P)_k \tag{4}$$

where K is the number of faces the cell has. Is it therefore sufficient to calculate the partial derivative of each face w.r.t. the point location. For doing this, the polygon of the face is decomposed into N triangles by defining an approximate face center $C_f(P)$ and forming a triangle from each edge of the polygon. Then it follows that

$$S_f(P) = \frac{1}{2} \sum_{i=1}^{I-1} \sqrt{\left[(P_i - C_f(P)) \times (P_{i+1} - C_f(P))\right]^2} \tag{5}$$

where I is the total number of points of the face polygon. It can be noticed that the face surface is a function of the point position and the face centre which in turn is also a function of the point position. Using the chain rule follows that,

$$\frac{dS_f(P, C_f(P))}{dP} = \frac{\partial S_f}{\partial P} + \frac{\partial S_f}{\partial C_f} \frac{dC_f(P)}{dP} \tag{6}$$

The last step is to write the face center as a function of the points that constitute the face and compute the derivative. In order to avoid cyclical dependencies, the face center is defined as the weighted sum of the edge centers,

$$C_f(P) = \frac{\frac{1}{2} \sum_{i=1}^{I-1} (P_i + P_{i+1}) \sqrt{(P_i - P_{i+1})^2}}{\sum_{i=1}^{I-1} \sqrt{(P_i - P_{i+1})^2}} \tag{7}$$

and which can be differentiated to produce,

$$\frac{dC_f(P)}{dP} = \frac{\frac{1}{2} \frac{\partial \sum_{i=1}^{I-1} (P_i + P_{i+1}) \sqrt{(P_i - P_{i+1})^2}}{\partial P}}{\sum_{i=1}^{I-1} \sqrt{(P_i - P_{i+1})^2}} - \frac{\frac{1}{2} \sum_{i=1}^{I-1} (P_i + P_{i+1}) \sqrt{(P_i - P_{i+1})^2}}{\left[\sum_{i=1}^{I-1} \sqrt{(P_i - P_{i+1})^2}\right]^2} \cdot \frac{\partial \sum_{i=1}^{I-1} \sqrt{(P_i - P_{i+1})^2}}{\partial P} \tag{8}$$

Following the same approach for the derivation of the cell volume, the element is decomposed into K (number of faces) pyramid elements taking as bases the polygonal faces and as the apex the cell center. It thus holds that,

$$V_c(P) = \sum_{k=1}^{K} V_p(P)_k \tag{9}$$

where

$$V_p(P) = \frac{1}{6} \sum_{i=1}^{I-1} (C_f(P) - C_c) \cdot (P_i - C_c) \times (P_{i+1} - C_c)) \tag{10}$$

Since the face center derivative w.r.t. the point locations has already been computed, the calculation of the volume derivative becomes almost straightforward. Having calculated all the component derivatives the total sphericity gradient can now be calculated using equation 3.

2.3 Optimization Method

Having obtained the objective sensitivity derivatives of all the points of the mesh, the next step is to find the new set of positions that maximizes the sphericity value. Thus, for each optimization step, the new position of a point can be calculated as,

$$\vec{P}^{new} = \vec{P}^{old} + \alpha \cdot \vec{G}_P \tag{11}$$

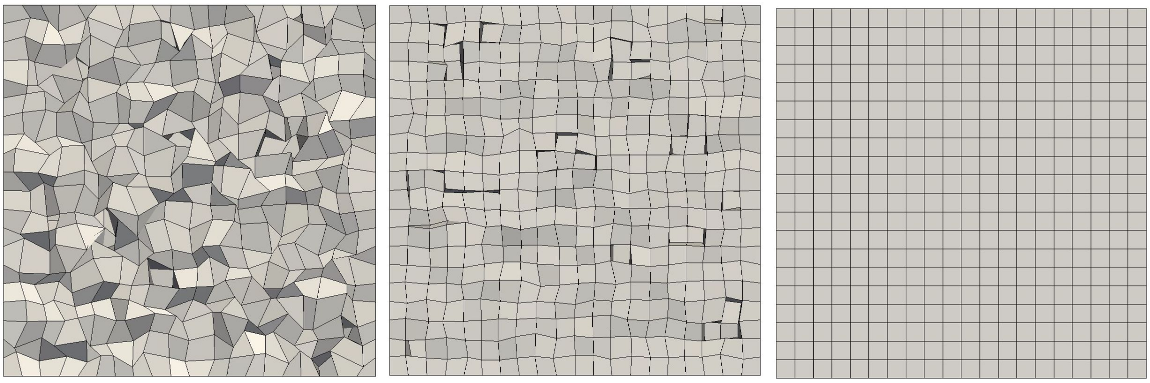
where α is the step length of the optimization cycle. Using a constant value of α is not optimal for the proposed application due to large variations in cell sizes encountered in different CFD meshes (up to 20 orders of magnitude or more). It is therefore necessary to have an adaptive step length. This can be achieved effectively by using second derivatives which also has the added benefit of speeding up the convergence. Based on Nocedal [3] a limited memory BFGS algorithm (LM-BFGS) is constructed for building and updating the inverse Hessian matrix of the second derivatives. Thus, equation 11 can be written as

$$\vec{P}^{new} = \vec{P}^{old} + \lambda H_P^{-1} \cdot \vec{G}_P \tag{12}$$

In general when using Newton and quasi-Newton methods the objective function has to be convex so the optimization algorithm can converge to an optimum value. In other words, the inverse Hessian matrix has to be positive definite. It has been observed that the sphericity function generally behaves in a convex way except from cases where there are negative oriented mesh cells. Therefore, instead of building one Hessian matrix for each point of the mesh which is being optimized, the goal is to create a global Hessian taking into consideration the effect of the displacement of one point, to all other points inside the mesh. Furthermore, the equation 12 can be written as

$$\begin{bmatrix} \vec{P}_1^{new} \\ \vdots \\ \vec{P}_n^{new} \end{bmatrix} = \begin{bmatrix} \vec{P}_1^{old} \\ \vdots \\ \vec{P}_n^{old} \end{bmatrix} + \begin{bmatrix} \lambda_1 & \cdots & \cdots \\ \vdots & \ddots & \vdots \\ \cdots & \cdots & \lambda_n \end{bmatrix} \cdot \mathbf{H}_{\mathbf{global}}^{-1} \cdot \begin{bmatrix} \vec{G}_1 \\ \vdots \\ \vec{G}_n \end{bmatrix} \tag{13}$$

where the H_{global}^{-1} is an $n \times n$ matrix with n to be the total number of points the mesh has. The global matrix is more robust is a sense that is always positive definite which allows the optimization algorithm to reliable converge to a new optimum set of point positions. At last, is



(a) Initial Mesh (b) 2nd Optimization Cycle (c) Optimized Mesh

Figure 1: Mesh optimization based on sphericity value, starting from a low quality mesh (left) and ending up to the optimized one with fully orthogonal cells(right)

fundamental the Wolfe conditions [4] to be satisfied at each and every step of the optimization procedure. This is achieved by assigning a proper step length value λ which will guarantee that the sphericity value, moving towards the direction $\mathbf{H}^{-1}\tilde{\mathbf{G}}$, will actually be increased. Figure 2.3 shows three snapshots of a simple mesh optimization case. Starting from low quality mesh and executing the optimization algorithm, keeping in the same time fixed the boundary points, it results a mesh with fully orthogonal cell elements.

2.4 Constraints

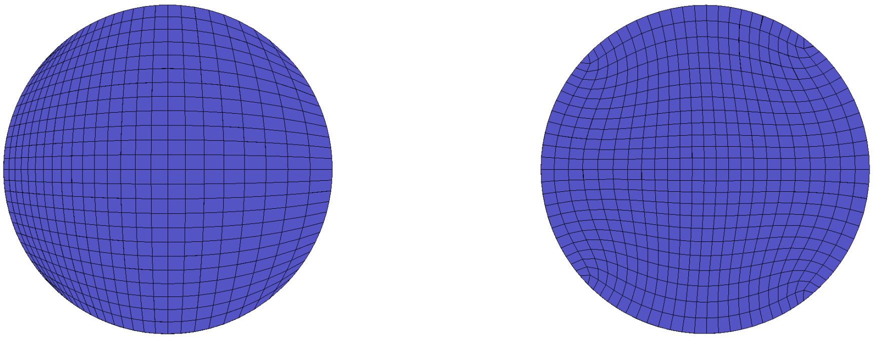
2.4.1 Boundary Points

Interior points are much easier to involve in the optimization as, in general, we want to preserve the original shape of the domain, which means that boundary points have to be constrained to the original surface. As shown on the figure 2.3, if the boundary points are constrained with zero displacement, then the optimization potential of the mesh can be severely curtailed. In order to allow for effective optimization it is necessary to let the boundary points slide on the surface, while respecting essential parts of the geometry like feature edges and feature points. A feature edge in this context is defined as a boundary edge in which, neighbouring faces have normal vectors that differ by more than a threshold value and a feature point is a boundary point that is connected to more than two feature edges.

Supposing that for a boundary point on the mesh the displacement direction vector $\vec{r}$ has been calculated according to the sphericity sensitivity. We decompose this direction into two components, one normal to the surface and the other parallel to the tangent plane of the surface. By eliminating the vertical component, each point on the boundary is allowed to move only in the tangent direction $\vec{r_t}$. Since the surface is often curved the point then has to be projected back onto the surface of the initial geometry. Figure 2.4.1 illustrates an optimization algorithm that is applied in a low quality mesh with curved edges where the boundary nodes are allowed to slide on the surface.

2.4.2 Layer Cells

In order to resolve highly directional anisotropic gradients many applications employ grids which are likewise directionally biased. In CFD analysis this phenomenon often manifests



(a) Initial Mesh (b) Optimized Mesh

Figure 2: Mesh optimization allowing the surface nodes to slide on the surface

itself in the use of stratified layers of cells adjacent to solid boundaries, which are used to capture high wall-normal gradients in this region. Layer cells typically have high aspect ratios, with wall normal dimensions being much smaller that tangential. Figure 3(a) illustrates an example of layers cells close to the boundary wall. Due to their high aspect ratio layer cells have very small sphericity values. Allowing the optimization algorithm to freely deform layer cells will result in their extrusion, as illustrated in figure 3(b), which will significantly reduce wall normal resolution. It is therefore important for the optimization procedure to maintain anisotropic aspect of layer cells. To do this, the displacement of the layer cell points, towards the growing direction of the layer cells, is constrained by eliminating the derivative component in that direction. Since the method is scale invariant the implementation of such a constraint inside the optimization framework can introduce a shrinking cell problem: due to its inability to expand the layers in the normal direction, the algorithm will try to reduce all other dimensions. If left unchecked and if the mesh topology allows, this will result in cells with very small volumes. Figure 3(c) shows the result of a mesh optimization procedure with constrained layer cells. The shrunk cells create problems both in terms of volume deviation and in the skewness of the layer cells.

To overcome this problem, a supplementary objective, in the form of volume uniformity, is introduced inside the optimization framework, to improve the behaviour in cases with high aspect ratio layer cells. Supposing again that a point of the computational domain neighbours M cells, the volume deviation of that point is given by the formula

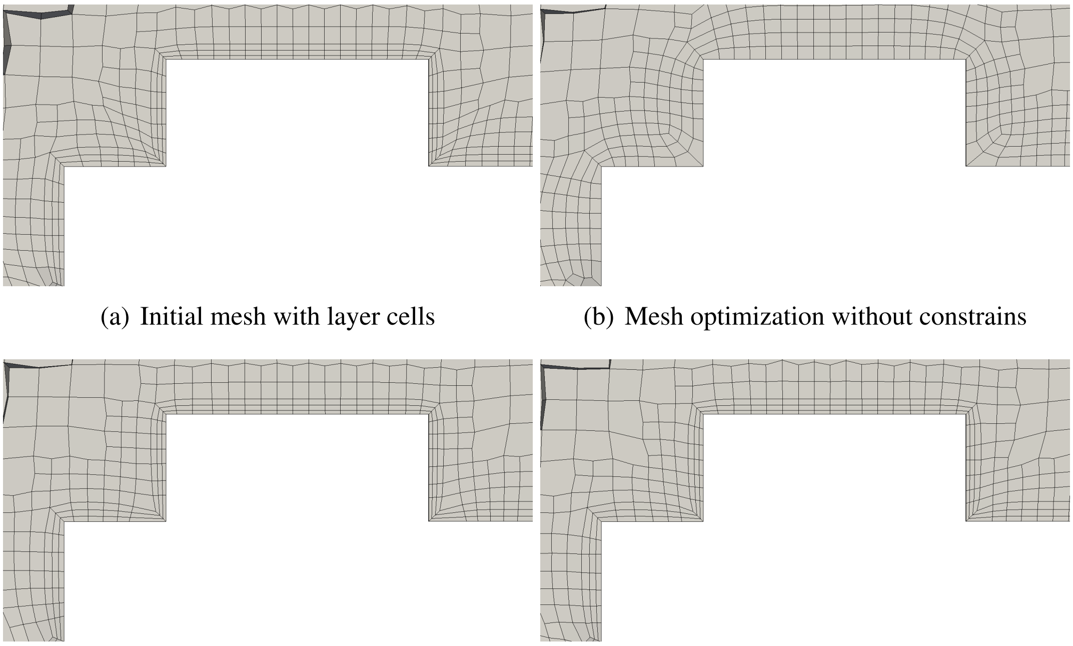
$$V_{dev}(P) = \frac{1}{M} \sum_{i=1}^{M} \left(\frac{V_i(P) - \bar{V}}{\bar{V}} \right)^2 \tag{14}$$

where $\bar{V}$ is the mean volume of the surrounding cells. Its easy to see that the mean volume is independent of the position of the point P. Thus the differentiation of the volume deviation is straightforward,

$$\frac{dV_{dev}(P)}{dP} = \frac{2}{M} \sum_{i=1}^{M} \left[\frac{V_i(P) - \bar{V}}{\bar{V}} \cdot \frac{dV_i(P)}{dP} \right] \tag{15}$$

The new optimization function now becomes

$$F_{obj} = Sph \cdot (1 - V_{dev}) \tag{16}$$



(a) Initial mesh with layer cells

(b) Mesh optimization without constrains

(c) Mesh optimization with constrains

(d) Mesh optimization including volume deviation

Figure 3: Mesh optimization procedure starting from the initial mesh (a), firstly without constraints (b), afterwards with layer cells constraints (c) and finally combining constraints with volume deviation derivative (d).

with a derivative of

$$\frac{dF_{obj}}{dP} = \frac{dSph}{dP} \cdot (1 - V_{dev}) - Sph \cdot \frac{dV_{dev}}{dP} \tag{17}$$

Be noted that the addition of the new objective function does not raise the computational cost of the algorithm, since the calculation of the derivative of the volume deviation depends only on the cell volume derivative (which had been already calculated in Eq.10). Figure 3(d) illustrates the optimized computational mesh after the introduction of the volume deviation objective function. As it can be seen the additional derivative maintains the same volume on every stack of layer cells and prevents also the formation of high skewness cells.

2.5 CFD Mesh Optimization Applications

In this section the effectiveness of the method for improving the mesh quality in large scale test cases will be examined. The three quality metrics that will be used to compare the non optimized with the optimized mesh are non-orthogonality, skewness and cell determinant. Non-orthogonality of a face is the angle between the line that connects two cell centres and the normal of their common face. Skewness measures the distance between the intersection of the line connecting two cell centres with their common face and the centre of that face and finally with the determinant we are referring to the determinant of the cell's Jacobian matrix. Both for non-orthogonality and skewness their optimum value is zero, instead the optimum value for the determinant is unity.

Considering the previous example (figure 2.4.2) of the optimization algorithm, table 2.5 contains the maximum, minimum and mean values of those three quality metrics for the meshes that are displayed in figures 3(a), fig. 3(c) and 3(d). It can be noticed that the introduction of the volume deviation affects mostly the maximum/minimum values of non-orthogonality and skewness.

Quality Metric	Max. Value	Min. Value	Mean Value
Non-Orthogonality	56.18^o	0^o	8.04^o
Skewness	2.63	0	0.19
Determinant	1	0.243	0.899

Table 1: Quality metrics for the initial mesh with layers (fig. 3(a)).

Quality Metric	Max. Value	Min. Value	Mean Value
Non-Orthogonality	56.32^o	0^o	7.14^o
Skewness	3.31	0	0.12
Determinant	1	0.257	0.912

Table 2: Quality metrics for the optimized mesh with layers (fig. 3(c)) without introducing the volume deviation objective.

Quality Metric	Max. Value	Min. Value	Mean Value
Non-Orthogonality	50.16^o	0^o	7.02^o
Skewness	3.01	0	0.11
Determinant	1	0.269	0.901

Table 3: Quality metrics for the optimized mesh with layers (fig. 3(d)) introducing also the volume deviation objective.

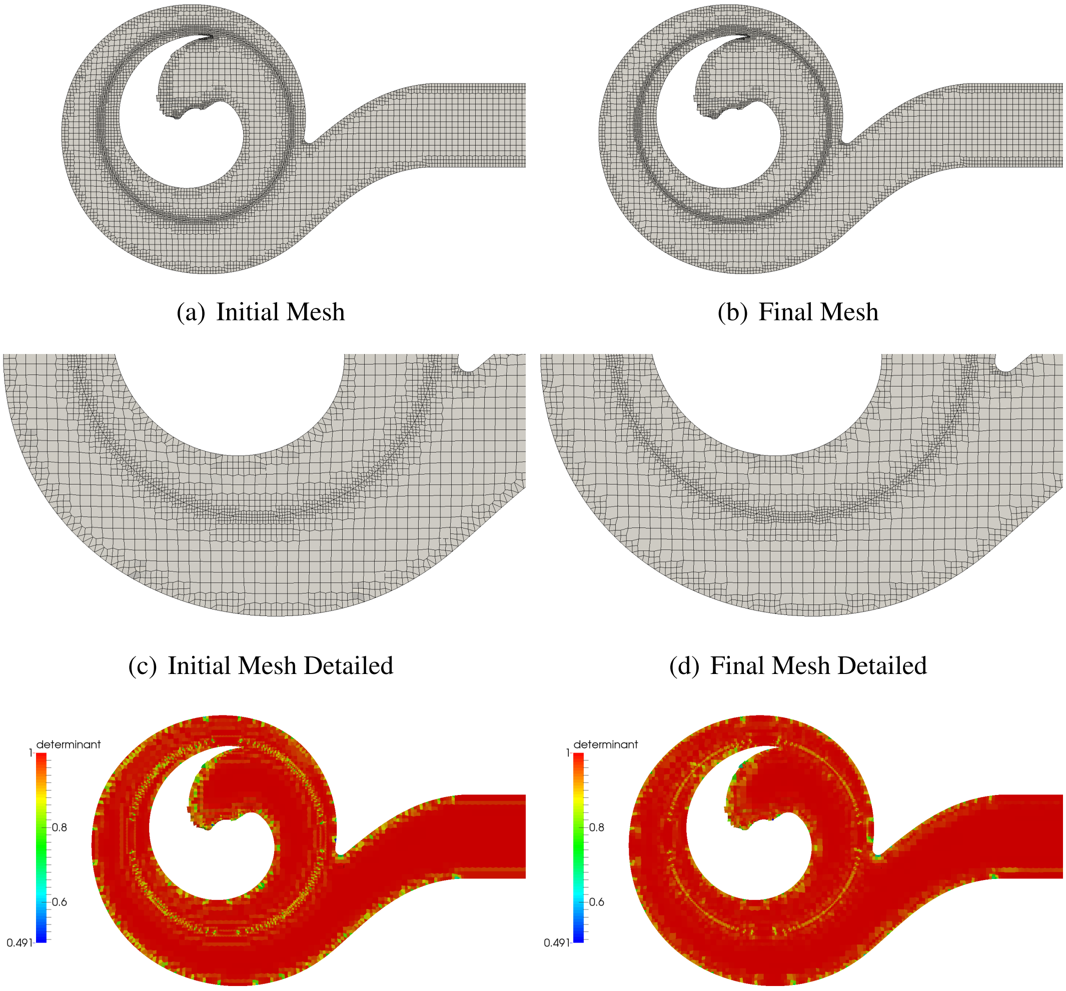
Quality Metrics	Max.Value		Min. Value		Mean Value	
	Before	After	Before	After	Before	After
Non-Orthogonality	69.27^o	68.59^o	0^o	0^o	7.57^o	7.22^o
Skewness	4.89	3.60	0	0	0.84	0.79
Determinant	1	1	0.621	0.657	0.979	0.984

Table 4: Quality metrics before and after the mesh optimization of the single channel pump.

Figure 2.5 shows the mesh optimization results for a single channel pump. The mesh is unstructured, having around 0.5 million cell elements consisting primarily of tetrahedra, prisms and other types of polyhedra. Images 4(a) to 4(d) show the mesh geometry before and after the optimization, while figures 4(e) and 4(f) show the cell determinant. Tables 2.5 show the maximum, minimum and the mean values of the quality metrics before and after the optimization. It can be seen that the maximization of the sphericity leads also to the maximization of the three quantities inside the domain. It should be noted that the algorithm has been calibrated to focus more on improvement of cells near the boundary, where the mesh is finer and the accuracy requirements are more demanding, than those further toward the interior of the domain.

The next case is the Ahmed-Body car (figure 2.5). Table 2.5 contain the quality metric quantities before and after the optimization of the computational mesh (1 million elements). Figures 5(a) and 5(b) show the skewness in the car boundary faces before and after the optimization.

It is important to examine how the improvement of the computational mesh affects the CFD calculations in terms of convergence speed. We solve the flow around the Ahmed-Body using the incompressible OpenFOAM solver (simpleFoam), both for the optimized and the non-



(a) Initial Mesh

(b) Final Mesh

(c) Initial Mesh Detailed

(d) Final Mesh Detailed

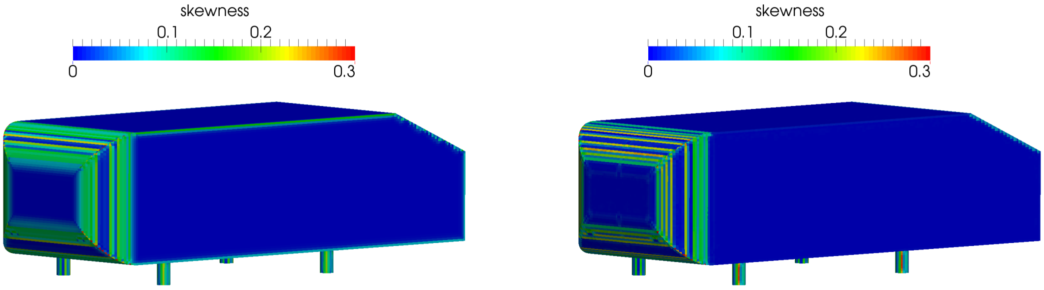
(e) Initial Mesh Cell Determinant

(f) Initial Mesh Cell Determinant

Figure 4: Mesh optimization results on a pump with half million cell elements. Figures (a) and (c) illustrate the geometry before optimization while (b) and (d) illustrate the geometry after the optimization. Finally figures (e) and (f) display the cell determinant before and after the optimization.

Quality Metrics	Max.Value		Min. Value		Mean Value	
	Before	After	Before	After	Before	After
Non-Orthogonality	48.68^o	47.23^o	0^o	0^o	3.93^o	3.88^o
Skewness	3.16	2.23	0	0	0.51	0.48
Determinant	1	1	0.617	0.749	0.990	0.991

Table 5: Quality metrics before and after the mesh optimization of the single Ahmed-Body.



(a) Initial Boundary Face Skewness

(b) Final Boundary Face Skewness

Figure 5: Face skewness on the boundary of the S-Bend duct before and after the sphericity optimization.

optimized mesh. In the optimized mesh the solver needs 6.5% less iterations to achieve convergence which reflects a 7.3% reduction on CPU time.

3 MORPHING FRAMEWORK INTEGRATION

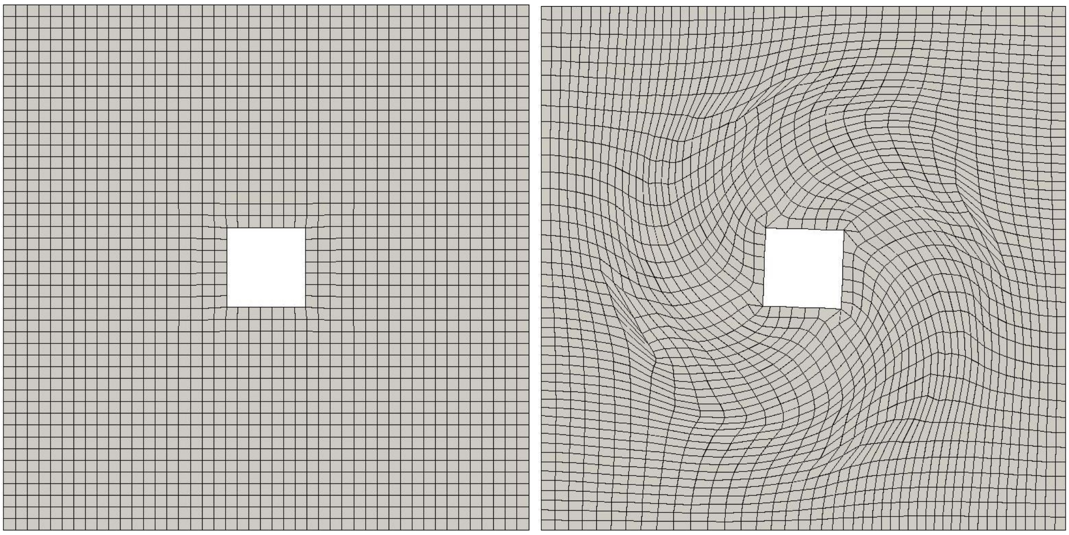
It has been demonstrated that the optimization algorithm can be used for improving the quality metrics on the computational mesh, leading to more accurate and reliable results. However the method can also be employed during dynamic mesh motion to prevent the occurrence of badly deformed elements, resulting in greater robustness and efficacy even during extreme deformation. In the context CFD analysis, a mesh displacement algorithm is indispensable in cases with moving boundaries, like shape optimization, aero-elasticity etc.

Many mesh displacement algorithms have been developed so far following a variety of approaches, like elastic medium analogy [8], [9] spring analogy [10] and Radial Basis Functions (RBF) [11] methods. In this paper the mesh displacement algorithm is based on the existence of an objective function (Sphericity) which quantifies the mesh quality. Solving the optimization problem for the maximization of the sphericity value, results in the new positions of the points inside the mesh.

Figure 3 illustrates the method's ability to maintain high cell quality in extreme deformation conditions such as a 90 degree box rotation. The rotation was made in 45 steps of 2 degrees running 50 optimization cycles for each rotation step. Further rotation of the box (over 93 degrees), will lead to an invalid mesh, primarily due to the fact that the corner vertices are not allowed to slip. Table 3 shows the quality metrics before and after the box rotation. The final mesh even after the extreme deformation maintains very low skewness values. Contrariwise the non-orthogonality has significantly increase, having four faces with non-orthogonality value over 70 degrees (a threshold value over which can be consider that non-orthogonal faces are starting to have impact in the CFD solution).

Quality Metrics	Max.Value		Min. Value		Mean Value	
	Before	After	Before	After	Before	After
Non-Orthogonality	5.77^o	77.18^o	0^o	0.01^o	0.23^o	16.54^o
Skewness	0.07	2.11	0	0	~ 0	0.16
Determinant	1	1	0.985	0.041	0.999	0.861

Table 6: Quality metrics before and after the rotation of the box by 90 degrees.



(a) Initial box position

(b) 90^o degrees rotating box

Figure 6: Mesh displacement around a box using sphericity optimization applying a 90^o degrees rotation.

3.1 Adjoint Shape Optimization

In the context of gradient based optimization, the adjoint method can be considered the most computational effective as the gradient calculation of the objective function is independent of the number of the design variables. That allows the creation of a very rich design space which leads to better optimum solutions. The adjoint method was introduced by Pironneau [5] and consequently Jameson [6] and Giles [7] et al. extended the method inside the scope of shape optimization. An in depth analysis of the adjoint method goes beyond the scope of this paper, thus only a brief description of the continuous adjoint formulation will be presented. For a more circumstantial analysis one can refer to [1], [2].

3.1.1 Continuous Adjoint Method

Further on, Einstein's summation convention applies for the lower-case indices, unless declared differently. Assuming a laminar flow of an incompressible fluid, the Navier-Stokes equations can be written as,

$$R^p = -\frac{\partial v_j}{\partial x_j} = 0 \tag{18a}$$

$$R_i^v = v_j \frac{\partial v_i}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\right] + \frac{\partial p}{\partial x_i} = 0 \qquad i = 1,2,3 \tag{18b}$$

where v_i is the velocity in direction of the Cartesian coordinates, p is the static pressure divided by the fluid's density ρ and ν is the kinematic viscosity. Afterwards a general objective function F is defined containing both surface S and volume Ω integrals,

$$F = \int_S F_s dS + \int_\Omega F_\Omega d\Omega \tag{19}$$

then F is augmented by the state equations leading to

$$F_{aug} = F + \int_\Omega u_i R_i^v d\Omega + \int_\Omega q R^p d\Omega \tag{20}$$

note that since the Navier-Stokes equations are satisfied it holds that $F_{aug} = F$.

The differentiating of the augmented objective function F_{aug} arises the adjoint equations for incompressible laminar flow.

$$R^q = \frac{\partial u_i}{\partial x_i} - \frac{\partial F}{\partial p} = 0 \tag{21a}$$

$$R_i^u = -v_j \frac{\partial u_i}{\partial x_j} + u_j \frac{\partial v_j}{\partial x_i} - \nu \frac{\partial}{\partial x_j}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) + \frac{\partial q}{\partial x_i} + \frac{\partial F}{\partial v_i} = 0 \tag{21b}$$

After solving the adjoint equations (which are similar to the Primal ones 18b,18a), the surface sensitivities of the objective function w.r.t. the surface normal of each surface node can be calculated as

$$G = \frac{\delta F_{aug}}{\delta b} = -\int_{S_w} \left[\nu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - q n_i\right] \frac{\partial v_j}{\partial x_k} \frac{\partial x_k}{\partial b} dS \tag{22}$$

3.1.2 Adjoint Sensitivities Smoothing

The design variables in this shape optimization problem are the surface normal displacements of each boundary node. Any oscillation (noise) in the sensitivity derivatives can lead to impractical, non-manufacturable, shapes which makes it necessary to apply filters that eliminate these oscillations. Based on [12] the filtered distribution of the sensitivity derivatives $\bar{G}$ is given by the convolution

$$\bar{G}(\xi) = \int_{S_w} K(r) G(\xi - r) dr, \qquad \forall \xi \epsilon S_w, \tag{23}$$

where r is the filter radius and ξ is the curved boundary coordinate. The filtered sensitivities can be calculated solving the equation

$$\bar{G} - \frac{\partial}{\partial \xi}\left(\varepsilon \frac{\partial \bar{G}}{\partial \xi}\right) = G \tag{24}$$

where ε is the smoothing intensity.

4 APPLICATIONS

4.1 Optimization of the S-Bend Duct

The final application, is the adjoint-based shape optimization of the S-Bend duct. The aim is to modify the shape of the duct so as to reduce the pressure losses while maintaining the geometry of the inlet and the outlet boundary. The flow is laminar with a fully developed profile

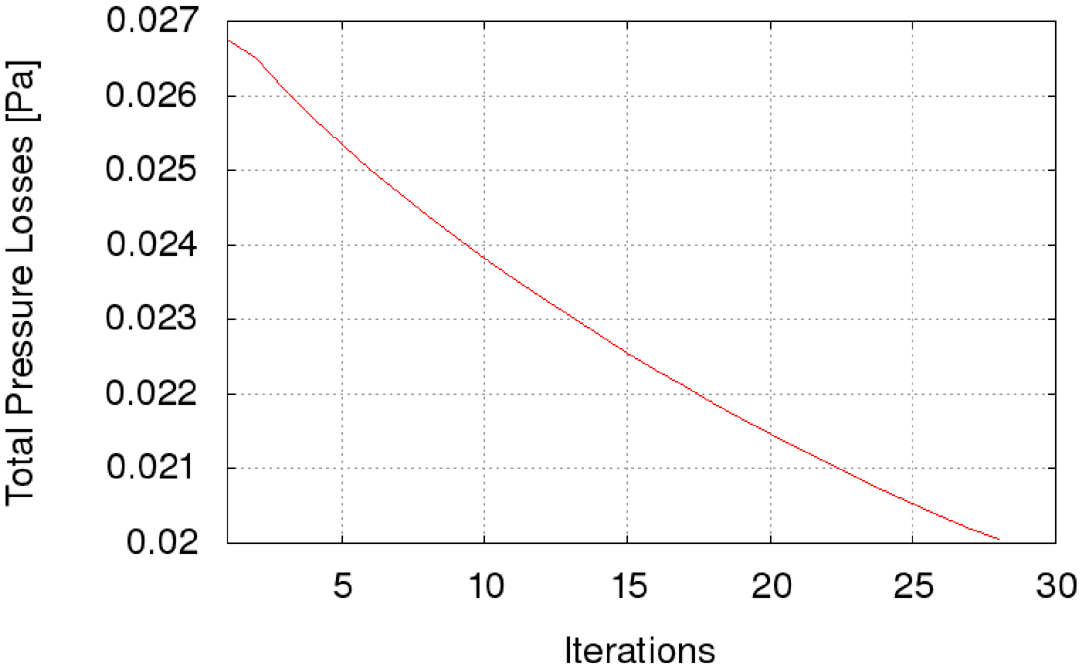
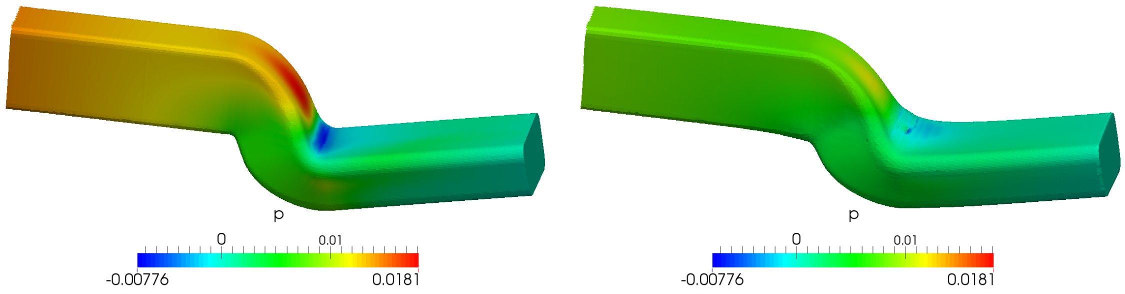


Figure 7: Total pressure losses for each shape optimization step.

at the inlet of the duct. Figure 4.1 illustrates the total pressure losses for every optimization step. After 28 optimization steps, the pressure losses were reduced by 25.01%. Figure 4.1 illustrates the geometry and the pressure distribution before and after the shape optimization of the duct, while figure 4.1 illustrates the difference between initial and smoothed sensitivities.

Table 4.1 shows the quality metrics for the initial and the final mesh. A high increase can be noticed at skewness metric having 3 faces over the acceptable skew value (≥ 4). Those faces were also incorrectly oriented preventing us to proceed further in to the optimization procedure.

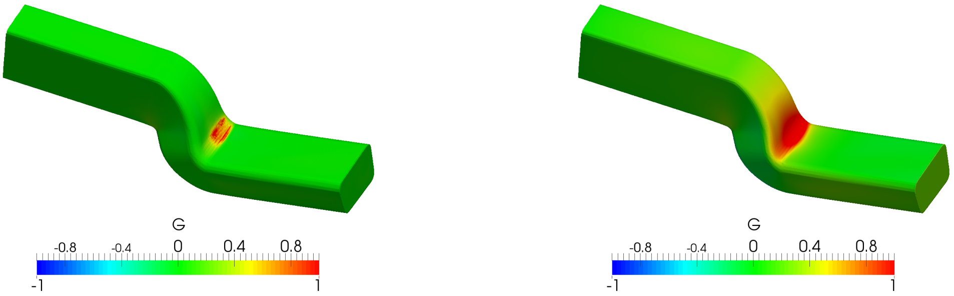
Using the mesh optimization algorithm as part of a mesh displacement procedure, maintains the quality of the mesh throughout multiple deformation steps. This reduces or removes the need for re-meshing after each optimization cycle, thus speeding up and improving the automation



(a) Initial shape and pressure distribution

(b) Optimized shape and pressure distribution

Figure 8: Shape geometry and pressure distribution before and after the adjoint shape optimization.



(a) Initial Sensitivity Map

(b) Smoothed Sensitivity Map

Figure 9: Initial and smoothed sensitivity map calculated with the adjoint method.

Quality Metrics	Max.Value		Min. Value		Mean Value	
	Before	After	Before	After	Before	After
Non-Orthogonality	60.09^o	73.59^o	0^o	0.01^o	2.47^o	5.26^o
Skewness	2.25	11.23	0	0	0.09	0.13
Determinant	1	1	0.118	0.014	0.999	0.990

Table 7: Quality metrics before and after the shape optimization of the S-Bend duct.

of the procedure.

5 CONCLUSIONS

An algorithm for optimizing finite volume meshes with arbitrary element topology was developed. The algorithm is based on a new heuristic quality metric termed sphericity. The maximization of sphericity inside the computational domain consistently produces improvement of all other quality metrics (skewness, non-orthogonality etc.) during testing, leading to more reliable results and solution times. The algorithm combines hand a derived gradient sphericity with a LM-BFGS method to speed up the solution. Heuristic techniques for limiting cell shrinkage were applied to improve effectiveness on meshes with mixed element topology and highly anisotropic cells. Basic constraints are employed to allow flexibility in the context of boundaries and feature elements. The algorithm was successfully employed in the context of adjoint based shape optimization of an S-bend Duct case.

ACKNOWLEDGEMENTS

The first author is an Early State Researcher participating in ITN *Industrial optimal design using adjoint CFD (IODA)* MSCA-ITN-2014-ETN: Marie Sklodowska-Curie Innovative Training Networks, Grant Agreement No 642959.

REFERENCES

[1] Zymaris, A., Papadimitriou, D., Giannakoglou, K., Othmer, C. Continuous adjoint approach to the Spalart- Allmaras turbulence model for incompressible flows. *Computers & Fluids,* 38(8):15281538, 2009.

[2] Zymaris, A. S., Papadimitriou, D. I., Giannakoglou, K. C. and Othmer C. Adjoint wall

functions: A new concept for use in aerodynamic shape optimization. *J. Comp. Phys.,* 229(13), 52285245., 2010.

[3] J. Nocedal Updating Quasi-Newton Matrices with Limited Storage *Mathematics of Computation,* Vol. 35, No. 151. (1980), pp. 773-782.

[4] P.Wolfe Convergence Conditions for Ascent Methods *SIAM Review 11* (2): 226000

[5] O. Pironneau On optimum design in fluid mechanics. *J. Fluid Mech.*, 64(1):97110, 1974

[6] Jameson, A. Optimum aerodynamic design using CFD and control theory. *AIAA-1995-1729-CP,* 1995.

[7] Giles, M. B., Pierce, N. A. An introduction to the adjoint approach to design. *Flow, Turbul. Combust.*, 65:393415, 2001

[8] T. Baker and P. A. Cavallo. Dynamic adaptation for deforming tetrahedral meshes. *AIAA Paper 1999-3253,* 1999.

[9] Amirante, D. and Hills, N.J. and Barnes, C.J., A moving mesh algorithm for aero-thermo-mechanical modelling in turbomachinery, *International Journal for Numerical Methods in Fluids*, vol. 70, 1118–1138, 2012

[10] J. Batina Unsteady Euler algorithms with unstructured dynamic mesh for complex-aircraft aerodynamic analysis. *AIAA Journal,* 29(3):327-333, 200.

[11] D. Sieger, S. Menzel and M. Botsch. RBF Morphing Techniques for Simulation-based Design Optimization. *Engineering with Computers,* 30(2):161174, 2014

[12] Stuck, A. and Rung, T., Filtered Gradients for Adjoint-based Shape Optimisation *20th AIAA Computational Fluid Dynamics Conference* 27 - 30 June 2011, Honolulu, Hawaii