

# Lawrence Berkeley National Laboratory

## Lawrence Berkeley National Laboratory

### **Title**

Discontinuous Galerkin solution of the Navier-Stokes equations on deformable domains

### **Permalink**

<https://escholarship.org/uc/item/4r26t0md>

### **Author**

Persson, P.-O.

### **Publication Date**

2009-01-29

# Discontinuous Galerkin Solution of the Navier-Stokes Equations on Deformable Domains

P.-O. Persson<sup>a</sup> J. Bonet<sup>b</sup> and J. Peraire<sup>c,\*</sup>

<sup>a</sup>*Department of Mathematics, University of California, Berkeley, Berkeley, CA 94720-3840, USA*

<sup>b</sup>*School of Engineering, Swansea University, Swansea SA2 8PP, UK*

<sup>c</sup>*Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA*

---

## Abstract

We describe a method for computing time-dependent solutions to the compressible Navier-Stokes equations on variable geometries. We introduce a continuous mapping between a fixed reference configuration and the time varying domain. By writing the Navier-Stokes equations as a conservation law for the independent variables in the reference configuration, the complexity introduced by variable geometry is reduced to solving a transformed conservation law in a fixed reference configuration. The spatial discretization is carried out using the Discontinuous Galerkin method on unstructured meshes of triangles, while the time integration is performed using an explicit Runge-Kutta method. For general domain changes, the standard scheme fails to preserve exactly the free-stream solution which leads to some accuracy degradation, especially for low order approximations. This situation is remedied by adding an additional equation for the time evolution of the transformation Jacobian to the original conservation law and correcting for the accumulated metric integration errors. A number of results are shown to illustrate the flexibility of the approach to handle high order approximations on complex geometries.

*Key words:* Discontinuous Galerkin, Deformable domains, Navier-Stokes, Arbitrary Lagrangian-Eulerian, Geometric Conservation

---

## 1. Introduction

There is a growing interest in high-order methods for fluid problems, largely because of their ability to produce highly accurate solutions with minimum numerical dispersion. The Discontinuous Galerkin (DG) method produces stable discretizations of the convective operator for any order discretization. Moreover, it can be used with unstructured meshes of simplices, which appears to be a requirement for real-world complex geometries. In this paper, we present a high order DG formulation for computing high order solutions to problems with variable geometries.

Time varying geometries appear in a number of practical applications such as rotor-stator flows, flapping flight or fluid-structure interactions. In such cases, it is necessary to properly account for the time variation of the solution domain if accurate solutions are to be obtained. For the Navier-Stokes equations, there has been a considerable effort in the development of Arbitrary Lagrangian Eulerian (ALE) methods to deal with

---

\* Corresponding author. Tel.: +1-617-253-1981; Fax.: +1-617-258-5143.  
*Email address:* [peraire@mit.edu](mailto:peraire@mit.edu) (J. Peraire).

these situations [1–6]. A common feature to these efforts is that the discretization of the equations is carried out on a deforming grid and thus the mesh geometry changes over time. It turns out that there are certain conditions that need to be satisfied by the discrete scheme in order to produce stable and accurate answers. The so-called Geometric Conservation Law (GCL), first introduced in [7], which ensures that constant flow solutions are preserved under arbitrary mesh deformations, is claimed to be a requirement to obtain schemes which are both first order spatially accurate [8] and stable in time [9]. While there appears to be some controversy as to the exact role that the satisfaction of the GCL plays in the stability and accuracy of the scheme [10,11], it seems that most practical implementations require the GCL to be satisfied in order to produce computationally acceptable results [3–6]. To our knowledge, only second (or at most third) order accuracy for space and time has been demonstrated for deformable domain applications using these variable metric formulations [12].

A very elegant formulation that has multiple advantages for time varying domains are the space-time DG formulations [13–15]. Space-time DG methods are fully conservative in space and time, are unconditionally stable and arbitrarily high order accurate in time. In addition, they naturally allow for time adaptivity provided suitable space-time meshes can be created, and they automatically satisfy the GCL. The main disadvantage of space-time DG formulations is that they do not allow for explicit time stepping or multi-step implicit procedures. In the purely hyperbolic case, it is possible to devise space time meshing algorithms that exploit causality and can result in rather efficient methods [16,15]. However, for parabolic problems, the advantages of space-time DG methods come at the cost of having to solve for all the unknowns for a given time slab simultaneously, which results in an increased number of degrees of freedom and higher computational cost.

An alternative approach is that presented in [17] using a high order finite difference method. Here, a time varying mapping is constructed between a fixed reference domain and the real time-varying domain geometry [18]. The original conservation law is first transformed to the reference configuration and then solved using a high order compact difference scheme. In this method, the actual computation is carried out on a fixed mesh and the variable domain geometry is accounted for through a modification of the fluxes in the conservation law. This approach is simple and allows for arbitrarily high order solutions to be obtained. Unfortunately, the satisfaction of the GCL is not guaranteed for general grid motions and correction terms are required to ensure preservation of constant solutions and to obtain accurate solutions in practice. One of the disadvantages of the correction terms proposed in [17] is that the resulting formulation is not strictly conservative and therefore its applicability in the presence of strong shocks is unclear.

In this paper, we follow a similar approach to that presented in [17]. That is, the equations are always discretized on a fixed reference domain. The spatial discretization is carried out using the Discontinuous Galerkin method on a mesh of triangles, and the time integration is done with an explicit Runge-Kutta scheme. In order to ensure that constant solutions in the physical domain are preserved exactly, we introduce an additional scalar equation in which the Jacobian of the transformation is integrated numerically using the same spatial and time discretization schemes. This numerically integrated Jacobian is used to correct for integration errors in the conservation equations. It turns out that for continuous mappings, this additional equation can be integrated locally within each element. Therefore, it can be segregated from the global system and dealt with separately, thus incurring a very small computational overhead. Several examples are presented and high orders of convergence are shown. Here, we consider polynomial approximations up to  $p = 5$  and fourth order time integration. We point out that for our algorithm, the satisfaction of the GCL is not a necessary condition for stability or accuracy. We have found that enforcing the satisfaction of the GCL improves the accuracy of the results, especially for low order approximations and long time integrations, but we also present some computations carried out with our basic scheme which are stable, optimally convergent and the algorithm employed fails to satisfy the GCL. Although the method presented can be used in 3D without changes, in this paper, we only consider 2D examples. For the examples considered, the mapping is given explicitly in terms of the boundary motion. We use blending functions to propagate the boundary movement into the domain. The motion of the boundary can be prescribed externally or determined using additional equations which relate it to the flow solution.

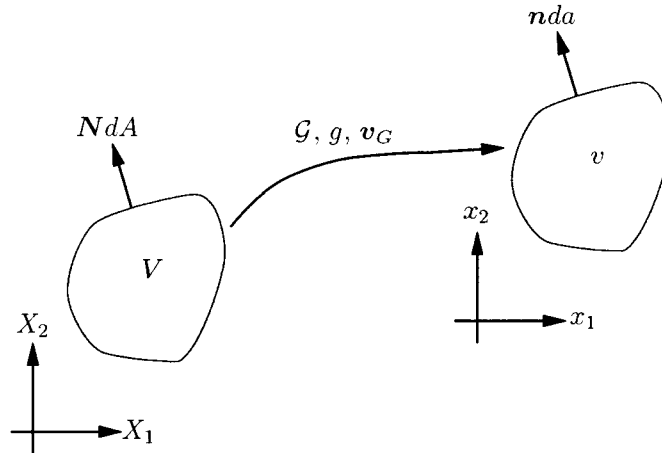


Fig. 1. Mapping between the physical and the reference domains.

## 2. ALE Formulation

When simulating problems involving time varying domains, the motion of the mesh must be accounted for in the solution process. We use an Arbitrary Lagrangian Eulerian (ALE) formulation, which allows us to move and deform the domains while preserving the high order accuracy. An important ingredient for our formulation is a time dependent mapping between a fixed reference domain and the physical deformable domain.

### 2.1. The Mapping

Let the physical domain of interest be denoted by  $v(t)$  and the fixed reference configuration be denoted by  $V$  (see figure 1). Also, let  $\mathbf{N}$  and  $\mathbf{n}$  be the outward unit normals in  $V$  and  $v(t)$ , respectively. We assume, for each time  $t$ , the existence of a smooth one-to-one time dependent mapping given by a diffeomorphism,  $\mathcal{G}(\mathbf{X}, t)$ , between  $V$  and  $v(t)$ . Thus a point  $\mathbf{X}$  in  $V$ , is mapped to a point  $\mathbf{x}(t)$  in  $v(t)$ , which is given by  $\mathbf{x} = \mathcal{G}(\mathbf{X}, t)$ . In addition, we assume that for all  $\mathbf{X}$ ,  $\mathbf{x} = \mathcal{G}(\mathbf{X}, t)$  is a smooth differentiable function of  $t$ . In order to transform the Navier-Stokes equations from the physical  $(\mathbf{x}, t)$  domain to the reference  $(\mathbf{X}, t)$  domain, we require some differential properties of the mapping. To this end, we introduce the mapping deformation gradient  $\mathbf{G}$  and the mapping velocity  $\mathbf{v}_G$  as

$$\mathbf{G} = \nabla_{\mathbf{X}} \mathcal{G}, \quad \mathbf{v}_G = \left. \frac{\partial \mathcal{G}}{\partial t} \right|_{\mathbf{X}}. \quad (1)$$

In addition, we denote the Jacobian of the mapping by  $g = \det(\mathbf{G})$ . We note that corresponding infinitesimal vectors  $d\mathbf{L}$  in  $V$  and  $d\mathbf{l}$  in  $v(t)$  are related by  $d\mathbf{l} = \mathbf{G}d\mathbf{L}$ . Also, the elemental volumes are related by  $dv = g dV$ . From this, we can derive an expression for the area change. Let  $d\mathbf{A} = \mathbf{N}dA$  denote an area element which after deformation becomes  $d\mathbf{a} = \mathbf{n}da$ . We then have that,  $dV = d\mathbf{L} \cdot d\mathbf{A}$  and  $dv = d\mathbf{l} \cdot d\mathbf{a}$ . Therefore, we must have that

$$\mathbf{n} da = g \mathbf{G}^{-T} \mathbf{N} dA, \quad \text{and} \quad \mathbf{N} dA = g^{-1} \mathbf{G}^T \mathbf{n} da. \quad (2)$$

### 2.2. Transformed Equations

As a starting point, we consider the compressible Navier-Stokes equations in the physical domain  $(\mathbf{x}, t)$  are written as a system of conservation laws

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U}, \nabla \mathbf{U}) = 0, \quad (3)$$

where  $\mathbf{U}$  is the vector of conserved variables and  $\mathbf{F}$  is a generalized column flux vector which components are the physical flux vectors in each of the spatial coordinate directions. Here  $\mathbf{F}$  incorporates both inviscid and viscous contributions. That is,  $\mathbf{F} = \mathbf{F}^{inv}(\mathbf{U}) + \mathbf{F}^{vis}(\mathbf{U}, \nabla \mathbf{U})$  and  $\nabla$  represents the spatial gradient operator in the  $\mathbf{x}$  variables. The detailed expressions for the flux vectors in the Navier-Stokes equations can be found in [18].

In order to obtain the corresponding conservation law written in the reference configuration we re-write the above equation in an integral form as

$$\int_{v(t)} \frac{\partial \mathbf{U}}{\partial t} dv + \int_{\partial v} \mathbf{F} \cdot \mathbf{n} da = \mathbf{0} \quad (4)$$

Note that the above expression follows directly from (3) by integrating over  $v(t)$  and applying the divergence theorem. It is now possible to utilize the mapping and evaluate these integrals in the reference configuration. Consider first the second term,

$$\int_{\partial v} \mathbf{F} \cdot \mathbf{n} da = \int_{\partial V} \mathbf{F} \cdot (g\mathbf{G}^{-T}\mathbf{N}) dA = \int_{\partial V} (g\mathbf{G}^{-1}\mathbf{F}) \cdot \mathbf{N} dA \quad (5)$$

Similarly, the first integral is transformed by means of Reynolds transport theorem to give,

$$\int_{v(t)} \frac{\partial \mathbf{U}}{\partial t} dv = \frac{d}{dt} \int_{v(t)} \mathbf{U} dv - \int_{\partial v} (\mathbf{U}\mathbf{v}_G) \cdot \mathbf{n} da \quad (6)$$

$$= \frac{d}{dt} \int_V g^{-1}\mathbf{U} dV - \int_{\partial V} (\mathbf{U}\mathbf{v}_G) \cdot (g\mathbf{G}^{-T}\mathbf{N}) dA \quad (7)$$

$$= \int_V \frac{\partial(g^{-1}\mathbf{U})}{\partial t} \Big|_X dV - \int_{\partial V} (g\mathbf{U}\mathbf{G}^{-1}\mathbf{v}_G) \cdot \mathbf{N} dA \quad (8)$$

Using the divergence theorem once again enables an equivalent local conservation law in the reference domain to be derived as,

$$\frac{\partial \mathbf{U}_X}{\partial t} \Big|_X + \nabla_X \cdot \mathbf{F}_X(\mathbf{U}_X, \nabla_X \mathbf{U}_X) = 0, \quad (9)$$

where the time derivative is at a constant  $\mathbf{X}$  and the spatial derivatives are taken with respect to the  $\mathbf{X}$  variables. The transformed vector of conserved quantities and corresponding fluxes in the reference space are,

$$\mathbf{U}_X = g\mathbf{U}, \quad \mathbf{F}_X = g\mathbf{G}^{-1}\mathbf{F} - \mathbf{U}_X\mathbf{G}^{-1}\mathbf{v}_G, \quad (10)$$

or, more explicitly,

$$\mathbf{F}_X = \mathbf{F}_X^{inv} + \mathbf{F}_X^{vis}, \quad \mathbf{F}_X^{inv} = g\mathbf{G}^{-1}\mathbf{F}^{inv} - \mathbf{U}_X\mathbf{G}^{-1}\mathbf{v}_G, \quad \mathbf{F}_X^{vis} = g\mathbf{G}^{-1}\mathbf{F}^{vis}, \quad (11)$$

and by simple chain rule,

$$\nabla \mathbf{U} = \nabla_X(g^{-1}\mathbf{U}_X)\mathbf{G}^{-T} = (g^{-1}\nabla_X\mathbf{U}_X - \mathbf{U}_X\nabla_X(g^{-1}))\mathbf{G}^{-T}. \quad (12)$$

### 3. Numerical Methods

In order to develop a Discontinuous Galerkin method, we rewrite the above equations as (9) as a system of first order equations

$$\frac{\partial \mathbf{U}_X}{\partial t} \Big|_X + \nabla_X \cdot \mathbf{F}_X(\mathbf{U}_X, \mathbf{Q}_X) = 0 \quad (13)$$

$$\mathbf{Q}_X - \nabla_X \mathbf{U}_X = 0 \quad (14)$$

Next, we introduce the ‘broken’ DG spaces  $\mathcal{V}^h$  and  $\Sigma^h$  associated with the triangulation  $\mathcal{T}^h = \{K\}$  of  $V$ . In particular,  $\mathcal{V}^h$  and  $\Sigma^h$  denote the spaces of functions whose restriction to each element  $K$  are polynomials of order  $p \geq 1$ .

Following [19], we consider DG formulations of the form: find  $\mathbf{U}_X^h \in \mathcal{V}^h$  and  $\mathbf{Q}_X^h \in \Sigma^h$  such that for all  $K \in \mathcal{T}^h$ , we have

$$\int_K \left. \frac{\partial \mathbf{U}_X^h}{\partial t} \right|_X \cdot \mathbf{V} \, dV - \int_K \mathbf{F}_X(\mathbf{U}_X^h, \mathbf{Q}_X^h) \cdot \nabla_X \mathbf{V} \, dV - \int_{\partial K} \mathbf{V} (\hat{\mathbf{F}}_X \cdot \mathbf{N}) \, dA = 0 \quad \forall \mathbf{V} \in \mathcal{V}^h \quad (15)$$

$$\int_K \mathbf{Q}_X^h \cdot \mathbf{P} \, dV + \int_K \mathbf{U}_X^h \cdot \nabla_X \cdot \mathbf{V} \, dV - \int_{\partial K} \hat{\mathbf{U}}_X^h (\mathbf{P} \cdot \mathbf{N}) \, dA = 0 \quad \forall \mathbf{P} \in \Sigma^h \quad (16)$$

Here, the numerical fluxes  $\hat{\mathbf{F}}_X \cdot \mathbf{N}$  and  $\hat{\mathbf{U}}_X$  are approximations to  $\mathbf{F}_X \cdot \mathbf{N}$  and to  $\mathbf{U}_X$ , respectively, on the boundary of the element  $K$ . The DG formulation is complete once we specify the numerical fluxes  $\hat{\mathbf{F}}_X \cdot \mathbf{N}$  and  $\hat{\mathbf{U}}_X$  in terms of  $(\mathbf{U}_X^h)$  and  $(\mathbf{Q}_X^h)$  and the boundary conditions. The flux term  $\hat{\mathbf{F}}_X \cdot \mathbf{N}$  is decomposed into its inviscid and viscous parts,

$$\hat{\mathbf{F}}_X \cdot \mathbf{N} = \hat{\mathbf{F}}_N^{\text{inv}}(\mathbf{U}_X^h) + \hat{\mathbf{F}}_N^{\text{vis}}(\mathbf{U}_X^h, \mathbf{Q}_X^h) \quad (17)$$

The numerical fluxes  $\hat{\mathbf{F}}_N^{\text{vis}}$  and  $\hat{\mathbf{U}}_X$  are chosen according to the Compact Discontinuous Galerkin (CDG) method [20]. This is a variant of the Local Discontinuous Galerkin (LDG) method [19] but has the advantage of being compact on general unstructured meshes.

The inviscid numerical flux  $\hat{\mathbf{F}}_N^{\text{inv}}(\mathbf{U}_X^h)$  is chosen according to the method proposed by Roe [21]. Note that this flux can be very easily derived from the standard Eulerian Roe fluxes by noting that the flux  $\mathbf{F}_X^{\text{inv}} \cdot \mathbf{N}$  can be written as

$$\mathbf{F}_X^{\text{inv}} \cdot \mathbf{N} = (\mathbf{F}^{\text{inv}} - \mathbf{U} \mathbf{v}_G) \cdot g \mathbf{G}^{-T} \mathbf{N} ,$$

where  $g \mathbf{G}^{-T} \mathbf{N}$  (from (2)) is always continuous across the interface (assuming that  $\mathcal{G}$  is continuous), and the eigenvalues and eigenvectors of the Jacobian matrix for  $\mathbf{F} - \mathbf{U} \mathbf{v}_G$  are trivially obtained from the Jacobian matrix for the standard Eulerian flux  $\mathbf{F}$ .

Time integration is performed explicitly using a fourth order Runge-Kutta scheme. Since we are largely interested in convection dominated flows, the time step required for stability is determined by the inviscid fluxes. When the deformation of the domain is prescribed, we chose the time step  $\Delta t$  such that  $\Delta t \lambda_{\text{max}} / h_X < C$ . Here,  $h_X$  is the characteristic mesh size,  $C$  is a constant that depends on the integration algorithm and the polynomial approximation order, and  $\lambda_{\text{max}}$  is the maximum eigenvalue of the Jacobian matrix  $d(\mathbf{F}_X^{\text{inv}} \cdot \mathbf{N}) / d\mathbf{U}_X$ , for any unit normal direction  $\mathbf{N}$ . We have found that for low order polynomial approximations, up to  $p = 5$  and for equally spaced nodes,  $C$  can be taken as  $2\sqrt{2}/p$ . It is well known that high order approximation the largest eigenvalue of the convective operator scales like  $p^2$ , but for low  $p$ , a linear approximation is found to work well in practice. The largest eigenvalue of  $d(\mathbf{F}_X^{\text{inv}} \cdot \mathbf{N}) / d\mathbf{U}_X$  can then be found explicitly and is given by  $\lambda_{\text{max}} = \alpha [(\mathbf{u} - \mathbf{v}_G) \cdot \mathbf{n} + c]$ . Here,  $\mathbf{n}$  is the mapped unit normal,  $\mathbf{u}$  is the fluid velocity and  $c$  is the speed of sound. Also,  $\alpha = dL/dl$  is a parameter that depends on the local mapping and relates distances in the directions of  $\mathbf{N}$  and  $\mathbf{n}$ , respectively. Compared to the non-deformable domain case, we see that the mapping modifies the allowable time step through the mapping velocity  $\mathbf{v}_G$  and through the parameter  $\alpha$ .

### 3.1. Geometric Conservation Law

It turns out that, for arbitrary mappings, a constant solution in the physical domain is not necessarily a solution of the discretized equations in the reference domain. Even though this error is typically very small for high order discretizations, the situation is quite severe for lower order approximations since the free-stream condition is not preserved identically. Satisfaction of the constant solution is often referred to as the Geometric Conservation Law (GCL) and is was originally discussed in [7]. The source of the problem is the inexact integration of the Jacobian  $g$  of the transformation by the numerical scheme.

First, we note that using expressions (2) together with the divergence theorem, it is straightforward to prove the so-called Piola relationships, which hold for arbitrary vectors  $\mathbf{W}$  and  $\mathbf{w}$ :

$$\nabla_X \cdot \mathbf{W} = g \nabla \cdot (g^{-1} \mathbf{G} \mathbf{W}), \quad \nabla \mathbf{w} = g^{-1} \nabla_X \cdot (g \mathbf{G}^{-1} \mathbf{w}).$$

When the solution  $\mathbf{U}$  is constant, say  $\bar{\mathbf{U}}$ , we have

$$\nabla_X \cdot \mathbf{F}_X = g \nabla \cdot (\mathbf{F} - \bar{\mathbf{U}} \mathbf{v}_G) = -g \bar{\mathbf{U}} \nabla \cdot \mathbf{v}_G = -[\nabla_X \cdot (g \mathbf{G}^{-1} \mathbf{v}_G)] \bar{\mathbf{U}}.$$

Therefore, for a constant solution  $\bar{\mathbf{U}}$ , equation (9) becomes

$$\left. \frac{\partial \mathbf{U}_X}{\partial t} \right|_X + \nabla_X \cdot \mathbf{F}_X = g^{-1} \bar{\mathbf{U}}_x \left( \left. \frac{\partial g}{\partial t} \right|_X - \nabla_X \cdot (g \mathbf{G}^{-1} \mathbf{v}_G) \right).$$

We see that the right hand side is only zero if the equation for the time evolution of the transformation Jacobian  $g$

$$\left. \frac{\partial g}{\partial t} \right|_X - \nabla_X \cdot (g \mathbf{G}^{-1} \mathbf{v}_G) = 0,$$

is integrated exactly by our numerical scheme. Since in general, this will not be the case, the constant solution  $\bar{\mathbf{U}}_x$  in the physical space will not be preserved exactly.

An analogous problem was brought up in the formulation presented in [17]. The solution proposed there was to add some corrections aimed at canceling the time integration errors. These terms are non-conservative and probably not suitable in the presence of strong shocks in the solutions. Here, we use a different approach. The system of conservation laws (9) is replaced by

$$\left. \frac{\partial (\bar{g} g^{-1} \mathbf{U}_X)}{\partial t} \right|_X - \nabla_X \cdot \mathbf{F}_X = \mathbf{0} \quad (18)$$

where  $\bar{g}$  is obtained by solving the following equation using the same numerical time integration scheme as for the remaining equations

$$\left. \frac{\partial \bar{g}}{\partial t} \right|_X - \nabla_X \cdot (g \mathbf{G}^{-1} \mathbf{v}_G) = 0. \quad (19)$$

We note that even though  $\bar{g}$  is an approximation to  $g$ , when the above equation is solved numerically its value will differ from that of  $g$  due to integration errors. It is straightforward to verify that (18) does indeed preserve a constant solution as desired. Finally, we point out that the fluxes in equation (19) do not depend on  $\mathbf{U}$  and are only a function of the mapping. These has two important implications. First, when the mapping is prescribed, equation (19) can be integrated independently to obtain  $\bar{g}$  in time. Then,  $\bar{g}$  in equation (18) has the effect of *modifying* the externally prescribed mapping. Since we expect  $\bar{g}$  and  $g$  to be close, the stability limit of equation (18) is the same as that of equation (9). Note that equation (19) has no stability limit since the fluxes do not depend on  $\bar{g}$ . Second, although we choose in practice to integrate (19) and (18) simultaneously, this incurs little overhead since the evaluation of the fluxes in (19) does not require communication with the neighboring elements.

### 3.2. Mappings for Deformable Domains

In order to solve problems on moving domains, we need a procedure to define a mapping  $\mathbf{x} = \mathcal{G}(\mathbf{X}, t)$  from the reference domain to the physical domain. One method that is commonly used in ALE simulations is to force the motion of the mesh nodes on the boundary and apply a mesh smoothing scheme to reconstruct the mapping in the domain interior. Another technique is to solve additional equations (usually elliptic) for  $\mathbf{x}$  in the interior. For some of these algorithms, it may not be obvious to obtain accurate values for the mapping derivatives as required by our fluxes.

In this paper, we use a different approach which produces explicit expressions for the mappings. These are differentiated analytically with respect to space and time to obtain the deformation gradient and the grid velocity. The procedure uses polynomial *blending functions*  $r_n(x)$  of odd degree  $n$ , with  $r(0) = 0$ ,  $r(1) = 1$ ,

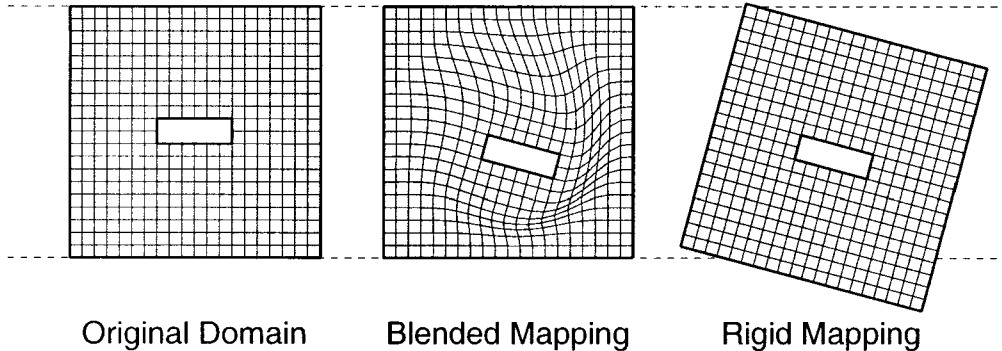


Fig. 2. Deformation by blending of the original domain and a rigidly displaced domain.

and with  $(n - 1)/2$  vanishing derivatives at  $x = 0$  and  $x = 1$ . For degree  $n = 3$  this blending polynomial equals  $r_3(x) = 3x^2 - 2x^3$ , and for  $n = 5$  it equals  $r_5(x) = 10x^3 - 15x^4 + 6x^5$ .

An example is shown Figure 2, where a square domain with a rectangular hole is deformed in such a way that the hole is displaced and rotated but the outer boundary is fixed. The mapping is defined by introducing a circle  $C$  centered at  $\mathbf{X}_C$  with a radius  $R_C$  that contains the moving boundary. The distance from a point  $\mathbf{X}$  to  $C$  is then  $d(\mathbf{X}) = \|\mathbf{X} - \mathbf{X}_C\| - R_C$ , where  $\|\cdot\|$  is the Euclidean length function.

Next, we compute a blending function in terms of the distance  $d(\mathbf{X})$ ,

$$b(d) = \begin{cases} 0, & \text{if } d < 0 \\ 1, & \text{if } d > D \\ r(d/D), & \text{otherwise.} \end{cases} \quad (20)$$

where  $D$  is chosen such that all points at a distance  $d(\mathbf{X}) \leq D$  are completely inside the domain. The mapping  $\mathbf{x} = \mathcal{G}(\mathbf{X}, t)$  is a blended combination of the undeformed domain and a rigidly displaced domain  $\mathbf{Y}(\mathbf{X})$ :

$$\mathbf{x} = b(d(\mathbf{x}))\mathbf{X} + (1 - b(d(\mathbf{x})))\mathbf{Y}(\mathbf{X}), \quad (21)$$

This expression will ensure that all points inside  $C$  will be mapped according to the rigid motion, that all points a distance  $D$  or larger from the circle will be unchanged, and that the points in-between will be mapped smoothly. Time-dependent mappings are treated in a straight-forward way using the same technique, and multiple boundaries moving independently are handled by introducing one blending function for each boundary. We obtain the mapping velocity  $\mathbf{v}_G$  and the deformation gradient  $\mathbf{G}$  by analytical differentiation of (21).

This approach clearly has some limitations, but for many cases it is remarkably simple and effective. It has the advantage that the mapping derivatives are easily obtainable since the mapping is given in explicit form. It is clearly that the regularity of the mapping must play an important role in the overall accuracy of the scheme. Here we use quintic  $r_5(x)$  polynomials in all our simulations. We expect that in general continuous piecewise polynomially defined mappings will not degrade the accuracy, provided any discontinuities in the derivatives are concentrated at the element boundaries.

### 3.3. Calculation of Flow Properties in Mapped Space

For some of the problems considered, it will be required to calculate integrals of boundary fluxes in physical space. This situation occurs, for instance, when we need to calculate the lift and drag forces on an object in which case we need to evaluate a boundary integral of the momentum fluxes over the boundary of the object. If we need to compute the integral of the physical fluxes over a curve  $c$  which maps to a curve  $C$  in the reference domain, the corresponding expression in the reference domain can be obtained as follows,



$$\int_c \mathbf{F} \cdot \mathbf{n} da = \int_C \mathbf{F} \cdot g \mathbf{G}^{-T} \mathbf{N} dA = \int_C (\mathbf{F}_X + \mathbf{U}_X \mathbf{G}^{-1} \mathbf{v}_G) \cdot \mathbf{N} dA, \quad (22)$$

where we have used expressions (2) and the definition of the fluxes.

## 4. Results

### 4.1. Euler Vortex with Variable Mapping

To validate our solver and show that we retain the high-order accuracy after the domain mapping, we solve an inviscid model problem consisting of a compressible vortex in a rectangular domain [22,23]. The vortex is initially centered at  $(x_0, y_0)$  and is moving with the free-stream at an angle  $\theta$  with respect to the  $x$ -axis. The analytic solution at  $(x, y, t)$  is given by

$$u = u_\infty \left( \cos \theta - \frac{\epsilon((y - y_0) - \bar{v}t)}{2\pi r_c} \exp\left(\frac{f(x, y, t)}{2}\right) \right) \quad (23)$$

$$v = u_\infty \left( \sin \theta + \frac{\epsilon((x - x_0) - \bar{u}t)}{2\pi r_c} \exp\left(\frac{f(x, y, t)}{2}\right) \right) \quad (24)$$

$$\rho = \rho_\infty \left( 1 - \frac{\epsilon^2(\gamma - 1)M_\infty^2}{8\pi^2} \exp(f(x, y, t)) \right)^{\frac{1}{\gamma-1}} \quad (25)$$

$$p = p_\infty \left( 1 - \frac{\epsilon^2(\gamma - 1)M_\infty^2}{8\pi^2} \exp(f(x, y, t)) \right)^{\frac{\gamma}{\gamma-1}} \quad (26)$$

where  $f(x, y, t) = (1 - ((x - x_0) - \bar{u}t)^2 - ((y - y_0) - \bar{v}t)^2)/r_c^2$ ,  $M_\infty$  is the Mach number,  $\gamma = c_p/c_v$ , and  $u_\infty, p_\infty, \rho_\infty$  are free-stream velocity, pressure, and density. The Cartesian components of the free-stream velocity are  $\bar{u} = u_\infty \cos \theta$  and  $\bar{v} = u_\infty \sin \theta$ . The parameter  $\epsilon$  measures the strength of the vortex and  $r_c$  is its size.

We use a domain of size 20-by-15, with the vortex initially centered at  $(x_0, y_0) = (5, 5)$  with respect to the lower-left corner. The Mach number is  $M_\infty = 0.5$ , the angle  $\theta = \arctan 1/2$ , and the vortex has the parameters  $\epsilon = 0.3$  and  $r_c = 1.5$ . We use periodic boundary conditions and integrate until time  $t_0 = \sqrt{10^2 + 5^2}$ , when the vortex has moved a relative distance of (10, 5).

We solve the Euler equations on a regular rectangular mesh. The domain is mapped according to the following expressions:

$$x_1(X_1, X_2, t) = X_1 + 2.0 \sin(\pi X_1/10) \sin(\pi X_2/7.5) \sin(2\pi t/t_0) \quad (27)$$

$$x_2(X_1, X_2, t) = X_2 + 1.5 \sin(\pi X_1/10) \sin(\pi X_2/7.5) \sin(4\pi t/t_0) \quad (28)$$

We note that at times  $t = 0$  and  $t = t_0$ , the mapping is the identity mapping which makes it straightforward to initialize and compare solutions. The solution and the deformed meshes at time  $t = 0$  and  $t = (3/8)t_0$  are shown in figure 3.

We solve for a variety of mesh sizes and polynomial orders, and measure the error in the  $L_2$ -norm. Figure 4 shows the convergence of the scheme when solving without mapping (equation (9), but setting  $\mathbf{x} = \mathbf{X}$ ), using a mapping with GCL correction (equations (18) and (19)), and using a mapping without the GCL correction (equation (9)). We note that we obtain optimal convergence  $O(h^{p+1})$  in all cases, and that the mapped cases have at most a factor of 8 larger error than the unmapped. The computation using the mapping is of course expected to be less accurate, since this particular mapping is fairly extreme and generates large variations in the resolution of the vortex. We also note that the use of the GCL correction is not noticeable except for low order approximations.

### 4.2. Free-stream Preservation

The above test has been performed with the modified scheme described above that ensures exact conservation of the free stream condition. To illustrate the effect of this modification, we solve using the same

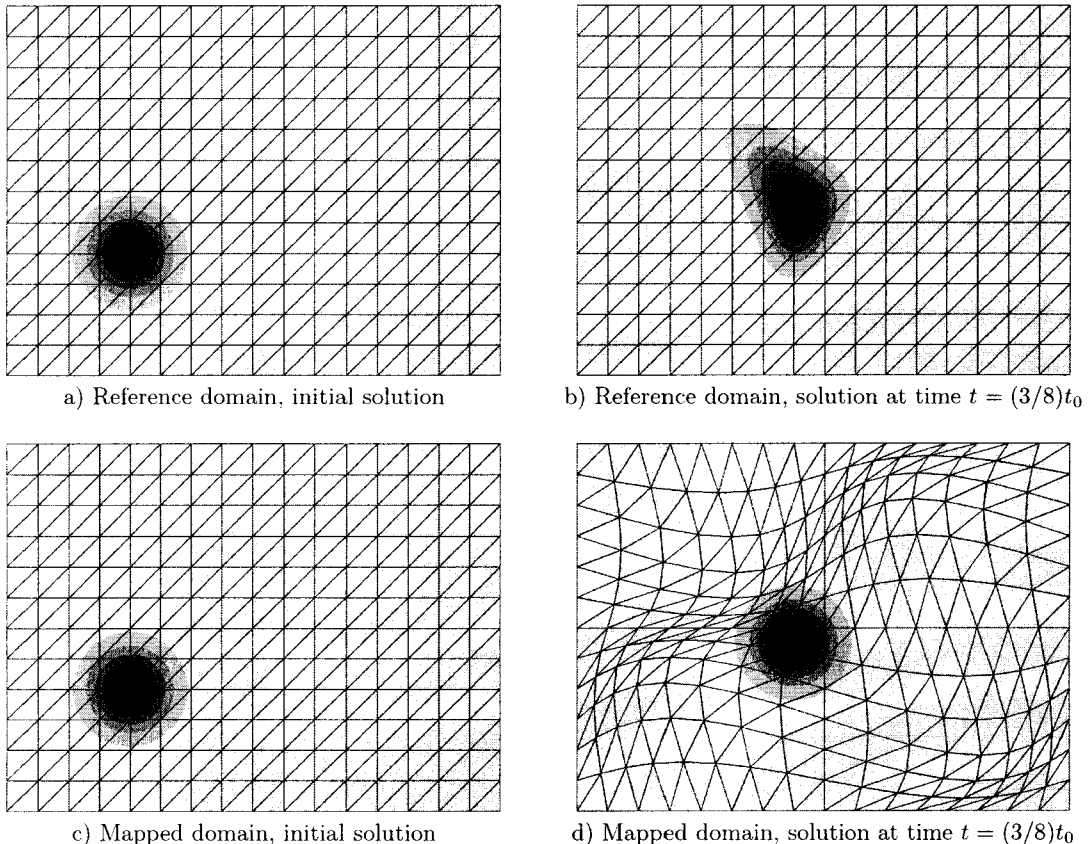


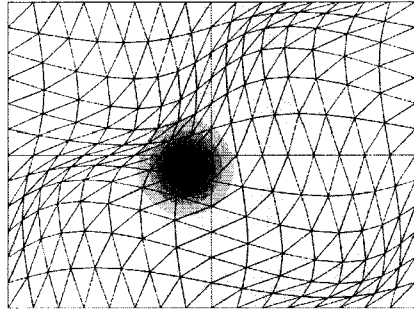
Fig. 3. The solutions at times  $t = 0$  and  $t = (3/8)t_0$ , plotted both in the reference domain and in the mapped domain. Note that the mapped mesh is used *only for visualization*, the calculations are done in the fixed reference space using mapped equations.

grid and time dependent mapping as in the previous example, but with uniform free-stream condition as the initial condition. We integrate in time until  $t = 1.0$  with and without the GCL correction, and plot the  $L_2$ -errors in figure 5.

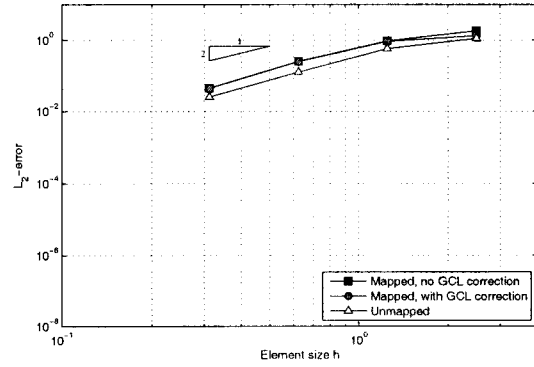
Although the error without the correction converges to zero as  $h \rightarrow 0$ , we note that it can be quite large, especially if the solution is not well-resolved. With the correction the errors are down to machine zero. We point out that in order to obtain these small errors when the GCL correction is used, it is necessary to use spatial integration rules for the non-linear fluxes in equations (15) and (16) which are sufficiently high. In order to obtain machine precision errors in when the GCL correction is used in this example, we had to employ integration rules which are exact for polynomials of order  $6p$  when the solution vector is interpolated using a polynomials of order  $p$ . For all the other results reported in this paper (including those that employ the GCL correction) we use using integration rules which are capable of integrating exactly only a polynomial of order  $3p$  when the solution vector is interpolated using polynomials of order  $p$ . We have found no noticeable differences in the numerical results when higher rules are employed.

#### 4.3. Flow around Oscillating Cylinder

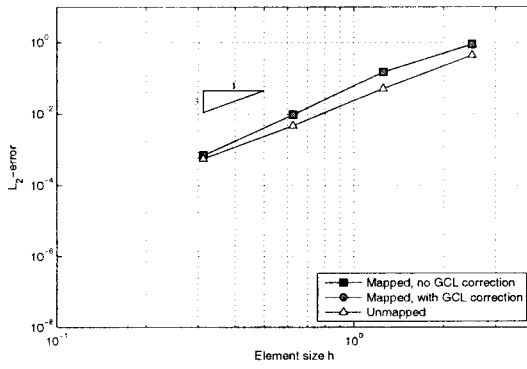
The main purpose of this example is to qualitatively validate the proposed blending approach for mesh motion. For this purpose, we consider the two dimensional viscous flow around an oscillating cylinder of radius 1. The Reynolds number with respect to the diameter is 400 and the Mach number is 0.2. The



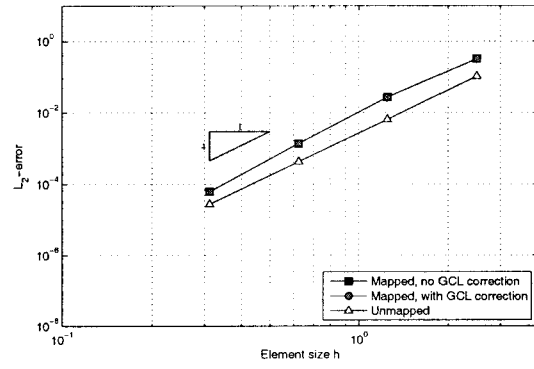
a) The Euler vortex problem,  $h = 1.25$ ,  $p = 4$



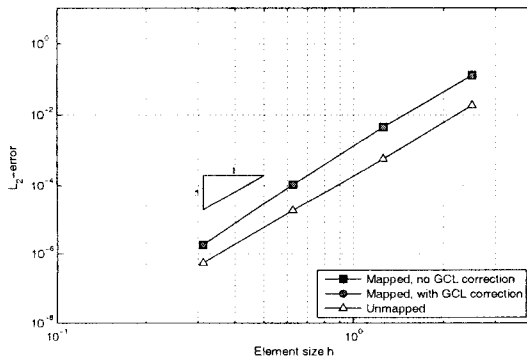
b) Polynomial order  $p = 1$



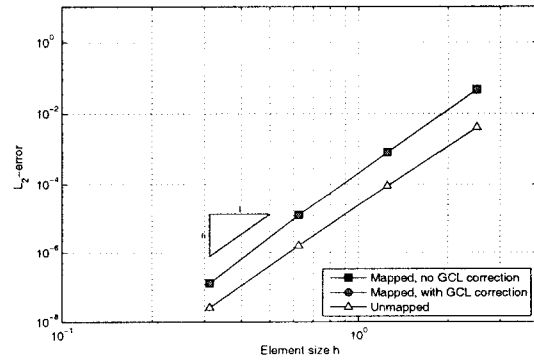
c) Polynomial order  $p = 2$



d) Polynomial order  $p = 3$



e) Polynomial order  $p = 4$



f) Polynomial order  $p = 5$

Fig. 4. The convergence of the Euler vortex problem, using the mapped scheme with and without GCL correction, as well as the unmapped scheme. Four mesh sizes are used and polynomial orders  $p = 1$  to  $p = 5$ . The optimal convergence  $h^{p+1}$  is achieved in all the cases, although the constant is larger for the mapped cases because of the non-uniform resolution of the vortex.

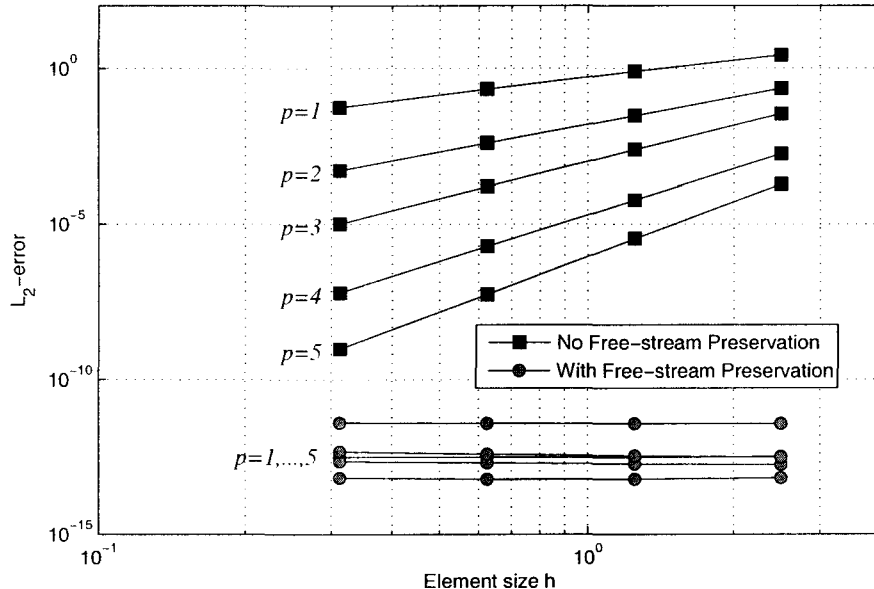


Fig. 5. The convergence of a pure free-stream problem with and without the free-stream preservation technique.

$y$ -displacement of the cylinder is given by

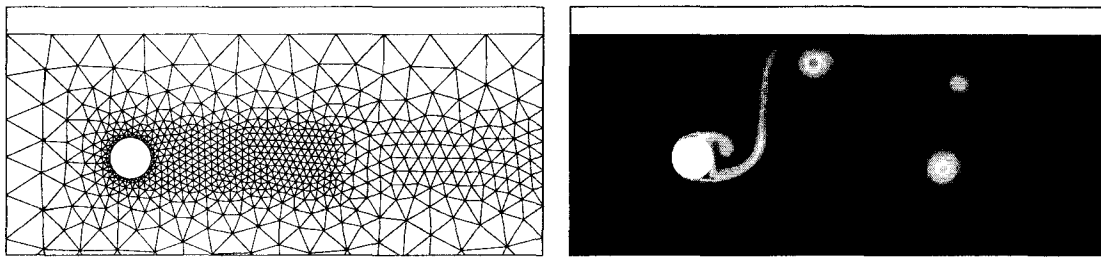
$$y_c(t) = A \sin(2\pi ft) \quad (29)$$

where  $A = 4/3$  and  $f = 0.1$ . Our unstructured triangular mesh consists of 1316 elements and we use polynomials of degree  $p = 4$  within each element.

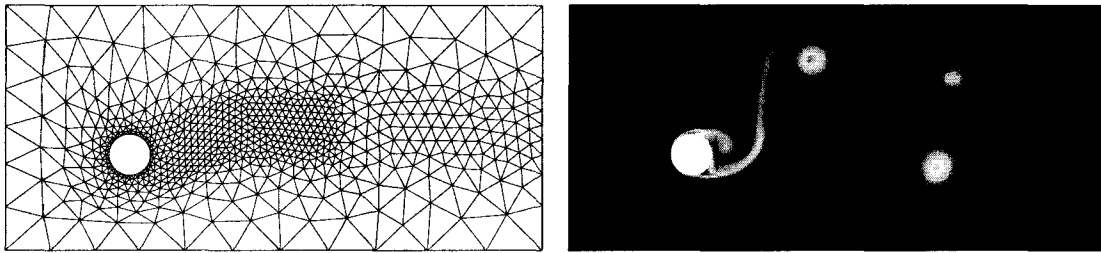
We integrate for seven periods (until  $t = 70$ ) using two mappings – rigid vertical motion and a smooth blending between a displacement of the circle and a fixed mapping. For the blended mapping we run the simulation with and without the GCL correction. The mapped meshes and the solutions using the three techniques are shown in figure 6 at time  $t = 17.5$ , together with the time evolution of the lift and the drag coefficients. In the cases of the rigid mapping and the blended mapping without GCL correction, equations (9) are solved directly. Since the vector of conserved variables has four components this results in a total of 78,960 degrees of freedom. For the non-rigid mapping with GCL correction, equations (18) and (19) are solved. In this case, the total number of degrees of freedom is 98,700. In all cases, we solve with the same time step size for a total of 70,000 time steps. Although a numerical quantification of the error is hard because of the large sensitivity of the exact solution to small changes in the data, the results are remarkably similar after a considerably large time integration interval. We note however, that some small oscillations can be seen in the free stream for the blended mapping without GCL correction case (figure 6(c)).

#### 4.4. Locomotion of a Free Oscillating Plate

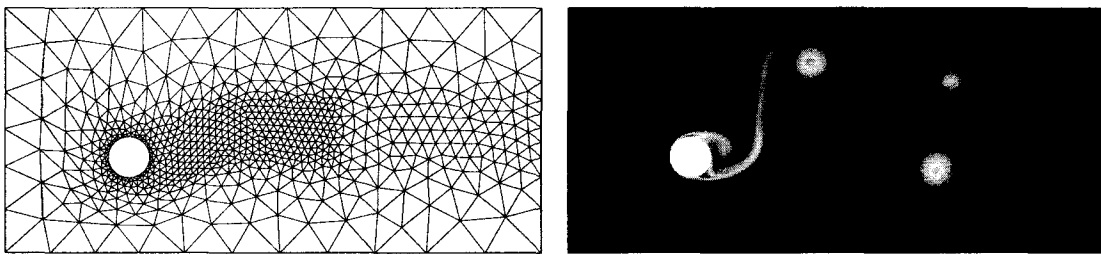
Next, we solve for the viscous flow around a vertically oscillating plate which is free to move in the horizontal direction. It has been shown experimentally [24] and explained using computational simulations [25] that in certain flow regimes the flow loses symmetry and pushes the body into locomotion. Here, we use our techniques to simulate a problem similar to the slender body case in [25], which was solved using a mixed Fourier/finite-difference method. By using a discontinuous Galerkin formulation we obtain several advantages: 1) The domain is discretized into an unstructured triangular mesh which is locally refined in the vicinity of the plate to obtain better computational efficiency, 2) The solutions are high order accurate even for a realistic thin plate geometry with sharp corners, and 3) The natural stabilization in the DG method



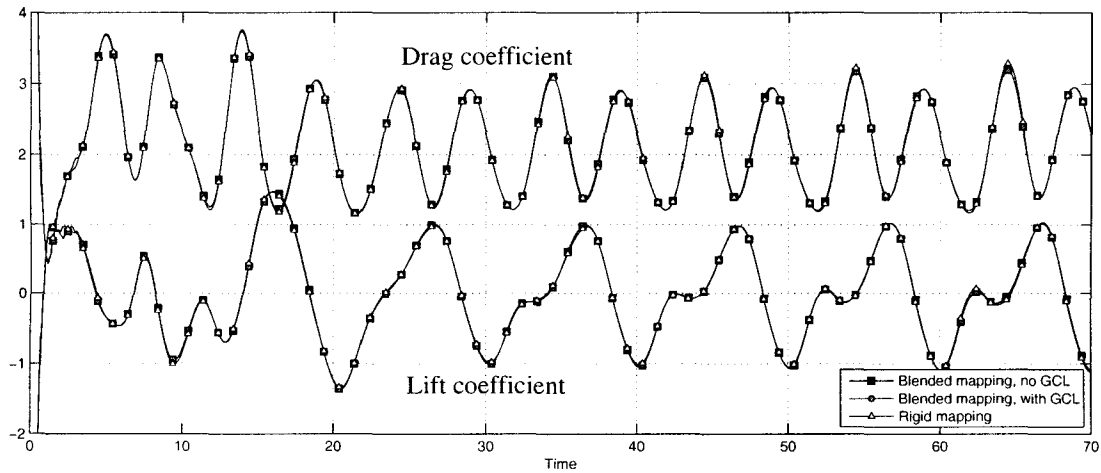
a) Rigid mapping, mapped mesh and entropy plot, time  $t = 17.5$



b) Blended mapping, with GCL correction, mapped mesh and entropy plot, time  $t = 17.5$



c) Blended mapping, no GCL correction, mapped mesh and entropy plot, time  $t = 17.5$



d) Time evolution of lift and drag coefficients for the three schemes

Fig. 6. Meshes and entropy plots at time  $t = 17.5$  for the the oscillating cylinder problem, and the time evolution of the drag and the lift coefficients. The deviations between the three schemes are small, although some oscillations can be seen in the entropy plot without GCL correction.

allows for large elements away from the plate (even larger than the size of the flow features) without causing numerical instabilities, which again results in high computational efficiency.

The width of the plate is  $L = 2$  and the height is 0.1, giving an aspect ratio of 20:1. The frequency Reynolds number is  $Re_{fr} = AfL/\nu = 35$ , where  $f$  is the frequency of oscillation,  $\nu$  is the kinematic viscosity of the fluid and  $A = L/2 = 1$  is the stroke amplitude. The speed of sound is chosen as ten times the maximum vertical velocity of the plate, giving an almost incompressible flow with Mach numbers around 0.1-0.2. In addition to the compressible Navier-Stokes equations, we solve for the horizontal velocity  $u_b$  of the plate using Newton's second law:

$$M\dot{u}_b = F_{\text{fluid}}, \quad (30)$$

where  $M$  is the mass of the plate, with density  $\rho_b = 32\rho$ , and  $F_{\text{fluid}}$  is the horizontal component of the total force from the fluid on the plate. We set the initial  $u_b$  to 0.01 to trigger the instability effect.

The mesh is shown in Fig. 7 (a), and we use polynomials of order  $p = 4$  within each element. The domain is mapped by a rigid motion, corresponding to the  $x, y$  translation of the plate. The DG formulation (13) and Newton's law (30) are fully coupled and integrated simultaneously in time using a fourth order Runge-Kutta method. This is a fully coupled fluid-structure interaction problem, even though the only degrees of freedom for the structure are the horizontal displacement and velocity. Strictly speaking, the allowable time step depends on the eigenvalues of the coupled fluid-structural system of equations. Here, we have reproduced the experiments and simulations in [24] and [25], which used a rather dense plate and hence, we did not observe the structural equations placing any additional restrictions on the allowable time step determined from the flow equations only. Had we used a much lighter plate, we would have expected larger horizontal accelerations in the plate and potentially some restrictions in the allowable time step. Plots of the solution (Mach number) at the times  $t = 1.3$ ,  $t = 4.3$ , and  $t = 6.3$  are shown in Fig. 7 (b)-(d). The flow is initially symmetric but after a few periods the instability causes the symmetry to break and eventually the plate is forced into a horizontal motion. The location of the plate in the  $x - y$  plane is shown in Fig. 8.

#### 4.5. Heaving and Pitching Foil in Wake

As an example of a more complex mapping, we solve for the flow around a NACA 0012 foil of length  $c = 1$  placed in the wake of a D-section cylinder of diameter  $d = 0.5$ , following the experimental study in [26]. Depending on the distance  $s$  between the center of the cylinder and the leading edge of the foil several modes of interaction can be identified. We consider the same three distances as in [26], namely, (1)  $s = 3.66d$ , (2)  $s = 4.25d$ , and (3)  $s = 5.5d$ . The cylinder and the foil oscillate in phase vertically, with amplitude  $A$  and frequency  $f$

$$h(t) = A \sin(2\pi ft). \quad (31)$$

The foil also pitches around the point located a distance  $\bar{c} = c/3$  from the leading edge, with the time-varying angle between the foil and the horizontal axis given by

$$\theta(t) = a \sin(2\pi ft + \pi/2). \quad (32)$$

We choose the free-stream velocity  $U = 1$  in the  $x$ -direction, the reduced frequency  $f = 0.2U/d = 0.4$ , and amplitudes  $A = d/2 = 0.25$  and  $a = \pi/6$ . The free-stream Mach number of the flow is 0.2 and the Reynolds number is 550.

The translations of the half-cylinder and the foil are rigid but essentially independent of each other, so we need a mapping based on a smooth blending of three meshes – the original mesh, a translation of the cylinder, and a translation and rotation of the foil. The method is illustrated in Fig. 9, where the reference mesh and the final blended mesh are shown at the time  $t = 13.5$ .

We ran the three simulations for 100 periods (until time  $t = 100$ ), and computed the drag coefficients on the half-cylinder and on the foil. Vorticity plots of the solutions at time  $t = 40.8$  are shown in Fig. 10, together with plots of the drag coefficients as functions of time. It is clearly seen that the interaction between the foil and the vortices from the cylinder is highly dependent on the distance  $s$ . For the larger separations, the vortex shedding locks into the frequency of oscillation and a periodic pattern is generated. This situation

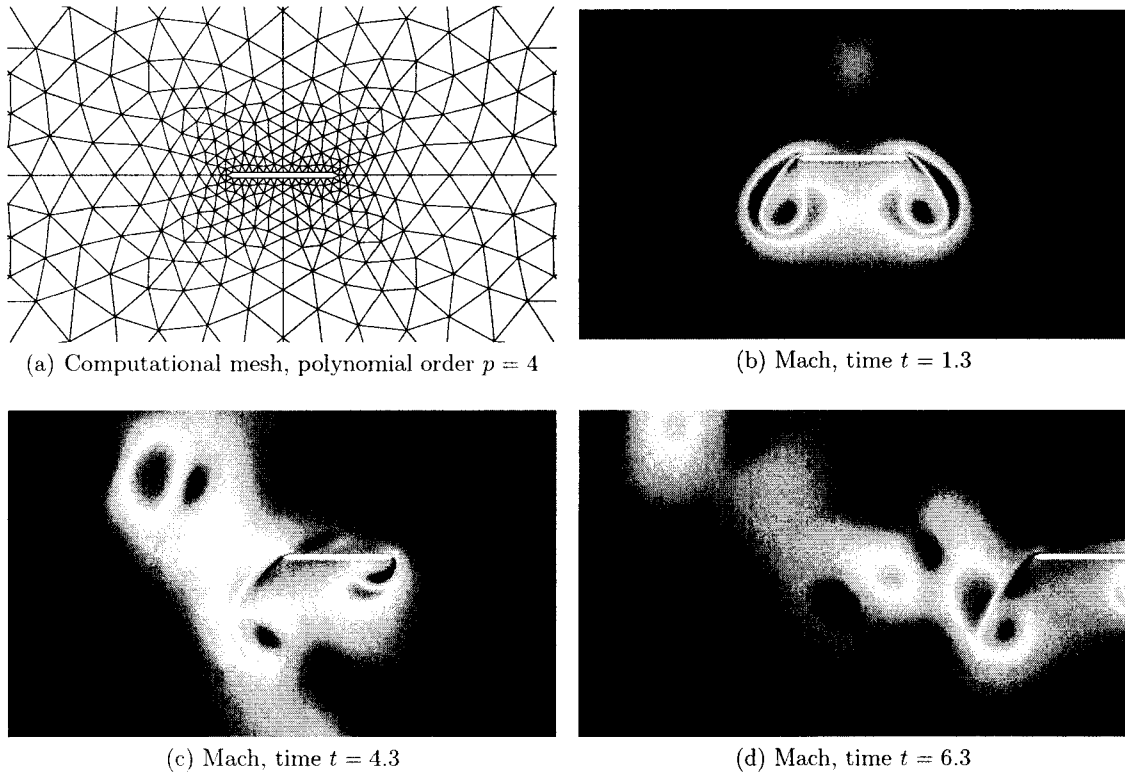


Fig. 7. The computational mesh and the flow fields around an oscillating plate which is free to move horizontally.

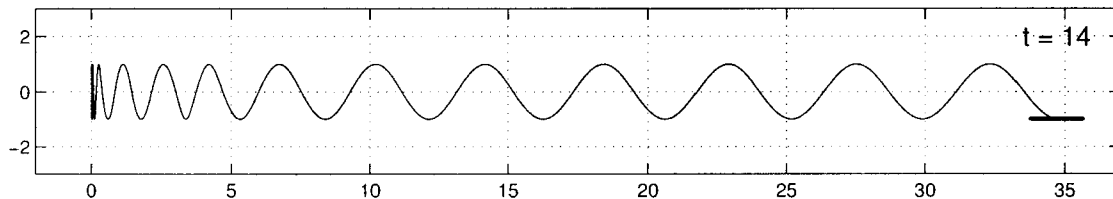


Fig. 8. The motion of the oscillating plate at time  $t = 14$ . Initially the body is located at  $x = 0$ , but due to an instability in the flow field the body is forced into a horizontal motion.

does not occur for the shorter distance where the flow appears to remain chaotic. We also note that in the cases where periodicity is established the average drag on the foil is negative indicating thrust generation. This is more clear in the intermediate case where effect of the foil is to attempt reduce the strength of the vortex street. For certain values of the amplitude and frequency of oscillation, not shown here, it is possible for the airfoil to completely reverse the vortex street and thus obtain net thrust for the combined cylinder and airfoil surfaces.

## 5. Acknowledgment

We would like to thank Mark Drela and David Willis for much valuable help with the applications. We would also like to acknowledge Anshul Mohnot for his help with the testing of the ALE formulation. Finally, we would like to thank the Singapore-MIT Alliance for the partial support of this work.

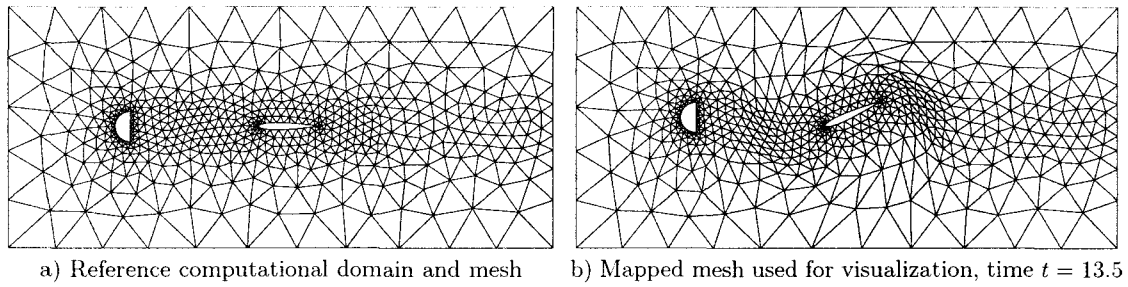


Fig. 9. The mapping for two independently moving bodies.

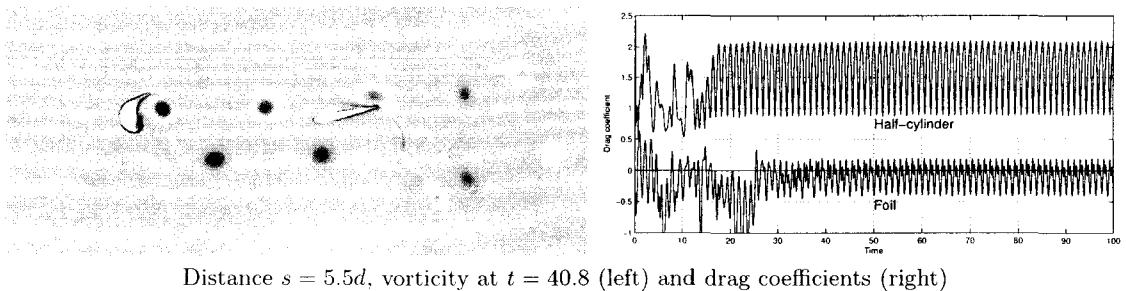
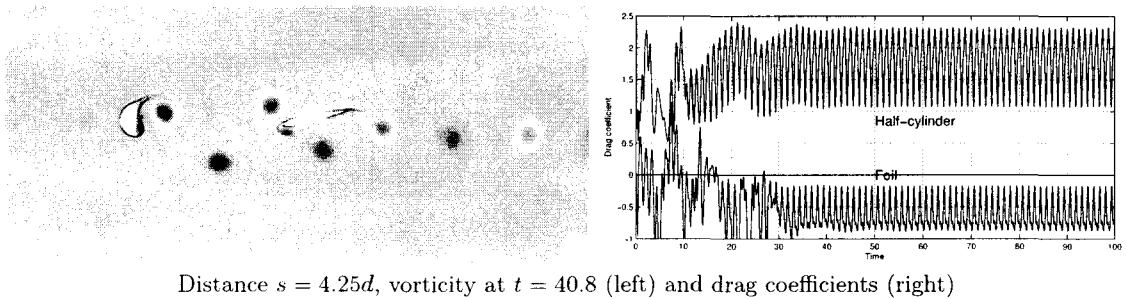
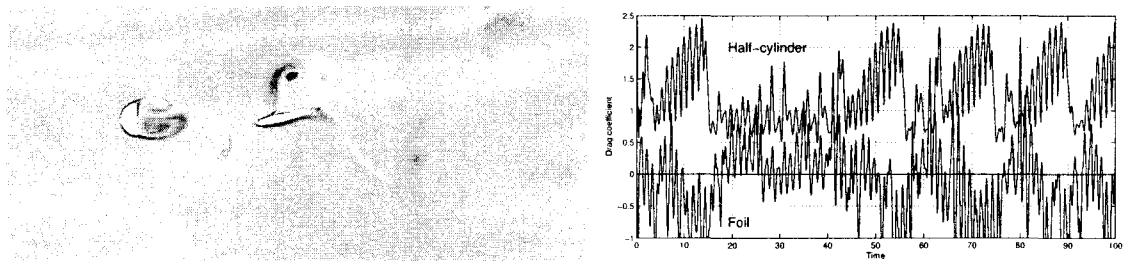


Fig. 10. The interaction between the foil and the vortices that are shed from the half-cylinder is highly dependent on the distance between the objects. For the three distances shown, the middle one produces the most regular behavior and the largest negative drag (positive thrust).



## References

- [1] J. Donea, Arbitrary Lagrangian–Eulerian finite element methods, in: T. Belytschko, T. Hughes (Eds.), *Computational Methods for Transient Analysis*, Vol. 1, Elsevier, Amsterdam, 1983, pp. 474–515.
- [2] A. Huerta, W. Liu, Viscous-flow with large free-surface motion, *Comput. Methods Appl. Mech. Engrg.* 69 (3) (1988) 277–324.
- [3] C. Farhat, P. Geuzaine, Design and analysis of robust ALE time-integrators for the solution of unsteady flow problems on moving grids, *Comput. Methods Appl. Mech. Engrg.* 193 (39-41) (2004) 4073–4095.
- [4] C. S. Venkatasubban, A new finite element formulation for ALE (arbitrary Lagrangian Eulerian) compressible fluid mechanics, *Internat. J. Engrg. Sci.* 33 (12) (1995) 1743–1762.
- [5] I. Lomtev, R. M. Kirby, G. E. Karniadakis, A discontinuous Galerkin ALE method for compressible viscous flows in moving domains, *J. Comput. Phys.* 155 (1) (1999) 128–159.
- [6] H. T. Ahn, Y. Kallinderis, Strongly coupled flow/structure interactions with a geometrically conservative ALE scheme on general hybrid meshes, *J. Comput. Phys.* 219 (2) (2006) 671–696.
- [7] P. D. Thomas, C. K. Lombard, Geometric conservation law and its application to flow computations on moving grids, *AIAA J.* 17 (10) (1979) 1030–1037.
- [8] C. Farhat, P. Geuzaine, C. Grandmont, The discrete geometric conservation law and the nonlinear stability of ALE schemes for the solution of flow problems on moving grids, *J. Comput. Phys.* 174 (2) (2001) 669–694.
- [9] H. Guillard, C. Farhat, On the significance of the geometric conservation law for flow computations on moving meshes, *Comput. Methods Appl. Mech. Engrg.* 190 (11-12) (2000) 1467–1482.
- [10] D. Boffi, L. Gastaldi, Stability and geometric conservation laws for ALE formulations, *Comput. Methods Appl. Mech. Engrg.* 193 (42-44) (2004) 4717–4739.
- [11] C. Förster, W. A. Wall, E. Ramm, On the geometric conservation law in transient flow calculations on deforming domains, *Internat. J. Numer. Methods Fluids* 50 (12) (2006) 1369–1379.
- [12] L. Formaggia, F. Nobile, Stability analysis of second-order time accurate schemes for ALE-FEM, *Comput. Methods Appl. Mech. Engrg.* 193 (39-41) (2004) 4097–4116.
- [13] T. J. R. Hughes, G. M. Hulbert, Space-time finite element methods for elastodynamics: formulations and error estimates, *Comput. Methods Appl. Mech. Engrg.* 66 (3) (1988) 339–363.
- [14] J. J. W. van der Vegt, H. van der Ven, Space–time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows: I. general formulation, *J. Comput. Phys.* 182 (2) (2002) 546–585.
- [15] R. Abedi, B. Petracovici, R. B. Haber, A space-time discontinuous Galerkin method for linearized elastodynamics with element-wise momentum balance, *Comput. Methods Appl. Mech. Engrg.* 195 (25-28) (2006) 3247–3273.
- [16] A. Üngör, A. Sheffer, Tent-Pitcher: A meshing algorithm for space–time discontinuous Galerkin methods, in: *Proceedings of the 9th International Meshing Roundtable*, Sandia Nat. Lab., 2000, pp. 111–122.
- [17] M. R. Visbal, D. V. Gaitonde, On the use of higher-order finite-difference schemes on curvilinear and deforming meshes, *J. Comput. Phys.* 181 (1) (2002) 155–185.
- [18] D. Anderson, J. Tanehill, R. Pletcher, *Computational Fluid Mechanics and Heat Transfer*, McGraw-Hill, New York, 1984.
- [19] B. Cockburn, C.-W. Shu, The local discontinuous Galerkin method for time-dependent convection-diffusion systems, *SIAM J. Numer. Anal.* 35 (6) (1998) 2440–2463 (electronic).
- [20] J. Peraire, P.-O. Persson, The compact discontinuous Galerkin (CDG) method for elliptic problems, *SIAM J. Sci. Comput.* 30 (4) (2008) 1806–1824.
- [21] P. L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, *J. Comput. Phys.* 43 (2) (1981) 357–372.
- [22] G. Erlebacher, M. Y. Hussaini, C.-W. Shu, Interaction of a shock with a longitudinal vortex, *J. Fluid Mech.* 337 (1997) 129–153.
- [23] K. Mattsson, M. Svård, M. Carpenter, J. Nordström, High-order accurate computations for unsteady aerodynamics, *Computers & Fluids* 36 (3) (2007) 636–649.
- [24] N. Vandenbergh, J. Zhang, S. Childress, Symmetry breaking leads to forward flapping flight, *J. Fluid Mechanics* 506 (2004) 147–155.
- [25] S. Alben, M. Shelley, Coherent locomotion as an attracting state for a free flapping body, *Proc. National Acad. Sciences United States Am.* 102 (32) (2005) 11163–11166.
- [26] R. Gopalkrishnan, M. S. Triantafyllou, G. S. Triantafyllou, D. Barrett, Active vorticity control in a shear flow using a flapping foil, *J. Fluid Mech.* 274 (1994) 1–21.