INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2008; **57**:1023–1045

Sensitivity analysis of low Reynolds number channel flow using the finite volume method

R. D. Kirkman and M. Metzger*,†

Department of Mechanical Engineering, University of Utah, Salt Lake City, UT 84112, U.S.A.

SUMMARY

An unsteady finite volume-based fractional step algorithm solved on a staggered grid has been developed for computing design sensitivity parameters in two-dimensional flows. Verification of the numerical code is performed for the case of low Reynolds number, pressure-driven flow through a straight channel, which has an exact steady-state solution to the Navier–Stokes equations. Sensitivity of the flow to the channel height, fluid viscosity, and imposed pressure gradient is considered. Three different numerical techniques for computing the design sensitivity parameters: finite difference, complex-step differentiation, and sensitivity equation method (SEM), are compared in terms of numerical error (relative to the exact solution), computational expense, and ease of implementation. Results indicate that, of all the three methods, complex step is the most accurate and requires the least computational time. In addition, treatment of the boundary conditions in SEM is addressed, within the framework of the present finite volume approach, with special attention given to parameter dependence in the boundary conditions. Error estimation based on the Grid Convergence Index provides a good indication of the exact error in the SEM solutions. An example of application of the use of sensitivity parameters to estimate the propagation of input uncertainty through the numerical simulation is also provided. Copyright © 2007 John Wiley & Sons, Ltd.

Received 28 February 2007; Revised 5 October 2007; Accepted 8 October 2007

KEY WORDS: sensitivity analysis; laminar flow; computational fluid dynamics

1. INTRODUCTION

With the proliferation of computational fluid dynamics (CFD), techniques and applications that augment CFD as a design and analysis tool have become increasingly important. Sensitivity analysis, when coupled with CFD, is one such example that provides insight on how design

---

*Correspondence to: M. Metzger, Department of Mechanical Engineering, University of Utah, Salt Lake City, UT 84112, U.S.A.

†E-mail: m.metzger@utah.edu

Contract/grant sponsor: National Science Foundation; contract/grant number: IIS-0428856

parameters influence a given flow field. While several studies have discussed sensitivity analysis techniques in the context of *finite element* methods [1–10], the main contribution of the present study is the development of a *finite volume*-based fractional step algorithm with staggered grid for use in computing sensitivity parameters. The finite volume method is particularly popular among engineers. In fact, two of the most heavily used commercial CFD packages (FLUENT and STAR-CD) are based on this method. In the finite volume method, the solution domain is divided into a discrete number of contiguous control volumes, over which the integral form of the conservation equations is applied. By construction, the method is conservative and can accommodate any type of grid, but favors a staggered grid for simplicity of the evaluation of the fluxes at control volume faces. Using the present finite volume CFD code as a computational engine, several different methods for calculating sensitivity parameters are compared including finite difference, complex-step differentiation, and the continuous sensitivity equation method (SEM). The CFD and numerical sensitivity algorithms are verified for the case of low Reynolds number, pressure-driven flow through a straight channel. This test case has an exact steady-state solution to the Navier–Stokes equations and, thus, permits analytic expressions to be obtained for the sensitivity parameters. Channel flow, which may be classified as a canonical wall-bounded flow, was chosen for verification in the present study because of the inherent parameter dependence on the velocity field and boundary conditions, as well as its relevance to longer-term goals of the authors as described further in Section 4.

Over the last several years, computational sensitivity analysis has been integrated into numerical schemes for parameter estimation [4], optimization [3, 5, 7], and uncertainty estimation [8, 11–13]. Numerical sensitivity analysis has potential to make significant contributions in these areas, especially in terms of the latter. Note, the terminology ‘error’ and ‘uncertainty’ are often used interchangeably, with ‘uncertainty’ generally implying the quantification of ‘error’. Some attempts have been made recently to establish explicit definitions of these terms, as well as guidelines for estimating and reporting uncertainty in numerical simulations [14–20]. For example, AIAA [17] defines uncertainty as ‘a potential deficiency in any phase or activity of the modeling process that is due to the lack of knowledge’, and error as ‘a recognizable deficiency in any phase or activity of modeling and simulation that is not due to lack of knowledge’. Nevertheless, error classification and uncertainty estimation are pre-eminent requirements in all simulation sciences, in order to establish confidence in the numerical technique.

The validation process, involving the direct comparison of numerical results with experimental data, yields a useful measure of confidence in the simulation. Non-negligible discrepancies between simulation and experiment are typically traced back to a combination of model uncertainty, input uncertainty, and numerical uncertainty [18]. The process of modeling a fluid flow may introduce errors in the CFD results, due to, for example, oversimplification of the continuum model [21] and/or turbulence model [22]. The present study only considers low Reynolds number flow so that model uncertainty is negligible. Input uncertainty refers to errors in the experimental measurements used for validation of the numerical simulation (e.g. the magnitude of the physical properties of the fluid and dimensions of the flow domain or objects embedded in the flow). Sensitivity analysis, in particular, provides a useful context for quantifying the link between input uncertainty and the subsequent propagation of this uncertainty into the simulation results. Along these lines, Putko *et al.* [13] recently developed a method to treat input parameter uncertainty and the propagation of that uncertainty through gradient-based design optimization schemes using first- and second-order sensitivity derivatives. Numerical uncertainty in CFD comes from computer round-off error, introduced by the representation of numbers using floating point precision, and truncation error, due to the discretization of time and physical space. Truncation error stems from truncating the Taylor series representation of derivatives, as well as the iterative nature of the solution process. The numerical uncertainty resulting from the spatial and temporal convergences of the discretization scheme is generally quantified using error estimators [15, 20, 23].

The outline of this paper is as follows. First, a brief overview of numerical sensitivity methods is provided. The finite volume CFD method and its extension to SEM are then discussed. The exact solutions for the velocity field and sensitivity parameters for the channel flow test case are presented. Sensitivity of the velocity to the channel height, imposed pressure gradient, and fluid viscosity is considered. Verification results for the field variables and sensitivities are shown, including a comparison of the performance of the three different numerical sensitivity methods. Finally, estimates of the numerical and input uncertainty are quantified.

2. SENSITIVITY ANALYSIS METHODS

Sensitivity analysis involves calculating the partial derivatives of the field variables with respect to the design parameters. Design parameters may be classified as physical parameters (such as the viscosity and density of the fluid) or geometric parameters (such as the height of the flow domain or the shape of an imbedded object in the flow). The sensitivity parameter associated with a field variable u and design parameter α is defined as

$$S_{u,\alpha}=\frac{\partial u}{\partial \alpha} \tag{1}$$

Note, the sensitivity parameter is often expressed as S_u without explicit reference to a particular parameter. In this study, P will imply a vector of important design parameters, $P=(P_1,\ldots,P_k)$. The sensitivity parameters associated with P provide a quantitative measure of the change in magnitude of the field variables at every point in the flow, due to changes in the value of P about a given baseline condition. As such, sensitivity analysis can be used to generate the gradients necessary for optimization algorithms [1–3]. In addition, local gradient information from the sensitivity analysis can be used to approximate flow fields ‘near’ the given baseline case [24, 25]. Sensitivity analysis may be performed using several different numerical techniques, including finite difference, complex-step differentiation, software differentiation, and continuous sensitivity equation method.

2.1. Finite difference method

Perhaps the most commonly used method for sensitivity analysis (and simplest to implement) is finite difference. The finite difference method uses a finite difference approximation to the sensitivity derivative. To first-order accuracy, the sensitivity parameter is approximated as

$$S_u=\frac{u(P+\Delta P)-u(P)}{\Delta P}+O(\Delta P) \tag{2}$$

In order to calculate the sensitivity parameter with finite difference, at least two different numerical simulations are required, one using the value P (referred to as the baseline case) and another using the value $P+\Delta P$ (referred to as the perturbation). Therefore, the finite difference method can be computationally expensive, especially for problems with a large number of design parameters.

DOI: 10.1002/fld

In addition, the truncation error depends on the step size ΔP. If ΔP is too large, the sensitivity derivative may not give a good approximation to the local gradient. On the other hand, if ΔP is too small, sensitivity derivatives may be of the order of the fluctuations in the u field and machine round-off errors become significant. Currently, no methodology exists for selecting the appropriate size for ΔP, other than a parameter study. This poses a problem because S_u can be quite sensitive to the value used for ΔP, as shown later in Section 5. Higher-order finite difference approximations certainly reduce the truncation error in S_u; however, at the cost of increasing the total number of numerical simulations that must be performed. For example, given k parameters of interest, a first-order accurate finite difference method for S_u would require $k+1$ CFD solves (one for the baseline and one for each parameter); whereas a second-order accurate method would require $2k$ CFD solves.

2.2. Complex-step method

The complex-step method is similar to the finite difference method; however, the approximation to the sensitivity derivative with respect to a complex parameter is $P=f(\overline{P}+\mathrm{i}\Delta P)$, where $\overline{P}$ denotes the nominal parameter value associated with a baseline case. Performing a Taylor series expansion of the real-valued variable u about the complex parameter P gives

$$u(\overline{P}+\mathrm{i}\Delta P)=u(\overline{P})+\left.\frac{\partial u}{\partial P}\right|_{\overline{P}}\mathrm{i}\Delta P-\frac{1}{2!}\left.\frac{\partial^2 u}{\partial P^2}\right|_{\overline{P}}\Delta P^2+O(\Delta P^3) \tag{3}$$

The sensitivity parameter follows directly from (3) as

$$S_u=\frac{\mathrm{Im}[u(\overline{P}+\mathrm{i}\Delta P)]}{\Delta P}+O(\Delta P^2) \tag{4}$$

The complex-step method is less sensitive to the value of ΔP because it does not involve a difference operation in the discrete representation of the derivative, only a function evaluation using complex arithmetic. Martins *et al.* [26] recommended using a step size ΔP on the order 10^{-8}, which reduced the normalized error in the sensitivity estimate to the same accuracy as the function evaluation. Although the complex-step method eliminates the parameter step-size ambiguity of the finite difference method, the cost trade-off increased the complication of the numeric code, which must accommodate complex variables and complex arithmetic.

2.3. Software differentiation

Software differentiation uses a pre-processor that receives an existing numeric source code and differentiates it line by line in order to generate a new source code that solves for the sensitivity parameters. As such, software differentiation uses a discretize-then-differentiate approach to differentiating the governing equations for sensitivity analysis. Software differentiation pre-processors have been developed for C [27], Fortran [28, 29], and MATLAB [30–32], among others. Software differentiation is not considered further in the present study.

2.4. Sensitivity equation method

The SEM requires that the original governing equation(s) of a given system, as well as the boundary and initial conditions, be differentiated with respect to the design parameters prior to discretization

DOI: 10.1002/fld

[2, 6, 33]. The new set of equations for the sensitivity parameters may be solved by implementing the same numeric algorithms and techniques as used with the original governing equations. At each time step in the numeric algorithm, the field variables are computed followed by the sensitivity parameters. Section 3 documents the application of SEM to the finite-volume-based numerical solution of the incompressible, two-dimensional Navier–Stokes equations.

2.5. *Normalized scaled sensitivity parameters*

Because raw (or unscaled) sensitivity parameters are dimensional quantities, with units dependent both on the field variable and physical/geometric parameter, direct comparison between the magnitude of one sensitivity parameter with another is not meaningful. Scaling the sensitivity parameters by a nominal value of the physical/geometric parameter allows different sensitivity parameters to be compared directly with consistent units. Furthermore, normalizing by a characteristic value of the field variable provides a measure of the relative influence of a given design parameter on the overall behavior of the flow field. The normalized scaled sensitivity parameter is denoted here with '$\tilde{\ }$' and defined as

$$\tilde{S}_{u,\alpha}=\left(\frac{\overline{\alpha}}{\overline{u}}\right)\frac{\partial u}{\partial \alpha} \tag{5}$$

where the overline indicates the nominal or characteristic value associated with the baseline (unperturbed) flow. For a design parameter α to exert a significant influence on the flow field, the value of at least one of its normalized scaled sensitivity parameters (one sensitivity parameter for each of the field variables) must be of order unity or greater when compared to the respective normalized field variable with order unity [34].

3. NUMERICAL MODEL

The time evolution of incompressible viscous flow is described by the unsteady Navier–Stokes equations

$$\frac{\partial(\rho u_i)}{\partial t}+\frac{\partial(\rho u_i u_j)}{\partial x_j}=-\frac{\partial p}{\partial x_i}+\frac{\partial}{\partial x_j}\left[\mu\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\right] \tag{6}$$

subject to the constraint of conservation of mass,

$$\frac{\partial(\rho u_i)}{\partial x_i}=0 \tag{7}$$

where x_i denotes the spatial coordinates, u_i represents the components of the velocity field, ρ is the fluid density, and μ is the absolute viscosity of the fluid. Note, a lower case p is used herein to denote pressure, while an upper case P is used to denote a design parameter.

3.1. *CFD method*

In this study, the numerical method used to solve (6) and (7) is a projection method, also known as a fractional-step or time-splitting method, first proposed by Harlow and Welch [35] and Chorin [36], and applied to canonical CFD problems using finite volumes by Kim and Moin [37].

The fractional-step method is based on the Hodge decomposition whereby any vector field ($\hat{u}_i$) can be decomposed into a divergence-free vector field (u_i) and the gradient of a scalar potential (ϕ),

$$\hat{u}_i = u_i + \nabla\phi \tag{8}$$

Hodge decomposition is employed in two steps: (1) the momentum equations are integrated in time to compute an intermediate velocity field $\hat{u}_i$ by the following equation:

$$\frac{\hat{u}_i - u_i^n}{\Delta t} = \frac{3}{2}H_i^n - \frac{1}{2}H_i^{n-1} \tag{9}$$

where

$$H_i = -\frac{\partial(\rho u_i u_j)}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right] \tag{10}$$

and (2) the intermediate velocity field is projected onto a divergence-free velocity field using

$$\frac{u_i^{n+1} - \hat{u}_i}{\Delta t} = -\frac{1}{\rho}\frac{\partial\phi}{\partial x_i} \tag{11}$$

The superscripts n represent the time step. Note, the integration of the momentum equation for the intermediate velocity field does not contain the pressure gradient term. The intermediate velocity field will not, in general, satisfy (7). By taking the divergence of (11) and requiring that $\nabla\cdot u_i^{n+1} = 0$, an equation for the scalar potential is as follows:

$$\frac{\partial^2\phi}{\partial x_j \partial x_j} = \frac{\rho}{\Delta t}\nabla\cdot\hat{u}_i \tag{12}$$

In this manner, the divergence of the intermediate velocity field is used as a source term to solve for the gradient of the scalar potential. The scalar potential ϕ must be corrected [37] in order to retrieve the pressure field,

$$p = \phi - \frac{v\Delta t}{2}\nabla^2\phi \tag{13}$$

The current fractional-step algorithm utilizes the finite volume method with a staggered uniform Cartesian grid. The spatial derivatives are discretized using second-order central differencing, and the temporal derivatives are advanced in time using Adams–Bashforth time integration. The CFD algorithm is solved using the *dimensional form* of the governing equations to maintain the dependence on the physical and geometric parameters for sensitivity analysis. The algorithm solves the momentum equations explicitly and solves for the scalar potential implicitly using a conjugate gradient method. Because only steady-state solutions are considered in this study, temporal convergence is monitored at each iteration. The temporal convergence criterion is based on the magnitude of the residual, defined here as the infinity norm of the relative difference in the value of the variable of interest (χ) between the current and previous time steps,

$$\text{res} = \left\|\frac{\chi^n - \chi^{n-1}}{\chi^n}\right\|_\infty \tag{14}$$

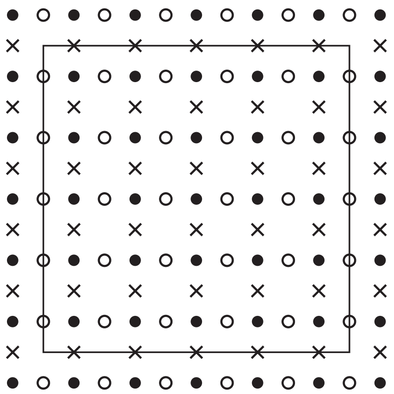

Figure 1. An example of staggered grid configuration with 5×5 cells. The solid lines represent the physical boundary. $\circ$ u-node, $\bullet$ p-node, and $\times$ v-node.

The CFD algorithm is solved on a staggered grid, an example of which is shown in Figure 1. Staggered grids are defined by separate node locations for each velocity component, as well as the scalar potential. The momentum equations are solved on the appropriate velocity grids to obtain the horizontal and vertical velocities at the u and v nodes, respectively; while, the scalar potential and continuity equations are solved on a third grid to obtain values at the p nodes (located at the cell centers of the combined velocity grids). Staggered grids are well suited to the finite volume approach in that conservation laws are evaluated over cells or control volumes. Staggered grids also tend to reduce the likelihood of convergence problems and oscillations in the pressure and velocity fields that can plague collocated grids [38]. In addition, with a staggered grid, the normal velocity at a boundary can be explicitly defined. The disadvantage, however, remains that tangential velocities at boundaries are not explicitly defined and must be approximated.

3.2. CFD algorithm verification

To verify the order of temporal and spatial discretization of the CFD method, the present code was used to numerically solve a periodic vortical flow that decays with time. The exact solution to the Navier–Stokes equations and conservation of mass for this particular flow is given by Kim and Moin [37]

$$u(x,y,t)=-\cos(x)\sin(y)\mathrm{e}^{-2t} \tag{15}$$

$$v(x,y,t)=\sin(x)\cos(y)\mathrm{e}^{-2t} \tag{16}$$

Note, the viscosity was taken as unity and the periodic boundary conditions were used in the simulation. The results of a grid refinement study are given in Figure 2, in terms of the grid size, Δx_i *versus* the L_2 norm of the error,

$$\varepsilon=\frac{\chi_\mathrm{a}-\chi_\mathrm{n}}{\chi_\mathrm{a}} \tag{17}$$

where χ represents the field variable of interest (in this case, the u velocity) and the subscripts a and n denote the analytical and numerical solutions, respectively. The manufactured analytical

DOI: 10.1002/fld

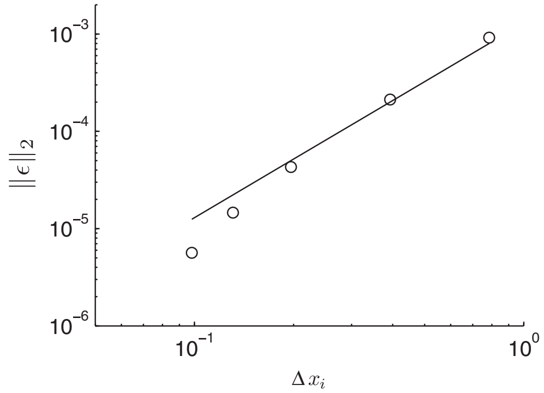

Figure 2. L_2 norm of u velocity error plotted against characteristic grid size for the test case of a decaying vortex. The Courant number remained constant for all data points. The solid line represents a second-order method.

solution for this flow is given by Kim and Moin [37]. The data in Figure 2 were obtained at $t=\pi/2$ with a uniform mesh having dimensions $\pi/2\leqslant x\leqslant 5\pi/2$ and $\pi/2\leqslant y\leqslant 5\pi/2$. All of the simulations were run with a constant ratio of $\Delta t/\Delta x$ (or Courant number). The results indicate that the present CFD algorithm follows second-order convergence for both temporal and spatial discretization.

3.3. Sensitivity equations

Equations (7), (6), and (12) may be differentiated with respect to parameter P to generate a new system of governing equations for the sensitivity parameters,

$$S_{u_i}=\frac{\partial u_i}{\partial P} \quad \text{and} \quad S_p=\frac{\partial p}{\partial P} \tag{18}$$

Using these definitions, the sensitivity equations are

$$\frac{\partial(\rho S_{u_i})}{\partial x_i}+\frac{\partial(\rho' u_i)}{\partial x_i}=0 \tag{19}$$

$$\begin{aligned}\frac{\partial(\rho S_{u_i})}{\partial t}+\frac{\partial(\rho' u_i)}{\partial t}+\frac{\partial(\rho' u_i u_j)}{\partial x_j}+\frac{\partial(\rho S_{u_i} u_j)}{\partial x_j}+\frac{\partial(\rho u_i S_{u_j})}{\partial x_j}\\ =-\frac{\partial S_p}{\partial x_i}+\frac{\partial}{\partial x_j}\left[\mu\left(\frac{\partial S_{u_i}}{\partial x_j}+\frac{\partial S_{u_j}}{\partial x_i}\right)\right]+\frac{\partial}{\partial x_j}\left[\mu'\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\right]\end{aligned} \tag{20}$$

The prime in Equations (19) and (20) represents the total derivative of the fluid properties ρ and μ with respect to P. Note, the governing equations for $S_{u,\mu}$ and $S_{u,\partial p/\partial x}$ contain additional source terms due to the presence of the physical parameters in the Navier–Stokes equations. Equations

(19) and (20) are solved using the same numerical algorithm presented in Section 3.1. The pressure sensitivity is obtained in a manner similar to (12) and (13),

$$\frac{\partial S_\phi}{\partial x_j \partial x_j} = \frac{\rho}{\Delta t} \nabla \cdot \hat{S}_{u_i} \tag{21}$$

$$S_p = S_\phi - \frac{v \Delta t}{2} \nabla^2 S_\phi \tag{22}$$

3.4. Boundary conditions for SEM

The boundary conditions necessary to solve the sensitivity equations come from implicit differentiation of the corresponding momentum and pressure boundary conditions, which may depend on one or more physical and/or geometric parameters. Consider the following non-homogeneous Dirichlet boundary condition for the velocity:

$$u(x_\mathrm{b}(P), y_\mathrm{b}(P); P) = c(x_\mathrm{b}(P), y_\mathrm{b}(P); P) \quad \text{on } \Gamma \tag{23}$$

where the subscript b denotes the boundary value. Differentiating (23) with respect to the parameter P gives

$$\frac{\partial u}{\partial P} + \frac{\partial u}{\partial x}\frac{\partial x_\mathrm{b}}{\partial P} + \frac{\partial u}{\partial y}\frac{\partial y_\mathrm{b}}{\partial P} = \frac{\partial c}{\partial P} + \frac{\partial c}{\partial x}\frac{\partial x_\mathrm{b}}{\partial P} + \frac{\partial c}{\partial y}\frac{\partial y_\mathrm{b}}{\partial P} \tag{24}$$

Solving (24) for $\partial u/\partial P$ (i.e. S_u) yields

$$S_u = \left[\frac{\partial c}{\partial P} + \frac{\partial c}{\partial x}\frac{\partial x_\mathrm{b}}{\partial P} + \frac{\partial c}{\partial y}\frac{\partial y_\mathrm{b}}{\partial P}\right] - \frac{\partial u}{\partial x}\frac{\partial x_\mathrm{b}}{\partial P} - \frac{\partial u}{\partial y}\frac{\partial y_\mathrm{b}}{\partial P} \quad \text{on } \Gamma \tag{25}$$

where the term inside the square bracket may be expressed as a total derivative, $D/DP \equiv \partial/\partial P + (\partial \mathbf{x}_\mathrm{b}/\partial P) \cdot \nabla$. If P represents a geometric parameter (height of the domain, for instance) information about the velocity gradients evaluated at the boundary are required, as well as the boundary parameterization $\partial \mathbf{x}_\mathrm{b}/\partial P$. For the case of a non-homogeneous Neumann boundary condition, consider the example

$$\frac{\partial u}{\partial y} = c(x_\mathrm{b}(P), y_\mathrm{b}(P); P) \quad \text{on } \Gamma \tag{26}$$

Differentiating (26) with respect to the parameter P yields

$$\frac{\partial S_u}{\partial y} = \frac{Dc}{DP} - \nabla\left(\frac{\partial u}{\partial y}\right) \cdot \left(\frac{\partial x_\mathrm{b}}{\partial P}, \frac{\partial y_\mathrm{b}}{\partial P}\right) \quad \text{for } x = 0 \tag{27}$$

These examples show that the boundary conditions for the sensitivity equations match the same type as that of the corresponding field variable (Dirichlet or Neumann), and may contain additional terms associated with the gradient of the field variable, the boundary parameterization associated with geometric sensitivities, and/or parameter dependence in the boundary value of the field variable. Further examples are given by Turgeon *et al.* [5] and Hristova *et al.* [9]. In general, initial conditions should be handled in the same manner as described above for the boundary conditions.

4. CHANNEL FLOW TEST CASE

Although limited to a few specific cases, exact solutions to the Navier–Stokes equations [39, 40] are used extensively in the CFD verification process. The method of manufactured solutions is another approach commonly used in CFD verification [9, 10, 41], see also Section 3.2. In this study, verification of the finite volume numerical methods for computing sensitivity parameters was performed using two-dimensional, low Reynolds number flow through a straight channel, which has an exact solution to the Navier–Stokes equations under steady-state, fully developed conditions. Channel flow was chosen as a test case here because it represents a canonical wall-bounded flow and, as such, constitutes a suitable stepping stone for more complicated turbulent boundary layer flows, which are of primary interest to the authors. Channel flow also involves an inherent dependence of the flow on the channel height, which allowed geometric sensitivities to be explored, as well as parameter dependence of the inlet velocity boundary condition.

4.1. Exact solution for steady-state, fully developed flow

Channel flow, also known as Poiseuille flow in honor of J. Poiseuille [42], is characterized by unidirectional flow between two stationary, parallel, flat plates separated by a distance h, whereby an externally applied streamwise pressure gradient drives the flow. The continuity and momentum equations are

$$\frac{\partial u}{\partial x}=0 \quad \text{and} \quad \mu\frac{\partial^2 u}{\partial y^2}=\frac{\partial p}{\partial x} \tag{28}$$

respectively, with boundary conditions

$$u(0)=0 \quad \text{and} \quad u(h)=0 \tag{29}$$

The solution to Poiseuille flow is

$$u(y)=\frac{-h^2}{2\mu}\frac{\partial p}{\partial x}\left[\frac{y}{h}-\left(\frac{y}{h}\right)^2\right] \tag{30}$$

The exact solution for the streamwise velocity u, as expressed in (30), depends on three design parameters: the fluid viscosity μ, the imposed pressure gradient $\partial p/\partial x$, and the gap height h. Analytic expressions for the sensitivity of u to these parameters are obtained by differentiating (30),

$$S_{u,\mu}=\frac{h^2}{2\mu^2}\frac{\partial p}{\partial x}\left[\frac{y}{h}-\left(\frac{y}{h}\right)^2\right] \tag{31}$$

$$S_{u,\partial p/\partial x}=\frac{-h^2}{2\mu}\left[\frac{y}{h}-\left(\frac{y}{h}\right)^2\right] \tag{32}$$

$$S_{u,h}=\frac{-y}{2\mu}\frac{\partial p}{\partial x} \tag{33}$$

The corresponding scaled normalized sensitivity parameters are

$$\tilde{S}_{u,\mu}=-6\left[\frac{y}{h}-\left(\frac{y}{h}\right)^2\right], \quad \tilde{S}_{u,\partial p/\partial x}=6\left[\frac{y}{h}-\left(\frac{y}{h}\right)^2\right] \quad \text{and} \quad \tilde{S}_{u,h}=6\frac{y}{h} \tag{34}$$

using normalizations based on the characteristic value of each parameter in the baseline (unperturbed) case, as well as the average velocity,

$$U=\frac{-h^2}{12\mu}\frac{\partial p}{\partial x} \tag{35}$$

All the normalized scaled sensitivity parameters in (34) have an order of magnitude of unity near the center of the channel. The $\partial p/\partial x$ and μ sensitivities have similar profiles but opposite sign, with a maximum absolute value of 1.5 along the channel centerline. Because the no-slip condition remains enforced at the upper and lower boundaries, increasing μ and $\partial p/\partial x$ will necessarily cause a nonlinear change in the u velocity across the gap. Owing to the symmetry of the domain, the maximum change would be expected along the channel centerline. The h-sensitivity parameter increases linearly with height across the channel, exhibiting a maximum value of 6 at the top boundary and a minimum value of 0 at the lower boundary. A positive perturbation in h and $\partial p/\partial x$ translates into an increase in the overall mass flow rate through the channel; while, the opposite is true for a positive perturbation in μ. It is important to realize that the mass flow rate (i.e. U) is not an independent design parameter in this scenario. This will have important ramifications with regard to the boundary conditions used in SEM (as discussed further in Section 4.3).

4.2. Numerical implementation

Numerical simulations for Poiseuille flow were performed using the CFD algorithm described in Section 3.1 at a Reynolds number based on average velocity of $Re_h=150$ (using $h=0.5\,\mathrm{m}$, $\mu=1.78\times10^{-5}\,\mathrm{kg\,m^{-1}\,s^{-1}}$, and $\rho=1.22\,\mathrm{kg\,m^{-3}}$). The computational domain spans $0\leqslant x/h\leqslant 12$ and $0\leqslant y/h\leqslant 1$, as shown in Figure 3. Simulations were run using grid resolutions of 10×120, 20×240, 30×360, 40×480, and 50×600. At each grid resolution, steady-state numerical solutions were obtained at four different temporal convergence criterions: 10^{-3}, 10^{-4}, 10^{-5}, and 10^{-6}. As indicated in Figure 3, the no-slip condition is enforced along the upper and lower boundaries, a uniform (average) velocity is specified at the inlet, and a homogeneous Neumann boundary condition in the x direction is maintained at the outlet for both velocity components. The boundary conditions for the scalar potential ϕ are homogeneous Neumann for all boundary faces with the exception of the outlet, where a homogeneous Dirichlet boundary condition is employed. The initial condition is $u(x,y)=U$ and $v(x,y)=0$.

Boundary conditions for the SEM follow according to Section 3.4 and have similar form as the velocity boundary conditions, with the exception of the inlet boundary condition, which possesses functional dependence on all three parameters, see Equation (35). Also the inlet and upper wall boundary conditions for the height sensitivity include the boundary parameterization as well as the velocity gradients, as discussed further in Section 4.3. At the inlet and upper wall, the velocity gradient must be computed and, due to the staggered grid, interpolated back to the velocity nodes. The initial condition for the sensitivity derivatives is $S_u(x,y)=S_v(x,y)=0$. Note, in general, the initial conditions should be handled similar to the boundary conditions, as discussed in Section 3.4. However, for simplicity here, homogeneous initial conditions were employed. This may have resulted in slightly slower convergence rates in the SEM solutions as reported in Section 5.2.

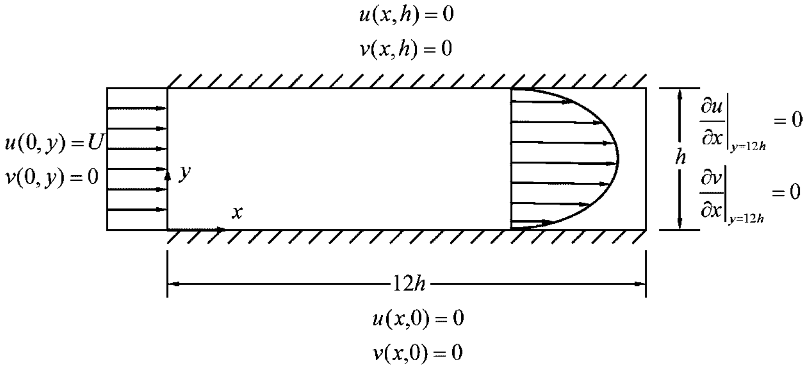

Figure 3. Schematic of the computational domain and velocity boundary conditions used in the finite volume solution.

4.3. Parameter dependence in the boundary condition

In the process of calculating the boundary conditions for SEM, particular care must be taken due to the terms in (25). Specifically, the inlet boundary condition for $S_{u,h}$ is given by

$$S_{u,h}(0,y)=\frac{\partial U}{\partial h}-\left(\frac{\partial y_{\mathrm{b}}}{\partial h}\right)\left.\frac{\partial u}{\partial y}\right|_{x=0} \tag{36}$$

The first term on the right-hand side of (36) is non-zero due to the functional dependence of the inlet velocity on the geometric parameter h, see (35). Recall, U is not an independent design parameter in this problem. The second term on the right-hand side of (36) is also non-zero. This follows from the fact that $\partial u/\partial y\neq 0$ at the upper and lower boundaries where the no-slip condition remains enforced, and from the boundary parameterization

$$y_{\mathrm{b}}=h\left(\frac{y}{\bar{h}}\right) \tag{37}$$

it follows that $\partial y_{\mathrm{b}}/\partial h=y/\bar{h}$. The sensitivity equations for μ and $\partial p/\partial x$ also have non-homogeneous Dirichlet inlet boundary conditions due to the functional dependence of U on these parameters. If the parameter dependence of U is neglected (or unknown), then incorrect boundary condition values will cause gross errors in the SEM solutions. The above observations highlight the importance of understanding parameter dependence in SEM, especially with regard to the boundary conditions on the equations governing the sensitivity derivatives.

4.4. Alternative design parameter scenarios

In this study, the design parameters are considered to be h, μ, and $\partial p/\partial x$. However, it is equally probable that one might consider U (the average velocity) as a design parameter, instead of the pressure gradient and viscosity. In this alternative scenario, the mass flow rate is controlled, while the pressure gradient and viscosity take on whatever values are necessary in order to satisfy (35) in the fully developed region. The aforementioned problems with regard to parameter dependence in the inlet boundary condition are not relevant in this scenario. The scaled normalized sensitivity parameter for U looks identical to that for the pressure gradient given in (34). However, the scaled

DOI: 10.1002/fld

normalized h sensitivity parameter is $\tilde{S}_{u,h} = 6[2(y/h)^2 - y/h]$, which is much different than the linear relationship given in (34). The ramifications are similar to the statements at the end of the previous section.

Another interesting point should be stated regarding the height sensitivity. If a transformation is made to shift the origin of the coordinate system to the channel centerline ($\zeta = y - h/2$), then the scaled normalized h sensitivity becomes $\tilde{S}_{u,h} = 3$, independent of the vertical position within the gap. This follows directly from differentiating the transformed exact solution, $u(\zeta) = 6U[-(\zeta/h)^2 + 1/4]$. The interpretation is that if the lower and upper boundaries of the channel are increased by an amount $\Delta h/2$, then the u velocity will increase uniformly across the channel gap. This contrasts the previous scenario (34) wherein the lower boundary remains fixed and the upper boundary moves an amount Δh, resulting in a maximum increase in the u velocity near the top boundary and minimal increase near the lower boundary. This will have ramifications on the use of sensitivity derivatives to estimate uncertainty bounds on the numerical velocity solution due to uncertainty in measurements of the channel height (as discussed further in Section 6.2).

5. NUMERICAL RESULTS

5.1. *Verification of velocity and pressure*

The numerical channel flow results for velocity and pressure using the 40×480 grid are shown in Figure 4 along with the exact solution for the fully developed case (30). The accuracy of the numerical velocity profile was found to depend more on the temporal convergence criterion rather than the grid resolution. Specifically, the error between the exact and numerical solutions had a maximum value of 2.5% for all grid resolutions tested herein, as long as res$\leqslant 10^{-5}$. The numerical pressure solution exhibits fully developed behavior for $x/h \geqslant 8$, as evinced by the constant pressure gradient. This agrees with the entry length prediction of $Le/h \approx 9$, based on the relation $Le/h \approx 0.06 Re_h$ from the integral momentum analysis of Sparrow *et al.* [43].

Table I shows the apparent order of the CFD/SEM numerical scheme for two different temporal convergence criteria: 10^{-5} and 10^{-6}. The apparent order of the scheme $\hat{p}$ (not to be confused

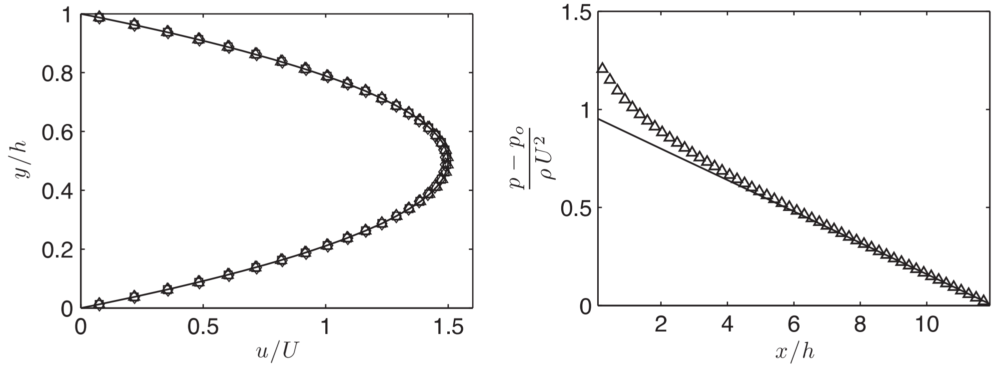

Figure 4. Numerical solution of the u velocity at $x/h = 11.4$ (left) and coefficient of pressure at $y/h = 0.5$ (right) using the 40×480 grid with temporal convergence criterion: ◇, 10^{-4}; □, 10^{-5}; △, 10^{-6}. The exact solution for steady-state, fully developed flow is denoted by the solid line.

DOI: 10.1002/fld

Table I. Apparent order of the numerical scheme for the CFD/SEM simulations.

Convergence (res)	Velocity (u)	Sensitivity		
		$S_{u,\mu}$	$S_{u,h}$	$S_{u,\partial p/\partial x}$
10^{-5}	4.42	2.03	0.72	2.04
10^{-6}	4.49	1.95	0.88	1.97

with the pressure p) is calculated using three of the five grid resolutions studied herein (10×120, 20×240, 40×480) according to [15] as

$$\hat{p} = \ln\left|\frac{f_3 - f_2}{f_2 - f_1}\right| \Big/ \ln(r) \tag{38}$$

where f_k denotes the numerical solution using the kth grid, with $k=1$ representing the finest grid resolution, and r denotes the ratio of grid spacings between successive grid refinements (in this case, $r=2$). The values listed in Table I represent the average value of $\hat{p}$ at $x/h=11.4$. Recall, the formal order of accuracy of the present finite volume scheme is two. Since the present flow evolves temporally, (38) was evaluated at a fixed residual. Interestingly, the apparent order of accuracy of the u velocity is about four. However, if the numerical simulation is run using a higher viscosity ($\mu=0.01\,\mathrm{kg\,m^{-1}\,s^{-1}}$ as opposed to $\mu=1.78\times 10^{-5}\,\mathrm{kg\,m^{-1}\,s^{-1}}$), then the observed order of accuracy reduces to two. This is due to the fact that the coarse-grid (10×120) simulation with the lower μ value has significantly higher error compared with the coarse-grid simulation with the higher μ value; whereas, the finer-grid simulations for both μ values showed very similar magnitude of error. Therefore, the increased magnitude of error in the coarse-grid simulation anomalously increases the apparent order of accuracy for the lower viscosity simulation. The μ and $\partial p/\partial x$ sensitivity parameters have an apparent order of accuracy of two; while, the h sensitivity parameter has an apparent order of accuracy less than one. Duvigneau and Pelletier [25] also discusses the difficulty in achieving accurate results for geometric sensitivities using SEM coupled with a *finite element* discretization approach.

5.2. Verification of sensitivity parameters

Sensitivity parameters were computed using finite difference, complex-step differentiation, and SEM. Numerical simulations were performed using the same geometry, grid resolutions, and temporal convergence criteria as described in Section 4.2. Results for the scaled normalized sensitivity parameters are shown in Figure 5 for the 40×480 grid with a convergence criteria of 10^{-5} on all quantities. All of the simulations exhibit excellent agreement with the exact solutions for the sensitivities given in (31)–(33). Note that h sensitivity was not computed using complex-step differentiation. As with the numerical solution for the velocity field, error in the sensitivity parameters seemed to depend more on the temporal convergence criterion, rather than grid resolution.

The influence of the parameter step size ΔP on the solutions for S_u using finite difference and complex-step differentiation methods is shown in Figure 6. Results were computed using the

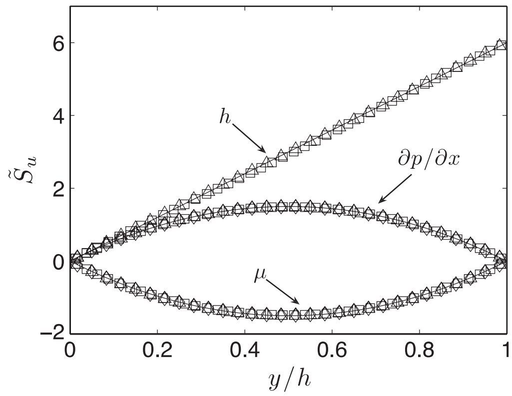

Figure 5. Exact solution (solid lines) of the scaled normalized sensitivity parameters compared with the numerical solution. ◇ Complex-step differentiation, △ central finite difference, and □ SEM.

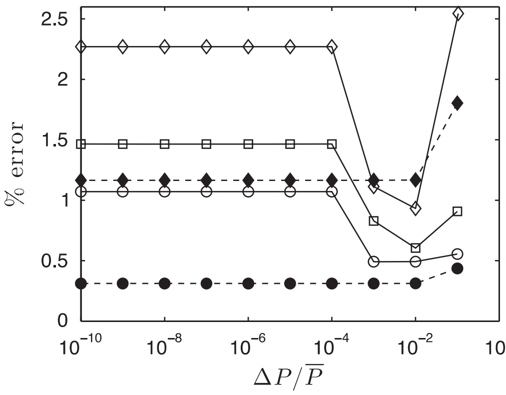

Figure 6. Parameter step size *versus* numerical error. Method: solid symbols, complex-step differentiation; open symbols, finite difference. Sensitivity parameter: ○ $S_{u,\partial p/\partial x}$, □ $S_{u,h}$, ◇ $S_{u,\mu}$.

40×480 grid with a temporal convergence criterion of 10^{-5} on the u velocity. The percent error is defined as

$$\%\ \text{error} = 100|\varepsilon| \tag{39}$$

with ε as given in (17). The error is only calculated in the fully developed region wherein the analytical solutions are valid. In the finite difference method, the error starts out relatively high for large ΔP. As ΔP decreases, the error decreases to a local minimum and then increases again to a constant value with further reduction in ΔP. Although the magnitude of the error and the local minimum varies for the different sensitivity parameters, the trend remains the same. Therefore, in order to ensure minimal error in the finite difference calculation, a parameter study (using a range of ΔP) must be performed for each sensitivity parameter of interest. In contrast, the

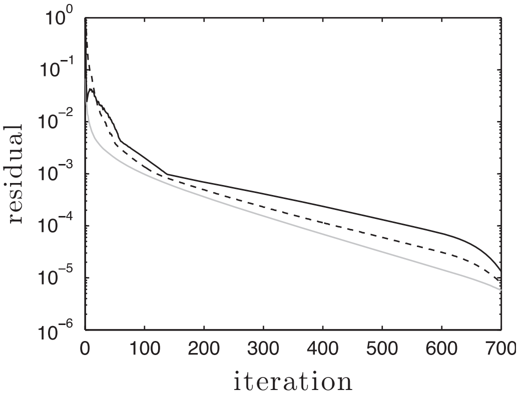

Figure 7. Residual curves in the SEM simulation: — u velocity, – $S_{u,\mu}$, -- $S_{u,h}$.

error associated with the complex-step differentiation method decays to a global minimum as ΔP decreases. This represents one of the main advantages of the complex-step differentiation method. A parameter study in terms of ΔP is unnecessary, as long as the step size remains small enough ($\Delta P < 1\times10^{-3}$ in this study, but Martins *et al.* [26] suggest $\Delta P < 1\times10^{-8}$). In this test case, for any given sensitivity parameter, the error in complex-step differentiation always remains less than that of the finite difference method for $\Delta P/\bar{P} < 1\times10^{-4}$. The present results agree with previous studies using *finite element*-based CFD algorithms for calculating sensitivity parameters [12, 26].

The residual curves of the u velocity, as well as the μ and h sensitivity parameters computed from SEM, are shown in Figure 7 for the 40×480 grid. The residual for the pressure gradient sensitivity follows similar trends as the other two sensitivity parameters show. While the curves display monotonic convergence trends for the explicit solve, the SEM solutions for $S_{u,\mu}$ and $S_{u,h}$ converge at a slightly slower rate than the solution for the velocity. Specifically, the residual of u decreases one decade (from 1×10^{-3} to 1×10^{-4}) in 285 iterations, whereas the residuals of the h and μ sensitivities require 330 and 410 iterations, respectively, for the same reduction in residual. Terminating the CFD/SEM solve when the residual of u reaches 1×10^{-4} results in a maximum error in $S_{u,\mu}$ of 20.54%, which decreases to 8.13% by allowing the residual of $S_{u,\mu}$ to reduce to 1×10^{-4}. For res $< 1\times10^{-4}$, the rate of convergence of the sensitivity solutions increases substantially. When the residual of $S_{u,\mu}$ reaches 1×10^{-5}, the maximum error in that quantity measures only 1.25%. From these results, it is recommended that the global convergence of the SEM computation be based on the criterion that the residual of all quantities, including field variables and sensitivity parameters, achieve the specified tolerance level. The implication is that finite volume-based SEM may require additional computational time beyond that for the velocity field alone.

Figure 8 shows the L_2 norm of the exact error in the CFD/SEM solutions as a function of temporal convergence criterion and parameterized by grid resolution. Results for the u velocity and $S_{u,\partial p/\partial x}$ indicate that the most accurate solutions are achieved with the 40×480 grid, and that negligible improvement in accuracy is obtained for residuals less than 10^{-5}. Furthermore, for residual values less than 10^{-5}, the error follows asymptotic convergence with respect to grid resolution. Results for μ and h sensitivities (not shown) follow similar trends.

DOI: 10.1002/fld

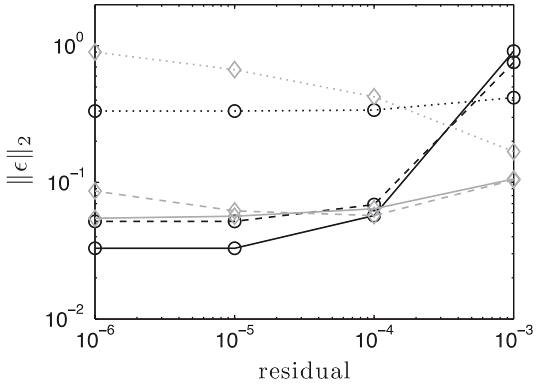

Figure 8. L_2 norm of the error in the CFD/SEM solutions at $x/h = 11.4$ as a function of residual and grid spacing. ○ u velocity, ◇ $S_{u,\partial p/\partial x}$, ··· 10×120 grid, −− 20×240 grid, – 40×480 grid.

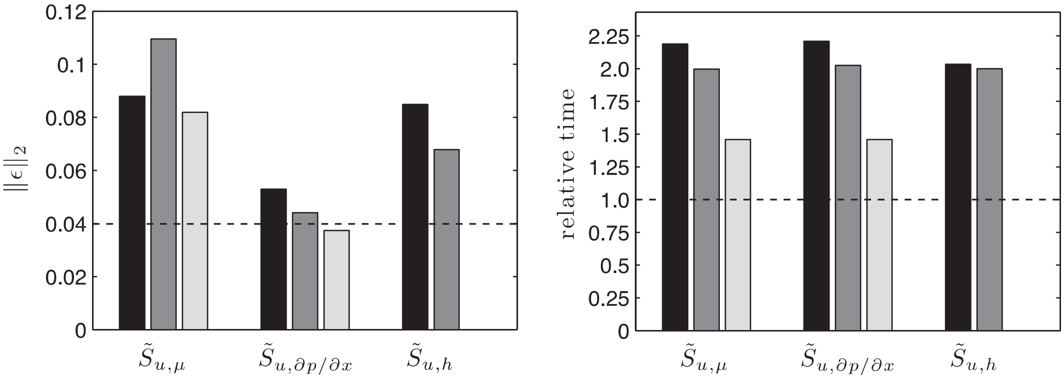

Figure 9. Performance comparison of the three different numerical sensitivity methods: (left) L_2 norm of the exact error, (right) normalized computational time. ■ SEM, ■ finite difference, ■ complex-step differentiation, −− benchmark from the CFD solution for the u velocity.

5.3. Comparison of sensitivity methods

Figure 9 compares the L_2 norm of the exact error in the fully developed numerical solutions of the scaled sensitivity parameters using the three different methods: finite difference, complex-step differentiation, and SEM. For reference, $\|\varepsilon\|_2$ based on the u velocity is indicated with a horizontal dashed line. All simulations utilized a grid of 40×480. For the finite difference method, the optimum value of $\Delta P/\overline{P}$ that yielded the minimum error was used. For complex step differentiation, a value of $\Delta P/\overline{P} = 1 \times 10^{-6}$ was used. In SEM, all quantities were allowed to converge to a residual value of 1×10^{-6}. Of the cases tested herein, complex step exhibits the least error. SEM exceeds finite difference in terms of error in both the height and pressure gradient sensitivities. Regardless of the method, the error in the h and μ sensitivities is about twice the error of the u velocity; while, the error in the $\partial p/\partial x$ sensitivity is about the same as that of the u velocity.

DOI: 10.1002/fld

The computational time for each sensitivity method, relative to the time required to compute the primitive variables, is also shown in Figure 9. These times represent one sensitivity parameter solve per numerical solution of the velocity field, and do not include post-processing time (as described further below). A first-order forward difference approximation of the sensitivity coefficients requires one solve for the baseline case, along with one additional solve for each of the design parameters, using the perturbed value of that parameter. To generate second-order sensitivity coefficients using the finite difference method requires two solves for each of the design parameters, perturbed above and below the corresponding baseline value. The finite difference time presented in Figure 9 represents second-order (central-difference) calculations, wherein the CFD algorithm was started from the same initial conditions as used in the baseline flow. As such, the computational time for the finite difference sensitivity method is approximately two times the order of a single primitive variable solve. In contrast, when the CFD algorithm is started using initial conditions based on the converged baseline solution, the computational time of the finite difference sensitivity method for one perturbed design parameter (relative to the time to solve the baseline case) decreases from 2.0 to 1.46.

The SEM solves a set of equations similar to that of the primitive variables. As such, the same numerical algorithms are employed for SEM as that for the primitive variables. In the specific case of the fractional-step algorithm used here, the majority of the computational time is spent implicitly solving the Poisson equation for the scalar potential. The time required to solve the Poisson equation in a fractional-step algorithm (for the primitive variables) can consume up to 90% of the total computational time when using iterative methods [44]. For the case where the diffusion terms are solved using the Crank–Nicholson method as well, the time required to solve the scalar potential Poisson equation is still more than two-thirds of the total computational time [45]. For the current work, the advection and diffusion terms (in the momentum and sensitivity equations) are solved explicitly in the fractional-step algorithm. The fact that the sensitivity equations are linear compared with the nonlinear momentum equations provides no additional computational time efficiency. The SEM, therefore, involves a computational solve time of the order of the primitive variable solution, as shown in the data in Figure 9. The complex-step method, while involving complex arithmetic, requires only the complex version of the baseline primitive variable solve. For the present fractional-step algorithm, the complex-step computational solve time is on the order of 1.5 times the primitive variable solution alone.

The data in Figure 9 represent the relative computational time only for one sensitivity parameter solve. For comparison, the data of Martins *et al.* [46], when reformulated in this way, show that the average computational time (relative to the baseline solution) per sensitivity solve per design parameter using the finite difference and complex-step methods is 1.79 and 1.91, respectively. The results from the present study indicate that the average computational time per sensitivity solve for the finite difference (initiated with the converged baseline solution following the approach of Martins *et al.* [46]) and complex-step methods is 1.46 and 1.50, respectively, which is not inconsistent with Martins *et al.* [46]. Note, however, that the CFD algorithm in Martins work is different from that used in this study, and therefore discrepancies in computational time are to be expected.

With respect to ease of implementation, finite difference has a clear advantage in that the values of the parameters of interest only need to be perturbed and the governing equations solved (once for each different parameter set) without any modification to the preexisting CFD code. Post-processing of the data to obtain the actual sensitivity parameters, however, requires additional computational time, and when considering geometric sensitivities, special care must be taken with

DOI: 10.1002/fld

interpolation to guarantee comparison of data at the same location in the mesh. In addition, the finite difference method requires a parameter study in ΔP and may not be appropriate for flows having no steady-state solution. The complex-step method is also relatively easy to implement in the framework of MATLAB (used for this study), and post-processing of the data only requires simple algebraic function evaluations. Furthermore, complex step does not require a parameter study in ΔP. Perhaps the most difficult method to implement was SEM, mostly due to the addition of source terms in the governing equations for some of the sensitivity parameters and the presence of parameter dependence in the boundary conditions. SEM does not, however, require any post-processing. In summary, complex-step differentiation appears to have some benefit over finite difference.

6. UNCERTAINTY ANALYSIS

The two types of uncertainty in the numerical solution considered here are numerical uncertainty and input uncertainty. Numerical uncertainty refers to an estimate of the exact error between the numerical solution and the exact solution. Input uncertainty refers to inexact measurement of parameters required as input to the numerical algorithm (e.g. fluid properties, dimensions of the geometry, inlet boundary conditions) and the propagation of that measurement uncertainty through the numerical simulation. Assuming zero uncertainty in the model, the sum of the numerical and input uncertainties yields the total uncertainty in the numerical solution.

6.1. Numerical uncertainty

Since analytical solutions are not available in most practical applications, numerical uncertainty bounds in the computational simulation must be estimated. The performance of the Grid Convergence Index (GCI) of Roache [47] as an error estimator for the sensitivity derivatives computed using SEM with the present finite volume approach is evaluated. GCI is defined as

$$\mathrm{GCI}=\frac{F_{\mathrm{s}}}{r^{\hat{p}}-1}\left|\frac{f_2-f_1}{f_1}\right| \tag{40}$$

where $\hat{p}$ is the apparent order of the scheme as given in (38) and F_{s} denotes a factor of safety. Figure 10 compares GCI with the exact error in u and S_u (computed with SEM) for residual tolerances of 10^{-5} and 10^{-6} using grid resolutions of 40×480 and 20×240. A value of $F_{\mathrm{s}}=3$ was used, as suggested by Roache [15], but may be more conservative than necessary for some of the sensitivities. Each data point in Figure 10 represents one node value along the line $x/h=11.4$. The average difference between the exact error and the GCI error estimator is 2.5%, based on an ensemble average of all the data shown in Figure 10. In particular, GCI tends to overestimate the error in $S_{u,h}$, while slightly underestimating the error in the u velocity and the two other sensitivity parameters. The GCI error does not depend significantly on the value of the residual (at least for the residual tolerances of 10^{-5} and 10^{-6} used here), except for u whereby GCI underestimates the true error twice as much with res$\leqslant 10^{-5}$ compared with res$\leqslant 10^{-6}$.

6.2. Input uncertainty

The previous sections have focused on *verification* of the numerical solution with respect to an exact solution. In the process of *validation* with experimental results, it is important to

DOI: 10.1002/fld

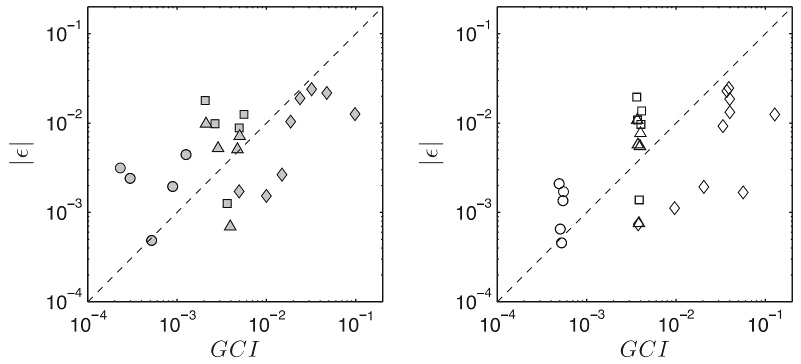

Figure 10. Comparison of the GCI error estimator *versus* exact error: (left) res$\leqslant 10^{-5}$ and (right) res$\leqslant 10^{-6}$. Symbol shapes: $\circ$ u, $\square$ $S_{u,\mu}$, $\diamond$ $S_{u,h}$, $\triangle$ $S_{u,\partial p/\partial x}$.

quantify the propagation of input data uncertainty into the numerical solution. For example, measurement of design parameters in the experiment is only accurate to within a given uncertainty, say $P \pm \Delta P$. Following Turgeon *et al.* [7], the resultant uncertainty bounds (worst-case scenario) on the numerical solution can be calculated using a first-order Taylor series expansion of the form

$$|u(x, y; P+\Delta P) - u(x, y; P)| \leqslant \sum_{i=1}^{3} |S_u(x, y; P_i)\Delta P_i| \tag{41}$$

where $P = (\mu, \partial p/\partial x, h)$. For the purpose of calculating bounds on the numerical solution due to input uncertainty, a value of 1.0% was used for $\Delta P/\overline{P}$.

Figure 11 shows the input uncertainty as error bars. The magnitude of the numerical uncertainty, in this case, is negligible relative to the input uncertainty. Therefore, error bars corresponding to the numerical uncertainty are omitted in Figure 11, since they fall within the thickness of the lines on the plot. In addition, uncertainty in the measurement of the geometry (i.e. channel height) dominates the overall input uncertainty in this case. One can see this directly from Figure 5, where it is observed that the μ and $\partial p/\partial x$ sensitivities are, on average across the gap, half of the magnitude of the u velocity; on the other hand, the h sensitivity is, on average, three times greater than the velocity field. Therefore, the flow is nearly an order of magnitude more sensitive to perturbations in the channel height.

The asymmetry in the error bars on the left plot stems from the form of the boundary parameterization, as given in (37). Because the lower wall is assumed fixed at $y = 0$, all of the input certainties in the channel height translate into an uncertainty in the location of the upper wall. This directly leads to negligible uncertainty in the velocity near $y = 0$ and maximum uncertainty near $y = h$. In the right plot of Figure 11, the coordinate system has been moved to the channel centerline. In this scenario, the input uncertainty in the channel height is split equally into uncertainties in the location of both the upper and lower walls, which leads to symmetric error bars. The latter scenario makes more sense in terms of validating the numerical simulation with experimental data.

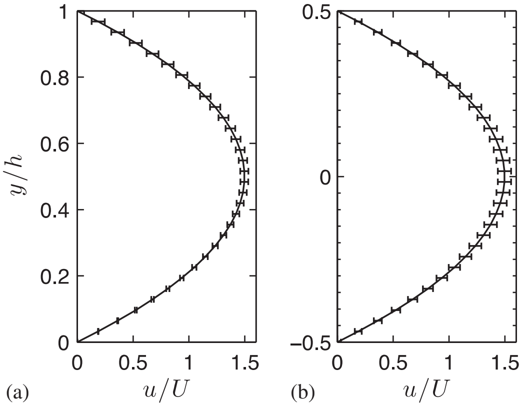

Figure 11. Uncertainty in the fully developed velocity profile. Origin located at (a) lower boundary and (b) channel center. Error bars represent the input uncertainty.

7. CONCLUSIONS

A finite volume-based fractional-step computational fluid dynamics (CFD) algorithm has been developed for the purpose of computing sensitivity parameters. Three different methods for generating sensitivity parameters, including finite difference, complex-step differentiation, and continuous sensitivity equation method (SEM), were compared in terms of accuracy and computational efficiency. The algorithms were verified using unsteady, two-dimensional flow through a straight channel, which has an exact solution under steady-state, fully developed conditions. This particular flow field possesses parameter dependence in the governing equations, as well as the boundary conditions. The design parameters of interest are the fluid viscosity, imposed pressure gradient, and channel height. Results indicate that errors in the sensitivity parameters are generally of the order of magnitude of the error in the velocity field, with the different sensitivity methods producing similar order of magnitude error for each parameter. Of all the three sensitivity methods, complex step exhibited the least error and required the least computational time. The Grid Convergence Index was shown to be an appropriate error estimator for the sensitivity parameters, when used with a factor of safety of 3. Sensitivities were used to illustrate how uncertainty in the actual values of the design parameters propagates into the numerical solution for the velocity field.

ACKNOWLEDGEMENTS

This work was supported by the National Science Foundation (IIS-0428856). The authors appreciate conversations with J. Burns, who introduced them to the field of sensitivity analysis.

REFERENCES

1. Borggaard J. The sensitivity equation method for optimal design. *Ph.D. Thesis*, Virginia Tech, Blacksburg, VA, 1994.
2. Borggaard J, Burns J. A PDE sensitivity equation method for optimal aerodynamic design. *Journal of Computational Physics* 1997; **136**(2):366–384.

3. Turgeon É, Pelletier D, Borggaard J. A continuous sensitivity equation approach to optimal design in mixed convection. *Numerical Heat Transfer*, *Part A* 2000; **38**:869–885.
4. Dowding K, Blackwell B. Sensitivity analysis for nonlinear heat conduction. *Journal of Heat Transfer* 2001; **123**:1–10.
5. Turgeon É, Pelletier D, Borggaard J. A general continuous sensitivity equation formulation for complex flows. *Numerical Heat Transfer*, *Part B* 2002; **42**:485–498.
6. Stanley L, Stewart D. *Design Sensitivity Analysis*. SIAM: Philadelphia, PA, 2002.
7. Turgeon É, Pelletier D, Borggaard J. Applications of continuous sensitivity equations to flows with temperature-dependent properties. *Numerical Heat Transfer*, *Part A* 2003; **44**:611–624.
8. Blackwell BF, Dowding KJ. Sensitivity analysis and uncertainty propagation of computational models. In *Handbook of Numerical Heat Transfer* (2nd edn), Minkowycz W, Sparrow E, Murthy J (eds). Wiley: New York, 2006; 443–470.
9. Hristova H, Étienne S, Pelletier D, Borggaard J. A continuous sensitivity equation method for time-dependent incompressible laminar flows. *International Journal for Numerical Methods in Fluids* 2006; **50**:817–844.
10. Duvigneau R, Pelletier D. On accurate boundary conditions for a shape sensitivity equation method. *International Journal for Numerical Methods in Fluids* 2006; **50**:147–164.
11. Turgeon É, Pelletier D, Borggaard J. Sensitivity and uncertainty analysis for variable property flows. *Proceedings of the 39th Aerospace Sciences Meeting*, Reno, NV, 2001; AIAA Paper 2001-0139.
12. Blackwell BF, Dowding KJ. Sensitivity and uncertainty analysis for thermal problems. *Fourth International Conference on Inverse Problems in Engineering*, Rio de Janeiro, Brazil, 2002.
13. Putko M, Taylor A, Newman P, Green L. Approach for input uncertainty propagation and robust design in CFD using sensitivity derivatives. *Journal of Fluids Engineering* 2002; **124**:60–69.
14. Freitas C. Journal of Fluids Engineering editorial policy statement on the control of numerical accuracy. *Journal of Fluids Engineering* 1993; **115**:330–340.
15. Roache P. Quantification of uncertainty in computational fluid dynamics. *Annual Review of Fluid Mechanics* 1997; **29**:123–160.
16. Mehta U. Credible computational fluid dynamics simulations. *AIAA Journal* 1998; **36**:665–667.
17. AIAA. Guide for the verification and validation of computational fluid dynamics simulations. *Technical Report*, *AIAA G-077-1998*, 1998.
18. Freitas CJ. The issue of numerical uncertainty. *Second International Conference on CFD in the Minerals and Process Industries*, CSIRO, Melbourne, Australia, 6–8 December 1999.
19. Stern F. Comprehensive approach to verification and validation of CFD simulations—Part 1: methodology and procedures. *Journal of Fluids Engineering* 2001; **123**:793–802.
20. Roy C. Review of code and solution verification procedures for computational simulation. *Journal of Computational Physics* 2005; **205**:131–156.
21. Mehta UB. Some aspects of uncertainty in computational fluid dynamics results. *Transactions of the ASME* 1991; **113**:538–543.
22. Pope S. *Turbulent Flows*. Cambridge University Press: New York, NY, 2000.
23. Pelletier D, Roache P. Verification and validation of computational heat transfer. In *Handbook of Numerical Heat Transfer*, Minkowycz W, Sparrow E, Murthy J (eds). Wiley: New York, 2006; 417–442.
24. Godfrey A, Eppard W, Cliff E. Using sensitivity equations for chemically reacting flows. *Seventh AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, St. Louis, MO, 2–4 September 1998; AIAA 98-4805.
25. Duvigneau R, Pelletier D. Evaluation of nearby flows by a shape sensitivity equation method. *Forty-third AIAA Aerospace Sciences Meeting*, Reno, NV, 10–13 January 2005; AIAA 2005-127.
26. Martins J, Kroo I, Alonso J. An automated method for sensitivity analysis using complex variables. *Proceedings of the 38th Aerospace Sciences Meeting*, Reno, NV, 2000; AIAA 2000-0689.
27. Bischof C, Roh L, Mauer-Oats A. ADIC: an extensible automatic differentiation tool for ANSI-C. *Software Practice and Experience* 1997; **27**(12):1427–1456.
28. Bischof C, Khademi P, Mauer A, Carle A. Adifor 2.0: automatic differentiation of Fortran 77 programs. *IEEE Computational Science and Engineering* 1996; **3**(3):18–32.
29. Bischof C, Bucker H, Rasch A. Sensitivity analysis of turbulence models using automatic differentiation. *SIAM Journal on Scientific Computing* 2004; **26**:510–522.
30. Borggaard J, Verma A. On efficient solutions to the continuous sensitivity equation using automatic differentiation. *SIAM Journal on Scientific Computing* 2000; **22**(1):39–62.

31. Bischof C, Land B, Vehreschild A. Automatic differentiation for MATLAB programs. *Proceedings in Applied Mathematics and Mechanics* 2003; **2**:50–53.
32. Bischof C, Bücker H, Vehreschild A. A macro language for derivative definition in ADiMat. In *Automatic Differentiation*: *Applications*, *Theory*, *and Implementations*, Bücker H, Corliss G, Hovland P, Naumann U, Norris B (eds). Springer: Berlin, 2005; 181–188.
33. Gunzburger M. Sensitivities, adjoints, and flow optimization. *International Journal for Numerical Methods in Fluids* 1999; **31**:53–78.
34. Dowding KJ, Blackwell BF. Joint experimental/computational techniques to measure thermal properties of solids. *Measurement Science and Technology* 1998; **9**:877–887.
35. Harlow F, Welch J. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of Fluids* 1965; **8**(12):2182–2189.
36. Chorin A. Numerical solution of the Navier–Stokes equations. *Mathematics of Computation* 1968; **22**:745–762.
37. Kim J, Moin P. Application of a fractional-step method to incompressible Navier–Stokes equations. *Journal of Computational Physics* 1985; **59**:308–323.
38. Ferziger J, Perić M. *Computational Methods for Fluid Dynamics*. Springer: New York, NY, 1996.
39. Wang C. Exact solutions to the steady-state Navier–Stokes equations. *Annual Review of Fluid Mechanics* 1991; **23**:159–177.
40. Drazin P, Riley N. *The Navier–Stokes Equations*. Cambridge University Press: New York, NY, 2006.
41. Roache P. Code verification by the method of manufactured solutions. *Journal of Fluids Engineering* 2002; **124**:4–10.
42. Poiseuille JL. Recherches expérimentelles sur le mouvement des liquides dans les tubes de trés petits diamètres. *Comptes Rendus de l'Académie des Sciences. Série I, Mathématique* 1840; **11**:961–967, 1041–1048.
43. Sparrow E, Lin S, Lundgren T. Flow development in the hydrodynamic entrance region of tubes and ducts. *Physics of Fluids* 1964; **7**:338–347.
44. Su M, Chen Q, Chiang C-M. Comparison of different subgrid-scale models of large eddy simulation for indoor airflow modeling. *Journal of Fluids Engineering* 2001; **123**(3):628–639. DOI: 10.1115/1.1378294.
45. Dong S, Meng H, Fox RO. Application of fractional-step scheme and finite volume method for simulating the flow past a surface-mounted mixing tab. *Numerical Heat Transfer*, *Part A* 2002; **41**:469–490.
46. Martins J, Sturdza P, Alonso J. The complex-step derivative approximation. *ACM Transactions on Mathematical Software* 2003; **29**:245–262.
47. Roache P. Perspective: a method for uniform reporting of grid refinement studies. *Journal of Fluids Engineering* 1994; **116**:405–413.