Real-time prediction of the deformation of microcapsules using Proper Orthogonal Decomposition

Carlos Quesada, Pierre Villon, Anne-Virginie Salsac

Real-time prediction of the deformation of microcapsules using Proper Orthogonal Decomposition

Carlos Quesada[a], Pierre Villon[b], Anne-Virginie Salsac[a*]

[a] *Biomechanics and Bioengineering Laboratory (UMR CNRS 7338), Université de Technologie de Compiègne–CNRS, Compiègne, France*
[b] *Roberval Laboratory, Université de Technologie de Compiègne, Compiègne, France*

Abstract

Microcapsules are liquid droplets enclosed by a thin elastic membrane. Being suspended in an external fluid, they undergo large deformations when flowing. Their deformation can be solved numerically, but the resolution of the fluid-structure interactions (FSI) requires extremely long computation times. This is a major constraint for instance when identifying the membrane mechanical properties from experimentations of microcapsules flowing in a microfluidic channel of similar size (Hu et al PFE 2013). We propose to apply Model Order Reduction (MOR) to predict in real time the steady-state capsule deformed shape, needed to determine the membrane elasticity. A database of the capsule deformed shapes was obtained numerically by solving the three-dimensional FSI through a finite element - boundary integral method coupling, and by varying systematically the two non-dimensionalized parameters of the problem: the capillary number, ratio of the viscous to the elastic forces, and the capsule-to-tube size ratio. Among the MOR techniques, we chose to apply Proper Orthogonal Decomposition (POD) onto the database, which provides a vector basis of principal components, defining a multi-dimensional vector space. The advantage is that, when all the capsule shapes of the database are mapped into this new vector space, they form a manifold (smooth hypersurface) that represents all the admissible solutions of the problem. We show that POD with a manifold walking technique can be successfully applied to microcapsule data sets, whether they are Lagrangian (e.g. known position vector fields) or Eulerian (i.e. when data is acquired experimentally or numerically using methods like level sets). In both

*Corresponding author
Email address: a.salsac@utc.fr (A.-V. Salsac)

cases, the problem dimensionality is reduced, the predicted capsule shapes are obtained within computation times of milliseconds, and they accurately fit the full FSI simulations. This paves the way to real-time computations for capsules in flow, while retaining all the physical ingredients of the FSI problem.

Keywords: Microcapsules, Reduced order model, Proper Orthogonal Decomposition, Manifold, Deformed capsule shape
2010 MSC: 58Z05, 65Z05, 76D07, 74K15, 74S15, 74S05

1. Introduction

Capsules are liquid droplets, enclosed within a thin deformable solid membrane, whose size ranges from a few microns to a few millimeters. Present in nature in the form of red blood cells, vesicles or eggs, capsules also have multiple applications across industries. Artificial microcapsules are, among others, used extensively in the biotechnological (Ma and Su, 2013), pharmaceutical (Cole et al., 2008; Yih and Al-Fandi, 2006), cosmetic (Miyazawa et al., 2000), food (Gibbs et al., 1999) and textile industries (Nelson, 2002) to protect material (active substances, aromas, drugs or transgenic cells) from the surrounding environment, and to control its targeted delivery when desired. Whether natural or artificial, capsules are typically suspended in an external fluid and are thus subjected to hydrodynamic forces when the fluid is in motion. This leads to strongly coupled fluid–structure interactions (FSI), which tend to be highly non-linear owing to the large deformations experienced by the capsule membrane (Barthès-Biesel, 2011).

Over the last three decades, a very large number of numerical studies have investigated capsule–flow interactions (see Barthès-Biesel (2016) for a review). They are complex to model, as microcapsules are three-dimensional closed objects with a liquid core and a hyperelastic membrane. Their three-dimensional deformation and flow are thus basically inertialess, which is non-conventional within the field of fluid–structure interaction numerical modeling. Since the end of the 1990's, the investigation of capsule dynamics has been complemented by an increasing number of experimental studies thanks to the advent and rise of microfluidic and microrheological technologies. The latter have indeed made it possible to study the motion and deformation of microcapsules with high accuracy (e.g. Chu et al. (2011); Koleva and Rehage (2012); De Loubens et al. (2014); Gires et al. (2016)).

Numerical and experimental approaches have many points in common. In both cases, the generated data may either be Lagrangian or Eulerian, depending on the numerical schemes and experimental acquisition technique. But the large body of previous studies on capsules and cells has shown that a 3D Lagrangian follow-up of the points of the capsule membrane is necessary to determine the overall dynamics of the deformable particle (e. g. Fischer et al. (1978); Abkarian et al. (2007); Dupont et al. (2016)). This explains why the classical numerical algorithm, since the pioneering work of Barthès-Biesel and Rallison (1981), consists of a Lagrangian tracking of the membrane nodes, whose time evolution is obtained from the integration of the capsule wall velocity provided by the fluid solver. But novel strategies based on a pure Eulerian description are, however, emerging to solve FSI problems (Cottet et al., 2008). They rely on a numerical analysis of the particle shapes and surfaces on a fixed framework in order to determine the strain and deformation fields without any explicit parametrization of the moving particle. Recently, they have been successfully applied to capsules in flow (Milcent and Maitre, 2016). Experimentally, the acquired data are by essence Eulerian, the only way to perform Lagrangian tracking being to place tracers on the capsule membrane, which is cumbersome and difficult to achieve on such small particles. Generally, experimental images thus consist of two-dimensional grayscale images, and their post-processing involves image processing techniques to extract relevant features such as the capsule deformed shape.

Another common feature of numerical and experimental approaches is the large set of data that are generated. In both cases, they necessitate large storage capacity and their post-processing tends to be time-consuming. Numerical simulations additionally face the issue of the long computational time required to solve the strongly coupled fluid-structure interaction (FSI) schemes.

Model order reduction (MOR) techniques can help decrease such model and data complexity by reducing the dimensionality of the generated data. Very significant time saving can be achieved by transforming data that depend on a large number of parameters into a small set of principal components that preserves the main original characteristics. Some of the usual MOR techniques are the Proper Orthogonal Decomposition (POD) (Jolliffe, 1986; Loeve, 1948), the Proper Generalized Decomposition (Ammar et al., 2006, 2007; Chinesta et al., 2013), and the Reduced Basis (Maday and Rønquist, 2002; Eftang, 2011) methods.

MOR techniques, which have never been applied to predict the deforma-

tion of capsules in flow, can offer the framework needed to solve the scientific challenges that remain to be met to generalize the use of microcapsules in health-related applications. Microcapsules indeed have the potential to become a solution of choice for encapsulation, as they can carry much larger quantities of active substances than the nanoparticles, which are typically used at present. But to become a robust and reliable vector solution, one needs to be able to predict their deformation and membrane tension under flow (and thus their potential rupture) and characterize the membrane resistance in order to find the optimal compromise between deformability and resistance for each application. New techniques of characterization have been devised (Chu et al., 2011; Hu et al., 2013; De Loubens et al., 2014; Gubspun et al., 2016) as microfluidics matured over the last decade: the shear/Young moduli of the capsule wall have been determined in microcapillary/extensional flows by comparing the experimental steady-state capsule profiles to databases of profiles calculated numerically. Such experiments are much less cumbersome to perform on entire batches of microcapsules than atomic force microscopy indentation or micropipette aspiration, and can be automatized. But the inherent issue associated with them is the need to run the full theoretical fluid-structure interaction model corresponding to the experiment to identify the mechanical property, as it is computationally costly. The objective of the paper is to analyze whether MOR (and in particular Proper Orthogonal Decomposition) is suitable to reduce the dimensionality of the complex problem and sufficiently precise to determine the steady-state deformation of a capsule in flow without launching the full-simulation. We aim at identifying if it is possible to accurately predict the capsule deformation in real time, and if it can be done with equal precision whether the data is Lagrangian or Eulerian.

We focus on the identification technique based on the steady-state flow of a microcapsule in a microfluidic channel of similar characteristic size (Chu et al., 2011; Hu et al., 2013). Its deformed profile (Figure 1) is known to be only a function of two non-dimensionalized parameters: the capsule-to-channel size ratio and the capillary number, ratio of the viscous to the elastic forces. The data sets used as input for the MOR procedure are the steady-state capsule shapes that we have computed off-line using a 3D FSI numerical scheme coupling the Finite Element (FE) method, used to solve for the capsule deformation, with the Boundary Integral (BI) Method, used to solve for the inner and outer fluid flows (Hu et al., 2012). They are computed varying the two controlling parameters in a systematic way and compiled in a comprehensive database. The advantage

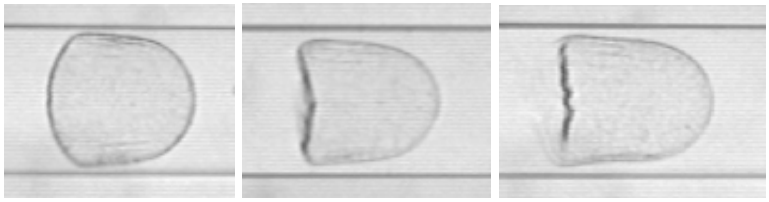


Figure 1: Pictures of ovalbumin microcapsules flowing in a microfluidic channel (cf Chu et al. (2011)).

of using numerical nodal data is that they are inherently Lagrangian, but can easily be transformed into a Eulerian field, by calculating, for instance, the signed distance of the points in space to the capsule membrane. We thus have the possibility to not only see whether reduced-order modeling approaches can be used to predict capsule shape, but also to compare the precision of POD when it is applied to Lagrangian and Eulerian similar-sized data sets.

We will first describe, in §2, the physics of the problem at stake and the numerical method of the FE-BI code, and detail how POD is applied to the database containing the steady-state capsule shapes. In §3, we will first analyze the precision of the capsule shapes reconstructed using POD by comparing them to the ones computed with the full FSI numerical code. We will try to predict the capsule deformation for parameter values that are not included within the database, by using a so-called manifold walking strategy. We will finally transform the Lagrangian numerical data into a Eulerian data field and compare the precision of the predictions for Lagrangian and Eulerian data sets. This will allow us to explore the potential of using POD to predict capsule shapes from experimental data (inherently Eulerian). In §4, we will eventually analyze the influence of the parameters of the POD method before providing concluding remarks in §5.

2. 3D numerical model of a capsule flowing in a tube and Proper Orthogonal Decomposition of the deformed capsule contours

2.1. Problem description

We consider an initially spherical capsule flowing in a microfluidic channel of comparable size. The channel is a long prismatic tube with a constant square cross-section of side 2ℓ (Fig. 2). The corners are rounded over one mesh in order to avoid solving for the flow in the vicinity of a sharp angle. The capsule is suspended in an incompressible Newtonian liquid of viscosity μ and density ρ that

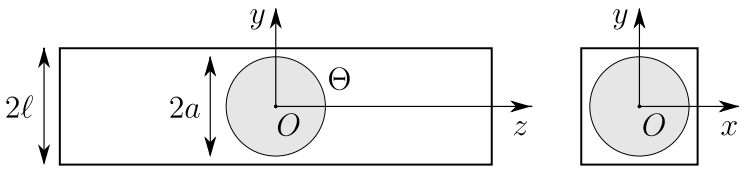


Figure 2: Longitudinal and cross-sectional views of the configuration modeled in the numerical simulation: an initially spherical capsule of radius a is positioned at the center O of a square-section microchannel of side 2ℓ and subjected to a Poiseuille flow oriented in the z-direction.

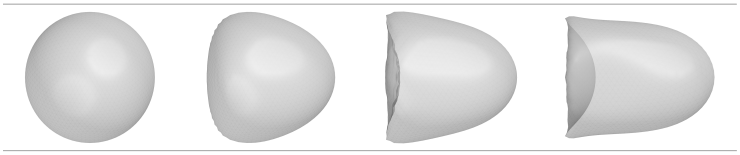
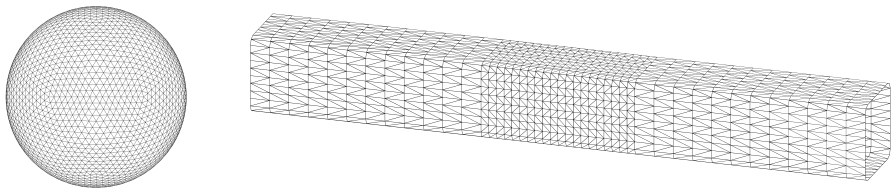


Figure 3: Snapshots, provided by the full 3D FSI numerical simulations, of the shapes for a capsule flowing along the microchannel from rest to steady state ($Ca = 0.12$ and $a/\ell = 0.9$). The capsule contours are given in the $x = 0$ plane at the non-dimensional times $Ut/\ell = 0$ (a), 0.2 (b), 0.6 (c) and 2.4 (d).

flows with mean velocity V. The velocity field in the absence of particle, $\boldsymbol{v}^{\infty}$, is the solution of Poiseuille flow within a duct with square cross-section (Pozrikidis, 1997; Hu et al., 2012).

At $t = 0$, the spherical capsule of radius a is placed at the middle O of the channel (Fig. 2). The capsule is chosen to be filled with the same liquid as the external fluid to avoid buoyancy effects and because the capsule profile is, at steady-state, independent of the properties of the internal fluid, as it is at rest. The capsule wall is assumed to be infinitely thin and made of an impermeable, hyperelastic isotropic material. It is considered to follow the neo-Hookean (NH) law, with a surface shear elastic modulus G_s and an area dilatation modulus $K_s = 3G_s$ (Barthès-Biesel et al., 2002). This law corresponds to a strain-softening behavior under large deformation. It has been chosen, because previous experimental studies have shown that the membrane of ovalbumin and human serum albumin capsules obey to this law, and that it is the most appropriate to describe their mechanical behaviour (Chu et al., 2011; Hu et al., 2013; Gubspun et al., 2016). When placed in flow, the elastic membrane deforms under the hydrodynamic forces (see the snapshots of the capsule shape provided at various instants of time in Fig. 3).



(a) (b)

Figure 4: Meshes of the undeformed spherical capsule (a) and of the prismatic channel (b).

2.2. Numerical resolution of the fluid-structure interactions

Owing to the micrometric size of the capsule, inertia effects are negligible for both the fluids and the capsule membrane. The fluid-structure interaction problem is thus quasi-steady and the fluid flows obey the Stokes equations. The problem is solved using the numerical method described in Walter et al. (2010) and (Hu et al., 2012), which couples a boundary integral method for the flow with a finite element method for the membrane mechanics.

The mesh of the capsule is generated by inscribing a regular icosahedron in a sphere, and subdividing its faces sequentially into 4 new elements. This is repeated until the desired number of elements is reached (Walter et al., 2010; Hu et al., 2012). In this work, the capsule mesh consists of $n = 2\,562$ nodes, i. e. 5 120 elements (Fig. 4a). The mesh of the tube has 1 905 nodes (i. e. 3 768 elements) and has been refined in its central region, where the capsule is placed (Fig. 4b).

The equations are solved within the frame of reference of the moving capsule, the origin O of the coordinate system (Fig. 2) being matched with the center of mass of the capsule at each time step. Thus, at each simulation time step, the velocity of the center of mass is calculated and the entire capsule is moved back by a corresponding distance, so that it remains centered in the refined domain of the tube.

The two main input parameters of the numerical method are the size or confinement ratio a/ℓ, which represents the ratio between the capsule initial radius and the channel cross dimension, and the capillary number $Ca = \mu\ V/G_s$, which is the ratio between the viscous forces and the elastic forces in the membrane. The outputs are the deformed capsule shape, the velocity of the capsule centroid and the elastic tension distribution in the membrane. The time required to calculate the flow and deformation of a capsule from its initial position at rest until it reaches steady-state ranges from a few days to a few weeks using a

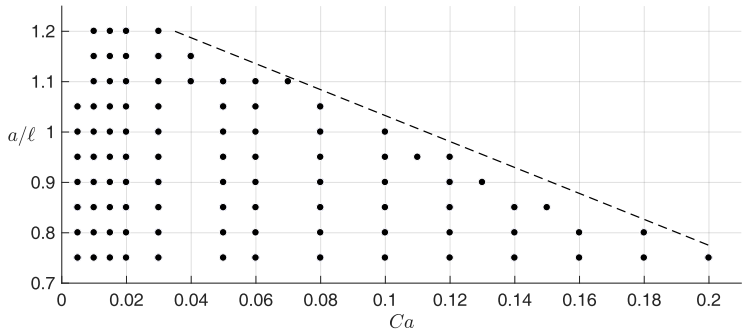


Figure 5: Values of the capillary number Ca and of the confinement ratio a/ℓ for which the steady-state capsule deformation have been included in the database. No steady-state deformation can be obtained above the dashed line for the neo-Hookean constitutive law.

computer with the current computational power, depending on the selected values of the 2 input parameters. The time step varying linearly with the capillary number, the computational time increases when the capillary number becomes very small compared to 1.

2.3. Database of capsule deformed shapes

Simulations of capsules in flow have been run varying the two governing parameters in the range $[0, 0.2]$ for the capillary number and $[0.75, 1.2]$ for the confinement ratio. Only the simulations for which a steady-state shape was reached were retained in the database. Let $\mathcal{P}$ be the set of values of capillary numbers Ca and confinement ratios a/ℓ for the 95 simulations that have been selected (Fig. 5). No deformed capsule shape exists for simulation parameters above the dashed line. This phenomenon, described in Barthès-Biesel (2011) and clearly illustrated in the experimental study of Chu et al. (2011), is a direct consequence of the fact that many capsules (like the ones with a proteic membrane) exhibit a strain-softening mechanical behavior. The database is then constituted of the set of the three-dimensional deformed capsules shapes computed at steady-state for the 95 simulations.

Fig. 6 shows examples of capsule contours included in the database. At low values of Ca, the initially spherical capsule takes a bullet-like shape at steady state (Figs. 6a and 6d). When the capillary number is increased, a change in concavity occurs at the rear of the capsule: the capsule then has a parachute-shape (Figs. 6b-c and 6e-f). The capsule is limited in the x- and y-directions by the channel walls, but extends in the z-direction under the hydrodynamic

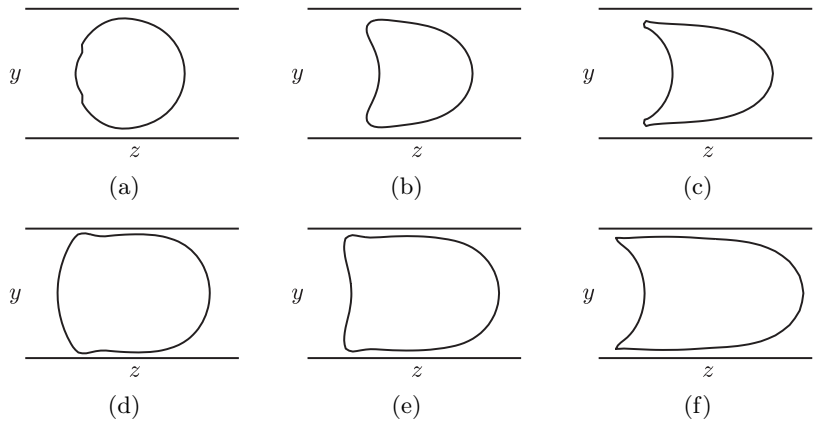


Figure 6: Examples of steady-state capsule contours in the $x = 0$ plane. (a) $Ca = 0.005, a/\ell = 0.85$, (b) $Ca = 0.06, a/\ell = 0.85$, (c) $Ca = 0.15, a/\ell = 0.85$ (d) $Ca = 0.01, a/\ell = 1.1$, (e) $Ca = 0.03, a/\ell = 1.1$, (f) $Ca = 0.07, a/\ell = 1.1$.

forces: the capsule length depends on the confinement ratio and varies between 1.3ℓ and 2.9ℓ in the cases shown in Fig. 6.

3. Proper Orthogonal Decomposition of the capsule deformed shapes

3.1. POD of the 3D Lagrangian position vectors

Proper Orthogonal Decomposition is an efficient statistical technique for data analysis, which allows to approximate a high-dimensional set of observations (containing probably correlated variables) with a reduced-dimension system of linearly uncorrelated variables called principal components (Antoulas and Sorensen, 2001). These result from the determination of a basis of orthogonal principal modes that can be viewed as representative of the most probable observations.

As general framework, let $\mathcal{C} = \{\Theta^c \mid c \in \mathbb{N}, c = 1 : N_c\}$ be the available data set. Each of the N_c capsule shapes, Θ^c, corresponds to given values of the parameter vector $\boldsymbol{\theta}$, defined such that $\theta_1 = Ca$ and $\theta_2 = a/\ell$. The capsule contour information Θ that is available from the 3D fluid-structure interaction simulations is the 3D Lagrangian position vectors at the n nodes located on the membrane of each of the $N_c = 95$ capsules. Let $\boldsymbol{x_i^c} = (x_{i1}^c, x_{i2}^c, x_{i3}^c)$ be the position vector for the node $(i)_{i=1:n}$ of the c^{th} capsule shape contained in the database. All the position vectors of the c^{th} capsule shape can then be rearranged in a single vector $\boldsymbol{x}^c = (x_{11}^c, x_{12}^c, x_{13}^c, \ldots, x_{n1}^c, x_{n2}^c, x_{n3}^c)$ of dimension

$N_x = 3n$. The proper orthogonal decomposition consists in finding an orthogonal basis of functions $(\boldsymbol{\psi}_m)_{m=1:M}$ (i. e. the principal components) that allows to approximate the position vector $\boldsymbol{x}^c$ by

$$\boldsymbol{x}^c \simeq \sum_{m=1}^{M} \alpha_m^c \boldsymbol{\psi}_m, \tag{1}$$

where M is the dimension of the POD basis (i. e. the number of modes) and α_m^c is the coefficient associated with the m^{th} principal component. Eq. (1) is written in the case of centered data (i.e. when the mean value of the vectors $\boldsymbol{x}^c$ is equal to the null vector $\mathbf{0}$). In the case of a non-zero mean value, it is otherwise classical to center the data by subtracting the mean value to each data point, before applying the POD decomposition.

One method to compute the orthogonal basis $(\boldsymbol{\psi}_m)_{m=1:M}$ is to create the data matrix $\boldsymbol{X}$ of dimension $N_x \times N_c$ with the node position vectors $(\boldsymbol{x}^c)_{c=1:N_c}$ arranged in columns, i. e. $\boldsymbol{X} = \left[\boldsymbol{x}^1\ \boldsymbol{x}^2\ \cdots\ \boldsymbol{x}^{N_c}\right]$, and to apply the Singular Value Decomposition (SVD) to it. The matrix $\boldsymbol{X}$ is then decomposed into

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}, \tag{2}$$

where the superscript $^{\mathrm{T}}$ stands for matrix transposition, $\boldsymbol{\Sigma}$ is an $N_x \times N_c$ rectangular diagonal matrix composed of the singular values of $\boldsymbol{X}$, and where $\boldsymbol{U}$ and $\boldsymbol{V}$ are orthogonal square matrices of orders N_x and N_c, respectively, which represent the left and right singular vectors of $\boldsymbol{X}$. Note that the singular values are arranged in decreasing order of variance (the accounted variability in the data), so that the first column of $\boldsymbol{U}$ is the mode associated with the highest coefficient. Indeed, the columns of the matrix $\boldsymbol{U}$ correspond to the principal components $\boldsymbol{\psi}_m$.

The advantage of building the orthogonal basis is that the least relevant information present in the data $(\boldsymbol{x}^c)_{c=1:N_c}$ can be removed, by only keeping the M first modes that are sufficient to approximate the position vectors with a good precision. By removing the high-order modes having very small or null coefficients, one creates a reduced-order model constituted of the first few principal components. The low-rank approximation of $\boldsymbol{X}$ (Eq. 1) is thus obtained by truncating $\boldsymbol{U}$. The corresponding α-coefficients are obtained by projecting the capsule position vectors $\boldsymbol{x}^c$ onto the new orthogonal basis of principal

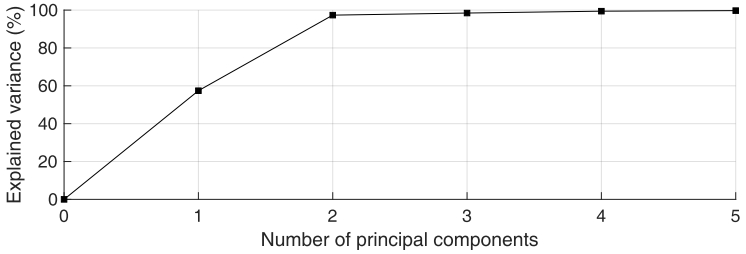


Figure 7: Cumulative explained variance of the first principal components, when POD is applied to the nodal position coordinates.

components:

$$\alpha_m^c = \boldsymbol{x}^c \cdot \boldsymbol{\psi}_m. \tag{3}$$

3.2. Validation of the POD method

In order to validate the application of Proper Orthogonal Decomposition to the deformed capsule shapes, the first point to look at is M_{max}, the number of principal components $\boldsymbol{\psi}$ having a non-zero variance. We find that, when the nodal position coordinates are used as design variables, the application of POD to the data set $\mathcal{C}$ returns $M_{max} = 94$. This is consistent with the fact that the number of capsule shapes is $N_c = 95$.

One also sees that the capsule shape, that was originally represented as 2 562 points in the three-dimensional coordinate system, and thus by $N_x = 3n = 7\,695$ design variables, can now be interpreted as a single point in an 94-dimensional space. This indicates that, by keeping the most relevant principal components only, the POD technique indeed allows to create a reduced basis and therefore reduce the dimensionality of the problem.

The second point to consider is the evolution of the cumulative explained variance with the number of principal components (or modes) M used in Eq. 1, in order to measure the relevance of the first singular values of $\boldsymbol{X}$. Fig. 7 shows the cumulative explained variance, expressed as a percentage, for the first five principal components. It shows that only the first two principal components majoritarily contribute to the variance, as they together account for about 97.3% of the total variance of the data set. This is consistent with the physics of the problem, as we know that the steady-state deformation of a capsule in a tube only depends on two independent variables: the capillary number Ca and the confinement ratio a/ℓ. The POD thus retrieves the intrinsic dimension of the problem.

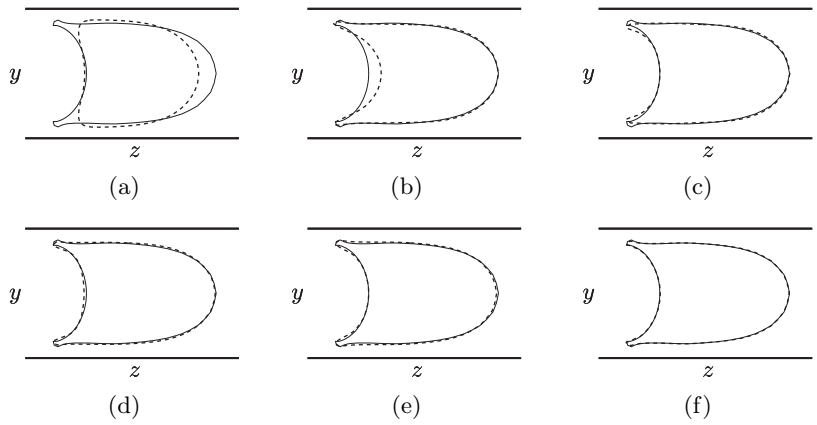


Figure 8: Contours in the $x = 0$ plane of the reconstructed capsules (dashed bold lines) superimposed to the original capsule provided by the full FSI simulation (continuous lines). The relative error number ε_{rel} decreases with the number of included principal components. (a) 1 principal component: $\varepsilon_{rel} = 8.8\%$, (b) 2 principal components: $\varepsilon_{rel} = 2.2\%$, (c) 3 principal components: $\varepsilon_{rel} = 2.1\%$, (d) 4 principal components: $\varepsilon_{rel} = 1.3\%$, (e) 5 principal components: $\varepsilon_{rel} = 1.0\%$, (f) 10 principal components: $\varepsilon_{rel} = 0.4\%$.

In order to further validate the application of POD on the 3D Lagrangian nodal position vector set and to show the implication that the cumulative explained variance (Fig. 7) has on the capsule shape, we propose to determine the error made when projecting a capsule of the data set $\mathcal{C}$ on a reduced basis and to show how it is influenced by the number of modes M included in the reduced basis. We reconstruct the capsule contours using Eq. 1 for increasing values of $(M)_{M=1:M_{max}}$ and compare the contours with the corresponding capsule deformed shape provided by the full FSI numerical simulation. Fig. 8 shows an example of comparison in the case $Ca = 0.12$ and $a/\ell = 0.95$, when the reconstructed contours are calculated including up to the first 10 modes. Qualitatively, one can see that, a very small number of modes is sufficient to guarantee a high-fidelity prediction of the capsule deformation. In such a case of very large capsule deformation, 3 modes are sufficient to reconstruct the capsule deformed shape with good accuracy and a quasi-perfect fit is reached when 10 modes are used.

To conduct a quantitative comparison and estimate the error made in predicting the capsule contour, we finally resort to the Hausdorff distances, which are object matching algorithms that evaluate the similarity between two contours composed of a set of points. We propose, in particular, to use the modified Hausdorff distance (Dubuisson and Jain, 1994), which is the average of

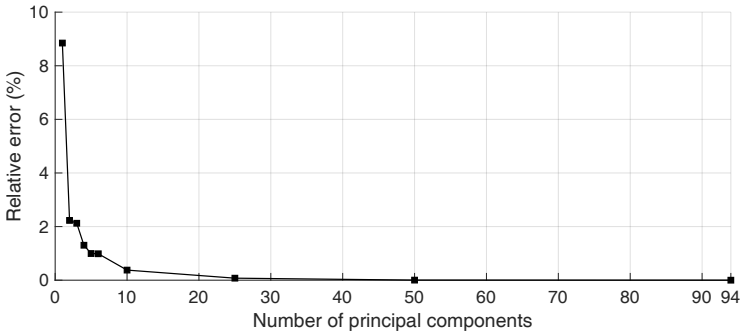


Figure 9: Effect of the number of selected principal components on the error of the reconstructed capsule. The relative error is computed for 1, 2, 3, 4, 5, 6, 10, 25, 50 and 94 principal components, represented by the dots.

the projected distances between the corresponding points of the two profiles, rather than the 'original' Hausdorff distance (Huttenlocher et al., 1993), which is the maximum value of all the projected distances. The modified Hausdorff distance is indeed more adequate to estimate the overall accuracy of the POD-reconstructed capsule profile. We shall call as relative error ε_{rel} the value of the modified Hausdorff distance normalized by the capsule initial radius a, and express it as a percentage. The values of ε_{rel} provided in the legend of Fig. 8 show that the accuracy increases with the number of included principal components, and that, even in the case of very large capsule deformation, the accuracy of the prediction is below 2.2% when only 2 modes are used in the POD reduced basis. Fig. 9 illustrates how the relative error ε_{rel} monotonically decreases with the number of principal components M for the same case as in Fig. 8, i. e. $Ca = 0.12$, $a/\ell = 0.95$. It confirms that five modes are sufficient to set the relative error below 1%, which is a very small number of modes.

4. Precision of POD on 3D Lagrangian data fields to predict capsule deformed shapes for parameter values not present in the database

4.1. Manifold for capsule shape predictions

Meng et al. (2016) have shown that it is improper to do direct interpolation between neighboring shapes using conventional shape morphing techniques (Duvigneau, 2006). In other words, the shape obtained by averaging two existing shapes has no reason to coincide with the shape computed for the mean value of their corresponding parameter values. One of the main reasons is that these morphing techniques cannot guarantee the *admissibility* of the predicted

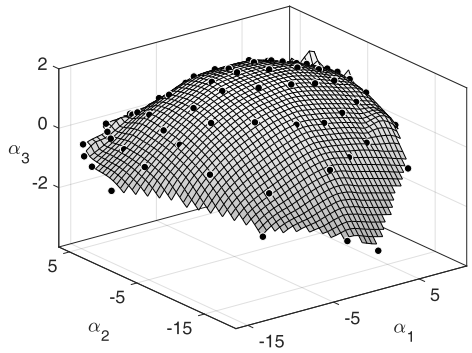


Figure 10: Representation of the coefficients $(\alpha_1, \alpha_2, \alpha_3)$ of all the capsule shapes contained in $\mathcal{C}$ (black dots). They lie on a manifold that connects all the admissible shapes. The manifold is the dome-like grey-colored 2D surface. It has a dimension of two, which is the intrinsic dimension of the problem.

shapes. To be *admissible*, the shape must indeed satisfy the constraints of the problem. In the case of the steady-state capsule shapes, the physical constraints are that a unique capsule deformed profile corresponds to each set of values of the parameters Ca and a/ℓ.

It is generally accepted that all the admissible shapes of a given problem lie on a smooth *manifold*, which is a hypersurface in the α-space (Meng et al., 2016; Raghavan et al., 2013a,b). The manifold is thus generated by projecting the database of shapes onto the POD basis: the capsule shape, originally represented as 2 562 nodes in the three-dimensional coordinate system, becomes a single point in the 94-dimensional α-space. Interestingly, the number of dimensions of the manifold is equal to the *intrinsic dimension* of the problem. Since our problem is governed by two parameters, we know in advance that the manifold associated with our database is a 2D-surface. It is represented in Fig. 10 when POD is applied to the database of Lagrangian nodal position vectors: only the points lying on this manifold correspond to admissible capsule shapes. We now propose to use the manifold to determine the admissible capsule deformed shape that corresponds to a set of parameter values that are not present in the database. This allows to verify whether we have the capability of predicting capsule deformations using POD.

4.2. Manifold-based methods of prediction

We use *manifold walking* schemes (Raghavan et al., 2013a) to predict capsule shapes for parameter values $\boldsymbol{\theta'}$ not included in the database. The technique consists in remaining on the manifold (Fig. 10) when inferring a shape from a

1	Linear piecewise interpolation.
2	Cubic spline interpolation.
3	1^{st}-order polynomial regression.
4	2^{nd}-order polynomial regression.
5	3^{rd}-order polynomial regression.
6	4^{th}-order polynomial regression.
7	5^{th}-order polynomial regression.
8	2^{nd}-order diffuse approximation. 12-point adaptive region, $A_1/A_2 = 0.4$.

Table 1: Summary of the manifold-based prediction approaches presently considered.

set of known admissible shapes and finding the α'-coefficients that best approximate it. We will test different manifold-based prediction approaches to find the α'-coefficients: surface interpolation (piecewise linear and bicubic spline interpolation), polynomial regression (of first to fifth order), and diffuse approximation. The objective is to find the one that will be the most adequate and the most precise to predict steady-state capsule shapes. All the mathematical details on the manifold walking schemes may be found in the Appendix.

4.3. Evaluation of the accuracy of the manifold-based predictions

In order to evaluate the capability of prediction of the manifold-based techniques and measure the precision of the predicted profiles, we propose to sequentially remove one data set from $\mathcal{P}$ and determine the capsule shape predicted for the parameter values corresponding to it (they become the components of $\boldsymbol{\theta}'$). The relative error ε_{rel} is calculated by comparing the predicted contour to the removed capsule shape (originally computed with the full FSI numerical model). This procedure is repeated for all the points of the database and for the different manifold-based techniques using the conditions indicated in Table 1. The results are presented as heat maps, which allow to visualize how the relative error evolves with the location in the parameter space and thus provide a complete overview of the accuracy of each technique. All the principal components provided by the POD are considered in the prediction of the shapes.

Representative heat maps of the relative error ε_{rel} are shown in Fig. 11 for 4 of the manifold-based prediction approaches: linear piecewise and cubic spline interpolations, 5^{th}-order polynomial regression, and 2^{nd}-order diffuse approximation. The darker the color tone, the greater the error. But one must stress that the error never exceeds the value of 2% in all cases: it directly proves that all these manifold-walking techniques are capable to predict a capsule shape

Θ' for parameter values not in $\mathcal{P}$ and that the accuracy of all these prediction techniques is very high.

Let us analyze more precisely how the prediction accuracy evolves within the parametric space. From a general and qualitative point of view, the heat maps of all the prediction approaches share common characteristics: (i) the predictions are the most precise in the central region of the parametric space; (ii) slightly less accurate estimations are obtained in the low capillary number region; (iii) predictions are more difficult in the region adjacent to the diagonal border. We propose to focus on the case of the cubic spline interpolation to further detail these 3 main comments. Fig. 11b clearly shows that the prediction error is the lowest in the central region of $\mathcal{P}$, where ε_{rel} is below 0.15% (even below 0.05% at some points), which means that the estimated and reference capsules are virtually indistinguishable from each other. This is illustrated in Fig. 12a, in which the 2 profiles are represented for the parameter set $Ca = 0.08$, $a/\ell = 0.85$. Capsules with low values of Ca are also well predicted, although ε_{rel} increases slightly, ranging from 0.1% to 0.5%. The slightly higher error value is due to the fact that, in that region, the concavity of the rear part of the capsules may change from one value to another, making predictions complex. An example is shown in Fig. 12b, in which the predicted profile has a small concavity whereas the actual data still has a convex rear shape. The largest inaccuracies are found in certain regions adjacent to the diagonal, where ε_{rel} just exceeds 1% (and even 1.5% at some points). This is linked to the fact that, close to the diagonal, fewer nearby points are available. The density in data points within the database has been adapted based on the observation of the error distribution. If the database has to cover the entire range of desired parameter values, the data point density also has to be increased in the regions where the capsule shape changes more abruptly with the parameters, as it is the case for the regions of low Ca and close to the dashed line. It is remarkable that, despite these border effects, the error never exceeds 2% and Fig. 12c shows that, even in these cases, the two profiles nicely fit with one another. One can thus conclude, that in spite of slight local imprecisions, the prediction of capsule deformation remains excellent within the entire parameter space.

The mean value of the relative error $\bar{\varepsilon}_{rel}$ over the entire parametric space has been computed in each case of Table 1, in order to have a quantitative measure to compare the different prediction approaches. It is provided, together with the respective standard deviations, in Fig. 13, which shows that 3 of the 4 techniques depicted in Fig. 11 sensibly have an equivalent high precision. Indeed, the mean

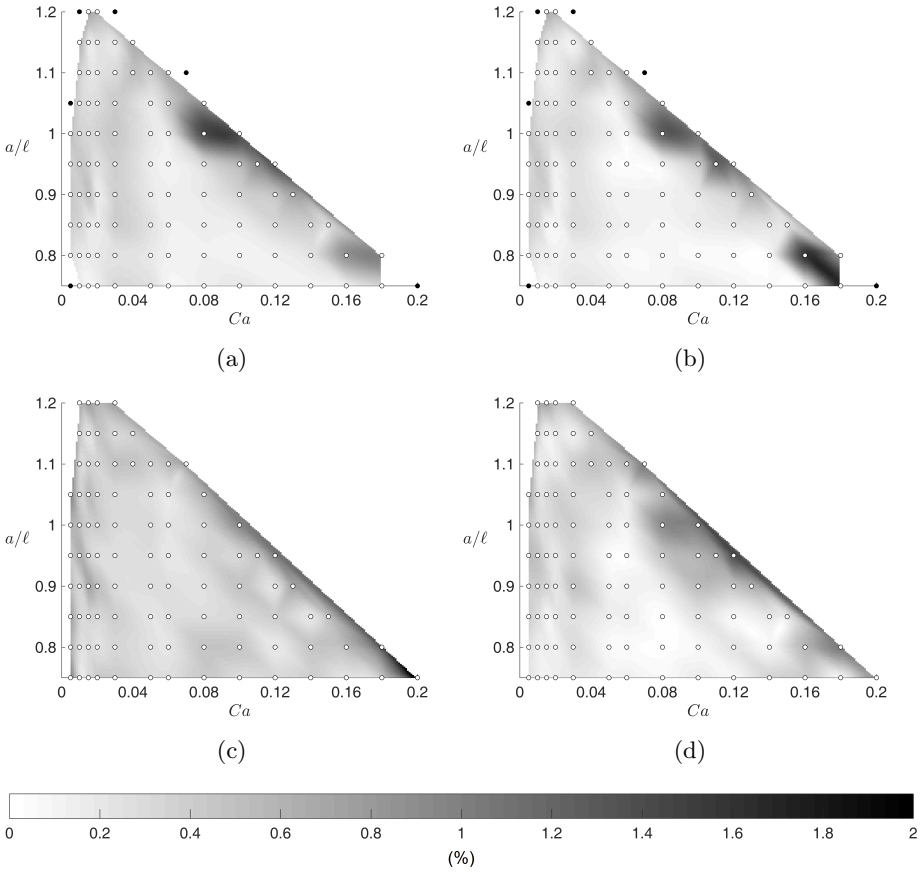


Figure 11: Heat maps of the relative error ε_{rel} of the capsule shape prediction, obtained when the nodal coordinates are used as design variables. The maps are derived by sequentially excluding a parameter set from the database $\mathcal{P}$. The error is calculated by comparing the predicted contour with the removed capsule shape (which was computed with the full FSI numerical model). (a) Piecewise linear interpolation, (b) cubic spline interpolation, (c) 5^{th}-order polynomial regression, (d) 2^{nd}-order diffuse approximation. The black dots correspond to the points, where capsule shape predictions could not be made by the method.

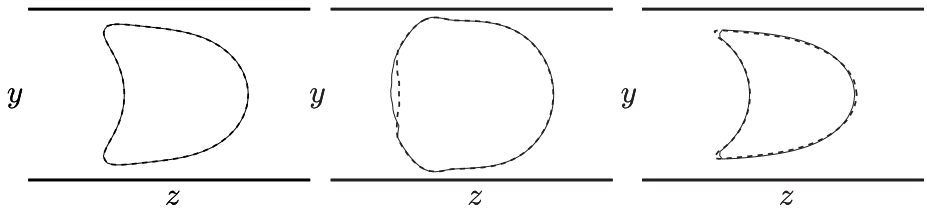


Figure 12: Capsule contours in the $x = 0$ plane predicted using the cubic spline interpolation. The dashed bold lines represent the predicted capsule contours, while the continuous lines represent the reference capsule shapes. (a) $Ca = 0.08, a/\ell = 0.85, \varepsilon_{rel} = 0.09\%$, (b) $Ca = 0.01, a/\ell = 0.95, \varepsilon_{rel} = 0.43\%$, (c) $Ca = 0.18, a/\ell = 0.75, \varepsilon_{rel} = 1.69\%$.

relative error is $\bar{\varepsilon}_{rel} = 0.32\%$ for the cubic spline interpolation, 0.33% for the 2^{nd}-order diffuse approximation is used, and 0.36% for the linear interpolation. The mean errors are, however, slightly higher for all the polynomial regression techniques regardless of the order of the regression. Fig. 13 shows that the accuracy increases with the order, but that a 5^{th}-order scheme is necessary to get close ($\bar{\varepsilon}_{rel} = 0.456\%$) to the precision of the other 3 methods.

We shall now provide additional remarks on the performance of each of the prediction techniques.

Interpolation approaches. The performance of the linear and cubic spline interpolation approaches are broadly analogous, which appears when one compares Fig. 11a and Fig. 11b. Examples of comparison of predicted vs computed contours may be found in Fig. 14a for linear interpolation and Fig. 12 for cubic interpolation: excellent fits are found in all cases. On average, the cubic spline interpolation presents slightly better results than the linear interpolation, but it is slightly less accurate when predicting capsule shapes close to the diagonal border at high values of Ca. It is noteworthy to make the general remark that interpolation methods cannot compute values outside the domain, as it has no *extrapolation* capabilities. No capsule shapes can thus be predicted close to the vertices of the parametric domain. This situation occurs for six observations in the data set, which are displayed as black dots on the heat maps.

Polynomial regression approaches. We have seen that the precision of the polynomial regression estimations greatly improves as the polynomial order increases (Fig. 13). It indicates that the first few orders of polynomial regression are not optimal for manifold-walking predictions (e. g. $\bar{\varepsilon}_{rel} = 2.7\%$ for the 1^{st}-order). Still, the results obtained with the 5^{th}-order polynomial regression (Fig. 11c)

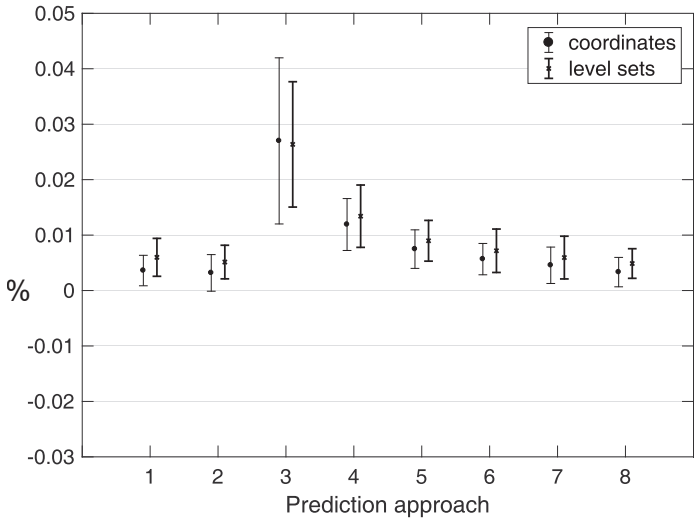


Figure 13: Mean and standard deviation of the 95 values of ε_{rel} for the prediction approaches listed in Table 1. Thin lines represent the results when POD is applied to nodal position coordinates, and bold lines when it is applied to a level set field.

present nearly uniform values of ε_{rel} in the central region, ranging between 0.2% and 0.5%. Higher values can be found along the diagonal boundary of the parametric domain (with a peak at 2.8% for $Ca = 0.2$), and also in the area of low Ca (with some values over 0.9%). An example of poorer capsule shape prediction is provided in Fig. 14b: it illustrates one common issue of high-order polynomial regressions, which is the risk of oscillations.

Diffuse approximation approaches. We have found that the diffuse approximation approaches provide excellent results as indicated by the heat map (Fig. 11d) and the example of capsule contour comparison given in Fig. 14c. The precision of the method is, however, conditioned by the shape of the neighborhood. It must be well chosen, so that the neighborhood is adapted to the manifold shape and contains a sufficient number data set points to ensure precision. Diffuse approximation performs, in general, better with 2^{nd}-order interpolation functions than with 1^{st}-order, but the difference is rather marginal. Unlike interpolation approaches, diffuse approximation allows to predict capsule shapes at the borders of the domain. Fig. 11d indicates that the predictions at the locations of the black dots (Fig. 11a) are actually very pertinent.

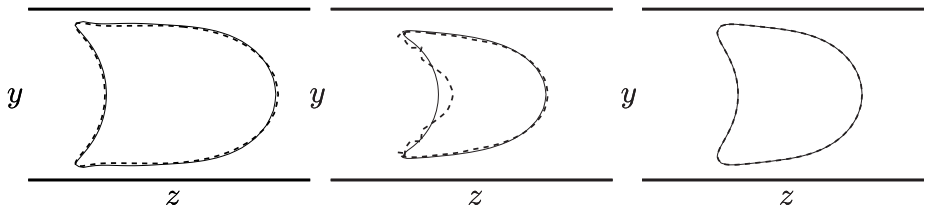


Figure 14: Capsule contours in the $x = 0$ plane. Dashed bold lines represent the predicted capsule contours, while the continuous lines represent the reference capsule shapes. (a) Linear interpolation: $Ca = 0.08, a/\ell = 1, \varepsilon_{rel} = 1.42\%$. (b) 5^{th}-order polynomial regression: $Ca = 0.2, a/\ell = 0.75, \varepsilon_{rel} = 2.81\%$. (c) 2^{nd}-order diffuse approximation: $Ca = 0.08, a/\ell = 0.85, \varepsilon_{rel} = 0.015\%$.

5. POD of 3D Eulerian data fields

Having established the capability of POD to predict capsule deformation when applied to Lagrangian data sets, we now investigate if it can equally be applied to Eulerian fields. In Lagrangian descriptions, fixed topologies of the shapes are assumed. The design variables can be any explicit parametrization of the surface boundaries, which can range from vectors of positions, lengths, angles or radii of the objects (Meng et al., 2016), to boundary splines, basis shapes or free-form deformations (Samareh, 1999; Sederberg and Parry, 1986). But the capsule datasets are not necessarily Lagrangian: they can also be in the form of a Eulerian field, whether they are obtained numerically (using purely Eulerian or meshless numerical schemes) or experimentally (the images being coded in greyscale). In such Eulerian problems, design variables can still be determined, but they require the use of implicit strategies. Some examples of implicit parametrizations include the use of pixelized or voxelized representations (Raghavan et al., 2010), phase fields (Bourdin and Chambolle, 2006), or level sets (Burger, 2003; Burger and Osher, 2005). The issue is that these methods inherently add errors onto the datasets. In this section, we thus investigate whether it is possible to directly apply POD on the Eulerian dataset.

Prior to applying POD, we first generate a scalar field from the Lagrangian datasets of the nodal position vectors obtained by the boundary element-finite element numerical method. We calculate the signed distance of all the points in space to the capsule membrane. After having developed a technique to reconstruct the 3D capsule contour from the level sets, we will compare the results to their Lagrangian counterparts and estimate the precision of POD when it is applied to 3D Eulerian data fields.

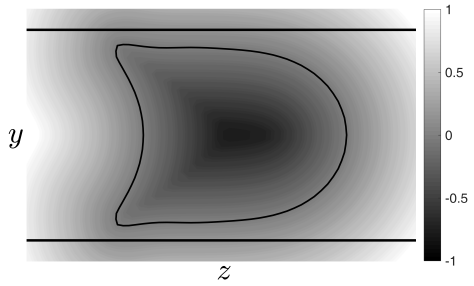


Figure 15: Representation in the $x = 0$ plane of the Eulerian field, generated by calculating the signed distance of the points of the domain to the contour of a capsule ($Ca = 0.06, a/\ell = 1$). The level set was obtained using a grid containing $p = 50\,176$ points.

5.1. Generation of level sets

We resort to level sets to emulate a Eulerian data set $\mathcal{C}_E$ from the Lagrangian database $\mathcal{C}$. To do so, we first create a three-dimensional regular grid capable of englobing all the capsule deformed contours for the parameters in $\mathcal{P}$. We then generate a three-dimensional scalar field for each of the capsule deformed shapes, by measuring the shortest Euclidean distances between the grid points and the capsule surface centered at the middle of the grid. The distances are signed: they are positive for the grid points located outside the shape and negative for the grid points located inside. The scalar field is stored in the form of a high-dimensional vector.

The level set grid has to be identical for all the shapes in $\mathcal{C}$ so that the new matrix $\boldsymbol{X}$ is correctly assembled. We choose a grid box that extends from -1.8ℓ to 1.9ℓ in the z-direction, and from -1.2ℓ to 1.2ℓ in the x- and y-directions. This size is enough to contain all the shapes in $\mathcal{C}$ with a margin on all sides (see Fig 15). The grid points are uniformly distributed across the box. Three grid point densities are tested: 8, 10 and 13 points per unit length ℓ. They, respectively, correspond to level set boxes of $30 \times 20 \times 20$ points (i.e. $p = 12\,000$ points), $38 \times 25 \times 25$ points (i.e. $p = 23\,750$ points), and $49 \times 32 \times 32$ points (i.e. $p = 50\,176$ points). One must note that the time required to compute the level set of a capsule largely increases with the number of grid points.

5.2. Application of POD to $\mathcal{C}_E$ and reconstruction of the capsule contours

The Eulerian data set $\mathcal{C}_E$ is composed of the N_c level sets computed for each of the capsule deformed shape of $\mathcal{C}$. In the Eulerian case, the vector $\boldsymbol{x}^c$ of the c^{th} capsule is $\boldsymbol{x}^c = \left(d_1^c, d_2^c, \ldots, d_p^c\right)$, where d_i^c is the shortest signed distance

from the i^{th} grid point to the surface of the c^{th} capsule shape, and p is the total number of points in the level set.

The POD technique is then applied to $\mathcal{C}_E$ following exactly the same method as described in section 3.1. It, similarly, returns $M_{max} = 94$ components with non-zero variance. But the difference is that the computed POD reduced bases are now bases of level sets, and not bases of shapes. In order to compare the results with the Lagrangian results, there is thus the need to reconstruct the capsule shapes from the level sets. One must note that applying POD to the Eulerian data set is computationally much costlier than in the Lagrangian case. Not only is the data set much bigger in size ($p \gg 3n$), but also an additional step is needed to reconstruct the capsule shape.

The capsule shapes are reconstructed from the level sets using the Marching Cubes algorithm (Lorensen and Cline, 1987), with which we generate a polygonal mesh of the isosurface $d = 0$, corresponding to the capsule surface. The size of the generated mesh is directly related to the density in grid points. The reconstructed capsule meshes are made of 969 nodes and 1 820 elements for the grid with $p = 12\,000$ points, of 1 407 nodes and 2 656 elements for the grid with $p = 23\,750$ points, and of 2 429 nodes and 4 628 elements for the grid with $p = 50\,176$ points. We have set the latter as the default grid, as it provides a mesh that has a similar amount of nodes and elements to the original capsule mesh, which has 2 562 nodes and 5 120 elements. This will guarantee the validity of the shape comparison.

Fig. 16 shows the cumulative explained variance for the first five components. The first two components account for 97.97% of the total variance of the data set, which, like in the POD on Lagrangian data, is consistent with the fact that the manifold is two-dimensional. It is interesting to note that the total variance of the first two modes is slightly higher than in the Lagrangian case. POD on Eulerian data appears as efficient, if not more, as POD on Lagrangian data.

5.3. Comparison of POD on Eulerian and Lagrangian data sets

The capsule shapes predicted by POD of level set data are compared, one by one, to the removed capsule shape, which was computed with the full FSI numerical model. As in the previous sections, we used all the principal components provided by POD to predict the shapes. The heat maps of the relative error ε_{rel} are shown in Fig. 17 for the same 4 manifold-based prediction approaches as in Fig. 11: linear piecewise and cubic spline interpolations, 5^{th}-order polynomial regression, and 2^{nd}-order diffuse approximation. The same color scales are used

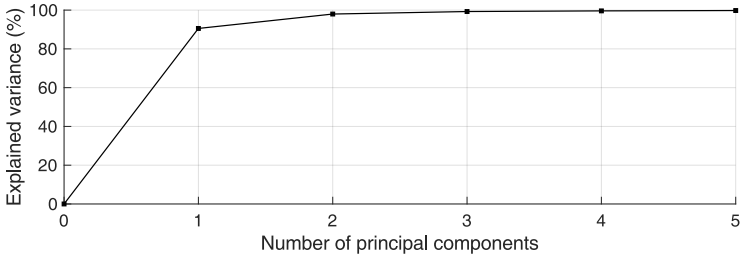


Figure 16: Cumulative explained variance of the first principal components, when POD is applied to the Eulerian level sets.

as previously to facilitate the comparison. The mean values of the relative error are provided in Fig. 13 for the different approaches listed in Table 1.

In general, the maps of the relative error obtained with Eulerian data sets (Fig. 17) are very similar to those obtained with Lagrangian data sets (Fig. 11). The predictions are the most accurate in the central region of the parametric domain, higher values of error being obtained for low values of Ca or parameter values close to the diagonal. The fact that the grey tones are globally more intense indicates that the predictions are generally less precise than in the Lagrangian case. But the error still hardly exceeds the value of 2%, which validates the concept of manifold-walking using Eulerian data to predict a capsule shape. The mean values of the relative error confirm that the results are slightly less precise than in the Lagrangian case. One can, for instance, note that $\bar{\varepsilon}_{rel} = 0.49\%$ for the 2^{nd}-order diffuse approximation, and 0.51% for the cubic spline interpolation, which indicates that the difference is not major.

6. Discussion on the influence of the number of principal components

In the previous sections, we have always used all the principal components provided by the POD to predict capsule shapes. We now propose to analyze the effects that the number of principal components has on the precision of the prediction and on the computational time. We will test different sum truncations, varying the value of M in Eq. 1.

We have computed the relative error, when removing the capsule shapes of the database one by one, and reconstructing them using 1, 2, 3, 4, 5, 6, 10, 25, 50 or all the 93 principal components. The evolution of the mean relative error with the number of modes is shown in Fig. 18 for the cubic spline interpolation, which is representative of the results obtained in all the cases. We retrieve the

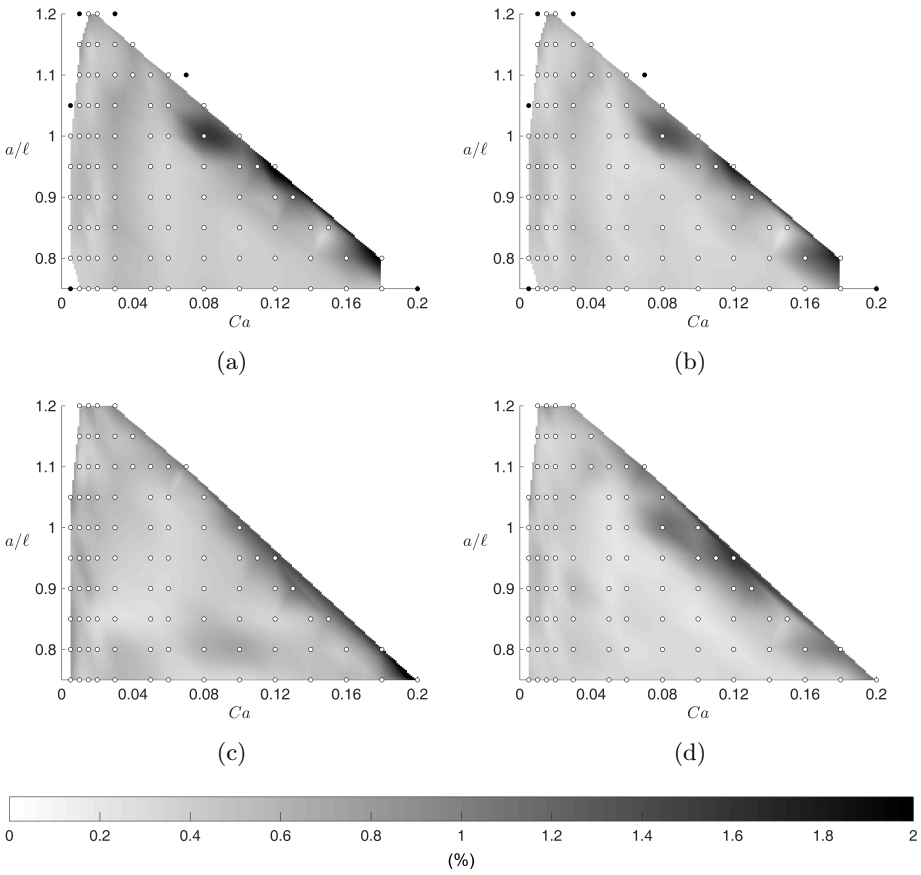


Figure 17: Heat maps of the relative error ε_{rel} of the capsule shape prediction, obtained when the level sets are used as design variables. The maps are derived by sequentially excluding a parameter set from the database $\mathcal{P}$. The error is calculated by comparing the contour, reconstructed from the predicted level set, to the removed capsule shape (which was computed with the full FSI numerical model). (a) Piecewise linear interpolation, (b) cubic spline interpolation, (c) 5^{th}-order polynomial regression, (d) 2^{nd}-order diffuse approximation. The black dots correspond to the points, where capsule shape predictions could not be made by the method.

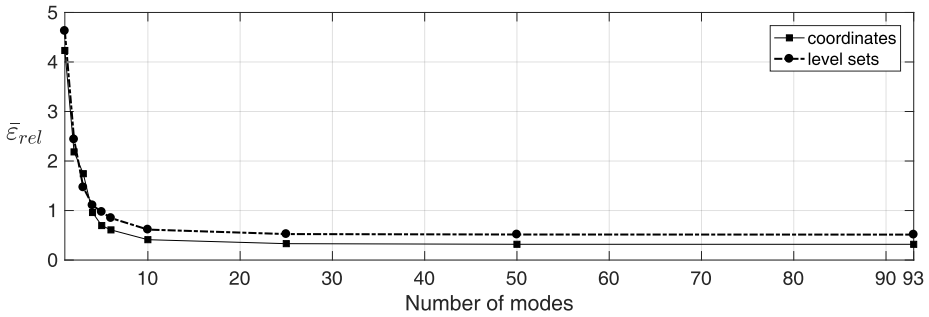


Figure 18: Evolution of $\bar{\varepsilon}_{rel}$ with the number of principal components in the case of cubic spline interpolation.

fact that the predictions errors for the level sets are slightly higher than for the nodal coordinate vectors. All the prediction techniques have values of $\bar{\varepsilon}_{rel}$ above 4%, when only one principal component is used in the prediction. The mean error decreases drastically with the number of principal components, until a plateau is reached from 10 principal components onwards. These results show that a very small number of modes is sufficient to provide very accurate results. The errors are below 1% when only 4 or 5 modes are used and, above 10, adding modes hardly increases the precision. This is due to the fact that almost all the information is contained in the very first few modes.

In order to determine the corresponding computational times needed to estimate the capsule shapes, we have run each of the computations 100 times, varying the number of principal components from 1 to 93 systematically. Naturally, the computation time depends to a large extent on how the prediction algorithms are programmed and on the characteristics of the computer on which they are run. The present algorithms have been written in Matlab, using the built-in function `fit` for interpolation and polynomial regression, and ad hoc functions for the diffuse approximation. Algorithms have been run on a desktop computer with an Intel Core i7 processor at 3.40 GHz and 16 GB of RAM memory.

Examples of the average values of the computational times are shown in Fig. 19, along with the standard deviation, for the cubic spline interpolation and 2^{nd}-order diffuse approximation. The time required by the prediction algorithms to estimate a capsule shape is directly proportional to the number of principal components chosen to generate the estimation. For instance, using cubic spline interpolation, the capsule shapes are predicted in 2.2 seconds using all the 93 principal components, and in approximately a third of that time us-

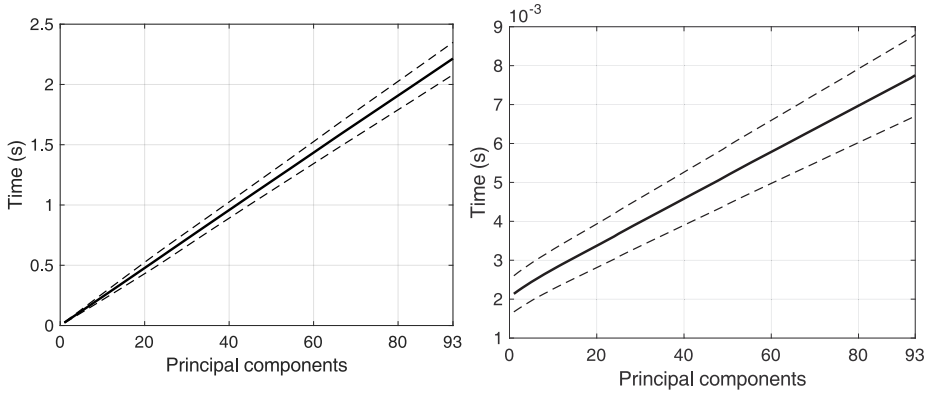


Figure 19: Average computational time required to predict a shape for a given number of principal components: (a) cubic spline interpolation, (b) 2^{nd}-order diffuse approximation. The dashed lines correspond to the average value $\pm$ the standard deviation.

ing 31 components. As a comparison, the corresponding maximum time using linear interpolation is 2.0 seconds. All the algorithms based on the generic Matlab built-in function `fit` take longer than those using ad hoc functions. When using the complete set of principal components, the predictions are obtained in 2–2.5 seconds with the built-in function (Fig. 19a), vs only 6.5–9 ms for the ad hoc function (Fig. 19b). As a comparison, the full FSI simulations take about 48 hours to run on the same computer. The POD technique thus represents a gain in time of 8 639 900% for the 2-second long estimates, and 2 159 999 900% for the 8-millisecond estimates.

7. Conclusion

In this work, we have implemented, for the first time, POD-based dimensionality reduction techniques to predict the deformation that a capsule has at steady state when flowing through a straight microfluidic channel. They allow to predict the capsule shape for any arbitrary value of the two input parameters of the problem: the capillary number Ca and the capsule-to-channel size ratio a/ℓ. We have shown that manifold-walking techniques provide very accurate predictions of the capsule shape, especially when interpolation and 2^{nd}-order diffuse approximation approaches are used to 'walk' on the hypersurface. Furthermore, one of the most outstanding features of these prediction techniques is the very small computational time. Results are obtained in just a couple of seconds, which sharply contrasts with the days required by the full Fluid-Structure Interaction numerical method.

The accuracy of the predictions is altogether excellent with mean values much below 1%. It is, however, not uniform for all the values of Ca and a/ℓ. Very high levels of precision are reached in the center region of the parametric domain. Near the domain borders, the predictions lose a little accuracy, because of the absence of information outside the domain. This occurs near the diagonal described in §2.3, beyond which no steady-state capsule shape exists for the neo-Hookean constitutive law. Strain-hardening laws, such as the law by Skalak et al. (1973), do not have this inherent limitation, since a steady-state capsule conformation exists for any value of the input parameters. The same bordering phenomenon occurs in the low capillary number region ($Ca < 0.04$), which adds the fact that the concavity of the rear part of the capsule may change suddenly with a small increment in Ca. It is, indeed, in this range of capillary numbers that the capsule transits from a slug-shape to a parachute-shape. This increases the difficulty of shape prediction, but this inherent weakness can be mitigated by increasing the density of pre-calculated forms in this region of the database.

We have also verified that POD-based prediction approaches can be applied to Lagrangian data (e. g. capsule nodal position vectors computed from numerical methods), as well as to Eulerian scalar fields (e. g. greyscale imaging data obtained from experiments). To emulate a Eulerian framework from our database, we have transformed the capsule shapes into scalar fields using level sets. The prediction results obtained on the level sets are approximately as accurate as the Lagrangian results. The slight difference in accuracy may be due to the fact that we had to go through a shape reconstruction step in order to estimate the accuracy. One can foresee that extremely high levels of accuracy would be kept (possibly higher than for Lagrangian data as suggested by Fig. 16), if the predicted level set fields are directly post-processed without conversion step.

The present method could be generalized to microcapsules flowing in channels that lead to non-steady-state solutions (e.g. microchannel with a sudden change in diameter like in Gires et al. (2016)). Besides obtaining the shape of the capsules at steady state, their shape can indeed also be computed at any particular value of the non-dimensional time Ut/l. The deformation can then be obtained for any value of the parameters Ca and a/l like in the present study. By using this approach and calculating the capsule shape at various instances, , one can see that it is possible to determine whether the obtained shape corresponds to a transient state or the steady state just by comparing the resulting shapes.

The advantage of POD over methods based on spherical harmonics or Fourier transforms is that it uses a base of dedicated functions rather than a universal function base. With POD, we indeed do the decomposition on a base that we build *a posteriori*, while the other methods use a decomposition on an *a priori* base. This results in much shorter series development and thus a reduced number of significant coefficients/modes (less than 10). The price to pay is the need to have existing results of the full simulation.

Once determined, the POD decomposition could then also be used to reduce the fluid and/or solid solvers, which will greatly reduce the computational time as we only need to account for less than 10 modes. Simulations in reduced bases was not the objective of the study, but one must stress that they are one application of the POD decomposition. Their prediction capability will be good in the case of parabolic problems (i.e. purely diffusive phenomena) but quite poor in the case of hyperbolic problems (i.e. transport phenomena).

In conclusion, the successful application of model-order reduction techniques to the prediction of capsule deformation paves the way for a new generation of faster and more efficient numerical methods in the field of microfluidics. We have seen that such techniques can be used for many applications, such as identification of mechanical properties, as they would drastically reduce the cost of the inverse analysis procedures.

Appendix

Different approaches have been tested to walk on the manifold when predicting the α'-coefficients: surface interpolation, polynomial regression, and diffuse approximation.

Surface interpolation. Interpolation is one possible method to build a continuous surface $S_m(\boldsymbol{\theta})$ for each principal component m: it is obtained such that, $\forall c,\ S_m(\boldsymbol{\theta^c}) = \alpha_m^c$. The coefficients α'_m, corresponding to any new parameter set $\boldsymbol{\theta'}$, are then given by: $\alpha'_m = S_m(\boldsymbol{\theta'})$. We have considered both piecewise bilinear interpolation and bicubic spline interpolation in order to evaluate the influence of the order of the interpolation technique.

For the piecewise linear interpolation, the surfaces S_m are approximated with a series of piecewise-defined planes. They thus have discontinuous first derivatives. The sub-domain of each plane is determined by applying the Delaunay triangulation algorithm to the set $\mathcal{P}$ of $\boldsymbol{\theta}$ points in the parametric space (Fig. 5).

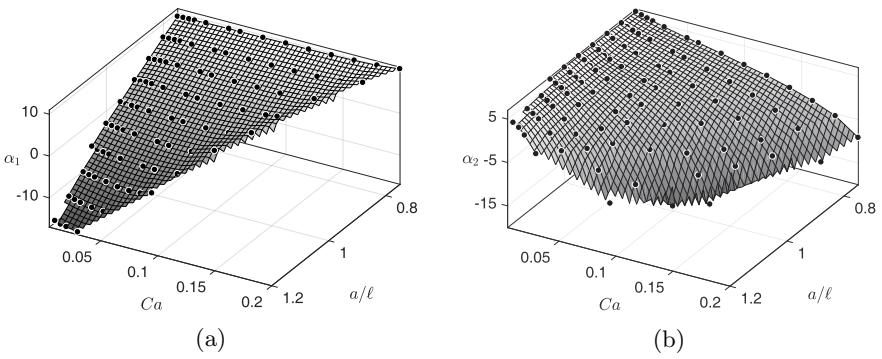


Figure 20: Surfaces S_1 (a) and S_2 (b) computed using piecewise linear interpolation. Note that the surfaces pass through all the points $\boldsymbol{\theta}$.

Let the point of interest $\boldsymbol{\theta'}$ be located inside the triangular sub-domain delimited by the points $\boldsymbol{\theta^i}, \boldsymbol{\theta^j}, \boldsymbol{\theta^k} \in \mathcal{P}$. The approximated value of α'_m is then obtained by finding the equation of the plane passing through the three points $\left(\theta_1^i, \theta_2^i, \alpha_m^i\right)$, $\left(\theta_1^j, \theta_2^j, \alpha_m^j\right)$, and $\left(\theta_1^k, \theta_2^k, \alpha_m^k\right)$ and evaluating the α_m value at (θ'_1, θ'_2). The surfaces S_1 and S_2, computed for the set $\mathcal{P}$, are shown in Fig. 20 as an illustration.

By contrast, the bicubic spline interpolation guarantees the continuity of the first derivatives and the cross-derivative of S_m: the surfaces are smoother, which is at the expense of a more complex algorithm. Within each sub-domain, the piecewise-defined surfaces are now computed using a third-order bivariate polynomial function with sixteen unknown coefficients, which are determined by particularizing the polynomial function and its derivatives with the values known at the neighboring points $\boldsymbol{\theta} \in \mathcal{P}$. More details on bicubic spline interpolation may be found in Boor (1962).

Polynomial regression. Another method to obtain continuous surfaces $S_m(\boldsymbol{\theta})$ for each principal component m is to use polynomial regression techniques, and express the α_m^c coefficients as an n^{th} degree polynomial of the variables θ_1 and θ_2. We have tested polynomial functions up to the fifth order to estimate the

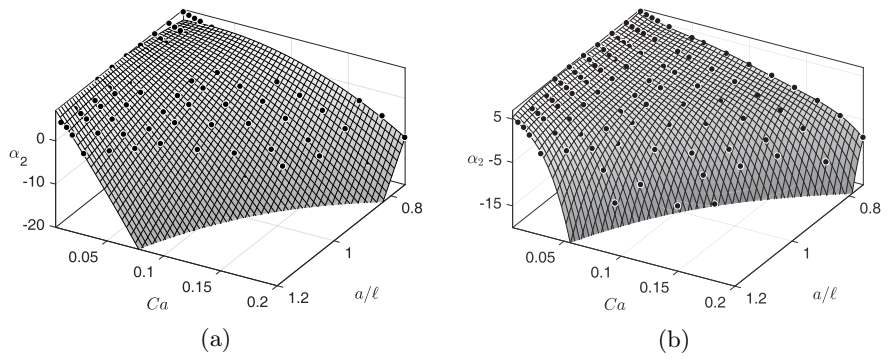


Figure 21: Examples of S_2 surfaces, computed using 2^{nd}-order (a) and 5^{th}-order (b) polynomial regression. Note that the surfaces do not pass through all the points $\boldsymbol{\theta}$. This is more visible for the 2^{nd}-order than the 5^{th}-order regression.

surfaces:

$$
\begin{aligned}
S_{1m}(\theta_1,\theta_2) &= \beta_{00} + \beta_{10}\theta_1 + \beta_{01}\theta_2,\\
S_{2m}(\theta_1,\theta_2) &= S_{1m} + \beta_{20}{\theta_1}^2 + \beta_{11}\theta_1\theta_2 + \beta_{02}{\theta_2}^2,\\
S_{3m}(\theta_1,\theta_2) &= S_{2m} + \beta_{30}{\theta_1}^3 + \beta_{21}{\theta_1}^2\theta_2 + \beta_{12}\theta_1{\theta_2}^2 + \beta_{03}{\theta_2}^3,\\
S_{4m}(\theta_1,\theta_2) &= S_{3m} + \beta_{40}{\theta_1}^4 + \beta_{31}{\theta_1}^3\theta_2 + \beta_{22}{\theta_1}^2{\theta_2}^2 + \beta_{13}\theta_1{\theta_2}^3 + \beta_{04}{\theta_2}^4,\\
S_{5m}(\theta_1,\theta_2) &= S_{4m} + \beta_{50}{\theta_1}^5 + \beta_{41}{\theta_1}^4\theta_2 + \beta_{32}{\theta_1}^3{\theta_2}^2 + \beta_{23}{\theta_1}^2{\theta_2}^3 + \beta_{14}\theta_1{\theta_2}^4 + \beta_{05}{\theta_2}^5,
\end{aligned}
$$

where β_{ij} are the polynomial coefficients, which are estimated using an ordinary least square method (Brandt, 1999). A major difference with interpolation techniques is that the condition $S_m(\boldsymbol{\theta}^c) = \alpha_m^c$ is not presently enforced: the generated surface does not necessarily pass through all α-coefficients. This is illustrated in Fig. 21, which shows the shapes of the S_2 surface computed with different orders of polynomial regression.

Diffuse approximation. The third approach, diffuse approximation (Nayroles et al., 1992), accurately maps any point $\boldsymbol{\theta'}$ in the parametric space to its corresponding point $\boldsymbol{\alpha'}$ on the manifold. To do so, the method uses a local weighted least squares fitting, that is valid in a small neighborhood centered on $\boldsymbol{\theta'}$ (Fig. 22a). Within this domain, the α'-coefficients can be locally approximated by

$$\alpha'_m(\boldsymbol{\theta'}) \approx \boldsymbol{p}(\boldsymbol{\theta'})^{\mathrm{T}}\boldsymbol{a}^m,$$

where $\boldsymbol{p}$ is a vector of independent functions, that are generally polynomials, and $\boldsymbol{a}^m$ is the vector containing the approximation coefficients (constant for each principal component m). For first-order functions,

$$\alpha'_m(\boldsymbol{\theta}') \approx a_0^m + \theta'_1\ a_1^m + \theta'_2\ a_2^m,$$

while for second-order functions,

$$\alpha'_m(\boldsymbol{\theta}') \approx a_0^m + \theta'_1\ a_1^m + \theta'_2\ a_2^m + {\theta'_1}^2\ a_3^m + \theta'_1\ \theta'_2\ a_4^m + {\theta'_2}^2\ a_5^m.$$

Given the distribution of the capsule shapes in the parametric space (Fig. 5), we have chosen to use ellipsoidal neighborhoods of major axis A_1 in the $\theta_1 = Ca$ direction and minor axis A_2 in the $\theta_2 = a/\ell$ direction (Fig. 22a). We have tested whether it was better to define the neighborhood using a criterion of size (providing values of A_1 and A_1/A_2) or using a criterion on the number of points of $\mathcal{P}$ included within the neighborhood. We have found that the best results are obtained when the neighborhoods have a variable size, keeping a constant size ratio $A_1/A_2 = 0.4$, that is adapted to contain 12 points regardless their position in $\mathcal{P}$.

The contribution made by the points $\boldsymbol{\theta}$ of the set $\mathcal{P}$ in the approximation of α'_m is weighted by their distance d to $\boldsymbol{\theta}'$. The normalized weight function

$$w(d) = \begin{cases} 2d^3 - 3d^2 + 1, & \text{if } d \leq 1 \\ 0, & \text{otherwise} \end{cases},$$

is plotted in Fig. 22b. The distance d is normalized to have a value of $d = 1$ at the boundary of the neighborhood. The neighborhood must, at least, include as many $\boldsymbol{\theta}$ points as the number of a_i^m coefficients, so that the system of equations is not over-determined.

The diffuse approximation finally uses a least squares method to determine the vector of coefficients $\boldsymbol{a}^m$, which are then given by:

$$\boldsymbol{a}^m = \left(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{P}\right)^{-1}\boldsymbol{P}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{\alpha}_m,$$

where $\boldsymbol{P}$ is a matrix consisting of the N_c column vectors $\boldsymbol{p}(\boldsymbol{\theta}^c)$, $\boldsymbol{W}$ is a diagonal matrix of order N_c with the weights and $\boldsymbol{\alpha}_m$ is a column vector containing the known values of the N_c coefficients α_m^c.

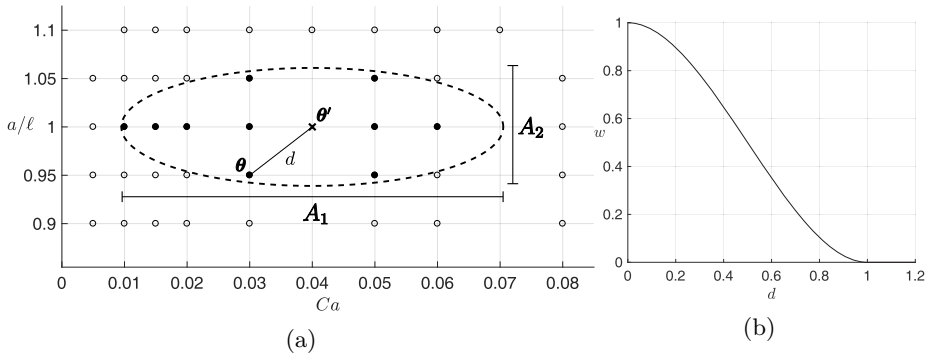


Figure 22: Elliptical neighborhood and weight function used in the diffuse approximation technique. (a) Close-up on an elliptical neighborhood with $A_1 = 0.061$ and $A_2 = 2\,A_1$ (dashed line), defined around the point $\boldsymbol{\theta'} = (Ca = 0.04, a/\ell = 1)$ in the parametric space $\mathcal{P}$. The neighborhood includes 10 capsule shapes (black dots). (b) Weight function w as a function of the distance d of the points of $\mathcal{P}$ to $\boldsymbol{\theta'}$.

Acknowledgements

This work was supported by the Labex MS2T through the program "*Investments for the future*" funded by the French Government and managed by the National Agency for Research (Reference ANR-11-IDEX-0004-02). It also received funding from the European Research Council (ERC) under the European UnionâĂŹs Horizon 2020 research and innovation programme (Grant agreement No. ERC-2017-COG - MultiphysMicroCaps).

References

Abkarian M, Faivre M, Viallat A. Swinging of red blood cells under shear flow. Phys Rev Lett 2007;98:188302.

Ammar A, Mokdad B, Chinesta F, Keunings R. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. Journal of Non-Newtonian Fluid Mechanics 2006;139(3):153–76.

Ammar A, Mokdad B, Chinesta F, Keunings R. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modelling of complex fluids: Part II: Transient simulation using space-time separated representations. Journal of Non-Newtonian Fluid Mechanics 2007;144(2):98–121.

Antoulas A, Sorensen D. Approximation of large-scale dynamical systems: an overview. Int J Appl Math Comput Sci 2001;11:1093–121.

Barthès-Biesel D. Modeling the motion of capsules in flow. Current Opinion in Colloid & Interface Science 2011;16(1):3–12.

Barthès-Biesel D. Motion and deformation of elastic capsules and vesicles in flow. Annual Review of Fluid Mechanics 2016;48:25–52.

Barthès-Biesel D, Diaz A, Dhenin E. Effect of constitutive laws for two dimensional membranes on flow-induced capsule deformation. Journal of Fluid Mechanics 2002;460:211–22.

Barthès-Biesel D, Rallison J. The time-dependent deformation of a capsule freely suspended in a linear shear flow. Journal of Fluid Mechanics 1981;113:251–67.

Boor CD. Bicubic spline interpolation. Studies in Applied Mathematics 1962;41(1-4):212–8.

Bourdin B, Chambolle A. The phase-field method in optimal design. In: IUTAM Symposium on Topological Design Optimization of Structures, Machines and Materials. Springer; 2006. p. 207–15.

Brandt S. Data Analysis: Statistical and Computational Methods for Scientists and Engineers. 3rd ed. Springer, 1999.

Burger M. A framework for the construction of level set methods for shape optimization and reconstruction. Interfaces and Free Boundaries 2003;5(3):301–29.

Burger M, Osher S. A survey on level set methods for inverse problems and optimal design. European Journal of Applied Mathematics 2005;16(2):263–301.

Chinesta F, Keunings R, Leygue A. The Proper Generalized Decomposition for Advanced Numerical Simulations: A Primer. Springer Science & Business Media, 2013.

Chu T, Salsac AV, Leclerc E, Barthès-Biesel D, Wurtz H, Edwards-Lévy F. Comparison between measurements of elasticity and free amino group content of ovalbumin microcapsule membranes: discrimination of the cross-linking degree. Journal of Colloid and Interface Science 2011;355(1):81–8.

Cole E, Cadé D, Benameur H. Challenges and opportunities in the encapsulation of liquid and semi-solid formulations into capsules for oral administration. Advanced Drug Delivery Reviews 2008;60(6):747–56.

Cottet GH, Maitre E, Milcent T. Eulerian formulation and level set models for incompressible fluid-structure interaction. ESAIM: Mathematical Modelling and Numerical Analysis 2008;42(3):471–92.

De Loubens C, Deschamps J, Georgelin M, Charrier A, Edwards-Lévy F, Leonetti M. Mechanical characterization of cross-linked serum albumin microcapsules. Soft Matter 2014;10:4561–8.

Dubuisson M, Jain A. A modified hausdorff distance for object matching. In: Pattern Recognition, 1994. Vol. 1-Conference A: Computer Vision & Image Processing., Proceedings of the 12th IAPR International Conference on. IEEE; volume 1; 1994. p. 566–8.

Dupont C, Delahaye F, Barthès-Biesel D, Salsac AV. Stable equilibrium configurations of an oblate capsule in shear flow. J Fluid Mech 2016;791:738–57.

Duvigneau R. Adaptive parameterization using free-form deformation for aerodynamic shape optimization. Ph.D. thesis; INRIA; 2006.

Eftang JL. Reduced basis methods for parametrized partial differential equations. Ph.D. thesis; Faculty of Information Technology, Mathematics and Electrical Engineering, Norwegian University of Science and Technology; 2011.

Fischer T, Stöhr-Liesen M, Schmid-Schönbein H. Red blood cell microrheology: Comparison of the behaviour of single rbc and liquid droplets in shear flow. Biorheology 1978;74(182):38–45.

Gibbs B, Kermasha S, Alli I, Mulligan C. Encapsulation in the food industry: a review. International Journal of Food Sciences and Nutrition 1999;50(3):213–24.

Gires PY, Barthès-Biesel D, Leclerc E, Salsac AV. Transient behavior and relaxation of microcapsules with a cross-linked human serum albumin membrane. Journal of the Mechanical Behavior of Biomedical Materials 2016;58:2–10.

Gubspun J, Gires PY, de Loubens C, Barthès-Biesel D, Deschamps J, Georgelin M, Léonetti M, Leclerc E, Edwards-Lévy F, Salsac AV. Size and concentration effect on the mechanical properties of cross-linked albumin capsule membranes. Colloid and Polymer Science 2016;294:1381–9.

Hu XQ, Salsac AV, Barthès-Biesel D. Flow of a spherical capsule in a pore with circular or square cross-section. Journal of Fluid Mechanics 2012;705:176–94.

Hu XQ, Sévénié B, Salsac AV, Leclerc E, Barthès-Biesel D. Characterizing the membrane properties of capsules flowing in a square-section microfluidic channel: Effects of the membrane constitutive law. Physical Review E 2013;87(6):063008.

Huttenlocher D, Klanderman G, Rucklidge W. Comparing images using the hausdorff distance. IEEE Transactions on Pattern Analysis and Machine Intelligence 1993;15(9):850–63.

Jolliffe I. Principal Component Analysis. Springer Verlag, 1986.

Koleva I, Rehage H. Deformation and orientation dynamics of polysiloxane microcapsules in linear shear flow. Soft Matter 2012;8:3681.

Loeve M. Functions aleatoires du second ordre. Processus Stochastique et Mouvement Brownien 1948;:366–420.

Lorensen W, Cline H. Marching cubes: A high resolution 3d surface construction algorithm. SIGGRAPH Comput Graph 1987;21(4):163–9.

Ma G, Su ZG. Microspheres and Microcapsules in Biotechnology: Design, Preparation and Applications. CRC Press, 2013.

Maday Y, Rønquist E. A reduced-basis element method. Comptes Rendus Mathématique 2002;335(2):195–200.

Meng L, Breitkopf P, Quilliec GL, Raghavan B, Villon P. Nonlinear shape-manifold learning approach: concepts, tools and applications. Archives of Computational Methods in Engineering 2016;:1–21.

Milcent T, Maitre E. Eulerian model of immersed elastic surfaces with full membrane elasticity. Commun Math Sci 2016;14:857–81.

Miyazawa K, Yajima I, Kaneda I, Yanaki T. Preparation of a new soft capsule for cosmetics. Journal of Cosmetic Science 2000;51(4):239–52.

Nayroles B, Touzot G, Villon P. Generalizing the finite element method: diffuse approximation and diffuse elements. Computational Mechanics 1992;10(5):307–18.

Nelson G. Application of microencapsulation in textiles. International Journal of Pharmaceutics 2002;242(1):55–62.

Pozrikidis C. Introduction to Theoretical and Computational Fluid Dynamics. Oxford university press, 1997.

Raghavan B, Breitkopf P, Tourbier Y, Villon P. Towards a space reduction approach for efficient structural shape optimization. Structural and Multidisciplinary Optimization 2013a;48(5):987–1000.

Raghavan B, Breitkopf P, Villon P. Pod-morphing, an a posteriori grid parametrization method for shape optimization. European Journal of Computational Mechanics/Revue Européenne de Mécanique Numérique 2010;19(5-7):671–97.

Raghavan B, Xia L, Breitkopf P, Rassineux A, Villon P. Towards simultaneous reduction of both input and output spaces for interactive simulation-based structural design. Computer Methods in Applied Mechanics and Engineering 2013b;265:174–85.

Samareh J. A survey of shape parameterization techniques. Technical Report; NASA; 1999.

Sederberg T, Parry S. Free-form deformation of solid geometric models. ACM SIGGRAPH Computer Graphics 1986;20(4):151–60.

Skalak R, Tozeren A, Zarda R, Chien S. Strain energy function of red blood cell membranes. Biophysical Journal 1973;V:245–64.

Walter J, Salsac AV, Barthès-Biesel D, Tallec PL. Coupling of finite element and boundary integral methods for a capsule in a stokes flow. International Journal for Numerical Methods in Engineering 2010;83(7):829–50.

Yih T, Al-Fandi M. Engineered nanoparticles as precise drug delivery systems. Journal of Cellular Biochemistry 2006;97(6):1184–90.