Master of Science Thesis

Variational Germano Optimization of Arbitrary Unresolved-Scale Models

Gabriel Maher

May 14, 2014

Aerodynamics

Faculty of Aerospace Engineering

TU Delft

Delft University of Technology

Variational Germano Optimization of Arbitrary Unresolved-Scale Models

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering
at Delft University of Technology

Gabriel Maher

May 14, 2014

Faculty of Aerospace Engineering · Delft University of Technology

Delft University of Technology

Copyright © Aerospace Engineering, Delft University of Technology
All rights reserved.

DELFT UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF AERODYNAMICS

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance the thesis entitled **"Variational Germano Optimization of Arbitrary Unresolved-Scale Models"** by **Gabriel Maher** in fulfillment of the requirements for the degree of **Master of Science**.

Dated: May 14, 2014

Supervisors:

Dr. S.J. Hulshoff

Dr. Ir. M.I. Gerritsma

Dr. R.P. Dwight

Dr. Ing. M. Ruess

Contents

Contents i

1 Introduction 1

2 Variational Multiscale Method 3

3 Variational Germano Method for Arbitrary SGS Models 5
3.1 Standard VGM solved with Newton's method 5
3.2 Least-squares Germano identity solved using the Broyden-Fletcher-Goldfarb-Shanno algorithm 6
3.2.1 Inexact line-search algorithm 7

4 Analysis of the VGM 9
4.1 Definitions of error 9
4.2 Difference between VGM and L_2 optimality 10
4.3 Germano residuals as a measure of SGS model quality 11

5 Advection-Diffusion Equation 13
5.1 Advection-Diffusion Germano residual investigation 14
5.2 Comparison of L_2 and Nodal projector 16
5.3 Performance of the VGM for different τ and mesh size with the L_2 projector 17

6 Burgers' Equation 21
6.1 Burgers' equation Germano residual investigation 22
6.2 Numerical results 23

7 Stokes Equations 29
7.1 Stokes equations Germano residual investigation 30
7.1.1 τ_m only models 30
7.1.2 Models with both τ_m and τ_c 31
7.2 Numerical results 33

8 Conclusion 37
8.1 Recommendations 38

Bibliography 39

Chapter 1

Introduction

The numerical solution of multiscale Partial Differential Equations (PDEs), i.e. PDEs where the solution contains a large range of scales that are dynamically important, can be a challenging task. This is due to the fact that the effort required to compute all the necessary scales may be intractable. In order to deal with multiscale PDEs, Hughes et al. introduced a Variational Multiscale Method (VMM) in [1, 2]. In the VMM, the solution of the PDE $\boldsymbol{u}$, is sum-decomposed as $\boldsymbol{u} = \boldsymbol{u}^h + \boldsymbol{u}'$, where $\boldsymbol{u}^h$ denote the numerically resolved scales and $\boldsymbol{u}'$ the unresolved scales. Since the effect of $\boldsymbol{u}'$ is non-negligible, by not computing $\boldsymbol{u}'$, $\boldsymbol{u}^h$ is not accurately computed. Therefore the influence of $\boldsymbol{u}'$ on $\boldsymbol{u}^h$ must be represented.

One approach for accounting for the influence of $\boldsymbol{u}'$ is to model $\boldsymbol{u}'$ using a subgrid-scale (SGS) model. In most cases only approximate models are available for $\boldsymbol{u}'$. These in turn, can depend on a number of parameters. Often optimal values for the model parameters cannot be exactly determined. As a solution to this problem, Oberai et al. showed that the variational Germano method (VGM) [3, 4] could be used to dynamically calibrate parameters in numerical methods. However the VGM has only seen limited application in combination with the VMM, see e.g. Oberai and Sondak [5] and Akkerman, van der Zee and Hulshoff [6], and in most cases only SGS models are used for which the relations appearing in the VGM can be solved analytically. Additionally, in the absence of objective reference data, it can be difficult to assess how effective the VGM is for the estimation of parameters on coarse meshes, as competing sources of error are present. Furthermore, it is not always clear which models are the most suitable for use with the VGM as demonstrated by Chen, Maher and Hulshoff in [7].

There is evidence that improved SGS models for use with the VMM can be obtained by including nonlinear dependencies and spatial variations, as shown by Gravemeier [8] and Calo [9]. In doing so, it may no longer be possible to solve the VGM relations analytically for the SGS model parameters. This motivates the need for a framework in which the VGM can be applied to more general SGS models.

In this thesis two methods are proposed that can numerically solve the VGM relations for arbitrary SGS Models. These are tested on a sequence of model equations for the Navier-Stokes equations, namely the advection-diffusion equation, Burgers' equation and the Stokes equations, all discretized using the VMM. Furthermore, a number of consequences for the selection of SGS models are derived from an analysis of the VGM.

In Chapter 2 an outline of the VMM is presented. The VGM and the numerical methods used for its solution are detailed in Chapter 3. A breakdown of the different

sources of error and how the VGM relates to them is presented in Chapter 4. Thereafter the proposed method of treating the VGM is applied to the advection-diffusion equation, Burgers' equation and the Stokes equations in Chapters 5, 6 and 7 respectively. Finally conclusions and recommendations for further work are discussed in Chapter 8.

Chapter 2

Variational Multiscale Method

In this thesis the VMM is used to model the scales which can not be resolved by the numerical discretization. To introduce the notation and concepts of the VMM the following abstract problem is considered:

$$\begin{aligned} &\text{find } \boldsymbol{u} \in \mathcal{V} : \\ &\mathcal{L}\boldsymbol{u} = \boldsymbol{f}, \text{ on } \Omega \end{aligned} \tag{2.1}$$

with appropriate boundary and initial conditions. $\mathcal{L}$ represents a potentially non-linear differential operator, $\mathcal{V}$ is the function space that contains the solution, Ω is the domain and $\boldsymbol{f}$ is a specified forcing function. The Galerkin variational form of (2.1) is shown in (2.2).

$$\begin{aligned} &\text{find } \boldsymbol{u} \in \mathcal{V} : \\ &B(\boldsymbol{w}, \boldsymbol{u}) = (\boldsymbol{w}, \boldsymbol{f}), \ \forall \boldsymbol{w} \in \mathcal{V} \end{aligned} \tag{2.2}$$

In this case $(\cdot, \cdot)$ is the standard L_2 inner-product on Ω and $B(\boldsymbol{w}, \boldsymbol{u}) := (\boldsymbol{w}, \mathcal{L}\boldsymbol{u})$.

In the VMM as proposed by Hughes et al. [1, 2], a numerical solution to (2.2) is sought in the finite dimensional function space $\mathcal{V}^h \subset \mathcal{V}$, where h represents a characteristic length of the discretization. $\mathcal{V}^h$ defines the projector $\mathbb{P}^h : \mathcal{V} \to \mathcal{V}^h$. The resolved scales are the part of $\boldsymbol{u}$ that can be represented in $\mathcal{V}^h$ and are given by $\boldsymbol{u}^h = \mathbb{P}^h\boldsymbol{u}$. The unresolved scales denote the part of $\boldsymbol{u}$ that is not contained in $\mathcal{V}^h$ and are given by:

$$\boldsymbol{u}' := \boldsymbol{u} - \mathbb{P}^h\boldsymbol{u} = (\mathbb{I} - \mathbb{P}^h)\boldsymbol{u} \in \mathcal{V}' := \mathcal{V} \backslash \mathcal{V}^h \tag{2.3}$$

A similar decomposition into resolved and unresolved scales can be made for $\boldsymbol{w}$.

For linear PDEs it can be shown that $\boldsymbol{u}'$ is governed by an unresolved scale Green's function, $\boldsymbol{\mathcal{G}}'$, see e.g. Hughes [1]:

$$\begin{aligned} \boldsymbol{u}'(\boldsymbol{x}) &= -\int_\Omega \boldsymbol{\mathcal{G}}(\boldsymbol{x}, \boldsymbol{y})'\boldsymbol{R}(\boldsymbol{y}) \, \mathrm{d}\Omega \\ \boldsymbol{R} &:= \mathcal{L}\boldsymbol{u}^h - \boldsymbol{f} \end{aligned} \tag{2.4}$$

where $\boldsymbol{R}$ is the resolved scale residual. For nonlinear PDEs this represents a first order perturbation approximation for $\boldsymbol{u}'$ as shown by Scovazzi [10]. In both cases the following expression is normally used:

$$\boldsymbol{u}' \approx -\boldsymbol{\tau}\boldsymbol{R} \tag{2.5}$$

where $\boldsymbol{\tau}$ is a global constant tensor. This implies that $\boldsymbol{\mathcal{G}'}$ is approximated as a scaled Dirac delta distribution:

$$\boldsymbol{\mathcal{G}'}(\boldsymbol{x}, \boldsymbol{y}) \approx \boldsymbol{\tau}\delta(\boldsymbol{x} - \boldsymbol{y}) \tag{2.6}$$

Hence $\boldsymbol{\tau}$ can be interpreted as an approximate unresolved scale Green's function, which suggests that $\boldsymbol{\tau}$ could be approximated by the following expression:

$$\boldsymbol{\tau} \approx \int_{\Omega} \boldsymbol{\mathcal{G}'} \, \mathrm{d}\Omega \tag{2.7}$$

However, for problems of interest $\boldsymbol{\mathcal{G}'}$ is generally not known. In those cases $\boldsymbol{\tau}$ can be approximated by a global constant tensor that depends on a vector of parameters $\vec{c}$ such that $\boldsymbol{\tau} \approx \boldsymbol{\tau}(\vec{c})$. The functional form of $\boldsymbol{\tau}(\vec{c})$ can then be obtained by parameterizing it as a function of the discretization parameters as shown by Akkerman [11]. Another method is to use a $\boldsymbol{\tau}$ from cases where it is possible to determine $\boldsymbol{\tau}$ exactly such as for the advection-diffusion equation see e.g. Hughes and Sangalli [12], or Shakib [13]. In both approaches optimal values for $\vec{c}$ may not be known. A solution to this problem is presented in the next chapter.

Substituting $\boldsymbol{u} = \boldsymbol{u}^h + \boldsymbol{u}' = \boldsymbol{u}^h - \boldsymbol{\tau}\boldsymbol{R}$ in (2.2) allows $\boldsymbol{u}^h$ to be determined by solving the variational problem in (2.8).

$$\begin{aligned} &\text{find } \boldsymbol{u}^h \in \mathcal{V}^h : \\ &B(\boldsymbol{w}^h, \boldsymbol{u}^h) + M(\boldsymbol{w}^h, \boldsymbol{u}^h; \vec{c}, \boldsymbol{f}, h) = (\boldsymbol{w}^h, \boldsymbol{f}), \ \forall \boldsymbol{w}^h \in \mathcal{V}^h \end{aligned} \tag{2.8}$$

$M(\boldsymbol{w}^h, \boldsymbol{u}^h; \vec{c}, \boldsymbol{f}, h)$ represents the additional terms that occur due to the substitution and account for the influence of $\boldsymbol{u}'$ on $\boldsymbol{u}^h$.

Chapter 3

Variational Germano Method for Arbitrary SGS Models

Application of the VMM leads to a subgrid-scale model for $\boldsymbol{u}'$ that depends on a vector of parameters $\vec{c}$. The definition of $\boldsymbol{u}'$ depends on the definition of $\mathcal{V}^h$. Therefore $\vec{c}$ will have values that depend on the numerical discretization used. In many cases optimal values for $\vec{c}$ are not known. One way of determining $\vec{c}$ is using a Variational Germano Method (VGM), which is based on the idea that a suitable SGS model will provide consistent representations of the solution for all reasonable levels of approximation. Considering the solution simultaneously for different levels of approximation leads to the VGM relations. If the SGS model is chosen such that the VGM relations can be solved for $\vec{c}$ analytically, the procedure as proposed by Oberai et al. [3, 4, 5], can be used. It is desirable to allow for the use of more general SGS models however. Therefore, in this chapter, numerical solution procedures are proposed for the VGM, allowing the determination of $\vec{c}$ for arbitrary functional forms of $\boldsymbol{\tau}$. The VGM does not require the exact solution of the PDE problem to be known, making it a suitable procedure for general nonlinear problems.

In the VGM a series of coarse nested function spaces, $\mathcal{V}^{h_N} \subset \mathcal{V}^{h_{N-1}} \subset ... \subset \mathcal{V}^{h_1} \subset \mathcal{V}^h$ are defined. On each $\mathcal{V}^{h_i}$ a coarse space solution $\boldsymbol{u}^{h_i}$ can be defined. The optimal $\boldsymbol{u}^{h_i}$ will be given by $\mathbb{P}^{h_i}\boldsymbol{u}$, where $\mathbb{P}^{h_i} : \mathcal{V} \to \mathcal{V}^{h_i}$ is a projector onto the coarse function space $\mathcal{V}^{h_i}$. As $\boldsymbol{u}$ is generally not known, $\boldsymbol{u}^{h_i}$ can instead be obtained via $\boldsymbol{u}^{h_i} = \mathbb{P}^{h_i}\boldsymbol{u}^h$. If $\boldsymbol{u}^h = \mathbb{P}^h\boldsymbol{u}$ then $\boldsymbol{u}^{h_i} = \mathbb{P}^{h_i}\mathbb{P}^h\boldsymbol{u}$. Therefore the $\boldsymbol{u}^{h_i}$ will still be optimal provided $\mathbb{P}^{h_i}\mathbb{P}^h = \mathbb{P}^{h_i}$ as indicated by Oberai in [4]. There are a number of projectors with this property. In this thesis the L_2 projector is primarily used.

3.1 Standard VGM solved with Newton's method

Optimal values for $\vec{c}$ can then be found by solving the variational inverse problem:

$$\text{find } \vec{c}:$$
$$B(\boldsymbol{w}^{h_i}, \boldsymbol{u}^{h_i}) + M(\boldsymbol{w}^{h_i}, \boldsymbol{u}^{h_i}; \vec{c}, \boldsymbol{f}, h_i) - (\boldsymbol{w}^{h_i}, \boldsymbol{f}) = 0,\ \forall \boldsymbol{w}^{h_i} \in \mathcal{V}^{h_i},\ i = 1, ..., N \tag{3.1}$$

(3.1) is known as the Germano identity and generally has more equations than there are coefficients $\vec{c}$. The amount of equations can be reduced by interpreting (3.1) in a global sense, i.e. taking all inner products over the entire domain Ω, see e.g. Oberai et al. [3, 4]. In this way (3.1) reduces to one system of equations per $\mathcal{V}^{h_i}$. Furthermore the equations in

each system can be summed to produce one equation. Then as many $\mathcal{V}^{h_i}$ must be defined as there are components of $\vec{c}$.

Once $\vec{c}$ has been obtained from the VGM it can be used to compute a new $\boldsymbol{u}^h$, which can then be used to repeat the VGM. This defines an iterative procedure for optimizing $\vec{c}$ as shown by Oberai and Wanderer [3]. In this thesis, for steady equations, the procedure was repeated until $\vec{c}$ no longer changed per iteration, or 50 iterations were exceeded.

When $\boldsymbol{\tau}$, and hence $M(\boldsymbol{w}^{h_i}, \boldsymbol{u}^{h_i}; \vec{c}, \boldsymbol{f}, h_i)$, depends nonlinearly on $\vec{c}$, it may not be possible to derive an analytical expression for $\vec{c}$ from (3.1). A numerical procedure can then be used to solve the system of equations. In this thesis the Newton-Raphson algorithm is proposed for solving the global VGM.

The global Germano identity on $\mathcal{V}^{h_i}$ is now denoted by $G^{h_i} := B(\boldsymbol{w}^{h_i}, \boldsymbol{u}^{h_i}) + M(\boldsymbol{w}^{h_i}, \boldsymbol{u}^{h_i}; \vec{c}, \boldsymbol{f}, h_i) - (\boldsymbol{w}^{h_i}, \boldsymbol{f})$. The system of Germano identities in (3.1) can be represented as the vector $\boldsymbol{G} := [G^{h_1}, ..., G^{h_N}]^T$. (3.1) is then equivalent to solving $\boldsymbol{G} = 0$. Taking a first order Taylor expansion of $\boldsymbol{G} = 0$, with respect to $\vec{c}$, about an initially chosen $\vec{c} = \vec{c}_0$, produces:

$$\boldsymbol{G}(\vec{c}) \approx \boldsymbol{G}(\vec{c}_0) + \boldsymbol{J}(\vec{c}_0)(\vec{c} - \vec{c}_0) = 0, \ \boldsymbol{J} = \begin{pmatrix} \frac{\partial G^{h_1}}{\partial c_1} & \cdots & \frac{\partial G^{h_1}}{\partial c_N} \\ \vdots & \ddots & \vdots \\ \frac{\partial G^{h_N}}{\partial c_1} & \cdots & \frac{\partial G^{h_N}}{\partial c_N} \end{pmatrix} \tag{3.2}$$

where $\boldsymbol{J}$ is the Jacobian of $\boldsymbol{G}$ with respect to $\vec{c}$. In this thesis exact algebraic expressions for the entries of $\boldsymbol{J}$ are used. Replacing $\vec{c}$ by $\vec{c}_{n+1}$ and $\vec{c}_0$ by $\vec{c}_n$, where n indicates the nth iteration, allows an iterative procedure for solving $\boldsymbol{G} = 0$ to be derived:

$$\vec{c}_{n+1} = \vec{c}_n - \boldsymbol{J}^{-1}(\vec{c}_n)\boldsymbol{G}(\vec{c}_n) \tag{3.3}$$

Here (3.3) is stopped when $||\boldsymbol{G}||_{L_2} < 1 \cdot 10^{-10}$.

3.2 Least-squares Germano identity solved using the Broyden-Fletcher-Goldfarb-Shanno algorithm

In the case where $\boldsymbol{w}^{h_i} = \sum_j \boldsymbol{\phi}_j^{h_i}$, where $\boldsymbol{\phi}_j^{h_i} \in \mathcal{V}^{h_i}$ are local basis functions, (3.1) can be split into local components as in (3.4).

$$\boldsymbol{r}_j^i := B(\boldsymbol{\phi}_j^{h_i}, \boldsymbol{u}^{h_i}) + M(\boldsymbol{\phi}_j^{h_i}, \boldsymbol{u}^{h_i}; \vec{c}, \boldsymbol{f}, h_i) - (\boldsymbol{\phi}_j^{h_i}, \boldsymbol{f}) \tag{3.4}$$

For each $\mathcal{V}^{h_i}$ in (3.1) the residual of the Germano identity can be assembled as $\sum_j \boldsymbol{r}_j^i$. However, it is possible for the $\boldsymbol{r}_j^i$ to be individually large while their sum may be small due to differing signs. This would make it seem that the chosen $\vec{c}$ satisfies (3.1) when it does not. The solution of (3.1) can then result in no further change for $\vec{c}$. Furthermore, numerical floating point errors may occur if the $\boldsymbol{r}_i^j$ have differing signs and very different magnitudes.

These issues can be resolved by using the least-squares form of the Germano identity as proposed by Oberai and Wang in [4]. Instead of assembling individual equations per $\mathcal{V}^{h_i}$, a least-squares residual, R_G, is constructed from all the $\boldsymbol{r}_j^i$:

$$R_G := \sum_{i=1}^{N} \sum_j (\boldsymbol{r}_j^i \cdot \boldsymbol{r}_j^i) \tag{3.5}$$

$\vec{c}$ is found by minimizing R_G.

In this thesis a further modification of the least-squares Germano identity is proposed. When the problem being solved consists of a system of PDEs, $\boldsymbol{r}_j^i$ will be a vector with one component per equation: $\boldsymbol{r}_j^i := [r_{1,j}^i, r_{2,j}^i, ..., r_{N_{eq},j}^i]$ where N_{eq} is the number of equations. In this case a separate least-squares residual can be assembled per equation:

$$R_{k,G} := \sum_{i=1}^{N} \sum_{j} (r_{k,j}^i)^2 \tag{3.6}$$

where k is the equation number for which the residual is being assembled. $\vec{c}$ is then found by minimizing all, or some of the $R_{k,G}$.

The least-squares formulation of the Germano identity requires the minimization of R_G, or $R_{k,G}$ to solve for $\vec{c}$. If a general least-squares residual R is considered, then R is minimized when $\nabla R = 0$. An efficient procedure for solving $\nabla R = 0$ is the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm. In this procedure, ∇R is approximated with a first order Taylor expansion about $\vec{c}_0$:

$$\nabla R(\vec{c}) \approx \nabla R(\vec{c}_0) + B(\vec{c}_0)(\vec{c} - \vec{c}_0) = 0 \tag{3.7}$$

where B is the Hessian of R with respect to $\vec{c}$. Again replacing $\vec{c}$ by $\vec{c}_{n+1}$ and $\vec{c}_0$ by $\vec{c}_n$, the following iterative procedure is derived:

$$\vec{c}_{n+1} = \vec{c}_n - \alpha_n B_n^{-1} \nabla R(\vec{c}_n) \tag{3.8}$$

α_n is a parameter that controls the step length and is obtained here from an inexact line search. $-B_n^{-1}\nabla R(\vec{c}_n) = \vec{p}_n$ is the search direction. B_n is an approximation to the Hessian at step n. B_{n+1} is obtained by first setting $B_0 = I$, the identity matrix, and then updated to B_{n+1} according to:

$$\begin{aligned} B_{n+1} &= B_n + \frac{y_n y_n^T}{y_n^T s_n} - \frac{B_n s_n s_n^T B_n}{s_n^T B_n s_n} \\ s_n &= \alpha_n(\vec{c}_{n+1} - \vec{c}_n) \\ y_n &= \nabla R(\vec{c}_{n+1}) - \nabla R(\vec{c}_n) \end{aligned} \tag{3.9}$$

To complete the algorithm it is noted that ∇R is evaluated using forward finite differences:

$$\frac{\partial R}{\partial c_i} \approx \frac{R(c_i + \epsilon) - R(c_i)}{\epsilon} \tag{3.10}$$

with $\epsilon = 1 \cdot 10^{-5}$. The algorithm is stopped when $||\nabla R||_{L_2} < 1 \cdot 10^{-7}$ or when $||\vec{p}_n||_{L_2} < 1 \cdot 10^{-4}$.

3.2.1 Inexact line-search algorithm

The inexact line-search algorithm is an algorithm commonly used with the BFGS algorithm for determining the step length parameter α, see e.g. Yuan [14]. The line-search is performed at every step of the BFGS algorithm. Given the search direction, $\vec{p}_n$, first an initial guess of $\alpha = 1$ is chosen. Then α is used to check if the strong Wolfe conditions as proposed by Wolfe [15], are satisfied:

$$R(\vec{c}_n + \alpha\vec{p}_n) \leq R(\vec{c}_n) + C_1 \alpha \vec{p}_n^T \nabla R(\vec{c}_n) \tag{3.11}$$

$$|\vec{p}_n^T \nabla R(\vec{c}_n + \alpha\vec{p}_n)| \leq C_2 |\vec{p}_n^T \nabla R(\vec{c}_n)| \tag{3.12}$$

with $C_1 = 10^{-4}$ and $C_2 = 0.9$. If the Wolfe conditions are satisfied, α is accepted and used to compute the next step of the BFGS algorithm. If either, or both of the Wolfe conditions are not satisfied α is decreased by a geometric scaling parameter:

$$\alpha_{l+1} = \beta\alpha_l \tag{3.13}$$

where l is the current step of the line-search algorithm and β is the geometric scaling factor. With the new α, the Wolfe conditions can be checked again and the procedure repeated if they are still not satisfied. $0 < \beta < 1$ must be selected to ensure the step length is decreased at each iteration. In this thesis $\beta = 0.25$ is used.

The line-search can be allowed to run until an acceptable α is found or a specified maximum number of iterations has been exceeded. In this thesis $10 - 20$ iterations is used as the maximum allowed. If after the maximum amount of iterations no acceptable α has been found, the BFGS algorithm can be stopped, or $\alpha \approx C_1$ can be selected and a BFGS iteration can be performed. In the latter case performing the BFGS iteration may lead to an improved estimate for the Hessian B_n which may result in an improved search direction $\vec{p}_n$ allowing the algorithm to proceed further. The latter procedure is used here.

Chapter 4

Analysis of the VGM

The VGM defined in the previous chapter depends on assumptions which introduce errors in the definition of an optimal $\vec{c}$. These errors are considered here in more detail. For this purpose, given a particular PDE, define:

- $\boldsymbol{u}$ as the exact solution
- $\mathbb{P}^h\boldsymbol{u}$ as the optimal solution for a particular finite element method (FEM) space $\mathcal{V}^h$, for a particular projector $\mathbb{P}^h$ defined by a norm $||\cdot||$.
- $\boldsymbol{u}^{h*}$ as the solution in $\mathcal{V}^h$ obtainable by varying the value of $\boldsymbol{\tau}$ such that $||\mathbb{P}^h\boldsymbol{u}-\boldsymbol{u}^{h*}||$ is minimized.
- $\boldsymbol{u}^h$ the solution in $\mathcal{V}^h$ obtained with a $\boldsymbol{\tau}$ with parameters $\vec{c}$ optimized with the VGM.

4.1 Definitions of error

In [5], Oberai and Sondak claim that the different errors of the VGM can be broken down and defined as follows:

- $\epsilon_{FEM} := ||\boldsymbol{u} - \mathbb{P}^h\boldsymbol{u}||$, error due to FEM basis
- $\epsilon_{SGS} := ||\mathbb{P}^h\boldsymbol{u}-\boldsymbol{u}^{h*}||$, error due to approximation of unresolved scales by $\boldsymbol{u'} \approx -\boldsymbol{\tau R}$ with a globally constant $\boldsymbol{\tau}$.
- $\epsilon_\tau := ||\boldsymbol{u}^{h*} - \boldsymbol{u}^h||$, error due to obtaining a different $\vec{c}$ when applying the VGM than that that minimizes $||\mathbb{P}^h\boldsymbol{u} - \boldsymbol{u}^{h*}||$.

The total error between $\boldsymbol{u}^h$ and $\boldsymbol{u}$ is then expressed as:

$$\epsilon := ||\boldsymbol{u} - \boldsymbol{u}^h|| = \epsilon_{FEM} + \epsilon_{SGS} + \epsilon_\tau \tag{4.1}$$

ϵ_{FEM} is an indication of how well the selected FEM basis can approximate the exact solution. It can be observed from ϵ_{SGS} how well $\boldsymbol{u'}$ is approximated. The interpretation of ϵ_τ is less clear. Oberai and Sondak [5] state that ϵ_τ is a measure of how effective the VGM is in determining optimal values for $\vec{c}$. However, in the following sections, evidence is provided that shows that ϵ_τ is dependent on the selected functional form of $\boldsymbol{\tau}$. Depending on the functional form of $\boldsymbol{\tau}$, ϵ_τ can be large or small and is therefore more an indicator of the suitability of a given SGS model for optimization with the VGM.

4.2 Difference between VGM and L_2 optimality

When using the VGM, with the L_2 projector, the goal is to obtain $\boldsymbol{u}^h \approx \mathbb{P}^h\boldsymbol{u}$, such that the projected error, $||\mathbb{P}^h\boldsymbol{u}-\boldsymbol{u}^h||_{L_2}$, is minimized, where $\mathbb{P}^h$ is the L_2 projector. Note that, in the case of the L_2 projector, by definition, when $\boldsymbol{u}^h \approx \mathbb{P}^h\boldsymbol{u}$, $||\boldsymbol{u}-\boldsymbol{u}^h||_{L_2}$, the standard L_2 error, is also approximately minimized. The solution of the VGM will determine how close $\boldsymbol{u}^h$ comes to $\mathbb{P}^h\boldsymbol{u}$. Hence it is important to consider how well the minimums of the VGM residuals correspond to the minimum of the projected error. An analysis of this correspondence is performed in this section.

For the purpose of discussion a case is considered here with two VGM residuals: $R_{1,G}$ and $R_{2,G}$. The numerical solution $\boldsymbol{u}^h$ implicitly depends on $\vec{c}$. $\vec{c}$ modifies the PDE that governs $\boldsymbol{u}^h$, this dependence can be represented by allowing $\boldsymbol{u}^h$ to be a function of $\vec{c}$. The residuals $R_{1,G}$ and $R_{2,G}$ both explicitly depend on $\vec{c}$ and implicitly via $\boldsymbol{u}^h(\vec{c})$. Taking into account the necessary dependencies, the projected error could be written in the following functional form:

$$\text{projected error} := F_{L_2}(R_{1,G}(\boldsymbol{u}^h(\vec{c}),\vec{c}), R_{2,G}(\boldsymbol{u}^h(\vec{c}),\vec{c})) \tag{4.2}$$

The $\vec{c}$ that minimises the projected error error must satisfy:

$$\frac{\mathrm{d}F_{L_2}}{\mathrm{d}\vec{c}} = \frac{\partial F_{L_2}}{\partial R_{1,G}}\left(\frac{\partial R_{1,G}}{\partial \boldsymbol{u}^h}\frac{\partial \boldsymbol{u}^h}{\partial \vec{c}} + \frac{\partial R_{1,G}}{\partial \vec{c}}\right) + \frac{\partial F_{L_2}}{\partial R_{2,G}}\left(\frac{\partial R_{2,G}}{\partial \boldsymbol{u}^h}\frac{\partial \boldsymbol{u}^h}{\partial \vec{c}} + \frac{\partial R_{2,G}}{\partial \vec{c}}\right) = 0 \tag{4.3}$$

which requires:

- $\frac{\partial F_{L_2}}{\partial R_{1,G}}\left(\frac{\partial R_{1,G}}{\partial \boldsymbol{u}^h}\frac{\partial \boldsymbol{u}^h}{\partial \vec{c}} + \frac{\partial R_{1,G}}{\partial \vec{c}}\right) = -\frac{\partial F_{L_2}}{\partial R_{2,G}}\left(\frac{\partial R_{2,G}}{\partial \boldsymbol{u}^h}\frac{\partial \boldsymbol{u}^h}{\partial \vec{c}} + \frac{\partial R_{2,G}}{\partial \vec{c}}\right)$
- or, $\frac{\partial R_{1,G}}{\partial \boldsymbol{u}^h}\frac{\partial \boldsymbol{u}^h}{\partial \vec{c}} = -\frac{\partial R_{1,G}}{\partial \vec{c}}$ and $\frac{\partial R_{2,G}}{\partial \boldsymbol{u}^h}\frac{\partial \boldsymbol{u}^h}{\partial \vec{c}} = -\frac{\partial R_{2,G}}{\partial \vec{c}}$
- or, $\frac{\partial R_{1,G}}{\partial \boldsymbol{u}^h}\frac{\partial \boldsymbol{u}^h}{\partial \vec{c}} = 0$, $\frac{\partial R_{1,G}}{\partial \vec{c}} = 0$, $\frac{\partial R_{2,G}}{\partial \boldsymbol{u}^h}\frac{\partial \boldsymbol{u}^h}{\partial \vec{c}} = 0$ and $\frac{\partial R_{1,G}}{\partial \vec{c}} = 0$

.

The last two points are equivalent to $\frac{\mathrm{d}R_{1,G}}{\mathrm{d}\vec{c}} = 0, \frac{\mathrm{d}R_{2,G}}{\mathrm{d}\vec{c}} = 0$, i.e. $R_{1,G}$ and $R_{2,G}$ must both be globally minimized for the same $\vec{c}$. In the Variational Germano Methods proposed to date, the optimum $\vec{c}$ is defined by $\frac{\partial R_{1,G}}{\partial \vec{c}} \approx 0$ and $\frac{\partial R_{2,G}}{\partial \vec{c}} \approx 0$. However, from (4.3) it is clear that this is not sufficient to find the minimum of the projected error. This requires $\frac{\partial R_{1,G}}{\partial \boldsymbol{u}^h}\frac{\partial \boldsymbol{u}^h}{\partial \vec{c}} \approx 0$ and $\frac{\partial R_{2,G}}{\partial \boldsymbol{u}^h}\frac{\partial \boldsymbol{u}^h}{\partial \vec{c}} \approx 0$ as well. Thus the VGM will only work if there is a single $\vec{c}$ that globally minimizes $R_{1,G}$ and $R_{2,G}$. In Chapter 7 it is shown that for a number of SGS models used in practice this is not the case. Therefore, in most cases the $\vec{c}$ obtained from the VGM will not be the $\vec{c}$ that minimizes the projected error.

In spite of not minimizing the projected error, using the VGM may result in a reduction of the projected error when compared to using standard values for $\vec{c}$. To achieve a further reduction, use would have to be made of adjoint methods to account for the dependence of F_{L_2} on $R_{1,G}$ and $R_{2,G}$. However, solving the adjoint problem may make the method prohibitively expensive for practical purposes.

For the purposes of discussion, in subsequent chapters, SGS models for the VGM are characterized as follows:

1. Concurrent models: have optimal coefficients that minimize all VGM residuals, and $||\mathbb{P}^h\boldsymbol{u}-\boldsymbol{u}^h||$ simultaneously.
2. Scale-invariant models: have optimal coefficients, those that minimize $||\mathbb{P}^h u-u^h||_{L_2}$, that are independent of h.

Presumably a concurrent, scale-invariant SGS model would likely realize the full potential of the VGM. The following section considers potential indicators for the suitability of an SGS model.

4.3 Germano residuals as a measure of SGS model quality

Exact SGS models have coefficients that are independent on the mesh size h and, when inserted into (2.8), $\boldsymbol{u}^h = \mathbb{P}^h\boldsymbol{u}$ is obtained as a solution. Consequently the Germano residuals, $R_{k,G}$, are simultaneously minimized by an exact SGS model. The projected solution is then $\boldsymbol{u}^{h_i} = \mathbb{P}^{h_i}\boldsymbol{u}^h = \mathbb{P}^{h_i}\mathbb{P}^h\boldsymbol{u} = \mathbb{P}^{h_i}\boldsymbol{u}$, which is the optimal solution on $\mathcal{V}^{h_i}$. Examples of exact SGS models are those obtained by Hughes and Sangalli for the advection-diffusion equation in [12]. An exact model can thus be defined as:

- Exact SGS Model: has coefficients that are independent of h, results in $||\mathbb{P}^h\boldsymbol{u}-\boldsymbol{u}^h||_{L_2}$ and $R_{k,G}$ being minimized simultaneously.

When optimizing a given SGS model with the VGM, it is assumed that the optimal coefficient values are those that minimize $R_{k,G}$ for the defined $\mathcal{V}^{h_i}$. By examining the variation of $R_{k,G}$ with $\vec{c}$, for $\mathcal{V}^{h_i}$ for different values of h_i, it is possible to determine whether the optimal coefficients of a given SGS model are scale-invariant. If the optimal coefficient values vary significantly as h_i is changed, the given SGS model can be rejected as a good approximation to the exact SGS model. It will be shown in the following chapters however that using the VGM can still lead to improved performance, even with scale-variant models, compared to using standard coefficient values.

Chapter 5

Advection-Diffusion Equation

To clarify the concepts introduced in Chapters 3 and 4, the application of the VGM to a simple linear equations is first considered. For this purpose the same advection-diffusion problem as in Akkerman [11] and Shakib [13] is used, which is given by:

$$\begin{aligned} &\text{find } u \in \mathcal{V} : \\ &a\frac{\partial u}{\partial x} - \nu\frac{\partial^2 u}{\partial x^2} = f \text{ on } \Omega := x \in]0,1[\\ &u(0) = u(1) = 0 \end{aligned} \tag{5.1}$$

where the space $\mathcal{V}$ is:

$$\mathcal{V} := \{v : v \in \mathcal{H}^1(\Omega); v(0) = v(1) = 0\} \tag{5.2}$$

with $\mathcal{H}^1(\Omega) := \{v : \int_\Omega v^2 + \frac{\partial v}{\partial x}^2 \, \mathrm{d}\Omega < \infty\}$ the Sobolev space of square-integrable functions with square-integrable derivatives on Ω. $a \in \mathbb{R}$ and $\nu \in \mathbb{R}$ are the advection velocity and diffusion coefficient respectively. Here $a = 1$ and $\nu = 1/100$ and $f = 1$ are used. The exact solution to (5.1) is given by:

$$u(x) = x - \frac{1 - e^{(Pe)x}}{1 - e^{Pe}} \tag{5.3}$$

where $Pe = a/\nu$ is the Peclet number. For high values of Pe, $u(x)$ forms a sharp boundary layer near $x = 1$. This can be seen in Figure 5.1 where the exact solution has been plotted for $Pe = 1, 10, 100$.

$\mathcal{C}^n(\Omega)$ is now defined as the space of functions with continuous derivatives up to order n on Ω. Furthermore, denote the partitioning of Ω into elements, K, of size h, by $\mathcal{T}_h$ and $\mathcal{Q}_n(K) := \{\sum_j c_j p_j(x) q_j(y) : p_j, q_j \text{ polynomials of degree } \leq n; x, y \in K\}$ as the space of products of polynomials on an element K, see e.g. Brenner and Scott [16]. Then the finite-dimensional subspace $\mathcal{V}^h \subset \mathcal{V}$ is defined as:

$$\mathcal{V}^h := \{v : \mathcal{C}^0(\Omega) \cap \mathcal{H}^1(\Omega); v(0) = v(1) = 0; \forall K \in \mathcal{T}_h; v_{|K} \in \mathcal{Q}_1(K)\} \tag{5.4}$$

which is the space of standard linear finite element functions on $\mathcal{T}_h$. Application of the

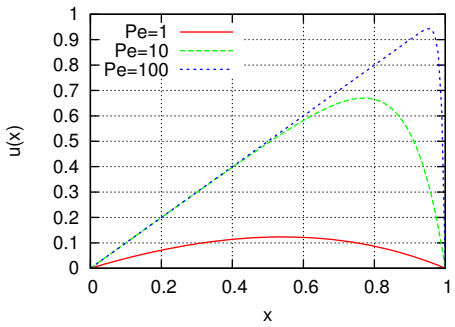


Figure 5.1: Exact solutions for advection-diffusion equation for varying Pe.

VMM to (5.1) then results in the following variational problem:

$$\begin{aligned}
&\text{find } u^h \in \mathcal{V}^h : \\
&\nu\left(\frac{\partial w^h}{\partial x}, \frac{\partial u^h}{\partial x}\right) - a\left(\frac{\partial w^h}{\partial x}, u^h\right) + M(w^h, u^h; \vec{c}, f, h) = (w^h, f) \;\forall\; w^h \in \mathcal{V}^h \\
&M(w^h, u^h; \vec{c}, f, h) = a\left(\frac{\partial w^h}{\partial x}, \tau\left(a\frac{\partial u}{\partial x} - f\right)\right)
\end{aligned} \tag{5.5}$$

Based on the proposed expressions of Akkerman [11] and Shakib [13], VMM SGS models with the following definitions of τ are investigated:

$$\tau_{linear} = c_1 h \tag{5.6}$$

$$\tau_{quadratic} = c_1 h + c_2 h^2 \tag{5.7}$$

$$\tau_{shakibVGM} = \left(c_1\left(\frac{a}{h}\right)^2 + c_2\left(\frac{\nu}{h^2}\right)^2\right)^{-\frac{1}{2}} \tag{5.8}$$

As a benchmark the results obtained using Shakib's modified τ from [13] are used:

$$\tau_{Shakib} = \left(4\left(\frac{a}{h}\right)^2 + 9\left(\frac{4\nu}{h^2}\right)^2\right)^{-\frac{1}{2}} \tag{5.9}$$

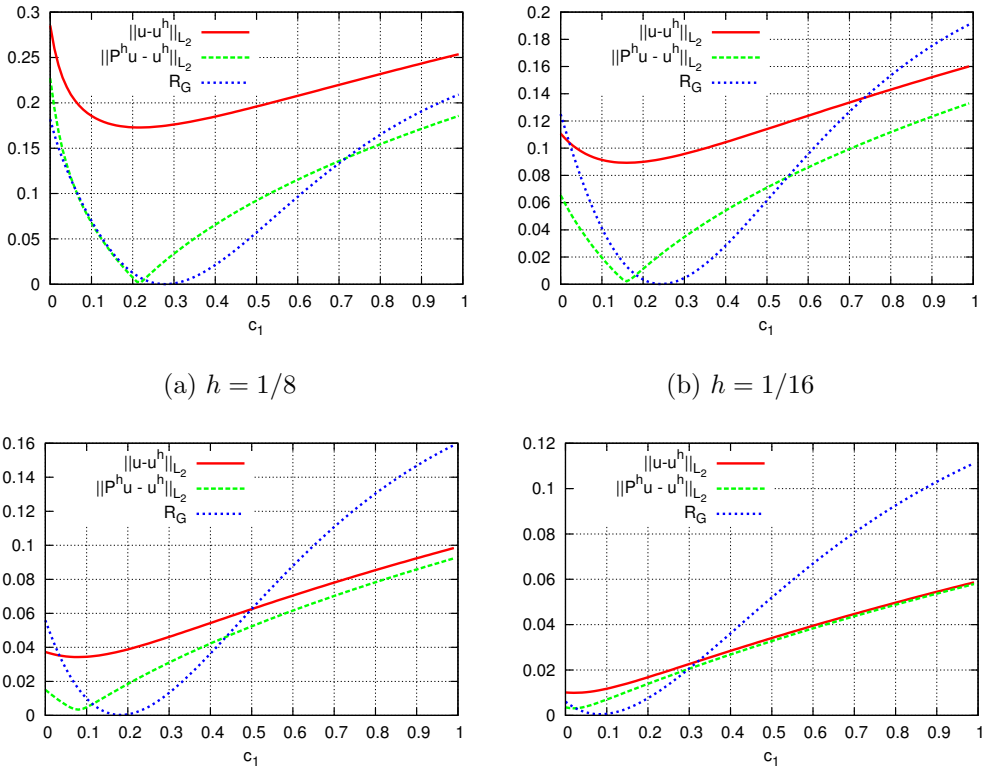
For the VGM, the coarse function spaces, $\mathcal{V}^{h_i}$ are defined with element sizes $h_i = 2^i h$. In the following sections the resulting Germano residuals are analyzed and computed results are presented.

5.1 Advection-Diffusion Germano residual investigation

As was discussed in Chapter 4, the the ability of the VGM to find optimal values for $\vec{c}$ depends on a number of conditions. The first is that $u^h \approx \mathbb{P}^h u$ should be obtainable using the expressions for τ presented above. Next the $\vec{c}$ found by minimizing the VGM residual should be the same as that for which the projected error is minimized. To examine the validity of these conditions, the variation of R_G with respect to c_1 and h are first studied in this section with $\tau = \tau_{linear} = c_1 h$. The errors $||u - u^h||_{L_2}$ and $||\mathbb{P}^h u - u^h||_{L_2}$ are also analyzed. It is emphasized here again that the projected error is of primary interest. The standard L_2 error is also shown however because it is often investigated in literature,

even if it is not the best measure of error from the VMM point of view. For the sake of simplicity, the other expressions for τ are not analyzed as they depend on multiple coefficients which results in multidimensional plots of R_G and the errors, which are more difficult to interpret.

Figure 5.2 shows graphs of the variation of the Germano residual and errors of interest with c_1 for different element sizes h. The first observation that should be made is that, in this case with $\tau = c_1 h$, it is always possible to obtain $u^h \approx \mathbb{P}^h u$ simply by varying c_1. In other words, $u' = -\tau R$ with a globally constant τ is a good approximation for the unresolved scales in this case. This means that $\epsilon_{SGS} = 0$ and the dominant errors are caused by the FEM basis and the parametrization of τ. Furthermore, it is directly evident that $\tau = c_1 h$ is not a scale-invariant model. The c_1 that minimizes R_G shifts from 0.3 to 0.1 with $h = 1/8$ and $h = 1/64$ respectively. It can also be seen that the c_1 that minimizes $||\mathbb{P}^h u - u^h||_{L_2}$ does not coincide with the c_1 that minimizes R_G. The lack of coincidence shows how optimizing scale-variant SGS models with the VGM may not lead to optimal coefficient values. The fact that the c_1 returned by the VGM decreases as the mesh is refined indicates that, in this case, a better τ must have a higher order dependence on h for the range being investigated here.



(a) $h = 1/8$ (b) $h = 1/16$

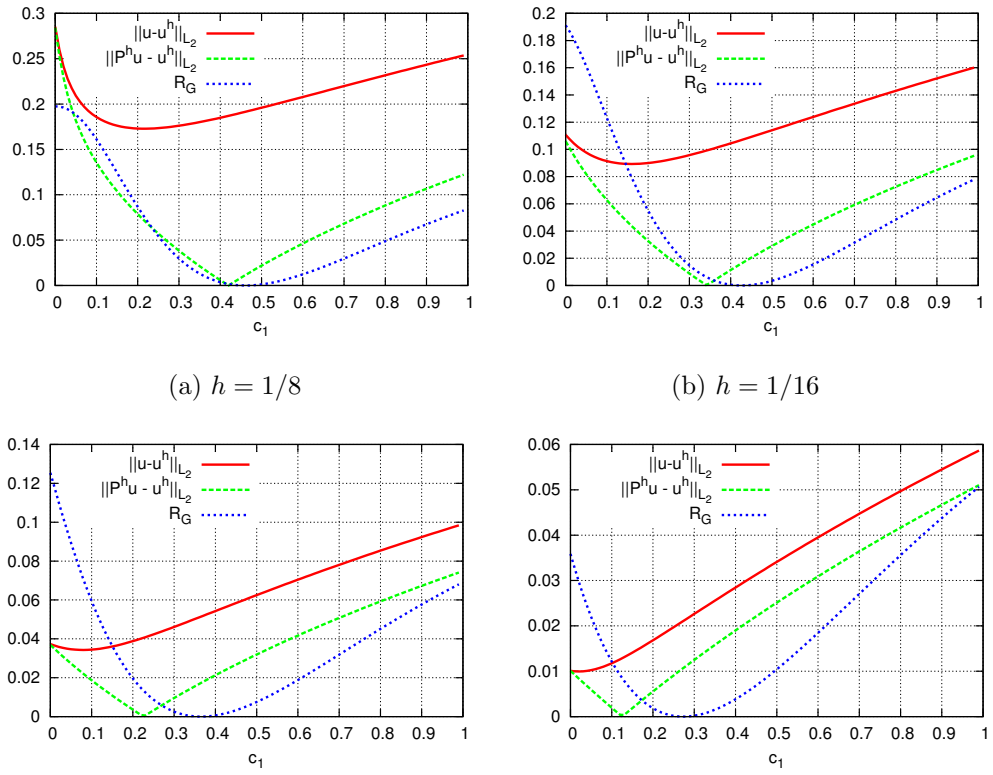
(c) $h = 1/32$ (d) $h = 1/64$

Figure 5.2: Variation of R_G, $||u - u^h||_{L_2}$ and $||\mathbb{P}^h u - u^h||_{L_2}$ with respect to c_1, with $\tau = c_1 h$, with varying element sizes h

5.2 Comparison of L_2 and Nodal projector

The selected projector, $\mathbb{P}^h$, largely determines the $\vec{c}$ that is obtained from the VGM and correspondingly also the u^h obtained. The numerical procedures proposed in this thesis have been designed to be able to handle arbitrary projectors, provided they are compatible with the VGM, that is $\mathbb{P}^{h_i}\mathbb{P}^h = \mathbb{P}^{h_i}$. To illustrate the generality of the procedures and the influence of the selected projector, in this section the VGM residuals are investigated for the nodal projector. Additionally, solutions obtained using the nodal projector are compared to those obtained with the L_2 projector, when using τ_{linear}.

In Figure 5.3 the Germano residuals, projected error and standard L_2 error are gain examined for different values of c_1. Now $\mathbb{P}^h$ is the nodal projector. It is again evident that $u^h \approx \mathbb{P}^h u$ can be obtained using a globally constant τ. The minimum of the standard L_2 error no longer corresponds to the same c_1 with which $u^h \approx \mathbb{P}^h u$ is obtained. This shows the incompatibility of considering the standard L_2 error with the VMM when different projector definitions are used. Similar to before, the c_1 that minimizes R_G does not correspond to the c_1 that minimizes R_G and τ_{linear} is also not scale-invariant under the nodal projector.



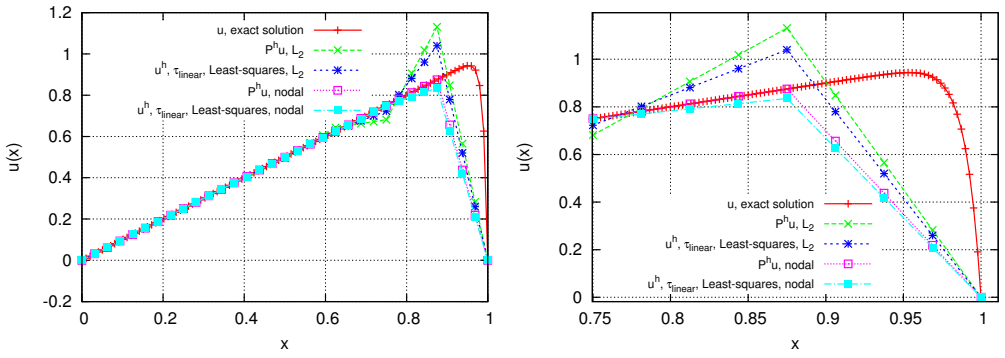
(a) $h = 1/8$ (b) $h = 1/16$

(c) $h = 1/32$ (d) $h = 1/64$

Figure 5.3: Variation of R_G, $||u-u^h||_{L_2}$ and $||\mathbb{P}^h u-u^h||_{L_2}$ with respect to c_1, with $\tau = c_1 h$, with varying element sizes h. $\mathbb{P}^h$ is now the nodal projector and is also used to construct the VGM residuals in this case.

In Figure 5.4, for a grid with 8 elements, the u^h obtained with τ_{linear} using the Least-squares VGM, with both the L_2 and nodal projector are shown. The corresponding projections of the exact solution, $\mathbb{P}^h u$ are also examined. As can be seen, the selection of $\mathbb{P}^h$

greatly influences $\mathbb{P}^h u$. Since the goal of the VMM and VGM is to obtain $u^h \approx \mathbb{P}^h u$, also u^h is largely determined by the selected projector. Using the nodal projector results in solutions which are non-oscillatory and have nodal values close to those of the exact solution. The solutions obtained with the L_2 projector are more oscillatory. In the solution of the VGM $c_1 = 0.46$ and $c_1 = 0.27$ were obtained when using the nodal projector and L_2 projector respectively. From Figures 5.2 and 5.3 it is clear that with these values of c_1 using the L_2 projector produced a lower L_2 error than when using the nodal projector.



(a) (b)

Figure 5.4: u^h obtained with Least-squares VGM, with $h = 1/8$, with τ_{linear}, using both the L_2 and nodal projector.

5.3 Performance of the VGM for different τ and mesh size with the L_2 projector

Here numerical results for the advection-diffusion equation are examined in more detail using only the L_2 projector. The numerical results were computed with $\mathcal{V}^h$ defined on uniform grids with $8, 16, 32$ and 64 elements, with respective element sizes of $h = 1/8, 1/16, 1/32$ and $h = 1/64$. On each grid, results were computed with τs optimized with both the Newton and Least-squares VGM. In Figure 5.5, for a grid with 8 elements, u^h is plotted along with the exact solution u and its projection onto the FEM basis $\mathbb{P}^h u$. u^h computed with $\tau = 0$ contains oscillations that characterize solutions of advection dominated equations obtained with the standard Galerkin method. The solutions obtained with non-zero τs closely match u and $\mathbb{P}^h u$ in the region $x < 0.8$. Near the right boundary $x = 1$, u^h with τ calibrated with either VGM procedure still match $\mathbb{P}^h u$ closely, but can no longer match u due to the limitations of the FEM basis.

In Figures 5.7a and 5.7b the L_2 error, $||u - u^h||_{L_2}$, and projected error, $||\mathbb{P}^h u - u^h||_{L_2}$, are plotted against the element size h. For large h it can be seen that the L_2 error is significantly higher when using $\tau = 0$ than when using τ_{shakib}, or the VGM calibrated τs. As the mesh is refined, i.e. h is made smaller, the differences are reduced. The results for the projected error confirm that the VMM and VGM are oriented towards obtaining $u^h \approx \mathbb{P}^h u$. Furthemore the projected error when calibrating with the VGM is always lower than when using τ_{shakib} without calibration. Finally, the results obtained with $\tau_{shakibVGM}$ produced the lowest projected errors showing the potential for nonlinear SGS models to perform well when combined with the VGM.

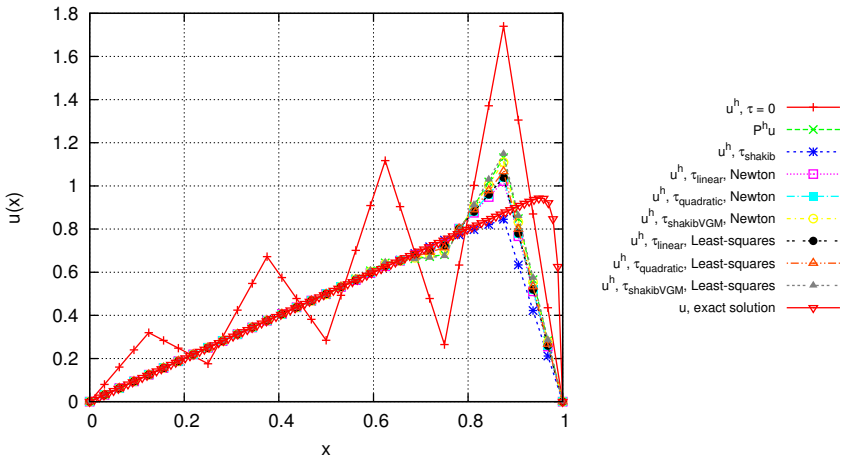


Figure 5.5: Exact, projected and numerical solutions obtained for the advection-diffusion equation with $h = 1/8$, with both the Newton and Least-squares VGM.

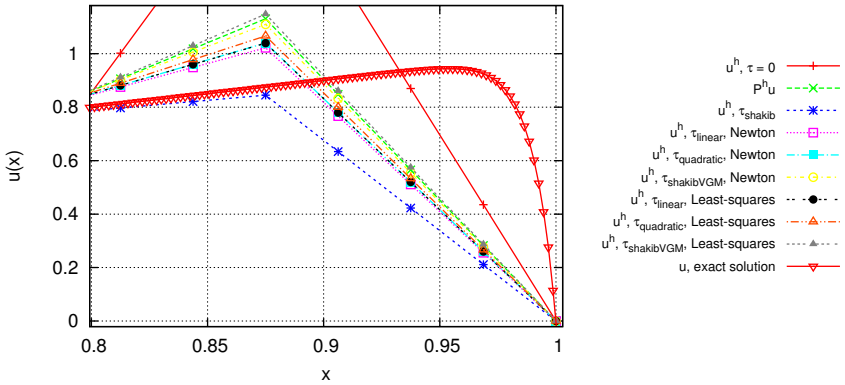
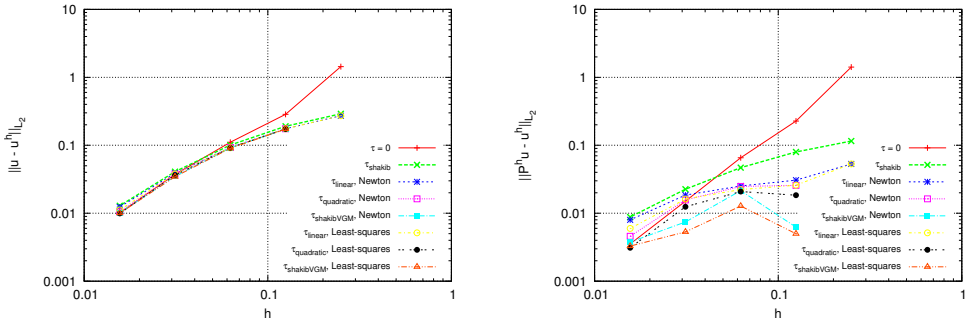


Figure 5.6: Advection-diffusion equation solutions, zoomed in near $x = 1$



(a) $||u - u^h||_{L_2}$ (b) $||\mathbb{P}^h u - u^h||_{L_2}$

Figure 5.7: L_2 error, $||u - u^h||_{L_2}$, and projected error, $||\mathbb{P}^h u - u^h||_{L_2}$, for the advection-diffusion equation plotted against h.

In Figure 5.8 the progression of the coefficients of each τ per iteration of the VGM is shown for $h = 1/32$. It can be seen that the coefficient values tend to converge within 10 Germano iterations for both the Newton and Least-squares VGM. In Figure 5.9 the

coefficient values at the final Germano iteration are plotted against h. As can be seen the $\vec{c}$ determined by the VGM can vary with h. This is not necessarily an indicator that the SGS model is scale variant in the absolute sense, however, nor that good results cannot be obtained. The coefficient values obtained via the Newton and Least-squares VGM do not differ significantly in most cases. Only when $h = 1/64$ the least-squares VGM produces a large negative c_1 for $\tau_{shakibVGM}$ whereas for with the Newton VGM a positive c_1 is obtained. However this difference does not seem to impact the performance of the method in terms of L_2 error or projected error.

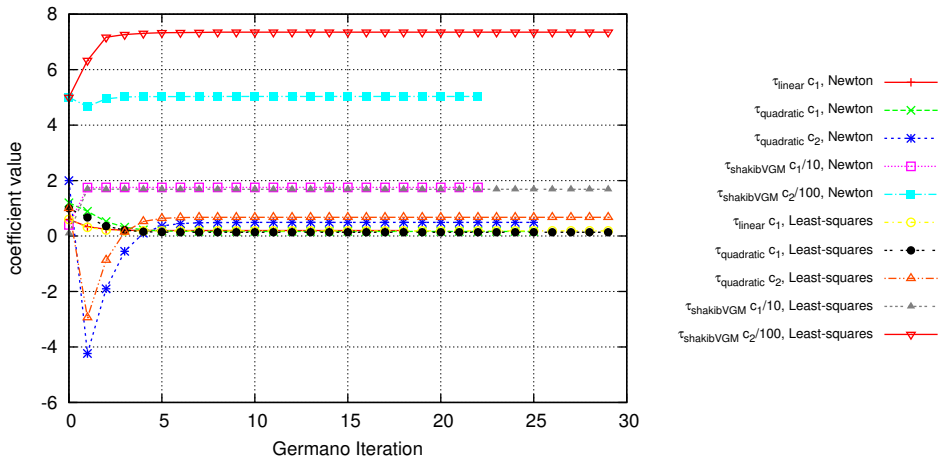
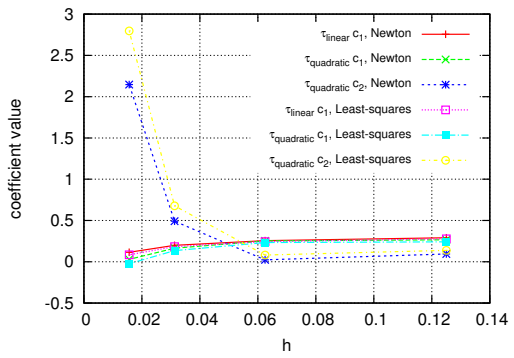
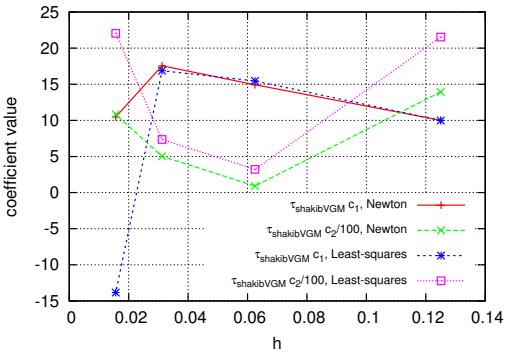


Figure 5.8: Coefficient values with respect to iteration of the VGM



(a) variation of τ_{linear} and $\tau_{quadratic}$ coefficients



(b) variation of $\tau_{shakibVGM}$ coefficients

Figure 5.9: Variation of converged coefficient values with respect to h.

The results of Figure 5.9 show some degree of invariance with h for c_1 for τ_{linear}. This is misleading however, since the SGS model is not scale-invariant which invalidates the assumptions used in the definition of R_G. Furthermore these results do not minimize R_G, but find $\vec{c}$ for $\frac{\partial R_G}{\partial \vec{c}} = 0$. This illustrates the very complex nature of interpreting the results of VGM mesh studies.

Chapter 6

Burgers' Equation

A significant advantage of the VGM is that it can be applied to equations where the analytical prediction of model coefficients becomes unfeasible. As an illustration this chapter considers the application of the VGM to optimize subgrid-scale models for the one-dimensional unsteady forced Burgers' equation. The problem is defined as:

$$\begin{aligned}
&\text{find } u(x,t) \in \mathcal{V} : \\
&\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} - \frac{1}{Re}\frac{\partial^2 u}{\partial x^2} = f, \text{ on } \Omega\times]0,T[\\
&u = 0, \text{ on } \delta\Omega \\
&u(x,0) = 0
\end{aligned} \tag{6.1}$$

where $\Omega :=]0,1[$ is the unit line, $\delta\Omega$ is the boundary of Ω, T is the final time and Re is a dimensionless parameter. $T = 25$ and $Re = 512$ are used here. The forcing function f is defined to be a spatially and temporally varying sine wave with an additional constant term:

$$f := 10\sin(t)\sin(2\pi x) + 11 \tag{6.2}$$

this ensures that u will form a sharp layer near the right boundary of the domain.

The FEM space $\mathcal{V}^h$ is again defined as the space of standard linear FEM basis functions as in (5.4). Application of the VMM, as described in Section 2, to (6.1) then results in the following variational problem:

$$\begin{aligned}
&\text{find } u^h \in \mathcal{V}^h : \\
&B(w^h,u^h) + M(w^h,u^h;\vec{c},f,h) = (w^h,f), \ \forall w^h \in \mathcal{V}^h \\
&u^h = 0, \text{ on } \delta\Omega \\
&u^h(x,0) = 0 \\
&B(w^h,u^h) := \left(w^h, \frac{\partial u^h}{\partial t}\right) - \frac{1}{2}\left(\frac{\partial w^h}{\partial x}, (u^h)^2\right) + \frac{1}{Re}\left(\frac{\partial w^h}{\partial x}, \frac{\partial u^h}{\partial x}\right) \\
&M(w^h,u^h;\vec{c},f,h) := \left(\frac{\partial w^h}{\partial x}, u^h\tau\mathcal{R}\right) \\
&\mathcal{R} := \frac{\partial u^h}{\partial t} + u^h\frac{\partial u^h}{\partial x} - f
\end{aligned} \tag{6.3}$$

where second order products of u' have been neglected and all second order spatial derivatives vanish due to the use of linear elements. Once τ is defined, (6.3) is integrated in time using the generalized-α method as proposed by Jansen et al in [17]. In this paper the time-step $\Delta t = 0.25$ is used.

The following definitions for τ are proposed, with coefficients c_n to be optimized with the VGM:

$$\tau_{linear} := c_1 h \tag{6.4}$$

$$\tau_{quadratic} := c_1 h + c_2 h^2 \tag{6.5}$$

$$\tau_{cubic} := c_1 h + c_2 h^2 + c_3 h^3 \tag{6.6}$$

For the least-squares VGM the following form of Shakib's τ for unsteady advective-diffusive systems is used:

$$\tau_{shakibVGM} := \left(\frac{4}{\Delta t^2} + c_1^2 \left(\frac{u}{h} \right)^2 + 100 c_2^2 \left(\frac{1}{Reh^2} \right)^2 \right)^{-\frac{1}{2}} \tag{6.7}$$

The squares of c_1 and c_2 are used as this prevents the square-root becoming negative which sometimes occurred with the standard definition of Shakib's τ. The factor 100 that pre-multiplies c_2 is to ensure that c_1 and c_2 have similar orders of magnitude which improves the performance of the BFGS algorithm. For the Newton VGM it was only possible to use the following form:

$$\tau_{shakibVGM} := \left(\frac{4}{\Delta t^2} + c_1 \left(\frac{u}{h} \right)^2 + 9 \left(\frac{4}{Reh^2} \right)^2 \right)^{-\frac{1}{2}} \tag{6.8}$$

This was due to the fact that when using a definition with two coefficients linear dependence was encountered which caused the Newton algorithm to become unstable.

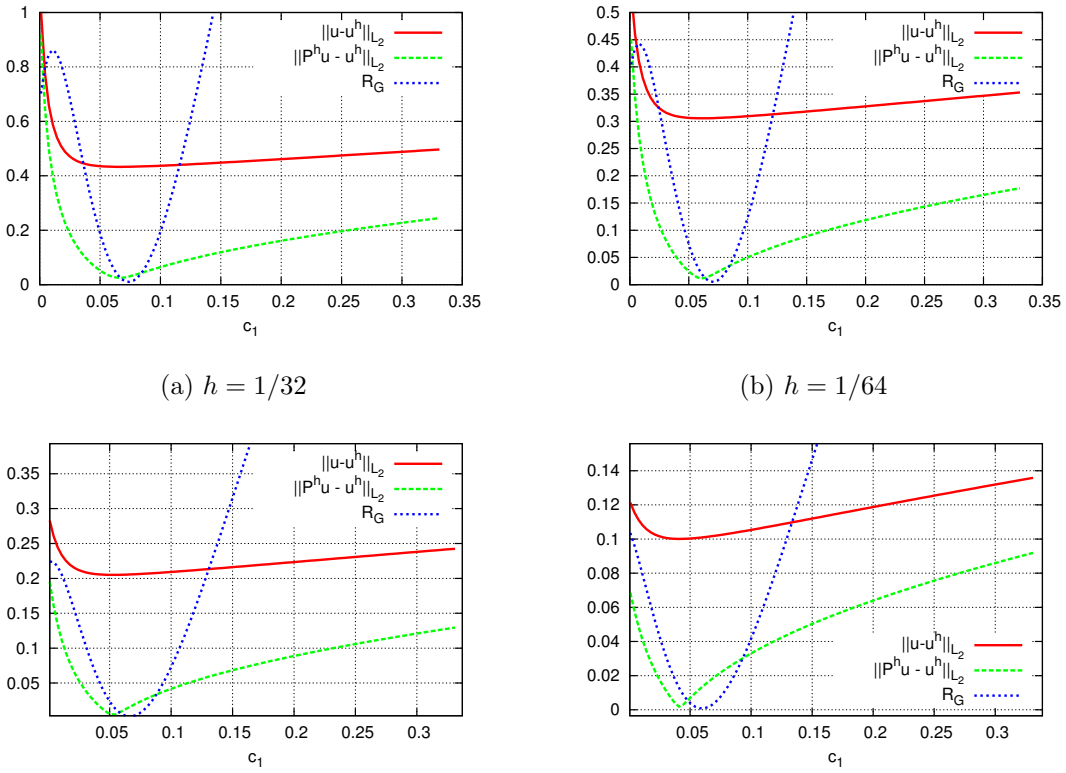
For the VGM the coarse functions spaces $\mathcal{V}^{h_i}$ are again defined by partitioning Ω into grids with elements of size $2^i h$. The VGM optimization is started after five time steps have been performed with the initial coefficient values. As a benchmark, the definition of τ as proposed by Shakib in [13] is used.

$$\tau_{shakib} := \left(\frac{4}{\Delta t^2} + 4 \left(\frac{u}{h} \right)^2 + 9 \left(\frac{4}{Reh^2} \right)^2 \right)^{-\frac{1}{2}} \tag{6.9}$$

6.1 Burgers' equation Germano residual investigation

In a similar fashion to the advection-diffusion equation, here the Germano residual, L_2 error $||u - u^h||_{L_2}$ and projected error $||\mathbb{P}^h u - u^h||_{L_2}$ are considered for Burgers' equation with $\tau = \tau_{linear} = c_1 h$. Figure 6.1 shows the variation of the quantities at $t = 25$ as c_1 is varied. A Direct Numerical Simulation (DNS) solution, computed on a grid with 1024 elements, is used to represent u. The plots are generated by performing simulations up to $t = 25$, recording the quantities and then restarting the simulation with a different value of c_1. Here too it is evident that it is possible to obtain $u^h \approx \mathbb{P}^h u$ by using a globally constant τ. This is a surprising result as for nonlinear equations such as Burgers' equation the unresolved-scale green's function theoretically does not exist and represents an approximation. A globally constant τ is a further approximation of the green's function. However, despite the very approximate nature of the SGS model, accurate results can still

be obtained in this case. Furthermore, contrary to the advection-diffusion case, the value of c_1 that minimizes R_G remains relatively constant at $c_1 \approx 0.07$. This value of c_1 is also close to the value that produces $u^h \approx \mathbb{P}^h u$ and minimizes the L_2 error. Therefore in the numerical results that follow it is expected that τ_{linear} will perform well with the VGM for the h range considered here.



(a) $h = 1/32$ (b) $h = 1/64$

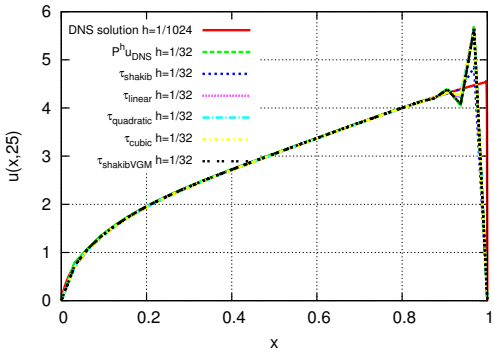
(c) $h = 1/128$ (d) $h = 1/256$

Figure 6.1: Variation of R_G, $||u - u^h||_{L_2}$ and $||\mathbb{P}^h u - u^h||_{L_2}$, at $t = 25$, with respect to c_1, with $\tau = c_1 h$, with varying element sizes h

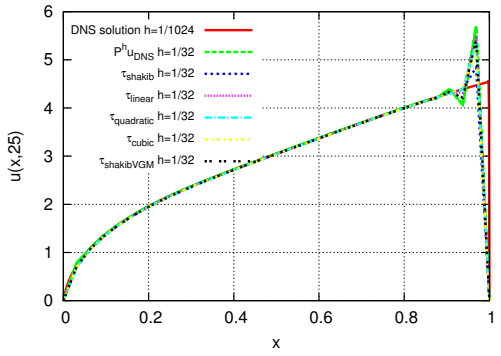
6.2 Numerical results

Results for Burgers' equation, with grids containing $32, 64, 128$ and 256 elements respectively, are now presented. Figure 6.2 shows plots of the instantaneous solutions obtained on a grid with 32 elements. Similar to the advection-diffusion results, away from the region $x = 1$ all solutions tends to overlap. Near $x = 1$ the coarse grid solutions cannot match the DNS solution, but the solutions obtained with the VGM match closely with the projection of the DNS solution.

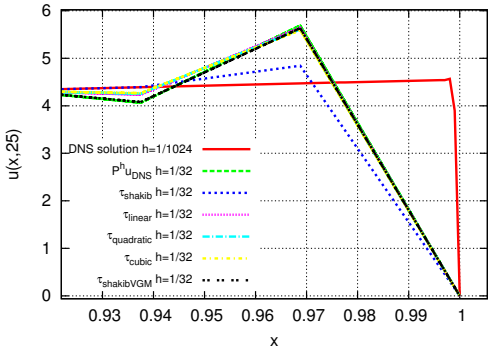
In Figure 6.3 the L_2 error and projected error of each numerical solution, with respect to the DNS solution, at the 100th time-step, is shown. In terms of the L_2 error there is not much difference between the results obtained with τ_{shakib} or the VGM calibrated τs. However, the projected errors obtained with the VGM are much lower than when using τ_{shakib}. In terms of projected error the Least-squares VGM outperforms the Newton VGM. But, with $\tau_{shakibVGM}$ with the Least-squares VGM the projected error increases as the mesh is refined.



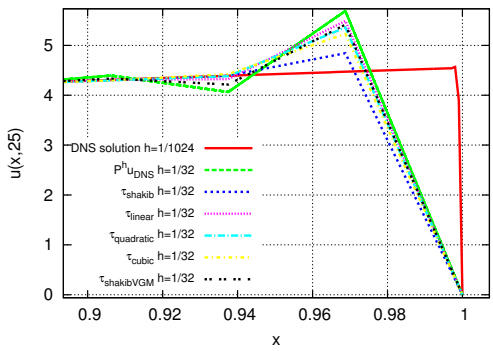
(a) Instantaneous solutions, least-squares VGM



(b) Instantaneous solutions, Newton VGM



(c) Instantaneous solutions near $x = 1$, least-squares VGM



(d) Instantaneous solutions near $x = 1$, Newton VGM

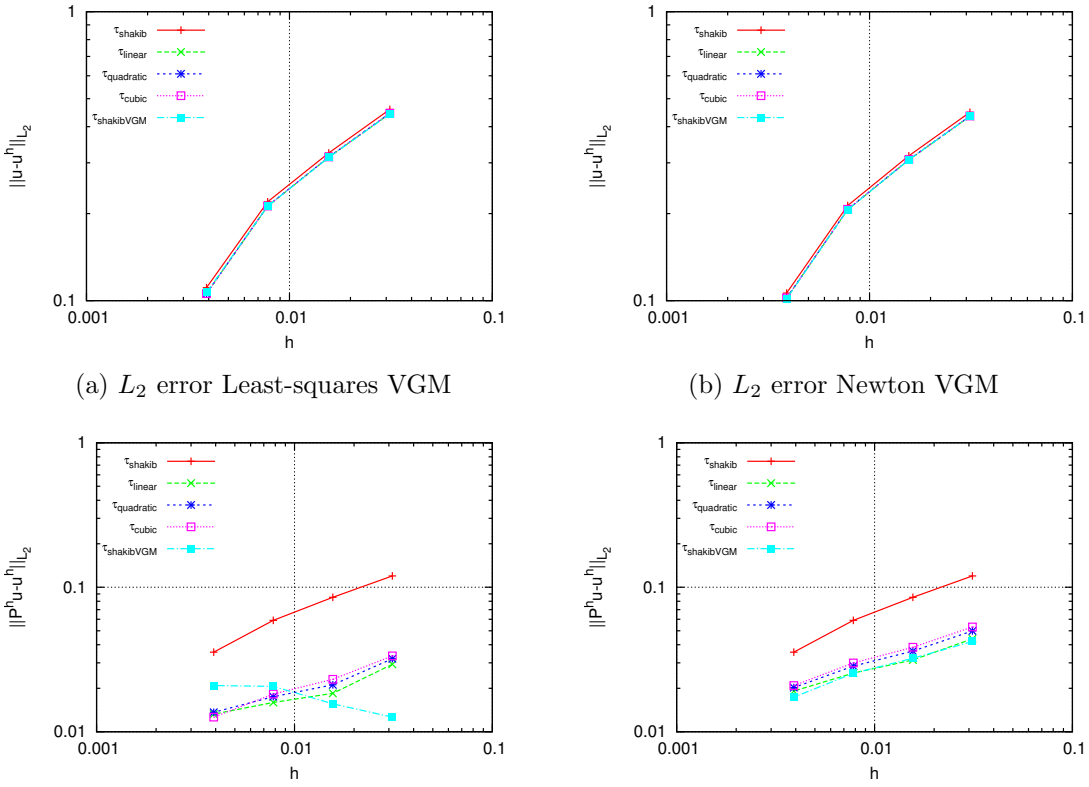
Figure 6.2: Instantaneous solutions for Burgers' equation at $t = 25$ with $h = 1/32$ for both Least-squares and Newton VGM

The coefficient values obtained with the VGM, at $t = 25$, for the different values of h are shown in Figure 6.4. It can be seen that the values of c_1 for τ_{linear}, $\tau_{quadratic}$ and τ_{cubic} do not vary significantly with h. In particular, for τ_{linear}, the values of c_1 obtained are close to the optimal values that minimize $||\mathbb{P}^h u - u^h||_{L_2}$. This indicates that indeed SGS models that are scale-invariant and concurrent result in models that work well with the VGM. Furthermore the values found from the solution of the VGM, i.e. $\frac{\partial R_G}{\partial \vec{c}} = 0$, are close to the global minimum of R_G, i.e. $\frac{dR_G}{d\vec{c}} = 0$. This result is a confirmation of the assertions made in Chapter 4.

The values of c_2 and c_3 tend to vary more with h showing that the addition of extra factors of h to the SGS models may be unnecessary in this case. c_1 for $\tau_{shakibVGM}$ with the Least-squares VGM also remains relatively unchanged as h is varied. τ_{linear} and $\tau_{shakibVGM}$ produce the lowest projected errors. It is interesting to note that the values of c_1 obtained for $\tau_{shakibVGM}$ with the VGM are more than twice the standard value of τ_{shakib}.

Figure 6.5 shows the coefficient values obtained at each time-step with $h = 1/32$. As can be seen the coefficients are not steady with time and follow a periodic pattern. The values of c_1 do not vary significantly with time for all τs, however c_2 and c_3 can show large variations.

Finally, in Figure 6.6 the computation time increase from using the VGM is considered. The ratio of time taken to complete a simulation without and with the VGM is plotted



(a) L_2 error Least-squares VGM

(b) L_2 error Newton VGM

(c) Projected error Least-squares VGM

(d) Projected error Newton VGM

Figure 6.3: Variation of L_2 error and projected error at the 100th time-step for both Least-squares and Newton VGM

as a function of h for all τs for both the Least-squares and Newton VGM. In most cases the computation time ratio is lower than 1.4 showing that the VGM does not significantly increase the computational effort required for a simulation. However for τ_{cubic} with the Least-squares VGM the computation time ratio is close to 1.8-1.9 for certain cases. This can be attributed to the BFGS algorithm taking a large number of steps to reach the stopping criterion for the τ_{cubic} case. For example for $\tau_{shakibVGM}$ on average 3-5 BFGS iterations were necessary to converge. For τ_{cubic} generally 10-17 iterations were needed.

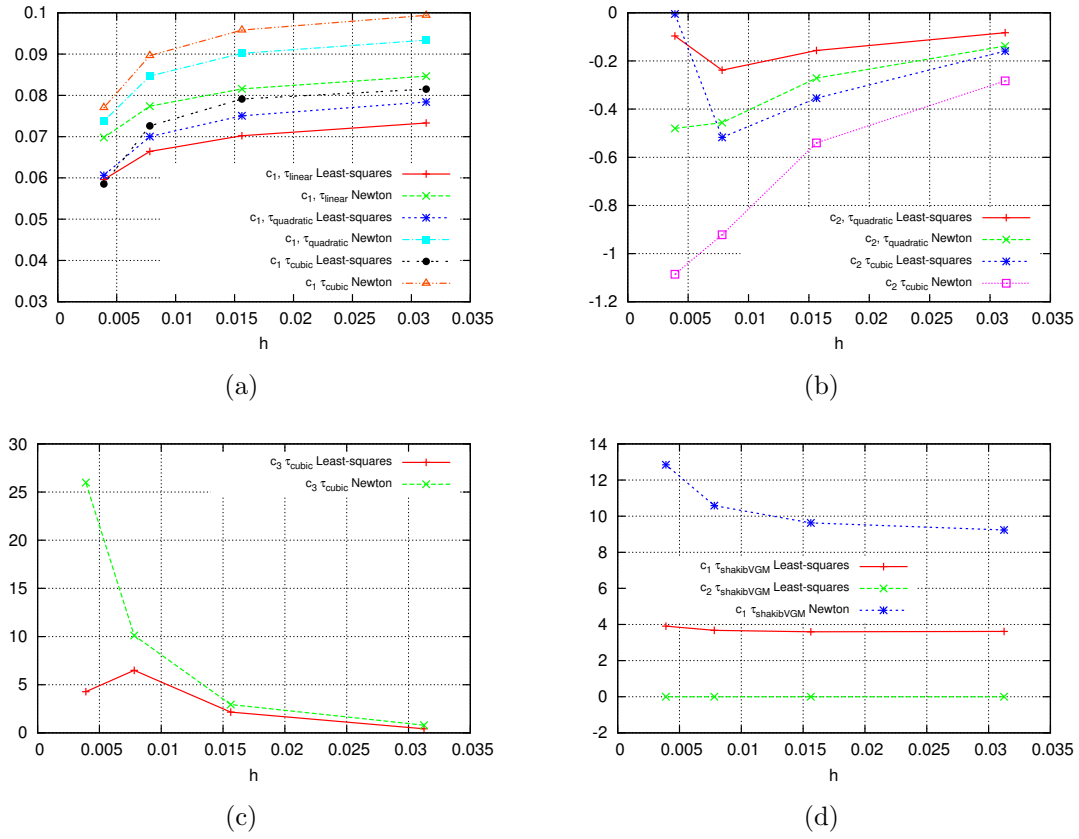


Figure 6.4: Variation of coefficient values at the 100th time-step with h for both Least-squares and Newton VGM

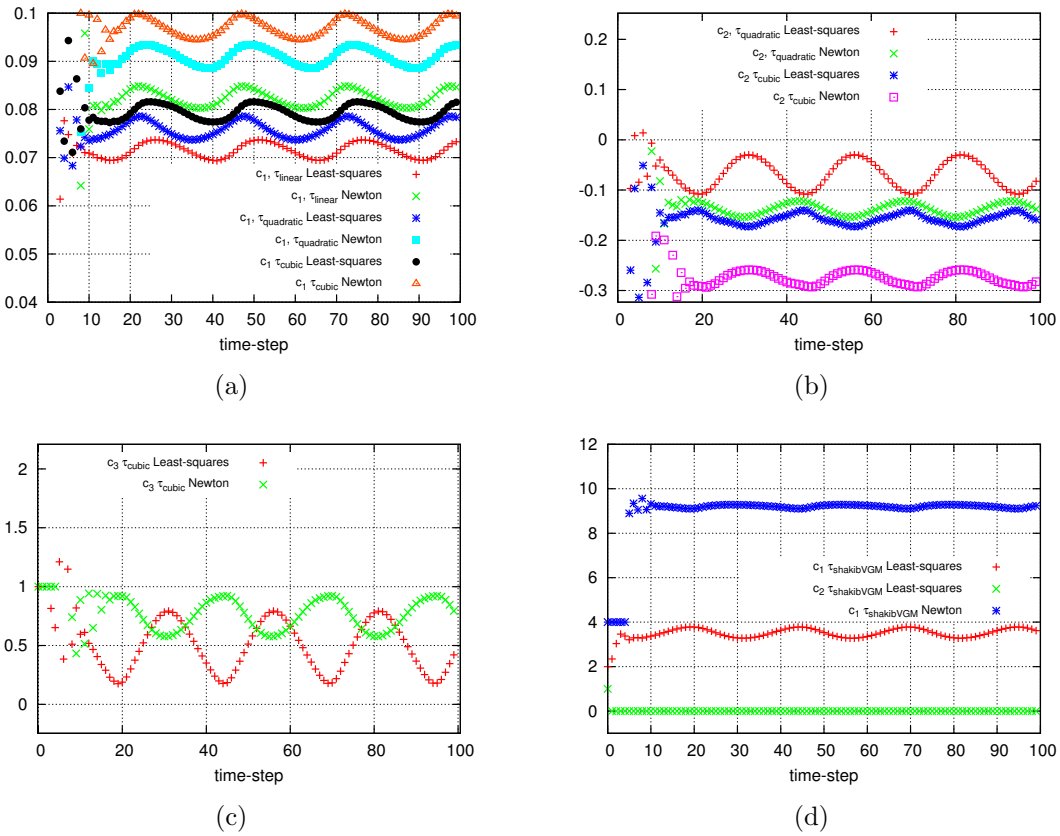


Figure 6.5: Coefficient values for τs calibrated with the Least-squares and Newton VGM per time-step with $h = 1/32$

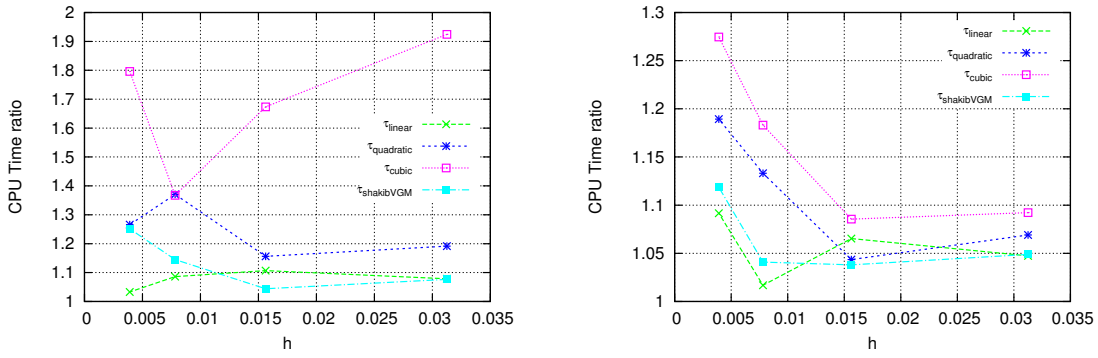


Figure 6.6: Computation time increase ratio when using the VGM as a function of h. Time is measured as time to complete 100 time-steps

Chapter 7

Stokes Equations

In the previous chapters single PDEs were considered which meant that a single R_G could be constructed. In this chapter a case is considered with multiple PDEs and $R_{k,G}$s. This makes the satisfaction of the concurrency requirment of Chapter 4 more difficult. In this chapter the steady two-dimensional Stokes equations are considered which leads to multiple VGM residuals for which this condition becomes relevant. Specifically, the following problem is considered:

$$\begin{aligned} &-\nabla\cdot(2\nu\nabla^s\boldsymbol{u})+\nabla p=\boldsymbol{f},\ \text{on }\Omega\\ &\nabla\cdot\boldsymbol{u}=0,\ \text{on }\Omega\\ &\boldsymbol{u}=0,\ \text{on }\delta\Omega,\quad \int_\Omega p\,\mathrm{d}\Omega=0 \end{aligned} \tag{7.1}$$

where $\Omega:=]0,1[\times]0,1[$ is the unit-square and $\nu\in\mathbb{R}$ is the viscosity parameter. $\nabla^s\boldsymbol{u}:=1/2(\nabla\boldsymbol{u}+\nabla\boldsymbol{u}^T)$ denotes the symmetric velocity gradient. $\boldsymbol{f}$ is the forcing vector and is defined here by inserting a manufactured solution of $\sin(4\pi x)\sin(4\pi y)$ for each solution variable.

The VMM is applied to (7.1), where both $\boldsymbol{u}$ and p contain resolved and unresolved scales such that $\boldsymbol{u}=\boldsymbol{u}^h+\boldsymbol{u}'$ and $p=p^h+p'$. The unresolved scales are then defined as:

$$\begin{aligned} &\boldsymbol{u}':=-\tau_m\boldsymbol{R}_m,\quad \boldsymbol{R}_m:=\nabla p^h-\boldsymbol{f}\\ &p':=-\tau_c R_c,\quad R_c:=\nabla\cdot\boldsymbol{u} \end{aligned} \tag{7.2}$$

Defining $\mathcal{V}^h$ and $\mathcal{Q}^h$ as spaces of standard bilinear finite element basis functions, in two dimensions on quadrilateral elements, with $v\in\mathcal{V}^h: v=0$ on $\delta\Omega$ and $v\in\mathcal{Q}^h: \int_\Omega v\,\mathrm{d}\Omega=0$, results in the following variational problem formulation:

$$\begin{aligned} &\text{find }\boldsymbol{u}^h\in\mathcal{V}^h\times\mathcal{V}^h, p^h\in\mathcal{Q}^h:\\ &(\nabla^s\boldsymbol{w}^h,2\nu\nabla^s\boldsymbol{u}^h)-(\nabla\cdot\boldsymbol{w}^h,p^h)+(q^h,\nabla\cdot\boldsymbol{u}^h)+(\nabla\cdot\boldsymbol{w}^h,\tau_c\nabla\cdot\boldsymbol{u}^h)\\ &+(\boldsymbol{\nabla}q^h,\tau_m(\boldsymbol{\nabla}p^h-\boldsymbol{f}))=(\boldsymbol{w}^h,\boldsymbol{f}),\ \forall\ \boldsymbol{w}^h\in\mathcal{V}^h\times\mathcal{V}^h, q^h\in\mathcal{Q}^h \end{aligned} \tag{7.3}$$

For the VGM optimization of τ_m and τ_c the coarse spaces are defined in a manner similar to the procedure used for the forced Burgers' equation, where each coarse space Ω is partitioned into elements with edge length 2^ih. Only The least-squares version of the Germano identity is used. With the Newton VGM the residual cancellation effect mentioned in Chapter 3 was encountered making the method unusable. The Least-squares

Germano residuals are split into residuals of the momentum and continuity equations. If the x-momentum, y-momentum and continuity equations are given the indexes $1, 2$ and 3 respectively, then τ_m is optimized by minimizing $R_{c,G} = R_{3,G}$ and τ_c is optimized by minimizing $R_{m,G} = R_{1,G} + R_{2,G}$, where $R_{k,G}$ is defined by (3.6).

Two definitions for τ_m and τ_c are selected from literature:

$$\tau_m := c_1 \frac{h^2}{24\sqrt{2}\nu}, \quad \tau_c := c_2 \nu \tag{7.4}$$

$$\tau_m := c_1 \frac{h^2}{24\sqrt{2}\nu}, \quad \tau_c := \frac{h\sqrt{\boldsymbol{u}\cdot\boldsymbol{u}}}{4} \tag{7.5}$$

where (7.4) was proposed by Franca and Hughes in [18] and (7.1) was proposed by Taylor et al in [19]. (7.4) and (7.5) are referred to as the linear τs and nonlinear τs from this point onwards. An additional case is defined by setting $\tau_c = 0$, which will be referred to as "τ_m only".

7.1 Stokes equations Germano residual investigation

In [7], Chen, Maher and Hulshoff compare the coefficients of SGS models obtained from a goal-oriented optimization procedure and the VGM for Stokes equations. Via the goal-oriented procedure, coefficients are found for which the L_2 error with respect to the exact solution is minimized. The goal-oriented coefficients can be used to assess the quality of the coefficients obtained from the VGM. The relevant results and discussion are reproduced here.

In this study model coefficients are found using both the goal-oriented procedure and the VGM. The goal-oriented procedure can be used to define coefficient reference values. The VGM results, however, are subject to limitations introduced by the approximation of the subgrid scales. In particular to what extent the SGS model simultaneously minimizes the VGM residuals, and to what extent its optimal coefficients are independent of mesh size.

7.1.1 τ_m only models

First the linear model (7.4) with $\tau_c = 0$ is considered.

$$(\tau_\mathrm{m} \text{ only}) : \tau_m = c_1 \frac{h^2}{24\sqrt{2}\nu} \tag{7.6}$$

Results for c_1 versus h obtained using the goal-oriented procedure are shown in Figure 7.1. The goal-oriented procedure converges uniformly for this case, requiring on average 10 iterations when starting from an initial guess of $c_1 = 1.0$. Clearly the (τ_m only) model is scale invariant. Also shown in Figure 7.1 are the values of c_1 obtained from the VGM, corresponding to $\frac{\partial R_{c,G}}{\partial \vec{c}} = 0$. These coefficients can be seen to vary with h.

The VGM results are considered in more detail in Figure 7.2, which shows the variation of $R_{m,G}$, $R_{c,G}$, the L_2 error and $||\mathbb{P}^h\boldsymbol{u} - \boldsymbol{u}^h||$ with c_1. For plotting purposes, $R_{m,G}$ and $R_{c,G}$ are rescaled to be one when $c_1 = 0$. It is evident that in all cases $||\boldsymbol{u} - \boldsymbol{u}^h||_{L_2} \approx ||\mathbb{P}^h\boldsymbol{u} - \boldsymbol{u}^h||_{L_2}$ and is minimized for the same value of c_1, which is identical to the value determined by the goal-oriented method. This shows that for these cases, L_2 optimality is equivalent to minimizing the projected error. However, the c_1 obtained from minimizing $R_{c,G}$ does not correspond to the value that minimizes either $R_{m,G}$ or $||\mathbb{P}^h\boldsymbol{u} - \boldsymbol{u}^h||$. Therefore

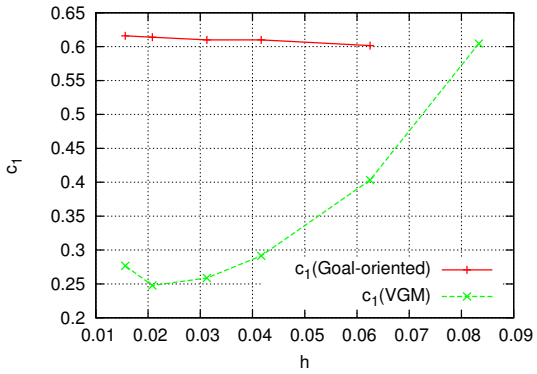
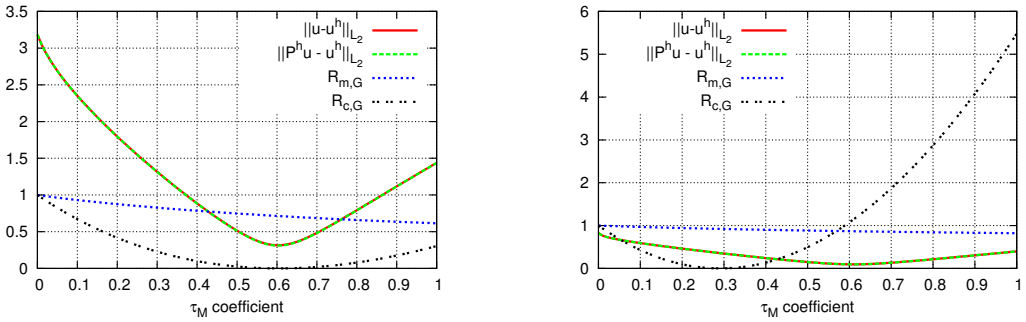


Figure 7.1: c_1 versus h for the (τ_m only) model

the (τ_m only) model is not concurrent (see Chapter 4) and information of the adjoint problem is needed to minimize the projected error. It is further clear that a scale-invariant SGS model may not retain constant $\vec{c}$ when calibrated with the VGM.



(a) (τ_m only) : $h = 1/12$ (b) (τ_m only) : $h = 1/24$

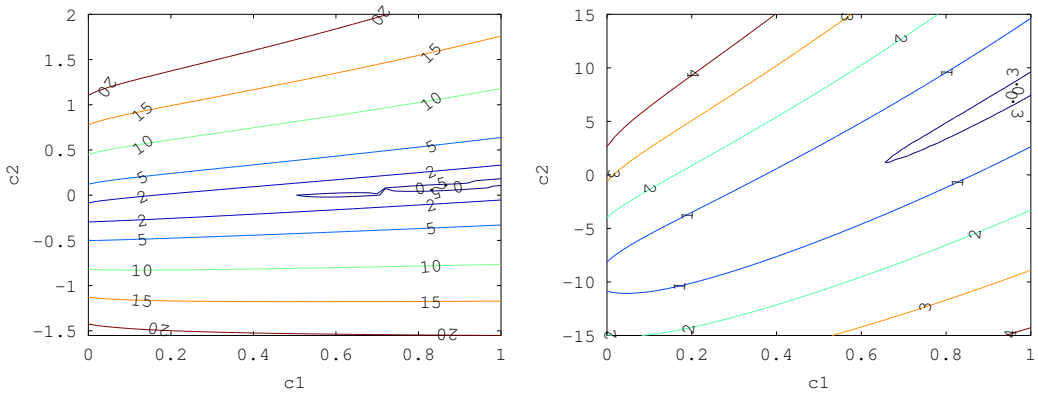
Figure 7.2: Plot of $R_{m,G}$, $R_{c,G}$, L_2 error and $||\mathbb{P}^h u - u^h||$ for different values of c_1 for the τ_m only model with $h = 1/12$ and $h = 1/24$.

7.1.2 Models with both τ_m and τ_c

Now the more conventional linear (7.4) and nonlinear (7.5) models which use both τ_m and τ_c are considered. Figure 7.3 shows contours of the L_2 error versus c_1 and c_2 for the manufactured solution on uniform meshes of 12×12, 24×24 and 48×48 elements. These are the two-dimensional counterparts of the $||\boldsymbol{u} - \boldsymbol{u}^h||_{L_2}$ curves of Figure 7.2. For both the linear and nonlinear models, as the mesh is refined, the L_2 error becomes less sensitive to c_1. For the linear models the location of minimum L_2 error remains relatively fixed for the different meshes, whereas for the nonlinear models the minimum location shifts.

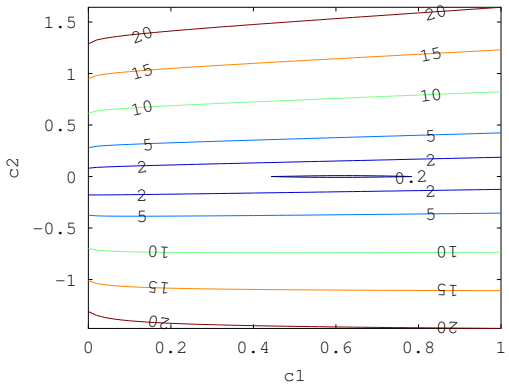
Optimal coefficients obtained with goal-oriented optimization are compared to those obtained with the VGM in Figure 7.4. The coefficients determined by the VGM vary widely in this range. Only the goal-oriented c_1 value is relatively constant, while the other coefficients show wide variation but possibly convergence for smaller h.

Figure 7.5 shows the behavior of $R_{c,G}$ and $R_{m,G}$ for the linear model. These indicate that the model lacks both concurrency and scale invariance in the h range of greatest interest. A similar conclusion can be obtained by examining $R_{c,G}$ and $R_{m,G}$ for the

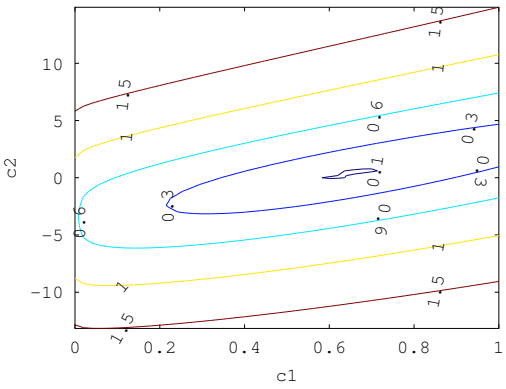


(a) 12×12 elements

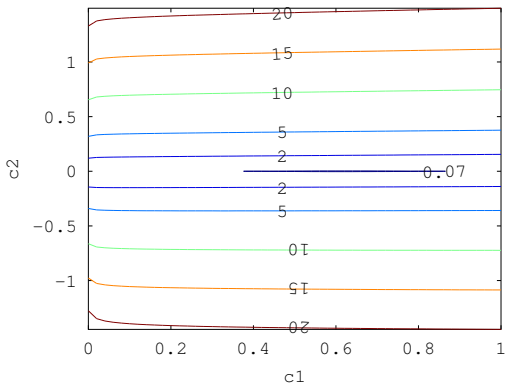
(b) 12×12 elements



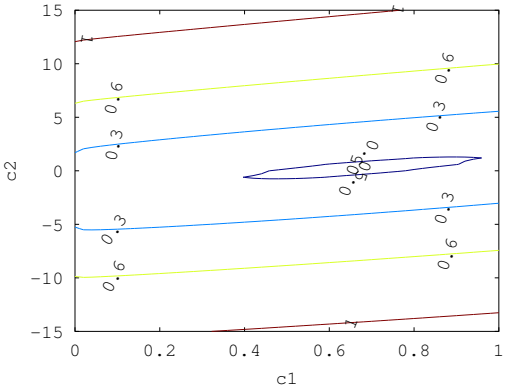
(c) 24×24 elements



(d) 24×24 elements



(e) 48×48 elements



(f) 48×48 elements

Figure 7.3: L_2 error variation with respect to c_1 and c_2. Left: linear SGS model, Right: nonlinear SGS model

nonlinear model in Figure 7.6. Thus information about the adjoint problem is again needed to be able to minimize the projected error. From Figures 7.5 and 7.6 it can be seen that the values of c_1 and c_2 obtained by the VGM are usually closer to the minimums of $R_{c,G}$ and $R_{m,G}$ than those obtained by the goal-oriented procedure. This is coherent with (4.3) in that obtaining coefficients that minimize the L_2, or projected error may not necessarily require a minimization of $R_{c,G}$ and $R_{m,G}$.

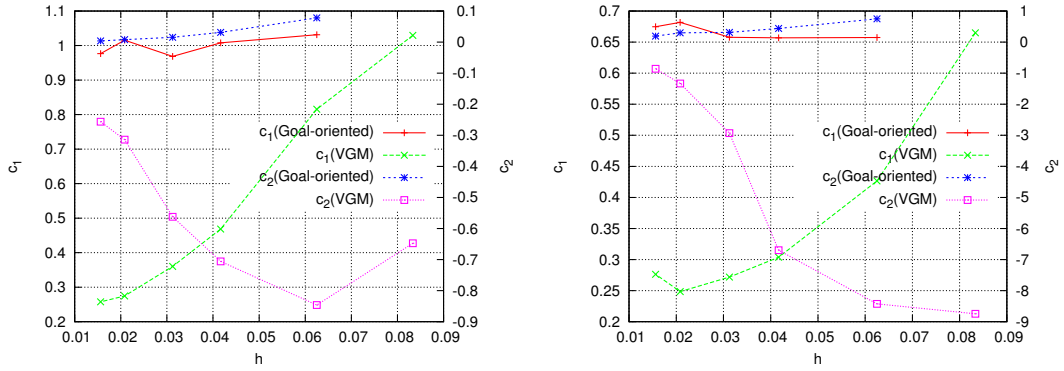


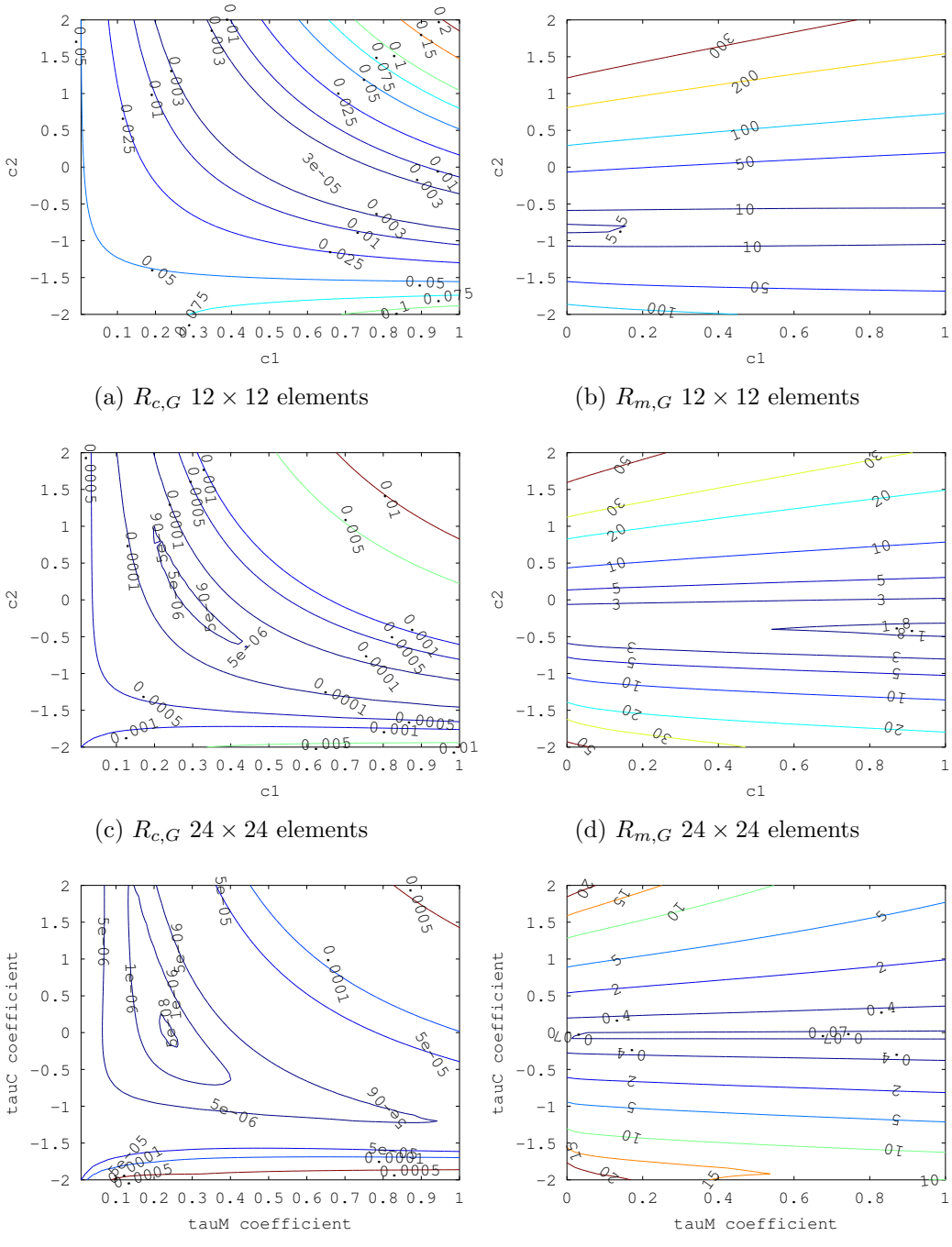
Figure 7.4: τ coefficients versus h. Left: linear stabilization parameters, right: nonlinear stabilisation parameters.

7.2 Numerical results

Numerical results are computed for the Stokes equations on grids with sizes: $h = 1/12$, $1/16$, $1/24$, $1/32$, $1/48$ and $h = 1/64$. A graph of the L_2 error for the different mesh sizes and τ definitions is shown in Figure 7.7. It is directly evident that the different definitions of τ result in very different error and convergence properties. The linear τs have the highest error and first-order convergence. The nonlinear τs perform better than the linear τs and comparably to the τ_m only case. This indicates that including a nonlinear dependence in τ may lead to improved performance, but that a correct parametrization of τ with h remains important.

Figure 7.7 also shows the consequence of the differences between the VGM and goal-oriented method. Although the VGM results converge uniformly with mesh size, the value of the VGM errors can be more than an order of magnitude greater than those of their goal-oriented counterparts. Note that the goal-oriented values for the non-linear model are also superior to those suggested in [19] by Taylor et al. Furthermore, while neither the linear, or nonlinear τs are scale-invariant or concurrent, the nonlinear τs produce vastly lower L_2 errors than the linear τs when used with the VGM. This makes it clear that when a model lacks concurrency, scale-invariance may not be required for it to work well with the VGM.

Figure 7.8 shows the values of c_1 and c_2 obtained after each Germano iteration, for each τ definition, for the case $h = 1/24$. All τ definitions converge within five iterations, except for the linear τs which take approximately 35 iterations to converge. Additionally, the BFGS algorithm converged to the specified tolerance for all τs. This is evidence that the BFGS algorithm is a suitable method for minimizing the least-squares Germano residuals.



(a) $R_{c,G}$ 12×12 elements

(b) $R_{m,G}$ 12×12 elements

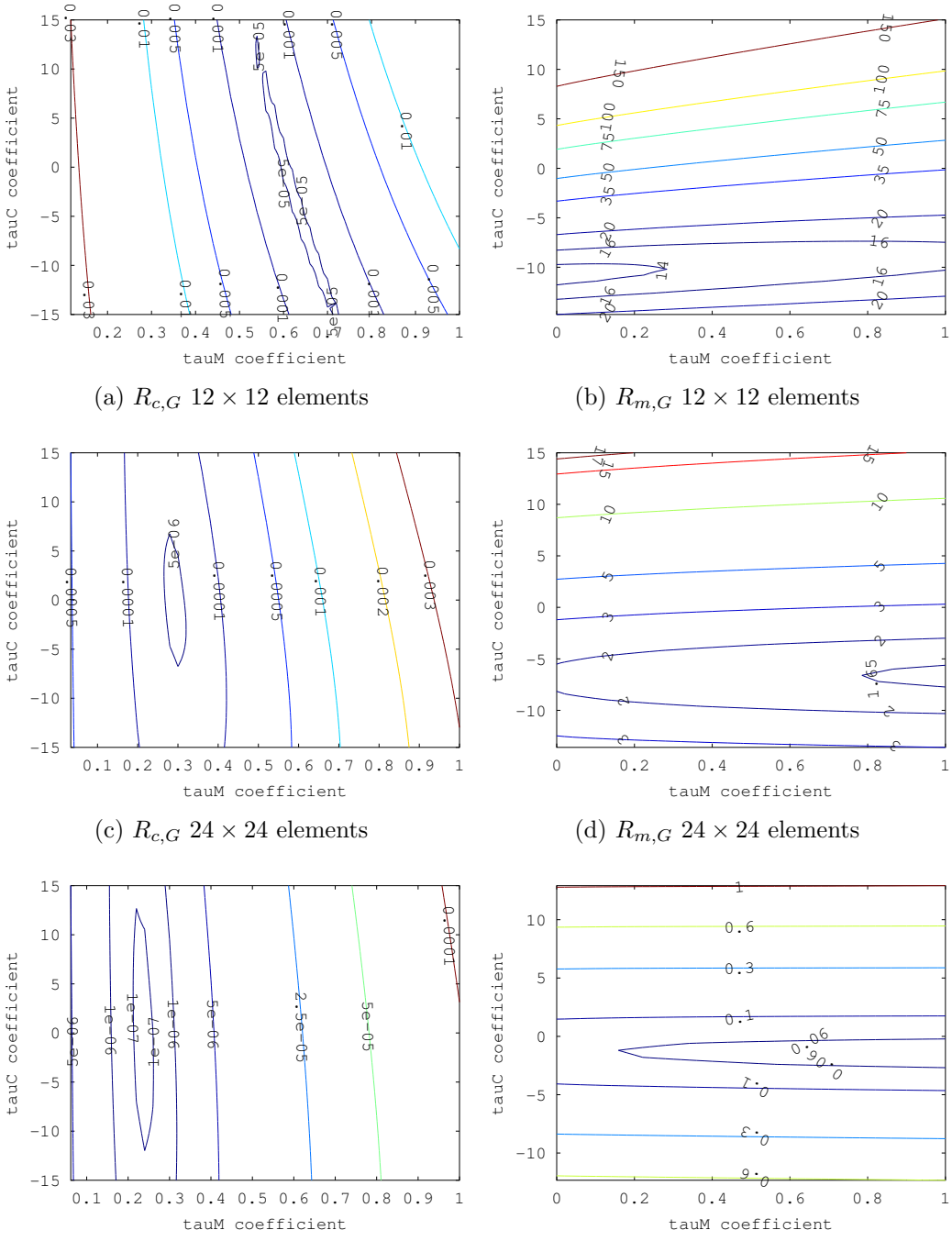
(c) $R_{c,G}$ 24×24 elements

(d) $R_{m,G}$ 24×24 elements

(e) $R_{c,G}$ 48×48 elements

(f) $R_{m,G}$ 48×48 elements

Figure 7.5: $R_{c,G}$ and $R_{m,G}$ contours for the linear model



(a) $R_{c,G}$ 12×12 elements

(b) $R_{m,G}$ 12×12 elements

(c) $R_{c,G}$ 24×24 elements

(d) $R_{m,G}$ 24×24 elements

(e) $R_{c,G}$ 48×48 elements

(f) $R_{m,G}$ 48×48 elements

Figure 7.6: $R_{c,G}$ and $R_{m,G}$ contours for the nonlinear model

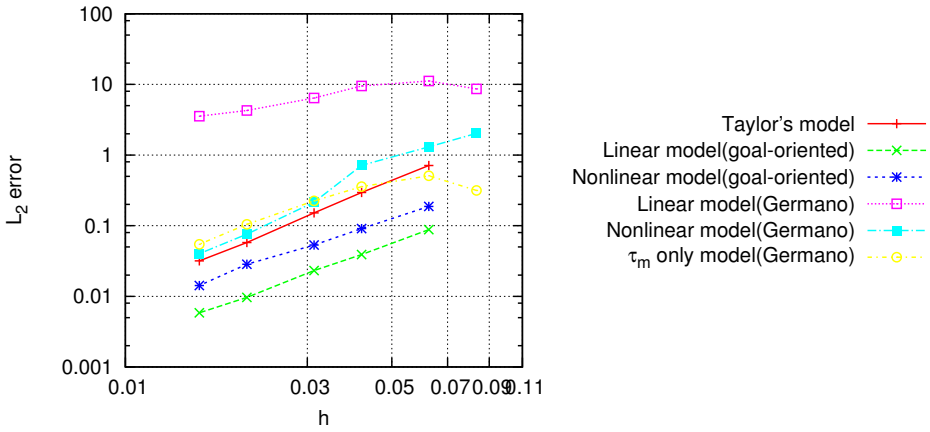
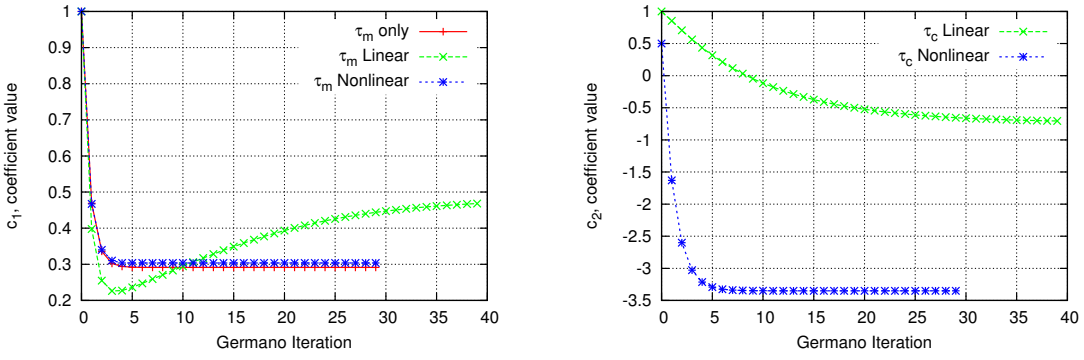


Figure 7.7: L_2 error convergence with h for VGM optimized Stokes equations



(a) c_1 coefficient value

(b) c_2 coefficient value

Figure 7.8: Values of c_1 and c_2 obtained with VGM iterations $h = 1/24$

Chapter 8

Conclusion

This thesis demonstrated how the Newton and BFGS algorithm could be used to solve the standard VGM relations and least-squares formulation respectively, for arbitrary forms of the τ parameter, including nonlinear τs, appearing in the VMM. The proposed procedure were also shown to be able to handle arbitrary projectors that are compatible with the VGM. When applied to the advection-diffusion equation, Burgers' equation and Stokes equations the algorithms always reached the specified stopping criteria and did not exceed the maximum alloted iterations. Additionally the increase in computational effort required was shown to be limited. However, it was shown that the Newton procedure for the VGM could not be used in cases where the local VGM residuals had varying signs. In that case the BFGS and least-squares VGM were successful. Application of the proposed procedures to different SGS models showed that nonlinear models tended to outperform linear models in terms of the L_2 and projected error. It was also shown that the parametrization of an SGS model can greatly influence how well the VGM will be able to optimize its coefficients.

Considering the error between the exact solution of a PDE and its numerical solution, Oberai and Sondak [5] showed that the error could be broken down into three sources. Error due to the used FEM basis, error due to the approximation of the unresolved-scales and an error due to the mismatch between optimal SGS model parameters and those found when using the VGM. The error due to the approximation of the unresolved scales was shown to be small for most cases. Even for Burgers' equation where the unresolved-scale Green's function theoretically does not exist; the error due to the unresolved-scales approximation was limited.

In analyzing the VGM residuals it was demonstrated that the parametrization of the SGS model determined how close the SGS model coefficients obtained from the VGM would be to the optimal coefficients. It was identified that to minimize the projected error by using the VGM an SGS Model is needed that can simultaneously globally minimize the VGM residuals and has scale-invariant optimal coefficients. The VGM residuals can be used to determine, a-priori, whether a particular SGS model meets these requirements. Furthermore, it was identified that in the absence of concurrency or scale-invariance the coefficients obtained from the VGM will most likely not minimize the projected error. In the case that either of the properties is absent, seeking the other is not required to still obtain improved performance when using the VGM, compared to using standard SGS model coefficients. However determining when a specific non-concurrent, or scale-variant SGS model would lead to improved performance remained difficult.

In conclusion, the VGM is a general method for the optimization of SGS models and in

many cases can lead to enhanced numerical procedures. The numerical methods proposed in this thesis for solving the VGM relations allow the optimization of a larger range of SGS models than before, which can lead to further improvements. Additionally it was demonstrated that the VGM residuals can be used to analyze SGS models that are likely to perform well. However, a full exploitation of this information will require methods that can globally minimize the VGM residuals, i.e. obtain $\frac{dR_{k,G}}{d\vec{c}} = 0$ as opposed to $\frac{\partial R_{k,G}}{\partial \vec{c}} = 0$, or problems where the difference between the two is not significant.

8.1 Recommendations

One issue, highlighted in particular by this thesis, was how the VGM may not always lead to optimal SGS model coefficients. In those cases, to minimize the projected error, it was argued that information about the adjoint equations would be necessary. Developing a VGM procedure that can take into account the adjoint information may then produce better coefficient values than those of the standard VGM. As such an interesting avenue for further research would be to investigate how to derive the adjoint formulation of the VGM relations and how to use them to obtain enhanced SGS model coefficient values. Taking into account the adjoint formulation is likely to lead to an increase in the computation time of the procedure. As such developing low-cost, or approximate methods to deal with the adjoint problem could also lead to promising developments.

Further work could be dedicated to extending the current method to the Navier-Stokes equations. However, in literature the VGM has not yet been applied to the Navier-Stokes equations with residual-based SGS models as used in this thesis. As such for the purpose of verification, the investigation could first be attempted by applying the procedures proposed in this thesis with an eddy-viscosity model for which an abundance of reference results are available. Once the procedure is confirmed to be working, the residual-based approach can be considered.

Finally, all numerical results in this thesis were computed using linear FEM basis functions. The method could be extended to higher-order basis functions, of unusual discretizations such as the partition of unity method, see e.g. Aquino [20]. However this would require an investigation of how $\boldsymbol{\tau}$ should be parameterized with increasing discretization order.

Bibliography

[1] T. J. R. Hughes, "Multiscale phenomena: Green's functions, the dirichlet-to-neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods," *Computer methods in applied mechanics and engineering*, vol. 127, pp. 387–401, 1995.

[2] T. J. R. Hughes, G. R. Feijóo, L. Mazzei, and J. Quincy, "The variational multiscale method - a paradigm for computational mechanics," *Computer methods in applied mechanics and engineering*, vol. 166, pp. 3–24, 1998.

[3] A. A. Oberai and J. Wanderer, "A dynamic approach for evaluating parameters in a numerical method," *International Journal for Numerical Methods in Engineering*, vol. 62, pp. 50–71, 2005.

[4] A. A. Oberai and Z. Wang, "Optimal numerical solutions of PDEs using the variational Germano identity," *Computer methods in applied mechanics and engineering*, vol. 197, pp. 2948–2962, 2008.

[5] A. A. Oberai and D. Sondak, "Application of the variational Germano identity to the variational multiscale formulation," *Communications in numerical methods in engineering*, vol. 27, pp. 335–344, 2011.

[6] I. Akkerman, K. G. van der Zee, and S. J. Hulshoff, "A variational Germano approach for stabilized finite element methods," *Computer methods in applied mechanics and engineering*, vol. 199, pp. 502–513, 2010.

[7] L. Chen, G. D. Maher, and S. J. Hulshoff, "Stabilisation parameter determination for the Stokes equations," *World congress of computational mechanics XI*, 2014.

[8] V. Gravemeier, *The variational multiscale method for laminar and turbulent incompressible flow*. PhD thesis, Institut für Baustatik der Universität Stuttgart, 2003.

[9] V. M. Calo, *Residual-based multiscale turbulence modeling: finite volume simulations of bypass transition*. PhD thesis, Stanford University, 2004.

[10] G. Scovazzi, *Multiscale methods in science and engineering*. PhD thesis, Stanford university, 2004.

[11] I. Akkerman, *Adaptive variational multiscale formulations using the discrete Germano approach*. PhD thesis, Delft University of Technology, 2009.

[12] T. J. R. Hughes and G. Sangalli, "Variational multiscale analysis: the fine-scale green's function, projection, optimization, localization, and stabilized methods," *SIAM Journal of numerical analysis*, vol. 45, pp. 539–557, 2007.

[13] F. Shakib, *Finite element analysis of the compressible Euler and Navier-Stokes equations*. PhD thesis, Stanford University, 1988.

[14] Y. Yuan, "A modified BFGS algorithm for unconstrained optimization," *IMA journal of Numerical Analysis*, vol. 11, pp. 325–332, 1991.

[15] P. Wolfe, "Convergence conditions for ascent methods," *SIAM review*, vol. 11, pp. 226–235, 1969.

[16] S. C. Brenner and L. R. Scott, *The Mathematical Theory of Finite Element Methods*. New York, USA: Springer, 2008.

[17] K. E. Jansen, C. Whiting, and G. M. Hulbert, "A generalized-alpha method for integrating the filtered Navier-Stokes equations with a stabilized finite element method," 2000.

[18] L. P. Franca and T. J. R. Hughes, "Convergence analyses of Galerkin least-squares methods for symmetric advective-diffusive forms of the Stokes and incompressible Navier-Stokes equations," *Computer methods in applied mechanics and engineering*, vol. 105, pp. 285–298, 1993.

[19] C. A. Taylor, T. J. R. Hughes, and C. K. Zarins, "Finite element modeling of blood flow in arteries," *Computer methods in applied mechanics and engineering*, vol. 158, pp. 155–196, 1998.

[20] W. Aquino, "Generalized finite element method using proper orthogonal decomposition," *International Journal for Numerical Methods in Engineering*, vol. 79, pp. 887–906, 2009.