

LEVEL-SET BASED TOPOLOGY OPTIMIZATION USING THE CONTINUOUS ADJOINT METHOD

Georgios K. Karpouzas¹, Eugene De Villiers²

^{1,2}Engys Ltd. - London SW18 3SX, UK

¹NTUA - Parallel CFD & Optimization Unit (PCOpt), Athens 15780, Greece

e-mail: {g.karpouzas, e.devilliers}@engys.com

Keywords: Optimization, Continuous Adjoint Method, Topology Optimization, Level-set Method, Sensitivity Statistical Analysis, One-Shot

Abstract. *In this paper, the development and application of the continuous adjoint of the incompressible Navier-Stokes coupled with the level-set method applied to topology optimization, is presented. The main focus of the paper is the representation and the evolution of the surface (interface between solid and fluid) via the level-set method. In this approach, the design variables are the signed distances between the cell centers and the interface. In the first step, the surface sensitivity derivatives are extended towards the normal direction of the interface, by solving the velocity extension equation. In the second step, the interface is evolved by solving a transport equation. After the evolution, the level-set field is no longer a signed distance field and as a result, has to be reinitialised as part of the final step. The main advantage of the level-set method is the computation of the curvature through the design variables. In this approach, curvature limitation is used as an objective function and can be minimized or maximized, which makes the final geometries smoother and more manufacturable. The presented method is applied in several industrial cases to show its efficacy.*

1 INTRODUCTION

One of the most fascinating fields in CFD research is adjoint optimization with its promise of automated design. The method has started to find its way into industry, where it is being used to optimize various products. Drag force minimization of a car, lift force maximization of aircraft wings and minimization of power losses in ducts are some typical optimization problems being so treated.

The adjoint method can be used either in shape or in topology optimization problems. The basic difference between the two is that, in shape optimization, the computational domain changes every optimization cycle using re-meshing or a mesh-deformation tool based on the surface sensitivities. In industrial cases re-meshing and mapping of results can significantly increase the computational cost. For this reason, starting with topology optimization and then switching to shape optimization is in many cases the most efficient way to solve an optimization problem.

Topology optimization was first introduced in structural mechanics by Bendsoe and Kikuchi [2]. They formulated and numerically solved equations in terms of material density to identify areas in which material should be added to increase structural stiffness. Several applications of topology optimization can be found in structural mechanics literature [3, 4]. In fluid mechanics, the porosity variable was first introduced in 2006 by C. Othmer [10]. In this approach, the porosity was added as an extra term in the RANS equations for laminar flows. Many examples of topology optimization including adjoint turbulence can be found in recent papers by PCOpt [7].

While topology optimization has been successfully applied in many cases, several issues remain in situations with high levels of complexity. These issues have been found to be primarily due to the steepest decent based porous blockage approach:

- The optimized shape is jagged and "noisy". After the optimization, the porosity field has to be filtered in order to extract the final shape. This technique produces inaccurate and potentially poorly manufacturable shapes.
- Porosity regions without any connection to the walls are created. The steepest descent method is unaware of the connectivity between the porosity regions and the walls. This can create "islands", which are impossible to manufacture.
- Blocking the flow with porosity is much easier than unblocking due to the small values of the volumetric sensitivities in the blocked areas. Unblocking thus takes longer than blocking operations, which can lead to poor intermediate results and long integration times.
- The optimization is inaccurate due to simplistic treatment of the surface. The absence of the surface location can cause accuracy problems, especially in turbulent cases, where the wall distance information is required from the turbulence models. Another problem derives from the categorisation of intermediate values of porosity as either solid or fluid and the effect this has on dependent functions.

Recently, differential equation-driven methods have emerged as an alternate approach for topology optimization. Implicit level-set methods are one such set of approaches in which the design domain is represented in terms of implicit functions and use the Hamilton-Jacobi equation as the evolution equation.

In this paper, a new method for topology optimization using the continuous adjoint method coupled with level-set method, is presented. The level-set method is inspired from a paper of J.A. Sethian [14] which details the use of the level-set method as a tracking interface and is made up of three parts: velocity extension, evolution, reinitialisation. More information about the level-set method can be found in literature ([8, 5, 9]). The interface can be tracked using the level-set variable φ , which is the signed distance from interface (see figure 1). In this paper, the velocity of the surface is the surface sensitivity calculated using the continuous adjoint method. The formulation of the continuous adjoint method can be found in literature [12, 15, 16, 13, 6, 11, 17, 7]. Statistical analysis algorithms for filtering the velocity of the interface, are implemented. The speed of the surface is controlled by specifying the mean and the maximum Courant number.

The proposed method is tested in two industrial applications. The first application is a 3D HVAC cooling duct and the second application consists of the inlet and the outlet ports of a gearpump, modelled as separate computational domains. The presented method is developed in an in-house optimization tool based on OPENFOAM®2.1.

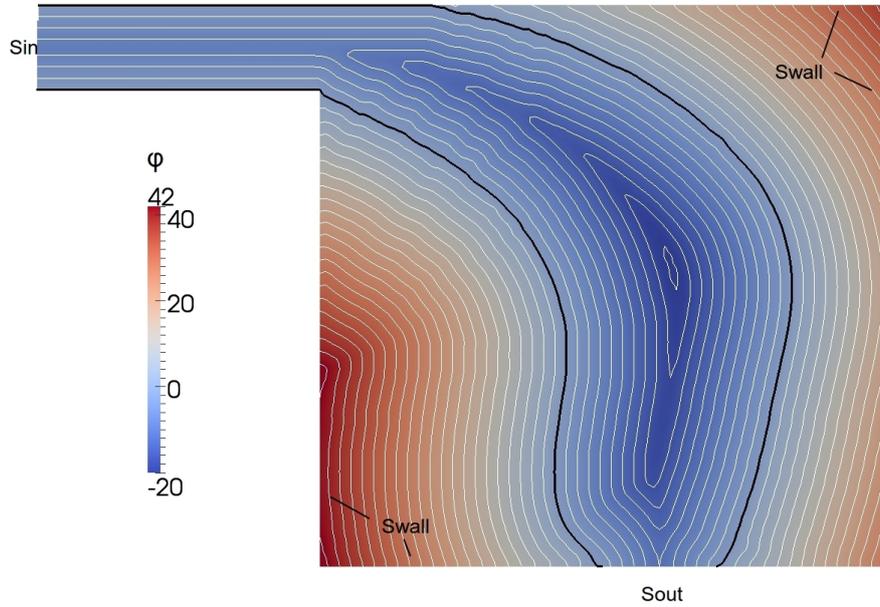


Figure 1: Level set field φ representation in a sample 2D case. S_{in} is the inlet, S_{out} is the outlet and S_{wall} are the walls of this case. The black iso-surface is the zero level set field and separates the fluid (negative values of φ) with solid part (positive values of φ) of the domain. The optimization algorithm is moving the solid-fluid interface in order to minimize or maximize the objective function.

2 TOPOLOGY OPTIMIZATION USING THE LEVEL-SET METHOD

2.1 THE LEVEL-SET METHOD

The level set method is a numerical technique for tracking interfaces and shapes. The level-set variable φ should satisfy the Eikonal equation:

$$|\nabla\varphi| = 1 \quad (1)$$

The difference from the distance functions is that they are monotonic and differentiable at the surface.

$$\varphi_i = \begin{cases} d_i & \text{if } i \in \text{solid part } \Omega_s \\ -d_i & \text{if } i \in \text{fluid part } \Omega_f \\ 0 & \text{if } i \in \text{interface } S \end{cases}$$

Any interface can be represented using the φ variable. The level-set method adds dynamics to the interface using the following transport equation or level-set equation.

$$\frac{\partial \varphi}{\partial t} + G_i \frac{\partial \varphi}{\partial x_i} = 0 \quad (2)$$

Where t refers to optimization cycles (pseudo-time) and G_i is the normal velocity of the interface. Using the divergence theorem, equation 2 can be alternatively written as:

$$\frac{\partial \varphi}{\partial t} + \frac{\partial(G_i \varphi)}{\partial x_i} - \varphi \frac{\partial G_i}{\partial x_i} = 0 \quad (3)$$

The normal velocity is extended towards the normal direction of the interface. This is achieved by solving the velocity extension equation 4.

$$W_i \frac{\partial G_n}{\partial x_i} = 0 \quad (4)$$

Where the W term is defined as follows:

$$W_i = \text{sign}(\varphi) \frac{\frac{\partial \varphi}{\partial x_i}}{|\frac{\partial \varphi}{\partial x_i}|} \quad (5)$$

Equation 4 has a very weak implicit part and is difficult to solve as a result. To facilitate the solution of the equation within the computational framework it is transformed using the divergence theorem and a small diffusion term is added to improve stability (equation 4):

$$\frac{\partial(W_i G_s)}{\partial x_i} - \frac{\partial W_i}{\partial x_i} G_s - k_{Lap} \frac{\partial^2 G_s}{\partial x_i^2} = 0 \quad (6)$$

where k_{Lap} controls the level of the diffusion term. The velocity of the interface is extended in a narrow band of cells by solving equation 6 for a few iterations.

After solving the level-set equation (eq. 3), the φ field is no longer a signed distance field. As a result the φ field has to be corrected by solving an additional equation which is called the reinitialisation equation.

$$\frac{\partial \varphi}{\partial t} + W_i \frac{\partial \varphi}{\partial x_i} = \text{sign}(\varphi) \quad (7)$$

Equation 7 is solved using narrow band approach by correcting the φ values of the cells near the surface. The φ values for the remaining cells are corrected using an advancing front

algorithm. In the first stage, the $\nabla\varphi$ term for the cells in the narrow band, is calculated iteratively (see eq. 9). In the second stage, equation 7 is solved explicitly.

$$\begin{aligned}\varphi_f^{CD} &= c_\alpha\varphi_\alpha + c_\beta\varphi_\beta \\ \varphi_f^{LUD} &= \varphi + \frac{\partial\varphi}{\partial x_i}\delta_i\end{aligned}\quad (8)$$

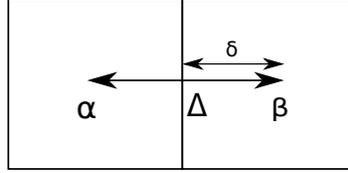


Figure 2: This figure represents the delta coefficients of the cells. The weights are calculated as $c_\beta = \frac{\delta}{\Delta}$, $c_\alpha = 1 - c_\beta$.

The interpolated φ values are used in eq. 9.

$$\begin{aligned}\nabla\varphi &= \frac{1}{\Omega} \left(\oint_S \varphi_f^{CD} n_i dS + \oint_S \varphi_f^{LUD} n_i dS \right) \\ \nabla\varphi &= f(\nabla\varphi)\end{aligned}\quad (9)$$

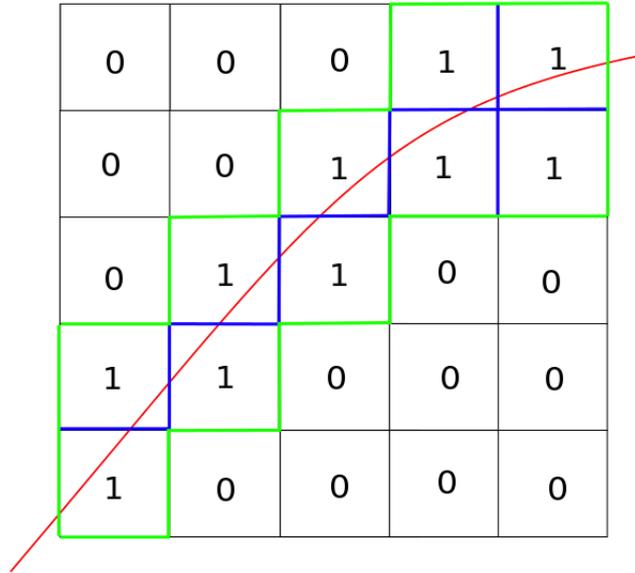


Figure 3: Schematic representation of the narrow band approach for a simple mesh. The cells marked with zero and one are the "dead" and "active" cells respectively. The φ values for the active cells are corrected using the reinitialisation equation. The interpolation scheme for the faces marked with green colour is **LUD** (linear upwind difference) whereas for the faces marked with blue colour is **CD** (central differencing). After the correction of the φ values in the active cells, advancing front algorithm is used for the calculation of the φ values of the dead cells.

2.2 THE CONTINUOUS ADJOINT METHOD

In this section, an alternative formulation of topology optimization, based on the level set method, is presented. The design variables of the level set topology optimization are variables φ , which express the signed distance of each internal node from the interface.

The primal equations used in this paper are the RANS (Reynolds-Averaged-Navier-Stokes). The design variables φ are defining a surface which separates the fluid and the solid part of the computational domain. In the solid part, the velocities are set to zero.

$$R^p = \frac{\partial v_j}{\partial x_j} = 0 \quad (10)$$

$$R_i^v = v_j \frac{\partial v_i}{\partial x_j} + \frac{\partial p}{\partial x_i} - \frac{\partial}{\partial x_j} \left[(\nu + \nu_t) \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) \right] \quad (11)$$

$$R_z = f(z) = \text{Convection} + \text{Diffussion} + \text{Production} + \text{Dissipation} = 0 \quad (12)$$

where v_i is the primal velocity, p is the primal pressure, ν and ν_t are the kinematic and the turbulent kinematic viscosity, respectively. R_z is considered to be an arbitrary turbulence model and z represents the turbulence variable(s). The distance from wall information, which is required in the turbulence models, is provided from the level-set variable φ . The user-defined objective function F can be defined on the surface and space integral (eq. 13).

$$F = \int_S F_S dS + \int_{\Omega} F_{\Omega} d\Omega \quad (13)$$

The objective function F is augmented by the state equations, R^p and R_{v_i} . In this paper, the turbulent kinematic viscosity ν_t is assumed frozen (frozen turbulence assumption).

$$F_{aug} = F + \int_{\Omega} q R^p d\Omega + \int_{\Omega} u_i R_i^v d\Omega \quad (14)$$

In general, the global variation (symbol δ) of any quantity Φ with respect to the design variable φ , is expressed as the sum of direct (symbol ∂) and grid-dependent variations.

$$\frac{\delta \Phi}{\delta \varphi} = \frac{\partial \Phi}{\partial \varphi} + \frac{\partial \Phi}{\partial x_i} \frac{\partial x_i}{\partial \varphi} \quad (15)$$

Using the Leibneiz theorem, the variation of the augmented objective function is:

$$\frac{\delta F_{aug}}{\delta \varphi} = \frac{\delta F}{\delta \varphi} + \int_{\Omega} q \frac{\partial R^p}{\partial \varphi} d\Omega + \int_{\Omega} u_i \frac{\partial R_i^v}{\partial \varphi} d\Omega + \int_S (u_i R_i^v + q R^p) \frac{\delta x_{\kappa}}{\delta \varphi} n_{\kappa} dS \quad (16)$$

After expanding the R_i^v and R^p terms in the equation 16, the $\frac{\partial v_i}{\partial \varphi}$ and $\frac{\partial p}{\partial \varphi}$ terms appear. The computational cost of these terms is extremely high because their calculation scales as the number of mesh elements squared. The adjoint method formulates the expressions of equation 16 using the Green-Gauss theorem and the multipliers of these terms are forced to become zero.

As a result, the calculation of the problematic terms is avoided. In equation 17 an example of the formulation using the Green-Gauss theorem is given.

$$\int_{\Omega} u_i v_j \frac{\partial}{\partial x_j} \left(\frac{\partial v_i}{\partial \varphi} \right) d\Omega = \int_S u_i v_i n_j \frac{\partial v_j}{\partial \varphi} dS - \int_{\Omega} \frac{\partial (u_i v_j)}{\partial x_j} \frac{\partial v_i}{\partial \varphi} d\Omega \quad (17)$$

The following equation (eq. 18) is the fully expanded $\frac{\partial F_{aug}}{\partial \varphi}$ (eq. 16).

$$\begin{aligned} \frac{\delta F_{aug}}{\delta \varphi} &= \int_S \left[u_j v_j n_i + u_i v_j n_j + (\nu + \nu_t) \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) n_j - q n_i + \frac{\partial F_S}{\partial v_i} \right] \frac{\partial v_i}{\partial \varphi} dS \\ &+ \int_{\Omega} \left\{ -v_j \left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \right) - (\nu + \nu_t) \frac{\partial}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) + \frac{\partial q}{\partial x_i} + \frac{\partial F_{\Omega}}{\partial v_i} \right\} \frac{\partial v_i}{\partial \varphi} d\Omega \\ &+ \int_S \left(u_j n_j + \frac{\partial F_S}{\partial p} \right) \frac{\partial p}{\partial \varphi} dS + \int_{\Omega} \left(-\frac{\partial u_j}{\partial x_j} + \frac{\partial F_{\Omega}}{\partial p} \right) \frac{\partial p}{\partial \varphi} d\Omega \\ &- \int_S (\nu + \nu_t) \frac{\partial}{\partial \varphi} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) n_j u_i dS + \int_S u_i R_i^v \frac{\delta x_k}{\delta \varphi} n_k dS + \int_S q R^p \frac{\delta x_k}{\delta \varphi} n_k dS \\ &+ \int_S \left(\frac{\partial F_S}{\partial x_k} + F_{\Omega} n_k \right) \frac{\delta x_k}{\delta \varphi} dS + \int_S F_S \frac{\delta (dS)}{\delta \varphi} \end{aligned} \quad (18)$$

The terms multiplied with $\frac{\partial v_i}{\partial \varphi}$ and $\frac{\partial p}{\partial \varphi}$ inside the space and surface integrals, are the adjoint equations and their adjoint boundaries, respectively. In this paper, the minimization of power loss defined over the surface integral is used as an objective function.

$$F = - \int_{S_{i,o}} \left(p + \frac{1}{2} v_i^2 \right) v_i n_i dS \quad (19)$$

For the power loss objective, the adjoint continuity and momentum equations are the following:

$$R_q = \frac{\partial u_i}{\partial x_i} = 0 \quad (20)$$

$$R_i^v = -v_j \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) + \frac{\partial q}{\partial x_i} - \frac{\partial}{\partial x_j} \left[(\nu + \nu_t) \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] = 0 \quad (21)$$

The boundary conditions are more complicated.

- At the boundaries where the primal velocity v_i is fixed (Dirichlet type), the variation of v_i is zero ($\frac{\partial v_i}{\partial \varphi} = 0$). As a result, the following boundary conditions are chosen:

$$\begin{aligned} u_{(n)} &= -\frac{\partial F_S}{\partial p} = v_i n_i \\ u_{(t)} &= u_i - u_j n_j n_i = 0 \\ \frac{\partial q}{\partial n} &= 0 \end{aligned} \quad (22)$$

- At the boundaries where the primal pressure p is fixed (Dirichlet type), the variation of p is zero ($\frac{\partial p}{\partial \varphi} = 0$). The boundary conditions are the following:

$$\begin{aligned}
 q &= u_j v_j + u_{\langle n \rangle} v_{\langle n \rangle} + (\nu + \nu_t) \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) n_j n_i - \frac{1}{2} v_i^2 - v_n^2 - p \\
 0 &= u_{\langle t \rangle} v_{\langle n \rangle} + (\nu + \nu_t) \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) n_j t_i - v_{\langle t \rangle} v_{\langle n \rangle}
 \end{aligned} \tag{23}$$

The remaining terms of the equation 18 are the surface sensitivities.

$$\frac{\delta F}{\delta \varphi} = - \int_{S_W} (\nu + \nu_t) \left(\frac{\partial u_i}{\partial n} + \frac{\partial u_{\langle n \rangle}}{\partial x_i} \right) \frac{\partial v_i}{\partial n} dS \tag{24}$$

The velocity in the level-set equation (see section 2.1) is calculated from the surface sensitivities.

$$G_j = (\nu + \nu_t) \left(\frac{\partial u_i}{\partial n} + \frac{\partial u_{\langle n \rangle}}{\partial x_i} \right) \frac{\partial v_i}{\partial n} S_j \tag{25}$$

where S_j represents the surface area vector. Curvature limitation is treated as an additional objective. The curvature objective sensitivities are added to the equation 25.

$$\begin{aligned}
 G_j^{lim} &= w_1 G_j + w_2 F_j^{lim} \\
 F_j^{lim} &= k n_j = \frac{\partial^2 \varphi}{\partial x_i^2} \frac{\partial \varphi}{\partial x_j}
 \end{aligned} \tag{26}$$

where k is the curvature, n_j is the normal vector of the surface, w_1 and w_2 are the weights of the velocities. The curvature objective is very important: it prevents the formation of small holes (inclusions) in the interface and strongly discourages the detachment of solid volumes that could form 'islands', leading to a much smoother and more manufacturable final geometry.

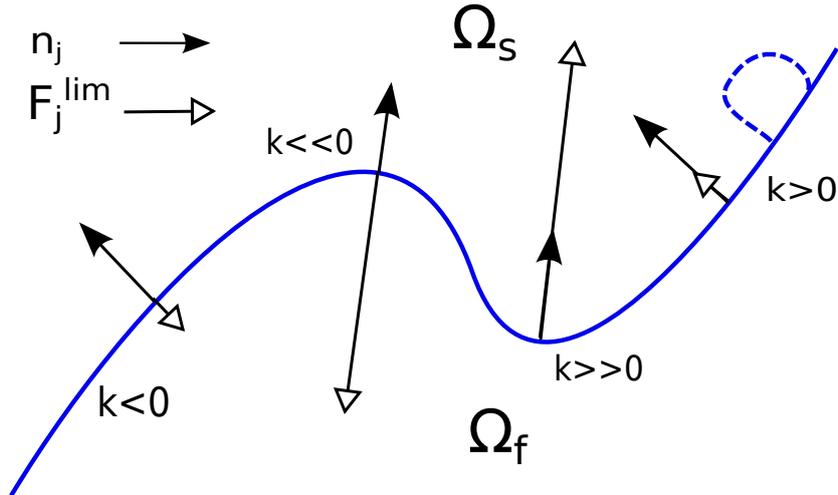


Figure 4: This figure represents the curvature objective, which makes the interface as smooth as possible. k is the curvature, n_j is the normal vector of the surface and the F_j^{lim} are the curvature sensitivities. For example, if the surface has a hole like the dashed line, the curvature will have a large value, whereas the other objectives small. After some optimization cycles, the hole will be filled and the optimized shape will be more manufacturable.

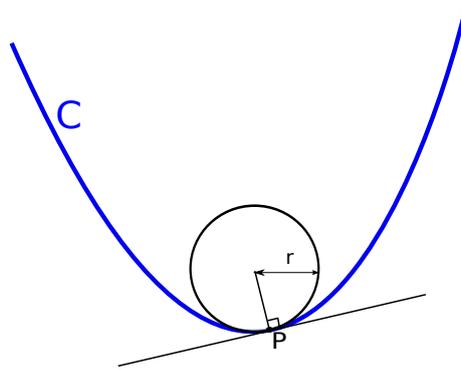


Figure 5: Physical meaning of the curvature k . Given any curve C and a point P on it, there is a unique circle which most closely approximates the curve near P , the osculating circle at P . The curvature of C at P is then defined to be the curvature of that circle. The value of the curvature of the C curve at the P point is $k = \frac{1}{r}$.

2.3 OPTIMIZATION ALGORITHM

The optimization algorithm is represented in figure 6 and includes the following steps:

1. Every optimization algorithm starts from an initial geometry. In most of the cases, the wall boundaries of the computational domain constitutes the starting interface. The optimization can also start from an interface provided by the user.
2. The primal and the adjoint equations are solved for a number of iterations.
3. The surface sensitivities for the selected objective functions, are calculated. The curvature is treated as an additional objective and is added to the surface sensitivities.
4. The calculated sensitivities are extended towards the normal direction of the interface by solving the velocity extension equation (eq. 6).

5. The velocities are filtered through the statistical analysis mechanism. The user provides only the max and mean desired Courant number for the transport equation, which is used in the next step.
6. Having the filtered extended velocities, the interface is evolving by solving the transport equation (eq. 3).
7. The level-set variable φ is no longer the signed distance. As a result, the solution of the reinitialisation equation (eq. 7) is required.
8. If the optimization algorithm is not converged the new interface is provided to step 2. If the optimization algorithm is converged, the interface is exported to an STL file, based on the zero iso-surface of the φ field.

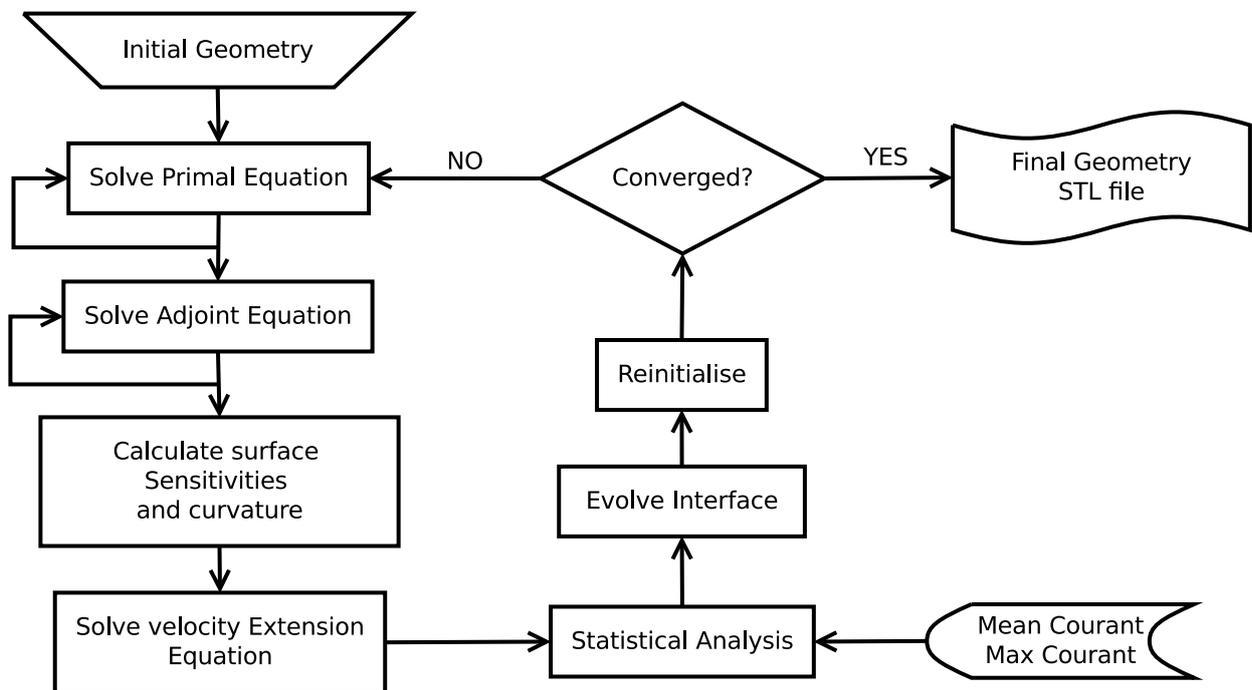


Figure 6: This figure represents the optimization algorithm.

3 APPLICATIONS

The first of the two industrial test cases is the optimization of a HVAC cooling duct. The second, pressure loss minimisation for the inlet and outlet ports of a high pressure gearpump.

3.1 HVAC COOLING DUCT

The presented method is used to optimize a 3D HVAC cooling duct (fig. 7).

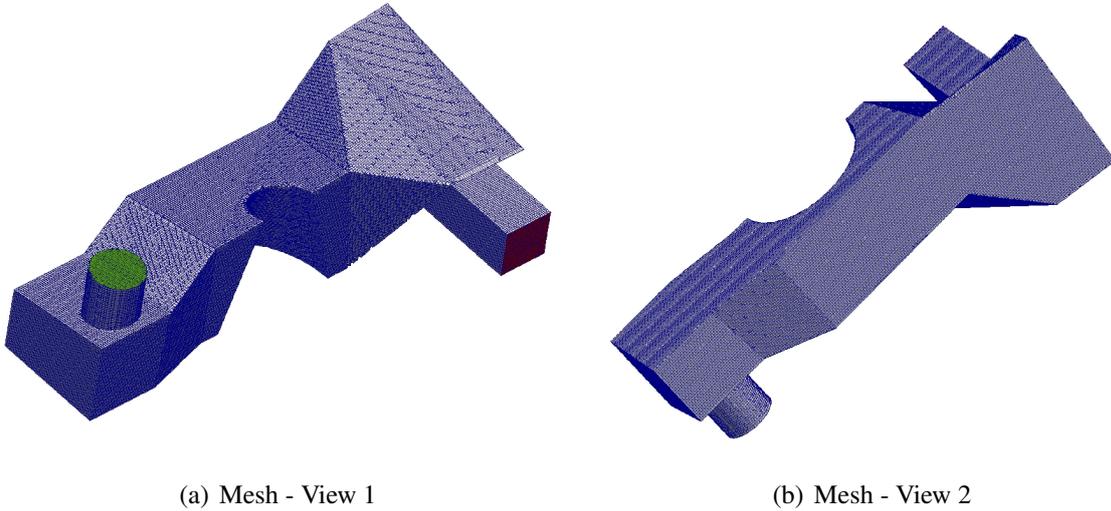


Figure 7: These figures represent the baseline mesh of the duct. The inlet and the outlet boundaries are the red and green patches, respectively. The mesh is generated using an in-house version of *snappyHexMesh* and has 600.694 cells.

The fluid is air with kinematic viscosity $\nu = 1.5881 \cdot 10^{-5} \text{ m}^2/\text{s}$ and density $\rho = 1.205 \text{ kg}/\text{m}^3$. The RANS flow equations is used for the primal problem. For the boundary conditions:

- At the inlet, the flow rate is $\dot{m} = 0.0133 \text{ kg}/\text{s}$.
- At the outlet, the pressure is $p = 0 \text{ Pa}$.

The *kOmega - SST* turbulence model is chosen, as the flow is turbulent.

The minimization of power losses (eq. 19) between the inlet and outlet is the chosen objective function. The inlet adjoint boundary condition is in accordance with a fixed (Dirichlet) primal velocity (eq. 22). The result of the optimization is presented in table 1.

	Duct
Baseline (W)	0.077
Optimized (W)	0.038
Percentage (%)	50.65

Table 1: Objective function values before and after optimization. The power losses were decreased about 50%.

The zero level-set isosurface (optimized shape), which is represented in figure 8 is smooth and manufacturable. The flow of the optimized shape (fig. 9) is much cleaner and without recirculation zones. The algorithm converges within fewer than 2000 optimization cycles (figure 10).

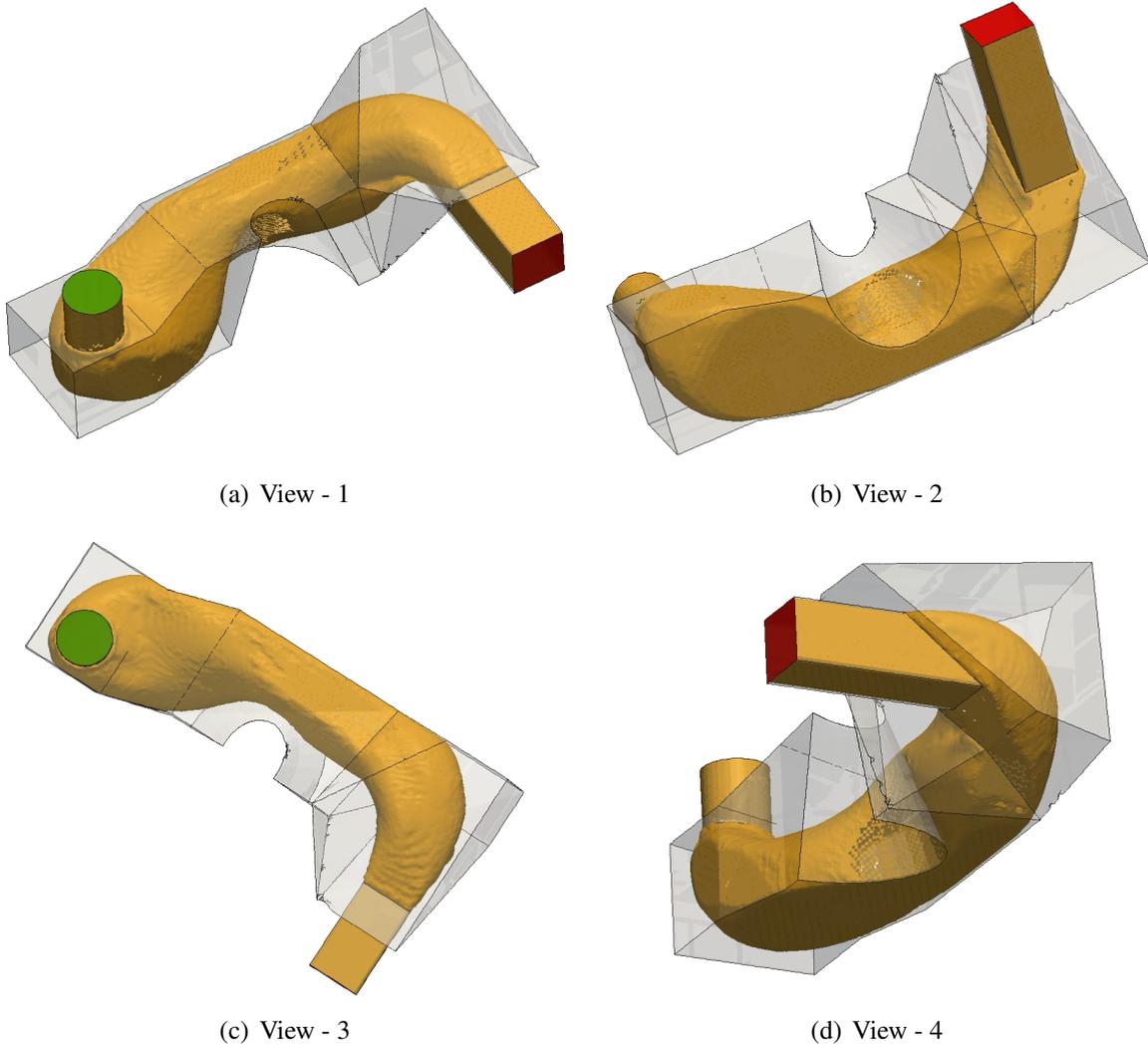


Figure 8: These figures represent the optimized shape from 4 different sides. The yellow and the transparent grey are the zero-level field and the baseline shape, respectively. The optimized shape is smooth and manufacturable.

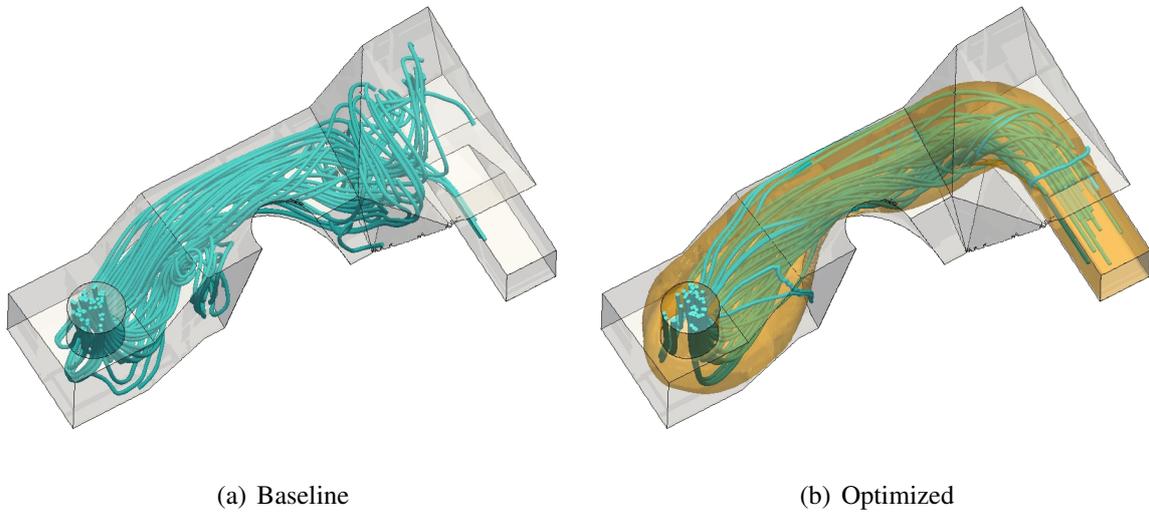


Figure 9: Streamlines before and after optimization. It is clear that the recirculation zones which appeared in the flow of the baseline mesh, are not present after the optimization.

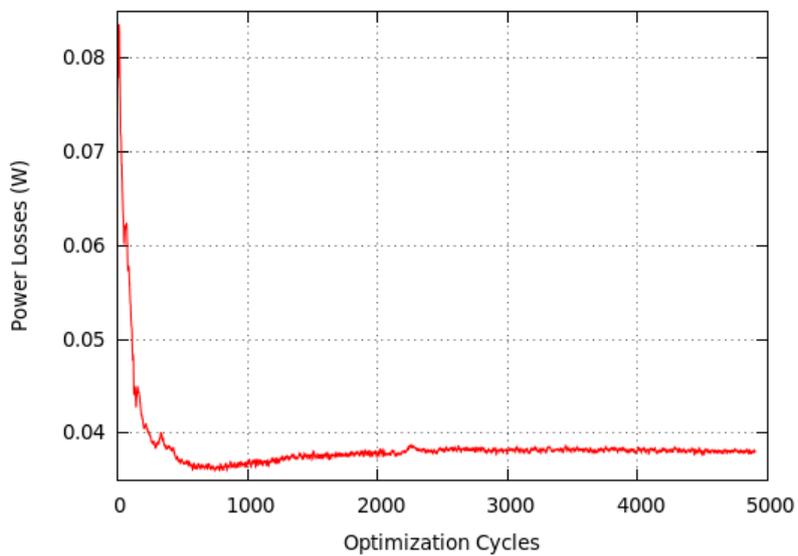


Figure 10: This figure represents the optimization convergence. The optimization algorithm converged after fewer than 2000 optimization cycles. The noise in the graph can be explained by: a) The user-defined mean and max Courant number. b) The lack of fully immersed boundaries.

3.2 GEARPUMP

In this application, the presented method is used to optimize the inlet and outlet port of a gearpump (fig. 11). The recirculation areas in figure 11(b) indicate that the two ducts (inlet and outlet port) are optimizable. The minimization of the power losses (eq. 19) is used as an objective function. The geometry is split into two different cases, which are presented in figure 12. The fluid is a lubricant oil with kinematic viscosity $\nu = 1.09047 \cdot 10^{-5} m^2/s$ and density $\rho = 829 kg/m^3$.

The primal problem has the following boundary conditions:

- Inlet port boundaries:
 - The mass flow at each outlet is $\dot{m} = 0.375 kg/s$.
 - The pressure is fixed to zero at the inlet $p = 0 Pa$.
- Outlet port boundaries:
 - The mass flow at the inlet is $\dot{m} = 0.75 kg/s$.
 - The pressure is fixed $p = 500 kPa$ at the outlet.

For the adjoint boundary conditions:

- The boundaries with fixed flowrate are treated as they were specified with fixed velocity (eq. 22).
- The boundaries with fixed pressure, equation 23 is used for adjoint boundary conditions.

When equation 23 is used for a primal inlet boundary condition, special measures have to be taken to prevent the occurrence of a singularity when the normal velocity to the boundary is negative. The same singularity generating mechanism can also cause instabilities when there is reversed flow at outlets.

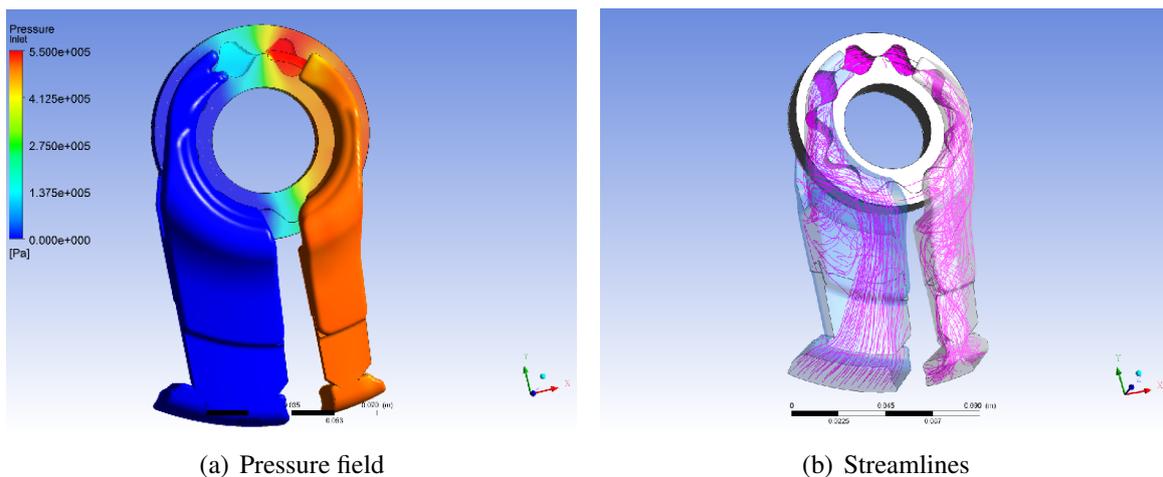


Figure 11: This figure represents the whole gearpump(inport, gear and output). The recirculation areas indicate that the power losses can be optimized.

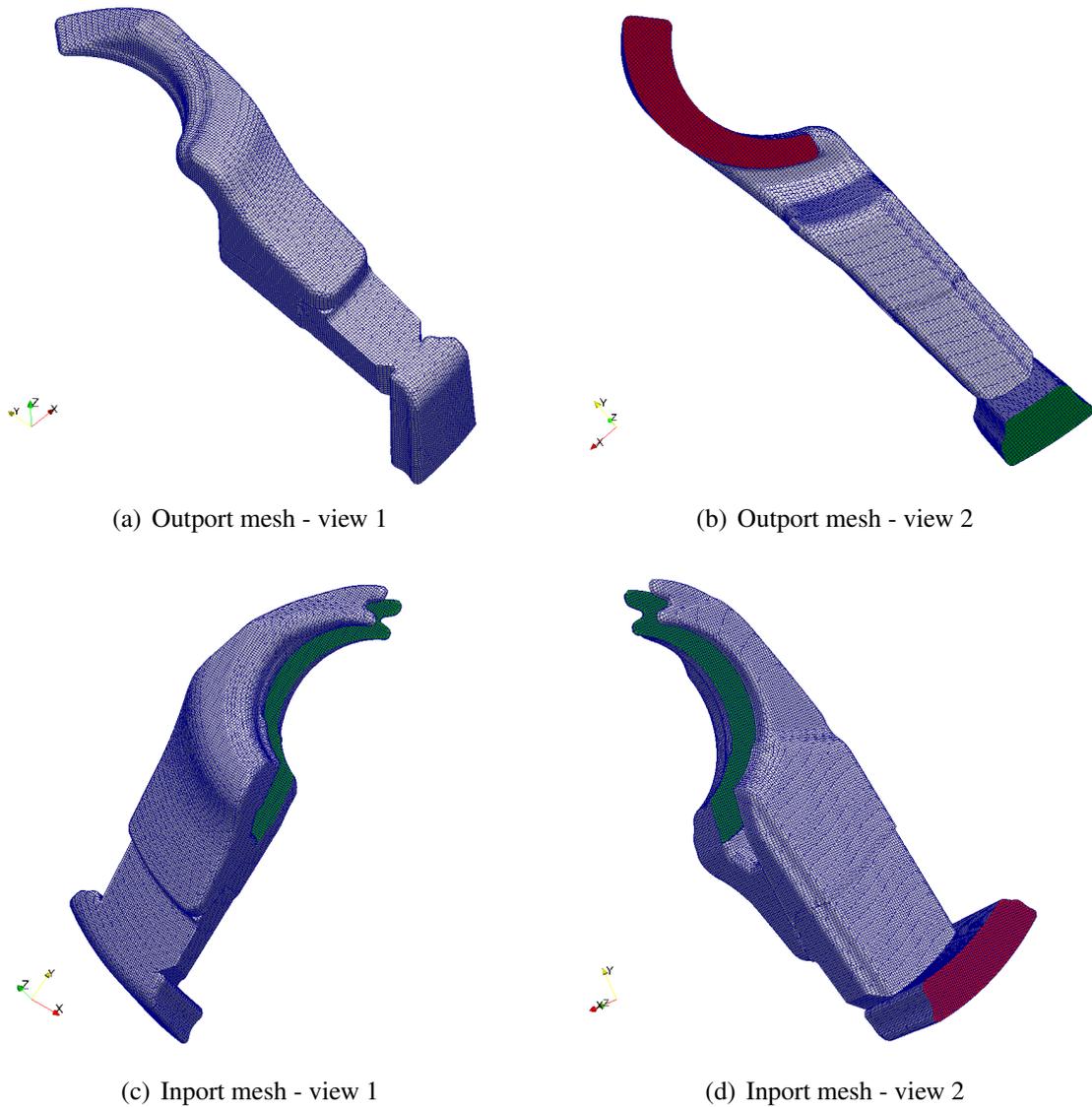
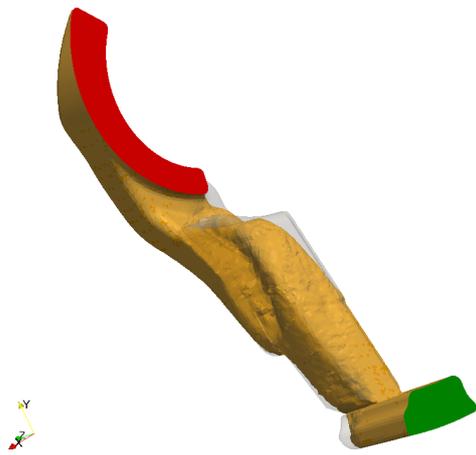


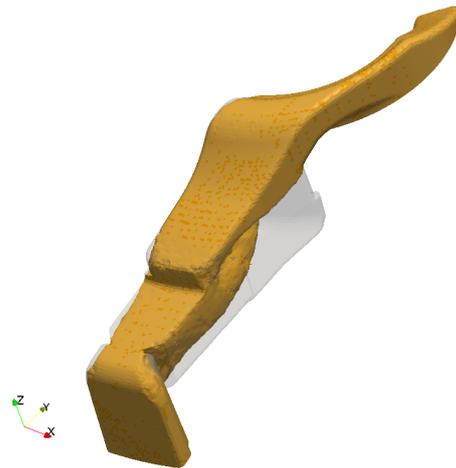
Figure 12: In this figure, the mesh for the inport and outport ducts, generated with an in-house version of *snappyHexMesh* mesh generator, is represented. The mesh size for the inport and outport is 330000 and 175000 cells respectively. The green patches are the outlets of each case, whereas the red patches are the inlets.

	Inport	Outport	GearPump
Baseline (W)	2.308	31.017	33.325
Optimized (W)	1.635	25.379	27.013
Percentage (%)	29.17	18.18	18.94
Re-meshed STL (W)	1.572	24.25	25.82
Percentage (%)	31.9	21.8	22.5

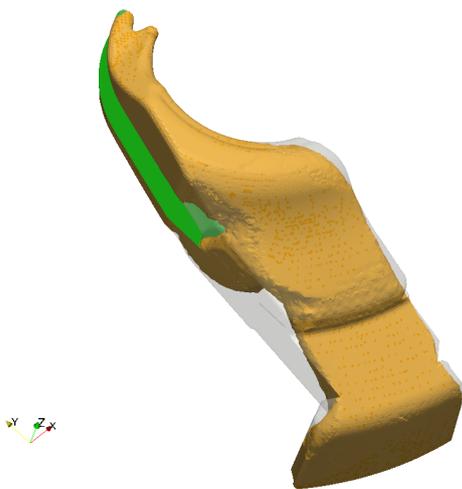
Table 2: Objective function values before and after optimization. The power losses of the gearpump decreased by about 18.94%. The optimized shape was re-meshed and the result was an improvement of 22.5% compared with the baseline shape.



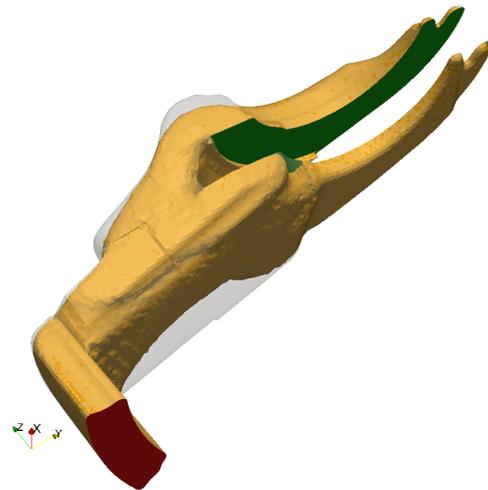
(a) Optimized output - view 1



(b) Optimized output - view 2



(c) Optimized input - view 1



(d) Optimized input - view 2

Figure 13: This figure represents the optimized shapes. The orange surface is the zero level set field, whereas the transparent grey surface consists the baseline mesh.

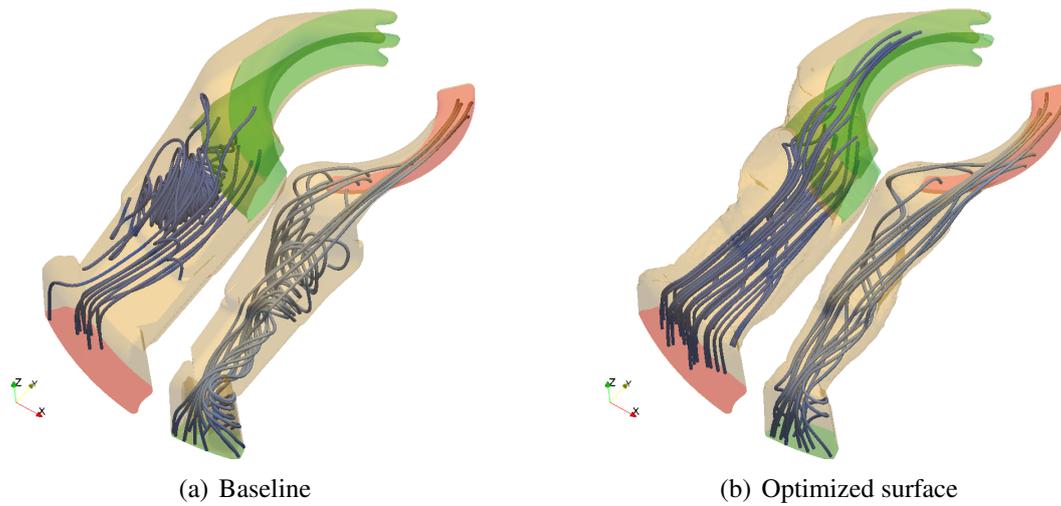


Figure 14: The left and right figure represents the streamlines for the baseline and optimized ducts. The optimization was clearly successful as the recirculation zones have all but disappeared.

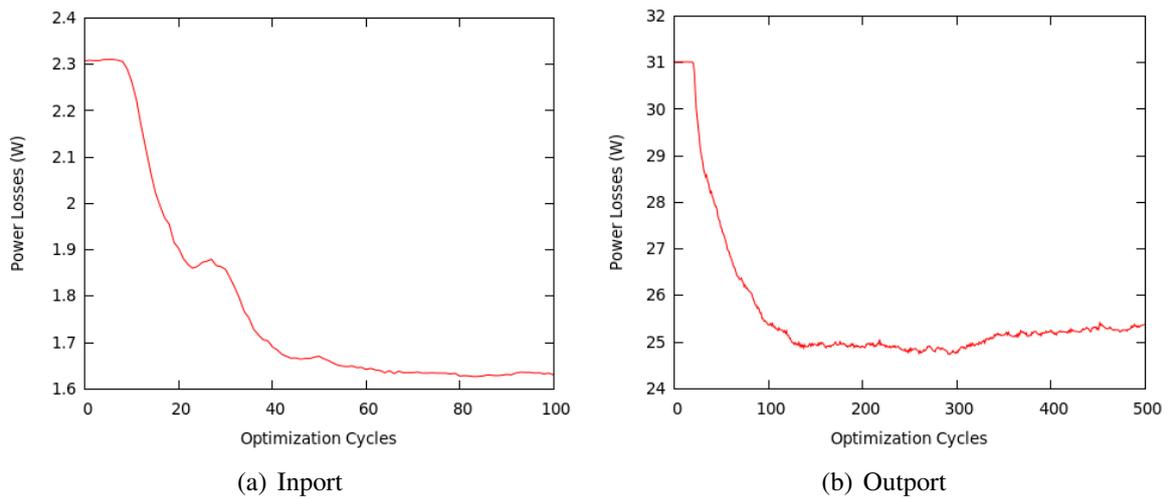


Figure 15: Optimization convergence of the objective function for the inport and output ducts. For the inport and the output the power losses decreased by 29.17% and 18.18% respectively. The power losses for the whole gearpump, were improved by 0.1894%. The optimization for the output is more important than the inport.

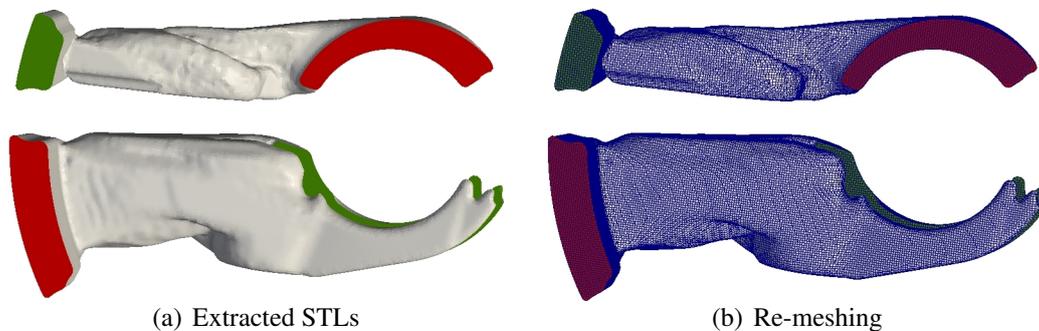


Figure 16: The left figure represents the STL file extracted from the zero level-set field in the last optimization cycle and can be used immediately for re-meshing (right figure).

4 CONCLUSIONS

The continuous adjoint method coupled with level-set method was presented and tested in industrial cases. The equations for the three parts (extension, evolution and reinitialization) of the level-set method are explained in detail. The applications show that the method is efficient and capable of use by industrial operators. For the HVAC cooling duct a 50% improvement of the objective function was obtained. In the second application, the power losses of the gearpump were decreased about 19%.

The main advantages of the level-set method in conjunction with the adjoint optimisation algorithm can be summarised as:

- Better control of the optimization. The presence of an interface makes topology optimization method more intelligible to the user. Moreover, the speed of the interface can be controlled by understandable variables such as mean and max Courant number.
- The optimized shape is smooth and it can be provided immediately as an STL file to the manufacturers. The smoothness is controlled by the curvature objective which is calculated implicitly (from the design variables).
- The usage of surface sensitivities and the existence of a curvature objective makes free standing "island" generation almost impossible.
- The signed distance field provides a solid basis for further improvements in accuracy. The level-set method can be coupled with immersed boundaries, which can significantly increase the accuracy of the interface representation. In theory, a sufficiently accurate immersed boundary would be indistinguishable from a real surface, blurring the lines between shape and topology optimisation. Furthermore, the wall-distance information, which is required by the turbulence models, can be used directly from the design variables. Finally, the design variables can be provided as an input to mesh adaptation algorithms. As a result, the user can control the y^+ by changing the refinement level of the interface.

5 ACKNOWLEDGEMENTS

Special thanks to **Aisin AW** company for providing us the gearpump geometry. The present work has been conducted under the auspices of the ITN Aboutflow FP7 EU project [1].

REFERENCES

- [1] Adjoint based optimization of industrial unsteady flows. <http://aboutflow.sems.qmul.ac.uk>.
- [2] M. Bendsoe and N. Kikuchi. Generating optimal topologies in structural design using a homogenization method. *Journal of Computer Methods in Applied Mechanics and Engineering*, 71:197–224, 1988.
- [3] M. Bendsoe and O. Sigmund. *Topology Optimization - Theory, Methods and Applications*. Springer Verlag, 2004.
- [4] T. Buhl, C. Pedersen, and O. Sigmund. Stiffness design of geometrically non-linear structures using topology optimization. *Structural and Multidisciplinary Optimization*, 19(2):93–104, 2000.

- [5] Y.C. Chang, T.Y. Hou, B. Merriman, and S. Osher. A level set formulation of eulerian interface capturing methods for incompressible fluid flows. *Journal of Computational Physics*, 124:449–494, 1996.
- [6] I.S. Kavvadias, G.K Karpouzas, E.M Papoutsis-Kiachagias, D.I Papadimitriou, and K.C Giannakoglou. Optimal flow control and topology optimization using the continuous adjoint method in unsteady flows. In *Evolutionary and Deterministic Methods for Design, Optimization and Control with Applications to Industrial and Societal Problems, Eurogen 2013*, Las Palmas de Gran Canaria Spain, October 7-9,2013.
- [7] E.A Kontoleontos, E.M. Papoutsis-Kiachagias, A.S. Zymaris, D.I. Papadimitriou, and K.C. Giannakoglou. Adjoint-based constrained topology optimization for viscous flows, including heat transfer. *Engineering Optimization*, 45(8):941–961, 2013.
- [8] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer, 2002.
- [9] S. Osher and J.A. Sethian. Fronts propagating with curvature–dependent speed: Algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics*, 79:12–49, 1988.
- [10] C. Othmer. A continuous adjoint formulation for the computation of topological and surface sensitivities of ducted flows. *International Journal for Numerical Methods in Fluids*, 58(8):861–877, 2008.
- [11] C. Othmer, E.D. Villiers, and H.G. Weller. Implementation of a continuous adjoint for topology optimization of ducted flows. In *18th AIAA Computational Fluid Dynamics Conference, Miami, FL*. AIAA 2007-3947,2007.
- [12] D.I. Papadimitriou and K.C. Giannakoglou. Aerodynamic shape optimization using first and second order adjoint and direct approaches. *Archives of Computational Methods in Engineering (State of the Art Reviews)*, 15(4):447–488, 2008.
- [13] D.I Papadimitriou and K.C. Giannakoglou. Aerodynamic design using the truncated newton algorithm and the continuous adjoint approach. *Numerical Methods in Fluids*, 68(6):724–739, 2012.
- [14] J.A. Sethian. Evolution, implementation, and application of level set and fast marching methods for advancing fronts. *Journal of Computational Physics*, 169:503–555, 2001.
- [15] A.S. Zymaris, D.I. Papadimitriou, K.C. Giannakoglou, and C. Othmer. Continuous adjoint approach to the Spalart-Allmaras turbulence model for incompressible flows. *Computers & Fluids*, 38(8):1528–1538, 2009.
- [16] A.S. Zymaris, D.I. Papadimitriou, K.C. Giannakoglou, and C. Othmer. Adjoint wall functions: A new concept for use aerodynamic shape optimization. *Journal of Computational Physics*, 229:5528–5245, 2010.
- [17] A.S. Zymaris, D.I. Papadimitriou, E. Papoutsis-Kiachagias, K.C. Giannakoglou, and C. Othmer. The continuous adjoint method as a guide for the design of flow control systems based on jets. *Engineering Computations*, 30(4):494–520, 2013.