Vietnam Journal of Mechanics, VAST, Vol. 39, No. 2 (2017), pp. 109 – 119
DOI:10.15625/0866-7136/8120

IMMERSED BOUNDARY METHOD COMBINED WITH PROPER GENERALIZED DECOMPOSITION FOR SIMULATION OF A FLEXIBLE FILAMENT IN A VISCOUS INCOMPRESSIBLE FLOW

Cuong Q. Le[1,2,*], **H. Phan-Duc**[1], **Son H. Nguyen**[1]
[1] *Ho Chi Minh City University of Technology and Education, Vietnam*
[2] *Binh Duong Vietnam – Korea Vocational College, Vietnam*
[*] E-mail: lecuong2109@gmail.com

Received April 11, 2016

Abstract. In this paper, a combination of the Proper Generalized Decomposition (PGD) with the Immersed Boundary method (IBM) for solving fluid-filament interaction problem is proposed. In this combination, a forcing term constructed by the IBM is introduced to Navier-Stokes equations to handle the influence of the filament on the fluid flow. The PGD is applied to solve the Poission's equation to find the fluid pressure distribution for each time step. The numerical results are compared with those by previous publications to illustrate the robustness and effectiveness of the proposed method.

Keywords: Immersed boundary method, fluid-structure interaction, proper generalized decomposition.

1. INTRODUCTION

The problems of interaction between the elastic bodies and incompressible viscous fluid flow are often encountered in the field of biofluid dynamics, such as the flapping flags and swimming fishes, etc. These problems are often complicated because of the complexity of the geometries as well as the free movement of bodies. Thus, the traditional methods have to adopt body-conforming grids. As a result, re-meshing process is needed after each time step to calculate the movement or deformation of bodies in the next step.

In this context, the Immersed boundary method (IBM) has emerged as a powerful simulation tool for tackling problems with complex bodies and conditions. This is because it possesses the ability to handle complex bodies without the need of expensive and cumbersome dynamic re-meshing strategies. The IBM was first proposed by Peskin [1] in 1977 to analyze the behavior of blood flow interacting with heart valves. Since then, the IBM was employed to solve various kinds of FSI problems [2–6].

FSI problems are usually defined in a multi-dimensional space which requires very long computation times and a large storage capacity to solve with traditional methods. Recently, a method finding the solution to problems in separate forms, called Proper Generalized Decomposition (PGD) [7], was proposed to reduce the computational cost of these problems. The PGD can build low-dimensional separated representations of the solutions without the knowledge of pre-computed data. As a result, the computational cost and complexity of the PGD for solving FSI problems can be reduced significantly. So far, the PGD was applied to numerous fields such as quantum chemistry [8], reliability analysis [9], composite structure [10], and so on. In terms of FSI problems, Dumon et al. [11] used the PGD to solve the Navier-Stokes equations for both steady and unsteady lid-driven cavity problems with different Reynolds numbers. The results from this study show that the PGD was eight times faster than the standard full grid solver. In 2013, Dumon et al. [12] combined the PGD with spectral discretization to deal with the Taylor Green and the lid-driven cavity problems. Recently in 2014, Leblond et al. [13], based on the space-time PGD definitions, proposed a priori low-dimentional space-time separated representations of the fluid fields. Nonetheless, as seen from the literature, studies conducted on the PGD for FSI problems are somewhat still limited and mainly focus on simple problems having simple geometry and no obstacle in the flow.

By exploiting the advantages of both the PGD and the IBM, the paper proposes an integration of the IBM into the PGD for solving incompressible fluid flow past a flexible filament. In this combination, the IBM formulation is first used to construct the interaction of fluid-structure by introducing a forcing term to the Navier-Stokes equation. Then, a projection-based PGD method is employed to find the solution for the coupled Navier-Stokes equations. By doing so, both of the superior abilities of the IBM and PGD are exploited effectively. The IBM formulation helps handle the complex boundary of the FSI problem while the PGD helps simplify the problem. Zhang et al. [14] studied experimentally the motion of flexible filaments in a flowing soap film. The other numerical simulation results presented interaction between incompressible viscous flows with filaments are presented in [15–18].

The paper is then organized as follows. Section 2 states the mathematical formulation of the problem. In section 3, the PGD is presented. Section 4 describes numerical method to solve the fluid-filament problem. In section 5, the numerical results are presented. Finally, some conclusions are withdrawn.

2. MATHEMATICAL FORMULATION

In this section, the mathematical formulations for the problem of interaction between a flexible filament (with one end fixed and one free) and incompressible viscous flow in Fig. 1 are presented. The Eulerian coordinate $(\mathbf{x} = (x, z))$ is used to describe the fluid domain and the Lagrangian coordinate (s) is employed to describe the filament.

The incompressible viscous fluid flow is governed by the Navier-Stokes equations and the continuity equation as

$$\rho\left(\mathbf{x},t\right)\left(\frac{\partial \mathbf{u}}{\partial t}+\left(\mathbf{u}\cdot\nabla\right)\mathbf{u}\right) = -\nabla p + \mu\Delta\mathbf{u} + \mathbf{f}\left(\mathbf{x},t\right) + \rho\left(\mathbf{x},t\right)\mathbf{g}, \tag{1}$$

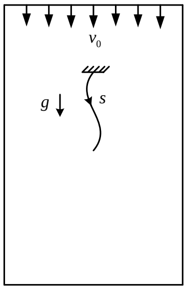

Fig. 1. Schematic diagram of the computational configuration: a filament in incompressible viscous flow

$$\nabla \cdot \mathbf{u} = 0, \tag{2}$$

where $\mathbf{u} = (u, \mathrm{w})$ is the velocity vector, p is the pressure, ρ is the density of system including the fluid and the flexible, μ is the dynamic viscosity, $\mathbf{f}$ is the momentum forcing applied to enforce no-slip boundary condition along the structure interface, and $\mathbf{g} = (0, -g)^T$ with g is the gravitational acceleration.

In Eq. (1) the term forcing $\mathbf{f}$ and the density ρ are defined as follows

$$\mathbf{f}(\mathbf{x}, t) = \int\limits_{\Gamma} \mathbf{F}(s, t)\delta(\mathbf{x} - \mathbf{X}(s, t))\, ds, \tag{3}$$

$$\rho(\mathbf{x}, t) = \rho_f + \int\limits_{\Gamma} \rho_s(s)\delta(\mathbf{x} - \mathbf{X}(s, t))\, ds, \tag{4}$$

where Γ is the boundary of filament, $\mathbf{F}$ is the Lagrangian force, $\mathbf{X}$ is the position of filament, $\delta(\mathbf{x}) = \delta(x)\,\delta(z)$ is the Dirac delta function, ρ_f is the fluid density, and ρ_s is the density of the flexible. The Lagrangian force consists of two terms: the stretching and compression force $\mathbf{F}_s$ and bending force $\mathbf{F}_b$

$$\mathbf{F}(s, t) = \mathbf{F}_s(s, t) + \mathbf{F}_b(s, t) = \frac{\partial}{\partial s}(T\boldsymbol{\tau}) - \frac{\partial^2}{\partial s^2}\left[\kappa_b \frac{\partial^2 \mathbf{X}(s, t)}{\partial s^2}\right], \tag{5}$$

where $T = K_s\left(\left|\partial \mathbf{X}/\partial s\right| - 1\right)$ is the tension in the filament, $\mathbf{\o} = \left(\partial \mathbf{X}/\partial s / \left|\partial \mathbf{X}/\partial s\right|\right)$ is the unit tangent defined at the each point of the filament.

The motion of the filament is defined as follows

$$\frac{\partial \mathbf{X}(s, t)}{\partial t} = \mathbf{U}(s, t) = \int\limits_{\Omega} \mathbf{u}(\mathbf{x}, t)\delta(\mathbf{x} - \mathbf{X}(s, t))\, d\mathbf{x}, \tag{6}$$

where Ω is the fluid domain.

The value of $\mathbf{X}(s, t = 0)$ is specified as the initial condition for the filament while the boundary condition is that $\mathbf{X}(s = 0, t)$ is constant. The Dirichlet boundary conditions $(u = 0, \mathrm{w} = v_0)$ are used for inlet and far-field boundaries while the homogeneous Neumann boundary conditions are used for outlet boundaries, and the fluid velocity is equal to zero on the left and right boundaries. The free stream velocity and zero pressure state is used as the initial state to the calculation.

3. PROPER GENERALIZED DECOMPOSITION

For the sake of clarity and without losing the general scope, PGD will be examined in the case of a 2D space decomposition. The problem is expressed as follows

$$\text{Find } U\left(x,y\right) \text{ as } l\left(U\right) = f \text{ in } \Omega, \tag{7}$$

where l is a linear differential operation and f is the prescribed function.

The approximated solution of the equation is assumed to be written in separated representations as

$$U\left(x,y\right) \approx \sum_{i=1}^{N} F_i\left(x\right) G_i\left(y\right). \tag{8}$$

The process of solving for Eq. (7) by the PGD is also an iterative procedure. The solutions at step $m+1$ is a product sum of functions of each space variable as

$$U_{m+1}\left(x,y\right) = \sum_{i=1}^{m} F_i\left(x\right) G_i\left(y\right) + R\left(x\right) S\left(y\right), \tag{9}$$

where $\sum_{i=1}^{m} F_i\left(x\right) G_i\left(y\right)$ is the sum of already-known solutions up to step m; $R\left(x\right) = F_{m+1}\left(x\right)$ and $S\left(y\right) = G_{m+1}\left(y\right)$ are additional solutions found in step $m+1$. Introducing this expression into Eq. (7), one obtains

$$l\left(U_m + R(x)S(y)\right) = f + Res_{m+1}, \tag{10}$$

where Res_{m+1} is defined by

$$Res_{m+1} = l\left(U_{m+1}\right) - f. \tag{11}$$

To find the solution for $R\left(x\right)$ and $S\left(y\right)$ from Eq. (10), an Alternating Directions Fixed-point Algorithm (ADFA) [19] is used. Details of the PGD is described as follows

Step 1: Calculation of the function $R\left(x\right)$

Assume that $S\left(y\right)$ is known, Eq. (10) is projected onto $S\left(y\right)$

$$\left\langle l\left(\sum_{i=1}^{m} F_i\left(x\right) G_i\left(y\right) + R\left(x\right) S\left(y\right)\right), S\left(y\right)\right\rangle_{L^2(Y)} = \left\langle f, S\left(y\right)\right\rangle_{L^2(Y)} + \left\langle Res_n, S\left(y\right)\right\rangle_{L^2(Y)}, \tag{12}$$

where $\langle .,.\rangle_{L^2(Y)}$ is the scalar products[1] on L^2 in the y direction. In addition with this approach, the residual must be orthogonal to $S\left(y\right)$, thus

$$\left\langle l\left(\sum_{i=1}^{m} F_i\left(x\right) G_i\left(y\right) + R\left(x\right) S\left(y\right)\right), S\left(y\right)\right\rangle_{L^2(Y)} = \left\langle f, S\left(y\right)\right\rangle_{L^2(Y)}. \tag{13}$$

Step 2: Calculation of the function $S\left(y\right)$

Up to now, the function $R\left(x\right)$ calculated in the previous step is considered as an already-known function. Using the same procedure in step 1, the differential equation for finding the function $S\left(y\right)$ can be easily constructed as

$$\left\langle l\left(\sum_{i=1}^{m} F_i\left(x\right) G_i\left(y\right) + R\left(x\right) S\left(y\right)\right), R\left(x\right)\right\rangle_{L^2(X)} = \left\langle f, R\left(x\right)\right\rangle_{L^2(X)}, \tag{14}$$

where $\langle .,. \rangle_{L^2(X)}$is the scalar products[1] on L^2 in the x direction.

Step 3: The checking convergence step

The iterative process of these two steps is conducted until the convergence criterion is satisfied. The stop criterion of Alternating Directions Fixed-point algorithm is given by

$$e = \left\| R^{(q)}\left(x\right) S^{(q)}\left(y\right) - R^{(q-1)}\left(x\right) S^{(q-1)}\left(y\right) \right\| < \varepsilon_{RS}, \tag{15}$$

where $R^{(q)}\left(x\right)$, $S^{(q)}\left(y\right)$, $R^{(q-1)}\left(x\right)$, and $S^{(q-1)}\left(y\right)$ are respectively $R\left(x\right)$, $S\left(y\right)$ functions found in the current step and the preceding step; ε_{RS} is the tolerance of convergence which is chosen to be enough small.

After the functions $R\left(x\right)$ and $S\left(y\right)$ in step $m+1$ are found by the above-mentioned ADFA, the solution for Eq. (7) now is added with an amount $R\left(x\right) S\left(y\right)$ by Eq. (9). Then, the iterative PGD is continued for the next step until the following convergence criterion is satisfied

$$E = \frac{\left\| Res_{m+1} \right\|_2}{\left\| f\left(x,y\right) \right\|} < \varepsilon, \tag{16}$$

where Res_{m+1} is defined by Eq. (11) is the residual of Eq. (7) with solutions found up to current step $m+1$; ε is the tolerance of convergence for problem which is chosen to be enough small.

4. NUMERICAL METHOD

4.1. The structure solver

4.1.1. *The stretching and compression force and bending forces*

A staggered grid is used in the Lagrangian coordinate system, as shown in Fig. 2. Let the filament be represented by a discrete collection of points, $s = k\Delta s$, where k is an integer which $k = 0$ at the free end and $k = N$ at the fixed end. The position of filament and the force density are defined at the integer points while the body tension force and the unit tangent are defined at the half-integer points.

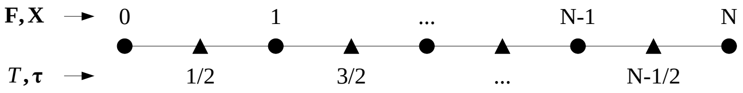

Fig. 2. Schematic diagram of the Lagrangian coordinate system for a flexible filament

The difference approximation to the first order derivative and second derivative for an arbitrary variable ϕ is defined as

$$\frac{\partial\phi\left(s\right)}{\partial s} \approx \frac{\phi\left(s+\frac{\Delta s}{2}\right) - \phi\left(s-\frac{\Delta s}{2}\right)}{\Delta s}, \tag{17}$$

$$\frac{\partial^2\phi\left(s\right)}{\partial s^2} \approx \frac{\phi\left(s-\Delta s\right) - 2\phi\left(s\right) + \phi\left(s+\Delta s\right)}{\left(\Delta s\right)^2}, \tag{18}$$

The stretching and compression force, and bending force in Eq. (5) is discretized as

$$(\mathbf{F}_s)_k = \frac{\left(T_{k+1/2}\boldsymbol{\tau}_{k+1/2} - T_{k-1/2}\boldsymbol{\tau}_{k-1/2}\right)}{\Delta s}, k = 1, 2, \ldots, N-1, \tag{19}$$

where the tension force and the unit tangent are discretized as

$$T_{k+1/2} = \kappa_s \left(\frac{|\mathbf{X}_{k+1} - \mathbf{X}_k|}{\Delta s} - 1 \right), \tag{20}$$

$$\boldsymbol{\tau}_{k+1/2} = \frac{\mathbf{X}_{k+1} - \mathbf{X}_k}{|\mathbf{X}_{k+1} - \mathbf{X}_k|}, \tag{21}$$

and the bending force is discretized as

$$(\mathbf{F}_b)_k = -\kappa_b \frac{\mathbf{C}_{k+1} - 2\mathbf{C}_k + \mathbf{C}_{k-1}}{(\Delta s)^2},\ k = 1, 2, \ldots, N-1, \tag{22}$$

where

$$\mathbf{C}_k = \frac{\mathbf{X}_{k+1} - 2\mathbf{X}_k + \mathbf{X}_{k-1}}{(\Delta s)^2},\ k = 1, 2, \ldots, N-1. \tag{23}$$

4.1.2. *The fluid – filament interaction*

With the stretching and compression force, and bending force are calculated as above, Lagrangian force density is then calculated by

$$\mathbf{F}^n = \mathbf{F}^n_s + \mathbf{F}^n_b\,. \tag{24}$$

The boundary force density is then used to calculate the force density acting on the fluid as

$$\mathbf{f}^n_{i,j} = \sum_{k=1}^{N} \mathbf{F}^n_k \delta_h \left(\mathbf{x}^n_{i,j} - \mathbf{X}^n_k \right) \Delta s, \tag{25}$$

where $\mathbf{F}^n_k$ at boundary configurations calculated as Eq. (23); $\mathbf{x}_{i,j}$ and $\mathbf{f}_{i,j}$ are respectively the fluid mesh point coordinates and its corresponding force density; $\delta_h(\mathbf{x})$ is discrete delta function in the form

$$\delta_h(\mathbf{x}) = \frac{1}{h^2} \varphi\left(\frac{x}{h}\right) \varphi\left(\frac{z}{h}\right), \tag{26}$$

where h is the Euler mesh size; $\phi(r)$ is one-dimensional discrete delta function, chosen as in [20]

$$\phi(r) = \begin{cases} \frac{1}{8}\left(3 - 2|r| + \sqrt{1 + 4|r| - 4r^2}\right), & 0 \le |r| \le 1 \\ \frac{1}{8}\left(5 - 2|r| - \sqrt{-7 + 12|r| - 4r^2}\right), & 1 \le |r| \le 2 \\ 0, & |r| \ge 2 \end{cases} \tag{27}$$

In case of the filament has mass, the density of system is calculated as

$$\rho^n_{i,j} = \rho_f + \sum_{k=1}^{N} \rho_s \delta_h \left(\mathbf{x}^n_{i,j} - \mathbf{X}^n_k \right) \Delta s. \tag{28}$$

The governing Navier-Stokes equation now is added with a force term reflecting the influence of filament on the fluid flow. It's noted that the problem is solved through a series of time steps. As a result, after solving the Navier-Stoke equation at a time step, the velocity of filament in the next step is interpolated based on results of Euler nodal fluid velocity $\mathbf{u}\left(\mathbf{x},t\right)$ of previous step as

$$\mathbf{U}_k^{n+1}\left(\mathbf{X}_k\right)=\sum_{i,j}\mathbf{u}^{n+1}\left(\mathbf{x}_{i,j}\right)\delta_h\left(\mathbf{x}_{i,j}^{n+1}-\mathbf{X}_k^{n+1}\right)h^2. \tag{29}$$

With the velocity of the filament known, the motion equation of the filament is discretization as

$$\frac{\mathbf{X}_k^{n+1}-\mathbf{X}_k^n}{\Delta t}=\mathbf{U}_k^{n+1}\left(\mathbf{X}_k\right). \tag{30}$$

4.2. The fluid solver

4.2.1. *The projection method*

In the paper, a projection method [21] is used to handle the coupling velocity-pressure in the Navier-Stokes equation. This method first solves the momentum equation at each time step to find an intermediate velocity. Then, with the obtained intermediate velocity, the main task of the problem becomes solving the Poisson's equation to find pressure for the fluid flow. By doing so, a reconstruction of estimated velocity and pressure with respect to the continuity equations is obtained for next time step. In summary, the discretized Navier-Stokes equation can be solved progressively through a series of time step by a procedure including 3 phases as

Phase 1: Directly calculate the intermediate velocity $\mathbf{u}^*$ without any notion for pressure gradient term as

$$\frac{\mathbf{u}^*-\mathbf{u}^n}{\Delta t}=-\left(\mathbf{u}^n\cdot\nabla\right)\mathbf{u}^n+\frac{\mu}{\rho^n}\Delta\mathbf{u}^n+\frac{1}{\rho^n}\mathbf{f}^n+\mathbf{g}, \tag{31}$$

where $\mathbf{u}^n$ is already-known velocity field at time step n.

Phase 2: The intermediate velocity $\mathbf{u}^*$ is used to calculate the pressure of the next time step

$$\Delta p^{n+1}=\frac{\rho^n}{\Delta t}\nabla\cdot\mathbf{u}^*. \tag{32}$$

This is a Poission's equation of pressure. After solving this Poission's equation, pressure values for time step $n+1$ is found.

Phase 3: Update the velocity for time step $n+1$ by

$$\mathbf{u}^{n+1}=\mathbf{u}^*-\frac{\Delta t}{\rho^n}\nabla p^{n+1}. \tag{33}$$

4.2.2. *Spatial discretization*

The spatial discretization is performed on a staggered grid for the resolution of the Navier-Stokes equation. With this kind of grid, the pressures are defined in the center of the cells, the horizontal velocities are defined on midpoints of the vertical cell interfaces, and the vertical velocities are defined on midpoints of the horizontal cell interfaces. Fig. 3

represents schematically a staggered grid. The equations are then solved with a finite difference method, using a second-order scheme.

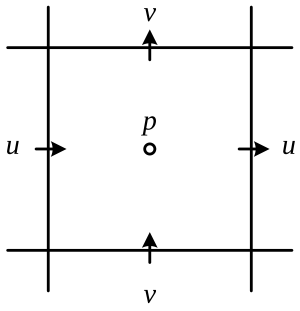

Fig. 3. Staggered grid

4.3. Summary of the numerical algorithm

The general process of the proposed numerical algorithm for simulating flexible filament in a uniform flow can be summarized as follows:

(1) At the time step n, we have known the velocity field $\mathbf{u}^n$ and the pressure field p^n and the position of the filament $\mathbf{X}^n$. From that, the Lagrangian momentum forcing $\mathbf{F}^n$ is calculated by using Eq. (24).

(2) The Lagrangian momentum forcing is spread to the Eulerian grids using Eq. (25). Eqs. (31), (32) and (33) are then solved to find the fluid velocity field and pressure field at time step $n+1$.

(3) The velocity of the filament $\mathbf{U}^{n+1}$ at the time step $n+1$ is then interpolated by using Eq. (29). Finally, the new position of the filament $\mathbf{X}^{n+1}$ is calculated by using Eq. (30). This ends one time step marching.

5. SIMULATION RESULTS

A filament is embedded in incompressible viscous fluid flow. The domain of problem is a rectangular domain that the width by 8.5 cm and the length by 17 cm. The filament with one end fixed and one free end. This problem is similar to one by Zhu and Peskin [15] except for the effect of air resistance is ignored. The parameters of simulation are shown in Tab. 1. In our simulation, a 200×400 mesh is used to discrete the fluid domain and 18 points is used to represent the filament.

Fig. 4 shows the vorticity around a massless filament. In this case, the inflow velocity is 280 cm/s, the filament length is 3 cm. The initial position of the filament is stretched straight and makes an angle of 5° from the vertical. After a few oscillations, the filament returns to its rest state (stretched straight and aligned with the flow direction). Thus, this is the globally stable state.

Fig. 5 shows the vorticity around a filament with mass. The parameters of simulation are the same as the above case, except that the mass of filament is nonzero. Fig. 6 shows the x coordinate of the free end of filament as a function of time. We found that the

Table 1. Parameters of the simulation

Inflow velocity (v_0)	280 cm/s
Dynamic viscosity (μ)	1.2×10^{-3} g/(cm.s)
Fluid density (ρ_f)	3×10^{-4} g/cm^2
Filament length (L)	3 cm
Filament density (ρ_s)	4×10^{-4} g/cm
Filament rigidity (κ_b)	0.1 erg.cm
Gravitational acceleration (g)	980 cm/s^2

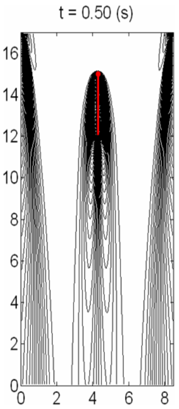

Fig. 4. Vorticity contours of massless filament in an incompressible viscous flow

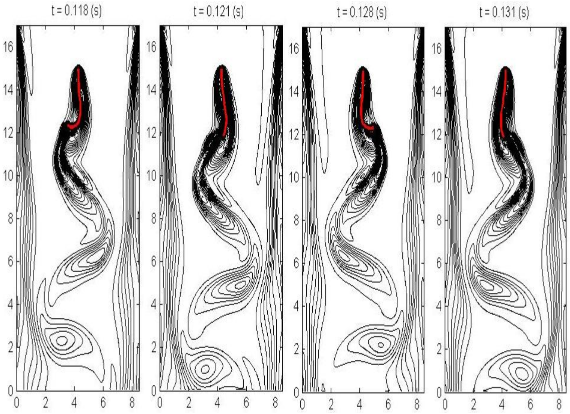

Fig. 5. Vorticity contours of filament with mass in an incompressible viscous flow

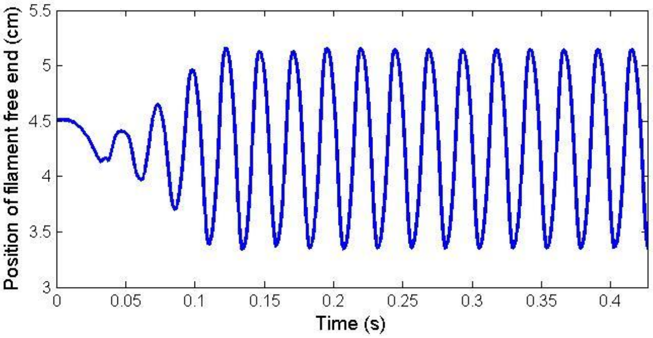

Fig. 6. The x - coordinate of the filament free end

oscillation of the filament is self-sustained and periodic in time. In our simulation, the flapping frequency (48 Hz) agrees very well with the observed experiment in [14] and the result of simulation in [15]. The total excursion of the free end is about 1.8 cm (1.5 cm in [15] and 2.1 cm in [16]).

6. CONCLUSION

The paper presents a new combination of the Immersed Boundary Method (IBM) and the Proper Generalized Decomposition (PGD) for solving the problem of interaction between incompressible viscous flows with a flexible filament. The numerical simulations show that combination of the IBM and the PGD can solve the FSI problems having complex shape boundary. Another advantage of the proposed method is the simplicity since it transforms a multi-dimensional problem to single-dimensional problems. As a result, the computational cost is reduced significantly. Moreover, in this method, the influence of the filament on the fluid flow is taken into account by a force terms added to the governing Navier-Stokes equation. This helps avoid the expensive cost of re-meshing or mesh-changing as in other grid-based methods. The present results are in good agreement with those by previous published studies. This verifies the robustness and effectiveness of the proposed method. In future studies, the proposed method will be extended to the other FSI problems in 2D and 3D space.

REFERENCES

[1] C. S. Peskin. Numerical analysis of blood flow in the heart. *Journal of Computational Physics*, **25**, (3), (1977), pp. 220–252. doi:10.1016/0021-9991(77)90100-0.

[2] A. L. F. Lima E Silva, A. Silveira-Neto, and J. J. R. Damasceno. Numerical simulation of two-dimensional flows over a circular cylinder using the immersed boundary method. *Journal of Computational Physics*, **189**, (2), (2003), pp. 351–370. doi:10.1016/s0021-9991(03)00214-6.

[3] D. Kim and H. Choi. Immersed boundary method for flow around an arbitrarily moving body. *Journal of Computational Physics*, **212**, (2), (2006), pp. 662–680. doi:10.1016/j.jcp.2005.07.010.

[4] Y. Kim and C. S. Peskin. 3-D Parachute simulation by the immersed boundary method. *Computers & Fluids*, **38**, (6), (2009), pp. 1080–1090. doi:10.1016/j.compfluid.2008.11.002.

[5] W.-X. Huang and H. J. Sung. An immersed boundary method for fluid-flexible structure interaction. *Computer Methods in Applied Mechanics and Engineering*, **198**, (33), (2009), pp. 2650–2661. doi:10.1016/j.cma.2009.03.008.

[6] D. Pan, X. Shao, J. Deng, and Z. Yu. Simulations of passive oscillation of a flexible plate in the wake of a cylinder by immersed boundary method. *European Journal of Mechanics-B/Fluids*, **46**, (2014), pp. 17–27. doi:10.1016/j.euromechflu.2014.02.001.

[7] A. Ammar, B. Mokdad, F. Chinesta, and R. Keunings. 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*, **139**, (3), (2006), pp. 153–176. doi:10.1016/j.jnnfm.2006.07.007.

[8] A. Ammar and P. Joyot. The nanometric and micrometric scales of the structure and mechanics of materials revisited: an introduction to the challenges of fully deterministic numerical descriptions. *International Journal for Multiscale Computational Engineering*, **6**, (3), (2008), pp. 191–213. doi:10.1615/intjmultcompeng.v6.i3.20.

[9] L. Gallimard, P. Vidal, and O. Polit. Coupling finite element and reliability analysis through proper generalized decomposition model reduction. *International Journal for Numerical Methods in Engineering*, **95**, (13), (2013), pp. 1079–1093. doi:10.1002/nme.4548.

[10] P. Vidal, L. Gallimard, and O. Polit. Proper generalized decomposition and layer-wise approach for the modeling of composite plate structures. *International Journal of Solids and Structures*, **50**, (14), (2013), pp. 2239–2250. doi:10.1016/j.ijsolstr.2013.03.034.

[11] A. Dumon, C. Allery, and A. Ammar. Proper general decomposition (PGD) for the resolution of Navier-Stokes equations. *Journal of Computational Physics*, **230**, (4), (2011), pp. 1387–1407. doi:10.1016/j.jcp.2010.11.010.

[12] A. Dumon, C. Allery, and A. Ammar. Proper generalized decomposition method for incompressible Navier-Stokes equations with a spectral discretization. *Applied Mathematics and Computation*, **219**, (15), (2013), pp. 8145–8162. doi:10.1016/j.amc.2013.02.022.

[13] C. Leblond and C. Allery. A priori space-time separated representation for the reduced order modeling of low Reynolds number flows. *Computer Methods in Applied Mechanics and Engineering*, **274**, (2014), pp. 264–288. doi:10.1016/j.cma.2014.02.010.

[14] J. Zhang, S. Childress, A. Libchaber, and M. Shelley. Flexible filaments in a flowing soap film as a model for one-dimensional flags in a two-dimensional wind. *Nature*, **408**, (2000), pp. 835–839. doi:10.1038/35048530.

[15] L. Zhu and C. S. Peskin. Simulation of a flapping flexible filament in a flowing soap film by the immersed boundary method. *Journal of Computational Physics*, **179**, (2), (2002), pp. 452–468. doi:10.1006/jcph.2002.7066.

[16] W.-X. Huang, S. J. Shin, and H. J. Sung. Simulation of flexible filaments in a uniform flow by the immersed boundary method. *Journal of Computational Physics*, **226**, (2), (2007), pp. 2206–2228. doi:10.1016/j.jcp.2007.07.002.

[17] E. Uddin, W.-X. Huang, and H. J. Sung. Interaction modes of multiple flexible flags in a uniform flow. *Journal of Fluid Mechanics*, **729**, (2013), pp. 563–583. doi:10.1017/jfm.2013.314.

[18] J. Favier, A. Revell, and A. Pinelli. Numerical study of flapping filaments in a uniform fluid flow. *Journal of Fluids and Structures*, **53**, (2015), pp. 26–35. doi:10.1016/j.jfluidstructs.2014.11.010.

[19] A. Ammar, M. Normandin, F. Daim, D. Gonzalez, E. Cueto, and F. Chinesta. Non incremental strategies based on separated representations: applications in computational rheology. *Communications in Mathematical Sciences*, **8**, (3), (2010), pp. 671–695. doi:10.4310/cms.2010.v8.n3.a4.

[20] C. S. Peskin. The immersed boundary method. *Acta Numerica*, **11**, (2002), pp. 479–517. doi:10.1017/cbo9780511550140.007.

[21] A. J. Chorin. Numerical solution of the Navier-Stokes equations. *Mathematics of Computation*, **22**, (104), (1968), pp. 745–762. doi:10.2307/2004575.