Shape Optimization of 3D Diaphragm Pumps using the Continuous Adjoint Approach to the Cut-Cell Method

***Konstantinos Samouchos[1], Dimitrios Kapsoulis[1], Xenofon Trompoukis[1], and Kyriakos Giannakoglou[1]**

[1]Parallel CFD & Optimization Unit, National Technical University of Athens, Greece.
[*]Presenting and Corresponding author: ksamouchos@gmail.com

Abstract

This paper is concerned with tools that can be used for the design/optimization of 3D valveless and valved diaphragm pumps. These pumps comprise a main chamber, an inlet and outlet duct and a periodically moving diaphragm that causes the unsteady/periodic fluid flow within the device; the inlet and outlet ducts might either be valveless diffusers or tubes of constant cross-section with valves (valved pumps). Obtaining the desired net mass-flow rate per period with the minimum (hopefully zero) back-flow at the exit (especially, in the absence of valves) are the usual objectives. The minimization of viscous losses within some pump's components such as inlet or outlet valved ducts could be an extra target. Regarding CFD analysis, apart from the cut-cell method mentioned in the title, an in-house flow solver using body-fitted unstructured grids is also used. Both codes are based on the finite volume approach by sharing the same numerical features and run on GPU-clusters (the former) and on many-processor platforms (the latter). To support gradient-based optimization, with either CFD tool, the continuous adjoint method computing the gradient of the objectives with respect to the design variables has been developed and programmed. According to the proposed parameterization scheme, all design variables are related to the diaphragm geometry and its periodic motion. An alternative optimization method that was also made available for the needs of this paper is a (gradient-free) evolutionary algorithm, assisted by metamodels (on-line trained artificial neural networks) and the principal component analysis for efficiently searching the design space. Using this tool, many-objective optimizations can be performed to compute the front of non-dominated solutions. This paper focuses on particular features of the analysis and optimization tools for this type of diaphragm pumps, including micropumps for medical applications.

Keywords: Cut-cell method; Adjoint method; Diaphragm pump; Valved pump; Evolutionary Algorithm.

Introduction

Valveless or valved diaphragm pumps are formed by an inlet and outlet duct as well as a chamber with a periodically moving diaphragm controlling the fluid's flow. Depending on the application, they are often preferred over bladed pumps since they can pump various types of fluids in a noiseless manner. They are manufactured in large or small scales, with the large (usually valved) ones used for cleaning tank bottoms or pumping sewage, while the small (usually valveless) ones (micropumps) used as medical analysis devices, in biochemical-processing applications or to deliver drug to patients [1]. The inlet and outlet ducts might be tubes of constant cross-section equipped with valves of different types (valved pumps) preventing back-flow. In medical applications and drug injections using valveless (micro)pumps, the exit flow must be uninterruptible and, in some cases, the valves are replaced by diffusers [2, 3]. Unfortunately, during their operation, back-flow might instantaneously occur at the outlet. Therefore, obtaining the desired net volume flux per period with minimum (hopefully zero) back-flow at the exit are standard objectives/constraints during design-optimization. In some cases, keeping viscous losses low in pump's components is an additional objective.

Regarding the CFD analysis of the diaphragm pumps this paper is dealing with, two in-house codes are used to model the periodic viscous flow within the pump. The first one, referred to as PUMA [4], uses body-fitted unstructured grids, continuously adapted to the moving diaphragm, using the spring analogy method. The software is GPU-enabled and uses the MPI protocol to perform on a many-GPU platform. However, in cases where large boundary movements occur or two bodies approach and finally touch each other, such as in valved pumps, grid deformation/adaptation to the changing boundaries becomes costly, delicate or even impossible. For these reasons, a flow solver based on the cut-cell variant [5] of the general class of Immersed Boundary Methods (IMB) [6] has been developed, too. Starting point is a Cartesian grid enclosing the flow domain. The background (coarse) grid remains stationary while the immersed solid bodies are allowed to move covering and uncovering grid cells. Grid is continuously refined close to the moving geometry, to increase the accuracy of the flow simulation. The two CFD codes are based on the finite volume approach and share the same numerical features.

For the optimization, the pump's diaphragm motion must be parameterized. The values of the design variables minimizing the selected objective functions are the unknowns in the optimization problem. The manufacturability of the diaphragm and the mechanism controlling its motion is beyond the scope of this paper.

The GB method is supported by the continuous adjoint method that computes the gradient of the objective function w.r.t. the design variables. The main advantage of the adjoint method is that its cost is independent of the number of the design variables. In case of more than one objectives, these are concatenated to a single function to be minimized. However, since the computation of the Pareto front of non-dominated solutions is of great value for the designer, an Evolutionary Algorithm (EA) is additionaly used. The EA is assisted by locally trained metamodels and the Kernel Principal Component method reducing the optimization's computational cost, which is the main drawback of EAs [7].

In this paper, CFD analyses and optimizations with either code and either optimization method, for diaphragm pumps or their components (such as inlet/outlet tubes equipped with a rotating disc-shaped shutter) are demonstrated.

Description/Parametrization of the Diaphragm Pumps

Fig. 1a shows the valveless micropump studied in this paper. Its geometry is based on an existing micropump found in the literature [2]. Regarding the valved pump it is assumed that they use cylindrical inlet and outlet ducts with a constant cross-section equipped with valves, fig. 1b, in place of the diffusers used in the valveless devices. The micropump's scale is of millimeters, with $1cm$ length and chamber's volume $\sim 40mm^3$. Valved pump's dimensions can be much larger, since this kind of devices are used for different purposes. Due to manufacturing reasons, the inlet and outlet diffusers or ducts and valves are identical. The pressure distribution inside the chamber and its volume is periodically changing by moving the diaphragm, this motion being essential for the pump's functionality. Technically, a piezoelectric device moves the diaphragm with a predetermined frequency, by pushing it from the inlet to the outlet causing the fluid to flow.

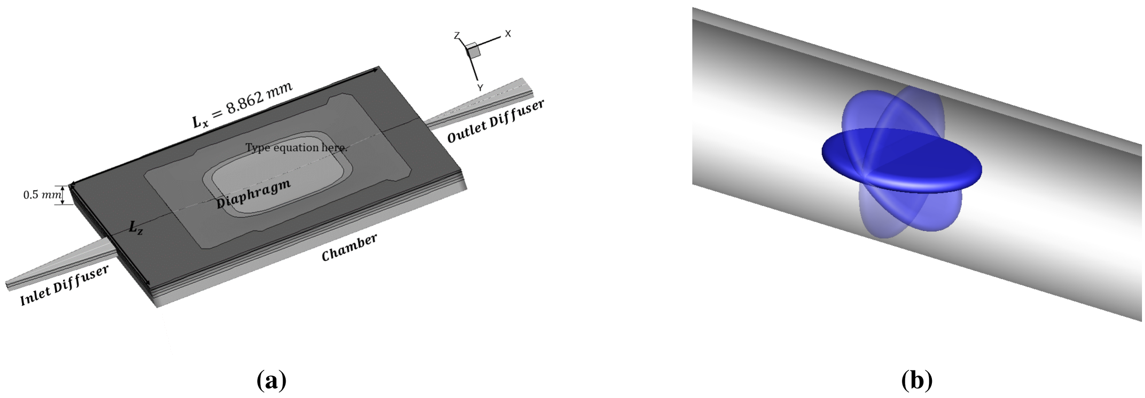


Figure 1: Left: Valveless diaphragm micropump with diffusers at the inlet and outlet. Right: Valved (butterfly valve) duct replacing the diffusers. The rotating disc, at three different positions, is shown in blue. The disc is always present in the flow, so it induces a total pressure drop, even when open.

The diaphragm's motion is parameterized, defining thus the design variables for the ensuing optimizations. The resting position of the diaphragm lays on the $y = 0$ plane, where the x and z axes point in the longitudinal and span-wise directions, respectively. The first two design variables (b_1, b_2) define a rectangular area limiting the moving part of the diaphragm, fig. 1. The maximum displacement $y_{max}(t)$ over all time instants is given by $y_{max}(t) = b_3 exp\big(-b_4(t-T)^2\big)\big(1-|1-\frac{2t}{T}|\big)$ where $T = 0.02\ sec$ is the period, b_3 controls the maximum overall displacement, which is much smaller than the chamber's height, achieved at the half period and b_4 controls the abruptness of the exponential function. The longitudinal deformation over time is defined by $y(x,t) = y^{max}(t)(6\tau_x^2 - 8\tau_x^3 + 3\tau_x^4),\ \ \tau_x = \frac{x+\delta x}{Dx}$, where $\delta x = b_5 min(x + L_x/2, L_x/2 - x)$ and $Dx = (1-b_6)\delta x$, with L_x and L_z being the total length and width of the chamber. Similarly, the span-wise diaphragm's deformation follows a similar parameterization. The design variables b_7 and b_8 define $\delta z = b_7 min(z + L_z/2, L_z/2 - z)$ and $Dz = (1-b_8)\delta z$, which are required to compute $y(z,t) = y_{max}(t)(6\tau_z^2 - 8\tau_z^3 + 3\tau_z^4),\ \ \tau_z = \frac{z+\delta z}{Dz}$. Different time instants of the deformation of the arbitrarily selected motion along the symmetry plane can be seen, later on, in fig. 11.

Flow Equations and Discretization

The 3D grid generator developed and used by the cut-cell method is likely the most challenging and important part of any cut-cell software. The implementation of an octree data structure makes grid generation fast with low memory requirements. Starting point is a uniform Cartesian grid covering the whole computational space including also the solid bodies (i.e. the pump's boundaries). Then, each cell intersected by the pump's boundaries is subdivided into eight sub-cells. The process is repeated until a minimum cell volume is reached. During the grid refinement process, each cell should have at most four neighbouring cells per face, preventing this way the creation of big cells next to much smaller ones. The computation of the exact intersection of cells with the pump's boundaries is based on the Sutherland-Hodgman algorithm [8]. The part of the cell inside the solid body is discarded and the remaining part creates a polyhedral finite volume (fig. 2a). Special treatment is needed whenever a cell splits into two or more polyhedra, which are treated as different finite volumes (fig. 2b).

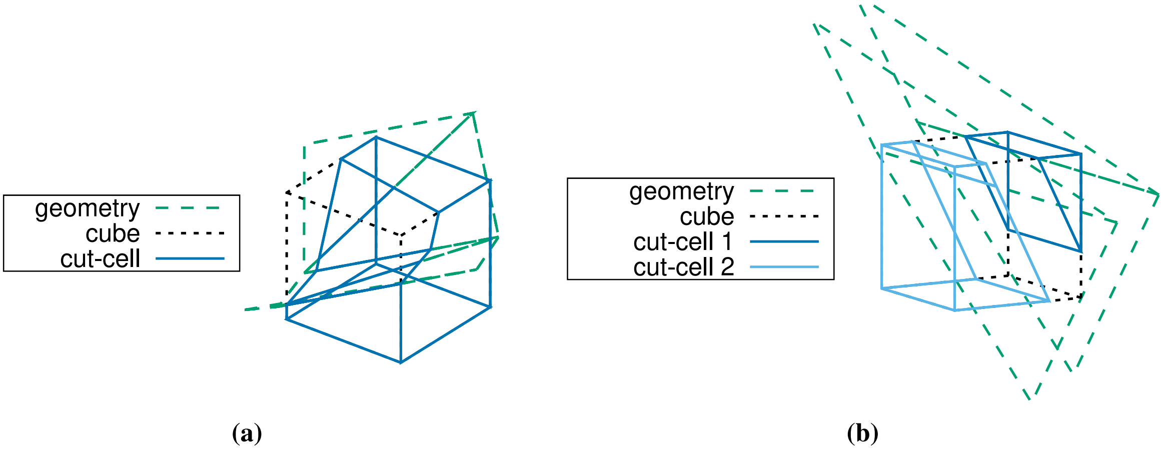


Figure 2: (a) A cube (black dashed line) is cut by the triangulated solid surface (green dashed line) creating a cut-cell (blue line). (b) A cube is separated into two different cut-cells. The flow and adjoint equations are solved only in the part of the cube lying inside the fluid domain.

A known downside of the cut-cell method is the formation of very small cut-cells leading to instabilities during the flow and adjoint numerical solution. A cell-merging technique is applied to surpass this difficulty. According to that, small cells are merged with bigger neighbours, named master-cells, creating hyper-cells. Geometrical criteria are used for the selection of the master-cell among the neighbours of a small-cell. In specific, for all neighbouring cells with volume greater than a threshold value, the one with the largest common face is chosen. If the first criterion fails, the neighbour with the biggest volume is chosen as the master-cell. Generally, more than two cells can be merged to form a hyper-cell.

PUMA [4] uses an unstructured grid with hexahedral elements, the generation of which will not be elaborated here, in the interest of space.

Flow prediction in a diaphragm pump requires the numerical solution of the unsteady incompressible Navier-Stokes equations. These are written as,

$$\Gamma_{ij}^{-1}\frac{\partial V_j}{\partial \tau}+\frac{\partial U_i}{\partial t}+\frac{\partial f_{ik}^{inv}}{\partial x_k}-\frac{\partial f_{ik}^{vis}}{\partial x_k}=0, \qquad i=1,4, \qquad \Gamma=\begin{bmatrix}\beta^2 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \tag{1}$$

where the Einstein notation is used. Index $k=1,2,3$ refers to the Cartesian directions; (x_1,x_2,x_3) stand for (x,y,z) and (u_1,u_2,u_3) to the corresponding Cartesian velocity components, $\vec{V}=\begin{bmatrix}p & u_1 & u_2 & u_3\end{bmatrix}^T$, $\vec{U}=\begin{bmatrix}0 & u_1 & u_2 & u_3\end{bmatrix}^T$, $\vec{f}_k^{inv}=\begin{bmatrix}u_k & u_1u_k+p\delta_k^1 & u_2u_k+p\delta_k^2 & u_3u_k+p\delta_k^3\end{bmatrix}^T$ and $\vec{f}_k^{vis}=\begin{bmatrix}0 & \tau_{1k} & \tau_{2k} & \tau_{3k}\end{bmatrix}^T$, where δ_i^j is the Kronecker symbol, p is the pressure divided by the density, t is the real time, τ is the pseudo-time and $\tau_{ik}=\nu(\frac{\partial u_i}{\partial x_k}+\frac{\partial u_k}{\partial x_i})$ are the viscous stresses. Finally, Γ is the preconditioning matrix used to stabilize the numerical solution, depending on β which stands for the pseudo-compressibility coefficient. Flows considered in this paper are laminar. In PUMA, the discretization of eqs. (1) is based on the vertex-centered finite volume approach. On the other hand, the cell-centered approach is used in the cut-cell software. In both solvers, inviscid flux f_{ik}^{inv} is computed based on the Roe's approximate Riemann solver, [9].

The diaphragm motion is taken into account in the flow equations' discretization. Both solvers use the Reynolds theorem to compute the "physical" time derivatives, namely, the integration of the time derivative of U_i in a deformable finite volume Ω_{FV} is given as

$$\begin{aligned}\int\limits_{\Omega_{FV}^{k+1}} \frac{\partial U_i}{\partial t}\, d\Omega &= \frac{d}{dt}\int\limits_{\Omega_{FV}^{k+1}} U_i\, d\Omega - \int\limits_{S_{FV}^{k+1}} u_m^{grid} n_m U_i\, dS \\ &\simeq \frac{1}{2\Delta t}\left(3U_i^{k+1}\Omega_{FV}^{k+1} - 4U_i^k\Omega_{FV}^k + U_i^{k-1}\Omega_{FV}^{k-1}\right) - \sum_{faces} u_m^{grid} n_m U_i^{k+1}\Delta s \end{aligned} \qquad (2)$$

where Ω_{FV}^{k+1}, Ω_{FV}^{k}, Ω_{FV}^{k-1} is the volume of the same finite volume at time instants $k+1$, k and $k-1$, respectively. u_m^{grid} stands for the grid velocity (i.e. the grid deformation rate) and is computed in a way ensuring the satisfaction of the geometric conservation law. n_m is the unit normal pointing outwards the finite volume and ΔS the area of the finite volume faces. PUMA morphs the grid at each time instant employing a spring analogy technique, [10]. Fig. 3 shows two grids (used by PUMA) where, on the left, the diaphragm is in its resting position whereas, on the right, the diaphragm is pushed down. Regarding the cut-cell method, the Cartesian grid is continuously adapted to the diaphragm's motion. First, a coarsening procedure applies and cells in the vicinity of the diaphragm are coarsened. Then, cells cut by the new diaphragm's geometry are refined and the neighbouring cells are adapted accordingly. The flow field computed at the previous time instant(s) is transferred to the new grid for computing time derivatives of the flow variables. Transfer is conservative by taking into account that cut-cells change in shape or migrate from the fluid to the solid region. Among other, solidified cells merge their values with those of neighbouring fluid cells. Fig. 4 shows three time-steps while the valve moves from open to closed position. A Cartesian grid, adapted to the valve's motion, is generated anew at each time-step.

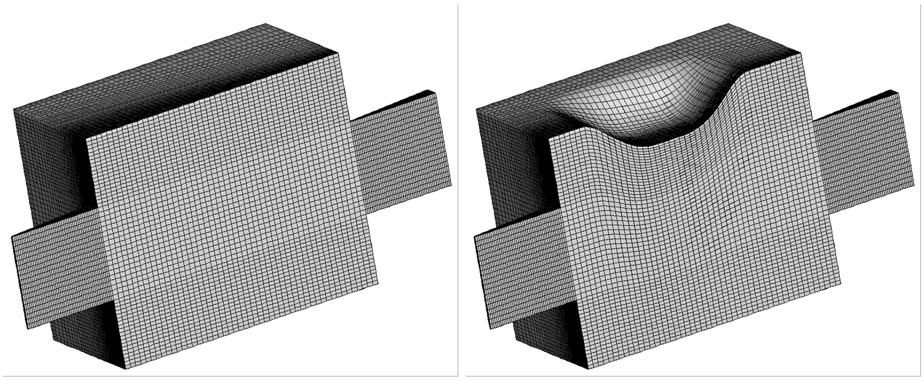
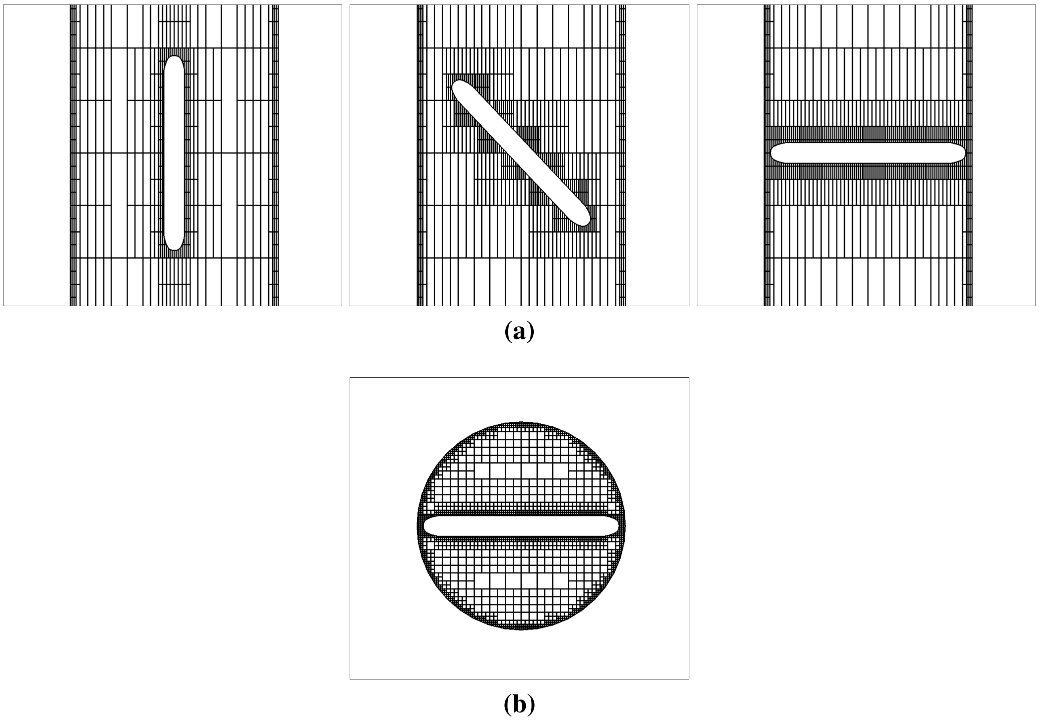


Figure 3: Perspective views of the unstructured CFD grids used by PUMA code at time instants 0 and $0.45T$. Axes not in scale, y-axis is 14 times greater.



(a)

(b)

Figure 4: (a) Adapted Cartesian grid for three positions of the butterfly valve. (b) View of the valve in the fully open position.

Adjoint Equations and Gradient Computation

Three metrics are used to measure the quality of the resulting flow in a diaphragm pump or its components (valved ducts), namely

- The volume of the fluid entering pump's exit (back-flow, Q_{bf}; a non-negative quantity, by definition) within a period T. It is defined as $Q_{bf} = -\int_T \int_{S_O} min(0, u_m n_m^O) dSdt$, where n^O is the unit normal to the pump's outlet (S^O).
- The net volume of fluid pumped within T, i.e. $Q_{net} = \int_T \int_{S_O} u_m n_m^O dSdt$.
- The total pressure losses ($Q_{pt} = p_t^I - p_t^O$) between (valved) duct's inlet and outlet.

Depending on the application, the objective function could be any of these metrics ($F \equiv Q$) or a combination of them; in regard to Q_{net} since the usual target is to achieve a desired volume flow rate, this might give an objective function of the form $F = |Q_{net} - Q_{net,ref}|$. Among other, below, a single-objective optimization of the pump is performed to minimize $F = \frac{Q_{bf}}{Q_{net}}$.

The gradient of F w.r.t. b_i (defined in section) is computed using the continuous adjoint method. The augmented objective function (to be differentiated instead of F) is

$$F_{aug} = F + \int_0^T \int_\Omega \Psi_i R_i d\Omega dt \tag{3}$$

where R_i are the residuals of the flow equations, Ψ_i are the adjoint variable fields, Ω is the fluid domain at t. By differentiating F_{aug} w.r.t. b_i and setting the multipliers of variations in the flow variables to zero, the unsteady adjoint equations along with their boundary conditions arise.

The unsteady adjoint equations are

$$\Gamma_{ji}^{-1}\frac{\partial \Psi_j}{\partial \tau} - \frac{\partial \bar{\Psi}_i}{\partial t} - A_{jik}\frac{\partial \Psi_j}{\partial x_k} - \frac{\partial f_{ik}^{A,vis}}{\partial x_k} = 0 \tag{4}$$

where $\vec{\bar{\Psi}} = \left[0\,\Psi_2\,\Psi_3\,\Psi_4\right]^T$, $\vec{f}_k^{A,vis} = \left[0\,\tau_{1k}^A\,\tau_{2k}^A\,\tau_{3k}^A\right]^T$ stands for the adjoint viscous flux and $\tau_{ik}^A = \nu\left(\frac{\partial \Psi_{i+1}}{\partial x_k} + \frac{\partial \Psi_{k+1}}{\partial x_i}\right)$ are the adjoint stresses. Similarly to the flow equations, a pseudo-time derivative multiplied with the inverse of the preconditioning matrix Γ has been added in eq. 4. The integration of the temporal term on deformable finite volumes is similar to the flow equations solver as in eq. 2.

For the pump's wall boundaries (including the diaphragm), the adjoint velocity is set to zero. Given the fact that the inlet/outlet duct shapes do not change in time, the adjoint inlet/outlet condition reads

$$\Psi_i A_{ijk} n_k^{I/O}\frac{\partial V_j}{\partial q_l^{I/O}} + \frac{\partial \hat{F}}{\partial q_l^{I/O}} = 0 \tag{5}$$

where $\hat{F}$ is the integrand of F. Working with Q_{bf} in particular, since function "min" cannot be differentiated, this is replaced with a sigmoid function in eq. 5. $q_l^{I/O}$ stands for flow quantities extrapolated from the CFD domain (like the velocity components at the outlet or the velocity magnitude at the inlet).

Considering that period T is constant, the gradient of F w.r.t. b_i becomes an expression of the computed primal and adjoint fields, as follows

$$\begin{aligned}\frac{\delta F}{\delta b_i} = &- \int_0^T \int_\Omega \left[\Psi_n \frac{\partial V_n}{\partial x_l}\frac{\partial}{\partial t}\left(\frac{\delta x_l}{\delta b_i}\right) + \left(\Psi_n \frac{\partial f_{nk}^{inv}}{\partial x_l} - \Psi_n \frac{\partial f_{nk}^{visc}}{\partial x_l}\right)\frac{\partial}{\partial x_k}\left(\frac{\delta x_l}{\delta b_i}\right)\right] d\Omega dt \\ &- \int_0^T \int_\Omega \tau_{km}^{adj}\frac{\partial u_k}{\partial x_l}\frac{\partial}{\partial x_m}\left(\frac{\delta x_l}{\delta b_i}\right) d\Omega dt + \int_0^T \int_{S^W}\left(\Psi_1 n_k - \tau_{km}^{adj} n_m\right)\frac{\delta u_k^{grid}}{\delta b_i} dS dt \\ &+ \int_0^T \int_{S^{I/O}} \frac{\partial \hat{F}}{\partial x_l}\frac{\delta x_l}{\delta b_i} dS dt\end{aligned} \tag{6}$$

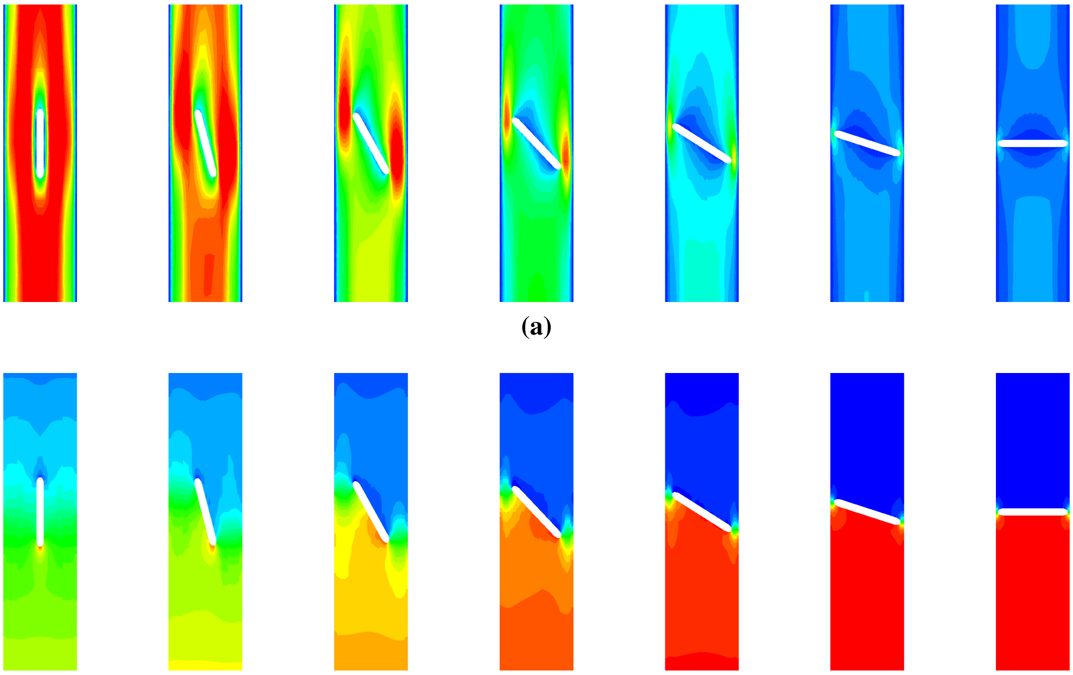
where S^I, S^W stand for the inlet and diaphragm surface, respectively. Note that in the cut-cell method, $\frac{\delta x_l}{\delta b_i}$ takes non-zero values only along the diaphragm.

PCA-Driven Metamodel-Assisted Evolutionary Algorithms

A (μ, λ) EA, with μ parents and λ offspring in each generation, is the background stochastic optimization tool. Metamodels assist the EA (Metamodel Assisted EA-MAEA), so as to reduce its computational cost. Metamodels are Radial Basis Function (RBF) network trained on-line during the evolution on individuals evaluated on the CFD tool and used, when needed, to predict the objective function value(s) of new individuals at practically negligible cost compared to a CFD run. In each generation (excluding the very first ones, in which there is no enough data to train the metamodels), each population member is pre-evaluated with personalized locally trained metamodels and, then, only the few most promising ones are re-evaluated with the CFD tool. Training patterns are selected, using several criteria, from the pool (database) of individuals evaluated during the preceding generations. Aiming at maximum performance, the Kernel Principal Component Analysis (PCA) method drives the EA as mentioned in [7]. Briefly, prior to applying the evolution operators, the design space is transformed into the feature space in which the evolution operators perform better and the metamodels are trained only with the significant variables as computed by the method. This EA assisted by metamodels and the PCA, referred to as the PCA-driven MAEA [11] [12] [7], has been developed within the EASY platform [13] of NTUA, is also used for the optimization.

Method Demonstration in a Valved Inlet/Outlet Duct

The valve is modeled as a disc-shaped shutter rotating around an axis/rod (fig. 1b). Applications like the flow inside a duct with a rotating valve, perfectly suit to the cut-cell CFD software, as this avoids morphing a body-fitted grid (without changing grid topology). Thus, only the cut-cell software is used here to compute the flow field inside the duct while the valve rotates from the open to the closed position. Velocity magnitude and pressure fields, for different valve's positions, are shown in fig. 5, respectively. Flow trajectories around the rotating disc are presented in fig. 6. The unsteady adjoint equations have also been solved giving the necessary information to compute the sensitivity derivatives of the Δp_t function, defined between the inlet and the outlet of the duct. A time history of the adjoint velocity magnitude and adjoint pressure fields is shown in fig. 7. 3D views of the adjoint field around the disc-shaped shutter, at different positions is shown in 8.



(a)

(b)

Figure 5: Top: Instantaneous velocity magnitude fields within the valved duct, at equally distributed time instants. Bottom: Instantaneous pressure fields at the same time instants.

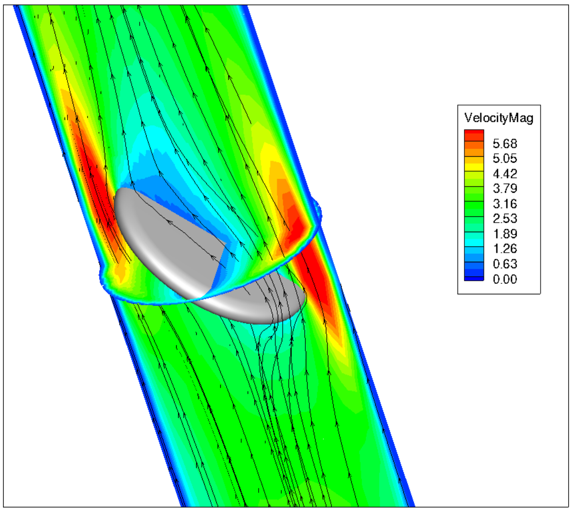
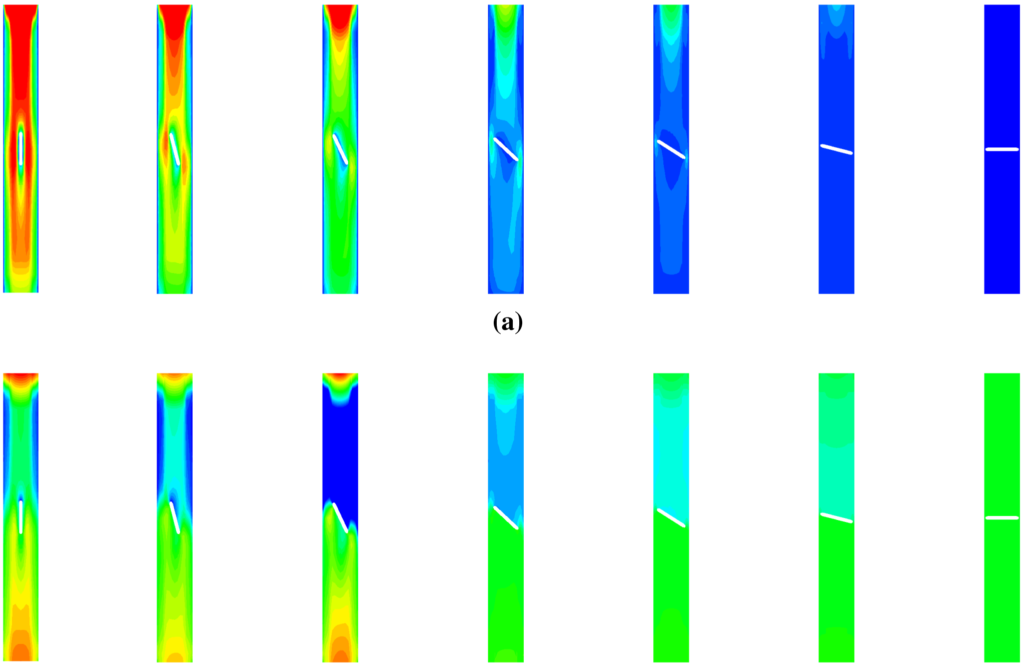


Figure 6: Flow trajectories within the valved duct with the disc half-open.


(a)

(b)

Figure 7: Top: Instantaneous adjoint velocity magnitude fields within the valved duct, at equally distributed time instants. Bottom:Instantaneous adjoint pressure fields at the same time instants.

Method Demonstration in a Valveless Micropump

A first valveless micropump (the way this has been constructed is beyond the scope of this paper) is optimized at first; this pump (to be referred to as the “reference” one; index “ref”) delivers the desired net volume of fluid per period ($T = 0.02s.$), with a non-negligible backflow rate though. Therefore, it was decided to run a two-objective optimization aiming at minimum $|Q_{net} - Q_{net,ref}|$ and minimum Q_{bf}. Next to this, two single-objective optimizations are also performed, both with the same target; one of them is based on the PCA-driven MAEA and the other on the GB method supported by the adjoint solver. It should be noted that, in the single-objective optimizations, the (common) target was to minimize $\frac{Q_{bf}}{Q_{net}}$; thus, any comparison with

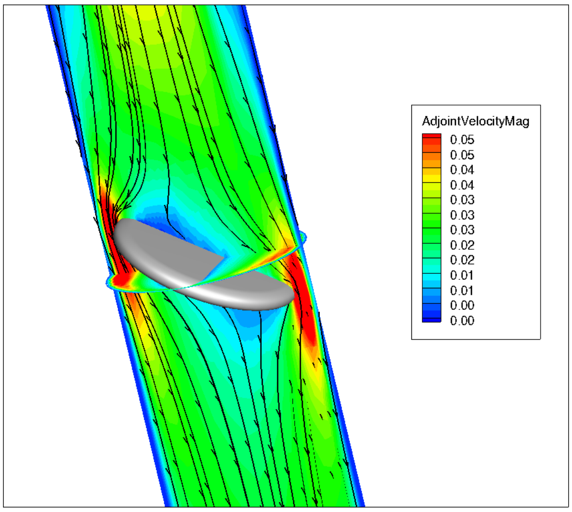


Figure 8: Adjoint flow trajectories (for Δp_t as the objective function) within the valved duct with the shutter half-open.

the two-objective optimization is made only for the purpose of comparing computational cost and not the quality of the optimal solutions obtained by the various methods.

The PCA-driven MAEA is configured with $\mu = 6$ and $\lambda = 12$ and both metamodels and PCA are activated after the first generation. The computational budget is limited to 200 CFD runs. The convergence history of the single-objective optimization is shown in fig. 10. For the single-objective problem, the resulted optimal motion has reduced the objective function $\left(\frac{Q_{bf}}{Q_{net}}\right)$ by two orders of magnitude from the initial/reference one. Regarding the two-objective optimization, the final front of non-dominated solutions is presented in fig. 9. Notice that the optimal solution of the first optimization is the bottom right end of the front.

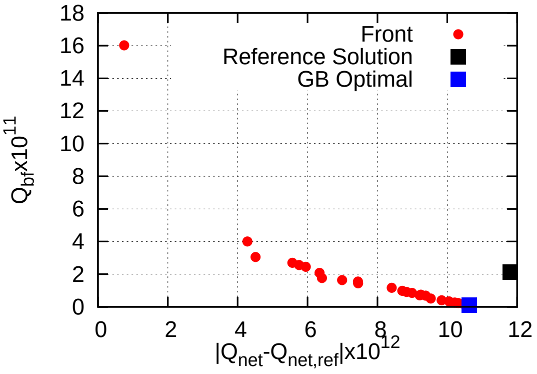


Figure 9: Front of non-dominated solutions (red points) computed by the PCA-driven MAEA in the two objective optimization. Reference (black point) and the optimal (found by the GB optimization) solution (blue point) are included, too.

In regard to the GB optimization, in which the cost of the solution of the adjoint equations is, on the average, equal to the cost of solving the flow equations, each optimization cycle approximately costs as much as two CFD calls. Steepest-descent is used with an adaptive step size per optimization step. The convergence of the GB optimization is shown in fig. 10. After 36 CFD calls, this practically leads to the same optimal solution with the one found by the

PCA-driven MAEA for the same objective. Note that the PCA-driven MAEA could have been stopped earlier since, during the last generations, the optimal solution does not change.

Regarding the flow quality metrics for the GB optimization, the net volume flow is increased from $Q_{net,ref} = 5.18mm^3$ for the reference motion to $Q_{net} = 6.23mm^3$ for the optimal one and the back-flow is reduced from $Q_{bf,ref} = 2.12mm^3$ to $Q_{bf} = 0.255mm^3$. The optimal and reference diaphragm's motion are shown at various time instants in figs. 11, 12, respectively. Fig. 13 shows that the optimal micropump has reduced the internal vortical flow in the last time instants and the maximum deformation of the diaphragm has been limited which assists at minimizing the back-flow at the exit.

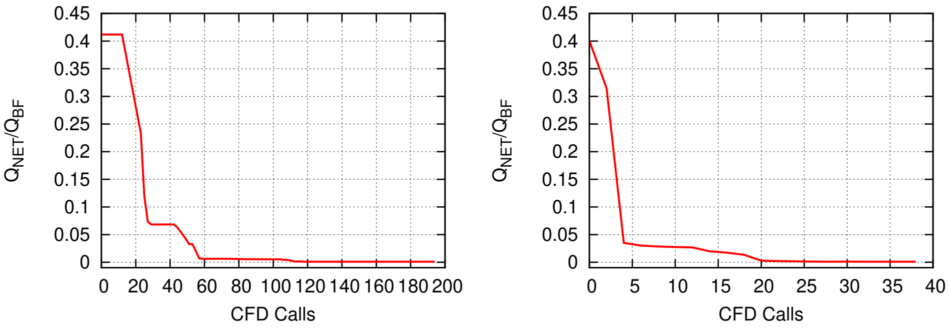


Figure 10: Convergence history of the micropump's optimization. (Left) PCA-driven MAEA, (right) Gradient-Based method.

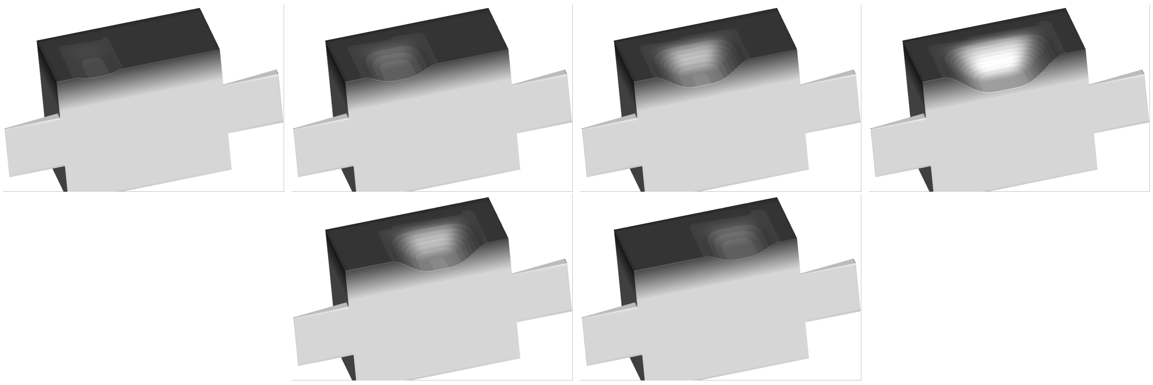


Figure 11: Reference diaphragm motion at the symmetry plane. Time instants: $0.20T$, $0.30T$, $0.40T$, $0.50T$, $0.60T$ **and** $0.70T$**, from top-left to bottom-right respectively.**

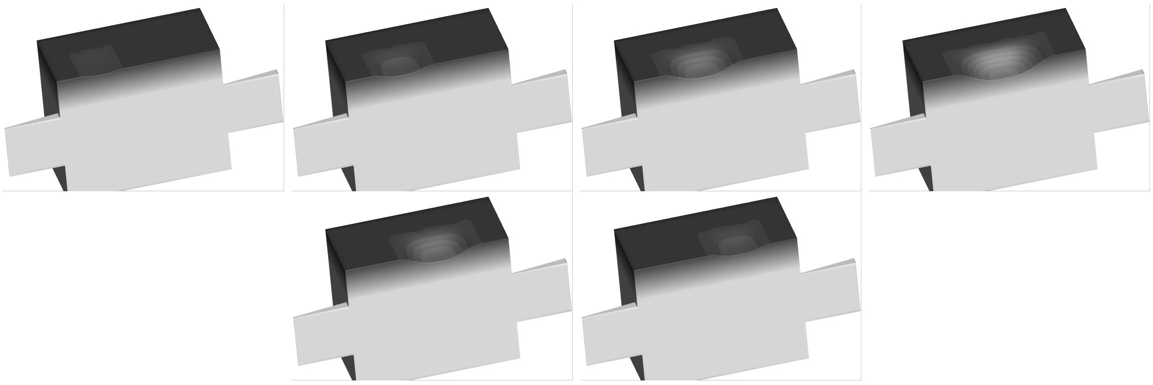


Figure 12: Optimal diaphragm motion at the symmetry plane. Time instants: $0.20T$, $0.30T$, $0.40T$, $0.50T$, $0.60T$ and $0.70T$, from top-left to bottom-right respectively.

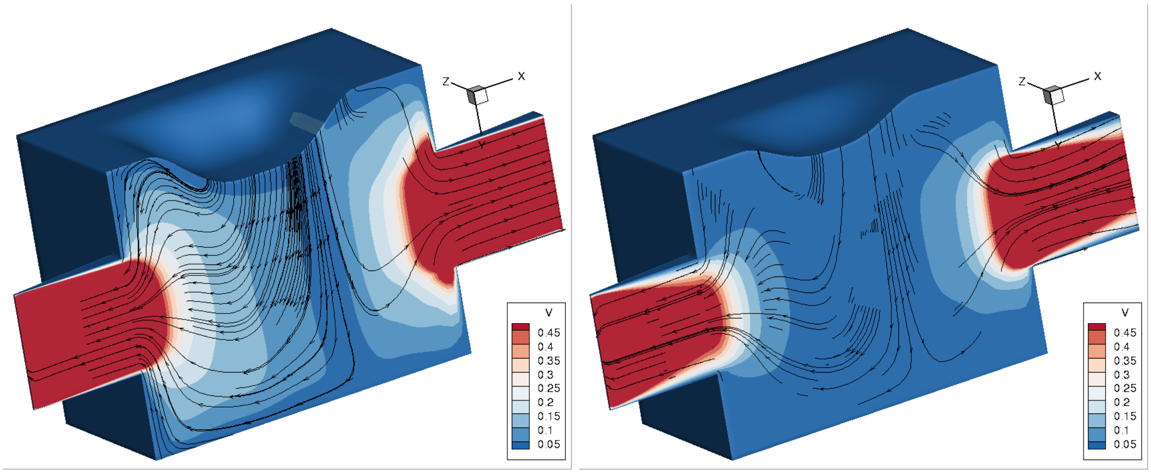


Figure 13: Velocity magnitude iso-areas on the symmetry plane at time instant $0.45T$. Reference (left) and optimized (right) motion. Axes not in scale, y-axis is 14 times greater.

Conclusion

This paper focuses on particular features of the analysis and optimization tools for valveless and valved diaphragm pumps. For both pump types, the diaphragm motion is parameterized and metrics quantifying the quality of the resulting flow inside the pump or its components are defined. Performing the optimization through gradient-free methods, their computation is all we need. In this paper, the gradient-free optimization is performed using a metamodel-assisted evolutionary algorithm, additionally supported by the principal component analysis of the population, for the purpose of cost reduction. For the use of gradient-based optimization, though, the objective functions in use should also be differentiated; for this purpose, the continuous adjoint method is employed. From the CFD viewpoint, it is very convenient to possess more than one tools. Our first CFD tool is based on a cut-cell method, being more appropriate in case of valves that periodically open and close. Next to this, a standard CFD solver with body-fitted unstructured grids is used. For both codes, the same adjoint methods for the same objective functions have been developed. Some selected analyses and optimizations have been performed and demonstrated the capabilities of the available tools. Their integration in an automatic workflow, which will involve both optimization methods and both CFD tools, is in progress.

Acknowledgment

Research presented in this paper has been funded by the Business Plan "Development of Human Resources, Education and Lifelong Learning" entitled "Support Researchers with Emphasis on Young Researchers" with the co-financing of Greece and the European Union. The project title is "*Design-Optimization of Diaphragm Pumps under Operational/Manufacturing Uncertainties using the Cut-Cell Method and Polynomial Chaos Expansion*".

References

[1] H. Chun-Wei, H. Song-Bin, and L. Gwo-Bin, "A microfluidic device for precise pipetting," *Journal of Micromechanics and Microengineering*, vol. 18, no. 3, pp. 35–39, 2008.

[2] L. Songjing, J. Liu, and D. Jiang, "Dynamic characterization of a valveless micropump considering entrapped gas bubbles," *Journal of Heat Transfer*, vol. 135, 2013.

[3] K. Samouchos, D. K. Kapsoulis, X. Trompoukis, and K. Giannakoglou, "Design of a diaphragm pump under uncertainties using the continuous adjoint to the cut-cell method.," in *6th European Conference on Computational Mechanics (ECCM 6) - 7th European Conference on Computational Fluid Dynamics (ECFD 7)*, June 11-15 2018.

[4] I. Kampolis, X. Trompoukis, V. Asouti, and K. Giannakoglou, "CFD-based analysis and two-level aerodynamic optimization on graphics processing units," *Computer Methods in Applied Mechanics and Engineering*, vol. 199, no. 9-12, pp. 712–722, 2010.

[5] K. Samouchos, S. Katsanoulis, and K. Giannakoglou, "Unsteady adjoint to the cut-cell method using mesh adaptation on GPUs," in *ECCOMAS Congress, Crete Island, Greece*, July.

[6] C. Peskin, "Flow patterns around heart valves: a numerical method," *Journal of Computational Physics*, vol. 10, no. 2, pp. 252–271, 1972.

[7] D. Kapsoulis, K. Tsiakas, X. Trompoukis, V. Asouti, and K. Giannakoglou, "Evolutionary multi-objective optimization assisted by metamodels, kernel pca and multi-criteria decision making techniques with applications in aerodynamics," *Applied Soft Computing*, vol. 1, no. 2, pp. 182–197, 2017.

[8] M. Aftosmis, M. Berger, and J. Melton, "Robust and efficient cartesian mesh generation for component-based geometry," *AIAA Journal*, vol. 36, 11 2000.

[9] P. Roe, "Approximate Riemann solvers, parameter vectors, and difference schemes," *Journal of Computational Physics*, vol. 43, no. 2, pp. 357–372, 1981.

[10] Y. Yang and S. Özgen, "Comparison of various spring analogy related mesh deformation techniques in two-dimensional airfoil design optimization," *Progress in Flight Physics*, vol. 9, pp. 189 – 204, 2017.

[11] A. Giotis and K. Giannakoglou, “Low-cost genetic optimization based on inexact pre-evaluations and the sensitivity analysis of design parameters,” *Advances in Engineering Software*, vol. 29, no. 2, pp. 129–138, 1998.

[12] M. Karakasis, A. Giotis, and K. Giannakoglou, “Inexact information aided, low-cost, distributed genetic algorithms for aerodynamic shape optimization,” *International Journal for Numerical Methods in Fluids*, vol. 43, no. 10-11, pp. 1149–1166, 2003.

[13] K. Giannakoglou. The EASY (Evolutionary Algorithms SYstem) software, http://velos0.ltt.mech.ntua.gr/EASY., 2008.