Journal of Computational Physics 228 (2009) 4588–4605

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

An arbitrary Lagrangian–Eulerian formulation for the numerical simulation of flow patterns generated by the hydromedusa *Aequorea victoria*

Mehmet Sahin *, Kamran Mohseni

Department of Aerospace Engineering Sciences, University of Colorado, 1111 Engineering Drive, Boulder, CO 80309, USA

ARTICLE INFO

Article history:
Received 5 April 2008
Received in revised form 7 March 2009
Accepted 20 March 2009
Available online 31 March 2009

Keywords:
ALE methods
Geometric conservation law
Finite volume
Jellyfish swimming

ABSTRACT

A new geometrically conservative arbitrary Lagrangian–Eulerian (ALE) formulation is presented for the moving boundary problems in the swirl-free cylindrical coordinates. The governing equations are multiplied with the radial distance and integrated over arbitrary moving Lagrangian–Eulerian quadrilateral elements. Therefore, the continuity and the geometric conservation equations take very simple form similar to those of the Cartesian coordinates. The continuity equation is satisfied exactly within each element and a special attention is given to satisfy the geometric conservation law (GCL) at the discrete level. The equation of motion of a deforming body is solved in addition to the Navier–Stokes equations in a fully-coupled form. The mesh deformation is achieved by solving the linear elasticity equation at each time level while avoiding remeshing in order to enhance numerical robustness. The resulting algebraic linear systems are solved using an ILU(k) preconditioned GMRES method provided by the PETSc library. The present ALE method is validated for the steady and oscillatory flow around a sphere in a cylindrical tube and applied to the investigation of the flow patterns around a free-swimming hydromedusa *Aequorea victoria* (crystal jellyfish). The calculations for the hydromedusa indicate the shed of the opposite signed vortex rings very close to each other and the formation of large induced velocities along the line of interaction while the ring vortices moving away from the hydromedusa. In addition, the propulsion efficiency of the free-swimming hydromedusa is computed and its value is compared with values from the literature for several other species.

Published by Elsevier Inc.

1. Introduction

Moving boundary problems in computational fluid dynamics have become of great interest due to their wide range of application areas. Examples include a wide variety of fluid-structure interaction problems such as wing flutter and tail buffeting, a large class of free-surface problems, fluid-particle interactions, swimming/flying animals, rotor-fuselage aerodynamic interactions, etc. In order to simulate the flow problems with moving boundaries, several numerical approaches have been presented in the literature including the arbitrary Lagrangian–Eulerian (ALE) method [20], the immersed boundary method [37,32] and the fictitious domain method [18].

In the ALE method, the mesh follows the interface between the fluid and solid boundary and the governing equations are discretized on an unstructured moving mesh. This differs from the standard Eulerian formulation in a way that the mesh movement has to fulfill special conditions in order to maintain the accuracy and the stability of the time integration scheme. This condition is satisfied by the enforcement of the so-called geometric conservation law (GCL) as coined by Thomas and

* Corresponding author. Tel.: +1 303 735 3526.
E-mail address: msahin.ae00@gtalumni.org (M. Sahin).

0021-9991/$ - see front matter Published by Elsevier Inc.
doi:10.1016/j.jcp.2009.03.027

Lombard [48]. The geometric conservation law requires that the volumetric increment of a moving cell must be equal to the summation of the volumes swept by its surfaces that close the volume. It can be interpreted such that a numerical scheme should preserve a uniform flow solution exactly independent of the mesh motion. Although the GCL is satisfied easily in the continuous sense, their discrete implementation may not be trivially satisfied. The ALE time integration scheme developed by Koobus and Farhat [26] is based on more continuous time integration of the fluxes. Such a scheme offers second-order accuracy in time obeying the GCL, but the integration will be computationally expensive. Geuzaine et al. [17] have showed that the GCL is neither a necessary nor a sufficient condition condition for an ALE scheme to preserve its order of time-accuracy established on fixed meshes. Recently, Mavriplis and Yang [29] have proposed a general framework for deriving high-order temporal schemes which respects the GCL. In the present work, a geometrically conservative arbitrary Lagrangian–Eulerian formulation is presented in the swirl-free cylindrical coordinates. The governing equations are multiplied by the radial distance r so that the GCL takes very simple form similar to that of the Cartesian coordinates. In addition, this will allow us to avoid singularities related to $1/r$ and $1/r^2$ terms in the cylindrical coordinates and result in better conditioned linear systems. Although similar modifications to the Navier–Stokes equations are presented in the literature [21], they are not considered in the view of the GCL.

The modified governing equations are discretized using the semi-staggered finite volume method [41,43] on all-quadrilateral unstructured moving meshes while allowing to the use of structured meshes as well. The continuity equation is satisfied exactly within each element and a special attention is given to satisfy the geometric conservation law at discrete level. The choice of the present semi-staggered approach leads to better pressure coupling compared to non-staggered (collocated) approach while being capable of handling non-Cartesian grids. In addition, it eliminates the need for a pressure boundary condition since it is defined at interior points. Furthermore, the summation of the continuity equation within each element can be exactly reduced to the domain boundary, which is important for the global mass conservation. But the most appealing feature of the method is leading to very simple algorithm consistent with the boundary and initial conditions required by the Navier–Stokes equations. Recently, the semi-staggered arrangement of variables has been used by Rida et al. [38], Kobayashi et al. [25] and Wright and Smith [50] for triangular, quadrilateral and hybrid meshes in 2D. The extension of the semi-staggered approximation to arbitrary Lagrangian–Eulerian form is reported by Hirt et al. [20] and Smith and Wright [47]. The convective fluxes in the momentum equations are approximated using both the second-order simple averages and the second-order upwind least square interpolation [2,4] in order to maintain stability at higher Reynolds numbers. The mesh motion is determined by solving the linear elasticity equation similar to the work of Refs. [22,14] at each time level while avoiding remeshing in order to enhance numerical robustness.

The resulting algebraic linear systems are solved using the GMRES method [40] with the restricted additive Schwarz preconditioner. The implementation of the preconditioned Krylov subspace algorithm and the restricted additive Schwarz preconditioner were carried out using the PETSc [3] software package developed at the Sandia National Laboratories. The computational meshes are partitioned using the METIS library [24] and within each sub-domain incomplete LU (ILU(k)) preconditioner is employed. In order to avoid the zero-block in the saddle point problem, we use an upper triangular right preconditioner which results in a scaled discrete Laplacian instead of a zero block in the original system. Unfortunately, this leads to a significant increase in the number of non-zero elements following the matrix-matrix multiplication. However, the new system may be efficiently preconditioned using ILU(k) preconditioner. The algebraic linear systems are solved in a fully-coupled manner including the equation of motion of a deforming body. This will lead to more robust solution techniques compared to SIMPLE, SIMPLER, etc. type decoupled solution techniques. Convergence of these decoupled solution techniques can often be problematic and may even result in nonconvergence. An extensive review on the fully-coupled iterative solvers for the incompressible Navier–Stokes equations may be found in [44].

The present ALE formulation is applied to the free-swimming oblate hydromedusa *Aequorea victoria* (crystal jellyfish) in order to investigate the propulsion mechanism of an oblate hydromedusa. The experimental observations indicate that the oblate medusae create more complex wake structure than those observed of more prolate jetting medusa and swim with jet-paddling mode of propulsion [30]. However, this propulsion mechanism for the oblate medusae is not well understood in the literature. The present numerical simulations for the free-swimming oblate hydromedusa *A. victoria* indicate the shed of the opposite signed vortex rings very close to each other and the formation of large induced velocities along the line of interaction while the ring vortices moving away from the hydromedusa. This mechanism of propulsion is very similar to the experimental observations of Dabiri et al. [12] for the hydromedusa *Aurelia aurita*. Although Ford et al. [15] have reported series of toroid vortex rings traveling along the medusan oral arms and tentacles, which is very similar to our fixed medusa simulations, this is particularly due to its very large tentacles causing the hydromedusa *Chrysaora quinquecirrha* hardly move through the surrounding fluid.

The remainder of this paper is organized as follows: In Section 2 the present ALE method is described along with the geometric conservation law and the mesh deformation technique. Validation of the present numerical method is given in Section 3. This is followed by the discussion on the flow patterns generated by the free swimming of the hydromedusa *A. victoria*. Conclusions are provided in Section 4.

2. Mathematical and numerical formulation

The incompressible Navier–Stokes equations that govern the swirl-free axisymmetric viscous fluid flow in the Eulerian cylindrical coordinates system (x, r) can be written in dimensionless form as follows: the continuity equation

$$-\frac{\partial u}{\partial x}-\frac{\partial v}{\partial r}-\frac{v}{r}=0 \tag{1}$$

the momentum equations

$$Re\left[\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial r}\right]+\frac{\partial p}{\partial x}=\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial r^2}+\frac{1}{r}\frac{\partial u}{\partial r} \tag{2}$$

$$Re\left[\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial r}\right]+\frac{\partial p}{\partial r}=\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial r^2}+\frac{1}{r}\frac{\partial v}{\partial r}-\frac{v}{r^2} \tag{3}$$

In these equations (u, v) are the axial and radial velocity components, respectively, p is the pressure and Re is the dimensionless Reynolds number. After multiplying the equations by the radial distance r and then modifying them, the continuity equation becomes

$$-\frac{\partial(ru)}{\partial x}-\frac{\partial(rv)}{\partial r}=0 \tag{4}$$

the momentum equations become

$$Re\left[r\frac{\partial u}{\partial t}+ru\frac{\partial u}{\partial x}+rv\frac{\partial u}{\partial r}\right]+\frac{\partial(rp)}{\partial x}=\frac{\partial}{\partial x}\left(r\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial r}\left(r\frac{\partial u}{\partial r}\right) \tag{5}$$

$$Re\left[r\frac{\partial v}{\partial t}+ru\frac{\partial v}{\partial x}+rv\frac{\partial v}{\partial r}\right]+\frac{\partial(rp)}{\partial r}-p=\frac{\partial}{\partial x}\left(r\frac{\partial v}{\partial x}+u\right)+\frac{\partial}{\partial r}\left(r\frac{\partial v}{\partial r}+v\right) \tag{6}$$

Integrating the differential Eqs. (4)–(6) over an arbitrary moving Eulerian–Lagrangian control volume $\Omega(t)$ with boundary $\partial\Omega(t)$ and using the Reynolds transport theorem [16] yield

$$-\oint_{\partial\Omega}(ru)dr+\oint_{\partial\Omega}(rv)dx=0 \tag{7}$$

$$Re\left[\frac{\partial}{\partial t}\int_{\Omega}(ru)dxdr+\oint_{\partial\Omega}(u-\dot{x})(ru)dr-\oint_{\partial\Omega}(v-\dot{r})(ru)dx\right]+\oint_{\partial\Omega}(rp)dr=\oint_{\partial\Omega}\left(r\frac{\partial u}{\partial x}\right)dr-\oint_{\partial\Omega}\left(r\frac{\partial u}{\partial r}\right)dx \tag{8}$$

$$\begin{aligned}&Re\left[\frac{\partial}{\partial t}\int_{\Omega}(rv)dxdr+\oint_{\partial\Omega}(u-\dot{x})(rv)dr-\oint_{\partial\Omega}(v-\dot{r})(rv)dx\right]-\oint_{\partial\Omega}(rp)dx-\int_{\Omega}pdxdr\\&=\oint_{\partial\Omega}\left(r\frac{\partial v}{\partial x}+u\right)dr-\oint_{\partial\Omega}\left(r\frac{\partial v}{\partial r}+v\right)dx\end{aligned} \tag{9}$$

where $(\dot{x}, \dot{r})$ are the axial and radial grid velocity components, respectively. The GCL in the cylindrical coordinate system takes the following form:

$$\frac{\partial}{\partial t}\int_{\Omega}rdxdr=\oint_{\partial\Omega}r\dot{x}dr-\oint_{\partial\Omega}r\dot{r}dx \tag{10}$$

This is very similar to the GCL equation in the Cartesian coordinate system. If the coordinate transformation $(y = r^2/2)$ proposed in Ref. [33] is used it can be exactly reduced to the same form. Although, this transformation does not allow us to avoid the singularities related to $1/r$ and $1/r^2$ terms, one may use the coordinate singularity treatment given in [35]. The above governing Eqs. (7)–(9) are discretized by extending the dilation-free semi-staggered finite volume method [41,43] to the cylindrical coordinates. However, the numerical discretization should ensure that the GLC is satisfied at the discrete level [48].

2.1. The discrete geometric conservation law

The geometric conservation law demands that the volumetric increment of a moving element must be equal to the summation of the volumes swept by its surfaces that close the volume. This can also be interpreted such that the grid positions and velocities are evaluated in a certain way that a numerical scheme preserves a constant solution exactly independent of the mesh motion. Although the GCL is satisfied easily in the continuous sense, their discrete implementation may not be trivially satisfied. In the present work, the GCL is satisfied over the dual finite volume constructed by connecting the centroids c_i of the quadrilateral elements which share a common vertex as seen in Fig. 1. The details of the two-dimensional dual control volume surrounding a node P at time levels n and $n+1$ is shown in Fig. 2. Then the GCL equation is discretized as follows:

$$\sum_{m=1}^{l}\left[\frac{r_G^{n+1}A^{n+1}-r_G^{n}A^{n}}{\Delta t}\right]_m=\sum_{m=1}^{l}\left[\bar{r}_G\dot{x}\frac{\Delta r^n+\Delta r^{n+1}}{2}\right]_m-\sum_{m=1}^{l}\left[\bar{r}_G\dot{r}\frac{\Delta x^n+\Delta x^{n+1}}{2}\right]_m \tag{11}$$

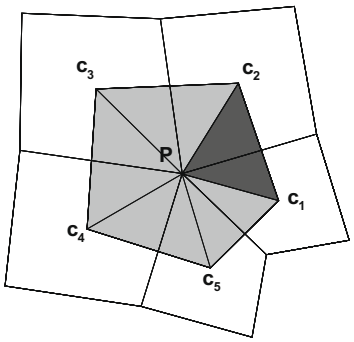


Fig. 1. Two-dimensional unstructured mesh with a dual control volume surrounding a node *P*.

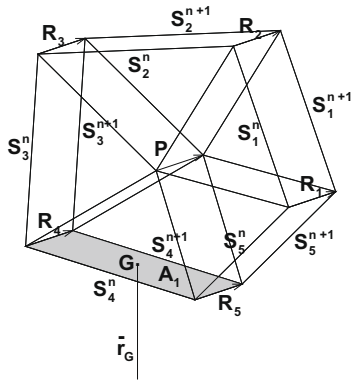


Fig. 2. A typical two-dimensional dual control volume surrounding a node P at time levels n and $n+1$. The shaded surface area A_1 is the area swept by the element edge $\mathbf{S}_1^n$, G is the geometric centroid of the shaded surface and $\bar{r}_G$ is the radial distance to the geometric centroid G.

where m is the summation over all dual control volume edges, l is the number of quadrilateral elements which share a common vertex, A^n and A^{n+1} are the surface area of the triangle elements at time levels n and $n+1$, respectively, r_G^n and r_G^{n+1} are the geometric centroid of the triangle elements and $\bar{r}_G$ is the geometric centroid of the surface swept by each edge $\mathbf{S}_1^n, \mathbf{S}_2^n, \mathbf{S}_3^n, \mathbf{S}_4^n$ and $\mathbf{S}_5^n$ shown in Fig. 2. The grid velocity components $(\dot{x}, \dot{r})$ are computed at the two endpoints of each dual control volume edge using

$$\dot{x} = \frac{x^{n+1} - x^n}{\Delta t} \tag{12}$$

$$\dot{r} = \frac{r^{n+1} - r^n}{\Delta t} \tag{13}$$

and the simple averages are used for the face value. The present discretization is very similar to the Crank–Nicholson scheme where the surface integral of the Eq. (10) is evaluated at both time levels n and $n+1$. However, the geometric centroid value $\bar{r}_G$ is used for the radial distance in the surface integral rather than the simple average value. The present discretization satisfies the GCL and the proof will be given below.

In fact, even though the left-hand side of the Eq. (10) is an area integral this integral is actually equal to the volume generated by rotating the surface element about the symmetry axis in the cylindrical coordinates when it is multiplied by 2π. Therefore, the GCL can be easily verified at the discrete level by keeping in mind that the volumetric increment of a moving cell should be equal to the summation of the volumes swept by the surfaces that close the volume. The volume generated by rotating the surface element about the symmetry axis can be easily computed using the Pappus's second theorem which states that the volume V of a solid of revolution generated by rotating a plane surface area A about an axis is equal to the product of the area A and the distance between the axis and its geometric centroid. Therefore, the GCL Eq. (10) can be written

$$\frac{V^{n+1} - V^n}{\Delta t} = \sum_{m=1}^{l} \left[\frac{r_G^{n+1} A^{n+1} - r_G^n A^n}{\Delta t} \right]_m = \sum_{m=1}^{l} (\bar{r}_G)_m A_m \tag{14}$$

where A_m is the area swept by each edge of the quadrilateral element shown in Fig. 2 and $\bar{r}_G$ is the geometric centroid of the area A_m. The area created by sweeping the edge $\mathbf{S}_1^n$ can be computed from

$$A_1 = \frac{1}{4}(\mathbf{S}_1^n + \mathbf{S}_1^{n+1}) \times (\mathbf{R}_1 + \mathbf{R}_2) \tag{15}$$

where the $\mathbf{S}_1^n, \mathbf{S}_1^{n+1}, \mathbf{R}_1$ and $\mathbf{R}_2$ are schematic in Fig. 2. In here, the terms $\mathbf{S}_1^n$ and $\mathbf{S}_2^{n+1}$ correspond to the surface integrals at time levels n and $n+1$ and the simple averages of the terms $\mathbf{R}_1$ and $\mathbf{R}_2$ correspond to the average grid velocity vector multiplied by Δt. This means that the GCL law is satisfied provided that the grid velocities are computed according to the Eqs. (12) and (13) and the r term in the surface integral of the Eq. (10) is computed from the geometric centroid of the surface swept by each edge.

2.2. Numerical discretization

The momentum Eqs. (8) and (9) are also discretized over the dual finite volume where the GCL is satisfied exactly at the discrete level. The reason to utilize the present centroid-dual polyhedral finite volume is due to its superior stability properties on highly irregular meshes compared to the medial-dual finite volume [1]. This is an important issue for ALE schemes where poor quality meshes could be encountered easily. The discrete contribution from cell c_1 to cell c_2 shown in Fig. 1 is given below for each term of the momentum equation in the axial direction.
The time derivative

$$Re\left[\frac{r_P^{n+1} + r_1^{n+1} + r_2^{n+1}}{3}\right]\left[\frac{u_P^{n+1} + u_1^{n+1} + u_2^{n+1}}{3\Delta t}\right]A_{P12}^{n+1} - Re\left[\frac{r_P^{n} + r_1^{n} + r_2^{n}}{3}\right]\left[\frac{u_P^{n} + u_1^{n} + u_2^{n}}{3\Delta t}\right]A_{P12}^{n} \tag{16}$$

The convective term due to mesh motion

$$-Re\ \bar{r}_G\left[\frac{\dot{x}_1 + \dot{x}_2}{2}\right]\left[\frac{u_1^{n+1} + u_2^{n+1}}{2}\right]\frac{\Delta r_{12}^{n+1} + \Delta r_{12}^{n}}{2} + Re\ \bar{r}_G\left[\frac{\dot{r}_1 + \dot{r}_2}{2}\right]\left[\frac{u_1^{n+1} + u_2^{n+1}}{2}\right]\frac{\Delta x_{12}^{n+1} + \Delta x_{12}^{n}}{2} \tag{17}$$

The convective term due to fluid motion

$$Re\left[\frac{r_1^{n+1} + r_2^{n+1}}{2}\right]\left[\frac{u_1^{n} + u_2^{n}}{2}\right]\left[\frac{u_1^{n+1} + u_2^{n+1}}{2}\right]\Delta r_{12}^{n+1} - Re\left[\frac{r_1^{n+1} + r_2^{n+1}}{2}\right]\left[\frac{v_1^{n} + v_2^{n}}{2}\right]\left[\frac{u_1^{n+1} + u_2^{n+1}}{2}\right]\Delta x_{12}^{n+1} \tag{18}$$

The pressure term

$$\left[\frac{r_1^{n+1} + r_2^{n+1}}{2}\right]\left[\frac{p_1^{n+1} + p_2^{n+1}}{2}\right]\Delta r_{12}^{n+1} \tag{19}$$

The viscous term

$$-\left[\frac{r_1^{n+1} + r_2^{n+1}}{2}\right]\left[\left(\frac{\partial u}{\partial x}\right)_1^{n+1} + \left(\frac{\partial u}{\partial x}\right)_2^{n+1}\right]\Delta r_{12}^{n+1} + \left[\frac{r_1^{n+1} + r_2^{n+1}}{2}\right]\left[\left(\frac{\partial u}{\partial r}\right)_1^{n+1} + \left(\frac{\partial u}{\partial r}\right)_2^{n+1}\right]\Delta x_{12}^{n+1} \tag{20}$$

In here, A_{P12} is the area between the points P, c_1 and $c_2, \bar{r}_G$ is the geometric centroid of the area swept by the dual edge $\mathbf{S}_1$ between the time levels n and $n+1$, $\Delta x_{12} = x_2 - x_1$ and $\Delta r_{12} = r_2 - r_1$. It should be noted that the use of the simple averages instead of $\bar{r}_G$ in the Eq. (17) will violate the GCL. Similarly, the use of $\bar{r}_G$ in the Eq. (18) will violate a constant solution since $\sum_{m=1}^{l} \bar{r}_{G_m} \Delta r_m - \sum_{m=1}^{l} \bar{r}_{G_m} \Delta x_m \neq 0$, where m is the summation over the dual finite volume edges. This means that the satisfaction of the GCL Eq. (10) alone does not preserve the constant solution exactly independent of the mesh motion. Although, the satisfaction of the GCL is enough for the constant solution in the Cartesian coordinates, this is due the fact that $\sum_{m=1}^{l} \Delta r_m - \sum_{m=1}^{l} \Delta x_m = 0$. Therefore, a special care should be given to the convective fluxes due to fluid motion and these fluxes should be integrated in a similar manner to the continuity equation. Otherwise, a source term will be created since $div\mathbf{u} \neq 0$ at the discrete level and it may lead the divergence of a numerical algorithm. The velocity values at the element centroids c_i is computed from the element vertex values using the simple averages and the gradient of velocity components are calculated from the Green's theorem.

$$\frac{\partial u}{\partial x} = \frac{1}{A}\oint_{\partial\Omega_e} u dr \tag{21}$$

$$\frac{\partial u}{\partial r} = -\frac{1}{A}\oint_{\partial\Omega_e} u dx \tag{22}$$

where the line integral on the right-hand side of the Eqs. (21) and (22) is evaluated using the mid-point rule on each of the element faces. The contribution from the other cells are also calculated in a similar manner. The discretization of the radial momentum equation follows very closely the ideas presented here and therefore is not repeated here. The continuity Eq. (7) is integrated within each elements and evaluated using the mid-point rule on each of the element faces. The discretization of above equations leads to a saddle point problem of the form:

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \tag{23}$$

where A_{11} is the convection diffusion operator, A_{12} is the pressure gradient operator and A_{21} is the divergence operator. In here A_{22} and b_2 are zero and $A_{12} \neq A_{21}^{\top}$. Although the system matrix of (23) is indefinite due to zero diagonal block, recent results indicate that indefiniteness of the problem does not represent a particular difficulty and a recent review of the iterative methods for solving large saddle point problems may be found in [6,39,44]. However, due to the zero diagonal block resulting from the divergence-free constraint, an ILU(k) type preconditioner cannot be used directly for the saddle point problem. Here, we consider an upper triangular right preconditioner in order to avoid problems arising from the zero block. The modified system becomes

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & 0 \end{bmatrix} \begin{bmatrix} I & -A_{12} \\ 0 & I \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} - A_{11}A_{12} \\ A_{21} & -A_{21}A_{12} \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ 0 \end{bmatrix}. \tag{24}$$

and the zero block is replaced with $-A_{21}A_{12}$, which is a scaled discrete Laplacian. Unfortunately, this leads to a significant increase in the number of non-zero elements due to the matrix–matrix multiplication. However, the new system may be solved efficiently by using preconditioned Krylov subspace methods. The implementation of the preconditioned Krylov subspace algorithm and the matrix–matrix multiplications were carried out using the PETSc [3] software package developed at the Sandia National Laboratories. Although there are several Krylov subspace algorithms readily available in the PETSc library, we only employ the GMRES algorithm [40] for the problems presented in this paper, due to its stability. The new system is precondioned using the restricted additive Schwarz preconditioner. The computational mesh is partitioned using the METIS library [24] and for each sub-block an ILU(k) preconditioner with reverse Cuthill–McKee ordering [11] is used.

In addition to the simple averages for convective term in the Eqs. (17) and (18) a second-order upwind least square interpolation [4,2] is also used in order to extrapolate the velocity values to the boundaries of the dual finite volume elements at high Reynolds numbers. The use of present least square approximation to the gradient term in order to compute the convective term results in the same coefficients computed from the second-order upwind interpolation on uniform Cartesian meshes. Therefore, our approximation is second-order similar to the second-order upwind interpolation.

2.3. Mesh motion

The ALE formulation requires a scheme for moving mesh vertices as the boundaries of a computational domain translate, rotate and deform. Several mesh deforming algorithms have been presented in the literature including the spring analogy [5], the elastic medium analogy [22], the edge swapping algorithm [13] and the remeshing algorithm [23]. In the present mesh moving scheme we assume that computational domain is made of linear elastic materials and the equations of linear elasticity are solved in order to obtain the displacement of each vertex along with the incompressible Navier–Stokes equations. The equations of the linear elasticity are given in the Cartesian coordinate system (x, y) by

$$\frac{\partial}{\partial x}\left[(\lambda + 2\mu)\frac{\partial q_1}{\partial x} + \lambda\frac{\partial q_2}{\partial y}\right] + \frac{\partial}{\partial y}\left[\mu\left(\frac{\partial q_1}{\partial y} + \frac{\partial q_2}{\partial x}\right)\right] = 0 \tag{25}$$

$$\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial q_1}{\partial y} + \frac{\partial q_2}{\partial x}\right)\right] + \frac{\partial}{\partial y}\left[(\lambda + 2\mu)\frac{\partial q_2}{\partial y} + \lambda\frac{\partial q_1}{\partial x}\right] = 0 \tag{26}$$

where (q_1, q_2) is the displacement vector and λ and μ are the Lame's constants. These terms can be expressed in terms of Young's modulus E and Poisson's ratio v as

$$\lambda = \frac{vE}{(1+v)(1-2v)}, \quad \mu = \frac{E}{2(1+v)} \tag{27}$$

Although the solution of the above elasticity equations with Dirichlet boundary condition is generally enough for small deformations it may lead to entanglement of small size quadrilateral meshes close to the domain boundary for large deformations. In order to enhance the robustness of the method the stiffness of small elements close to the domain boundary are increased by modifying the Young's module based on the distance function $d(E = A + B/d)$. In addition, λ and μ values are replaced by $\lambda = -E$ and $\mu = E$. The elasticity equations are discretized on the initial mesh in a similar manner to the momen-

tum equations given in Section 2.2. Therefore, the global stiffness matrix and its preconditioner are constructed only once during the first step. The quadrilateral elements close to wall surfaces exhibit good orthogonality for the present parameters. However, the use of a constant E value with $\lambda = -E$ and $\mu = E$ leads to the Laplace equation and it is not as robust as the solution of the elasticity equations. The robustness of the method can be further increased by defining more complex non-linear relation between the strain tensor and the displacement vector as given in Ref. [14]. However, this gain is accomplished with a significant increase in computational cost.

3. Numerical results

In this section numerical results are presented using the present arbitrary Lagrangian–Eulerian method described in Section 2. The first numerical results are presented for validation and assessment of accuracy. Then the results are followed by the numerical solution of the flow patterns around the free-swimming hydromedusa *A. victoria*. The present numerical calculations have been performed on the Phantom Linux Cluster and the IBM Machine Bluefire at NCAR.

3.1. Numerical validation

The first set of validation cases corresponds to the steady flow past a sphere in a cylindrical tube. For this problem we consider a sphere of radius R positioned symmetrically within a cylindrical tube of radius $2R$. The length to diameter ratio of the tube is approximately 20. The dimensionless parameters are the Reynolds number $Re = 2\rho\langle U\rangle R/\mu$ and the Strouhal number $St = \omega R/\langle U\rangle$. The physical parameters are the density ρ, the average velocity at the inlet $\langle U\rangle$, the viscosity of the fluid μ and the angular frequency ω. The boundary conditions are the fully developed pipe (Hagen–Poiseuille) flow velocity boundary conditions at the inlet and natural (traction-free) boundary conditions at the outlet. No-slip zero velocity boundary conditions are imposed on all solid walls. Along the axis of symmetry the radial velocity component is set to zero meanwhile the axial momentum equation is solved numerically for the axial velocity component. In order to investigate the mesh dependency of the solutions, four different meshes are employed: mesh M1 with 20,552 node points and 20,064 elements, mesh M2 with 78,628 node points and 77,670 elements, mesh M3 with 318,728 node points and 316,794 elements and mesh M4 with 1,059,082 node points and 1,055,740 elements. The successive meshes are generated by multiplying the mesh sizes by 1/2 in each direction and the details of the meshes are provided in Table 1. All the meshes are stretched next to the sphere surface and the lateral solid walls as seen in Fig. 3 and the mesh refinements in these regions are nested within the successive coarse meshes in order to have more uniform mesh refinement. The rest of computational domain is meshed using the paving algorithm provided in the CUBIT mesh generation environment [7]. The governing equations are solved together with the associated boundary conditions at the several different Re numbers as given in Table 2 and the computed drag coefficient $C_d = 2F_x/\rho\langle U\rangle^2\pi R^2$ at $Re = 100$ is compared with the result of Sheard and Ryan [46]. The numerical value of Sheard and Ryan [46] is obtained using the spectral element method and a good agreement is seen between the present value and that of Sheard and Ryan [46]. In order to estimate the spatial convergence the present method a relationship is sought between the minimum tangential mesh spacing on the sphere surface $\Delta S_{min}/R$ and the drag coefficient C_d in the form of $C_d = A + B(\Delta S_{min}/R)^n$ similar to the reference [42]. For this purpose, we use the meshes M2–M4 since the computational mesh M1 is relatively very coarse. Then the drag coefficient based on mesh spacing is estimated to be $C_d = 3.8189 + 2.7378(\Delta S_{min}/R)^{2.03}$. Therefore, the present numerical scheme can actually achieve second-order spacial accuracy in the swirl-free cylindrical coordinates since the governing equations are multiplied with the radial distance r so that the singularities related to $1/r$ and $1/r^2$ terms are removed. Although, the computed zero mesh size drag coefficient is slightly lower than the value of Sheard and Ryan [46] this is due to fact that the value computed by the authors is not mesh independent. The computed contours of u-velocity component with streamlines are given in Fig. 4 for $Re = 1$ and $Re = 100$. The streamlines indicate significant delay of the critical Reynolds number for the flow separation or wake formation beyond $Re_{crit} > 20$ compared to the unbounded case. In addition, the computed drag factor $F^* = F_x/6\pi\mu\langle U\rangle R$ is compared with the numerical result of Owens and Phillips [36] for the Stokes flow around a rigid sphere falling along the axis of a cylindrical tube with a radius of $2R$. The numerical results of Owens and Phillips [36] is also obtained using the spectral element method and the authors showed that the drag factor value is converged to a mesh independent value ($F^* = 5.9474$) as the mesh is refined. The mesh convergence of the drag factor F^* is also given in here using the meshes in Table 1 and the results are compared with the value of Owens and Phillips [36]. The comparison shows excellent agreement and the results converge towards the mesh independent value of Owens and Phillips [36] as seen in Table 3. If we repeat the convergence analysis

Table 1
Description of quadrilateral meshes for the viscous flow past a sphere in a cylindrical tube. $\Delta S_{min}/R$ and $\Delta r_{min}/R$ are the minimum tangential and normal mesh spacing on the sphere surface.

Mesh	Number of nodes	Number of elements	$\Delta S_{min}/R$	$\Delta r_{min}/R$	DOF
M1	20,552	20,064	0.078540	0.005499	61,168
M2	78,628	77,670	0.039270	0.002544	234,926
M3	318,728	316,794	0.019635	0.001224	954,250
M4	1,059,082	1,055,740	0.009817	0.000601	3,173,904

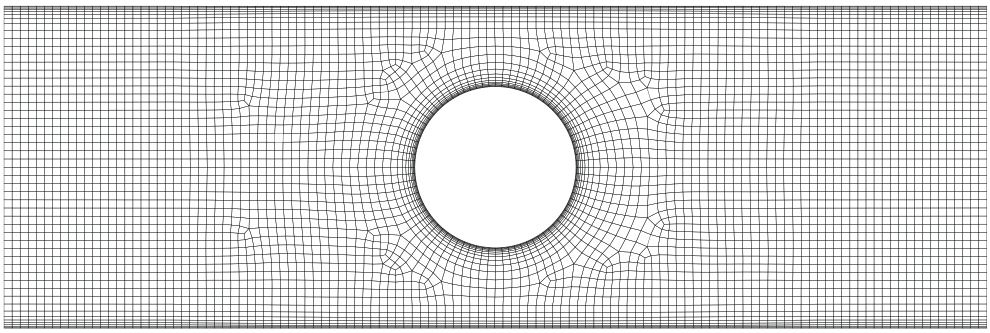


Fig. 3. The computational coarse mesh M1 for the flow past a sphere in a circular tube with 20,552 and 20,064 elements.

Table 2
The comparison of the drag coefficient $C_d = 2F_x/\rho\langle U\rangle^2\pi R^2$ for the flow past a sphere in a circular tube.

Re	M1	M2	M3	M4	Sheard and Ryan [46]
1	2.3978×10^2	2.3980×10^2	2.3984×10^2	2.3984×10^2	–
10	2.4580×10^1	2.4547×10^1	2.4545×10^1	2.4545×10^1	–
20	1.2893×10^1	1.2858×10^1	1.2853×10^1	1.2852×10^1	–
40	7.2053	7.1783	7.1730	7.1719	–
100	3.8341	3.8227	3.8198	3.8191	3.8196

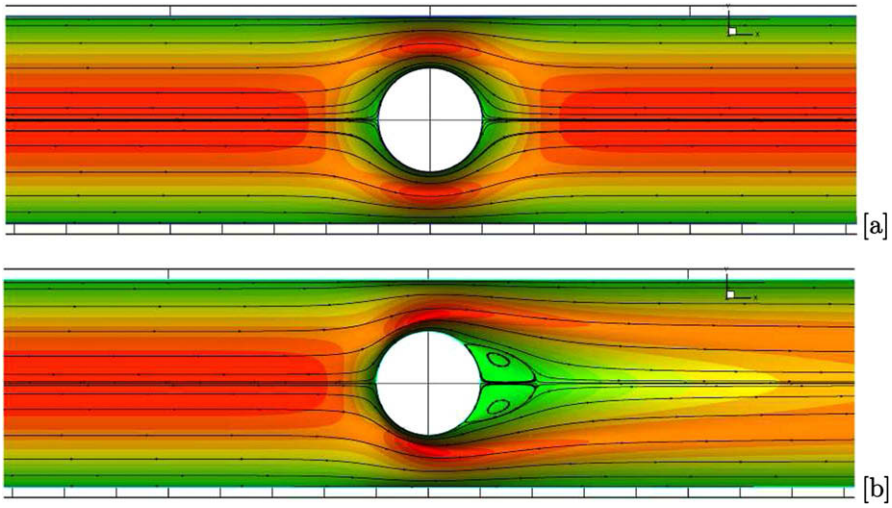


Fig. 4. The computed contours of u-velocity component with streamlines for the flow past a sphere in a cylindrical tube at $Re = 1$ [a] and $Re = 100$ [b] on mesh M3.

Table 3
The comparison of the drag factor $F^* = F_x/6\pi\mu\langle U\rangle R$ for a free falling sphere in a cylindrical tube for the Stokes flow.

	M1	M2	M3	M4	Owens and Phillips [36]
F^*	5.9424	5.9455	5.9469	5.9473	5.9474

given above, the drag factor is found to be $F^* = 5.9474 - 1.9590(\Delta S_{min}/R)^{2.16}$ and the zero mesh size drag factor is in excellent agreement with the value of Owens and Phillips [36]. The convergence of the error in the drag factor is also shown with the mesh refinement in Fig. 5 using the mesh independent drag factor.

The second set of validation cases corresponds to the flow around a rigid sphere undergoing sinusoidal transverse oscillation in a cylindrical tube with a specified amplitude and frequency. For this problem, we used the meshes M1–M3 given in Table 1 since the time dependent calculations are prohibitively expensive on mesh M4. The sphere center is oscillating sinusoidally such that the location of the sphere center is given by $x = [U_0/\omega]sin(\omega t)$ where t is the time and U_0 is the maximum

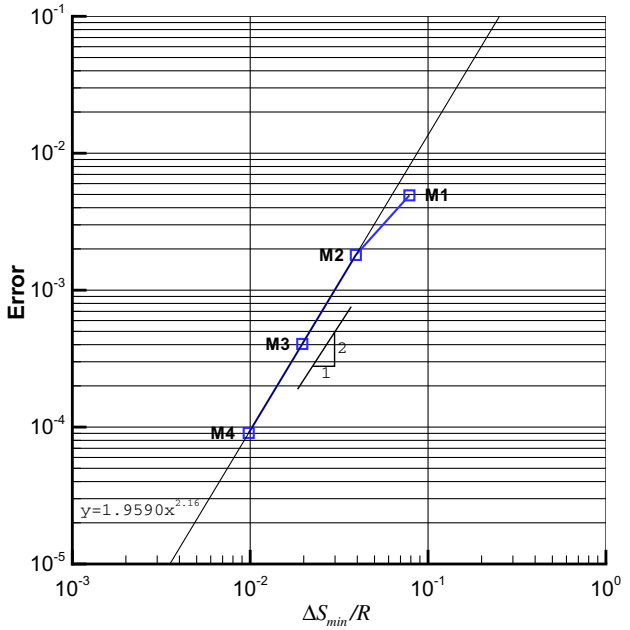


Fig. 5. The convergence of error in the drag factor F^* with the mesh refinement for a free falling sphere in a cylindrical tube for the Stokes flow.

velocity. The dimensionless parameters Re and St are based on the maximum velocity U_0. The boundary conditions are set to no-slip grid velocities on all boundaries. The calculations are first carried out for very small amplitude oscillations where the non-linear convective terms can be neglected and analytical solutions become possible. Mei et al. [31] have presented solutions for the motion of a sphere oscillating with a small amplitude inside a cylindrical tube. The present numerical results with $\Delta t = T/200$ are compared with the results of Mei et al. [31] for a Stokes number $\epsilon = \sqrt{StRe/4} = 64$ by setting $Re = 40$ and $St = 409.6$. The comparison of the axial velocity component along the center line is given in Fig. 6 and from the comparison, we can see that the present numerical results on the meshes M2 and M3 are almost identical with the finite difference solution of Mei et al. [31]. Although, the authors presented an asymptotic solution and a good agreement is seen within the Stokes layers, such a solution does not satisfy the mass conservation along the centerline and significant differences are observed outside the Stokes layers. In addition, we also present numerical results with a larger displacement of $0.5R$ where

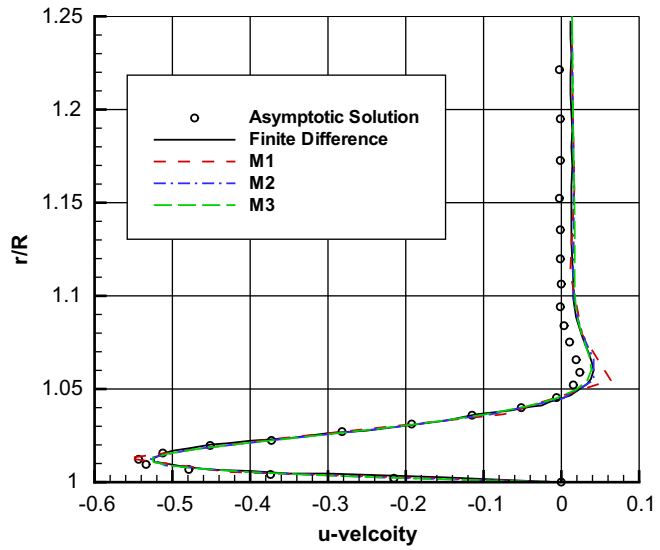


Fig. 6. The comparison between the present simulations and the results of Mei et al. [31] (asymptotic solution (∘) and finite difference solution (—)) for the u-velocity component on the centerline.

the non-linear terms no longer be neglected and the GCL can be verified. The dimensionless Stokes number is taken to be $\epsilon = \sqrt{20}$ by setting $Re = 40$ and $St = 2$. The computed drag coefficient is given as a function of time in Fig. 7 on meshes M1–M3 and the plotted curves are almost indistinguable from one another. For more precise comparison, the maximum value of the drag coefficient for each mesh is provided in Table 4. The computed contours of u-velocity component with streamlines are also given at the several time levels $t = 10\pi, t = 10\pi + T/4, t = 10\pi + T/2$ and $t = 10\pi + 3T/4$ in Fig. 8. The streamlines indicate the formation of very large separation bubbles on the cylindrical tube wall. The temporal convergence of the present method on deforming meshes is tested on mesh M2 by setting times steps Δt to $T/100, T/200$ and $T/400$. The computed maximum drag coefficient corresponding to each time step is provided in Table 5 and the same convergence analysis given above reveals that $C_{d\,max} = 8.3443 + 2.5729\Delta t^{1.05}$. The estimated error in the maximum drag coefficient is also plotted as a function of time step in Fig. 9 using the estimated maximum drag coefficient for $\Delta t \to 0$. The first-order temporal accuracy established on the moving meshes should be expected since the satisfaction of the GCL is enough to preserve first-order time-accuracy of the given scheme on a moving mesh [19]. The satisfaction of the GCL can be easily verified by setting uniform boundary and initial conditions everywhere ($u = 1, v = 0$ and $p = 0$) and solving the flow field around the transversely oscillating sphere for several cycles. The calculations indicate that that the initial RMS value for the GMRES solver is less than 10^{-13} and the iterative solver did not perform any iteration. Therefore, the uniform flow field is preserved up to machine precision and the numerical scheme actually obeys the GCL.

3.2. The flow around the hydromedusa A. victoria

In this section the flow pattern around the free-swimming medusa *A. victoria* is computed using the present ALE formulation. The kinematics of the swimming hydromedusa *A. victoria* was obtained by direct video recordings obtained from Dr. Sean Colin at Roger Williams University. A sample snapshot of the free-swimming hydromedusa is shown in Fig. 10. The geometry of the medusa at each snapshot is approximated by NURBS curves and Fourier-series interpolation is used in time. The maximum bell radius of the medusa is approximately $R_{max} = 2.3$ cm and the period of one cycle T is being approximately equal to 1.16 s. The maximum contraction ratio of the bell margin is $R_{min}/R_{max} = 0.72$. The kinematic viscosity of water is taken to be $\mu/\rho = 1 \times 10^{-2}$ cm^2/s. Although the medusa is capable of changing its body density compared to the density of surrounding water, it is assumed to be equal to the density of water. To compute the velocity of the medusa, the following equation of motion is solved in addition to the Navier–Stokes equations in a fully coupled form.

$$V\frac{dU}{dt} = F_x \tag{28}$$

where V is the volume occupied by the medusa ($V = 9.78$ cm^3), U is the instantaneous velocity of the medusa mass center in a motionless frame and F_x is the total force acting on the medusa in the axial direction given by

$$F_x = \frac{2\pi}{Re}\left[-\oint(rp)dr + \oint\left(r\frac{\partial u}{\partial x}\right)dr - \oint\left(r\frac{\partial u}{\partial r}\right)dx\right] \tag{29}$$

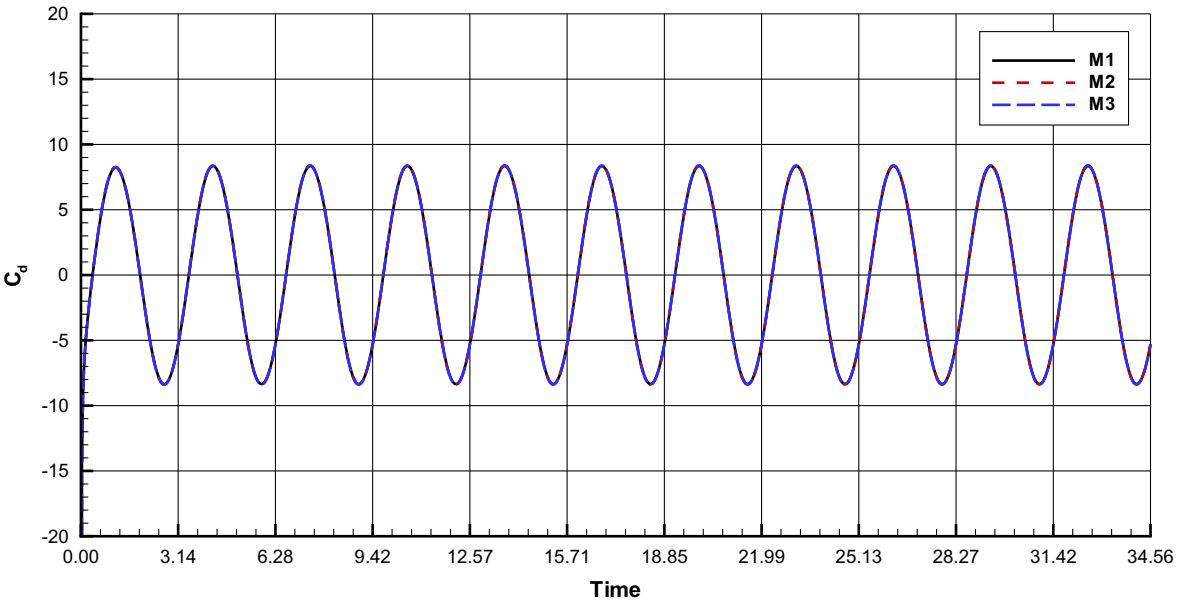


Fig. 7. Time variation of the drag coefficient for an oscillating sphere in a circular tube at $Re = 40$ and $St = 2$ with $\Delta t = T/200$.

Table 4
The convergence of the maximum drag coefficient with the mesh refinement for an oscillating sphere in a circular tube at $Re = 40$ and $St = 2$ with $\Delta t = T/200$.

	M1	M2	M3
$C_{d_{max}}$	8.3434	8.3770	8.3883

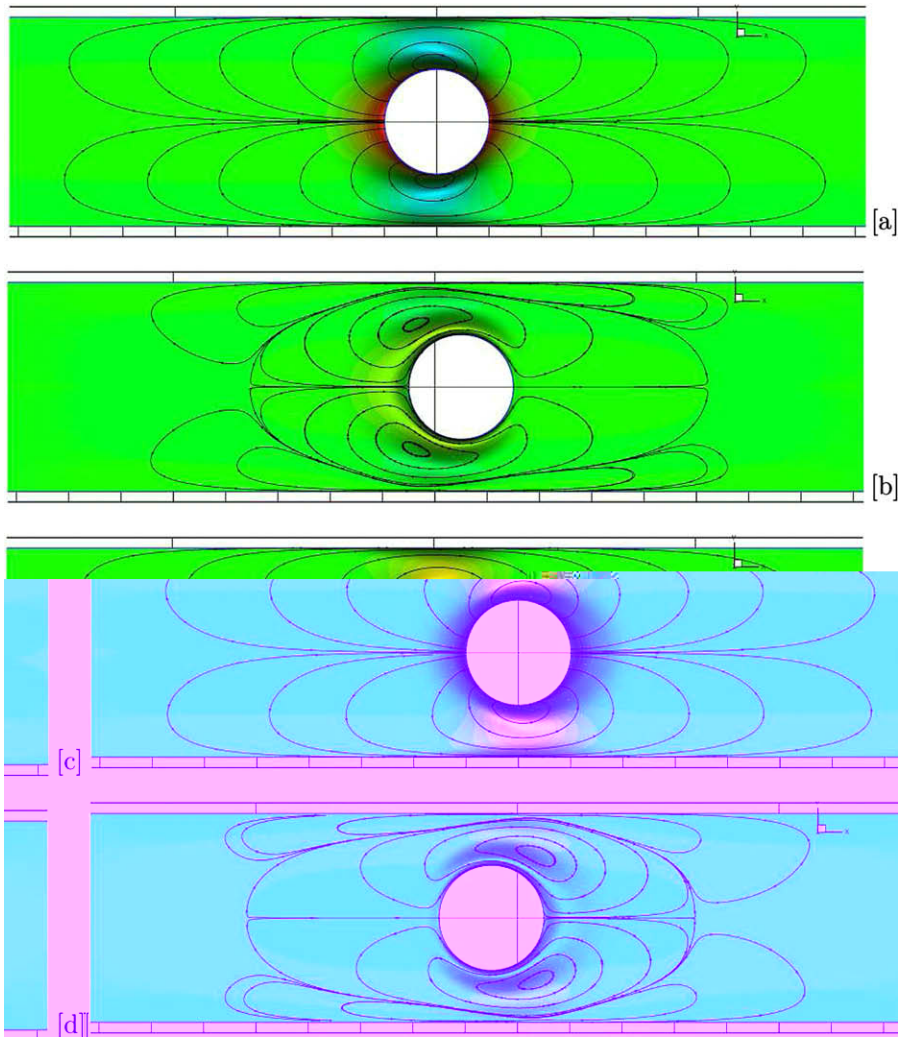


Fig. 8. The sequences of u-velocity component contours with streamlines around a transversely oscillating sphere in a cylindrical tube at times $t = 10\pi$ [a], $t = 10\pi + T/4$ [b], $t = 10\pi + T/2$ [c] and $t = 10\pi + 3T/4$ [d] on mesh M3.

Table 5
The temporal convergence of the maximum drag coefficient on mesh M2 for an oscillating sphere in a circular tube at $Re = 40$ and $St = 2$.

Δt	$T/100$	$T/200$	$T/400$
$C_{d_{max}}$	8.4120	8.3770	8.3601

Although the above equations between the medusa velocity U and the Navier–Stokes equations are decoupled they are coupled through the Dirichlet boundary condition on the medusa surface. Two different computational meshes have been considered in order to investigate the mesh dependency of the present solutions: a coarse mesh M1 with 63,099 vertices and 62,610 quadrilateral elements and a fine mesh M2 with 205,714 vertices and 204,784 quadrilateral elements. The successive meshes are generated by multiplying the mesh sizes by 1/2 in each direction and the meshes are stretched next to the medusa surface in order to resolve the viscous flow within the boundary layer as it may be seen in Fig. 11. The rectangular far

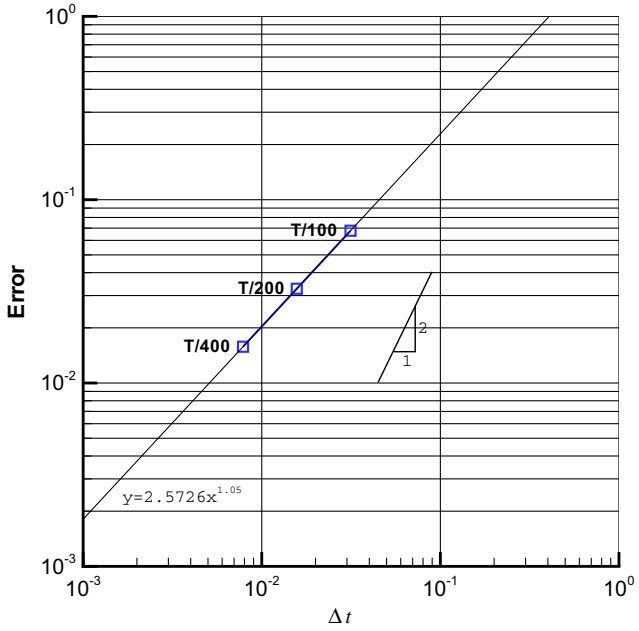


Fig. 9. The temporal convergence of error in the maximum drag coefficient for an oscillating sphere in a circular tube at $Re = 40$ and $St = 2$ on mesh M2.

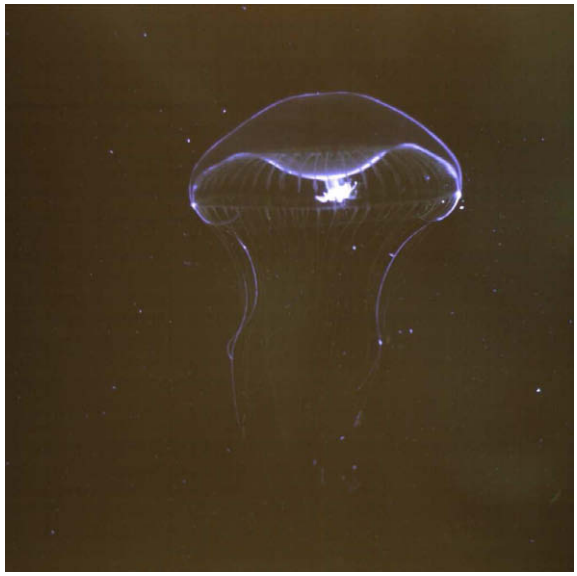
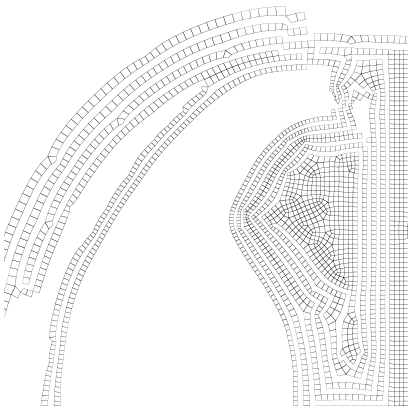


Fig. 10. A snapshot of a free-swimming hydromedusa *Aequorea Victoria*. (From Dr. Sean Colin at Roger Williams University).

field computational domain boundary starts $24R_{max}$ upstream of the medusa and ends $64R_{max}$ downstream of the medusa. The lateral far field boundaries are $24R_{max}$ away from the medusa centroid. The meshes are deformed in every time step using the scheme described in Section 2.3 and the coarse mesh M1 corresponding to the minimum bell diameter is shown in Fig. 11(b). It is remarkable that even for such a large mesh deformation, the final mesh still remains to be valid. However, the solution of the Laplace equation for the same problem results in entanglement of elements. The algebraic linear systems are solved using the GMRES algorithm [40] preconditioned by the restricted additive Schwarz algorithm and within each sub-block an ILU(5) preconditioner with reverse Cuthill-McKee ordering [11] is used. The relative residual is set to 10^{-8}. The time-dependent calculations on mesh M1 converge within approximately 60 iterations per time step and require approximately 5 s to construct and solve the fully coupled Navier–Stokes equations on the IBM Machine Bluefire using 8 nodes. Meanwhile, the calculations on mesh M2 converge within approximately 140 iterations and require approximately



10 s on 32 nodes. The time required for the mesh deformation is far less than the time required for the solution of the fully coupled Navier–Stokes equations.

The instantaneous variations of the vorticity field around the free-swimming medusa are presented in Fig. 12 at several time levels on mesh M2. The vorticity field indicates the formation of a starting vortex ring as the bell contracted during the power stroke. The starting vortex ring is formed very close to the opposite signed stopping vortex ring which is created within the subumbrellar cavity during the bell relaxation of the previous cycle. Therefore, the starting vortex ring interacts with the the stopping vortex within the subumbrellar cavity and creates very large induced velocities along the line of interaction while the ring vortices moving away from the hydromedusa. The large induced velocities and the location of the vortex rings may be seen more clearly from the contours of u-velocity component with the streamlines given in Fig. 13. From the streamlines in Figs. 13(b–d) we observe that the induced velocities between the two opposite signed vortex rings are not exactly parallel to the axis of the symmetry even though it is desirable for maximum thrust. The hydromedusa may control the direction of the induced velocities in different combinations in order to maneuver. In the wake the stopping vortex rings dissipate more quickly due to the stretching effect of the local flow velocity. Eventually the starting vortex rings form a series of vortex rings in the wake which generate an opposite force to trust the medusa body forward. In addition to the jet propulsion mechanism, the hydromedusa's bell margin acts like a paddle as the hydromedusa rows through the surrounding fluid. This motion creates a low pressure field on the leeward side of the bell margin. If the highly flexible bell margin deforms in a way that the leeward side of the bell margin has a projected area in the direction of motion it creates thrust. This propulsion mechanism is particularly important during the refill of the subumbrellar cavity where the stopping vortex causes significant pressure drag. Therefore, the oblate hydromedusa *A. victoria* uses both paddling and jet propulsion mechanism together to propel itself through the surrounding fluid. The present description of the propulsion mechanism is very similar to the experimental observations of Dabiri et al. [12] even though the authors have examined the swimming cycle of the hydromedusa *Aurelia aurita*. Although Ford et al. [15] have reported series of toroid vortex rings traveling along the medusan oral arms and tentacles, which is very similar to our fixed medusa simulations, this is particularly due to its very large tentacles causing the hydromedusa *Chrysaora quinquecirrha* hardly move through the surrounding fluid.

The time variation of the instantaneous medusa velocity is given in Fig. 14 for the meshes M1–M2 using $\Delta t = T/200$. The computed average swimming velocity $\langle U \rangle$ is computed to be 1.4530 cm/s and 1.4625 cm/s for the meshes M1 and M2, respectively. In order to investigate the effect of the temporal resolution, the time step is set to $\Delta t = T/400$ on the mesh M1 and the computed average swimming velocity is found out to be 1.4676 cm/s. The computed average swimming velocities are almost independent of the mesh size as well as the time step. The computed time variation of the free-swimming medusa velocity is in good agreement with the experiment results of Colin and Costello for the same species of the hydromedusa with $R_{max} = 2.5$ cm [9,10]. The numerical results replicate the oscillation seen in the experiments. It is remarkable that both the average swimming velocity and the amplitude of the velocity oscillations seem to be in good agreement with the experimental values. Based on the average medusa velocity on mesh M2 and the maximum bell diameter, the dimensionless parameters Reynolds and Strouhal numbers are computed to be 672 and 8.47, respectively. However, if the Strouhal is defined in analogy with bluff body flows as $St = A/T\langle U \rangle$, where A is the total excursion of the bell margin, the Strouhal number yields a value of 0.50. This value is close to the range of the Strouhal numbers (0.25–0.35) reported by Triantafyllou et al. [49] for most fishes. In addition to the free-swimming velocity, the mesh convergence is given for the reconstructed three-dimensional wake structure behind the free-swimming hydromedusa *A. victoria* in Fig. 15. The plotted absolute vorticity isosurfaces on both meshes corresponds to the time levels of Figs. 12(f) and 13(f) and are almost indistinguable from one another. The present wake structure is quite similar to

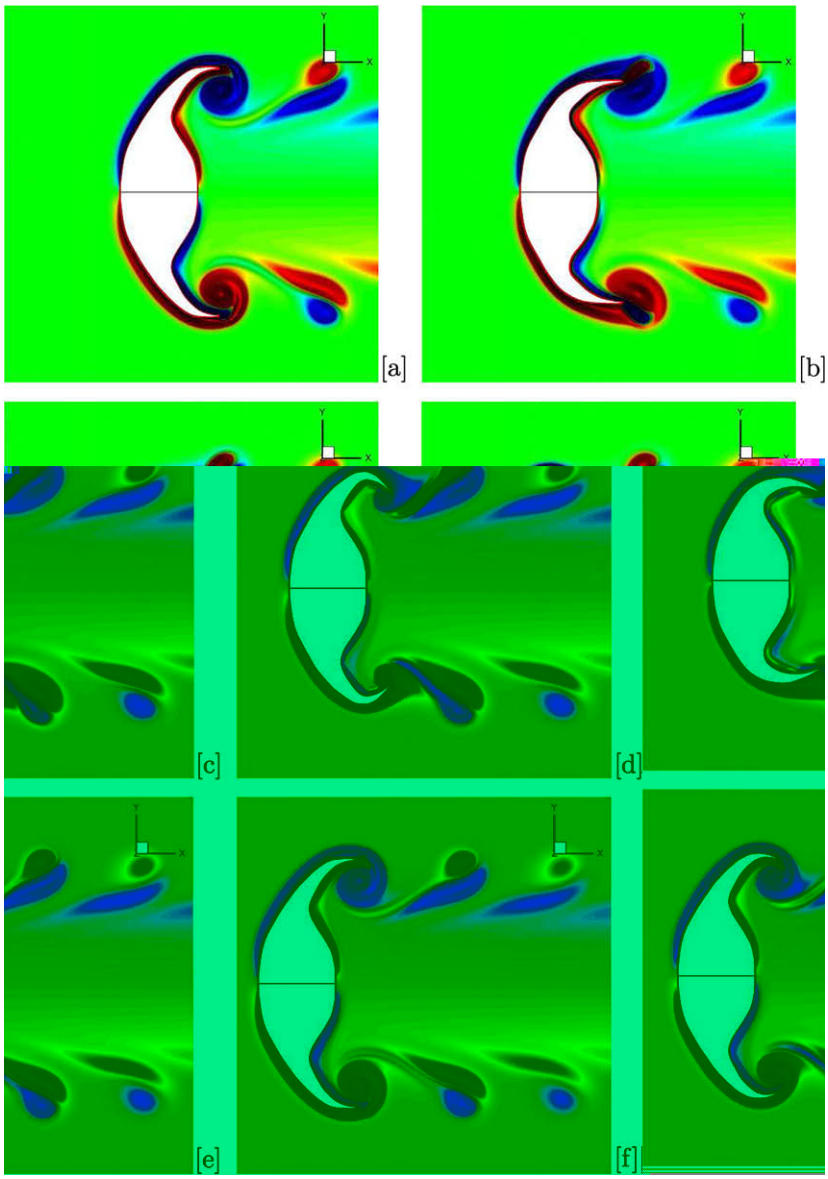


Fig. 12. The sequences of vorticity contours around a free-swimming hydromedusa *Aequorea victoria* taken at times $t = t_0$ [a], $t = t_0 + T/5$ [b], $t = t_0 + 2T/5$ [c], $t = t_0 + 3T/5$ [d], $t = t_0 + 4T/5$ [e] and $t = t_0 + T$ [f] on mesh M2.

the wake structure depicted by Dabiri et al. in Fig. 3 of [12]. The time variation of the thrust coefficient, the skin friction drag coefficient and the total drag coefficient are also shown against time in Fig. 16 and the symbols "$\circ$" represent the time levels corresponding to Figs. 12 and 13. As it may be seen, the skin friction acts always in the direction against movement. Although the trust coefficient generally acts in the direction of motion, we observe negative effects during the end of the bell contraction phase and just before the end of the bell relaxation phase. The maximum trust is seen to appear during the formation of the starting vortex ring at $t = 34.10$ s. At this point the viscous drag force is also relatively very large. The trust observed during the bell relaxation phase around at $t = 34.58$ s is due to the paddling motion of the bell margin.

The Froude efficiency [28] is used to evaluate and compare the propulsion efficiency and it is defined as:

$$\eta = \frac{\langle T \rangle \langle U \rangle}{\langle P \rangle} \tag{30}$$

where $\langle T \rangle$ is the average trust due to the pressure term and $\langle P \rangle$ is the average total power supplied by the medusa for one cycle. The work done by the medusa is computed as a time integral of the power output from its surface to the surrounding fluid:

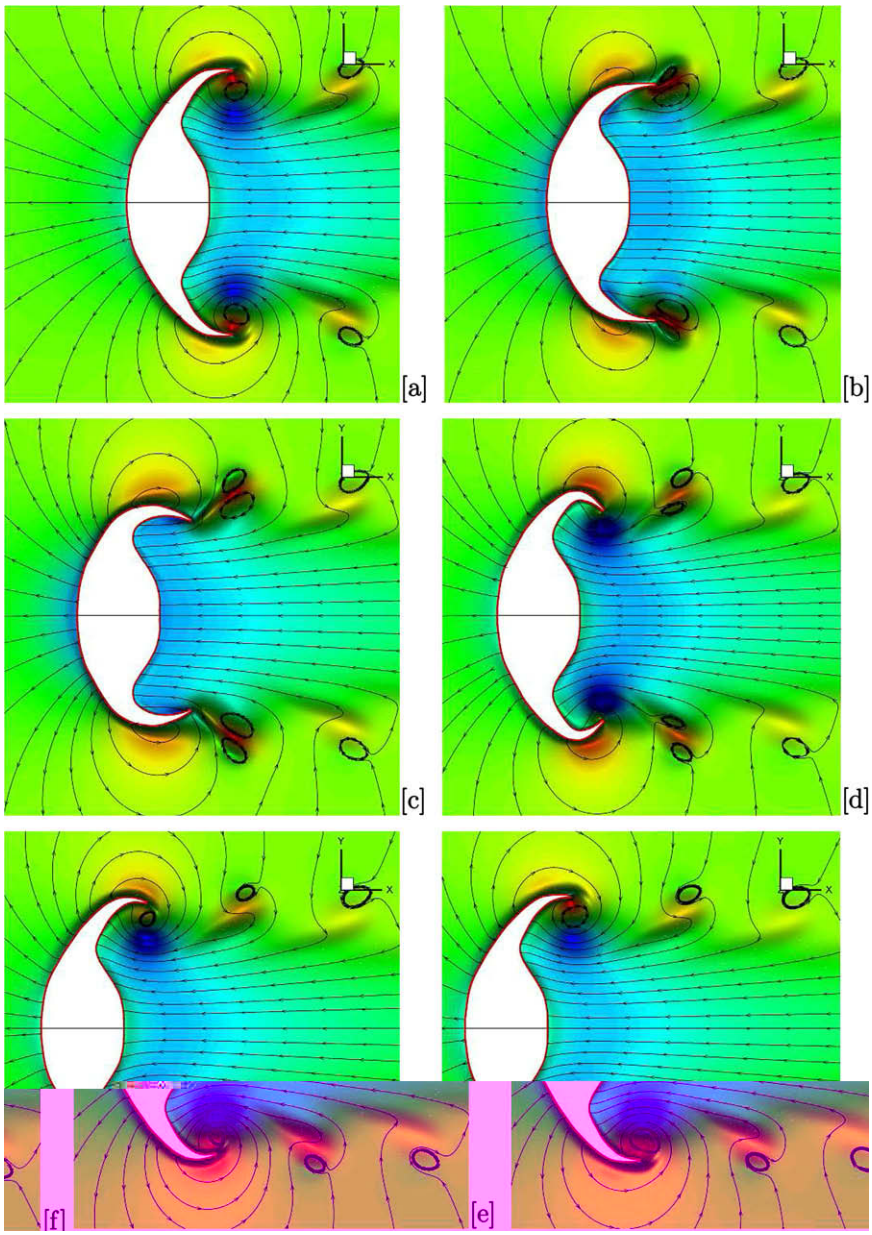


Fig. 13. The sequences of u-velocity component contours with streamlines around a free-swimming hydromedusa *Aequorea victoria* taken at times $t = t_0$ [a], $t = t_0 + T/5$ [b], $t = t_0 + 2T/5$ [c], $t = t_0 + 3T/5$ [d], $t = t_0 + 4T/5$ [e] and $t = t_0 + T$ [f] on mesh M2.

$$\langle P \rangle = \frac{1}{T}\int_{t_0}^{t_0+T}\oint \mathbf{n}\cdot(\boldsymbol{\sigma}\cdot\mathbf{u})dSdt \tag{31}$$

where $\mathbf{n}$ is the surface normal vector, $\boldsymbol{\sigma}$ is the stress tensor including the pressure term and $\mathbf{u}$ is the velocity vector on the deforming body surface. The propulsion efficiency of the hydromedusa *A. victoria* on mesh M2 is computed to be approximately 37% which is relatively lower than the values reported for most fishes [45] but higher than the efficiency of microorganism at very low Reynolds numbers (for example, 2% for the bacterium *Escherichia coli* [8]). The low value of the propulsion efficiency is due to relatively low Reynolds number at which the shed vortex rings dissipate rapidly away from the hydromedusa and highly oblate shape of the hydromedusa. The calculations with larger medusa size show significant increase in the medusa velocity indicating that the medusa may evolve much larger sizes and still be able to move through the surrounding fluid. The present thrust mechanism may offer low speed thrust and maneuvering capabilities for underwater vehicles and robots similar to the vortex ring generator in Refs. [34,27].

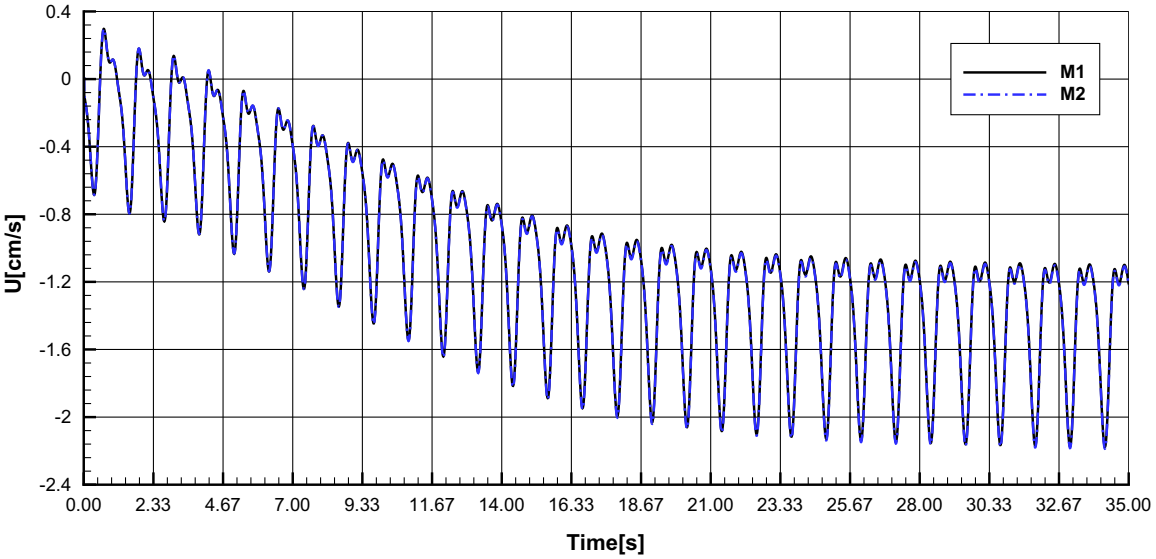


Fig. 14. The mesh convergence is given for the instantaneous velocity of a free-swimming hydromedusa *Aequorea victoria* starting from the rest.

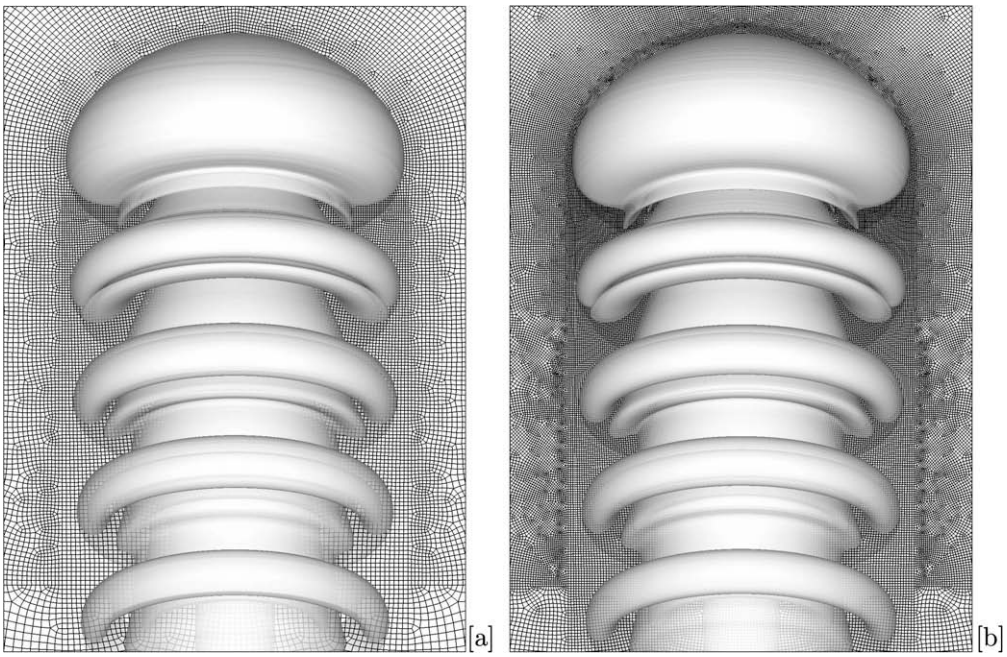


Fig. 15. The mesh convergence is given for the wake structure behind a free-swimming hydromedusa *Aequorea victoria* on meshes M1 [a] and M2 [b].

4. Conclusions

In the present work, a new geometrically conservative arbitrary Lagrangian–Eulerian formulation is presented in the swirl-free cylindrical coordinates. The continuity equation is satisfied exactly within each element and a special attention is given to satisfy the geometric conservation law at the discrete level. The present ALE method is validated for the steady and oscillatory flow around a sphere in a cylindrical tube and applied to the investigation of the flow patterns around a free-swimming oblate hydromedusa *A. victoria*. The present results are, as far as we are aware, the first numerical calculations towards understanding the underlying propulsion mechanisms of an oblate medusa. The primary calculations indicate the shed of the opposite signed vortex rings very close to each other and the formation of large induced velocities along the line of interaction while the ring vortices moving away from the hydromedusa. These numerical observations are in

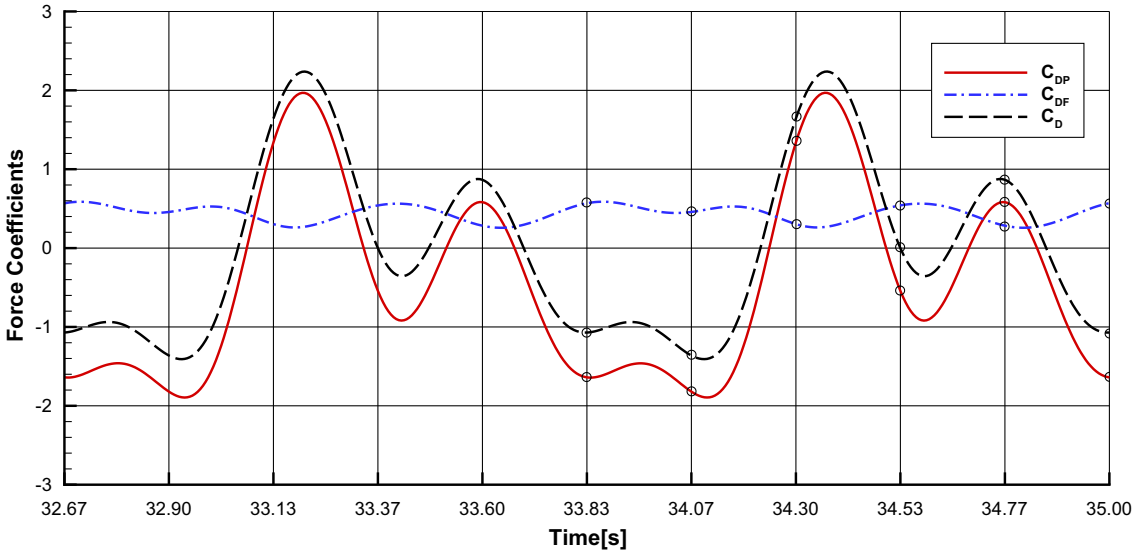


Fig. 16. The time-variation of instantaneous force coefficients acting on a free-swimming hydromedusa *Aequorea victoria* on mesh M2. C_{DP} is the thrust coefficient, C_{DF} is the skin friction drag coefficient and C_D is the total drag coefficient. The symbols "$\circ$" represent the time levels corresponding to Figs. 12 and 13. The average thrust is $\langle C_{DP} \rangle = -0.43$.

accord with the experimental results available in the literature. In addition, the propulsion efficiency of the free-swimming hydromedusa is computed and its value is compared with values from the literature for several other species.

Acknowledgments

This work was partially supported by the National Science Foundation and the Air Force Office of Scientific Research. The authors would like to thank Dr. Sean Colin at Roger Williams University for providing his original video recordings as well as his discussions on the hydromedusa *A. victoria* swimming. The authors would like to thank Dr. Greg Sheard at Monash University for computing the drag coefficient data in Table 2 which is not reported in [46]. The authors gratefully acknowledge the use of the IBM Machine Bluefire at NCAR.

References

[1] H.T. Ahn, G. Carey, An enhanced polygonal finite-volume method for unstructured hyrid-meshes, Int. J. Numer. Meth. Fluids 43 (2007) 29–46.
[2] W.K. Anderson, D.L. Bonhaus, An implicit upwind algorithm for computing turbulent flows on unstructured grids, Comp. Fluids 23 (1994) 1–21.
[3] S. Balay, K. Buschelman, V. Eijkhout, W.D. Gropp, D. Kaushik, M.G. Knepley, L.C. McInnes, B.F. Smith, H. Zhang, PETSc Users Manual. ANL-95/11, Mathematic and Computer Science Division, Argonne National Laboratory, 2004, <http://www-unix.mcs.anl.gov/petsc/petsc-2/>.
[4] T.J. Barth, A 3-D Upwind Euler Solver for Unstructured Meshes, AIAA Paper 91-1548-CP, 1991.
[5] J.T. Batina, Unsteady Euler airfoil solutions using unstructured dynamic meshes, AIAA J. 28 (1990) 1381–1388.
[6] M. Benzi, G.H. Golub, J. Liesen, Numerical solution of saddle point problems, Acta Numer. 14 (2005) 1–137.
[7] T.D. Blacker, S. Benzley, S. Jankovich, R. Kerr, J. Kraftcheck, R. Kerr, P. Knupp, R. Leland, D. Melander, R. Meyers, S. Mitchell, J. Shepard, T. Tautges, D. White, CUBIT Mesh Generation Enviroment Users Manual, vol. 1, Sandia National Laboratories, Albuquerque, NM, 1999.
[8] S. Chattopadhyay, R. Moldovan, C. Yeung, X.L. Wu, Swimming efficiency of bacterium Escherichia coli, Proc. Natl. Acad. Sci. 103 (2006) 13712–13717.
[9] S.P. Colin, J.H. Costello, Reletionship between morpjology and hydrodynamics during swimming by the hyromedusae *A. victoria* and *Aglantha didtale*, Sci. Mar. 60 (1996) 35–42.
[10] S.P. Colin, J.H. Costello, Morphology, swimming performance and propulsive mode of six co-occuring hyromedusae, J. Exp. Biol. 205 (2002) 427–437.
[11] E. Cuthill, J. McKee, Reducing the bandwidth of sparce symmetric matrices, in: Twenty Fourth ACM National Conference, 1969, pp. 157–172.
[12] J.O. Dabiri, S.P. Colin, J.H. Costello, M. Gharib, Flow patterns generated by oblate medusan: field measurements and laboratory analysis, J. Exp. Biol. 208 (2005) 1257–1265.
[13] M. Dai, D.P. Schmidt, Adaptive tetrahedral meshing in free-surface flow, J. Comput. Phy. 208 (2005) 228–252.
[14] R.P. Dwight, Robust mesh deformation using the linear elasticity equations, in: H. Deconinck, E. Dick, (Eds.), Computational Fluid Dynamics, Springer, 2006.
[15] M.D. Ford, J.H. Costello, K.B. Heidelberg, J.E. Purcell, Swimming and feeding by the scypomedusa *Chrysaora quinquecirrha*, Mar. Biol. 129 (1997) 355–362.
[16] Ch. Förster, W.A. Wall, E. Ramm, On the geometric conservation law in transient flow calculations on deforming domains, Int. J. Numer. Meth. Fluids 50 (2005) 1369–1379.
[17] P. Geuzaine, C. Grandmont, C. Farhat, Design and analysis of ALE schemes with provable second-order time-accuracy for inviscid and viscous flow simulations, J. Comput. Phys. 191 (2003) 206–227.
[18] R. Glowinskia, T.-W. Pana, T.I. Heslab, D.D. Josephb, A distributed Lagrange multiplier/fictitious domain method for particulate flows, Int. J. Multiphase Flow 25 (1999) 755–794.
[19] H. Guillard, C. Farhat, On the significance of the geometric conservation law for floow computations on moving meshes, Comput. Meth. Appl. Mech. Eng. 190 (2000) 1467–1482.
[20] C.W. Hirt, A.A. Amsden, J.L. Cook, An arbitrary Lagrangian–Eulerian computing method for all flow speeds, J. Comput. Phys. 14 (1974) 227–253.

[21] M.J. Jensen, Numerical Simulation of Interface Dynamics in Microfluidics, Ph.D. Thesis, Technical University of Denmark, 2005.
[22] A. Johnson, T. Tezduyar, Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces, Comput. Meth. Appl. Mech. Eng. 119 (1994).
[23] A. Johnson, T. Tezduyar, Advanced mesh generation and update methods for 3D flow simulations, Comput. Mech. 23 (1999) 130–143.
[24] G. Karypis, V. Kumar, A fast and high quality multilevel scheme for partitioning irregular graphs, SIAM J. Sci. Comput. 20 (1998) 359–392.
[25] M.H. Kobayashi, J.M.C. Pereira, J.C.F. Pereira, A conservative finite-volume second-order-accurate projection method on hybrid unstructured grids, J. Comput. Phys. 150 (1999) 40–75.
[26] B. Koobus, C. Farhat, Second-order time-accurate and geometricaly conservative implicit schemes for flow computations on unstructured dynamic meshes, Comput. Meth. Appl. Mech. Eng. 170 (1999) 103–129.
[27] M. Krieg, K. Mohseni, Trust characterization of a bio-inspired vortex ring generator for locomotion of underwater robots, IEEE J. Oceanic Eng. 33 (2008) 123–132.
[28] M.J. Lighthill, Mathematical Biofluid Dynamics, SIAM, Philadelphia, 1975.
[29] D.J. Mavriplis, Z. Yang, Constraction of the discrete geometric conservation law for high-order time-accurate simulations on dynamic meshes, J. Comput. Phys. 213 (2006) 557–573.
[30] M.J. McHenry, J. Jed, The ontogenetic scaling of hydrodynamics and swimming performance in jellyfish (*Aurelia aurita*), J. Exp. Biol. 206 (2003) 4125–4137.
[31] R. Mei, J. Xiong, R. Tran-Son-Tay, Motion of a sphere oscillating at low Reynolds numbers in a viscoelastic-fluid-filled cylinderical tube, J. Non-Newtonian Fluid Mech. 66 (1996) 169–192.
[32] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. 37 (2005) 239–261.
[33] K. Mohseni, Statistical equilibrium theory for axisymmetric flows: Kelvin's variational principle and an explanation for the vortex ring pinch-off process, Phys. Fluids 13 (2001) 1924–1931.
[34] K. Mohseni, Pulsatile vortex generators for low-speed maneuvering of small underwater vehicles, Ocean Eng. 33 (2006) 2209–2223.
[35] K. Mohseni, T. Colonius, Numerical treatment of polar coordinate singularities, J. Comput. Phy. 157 (2000) 787–795.
[36] R.G. Owens, T.N. Phillips, Steady viscoelastic flow past a sphere using spectral elements, Int. J. Num. Meth. Eng. 39 (1996) 1517–1534.
[37] C.S. Peskin, The fluid dynamics of heart valves: experimental, theoretical, and computational methods, Ann. Rev. Fluid Mech. 14 (1982) 235–259.
[38] S. Rida, F. McKenty, F.L. Meng, M. Reggio, A staggered control volume scheme for unstructured triangular grids, Int. J. Numer. Meth. Fluids 25 (1997) 697–717.
[39] M. Rozloznik, Saddle point problems, iterative solution and preconditioning: a short overview, in: I. Marek (Ed.), Proceedings of the XVth Summer School Software and Algorithms of Numerical Mathematics, University of West Bohemia, Pilsen, 2003, pp. 97–108.
[40] Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Statist. Comput. 7 (1986) 856–869.
[41] M. Sahin, A preconditioned semi-staggered dilation-free finite volume method for the incompressible Navier–Stokes equations on all-hexahedral elements, Int. J. Numer. Meth. Fluids 49 (2005) 959–974.
[42] M. Sahin, R.G. Owens, A novel fully-implicit finite volume method applied to the lid-driven cavity problem. Part II. Linear stability analysis, Int. J. Numer. Meth. Fluids 42 (2003) 79–88.
[43] M. Sahin, H.J. Wilson, A semi-staggered dilation-free finite volume method for the numerical solution of viscoelastic fluid flows on all-hexahedral elements, J. Non-Newtonian Fluid Mech. 147 (2007) 79–91.
[44] P.R. Schunk, M.A. Heroux, R.R. Rao, T.A. Baer, S.R. Subia, A.C. Sun, Iterative Solvers and Preconditioners for Fully-coupled Finite Element Formulations of Incompressible Fluid Mechanics and Related Transport Problems, SAND2001-3512J, Sandia National Laboratories Albuuquerque, New Mexico, 2001.
[45] M. Sfakiotakis, D.M. Lane, J.B.C. Davies, Review of fish swimming modes for aquatic locomotion, IEEE J. Oceanic Eng. 24 (1999) 237–252.
[46] G.J. Sheard, K. Ryan, Pressure-driven flow past spheres moving in a circular cylinder, J. Fluid Mech. 592 (2007) 233–262. The value of Sheard & Ryan given in Table 2 is not reported in [46].
[47] R.W. Smith, J.A. Wright, An implicit edge-based ALE method for the incompressible Navier–Stokes equations, Int. J. Numer. Meth. Fluids 43 (2003) 253–279.
[48] P.D. Thomas, C.K. Lombard, Geometric conservation law and its application to flow computations on moving grids, AIAA J. 17 (1979) 1030–1037.
[49] G.S. Triantafyllou, M.S. Triantafyllou, M.A. Grosenbaugh, Optimal thrust development in oscillating foils with application to fish propulsion, J. Fluid Struct. 7 (1993) 205–224.
[50] J.A. Wright, R.W. Smith, An edge-based method for the incompressible Navier–Stokes equations on polygonal meshes, J. Comput. Phys. 169 (2001) 24–43.