UNIVERSITY OF CALIFORNIA, SAN DIEGO

New algorithms for the direct numerical simulation of turbulent flows past compliant bodies and the optimization of highly constrained PDE systems

A dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy
in
Engineering Sciences (Mechanical Engineering)

by

Anish Avinash Karandikar

Committee in charge:

Professor Thomas R. Bewley, Chair
Professor Philip E. Gill
Professor Michael Holst
Professor Alison L. Marsden
Professor Constantine Pozrikidis

2008

The dissertation of Anish Avinash Karandikar is approved, and it is acceptable in quality and form for publication on microfilm:

Chair

University of California, San Diego

2008

DEDICATION

Dedicated with love and gratitude to my mother and father,
my guiding light.

EPIGRAPH

Jo Kachu Kiya so Tum Kiya, Mein Kuch Kiya Nahi
Kahe Koi Jo Mein Kiya, Tum Hi The Mujh Mahi

Whatever was done, was done by You, nothing by me
Whatever is said done by me, 'twas You dwelling in me

- Sant Kabir, The Mystic Poet

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

PREFACE

The effect of a deformable compliant surface wrapped around a bluff body in an incompressible flow is explored in this thesis. To study the interaction between the flow and the compliant surface, we consider a canonical bluff body and an infinitely long circular cylinder wrapped with a compliant surface. Flow past the compliant cylinder is studied numerically using direct numerical simulation (DNS) for both laminar and turbulent flow. Incompressible flow past a compliant cylinder for both 2D and 3D flows is described in the first two chapters.

Chapter I: Accurate simulation of unsteady 2D flow past a compliant cylinder

In this chapter, results for the laminar flow are presented. Simulations were performed at a Reynolds number of 80 based on the cylinder diameter and the free stream velocity. At this Reynolds number, the von Karman vortex shedding pattern is observed in the wake of the cylinder. This flow is laminar, two-dimensional and unsteady, with vortices generated alternately in the wake region. This flow pattern is used to validate the code in the laminar 2D case. Further, statistics for flow past a compliant cylinder are presented for different values of surface compliance. For modeling the compliant surface, a simple spring-mass-damper system is used which is governed by a decoupled 2nd order ordinary differential equation for each grid point. As the surface is made more compliant, a decrease in the peak lift coefficient and an increase in the Strouhal number and time average drag are observed.

Chapter II: Direct numerical simulation of turbulent 3D flow past a circular cylinder with a compliant surface

In this chapter, results for turbulent flow past a circular cylinder at Reynolds number 300 are presented. At a Reynolds number of approximately

180, flow past a cylinder becomes three-dimensional, and at a Reynolds number of 300, fully turbulent 3D vortex shedding in the cylinder wake is observed. The Reynolds number 300 case is used for validation of the 3D turbulent code. Flow statistics for turbulent flow past a cylinder with a compliant surface are also presented. As in the 2D case, a spring-mass-damper system is used to model the compliant surface. Similar to the 2D case, a drop in the peak lift coefficient and an increase in time average drag is seen as the surface is made compliant.

Chapter III: An iterative left/right LP method for the efficient optimization of a highly constrained resistor network

In this chapter, a new method to optimize a highly constrained PDE system is introduced. Optimization of a variable channel width/porosity liquid cooled heat exchanger, modeled by a electrical network of resistors was studied and a solution method was developed that is scalable and passes efficiently to the PDE limit. The full problem is a quadratically constrained quadratic program (QCQP). By fixing certain variables and allowing others to be free, the full problem can be cast into two alternate linear programs (LP). This method scales well with problem dimension and provides us with a tool for optimizing other similar large scale highly constrained PDE system optimization problems.

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and gratitude to my advisor, Professor Thomas Bewley. He is truly a man of vision, intellect and insight and I counted on his wisdom to guide me through my dissertation and in my studies at UC San Diego. Trite as it may sound, the simple fact remains that I could not have pulled it off without his guidance.

My family have been a pillar of support throughout this endeavor. In beautiful La Jolla, I found a home away from home, and in my fellow graduate students I found a family away from family. I would especially like to mention my close friends here at UCSD, Suchit Jhunjhunwala, Himanshu Bal, Manmohan Chandraker, Kiran Kalyan, Manish Amde and Saumya Chandra; who all made this excursion a plenteous and mirthful one.

I am thankful to my lab seniors, Haoxiang Luo and Laura Cervino, who helped me out in every possible way in my earlier years in the lab. Thanks are also due to my colleagues Paul Belitz, Chris Colburn, Joe Cessna and Brett Nadler, who have helped and supported me time and again in my work.

I gratefully acknowledge the Office of Naval Research and the Department of Mechanical and Aersopace Engineering for financial support during this work.

Chapter 1, in full, has been submitted for publication to Journal of Computational Physics as, Anish Karandikar and T. R. Bewley, *Accurate simulation of unsteady 2D flow past a compliant cylinder*.

Chapter 2, in full, has been submitted for publication to Journal of Computational Physics as, Anish Karandikar and T. R. Bewley, *Direct numerical simulation of turbulent 3D flow past a circular cylinder with a compliant surface*.

Chapter 3, in full, has been submitted to SIAM Journal of Control and Optimization as, Anish Karandikar and T. R. Bewley, *An iterative left/right LP method for the efficient optimization of a highly constrained resistor network*.

VITA

2003	B. E. in Mechanical Engineering Government College of Engineering, Pune (COEP), India
2005	M. S. in Engineering Sciences (Mechanical Engineering) University of California San Diego, USA
2008	Ph. D. in Engineering Sciences (Mechanical Engineering) University of California, San Diego, USA

PUBLICATIONS AND PRESENTATIONS

Accurate simulation of unsteady 2D flow past a compliant cylinder, Submitted to *Journal of Computational Physics*

Direct numerical simulation of turbulent 3D flow past a circular cylinder with a compliant surface, Submitted to *Journal of Computational Physics*

An iterative left/right LP method for the efficient optimization of a highly constrained resistor network, Submitted to *SIAM Journal of Control and Optimization*

DNS of turbulent flow past a bluff body with a compliant tensegrity surface, Presented at *American Physical Society, Division of Fluid Dynamics, 60th Annual Meeting, 2007*

Multigrid-accelerated DNS of Turbulent Flows past Bluff Bodies with Compliant Surfaces, Presented at *UCSD Jacobs School of Engineering, Annual Research Expo, 2006*

WORK EXPERIENCE

The Mathworks Inc
Intern in Quality Engineering (Control Systems Toolbox)
June 2006 to September 2006

Department of MAE, UC San Diego
Graduate Student Researcher
January 2005 to June 2006

Department of MAE, UC San Diego
Teaching Assistant
Graduate level courses in Numerical Methods, Computational Fluid Dynamics and Thermodynamics

ABSTRACT OF THE DISSERTATION

New algorithms for the direct numerical simulation of turbulent flows past compliant bodies and the optimization of highly constrained PDE systems

by

Anish Avinash Karandikar
Doctor of Philosophy in Engineering Sciences (Mechanical Engineering)
University of California, San Diego, 2008
Professor Thomas R. Bewley, Chair

This work develops an efficient and accurate new method for direct numerical simulation of laminar and turbulent flow past a circular cylinder with a deformable (compliant) surface. It studies the interaction of the incompressible flow with the compliant cylinder. From the outset, this is defined as an optimization problem, in which we seek to minimize aeroacoustic noise generated by dipole sound sources on the compliant surface at low Mach numbers.

We build on a unique method developed in our lab for simulating turbulent flow in a channel with compliant walls. This method is accurate and efficient for large surface deformations. We adapt this method for the cylindrical polar coordinate system to study flow past a compliant cylinder. In this method, a time-dependent coordinate transformation is used to map the deformed flow domain to a regular computational domain. The governing Navier Stokes equations are formulated in the cylindrical polar form and not the contravariant form, as the latter is computationally expensive to simulate. The compliant surface is modeled by a simple spring-mass-damper system.

As surface compliance is increased, a decrease in the peak lift coefficient

for the compliant cylinder is observed both in the laminar 2D case at $Re = 80$, as well as the turbulent 3D case at $Re = 300$. On the other hand, the frequency of vortex shedding and the time-average drag both increase with surface compliance.

This work also develops a new method for optimizing highly-constrained PDE systems by splitting up the governing equations into parallel linear programs, thus achieving scalability. It explores optimization of a single-phase fluid heat exchanger to minimize the power required to drive coolant through it by appropriately adjusting the channel width or channel porosity. The Stokes flow in the heat exchanger is modeled as a resistor network, while the flow rate and pressures in the flow are analogous to currents and voltages in the resistor network. The method developed and demonstrated on the resistor network problem extends naturally to the optimization of the variable channel width/porosity distribution in the heat exchanger model.

I

Accurate simulation of unsteady 2D flow past a compliant cylinder

I.A Introduction

In this paper, we develop an efficient and accurate method to perform two dimensional Direct Numerical Simulation (DNS) of flow past a circular cylinder with a compliant, deformable surface. The surface and flow interact with each other causing flow-structure interaction. We start by defining the problem in an optimization context, in which, we aim to minimize the aeroacoustic noise generated by the flow past a compliant cylinder by appropriately changing the surface compliance parameters. Section I.B describes the problem setup in detail including the optimization cost function definition and the compliant surface model used for this simulation. In section I.C we describe the equations that govern the incompressible flow and the method used to transform the irregular physical domain to a regular computational domain. Section I.D describes the numerical algorithm used to perform DNS of the flow. In section I.E, we describe the results for validation of the code for a solid cylinder case with compliance turned off for Reynolds number 80. At this Reynolds number, flow past a circular cylinder is unsteady and two dimensional with alternate vortex shedding in the wake. Finally, in section III.F we summarize the results of the DNS performed for the compliant surface cases at Reynolds number 80 for different surface compliance parameter regimes.

I.A.1 Turbulent flow in a channel with compliant walls

Turbulent flow in a channel with deformable walls has been extensively studied by, for example Carlson et al (15) and Mito et al (16). The presence of a compliant surface has been proven to delay laminar-to-turbulent transition in channel flow. This has been studied analytically by Carpenter et al in (19), (20) & (21) using linear stability methods. But the linear stability theory does not shed much light on the fully turbulent regime, and so numerical methods must be used for simulation of turbulent flow in compliant channels.

In (12), Endo et al performed direct numerical simulation of turbulence in a channel with a compliant surface and reported 2 to 3% drag reduction. This result was challenged by Rempfer et al in (3) for the reason of insufficient averaging time. Rempfer et al found no discernible impact of wall compliance on long term turbulence statistics. Our lab has developed an alternative numerical scheme (see (1) and (2)) for DNS of turbulent flow in a channel with compliant walls. This scheme differs in some significant details from previous methods, and is stable and accurate even for large deformations of the compliant channel walls. The results obtained in our study were consistent with those of Rempfer et al for small wall deformations. A new structural paradigm called a tensegrity fabric was also introduced in this work to model the compliant channel walls. Unfortunately, a parameter regime for the wall compliance properties has not yet been identified using this code that indicates the possibility for sustained drag reduction in fully developed turbulent channel flow using a compliant surface.

The present work extends the method developed in (1) to cylindrical polar coordinates. We handle the arbitrarily moving bluff body surface with care and accuracy to simulate flow past a compliant cylinder.

I.A.2 Two dimensional flow past a compliant cylinder

As mentioned above, this work performs DNS of two dimensional flow past a compliant cylinder. We use a velocity-pressure formulation, with a time-dependent domain transformation to handle the moving cylinder surface. Spatial differentiation is performed using a Discrete Fourier Transform in the azimuthal direction, while radial derivatives are computed using a second order central finite difference scheme. Temporal integration is performed using a low storage RKW3/CN scheme, with certain wall normal derivatives handled implicitly, and other terms explicitly. The fractional step method is used to project the intermediate velocity field to a divergence-free subspace at the end of each RK substep. The resulting Poisson equation is solved in Fourier space iteratively.

In a larger context, this is set up as an optimization problem to minimize the aeroacoustic noise produced by the dominant dipole sources on the surface, similar to (5) and (14). In future work, we plan to use a derivative-free optimization method based on the Surrogate Management Framework (SMF) (18) to minimize the noise generated by the flow past the compliant bluff body. This method is especially suited to optimize problems with expensive function evaluations. Our method extends the idea of n-dimensional sphere packing to increase efficiency of the optimization routine.

I.B Problem statement

To set the stage for follow-on work, we will pose the present formulation in an optimization context. We consider two-dimensional incompressible flow past a nominally circular cylinder. The cylinder has a solid core (See Figure I.1) and is surrounded by a compliant surface. The flow interacts with the compliant surface on top of the cylinder and vortex shedding occurs in the wake of the cylinder. The imposed inflow boundary conditions are uniform flow from left to right. The flow structure interaction and the flow itself produce aeroacoustic noise. For an acoustically compact surface, the noise generated is governed by Curle's extension to Lighthill's acoustic analogy. The cost function in the optimization is an integral of the source terms in Lighthill's equation on the surface of the cylinder. The independent parameters to be optimized are physical properties of various regions of the compliant surface that determine how soft any particular region of the compliant surface is. The surface is modeled by a simple spring-mass-damper type system with associated stiffness and damping coefficients.

The main part of this problem is accurate simulation of the flow in the wake of the compliant cylinder. The boundary conditions for this flow at the cylinder wall are dynamic, and have to handled correctly. We use a domain transformation method to simulate the flow, by first transforming the governing Navier

Stokes equations together with the continuity equation to the computational domain while incorporating the moving irregular boundary of the domain.

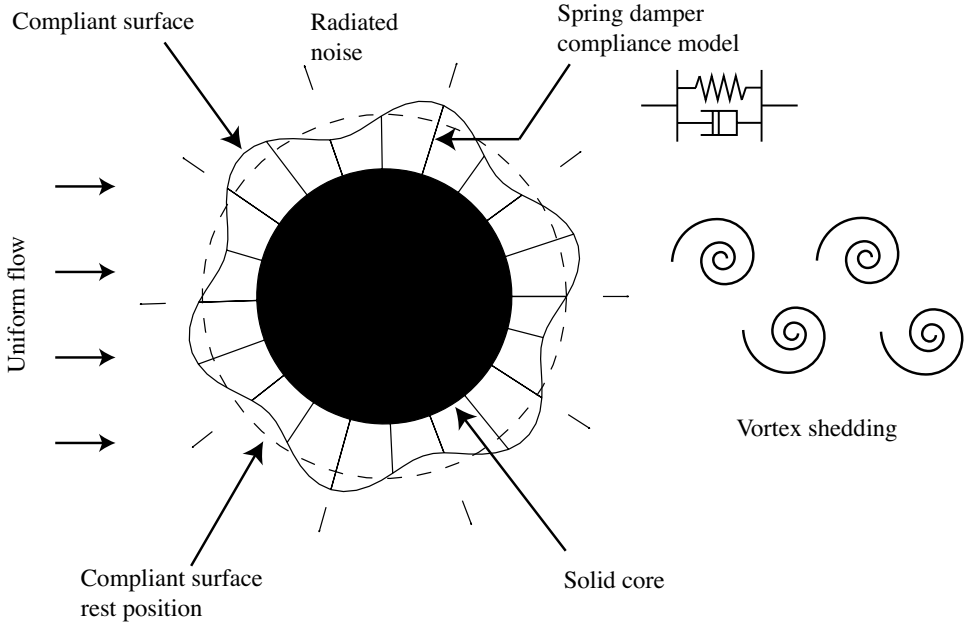

Figure I.1: Schematic of problem statement

I.B.1 Noise generation

We first begin by examining the cost function that we are optimizing. According to Ffowcs-William and Hawkings, the density fluctuation (ρ') for an observer locate at position $\mathbf{x}$ in the presence of arbitrarily moving walls is,

$$\rho'(\mathbf{x},t) = \frac{1}{4\pi c_0}\left[\frac{\partial^2}{\partial x_i \partial x_j}\int\int_{\nu(\tau)}\left(\frac{T_{ij}}{r|D|}\right)_{\tau e} d\sigma - \frac{\partial}{\partial x_i}\int\int_{s(\tau)}\left(\frac{f_i}{r|D|}\right)_{\tau e} ds(\sigma) - \frac{\partial}{\partial x_i}\int\int_{\nu c(\tau)}\left(\frac{\rho_0 a_i}{r|D|}\right)_{\tau e} d(\sigma) + \frac{\partial^2}{\partial x_i \partial x_j}\int\int_{\nu c(\tau)}\left(\frac{\rho_o v_i v_j}{r|D|}\right)_{\tau e} d\sigma\right]$$

where, T_{ij} is the Lighthill tensor, $\nu(\tau)$ is the source region, f_i is the force due to interaction of the flow with the moving surface $s(\tau)$, and a_i and v_i are acceleration and velocity of the volume $\nu_c(\tau)$ enclosed by surface $s(\tau)$. The distance between acoustic source and observer is $\mathbf{r}$, while D is the Doppler factor

given by,

$$D = 1 - \frac{\mathbf{r}}{c_0 r}\frac{\partial \mathbf{y}}{\partial \tau}$$

See (4) for further details. For subsonic flows, the noise generation is dominated by dipole sources on the surface as shown in (5) and (14). In this case, the acoustic density (ρ) can be expressed as,

$$\rho \approx \frac{M^3}{4\pi}\frac{y_i}{|\mathbf{y}|^2}\frac{\partial}{\partial t}\int_S n_j p_{ij}(\mathbf{x}, t - M|\mathbf{x}|)d^2\mathbf{x}$$

where, M is the free stream Mach number, n_j is the direction cosine of the outward normal to the cylinder surface S, and $p_{ij} = p\delta_{ij} - \tau_{ij}$ is the compressive stress tensor.

Thus, a cost function proportional to power of radiated acoustic noise per unit span of the cylinder can be stated as,

$$J_1 = \overline{\left(\frac{\partial}{\partial}\int_S n_j p_{1j}(\mathbf{x}, t)d^2\mathbf{x}\right)^2} + \overline{\left(\frac{\partial}{\partial}\int_S n_j p_{2j}(\mathbf{x}, t)d^2\mathbf{x}\right)^2}$$

where, the overbar indicates time-averaged values, after the flow has reached statistical steady state.

I.B.2 Compliant surface model

The compliant surface is modeled by a simple spring-mass-damper system. Each point on the compliant surface is assumed to be attached to a spring-damper system which is only allowed to deform in the wall normal (radial) direction (See Figure I.2). The dashed line shows the nominal (rest) position of the compliant surface, with no forces acting on it, and the solid line is the compliant surface.

The spring-damper system has an associated stiffness coefficient (k), damping coefficient (c) and mass (m). The governing equation for dynamics of this system is given by a second order decoupled ODE for each grid point on the surface of the cylinder,

$$m\frac{\partial^2 \eta}{\partial t^2} + c\frac{\partial \eta}{\partial t} + k\eta = f$$

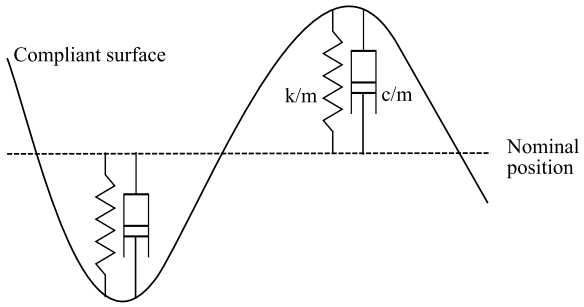

Figure I.2: Compliant surface model

where f is the external force applied to the system and η is the diplacement of the compliant surface from the nominal (rest) position. In the present flow structure interaction, this force is generated by the fluid flow, and appears in the form of pressure at the cylinder surface (p').

On dividing by m, the above equation can be written as,

$$\frac{\partial^2 \eta}{\partial t^2} + \left(\frac{c}{m}\right)\frac{\partial \eta}{\partial t} + \left(\frac{k}{m}\right)\eta = \frac{p'}{m}$$

In the code, the dynamics of the flow are simulated for one timestep, then the pressure at the cylinder surface is used as input to compute the dynamics of the compliant surface for one timestep. Then, the position and velocity of the compliant surface are input as boundary condition for the flow solver, and so on back and forth between the flow solver and the structure dynamics solver.

For optimization, the surface of the cylinder is divided into three zones (See Figure I.3). Considering $\theta = 0$ as the downstream stagnation point, with the flow going from left to right, Zone I is defined by $|\theta| \in [2\pi/3, \pi]$. Zone II is defined by $|\theta| \in [\pi/3, 2\pi/3]$ and Zone III is defined by $|\theta| \in [0, \pi/3]$.

The free variables in the optimization are the scaled stiffness coefficient (k/m) and damping coefficient (c/m) for the three zones, defined as,

$$\mathbf{x} = \left[\left(\frac{k}{m}\right)_I, \left(\frac{k}{m}\right)_{II}, \left(\frac{k}{m}\right)_{III}, \left(\frac{c}{m}\right)_I, \left(\frac{c}{m}\right)_{II}, \left(\frac{c}{m}\right)_{III}\right]$$

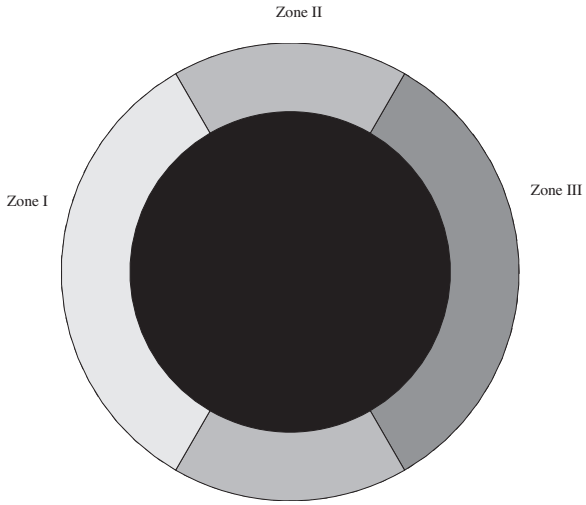

Figure I.3: Zones of compliance model

I.B.3 Cost function definition

The overall cost function for the optimization is defined as a combination of the cost related to noise generation (Section I.B.1), and the compliance parameters (Section I.B.2). The reason for adding compliance parameters to the cost function is to prevent the surface from becoming arbitrarily soft. Thus, the optimization algorithm has a finite amount of compliance (softness) that it needs to distribute appropriately over the surface on the zones of the cylinder. Thus, the negative of the stiffness and damping ratios (k/m) and (c/m) are penalized in the second part of the cost function.

$$J_2 = C - \sum_{i \in [I,II,III]} \left[\left(\frac{k}{m} \right)_i + \left(\frac{c}{m} \right)_i \right]$$

where C is a heuristically determined constant.

Thus, the final cost function is defined as,

$$J = \alpha_1 J_1 + \alpha_2 J_2$$

where, α_1 and α_2 are appropriately chosen weights. Thus, the overall optimization problem is to find a $\mathbf{x}$ that minimizes $J(\mathbf{x})$.

I.B.4 Domain transformation

An important part of this code is the handling of the arbitrarily moving cylinder surface. The physical domain is irregular. We use a time-dependent coordinate transformation similar to (1) to map the irregular physical domain to a regular computational domain. We have developed a new flow solver from scratch which is significantly different from previous works to accurately capture the flow-structure interaction.

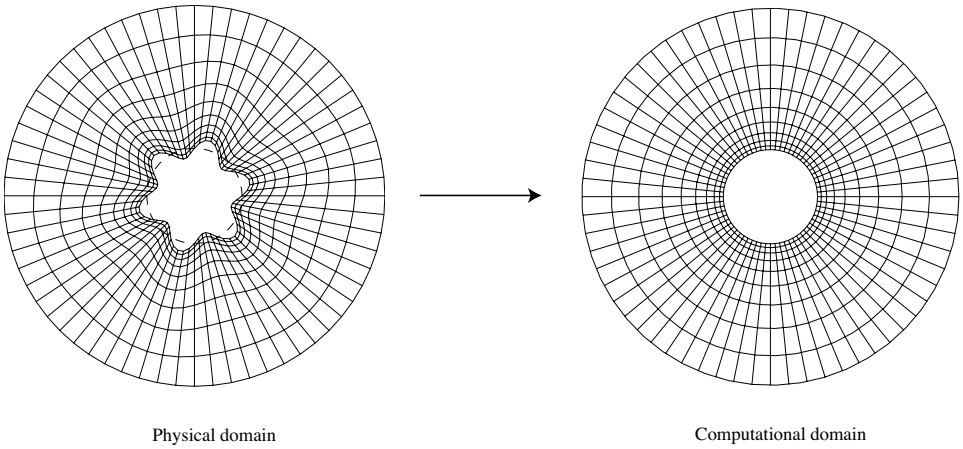

Figure I.4: Domain transformation

Figure I.4 shows the ciruclar cylinder with deformed compliant surface at the left. The dashed line shows the nominal (rest) position of the cylinder. The grid is stretched only in the radial (r) direction. Due to this nonorthogonal coordinate transformation, the contravariant formulation of the Navier Stokes equations is very involved, and computationally expensive to compute. So, in this present work, as in (1), we use a simple cylindrical polar coordinate system formulation as detailed in Section I.C. The regular computational domain is shown in Figure I.4 on the right. Similar to (1), the wall parallel derivatives in the azimuthal direction are computed spectrally, using the Discrete Fourier Transform. In the wall normal (radial) direction, the derivatives are computed using a second-order finite difference scheme. This allows us to treat most of the derivatives in the radial

direction implicitly in the time integration scheme, thus increasing the stability of the code in the presence of wall fluctuations.

I.C Governing Equations

The governing equations are the incompressible Navier Stokes equations together with the continuity equation in cylindrical polar coordinates.

$$\begin{aligned}\frac{\partial u_r}{\partial t} &= -\frac{1}{r}\frac{\partial\left(ru_ru_r\right)}{\partial r}+\frac{u_\theta^2}{r}-\frac{1}{r}\frac{\partial(u_ru_\theta)}{\partial\theta}-\frac{2\nu}{r^2}\frac{\partial u_\theta}{\partial\theta}+\frac{\nu}{r^2}\frac{\partial^2u_r}{\partial\theta^2} \\ &+ \nu\left\{\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial u_r}{\partial r}\right)-\frac{u_r}{r^2}\right\}-\frac{1}{\rho}\frac{\partial p}{\partial r}\end{aligned} \tag{I.1}$$

$$\begin{aligned}\frac{\partial u_\theta}{\partial t} &= -\frac{1}{r}\frac{\partial\left(ru_ru_\theta\right)}{\partial r}-\frac{u_ru_\theta}{r}-\frac{1}{r}\frac{\partial(u_\theta u_\theta)}{\partial\theta}+\frac{2\nu}{r^2}\frac{\partial u_r}{\partial\theta}+\frac{\nu}{r^2}\frac{\partial^2u_\theta}{\partial\theta^2} \\ &+ \nu\left\{\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial u_\theta}{\partial r}\right)-\frac{u_\theta}{r^2}\right\}-\frac{1}{\rho}\frac{1}{r}\frac{\partial p}{\partial\theta}\end{aligned} \tag{I.2}$$

$$\nabla\cdot\mathbf{u}=\frac{1}{r}\frac{\partial(ru_r)}{\partial r}+\frac{1}{r}\frac{\partial u_\theta}{\partial\theta}=0 \tag{I.3}$$

The coordinate transformation is nonorthogonal, as the grid is deformed only in the radial direction. As defined in Section I.B.2, $\eta(\theta,t)$ is the displacement of the compliant cylinder surface from its nominal position, the following time-dependent coordinate transformation may be used to map the irregular physical onto a regular cylindrical domain

$$\begin{aligned}r &= (\xi_r-R_I)\cdot\left[\frac{R_O-(R_I+\eta)}{R_O-R_I}\right]+(R_I+\eta) \\ \theta &= \xi_\theta \\ t &= \tau\end{aligned}$$

where (r,θ) denote cylindrical polar coordinates, and (ξ_r,ξ_θ) denote transformed coordinates in the computational domain. R_I is the nominal position of the cylinder surface at rest, and R_O is the outer radius of the computational domain. The Jacobian of the spatial transformation, and its determinant are,

$$C = \frac{\partial \mathbf{x}}{\partial \xi} = \begin{pmatrix} \frac{\partial r}{\partial \xi_r} & \frac{\partial r}{\partial \xi_\theta} \\ \frac{\partial \theta}{\partial \xi_r} & \frac{\partial \theta}{\partial \xi_\theta} \end{pmatrix} = \begin{pmatrix} \frac{\partial r}{\partial \xi_r} & \frac{\partial r}{\partial \xi_\theta} \\ 0 & 1 \end{pmatrix}$$

$$J = |C| = \frac{\partial r}{\partial \xi_r} = \frac{R_O - (R_I + \eta)}{R_O - R_I}$$

The inverse spatial transformation is,

$$C^{-1} = \frac{\partial \xi}{\partial \mathbf{x}} = \begin{pmatrix} \frac{1}{J} & -\frac{1}{J}\frac{\partial r}{\partial \xi_\theta} \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \frac{\partial \xi_r}{\partial r} & \frac{\partial \xi_r}{\partial \theta} \\ \frac{\partial \xi_\theta}{\partial r} & \frac{\partial \xi_\theta}{\partial \theta} \end{pmatrix}$$

Defining the nontrivial elements of C^{-1} as,

$$\begin{aligned}
\phi_r &= 1/J = 1 \Big/ \left(\frac{\partial r}{\partial \xi_r}\right) = \frac{R_O - R_I}{R_O - (R_I + \eta)} \\
\phi_\theta &= \frac{\partial \xi_r}{\partial \theta} = -\frac{1}{J}\frac{\partial r}{\partial \xi_\theta} = -\frac{R_O - R_I}{R_O - (R_I + \eta)} \cdot \left(\frac{R_O - \xi_r}{R_O - R_I}\right) \eta_\theta \\
&= -\frac{R_O - \xi_r}{R_O - (R_I + \eta)} \eta_\theta \\
\phi_t &= \frac{\partial \xi_r}{\partial t} = -\frac{1}{J}\frac{\partial r}{\partial t} = -\frac{R_O - R_I}{R_O - (R_I + \eta)} \cdot \left(\frac{R_O - \xi_r}{R_O - R_I}\right) \eta_t \\
&= -\frac{R_O - \xi_r}{R_O - (R_I + \eta)} \eta_t
\end{aligned}$$

and using the chain rule, we can apply the substitutions,

$$\begin{aligned}
\frac{\partial}{\partial t} &\rightarrow \frac{\partial}{\partial \tau} + \phi_t \frac{\partial}{\partial \xi_r} \\
\frac{\partial}{\partial \theta} &\rightarrow \frac{\partial}{\partial \xi_\theta} + \phi_\theta \frac{\partial}{\partial \xi_r} \\
\frac{\partial}{\partial r} &\rightarrow \phi_r \frac{\partial}{\partial \xi_r}
\end{aligned}$$

to express the derivatives in (I.1), (I.2) and (I.3) in terms of the new coordinate system. With these substitutions, the transformed r and θ momentum equations, and the conitnuity equation, are written out explicitly in Section I.G.

I.D Numerical algorithm

The numerical algorithm used for solving the unsteady incomspressible Navier Stokes equation is similar to the one used in (1), in which turbulent flow in a channel with deformable walls is simulated using DNS. In (1), the grid was deformed only in the wall normal direction (y), as in this code the grid is deformed only in the radial direction. We use a hybird pseudospectral/finite-difference method for spatial discretization, and a mixed low storage Runge-Kutta-Wray/Crank-Nicholson (RKW3/CN) method for temporal discretization with certain terms of (I.1) and (I.2) treated implicitly using CN, and other terms treated explicitly (see Section I.G for details). In addition to the flow variables, the geometry-related time-varying coefficients, ϕ_r, ϕ_θ, and ϕ_t need to be spatially discretized.

The azimuthal direction is naturally periodic for the O-type grid used. No-slip and no-penetration boundary conditions are assumed at the cylinder surface. Uniform flow is imposed on the upstream side of the outer boundary with a velocity of U_∞, while physical outflow boundary conditions as in (7) are used in the downstream side of the boundary.

A fractional step method is used to advance the flow and pressure fields in time. At each RKW3 substep, an intermediate flow field is computed which is not divergence-free. A Poisson equation with scaled divergence as right hand side is solved, and the solution is used to project the flow field to the divergence free supspace, and to compute the pressure update. The Poisson equation has to be solved in Fourier space, but as the various Fourier modes can not be fully decoupled, the pressure equation needs to be solved iteratively. Details for this algorithm are provided in Section I.D.3.

I.D.1 Spatial discretization

The grid is chosen to be evenly spaced and non-staggered in the azimuthal (ξ_θ) direction so that spatial derivatives can be computed spectrally. In the wall-

normal direction (ξ_r), the grid is staggered and stretched using a hyperbolic tangent function, to give sufficient grid resolution in the near-wall region. The variables u_θ and p are discretized on the family of grid points $j = 0, 1, ..., NR$, where $j = 0$, corresponds to the cylinder surface and $j = NR$ corresponds to the outer edge of the computational domain. The variables u_r, ϕ_r, ϕ_θ and ϕ_t are discretized on the family of grid points $j = \frac{1}{2}, 1 + \frac{1}{2}, ..., NR - \frac{1}{2}$, where gridpoint $j = n + \frac{1}{2}$ is midway between the gridpoints $j = n$, and $j = n + 1$.

With the arrangement of the discrete variables on the numerical grid as described above, the two components of the momentum equation are enforced at the grid points at which the corresponding velocity components are discretized, and the continuity equation is solved on the cells centered at the pressure points.

I.D.2 Temporal discretization

A low storage 3rd-order Runge-Kutta-Wray/Crank-Nicholson scheme is used to advance the flow from time step m to the next time step $m + 1$. At each RK substep k ($k = 1, 2, 3$), a fractional step method as in (8) is used to march the flow and project the flow field onto a divergence-free subspace. Certain terms in the momentum equations are treated implicitly using the CN scheme, while other terms are treated explicitly using the RKW3 scheme. The details of these terms are given in Section I.G.

I.D.3 Fractional step method

During each RK substep, while marching the current velocity field ($\mathbf{u}^k$) to the velocity field at the next time step ($\mathbf{u}^{k+1}$), an intermediate flow field, $\mathbf{u}^*$, is obtained. This velocity field is not divergence free. After this step, a projection equation is solved to make the velocity field divergence free. This projection equation is a Poisson equation with its right hand side given by divergence of the intermediate velocity field scaled by a factor,

$$\nabla^2\phi = \frac{1}{2\beta_k\Delta t}(\nabla \cdot \mathbf{u}^*)$$

The solution to this equation is also the pressure update to get the new pressure field as,

$$p^{k+1} = p^k + \phi$$

Also, the intermediate velocity is projected onto the divergence free space as,

$$\mathbf{u}^{k+1} = \mathbf{u}^* - (2\beta_k\Delta t)\,\nabla\phi$$

The Laplacian, divergence and gradient operators in the transformed coordinate system are defined explicitly in Section I.G.

Numerically, the Poisson equation has to be solved in Fourier space iteratively. This has to be done because the Laplacian, divergence and gradient operators cause coupling of wavenumbers in the azimuthal direction. This happens due to the non-constant coeffiecients ϕ_r and ϕ_θ. We split the Laplacian operator into two parts. The terms with the various Fourier modes decoupled are treated implicitly ($L^{imp}(\cdot)$), while the rest of the terms are treated explicitly ($L^{exp}(\cdot)$), such that, $L = L^{imp} + L^{exp}$. The following algorithm describes the numerical method used to solve the pressure update Poisson equation iteratively in Fourier space.

Algorithm for iterative pressure update Poisson equation solver

- Compute scaled divergence in d
- Set ϕ^0 to zero or initial guess
- For $i = 0 : itmax$
- Compute right hand side, $r = d - L^{exp}(\phi^i)$, in physical space
- Convert r to Fourier space, $\hat{r}$

- Solve $L^{imp}(\hat{\phi}^{i+1}) = \hat{r}$, for $\hat{phi}^{i+1}$
- Convert $\hat{\phi}^{i+1}$ to physical space, ϕ^{i+1}
- Compute residual, $e = d - L(\phi^{i+1})$
- If norm of residual, $|e|$ is less than tolerance, Exit For Loop
- End For Loop
- Set, $\mathbf{u}^{k+1} \rightarrow \mathbf{u}^* - \nabla\phi$
- Set, $p^{k+1} \rightarrow p^k + \phi$

I.E Code validation

Simulations of laminar flow over a circular cylinder with no compliance (solid surface) were performed at a Reynolds number of 80. The Reynolds number was based on free stream velocity (U_∞) and the cylinder diameter ($D = 2\,R_I$). Results from two simulations with different grid sizes are presented here for verification of the code. In Case 1, the grid size chosen was 64×64 grid points in the radial and azimuthal directions. In the Case 2, 128×128 grid points in the radial and azimuthal direction were considered. $Re = 80$ is a well known test case for flow past a circular cylinder, producing alternate positive and negative vortices, in the wake region. Various flow quantities were measured for the two cases, and are presented in Table I.1. For both cases, the nominal cylinder radius was taken as, $R_I = 0.5$, and the radius of the outer edge of the computational domain was taken as, $R_O = 30$.

For the two cases, the parameters considered were mean drag coefficient ($\bar{C}_D$), maximum lift coefficient (C_L^{max}), and Strouhal number (St) which is defined as,

$$St = \frac{f\,D}{U_\infty}$$

Table I.1: Validation results for solid cylinder with $Re = 80$

Case	NR	NTH	C_D	C_L^{max}	St
1	64	64	1.18	0.22	0.17
2	128	128	1.2	0.23	0.16
Reference (11)	-	-	1.05	0.2	0.16

where f is the vortex shedding frequency. Also the computational domain size for the two cases are given as NR and NTH in the radial and azimuthal directions.

Figure I.5 shows instantaneous isocontours of axial vorticity (ω_z) for 2D flow past a solid circular cylinder at $Re = 80$ for Case 2 ($NR = 128, NTH = 128$) at $tU_\infty/D = 107$. Both positive and negative vorticity values are plotted for the range $\omega_z \in \pm[0.5, 8]$. The center of the circular cylinder is located at (0,0) with radius, $R_I = 0.5$.

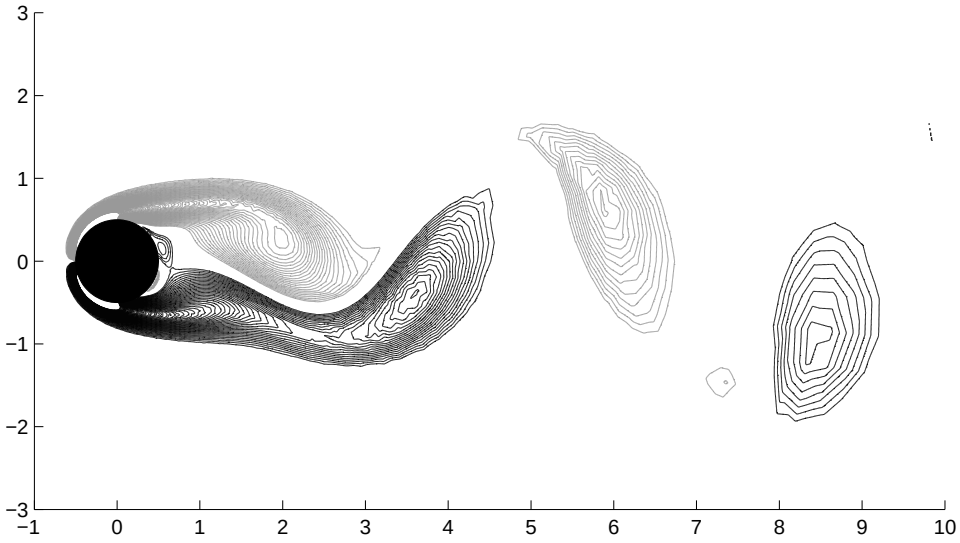

Figure I.5: Isocontous of z-vorticity in Case 2 for solid surface (no compliance) and $Re = 80$. The plot is at $tU_\infty/D = 107$. The dark lines show positive vorticity values, and the light lines show negative vorticity values.

Figure I.6 shows instantaneous isocontours of axial vorticity (ω_z) for 2D

flow past a solid circular cylinder at $Re = 80$ for Case 2 ($NR = 128, NTH = 128$) at $tU_\infty/D = 110$. Both positive and negative vorticity values are plotted for the range $\omega_z \in \pm[0.5, 8]$. The center of the circular cylinder is located at (0,0) with radius, $R_I = 0.5$.

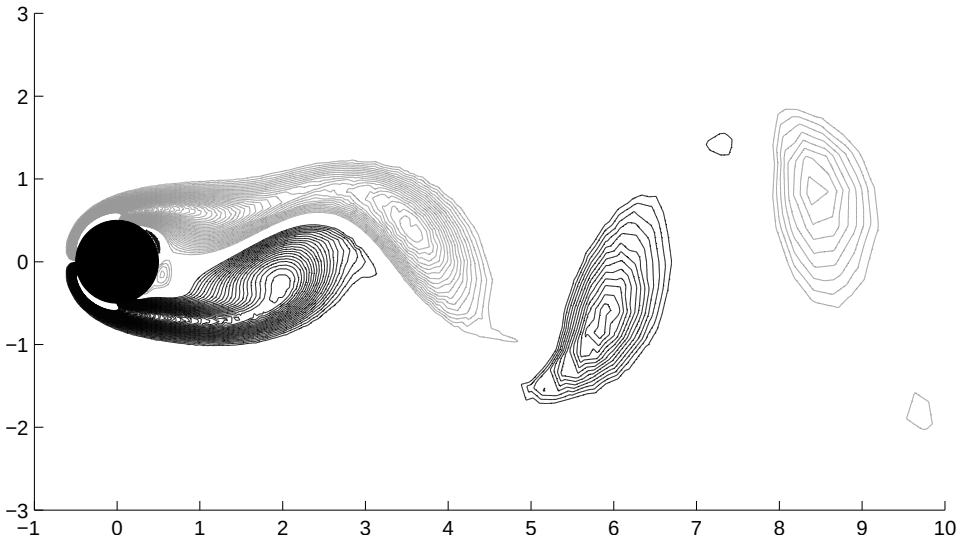

Figure I.6: Isocontous of z-vorticity in Case 2 for solid surface (no compliance) and $Re = 80$. The plot is at $tU_\infty/D = 110$. The dark lines show positive vorticity values, and the light lines show negative vorticity values.

I.F Results and discussion

We ran the simulation for two cases with the surface compliance turned on. These will be referred to as Case 3 and Case 4. For both cases, the grid size employed was 64×64 grid points. Case 1 is as defined in Section I.E for a solid cylinder. Case 3 has $(k/m) = (c/m) = 10$ for all three compliance zones. Case 4 has $(k/m) = (c/m) = 1$ for all three compliance zones.

Figure I.7 shows a comparison of the lift coefficients for cases 1, 3 and 4. Case 1 is denoted by solid line, Case 3 by dashed line and Case 4 by dash-dot line. The X-axis denotes non dimensional time (tU_∞/D). We can see that the peak lift

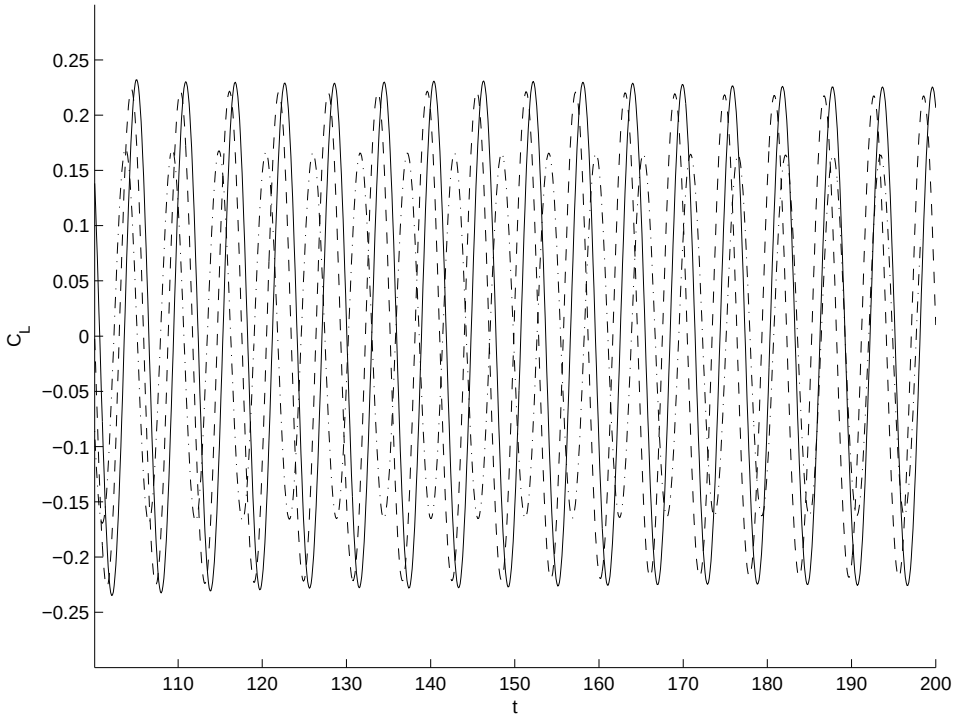

Figure I.7: Comparison of lift coefficient for Cases 1, 3 and 4. The X axis is nondimensional time (tU_∞/D). Case 1 is denoted by solid line, Case 3 by dashed line and Case 4 by dash-dot line.

coefficient goes down as the values of (k/m) and (c/m) are decreased, i.e. softness of the structure is increased. This is in agreement with Endo et al (11).

Figure I.8 shows a comparison of the drag coefficients for cases 1, 3 and 4. Again, Case 1 is denoted by solid line, Case 3 by dashed line and Case 4 by dash-dot line. The X-axis denotes non dimensional time (tU_∞/D). We can see that after statistical steady state has been reached, the time average drag *increases*, as the values of (k/m) and (c/m) are decreased. In other words, as the surface becomes more compliant, the time average drag increases.

Figure I.9 shows the shape of the compliance structure at $tU_\infty/D = 100$ for Case 3. The dashed line shows the nominal position of the cylinder.

Figure I.10 shows instantaneous streamline plots together with the compliant structure deformation (denoted by the circles) for one shedding cycle for

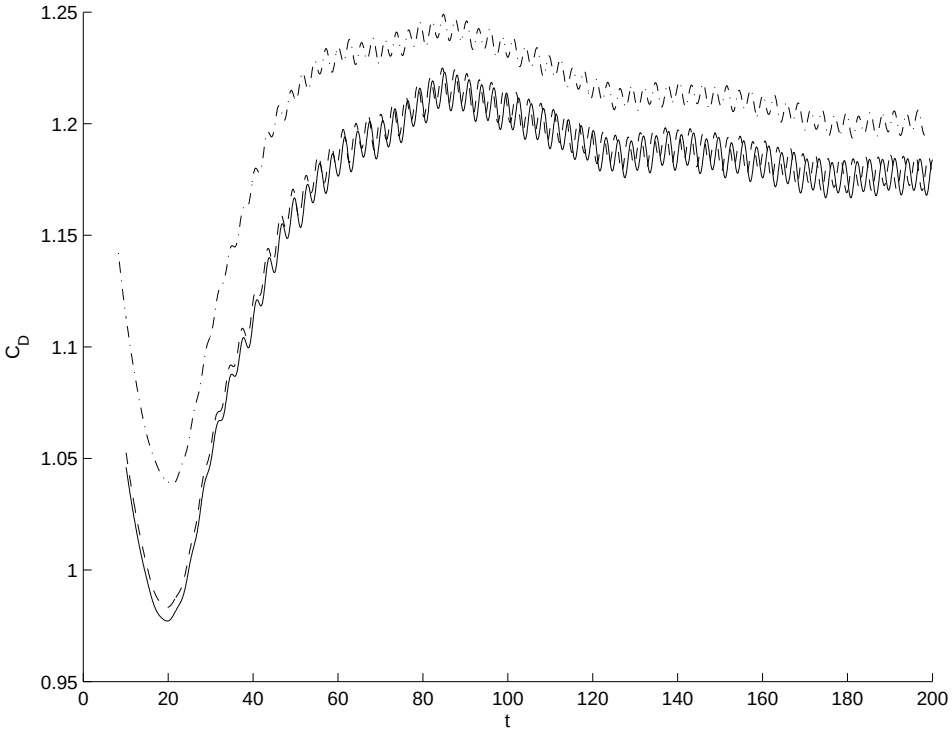

Figure I.8: Comparison of drag coefficient for Cases 1, 3 and 4. The X axis is nondimensional time (tU_∞/D). Case 1 is denoted by solid line, Case 3 by dashed line and Case 4 by dash-dot line.

Case 4. Figure I.11 shows the displacement, velocity and normalized pressure force at each grid point on the compliant cylinder surface for Case 4 for one complete shedding cycle. Note that the snapshots in Figures I.10 and I.11 correspond to the same six time instants and represent one complete vortex shedding cycle.

I.G Transformed momentum and continuity equations

In the momentum equations, the terms treated explicitly, using RKW3 scheme are denoted by A_r and A_θ, while the terms treated implicitly with the CN scheme are denoted by B_r and B_θ. The pressure terms, handled by the frac-

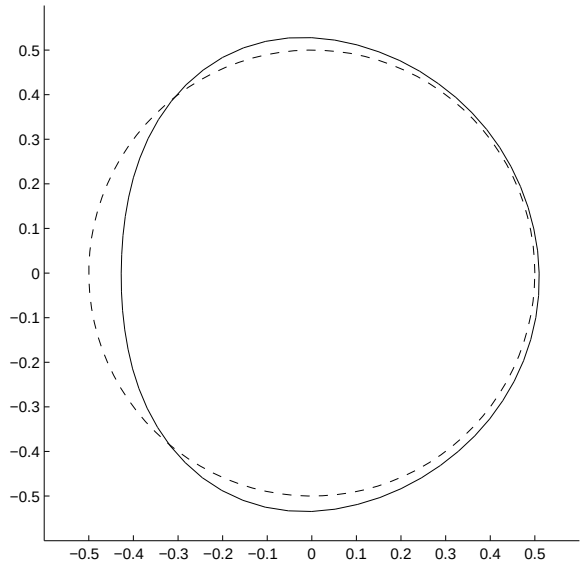

Figure I.9: The shape of the compliance structure (solid) together with the nominal position of the cylinder (dashed) for Case 3 at $tU_\infty/D = 100$.

tional step update, are denoted by P_r and P_θ. Then, the complete transformed momentum and continuity equations are written out as,

$$\begin{aligned}
\frac{\partial u_r}{\partial \tau} &= A_r + B_r + P_r \\
\frac{\partial u_\theta}{\partial \tau} &= A_\theta + B_\theta + P_\theta \\
0 &= \frac{\phi_r}{r}\frac{\partial (r u_r)}{\partial \xi_r} + \frac{1}{r}\frac{\partial u_\theta}{\partial \xi_\theta} + \frac{\phi_\theta}{r}\frac{\partial u_\theta}{\partial \xi_r}
\end{aligned}$$

where,

$$A_r = -\phi_t \frac{\partial u_r}{\partial \xi_r} - \frac{1}{r}\frac{\partial (u_r u_\theta)}{\partial \xi_\theta} - \frac{\phi_\theta}{r}\frac{\partial (u_r u_\theta)}{\partial \xi_r} + \frac{u_\theta^2}{r} - \frac{2\nu}{r^2}\frac{\partial u_\theta}{\partial \xi_\theta} - \frac{2\nu\phi_\theta}{r^2}\frac{\partial u_\theta}{\partial \xi_r} - \frac{\phi_r}{r}\frac{\partial (r u_r u_r)}{\xi_r} + S_r$$

$$B_r = \nu\left[\frac{\phi_r^2}{r}\frac{\partial}{\partial \xi_r}\left(r\frac{\partial u_r}{\partial r}\right) - \frac{u_r}{r^2}\right]$$

$$P_r = -\frac{\phi_r}{\rho}\frac{\partial p}{\partial \xi_r}$$

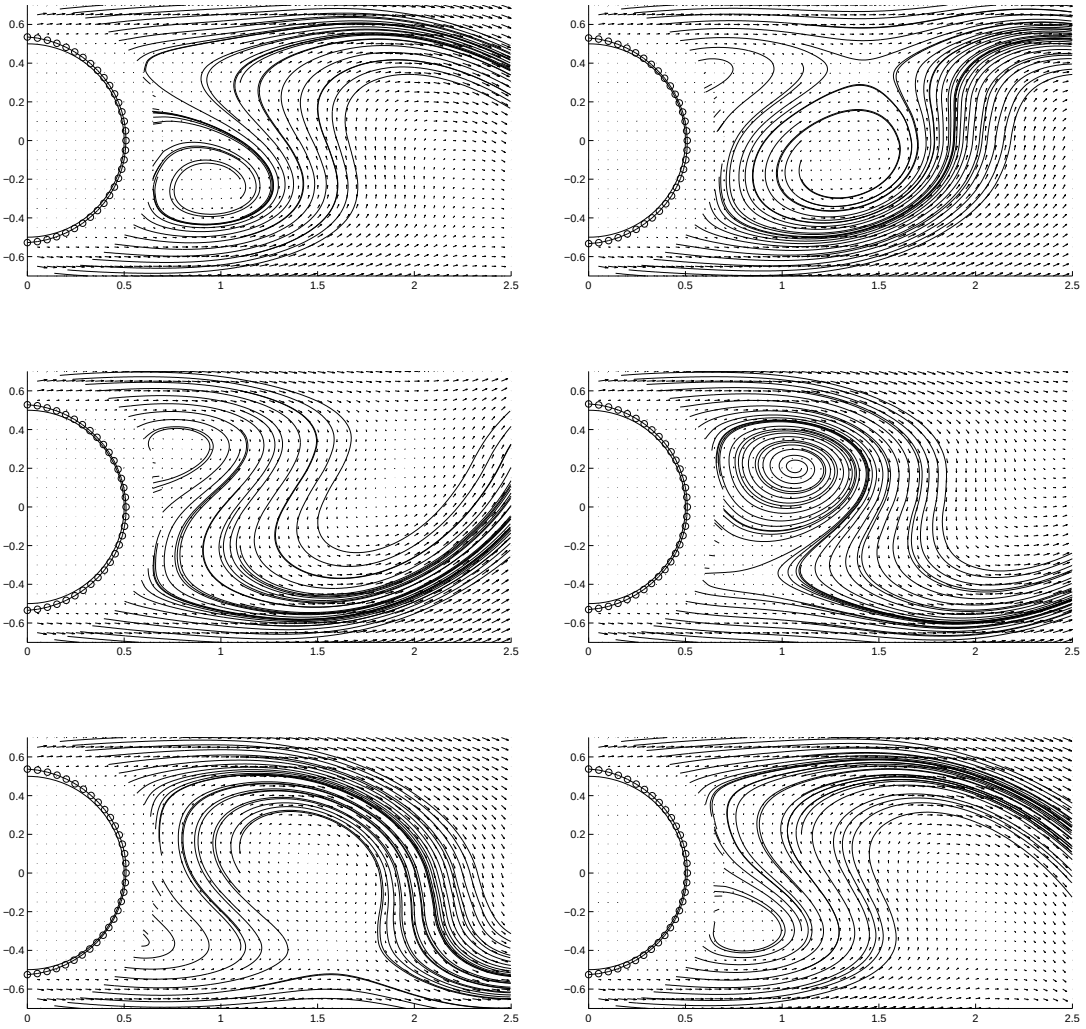

Figure I.10: Instantaneous streamline plots for one shedding cycle for Case 4

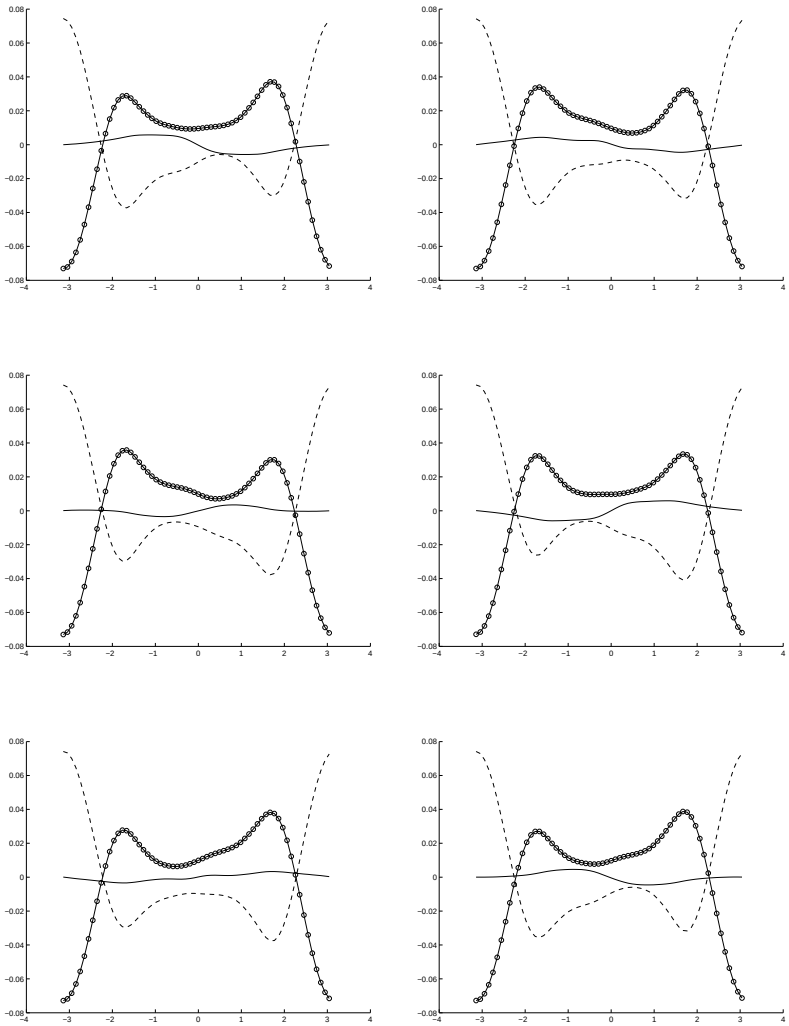

Figure I.11: Instantaneous plots of cylinder surface displacement (circle), surface velocity (solid) and normalized pressure (dashed) as a function of position on cylinder (θ) for Case 4

where the last term of A_r is computed as,

$$S_r = \frac{\nu}{r^2}\frac{\partial^2 u_r}{\partial \theta^2} = \frac{\nu}{r^2}\frac{\partial h}{\partial \xi_\theta} + \frac{\nu\phi_\theta}{r^2}\frac{\partial h}{\partial \xi_r}, \quad h = \frac{\partial u_r}{\partial \theta} = \frac{\partial u_r}{\partial \xi_\theta} + \phi_\theta \frac{\partial u_r}{\partial \xi_r}$$

and,

$$A_\theta = -\phi_t \frac{\partial u_\theta}{\partial \xi_r} - \frac{\phi_r}{r}\frac{\partial (r u_r u_\theta)}{\partial \xi_r} - \frac{u_r u_\theta}{r} - \frac{1}{r}\frac{\partial (u_\theta u_\theta)}{\partial \xi_\theta} - \frac{\phi_\theta}{r}\frac{\partial (u_\theta u_\theta)}{\partial \xi_r} + \frac{2\nu}{r^2}\frac{\partial u_r}{\partial \xi_\theta} + \frac{2\nu\phi_\theta}{r^2}\frac{u_r}{\xi_r} + S_\theta$$

$$B_\theta = \nu\left[\frac{\phi^2}{r}\frac{\partial}{\partial \xi_r}\left(r\frac{\partial u_\theta}{\partial \xi_r}\right) - \frac{u_\theta}{r^2}\right]$$

$$P_\theta = -\frac{1}{\rho}\frac{1}{r}\frac{\partial p}{\partial \xi_\theta} - \frac{1}{\rho}\frac{\phi_\theta}{r}\frac{\partial p}{\partial \xi_r}$$

where the last term of A_θ is computed as,

$$S_\theta = \frac{\nu}{r^2}\frac{\partial^2 u_\theta}{\partial \theta^2} = \frac{\nu}{r^2}\frac{\partial k}{\partial \xi_\theta} + \frac{\nu\phi_\theta}{r^2}\frac{\partial k}{\partial \xi_r}, \quad k = \frac{\partial u_\theta}{\partial \theta} = \frac{\partial u_\theta}{\partial \xi_\theta} + \phi_\theta \frac{\partial u_\theta}{\partial \xi_r}$$

The transformed Laplacian operator is defined as,

$$\begin{aligned} L(\phi) &= \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial \phi}{\partial r}\right) + \frac{1}{r^2}\frac{\partial^2 \phi}{\partial \theta^2} \\ &= \frac{\phi_r^2}{r}\frac{\partial}{\partial \xi_r}\left(r\frac{\partial \phi}{\partial \xi_r}\right) + \frac{1}{r^2}\frac{\partial L_\theta}{\partial \xi_\theta} + \frac{\phi_\theta}{r^2}\frac{\partial L_\theta}{\partial \xi_r} \end{aligned}$$

where,

$$L_\theta = \frac{\partial \phi}{\partial \xi_\theta} = \frac{\partial \phi}{\partial \xi_\theta} + \phi_\theta \frac{\partial \phi}{\partial \xi_r}$$

The transformed implicit part of Laplacian is defined as,

$$L^{imp}(\phi) = \frac{1}{\xi_r}\frac{\partial}{\partial \xi_r}\left(\xi_r \frac{\partial \phi}{\partial \xi_r}\right) + \frac{1}{\xi_r^2}\frac{\partial^2 \phi}{\partial \xi_\theta^2}$$

It follows that the explicit part of the Laplacian is given by,

$$L^{exp}(\phi) = L(\phi) - L^{imp}(\phi)$$

The transformed divergence operator is given by,

$$\begin{aligned}\nabla \cdot \mathbf{u} &= \frac{1}{r}\frac{\partial}{\partial r}(r\, u_r) + \frac{1}{r}\frac{\partial u_\theta}{\partial \theta} \\ &= \frac{\phi_r}{r}\frac{\partial}{\partial \xi_r}(r\, u_r) + \frac{1}{r}\frac{\partial u_\theta}{\partial \xi_\theta} + \frac{\phi_\theta}{r}\frac{\partial u_\theta}{\partial \xi_r}\end{aligned}$$

The transformed gradient operator is given by,

$$\begin{aligned}\nabla\phi &= \left(\frac{\partial \phi}{\partial r}, \frac{1}{r}\frac{\partial \phi}{\partial \theta}\right) \\ &= \left(\phi_r\frac{\partial \phi}{\partial \xi_r}, \frac{1}{r}\frac{\partial \phi}{\partial \xi_\theta} + \frac{\phi_r}{r}\frac{\partial \phi}{\partial \xi_r}\right)\end{aligned}$$

Thus, it can be verified that $\nabla\cdot(\nabla\phi) = \nabla^2\phi$ in the transformed coordinate system, which is necessary for the fractional step method to correctly remove divergence form the intermediate flow field.

I.H Conclusion

We have developed a new method for accurate direct numerical simulation of incompressible flow past a circular cylinder with a deformable surface based on (1) and (2). We have validated the code for the 2D unsteady vortex shedding case at $Re = 80$. We have also presented results for flow past a compliant cylinder for different values of the surface compliance parameters. Our results show that the Strouhal number increases and the peak lift coefficient decreases as compliance (or softness) of the cylinder surface is increased. This result is in agreement with Endo et al (11). Our results show that the time average drag *increases* as the surface is made more compliant.

In future work, this method will be used for optimization of 2D unsteady flow past compliant cylinders to minimize the aeracoustic noise generated by the flow-structure interaction. In a follow up paper, we describe the validation and results for incompressible turbulent 3D flow past a compliant cylinder for $Re = 300$.

This chapter, in full, has been submitted for publication to Journal of

Computational Physics as, Anish Karandikar and T. R. Bewley, *Accurate simulation of unsteady 2D flow past a compliant cylinder.*

Bibliography

[1] H. Luo, T. R. Bewley, Accurate simulation of near-wall turbulence over a compliant tensegrity fabric, SPIE Paper 5757-17.

[2] H. Luo, T. R. Bewley, On the contravariant form of the Navier-Stokes equation in time-dependent curvilinear coordinate systems, J. Comp. Phys. 199, p. 355.

[3] S. Xu, D. Rempfer, J. Lumley, Turbulence over a compliant surface: numerical simulation and analysis, J. Fluid Mech. 478, p. 11.

[4] H. Reese, T. Carolus, C Kato, Numerical prediction of the aeroacoustic sound sources in a low pressure axial fan with inflow distortion, Fan Noise 2007.

[5] A. L. Marsden, M. Wang, J. E. Dennis, P. Moin, Optimal aeroacoustic shape design using the surrogate management framework, Optimization and Engineering, 5(2) 2004, p. 235.

[14] A. L. Marsden, M. Wang, J. E. Dennis, P. Moin, Suppression of vortex-shedding noise via derivative-free shape optimization, Physics of Fluids, 16(10) 2004, p. L83.

[7] N. Hasan, S. F. Anwer, S. Sanghi, On the outflow boundary condition for external incompressible flows: A new approach, J. Comp. Phys. 206, 2005, p. 661.

[8] T. R. Bewley, P. Moin, R. Temam, DNS-based predictive control of turbulence: an optimal benchmark for feedback algorithms, J. Fluid. Mech. 447, 2001, p. 179.

[9] M. Braza, P. Chassing, H.H. Minh, Numerical study and physical analysis of the pressure and velocity fields in the wake of a circular cylinder, J. Fluid Mech. 165, 1986, p. 79.

[10] H.M. Blackburn, R.D. Henderson, A study of two dimensional flow past an oscillating cylinder, J. Fluid Mech. 385, 1999, p. 255.

[11] T. Endo, R. Himeno, Direct numerical simulation of flow across a compliant cylinder, RIKEN Review 48, October 2002.

[12] T. Endo, R. Himeno, Direct numerical simulation of turbulent flow over a compliant surface, J. Turbulence 3(007), p 1.

[13] N. Curle, The influence of solid boundary upon aerodynamic sound, Proc. Roy. Soc. Lond. A 231, 1955, p. 505.

[14] J. E. Ffowcs Williams, L. H. Hall, Aerodynamic sound generation by turbulent flow in the vicinity of a scattering half plane. J. Fluid Mech. 40, 1970, p. 657.

[15] M. Rosenfeld, D. Kwak, Time-dependent solutions of viscous incompressible flows in moving co-ordinates. Int. J. Numerical Methods in Fluids 13, 1991.

[15] H. Carlson, G. Berkooz, J. L. Lumley, Direct numerical simulation of flow in a channel with complex, time-dependent wall geometries: A pseudospectral method, J. Comp. Phys. 121, 1995, p. 155.

[16] Y. Mito, N. Kasagi, DNS study of turbulence modification with streamwise-uniform sinusoidal wall-oscillation, Int. J. Heat & Fluid Flow 19, 1998, p. 470.

[18] A. Booker, J. Dennis Jr, P. Frank, D. Serafini, V. Torczon, M. Trosset, A rigorous framework for optimization of expensive functions by surrogates, Structural and Multidisciplinary Optimization, 17, 1999, p. 113.

[19] P. Carpenter, A. Garrad, The hydrodynamic stability of flow over Kramer-type compliant surfaces, J. Fluid Mech. 155, p. 465.

[20] P. Carpenter, P. Morris, The effect of anisotropic wall compliance on boundary-layer stability and transition, J. Fluid Mech. 218, p. 171.

[21] C. Davies, P. Carpenter, Numerical simulations of the evolution of Tollmien-Schlichting waves over nite compliant panels, J. Fluid Mech. 335, p. 361.

II

Direct numerical simulation of turbulent 3D flow past a circular cylinder with a compliant surface

II.A Introduction

This work performs direct numerical simulation of incompressible turbulent flow past a circular cylinder with a deformable surface. A time-dependent coordinate transformation is applied to map the deformed cylindrical polar domain to a regular computational domain. We employ a pseudospectral method to compute spatial derivatives in the periodic azimuthal and axial directions. Derivatives in the radial (wall-normal) direction are computed using a second order finite difference scheme.

In Section II.A.1 we present a brief survey of previous work done in the DNS of turbulent flow in a channel with compliant walls. This work extends some of the methods used in compliant channel flow to simulate flow past a compliant cylinder. In Section II.A.2, we describe the model used for the compliant surface, and an introduction to the domain transformation method used is given in Section II.A.3. The governing equations for the turbulent flow are the full three dimensional incompressible Navier Stokes equations which are described in detail in Section II.B. In this section, we also describe the coordinate transformation as applied to the Navier Stokes equations to account for the moving cylinder surface. In Section II.C, we describe the numerical method used to perform DNS of turbulent flow past a compliant cylinder. This section describes in detail the spatial and temporal discretization methods employed in the present code, and also details of the fractional step method used to enforce continuity at the end of each Runge-Kutta substep.

We present validation results for the code at Reynolds number 300 for a solid cylinder, with no surface compliance in Section II.D.1. At this Reynolds number, flow past a circular cylinder is three dimensional and turbulent with alternate vortex structures in the wake region. The statistics produced by our code match well with results presented in (5) for turbulent flow past a solid circular cylinder at Reynolds number 300. Finally, in Section II.D.2, we present results

for turbulent flow past a cylinder with a compliant surface. We found that as the compliance (or softness) of the cylinder surface is increased, the peak lift coefficient goes down, while the time average drag *increases*.

The 3D Navier Stokes equations and the continuity equations in transformed coordinates are explicitly written out in the appendix Section II.E.

II.A.1 Turbulent flow in a compliant channel

In this work, we perform DNS of turbulent flow past a cylinder with a compliant surface. This work is a conceptual extension of previous studies involving turbulent flow in a channel with compliant walls. Many researchers have studied this problem before, see for example Carpenter et al (8), (9) & (10), which deal with, among other things, the delay of laminar to turbulent transition in compliant channels. Endo et al (11) performed direct numerical simulation of turbulent flow in a compliant channel and reported 2 to 3% drag reduction in the presence of deformable walls. Rempfer et al (4) challenged this result for the reason of insufficient averaging time. Rempfer et al found no discernible change in long term turbulence statistics due to the presence of compliant walls in a channel flow.

In (1) and (2), Luo & Bewley developed an alternative numerical scheme for direct numerical simulation of turbulent flow in a channel with deformable walls. This scheme is stable and accurate for large deformations of the channel walls. The results obtained in these studies were consistent with the results of Rempfer et al for small wall deformation. Also, a new structural paradigm called a tensegrity fabric was introduced in (1) and (2) to model the compliant channel walls. The present work extends the method developed by Luo & Bewley to cylindrical polar coordinates. We handle the effects of the arbitrarily moving cylinder surface on the flow with care and accuracy to simulate turbulent flow past a compliant cylinder.

II.A.2 Compliant surface model

The compliant surface is modeled by a simple spring-mass-damper system. Each point on the compliant surface is assumed to be attached to a spring-mass-damper system which is only allowed to deform in the radial direction (See Figure II.1). The dashed line shows the nominal (rest) position of the compliant surface with no forces acting on it. This system has an associated stiffness coefficient (k), damping coeffiecient (c) and mass (m). The governing equation is given by a second order decoupled ODE for each grid point on the surface of the cylinder, as,

$$m\frac{\partial^2\eta}{\partial t^2} + c\frac{\partial\eta}{\partial t} + k\eta = f$$

where f is the external force applied to the compliant surface, and η is the displacement of the surface from its rest position. In this flow structure interaction, the external force acting on the compliant surface appears as the normalized pressure force acting on the surface due to the flow field surrounding it (p'). Then, the equation is divided by m to get,

$$\frac{\partial^2\eta}{\partial t^2} + \left(\frac{c}{m}\right)\frac{\partial\eta}{\partial t} + \left(\frac{k}{m}\right)\eta = \frac{p'}{m}$$

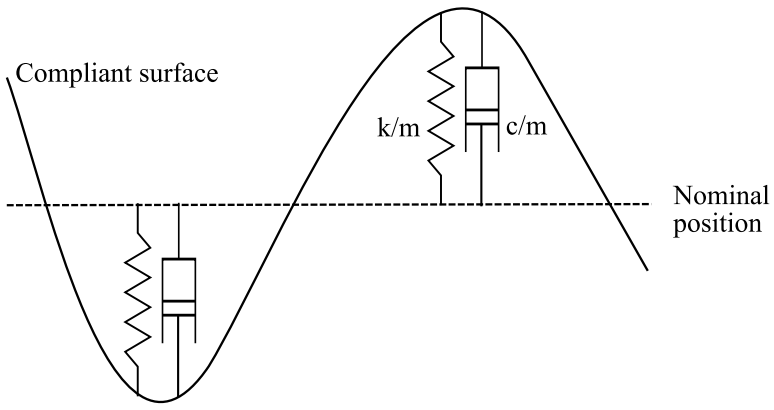

Figure II.1: Compliant surface model

In the numerical code, the dynamics of flow are simulated for one full timestep, and the pressure at the cylinder surface is used as input to compute

dynamics of the compliant surface for one timestep. Then, the position and velocity of the surface are used as boundary conditions for the flow solver, and so on, back and forth.

The cylinder surface is divided into three zones, defined as,

Zone I: $2\pi/3 \leq |\theta| \leq \pi$
Zone II: $\pi/3 \leq |\theta| \leq 2\pi/3$
Zone III: $0 \leq |\theta| \leq \pi/3$

and the physical parameters defining the compliant surface are assumed constant over each of these three zones. Thus, the stiffness coefficient (k/m) and damping coefficient (k/m) for each zone defined above constitute the problem parameters. Note that $\theta = 0$ corresponds to the downstream stagnation point with the flow over the cylinder going from left to right.

II.A.3 Domain transformation

An important part of this code is to accurately capture the effect of the arbitrarily moving cylinder surface on the flow. The physical domain is irregular in this problem. The inner edge of this warped domain is illustrated in Figure II.2. Note that the grid is stretched only in the radial direction, and there is no stretching in the azimuthal or axial directions. The nonorthogonal coordinate transformation makes the contravriant formulation of the Navier Stokes equations is very involved and expensive to compute. Thus, in the present work, we use a simple cylindrical polar coordinate system as detailed in Section II.B and a time-dependent coordinate transformation similar to (1) to map the irregular physical domain to a regular computational domain (Figure II.3). The wall parallel derivatives in the azimuthal and axial directions are computed spectrally using the Discrete Fourier Transform. In the wall normal (radial) direction, the derivatives are computed using a second-order central finite difference scheme. This allows us to compute most of the derivatives in the radial direction implicitly in the time integration scheme, thus increasing the stability of the code in presence of large wall deformations.

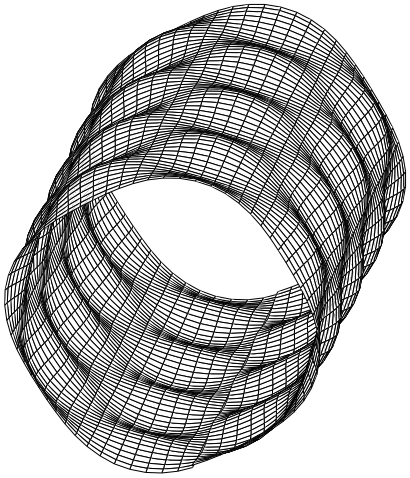

Figure II.2: Inner edge of the irregular physical domain (compliant surface)

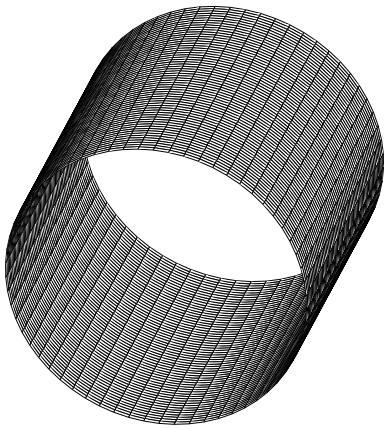

Figure II.3: Inner edge of the regular (transformed) computational domain

II.B Governing Equations

The governing equations are the incompressible Navier Stokes equations together with the continuity equation.

$$\begin{aligned}\frac{\partial u_r}{\partial t} &= -\frac{1}{r}\frac{\partial\left(ru_ru_r\right)}{\partial r}+\frac{u_\theta^2}{r}-\frac{1}{r}\frac{\partial(u_ru_\theta)}{\partial\theta}-\frac{2\nu}{r^2}\frac{\partial u_\theta}{\partial\theta}-\frac{\partial(u_ru_z)}{\partial z} \\ &+ \nu\left[\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial u_r}{\partial r}\right)-\frac{u_r}{r^2}+\frac{1}{r^2}\frac{\partial^2u_r}{\partial\theta^2}+\frac{\partial^2u_r}{\partial z^2}\right]-\frac{1}{\rho}\frac{\partial p}{\partial r}\end{aligned} \tag{II.1}$$

$$\begin{aligned}\frac{\partial u_\theta}{\partial t} &= -\frac{1}{r}\frac{\partial\left(ru_ru_\theta\right)}{\partial r}-\frac{u_ru_\theta}{r}-\frac{1}{r}\frac{\partial(u_\theta u_\theta)}{\partial\theta}+\frac{2\nu}{r^2}\frac{\partial u_r}{\partial\theta}-\frac{\partial(u_\theta u_z)}{\partial z} \\ &+ \nu\left[\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial u_\theta}{\partial r}\right)-\frac{u_\theta}{r^2}+\frac{1}{r^2}\frac{\partial^2u_\theta}{\partial\theta^2}+\frac{\partial^2u_\theta}{\partial z^2}\right]-\frac{1}{\rho}\frac{1}{r}\frac{\partial p}{\partial\theta}\end{aligned} \tag{II.2}$$

$$\begin{aligned}\frac{\partial u_z}{\partial t} &= -\frac{1}{r}\frac{\partial(ru_ru_z)}{\partial r}-\frac{1}{r}\frac{\partial(u_\theta u_z)}{\partial\theta}-\frac{\partial(u_zu_z)}{\partial z} \\ &+ \nu\left[\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial u_z}{\partial r}\right)+\frac{1}{r^2}\frac{\partial^2u_z}{\partial\theta^2}+\frac{\partial^2u_z}{\partial z^2}\right]-\frac{1}{\rho}\frac{\partial p}{\partial z}\end{aligned} \tag{II.3}$$

$$\nabla\cdot\mathbf{u}=\frac{1}{r}\frac{\partial(ru_r)}{\partial r}+\frac{1}{r}\frac{\partial u_\theta}{\partial\theta}+\frac{\partial u_z}{\partial z} \tag{II.4}$$

The coordinate transformation is nonorthogonal, as the grid is deformed only in the radial direction. As defined in II.A.2, $\eta(\theta,z,t)$ is the displacement of the compliant cylinder surface from its nominal position. Then, following time-dependent coordinate transformation may be used to map the irregular physical onto a regular cylindrical domain

$$\begin{aligned}r &= (\xi_r-R_I)\cdot\left[\frac{R_O-(R_I+\eta)}{R_O-R_I}\right]+(R_I+\eta) \\ \theta &= \xi_\theta \\ z &= \xi_z \\ t &= \tau\end{aligned}$$

where (r,θ,z) denote cylindrical polar coordinates, and (ξ_r,ξ_θ,ξ_z) denote transformed coordinates in the computational domain. R_I is the nominal position of

the cylinder surface at rest, and R_O is the outer radius of the computational domain. The Jacobian of the spatial transformation, and its determinant are,

$$C = \frac{\partial \mathbf{x}}{\partial \xi} = \begin{bmatrix} \frac{\partial r}{\partial \xi_r} & \frac{\partial r}{\partial \xi_\theta} & \frac{\partial r}{\partial \xi_z} \\ \frac{\partial \theta}{\partial \xi_r} & \frac{\partial \theta}{\partial \xi_\theta} & \frac{\partial \theta}{\partial \xi_z} \\ \frac{\partial z}{\partial \xi_r} & \frac{\partial z}{\partial \xi_\theta} & \frac{\partial z}{\partial \xi_z} \end{bmatrix} = \begin{bmatrix} \frac{\partial r}{\partial \xi_r} & \frac{\partial r}{\partial \xi_\theta} & \frac{\partial r}{\partial \xi_z} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$J = |C| = \frac{\partial r}{\partial \xi_r} = \frac{R_O - (R_I + \eta)}{R_O - R_I}$$

The inverse spatial transformation is,

$$C^{-1} = \frac{\partial \xi}{\partial \mathbf{x}} = \begin{bmatrix} \frac{1}{J} & -\frac{1}{J}\frac{\partial r}{\partial \xi_\theta} & -\frac{1}{J}\frac{\partial r}{\partial \xi_z} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \frac{\partial \xi_r}{\partial r} & \frac{\partial \xi_r}{\partial \theta} & \frac{\partial \xi_r}{\partial z} \\ \frac{\partial \xi_\theta}{\partial r} & \frac{\partial \xi_\theta}{\partial \theta} & \frac{\partial \xi_\theta}{\partial z} \\ \frac{\partial \xi_z}{\partial r} & \frac{\partial \xi_z}{\partial \theta} & \frac{\partial \xi_z}{\partial z} \end{bmatrix}$$

Defining the nontrivial elements of C^{-1} as,

$$\begin{aligned}
\phi_r &= 1/J = 1 \Big/ \left[\frac{\partial r}{\partial \xi_r}\right] = \frac{R_O - R_I}{R_O - (R_I + \eta)} \\
\phi_\theta &= \frac{\partial \xi_r}{\partial \theta} = -\frac{1}{J}\frac{\partial r}{\partial \xi_\theta} = -\frac{R_O - R_I}{R_O - (R_I + \eta)} \cdot \left[\frac{R_O - \xi_r}{R_O - R_I}\right] \eta_\theta \\
&= -\frac{R_O - \xi_r}{R_O - (R_I + \eta)} \eta_\theta \\
\phi_z &= \frac{\partial \xi_r}{\partial \theta} = -\frac{1}{J}\frac{\partial r}{\partial \xi_z} = -\frac{R_O - R_I}{R_O - (R_I + \eta)} \cdot \left[\frac{R_O - \xi_r}{R_O - R_I}\right] \eta_z \\
&= -\frac{R_O - \xi_r}{R_O - (R_I + \eta)} \eta_z \\
\phi_t &= \frac{\partial \xi_r}{\partial t} = -\frac{1}{J}\frac{\partial r}{\partial t} = -\frac{R_O - R_I}{R_O - (R_I + \eta)} \cdot \left[\frac{R_O - \xi_r}{R_O - R_I}\right] \eta_t \\
&= -\frac{R_O - \xi_r}{R_O - (R_I + \eta)} \eta_t
\end{aligned}$$

and using the chain rule, we can apply the substitutions,

$$\frac{\partial}{\partial t} \rightarrow \frac{\partial}{\partial t} + \phi_t \frac{\partial}{\partial \xi_r}$$
$$\frac{\partial}{\partial \theta} \rightarrow \frac{\partial}{\partial \xi_\theta} + \phi_\theta \frac{\partial}{\partial \xi_r}$$
$$\frac{\partial}{\partial z} \rightarrow \frac{\partial}{\partial \xi_z} + \phi_z \frac{\partial}{\partial \xi_r}$$
$$\frac{\partial}{\partial r} \rightarrow \phi_r \frac{\partial}{\partial \xi_r}$$

to express the derivatives in (II.1), (II.2), (II.3) and (II.4) in terms of the new coordinate system. With these substitutions, the transformed r, θ, z momentum equations, and the conitnuity equation, are written out explicitly in Section II.E.

II.C Numerical algorithm

The numerical algorithm used for solving the unsteady incomspressible Navier Stokes equation is similar to the one used by Luo & Bewley in (1), in which DNS of turbulent flow in a channel with deformable walls is performed. In this code, we use a hybird pseudospectral/finite-difference method for spatial discretization, and a mixed low storage Runge-Kutta-Wray/Crank-Nicholson (RKW3/CN) method for temporal discretization. In addition to the flow variables, the geometry-related time-varying coefficients, ϕ_r, ϕ_θ, ϕ_z and ϕ_t need to be spatially discretized.

The azimuthal direction is naturally periodic for the O-type grid used. For simulating an infinitely long cylinder, we consider a cylinder with a finite length and apply periodic boundary conditions in the axial direction. No-slip and no-penetration boundary conditions are assumed at the cylinder surface. Uniform flow is imposed on the upstream side of the outer boundary with a velocity of U_∞, while physical outflow boundary conditions as in (7) are used in the downstream side of the boundary.

As the various Fourier modes can not be fully decoupled, the pressure

equation needs to be solved iteratively. Details of this algorithm are discussed in Section II.C.3.

II.C.1 Spatial discretization

The grid is chosen to be evenly spaced and non-staggered in the azimuthal (ξ_θ) and axial (ξ_z) directions so that spatial derivatives can be computed spectrally. In the wall-normal direction (ξ_r), the grid is staggered and stretched using a hyperbolic tangent function, to give sufficient grid resolution in the near-wall region. The variables u_θ, u_z and p are discretized on the family of grid points $j = 0, 1, ..., NR$, where $j = 0$, corresponds to the lower wall and $j = NR$ corresponds to the upper wall. The variables u_r, ϕ_r, ϕ_θ, ϕ_z and ϕ_t are discretized on the family of grid points $j = \frac{1}{2}, 1 + \frac{1}{2}, ..., NR - \frac{1}{2}$, where the $j = n + \frac{1}{2}$ gridpoint is midway between the gridpoints $j = n$, and $j = n + 1$.

With the arrangement of the discrete variables on the numerical grid as described above, the three components of the momentum equation are enforced at the grid points at which the corresponding velocity components are discretized, and the continuity equation is solved on the cells centered at the pressure points.

II.C.2 Temporal discretization

A low storage 3rd-order Runge-Kutta-Wray/Crank-Nicholson scheme similar to the one used by Bewley et al in (12) is used to advance the flow from time step m to the next time step $m+1$. At each RK substep k ($k = 1, 2, 3$), a fractional step method (Bewley et al, 2001) is used to march the flow and project the flow field onto a divergence-free subspace.

Let the operators A_r, A_θ and A_z represent the terms treated explicitly (RKW3) and B_r, B_θ and B_z represent the terms treated implicitly (CN) in the Navier Stokes equations.

$$\frac{u_r^{k+1} - u_r^k}{\Delta t} = \beta_k(B_r^{k+1} + B_r^k) + \gamma_k A_r^k + \zeta_k A_r^{k-1} - 2\beta_k P_r^k$$
$$\frac{u_\theta^{k+1} - u_\theta^k}{\Delta t} = \beta_k(B_\theta^{k+1} + B_\theta^k) + \gamma_k A_\theta^k + \zeta_k A_\theta^{k-1} - 2\beta_k P_\theta^k$$
$$\frac{u_z^{k+1} - u_z^k}{\Delta t} = \beta_k(B_z^{k+1} + B_z^k) + \gamma_k A_z^k + \zeta_k A_z^{k-1} - 2\beta_k P_z^k$$

Note that P_r, P_θ and P_z represent the pressure gradient terms in the three components of the Navier Stokes equations which are treated explicitly. All the operators in the above equations are explicitly written out for the transformed coordinate system in Section II.E. For the RKW3/CN constants, β_k, γ_k and ζ_k, the reader is referred to (12).

II.C.3 Fractional step method

During each RK substep, while marching the current velocity field ($\mathbf{u}^k$) to the velocity field at the next time step ($\mathbf{u}^{k+1}$), an intermediate flow field, $\mathbf{u}^*$, is obtained. This velocity field is not divergence free. After this step, a projection equation is solved to make the velocity field divergence free. The projection equation is a Poisson equation with its right hand side given by divergence of the intermediate velocity field scaled by a factor,

$$\nabla^2\phi = \frac{1}{2\beta_k\Delta t}(\nabla \cdot \mathbf{u}^*)$$

The solution to this equation (ϕ) is the pressure update to get the new pressure field,

$$p^{k+1} = p^k + \phi$$

Also, the intermediate velocity is projected onto the divergence free space as,

$$\mathbf{u}^{k+1} = \mathbf{u}^* - (2\beta_k \Delta t)\, \nabla\phi$$

The Laplacian ($\nabla^2 \cdot$), divergence ($\nabla \cdot$) and gradient (∇) operators in the transformed coordinate system are defined explicitly in Section II.E.

The Poisson equation has to be solved in Fourier space iteratively. This has to be done because the Laplacian, divergence and gradient operators cause coupling of wavenumbers in the azimuthal direction, due to the presence of non-constant coeffiecients ϕ_r, ϕ_θ and ϕ_z. We split the Laplacian operator into two parts. The terms with the various Fourier modes decoupled are treated implicitly ($L^{imp}(\cdot)$), while the rest of the terms are treated explicitly ($L^{exp}(\cdot)$), such that, $L = L^{imp} + L^{exp}$. The following algorithm describes the numerical method used to solve the Pressure update Poisson equation iteratively in Fourier space.

Algorithm for iterative pressure update Poisson equation solver

- Compute scaled divergence in d
- Set ϕ^0 to zero or initial guess
- For $i = 0 : itmax$
- Compute right hand side, $r = d - L^{exp}(\phi^i)$, in physical space
- Convert r to Fourier space, $\hat{r}$
- Solve $L^{imp}(\hat{\phi}^{i+1}) = \hat{r}$
- Convert $\hat{\phi}^{i+1}$ to physical space, ϕ^{i+1}
- Compute residual, $e = d - L(\phi^{i+1})$
- If norm of residual, $|e|$ is less than tolerance, Exit For Loop
- End For Loop
- Set, $\mathbf{u}^{k+1} \rightarrow \mathbf{u}^* - \nabla\phi$
- Set, $p^{k+1} \rightarrow p^k + \phi$

II.D Results and discussion

II.D.1 Code validation

We ran the flow code to simulate turbulent flow past a solid circular cylinder at Reynolds number of 300 with compliance turned off. At Reynolds numbers greater than $Re \approx 200$, the flow becomes three dimensional with large streamwise structures developing in it. Around reynolds number of 260, spanwise vortical structures are observed in the flow. The grid size used for this test case was $(NR, NTH, NZ) = (64 \times 64 \times 64)$ grid points. The nominal inner radius of the solid cylinder was taken as $R_I = 0.5$, and the radius of the outer edge of the computational domain was taken as $R_O = 20$. The flow was initialized with a uniform flow, and was run till $tU_\infty/D = 50$ by which time, all initial transients had washed out of the domain.

Figure II.4 shows isosurfaces of streamwise vorticity (ω_x) in a three-dimensional view at a time instant $tU_\infty/D = 45.1$. Figure II.5 shows isosurfaces of streamwise vorticity (ω_x) in a top view for the same time instant, while Figure II.6 shows isosurfaces of streamwise vorticity (ω_x) in a side view. Figures II.7 to II.10 show isocontours of streamwise vorticity for this time instant at 4 equispaced z-locations for values of $\omega_x = \pm 0.7$. The positove contours are plotted with dark lines, and the negative contours with light lines.

Figure II.11 shows isosurfaces of streamwise vorticity (ω_x) in a three-dimensional view at a time instant $tU_\infty/D = 47.6$. Figure II.12 shows isosurfaces of streamwise vorticity (ω_x) in a top view for the same time instant, while Figure II.13 shows isosurfaces of streamwise vorticity (ω_x) in a side view. Figures II.14 to II.17 show isocontours of streamwise vorticity for this time instant at 4 equispaced z-locations for values of $\omega_x = \pm 0.7$. The positove contours are plotted with dark lines, and the negative contours with light lines.

These two time instants ($tU_\infty/D = 45.1$ and $tU_\infty/D = 47.6$) for which the vorticity plots are shown, correspond to two extremes on the lift coefficient (C_L)

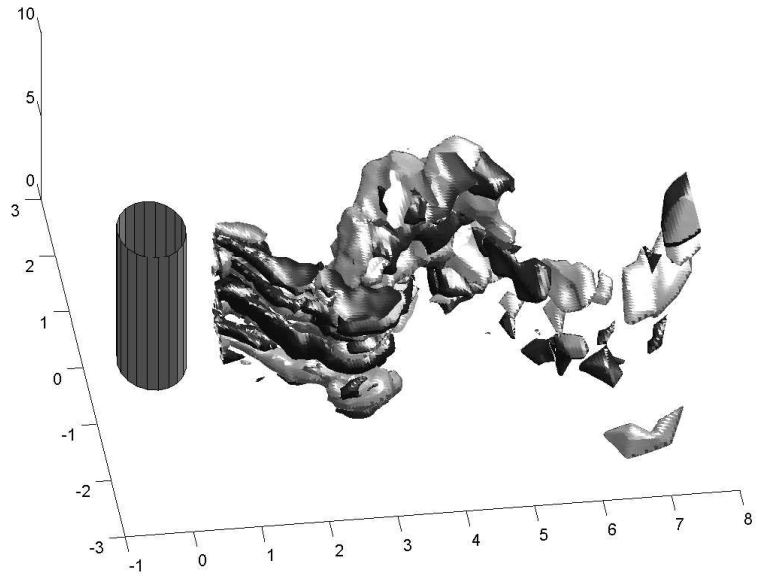

Figure II.4: 3D view of isosurfaces of streamwise vorticity for solid cylinder at $tU_\infty/D = 45.1$ for $\omega_x = +1$ (black) and $\omega_x = -1$ (grey)

evolution for the flow (See Figure II.20). For all plots, the solid circular cylinder is at $(0, 0)$ with a nominal radius of $R_I = 0.5$. The X-axis is the streamwise direction and the Z-axis is the axial direction.

Figure II.18 shows time-average and span-average streamwise velocity (u_x/U_∞) for different downstream locations, at $x/D = 1.2$, 1.5, 2, 2.5 and 3. Figure II.19 shows time-average and span-average crossflow velocity (u_y/U_∞) for different downstream locations, at $x/D = 1.2$, 1.5, 2, 2.5 and 3. These results match well with the results presented by Kravchenko et al (5) for turbulent flow past a solid circular cylinder at a Reynolds number of 300.

II.D.2 Results for compliant cylinder case

We ran the code with compliance turned on for $tU_\infty/D = 10$ time units with the flow field initialized by a fully developed turbulent flow past cylinder at $Re = 300$. The same grid size and cylinder and domain dimensions as in the

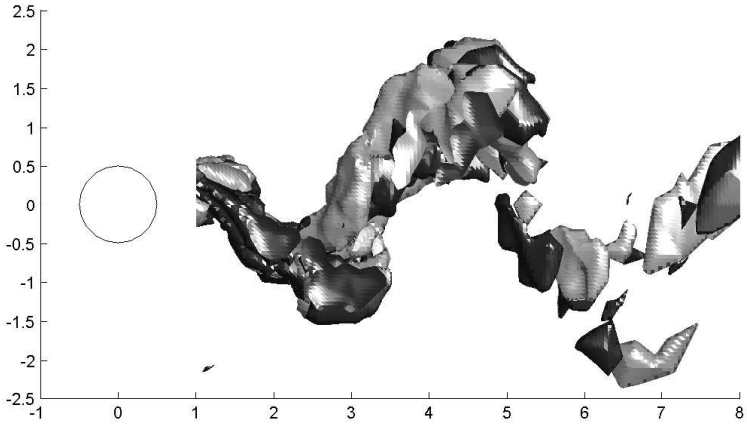

Figure II.5: Top view of isosurfaces of streamwise vorticity for solid cylinder at $tU_\infty/D = 45.1$ for $\omega_x = +1$ (black) and $\omega_x = -1$ (grey)

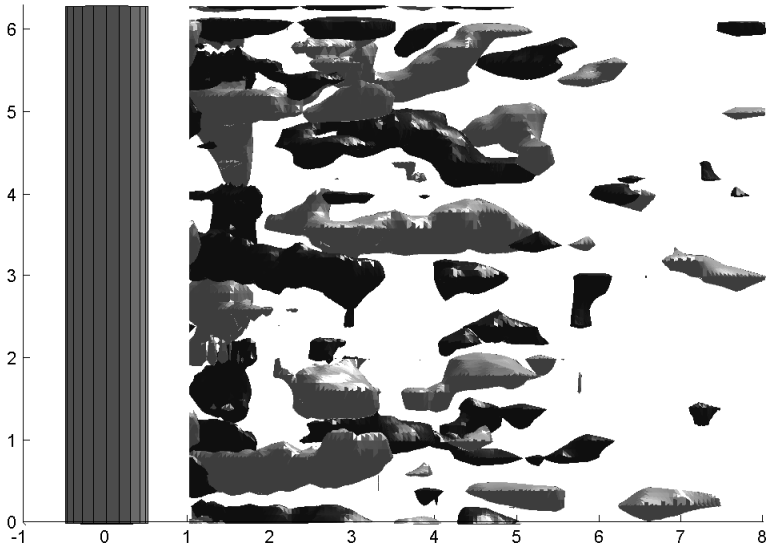

Figure II.6: Side view of isosurfaces of streamwise vorticity for solid cylinder at $tU_\infty/D = 45.1$ for $\omega_x = +1$ (black) and $\omega_x = -1$ (grey)

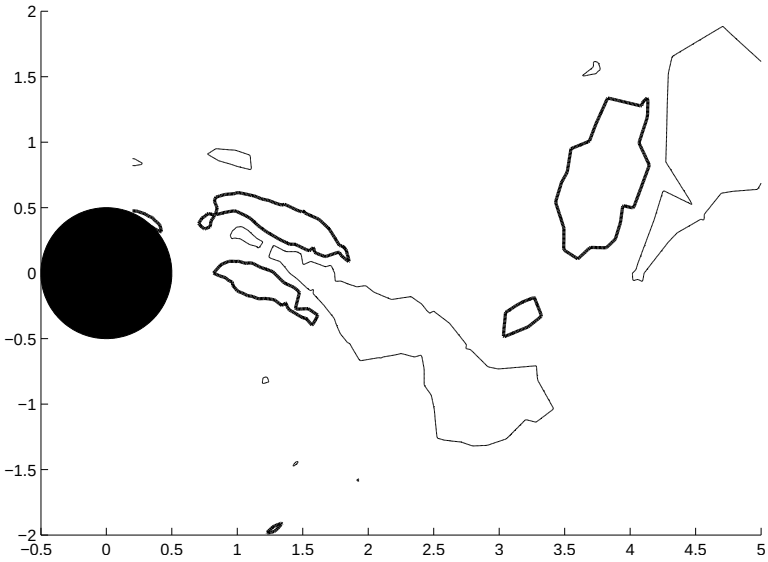

Figure II.7: Isocontours of streamwise vorticity for solid cylinder at $tU_\infty/D = 45.1$ at $z = 0$ for $\omega_x = +0.7$ (dark) and $\omega_x = -0.7$ (light)

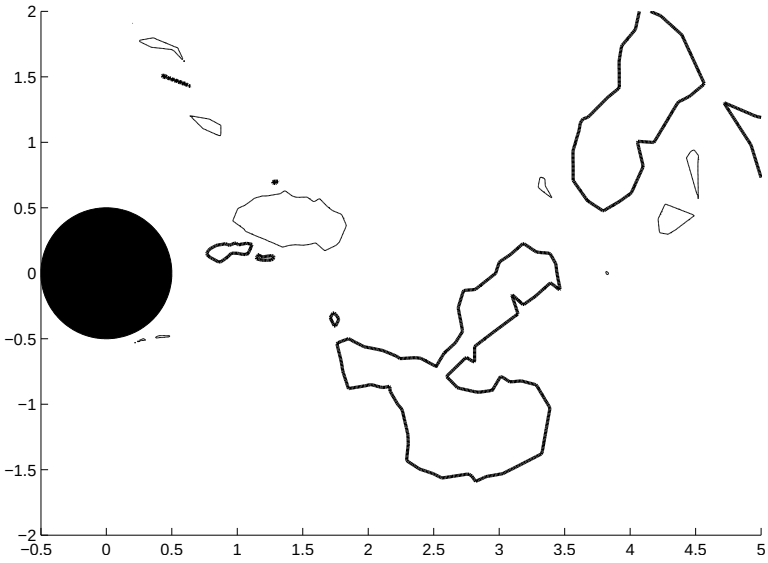

Figure II.8: Isocontours of streamwise vorticity for solid cylinder at $tU_\infty/D = 45.1$ at $z = \pi/2$ for $\omega_x = +0.7$ (dark) and $\omega_x = -0.7$ (light)

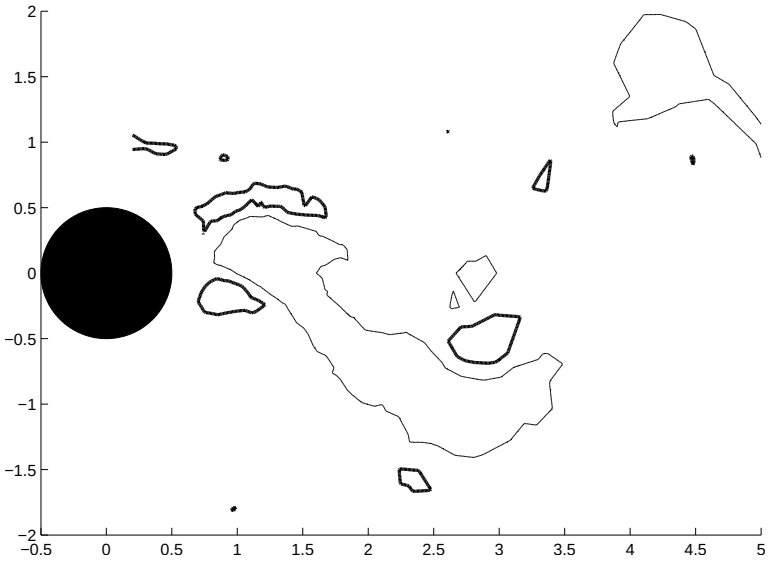

Figure II.9: Isocontours of streamwise vorticity for solid cylinder at $tU_\infty/D = 45.1$ at $z = \pi$ for $\omega_x = +0.7$ (dark) and $\omega_x = -0.7$ (light)

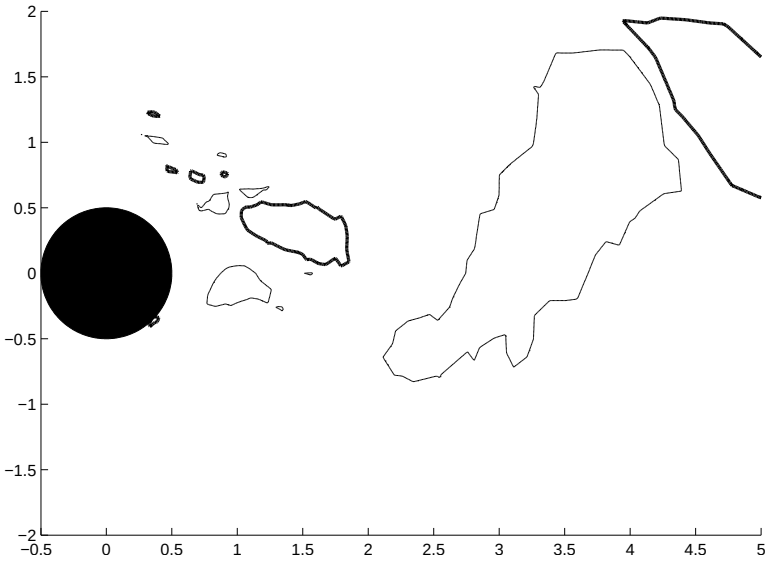

Figure II.10: Isocontours of streamwise vorticity for solid cylinder at $tU_\infty/D = 45.1$ at $z = 3\pi/2$ for $\omega_x = +0.7$ (dark) and $\omega_x = -0.7$ (light)

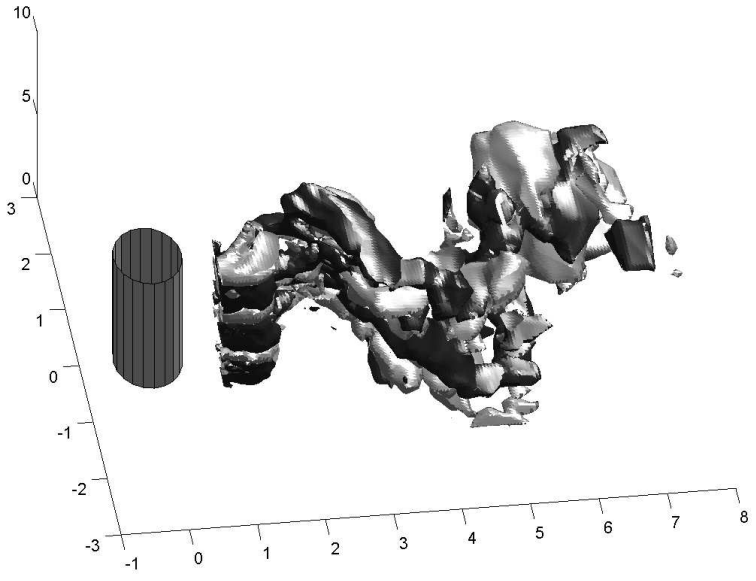

Figure II.11: 3D view of isosurfaces of streamwise vorticity for solid cylinder at $tU_\infty/D = 47.6$ for $\omega_x = +1$ (black) and $\omega_x = -1$ (grey)

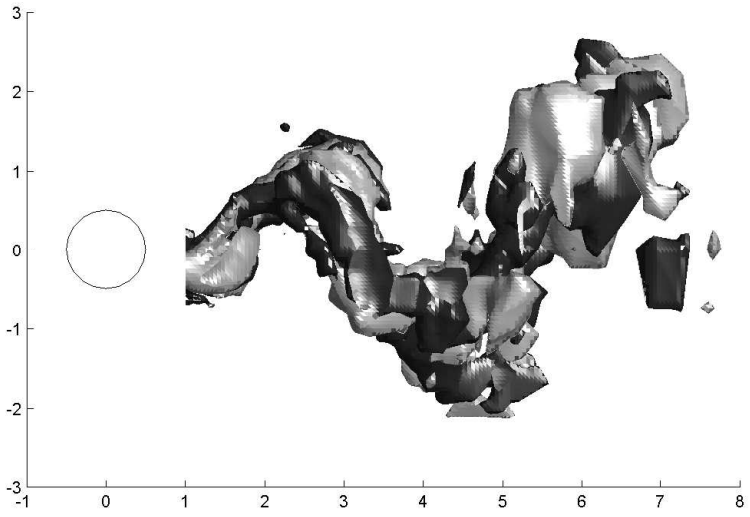

Figure II.12: Top view of isosurfaces of streamwise vorticity for solid cylinder at $tU_\infty/D = 47.6$ for $\omega_x = +1$ (black) and $\omega_x = -1$ (grey)

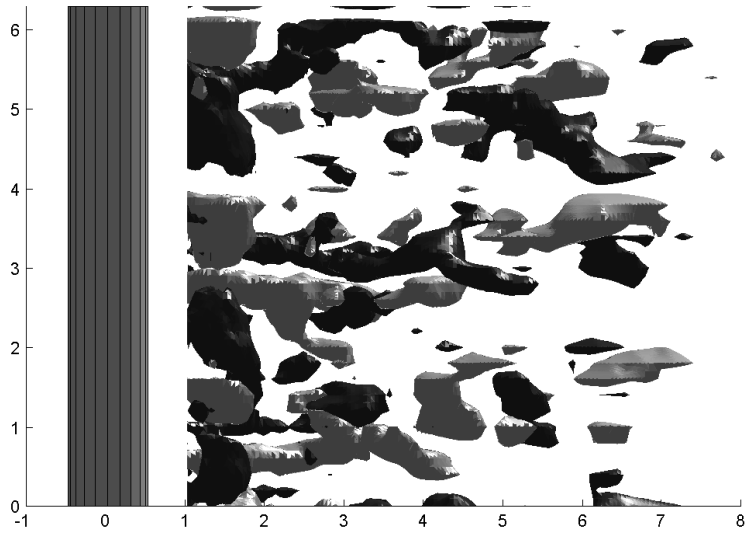

Figure II.13: Side view of isosurfaces of streamwise vorticity for solid cylinder at $tU_\infty/D = 47.6$ for $\omega_x = +1$ (black) and $\omega_x = -1$ (grey)

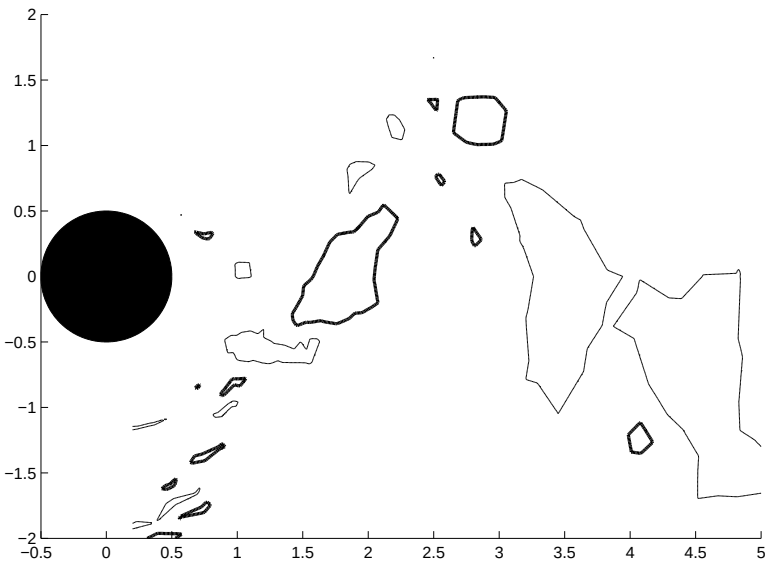

Figure II.14: Isocontours of streamwise vorticity for solid cylinder at $tU_\infty/D = 47.6$ at $z = 0$ for $\omega_x = +0.7$ (dark) and $\omega_x = -0.7$ (light)

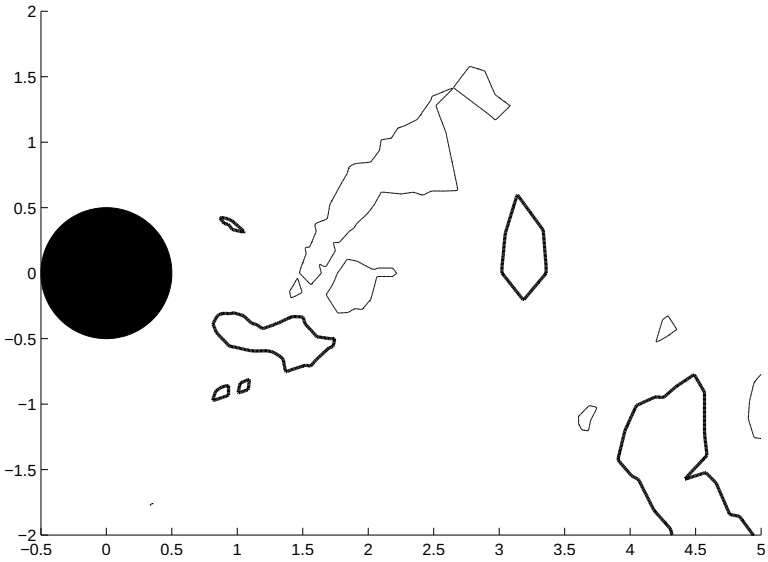

Figure II.15: Isocontours of streamwise vorticity for solid cylinder at $tU_\infty/D = 47.6$ at $z = \pi/2$ for $\omega_x = +0.7$ (dark) and $\omega_x = -0.7$ (light)

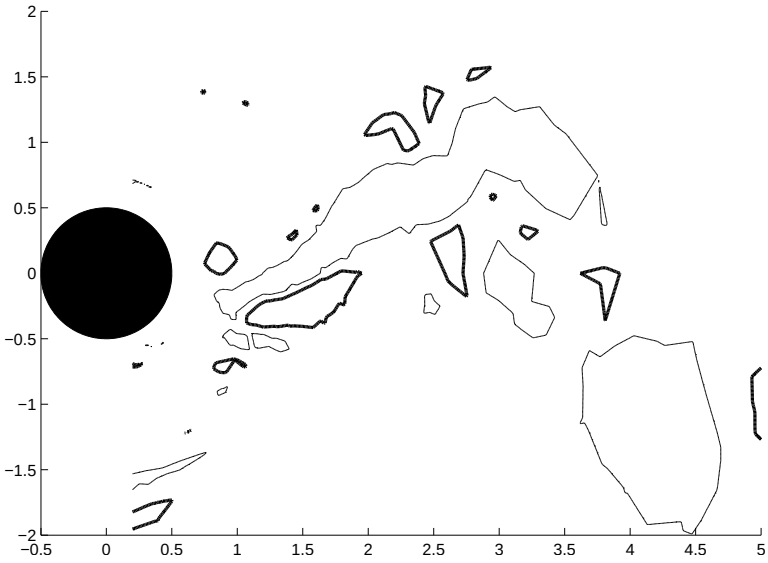

Figure II.16: Isocontours of streamwise vorticity for solid cylinder at $tU_\infty/D = 47.6$ at $z = \pi$ for $\omega_x = +0.7$ (dark) and $\omega_x = -0.7$ (light)

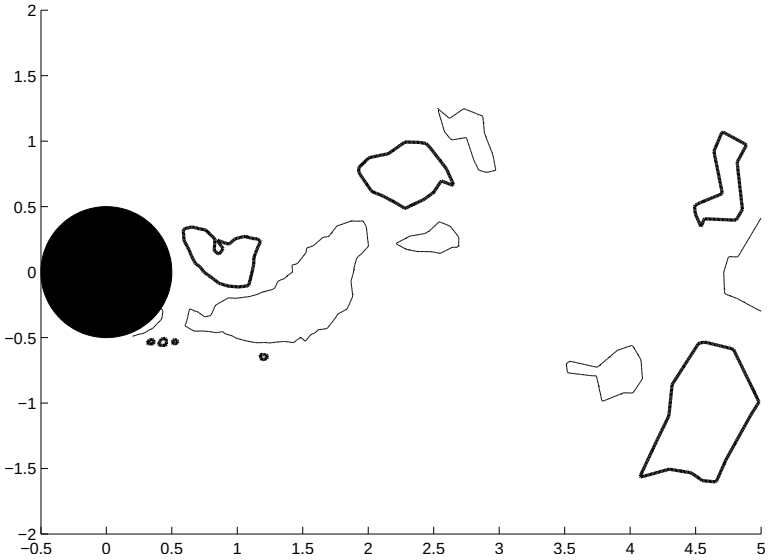

Figure II.17: Isocontours of streamwise vorticity for solid cylinder at $tU_\infty/D = 47.6$ at $z = 3\pi/2$ for $\omega_x = +0.7$ (dark) and $\omega_x = -0.7$ (light)

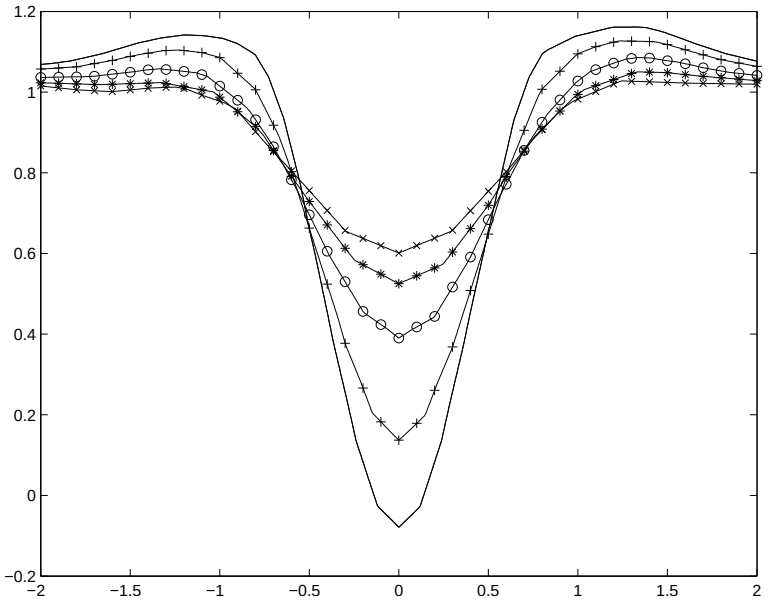

Figure II.18: Time-averaged mean streamwise velocity (u_x/U_∞) at 5 different downstream locations for solid cylinder case. The average velocities are plotted for $x/D = 1.2$ (solid), $x/D = 1.5$ (+), $x/D = 2$ ($\circ$), $x/D = 2.5$ ($*$), & $x/D = 3$ ($\times$). These values compare well with the results of Kravchenko et al. at $Re = 300$.

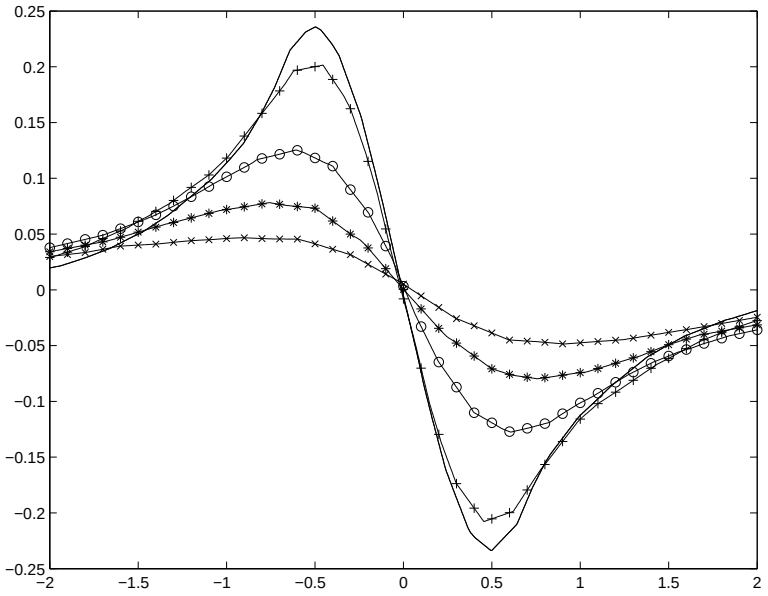

Figure II.19: Time-avearged mean crossflow velocity (u_y/U_∞) at 5 different downstream locations for solid cylinder case. The average velocities are plotted for $x/D = 1.2\,(\text{solid})$, $x/D = 1.5\,(+)$, $x/D = 2\,(\circ)$, $x/D = 2.5\,(*)$, & $x/D = 3\,(\times)$. These values compare well with the results of Kravchenko et al. at $Re = 300$.

validation case were used for the compliant surface case. The surface parameters were set to $(c/m) = (k/m) = 1$ for all three compliance zones. The velocity of the compliant surface in the radial direction was used as inner boundary conditions for the radial velocity field to satisfy no penetration boundary conditions. The flow field was initialized with a fully developed turbulent flow past a solid cylinder at Reynolds number of 300.

Figure II.20 shows the evolution of the lift coefficient for the solid cylinder and compliant cylinder cases. The solid line shows lift coefficient for turbulent flow past a solid cylinder for Reynolds number 300 and the dashed line shows lift coefficient for flow past a compliant cylinder with $(k/m) = 1$ and $(c/m) = 1$ for all three compliance zones for the same Reynolds number. We see a drop in the peak lift coefficient but no significant decrease in the Strouhal number as seen in the 2D case. The X-axis denotes nondimensional time tU_∞/D, and the compliant cylinder plot is shifted ahead to match with the fully developed solid cylinder plot. This result is similar in nature to 2D flow past a compliant cylinder as seen in (3).

Figure II.21 shows the evolution of the drag coefficient for the two cases. The solid line shows drag coefficient for turbulent flow past a solid cylinder for Reynolds number 300 and the dashed line shows drag coefficient for flow past a compliant cylinder with $(k/m) = 1$ and $(c/m) = 1$ for all three compliance zones for the same Reynolds number. We see an increase in time-average drag for the compliant cylinder case. The X-axis denotes nondimensional time tU_∞/D. The compliant cylinder drag is plotted from $tU_\infty/D = 50$ to $tU_\infty/D = 60$ and is denoted with a dashed line. The solid cylinder drag evolution is denoted by a solid line.

Figures II.22 to II.25 show surface plots for the compliant surface for four different time instants, $tU_\infty/D = 5.1$, 6.5, 7.6 and 9.3. The first and third time instants correspond to the maximum and minimum lift coefficients in the alternate vortex shedding cycle. The other two time instants occur in the middle of the shedding cycle. See Figure II.20 for evolution of the lift coefficient in the

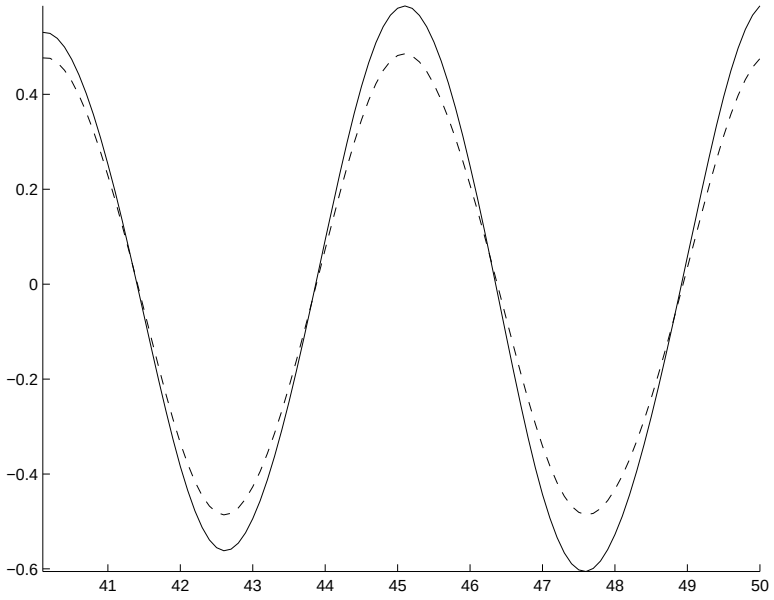

Figure II.20: Comparison of lift coefficient for solid cylinder case (solid) and compliant cylinder case (dashed) for $Re = 300$

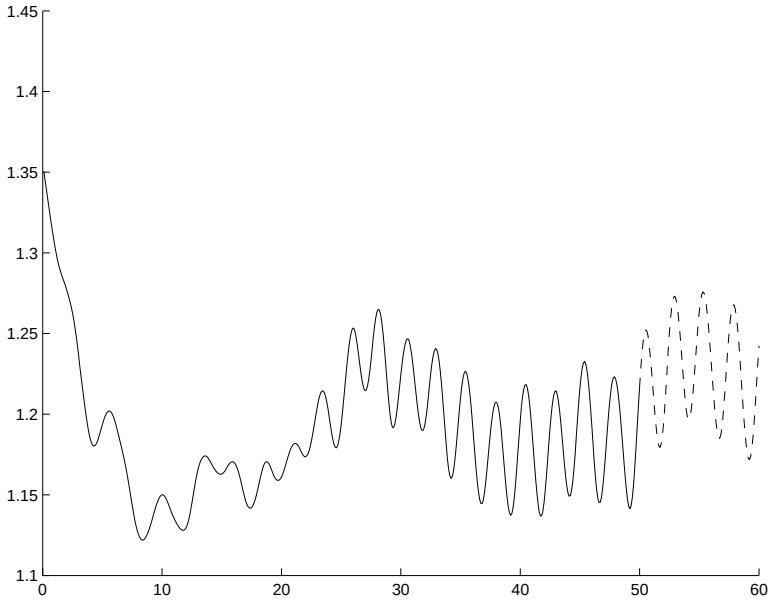

Figure II.21: Evolution of drag coefficient for solid cylinder case (solid) and compliant cylinder case (dashed) for $Re = 300$

compliant cylinder case.

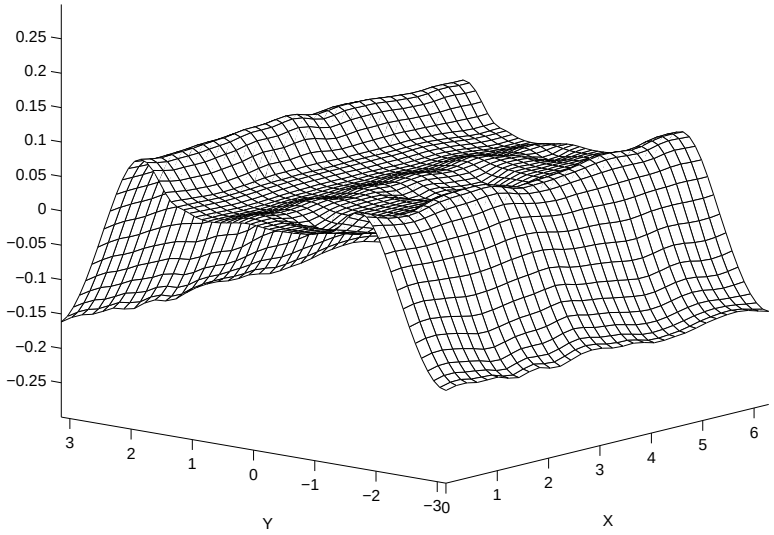

Figure II.22: Compliant surface at $tU\infty/D = 5.1$ for Re = 300. The X-axis axis shows axial coordinate (z), and the Y-axis shows azimuthal coordinate (θ)

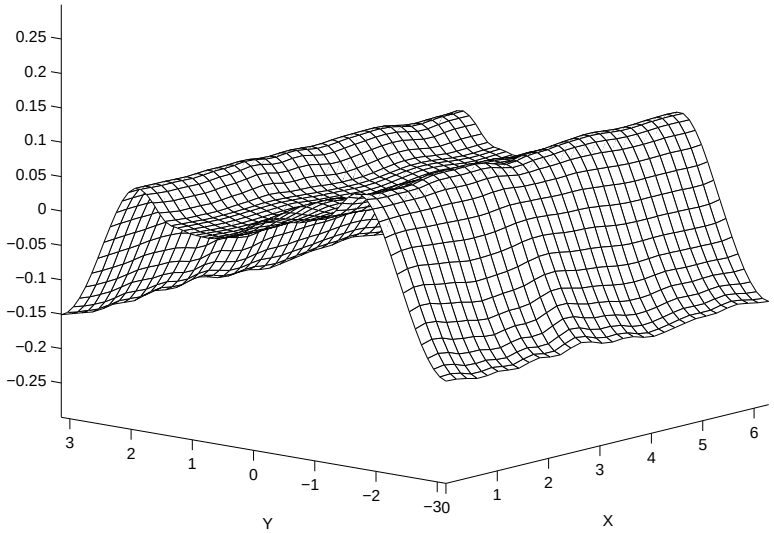

Figure II.23: Compliant surface at $tU\infty/D = 6.5$ for Re = 300. The X-axis axis shows axial coordinate (z), and the Y-axis shows azimuthal coordinate (θ)

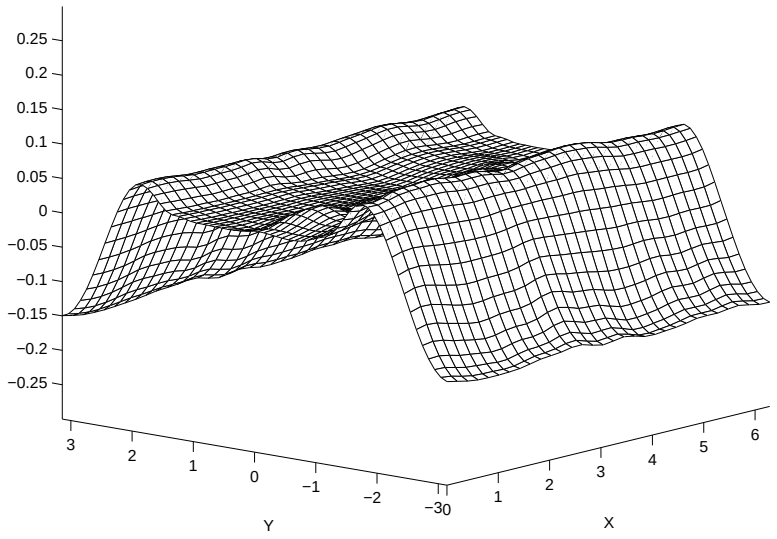

Figure II.24: Compliant surface at $tU\infty/D = 7.6$ for Re = 300. The X-axis axis shows axial coordinate (z), and the Y-axis shows azimuthal coordinate (θ)

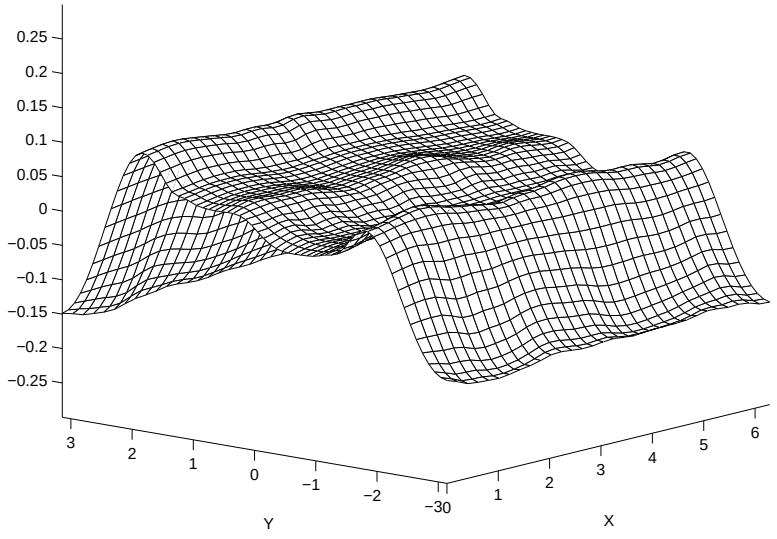

Figure II.25: Compliant surface at $tU\infty/D = 9.3$ for Re = 300. The X-axis axis shows axial coordinate (z), and the Y-axis shows azimuthal coordinate (θ)

II.E Transformed momentum and continuity equations

In the momentum equations, the terms treated explicitly, using RKW3 scheme are denoted by A_r, A_θ, and A_z while the terms treated implicitly with the CN scheme are denoted by B_r, B_θ and B_z. The pressure terms, handled by the fractional step update, are denoted by P_r, P_θ and P_z. Then, the complete transformed momentum and continuity equations are written out as,

$$\begin{aligned}
\frac{\partial u_r}{\partial \tau} &= A_r + B_r + P_r \\
\frac{\partial u_\theta}{\partial \tau} &= A_\theta + B_\theta + P_\theta \\
\frac{\partial u_z}{\partial \tau} &= A_z + B_z + P_z \\
0 &= \frac{\phi_r}{r}\frac{\partial (r u_r)}{\partial \xi_r} + \frac{1}{r}\frac{\partial u_\theta}{\partial \xi_\theta} + \frac{\phi_\theta}{r}\frac{\partial u_\theta}{\partial \xi_r} + \frac{\partial u_z}{\partial \xi_z} + \phi_z \frac{\partial u_z}{\partial \xi_r}
\end{aligned}$$

where the r-momentum terms are given by,

$$\begin{aligned}
A_r = &-\phi_t \frac{\partial u_r}{\partial \tau} - \frac{\phi_r}{r}\frac{\partial (r u_r u_r)}{\partial \xi_r} + \frac{u_\theta^2}{r} - \frac{1}{r}\frac{\partial (u_r u_\theta)}{\partial \xi_\theta} - \frac{\phi_\theta}{r}\frac{\partial (u_r u_\theta)}{\partial \xi_r} - \frac{2\nu}{r}\frac{\partial u_\theta}{\partial \xi_\theta} - \frac{2\nu}{r}\phi_\theta \frac{\partial u_\theta}{\partial \xi_r} \\
&- \frac{\partial (u_r u_z)}{\partial \xi_z} - \phi_z \frac{\partial (u_r u_z)}{\partial \xi_r} + \frac{\nu}{r^2}\frac{\partial h_{r\theta}}{\partial \xi_\theta} + \frac{\nu}{r^2}\phi_\theta \frac{\partial h_{r\theta}}{\partial \xi_r} + \nu \frac{\partial h_{rz}}{\partial \xi_z} + \nu \phi_z \frac{\partial h_{rz}}{\partial \xi_r} \\
B_r = &\ \nu \left[\frac{\phi_r^2}{r}\frac{\partial}{\partial \xi_r}\left(r \frac{\partial u_r}{\partial \xi_r} \right) - \frac{u_r}{r^2} \right] \\
P_r = &-\frac{1}{\rho}\phi_r \frac{\partial p}{\partial \xi_r}
\end{aligned}$$

where,

$$\begin{aligned}
h_{r\theta} &= \frac{\partial u_r}{\partial \theta} = \frac{\partial u_r}{\partial \xi_\theta} + \phi_\theta \frac{\partial u_r}{\partial \xi_r} \\
h_{rz} &= \frac{\partial u_r}{\partial z} = \frac{\partial u_r}{\partial \xi_z} + \phi_z \frac{\partial u_r}{\partial \xi_r}
\end{aligned}$$

The θ-momentum terms are given by,

$$A_\theta = -\phi_t \frac{\partial u_\theta}{\partial \xi_r} - \frac{\phi_r}{r}\frac{\partial (r u_r u_\theta)}{\partial \xi_r} - \frac{u_r u_\theta}{r} - \frac{1}{r}\frac{\partial (u_\theta u_\theta)}{\partial \xi_\theta} - \frac{\phi_\theta}{r}\frac{\partial (u_\theta u_\theta)}{\partial \xi_r} + \frac{2\nu}{r^2}\frac{\partial u_r}{\partial \xi_\theta} + \frac{2\nu}{r^2}\phi_\theta \frac{\partial u_r}{\partial \xi_r}$$
$$- \frac{\partial (u_\theta u_z)}{\partial \xi_z} - \phi_z \frac{\partial (u_\theta u_z)}{\partial \xi_r} + \frac{\nu}{r^2}\frac{\partial h_{\theta\theta}}{\partial \xi_\theta} + \frac{\nu}{r^2}\phi_\theta \frac{\partial h_{\theta\theta}}{\partial \xi_r} + \nu \frac{\partial h_{\theta z}}{\partial \xi_z} + \nu \phi_z \frac{\partial h_{\theta z}}{\partial \xi_r}$$
$$B_\theta = \nu \left[\frac{\phi_r^2}{r}\frac{\partial}{\partial \xi_r}\left(r \frac{\partial u_\theta}{\partial \xi_r} \right) - \frac{u_\theta}{r^2} \right]$$
$$P_\theta = -\frac{1}{\rho}\frac{1}{r}\frac{\partial p}{\partial \xi_\theta} - \frac{1}{\rho}\frac{1}{r}\phi_\theta \frac{\partial p}{\partial \xi_r}$$

where,

$$h_{\theta\theta} = \frac{\partial u_\theta}{\partial \theta} = \frac{\partial u_\theta}{\partial \xi_\theta} + \phi_\theta \frac{\partial u_\theta}{\partial \xi_r}$$
$$h_{\theta z} = \frac{\partial u_\theta}{\partial z} = \frac{\partial u_\theta}{\partial \xi_z} + \phi_z \frac{\partial u_\theta}{\partial \xi_r}$$

The z-momentum terms are given by,

$$A_z = -\phi_t \frac{\partial u_z}{\partial \xi_r} - \frac{\phi_r}{r}\frac{\partial (r u_r u_z)}{\partial \xi_r} - \frac{1}{r}\frac{\partial (u_\theta u_z)}{\partial \xi_\theta} - \frac{\phi_\theta}{r}\frac{\partial (u_\theta u_z)}{\partial \xi_r} - \frac{\partial (u_z u_z)}{\partial \xi_z} - \phi_z \frac{\partial (u_z u_z)}{\partial \xi_r}$$
$$+ \frac{\nu}{r^2}\frac{\partial h_{z\theta}}{\partial \xi_\theta} + \frac{\nu}{r^2}\phi_\theta \frac{\partial h_{z\theta}}{\partial \xi_r} + \nu \frac{\partial h_{zz}}{\partial \xi_z} + \nu \phi_z \frac{h_{zz}}{\xi_r}$$
$$B_z = \nu \left[\frac{\phi_r^2}{r}\frac{\partial}{\partial \xi_r}\left(r \frac{\partial u_z}{\partial \xi_r} \right) \right]$$
$$P_z = -\frac{1}{\rho}\frac{\partial p}{\partial \xi_z} - \frac{1}{\rho}\phi_z \frac{\partial p}{\partial \xi_r}$$

where,

$$h_{z\theta} = \frac{\partial u_z}{\partial \theta} = \frac{\partial u_z}{\partial \xi_\theta} + \phi_\theta \frac{\partial u_z}{\partial \xi_r}$$
$$h_{zz} = \frac{\partial u_z}{\partial z} = \frac{\partial u_z}{\partial \xi_z} + \phi_z \frac{\partial u_z}{\partial \xi_r}$$

The transformed Laplacian operator is defined as,

$$\begin{aligned} L(\phi) &= \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial \phi}{\partial r}\right) + \frac{1}{r^2}\frac{\partial^2 \phi}{\partial \theta^2} + \frac{\partial^2 \phi}{\partial z^2} \\ &= \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial \phi}{\partial r}\right) + \frac{1}{r^2}\frac{\partial}{\partial \theta}\left(\frac{\partial \phi}{\partial \theta}\right) + \frac{\partial}{\partial z}\left(\frac{\partial \phi}{\partial z}\right) \\ &= \frac{\phi_r^2}{r}\frac{\partial}{\partial \xi_r}\left(r\frac{\partial \phi}{\partial \xi_r}\right) + \frac{1}{r^2}\frac{\partial L_\theta}{\partial \xi_\theta} + \frac{\phi_\theta}{r^2}\frac{\partial L_\theta}{\partial \xi_r} + \frac{\partial L_z}{\partial \xi_z} + \phi_z\frac{\partial L_z}{\partial \xi_r} \end{aligned}$$

where,

$$\begin{aligned} L_\theta &= \frac{\partial \phi}{\partial \theta} = \frac{\partial \phi}{\partial \xi_\theta} + \phi_\theta\frac{\partial \phi}{\partial \xi_r} \\ L_z &= \frac{\partial \phi}{\partial z} = \frac{\partial \phi}{\partial \xi_z} + \phi_z\frac{\partial \phi}{\partial \xi_r} \end{aligned}$$

The transformed implicit part of Laplacian is defined as,

$$L^{imp}(\phi) = \frac{1}{\xi_r}\frac{\partial}{\partial \xi_r}\left(\xi_r\frac{\partial \phi}{\partial \xi_r}\right) + \frac{1}{\xi_r^2}\frac{\partial^2 \phi}{\partial \xi_\theta^2} + \frac{\partial^2 \phi}{\partial \xi_z^2}$$

It follows that the explicit part of the Laplacian is given by,

$$L^{exp}(\phi) = L(\phi) - L^{imp}(\phi)$$

The transformed divergence operator is given by,

$$\begin{aligned} \nabla \cdot \mathbf{u} &= \frac{1}{r}\frac{\partial}{\partial r}(ru_r) + \frac{1}{r}\frac{\partial u_\theta}{\partial \theta} + \frac{\partial u_z}{\partial z} \\ &= \frac{\phi_r}{r}\frac{\partial}{\partial \xi_r}(ru_r) + \frac{1}{r}\frac{\partial u_\theta}{\partial \xi_\theta} + \frac{\phi_\theta}{r}\frac{\partial u_\theta}{\partial \xi_r} + \frac{\partial u_z}{\partial \xi_z} + \phi_z\frac{\partial u_z}{\partial \xi_r} \end{aligned}$$

The transformed gradient operator is given by,

$$\begin{aligned} \nabla\phi &= \left(\frac{\partial \phi}{\partial r}, \frac{1}{r}\frac{\partial \phi}{\partial \theta}, \frac{\partial \phi}{\partial z}\right) \\ &= \left(\phi_r\frac{\partial \phi}{\partial \xi_r}, \frac{1}{r}\frac{\partial \phi}{\partial \xi_\theta} + \frac{\phi_r}{r}\frac{\partial \phi}{\partial \xi_r}, \frac{\partial \phi}{\partial \xi_z} + \phi_z\frac{\partial \phi}{\partial \xi_r}\right) \end{aligned}$$

Thus, it can be verified that $\nabla \cdot (\nabla \phi) = \nabla^2 \phi$ in the transformed coordinatesystem, which is necessary for the fractional step method to correctly remove divergence form the intermediate flow field.

II.F Conclusion

We have developed an accurate and stable method to perform direct numerical simulation of turbulent flow past a circular cylinder with a compliant surface. This work is an extension of a previous method developed by us for DNS of 2D unsteady flow past a compliant cylinder (see (3)). We have validated the code for flow past a solid cylinder at a Reynolds number of 300. We have also presented results for turbulent flow past a circular cylinder with a compliant surface in this work. According to our results, the peak lift coefficient of 3D flow past a cylinder *decreases* as surface compliance is increased. Also, the time average drag *increases* as compliance is added.

In future work, we plan to use this method together with an efficient surrogate management framework based (see (13)) optimization routine to minimize the aeroacoustic noise produced by the turbulent flow. This method can further be used to minimize blade vortex interaction noise in helicopter blades, and to minimize sound produced by a submarine in its wake.

This chapter, in full, has been submitted for publication to Journal of Computational Physics as, Anish Karandikar and T. R. Bewley, *Direct numerical simulation of turbulent 3D flow past a circular cylinder with a compliant surface.*

Bibliography

[1] H. Luo, T. R. Bewley, Accurate simulation of near-wall turbulence over a compliant tensegrity fabric, SPIE Paper 5757-17.

[2] H. Luo, T. R. Bewley, On the contravariant form of the Navier-Stokes equation in time-dependent curvilinear coordinate systems, J. Comp. Phys. 199, p. 355.

[3] A. Karandikar, T. R. Bewley, Accurate simulation of unsteady 2D flow past a compliant cylinder, Submitted to J. Comp. Phys.

[4] S. Xu, D. Rempfer, J. Lumley, Turbulence over a compliant surface: numerical simulation and analysis, J. Fluid Mech. 478, p. 11.

[5] Arthur G. Kravchenko, Parviz Moin, Karim Shariff, B-spline method and zonal grids for simulations of complex turbulent flows, Journal of Computational Physics 151, 1999, p. 757.

[6] T. Endo, R. Himeno, Direct numerical simulation of flow across a compliant cylinder, RIKEN Review 48, October 2002.

[7] N. Hasan, S. F. Anwer, S. Sanghi, On the outflow boundary condition for external incompressible flows: A new approach, J. Comp. Phys. 206, 2005, p. 661.

[8] P. Carpenter, P. Morris, The effect of anisotropic wall compliance on boundary-layer stability and transition, J. Fluid Mech. 218, p. 171.

[9] P. Carpenter, A. Garrad, The hydrodynamic stability of flow over Kramer-type compliant surfaces, J. Fluid Mech. 155, p. 465.

[10] C. Davies, P. Carpenter, Numerical simulations of the evolution of Tollmien-Schlichting waves over nite compliant panels, J. Fluid Mech. 335, p. 361.

[11] T. Endo, R. Himeno, Direct numerical simulation of turbulent flow over a compliant surface, J. Turbulence 3(007), p 1.

[12] T. R. Bewley, P. Moin, R. Temam, DNS-based predictive control of turbulence: an optimal benchmark for feedback algorithms, J. Fluid. Mech. 447, 2001, p. 179.

[13] A. Booker, J. Dennis Jr, P. Frank, D. Serafini, V. Torczon, M. Trosset, A rigorous framework for optimization of expensive functions by surrogates, Structural and Multidisciplinary Optimization, 17, 1999, p. 113.

[14] A. L. Marsden, M. Wang, J. E. Dennis, P. Moin, Suppression of vortex-shedding noise via derivative-free shape optimization, Physics of Fluids, 16(10) 2004, p. L83.

[15] H. Carlson, G. Berkooz, J. L. Lumley, Direct numerical simulation of flow in a channel with complex, time-dependent wall geometries: A pseudospectral method, J. Comp. Phys. 121, 1995, p. 155.

[16] Y. Mito, N. Kasagi, DNS study of turbulence modification with streamwise-uniform sinusoidal wall-oscillation, Int. J. Heat & Fluid Flow 19, 1998, p. 470.

III

An iterative left/right LP method for the efficient optimization of a highly constrained resistor network

III.A Introduction

The present work was motivated by the recent advent of porous-media heat exchangers with porosity that may be tailored for any given application, thus apparently providing the opportunity to significantly improve efficiency of distributed heat exchangers for electronics cooling applications. In simple configurations, such as when all the heat producing components are localized in one specific area, the porosity distribution in such a heat exchanger can be optimized by inspection. In the more general case, the distribution of porosity that leads to the most efficient coolant flow is not obvious, and an efficient optimization procedure is necessary.

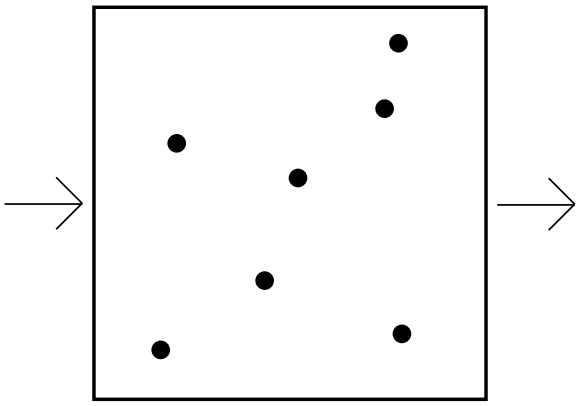

Figure III.1: Schematic of an electronic circuit board that needs to be cooled with hot spots marked with dark circles. The flow of the coolant is in a variable-width channel within the printed circuit board from left to right.

A similar configuration in which tunable cooling can be acheived is a variable channel width heat exchanger. This configuration is, in fact, significantly more efficient for liquids with high thermal conductivity. Figure III.1 shows a schematic for a liquid-cooled electronic circuit board with certain hot spots spread out over the domain as indicated by the dark circles. The coolant flows from left to right. The optimization problem considered is to deteremine the width of the channel, δ, as a function of x and y that produces the optimal cooling. [Note that

Table III.1: Electrical analogy for variable width channel flow

Flow (Cooling) Problem	**Electrical (Heating) Problem**
Pressure	Voltage
Flow rate	Current
Inverse of channel width	Resistance
$h(x,y)$ = Heating input (specified)	—
$T(x,y) \leq T_{max}$ (Max. temperature)	$H(x,y) \geq H_{min}(x,y)$ (Min. heating)

the origin is assumed to be at the bottom-left corner of the circuit board, and the x-axis is aligned with the bottom boundary]. By optimal cooling we mean a distribution of channel widths in the heat exchanger such that the maximum coolant temperature is bounded everywhere in the domain (say $\leq 80°C$), and the power required to drive the coolant through the heat exchanger is minimum.

The development of efficient interior point linear programming (LP) solvers have made available a powerful tool to simplify constrained quadratic minimization problems for PDE systems. The benefits garnered should extend to a range of other problems in flow control. See (1) and (2) for further discussion.

In order to study the essential ingredients of the variable porosity or variable channel width heat exchanger problems, we examine a warm-up problem with a similar structure, a heating blanket problem. We apply the electrical analogy to a low Reynolds number Stokes flow in a channel, and consider as a first model, a grid of horizontal and vertical resistors. The resistances in the electrical problem correspond to the inverse of channel width / porosoity in the fluid case. The currents in the heating problem correspond to the velocity or flow rate in the flow problem. The voltages in the electrical problem correspond to the pressures in the flow case. Finally, instead of minimizing the power required to drive the flow through the heat exchanger (while staying below a maximum allowed coolant temperature) as done in the flow problem, we minimize the power required to drive the total current in the circuit while maintaining a *minimum* heating requirement at specified locations on the heating grid. This analogy is summarized in Table III.1.

In section III.B we describe, the notation used for the heating blanket problem, and also give a formal statement of the optimization problem. In section III.C we describe, the solution method adopted to solve the heating blanket optimization problem by splitting the problem into two parallel left/right linear programming problems. In section III.D, we describe an alternate method for solving the heating blanket optimization problem by splitting the problem into an inner LP problem and an outer gradient based optimization problem, and also illustrate some of the difficulties faced in this approach. In section III.E, we solve a 10×10 sized heating blanket problem using the left/right LP solver and compare the solution with a full nonlinear optimization solver (SNOPT). In section III.F we summarize, the results for three problems of increasing grid size (32×32, 64×64 and 128×128) in each direction using the left/right LP solver.

III.B The resistor network

In this section, as a warm-up to solving the heat exchanger problem, we examine a heating blanket in the form of a grid of horizontal and vertical resistors (See Figure III.2) with a specified minimum heating requirement on certain locations on the grid. The notation for the resistor grid heating blanket is specified in the following subsection.

III.B.1 Notation for resistor network

Referencing Figure III.2, the following notation is used:

Interior nodes:	$\{i, j\}$	for $i = 1, \ldots, N-1;$	$j = 0, \ldots M;$
Interior cells:	$\{i + 1/2, j + 1/2\}$	for $i = 0, \ldots, N-1;$	$j = 0, \ldots M-1;$
Horizontal paths:	$\{i + 1/2, j\}$	for $i = 0, \ldots, N-1;$	$j = 0, \ldots M;$
Vertical paths:	$\{i, j + 1/2\}$	for $i = 1, \ldots, N-1;$	$j = 0, \ldots M-1.$

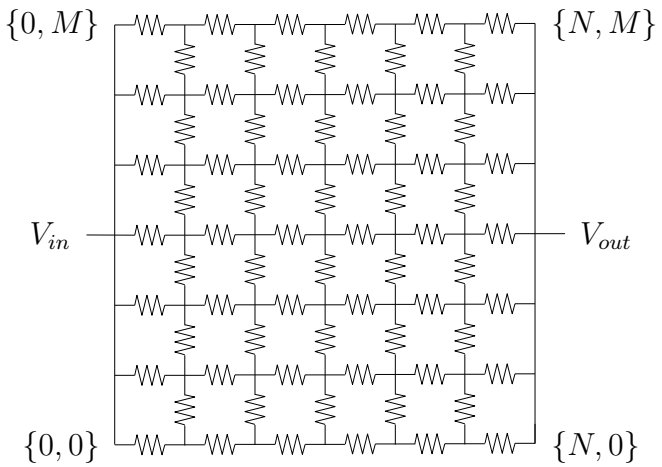

Figure III.2: A simple resistor network. We first optimize the values of the resistors and the total voltage drop $V_{in} - V_{out}$ to minimize the total power expended while a prespecified heating distribution.

The voltage $V^{i,j}$ is defined on every interior node. A horizontal resistor $R_1^{i+1/2,j}$ and current component $I_1^{i+1/2,j}$ are defined for every horizontal path and a vertical resistor $R_2^{i,j+1/2}$ and current component $I_2^{i,j+1/2}$ are defined for every vertical path. Kirchoff's current law (KCL) for every interior node may be written as,

$$(I_1^{i+1/2,j} - I_1^{i-1/2,j}) + (I_2^{i,j+1/2} - I_2^{i,j-1/2}) = 0. \tag{III.1}$$

The voltage drop for every horizontal and vertical path, respectively, may be written as,

$$(V^{i+1,j} - V^{i,j}) + R_1^{i+1/2,j}\, I_1^{i+1/2,j} = 0 \tag{III.2a}$$

$$(V^{i,j+1} - V^{i,j}) + R_2^{i,j+1/2}\, I_2^{i,j+1/2} = 0. \tag{III.2b}$$

The boundary conditions on this system are given by,

$$V^{0,*} = V_{in}, \qquad V^{N,*} = V_{out}, \qquad I_2^{*,-1/2} = I_2^{*,M+1/2} = 0, \tag{III.3}$$

where a star (*) in the superscript means all values of the index for which the corresponding variable is defined. Without loss of generality, we define $V_{out} = 0$. The range of acceptable resistor values are:

$$\underline{R} \le R_1^{*,*} \le \overline{R}, \qquad \underline{R} \le R_2^{*,*} \le \overline{R}. \tag{III.4}$$

The heating requirement on every interior cell is given by

$$\begin{aligned}\frac{1}{2}\Big[&\Big(V^{i+1,j}-V^{i,j}\Big)I_1^{i+1/2,j}+\Big(V^{i+1,j+1}-V^{i,j+1}\Big)I_1^{i+1/2,j+1}\\&+\Big(V^{i,j+1}-V^{i,j}\Big)I_2^{i,j+1/2}+\Big(V^{i+1,j+1}-V^{i+1,j}\Big)I_2^{i+1,j+1/2}\Big]\geq H^{i+1/2,j+1/2},\end{aligned}\tag{III.5}$$

where we have defined the boundary resistor values $R_2^{0,*}=R_2^{N,*}=0$.

Defining the total current in the network as

$$I_{total}=\sum_{j=0}^{M}I_1^{i+1/2,j}\qquad\text{for any } i.\tag{III.6}$$

III.B.2 Problem statement

We now give our first formal statement of the heating blanket problem.

Problem A: Given specifications on the minimum and maximum resistor values $\underline{R}$ and $\overline{R}$ and the required heating distribution $H^{*,*}>0$, minimize the total power consumed,

$$J=(V_{in}-V_{out})I_{total},$$

with respect to the resistor values $R_1^{*,*}$ and $R_2^{*,*}$ and the voltage drop $V_{in}-V_{out}$ while satisfying the equality constraints (III.1)-(III.3) and the inequality constraints (III.4)-(III.5).

We can assume, without loss of generality, that $V_{out}=0$. Thus, the modified cost function is,

$$J=V_{in}\cdot I_{total}\tag{III.7}$$

At the outset, this looks like a fairly easy problem. Indeed, if $\overline{R}=\infty$, one can minimize the power consumed simply by setting $R_2^{*,*}=\infty$ (so the current on each vertical path is zero) and selecting the voltage drop $V_{in}-V_{out}$ to be sufficiently large. In this case, it is possible to apply (III.5) as an equality constraint on every cell and effectively ignore the inequality constraints (III.4), as none of these inequality constraints will be binding in the optimized solution for sufficiently large $V_{in}-V_{out}$. Problem A then converts to a reduced set of nonlinear equations which

is easily solved via standard techniques, such as Newton-Raphson. As the power consumed on each cell is the minimum that meets the specifications (III.5), this solution is globally optimal.

Similarly, if $\underline{R} = 0$, one can minimize the power consumed by setting $R_2^{*,*} = 0$ (so the voltage on each vertical line is constant) and selecting the voltage drop $V_{in} - V_{out}$ to be sufficiently small. In this case, it is also possible to apply (III.5) as an equality constraint on every cell and effectively ignore the inequality constraints (III.4), as none of these inequality constraints will be binding in the optimized solution for sufficiently small $V_{in} - V_{out}$. Problem A then converts to a reduced set of nonlinear equations which is easily solved via standard techniques. As the power consumed on each cell is the minimum that meets the specifications (III.5), this solution is also globally optimal.

When the range of acceptable resistor values is small, such as $\overline{R}/\underline{R} = 10$, the many binding inequality constraints on this problem appear to make it much more difficult. However, there is a novel way to get around this problem which we describe in the following section.

III.C Solution method

The optimization problem described by (III.1) - (III.7) is a quadraticlaly constrained quadratic programming problem (QCQP). QCQP's are, in the general case NP-hard problems. They do not scale well with problem dimension. For the heating blanket problem, the heating requirement (III.5) is a quadratic inequality constraint. Kirchoff's Voltage Law (III.2) are the quadratic equality constraints. The cost function itself (III.7) is of course quadratic as it is given by $V_{in} \cdot I_{total}$.

To reduce the computational effort involved in solving the optimization problem and finding a local minimum, we split the problem into two parallel Linear Programming (LP) problems: LeftLP and RightLP. The strategy is to split the free variables into two sets, left and right, such that the governing equations (and

cost function) are *linear* in any particular set of variables, while the other set is held constant.

One complete iteration of the left/right solver comprises of one run of a LP solver for the LeftLP for a specified number of maximum iterations and one run of LP solver for the RightLP for a specified number of maximum iterations. Further, the solution procedure can be modified to first solve for the given heating requirement on a coarse grid (say 8×8 grid of resistors), then use this solution to initialize a solve on a finer grid (say 16×16 grid), and so on till required refinement of the solution is achieved.

III.C.1 LeftLP

The free and fixed variables for the LeftLP are defined as,

$$\text{Free:} \quad \begin{pmatrix} V_{in} \\ V^{*,*} \\ R_1^{*,*} \\ R_2^{*,*} \end{pmatrix} \qquad \text{Fixed:} \quad \begin{pmatrix} I_{total} \\ I_1^{*,*} \\ I_2^{*,*} \end{pmatrix}$$

In the LeftLP, $\mathbf{x}$ are the free variables while $\mathbf{y}_c$ are fixed. The governing equations are linear in $\mathbf{x}$, and the problem can be cast as an LP. Kirchoff's Voltage Law (III.2) is linear in the voltages and resistances with fixed currents. For convenience, they are listed here again with the fixed variables marked.

$$\begin{aligned} (V^{i+1,j} - V^{i,j}) + R_1^{i+1/2,j} \overbrace{I_1^{i+1/2,j}}^{\text{fixed}} &= 0 \\ (V^{i,j+1} - V^{i,j}) + R_2^{i,j+1/2} \underbrace{I_2^{i,j+1/2}}_{\text{fixed}} &= 0. \end{aligned}$$

The acceptable range of resistance values (III.4) are bounds on the resistance part of $\mathbf{x}$. Also the inequality constraints (III.5) are linear in voltages. Thus

the LeftLP can be expressed as,

$$\begin{aligned} & min\, \mathbf{c}_1^T \mathbf{x} \\ & s.t.\ A\,\mathbf{x} = 0 \\ & \quad B\,\mathbf{x} \geq \mathbf{h}_1 \end{aligned} \tag{III.8}$$

In the LP formulation, the cost function vector, $\mathbf{c}_1$, comes from (III.7) The A-matrix is a function of $\mathbf{y}_c$ and comes from a combination of Kirchoff's Volatge Law (III.2), Kirchoff's Current Law (III.1) and the equation enforcing total current in the network (III.6). The B-matrix (which is also a function of $\mathbf{y}_c$) and $\mathbf{h}_1$ come from the heating requirement (III.5). Thus the constraint matrices on the LeftLP are imposed by a solution of the RightLP and vice versa.

Figure III.3 shows a typical sparsity pattern for an equality matrix (A) for the LeftLP for a 10 by 10 grid. Figure III.4 shows a typical sparsity pattern for an inequality matrix (B) for the LeftLP for a 10 by 10 grid.

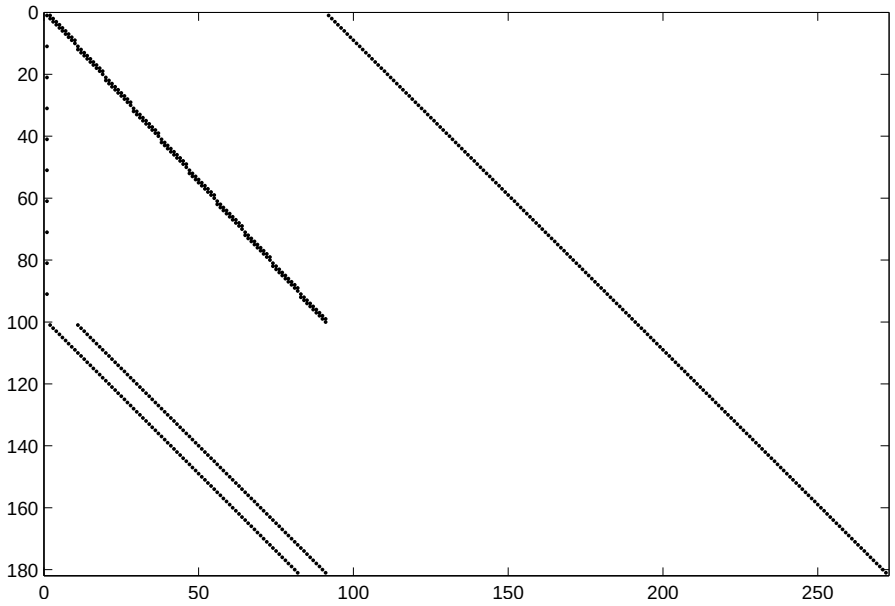

Figure III.3: Sparsity pattern for equality matrix of LeftLP (10×10 grid)

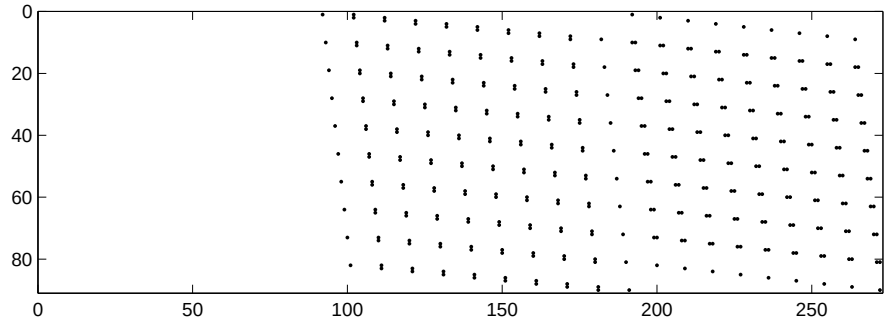

Figure III.4: Sparsity pattern for inequality matrix of LeftLP (10×10 grid)

III.C.2 RightLP

To cast the right problem as a linear program, we have to define a new set of free variables, conductance (G), defined as,

$$G_1^{*,*} = \frac{1}{R_1^{*,*}}, \qquad G_2^{*,*} = \frac{1}{R_2^{*,*}}$$

where G_1 is the conductance of horizontal resistors, and G_2 is the conductance of vertical resistors.

Thus, Kirchoff's Voltage Law (III.2) can be written in terms of conductances as,

$$\overbrace{(V^{i+1,j} - V^{i,j})}^{\text{fixed}} G_1^{i+1/2,j} + I_1^{i+1/2,j} = 0 \tag{III.9a}$$

$$\underbrace{(V^{i,j+1} - V^{i,j})}_{\text{fixed}} G_2^{i,j+1/2} + I_2^{i,j+1/2} = 0. \tag{III.9b}$$

for horizontal and vertical path. Note that (III.9) is exactly equivalent to (III.2).

Now, for the RightLP, the free and fixed variables are,

$$\text{Free:} \quad \begin{pmatrix} I_{total} \\ I_1^{*,*} \\ I_2^{*,*} \\ G_1^{*,*} \\ G_2^{*,*} \end{pmatrix} \qquad \text{Fixed:} \quad \begin{pmatrix} V_{in} \\ V^{*,*} \end{pmatrix}$$

Notice that, the RightLP mirrors the LeftLP. The $\mathbf{y}$ variables are free while $\mathbf{x}_c$ are fixed. The governing equations (with (III.2) which is in terms of resistances replaced by (III.9) which is in terms of conductances) are linear in the free variables, and the right problem can be cast as a linear program.

Kirchoff's Voltage Law, recast as (III.9) is linear in the conductances and currents on the horizontal and vertical resistors with fixed voltages. Kirchoff's Current Law (III.1) is also linear on the currents in the resistors. Also, the equation (III.6) enforcing total current in the network for any i is a linear relation. These three things together constitute the linear equality relations of the LP. As in the LeftLP, the heating requirement (III.5) poses the linear inequality relations in the linear program. Note that the bounds on the conductances are flipped with respect to the bounds on resistances (III.4), due to the reciprocal relationship between resistance and conductance. By taking reciprocal of (III.4) we have,

$$\frac{1}{\underline{R}} \geq \frac{1}{R_1^{*,*}} \geq \frac{1}{\overline{R}}, \qquad \frac{1}{\underline{R}} \geq \frac{1}{R_2^{*,*}} \geq \frac{1}{\overline{R}}$$

Defining the lower ($\underline{G}$) and upper ($\overline{G}$) limits for conductances as,

$$\underline{G} = \frac{1}{\overline{R}}, \qquad \overline{G} = \frac{1}{\underline{R}}$$

we have the bounds on the conductance part of $\mathbf{y}$ as,

$$\underline{G} \leq G_1^{*,*} \leq \overline{G}, \qquad \underline{G} \leq G_2^{*,*} \leq \overline{G}. \tag{III.10}$$

Now, after this manipulation, the RightLP can be expressed as a linear

program as,

$$\begin{aligned} & min\, \mathbf{c}_2^T \mathbf{y} \\ & s.t.\ C\,\mathbf{y} = 0 \\ & \quad D\,\mathbf{y} \geq \mathbf{h}_2 \end{aligned} \tag{III.11}$$

Again, the cost function vector $\mathbf{c}_2$ comes from (III.7). The equality matrix C is a function of $\mathbf{x}_c$ comes from the restatement of Kirchoff's Voltage Law in terms of conductance (III.9), Kirchoff's Current Law (III.1) and the equation enforcing total current in the circuit (III.6). The inequality matrix D (which is also a function of $\mathbf{x}_c$) and $\mathbf{h}_2$ for the LP come from the heating requirement (III.5). Thus the constraint matrices on the RightLP are imposed by a solution of the LeftLP and vice versa.

Figure III.5 shows a typical sparsity pattern for an equality matrix (C) for the RightLP for a 10 by 10 grid. Figure III.6 shows a typical sparsity pattern for an inequality matrix (D) for the RightLP for a 10 by 10 grid.

Note that each complete iteration for the left/right LP solver consists of 1 LeftLP run for a specified number of maximum inner iterations and 1 RightLP run for a specified number of maximum inner iterations.

With the left and right LP's defined, each LP is solved for a predetermined number of iterations alternately. First a few iterations of the LeftLP are performed, and then the solution is imposed as constraints (through the constraint matrices) on the RightLP. Then the RightLP is iterated for a few steps and its solution is imposed on the LeftLP through its constraint matrices, and so on.

III.D Alternate method (sensitivity analysis) and its problems

Another possible method to solve this problem is detailed in this section. In this method, the problem variables are split into inner and outer parts

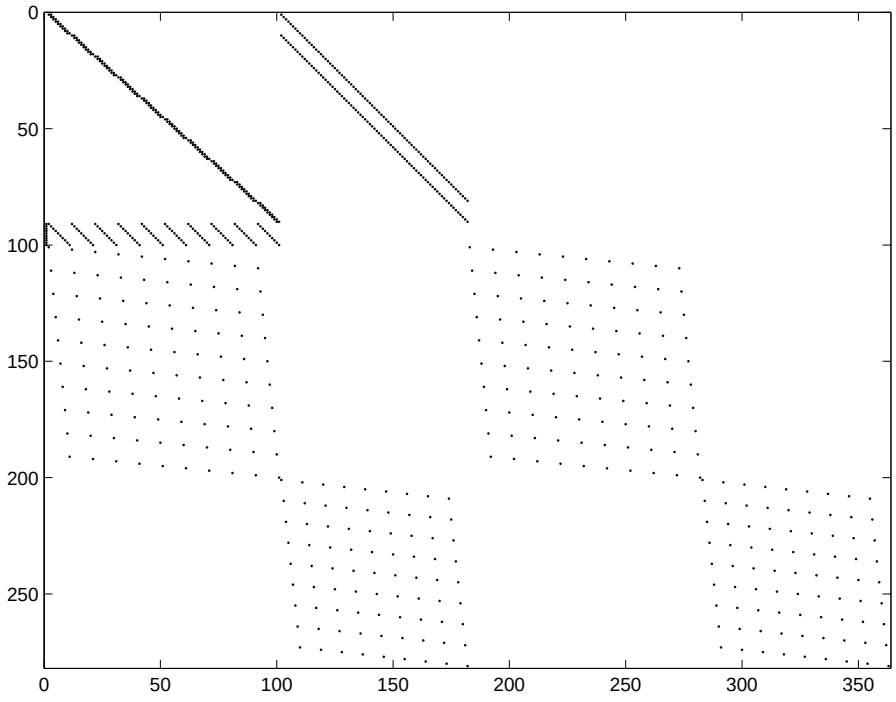

Figure III.5: Sparsity pattern for equality matrix of RightLP (10×10 grid)

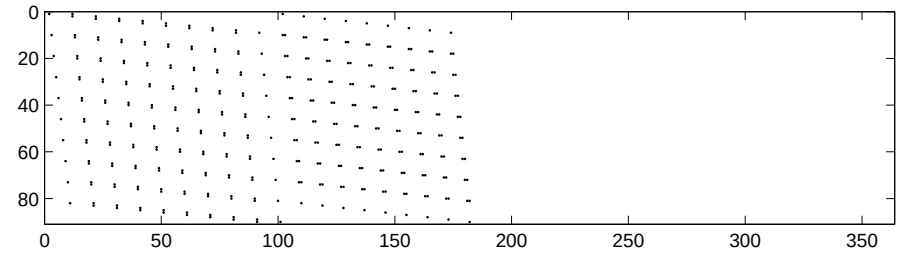

Figure III.6: Sparsity pattern for inequality matrix of RightLP (10×10 grid)

instead of left and right LPs. Note that for a given current distribution, Kirchoff's Voltage Law (III.2), the inequality heating requirement (III.5) and the objective function (III.5) are all linear in the voltages and resistances in the circuit. Thus, the inner/outer formulation of the problem is given as,

Inner Problem: Given specifications on the minimum and maximum resistor values $\underline{R}$ and $\overline{R}$, the required heating distribution $H^{*,*} > 0$, and a prescribed (feasible) current distribution parameterized by $\{I_{total}, I_1^{*,*}, I_2^{*,*}\}$, minimize the total power consumed, J (defined in (III.7)), with respect to V_{in}, $V^{*,*}$, $R_1^{*,*}$, and $R_2^{*,*}$, subject to the equality constraints (III.2)-(III.3) and the inequality constraints (III.4)-(III.5).

Outer Problem: Minimize the optimized power consumption, J, with respect to the current distribution $\{I_{total}, I_1^{*,*}, I_2^{*,*}\}$, where the feasibility of a particular current distribution and the corresponding optimized power consumption J is found by solving problem B.1.

As noted above, the inner problem is linear in its unknowns and can be solved effeciently as an LP. Thus given an initial feasible current distribution, the outer problem can be solved by repeated solution of the inner problem. Also, in addition to solving the inner problem the LP algorithm gives gradient information which incorporates the binding feasibility constraints, thus guiding the next step in the iterative solution refinement of the outer problem.

The unknowns in the inner and outer problems are defined as,

$$\mathbf{x}_{inner} = \begin{pmatrix} V_{in} \\ V^{*,*} \\ R_1^{*,*} \\ R_2^{*,*} \end{pmatrix} \qquad \text{and} \qquad \mathbf{y}_{outer} = \begin{pmatrix} I_{total} \\ I_1^{*,*} \\ I_2^{*,*} \end{pmatrix}$$

An important requirement for the success of this algorithm is the ability for the LP solver to determine gradient information from the inner problem, so that the outer gradient based solver can use this information to reduce function evaluations. This gradient information is derived from the sensitivity analysis of the inner LP. But the sensitivity of the inner LP to perturbations of the constraint matrices or the right hand sides is a nonsmooth function. This can be seen in a simple example.

III.D.1 Nonsmoothness of LP optimum with respect to perturbations to right hand side of constraints

Consider an LP in canonical form,

$$\begin{aligned} &min \quad J = \mathbf{c}^T \mathbf{x} \\ &s.t. \quad A\,\mathbf{x} \leq \mathbf{b}_0, \quad \mathbf{x} \geq 0 \end{aligned}$$

We consider a test case, called model LP with,

$$\mathbf{c} = \begin{pmatrix} -5 \\ -4 \\ -6 \end{pmatrix}, \quad A = \begin{pmatrix} 1 & -1 & 1 \\ 3 & 2 & 4 \\ 3 & 2 & 0 \end{pmatrix}, \quad \mathbf{b}_0 = \begin{pmatrix} 20 \\ 30 \\ 30 \end{pmatrix}$$

The optimum of this base LP is, $\mathbf{x}^* = [\ 0\ 15\ 0\ ]^T$ with optimum cost function, $J^* = -60$. We now perturb the second element of the RHS vector, $\mathbf{b}_0$ by a small amount t as,

$$\mathbf{b} = \mathbf{b}_0 + t \cdot \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

The amount of perturbation is $t = \pm 0.1$ in intervals of 0.01. For each perturbed right hand side, the LP is solved and optimal cost function is plotted against perturbation. The analytical slope of the cost function with respect to RHS perturbations is given by sensitivity analysis as,

$$\frac{\partial J^*}{\partial t} = -\lambda_i \tag{III.12}$$

where λ_i is the Lagrangian multiplier corresponding to the i-th constraint. In this case $i = 2$. See (3) for more details.

Figure III.7 shows that there is a discontinuity in the slope at $t = 0$ in the cost function as a function of perturbation. The optimal cost function (J^*) is nondifferentiable with respect to pertubration t at $t = 0$. The predicted slope of the cost function does not match the numerically computed slope as (III.12) only

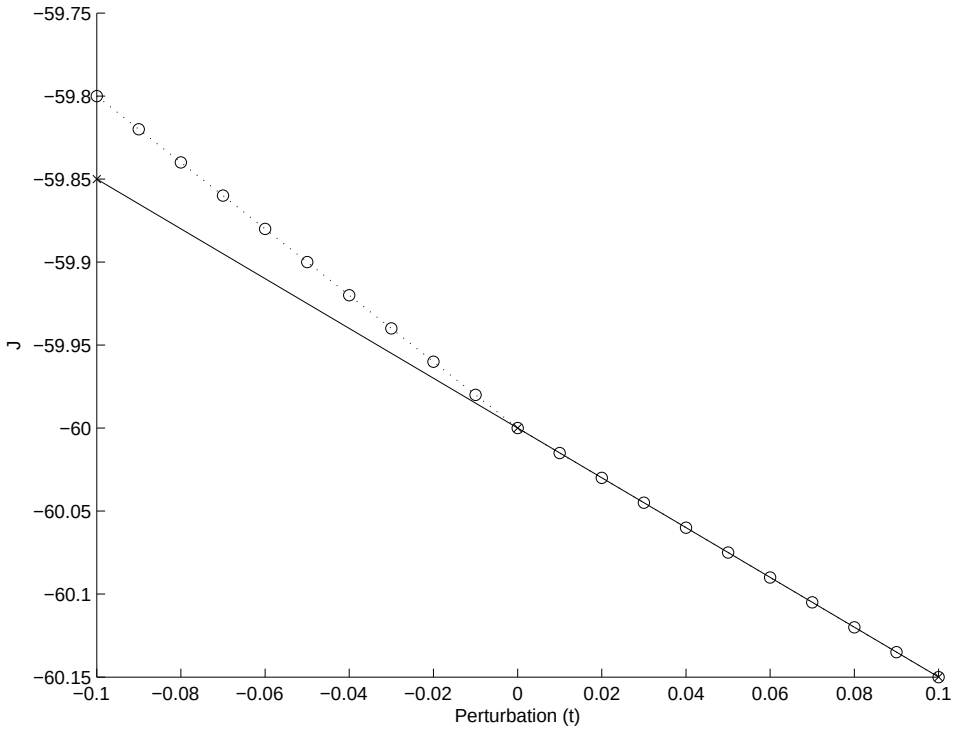

Figure III.7: Analytical (solid) and numerical (circle) sensitivities of optimal cost function of a model LP w.r.t to perturbations in right hand side

holds for non-degenrate point, and $t = 0$ corresponds to a degenerate LP. Thus we can not compute slope of the cost function for the outer problem from sensitivity analysis of the inner LP.

III.D.2 Nonsmoothness of outer cost function

As a further illustration of nonsmoothness of the outer problem, with respect to its variables, we consider a surface plot of the outer cost function as defined above, with respect to two of the outer variables in the inner/outer formulation.

We restate the governing equations by parameterizing all current distributions which satisfy Kirchoff's Current Law by a streamfunction, ψ, defined on all interior cell. The relationship between streamfunction and currents is given by,

$$\begin{aligned} I_1^{i+1/2,j} &= \quad (\psi^{i+1/2,j+1/2} - \psi^{i+1/2,j-1/2}) \\ I_2^{i,j+1/2} &= -(\psi^{i+1/2,j+1/2} - \psi^{i-1/2,j+1/2}), \end{aligned} \tag{III.13}$$

with boundary conditions on ψ given by,

$$\psi^{*,-1/2} = 0, \qquad \psi^{*,M+1/2} = I_{total}$$

We then examine combinations of I_{total} and $\psi^{3/2,1/2}$ for a grid with $N = 3, M = 1$. The various current distributions arising as a result, are imposed as constraints on the inner LP, and the corresponding optimum points are visualized as a surface plot in Figure III.8. If the inner LP is infeasible for a particular combination of (I_{total}, $\psi^{3/2,1/2}$), the cost function for that point is set arbitrarily to 300 for aiding visualization.

From the plot, we can see that outer problem is a nonconvex, nonsmooth function of the currents (expressed as I_{total} and $\psi^{*,*}$), thus gradient based techniques alone can not be used to solve this inner/outer formulation of the problem.

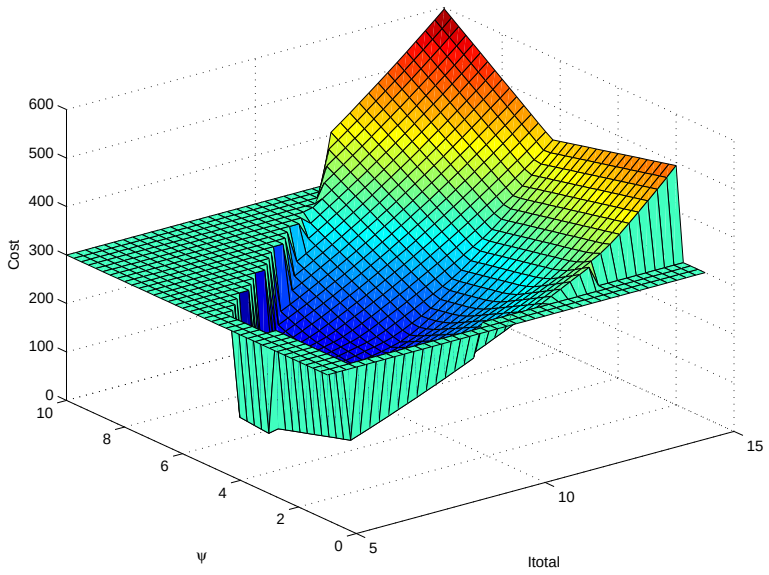

Figure III.8: Nonsmoothness of outer cost function in the inner/outer formulation.

III.E Comparison of left/right LP solver with full nonlinear solver (SNOPT)

We perfomed a comparison of the left/right LP solver with a general nonlinear solver (SNOPT) for a 10×10 heating grid. In both cases, the heating requirement was the same. The imposed heating requirement for both problems is plotted in Figure III.9. There were two points on the grid with non-zero heating requirements as shown in the figure. Both problems had acceptable resistor limits of $\underline{R} = 1$ and $\overline{R} = 10$, giving a restrictive resistor limit ratio of 10. For details on the nonlinear solver used, the reader is referred to (4).

III.E.1 Left/Right LP solver case

For the first case, we used a left/right LP solver as described in section III.C. The solver was run for 50 complete iterations. Each complete iteration comprised of one LeftLP run using an LP solver run for a maximum of 10 iterations, and one RightLP run with a maximum of 10 iterations.

Figure III.10 shows the actual heating produced by the solution of the left/right solver after 50 complete (one LeftLP + one RightLP) iterations. It also shows a superposition of actual and required heating for the 10×10 grid in the left/right solver case.

III.E.2 Nonlinear solver case

For comparison, the same problem (with the same heating requirement as shown in Figure III.9) was posed and solved as a nonlinear program using the SNOPT general nonlinear optimization algorithm for 50 iterations. See (4) for details on SNOPT. The nonlinear program was posed as,

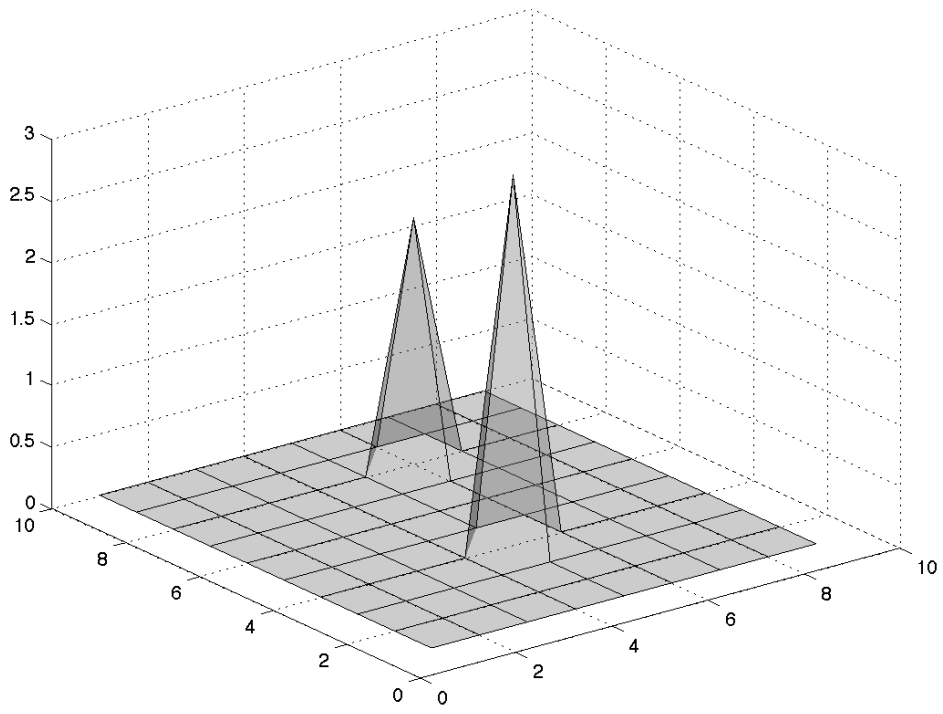

Figure III.9: Heating requirement on a 10×10 grid of resistors in watt

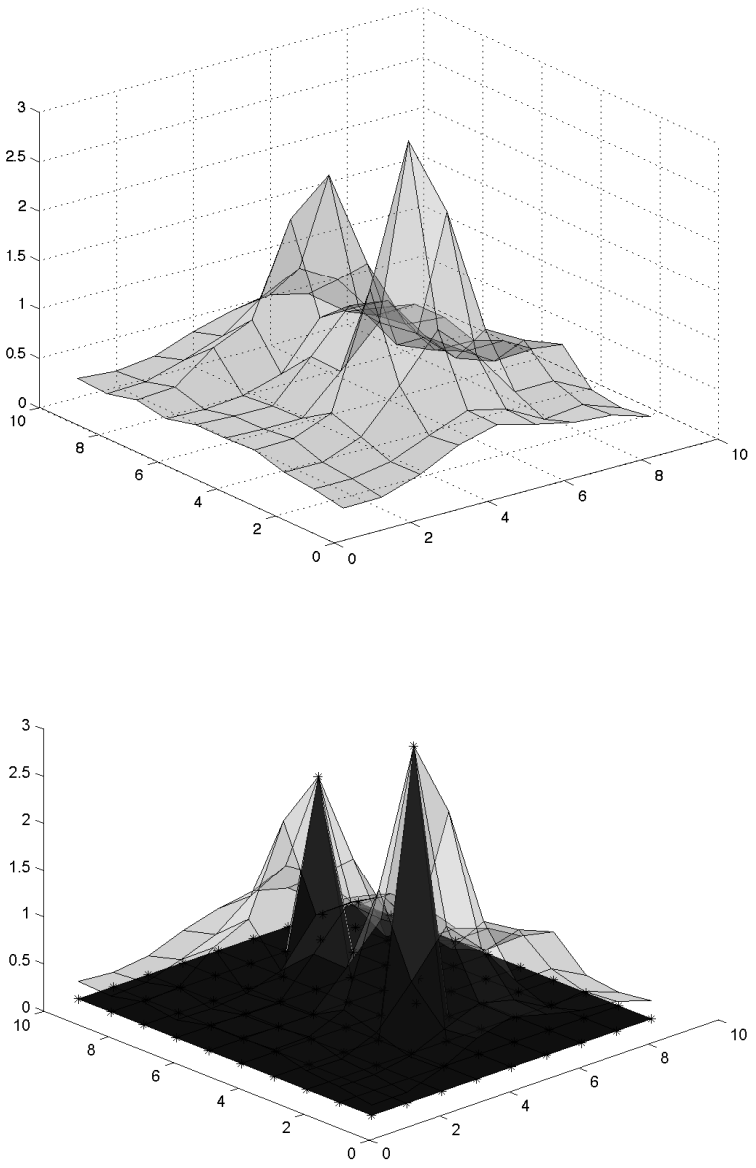

Figure III.10: Actual heating produced by left/right LP solver (top) and superimposition of actual and required heating (bottom) for a 10×10 grid

$$
\begin{aligned}
min \quad & f(\mathbf{x}) \\
s.t. \quad & g_1(\mathbf{x}) = 0, \quad \mathbf{l}_g \leq g_2(\mathbf{x}) \leq \mathbf{u}_g, \\
& \mathbf{l}_A \leq A\,\mathbf{x} \leq \mathbf{u}_A, \quad \mathbf{l}_b \leq \mathbf{x} \leq \mathbf{u}_b
\end{aligned}
$$

where the $\mathbf{x}$ vector is defined as,

$$\mathbf{x} = \begin{pmatrix} V_{in} \\ V^{*,*} \\ R_1^{*,*} \\ R_2^{*,*} \\ I_{total} \\ I_1^{*,*} \\ I_2^{*,*} \end{pmatrix}$$

The nonlinear (quadratic) cost function is simply $V_{in} \cdot I_{total}$ from (III.7), assuming $V_{out} = 0$ wihtout loss of generality. The nonlinear function $g_1(\mathbf{x})$ corresponds to the nonlinear equality constraints in the governing equations, namely Kirchoff's Voltage Law (III.2). This also is a quadratic function. The nonlinear function $g_2(\mathbf{x})$ corresponds to the inequality constraints in the problem, i.e. the heating requirement (III.5) which are again quadratic in nature. Kirchoff's Current Law (III.1) and the equation enforcing total current in the circuit (III.6) are incorporated in the A-matrix with the lower and upper bounds ($\mathbf{l}_A$ and $\mathbf{u}_A$) set to 0 to enforce equality. The lower and upper bounds on the variables themselves ($\mathbf{l}_b$ and $\mathbf{u}_b$) come from the acceptable resistor values (III.4).

Figure III.11 shows the actual heating produced by the solution of the nonlinear solver (SNOPT) after 50 iterations on a 10×10 grid. It also shows a superposition of actual and required heating for the 10×10 grid in the nonlinear solver case.

III.E.3 Comparison of left/right LP solver with nonlinear solver

Figure III.12 shows a comparison of convergence history of the two solvers for 50 iterations. We can see that, the left/right LP solver is competitive with the full nonlinear (SNOPT) solver. The solution obtained by the nonlinear solver is

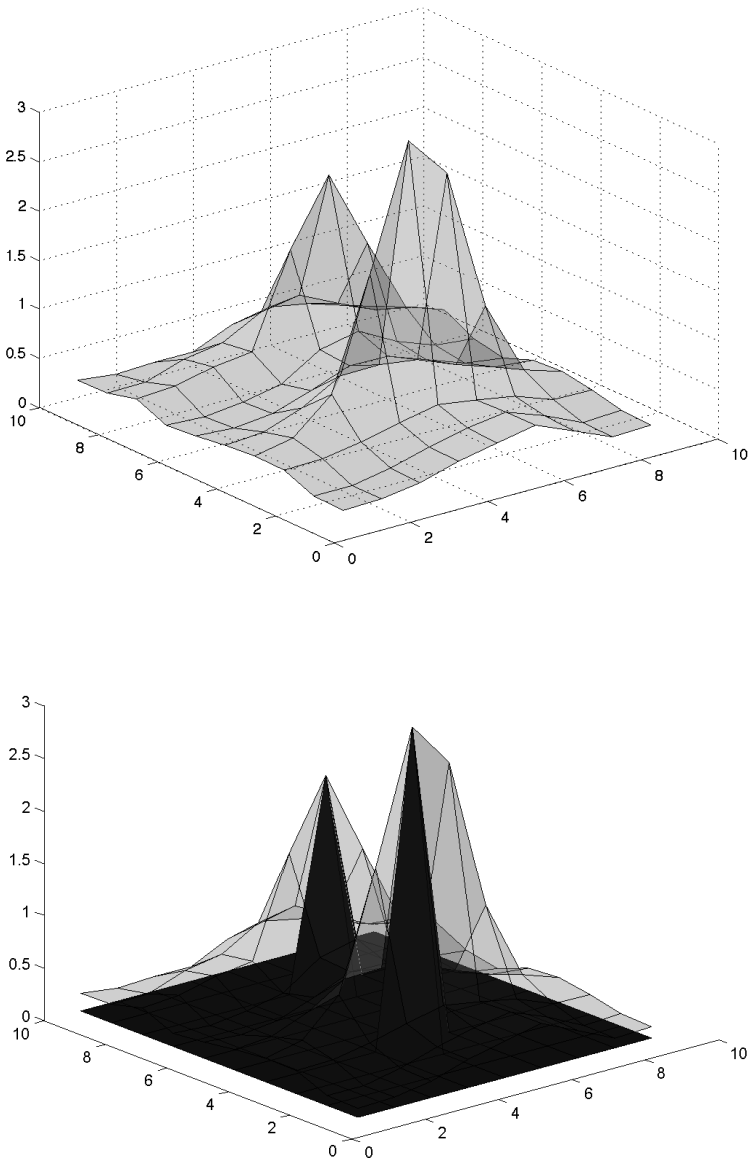

Figure III.11: Actual heating produced by nonlinear solver (top) and superimposition of actual and required heating (bottom) for a 10×10 grid

slightly better than the left/right solver. But it is important to note that for bigger grid sizes, (e.g. 64×64 or 128×128), the left/right LP solver quickly converges to good solutions, while the nonlinear solver is not able to handle such big grids as the length of the solution vector is $\approx O(10^6)$.

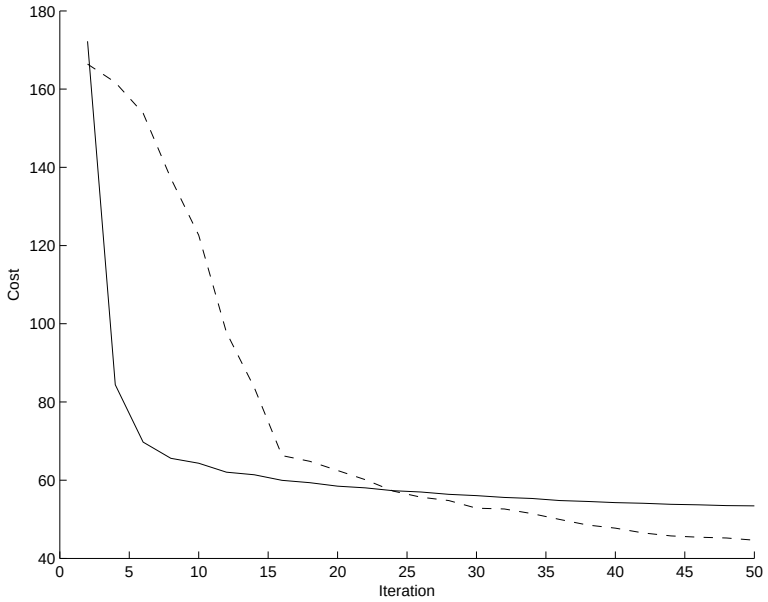

Figure III.12: Comparison of convergence history of left/right LP solver (solid) and nonlinear solver (dashed)

III.F Results and discussion

We ran the left/right LP solver for three grid sizes, 32×32, 64×64 and 128×128. These three cases will be referred to as Case 1, Case 2 and Case 3 respectively. For all three cases, the heating requirement was specified at the start, and the solver was run for 20 iterations. The range of acceptable resistor values for all three cases was $\underline{R} = 0.1$ and $\overline{R} = 10$.

Case 1: In this case, the grid size was chosen to be 32×32 grid points in the X and Y directions. Figure III.13 shows the heating requirement imposed on the solver together with the actual heating produced by the circuit as optimized by the left/right LP solver after 20 iterations. Figure III.14 shows the resistor distribution in the circuit. It is an isocontour plot of average horizontal and vertical resistance values at all nodes of the circuit.

Case 2: In this case, the grid size was chosen to be 64×64 grid points in the

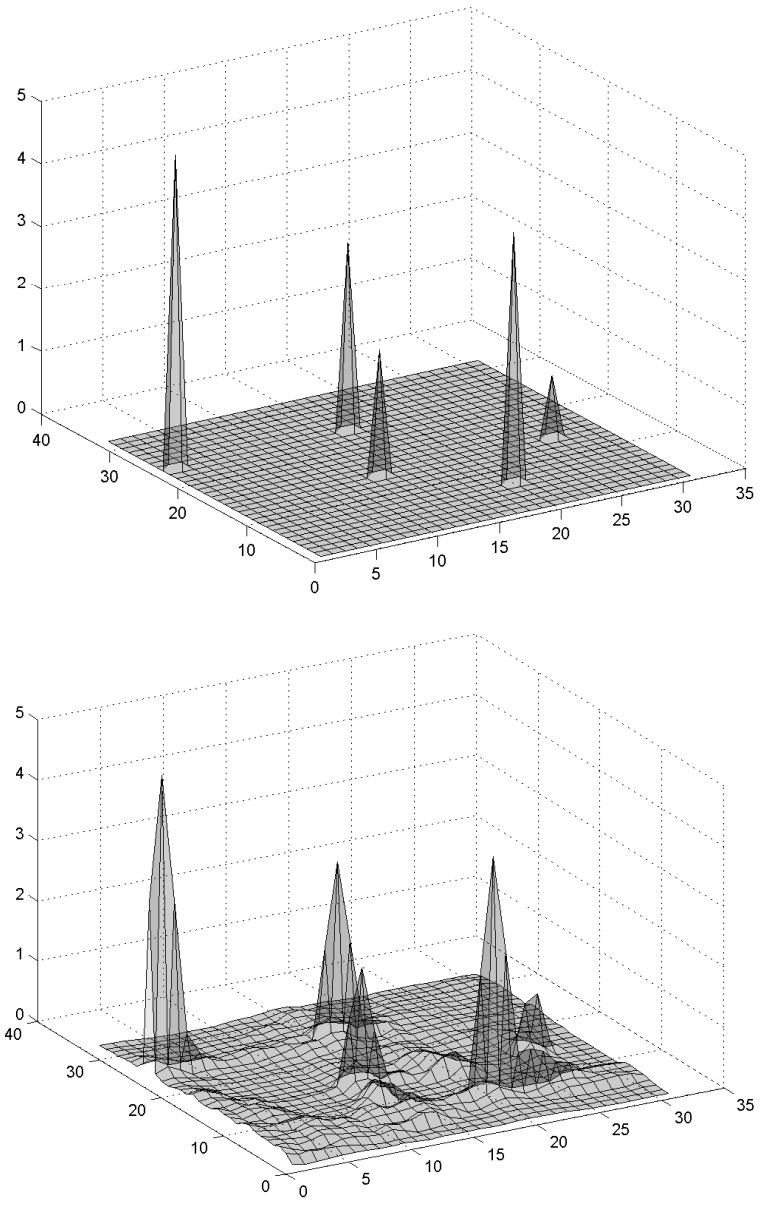

Figure III.13: Case 1: Required heating (top) and actual heating produced (bottom) by left/right LP solver for 32×32 grid.

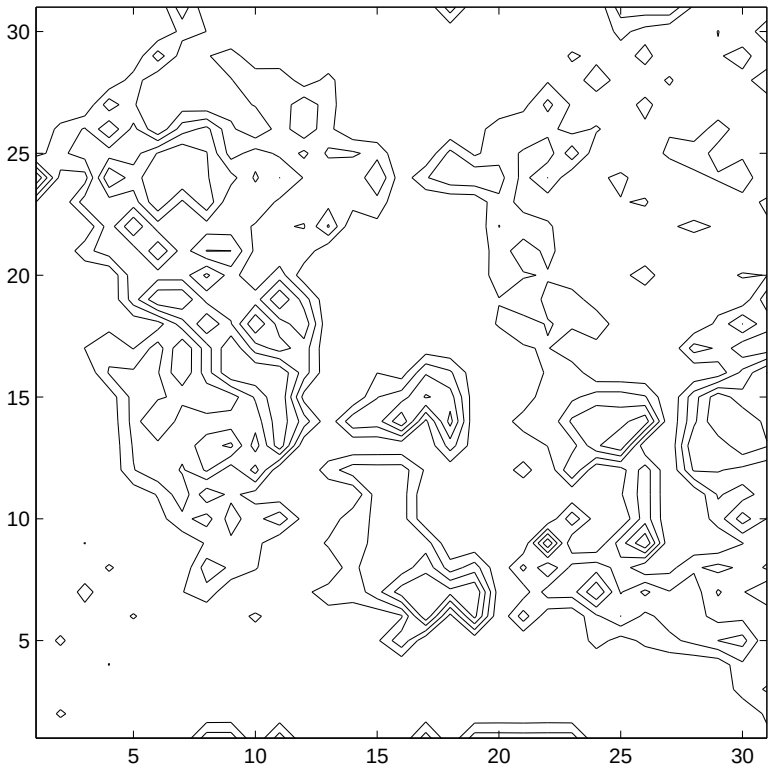

Figure III.14: Isocontours of average of horizontal and vertical resistance values for 32×32 grid. Isocontours for resistance values of 2, 4, 6 and 8 are plotted.

X and Y directions. Figure III.15 shows the heating requirement imposed on the solver together with the actual heating produced by the circuit as optimized by the left/right LP solver after 20 iterations. Figure III.16 shows the resistor distribution in the circuit. It is an isocontour plot of average horizontal and vertical resistance values at all nodes of the circuit.

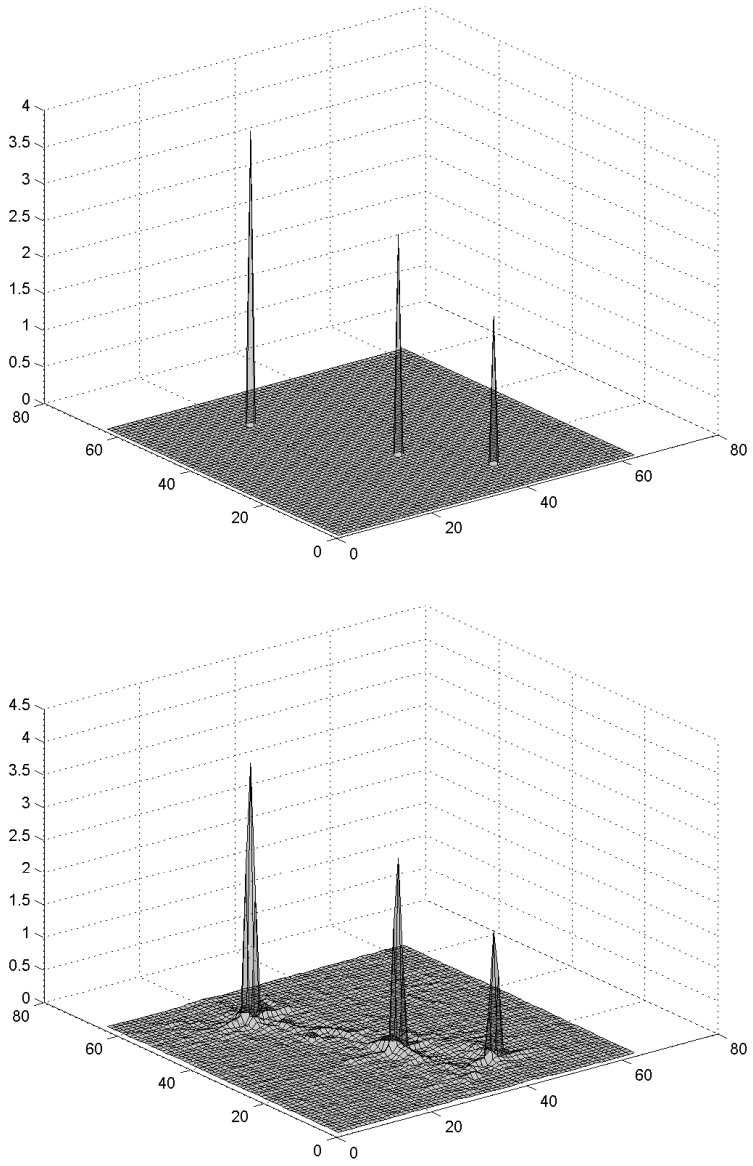

Figure III.15: Case 2: Required heating (top) and actual heating produced (bottom) by left/right LP solver for 64×64 grid

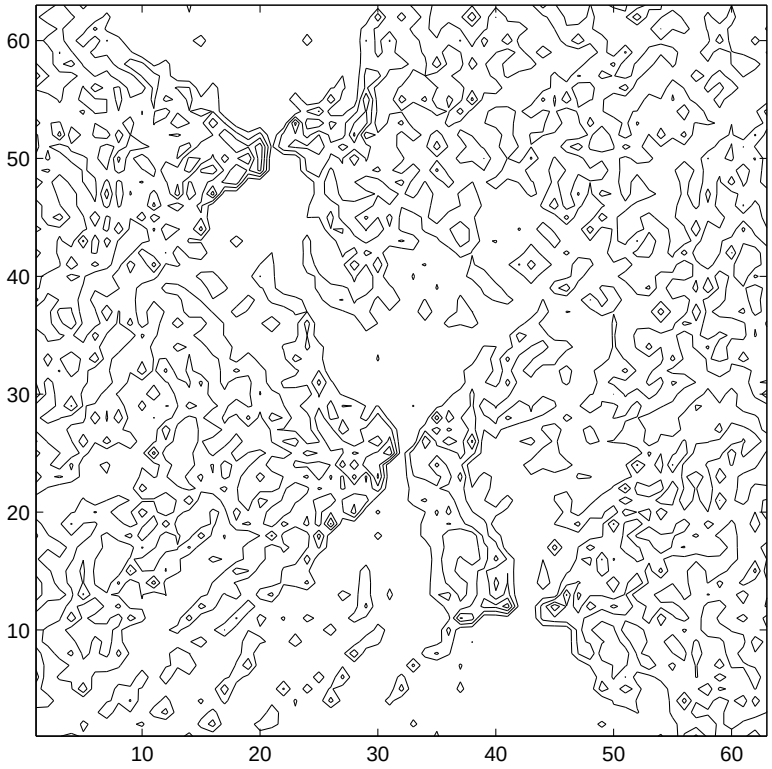

Figure III.16: Isocontours of average of horizontal and vertical resistance values for 64×64 grid. Isocontours for resistance values of 2, 4, 6 and 8 are plotted.

Case 3: In this case, the grid size was chosen to be 128×128 grid points in the X and Y directions. Figure III.17 shows the heating requirement imposed on the solver together with the actual heating produced by the circuit as optimized by the left/right LP solver after 20 iterations. Figure III.18 shows the resistor distribution in the circuit. It is an isocontour plot of average horizontal and vertical resistance values at all nodes of the circuit.

III.G Conclusion

We have illustrated a powerful new method to optimize a highly constrained resistor network problem, which passes efficiently to the PDE limit. This heating resistor network serves as a good model for a low Reynolds number liq-

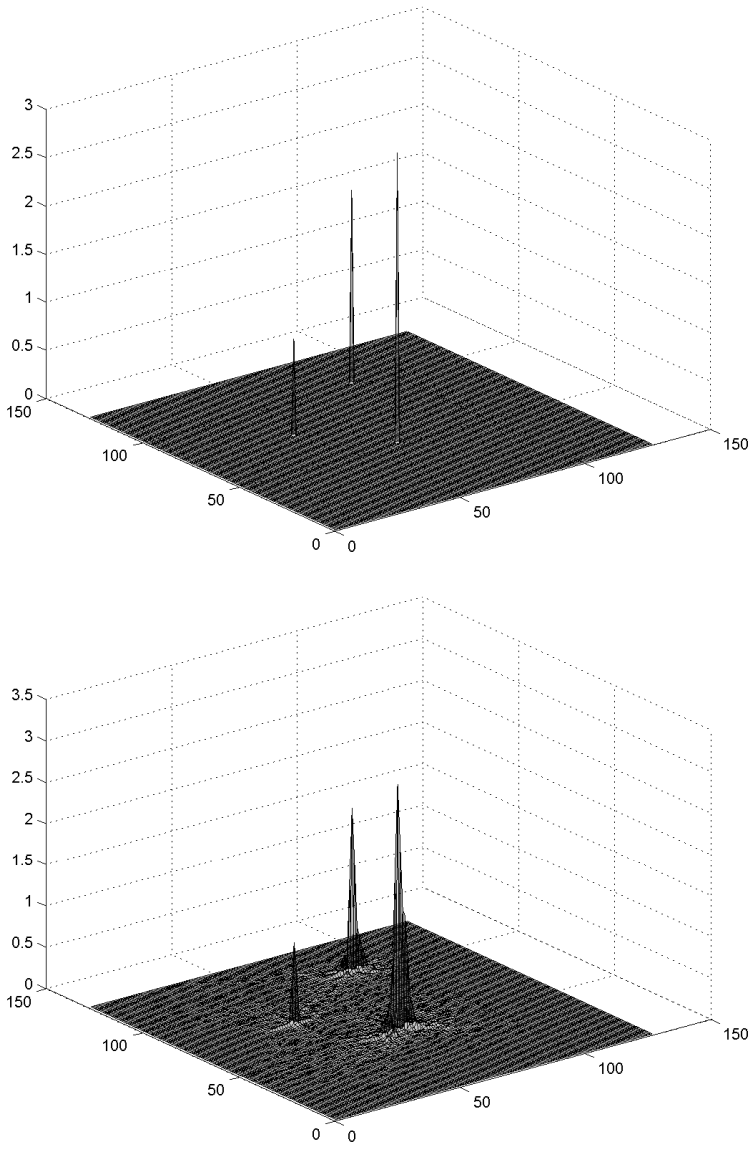

Figure III.17: Case 3: Required heating (top) and actual heating produced (bottom) by left/right LP solver for 128×128 grid

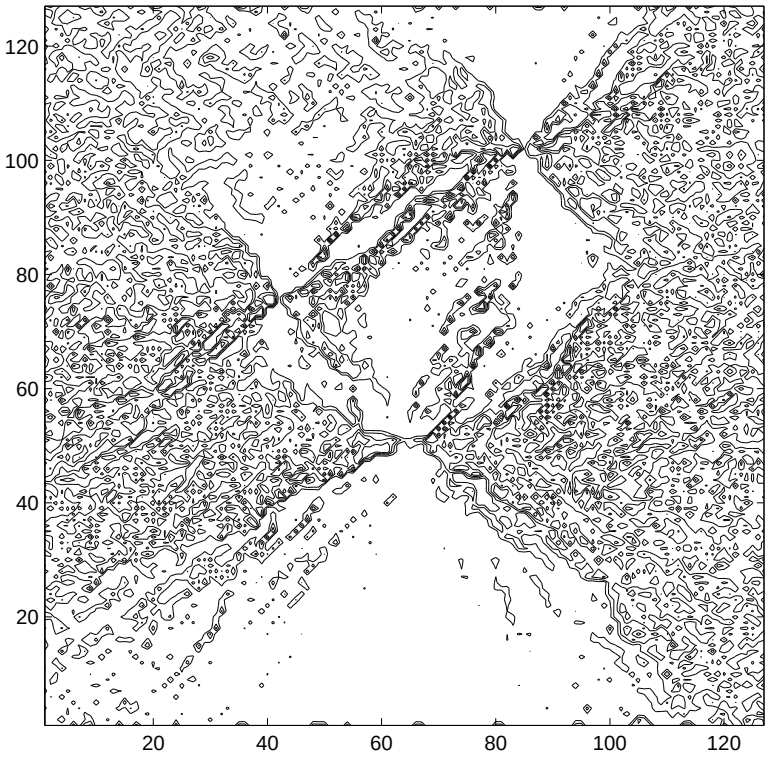

Figure III.18: Isocontours of average of horizontal and vertical resistance values for 128×128 grid. Isocontours for resistance values of 2, 4, 6 and 8 are plotted.

uid cooled heat exchanger with variable cahnnel width or variable porosity. In future work, the left/right LP solver method will be extended to solve the variable channel width/porosity heat exchanger problem by extending the resistor network analogy to low Reynolds number Stokes flow.

One of the issues in the current solution method is the non-smoothness of the resistor distribution in the solution produced by the left/right LP solver. We plan to solve this problem by enforcing smoothness on the allowable solutions for the resistor distribution (See Figures III.14, III.16 and III.18). This will be done by using fewer grid points to specify the resistances in the circuits, and using a smooth interpolant to determine resistor values in between these grid points. Essentially, the resistances in the network, will be defined on a coarse grid, with fewer grid points in each direction (by a factor of 2 or 4), while the rest of the governing equations will be defined on the fine grid. Another issue that needs addressing is the decoupling between horizintal and vertical resistor values. Once these issues are appropraitely handled, this method will be extended to solve the liquid cooled heat exchanger problem.

This chapter, in full, has been submitted to SIAM Journal of Control and Optimization as, Anish Karandikar and T. R. Bewley, *An iterative left/right LP method for the efficient optimization of a highly constrained resistor network.*

Bibliography

[1] T. R. Bewley, Flow Control: New Challenges for a New Renaissance, Progress in Aerospace Sciences 37, 2001, p. 21.

[2] Stephen J. Wright, Primal-Dual Interior-Point Methods, SIAM Publications, 1997.

[3] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, 2004.

[4] P. Gill, W. Murray, M. A. Saunders, SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization, SIAM Review 47 (1), 2005, p. 99.

[5] P. E. Gill, W. Murray, M.H. Wright, Practical Optimization, London, Academic Press, 1981.

[6] J. T. Betts, P. D. Frank, A sparse nonlinear optimization algorithm, J. Optim. Theory Appl. 82, 1994, p. 519.

[7] R. H. Byrd, J. C. Gilbert, J. Nocedal, A Trust Region Method Based on Interior Point Techniques for Nonlinear Programming, Mathematical Programming 89 (1), 2000, p. 149.