INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2003; **42**:1135–1154 (DOI: 10.1002/fld.584)

Solution adaptive grids applied to low Reynolds number flow

G. de With[1,*,†], A. E. Holdø[1] and T. A. Huld[2]

[1]*Department of Aerospace Automotive and Mechanical Engineering, CFD Research Group, University of Hertfordshire, Hatfield, U.K.*
[2]*Institute for Systems, Informatics and Safety, Joint Research Centre, Ispra (Va) 21020, Italy*

SUMMARY

A numerical study has been undertaken to investigate the use of a solution adaptive grid for flow around a cylinder in the laminar flow regime. The main purpose of this work is twofold. The first aim is to investigate the suitability of a grid adaptation algorithm and the reduction in mesh size that can be obtained. Secondly, the uniform asymmetric flow structures are ideal to validate the mesh structures due to mesh refinement and consequently the selected refinement criteria. The refinement variable used in this work is a product of the rate of strain and the mesh cell size, and contains two variables C_m and C_{str} which determine the order of each term. By altering the order of either one of these terms the refinement behaviour can be modified. Copyright © 2003 John Wiley & Sons, Ltd.

1. INTRODUCTION

The present study deals with the simulation of vortex shedding behind a circular cylinder in the laminar flow regime. Since the early works of Bénard [1] and Von Karman [2] the phenomenon of vortex shedding behind cylinders at low Reynolds numbers has attracted considerable attention. Numerous investigations have provided much detailed information on various aspects of this problem. For example, the longitudinal and lateral spacing of vortices, the time frequency of the vortex pattern with Reynolds numbers and vortex street drag have been widely studied by Kovasznay [3], Roshko [4], Abernathy and Kronauer [5] and Bearman [6].

Early work on the numerical modelling of vortex shedding was presented by Son and Hanratty [7], since then the study of flow behind bluff bodies by means of computational fluid dynamics (CFD) models has provided a further understanding of the mechanisms by which vortex shedding takes place [8–13]. CFD investigations have shown a solid agreement between numerical and experimental data; however, Posdziech and Grundmann [8] have demonstrated that mesh size and mesh domain have to be chosen sufficiently large to obtain accurate

*Correspondence to: G. de With, CFD Research Group, Department of Aeronautical, Civil and Mechanical Engineering, University of Hertfordshire, Hatfield Campus, College Lane, Hatfield, Herts AL10 9AB, U.K.
†E-mail: G.de-With@herts.ac.uk

Received 14 August 2002
Revised 20 March 2003

and reliable numerical data. Numerical requirements of this kind are a severe limitation in an environment where computational resources are scarce. In order to overcome the mesh requirements and still obtain results of good quality the present work investigates the suitability of a dynamic grid adaptation (DGA) algorithm in order to reduce the grid requirements. Frequently meshes are generated where due to mesh construction methods the resolution is high in regions where gradients are not significant. Consequently, such meshes have a large number of redundant cells. Conversely, due to time variations in the computed flow field, there will be regions where the mesh resolution is insufficient.

Although, there have been many developments related to adaptive methods and adaptive grid strategies [14–17], comparatively little work is reported on grid adaptation with unsteady flows. The aim of this work is to present a refinement variable suitable for vortex shedding behind a cylinder in the laminar flow regime. The Reynolds numbers which are investigated range from 50 to 75, under these conditions the flow is expected unsteady, laminar and two-dimensional. The refinement variable proposed herein is a product of local rate of strain and element mesh size. The refinement variable has included two variables with which the magnitude of both terms can be altered to accomplish a unique refinement behaviour.

In this work the REACFLOW CFD code has been used. Transient simulations are carried out with both conventional pre-defined mesh and with DGA. For validation of the numerical data a comparison is made with experimental data of Paranthoën *et al.* [18]. The Reynolds numbers under investigation are associated with an unsteady asymmetric flow field. Nevertheless, for Reynolds numbers in the order of 50 the onset of vortex shedding due to an absolute instability is very weak. As integral part of the validation of the refinement variable it will be investigated what refinement conditions are most suitable to initiate vortex shedding in the simulation.

2. SOLUTION ADAPTIVE MESH

One method to reduce the numerical error is by reducing the size of the discretization elements. However, an excessive increase in mesh resolution would lead to insurmountable demand for computational power. As a consequence a mesh sensitivity analysis is required to find consensus in numerical error and computational demand. This requires the user to estimate beforehand the regions of interest. Research has shown that the flow prediction is sensitive to the mesh resolution, this numerical behaviour indicates a need for dynamically generated mesh, which adapt the mesh to the flow field to cover interesting flow phenomena in high detail. A well developed and robust DGA algorithm can lead to a reduction in both numerical error and computational requirements. This appealing feature has encouraged the CFD society to develop a whole range of algorithms for DGA of different applicability and complexity.

2.1. Refinement strategies

In recent years a whole series of strategies have been developed, a small number of these strategies are preprocessing based and focus on a geometry-based mesh refinement. For CFD purposes the majority of the strategies is focusing on a solution-based mesh refinement. This work will deal with the solution-based mesh refinement. Four approaches can be distinguished:

R-refinement (grid movement or grid relocation): In this approach a basically fixed number of grid points is used, but the point positions are moved in such a way so that the grid points concentrate in critical spatial areas [19]. R-refinement plays an important and growing role in CFD, especially where the fluid interacts with moving walls (as in piston engines or structures yielding to an overpressure).

H-refinement: In this approach a variable number of grid points is used. Points are added to, or removed from, the grid according to the local requirements, without changing the positions of the other grid points. As a result the grid is locally refined or coarsened [16].

M-refinement: A third family of refinement strategies is based on the existence of automatic grid generators. The grid generator is used in combination with an error indicator based on the present discretization and solution to remesh the computational domain either globally or locally, in order to produce a more suitable discretization [20–22]. The main advantages offered by adaptive remeshing methods are twofold. The possibility of stretching elements when adapting features that are of lower dimensionality than the problem at hand, secondly the ability to accommodate in a straightforward manner problems with moving bodies or free surfaces.

P-refinement: In this approach a principally fixed grid is used, but an adaptive solution is obtained by locally varying the order of spatial discretization. Methods which do accomplish the addition of higher-order shape functions are either the conventional polynomials [23], spectral element functions [24] or hierarchical shape functions [25]. P-refining methods are increasingly used in finite-element methods, often together with h-refining methods [16]. The refinement strategy being used in the REACFLOW code is the so-called h-refinement, where grid points are added and removed based on the error estimation and the error bounds, defined at the start of the simulation. The refinement of the mesh always takes place across the longest edge of the triangular element, which ensures that triangles with very high aspect ratio are not formed.

Conformal and non-conformal grid adaptation: The refinement strategy is based on a conformal insertion of new grid points. When a new grid point is inserted into the grid neighbouring elements and/or control volumes may also be changed. If the change in the elements is such that all elements will remain the same type, i.e. elements remain of triangular shape, the refinement is called conformal. Such a refinement is shown in the left-hand side of Figure 1 for a two-dimensional triangle-based grid. When the new grid point is inserted the two neighbouring triangles are both divided so that the new triangles all have the new grid point as a vertex. In this way, all triangular elements will continue to have exactly three neighbouring points. In the non-conformal refinement shown on the right-hand side of Figure 1, only one of the neighbouring elements is divided. In the other element the new grid point appears as a ‘hanging node’ somewhere along the side of the element.

Among the two alternatives, it is not entirely clear what is the most advantageous [26]. The non-conformal methods seem to be able to concentrate the new grid points better in the area where they are needed. On the other hand it will introduce some extra complexity into the code. Huld and Wilkening [26] mentioned that in a node-based system, where the number of neighbours to a given volume is already variable, it is an important simplification that at least the number of neighbours to each element will be known in advance.

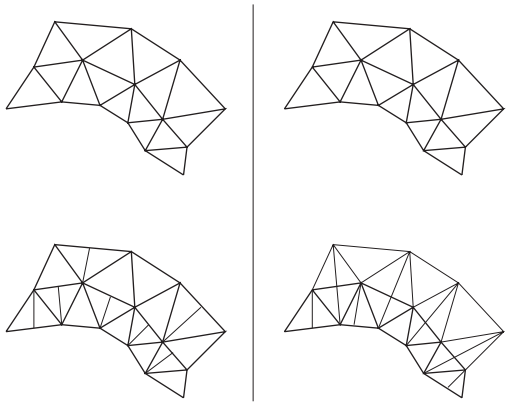

Figure 1. Non-conformal and conformal grid adaptation.

2.2. REACFLOW DGA algorithm

The DGA algorithm implemented in the REACFLOW CFD code has adopted the h-refinement technique. The algorithm uses the longest edge bisection method [27–29] and refinement is based on local refinement criteria. The DGA algorithm as implemented in the REACFLOW CFD code consists of six main steps, and are listed below:

1. Identify elements for mesh refinement that satisfy the refinement criterion.
2. Neighbouring elements to the marked element are also selected for refinement.
3. Process elements marked for refinement. To avoid constructing elements with very small solid angles, an element is always divided by bisecting the longest edge of that element.
4. Repeat the search for the longest edge starting from the originally defined edge and bisect the edge found, until the process arrives at the originally defined edge. In this way, the refinement will be performed in the entire region from the edge found in the first step back to the edge where the refinement process was started.
5. Update the other geometrical features, such as the segments of the surfaces of the control volumes, the boundary facets, etc.
6. Compute the state in the new control volumes. At present the element-based variables in the new elements are simply the ones in the original element.

An illustration of the mesh refinement is presented in Figure 2. A special remark should be made for mesh refinement in the vicinity of the wall. In the region adjacent to the boundary, the wall segments are assumed to be straight and new nodes on the wall are positioned by interpolating between both existing wall nodes. No nodes are removed from the initial mesh to ensure that the geometry remains in tact, even when there is minimum flow development along the edges of the model.

The main ingredients in the DGA algorithm, are the method for error indication and the refinement variable. The selection of a suitable error indication method and refinement variable is crucial for designing an effective and powerful refinement algorithm.

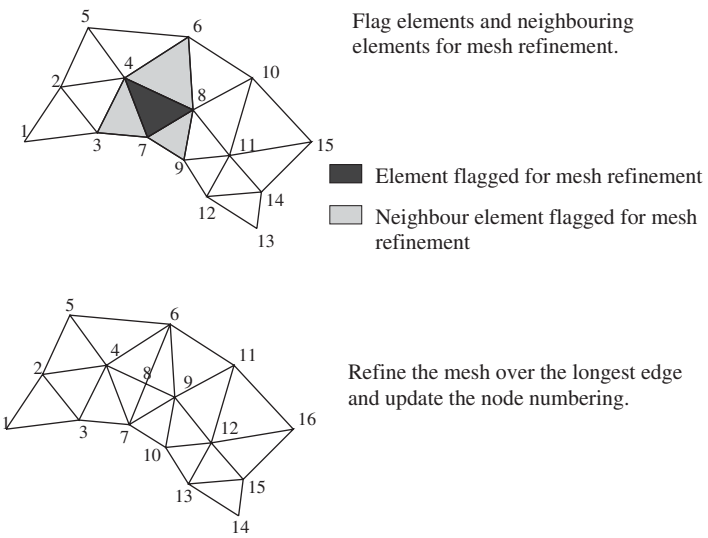

Figure 2. Mesh adaptation method.

2.2.1. Method for error indication. In flow simulations, the regions of interest will often include those with strong gradients. The issue of grid refinement is to modify the grid and focus on the regions of interest.

The information about the regions where the grid should be refined or coarsened ideally should be provided by an error estimator. Error estimation for fluid flow calculations is not an easy task. The Navier–Stokes equations are a coupled, non-linear system and errors present in any one of these fields in general will effect the solutions of all others, in a non-linear manner difficult to describe accurately. The discretization error describes the deviation from the analytical solution of the set of differential equations, but is not directly accessible. However, it is possible to construct approximations to this error.

The method for error indication implemented is an estimation of the local interpolation error. For a given variable of interest, q, an interpolated value for a grid point x_i can be defined. This value $\hat{q}_i$ can be defined in various ways. However, in the present work, the arithmetical mean is defined over the neighbour grid points:

$$\hat{q}_i \equiv \frac{1}{N} \sum_{j=1}^{N} q_j \tag{1}$$

where q_j are the values of the variable at the jth of the N neighbours of the grid point x_i. The interpolated value may now be compared with the actual value in that grid point, q_i. If the absolute value of the difference

$$e_i \equiv |q_i - \hat{q}_i| \tag{2}$$

is greater than a predefined value, a new grid point will be inserted in the vicinity.

An error estimation of this kind assumes the solution to be smooth and will lead to mesh refinement in the region where there is a variation in the refinement variable. An

advantage of the interpolation method, in comparison to other methods, is that all neighbour elements are included in the determination of a new node, leading to a smoother mesh refinement.

2.2.2. Refinement variable. The determination of an appropriate refinement variable to approximate the discretization error depends very much on the physical problem investigated. Widely used refinement variables in this respect are the density, pressure and velocity gradients, as well as the turbulent viscosity. However, the discretization error describing the deviation from the analytical solution of the set of differential equations is not directly accessible. Therefore, the refinement variable is a compromise between the quality of the error indication and the computational complexity of the refinement variable.

Previous work by De With *et al.* [30] concluded that a modified turbulent viscosity is a suitable refinement variable for high Reynolds number flows. In the laminar case which is the subject of the present work, the important quantities are rate of strain $|\bar{S}|$ and mesh cell size Δ. The refinement variable is a function of $|\bar{S}|$ and Δ, as shown in Equation (3). The refinement variable can be considered as a mesh-Reynolds number, in case C_{m} and C_{str} are set to 1, as shown in Equation (5). The relation holds in a simulation where the dynamic viscosity μ is constant, which is the case in an isothermal and incompressible flow.

$$q_i = \rho \Delta^{2C_{\mathrm{m}}} |\bar{S}|^{C_{\mathrm{str}}} \tag{3}$$

$$Re_{\Delta} = \frac{\rho \Delta^2 |\bar{S}|}{\mu} \tag{4}$$

$$q_i \propto Re_{\Delta} \leftrightarrow \rho \Delta^{2C_{\mathrm{m}}} |\bar{S}|^{C_{\mathrm{str}}} \propto \frac{\rho \Delta^2 |\bar{S}|}{\mu} \tag{5}$$

By varying the importance or weight of either the cell size Δ or the strain rate $|\bar{S}|$ another refinement behaviour can be established. The weight for the cell size and the strain rate is controlled by the two variables C_{m} and C_{str}, respectively. The variables C_{m} and C_{str} have to be defined at the start of the simulation, and are constant throughout the numerical domain. A higher weight of the mesh cell size will lead to a more solution independent mesh refinement, while an increase in weight of the strain rate will lead to a mesh refinement in the regions where high velocity gradients do occur. As suggested in the work of De With *et al.* [30] the subgrid length scale Δ is calculated in accordance with equation $\Delta = \sqrt{A_{\mathrm{tr}}}$.

3. REACFLOW CFD SOLVER

For the simulations carried out in this work the CFD code REACFLOW [31] has been utilized. REACFLOW calculates the solution to the multicomponent, variable-density incompressible Navier–Stokes equations. The numerical method used is similar to the predictor–corrector method of Chorin [32].

The predictor–corrector method is based on the idea of performing the time integration by means of a fractional step. The convective part of the equations is integrated in the first substep, whereas in the second substep a velocity correction is found in the form of a

pressure gradient so as to make the end-of-step velocity field divergenceless ($\partial u_i/\partial x_i = 0$). The pressure p is discretized in a vertex-centred fashion, where each pressure value is defined on a *control volume*, delimited by the medians in all the triangles surrounding the given vertex. This choice of discretization has been made to suppress the 'checkerboard' instability. The pressure gradient would then be known for each triangle allowing for an easy construction of the diffusion-like stiffness matrix associated with the pressure equation to be solved at each time level.

The effects associated with the pressure propagate across the computational domain instantaneously. To account for this source of stiffness in the equations, an implicit discretization of these terms is mandatory. Assuming an explicit time integration for the other terms, the governing equations can be obtained by spatial integration over the triangular control volume Ω_i.

$$|\Omega_i| \frac{\rho_{\gamma i}^{n+1} - \rho_{\gamma i}^{n}}{\Delta t} + \int_{\Omega_i} [u \cdot \nabla \rho_\gamma] \, \mathrm{d}\Omega = 0$$

$$|\Omega_i| \frac{u_i^{n+1} - u_i^{n}}{\Delta t} + \int_{\Omega_i} [(u \cdot \nabla)u]^n \, \mathrm{d}v + \int_{\Omega_i} \left[\frac{\nabla p}{\rho}\right]^{n+1} \mathrm{d}\Omega = \int_{\Omega_i} \left[\frac{\nabla \cdot \tau + \rho g}{\rho}\right]^{n} \mathrm{d}\Omega \tag{6}$$

$$|\Omega_i| \frac{e_i^{n+1} - e_i^{n}}{\Delta t} + \int_{\Omega_i} [u \cdot \nabla e]^n \, \mathrm{d}\Omega = - \int_{\Omega_i} \left[\frac{\nabla \cdot q}{\rho}\right]^{n} \mathrm{d}\Omega$$

$|\Omega_i|$ represents the measure of the triangular control volume Ω_i, and the quantities $\rho_{\gamma i}$, u_i and e_i denote the volume-averaged primitive quantities on Ω_i.

The advective terms of the form $(u \cdot \nabla)W$ (W being a general primitive variable) are discretized in a classical fully upwind manner. One can make use of Green's theorem to write

$$\int_{\Omega_i} [u \cdot \nabla W]^n \, \mathrm{d}\Omega \approx u_i^n \cdot \int_{\Omega_i} [\nabla W] \, \mathrm{d}\Omega = u_i^n \cdot \oint_{\partial\Omega_i} \eta W^n \, \mathrm{d}\Sigma \tag{7}$$

where $\partial\Omega_i$ denotes the contour of the triangular control volume Ω_i, $\mathrm{d}\Sigma$ represents an elemental arc of $\partial\Omega_i$ and η is the normal unit vector outwardly oriented of $\partial\Omega_i$. According to the upwind method, the value of the primitive variable W on the control volume contour is approximated by W_i (corresponding to the current control volume) if the scalar product $u \cdot \eta$ is positive, or by W_j (corresponding to the neighbour control volume Ω_j) if the scalar product $u \cdot \eta$ is negative. The fully upwind discretization scheme provides only first-order spatial accuracy for the convective term, if combined with a piecewise linear description of the primitive variables within the triangular control volumes.

The above-described technique permits the explicit integration of the mass and energy equations to obtain ρ_γ^{n+1} and e^{n+1} from the values of the primitive variables at time n. The convective transport is made across the element sides. To make the numerical procedure complete, the pressure field p^{n+1} must now be determined so that the end-of-step velocity field verifies the solenoidal condition:

$$\nabla \cdot u^{n+1} = 0 \tag{8}$$

Denoting ξ_k the classical finite-element shape function corresponding to linear three-node elements for node k, and by l(i) the nodes belonging the triangular control volume Ω_i, the integral term in equation involving the pressure gradient can be approximated as

$$\int_{\Omega_i} \left[\frac{\nabla p}{\rho}\right]^{n+1} \mathrm{d}\Omega = \frac{|\Omega_i|}{\rho_i^{n+1}} \Sigma_{k\in l(i)} \nabla \xi_k \, p_k^{n+1} \tag{9}$$

Introducing Equation (9) into the discretized momentum equation yields

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{|\Omega_i|} \int [(u\cdot\nabla)u]^n \,\mathrm{d}\Omega + \frac{\Delta t}{|\Omega_i|} \int_{\Omega_i} \left[\frac{\nabla\cdot\tau + \rho g}{\rho}\right]^n \mathrm{d}\Omega$$
$$-\Delta t \frac{1}{\rho_i^{n+1}} \Sigma_{k\in l(i)} \nabla\xi_k \, p_k^{n+1} \tag{10}$$

The solenoidal condition expressed by Equation (8) can be discretized applying the divergence theorem to the control volume Ξ_j surrounding the node j,

$$\int_{\Xi_j} \nabla\cdot u^{n+1} \,\mathrm{d}\Xi = \oint N\cdot u^{n+1} \,\mathrm{d}\Upsilon = 0 \tag{11}$$

where $\mathrm{d}\Upsilon$ is a boundary element of the control volume Ξ_j. The boundary integral can be approximated as follows:

$$\Sigma_{\Xi_j} \nabla\cdot u_{\mathrm{m}}^{n+1} = 0 \tag{12}$$

where $T(j)$ is the set of triangles surrounding node j, and the vectors $N_{m,j}$ (referred to node j and triangle m) are a zero-divergence basis, defined as a discretization of the normal N. If the control volume Ξ_j is created by means of the triangle medians, it is easy to verify that, in each triangle m one has

$$\nabla\xi_j = \frac{N_{m,j}}{2|\Omega_{\mathrm{m}}|} \tag{13}$$

Denoting the elementwise vector quantity

$$S_i^n = u_i^n - \frac{\Delta t}{|\Omega_i|} \int_{\Omega_i} [u\cdot\nabla u]^n \,\mathrm{d}\Omega + \frac{\Delta t}{|\Omega_i|} \int_{\Omega_i} \left[\frac{\nabla\cdot\tau + \rho g}{\rho}\right]^n \mathrm{d}\Omega \tag{14}$$

substitution of Equation (10) into Equation (12) yields the final equation for the nodal pressure values p_j^{n+1}:

$$\Delta t \Sigma_{m\in T(j)} \frac{1}{2\rho_{\mathrm{m}}^{n+1}} \frac{N_{m,j}}{|\Omega_i|} \sum_{k\in l(m)} N_{m,k} \, p_k^{n+1} \Sigma = \Sigma_{m\in T(j)} N_{m,j} S_{\mathrm{m}}^n \tag{15}$$

Note that at this point the advanced density ρ^{n+1} is already known, since the density equations are purely convective. The system of equations for pressure p_j^{n+1} has to be solved at each time step. It is the discrete form of the Poisson elliptic problem for pressure which is encountered in many predictor–corrector methods. The resulting system matrix is symmetric either for the

single- and multicomponent formulation. However, for the single component case the system matrix does not change in time, thanks to the uniformity of the density, whereas for the multicomponent case, the spatial changes in density imply that the system matrix has to be recomputed at each time step. As a consequence, one has the additional computational cost of forming the system, and also of factorizing the preconditioning matrix.

Once the system has been solved, the pressure gradient ∇p^{n+1} is known in each triangular control volume Ω_i, and the velocity field is updated according to the governing equations:

$$u_i^{n+1} = S_i^n + \int_{\Omega_i} \left[\frac{\nabla p}{\rho}\right]^{n+1} \mathrm{d}\Omega \tag{16}$$

The proposed numerical scheme provides first-order accuracy in space and also first-order accuracy in time.

If W_i is assumed constant (equal to $\bar{W}_j$), the resulting numerical scheme will be first-order accurate in space. Choosing a linear dependency in space for W_i nearly second-order accuracy in space can be obtained. To find the associated constant gradient the gradient of W in the control volumes is determined. This can be found using Green's theorem over the control volume boundary, using the (assumed constant) values of W inside each element touching the control volume. The gradient in the element may then be found by the following rule:

$$|w|_e = \begin{cases} k \min_{CV|e} |W|_{CV|e} & \text{if } k \min_{CV|e} |W|_{CV|e} < \min^*_{CV|e} |W|_{CV|e} \\ \min^*_{CV|e} |W|_{CV|e} & \text{if } k \min_{CV|e} |W|_{CV|e} > \min^*_{CV|e} |W|_{CV|e} \end{cases} \tag{17}$$

where $\min_{CV|e}$ is the minimum over the control volumes touching element e, and $\min^*$ is the minimum of all the values but the minimum (i.e. the second-smallest value). If $k = 1$ the minimum of the gradients in the control volumes is always chosen. This corresponds to the *minmod* limiter [33]. In order to preserve the total variation diminishing (TVD) property it must be demanded that $k \leqslant 2$, with $k = 2$ corresponding to the *Superbee* limiter [33]. The minmod limiter is the most diffusive one, but possibly yields the more robust numerical solutions, whereas the superbee limiter yields sharper profiles near discontinuities.

4. COMPUTATIONAL DOMAIN AND SIMULATIONS

The computational domain and boundary conditions used for the simulations with DGA are shown in Figure 3. At the inlet boundary, a uniform velocity profile is imposed, while on the outlet boundary pressure is imposed and velocity is set free. No-slip boundary conditions are applied to the cylinder wall and all velocity components are set to zero. On both horizontal boundaries velocity is imposed; the velocity in the y direction is set to $v = 0$ so that no flow can be drained off or entrained via these boundaries. To ensure no mesh refinement will take place along the horizontal boundaries, the x-component of the velocity is set equal to the velocity imposed at the inlet in order not to maintain any velocity gradient along the horizontal boundaries.

A total of 12 simulations are presented in this work and the simulations are carried out at three different Reynolds numbers, 50, 54 and 75, respectively. For each Reynolds number two simulations with conventional grid are presented using a mesh with 4000 and 10 000 nodes, respectively. For the other simulations, a DGA algorithm is applied using an initial mesh

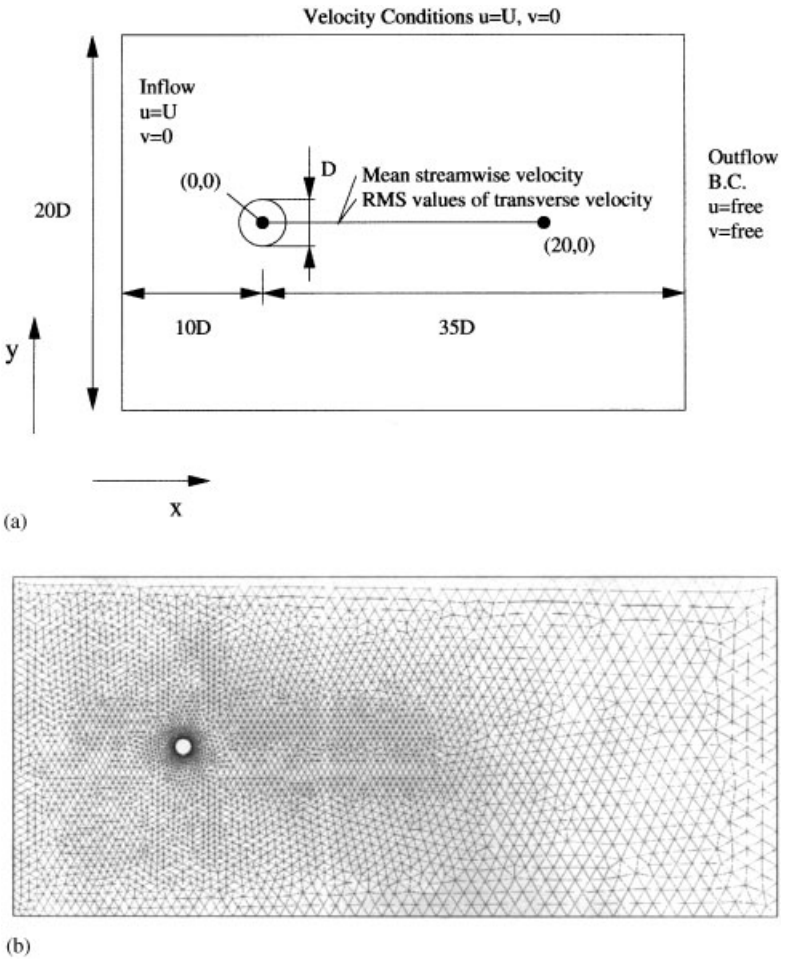

Figure 3. Dimensions of the computational domain: (a) Dimensions in ratio of cylinder diameter. (b) Initial mesh containing 4000 nodes.

containing 4000 nodes. The product of the rate of strain and mesh cell size is used as a grid adaptation variable to enhance the mesh during the flow calculations. The configuration of the DGA algorithm requires specification of the DGA variables C_{m} and C_{str}, threshold values for mesh refinement and coarsening, as well as a maximum mesh size to restrict excessive mesh refinement. Details of the DGA configuration are presented in Table I. The number of nodes towards the end varies each time step, but an approximation is used to indicate the all over mesh size.

The numerical data presented in this work is compared with the work of Paranthoën *et al.* [18]. Four main characteristics of the flow prediction are compared with Paranthoën's work. These are the Strouhal number St[‡] and the recirculation length, the streamwise evolution

[‡]Strouhal number is the frequency of the periodic wave relative to cylinder diameter and inlet velocity $St = fD/u_\infty$.

Table I. Configuration of DGA algorithm for the two-dimensional simulations, including the threshold values for mesh refinement and coarsening as well the minimum mesh resolution in terms of a microscale.

	Refinement e_i	Coarsening e_i	Microscale (m)
$50 \mid C_m = 1 \mid C_{str} = 1$ nodes $= 10.3 \times 10^3$	1.75×10^{-3}	8.75×10^{-4}	7×10^{-3}
$50 \mid C_m = 1 \mid C_{str} = 1.5$ nodes $= 7.2 \times 10^3$	1.50×10^{-3}	7.50×10^{-4}	7×10^{-3}
$54 \mid C_m = 1 \mid C_{str} = 1$ nodes $= 8.4 \times 10^3$	2.50×10^{-3}	1.25×10^{-4}	7×10^{-3}
$54 \mid C_m = 1 \mid C_{str} = 1.5$ nodes $= 7.5 \times 10^3$	1.50×10^{-3}	7.50×10^{-4}	7×10^{-3}
$75 \mid C_m = 1 \mid C_{str} = 1$ nodes $= 9.5 \times 10^3$	2.50×10^{-3}	1.25×10^{-4}	7×10^{-3}
$75 \mid C_m = 1 \mid C_{str} = 1.5$ nodes $= 8.8 \times 10^3$	1.50×10^{-3}	7.50×10^{-4}	7×10^{-3}

of the mean centreline velocity in the cylinder wake and the streamwise evolution of the centreline root mean square (RMS) values of the transverse velocity in the cylinder wake. The streamwise profiles are plotted over a distance of 20 cylinder diameters as shown in Figure 3.

5. BASIC FLOW FEATURES

In this work, a variety of simulations is presented and compared with the experimental data of Paranthoën *et al.* [18]. In Figure 4 a vector plot is given which shows the general features of the flow. The simulations with conventional mesh and DGA show good agreement with experimental data at each Reynolds number. The Strouhal number St at a Reynolds number of 50 is predicted to be 0.125 and is in full agreement with the experimental data. The simulations have shown good agreement with both conventional mesh and DGA as shown in Table II. The Strouhal number St predicted at higher Reynolds number (75) shows a small deviation when using a coarser mesh. The reduction in Strouhal number St indicates that numerical diffusion is reducing the effective Reynolds number.

The streamwise evolution of the mean centreline velocity is predicted well despite minor deviation, as shown in Figure 6. In contrast to the Strouhal number St, the streamwise velocities and the RMS values of the transverse velocity at a Reynolds number of 50 are dependent on the utilized mesh. The velocity and RMS profiles show a more diffused flow field in case the mesh is insufficiently fine. A similar trend is seen at higher Reynolds numbers also and to obtain good agreement higher mesh resolution is required as shown in the simulations with conventional mesh (10 000 nodes) and DGA (Figures 6 and 7).

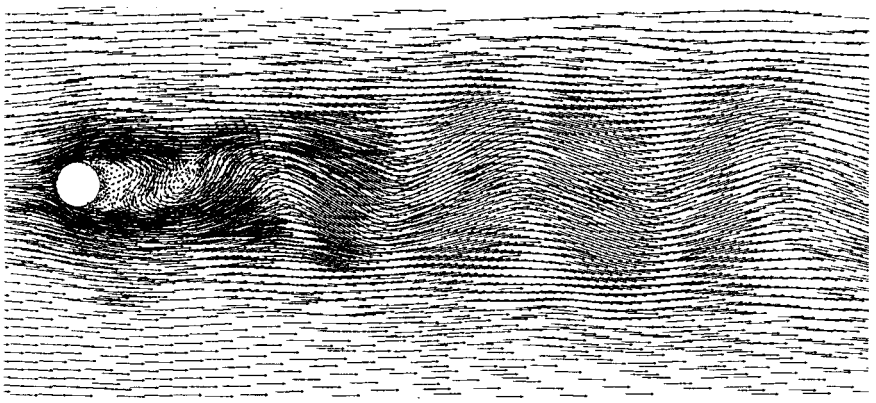

Figure 4. Vectorplot of a fully developed vortex shedding, $75 \mid C_{\mathrm{m}} = 1 \mid C_{\mathrm{str}} = 1 \mid \mathrm{nodes} = 9.5 \times 10^3$.

Table II. Strouhal number St and mean length of recirculation L_r for Reynolds numbers of 50, 54 and 75, respectively.

	St	L_{r}
50 | Paranthoën *et al.*	0.125	2.45
50 | NO-DGA | nodes $= 4 \times 10^3$	0.125	2.45
50 | NO-DGA | nodes $= 10 \times 10^3$	0.125	2.45
50 | $C_{\mathrm{m}} = 1 \mid C_{\mathrm{str}} = 1 \mid$ nodes $= 10.3 \times 10^3$	0.127	2.25
50 | $C_{\mathrm{m}} = 1 \mid C_{\mathrm{str}} = 1.5 \mid$ nodes $= 7.2 \times 10^3$	0.129	2.25
54 | Paranthoën *et al.*	0.130	2.20
54 | NO-DGA | nodes $= 4 \times 10^3$	0.131	2.25
54 | NO-DGA | nodes $= 10 \times 10^3$	0.135	2.25
54 | $C_{\mathrm{m}} = 1 \mid C_{\mathrm{str}} = 1 \mid$ nodes $= 8.4 \times 10^3$	0.135	2.25
54 | $C_{\mathrm{m}} = 1 \mid C_{\mathrm{str}} = 1.5 \mid$ nodes $= 7.5 \times 10^3$	0.134	2.25
75 | Paranthoën *et al.*	0.150	1.50
75 | NO-DGA | nodes $= 4 \times 10^3$	0.145	1.85
75 | NO-DGA | nodes $= 10 \times 10^3$	0.153	1.70
75 | $C_{\mathrm{m}} = 1 \mid C_{\mathrm{str}} = 1 \mid$ nodes $= 9.5 \times 10^3$	0.151	1.75
75 | $C_{\mathrm{m}} = 1 \mid C_{\mathrm{str}} = 1.5 \mid$ nodes $= 8.8 \times 10^3$	0.153	1.70

To obtain good agreement with experimental data a mesh containing 4000 nodes is insufficient. Despite the limitations of a coarser mesh, a semi-steady vortex street is accomplished substantially faster when using a coarse mesh. This phenomenon can be seen in Figure 8 which shows a time plot of the transverse velocity in the wake behind the cylinder. The figure shows the onset of a vortex motion is substantially delayed when using a mesh containing 10 000 nodes.

6. DISCUSSION

6.1. The use of a DGA algorithm

The numerical predictions obtained with a mesh containing 4000 nodes have shown all the major characteristics of the flow field. The vortex shedding and the shedding frequency are predicted well for the simulations with coarse mesh. Nevertheless, a comparison of the streamwise velocity as well the RMS values of the transverse velocity has shown some deviation. This deviation was minimized in the simulations with further mesh refinement both conventional and with DGA. The deviation with experimental data for simulations using a coarse mesh (4000 nodes) indicates that the mesh is insufficient to capture all unsteady flow phenomena. The results imply that even for a low Reynolds number problem as investigated in this work, accurate modelling still requires sufficient mesh resolution.

In the simulations with a conventional mesh containing 10 000 nodes a very close agreement has been accomplished. Not only the general flow characteristics are predicted well, but also the prediction of mean streamwise velocity and RMS values of the transverse velocity are in close agreement. However, to obtain this level of agreement a fine mesh has to be applied *a priori*. To apply a fine mesh in advance is a severe limitation in many numerical investigations. Good agreement can be obtained by applying a fine mesh, however it comes with computational costs.

The numerical data obtained from the simulations with DGA have shown a good level of agreement without *a priori* applying a fine mesh. The key to a successful mesh refinement is the refinement variable. In this work a refinement variable is used which is a product of the element size and rate of strain, as shown in Equation (3). When using values of 1 for both C_m and C_{str} the variable is proportional to the mesh-Reynolds number. By varying the values for C_m and C_{str}, the dimensions for the refinement variable consequently change leading to another refinement behaviour, as shown in Figure 5.

With increasing values for C_{str} emphasis is given to the rate of strain leading to a narrow mesh refinement in the vicinity of the wall. In contrast, a decrease in values for C_{str} leads to a smooth mesh refinement in a wider area. The numerical results indicate that the mesh refinement is most effective when choosing C_m and C_{str} to be 1 and 1.5, respectively. Under these conditions, a solid agreement with experimental can be obtained with roughly 8000 nodes. Although good agreement has been obtained with the other simulations using DGA, the required number of grid points to accomplish the same level of agreement was higher. In comparison with simulations whereby C_{str} is set 1, the use of a higher value for C_{str} leads to an increase in mesh resolution in the vicinity of the wall where high velocity gradients are apparent. To ensure an acceptable mesh resolution, the minimum element size was bounded. Without this limitation the mesh would become unnecessarily fine, leading to a reduction in time increment and consequently an increase in numerical computations. The work of De With *et al.* [30] has shown that for a vortex shedding in the sub-critical flow regime at Reynolds numbers of 1.4×10^5 suitable values for C_m and C_{str} are 1 and 1.5, respectively. However, for flows at higher Reynolds numbers, it was unnecessary to limit the mesh resolution. Instead, the modelling of the smaller turbulent scales in the vicinity of the wall where of crucial importance to the flow field. The differences in the refinement variable configuration reflect the nature of the problem. In the sub-critical flow regime, shear layer instabilities are leading

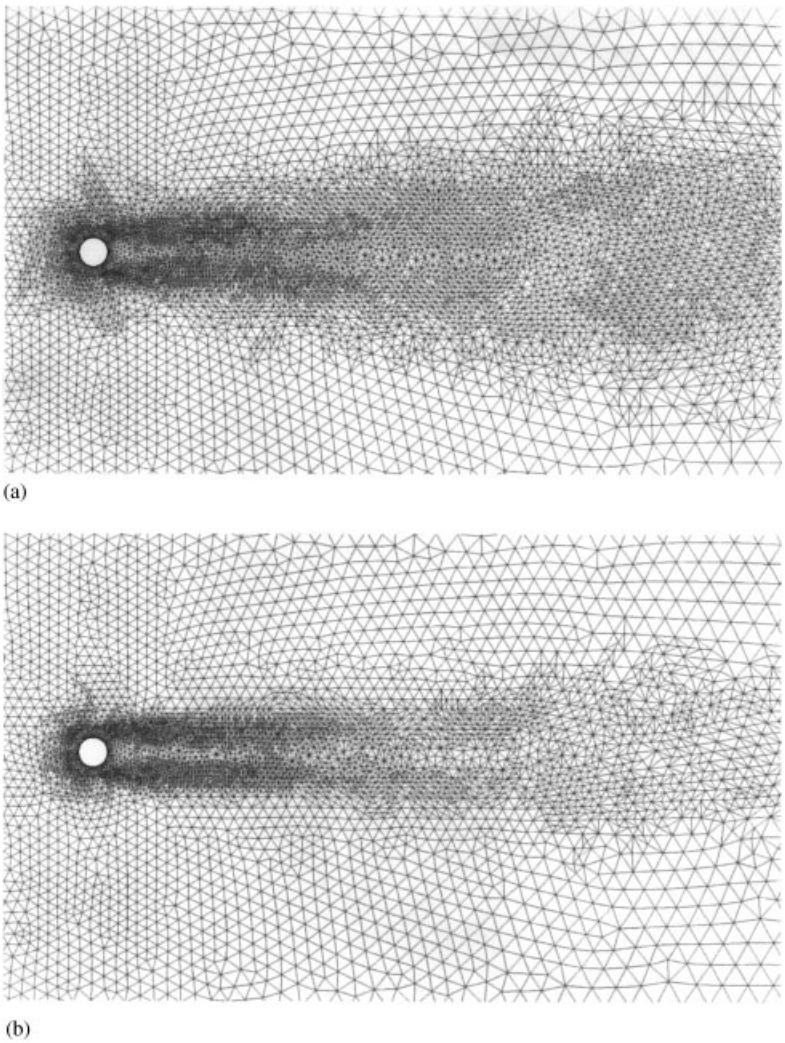

Figure 5. Plot of the numerical domain towards the end of the simulation: (a) $50 \mid C_m = 1 \mid C_{str} = 1 \mid 10\,300$ nodes. (b) $50 \mid C_m = 1 \mid C_{str} = 1.5 \mid 7200$ nodes.

to fine scale structures. The interaction between the fine scales is leading to a transition from laminar to turbulent flow in the thin shear layers just after the point of separation. In contrast, the flow investigated in the present work is dominated by primary Karman vortices. The vortex shedding originates from the counterrotating vortex pairs. The counterrotating vortices cover a wider sparse area in comparison with the fine-scale structures at high Reynolds numbers. Consequently, a smoother refinement behaviour is required with values for C_m and C_{str} in the order of 1. The results have shown that a refinement variable as proposed in this work can work well for laminar flow problems.

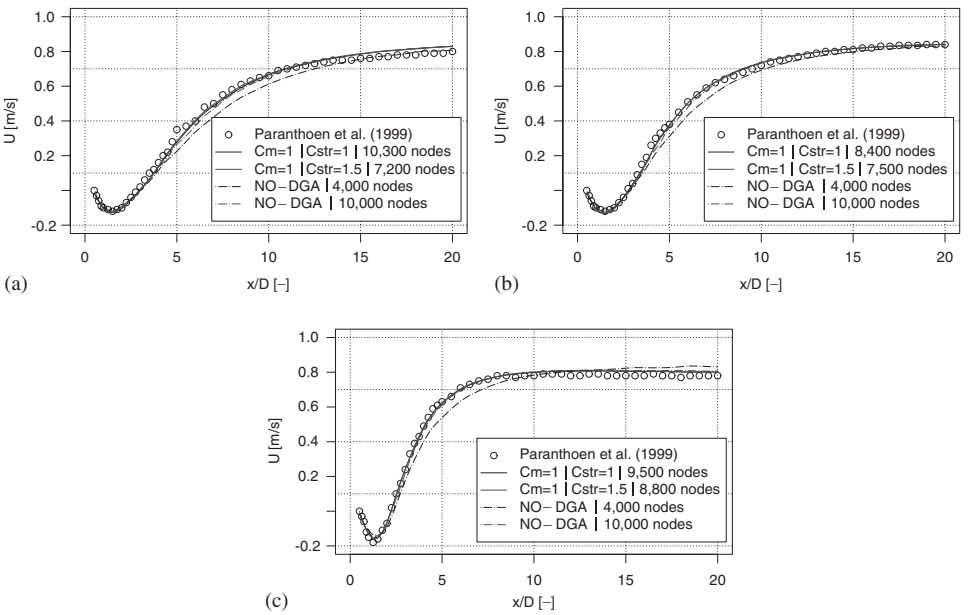

Figure 6. Streamwise evolution of the mean centreline velocity in the cylinder wake, Reynolds number 50, 54 and 75: (a) Reynolds number 50; (b) Reynolds number 54; (c) Reynolds number 75.

The use of a grid adaptation algorithm has led to an increase in grid points as the simulation progressed, the initial mesh used at the start of the simulation was containing 4000 nodes, and towards the end of the simulation the number of grid points had roughly doubled. Nevertheless, the grid points that have been added to the mesh during the simulation were required to obtain good agreement. In contrast, the mesh refinement in a conventional mesh depends on the users judgement and his knowledge about the flow under investigation.

An indication of the CPU time requirements on a Intel Pentium II 450 MHz, for simulations with both conventional mesh and grid adaptation is shown in Table III. The CPU requirements indicate that the use of a DGA algorithm enabled a reduction in simulation time of roughly 50% for a simulation of comparable size. The reduction in simulation time is partly accomplished by a significant mesh reduction during the initial development of the flow. A second reason for the reduction in CPU requirements is due to the early onset of vortex shedding which enables a reduction of the physical simulation time. The substantial difference in CPU requirements between the simulation with C_{m} and C_{str} set to 1 and the simulation with C_{m} and C_{str} set to 1 and 1.5, respectively, is due to the nature of the refinement variable. When selecting an increased value for C_{str} the refinement becomes more focussed, which automatically enforces the time increment to decrease.

6.2. *Onset of vortex shedding*

The vortex shedding at Reynolds numbers in the laminar flow regime originates from an instability in the counterrotating vortices. For flows below the critical Reynolds number of 45, it is expected that any instability or perturbation is of a convective nature [34]. Under

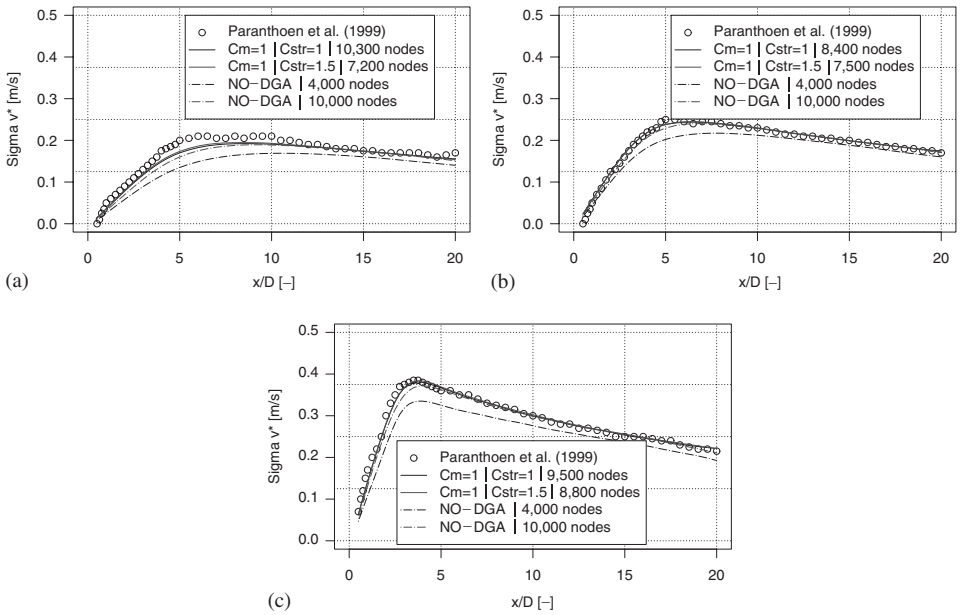

Figure 7. Streamwise evolution of the centreline RMS values of the transverse velocity in the cylinder wake, Reynolds number 50, 54 and 75: (a) Reynolds number 50; (b) Reynolds number 54; (c) Reynolds number 75.

such conditions, a disturbance grows with time, but is concurrently convected away leaving the wake undisturbed. For higher Reynolds numbers above 45, instabilities are of an absolute nature and initial disturbances grow at any fixed location and, after non-linearities have limited the growth of the disturbance, a self-sustained oscillation of the wake is established. The simulations have shown that with increasing Reynolds number the flow becomes more sensitive to any kind of perturbation and as a consequence, vortex shedding starts to develop already in an early stage of the simulation.

Explicit perturbations to trigger an absolute instability were not applied, instead asymmetric mesh, rounding errors and numerical dispersion were sufficient to perturbate the flow field. The significance of the mesh onto the onset of vortex shedding is shown in Figure 8, in here the simulation with coarse mesh (4000 nodes) shows stronger fluctuation in the transverse velocity. When using a finer mesh the amplification of initial velocity fluctuation is weaker and the development of a steady periodic vortex shedding is reached later. For simulations using a DGA algorithm, the time to reach a semi-steady-state situation is of the same order as the simulations with coarse mesh (4000 nodes). In contrast, a simulation with a conventional fine mesh (10 000 nodes), requires more time to develop a semi-steady vortex shedding.

Holdø and Weibust [35] concluded that for laminar flow around a bluff body an initial disturbance or perturbation is required to initiate vortex shedding when simulating the flow field with a high-density mesh. In accordance with their work, a higher mesh resolution delayed the onset of vortex shedding. It is suggested that this phenomenon occurs due to

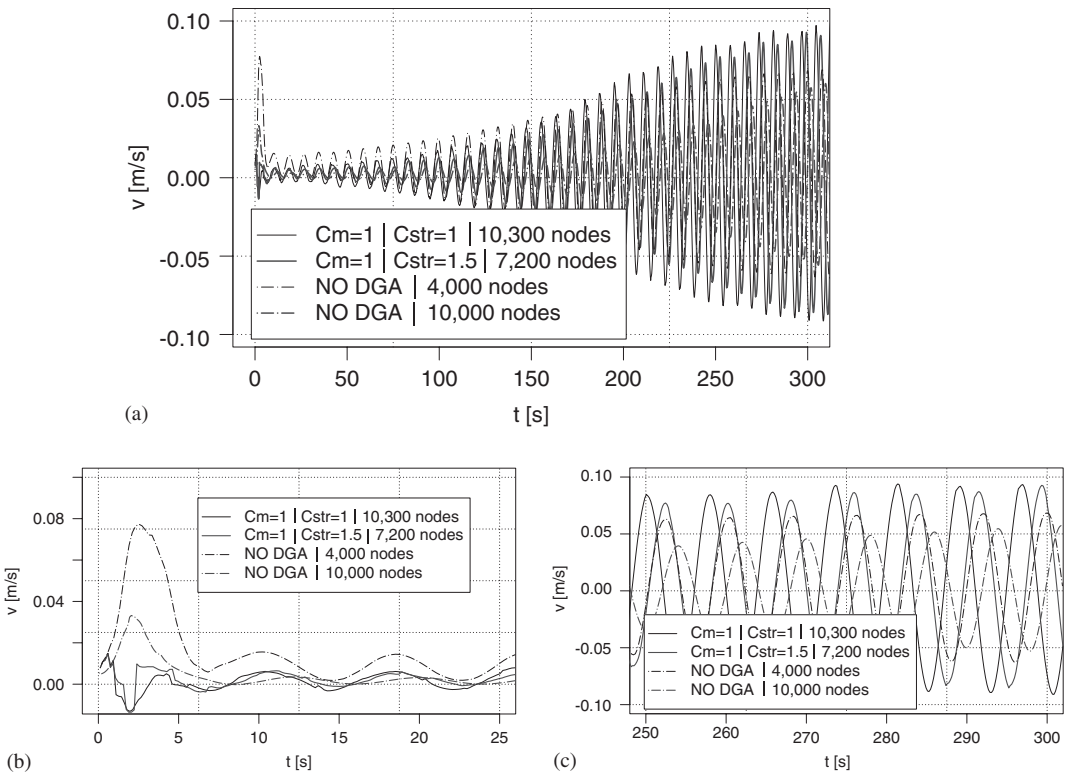

Figure 8. Time history of the transverse velocity at $x = 1.75D$ and $y = 0$ for Reynolds number 50: (a) Time is 0–300 (s); (b) Time is 0–25 (s); (c) Time is 250–300 (s).

Table III. Required CPU time on an Intel Pentium II 450 MHz for the calculations at a Reynolds number of 54.

Simulation name	CPU time (h)
54 | NO-DGA | nodes $= 4 \times 10^3$	3
54 | NO-DGA |nodes $= 10 \times 10^3$	12
54 $\mid C_{\mathrm{m}} = 1 \mid C_{\mathrm{str}} = 1 \mid$ nodes $= 8.4 \times 10^3$	5
54 $\mid C_{\mathrm{m}} = 1 \mid C_{\mathrm{str}} = 1.5 \mid$ nodes $= 7.5 \times 10^3$	10

dispersion and asymmetry in the coarser mesh. In most CFD algorithms, numerical viscosity is the dominant error. Errors of this type lead to a more diffused solution. Nevertheless, in a flow field which requires a perturbation or instability, numerical dispersion and asymmetry in the mesh are of crucial importance, even when they come together with numerical diffusion. Numerical diffusion and dispersion are generally considered to be weaknesses in the numerical algorithm. However, they are crucial for the initiation of vortex shedding at low Reynolds

numbers and they mimic the existence of small flow motions that are apparent in windtunnel flow.

6.3. *Conflicting mesh requirements for low Reynolds numbers*

The numerical results have shown a drawback of the use of conventional mesh for laminar flow. To obtain good agreement with experimental data the mesh must be sufficiently fine. However, to initiate vortex shedding a coarse mesh is more suitable. As shown in the work of Holdø and Weibust [35], these requirements can lead to a need for explicit perturbations. In an ideal flow set-up whereby the flow inside the numerical domain only covers one bluff body, special boundary conditions can be applied to encourage vortex shedding. However, in case the laminar vortex shedding is part of a larger system then explicit perturbation becomes very arduous, if not all together impossible.

The present work has shown that a DGA algorithm is not only suitable to reduce the computational needs, but helps to trigger vortex shedding early in the simulation process. Consequently, simulation times can be significantly reduced.

7. CONCLUSIONS

A conventional mesh used in a numerical CFD investigation has significant drawbacks. Either the mesh is too coarse and there is an unacceptable increase in numerical viscosity, or an *a priori* fine mesh is used with a high number of redundant cells. In the former, the simulation is masked by numerical viscosity leading to a diffused flow prediction, while in the latter it is most likely that a significant percentage of the mesh will have only a marginal influence on the flow prediction.

There are several advantages in using the DGA algorithm in low Reynolds number flows:

- In a low Reynolds number flow, the mesh requirements are substantially less in comparison with a high Reynolds number turbulent flow. Nevertheless, mesh requirements are apparent and, in an unsteady flow, the regions which require a higher mesh resolution vary with time. In such flow the use of a DGA algorithm can reduce the total mesh size substantially.
- The refinement variable presented in this work is a product of rate of strain and element size.
- An effective refinement variable is accomplished with values of 1 and 1.5 respectively for C_{m} and C_{str}. However, it is necessary to limit the mesh refinement in the vicinity of the wall to bound excessive mesh refinement.
- No explicit perturbation is used to initiate vortex shedding. Instead the vortex shedding was initiated by rounding errors, asymmetric mesh and numerical dispersion. The use of a DGA algorithm enabled a coarse mesh at the start of the simulation, which led to a significant increase in the development of a vortex street. As the simulation is progressing, the mesh is refined and good agreement with experimental data is obtained.

APPENDIX A: NOMENCLATURE

A_{tr}	element surface
C_{m}	adaptation variable for mesh
C_{str}	adaptation constant for strain
D	cylinder diameter
e	estimated error
k	flux limiter variable
n	instantaneous number of time step
N	number of neighbour elements
p	pressure
q	refinement variable
$\hat{q}$	interpolated refinement variable
Re	Reynolds number
Re_{Δ}	mesh-Reynolds number
S	elementwise vector
$\lvert\bar{S}\rvert$	magnitude of strain
St	Strouhal number
T_j	triangles surrounding node j
t	time
u	streamwise velocity component
u^*	streamwise intermediate velocity component
v	lateral velocity component
W	primitive variable
x	space in streamwise direction
y	space in lateral direction
Δ	subgrid length scale
Ξ	control volume
Σ	elemental arc
Υ	boundary element
Ω	discretization element
η	normal unit vector
μ_{t}	turbulent viscosity
μ	kinematic viscosity
ν	element periphery
ξ	classical shape function
ρ	density

REFERENCES

1. Bénard H. Formation de centres de giration à l'arrière d'un obstacle en mouvement. *C. R. Academy of Science, Paris* 1908; **147**:839–842.
2. Von Karman T. über den mechanismus des widerstandes den ein bewegter körper in einer flüssigkeit erfährt. *Gott. Nachr.* 1911; 509–517.
3. Kovasznay LSG. Hot wire investigation of the wake behind a cylinder at low Reynolds numbers. *Proceedings of the Royal Society* 1949; **198**:174–190.
4. Roshko A. On the development of turbulent wakes from vortex streets. *NACA report* (1191), 1953.
5. Abernathy FH, Kronauer RE. The formation of vortex street. *Journal of Fluid Mechanics* 1962; **13**:1–20.

6. Bearman PW. On vortex street wakes. *Journal of Fluid Mechanics* 1967; **28**:625–641.
7. Son JS, Hanratty TJ. Numerical solution for the flow around a circular cylinder at Reynolds numbers of 40, 200 and 500. *Journal of Fluid Mechanics* 1969; **35**:369–386.
8. Posdziech O, Grundmann R. A systematic approach to the numerical calculation of fundamental quantities of the two-dimensional flow over a circular cylinder. *Journal of Fluids and Structures* 2000, submitted.
9. Thompson M, Hourigan K, Sheridan J. Three-dimensional instabilities in the wake of a circular cylinder. *Experiments on Thermal Fluid Science* 1996; **12**(2):190–196.
10. Zhang HQ, Fey U, Noack BR, König M, Eckelmann H. On the transition of the cylinder wake. *Physics of Fluids* 1995; **7**(4):779–794.
11. Henderson RD. Nonlinear dynamics and pattern formation in turbulent wake transition. *Journal of Fluid Mechanics* 1997; **352**:65–112.
12. Persillon H, Braza M. Physical analysis of the transition to turbulence in the wake of a circular cylinder by three-dimensional Navier–Stokes simulation. *Journal of Fluid Mechanics* 1998; **365**:23–88.
13. Kravchenko AG, Moin P, Shariff K. B-spline method and zonal grids for simulations of complex turbulent flows. *Journal of Computational Physics* 1999; **151**:757–789.
14. Eiseman PR. Adaptive grid adaptation. *Computer Methods in Applied Mechanics and Engineering* 1987; **64**:321–376.
15. Habashi WG, Dompierre J, Bourgault Y, Ait-Ali-Yahia D, Fortin M, Vallet MG. Anisotropic mesh adaptation: towards user-independent, mesh-independent and solver-independent cfd. part 1: general principles. *International Journal for Numerical Methods in Fluids* 2000; **32**:725–744.
16. Lohner R. Mesh adaptation in fluid mechanics. *Engineering Fracture Mechanics* 1995; **50**(5/6):819–847.
17. Muzaferija S, Gosman D. Finite-volume cfd procedure and adaptive error control strategy for grids of arbitrary topology. *Journal of Computational Physics* 1997; **138**:766–787.
18. Paranthoën P, Browne LWB, Le Masson S, Dumouchel F, Lecordier JC. Characteristics of the near wake of a cylinder at low Reynolds numbers. *European Journal of Mechanics B/Fluids* 1999; **18**:659–674.
19. Diaz AR, Kikuchi N, Taylor JE. A method for grid optimization for the finite-element method. *Computer Methods in Applied Mechanics and Engineering* 1983; **41**:29–45.
20. Hetu JF, Pelletier DH. Adaptive remeshing for viscous incompressible flows. *AIAA* 1992; **30**:1986–1992.
21. Lohner R. An adaptive finite element solver for transient problems with moving bodies. *Computers and Structures* 1988; **30**:303–317.
22. Lohner R. Adaptive remeshing for transient problems. *Computer Methods in Applied Mechanics and Engineering* 1989; **75**:195–214.
23. Babuska I, Zienkiewicz OC, Gago J, de A Oliverira ER. *Accuracy Estimates and Adaptive Refinements in Finite Element Computations*. Wiley: New York, 1986.
24. Mavriplis C. A posteriori error estimators for adaptive spectral element techniques. *Proceedings of the 8th GAMM Conference on Numerical Methods and Fluid Mechanics* 1990; **29**:333–342.
25. Zienkiewicz OC, Zhu JZ. A simple error estimator and adaptive procedure for practical engineering analysis. *International Journal for Numerical Methods of Engineering* 1987; **24**:337–357.
26. Huld T, Wilkening H. *Adaptive Grid Methods for Unsteady CFD Problems*. Joint Research Centre of the European Commission, T.P. 723, I-21023 Ispra, 1998.
27. Nambiar RV, Valera RS, Lawrence LL, Morgan RB, Amil D. An algorithm for adaptive refinement of triangular element meshes. *International Journal for Numerical Methods in Engineering* 1993; **36**:499–509.
28. Rivara MC. Algorithms for refining triangular grids suitable for adaptive and multigrid techniques. *International Journal for Numerical Methods in Engineering* 1984; **20**:745–756.
29. Zhang H, Trepanier YJ, Reggio M, Camarero R. A Navier–Stokes solver for stretched triangular grids. In *AIAA 30th Aerospace Science Meeting*, pages Paper 92–0183, 1992.
30. De With G, Holdø AE, Huld TA. The use of dynamic grid adaptation algorithms for the modelling of flow around a circular cylinder in sub-critical flow regime. *International Journal for Numerical Methods in Fluids* 2003; **41**:789–808.
31. Soria A, Ruel F. *Finite Volume Analysis of Multicomponent Incompressible Viscous Flow*. Joint Research Centre of the European Commission, special publication no. i.94.41 edition, 1994.
32. Chorin AJ. Numerical solution of the Navier–Stokes equations. *Mathematics of Computations* 1968; **23**: 341–353.
33. Hirsch C. *Numerical Computation of Internal and External Flows*, vol. 1. Wiley, New York, 1988.
34. Triantafyllou GS, Triantafyllou MS, Chryssostomidis C. On the formation of vortex streets behind stationary cylinders. *Journal of Fluid Mechanics* 1986; **170**:461–477.
35. Holdø AE, Weibust E. Effects due to initial perturbations on the modelling of vortex shedding from a bluff body in a laminar flow. *Computers and Fluid Dynamics* 1996 **6**:337–344.

