Proper Orthogonal Decomposition Extensions and Their Applications in Steady Aerodynamics

by

Bui Thanh Tan

B.Eng. in Aeronautics
Ho Chi Minh City University of Technology, Vietnam, 2001

SUBMITTED TO THE SMA OFFICE IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF ENGINEERING
IN HIGH PERFORMANCE COMPUTATION FOR ENGINEERED SYSTEMS (HPCES)
AT THE
SINGAPORE-MIT ALLIANCE

2003

Signature of author:...
HPCES Programme

Certified by:...
Assoc. Prof. Murali Damodaran
SMA Fellow, NTU
Thesis Advisor

Certified by:...
Assist. Prof. Karen Willcox
SMA Fellow, MIT
Thesis Advisor

Accepted by:...
Assoc. Prof. Khoo Boo Cheong
Programme Co-Chair
HPCES Programme

Accepted by:...
Prof. Jaime Peraire
Programme Co-Chair
HPCES Programme

Proper Orthogonal Decomposition Extensions and Their Applications in Steady Aerodynamics

by

Bui Thanh Tan

In Partial Fulfillment of the Requirements for the
Degree of Master of Engineering in High Performance
Computation for Engineered Systems (HPCES)

ABSTRACT

The development and application of model reduction techniques have been widely investigated for problems in unsteady aerodynamic systems, driven mainly by the need to develop effective economical computational models capturing as much flow physics as possible. These techniques which have been based on a number of approaches such as balanced truncation, Arnoldi, proper orthogonal decomposition etc are designed to obtain a reduced-order model for both linear and nonlinear systems. However there has been a paucity in the application of these techniques in the area of steady aerodynamics. The work reported in this thesis demonstrates how the proper orthogonal decomposition (POD) technique can be extended to a number of applications in steady aerodynamics.

In the first instance the POD approach is coupled with a cubic spline interpolation procedure to develop reliable fast, low-order models for accurately predicting steady aerodynamic flowfields for arbitrary values or variation in flow parameters such as the angle of attack or inflow Mach number. Results on the prediction of steady transonic aerodynamic flowfield solution past an airfoil at arbitrary values of angle of attack or Mach number show that accurate flowfield predictions can be obtained, including cases that were not sampled in the ensemble of snapshots.

The second extension concerns a "gappy" POD technique for the reconstruction of incomplete or inaccurate aerodynamic flowfield data. The first case corresponds to the complete reconstruction of pressure field around an airfoil from the knowledge of pressure data defined only at the airfoil surfaces. The second case corresponds to the one in which POD snapshots are reconstructed from an incomplete set of aerodynamic data. Gappy POD is shown to be an effective technique for reconstruction of complete aerodynamic flow field data from limited measurements or incomplete data. This approach demonstrates an effective way in which experimental and computational aerodynamic data can be combined to predicate accurate aerodynamic flow fields cheaply.

Finally it is shown how an extension of the gappy POD can be used to formulate and cast inverse airfoil shape design and flowfield prediction for an arbitrary airfoil problems, in which steady aerodynamic plays a role, as gappy data problems. The extension of the methodology to constrained airfoil shape design problem is also discussed. The gappy design methodology has been shown to be a simple, effective and an efficient airfoil shape design technique.

Thesis Advisors:

1. Assoc. Prof. Murali Damodaran, SMA Fellow, NTU
2. Assist. Prof. Karen Willcox, SMA Fellow, MIT

ACKNOWLEDGMENT

It is a great pleasure to thank my first advisor, Prof. Damodaran Murali, for training me in computational fluid dynamics and my second advisor, Prof. Karen Willcox, for sharing with me her deep knowledge in proper orthogonal decomposition.

I would also like to thank Prof. Nguyen Thien Tong and Prof. Nhan Phan Thien for encouraging me to study under the Singapore-MIT Alliance (SMA). Without their guidance, I would not have accomplished this work.

I am very pleased to thank all of my friends from Vietnam, Malaysia, China, India and Singapore from the third batch of High Performance Computation for Engineered Systems (SMA-HPCES 2001) for their help and encouragement which made my life in SMA so interesting that it will remain etched in my memory forever.

This work has been supported through a SMA scholarship and I would like to thank the Singapore Government for creating this scholarship for students around the world. I am so grateful to all Professors in the SMA-HPCES program for teaching me many useful subjects for my research work.

Finally, I would like to thank my family for their continual support and encouragement during my research.

DEDICATION

To my mom

Hà Thị Mận

my wife

P.T. Công Minh

and my brother

Hô` Bão Hân

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

The development of reduced-order models is primarily motivated by the desire to have detailed knowledge of the physics of the problem being investigated at the expense of fast, cheap and reliable prediction tools. Efforts at deriving reduced-order modelling techniques have been successful in many areas, including thermal chemical processing in Newman [1], steady analysis and design of inviscid airfoils in LeGresley and Alonso [2], and derivation of reduced-order dynamical models as in Holmes et al. [3].

Antoulas et al. [4] have classified model reduction methods into two broad categories, namely methods based on the singular value decomposition (SVD) and methods based on moment matching based methods. The SVD based methods consist of Hankel-norm approximation, balanced truncation, singular perturbation, cross grammian and proper orthogonal decomposition (POD) while the moment matching based methods are the families of Lanczos and Arnoldi procedures. In the first category, POD turns out to be a popular method for nonlinear systems. In the second category, one of the approaches for nonlinear systems is to use the Arnoldi type projection based on quadratic or even higher degree approximation as in Chen [5]. However most of these attempts are designed either for some specific types of nonlinear systems or have not been practically applied to general nonlinear model order reduction problems. In view of this then it may be safe to say that for highly nonlinear systems in aerodynamics governed by Euler or Navier-Stokes equations, the POD could be an appropriate choice for model order reduction.

The computation of non-linear unsteady aerodynamic flows by numerically solving the Euler and Navier-Stokes equations is a computationally intensive task even when implemented on modern state-of-the art computing platforms. The development of reduced-order models provides a more efficient as well as a computationally economic way for investigating these problems. So far, most of the current efforts in model reduction have been focused mainly on unsteady aerodynamics. However it is also possible to use model-order reduction methods for a number of application problems in steady aerodynamics as well.

This thesis will address a number of applications of model-order reduction in steady compressible flow. The first application has implications for the efficient computation of steady aerodynamic flow fields for variation in flow parameters such as angle of attack and flow Mach number. Although Euler and Navier-Stokes flow solvers based on computational fluid dynamics technology turn out to be useful tools for providing accurate numerical result, they are generally of a high order and generally expensive and hence may not be very efficient for certain cases. The second application considered in this work deals with the reconstruction of incomplete aerodynamic data obtained experimentally or damaged data (due to technical or natural circumstances some data is marred). Finally model-order reduction via gappy POD technique is shown to be a simple, fast and an efficient way for modelling inverse and constrained airfoil shape design optimization problems which are normally very expensive computational tasks in which the flow solver is deployed during each design iteration to evaluate the objective function which is being minimized during the optimization process. Model order reduction techniques in airfoil shape design can be used to overcome this computationally expensive part of the design task. As this research work aims to demonstrate the benefits of model-order reduction for these problems in steady compressible aerodynamics, steady aerodynamic flow fields around an airfoil are considered to illustrate all these three applications. Euler equations of fluid motion are solved numerically to model the compressible flowfields around an airfoil and to generate the ensemble of aerodynamic flowfield snapshots which forms the basis for the application of POD method. POD is selected as the model-order reduction technique of choice for this study in view of the nonlinear nature of the Euler equations.

The proper orthogonal decomposition, also known as Karhunen Loève expansion and principle components analysis, has been widely used for a wide range of applications. POD analysis yields a set of empirical modes, which describes the dominant behavior or dynamics of given problem. Sirovich [6] introduced the method of snapshots as a way for efficiently determining the POD basis vectors (modes) for large problems. This technique can be used for a variety of applications, including derivation of reduced-order dynamical models for fluid dynamic applications as in Holmes et al. [3]. In particular, the method of snapshots has been widely applied in conjunction with computational fluid dynamic (CFD) formulations to obtain reduced-order models for unsteady aerodynamic applications as in Dowell et al. [7], Hall et al. [8], Romanowski [9] and Beran and Silva [10]. A set of instantaneous flow solutions (snapshots) is obtained using the CFD method. The POD process then computes a set of basis functions from these snapshots, which is optimal in the sense that, for any given basis size, the error between the original and reconstructed data is minimized. Reduced-order models can be derived by projecting the CFD model onto the reduced space spanned by the POD modes. The original CFD model is usually governed by the nonlinear partial differential equations (PDEs), i.e. Navier-Stokes or Euler equations, which are very computationally expensive especially for a flow past a complex configuration such as an airplane. Moreover, the model is computationally intensive and hence unsuitable if coupling of the CFD model with another disciplinary model is required, for example real-time feedback control scheme as in Lassaux [11]. The reduced-order model, however, is just a small system of ordinary differential equations (ODEs) which can be solved efficiently. In view of this, the application of POD method for obtaining a model of reduced complexity has been extended to many other applications. Typical examples are a number of heat transfer problems in Newman [1], steady analysis and design of inviscid airfoils in LeGresley and Alonso [2], and derivation of reduced-order dynamical models as in Holmes et al. [3].

As a common practice in dealing with nonlinear PDEs, the linearized approximate equations are studied under some assumptions such as small disturbances of the inflow parameters as in Hall et al. [8]. Although these linearized models are significantly simple compared to the original nonlinear models, they usually have a prohibitively high number of states and hence they are still very expensive to simulate. Recently, Hall et al. [8], Kim [12] and Willcox et al.[13] have developed efficient POD methods for linearized systems. For the case of problems exhibiting spatial symmetry, instead of obtaining the snapshots from a time-dependent simulation which is computational expensive, the linearized system in time domain can first be transformed to frequency domain and then the POD snapshots are computed efficiently and accurately over the frequency range of interest. However the major restriction of the linearized model is the assumption of small perturbation unsteady flow and hence it is not suitable for the cases in which the nonlinear effect may be important and perturbations are large.

While the use of POD to capture the time variation of fluid dynamic problems has been widespread, the development of reduced-order models to capture parametric variation is less commonly available in technical literature. Epureanu et al. [14] has used POD to develop reduced-order models for turbomachinery flows with sampling in both time and over a range of interblade phase angles. The resulting reduced models were then applied to flows at varying Mach numbers, although the snapshot ensemble is computed at a single Mach number condition. Accurate results were obtained for Mach numbers close to that used in the snapshots. In another example that addresses parametric variation, Ly and Tran [15] have developed a fast computation that uses a POD basis to predict the steady-state temperature distribution of flow in a square cavity as the Rayleigh number is varied and in this case each snapshot is the steady temperature distribution corresponding with each value of Rayleigh number. This method is a simple combination of the POD basis and an interpolation procedure but shown to be effective and accurate.

In the context of the characterization of human faces, Sirovich and Kirby [16] have employed the POD procedure as a useful tool for human face compression. In this application, each distinguishable digital picture of a human face (snapshot) is mapped to be a rectangular matrix whose elements are integers in the interval $[0, 255]$. The POD procedure is then performed to obtain the POD modes. For any given face which is not a member of ensemble is projected onto the reduced space spanned by the dominant POD basis vectors to calculate the POD coefficients. This small number of POD coefficients is used to represent the face with a small error instead of a big matrix. The results show that roughly 100 : 1 compression ratio (ratio between the number of elements of a matrix representing a face and the number of POD coefficients corresponding with that face) is obtained. A further question in the characterization of human faces is: How much damage, by deletion of pixels, can such an image suffer and still be recovered accurately? Such a question arises due to technical or natural circumstances in which the available data is incomplete or partial. Everson and Sirovich [17] have presented a variation of the basic POD method that handles incomplete or “gappy” data sets. This method actually makes use of the conventional POD method for solving an unconstrained optimization problem which approximately reconstructs the full data from the incomplete one. Given a set of POD modes, an incomplete data vector can be reconstructed accurately by solving a small linear system of equations derived from the optimal conditions. Moreover, if the snapshots themselves are damaged or incomplete, an iterative method can be used to derive the POD basis. The POD basis is in turn used to reconstruct the incomplete data. This method has been successfully applied for reconstruction of images, such as human faces, from partial data.

In the field of airfoil design, Lighthill [18] was the pioneer in airfoil inverse design using the method of conformal mapping. This method was limited to only incompressible flow. The extension of this method to compressible flow was presented later in McFadden [19]. By introducing the finite-difference method to evaluate the sensitivity derivatives, Hicks and Henne [20] first attempted to solve the constrained aerodynamic optimization problem. Since then, gradient-based methods have been used for aerodynamic design. Recently, Jameson [21] has applied control theory in the context of airfoil shape design optimization using Euler and Navier-Stokes problems. In this method, although an adjoint equation needs to be solved to obtain the gradient, the total cost for computing the gradients is independent of the number of design variables and hence it is better than finite-difference methods. Since no reduction was employed in the conformal mapping or the gradient-based methods, the cost for the design task is still enormous. In order to reduce the cost in the design work, LeGresley and Alonso [2] have used the POD technique for both direct and inverse airfoil design problems. In the present work, a set of aerodynamic flowfield solutions corresponding to the airfoil profiles created by perturbing the airfoil shape design variables from the parametric representation of base airfoil shape is computed using the Euler solver. The POD basis is then computed and used to construct a reduced order model for Euler equations to compute a new, approximate solution for an arbitrary airfoil at significantly lower computational costs. In this manner, both direct and inverse airfoil design problems can be done efficiently using a gradient-based optimization procedure with the information from the reduced order model.

1.2 Objectives

Applications of POD technique and its extensions for steady aerodynamics problems considered in this study, to the best of the author's knowledge, have not been abundantly available in technical literature. Though a simple combination of the POD basis and interpolation procedure could serve as a straightforward method for estimating steady aerodynamic flow problems, it has not been applied for such cases yet. Therefore the first extension is to couple the conventional POD approach with a cubic spline interpolation in order to develop fast, low-order models that accurately capture the variation in parameters such as the angle of attack (AOA) or inflow Mach number in which the application of the conventional POD is not obvious. The results will show that the steady state transonic flow solution for an arbitrary of parameter, i.e. angle of attack or Mach number, can be obtained efficiently without running the CFD solver again though the ensemble of snapshots consists of the solutions from the CFD solver over a parameter range of interest.

The second extension is a "gappy" POD technique for the reconstruction of incomplete or inaccurate aerodynamic data. For the first time, the proper orthogonal decomposition for incomplete (gappy) data is applied successfully to a range of aerodynamic problems. Incomplete aerodynamic data may arise in several situations. Firstly, a limited set of data may be available from experimental measurements. The gappy POD provides a way to reconstruct full flowfield information, using a combination of the available experimental and supplemental computational data. As a typical example in this work, an incomplete flowfield whose values are only partially distributed on an airfoil will be reconstructed. From the results, the reconstruction of the flowfield is accurately obtained even if a large amount of incomplete data is considered. Secondly, certain data may be missing because it is not known. For example, one may have a set of snapshots which corresponds to a set of airfoil shapes and their respective flowfields. Given a new airfoil shape, the gappy POD provides a way to quickly estimate the corresponding flowfield.

Conversely, the gappy POD can be used as a new approach to solve the problem of inverse design: given a target pressure distribution, the optimal airfoil shape can be determined by appropriate interpolation of known designs. In order to support the gappy POD in inverse design problem, a key modification to the conventional way of constructing the ensemble of snapshots is introduced where the snapshots are augmented to contain both airfoil coordinates and pressure distribution. It will be shown that, this new extension, which allows one to cast an airfoil inverse design task as a gappy data problem, is simple, efficient, and is found to work well on a range of examples, including both subsonic and transonic cases. It is natural to extend the POD method for constrained airfoil design problem. Although further refinement is necessary, the POD-based method is a promising method to solve the constrained airfoil design problem.

1.3 Outline

Chapter 2 briefly reviews the numerical method for the flow solver based on finite volume formulation proposed by Jameson et al. [22] for the solution of inviscid compressible aerodynamic flow. This flow solver is to generate the ensemble of aerodynamic flowfield snapshots, for both subsonic and transonic flows, which forms the basis for the application of POD. Chapter 3 gives a brief summary of the theoretical background on proper orthogonal decomposition technique and its extensions to parametric application and data reconstruction. Chapter 4 presents a number of application problems in steady aerodynamics investigated in this work in which the theory elucidated in Chapter 3 is employed using the snapshots of computed aerodynamic flowfield as outlined in Chapter 2. The first set of applications consists of one and two-parameter dependent aerodynamic problems in which the angle of attack and Mach number are varying parameters and the task at hand is to predict a steady state solution for specific values of these parameters which are not used to generate the ensemble of flowfield snapshots. Additional problems such as estimation of the greatest lift coefficient for a given angle of attack range and tracking the angle of attack for a given flowfield around an airfoil (defined later) are also presented to demonstrate the robustness of the POD extensions for a range of applications. The second set application problems focuses on reconstruction of damaged aerodynamic flowfield data. The first case considered is the reconstruction of the complete flow pressure field around the airfoil from the knowledge of values of pressure defined at discrete points on the airfoil surface. The second case considered corresponds to the reconstruction of POD snapshots from an incomplete set of aerodynamic data. The final set of applications considered deals with the solution of inverse airfoil shape design optimization and flowfield prediction for an arbitrary airfoil problems within the framework of the gappy POD method. The extension of the methodology to a simple constrained airfoil shape design problem is also discussed. Chapter 5 presents the results and discussions of all the problems addressed in this thesis while Chapter 6 concludes this inquiry with recommendations for future work in this area.

Chapter 2

Compressible Aerodynamic Flow Simulation

2.1 Introduction

The application of reduced order modelling such as POD in compressible aerodynamics requires the generation of an ensemble of aerodynamic flowfield snapshots. Flowfield snapshots can be generated by way of computation or by experimental techniques. In this work the ensemble of aerodynamic flowfield snapshots has been generated numerically using a compressible flow solver. As this work is primarily concerned with POD applications in compressible airfoil aerodynamics, the flow model used for generating the snapshots corresponds to steady inviscid compressible flow past an airfoil governed by Euler equations. In this chapter, a well-known finite volume multi-stage time-stepping numerical scheme for solving the steady inviscid compressible flow proposed by Jameson et al. [22] (extension for unsteady compressible aerodynamics can be found in Damodaran [23]) is briefly outlined.

2.2 Flow Model for Inviscid Flow

The two dimensional inviscid flows are governed by the Euler equations as follows

$$\frac{\partial U}{\partial t} + \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} = 0 \tag{2.1}$$

where

$$\begin{aligned} &U = (\rho, \rho u, \rho v, \rho E)^T, \\ &f = (\rho u, \rho u^2 + p, \rho u v, \rho u H)^T, \\ &g = (\rho v, \rho u v, \rho v^2 + p, \rho v H)^T \end{aligned} \tag{2.2}$$

where p, ρ, u, v, E, H denote the pressure, density, Cartesian velocity components, total energy and total enthalpy respectively. For a perfect gas

$$E = \frac{p}{(\gamma - 1)\rho} + \frac{1}{2}(u^2 + v^2),\ H = E + \frac{p}{\rho} \tag{2.3}$$

where γ is the ratio of specific heats of air.

2.3 Finite Volume Formulation

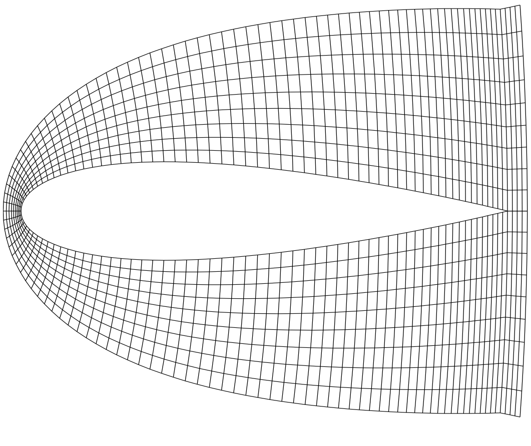


Figure 2.1: Structured C-grid for finite volume scheme

A C-grid is generated around the airfoil to discretize the flow domain into quadrilateral cells to implement the numerical algorithm. A structured C-grid in the vicinity of the airfoil is shown in Figure 2.1. The upstream inflow boundary is placed at about 100 chords from the leading edge of the airfoil while the downstream boundary is about 20 chords from the trailing edge of the airfoil.

The computational domain is divided into quadrilateral cells whose cell centers are denoted by the subscripts i, j as in Figure 2.2. The values of the dependent variables are assumed to be uniform in the cell and stored at the center of the cell. Application of the integral of conservation laws (2.1) i.e.

$$\frac{\partial}{\partial t}\int\int_D U\, dxdy + \oint_{\partial D}(f dy - g dx) = 0 \tag{2.4}$$

leads to a system of ordinary differential equations (ODEs)

$$\frac{d}{dt}(h_{i,j}\, U_{i,j}) + Q_{i,j} = 0 \tag{2.5}$$

where $h_{i,j}$ is the cell area, and $Q_{i,j}$ is the net flux of the cell which is given by

$$\sum_{k=1}^{4}(\Delta y_k f_k - \Delta x_k g_k) \tag{2.6}$$

where Δx_k and Δy_k are the increments of x and y along the edge with appropriate signs, f_k and g_k are the values of flux vectors f and g on the k^{th} edge and the sum is over the four edges of the cell. Each element of the flux vectors is calculated as the average of the values in the cells on either side of each edge. For example

$$f_4 = \frac{1}{2}(f_{i-1,j} + f_{i,j}) \tag{2.7}$$

The flux balance for the x-momentum is then

$$\frac{d}{dt}\{(\rho u)_{i,j}\, h_{i,j}\} + \sum_{k=1}^{4}\{Q_k(\rho u)_k + \Delta y_k p_k\} = 0 \tag{2.8}$$

The numerical scheme has second order accuracy spatially.

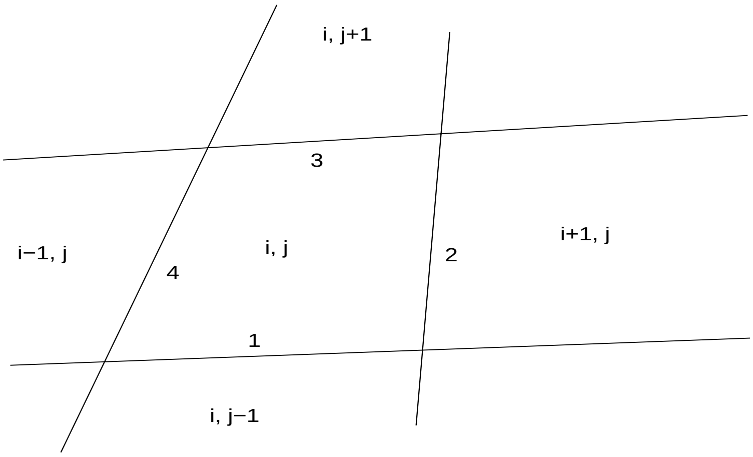


Figure 2.2: Discretization in space

In order to suppress the tendency for odd and even point decoupling and to limit undesirable overshoots near discontinuities such as shock waves, equation (2.5) has to be supplemented with artificial viscosity terms D_{ij} i.e.

$$\frac{d}{dt}(h_{i,j}\, U_{i,j}) + Q_{i,j} - D_{i,j} = 0 \tag{2.9}$$

The artificial viscosity terms are a blend of the second and fourth differences in flow variables. For example, the artificial viscosity terms for the density equation is

$$D_{i,j} = d_{i+1/2,j} - d_{i-1/2,j} + d_{i,j+1/2} - d_{i,j-1/2} \tag{2.10}$$

where the dissipative flux $d_{i+1/2,j}$ is defined by

$$d_{i+1/2,j} = \frac{h_{i+1/2,j}}{\Delta t}\left\{\varepsilon^{(2)}_{i+1/2,j}(\rho_{i+1,j} - \rho_{i,j}) - \varepsilon^{(4)}_{i+1/2,j}(\rho_{i+2,j} - 3\rho_{i+1,j} + 3\rho_{i,j} - \rho_{i-1,j})\right\} \tag{2.11}$$

where $\varepsilon^{(2)}$ and $\varepsilon^{(4)}$ are adaptive parameters. They are defined as

$$\varepsilon^{(2)}_{i+1/2,j} = k^{(2)} max(\upsilon_{i+1,j}, \upsilon_{i,j}) \tag{2.12}$$

and

$$\varepsilon^{(4)}_{i+1/2,j} = max(0, k^{(4)} - \varepsilon^{(2)}_{i+1/2,j}) \tag{2.13}$$

where

$$\upsilon_{i,j} = \frac{|p_{i+1,j} - 2p_{i,j} + p_{i-1,j}|}{|p_{i+1,j}| + 2|p_{i,j}| + |p_{i-1,j}|} \tag{2.14}$$

As pointed out in Jameson et al. [22], typical values of the constants $k^{(2)}$ and $k^{(4)}$ are

$$k^{(2)} = \frac{1}{4},\ \ k^{(4)} = \frac{1}{256} \tag{2.15}$$

Similarly, the dissipative terms for the momentum and energy equations are constructed by replacing ρ by ρu, ρv or ρH. Note that ρH is used rather than ρE in the energy equation because the difference equation will converge to the solution $H = H_\infty$ in the steady state.

2.4 Time Stepping Scheme

Equation (2.9) is advanced in time from a set of initial conditions using a multi-stage time stepping scheme as follows

$$\begin{aligned}
&U^{(0)} = U^{(n)} \\
&U^{(1)} = U^{(0)} - \tfrac{\Delta t}{2h} Q U^{(0)} + \tfrac{\Delta t}{2h} D U^{(0)} \\
&U^{(2)} = U^{(0)} - \tfrac{\Delta t}{2h} Q U^{(1)} + \tfrac{\Delta t}{2h} D U^{(0)} \\
&U^{(3)} = U^{(0)} - \tfrac{\Delta t}{h} Q U^{(2)} + \tfrac{\Delta t}{h} D U^{(0)} \\
&U^{(4)} = U^{(0)} - \tfrac{\Delta t}{6h} \left(Q U^{(0)} + 2 Q U^{(1)} + 2 Q U^{(2)} + Q U^{(3)} \right) + \tfrac{\Delta t}{h} D U^{(0)} \\
&U^{(n+1)} = U^{(4)}
\end{aligned} \tag{2.16}$$

The dissipative values are frozen at the values in the first stage for computational economy while preserving a second order temporal accuracy. The bound on the time step is $\Delta t \leq \lambda \Delta x$, where λ is the CFL number. The scheme above is stable for the CFL number of $2\sqrt{2}$ [22].

2.5 Convergence Acceleration Strategies

Local time steps are used in the code but it is not enough to speed up the convergence to a steady state. Some novel mechanisms to accelerate numerical solution to the steady state are discussed in this section. It will be shown that the steady state can be obtained within some hundred time steps by using these techniques together with local time-stepping.

2.5.1 Enthalpy Damping

It is well known that the total specific enthalpy is constant along each streamline in the steady state for Euler equations. This constant value is known from the inflow condition. Therefore one can add artificial terms to each equation that depend on the deviation of the local enthalpy from the steady state enthalpy. Note that these forcing terms are zero in the steady state and hence do not affect the steady state solution. These terms are chosen so as to speed up the convergence to a steady state. In particular, the Euler equations with enthalpy damping are given

$$\begin{array}{l} \frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \alpha\rho(H - H_\infty) = 0 \\ \frac{\partial(\rho u)}{\partial t} + \frac{\partial(\rho u^2+p)}{\partial x} + \frac{\partial(\rho uv)}{\partial y} + \alpha\rho u(H - H_\infty) = 0 \\ \frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho uv)}{\partial x} + \frac{\partial(\rho v^2+p)}{\partial y} + \alpha\rho v(H - H_\infty) = 0 \\ \frac{\partial(\rho E)}{\partial t} + \frac{\partial(\rho uH)}{\partial x} + \frac{\partial(\rho vH)}{\partial y} + \alpha(H - H_\infty) = 0 \end{array} \tag{2.17}$$

2.5.2 Implicit Residual Smoothing

The CFL number is one of the most important parameter in any numerical scheme in fluid dynamics applications. It is proportional to the time step and hence one way to speed up the convergence is to find a way to increase the allowable Courant number. This can be done by smoothing the residuals. At each stage of the time stepping scheme, one could replace the residual R_i by $\bar{R}_i$ by an implicit residual smoothing procedure given by

$$-\varepsilon\bar{R}_{i-1} + (1+2\varepsilon)\bar{R}_i - \varepsilon\bar{R}_{i+1} = R_i \tag{2.18}$$

where

$$\varepsilon = \frac{r}{(1-r)^2}, \quad r < 1 \tag{2.19}$$

For an infinite interval, equation (2.18) has the explicit solution

$$\bar{R}_i = \frac{1-r}{1+r} \sum_{q=-\infty}^{\infty} r^{|q|}\, R_{i+q} \tag{2.20}$$

2.5.3 Multigrid Strategies

Originally, the multigrid scheme is designed to accelerate the elliptic system to a steady state. However, it was shown to be possible to speed up the evolution of a hyperbolic system by using large time steps on coarse grids (because the same Courant number is used for all grids), so that the disturbances will pass through the outer boundary more rapidly. It is from the fact that the time step for a coarser mesh is larger than the one for a fine mesh in an explicit scheme, therefore a wave travels further for a given number of time steps. For the multistage scheme above, a simple multigrid scheme can be implemented by doubling the mesh spacing. Values of the flow variables U are transferred to a coarser grid by the rule

$$U_{2h}^{(0)} = \sum \frac{V_h\, U_h}{V_{2h}} \tag{2.21}$$

where the subscripts denote values of the mesh interval, V is the cell area and the sum is over the four cells on the fine grid constructing each cell of the coarse grid. A good point of this transformation is that it conserves mass, momentum and energy. Next a forcing function is defined as

$$P_{2h} = \sum R_h(U_h) - R_{2h}(U_{2h}^{(0)}) \tag{2.22}$$

where R is the residual of the difference scheme. The solution on a coarser grid is then constructed as

$$\begin{aligned} U_{2h}^{(1)} &= U_{2h}^{(0)} - \alpha_1 \Delta t[R_{2h}^{(0)} + P_{2h}], \\ &\vdots \\ U_{2h}^{(m+1)} &= U_{2h}^{(0)} - \alpha_m \Delta t[R_{2h}^{(m)} + P_{2h}] \end{aligned} \tag{2.23}$$

where α_m are constants, $R^{(m)}$ is the residual at the m^{th} stage. In the first stage of the scheme, P_{2h} cancels $R_{2h}^{(0)}$, therefore the evolution of the coarse grid is driven by the residuals on the fine grid. As a result, the final solution on the fine grid is independent of the choice of boundary conditions on the coarse grids. Finally, the correction calculated on each grid is passed back to the next finer grid by bilinear interpolation.

In the code, a simple saw-tooth cycle is used for multigrid strategy such that an intermediate Euler calculation is never needed for the correction back from each grid to the next finer grid. The details of these steps are outlined in Jameson [24].

2.6 Boundary Conditions

In any numerical scheme, one has to update the boundary conditions on the airfoil surface and at the far-field boundaries which simulate flow conditions far away from the airfoil after each iteration towards the solution. Since the normal component of the velocity on the airfoil surfaces is zero, only the pressure on the airfoil surface needs to be known in the finite volume scheme. This pressure can be estimated from the normal momentum equation. For subsonic flows, one needs to fix three characteristic variables based on flow conditions at infinity and extrapolate one characteristic variable from the interior of the computational domain at the inflow boundary and one characteristic variable is fixed using flow conditions at infinity and three characteristic variables are extrapolated from the interior of the computational domain at the outflow boundary. This is because there are three incoming characteristics at the inflow and one at the outflow. For supersonic flow all characteristic variables are fixed based on flow conditions at infinity at the inflow boundary while all characteristic variables are extrapolated from the interior of the computational domain at the outflow boundary. For nonlinear systems such as Euler equations, it is also possible to use one-dimensional Riemann variables based on normal and tangential velocity components at the far-field in place of characteristic variables and use two other parameters such as the measure of entropy and tangential velocity component at the far-field boundary as the four variables which are to be fixed or extrapolated. Therefore, at the inflow and outflow boundaries the measure of the entropy $S = p/\rho^{\gamma}$, the tangential velocity q_t, the Riemann invariant corresponding to the incoming waves $q_n - 2c/(\gamma - 1)$ and the Riemann invariant corresponding to the outgoing wave $q_n + 2c/(\gamma - 1)$, where q_n is the normal velocity and c is the speed of sound, can be used to specify inflow and outflow boundary conditions. For example, at the outflow boundary, the Riemann invariant corresponding to the incoming wave is fixed and the other variables are extrapolated. From these equations the values of the flow variables at all the boundaries can be determined.

2.7 Numerical Solution

Transonic flow past a NACA 0012 airfoil at Mach number of 0.8 and at angle of attack of 0.45 is computed for validating the code in transonic regime. The initial condition with $\rho = 1$ and $p = 1$ is used for all examples in this thesis. The results are output after 200 iterations. Figure 2.3 shows that the residual which here is the log of the RMS value of the density in the computational domain decreases rapidly to the order of about 10^{-5} after only 200 iterations. Figure 2.3 also shows the convergence of the number of supersonic points in the flowfield i.e. the supersonic pockets which form in the vicinity of the airfoil in transonic flow. The convergence of lift (C_L) and drag (C_D) coefficients is shown in Figure 2.4(a). Figure 2.5(a) shows the computed pressure contours. Figure 2.4(b) shows that the entropy increases on both upper and lower surfaces where strong shock waves form on both surfaces thereby confirming entropy rise across shock waves. The presence of the shock waves can also be seen very clearly in Figure 2.5(b), where C_p^* is the sonic line.

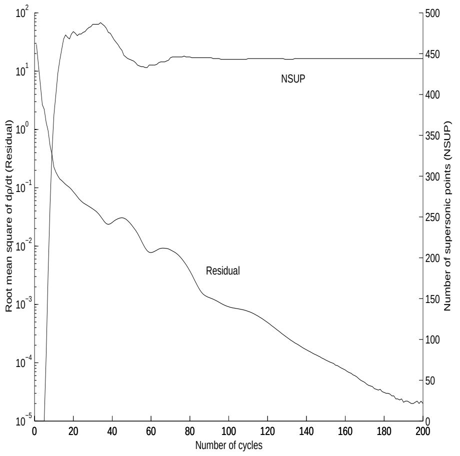
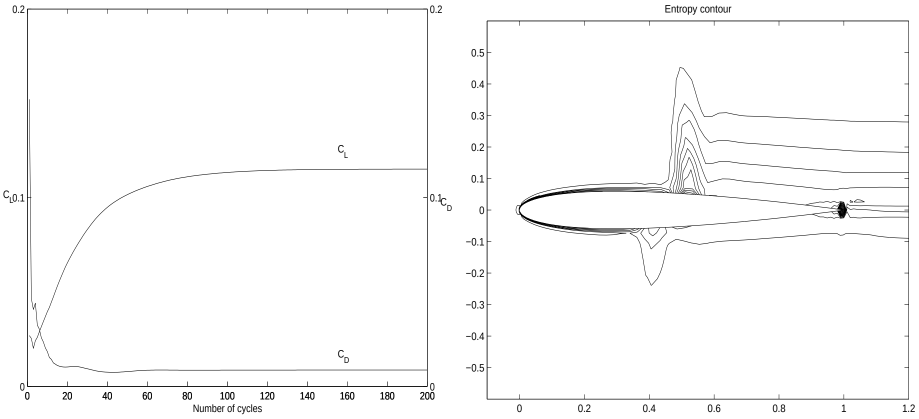


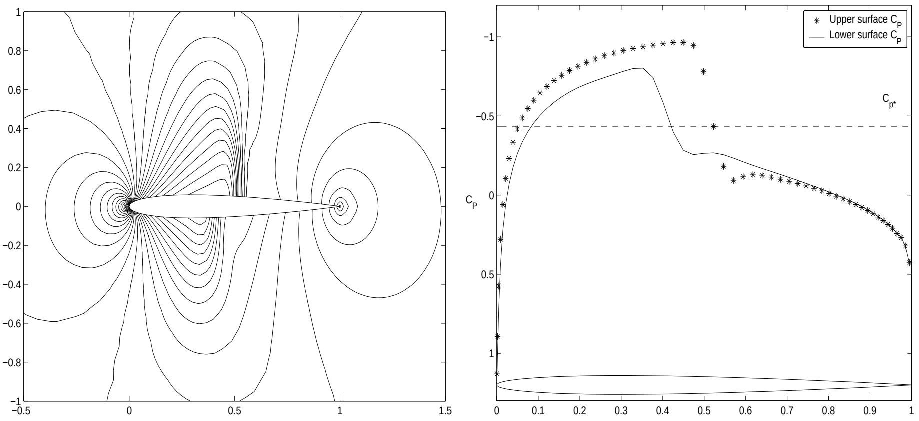
Figure 2.3: The convergence of the number of supersonic points and residual for the NACA 0012 with M = 0.8 and AOA = 0.45



(a) The convergence of lift and drag coefficients

(b) Entropy contours

Figure 2.4: The convergence of lift and drag coefficients and entropy contours for the NACA 0012 with M = 0.8 and AOA = 0.45



(a) Pressure contours

(b) Pressure coefficients

Figure 2.5: Pressure contours and pressure coefficients for the NACA 0012 with M = 0.8 and AOA = 0.45

Chapter 3

Proper Orthogonal Decomposition Extensions

3.1 Introduction

The well-known proper orthogonal decomposition (POD) technique which appears in different forms such as Karhunen-Loève decomposition (KLD), principal component analysis (PCA), and singular value decomposition (SVD) has been used in many physical applications. The connections and the equivalence of these methods have been discussed in Liang et al. [25]. In this chapter, the basic POD method will first be outlined, followed by a description of the extensions with interpolation and incomplete data sets. The origin of the POD procedure for fluid dynamic applications is mainly based on the study of Holmes et al. [3] which also provides insight into details and proofs. More details on POD from the second order stochastic process point of view can also be found in Wong [26], and Newman [1]. The treatment of reconstruction of missing data is a detailed version of the gappy POD method from Everson and Sirovich [17].

3.2 Proper Orthogonal Decomposition (POD)

The basic POD procedure is summarized briefly here. The optimal POD basis vectors Φ are chosen to maximize the cost:[3]

$$\max_{\Psi} \frac{\left\langle |(U,\Psi)|^2 \right\rangle}{(\Psi,\Psi)} = \frac{\left\langle |(U,\Phi)|^2 \right\rangle}{(\Phi,\Phi)} \tag{3.1}$$

where (U,Φ) is the inner product of the basis vector Φ with the field $U(x,t)$, x represents the spatial coordinates, t is time, and $<>$ is the time-averaging operation. It can be shown that the POD basis vectors are eigenfunctions of the kernel K given by

$$K(x,x') = \langle U(x,t), U^*(x',t) \rangle \tag{3.2}$$

where U^* denotes the hermitian of U. The method of snapshots, introduced by Sirovich [6], is a way of determining the modes Φ without explicitly calculating the kernel K. Consider an ensemble of instantaneous field solutions, or "snapshots". It can be shown that the eigenfunctions of K are linear combinations of the snapshots as follows

$$\Phi = \sum_{i=1}^{m} \beta_i U^i \tag{3.3}$$

where U^i is the solution at a time t_i and the number of snapshots, m, is large. For fluid dynamic applications, the vector U^i contains the flow unknowns at a given time at each point in the computational grid. The coefficients β_i can be shown to satisfy the eigen-problem

$$R\beta = \Lambda\beta \tag{3.4}$$

where R is known as the correlation matrix

$$R_{ik} = \frac{1}{m}\left(U^i, U^k\right) \tag{3.5}$$

The eigenvectors of R determine how to construct the POD basis vectors [using (3.3)], while the eigenvalues of R determine the importance of the basis vectors. These eigenvalues may be interpreted as "the mean energy of the flow field $U(x,t)$ projected on the Φ^i axis in the subspace spanned by POD basis vectors $\left\{\Phi^i\right\}_{i=1}^m$". The relative "energy" (measured by the 2-norm) captured by the i^{th} basis vector is then given by $\lambda_i / \sum_{j=1}^m \lambda_j$. The approximate prediction of the field U is then given by a linear combination of the eigenfunctions

$$U \approx \sum_{i=1}^{p} \alpha_i \Phi^i \tag{3.6}$$

where $p << m$ is chosen to capture the desired level of energy, Φ^i is the i^{th} POD basis vector, and the POD coefficients α_i must be determined as a function of time.

The basic POD procedure outlined above considers time-varying flows by taking a series of flow solutions at different instants in time. The procedure can also be applied in parameter space i.e. obtaining flow snapshots while allowing a parameter to vary as in Epureanu et al. [14]. The parameter of interest could, for example, be the flow freestream Mach number, airfoil angle of attack or airfoil shape.

Two properties of POD are outlined here. The first property states that the linear subspaces spanned by the POD basis vectors $\left\{\Phi^k\right\}_{k=1}^m$ and the snapshots $\left\{U^i\right\}_{i=1}^m$ coincide exactly [3]. This implies that, the POD basis functions can represent exactly the observed phenomena described by the snapshots. The second property states that the POD basis vectors are optimal in a sense that the first p POD basis vectors always capture more energy on average than p vectors of any other orthonormal basis [3]. In other words, let

$$U = \sum_{i=1}^{p} a_i \Phi^i$$

be the decomposition with respect to the POD basis $\left\{\Phi^i\right\}_{i=1}^{m}$ and let

$$U = \sum_{i=1}^{p} b_i \psi^i$$

be the decomposition with respect to any other arbitrary orthonormal basis $\left\{\psi^i\right\}_{i=1}^{m}$, then

$$\sum_{i=1}^{p} \left\langle \left|(U, \Phi^i)\right|^2 \right\rangle = \sum_{i=1}^{p} \lambda_i \geq \sum_{i=1}^{p} \left\langle \left|(U, \psi^i)\right|^2 \right\rangle \tag{3.7}$$

This property is exploited in this thesis to obtain a fast computational algorithm in many of the applications in compressible aerodynamics.

3.3 POD with Interpolation (PODI)

The basic POD procedure outlined in the previous section considered time-varying flows by taking a series of flow solutions at different instants in time. The procedure could also be applied in parameter space, that is, obtaining flow snapshots while allowing a parameter to vary. Assuming that the parameter of interest is denoted by δ which could be, for example, the flow freestream Mach number or airfoil angle of attack.

A procedure for rapid prediction of the flow solution U for any value of δ is as follows:

1. Let $\left\{U^{\delta_i}\right\}_{i=1}^{m}$ be the set of snapshots corresponding to the set of parameter values $\{\delta_i\}_{i=1}^{m}$.

2. Perform the basic POD procedure described above on $\left\{U^{\delta_i}\right\}_{i=1}^{m}$ to obtain the orthonormal POD basis $\left\{\Phi^k\right\}_{k=1}^{m}$.

3. The reconstruction of each snapshot is given by

$$U^{\delta_i} = \sum_{j=1}^{p} \alpha_j^{\delta_i} \Phi^j \tag{3.8}$$

where $p < m$ is the number of modes used in the reconstruction. The POD coefficients $\alpha_j^{\delta_i}$ are given by

$$\alpha_j^{\delta_i} = \left(\Phi^j, U^{\delta_i}\right) \tag{3.9}$$

4. If $\left\{\alpha_j^{\delta_i}\right\}_{i=1}^{m}$ is a smooth function of δ, interpolation can be used to determine the POD coefficients for intermediate values of δ that were not included in the original ensemble. The prediction of U^{δ} at any value of δ via the POD expansion is given by (3.6)

$$U^{\delta} = \sum_{j=1}^{p} \alpha_j^{\delta} \Phi^j \tag{3.10}$$

where the coefficients α_j^{δ} are found by cubic spline interpolation of the set $\left\{\alpha_j^{\delta_i}\right\}_{i=1}^{m}$. Note that no discussion of a smoothness requirement was given in Ly and Tran [15]; however, this is important for the interpolated result to be reliable.

3.4 POD for Reconstruction of Missing Data (Gappy POD)

In CFD applications, POD has predominantly been used for deriving reduced-order models via projection of the governing equations onto the reduced space spanned by the POD basis vectors. Here, a different application of the method is considered, which is based on the gappy POD procedure developed by Everson and Sirovich [17] for the reconstruction of human face images from incomplete data sets. In this thesis, the gappy POD methodology will be extended for consideration of fluid dynamic applications. The gappy POD procedure is first described.

The first step is to define a "mask" vector which describes for a particular flow vector where data is available and where data is missing. For example, for the flow solution U^k, the corresponding mask vector n^k is defined as follows:

$$\begin{aligned} n_i^k &= 0 \text{ if } U_i^k \text{ is missing or incorrect} \\ n_i^k &= 1 \text{ if } U_i^k \text{ is known} \end{aligned}$$

where U_i^k denotes the i^{th} element of the vector U^k. For convenience in formulation and programming, zero values are assigned to the elements of the vector U^k where the data is missing, and pointwise multiplication is defined as $\left(n^k, U^k\right)_i = n_i^k U_i^k$. Then the gappy inner product is defined as $(u,v)_n = ((n,u),(n,v))$, and the induced norm is $(\|v\|_n)^2 = (v,v)_n$.

Let $\left\{\Phi^i\right\}_{i=1}^m$ be the POD basis for the snapshot set $\left\{U^i\right\}_{i=1}^m$, where all snapshots are completely known. Let g be another solution vector that has some elements missing, with corresponding mask vector n. Assume that there is a need to reconstruct the full or "repaired" vector from the incomplete vector g. Assuming that the vector g represents a solution whose behavior can be characterized with the existing snapshot set, an expansion of the form (3.6) can be used to represent the intermediate repaired vector $\tilde{g}$ in terms of p POD basis functions as follows:

$$\tilde{g} \approx \sum_{i=1}^{p} b_i \Phi^i \tag{3.11}$$

To compute the POD coefficients b_i, the error, E, between the original and repaired vectors must be minimized. The error is defined as

$$E = \|g - \tilde{g}\|_n^2 \tag{3.12}$$

using the gappy norm so that only the original existing data elements in g are compared. The coefficients b_i which minimize the error E can be found by differentiating (3.12) with respect to each of the b_i in turn. This leads to the linear system of equations of the form

$$Mb = f \tag{3.13}$$

where $M_{ij} = \left(\Phi^i, \Phi^j\right)_n$ and $f_i = \left(g, \Phi^i\right)_n$. Solving equation (3.13) for b and using (3.11), the intermediate repaired vector $\tilde{g}$ can be obtained. Finally, the complete g is reconstructed by replacing the missing elements in g by the corresponding repaired elements in $\tilde{g}$, i.e. $g_i = \tilde{g}_i$ if $n_i = 0$.

3.5 POD with an Incomplete Snapshot Set (Iterative gappy POD)

The gappy POD procedure can be extended to the case where the snapshots themselves are not completely known. In this case, the POD basis can be constructed using an iterative procedure. Consider a collection of incomplete data $\left\{g^k\right\}_{k=1}^m$, with an associated set of masks $\left\{n^k\right\}_{k=1}^m$. The first step is to fill in the missing elements of the snapshots using average values as follows:

$$h_i^k(0) = \begin{cases} g_i^k & \text{if } n_i^k = 1 \\ \bar{g}_i & \text{if } n_i^k = 0 \end{cases} \tag{3.14}$$

where $\bar{g}_i = \frac{1}{P_i}\sum_{k=1}^m g_i^k$, $P_i = \sum_{k=1}^m n_i^k$ and $h^k(l)$ denotes the l^{th} iterative guess for the vector h^k. A set of POD basis vectors can now be computed for this snapshot set, and iteratively used to refine the guess for the incomplete data. The procedure can be summarized as follows, beginning with $l = 0$:

1. Use the basic POD procedure on the snapshot set $\left\{h^k(l)\right\}_{k=1}^m$ to obtain the POD basis vectors for the current iteration, $\left\{\Phi^k(l)\right\}_{k=1}^m$.

2. Use the first p of these POD basis vectors to repair each member of the snapshot ensemble, as described in the previous section. The intermediate repaired data for the current iteration is given by

$$\tilde{h}^k(l) = \sum_{i=1}^p b_i^k(l)\Phi^i(l) \tag{3.15}$$

3. The values from these intermediate repaired data are now used to recon-

struct the missing data for the next iteration as follows

$$h_i^k(l+1) = \begin{cases} h_i^k(l) & \text{if } n_i^k = 1 \\ \tilde{h}_i^k(l) & \text{if } n_i^k = 0 \end{cases} \tag{3.16}$$

4. Set $l = l + 1$ and go to step 1.

The above iterative procedure should be repeated until the maximum number of iterations is reached or until the algorithm has converged. When evaluating convergence, one can consider both the POD eigenvalues and the POD eigenvectors as will be demonstrated in the results in Chapter 5. It is important to know that in this procedure the POD modes are constructed at the same with the reconstruction of the incomplete snapshot set. Once the POD modes are found, the gappy POD method in section 3.4 can be used for reconstructing missing data in any new incomplete snapshot, which is not a member of the incomplete snapshot set.

Chapter 4

Application Problems in Steady Aerodynamics Using POD Extensions

4.1 Introduction

In this chapter, a set of application problems in steady aerodynamics is addressed to demonstrate how the POD extensions developed in chapter 3 are applied to the ensemble of aerodynamic flowfield snapshots generated using the compressible flow solver described in chapter 2. The first set of problems deals with one and two-parameter dependent aerodynamic problems in which the angle of attack and Mach number are the parameters which are allowed to vary and the goal is to predict a steady state solution with prescribed values of the parameters which are not used to generate the ensemble of flowfield snapshots. Additional problems such as estimation of the greatest lift coefficient for a given angle of attack range and tracking the angle of attack for a given flowfield around an airfoil are also presented in the first set to demonstrate the robustness of the POD extensions for a range of applications. The second set of problems concerns the problem of reconstruction of damaged data in aerodynamic flowfield data sets. The first example considered here is the case in which the complete flow pressure field is reconstructed from the knowledge of pressure values defined only on the airfoil surface. The second example concerns the reconstruction of POD snapshots from an incomplete set of aerodynamic data. The final set of problems concerns the airfoil shape design problem using the gappy POD method where it will be shown that the gappy POD offers a very simple and efficient approach for airfoil design problems. In this chapter these problems will be described in detail while the results and discussion pertaining to these problems will be presented in chapter 5.

4.2 Parametric Applications in Steady Transonic Aerodynamics

The first application of POD extension for parametric applications in transonic aerodynamics concerns the prediction of the flow field around the NACA 0012 airfoil at an arbitrary angle of attack (AOA) and Mach number (M). In this problem, M and AOA are the parameters which vary and the goal of this exercise is to predict the flow field (pressure field for example) at an arbitrary M or AOA which is not a member of ensemble of flowfield snapshots generated from the CFD solver over a range of M or AOA.

The second application concerns the determination of the angle of attack such that C_L, the lift coefficient of the airfoil, is maximized. It will be shown that this problem can also be solved efficiently by coupling PODI method and the well known Golden Section search method outlined in Krishnamurty [28].

The last example considered for parametric applications is the tracking problem which is to find a value of AOA such that the Mach number field distribution in the vicinity of the airfoil estimated from the POD expansion $V_p = \sum_{i=1}^{p} \alpha_i \Phi_V^i$ is close to the given Mach field V, where $\left\{\Phi_V^i\right\}_{i=1}^{m}$ is the POD basis for the Mach field V. The formulation is then to find an AOA which minimizes the cost functional

$$J = \|V - V_p\|_2^2 \tag{4.1}$$

Once again, PODI method coupled with the Golden Section Search method appears to be one of the suitable methods for solving this problem.

4.3 Reconstruction of Missing Aerodynamic Data

Incomplete or damaged aerodynamic data may occur in a variety of situations discussed in Chapter 1. For example, a limited set of data may be available from experimental measurements such as the pressure measurements at discrete pressure tapping locations on the surface of an airfoil. Stored aerodynamic flowfield data from CFD simulations may be damaged due to computer storage device failures and these data may be required for other subsequent flowfield simulations. POD and its extensions offer an attractive way for reconstructing the missing or lost data and this application is considered by way of a few illustrative examples. The goal of this application is to show that the missing (damaged) aerodynamic data can be reconstructed (or repaired) efficiently using the gappy POD method. The first example considered is the case in which the complete pressure field is reconstructed from pressure values known only on the airfoil surface. The second example considered is the one in which the POD snapshots are constructed from an incomplete set of aerodynamic data i.e. some elements of each flow field data are missing or lost.

4.4 Airfoil Design Optimization

POD and its extensions also offer a new approach for handling inverse airfoil shape design and constrained airfoil shape design optimization and the approach is outlined here briefly.

4.4.1 Inverse Design Using the Gappy POD Method

Gappy POD approach can be extended to handle the inverse airfoil shape design problem. In particular, a new variant of the method is proposed for the inverse design of a two-dimensional airfoil, i.e. given a target airfoil surface pressure distribution P^*, the inverse airfoil design problem corresponds to the determination of an optimal airfoil shape whose surface pressure distribution P minimizes the cost

$$J = \|P^* - P\|_2^2 \tag{4.2}$$

In order to solve this inverse airfoil shape design problem using the gappy POD method, the ensemble of flowfield snapshots is first redefined. Instead of having the ensemble simply containing only flow variables, each snapshot is augmented to also contain airfoil coordinates in view of the fact that the shape of the target airfoil sought is unknown and there must be a mechanism for POD to generate that unknown. The augmented snapshot set is defined as $\{V^i\}_{i=1}^m$

where each snapshot contains airfoil surface pressure distribution P^i and the corresponding set of airfoil coordinates C^i:

$$V^i = \begin{bmatrix} C^i \\ P^i \end{bmatrix} \tag{4.3}$$

The target vector $V^* = [C^{*T} \, P^{*T}]^T$ can then be considered as an incomplete data vector, where P^* is known and C^* is unknown and yet to be determined. Thus, the gappy POD procedure can be used to determine the optimal airfoil shape, using the procedure outlined in section 3.4 and minimizing the cost in (4.2) with respect to the gappy norm

$$J = \|V^* - \tilde{V}\|_n^2 \tag{4.4}$$

where n is the mask vector corresponding to V^* and the intermediate repaired vector $\tilde{V}$ is represented by a linear combination of basis vectors as in equation (3.11).

The inverse airfoil shape design problem has been transformed to a problem involving the reconstruction of missing data. In order to determine the airfoil shape, a system of linear equations (3.13) must be solved, with size equal to the number of POD basis functions. The gappy POD method will then produce not only the optimal airfoil shape, but also the corresponding surface pressure distribution. If additional flow field information is desired, such as pressure distribution off the surface or other flow variables, these data could also be included in the augmented snapshots set.

The POD eigenvalues provide guidance as on the number of POD modes that should be included in the basis. Typically, one will include p basis vectors so that the relative energy captured, i.e. $\sum_{i=1}^{p} \lambda_i / \sum_{j=1}^{m} \lambda_j$, is greater than some threshold, typically taken to be 99% or higher. This energy measure determines how accurately a snapshot in the original ensemble can be reconstructed using the POD basis; however, it does not provide any information regarding the accuracy of reconstructing a new vector. For the inverse airfoil shape design problem, it is therefore important to monitor the value of the cost function J. One may choose enough POD basis functions to capture 99% or more of the snapshot energy, but the optimal value of J remains unacceptably high. This indicates that the subspace spanned by the chosen snapshots is not sufficiently adequate enough to capture the desired design airfoil. Approaches for addressing this issue will be discussed in the chapter 5.

4.4.2 Constrained Airfoil Design Optimization

The gappy POD method, outlined in section 3.4 is a good method for solving the inverse airfoil shape design or unconstrained optimization problem using POD based method in which a system of linear equations is obtained by setting the first derivatives of the objective function to zero. However the gappy POD method cannot be used for a constrained airfoil design optimization problem in view of the imposed constraints which have to be taken into consideration for the optimization process. However, POD can still be used for solving the constrained design problem rapidly and this is outlined in below.

A simple constrained airfoil design optimization can be defined as follows.

$$\begin{aligned} &minimize \qquad C_D \\ &subject\ to: \quad C_L \geq C_{L_o} \end{aligned} \tag{4.5}$$

where C_D and C_L are the drag and lift coefficients respectively, C_{L_o} is the lift coefficient corresponding to the initial airfoil. The goal here is to alter the shape of the airfoil to reduce the drag yet still maintaining the lift coefficient. Many methods have been used to solve this problem. Here, a method which is developed based on a compact way of constructing the ensemble of aerodynamic flowfield snapshots discussed in section 4.4.1 is used to construct the ensemble defined as

$$V^i = \begin{bmatrix} C^i \\ P^i \end{bmatrix} \tag{4.6}$$

where C^i and P^i are the airfoil coordinates and pressure distribution, respectively. The POD procedure is then applied on this ensemble of snapshots to obtain the POD basis functions

$$\Phi^i = \begin{bmatrix} \Phi^i_C \\ \Phi^i_P \end{bmatrix} \tag{4.7}$$

where Φ^i_C and Φ^i_P are the coordinate and pressure parts of POD basis vector Φ^i. The vector containing the optimal airfoil and its corresponding pressure distribution of interest is a linear combination of POD basis vectors i.e.

$$\begin{bmatrix} C \\ P \end{bmatrix} = \sum_{i=1}^{p} a_i \begin{bmatrix} \Phi^i_C \\ \Phi^i_P \end{bmatrix} \tag{4.8}$$

where POD coefficients a_i are unknown and are treated as the design variables. It is important to note here that both airfoil coordinates and its corresponding pressure distribution are a function of the design variables a_i. Hence the statement of the constrained airfoil optimization (4.5) is equivalent to the following statement:

$$\begin{aligned} &\textit{minimize} \qquad C_D = f(a_1, \ldots, a_p) \\ &\textit{subject to}: \quad C_L = g(a_1, \ldots, a_p) \geq C_{L_o} \end{aligned} \tag{4.9}$$

The advantage here is that the design variables are now functioning as the POD coefficients and the constrained optimization problem (4.9) is transformed into a nonlinear convex programming problem which can be solved efficiently using any commercial software such as AMPL [29]. The most important aspect of this approach is that no further CFD simulations are required for executing the process of solving the optimization problem (4.9). Once the design variables are found, the airfoil and its corresponding pressure are obtained immediately using equation (4.8).

4.5 Prediction of Flowfield for an Arbitrary Airfoil

As a final application problem, the gappy POD method is used for predicting the aerodynamic flow field around an arbitrary airfoil. The mathematical model is exactly the same as the one in section 4.4.1 except that the available data now corresponds to coordinates C^* of a given airfoil shape while the pressure distribution P^* (or the grid coordinates G^* if the grid generation is of interest) is now treated as missing data. As a result, each snapshot consists of two parts, airfoil coordinates and pressure field or airfoil coordinates and grids. Once again, the other flow fields can be added to the snapshots so that the pressure field or grids and the other fields can be obtained at the same time.

Chapter 5

Results and Discussion

5.1 Introduction

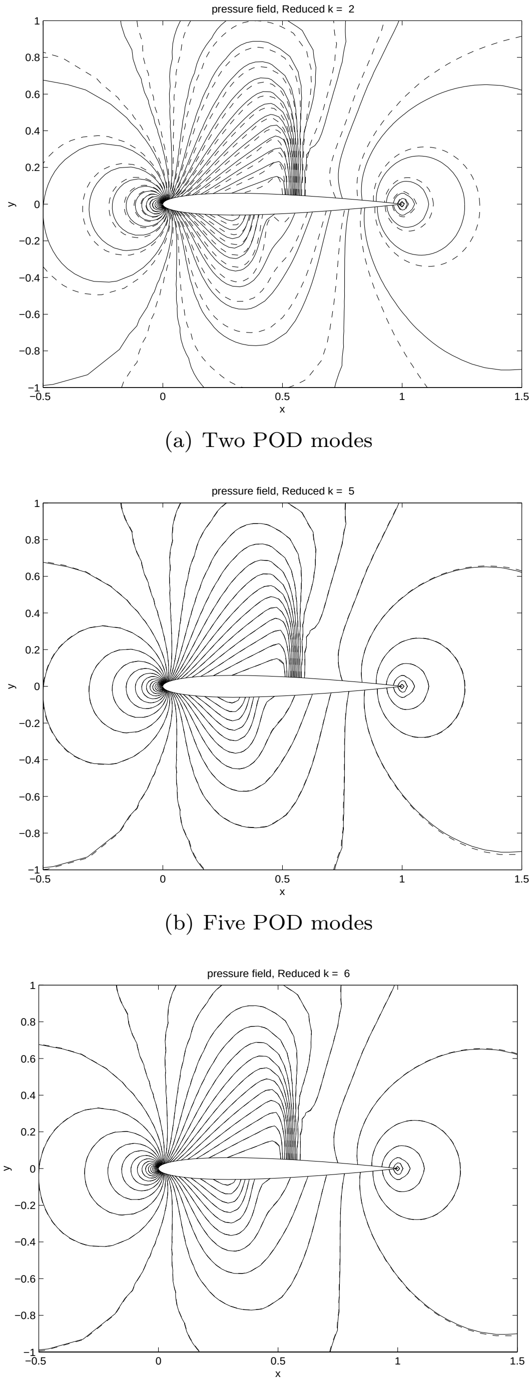
This chapter presents the results and discussion for all the problems described in chapter 4 to demonstrate the utility of POD and its extensions for various useful applications in steady aerodynamic flows. The first set of results concerns the prediction of a steady state solution in which the angle of attack and Mach number act as variable parameters. Next set of results corresponds to estimation of the greatest lift coefficient for a given angle of attack range and tracking the angle of attack for a given flowfield around an airfoil. This is followed by results dealing with examples of reconstruction of damaged aerodynamic data corresponding to the case in which the complete pressure field in the vicinity of the airfoil is reconstructed from pressure values defined only on the airfoil surface and the case in which the POD snapshots are reconstructed from an incomplete set of aerodynamic data. The final set of results pertains to applications of POD and its extensions to the airfoil design problem and prediction of flow field for an arbitrary airfoil.

5.2 Parametric Applications in Transonic Aerodynamics

In this section, results obtained using POD and its extensions are presented for a variety of cases to demonstrate the interpolation technique for parametric applications. The steady transonic flow past a NACA 0012 airfoil at a baseline freestream Mach number of 0.8 is considered for all examples considered here.

5.2.1 Flowfield Prediction at an Arbitrary Angle of Attack

The flow field chosen to demonstrate here is the pressure field, the application for the other flowfields is straightforward. The ensemble of aerodynamic flowfield snapshots contains 26 pressure field solutions (generated using the compressible flow solver outlined in Chapter 2) corresponding to 26 values of angle of attack in the interval $[0, 1.250]$ with angle step 0.05 degrees. The prediction of pressure field at angle of attack of $\alpha = 0.77$ degrees which is not a member of the ensemble of the flowfield snapshots is considered. Figures 5.1(a) to (c) compare the predicted pressure (shown in solid contour lines) and the exact pressure (shown in dotted contours) corresponding to the use of two, five and six POD modes respectively. Here the term “exact” refers to the solution computed using the flow solver by setting the angle of attack to 0.77. It can be clearly seen that by using six POD modes good agreement between the predicted pressure contours and the numerically computed exact pressure field could be reached. Figure 5.2 shows the convergence of the predicted lift coefficient to the exact one computed from the CFD solver when the number of POD modes increases. The convergent value can be obtained only with seven POD modes, showing



(a) Two POD modes

(b) Five POD modes

(c) Six POD modes

Figure 5.1: Comparison of predicted pressure contours (dash) and CFD pressure contours (solid) for a Mach number of 0.8 and angle of attack of 0.77°.

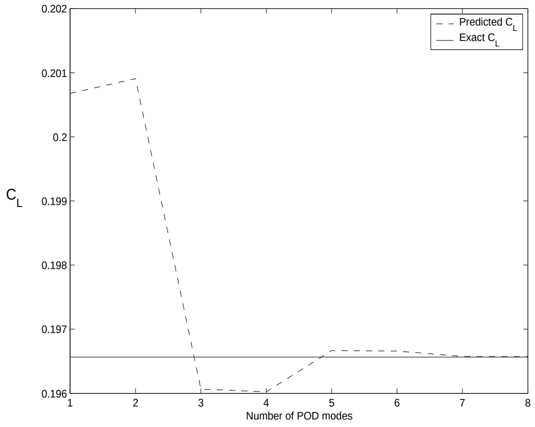


Figure 5.2: The convergence of lift coefficient versus the number of POD modes for a Mach number of 0.8 and angle of attack of 0.77°.

that in the case when the angle of attack is the only varying parameter, the PODI method can give accurate results with a small number of POD modes.

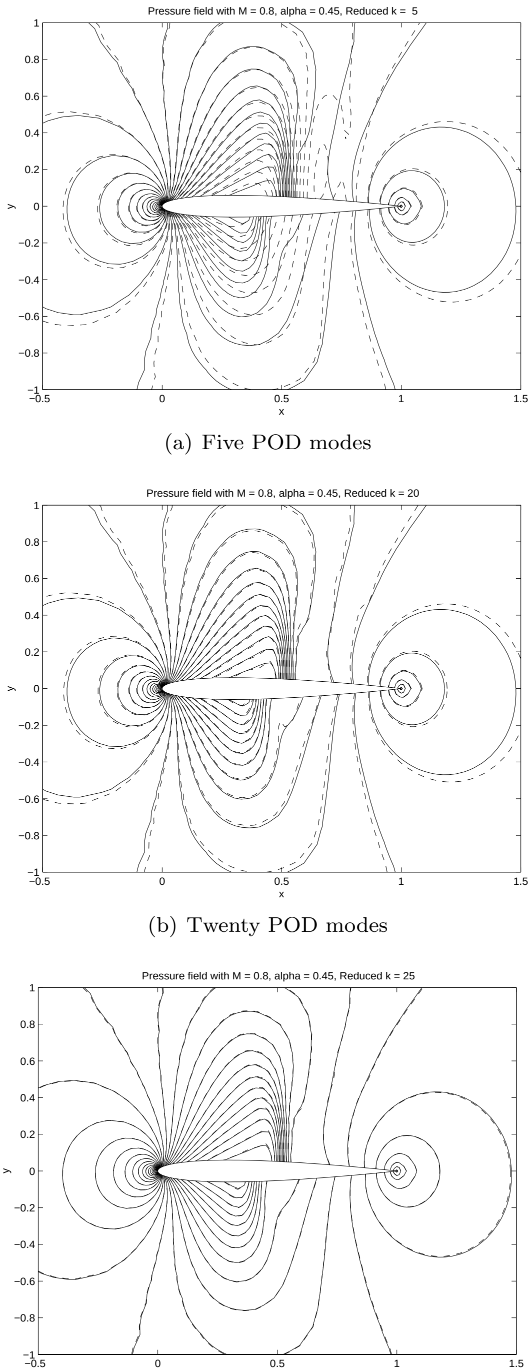
Flow field prediction for an arbitrary Mach number for steady transonic aerodynamic flow can also be done in the same manner. Results for this case, though not shown here, can be computed in a straightforward manner.

5.2.2 Flowfield Prediction for an Arbitrary Pair of Angle of Attack and Mach Number

The problem considered in this section corresponds to the prediction of steady flow when variations in both angle of attack and Mach number are considered. For this case, the Mach number interval [0.75, 0.85] is divided into 20 uniform intervals, and the angle of attack interval $[0°, 1.25°]$ is divided into 10 uniform intervals. Hence, the total number of snapshots in the ensemble is 231 which is the product of 21 values of Mach number and 11 values of angle of attack. Based on this ensemble of flowfield snapshots, interpolation is used to predict the flow at any arbitrary Mach number and angle of attack within the range considered. For demonstration, POD will be applied to the pressure field only; the procedure for the other flow fields is straightforward.

The first prediction considered is for the pair ($\alpha = 0.45$, $M = 0.8$), in which $M = 0.8$ is one of the values used to generate the snapshots while $\alpha = 0.45$ is not. Figures 5.3 compares the pressure field (shown by dashed contour lines) computed using POD with the exact or computed pressure field (in solid contour lines) corresponding to ($\alpha = 0.45, M = 0.8$). It can be seen from Figure 5.3(a), that using five POD modes results in large discrepancies in many regions of the flowfield when compared with the exact pressure contours corresponding to this case. However, as the number of POD modes is increased to 25, as in Figure 5.3(c), the predicted pressure contours match closely with the exact ones. It should be noted that the cost for pressure prediction with 20, as in Figure 5.3(b), or 25 modes does not differ greatly, since the method requires only interpolation of the scalar POD coefficients. The number of modes can therefore be increased to obtain the desired level of accuracy.

Figure 5.4 compares the distribution of surface pressure coefficients on the upper and lower surfaces of the airfoil corresponding to each of the predicted pressure field shown in Figure 5.3. One interesting point to note from the pressure contour plots here is that when a small number of POD modes are used, the pressure contours far away from the airfoil surface show large deviations from the exact contours while the pressure distribution on the airfoil surface appears to be in agreement with the exact one. This can be seen clearly by comparing Figure 5.3(b) and Figure 5.4(b). Therefore, beyond a certain point,

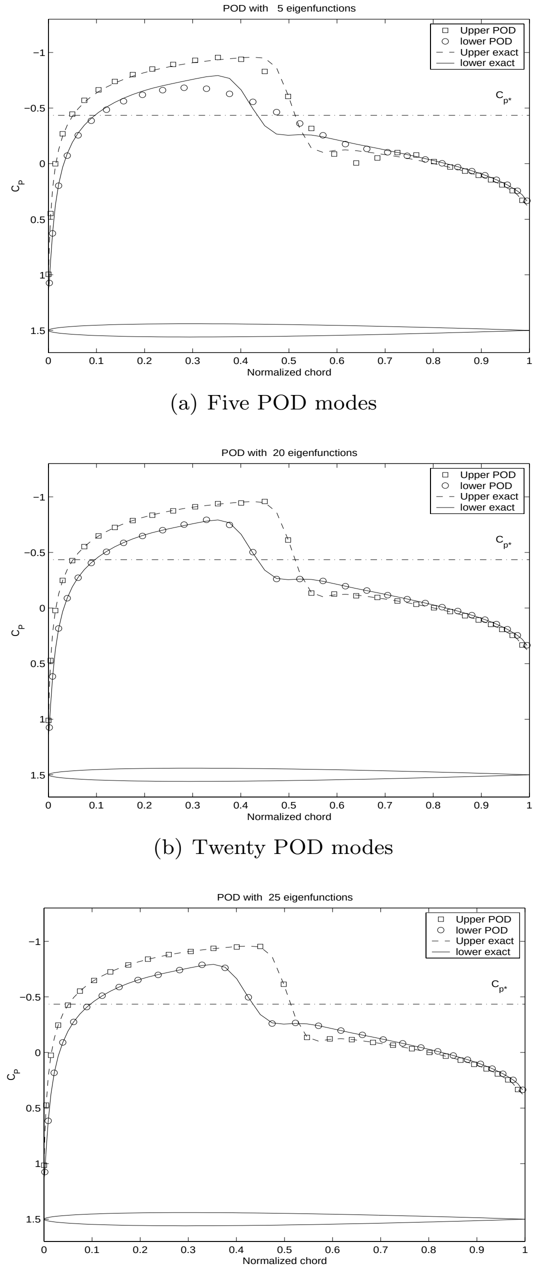


(a) Five POD modes

(b) Twenty POD modes

(c) Twenty five POD modes

Figure 5.3: Comparison of predicted pressure contours (dash) and exact pressure contours (solid) for a Mach number of 0.8 and angle of attack of 0.45°.



(a) Five POD modes

(b) Twenty POD modes

(c) Twenty five POD modes

Figure 5.4: Comparison of predicted upper pressure coefficients (square), predicted lower pressure coefficients (circle), exact upper pressure coefficients (dash) and exact lower pressure coefficients (solid) for M = 0.8 and AOA = 0.45°. C_{p^*} is the sonic pressure coefficient.

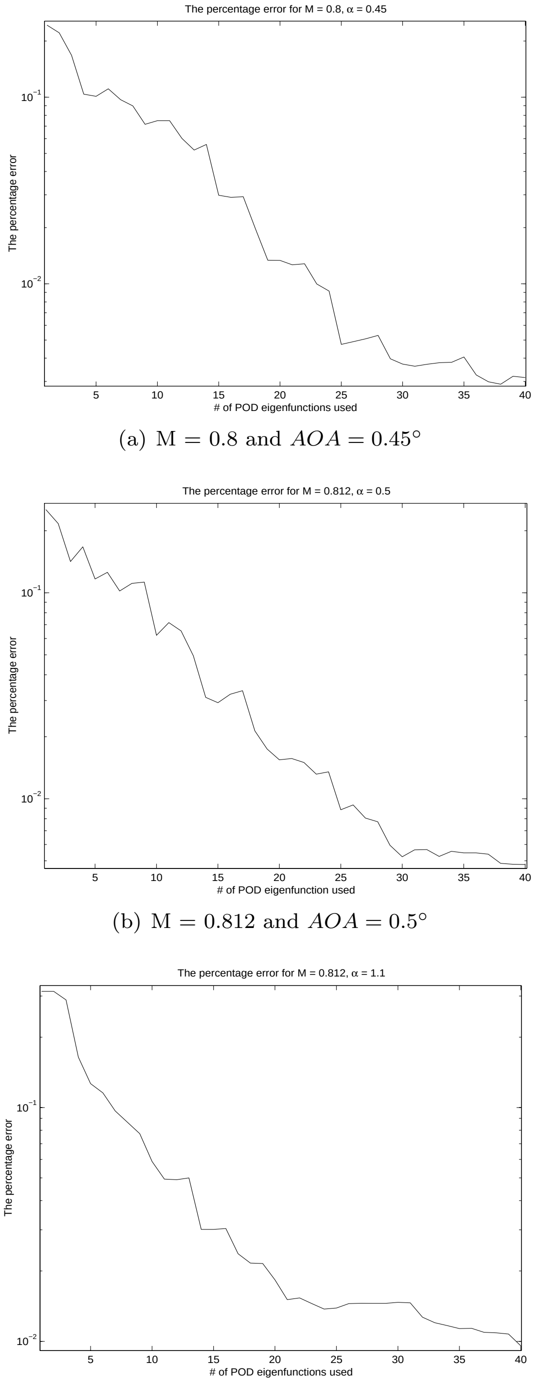
increasing the number of POD modes only contributes to the improvement in the agreement of the farfield predicted pressure fields with the exact one and does not result any major effect on the surface pressure distribution.

Assume that the discrepancy or error between the predicted and exact quantities is expressed as

$$e = max|(U - U_p)./U| = \|(U - U_p)./U\|_\infty \tag{5.1}$$

which is the the maximum percentage error between two corresponding elements of the CFD solution U and p^{th}-order reconstructed solution U_p and "./" means elementwise division. Figure 5.5(a) shows the log value of the maximum percentage error versus the number of POD modes used to predict the pressure distribution. It can be seen that the error decreases rapidly as the number of modes is increased from one to 25.

The next prediction considered is for the pair ($\alpha = 0.5$, $M = 0.812$), in which $\alpha = 0.5$ is one of the values used to generate the snapshots but $M = 0.812$ is not. Figure 5.6 compares the predicted pressure field contours (dashed lines) with the exact pressure field contours (solid lines) corresponding to ($\alpha = 0.5, M = 0.812$). Experimentation with POD modes by the author during the course of this work has shown that POD prediction is more sensitive for variations in Mach number than with variations in angle of attack. Hence it is expected that more POD modes are required to attain results comparable to exact solutions for this case. Figures 5.6(c) and 5.5(b) show that 30 modes are required to achieve a close agreement between the predicted flow field and the exact flow field.

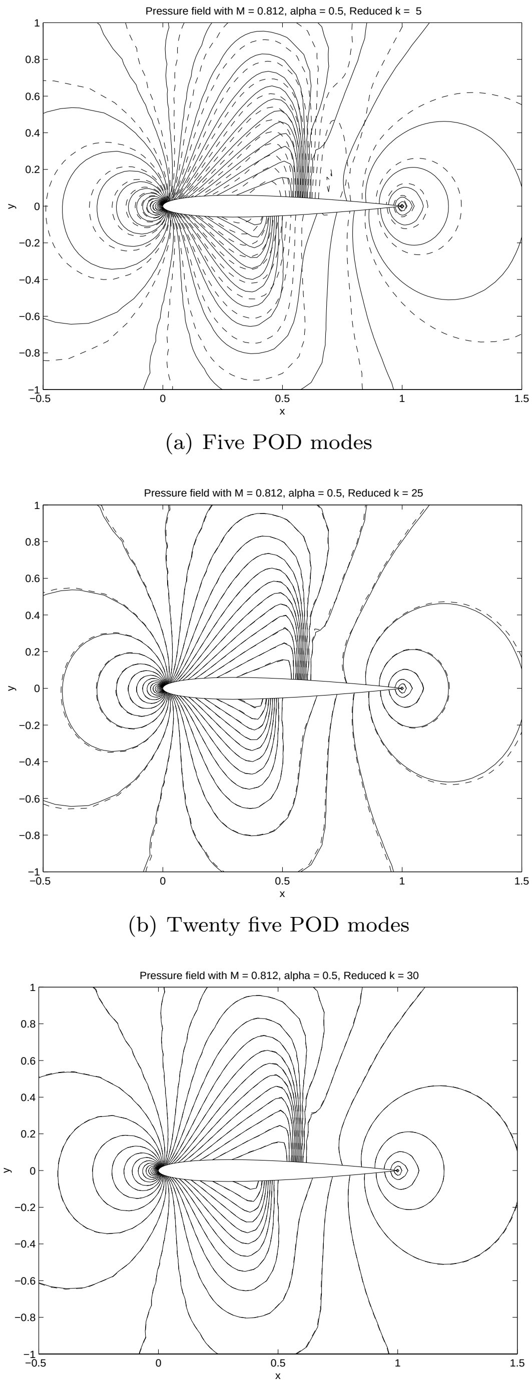


(a) M = 0.8 and $AOA = 0.45°$

(b) M = 0.812 and $AOA = 0.5°$

(c) M = 0.812 and $AOA = 1.1°$

Figure 5.5: Variation of percentage error versus the number of POD modes in log scale.

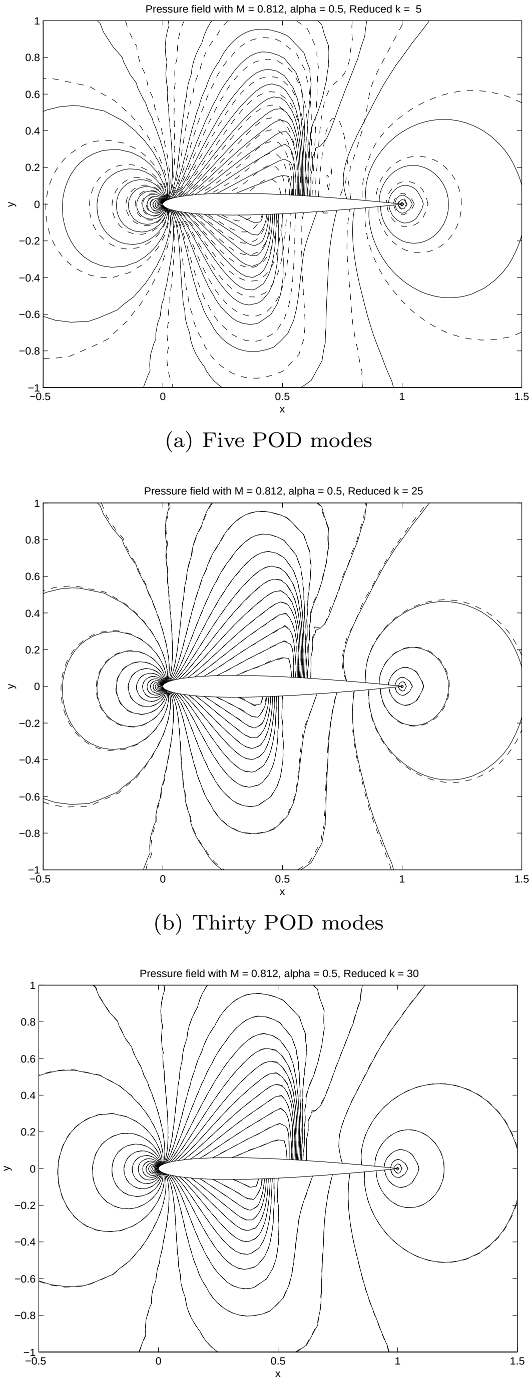


(a) Five POD modes

(b) Twenty five POD modes

(c) Thirty POD modes

Figure 5.6: Comparison of predicted pressure contours (dash) and exact pressure contours (solid) for a Mach number of 0.812 and angle of attack of 0.5°.



(a) Five POD modes

(b) Thirty POD modes

(c) Thirty five POD modes

Figure 5.7: Comparison of predicted pressure contours (dash) and exact pressure contours (solid) for a Mach number of 0.812 and angle of attack of 1.1°.

The final example considered for this set is the prediction for the pair ($\alpha = 0.45, M = 0.812$) in which both $\alpha = 0.45$ and $M = 0.812$ are not values used to generate the snapshots. Therefore, it is expected that a greater number of POD modes are needed to obtain close agreement with the exact predictions. Figure 5.7 compares the predicted pressure field contours (dashed lines) with the exact pressure field contours (solid lines) for this case. As expected, Figure 5.7(b) shows that using 30 modes leads to discrepancies which are larger than that shown in Figure 5.6(c) with the same number of modes. It can be seen in Figure 5.5(c) that at least 40 modes are required to get the same level of agreement and accuracy with the exact flowfield obtained in the previous two examples.

These results show that the POD method combined with interpolation enables one to derive simple low-order models that accurately predict steady-state pressure fields over a range of parameter values such as Mach number and angle of attack. The approach can be extended to the case where more than two parameters vary. For example, one might wish to include geometric properties of the airfoil in order to apply these models in an airfoil shape design optimization context. While the number of snapshots in this case might be large, the method is straightforward to apply.

5.2.3 Estimation of the Greatest Lift Coefficient for a Given Angle of Attack Range

Here the problem of estimating the greatest lift coefficient in a given range of angle of attack is considered within the framework of POD and its extensions. The NACA 0012 airfoil is considered at its baseline freestream Mach number of

0.8, and 51 aerodynamic flowfield snapshots are computed for uniformly spaced values of angle of attack in the interval $[-1.25°, 1.25°]$ with a step of $0.05°$. As a demonstration, the values of the angle of attack in the interval [-1.25, 1.25] are used to find the largest value of the lift coefficient C_L in that range. Fifteen modes are used for this case. As the the flow model used is inviscid, it is expected that in the given range of the angle of attack, the largest lift coefficient corresponds to the case when the angle of attack is 1.25 degrees while in the general case the maximum may be at some value inside the interval.

Tolerance	0.1	0.01	0.001	0.0001
No. of iterations	6	11	15	20
AOA	1.1936	1.2449	1.2492	1.2499

Table 5.1: The tolerance and corresponding result by Golden Search method for greatest lift coefficient problem.

Table 5.1 shows that the solution converges to the expected value 1.25 as the tolerance level for the desired level of accuracy is refined. The same approach can be used to estimate the maximum lift coefficient for viscous aerodynamic flowfields predicted using an ensemble of flowfield snapshots computed by solving the Navier-Stokes equations for the span of angle of attack which includes the stall angle.

5.2.4 Tracking the Angle of Attack for a Given Flowfield Around an Airfoil

For demonstration purposes, the flowfield of interest in this example is the Mach field V around the NACA 0012 airfoil corresponding to $AOA = 0.77$, assuming that the flowfield information and other data corresponding to AOA of 0.77 are unknown and undefined in the ensemble of flowfield snapshots.

The NACA 0012 airfoil is considered at its baseline freestream Mach number of 0.8, and 51 snapshots are computed corresponding to uniformly spaced values of angle of attack in the interval $[-1.25°, 1.25°]$ with a step of 0.05°. The Golden Section Search method together with the interpolation-based POD are used here to find the minimum cost function defined in (4.1). The angle of attack spans the interval [-1.25, 1.25], and 15 POD modes are used. It is noted that the minimum cost function and the angle of attack corresponding the given Mach field are obtained at the same time.

Tolerance	0.1	0.01	0.001	0.0001
No. of iterations	7	12	17	21
AOA	0.7815	0.7723	0.7699	0.7700

Table 5.2: The tolerance and corresponding result by Golden Search method for the angle of attack tracking problem.

Table 5.2 shows that the result converges to the expected $AOA = 0.77$ as the tolerance level is refined. Once the AOA is determined, all other flow variable fields corresponding to the Mach field can be obtained immediately by applying the method in the section 5.2.2. The application of the tracking problem for the other fields operates in the same manner and is straightforward.

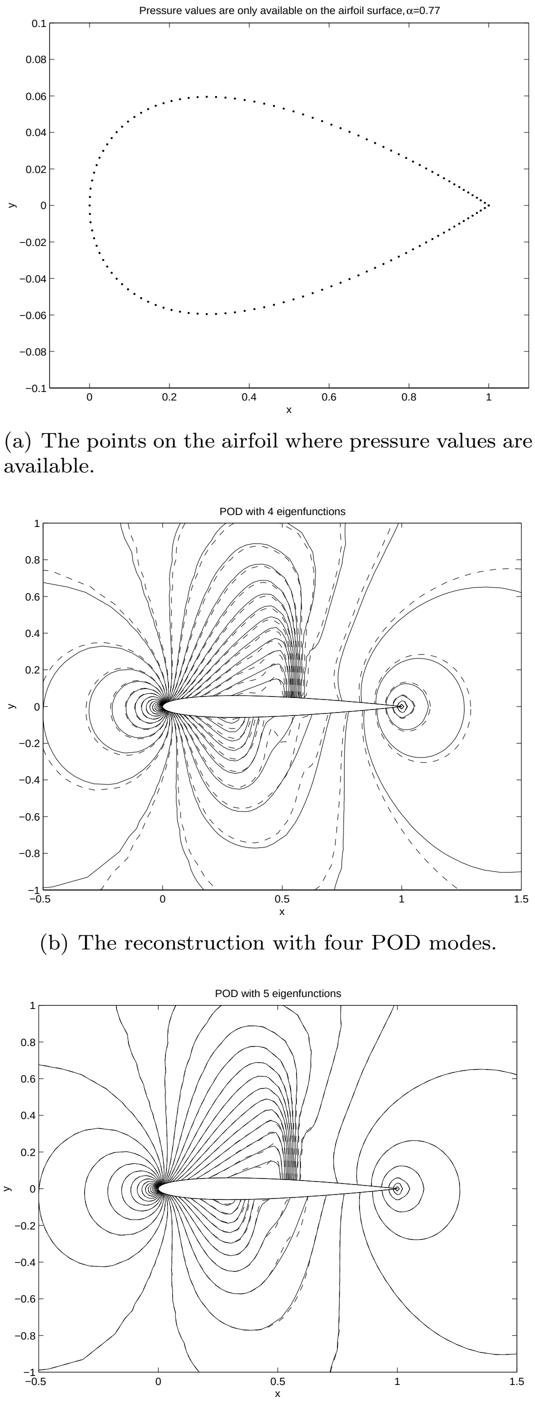
5.3 Reconstruction of Missing Aerodynamic Data

5.3.1 Reconstruction of Aerodynamic Flowfield Data

The case considered is the NACA 0012 airfoil at a freestream Mach number of 0.8. To create the POD basis, 51 snapshots are computed at uniformly spaced values of angle of attack in the interval $\alpha = [-1.25°, 1.25°]$ with a step of 0.05°. An aerodynamic flowfield is then generated by computing the flow solution at

$\alpha = 0.77°$ and at the same Mach number as the base Mach number of 0.8 (which is not one of the snapshots). An incomplete aerodynamic data field is then created by only retaining the pressure values on the surface of the airfoil. This mimics the situation of a typical wind-tunnel airfoil surface pressure measurement which consists of available data only at discrete locations on the airfoil surface. The total number of pressure values in the complete flowfield is 6369 and the number of pressure values on the airfoils surface is 121, hence for this case about 98% of the data is considered missing. The goal is then to reconstruct the entire pressure flowfield in the vicinity of the airfoil using the gappy POD method and to compare the prediction with the exact solution corresponding to the original CFD solution. The gappy POD method can also be viewed as an approach that provides a way to combine experimental data with computational results in order to reconstruct the entire aerodynamic flowfield. Figure 5.8(a) shows the points on the NACA 0012 airfoil surface where pressure field values are made available. Figures 5.8(b) and (c) compare the reconstructed pressure field contours using four and five POD modes, respectively, with the complete flowfield contours obtained from the original CFD solution. As expected, the greater the number of modes used, the more accurate is the quality of the reconstructed flowfield. With the knowledge of limited surface pressure data available, the complete pressure field in the vicinity of the airfoil can be determined very accurately with only five POD modes, showing that the gappy POD methodology for data reconstruction works effectively for aerodynamic flowfield data.

A question of interest is whether or not the complete pressure field can be



(a) The points on the airfoil where pressure values are available.

(b) The reconstruction with four POD modes.

(c) The reconstruction with five POD modes.

Figure 5.8: The reconstruction of the pressure field from the airfoil surface pressure distribution (dash), compared with the original contours (solid).

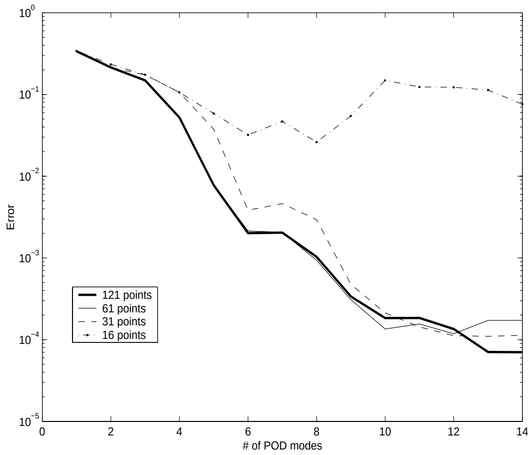


Figure 5.9: The 2-norm of the pressure reconstruction error versus the number of POD modes. For all four cases, the pressure measurement points were equally spaced over the airfoil surface.

accurately reconstructed with knowledge of surface data at a fewer data points than that considered earlier. Figure 5.9 shows the variation of the 2-norm of the error between the exact and reconstructed pressure field for different cases corresponding to different number of available surface data points. In each case, the number of available surface measurements is evenly distributed around the airfoil surface. It can be seen from Figure 5.9 that when the number of available data points is reduced by half or even when only 31 surface pressure measurements are available, the complete pressure field can be constructed accurately using about ten POD modes. If the number of available data points is reduced to say 16 surface data points, then it can be seen that the prediction becomes unreliable. It can also be noted that, given N measurements, a uniformly-spaced distribution is unlikely to yield the best reconstruction of the flowfield. The question on the best distribution of the surface data points for collecting surface pressure measurements is being addressed in ongoing research to see how the gappy POD method can be used to determine optimal pressure sensor locations on the airfoil surface to determine the complete pressure field in the vicinity of the airfoil effectively and economically.

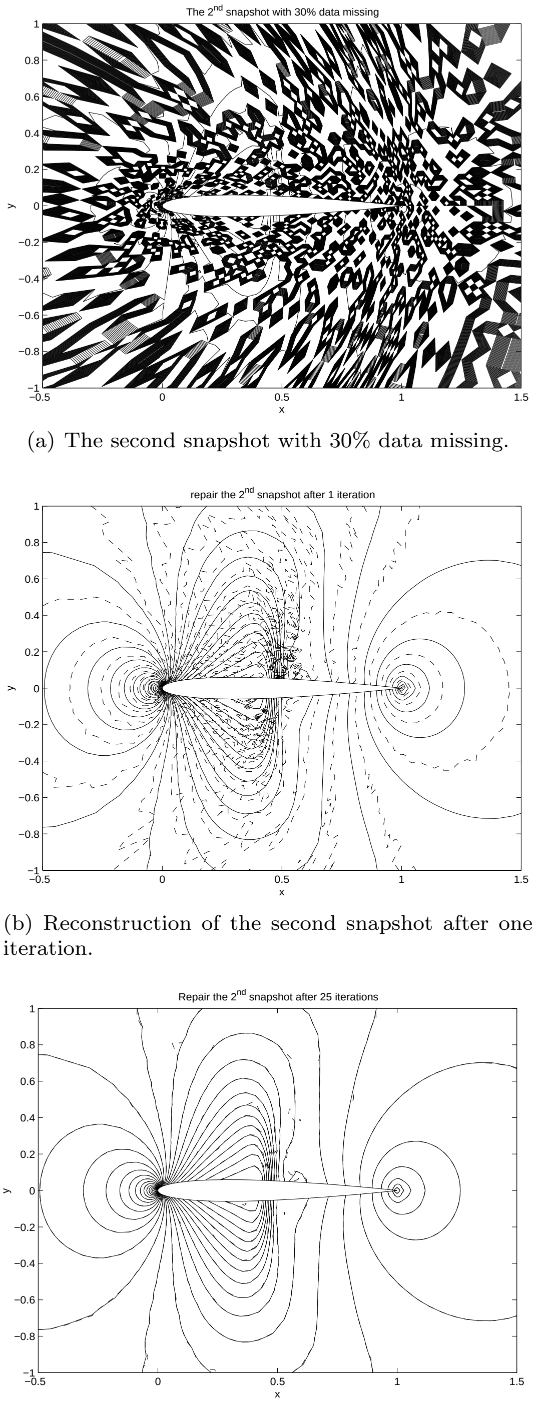
5.3.2 Incomplete Flowfield Snapshot Set

In the second example, the construction of a set of POD basis vectors from an incomplete set of aerodynamic flowfield snapshots and the reconstruction of the incomplete set itself are considered. This problem may again be of interest if coarse or partial flow field data is available from experimental results. Using the gappy POD methodology, experimental and computational data with differing levels of resolution can be combined effectively to determine dominant flow modes. Once the POD basis vectors are constructed, any incomplete data which does not belong to the incomplete set can be reconstructed as in section 5.3.1.

Again, the NACA 0012 airfoil is considered at a freestream Mach number of 0.8. An ensemble consisting of 26 flowfield snapshots corresponding to steady pressure flowfields at angles of attack in the range $\alpha = [0°, 1.25°]$, uniformly spaced with an interval of 0.05° is used for this study. To create the incomplete snapshot set for this example, 30% of the pressure data of each snapshot is discarded randomly. The iterative gappy POD algorithm is then used to repair the data as follows. By first repairing the missing data points in each snapshot with the average over available data at that point, a new ensemble of data which has no missing values is created. Using this new ensemble, a first approximation to the POD basis is constructed. Then, each snapshot in the ensemble is repaired using the first approximation of the POD basis. This repaired ensemble is then used to construct a second approximation to the POD basis. For the example in this section, the iterative procedure above is stopped after about 50 iterations.

In Figure 5.10, the second snapshot with 30% pressure field data missing is repaired using the procedure described above with five POD modes, which contain 99.99% of the flow energy. Figure 5.10(a) shows the original damaged snapshot. After one iteration, the repaired snapshot in Figure 5.10(b) begins to resemble the CFD solution; however, large discrepancies remain at various spatial locations. Figure 5.10(c) shows the repaired snapshot after 25 iterations and the resulting pressure field resembles closely with the original flowfield. Figure 5.11 shows the repairing process for the 23^{rd} snapshot. Compared with the pressure contours corresponding to the one in which 30% of the data has been assumed missing as shown in Figure 5.11(a), the reconstruction in Figure 5.11(c) appears to be in very close agreement with the original CFD result and this has been accomplished using only seven iterations. It can be seen that the convergence of the reconstruction process depends on the details of the particular snapshot under consideration. In particular, it depends on the structure of the flowfield snapshot and how much data is missing. For the 23^{rd} snapshot shown in Figure 5.11, the convergence rate for reconstruction is much faster than that of the second snapshot shown in Figure 5.10.

The convergence of the POD eigenvalue spectrum of the incomplete ensemble is shown in Figure 5.12. It can be seen that after one iteration the first two eigenvalues have converged, while convergence of subsequent eigenvalues requires more iterations. For example, after 45 iterations, it can be seen that only the first five eigenvalues have converged; however, these five modes account

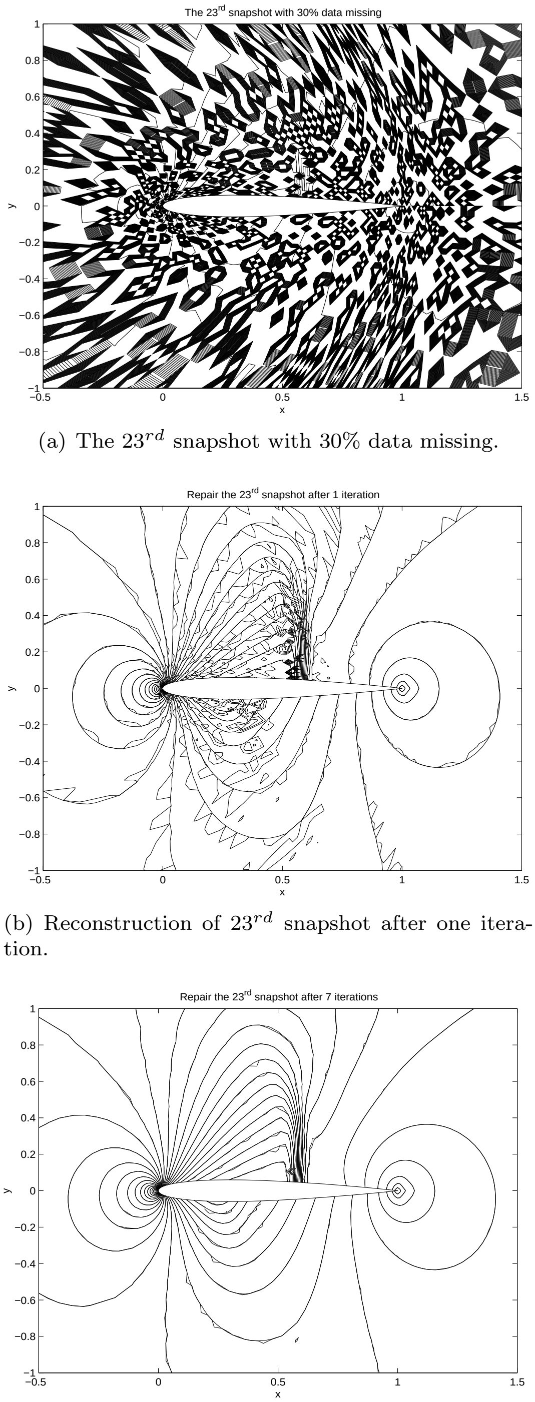


(a) The second snapshot with 30% data missing.

(b) Reconstruction of the second snapshot after one iteration.

(c) Reconstruction of the second snapshot after 25 iterations.

Figure 5.10: Reconstruction of the second snapshot (dash), compared with the original contours (solid).



(a) The 23^{rd} snapshot with 30% data missing.

(b) Reconstruction of 23^{rd} snapshot after one iteration.

(c) Reconstruction of 23^{nd} snapshot after seven iterations.

Figure 5.11: Reconstruction of the 23^{rd} snapshot (dash), compared with the original contours (solid).

for almost all of the flow energy (99.99%).

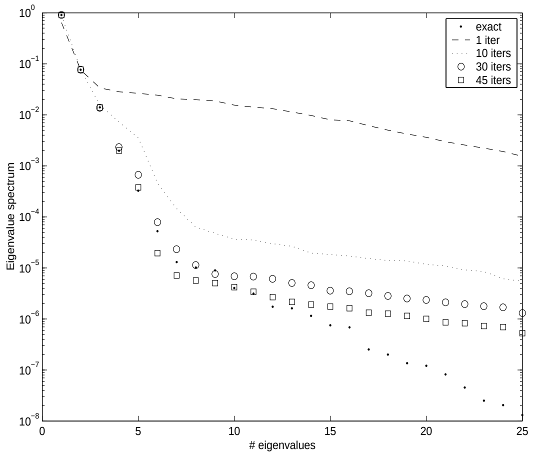
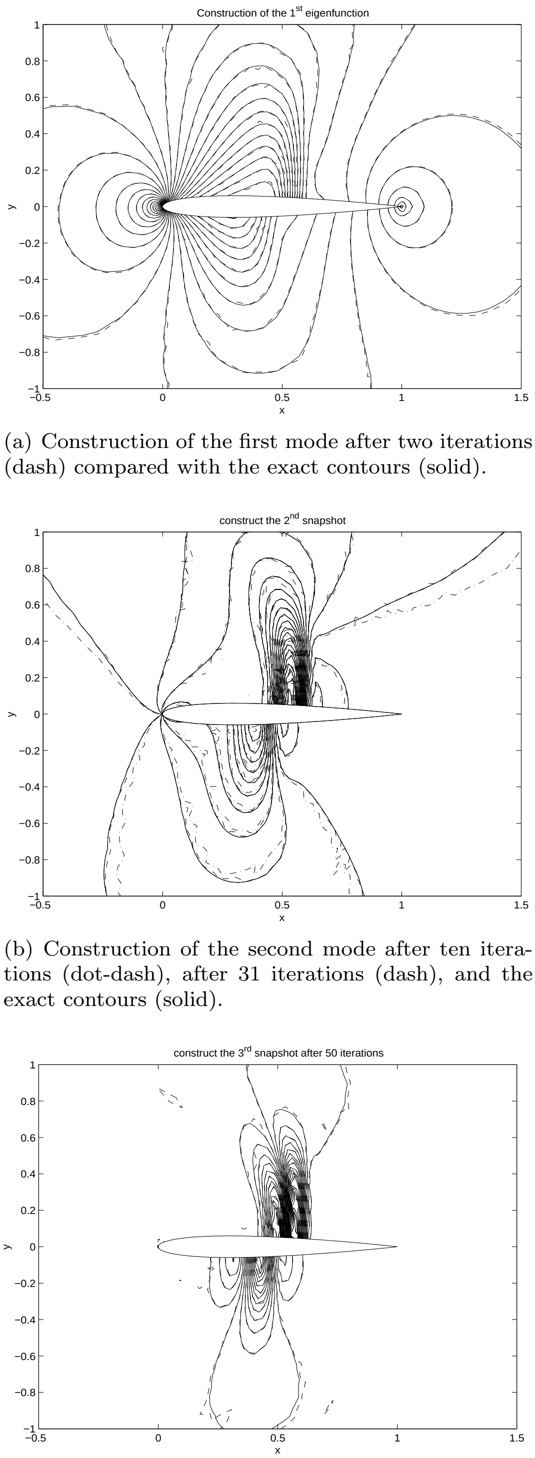


Figure 5.12: The eigenvalue spectrum for construction of the POD basis from an incomplete snapshot set. Shown are the POD eigenvalues at various stages in the iterative process.

The rate of convergence for the construction of the POD modes is also investigated. This is done by comparing the POD basis vectors constructed using a complete data set with those constructed from the incomplete set. In Figure 5.13(a), it can be seen that, after just two iterations, the first mode constructed with incomplete data matches well with the desired result. However, in Figure 5.13(b), the construction of the second mode after ten iterations still shows significant deviation from the actual mode. At least 31 iterations are needed to obtain the second mode accurately. Figure 5.13(c) shows that after 50 iterations the estimate of the third POD mode is reasonable but not fully converged.

It is observed that the more energetic a POD mode, the faster the rate of construction from a given patial flowfield data set. The first mode, which captures about 90.65% energy, requires only two iterations to converge very



(a) Construction of the first mode after two iterations (dash) compared with the exact contours (solid).

(b) Construction of the second mode after ten iterations (dot-dash), after 31 iterations (dash), and the exact contours (solid).

(c) Construction of the third mode after 50 iterations (dash) compared with the exact contours (solid).

Figure 5.13: Construction of POD modes from an incomplete snapshot set.

closely to the desired result; however, the second and the third modes, which capture about 7.7% and 1.4%, require about 31 and 50 iterations, respectively. This trend is observed for subsequent modes and may be related to the fact that successive POD modes often correspond to higher spatial frequency flow structures. For example, it can be seen in Figure 5.13 that the second and third modes contain progressively more high-frequency shock structure. It is therefore not surprising that construction of higher modes requires more iterations.

Finally, the computational cost of the iterative gappy POD procedure is considered. At each iteration, the reconstruction of m snapshots, requires the solution of m systems of the form (3.13). Furthermore, at each iteration, an eigenvalue problem of size m must also be solved to determine the POD modes. Implementation of the method using five POD modes for each iteration on a 1.6 GHz Pentium 4 personal computer, took less than two seconds.

5.4 Airfoil Design Optimization

5.4.1 Inverse Design Using Gappy POD Method

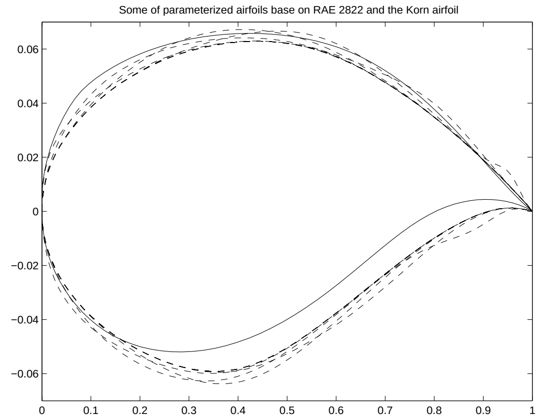
In this section, a set of examples demonstrates how the gappy POD method can be applied to the problem of inverse airfoil design. An ensemble of snapshots is first generated as in (4.3) by choosing a set of airfoil shapes and obtaining their corresponding surface pressure distributions. (Other flowfield quantities can also be included to form the snapshots if they are of interest.) Here these two quantities are chosen because they are of prime interest for airfoil shape design problems which are concerned with the determination of airfoil shapes which can produce prescribed airfoil surface pressure distributions or which can satisfy specific constraints on aerodynamic force and moment coefficients. While in this thesis, CFD results were used to create the snapshots, in practice, the flow data could be obtained from computational simulations, experimental results, tabulated data, or a combination thereof. The goal here is to apply the gappy POD method to determine the optimal airfoil shape that can produce a given target pressure distribution, which is not contained in the snapshot collection. Here a set of airfoil shape design optimization problems is considered in the subsonic and transonic regimes using the framework outlined in section 4.4.

Subsonic Regime

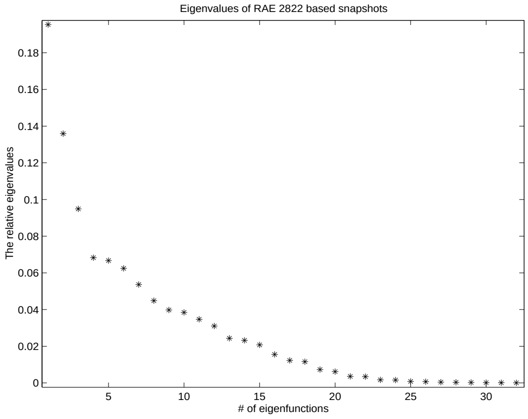
The first example considered is that of inverse airfoil design in the subsonic regime. An ensemble of airfoil profiles is created by considering the base RAE 2822 airfoil and adding a series of Hicks-Henne bump functions [20], which permit one to render smooth changes in the airfoil profile geometry. These functions are traditionally used for parametric representation of airfoil sections in airfoil design optimization problems. The Hicks-Henne bump functions are defined as

$$y(x) = A\left\{\sin\left[\pi x^{log(1/2)/log(t_1)}\right]\right\}^{t_2}, \ \ 0 \leq x \leq 1 \tag{5.2}$$

where A is the magnitude of the maximum bump amplitude, t_2 is used to control the width of the bump and $x = t_1$ is the location of the maximum bump amplitude. Thirty one bump functions are created using $t_2 = 4, A = 0.005$ and added to points on each of the upper and lower surfaces of the RAE 2822 airfoil to create a total of 63 airfoil snapshots, some of which are shown in Figure 5.14(a). The aerodynamic flowfields around these airfoils inclined at zero angle of attack and at a freestream Mach number of 0.5 are then computed using the flow solver. These airfoils and their corresponding surface pressure distributions extracted from the computed flowfield solutions are used to form the ensemble of snapshots.



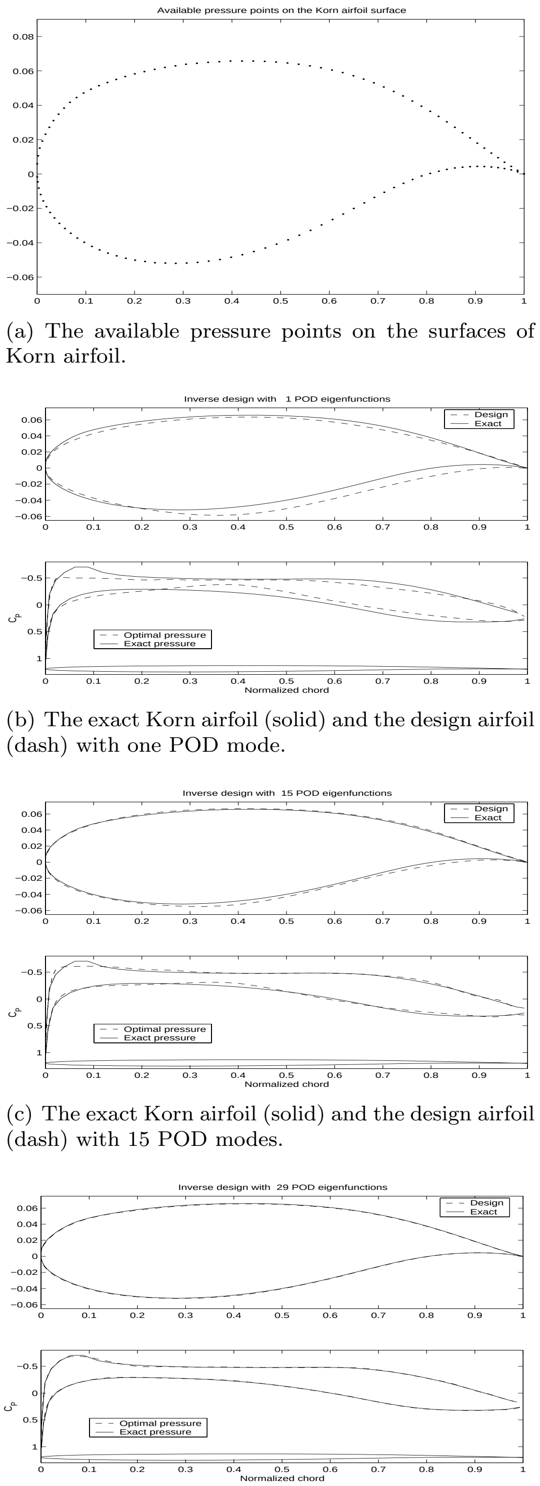
(a) The airfoil snapshots from baseline RAE 2822 (dash) and the Korn airfoil (solid).



(b) POD eigenvalues from the airfoil snapshots.

Figure 5.14: The airfoil snapshots from baseline RAE 2822 (dash) and the Korn airfoil (solid) and corresponding POD eigenvalues.

The airfoil surface pressure distribution around the Korn airfoil whose geometry is shown in Figure 5.14(a) and which is inclined at the same angle of attack and Mach number as the ensemble of airfoils is computed using the flow



(a) The available pressure points on the surfaces of Korn airfoil.

(b) The exact Korn airfoil (solid) and the design airfoil (dash) with one POD mode.

(c) The exact Korn airfoil (solid) and the design airfoil (dash) with 15 POD modes.

(d) The exact Korn airfoil (solid) and the design airfoil (dash) with 29 POD modes.

Figure 5.15: Korn airfoil design.

solver and specified as the design target airfoil surface pressure distribution P* as defined in section 4.4. It can be seen in Figure 5.14(a) that, while the Korn airfoil shares some similarities with the RAE 2822-based snapshot set, its camber and thickness are quite distinct. This example thus represents a challenge to test the effectiveness of the gappy POD to carry out reliable inverse airfoil design tasks. The first 32 POD eigenvalues corresponding to the snapshot set are shown in Figure 5.14(b). It can be seen that the first 21 POD modes contain 99% of the system energy. Figure 5.15(a) shows the points on the Korn airfoil where target pressure values P* are specified. The inverse airfoil design problem then corresponds to the determination of an optimal airfoil shape whose surface pressure distribution P minimizes the cost function defined by equation 4.2. Using the gappy POD procedure outlined in section 4.4, the corresponding optimal airfoil shape can then be determined easily. Figures 5.15(b), (c) and (d) compare the designed airfoil and surface pressure distribution using the gappy POD using one, 15 and 29 POD modes, respectively with the exact Korn airfoil. It can be seen that as the number of POD modes is increased, the predicted shape and its pressure distribution agree more closely with the exact solutions. The corresponding values of the cost J are given in Table 5.3. By using 29 POD modes, which accounts for 99.97% of the snapshot energy, it can be seen that the error in the pressure distribution is very small. It is noted that if more than 29 POD modes are used, the improvement is negligible.

Additional inverse design optimization problems using the same ensemble of snapshots as in the previous inverse design problem for three different specification of target pressure distributions are considered to illustrate certain other

Airfoil	Number of POD modes	Cost, J
Korn-63 snapshots	1	0.022
Korn-63 snapshots	15	0.0047
Korn-63 snapshots	29	2.9426e-004

Table 5.3: Optimal cost versus number of POD modes for the subsonic inverse design of Korn airfoil.

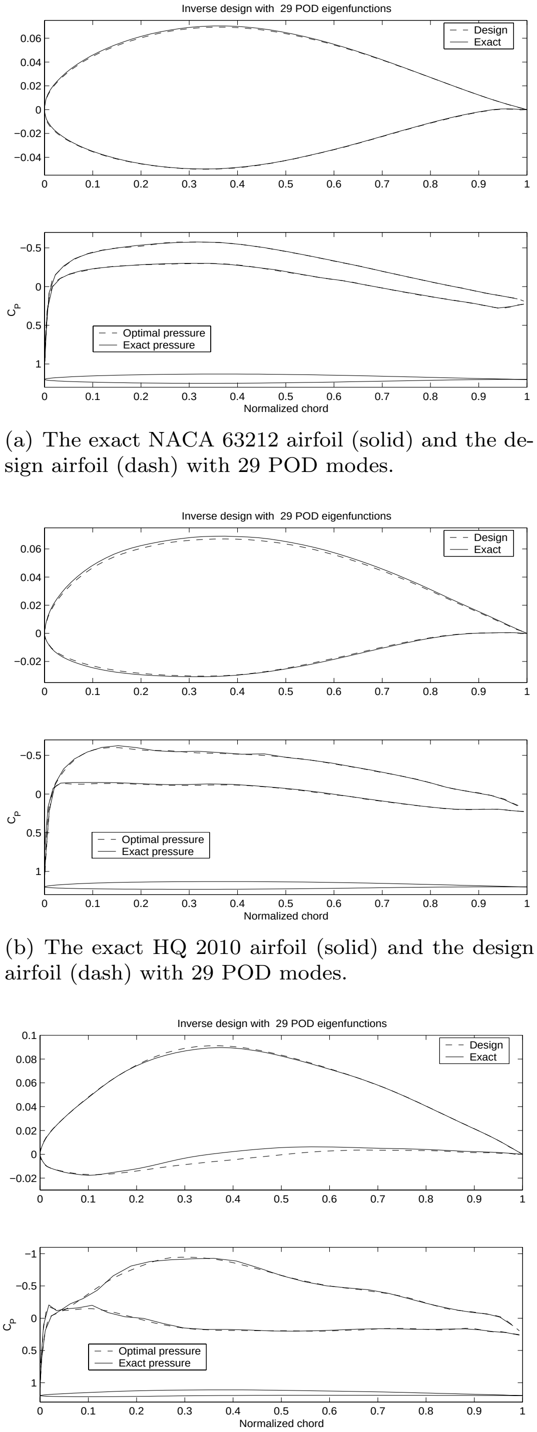
Airfoil	Number of POD modes	Cost, J
NACA 63212-63 snapshots	29	6.2673e-004
HQ 2010-63 snapshots	29	0.0061
GOE 117-63 snapshots	29	0.0056
NACA 63212-90 snapshots	29	1.9532e-004
HQ 2010-90 snapshots	29	2.5858e-004
GOE 117-90 snapshots	29	0.0014
NACA 63212 with restart	43	1.7435e-004
HQ 2010 with restart	32	2.0622e-004
GOE 117 with restart	39	6.8127e-004

Table 5.4: Optimal cost versus number of POD modes for subsonic inverse design cases of NACA 63212, HQ 2010 and GOE 117 airfoils.

features of the gappy POD application in inverse airfoil design optimization problems. Target airfoil surface pressure distributions corresponding to the NACA 63212, Quabeck 2.0/10 R/C sailplane HQ 2010, and GOE 117 airfoils [27] at the same angle of attack and Mach number are computed and specified for the inverse airfoil design practice. The inverse airfoil design shapes and their surface pressure distributions resulting from the application of gappy POD are compared with the target airfoil shapes and the pressure distributions corresponding to the target airfoils in Figure 5.16. Although 29 POD modes are used for the NACA 63212 in Figure 5.16(a), there is still a small region on the upper surface near the leading edge which cannot be resolved accurately. The situation is worse for the HQ 2010 airfoil. As shown in Figure 5.16(b), there are slight differences in the shape in some regions on the upper and lower surfaces.

These slight discrepancies need to be improved further so that the agreement with the target HQ 2010 airfoil will be better. For the case corresponding to the target airfoil GOE 117, as can be seen from Figure 5.16(c), the designed airfoil shape is still far away from the exact shape of the GOE 117 airfoil, indicating that this geometry, which differs considerably from the baseline RAE 2822 airfoil, is not contained in the subspace spanned by the snapshot set considered. From Table 5.4, the values of the cost J can be seen to be larger than those corresponding to the Korn airfoil in Table 5.3, especially in the case of the HQ 2010 and GOE 117 airfoils. Hence further explorations to improve the designed airfoil shape need to be developed.

One approach to improve the resulting airfoil shapes from inverse design to correspond to those of the target airfoil is to increase the richness of the subspace spanned by the POD basis vectors. This can be achieved by including more snapshots in the ensemble. One could for example include additional airfoil snapshots. Here 27 airfoil shapes parameterized from the NACA 64A410 airfoil are added to the ensemble with 63 airfoils considered in the previous example to form a new ensemble of 90 airfoils and the corresponding computed surface pressure snapshots. The 64A410 airfoil has been chosen for some similarities which exist with the three target airfoils hoping that this would add some geometrical richness into the ensemble so as to improve the resulting inverse design shapes. As shown in Figures 5.17(a) and (b), by using this new ensemble and by using the same number of basis vectors i.e 29 POD modes as in the previous cases, the designed airfoil shapes are now almost identical to the exact NACA 63212 and HQ 2010 airfoils. However, the design airfoil in Figure 5.17(c) still

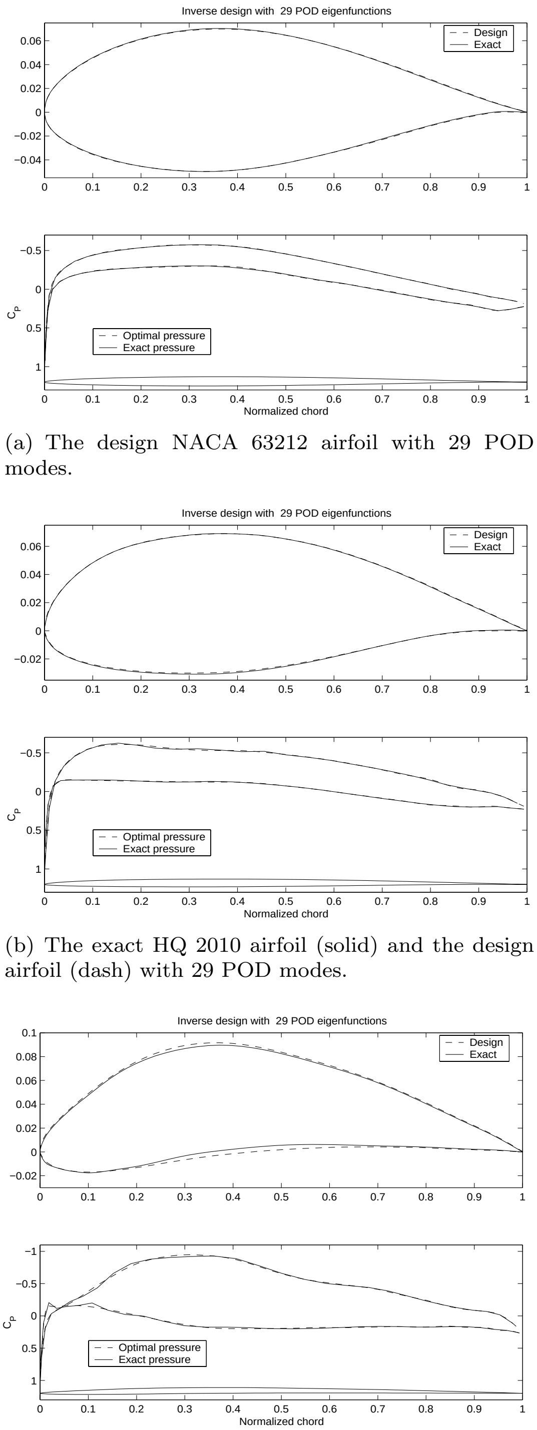


(a) The exact NACA 63212 airfoil (solid) and the design airfoil (dash) with 29 POD modes.

(b) The exact HQ 2010 airfoil (solid) and the design airfoil (dash) with 29 POD modes.

(c) The exact GOE 117 airfoil (solid) and the design airfoil (dash) with 29 POD modes.

Figure 5.16: The design (dash) airfoils from 63 RAE 2822 based snapshots.



(a) The design NACA 63212 airfoil with 29 POD modes.

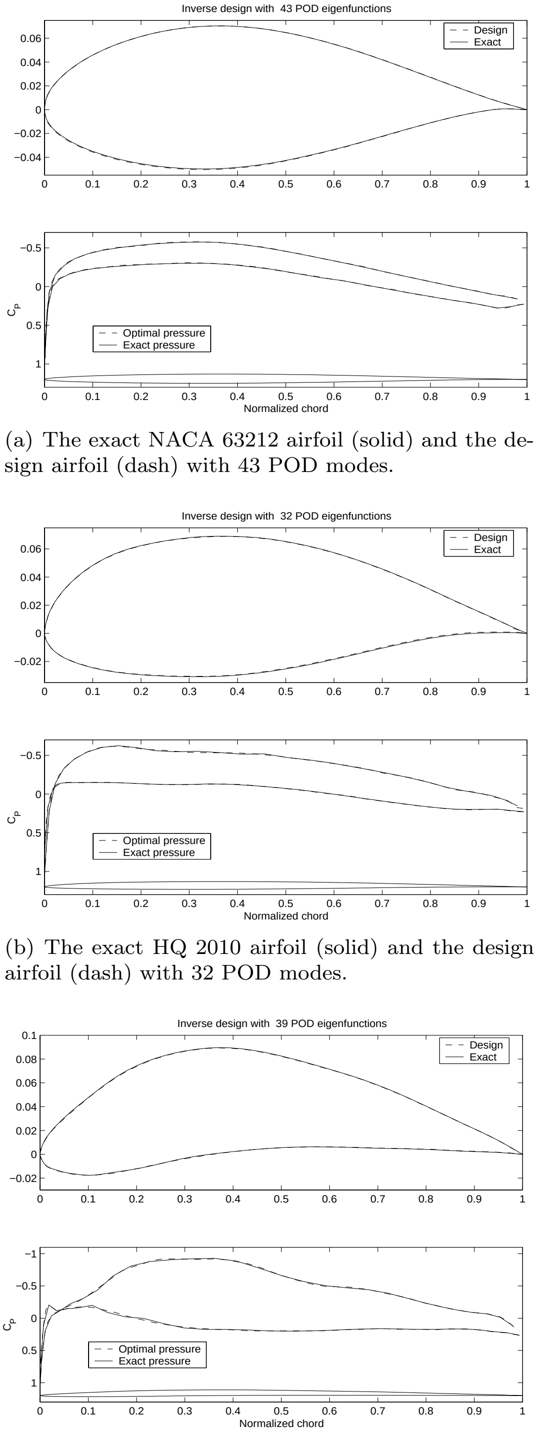
(b) The exact HQ 2010 airfoil (solid) and the design airfoil (dash) with 29 POD modes.

(c) The exact GOE 117 airfoil (solid) and the design airfoil (dash) with 29 POD modes.

Figure 5.17: The design airfoils (dash) airfoils from 90 snapshots with baseline RAE 2822 and NACA 64A410.

shows some discrepancies with the exact GOE 117 airfoil. The better behavior of the new ensemble is also shown in Table 5.4 by the smaller values of the cost J. The exact airfoil could be obtained if further snapshots are added to the set; however, this implies some a priori knowledge of the desired result so that appropriate snapshots may be chosen.

A better way to improve the design airfoil is proposed in Legresley and Alonso [2], where an available designed airfoil at some iteration is used as an intermediate baseline shape, to which some bump functions are added to generate a new set of snapshots. This new snapshot collection is then used to compute a new set of POD modes and thus restarts the design procedure. Here, a similar method for the gappy POD procedure can be done. For the case corresponding to the target airfoil NACA 63212, the designed airfoil reached using 29 POD modes as shown in Figure 5.16(a) is used as an intermediate baseline airfoil, to which 60 bump functions using $t_2 = 4, A = 0.005$ are added to obtain a new ensemble of 61 airfoil snapshots. A similar procedure is used for the intermediate design airfoils shown in Figures 5.16(b) and (c) for the HQ 2010 and GOE 117 cases, respectively. It can be seen in Figure 5.18 that the designed airfoils now match the exact airfoils very well. In order to obtain a cost function reduction on the order 10^{-4}, it is found that about 43, 32 and 39 POD modes are required for the NACA 63212, HQ 2010, and GOE 117 airfoils, respectively. The cost function values attained using the restart procedure are shown in Table 5.4. The results from this restarted gappy POD procedure are much better than those obtained from using the 90-snapshot ensemble above. Moreover, by allowing multiple restarts, this procedure enables the consideration of an inverse design



(a) The exact NACA 63212 airfoil (solid) and the design airfoil (dash) with 43 POD modes.

(b) The exact HQ 2010 airfoil (solid) and the design airfoil (dash) with 32 POD modes.

(c) The exact GOE 117 airfoil (solid) and the design airfoil (dash) with 39 POD modes.

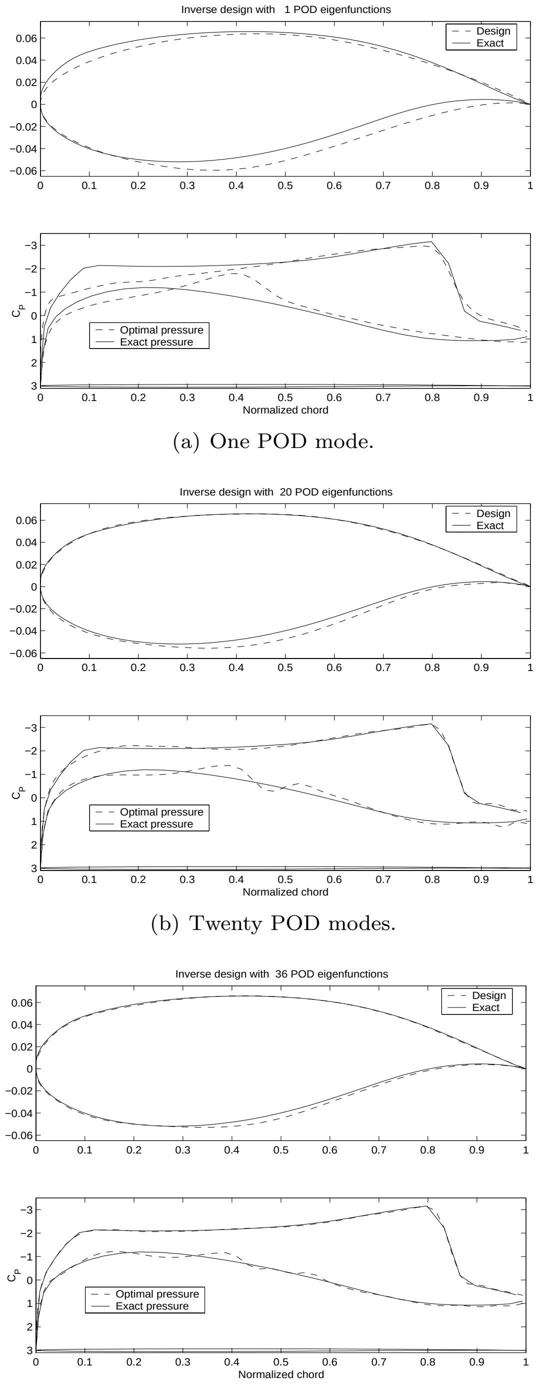
Figure 5.18: The design (dash) airfoils from restarted gappy POD method.

whose geometry differs significantly from that of the baseline airfoil.

Transonic Regime

All the examples in the previous section are at a freestream Mach number of $M = 0.5$ which is subsonic and free of shock waves. It is envisaged that in the transonic regime, the inverse airfoil design task using gappy POD will be more difficult due to the presence of shocks and this motivates the current investigation for inverse transonic airfoil design using gappy POD.

The ensemble consisting of 63 airfoils based on the RAE 2822 airfoil used in the example on the inverse airfoil design corresponding to Korn airfoil target pressure distribution is used for this study but the corresponding flow solutions are now computed at a freestream Mach number of $M = 0.8$. Figures 5.19(a)-(c) compare the shapes of the designed airfoil attained using the gappy POD inverse design corresponding to the Korn airfoil target pressure distribution at $M = 0.8$ using different POD modes ranging from one, 20 and 36. It can be seen that, in the transonic case, even when 36 POD modes, which capture 99.997% of the energy, are used, the lower surface of the designed airfoil cannot be resolved accurately and the cost function J which has reduced to a value of 0.0230 is still large. However, the shape of the designed airfoil can be improved by restarting the gappy POD method as described in the previous section. The designed airfoil shown in Figure 5.19(c) is now used as the intermediate baseline airfoil shape and a new set of airfoil snapshots is created using 60 bump functions with $t_2 = 4, A = 0.001$. Figures 5.20(a)-(c) compare the shapes of the designed airfoil for one, 20 to 29 POD modes created from the new ensemble with the original target Korn airfoil. Figure 5.20(a) shows that using just one of the new

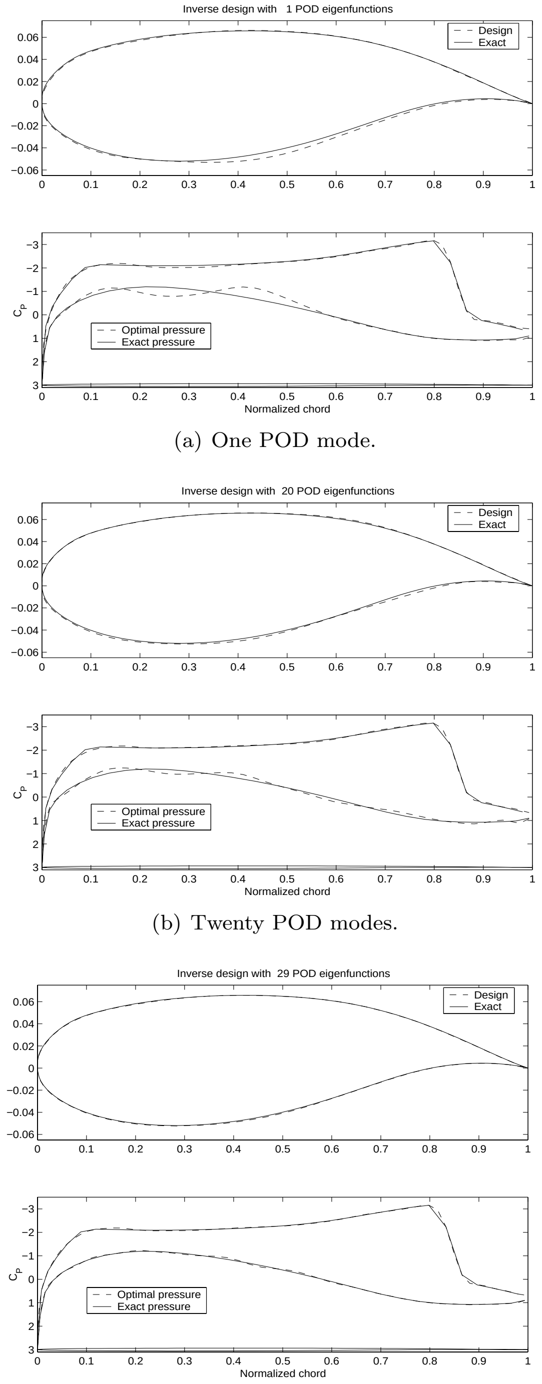


(a) One POD mode.

(b) Twenty POD modes.

(c) Thirty six POD modes.

Figure 5.19: The exact Korn (solid) and the design (dash) airfoils from 63 RAE based airfoils in transonic regime.



(a) One POD mode.

(b) Twenty POD modes.

(c) Twenty nine POD modes.

Figure 5.20: The exact Korn (solid) and design (dash) airfoils from new ensemble of snapshots based on the intermediate airfoil in transonic flow (restart).

POD modes yields a result which is very similar to the intermediate baseline airfoil shown in Figure 5.19(c). The designed airfoil with 20 POD modes is significantly improved as can be seen in Figure 5.20(b). If the design process uses 29 POD modes then it can be seen from Figure 5.20(c) that the designed airfoil is in good agreement with the Korn airfoil shape. This example shows that the design of an airfoil using gappy POD requires considerably more effort in the transonic regime than in a subsonic regime. Although the restarted gappy POD procedure can be used to obtain more accurate results, i.e. smaller values of the cost function J; the gappy design procedure becomes more expensive with more restarts.

In general, several restarts of the gappy POD could be required in order to obtain the desired target pressure distribution, especially if the designed airfoil shape is significantly different from those included in the snapshot set, or if significant nonlinearities exist in the flow. For each restart, the CFD solver must be used to obtain the flow solutions corresponding to the new set of parameterized airfoils and this constitutes the most expensive part of the computation. One could also utilize different approaches to generate the new airfoil set. For example, the parameters A, t_1 and t_2 for the bump functions could be varied. Considering more bump functions would result in a larger snapshot set, but may reduce the number of restarts required.

5.4.2 Constrained Airfoil Design Optimization

For the constrained airfoil design optimization problem outlined in section 4.4.2 an initial guess of an airfoil shape is required to start the design optimization process. In this section, the baseline RAE 2822 airfoil at AOA of 0 degree and Mach number of 0.5 is parameterized by 20 Hick-Henne bump functions to create 21 airfoil snapshots including the baseline. The initial airfoil guess is assumed to be the airfoil corresponding to the average of all the airfoil snapshots in the ensemble. Figure 5.21 compares the designed airfoil with the initial airfoil to show the shape alteration to the initial profile to attain the satisfaction of the cost function and constraints defined by Equation 4.9 using the procedure outlined in Equations 4.5-4.9 and based on five POD basis functions. Table 5.5 compares the lift and drag coefficients of the initial and the designed airfoils. It can be seen that drag reduction has been obtained for the designed airfoil compared with the initial airfoil while the lift coefficient is maintained according to the constraint requiring the preservation of the lift coefficient while reducing the drag coefficient. The number of design variables used here is five and the converged solution to the designed airfoil shape problem (4.9) is reached after about 24 iterations in 0.04 seconds (CPU time) using a 1.6 GHz Pentium 4 personal computer.

	initial airfoil	optimal airfoil
C_L	0.2966	0.2969
C_D	0.000545	0.0003

Table 5.5: The lift and drag coeficient for the initial and optimal airfoils, five POD modes.

To validate the results such as pressure distribution, flowfield contours etc estimated from the outcomes of the constrained problem i.e. Equation (4.9), the flowfield around the resulting designed airfoil is computed and the airfoil surface pressure distribution is extracted using the flow solver. Figure 5.22

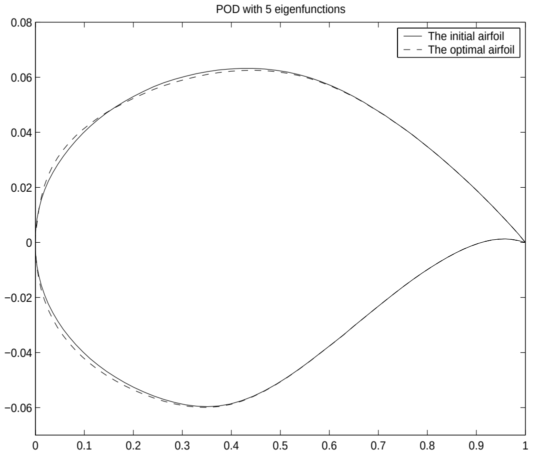


Figure 5.21: The initial airfoil (solid) and the design airfoil with five POD modes.

compares the airfoil surface pressure distribution from Equation (4.8) with that computed for the designed airfoil using the flow solver. It can be seen that both distributions match each other very well for this case.

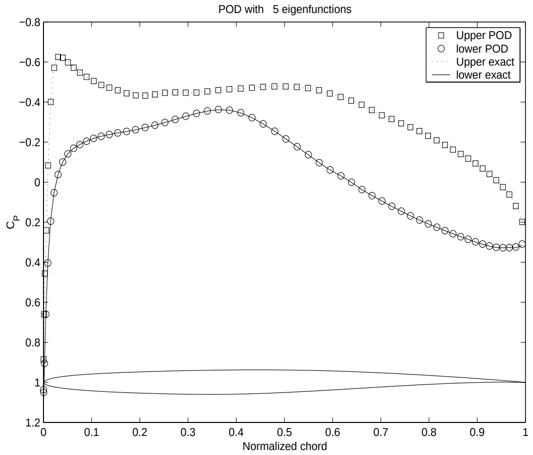


Figure 5.22: The pressure coefficients comparison for the design airfoil with five POD modes.

It is expected that a better improvement in lift and drag coefficients for the design airfoil can be obtained when more POD modes (and hence the number of design variables) are used. Figure 5.23 compares the initial and designed airfoil shapes for the case when ten POD modes are used. From Table 5.6 it can be seen that improvement in lift and drag coefficients is now obtained. For this case it can be seen that the airfoil surface pressure distributions from (4.8) and from the flow solver for the designed airfoil are identical as shown in Figure 5.24. However it is observed that although much more improvement is achieved when the number of POD modes is increased, the designed airfoil is somewhat unphysical. A possible reason is because the current model has only taken advantage of numerics on the ensemble of snapshots in order to solve the optimization problem (4.9). Therefore although an optimal solution can be obtained with negligible cost, it may not guarantee to result in a physical airfoil. Further constraints such as the preservation of the cross-sectional area imposed as a constraint maybe needs to be incorporated in this approach to maintain its advantage and also to lead to a physical solution.

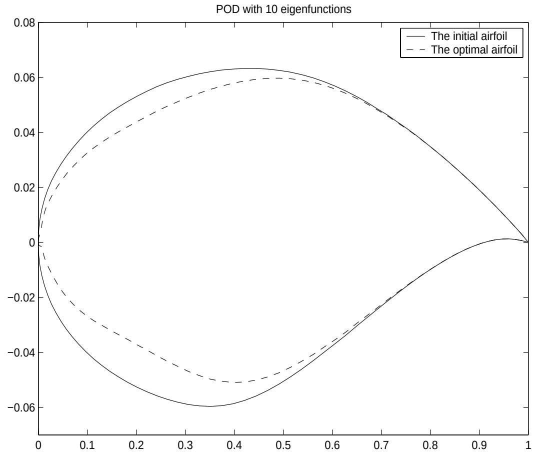


Figure 5.23: The initial airfoil (solid) and the design airfoil with ten POD modes.

	initial airfoil	optimal airfoil
C_L	0.2966	0.3141
C_D	0.000545	0.00032

Table 5.6: The lift and drag coeficient for the initial and optimal airfoils, ten POD modes.

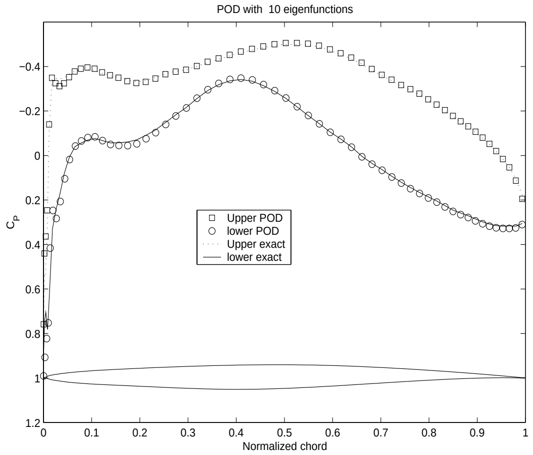
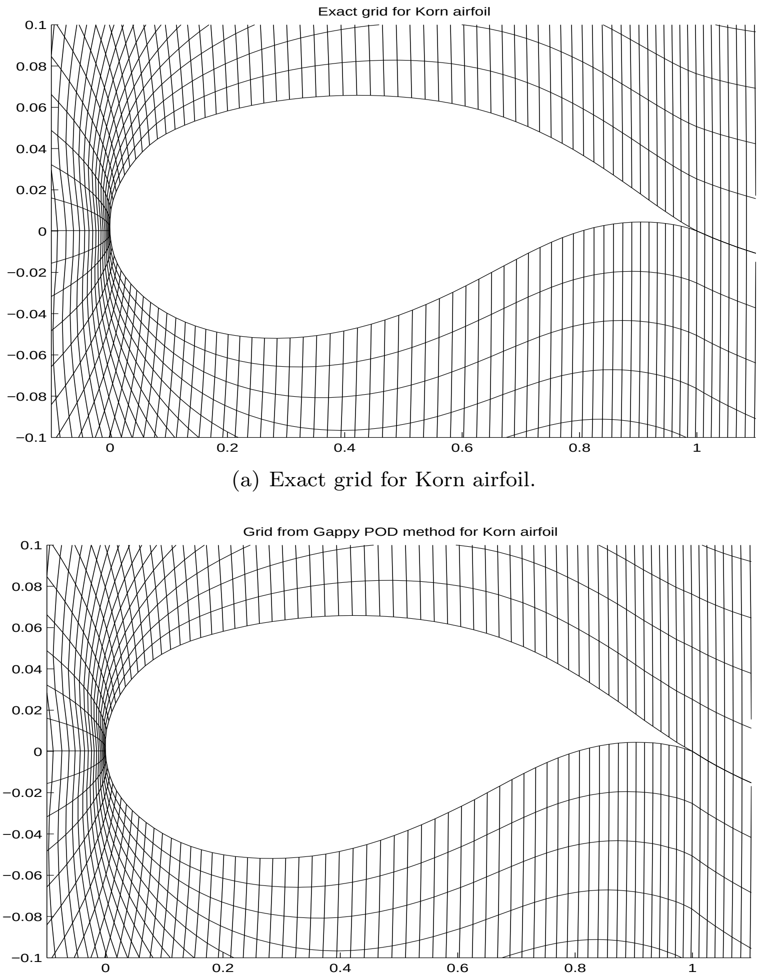


Figure 5.24: The pressure coefficients comparison for the design airfoil with ten POD modes.

5.5 Prediction of Flowfield around an Arbitrary Airfoil

5.5.1 Grid Generation Around an Arbitrary Airfoil

In this section, an attempt to generate the grids around an arbitrary airfoil via gappy POD method is considered. As outlined in section 4.5, if the grid generation around an arbitrary airfoil is of interest, the coordinates C^* of a given airfoil shape are available data while the grids G^* are treated as missing data. For example, the coordinates of the Korn airfoil are specified as available data while the grid coordinates around it are unknown and treated as missing data. The ensemble of 63 snapshots around the baseline airfoil RAE 2822 in



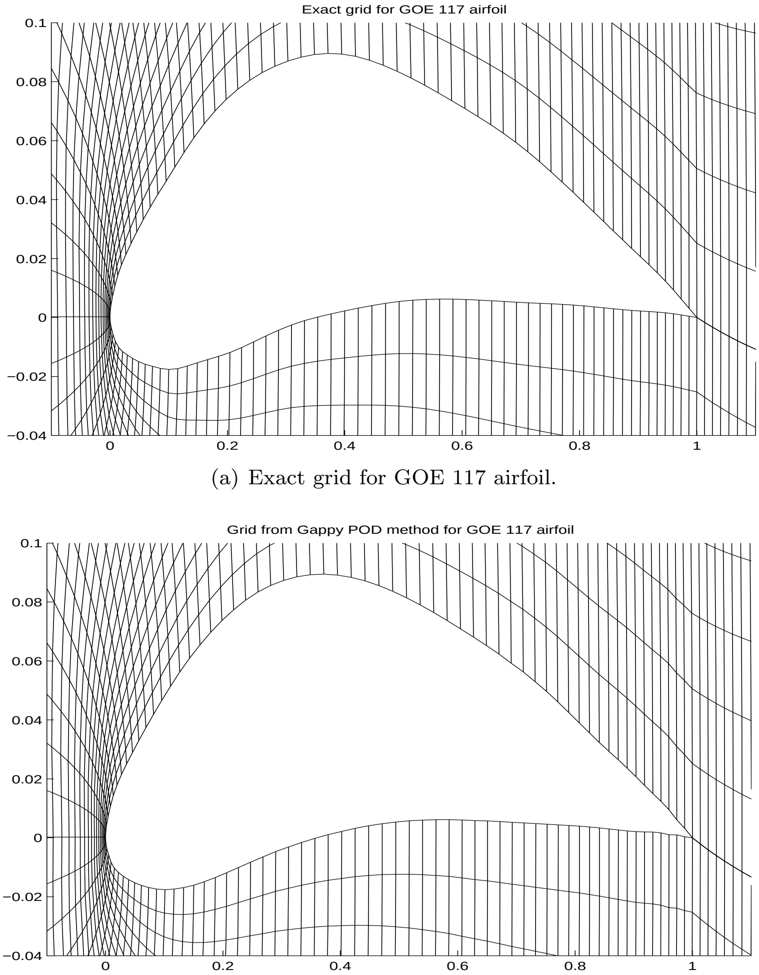
Exact grid for Korn airfoil

(a) Exact grid for Korn airfoil.

Grid from Gappy POD method for Korn airfoil

(b) Grid for Korn airfoil from gappy POD method.

Figure 5.25: The grid generation with 29 POD modes for Korn airfoil from RAE based snapshots.



(a) Exact grid for GOE 117 airfoil.

(b) Grid for GOE 117 airfoil from gappy POD method.

Figure 5.26: The grid generation with 29 POD modes for GOE 117 airfoil from RAE based snapshots.

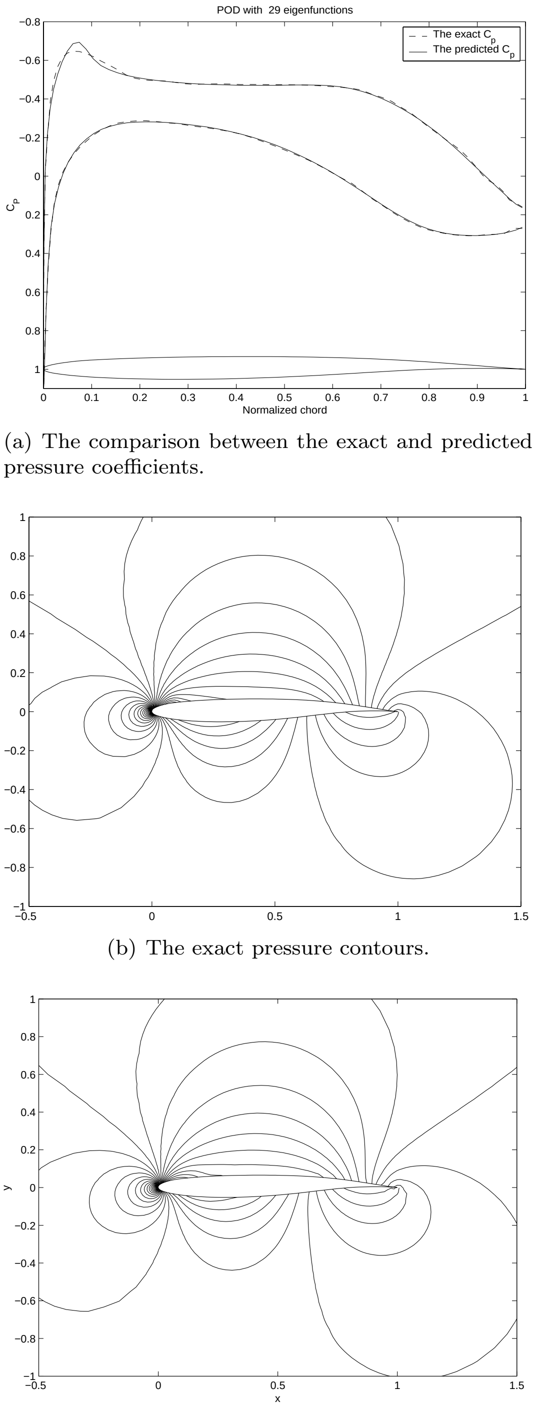
section 5.4.1 is used again and 29 POD basis functions which capture 99.994% of energy are used. Figures 5.25(a) and (b) are the exact grids from the CFD solver and those from gappy POD method, respectively. It can be seen that the grids from gappy POD method are very close to the exact ones. Even in the case of the GOE 117 airfoil whose geometry is very different from the baseline, the grids generated by gappy POD in Figure 5.26(b) are in good agreement with the exact ones in Figure 5.26(a). It has been shown that grids around an airfoil can be generated by gappy POD method. The resulting grids are very close to the exact ones even if there are significant changes in geometries between given and baseline airfoils.

5.5.2 Prediction of Flowfield around an Arbitrary Airfoil

This section presents some examples in predicting the aerodynamic flow field around an arbitrary airfoil as derived in section 4.5. The first example in this section is the case when the coordinates of the Korn airfoil are the available data while the pressure field is missing data. The ensemble in the previous section is used and again 29 POD basis functions are used for predicting pressure distribution. Although the Korn airfoil has different thickness and camber and hence is not a member of snapshots, it is quite similar to the RAE 2822 airfoil. As a result, in Figure 5.27(a), the pressure coefficients on the surface of the Korn airfoil match the exact ones generated from the flow solver accurately except a very small region on the upper surface near the leading edge. Figure 5.27(c) shows that the predicted pressure contours are quite close to the exact contours computed from the CFD solver in Figure 5.27(b).

Next, pressure field predictions are done for NACA 63212, HQ 2010 and GOE 117 airfoils with 31 POD basis functions. The order of these airfoils implies small to big changes in geometry compared to the baseline airfoil RAE 2822. As expected, Figure 5.28(a) shows that the predicted pressure coefficients for NACA 63212 airfoil are very close to the exact ones while there are some small discrepancies in the resulting pressure coefficients in Figure 5.28(b) for HQ 2010 airfoil and big discrepancies in the resulting pressure coefficients in Figure 5.28(c) for GOE 117 airfoil.

These examples show that Gappy POD can be used as an efficient tool for predicting flowfields for an arbitrary airfoil. This method may be useful for guiding direct airfoil design methods which involve the specification of a section geometry and the calculation of pressures and performance. One evaluates the given shape and then modifies the shape to improve the performance.

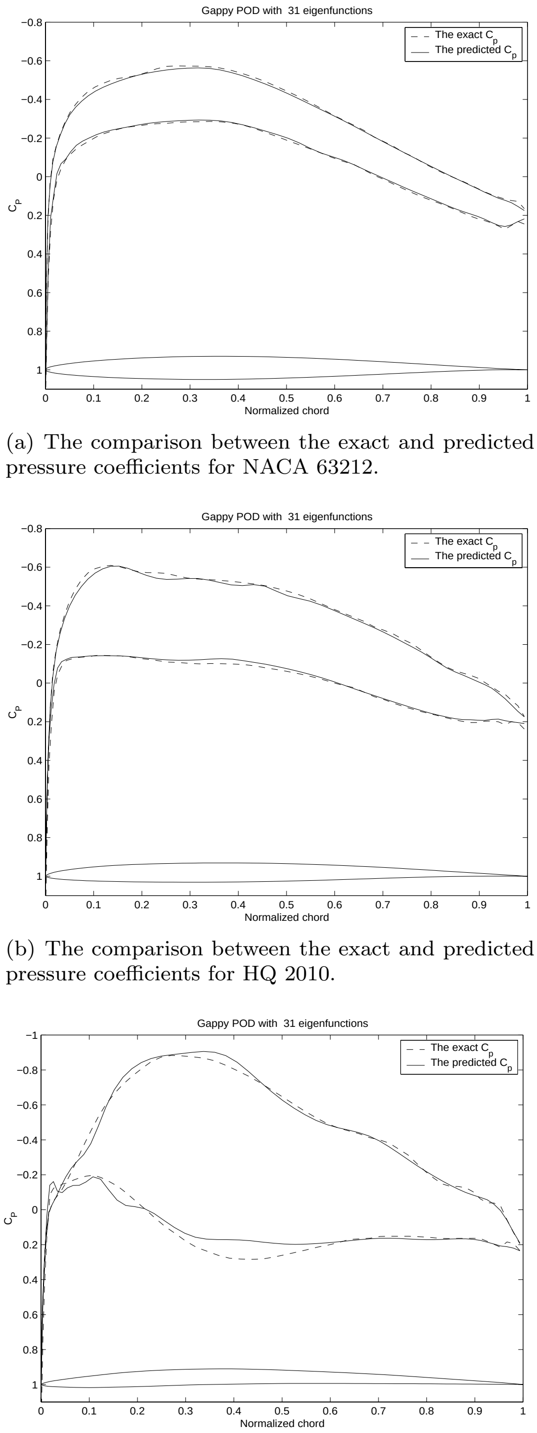


(a) The comparison between the exact and predicted pressure coefficients.

(b) The exact pressure contours.

(c) The predicted pressure contours.

Figure 5.27: The pressure field prediction with 29 POD modes for Korn airfoil from RAE based snapshots.



(a) The comparison between the exact and predicted pressure coefficients for NACA 63212.

(b) The comparison between the exact and predicted pressure coefficients for HQ 2010.

(c) The comparison between the exact and predicted pressure coefficients for GOE 117.

Figure 5.28: The pressure field predictions for NACA 63212, HQ 2010 and GOE 117 airfoils from RAE based snapshots.

Chapter 6

Conclusions and Recommendations

A challenge arises in steady aerodynamics is that how the steady solution can be obtained efficiently. Although CFD turns out to be a useful tool to provide an accurate result, it is high order and expensive in most of the cases in aerodynamics. Reconstruction of incomplete data in experiments or damaged data (due to technical or natural circumstances some data is marred) is another challenge in time-independent aerodynamics. Recent effort in characterization of human faces has been successful in reconstruction of incomplete or marred faces. One of the well known problem in time-independent aerodynamics may be the inverse airfoil design in which an appropriate airfoil is found from a given pressure distribution. A number of methods have been designed to solve this problem but it is still open for a new method. Attempts to deal with these challenges have been proposed in this thesis.

The POD basis has been shown to be efficient for capturing the relevant flow information for steady transonic aerodynamic applications in which the flow solution depends on the values of some varying parameters such as the angle of attack or Mach number. By coupling the POD basis with an interpolation method, models are obtained that give accurate flow field predictions. These predictions do not require a projection onto the CFD governing equations which may be difficult and expensive, but rather just a collection of flow snapshots that covers the parameter ranges of interest, hence they can be obtained very fast. The interpolation approach is applicable to any problem whose properties of interest are a smooth function of the parameters under consideration.

The PODI method has been employed for steady problems so far, one of the extensions could be the application of the PODI method for obtaining a fast computation for parameter-dependent unsteady aerodynamic in which angle of attack or Mach number and time are varying parameters. One of the possible ways is to take the snapshots at discrete point of time directly from nonlinear unsteady simulation for PODI model, and then PODI will predict the solution at any point of time. However, such a way is expensive and may not be suitable since the unsteady simulation needs to be performed beforehand.

The gappy POD has been applied to a number of aerodynamic applications. In particular, the method has been shown to be very effective for reconstructing flow fields from incomplete aerodynamic data sets. This approach may be useful in many real applications where experimental and computational results must be combined. While the rate of convergence of the reconstruction depends on the amount of missing data and the structure of the flow field, the method was found to work effectively for all problems considered. Ongoing research is investigating the question of optimal sensor placement, that is, how many sensors which are really necessary and where to put them on the airfoil surfaces such that the whole flow field can be reconstructed accurately.

A new extension of the methodology has also been proposed for inverse design of airfoil shapes. Given a database of airfoil shapes and pressure distributions, it has been shown that the gappy POD approach can be used to design an airfoil to match a specified pressure target. Even when the geometry of the target airfoil is different significantly to those contained within the original database, a systematic restart procedure can be used to obtain accurate results. While demonstrated for aerodynamic problems in this thesis, the Gappy POD can be applied straightforward to the other areas.

It is clear that the geometry disturbances provided by Hicks-Henne bump functions may not be the best way to generate the airfoil geometry subspace. It is expected that the bigger the subspace is the better the gappy POD result is and hence less restarts need to be performed as well as the design task will be less expensive. One possible way is to vary the parameters in the Hicks-Henne function to find the best combination but it may be impossible to test all the values of these parameters since they can be any positive real numbers. Another possible way is to use another kind of bump function for disturbing the airfoil geometry [30].

The research has shown that although the flowfield prediction for an arbitrary airfoil can be solved efficiently by the gappy POD method, some improvement needs to be done to obtain an accurate result in the case when the geometry of given airfoil is significantly different from those in the ensemble of snapshots. The result can be improved if further snapshots are added to the ensemble; however, this implies some a priori knowledge of the desired result so that appropriate snapshots may be chosen. The gappy POD method for predicting flowfields around an arbitrary airfoil may be useful for guiding direct airfoil design methods which involve the specification of a section geometry and the calculation of pressures and performance. One evaluates the given shape and then modifies the shape to improve the performance.

It has been shown that the constrained airfoil design optimization turns out to be very simple in the context of POD-based method in the sense that the result can be obtained very quickly without running the CFD solver again. However, the method does not guarantee to give a physical airfoil. Further development needs to be incorporated in this approach to maintain its advantage and to lead to a physical solution as well.

While the gappy POD method has been useful for experiment and steady aerodynamic, it may be extended for unsteady aerodynamic. For example, in the supersonic engine inlet problem, the propulsive efficiency will decrease and the drag will increase if the shock stands in front of the engine throat. Hence it is necessary to know the position of the shock in order to make sure that it is right behind the engine throat through some control mechanism when there are some upstream disturbances. In this case, the upstream disturbances can be treated as the available data while the shock position is the missing data. Therefore, with a given upstream disturbance, investigating the gappy POD method to see whether it can predict the shock position is an interesting task for future work.

While in this thesis, the methods are demonstrated for two-dimensional inviscid compressible flow past an airfoil, it can be applied straightforwardly to both two-dimensional and three-dimensional viscous flows.

Bibliography

[1] Newman, A. J., “Modeling and Reduction with Application to Semiconductor Processing” Ph.D thesis, University of Maryland, College Park, 1999

[2] LeGresley, P.A. and Alonso, J.J., “Investigation of Non-Linear Projection for POD Based Reduced Order Models for Aerodynamics”, AIAA Paper 2001-0926, presented at 39th Aerospace Sciences Meeting and Exhibit, Reno, NV, January 2001.

[3] Holmes, P., Lumley, J. L., and Berkooz, G., “Turbulence Coherent structures, Dynamical Systems and Symmetry” Cambridge University Press, 1996.

[4] Antoulas, A.C., Sorensen, D.C., and Gugercin, S., “A survey of model reduction methods for large scale systems”, Contemporary Mathematics, AMS Publications, 280: 193-219, 2001.

[5] Chen Yong, “Model Order Reduction for Nonlinear Systems”, Master’s thesis, Massachusetts Institute of Technology, 1999.

[6] Sirovich, L., “Turbulence and the Dynamics of Coherent Structures, Part 1: Coherent Structures” Quarterly of Applied Mathematics, Vol. 45, No. 3, October 1987, pp. 561-571.

[7] Dowell, E.H. and Hall, K.C. and Thomas, J.P. and Florea, R. and Epureanu, B.I. and Heeg, J., "Reduced Order Models in Unsteady Aerodynamics", AIAA Paper 99-1261, 1999.

[8] Hall, K. C. and Thomas, J. P. and Dowell, E. H., "Reduced-Order Modeling of Unsteady Small-Disturbance Flows Using a Frequency-Domain Proper Orthogonal Decomposition Technique", AIAA Paper 99-0655, 1999.

[9] Romanowski, M.C.,"Reduced Order Unsteady Aerodynamic and Aeroelastic Models using Karhunen-Loéve Eigenmodes", AIAA Paper 96-194, 1996.

[10] P. Beran and W. Silva, "Reduced-Order Modeling: New Approaches for Computational Physics", AIAA Paper 2001-0853, 2001.

[11] Lassaux, G., "High-Fidelity Reduced-Order Aerodynamic Models: Application to Active Control of Engine Inlets.", Master's thesis, Massachusetts Institute of Technology, 2002.

[12] Kim,T.,"Frequency-Domain Karhunen-Loeve Method and Its Application to Linear Dynamic Systems", AIAA Journal, vol. 36, no. 11, pages 2117-2123, 1998.

[13] Willcox, K.E. and Paduano, J.D. and Peraire, J. and Hall, K.C., "Low Order Aerodynamic Models for Aeroelastic Control of Turbomachines", AIAA Paper 99-1467, 1999.

[14] B. I. Epureanu, Dowell, E. H. and Hall, K.C., "A Parametric Analysis of Reduced Order Models of Potential Flows in Turbomachinery Using Proper Orthogonal Decomposition", 2001-GT-0434, Proceedings of ASME TURBO EXPO 2001, New Orleans, Louisiana, June 2001.

[15] Hung V. Ly and Hien T. Tran, “Modeling and Control of Physical Processes using Proper Orthorgonal Decomposition”, Journal of Mathematical and Computer Modeling, Feb 12 1999.

[16] Kirby, M., and Sirovich, L., “Application of the Karhunen-Loeve Procedure for the Characterization of Human faces”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 12, No. 1, January 1990.

[17] R. Everson and L. Sirovich, “The Karhunen-Loeve for Gappy Data”, J.Opt.Soc.Am., 12: 1657-1664, 1995.

[18] Lighthill, M.J., “A New Method of Two-dimensional Aerodynamic Design”, ARC, Rand M 2112.

[19] McFadden, G.B., “An Artificial Viscosity Method for the Design of Supercritical Airfoils”, New York University report No. C00-3077-158.

[20] Hicks, R.M. and Henne, P.A., “Wing Design by Numerical Optimization.”, Journal of Aircraft, 15:407-412, 1978.

[21] Jameson, A., “Aerodynamic Design via Control Theory”, Journal of Scientific Computing, 3:233-260,1988.

[22] Jameson, A., Schmidt, W. and Turkel, E., “Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes”, AIAA Paper 81-1259 14th Fluid and Plasma Dynamics Conference, Palo Alto, California, 1981.

[23] Damodaran, M., “Finite Volume Computation of Unsteady Inviscid Rotational Transonic Flows Past Airfoils in Rigid Body Motion”, AIAA Paper 88-0006, 26th Aerospace Sciences Meeting, Reno, Nevada, 1988.

[24] Jameson, A., “Solution of the Euler Equations for Two Dimensional Transonic Flow by a Multigrid Method.”, MAE Report No. 1613, June 1983.

[25] Liang, Y.C., Lee, H.P., Lim, S.P., Lin, W.Z., Lee, K.H. and Wu, C.G., “Proper Orthogonal Decomposition and its Applications-Part I: Theory”, Journal of Sound and Vibration (2002) 252(3), page 527-544.

[26] Eugene Wong, “Stochastic Processes in Information and Dynamical Systems”, McGraw-Hill, 1971.

[27] http://www.aae.uiuc.edu/m-selig/ads/coord_database.html#M

[28] Sundar Krishnamurty, “Advanced Mechanical Engineering Design II.”, MIE 761 Lecture Notes, Spring 1998.

[29] Robert Fourer, David M. Gay, and Brian W. Kernighan,“AMPL: A Modeling Language for Mathematical Programming”, Brooks/Cole Publishing Company, 2002.

[30] http://www.ae.gatech.edu/ lsankar/AE3903/9