Advances in Applied Mathematics and Mechanics
Adv. Appl. Math. Mech., Vol. **1**, No. 3, pp. 319-340 (2009)

June 2009

The Locally Conservative Galerkin (LCG) Method — a Discontinuous Methodology Applied to a Continuous Framework

Rhodri L. T. Bevan[1], Raoul vanLoon[1] and Perumal Nithiarasu[1,*]

[1] *Civil and Computational Engineering Centre, School of Engineering, Swansea University, Swansea SA2 8PP, UK*

Received 15 December 2008; Accepted (in revised version) 07 March 2009

Available online 22 April 2009

Abstract. This paper presents a comprehensive overview of the element-wise locally conservative Galerkin (LCG) method. The LCG method was developed to find a method that had the advantages of the discontinuous Galerkin methods, without the large computational and memory requirements. The initial application of the method is discussed, to the simple scalar transient convection-diffusion equation, along with its extension to the Navier-Stokes equations utilising the Characteristic Based Split (CBS) scheme. The element-by-element solution approach removes the standard finite element assembly necessity, with an face flux providing continuity between these elemental subdomains. This face flux provides explicit local conservation and can be determined via a simple small post-processing calculation. The LCG method obtains a unique solution from the elemental contributions through the use of simple averaging. It is shown within this paper that the LCG method provides equivalent solutions to the continuous (global) Galerkin method for both steady state and transient solutions. Several numerical examples are provided to demonstrate the abilities of the LCG method.

AMS subject classifications: 35D99, 65-05, 76D99, 76Z05

Key words: Explicit local flux conservation, element-by-element solution, local conservation, LCG, convection-diffusion, CBS, artificial compressibility, incompressible flow.

1 Introduction

The locally conservative Galerkin (LCG) method, introduced in 2004 [1], has been developed and employed in an increasing number of applications. It is however important to note that its potential has not yet been fully realised. In this paper an overview

*Corresponding author.
URL: `http://www.swansea.ac.uk/staff/academic/Engineering/nithiarasuperumal/`
Email: P.Nithiarasu@swansea.ac.uk (P. Nithiarasu), R.vanloon@swansea.ac.uk (R. Van Loon), 191381@swansea.ac.uk (R. L. T. Bevan)

of the method is provided along with several numerical examples to demonstrate its ability.

The conservation of the Galerkin finite element method has been investigated by many researchers [2–7] and the method has been shown to be globally conservative if Neumann boundary conditions are imposed. Local conservation is seen as a highly valued property possessed by both finite volume methods and discontinuous Galerkin (DG) finite element methods. Local (element-wise) conservation may be of advantage when considering problems involving discontinuities or interfaces in problems such as porous-medium interfaces and fluid structure interaction.

In order to achieve a timely solution, the use of parallel computation is a desirable property for a method to possess. In order to parallelise the global Galerkin method domain decomposition techniques are required [8–10]. An alternative to decomposing the domain would be to use an element-by-element solution scheme. Such a scheme is already decomposed into multiple subdomains and as such is more applicable to parallel computation.

To rectify some of the inherent drawbacks of the continuous (or global) Galerkin (GG) method, the discontinuous Galerkin method (DGM) [11–22] has been developed. The DGM allows for element by element solution and can therefore be easily parallelised [23, 24]. Discontinuous methods can also utilise varying orders of approximation to locally capture a more refined solution [25]. Use of DG methods within industry is often hampered by the large CPU and memory requirements, due to the requirements of storing multiple solutions at a node as well as solving for additional flux variables. Therefore, researchers have been seeking a DG method with the structure of a continuous Galerkin (CG) method [26].

The approach adopted within this paper allows for the introduction of the interface fluxes within the continuous (global) framework [1, 27, 28], rather than adopt a discontinuous Galerkin method to a continuous Galerkin framework. This method can be readily adapted to existing industrial codes with a minimum of modification. The adoption of the interface fluxes does not necessarily have to break the shape function spaces but can be constructed within the continuous framework. The proposed method, corrected LCG, is identical to both the standard and stabilised versions of the continuous Galerkin method for internal nodes. The LCG method and the CG method are not identical on the global boundaries due to the nature of the CG method itself. The LCG method is implicitly globally conservative and does not require the extra calculations CG would require to be globally conservative. The LCG method utilises an element-by-element solution approach removing the standard finite element assembly, with a edge (2D) or face (3D) flux providing continuity between these elemental subdomains. This removal of the standard assembly allows the elemental matrices to be computed, inverted and stored at the pre-processing stages of a computation. This is possible since often these matrices are functions of an element property, such as volume. In an Eulerian frame of reference this property does not change, therefore control over individual elements gives a great deal of freedom to optimise memory requirements. Investigation has shown that the majority of LCG forms, especially the implicit

forms, are much faster than their continuous (or global) Galerkin counter parts [1,27].

In the LCG method, the face flux is weakly imposed via a small post-processing calculation at every time step. The formulation is written in a space-time framework, utilising a pseudo time step as an iterative mechanism. Transient solutions can be recovered through use of a dual time stepping approach. The method is both locally and globally conservative and maintains many of the advantages DG methods offer, but at a lower computational cost. This is due to the use of a small post processing calculation to determine the face fluxes. Memory requirements are not as intensive as with DG methods, since multiple solutions at a node can be discarded after each time step.

The paper is organised into the following sections. In Section 2, the convection-diffusion equation is introduced. Section 3 discusses the locally conservative Galerkin spatial discretisation of the convection-diffusion problem. Section 4 investigates the solutions produced by both the locally conservative Galerkin method and the continuous Galerkin method. In Section 5 the Navier-Stokes equations and their temporal discretisation, using the characteristic based split (CBS) scheme, are briefly introduced. In Section 6 the locally conservative Galerkin spatial discretisation of the semi-discrete equations is discussed. Discussion in this section includes the matrices arising from the spatial discretisation and a brief summary of local and dual time stepping procedures. Section 7 contains numerical examples to demonstrate the validity and accuracy of the LCG method against both continuous Galerkin solutions and benchmark values. These examples consist of 2D steady state heat conduction problem, a transient 2D flow around a circular cylinder at a Reynolds number of 100, a 3D steady state lid driven cavity flow problem at a Reynolds number of 100, and a patient specific 3D carotid bifurcation. Finally, in Section 8, some important conclusions are discussed along with the possibilities for future research.

2 Convection-diffusion problem

Consider the following simple scalar transient convection-diffusion equation:

$$\frac{\partial \phi}{\partial t} + \frac{\partial F_i}{\partial x_i} = 0, \tag{2.1}$$

where β is a constant, ϕ is a scalar variable and F_i is the flux. In general, the flux has both diffusive and convective components, although it can be purely diffusive or convective. The equation is complete with the following initial and boundary conditions.

$$\phi(\mathbf{x}, t=0) = \phi_0(\mathbf{x}), \quad \forall \mathbf{x} \in \Omega, \tag{2.2a}$$

$$\phi = \overline{\phi}(\mathbf{x}, t), \quad \text{on} \Gamma_\phi, \tag{2.2b}$$

$$F_n = \overline{F}(\mathbf{x}, t), \quad \text{on} \Gamma_f. \tag{2.2c}$$

The boundary domain Γ of Ω is decomposed as $\Gamma = \Gamma_\phi \cap \Gamma_f$, where Γ_ϕ and Γ_f represent the Dirichlet and Neumann partitions respectfully. The semi-discrete form of Eq.

(2.1) in its explicit form is defined as

$$\frac{\phi^{n+1}-\phi^n}{\Delta t}=-\frac{\partial F_i^n}{\partial x_i}, \tag{2.3}$$

with the spatial discretisation of the scalar variable ϕ approximated as

$$\phi \approx \widetilde{\phi}=\sum_{a=1}^{m} N_a\phi_a=\mathbf{N}\mathbf{\Phi}, \tag{2.4}$$

where $\widetilde{\phi}$ indicates an approximate quantity, $\mathbf{N}$ is the shape functions and subscript a indicates nodes. The Galerkin weighted residual form of Eq. (2.3) is defined as

$$\int_\Omega N_aN_b\frac{\Delta\widetilde{\phi}}{\Delta t}d\Omega=\int_\Omega N_a\frac{\partial\widetilde{F_i^n}}{\partial x_i}d\Omega, \tag{2.5}$$

where $\widetilde{F_i^n}$ denotes the approximate flux, i denotes the spatial direction, n denotes the n^{th} time step and $\Delta\widetilde{\phi}=\widetilde{\phi}^{n+1}-\widetilde{\phi}^n$. Performing integration by parts on the RHS term of Eq. (2.5) gives

$$\int_\Omega N_aN_b\frac{\Delta\widetilde{\phi}}{\Delta t}d\Omega=\int_\Omega\frac{\partial N_a}{\partial x_i}\widetilde{F_i^n}d\Omega-\int_\Gamma N_a\widetilde{F_i^n}n_id\Gamma, \tag{2.6}$$

where n_i represents the components of the outward boundary normal vector.Note that Eq. (2.6) represents an assembled system of simultaneous equations using the standard (continuous) Galerkin discretisation. This equation is valid for any domain divided into non-overlapping continuous elements. The loading vector $\{\mathbf{f}\}$ may be determined using only a small post-processing calculation, as opposed to DG methods which utilise extra equations to determine the interface flux, with flux continuity often determined using Lagrangian multipliers [21].

3 The LCG discretisation

In the locally conservative Galerkin method the variable and its fluxes are explicitly conserved over each individual element. If it is possible to ensure that the flux crossing a common face between two elements is equal and opposite then Eq. (2.6) is valid between individual elements. The LCG method conserves the fluxes and the variable through calculation of a numerical face flux. This ensures continuity between neighbouring elements. This process is equivalent to treating the global domain as a group of elemental subdomains, each with its own time-dependent Neumann boundary conditions prescribed at each timestep.

Assuming accurate calculation of the numerical face flux, Eq. (2.6) can be rewritten for an elemental subdomain Ω_e as

$$\int_{\Omega_e}\mathbf{N}^\mathbf{T}\mathbf{N}\frac{\Delta\mathbf{\Phi}}{\Delta t}d\Omega_e=\int_{\Omega_e}\frac{\partial\mathbf{N}^\mathbf{T}}{\partial x_i}\widetilde{F}_i^n d\Omega_e-\int_{\Gamma_e}\mathbf{N}^\mathbf{T}\hat{\widetilde{F}}_i^n n_id\Gamma_e, \tag{3.1}$$

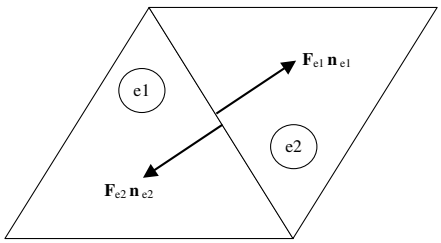


Figure 1: Flux crossing a common face (edge) between two triangular elements.

where subscript e indicates an element and Γ_e indicates an element face. The face flux term is replaced with a numerical flux $\hat{\hat{F}}_i^n$ across the elemental boundary. In matrix form, Eq. (3.1) can be rewritten as

$$[\mathbf{M}_e]\{\Delta\mathbf{\Phi}\} = \Delta t\left([\mathbf{K}_e]\{\mathbf{\Phi}\}^n + \{\mathbf{f}_{\mathbf{\Gamma_e}}\}^n\right), \tag{3.2}$$

where

$$\{\Delta\mathbf{\Phi}\} = \{\mathbf{\Phi}\}^{n+1} - \{\mathbf{\Phi}\}^n,$$

$[\mathbf{M}_e]$ denotes the elemental mass matrix, $[\mathbf{K}_e]$ denotes the stiffness matrix and $\{\mathbf{f}_{\mathbf{\Gamma_e}}\}$ denotes the face flux vector. The numerical face flux is calculated utilising a postprocessing approach, with the loading vector $\{\mathbf{f}_{\mathbf{\Gamma_e}}\}$ determined at the n^{th} time level regardless of the time discretisation. To determine nodal values for the numerical face flux, several postprocessing approaches are available. These include superconvergence patch recovery (SPR) and area-weighting the gradients, although only a simple averaging approach is utilised in order to reduce complexity. Once the nodal values are obtained the following condition (Fig. 1) is enforced

$$F_{e1}n_{e1} = F_{e2}n_{e2}. \tag{3.3}$$

An implicit solution procedure is also possible and involves treating matrix $[\mathbf{K}_e]$ implicitly. Eq. (3.2) may be rewritten as

$$[\mathbf{M}_e + \Delta t\mathbf{K}_e]\{\mathbf{\Phi}\}^{n+1} = [\mathbf{M}_e]\{\mathbf{\Phi}\}^n + \Delta t\{\mathbf{f}_{\mathbf{\Gamma_e}}\}^n. \tag{3.4}$$

In both the explicit and the implicit solution procedure, the mass matrix $[\mathbf{M}_e]$ can either be kept as a consistent mass matrix or lumped. A lumping procedure for a linear triangular element would involve row summation.

The LCG method involves solving elemental discrete equations, with the connections between the elements established via fluxes crossing the common faces. This leads to multiple solutions at a node for the scalar variable ϕ, and in order to achieve a unique solution, an arithmetic mean of the nodal values obtained from the different elements is undertaken.

4 Comparison between continuous Galerkin and LCG method

To provide a comparison between the continuous Galerkin and the locally conservative Galerkin method, a two–dimensional transient explicit convection-diffusion prob-

lem is considered. The continuous Galerkin method for an interior node a (see Fig. 2) using linear elements is given as

$$(\sum_{e=1}^{5} \mathbf{M_e}) \{\Delta\phi_a\} = \Delta t (\sum_{e=1}^{5} \mathbf{K_e}) \{\mathbf{\Phi}\}^n, \tag{4.1}$$

where the mass vector $[\mathbf{M}]$ is lumped, $\mathbf{K_e}$ is element $_\mathbf{e}$'s contribution to $[\mathbf{K}]$.

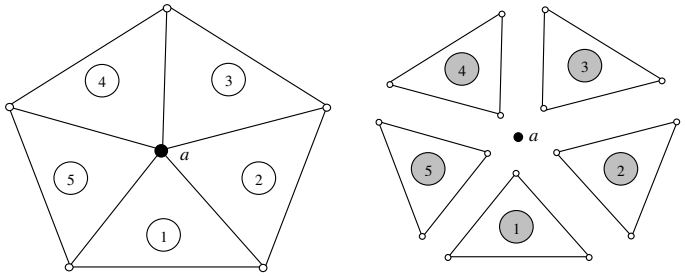


Figure 2: Interior node a - Element connectivity.

The LCG method gives multiple nodal equations due to its element by element solution approach. With the introduction of Neumann type boundary flux conditions at elemental interfaces, nodal values of the fluxes are required. Nodal values of convection fluxes can be calculated at the nodes, however diffusive fluxes are constant over the element and hence nodal values of these fluxes are determined by averaging the values over the connected elements (see Eq. (4.2)).

$$(\frac{\partial\hat{\phi}}{\partial x_i})_a = \frac{1}{ne} \sum_{e=1}^{ne} (\frac{\partial\phi}{\partial x_i})_e. \tag{4.2}$$

The LCG nodal equation for element 1 is

$$[\mathbf{M_1}] \{\Delta\phi_a\} = \Delta t \left([\mathbf{K_1}] \{\mathbf{\Phi}\}^n + \{\mathbf{f_1}\}^n \right), \tag{4.3}$$

where $[\mathbf{M_1}]$ represents the mass vector for element 1. It follows that Eq. (4.3) can be repeated for all of the five connected elements. Using the simple averaging technique to produce a unique solution gives

$$\begin{aligned} &\big([\mathbf{M_1}] + [\mathbf{M_2}] + [\mathbf{M_3}] + [\mathbf{M_4}] + [\mathbf{M_5}]\big)\{\Delta\phi_a\} \\ &= \frac{1}{5}\Delta t\Big([\mathbf{K_1}]\{\mathbf{\Phi}\}^n + \{\mathbf{f_1}\}^n + [\mathbf{K_2}]\{\mathbf{\Phi}\}^n + \{\mathbf{f_2}\}^n + [\mathbf{K_3}]\{\mathbf{\Phi}\}^n \\ &\qquad + \{\mathbf{f_3}\}^n + [\mathbf{K_4}]\{\mathbf{\Phi}\}^n + \{\mathbf{f_4}\}^n + [\mathbf{K_5}]\{\mathbf{\Phi}\}^n + \{\mathbf{f_5}\}^n \Big), \end{aligned} \tag{4.4}$$

which can be simplified on consideration that, by principle, the numerical boundary flux terms will cancel out

$$(\sum_{e=1}^{5} \mathbf{M_e}) \{\Delta\phi_a\} = \frac{1}{5}\Delta t (\sum_{e=1}^{5} \mathbf{K_e}) \{\mathbf{\Phi}\}^n. \tag{4.5}$$

From comparison between Eqs. (4.5) and (4.1) it is clear that the LCG method and the CG method will provide the same steady state solution, and in the case where the individual elements are of identical size, an identical transient solution also. In order for the LCG method to provide an identical solution for a transient problem on a non-uniform mesh then a correction factor must be applied. Eq. (4.3) can be rearranged as

$$\{\phi_a\}^{n+1} = \{\phi_a\}^n + [\mathbf{M_1}]^{-1}\,\Delta t\left([\mathbf{K_1}]\{\mathbf{\Phi}\}^n + \{\mathbf{f_1}\}^n\right). \tag{4.6}$$

The correction factor ψ is applied to the mass matrix term on the RHS, and is of the form

$$\psi = \left(ne\,[\mathbf{M}]^{-1} - [\mathbf{M_e}]^{-1}\right). \tag{4.7}$$

5 Incompressible Navier-Stokes equations and the CBS scheme

5.1 The Navier-Stokes equations for incompressible flow

The artificial compressibility based Navier-Stokes equations may be written as

$$\text{Continuity} \qquad \frac{1}{\beta^2}\frac{\partial p}{\partial t} + \rho\frac{\partial u_i}{\partial x_i} = 0, \tag{5.1}$$

$$\text{Momentum} \qquad \frac{\partial u_i}{\partial t} + u_j\frac{\partial u_i}{\partial x_j} + \frac{1}{\rho}\frac{\partial p}{\partial x_i} - \frac{1}{\rho}\frac{\partial \tau_{ij}}{\partial x_j} = 0, \tag{5.2}$$

where u_i represent the cartesian components of the velocity vector, p the pressure, β an artificial compressibility parameter and ρ the fluid density [29–33]. Deviatoric stress components τ_{ij} are related to velocity gradients by

$$\tau_{ij} = \mu\Big(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\Big), \tag{5.3}$$

where μ is the dynamic viscosity. The non-dimensional form of Eqs. (5.1) and (5.2) are rewritten as:

$$\text{Continuity} \qquad \frac{1}{\beta^{*2}}\frac{\partial p^*}{\partial t^*} = -\rho^*\frac{\partial u_i^*}{\partial x_i^*}, \tag{5.4}$$

$$\text{Momentum} \qquad \frac{\partial u_i^*}{\partial t^*} = -u_j^*\frac{\partial u_i^*}{\partial x_j^*} + \frac{1}{Re}\frac{\partial^2 u_i^*}{\partial x_j^{*2}} - \frac{\partial p^*}{\partial x_i^*}, \tag{5.5}$$

where asterisks denote non-dimensional variables defined as

$$x_i^* = \frac{x_i}{L}; \qquad u_i^* = \frac{u_i}{u_\infty}; \qquad t^* = \frac{tu_\infty}{L}; \tag{5.6}$$

$$p^* = \frac{p}{\rho u_\infty^2}; \quad \beta^* = \frac{\beta}{u_\infty}; \qquad Re = \frac{\rho_\infty u_\infty L}{\mu_\infty}. \tag{5.7}$$

Here Re is the Reynolds number, the subscript ∞ denotes a reference quantity and L is a reference length. The problem definition is completed by selecting appropriate initial and boundary conditions. Asterisks will now be dropped for simplicity of presentation.

5.2 The semi-discrete CBS scheme

The characteristic based split (CBS) scheme is well established in the continuous (global) Galerkin framework, and can be employed for simulations involving both incompressible and compressible flow [32, 34, 35]. The artificial compressibility technique has been incorporated into the CBS family since 2003 [29, 30, 36, 37]. For a recent review of the abilities of the CBS scheme, the reader is referred to the work of Nithiarasu *et al.* [38].

The LCG spatial discretisation procedure can be coupled to the semi-discrete CBS scheme to obtain an element-by-element solution technique. The CBS algorithm utilises a fractional step involving a split to circumvent the LBB restriction. The three steps of the CBS scheme can be described as:

1. Solve for the auxiliary intermediate velocity field ${u_i}^{\dagger}$;
2. Solve for the pressure field p;
3. Determine the corrected velocity field u_i.

The first step of the CBS scheme follows from a rearranged Eq. (5.5) without the pressure term

$$\frac{\partial {u_i}^{\dagger}}{\partial t} + \frac{\partial F_{ij}}{\partial x_j} = 0, \tag{5.8}$$

where the flux term F_{ij} is defined as

$$F_{ij} = \Big(u_j u_i - \frac{1}{Re}\frac{\partial u_i}{\partial x_j} \Big). \tag{5.9}$$

Applying the *simple explicit* characteristic temporal discretisation [39] to Eq. (5.8) gives

$$\frac{{u_i}^{\dagger} - {u_i}^{n}}{\Delta t} = -\big(\frac{\partial F_{ij}}{\partial x_j}\big)^n + \frac{\Delta t}{2} u_k \frac{\partial}{\partial x_k}\big(\frac{\partial F_{ij}}{\partial x_j}\big)^n. \tag{5.10}$$

This forms the first step of the LCG-CBS scheme. The corrected value of ${u_i}^{n+1}$ is given by

$${u_i}^{n+1} = {u_i}^{\dagger} - \Delta t\big(\frac{\partial p}{\partial x_i}\big)^n + \frac{\Delta t^2}{2} u_k \frac{\partial}{\partial x_k}\big(\frac{\partial p}{\partial x_i}\big)^n, \tag{5.11}$$

which is actually the third step of the scheme. The second step of the scheme is the determination of the pressure field. The pressure equation used is based on the conti-

nuity relation given by Eq. (5.4)

$$\frac{1}{\beta^2}\frac{\Delta p}{\Delta t} = -\rho\frac{\partial {u_i}^{n+1}}{\partial x_i}. \tag{5.12}$$

Using Eq. (5.11) to eliminate ${u_i}^{n+1}$ and neglecting terms higher than second-order, the pressure equation becomes

$$\frac{1}{\beta^2}\frac{\Delta p}{\Delta t} = -\rho\frac{\partial}{\partial x_i}\left({u_i}^{\dagger} - \Delta t(\frac{\partial p}{\partial x_i})^n\right). \tag{5.13}$$

Eq. (5.13) is the second step of the algorithm. Assuming the pseudo compressibility allows for a fully explicit scheme. The artificial compressibility parameter, β, has been successfully used since 2003 [29–33, 36–38, 40]. The value of β can be a designated constant within the domain although the recommended technique, adopted here, is to calculate β locally. This local value of β is based on both convective and diffusive time-step restrictions [29, 30, 32]. This accommodates both flow regimes (convection and diffusion dominated) within a problem at a particular Reynolds number. In this work the relation

$$\beta = \max\left(\epsilon, \nu_{conv}, \nu_{diff}\right), \tag{5.14}$$

is employed. The constant ϵ, is to ensure that β does not approach zero, and typically takes the value of $0.1 \leq \epsilon \leq 0.5$. ν_{conv} is the local convective velocity and ν_{diff} is the local diffusive velocity. These velocities are calculated from the non-dimensional relations [29]:

$$\nu_{conv} = \sqrt{u_i u_i}, \qquad \nu_{diff} = (hRe)^{-1}. \tag{5.15}$$

6 LCG discretization of CBS scheme

As with the continuous Galerkin method, the variables are approximated by the standard finite element spatial discretisation as:

$$u_i \approx \tilde{u_i} = \mathbf{N}\mathbf{u_i} \quad \text{and} \quad p \approx \tilde{p} = \mathbf{N}\mathbf{p}, \tag{6.1}$$

where $\mathbf{N}$ are the shape functions. In an LCG discretisation, the domain can be broken into elemental subdomains if required. Introducing the Galerkin weighting and integrating by parts, neglecting third- and higher-order terms gives the final matrix form of the first step as

$$[\mathbf{M1_e}]\{\Delta\mathbf{u_i}^{\dagger}\} = \Delta t\Big([\mathbf{K1_e}]\{\mathbf{u_i}\}^n + [\mathbf{K1_e^{cg}}]\{\mathbf{u_i}\}^n + \{\mathbf{f\hat{1}_e}\}^n\Big), \tag{6.2}$$

$$[\mathbf{M1_e}] = \int_{\Omega_e}\mathbf{N^T}\mathbf{N}d\Omega_e, \qquad [\mathbf{K1_e}] = \int_{\Omega_e}\frac{\partial\mathbf{N^T}}{\partial x_j}\Big(u_j\mathbf{N} - \frac{1}{Re}\frac{\partial\mathbf{N}}{\partial x_j}\Big)d\Omega_e,$$

$$[\mathbf{K1_e^{cg}}] = \int_{\Omega_e}(\frac{\Delta t}{2}u_k)\frac{\partial\mathbf{N^T}}{\partial x_k}u_j\frac{\partial\mathbf{N}}{\partial x_j}d\Omega_e,$$

$$\{\mathbf{f\hat{1}_e}\}^n = -\int_{\Gamma_e} \mathbf{N^T} u_j \mathbf{N} n_j d\Gamma_e \{\mathbf{u_i}\}^n + \frac{1}{Re}\int_{\Gamma_e} \mathbf{N^T N} n_j d\Gamma_e \{\frac{\partial \hat{\mathbf{u}}_\mathbf{i}}{\partial \mathbf{x_j}}\}^\mathbf{n} + \int_{\Gamma_e} (\frac{\Delta t}{2} u_k) \mathbf{N^T} u_j \mathbf{N} n_k d\Gamma_e \{\frac{\partial \hat{\mathbf{u}}_\mathbf{i}}{\partial \mathbf{x_j}}\}^n. \tag{6.3}$$

The boundary terms are replaced with a numerical flux $\mathbf{f\hat{1}_e}$, calculated at time n on the element boundaries. $\mathbf{K1_e^{cg}}$ denotes the characteristic Galerkin stabilisation term. The final matrix form of the second step is

$$[\mathbf{M2_e}]\{\Delta \mathbf{p}\} = \Delta t\Big([\mathbf{C2_e}]\{\mathbf{u_i}^\dagger\} - \Delta t[\mathbf{K2_e}]\{\mathbf{p}\}^n + \{\mathbf{f\hat{2}_e}\}\Big), \tag{6.4}$$

$$[\mathbf{M2_e}] = \int_{\Omega_e} \mathbf{N^T} (\frac{1}{\beta^2})^n \mathbf{N} d\Omega_e, \qquad [\mathbf{C2_e}] = \int_{\Omega_e} \frac{\partial \mathbf{N^T}}{\partial x_i} \mathbf{N} d\Omega_e,$$

$$[\mathbf{K2_e}] = \int_{\Omega_e} \frac{\partial \mathbf{N^T}}{\partial x_i} \frac{\partial \mathbf{N}}{\partial x_i} d\Omega_e,$$

$$\{\mathbf{f\hat{2}_e}\}^n = -\int_{\Gamma_e} \mathbf{N^T N} n_i d\Gamma_e (\{\mathbf{u_i}^\dagger\} - \Delta t \{\frac{\partial \hat{\mathbf{p}}}{\partial \mathbf{x_i}}\}^\mathbf{n}). \tag{6.5}$$

The final matrix form of the third step is

$$[\mathbf{M3_e}]\{\mathbf{u_i}\}^{n+1} = [\mathbf{M3_e}]\{\mathbf{u_i}\}^\dagger + \Delta t\Big([\mathbf{K3_e}]\{\mathbf{p}\} - [\mathbf{K3_e^{cg}}]\{\mathbf{p}\} + \{\mathbf{f3_e}\}\Big)^n, \tag{6.6}$$

$$[\mathbf{M3_e}] = \int_{\Omega_e} \mathbf{N^T N} d\Omega_e, \qquad [\mathbf{K3_e}] = \int_{\Omega_e} \frac{\partial \mathbf{N^T}}{\partial x_i} \mathbf{N} d\Omega_e,$$

$$\big[\mathbf{K3_e^{cg}}\big] = \int_{\Omega_e} (\frac{\Delta t}{2} u_k) \frac{\partial \mathbf{N^T}}{\partial x_k} \frac{\partial \mathbf{N}}{\partial x_i} d\Omega_e,$$

$$\{\mathbf{f3_e}\}^n = -\int_{\Gamma_e} \mathbf{N^T N} n_i d\Gamma_e \{\mathbf{p}\}^n + \int_{\Gamma_e} (\frac{\Delta t}{2} u_k) \mathbf{N^T N} n_k d\Gamma_e \{\frac{\partial \hat{\mathbf{p}}}{\partial \mathbf{x_i}}\}^n. \tag{6.7}$$

6.1 Local time-stepping

A local time-step is introduced at each node a, for the artificial compressibility LCG-CBS method is defined as

$$\Delta t_a = \min\Big(\frac{h_a}{\sqrt{u_i u_i} + \beta}, \frac{h^2}{2} Re\Big), \tag{6.8}$$

where h_a is an appropriate element size, which in this work is defined as

$$h_a = \min(3V/A_o)_e, \tag{6.9}$$

in three dimensional problems. V denotes element volume, and A_o denotes the area of the opposite face. The minimum value is selected among the number of elements e connected to the node a. In two dimensions, the relation is defined as

$$h_a = \min(2A/L_o)_e, \tag{6.10}$$

where A represents the area of the element and L_o the opposite side length.

The artificial compressibility parameter β in Eq. (6.8) is calculated from Eq. (5.14). The inclusion of β in Eq. (6.8) allows the artificial wave speed to be included into the local time-step limit. The calculated Δt_a is in practice multiplied by a non-zero safety factor below unity, with the actual value of the safety factor dependent on the problem being simulated and the mesh used. The approach adopted for this study does not handle the convection part of Δt using the Courant-Friedrichs-Lewy (CFL) number condition as advocated by many authors in order to avoid tuning the CFL number and the diffusion time step parameter [29].

The convergence criteria of the artificial compressibility LCG-CBS scheme is defined using the velocity residual error of the solution. The L_2 residual norm of the velocity is given as

$$\|\mathbf{u}\|_{L_2} = \Big(\frac{\sum_{i=1}^{nnode} (|u|_i^{n+1} - |u|_i^n)^2}{\sum_{i=1}^{nnode} (|u|_i^{n+1})^2} \Big)^{-1/2}, \tag{6.11}$$

where *nnode* is the total number of nodes in the mesh. The criterion for reaching steady-state is that this error should be reduced to a value of 10^{-5} or less.

6.2 Recovering a transient solution via a dual time-stepping approach

For the solution of transient incompressible-flow problems, a dual time-stepping procedure is utilised. The method has been shown to be very successful for both preconditioned artificial-compressibility, finite volume [31, 40, 41] and the continuous Galerkin fully-explicit CBS schemes [29, 30, 32, 36–38].

To determine the transient solution, a real time stepping term is added to the momentum equation. This term is added to the third step of the CBS scheme to remain consistent with the continuous Galerkin approach [29, 30, 36–38]. The third step becomes

$$\{\mathbf{u_i}\}^{n+1} = \{\mathbf{u_i}\}^{\dagger} + [\mathbf{M3_e}]^{-1}\Delta t \Big([\mathbf{K3_e}]\{\mathbf{p}\} - [\mathbf{K3_e^{cg}}]\{\mathbf{p}\} + \{\mathbf{f3_e}\} \Big)^n - \frac{\Delta t}{\Delta \tau}\{\Delta \mathbf{u_i}\}^{\tau}, \tag{6.12}$$

where $\Delta\tau$ is the real time-step. When utilising the dual time-stepping procedure, local Δt becomes a (pseudo) time-stepping iterative mechanism. In order to get second-order real time accuracy, $\{\Delta \mathbf{u_i}\}^{\tau}$ is approximated with an implicit second-order backward-difference formula

$$\{\mathbf{u_i}\}^{\tau} = \frac{1}{2}\Big(3\{\mathbf{u_i}\}^{m+1} - 4\{\mathbf{u_i}\}^{m} + \{\mathbf{u_i}\}^{m-1} \Big). \tag{6.13}$$

In the above equation, $\{\mathbf{u_i}\}^{m+1}$ is the n^{th} pseudo-time level value within the pseudo-time loop, $\{\mathbf{u_i}\}^{m}$ is the steady-state solution at the last real time-step and $\{\mathbf{u_i}\}^{m-1}$ is

the steady-state solution at one real time-step before the last. This requires the storage of both the latter vectors at the end of each real-time step. The real time-step size is unrestricted due to the scheme's implicit nature and only governed by the quality of the transient solution required. The pseudo time-step, however, is locally calculated and subjected to the usual stability conditions.

7 Numerical examples

7.1 A 2D steady state heat conduction problem

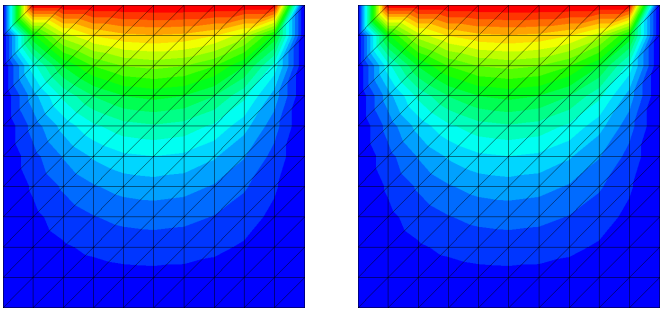
This problem is a standard benchmark problem. The problem consists of a heat conduction problem in a two dimensional square plate of unit size, subject to Dirichlet boundary conditions on all external faces. The vertical sides, as well as the bottom side are subjected to a constant temperature of 100°C, with the top side of the plate is subjected to a constant temperature of 500°C. The initial temperature value of the plate is assumed to be 0°C. The specific heat c_p, density ρ and thermal diffusivity k are assumed to be 1.0. Two structured meshes were utilised. Mesh A (A) consisted of 200 linear triangular elements and 121 nodes, Mesh B (B) contained 800 linear triangular elements and 441 nodes. The analytical solution to this problem is given by [42]

$$T(x_1,x_2)=(T_{top}-T_{side})\frac{2}{\pi}\sum_{n=1}^{\infty}\frac{(-1)^{n+1}+1}{n}\sin\left(\frac{n\pi x_1}{w}\right)\frac{\sinh(n\pi x_2/w)}{\sinh(n\pi H/w)}+T_{side}. \quad (7.1)$$

Using Eq. (7.1) gives an analytical solution of 200°C at the centre node of the plate $(0.5,0.5)$. The problem was analysed using both the explicit and implicit variations of the LCG and continuous Galerkin method. Use of a consistent mass matrix produced the least accurate results for both the LCG and continuous Galerkin methods. The solution of the lumped mass matrix explicit variant of the LCG and continuous Galerkin methods using Mesh A is shown in Fig. 3 and the implicit variant (lumped mass) in Fig. 4. Table 1 gives the results obtained for the centre point $(0.5,0.5)$ from the various versions. Comparison of the CPU times to completion for the different versions can be found in Table 1 for a fixed time step size of $5.0e^{-4}$.

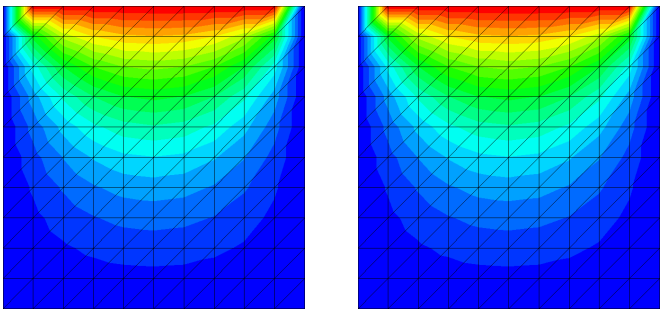
Table 1: 2D Steady State conduction in a square plate. Comparison of temperature calculated at the centre of the plate and comparison of CPU times for preprocessing and iterations. A: mesh A; B: mesh B; T: temperature; t: time; E.: explicit; I.: implicit; cm: consistent mass; lm: lumped mass.

Method	T (A) °C	T (B) °C	t (A) s	t (B) s
E. LCG (cm)	206.200	-	0.110	-
E. LCG (lm)	200.000	200.000	0.093	0.297
I. LCG (cm)	201.536	199.841	0.141	0.468
I. LCG (lm)	199.478	199.707	0.141	0.469
E. GG (cm)	202.270	-	1.313s	-
E. GG (lm)	200.000	200.000	0.062	0.156
I. GG (cm)	200.774	199.708	1.375	286.672
I. GG (lm)	199.445	199.595	1.453	292.718



(a) LCG (b) GG

Figure 3: 2D Steady State conduction in a square plate using Mesh A. Explicit (lumped mass) solutions.



(a) LCG (b) GG

Figure 4: 2D Steady State conduction in a square plate using Mesh A. Implicit (lumped mass) solutions.

It can be seen from Table 1 that the explicit (lumped mass) version of the LCG and continuous Galerkin methods produced identical solutions that agreed with the analytical solution. The implicit (lumped mass) version of the methods was also in good agreement with each other and the analytical solution, with the LCG solution being marginally closer to the analytical solution. The explicit consistent mass versions performed the least accurately, with the LCG version performing worst. The implicit (consistent mass) versions improve with the mesh refinement to provide good agreement with the analytical solution. From Fig. 3 and Fig. 4 it is possible to see that all possess high symmetry and are in excellent agreement. From Table 1 it is possible to see that the explicit (lumped mass) variant of the LCG and continuous Galerkin method possess the fastest convergence time. The LCG version requires more time to complete due to the post-processing flux calculation requirement. The three versions of the LCG method that require the inversion of the elemental matrix require a fraction more time to complete, making them significantly faster than their corresponding continuous Galerkin counterpart.

7.2 The 2D transient flow past a cylinder at Re=100

This problem is a popular test case for validating the transient part of numerical schemes [29, 30, 32, 36, 40, 43]. Problem geometry and boundary conditions can be

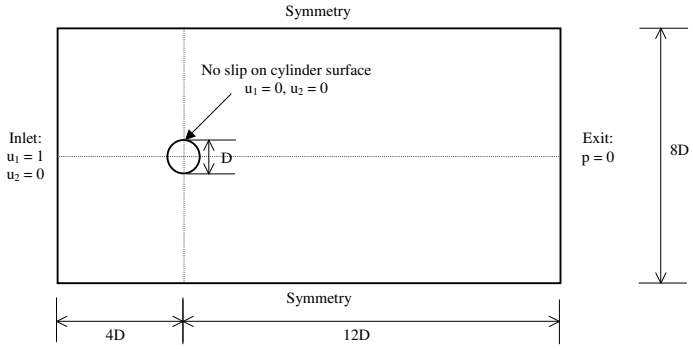


Figure 5: Transient flow past a cylinder. Geometry and boundary conditions.

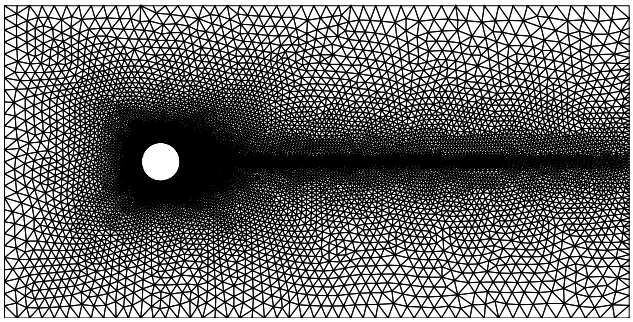
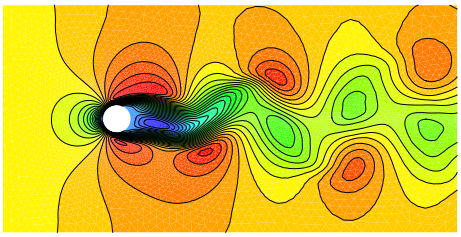
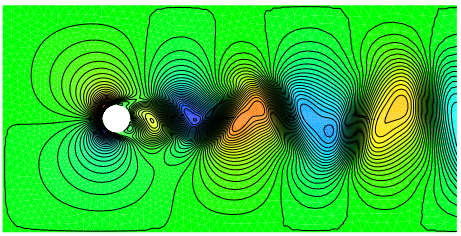


Figure 6: Transient flow past a cylinder. Unstructured mesh.

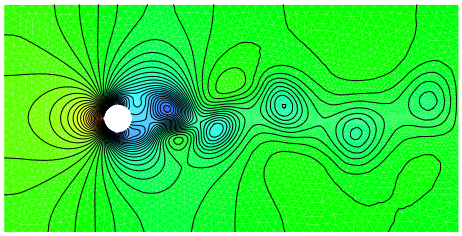
found in Fig. 5. The computational domain is $16D$ in length and $8D$ in width. The cylinder centroid is located at a distance of $4D$ from the inlet and along the horizontal centre line. At the inlet boundary, the horizontal and vertical velocity components are prescribed as unity and zero respectively. At the outlet boundary, the pressure is specified as zero. Slip conditions are imposed at the top and bottom of the domain.



(a) u_1 velocity contours



(b) u_2 velocity contours



(c) Pressure contours

Figure 7: Transient flow past a cylinder. Computed solution at a non-dimensional real time of 150.

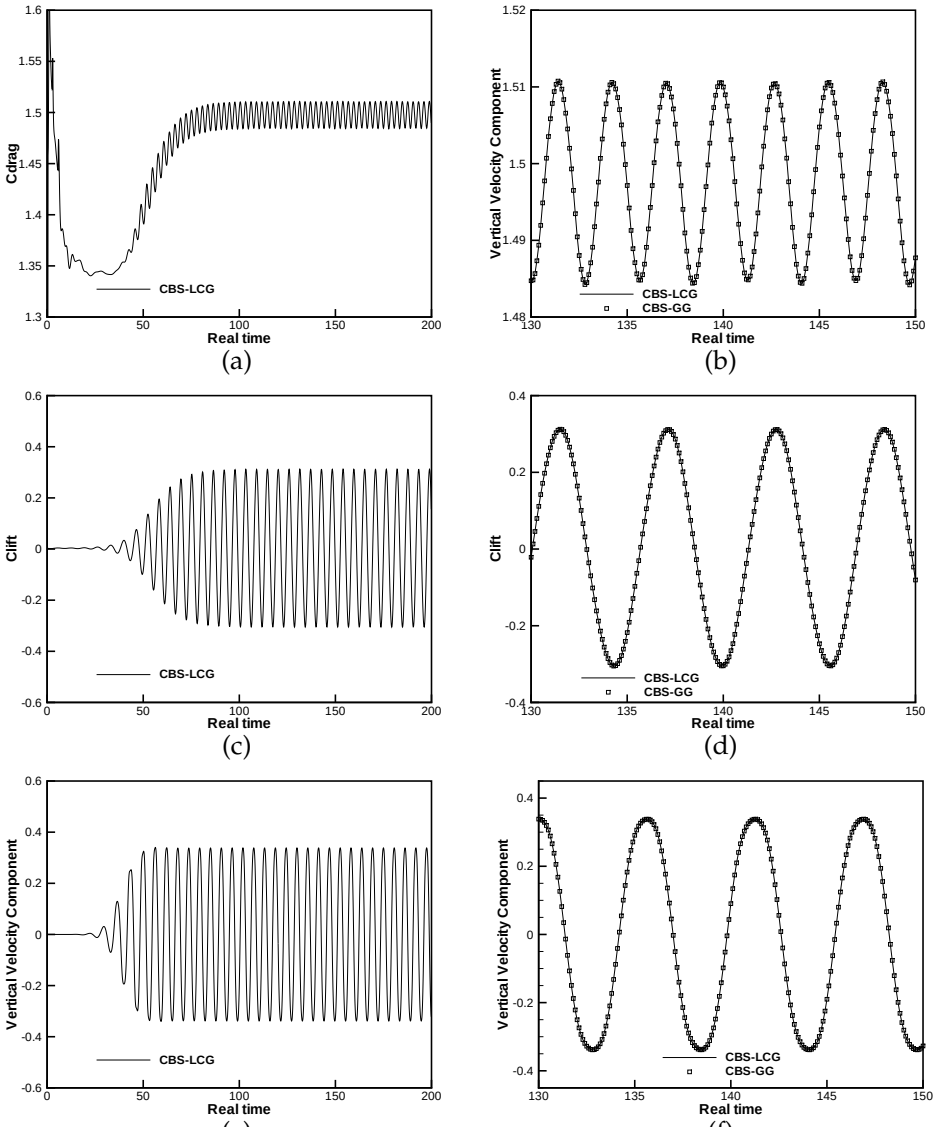


Figure 8: Transient flow past a cylinder. Computed coefficients of drag and lift, along with computed vertical velocity component at vertical midpoint at exit. All are plotted as a function of the non-dimensional, real time. (a) Full history of drag coefficient c_d. (b) Comparison of c_d with continuous Galerkin solution. (c) Full history of lift coefficient c_l. (d) Comparison of c_l with continuous Galerkin solution. (e) Vertical velocity component u_2 at central exit. (f) Comparison of u_2 at exit with continuous Galerkin solution.

No-slip conditions are prescribed on the cylinder. Initial horizontal velocity is unity and both the initial vertical velocity and pressure are zero.

A single unstructured mesh was used in this study and it is shown in Fig. 6. It contains 19,650 linear triangular elements and 9988 nodes. This mesh has been designed using previous knowledge on the formation of the unsteady wake and subsequent vortex shedding to refine specific regions [44,45]. The real time-step size utilized is 0.1

and simulations were carried out to a real non-dimensional time of 200.

Qualitative results are shown in Fig. 7. The contours of horizontal and vertical velocities along with pressure are shown for the real non-dimensional time of 150. All results are of high quality with no non-physical oscillations.

In order to perform a quantitative analysis of the results, the real-time history of both the lift and drag coefficients, along with the variation of the vertical velocity component at the vertical midpoint of the exit is shown in Fig. 8. It can be seen that the results of the LCG and the continuous (global) Galerkin methods are in excellent agreement, with negligible difference between the results of the two methods.

7.3 The 3D lid-driven cavity flow problem

The problem considered is the incompressible fluid flow in a lid-driven cavity. This problem is a standard test case and comparison of the velocity profiles can be made with well known benchmark data [46]. The cavity geometry is defined as a cube with unit length. No-slip boundary conditions are applied to the left, right and bottom walls and the flow inside the cavity is generated by the motion of the top surface, which travels in the horizontal direction. The two remaining walls are subjected to a

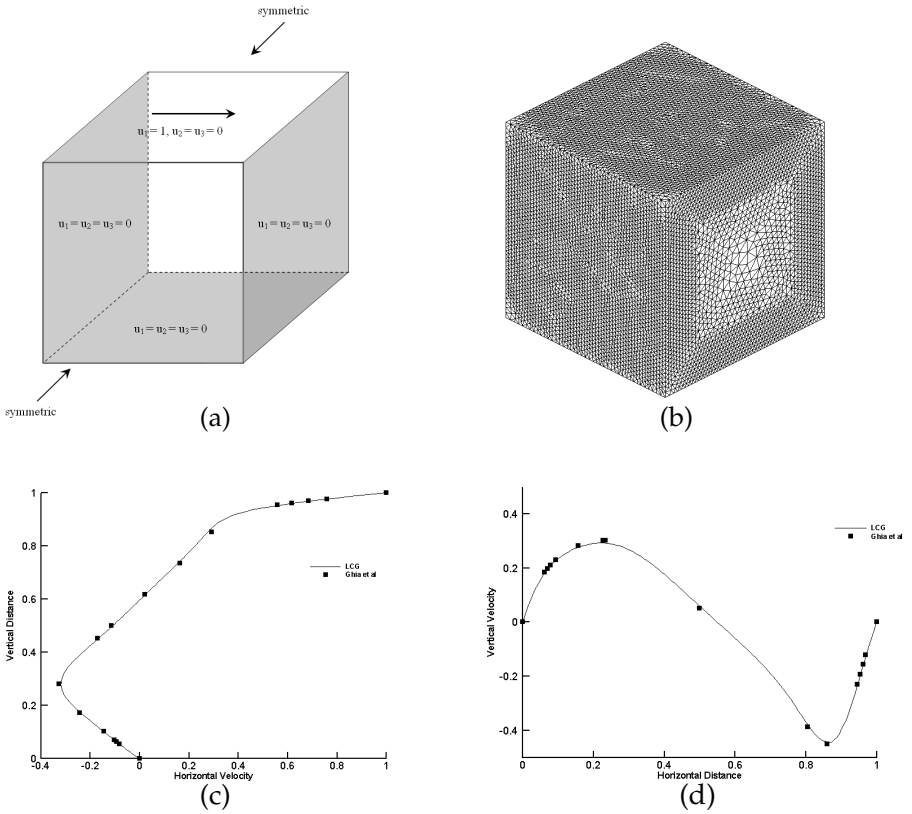


Figure 9: Flow in a 3D lid-driven cavity at Re=400. (a) Flow in a 3d lid-driven cavity. Geometry and boundary conditions. (b) Finite element mesh, Elements 514,297 Nodes 92,405. (c) Comparison of horizontal velocity distribution along mid-vertical line with Ghia et al.. (d) Comparison of vertical velocity distribution along mid-horizontal line with Ghia et al.

symmetrical boundary condition. The geometrical and boundary conditions are given in Fig. 9(a). The mesh Fig. 9(b) consisted of 514,297 linear tetrahedral elements and 92,405 nodes, with refinement around the solid walls.

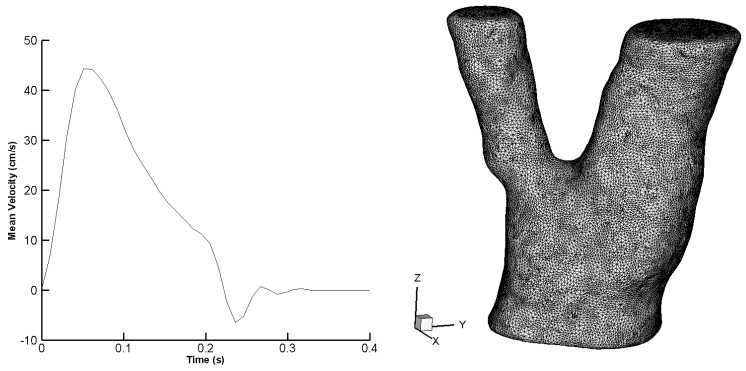
Fig. 9 shows the results obtained for the lid-driven cavity flow problem at a Reynolds number of 400. Fig. 9(c) shows the comparison of the u_1 velocity distribution along the mid-vertical line with the numerical benchmark solution of Ghia et al [46]. It can be seen that the corrected LCG results obtained are in excellent agreement with the benchmark solution. Fig. 9(d) shows the comparison of the u_3 velocity distribution along the mid-horizontal line with the benchmark solution and it is also in excellent agreement with the benchmark solution.

7.4 A 3D patient-specific carotid bifurcation

The unique geometry of each individual's carotid artery can influence the blood flow dynamics within the artery, and hence affect the risk and location of plaque build-up and atherosclerosis. A method that has both local and global conservation and that can capture the complexities of the flow dynamics and produce solutions in a timely manner would be of particular application in this instance.

Here, blood flow through a carotid bifurcation is investigated. Other studies are available on blood flow within the carotid artery [47–50]. Fluid dynamics analysis of this problem can be utilised to investigate and understand problems associated with the carotid artery. Although atherogenesis is linked to arterial wall shear stresses (WSS), the uneven distribution of the atherosclerotic lesions are not yet fully understood. Atherosclerosis is an inflammatory disease linked to the geometry of the arterial walls and is found in regions of complex hemodynamics, such as bifurcations.

The patient-specific carotid bifurcation geometry was constructed from a set of anonymous CT images provided by Singleton Hospital, Swansea, UK. The CT scans consisted of 390 axial slices from the thorax to the nasal passage. Data preparation, segmentation and mesh generation was performed using the data software AMIRA (Mercury Computer Systems, Chelmsford, MA, USA). A no-slip boundary condition was applied to the walls of the artery. A velocity profile was developed from the



(a) (b)

Figure 10: (a) Mean Velocity Profile. (b) Carotid bifurcation mesh approximately 1.2 million elements.

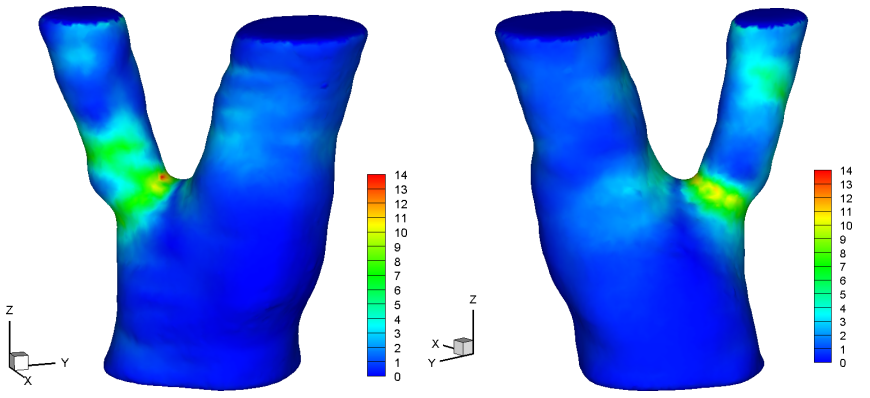


Figure 11: Wall Shear Stress distribution within the carotid bifurcation (a) Posterior, (b) Anterior.

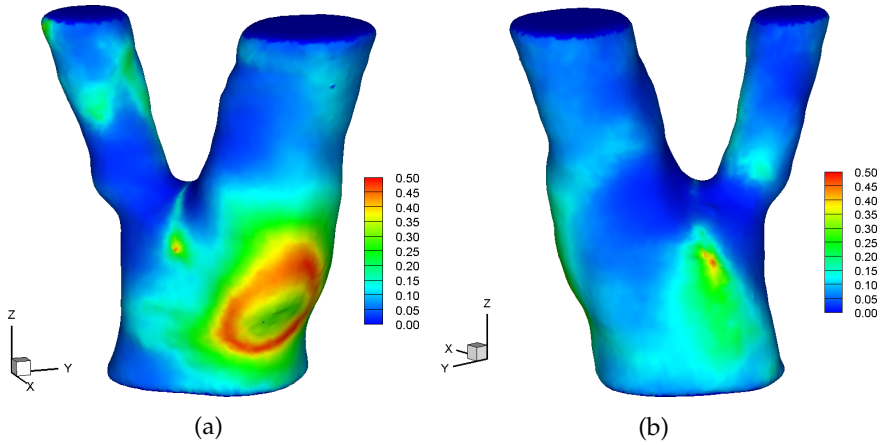


Figure 12: Oscillating Shear Index distribution within the carotid bifurcation (a) Posterior, (b) Anterior.

fully developed unsteady solution known as Womersley flow [48]. This is a 2D linear viscous solution of the Navier-Stokes equations in cylindrical tubes. A detailed description can be found in [51].

The blood density and dynamic viscosity were taken to be ρ=1.0 g/cm^3 and ν=0.035 g/cm s. The Womersley velocity profile was discretized into 40 (real) time steps. The peak velocity was set to 66 cm/s while the peak mean velocity was 45 cm/s Fig. 10(a). The inlet velocity profile was determined and mapped to the surface of the inlet. Based upon the inlet mass flow rate the velocity profiles for the interior and exterior carotid arteries was determined and applied, with the outlet flow split between the interior (60% flow rate of inlet) and the exterior (40% flow rate of inlet) carotid artery.

One of the primary indicators for the location of atheroma is the presence of low or oscillating shear stress on the arterial wall. The time averaged wall shear stress distribution is shown in Fig. 11. A low WSS below 0.5Pa is believed to stimulate an atherogenic phenotype. A high WSS is found at the apex, with a localised region expe-

riencing a WSS greater than 10Pa. This region is closer to the inner wall of the external carotid artery which is in agreement with the results of Younis *et al* [48]. The oscillatory shear index (OSI) [52, 53] is used to quantify the transient shear stress dynamics experienced by the endothelial cells. The oscillatory shear index is defined by

$$OSI = \frac{1}{2}\Big(1 - \frac{\tau_{mean}}{\tau_{abs}}\Big), \tag{7.2}$$

and will always lie within the range $0 \leq OSI \leq 0.5$. The oscillatory shear index distribution for the patient specific model used is shown in Fig. 12. It can be seen that a region within the Common Carotid Artery (CCA) displays a high OSI value, indicating a region where atherosclerotic plaque is predicted to form.

8 Conclusions and future work

This paper summarises the LCG method and its application to fluid dynamics problems. Numerical examples demonstrate the ability of the LCG method to provide equivalent solutions to the continuous Galerkin method. The ability to have a discontinuous method without the disadvantages of a true DG method cannot be understated. A timely solution to a patient-specific problem is highly desirable in future application as a diagnostic tool. Although the LCG method has been developed and employed on different fluid dynamics problems it has not yet reached its full potential. Although the method allows for discontinuity capturing at the element interfaces, this has yet to be implemented. Extension of the method to problems involving fluid-structure interaction is seen as the next step. Integration of the method into multi-dimensional models and higher order elements is also called for.

References

[1] P. NITHIARASU, *A simple locally conservative Galerkin (LCG) finite element method for transient conservation equations*, Numer. Heat Transfer, Part B Fundamentals, 46 (2004), pp. 357–370.

[2] A. MIZUKAMI, *A mixed finite element method for boundary flux computation*, Computer Methods in Applied Mechanics and Engineering, 57 (1986), pp. 239–243.

[3] P. M. GRESHO, R. L. LEE, R. L. SANI, M. K. MASLANIK AND B. E. EATON, *The consistent galerkin fem for computing derived boundary quantities in thermal and/or fluids problems*, Int. J. Numer. Meth. Fl., 7 (1987), pp. 371–394.

[4] T. J. R. HUGHES, L. P. FRANCA, I. HARARI, M. MALLET, F. SHAKIB AND T. E. SPELCE, *Finite element method for high–speed flows:consistent calculation of boundary flux*, Paper No.AIAA-87-0556, AIAA 25th Aerospace Sciences Meeting, Reno, Nevada, 1987.

[5] M. OSHIMA, T. J. R. HUGHES AND K. JANSEN, *Consistent finite element calculation of boundary and internal fluxes*, Int. J. Comput. Fluid D., 9 (1998), pp. 227.

[6] T. J. R. HUGHES, G. ENGEL, L. MAZZEI AND M. G. LARSON, *The continuous Galerkin method is locally conservative*, J. Comput. Phys., 163 (2000), pp. 467–488.

[7] T. J. R. HUGHES AND G. N. WELLS, *Conservation properties for the galerkin and stabilised forms of the advection-diffusion and incompressible Navier-Stokes equations*, Comput. Method. Appl. M., 194 (2005), pp. 1141–1159.
[8] J. H. BRAMBLE, J. E. PASCIAK AND A. H. SCHATZ, *The construction of preconditioners for elliptic problems by substructuring.* I, Math. Comput., 47 (1986), pp. 103–134.
[9] I. G. GRAHAM, P. O. LECHNER AND R. SCHEICHL, *Domain decomposition for multiscale* PDE*s*, Numer. Math., 106 (2007), pp. 589–626.
[10] M. PICASSO, J. RAPPAZ AND V. REZZONICO, *Multiscale algorithm with patches of finite elements*, Comm. Numer. Meth. Eng., 24 (2008), pp. 477–491.
[11] W. H. REED AND T. R. HILL, *Triangular mesh methods for the neutron transport equation*, Technical report LA-UR-73-479, 1973.
[12] G. CHAVENT AND G. SALZANO, *A finite element method for the 1d water flooding problem with gravity*, J. Comput. Phys., 45 (1982), pp. 307–344.
[13] G. CHAVENT AND B. COCKBURN, *The local projection p^0p^1 - discontinuous Galerkin finite element method for scalar conservation laws*, RAIRO Mod*é*l. Math. Anal.Numer., 23 (1989), pp. 565–592.
[14] B. COCKBURN AND C-W. SHU, *TVB Runge-Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws II: General framework*, Math. Comp., 52 (1989), pp. 411–435.
[15] B. COCKBURN, S. Y. LIN AND C-W. SHU, *TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: One dimensional systems*, J. Comput. Phys., 84 (1989), pp. 90–113.
[16] B. COCKBURN, S. HOU AND C-W. SHU, *TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: The multidimensional case*, Math. Comp., 54 (1989), pp. 545–581.
[17] B. COCKBURN AND C-W. SHU, *The Runge-Kutta local projection p^1 discontinuous Galerkin finite element method for scalar conservation laws*, RAIRO Mod*é*l. Math. Anal. Numer., 25 (1991), pp. 337–361.
[18] B. COCKBURN AND C-W. SHU, *The Runge-Kutta discontinuous Galerkin finite element method for conservation laws V: Multidimensional systems*, J. Comput. Phys., 141 (1998), pp. 337–361.
[19] B. COCKBURN, *Disconuous Galerkin methods for convection dominated problems*, High-Order Methods for Computational Science and Engineering, Berlin, Springer-Verlag, 1999.
[20] B. COCKBURN, G. E. KARNIADAKIS AND C-W. SHU (EDS.), The Development of Discontinuous Galerkin Methods. Theory, Computation and Applications. Lecture Notes in Computational Science and Engineering, Springer, Berlin, 2000.
[21] O. C. ZIENKIEWICZ, R. L. TAYLOR, S. J. SHERWIN AND J. PEIRO, *On discontinuous Galerkin methods*, Int. J. Numer. Meth. Eng., 58 (2003), pp. 1119–1148.
[22] B. COCKBURN, G. KANSCHAT AND D. SCHÖTZAU, *The local discontinuous galerkin methods for linear incompressible flow: A review*, Computer and Fluids (special issue: Residual based methods and discontinuous galerkin schemes), Computers and Fluids, 34 (2005), pp. 491–506.
[23] R. BISWAS, K. D. DEVINE AND J. E. FLAHERTY, *Parallel, adaptive finite element methods for conservation laws*, Applied Numerical Mathematics, 14 (1994), pp. 255–283.
[24] T. EKEVID AND N-E. WIBERG, *A comparison of parallel implementation of explicit* DG *and central difference method*, Commun. Numer. Meth. Eng., 18 (2002), pp. 585–597.
[25] C. E. BAUMANN AND J. T. ODEN, *A discontinuous hp finite element method for convection-*

diffusion problems, Comput. Method. Appl. M., 175 (1999), pp. 311–341.

[26] THOMAS J. R. HUGHES, GUGLIELMO SCOVAZZI, PAVEL B. BOCHEV AND ANNALISA BUFFA, *A multiscale discontinuous Galerkin method with the compuational structure of a continuous Galerkin method*, Comput. Method. Appl. M., 195 (2006), pp. 2761–2787.

[27] C. G. THOMAS AND P. NITHIARASU, *An element-wise, locally conservative Galerkin (*LCG*) method for diffusion and convection-diffusion problems*, Int. J. Numer. Meth. Eng., 73(5) (2008), pp. 642–664.

[28] C. G. THOMAS, P. NITHIARASU AND R. L. T. BEVAN, *The locally conservative Galerkin (LCG) method for solving incompressible Navier-Stokes equations*, Int. J. Numer. Meth. Fl., 57 (2008), pp. 1771–1792.

[29] P. NITHIARASU, *An efficient artificial compressibility (AC) scheme based on the charcteristic based split (CBS) method for incompressible flows*, Int. J. Numer. Meth. Eng., 56 (2003), pp. 1815–1845.

[30] P. NITHIARASU, J. S. MATHUR, N. P. WEATHERILL AND K. MORGAN, *Three dimensional incompressible flow calculations using the characteristic based split (CBS) scheme*, Int. J. Numer. Meth. Fl., 44 (2004), pp. 1207–1229.

[31] A. G. MALAN, R. W. LEWIS AND P. NITHIARASU, *An improved unsteady, unstructured, artificial compressibility, finite volume scheme for viscous incompressible flows: Part I. Theory and implementation*, Int. J. Numer. Meth. Eng., 54 (2002), pp. 695–714.

[32] O. C. ZIENKIEWICZ, R. L. TAYLOR AND P. NITHIARASU, The Finite Element Method for Fluid Dynamics, Elsevier Butterworth Heinemann, London, 2005.

[33] D. DRIKAKIS, P. A. GOVATSOS AND D. E. PAPANTONIS, *A characteristic-based method for incompressible flows*, Int. J. Numer. Meth. Fl., 19 (1994), pp. 667–685.

[34] O. C. ZIENKIEWICZ AND R. CODINA, *A general algorithm for compressible flow - part i. the split characteristic-based scheme*, Int. J. Numer. Meth. Fl., 20 (1995), pp. 869–885.

[35] O. C. ZIENKIEWICZ, K. MORGAN, B. V. K. S. SAI, R. CODINA AND M. VAZQUEZ, *A general algorithm for compressible flow - part ii. tests on the explicit form*, Int. J. Numer. Meth. Fl., 20 (1995), pp. 887–913.

[36] P. NITHIARASU AND O. C. ZIENKIEWICZ, *Analysis of an explicit and matrix free fractional step method for incompressible flows*, Comput. Method. Appl. M., 195 (2006), pp. 5537–5551.

[37] P. NITHIARASU, *A matrix free fractional step method for static and dynamic incompressible solid mechanics*, Int. J. Comp. Meth. Eng. Sci and Mech., 7 (2006), pp. 369–380.

[38] P. NITHIARASU, R. CODINA AND O. C. ZIENKIEWICZ, *The characteristic based split scheme — a unified approach to fluid dynamics*, Int. J. Numer. Meth. Eng., 66 (2006), pp. 1514–1546.

[39] R. LÖHNER, K. MORGAN AND O. C. ZIENKIEWICZ, *The solution of non-linear hyperbolic equation systems by the finite element method*, Int. J. Numer. Meth. Fl., 4 (1984), pp. 1043–1063.

[40] A. G. MALAN, R. W. LEWIS AND P. NITHIARASU, *An improved unsteady, unstructured, artificial compressibility, finite volume scheme for viscous incompressible flows: Part II. Application*, Int. J. Numer. Meth. Eng., 54 (2002), pp. 715–729.

[41] A. L. GAITONDE, *A dual-time method for two-dimensional unsteady incompressible flow calculations*, Int. J. Numer. Meth. Eng., 41 (1998), pp. 1153–1166.

[42] J. P. HOLMAN, Heat Transfer, McGraw-Hill Publishers, Singapore, 7th edition, 1992.

[43] P. A. B. DESAMPAIO, P. R. M. LYRA, K. MORGAN AND N. P. WEATHERILL, *Petrov Galerkin solutions of the incompressible Navier-Stokes equation in primitive variables with adaptive remeshing*, Comput. Method. Appl. M., 106 (1992), pp. 143–178.

[44] R. C. PANTON, Incompressible Flow, Wiley, New York, 1984.

[45] H. SCHLICHTING, Boundary Layer Theory, 6th ed., McGraw-HillPublications, 1968.

[46] U. GHIA, K. N. GHIA AND C. T. SHIN, *High-re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method*, J. Comput. Phys., 48 (1982), pp. 387–411.

[47] P. PAPATHANASOPOULOU, S. ZHAO, U. KÖHLER, M. B. ROBERTSON, Q. LONG AND P. HOSKINS, MRI *measurement of time-resolved wall shear stress vectors in a carotid bifurcation model, and comparison with* CFD *predictions*, J. Magn. Reson. Im., 17 (2003), pp. 153–162.

[48] H. F. YOUNIS, M. R. KAAZEMPUR-MOFRAD, R. C. CHAN, A. G. ISASI, D. P. HINTON, A. H. CHAU, L. A. KIM AND R. D. KAMM, *Hemodynamics and wall mechanics in human carotid bifurcation and its consequences for atherogenesis – investigation of inter–individual variation*, Biomechanics and Modeling in Mechanobiology, 3 (2004), pp. 17–32.

[49] M. R. KAAZEMPUR-MOFRAD, A. G. ISASI, H. F. YOUNIS, R. C. CHAN, D. P. HINTON, G. SUKHOVA, G. M. LAMURAGLIA, R. T. LEE AND R. D. KAMM, *Characterization of the atherosclerotic carotid bifurcation using* MRI *finite element modeling and histology*, Ann. Biomed. Eng., 32 (2004), pp. 932–946.

[50] K. T. NGUYEN, C. D. CLARK, T. J. CHANCELLOR AND D. V. PAPAVASSILIOU, *Carotid geometry effects on blood flow and on risk for vascular disease*, J. Biomech., 41 (2008), pp. 11–19.

[51] J. P. MYNARD AND P. NITHIARASU, *A 1d arterial blood flow model incorporating ventricular pressure, aortic valve and regional coronary flow using the locally conservative galerkin (*LCG*) method*, Commun. Numer. Meth. En., 24 (2008), pp. 367–417.

[52] D. N. KU, D. P. GIDDENS, C. K. ZARINS AND S. GLAGOV, *Pulsatile flow and atherosclerosis in the human carotid bifurcation – positive correlation between plaque location and low and oscillating shear-stress*, Atherosclerosis, 5(3) (1985), pp. 293–302.

[53] A. D. JEAYS, P. V. LAWFORD, R. GILLOTT, P. SPENCER, D. C. BARBER, K. D. BARDHAN AND D. R. HOSE, *Characterisation of the haemodynamics of the superior mesenteric artery*, J. Biomech., 40 (2007), pp. 1916–1926.