Solving Eigenproblems with Application in Collapsible Channel Flows

submitted by

Yujue Hao

for the degree of PhD

of the

University of Glasgow

2013

Signature of Author ....

Yujue Hao

Abstract

Collapsible channel flows have been attracting the interest of many researchers, because of the physiological applications in the cardiovascular system, the respiratory system and urinary system. The linear stability analysis of the collapsible channel flows in the Fluid-Beam Model can be finalized as a large sparse asymmetric generalized eigenvalue problem, where the stiffness matrix is sparse, asymmetric and nonsingular, and the mass matrix is sparse, asymmetric and singular. The dimensions of the both matrices can reach about ten thousand or more, and the traditional QZ Algorithm is so expensive for this size of eigenvalue problem, due to its large requirement of computational resources and the quite long elapsed time. Unlike the traditional direct methods, the projection methods are much more efficient for solving some specified eigenpairs of the large scale eigenvalue problems, because normally a small subspace is made use of, and the original eigenvalue problem is projected to this small subspace. With this projection, the size of the eigenvalue problem is reduced significantly, and then the small dimensional eigenvalue problem can be easily and rapidly worked out by employing a traditional solver. Combined with a restarting strategy, this can be used to solve large dimensional eigenvalue problem much more rapidly and precisely. So far as we know, the Implicitly Restarted Arnoldi iteration(IRA) is considered as one of the most effective asymmetric eigenvalue solvers. In order to improve the efficiency of linear stability analysis in collapsible channel flows, an IRA method is employed to the linear stability analysis of collapsible channel flows in FBM. A Frontal Solver, which is an efficient solver of large sparse linear system, is also used to replace the process of shift-and-invert transformation. After applying these two efficient solvers, the new eigenvalue solver of collapsible channel flows—Arnoldi method with a Frontal Solver(AR-F), not only gets rid of the restriction of memory storage, but also reduces the computational time observably. Some validating and testing work have been done to variety of meshes. The AR-F can solve the eigenvalues with largest real parts very quickly, and can also solve the large scale eigenvalue problems, which cannot be solved by the QZ Algorithm, whose results have been proved to be correct with the unsteady simulations. Compared with the traditional QZ Algorithm, not only a great deal of elapsed time is saved, but also the increasing rate of the operation numbers is dropped to $O(n)$ from $O(n^3)$ of QZ Algorithm. With the powerful AR-F, the stability problems of refined meshes in collapsible channel flows are no long a barrier to the study. So AR-F is used to solve the eigenvalue problems from two refined meshes of the two different boundary conditions(pressure-driven system and flow-driven system), and the two neutral curves obtained are both revised and extended. This is the first time that IRA is made use of in the problem of fluid-structure interaction, and this is also a critical footstone to adopt a three dimensional model over FBM. Recently, the energy analysis and the energetics are the centre of research in collapsible channel flow. Because the linear stability analysis is much more accurate and faster than the unsteady simulation, the energy solutions from eigenpairs are also achieved in this thesis. The energy analysis with eigenpairs has its own advantages: the accuracy, the timing, the division, any mode and any point. In order to analyze the energy from eigenpairs much more clearly, the energy results with different initial solutions are presented first, then the energy solutions with eigenpairs are validated with those presented by Liu et al. in the pressure-driven system. By using the energy analysis with eigenpairs, much more energy results in flow-dirven system are obtained and analyzed.

Keywords : Collapsible channel flow, Linear stability analysis, Large asymmetric generalized eigenvalue problem, Implicitly Restarted Arnoldi iteration, Frontal solver, Energy analysis.

Acknowledgement

I would like to thank my supervisor—Prof Xiaoyu Luo, for her generous offer and grants and continuous supervision, support and encouragement throughout the four years of my study, and also for her patience on proofreading and commenting on this thesis.

Special thanks to everybody in the School of Mathematics and Statistics, University of Glasgow, for their guidance, material contribution, collaboration and helpful support during my study here. Especially for Dr. Steven Roper, who helped a lot with the computational technologies, the operation of Linux system and LaTex.

Finally, I also would like to show my appreciation to Prof Zongxi Cai in Tianjin University in China for his technical support to the project of collapsible channel flows.

Contents

1 Introduction 1

2 Literature overview and preliminaries 3

2.1 Collapsible channel flows . . . 3

2.1.1 Zero-dimensional model . . . 3

2.1.2 One-dimensional model . . . 3

2.1.3 Two-dimensional model . . . 4

2.1.4 Three-dimensional model . . . 5

2.2 Large scale eigenvalue problem . . . 6

2.2.1 Projection method . . . 7

2.3 Conclusion . . . 13

3 Methodology 14

3.1 Model and equation . . . 14

3.1.1 Fluid-Beam Model . . . 15

3.1.2 Dimensionless parameters and governing equations . . . 15

3.1.3 Boundary conditions . . . 16

3.1.4 Steady solution, unsteady simulation and eigenvalue problem . . . 17

3.2 Numerical methods . . . 20

3.2.1 Frontal solver . . . 20

3.2.2 QR/QZ algorithm . . . 26

3.2.3 Arnoldi method . . . 28

3.3 Conclusion . . . 35

4 New Arnoldi Eigensolver 36

4.1 Arnoldi method with global matrices(AR-G) . . . 36

4.2 Arnoldi method with a frontal solver(AR-F) . . . 40

4.3 Validations . . . 44

4.3.1 Flow-driven system . . . 44

4.3.2 Pressure-driven system . . . 46

4.4 Numerical results 47

4.4.1 Computational time 47

4.4.2 Neutral stability curves revised 49

4.4.3 Mode-1 in flow-driven system 51

4.4.4 Neutral stability curves in refined meshes 53

4.5 Conclusion & Further study 66

4.5.1 Conclusion 66

4.5.2 Further study 67

5 Energy analysis with eigenpairs 69

5.1 Approaches to energy solution with eigenpairs 69

5.2 Validation of the energy analysis in pressure-driven system 73

5.3 Energy analysis of flow-driven system 82

5.4 Division of mean flow and oscillatory energy 100

5.4.1 Pressure-driven system 101

5.4.2 Flow-driven system 104

5.5 Conclusion 106

6 Discussion & Conclusion 108

6.1 Achievement 108

6.2 Discussion 108

6.2.1 Eigensolvers 108

6.2.2 Energy analysis 110

6.3 Limitation 112

6.4 Future Work 113

6.5 Conclusion 113

List of Figures

1-1 The Starling Resistor in experiment. The subfigure (a) shows the major part of the collapsible tube, and (b) illustrates the phenomenon of the collapse. 2

3-1 The configuration of collapsible channel flow in fluid-beam model(no scaling). . 15
3-2 The structure of a stiffness matrix $\mathbf{K}$(blue only indicates the positions of nonzero entries, while white means zero entries), and this is not a geometrical symmetry. 19
3-3 The structure of a mass matrix $\mathbf{M}$(blue only indicates the positions of nonzero entries, while white means zero entries), and this is not a geometrical symmetry. 20
3-4 A three triangular element mesh for illustrating the process of a frontal solver . 22
3-5 A three-element, five-node structure for illustrating matrix-vector product . . . 24
3-6 $\mathbf{V}_{k+p}\mathbf{H}_{k+p} + \mathbf{f}_{k+p}\mathbf{e}^T_{k+p}$, which is the right hand side of (3.40). 33
3-7 $\mathbf{V}^+_{k+1}\mathbf{H}^+_{k+1} + \mathbf{f}^+_{k+1}\mathbf{e}^T_{k+1}$, which shows the right hand side of (3.46). 33
3-8 $\mathbf{V}^+_k\mathbf{H}^+_k + (\beta^+_{k+1}\mathbf{v}^+_{k+1} + \sigma_k\mathbf{f}_{k+1})\mathbf{e}^T_k$, which is the right hand side of (3.47), and the next iteration can start from here. 34

4-1 The flowchart of Arnoldi method with Global matrices, where the full expression of the large global matrices is required. 39
4-2 The structure of the matrix $\mathbf{S}(= \mathbf{K}^{-1}\mathbf{M})$, which has lost the original sparsity (blue indicates the positions of nonzero entries, while white means those of zero entries) 40
4-3 The flowchart of Arnoldi method with a Frontal solver, where the full expression of large global matrices is replaced by a frontal solver tactfully. 43
4-4 Comparison of computational time against matrix size using the QZ, AR-G & AR-F solvers. 49
4-5 Mode-2 neutral curve obtained using the AR-F solver for various numbers of degree of freedom $n \times n$, where n=933, 2063, 2629, 3942, 5117 and 7325. The neutral curve obtained by Luo et al. [85] is shown in the solid curve, with $n = 6152$. In all cases, the non-dimensional transmural pressure $P_e - P_d$ is fixed at 1.95. 51
4-6 Unstable Mode-1 at $Re = 480$, $P_e = 1.95$,$\beta = 1.0 \times 10^{10}$, $C_\lambda = 2.45 \times 10^5$ . . . 53

4-7 Neutral curve of the flow-driven system in a refined mesh($50 \times (200+140+100)$ and number of degree of freedom is 55392). The dashed line is the neutral curve presented by Luo et al.(see Figure-3 in [85]) 55

4-8 N2-1: $Re = 700$, $C_\lambda = 1672$, $\omega_r = +3.30023 \times 10^{-5}$, $\omega_i = 2.40366$, Mode-2 . . . 56

4-9 N2-2: $Re = 600$, $C_\lambda = 1797.5$, $\omega_r = -1.14649 \times 10^{-6}$, $\omega_i = 2.42381$, Mode-2 . . 57

4-10 N2-3: $Re = 500$, $C_\lambda = 1800$, $\omega_r = -6.13079 \times 10^{-6}$, $\omega_i = 2.35544$, Mode-2 . . . 57

4-11 N2-4: $Re = 300$, $C_\lambda = 1937$, $\omega_r = 1.76015 \times 10^{-6}$, $\omega_i = 2.03575$, Mode-2 57

4-12 N2-5: $Re = 235$, $C_\lambda = 2410$, $\omega_r = 1.93529 \times 10^{-7}$, $\omega_i = 1.72480$, Mode-2 57

4-13 N2-6: $Re = 220$, $C_\lambda = 2727$, $\omega_r = 1.22905 \times 10^{-6}$, $\omega_i = 1.57726$, Mode-2 58

4-14 N2-7: $Re = 213$, $C_\lambda = 2436$, $\omega_r = 6.90523 \times 10^{-7}$, $\omega_i = 1.43017$, Mode-2 58

4-15 N2-8: $Re = 212$, $C_\lambda = 1600$, $\omega_r = 2.39754 \times 10^{-4}$, $\omega_i = 1.30612$, Mode-2 58

4-16 N2-9: $Re = 240$, $C_\lambda = 558$, $\omega_r = -2.62525 \times 10^{-6}$, $\omega_i = 1.24025$, Mode-2 . . . 58

4-17 N2-10: $Re = 300$, $C_\lambda = 447$, $\omega_r = -3.63519 \times 10^{-5}$, $\omega_i = 1.27266$, Mode-2 . . . 59

4-18 N2-11: $Re = 380$, $C_\lambda = 397.5$, $\omega_r = -3.62141 \times 10^{-6}$, $\omega_i = 1.24308$, Mode-2 . . 59

4-19 N2-12: $Re = 440$, $C_\lambda = 369$, $\omega_r = 9.00410 \times 10^{-6}$, $\omega_i = 1.20026$, Mode-2 59

4-20 N3-1: $Re = 300$, $C_\lambda = 335.61$, $\omega_r = -5.67991 \times 10^{-6}$, $\omega_i = 4.07781$, Mode-3 . . 59

4-21 N3-2: $Re = 250$, $C_\lambda = 311$, $\omega_r = 7.02530 \times 10^{-5}$, $\omega_i = 3.61682$, Mode-3 60

4-22 N4-1: $Re = 231.15$, $C_\lambda = 60$, $\omega_r = 9.69742 \times 10^{-6}$, $\omega_i = 3.85157$, Mode-4 . . . 60

4-23 The mode-1 neutral curve of the pressure-driven system, obtained for $P_e - P_d = 1.95$. The neutral points below N1-2 were initially obtained by Liu et al. using the QZ solver, which are identical to the results of the AR-F solver. The points between N1-1 and N1-2 are obtained using the AR-F solver, which agree well with those obtained by Liu et al. using an unsteady solver, see Table 2. The eight points above N1-1 are the new points obtained using the AR-F solver for a mesh denser mesh. 61

4-24 N1-1: $Re = 226.62$, $Pud = 1.21$, $C_\lambda = 2\times10^7$, $\omega_r = 7.08475\times10^{-6}$, $\omega_i = 3.61910$, Mode-1 62

4-25 N1-2: $Re = 122.16$, $Pud = 0.70815$, $C_\lambda = 5500$, $\omega_r = 4.59145 \times 10^{-6}$, $\omega_i = 1.01443$, Mode-1 62

4-26 N1-3: $Re = 113.83$, $Pud = 0.6872$, $C_\lambda = 2000$, $\omega_r = -3.31709 \times 10^{-6}$, $\omega_i = 0.84649$, Mode-1 62

4-27 N1-4: $Re = 107.89$, $Pud = 0.7031$, $C_\lambda = 927$, $\omega_r = -1.60704 \times 10^{-5}$, $\omega_i = 0.72597$, Mode-1 63

4-28 N1-5: $Re = 115.61$, $Pud = 1.0$, $C_\lambda = 308$, $\omega_r = -9.00310 \times 10^{-6}$, $\omega_i = 0.51888$, Mode-1 63

4-29 $Re = 237.02$, $Pud = 1.263$, $C_\lambda = 3 \times 10^7$, $\omega_r = -2.17656 \times 10^{-5}$, $\omega_i = 3.96708$, Mode-1 63

4-30 $Re = 241.64$, $Pud = 1.2895$, $C_\lambda = 3.5 \times 10^7$, $\omega_r = 3.5902 \times 10^{-6}$, $\omega_i = 4.14419$, Mode-1 64

4-31 $Re = 253.02$, $Pud = 1.3152$, $C_\lambda = 4 \times 10^7$, $\omega_r = -2.73466 \times 10^{-6}$, $\omega_i = 4.32351$, Mode-1 64

4-32 $Re = 264.91$, $Pud = 1.4127$, $C_\lambda = 6 \times 10^7$, $\omega_r = -5.71267 \times 10^{-6}$, $\omega_i = 5.03695$, Mode-1 64

4-33 $Re = 281.09$, $Pud = 1.4989$, $C_\lambda = 8 \times 10^7$, $\omega_r = 6.63375 \times 10^{-6}$, $\omega_i = 5.71320$, Mode-1 65

4-34 $Re = 301.79$, $Pud = 1.6083$, $C_\lambda = 1.1 \times 10^8$, $\omega_r = -2.08839 \times 10^{-6}$, $\omega_i = 6.63706$, Mode-1 65

4-35 $Re = 322.68$, $Pud = 1.72775$, $C_\lambda = 1.5 \times 10^8$, $\omega_r = 4.88966 \times 10^{-7}$, $\omega_i = 7.72416$, Mode-1 65

4-36 $Re = 347.15$, $Pud = 1.849$, $C_\lambda = 2 \times 10^8$, $\omega_r = 4.53263 \times 10^{-6}$, $\omega_i = 8.91252$, Mode-1 66

5-1 $P1 - D$ plot of different perturbations to a neutral point ($Pud = 1.31$, $C_\lambda = 3.0 \times 10^7$) 74

5-2 The enlargement of zone A in Figure 5-1 of $P1 - D$ plot 75

5-3 N1-1: $Pud = 1.21$, $C_\lambda = 2.0 \times 10^7$, the blue solid curve is from unsteady simulation [80], the red solid curve is from energy solution from eigenpairs, and the dashed curve is the normalized energy solution from eigenpairs. 76

5-4 N1-2: $Pud = 0.70815$, $C_\lambda = 5500$, the blue solid curve is from unsteady simulation [80], the red solid curve is from energy solution from eigenpairs, and the dashed curve is the normalized energy solution from eigenpairs. 77

5-5 N1-3: $Pud = 0.6872$, $C_\lambda = 2000$, the blue solid curve is from unsteady simulation [80], the red solid curve is from energy solution from eigenpairs, and the dashed curve is the normalized energy solution from eigenpairs. 78

5-6 N1-4: $Pud = 0.7031$, $C_\lambda = 927$, the blue solid curve is from unsteady simulation [80], the red solid curve is from energy solution from eigenpairs, and the dashed curve is the normalized energy solution from eigenpairs. 79

5-7 N1-5: $Pud = 1.0$, $C_\lambda = 308.4$, the blue solid curve is from unsteady simulation [80], the red solid curve is from energy solution from eigenpairs, and the dashed curve is the normalized energy solution from eigenpairs. 80

5-8 N2-1: $Re = 700$, $C_\lambda = 1672$, $\omega_r = +3.30023 \times 10^{-5}$, $\omega_i = 2.40366$, Mode-2 . . . 84

5-9 N2-2: $Re = 600$, $C_\lambda = 1797.5$, $\omega_r = -1.14649 \times 10^{-6}$, $\omega_i = 2.42381$, Mode-2 . . 85

5-10 N2-3: $Re = 500$, $C_\lambda = 1800$, $\omega_r = -6.13079 \times 10^{-6}$, $\omega_i = 2.35544$, Mode-2 . . . 86

5-11 N2-4: $Re = 300$, $C_\lambda = 1937$, $\omega_r = 1.76015 \times 10^{-6}$, $\omega_i = 2.03575$, Mode-2 87

5-12 N2-5: $Re = 235$, $C_\lambda = 2410$, $\omega_r = 1.93529 \times 10^{-7}$, $\omega_i = 1.72480$, Mode-2 88

5-13 N2-6: $Re = 220$, $C_\lambda = 2727$, $\omega_r = 1.22905 \times 10^{-6}$, $\omega_i = 1.57726$, Mode-2 89

5-14 N2-7: $Re = 213$, $C_\lambda = 2436$, $\omega_r = 6.90523 \times 10^{-7}$, $\omega_i = 1.43017$, Mode-2 90

5-15 N2-8: $Re = 212$, $C_\lambda = 1600$, $\omega_r = 2.39754 \times 10^{-4}$, $\omega_i = 1.30612$, Mode-2 91

5-16 N2-9: $Re = 240$, $C_\lambda = 558$, $\omega_r = -2.62525 \times 10^{-6}$, $\omega_i = 1.24025$, Mode-2 . . . 92

5-17 N2-10: $Re = 300$, $C_\lambda = 447$, $\omega_r = 3.63519 \times 10^{-5}$, $\omega_i = 1.27266$, Mode-2 93

5-18 N2-11: $Re = 380$, $C_\lambda = 397.5$, $\omega_r = -3.62141 \times 10^{-6}$, $\omega_i = 1.24308$, Mode-2 . . 94

5-19 N2-12: $Re = 440$, $C_\lambda = 369$, $\omega_r = 9.00410 \times 10^{-6}$, $\omega_i = 1.20026$, Mode-2 95

5-20 N3-1: $Re = 300$, $C_\lambda = 335.61$, $\omega_r = -5.67991 \times 10^{-6}$, $\omega_i = 4.07781$, Mode-3 . . 96

5-21 N3-2: $Re = 250$, $C_\lambda = 311$, $\omega_r = 7.02530 \times 10^{-5}$, $\omega_i = 3.61682$, Mode-3 97

5-22 N4-1: $Re = 231.15$, $C_\lambda = 60$, $\omega_r = 9.69742 \times 10^{-6}$, $\omega_i = 3.85157$, Mode-4 . . . 98

List of Tables

3.1 Common boundary condition 16

3.2 Specified boundary condition 17

4.1 Neutral points of $(8 \times (10 + 20 + 10))$ mesh 45

4.2 Neutral points of $(16 \times (30 + 60 + 60))$ mesh 46

4.3 The Mode-1 validating points of the pressure-driven system, obtained for $P_e - P_d = 1.95$. 47

4.4 Elapsed time(in seconds) of different meshes at a same group of control parameters 48

4.5 Unstable eigenmodes in flow-driven system 52

4.6 Neutral points of flow-driven system$(50 \times (200+140+100)$ and number of freedom is 55392) 54

4.7 The mode-1 neutral curve of the pressure-driven system, obtained for $P_e - P_d = 1.95$. The neutral points below N1-2 were initially obtained by Liu et al. [80] using the QZ solver, which are identical to the results of the AR-F solver. The points between N1-1 and N1-2 are obtained using the AR-F solver, which agree well with those obtained by Liu et al. using an unsteady simulation, see Table 2. The eight points above N1-1 are the new points obtained using the AR-F solver for a denser mesh. 61

5.1 Definition of the energy analysis 71

5.2 The lists of perturbations to a neutral point($Pud = 1.31$, $C_\lambda = 3.5 \times 10^7$) . . . 73

5.3 The average energy quantities at the operating points in pressure-driven system 82

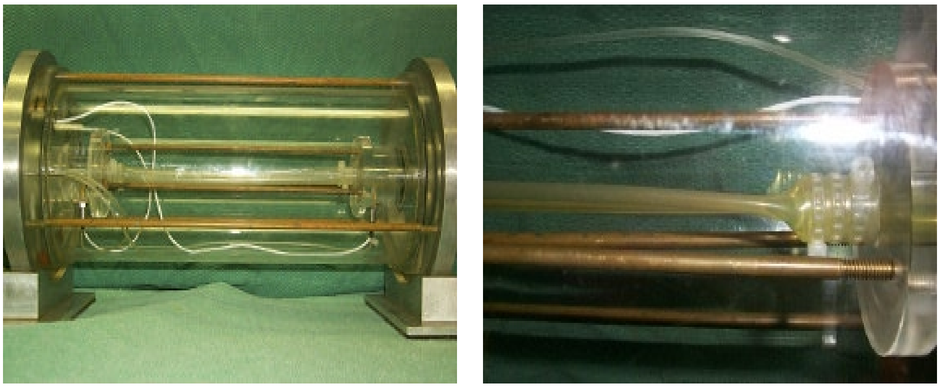
5.4 Operating points from the neutral curve in flow-driven system $(50 \times (200 + 140 + 100)$, and number of degree of freedom is 55392) 83

5.5 The average energy quantities at the operating points in flow-driven system . . 100

Chapter 1

Introduction

A collapsible tube is a tube with sufficiently flexible walls that it can elastically accommodate deformation to a highly noncircular cross-section, when external pressure exceeds internal pressure. The root of “collapsible tube problem” and “collapsible channel problem” come from the compressed elastic tube conveying fluid flow. If the tube tends to be fairly flexible, due to the difference between the external pressure and the internal pressure distribution throughout the fluid fields, a non-axisymmetrical collapse happens. In particular, a very small change of the difference between the external pressure and internal pressure would cause a large buckling displacement of the elastic tube or elastic beam. When there is flow through the tube, the nonlinearity of this relation between lumen area and transmural pressure allows the flow to affect markedly the shape of the tube thereby intimately coupling the fluid and solid mechanics. Consequently, these fluid-structure interactions lead to a great number of interesting phenomena, for example, flowrate limitation, pressure limitation and self-excited oscillation. Because of the physical resemblance between the research of the collapsible tube and physiological flow traveling in conduits of human body, such as the blood vessels, airways and so on, the study of flows in collapsible tube is therefore of considerable interest in many biomedical or physiological applications.



(a) Starling Resistor

(b) Collapsible Tube

Figure 1-1: The Starling Resistor in experiment. The subfigure (a) shows the major part of the collapsible tube, and (b) illustrates the phenomenon of the collapse.

A canonical problem in physiological fluid mechanics concerns the origin of self-sustained oscillations in the Starling Resistor [69], which was originally used by the physiologists to understand the phenomenon of flow limitation in the returning to the heart of human being. A flow driven through a segment of pre-pressed collapsible tube was involved in the experiment. The high compliance of the tube, coupled to an internal flow, make the system susceptible to a range of instabilities, which widely appear in cardiovascular system, respiratory system and urinary system. In the cardiovascular system, Aoki et al. used their model to predict diseased arteries [1]; Binns et al studied the ischemic attack with the effect from stenosis on wall motion [14]; Ku simulated the blood flow in arteries [70] and investigated the unsteady flow phenomena in it. The cervical venous hum and the Korotkoff sound can also be generated by the collapse of blood vessels. In the respiratory system, flutter instabilities have been held responsible for the origin of respirator wheezes during force expiration [29, 37, 30]; Berke et al. [8] modeled a collapsible tube system for the production of speech; a nonlinear dynamical system was investigated by Fee et al. [22] for demonstating the primary mechanism behind the dynamic flow-structre interactions; Balint [6] presented that “flutter” or “static divergence” would cause the instabilities of the palate. In the urinary system, the urethra behaves as a collapsible tube and is affected by flow-limitation [35, 36].

Chapter 2

Literature overview and preliminaries

2.1 Collapsible channel flows

Inspired by the dynamical phenomena exhibited in the experiment of Starling Resistor, in order to disclose the mechanism of the collapsible channel flows, a great many of different models were created by many researchers for their investigations.

2.1.1 Zero-dimensional model

The zero-dimensional models were adopted by Conrad [19], Schoendorfer et al. [131], Pedley [112], and Bertram et al. [10]. Especially, Bertram et al. [9] found that the energy loss in the downstream at the point of strongest collapse played an important role in developing self-induced oscillation. However, the zero-dimensional model has its failure to incorporate many real mechanical feature.

2.1.2 One-dimensional model

Shapiro [133] created a one-dimensional model via employed a long-wavelength approximation to the Navier-Stokes equations. However, the experiments by Bertram et al. [13] and the computations by Luo et al. [89] cast doubt on the causal link between self-sustained oscillation and the flow limitation. Afterwards, that model is not reliable for describing the flow in the experiment of Starling Resistor. Kececioglu et al. [68, 67] improved the law of tube to make it better, and suggested that the flow in that tube is as same as the flow between two parallel membranes, which inspired the two-dimensional model to come into being. Bertram et al. [9]

showed that the steady flow could be always found, but no self-induced oscillation was reported. And Cancelli et al. [16] improved the inviscid momentum equation at the downstream of the narrowest point to suggest a reasonable but still simple model of energy loss. After that, Jensen et al. [54] investigated the existence of steady solution via applying the phase-plane techniques to the model of Cancelli et al. [16], and they presented the existence of multiple steady states. The distinct modes of oscillation found by Jensen [50, 13] in their own range of frequencies were reproduced by Bertram et al [12, 11]. Some other researchers [94, 51, 93, 38, 46] also did some work in this field, all of which showed the complicated self-excited oscillations.

Stewart et al. [141] performed their studies on a one-dimensional model. By means of solving an eigenvalue problem, they revealed the Static Divergence(non-oscillatory instability) and the Travelling-Wave Flutter(oscillatory instability), and disclosed that the static modes arise through transcritical bifurcations, and oscillatory modes arise through Hopf bifurcations. Afterwards, Stewart et al. conducted an asymptotic approximation to the oscillatory mode-1 in the case of $T >> 1$, and introduced the energy budget analysis of the instabilities.

2.1.3 Two-dimensional model

Fluid-Membrane Model(FMM)

According to the suggestion of McClurken et al. [68, 67], Pedley [113] initiated the two-dimensional collapsible channel model, which is well known as "Fluid-Membrane Model(FMM)". In this model, a part of the upper wall in a rigid channel is replaced by a membrane, with longitudinal tension T, and neither the stiffness nor inertia of the membrane was considered. A steady Poiseuille flow with an invariant flowrate is assumed at the entrance of the channel. The pressure $P_{ed} = P_e - P_d$ is kept as a constant, where P_e and P_d are the external pressure and the pressure at downstream, respectively. Pedley pointed out that the necessary condition that the steady solution exists is that T remains positive at every node on the membrane, and suggested that when the flowrate reaches a critical value, the whole configuration should exhibit a bulge-out at the upstream. Lowe et al. [81] segregated the equations of fluid and structure, where the equations were discretized independently, and then after the computation, the solutions were coupled together. However, they failed to obtain a converged steady solution at a low value of the membrane tension. Following Pedley's theory, Luo et al. [86] made the fluid and the structure coupled. By means of numerical spines [123], Rast [116] adopted a fully-coupled approach in his computations, and the discretized Finite Element Equations were solved simultaneously via Newton-Raphson method and a frontal solver. Rast disclosed the steady streamlines when the flow goes through the channel with a large collapse. After investigating the steady flow in 2-D collapsible channel in 1995, Luo et al. [87] deliberated on the instability of the steady solution, by studying the time-dependent problem. Luo et al. [87] performed the unsteady simulation by adding a perturbation to the steady solution, and showed the instantaneous streamline of the fluid under a large amplitude self-excited oscillation. Luo et al. [88] emphasized the effects of the wall inertia again in 1998, and they indicated the three shortcomings(1, the direction of the wall movement has to be assumed; 2, axial stiffness is ignored and the longitudinal stretch of the elastic wall is only banlanced by a uniform tension and the transmural pressure; 3, bending stiffness is neglected as well.), due to the several *ad hoc* assumptions in 2001 [83]. All of these are overcome by the Fluid-Beam Model(FBM).

Another two-dimensional Fluid-Membrane Model was constructed by Jensen et al. [53]. With a uniform external pressure, Jensen & Heil simulated high-frequency self-excited oscillations in a collapsible-channel flow [52].

Fluid-Beam Model(FBM)

The Fluid-beam Model was firstly presented by Cai et al. [15] in 2003, where some *ad hoc* assumptions are abolished, and both extensional stiffness and bending stiffness are considered(full details of FBM are introduced in Chapter 3). Due to the numerical perturbations of the unsteady simulation are not strictly infinitesimal, the small amplitude self-induced oscillations cannot be considered to be coming from the linear instability of the system. In order to investigate this, they involved the discretised Orr-Sommerfeld eigenvalues system in the FBM in 2004 [84], where they identified the “Tongue Shape” neutral stability curve at the first time, which was revised with a commercial finite element software—ADINA by Luo et al. [82]. Luo et al. continued the research of the instability of the FBM, and presented the much more complicated “CASCADE” structure [85] of $Re - C_\lambda$ space in flow-driven system. Later in 2012, Liu et al. [80] carried out the stability of pressure-driven system in FBM, and introduced the energy distribution analysis to the FBM. During the research work of the stability in the collapsible channel flows, the eigenvalue problem of linear stability is much more accurate and efficient than the unsteady simulation. However, the traditional QZ algorithm which is one of the current eigensolvers in collapsible channel flow cannot play a significant role while the number of degree of freedom increases. In order to investigate the stability problems of FBM better, it is instructive to find a much more efficient eigensolver, which is also a necessary work before proposing the three-dimensional model.

2.1.4 Three-dimensional model

With the development of science and technology, the study of three-dimensional model comes to be much easier than before. After examining the post-buckling large deformation of cylindrical shells conveying viscous flow with Pedley [43] in 1996, where the tube wall was modeled with the theory of nonlinear Kirchhoff-Love shell, Heil [40] extended his research to the stability of cylindrical shell and determined the most unstable buckling modes. Owing to the successful agreement between the computation and experiment found by Heil [41]. By using 3D steady stokes equations to describe the flow, Hazel and Heil [39] created a 3D model with employing 3D steady Navier-Stockes equations and applying the thin shell theory to the structure. Marzo et al. [90] presented the 3D simulations for a steady flow in thick-walled tubes, via employing a FEM software FIDAP. They presented that the thickness of the wall affects the steady behaviour of the wall, and when the wall was thin, their results agreed with Hazel and Heil [39] very well. Heil et al. [44] presented a research on a 3D model of unsteady flows in a high-frequency oscillated collapsible tubes. Later in 2010, Whittaker et al. disclosed the energetics of flow through a rapidly oscillating tube in both general theory [147] and the application in an elliptical tube [145]. Heil et al. [42] introduced a numerical simulations of self-exited oscillations(flow-driven system) in three-dimensional collapsible tubes. Whittaker et al. accounted for the energetics of flow with slowly varying amplitude [146].

Cai and Luo are structuring their 3D model basing on their FBM, and what they used to be worried about is the eigensolver which has been researched in this thesis.

2.2 Large scale eigenvalue problem

Large matrix problems include solving linear equations, locating eigenpairs and so on. Theory, method and software package of the numerical methods for these problems are very important issue in the computing mathematics and technology of engineering computing. And these problems are the basic and significant parts of the engineering computing, especially the middle or large asymmetric matrix problems. Because of the complexity in the theory, people must face a great deal of trouble in the study of the theory and the design of the software. However, a challenge to these hard problems attracts many experts all over the world to study this field, and they have made great progress.

Eigenvalue problems occur frequently in many engineering calculations, such as computational fluid mechanics [65], statistics [115], structural engineering [117, 7], quantum physics [132], chemical engineering [114], economical modelling [106], aerospace industry [66], meteorologic modelling [20], electronic supply [3, 97], signal processing and controlling [91, 23] and some other fields. Given the large matrices involved, such problems are usually solved using numerical methods. Research on the numerical methods of large asymmetric eigenvalue problems has been a central focus over the last 50 years.

There are three kinds of eigenvalue problems: standard ($\mathbf{Ax} = \lambda\mathbf{x}$), generalized eigenvalue problem($\mathbf{Ax} = \lambda\mathbf{Bx}$) and polynomial eigenvalue problems(($\mathbf{A}_0 + \lambda\mathbf{A}_1 + \ldots + \lambda^n\mathbf{A}_n)x = \mathbf{0}$) . The $\mathbf{A}$ and $\mathbf{B}$ are $n \times n$ real(complex) matrices. Although the dimensions of the eigenvalue problems are several thousands, or above ten thousand even several million, the number of the eigenvalues needed are usually very small, just several leading eigenvalues which have the largest(smallest) real(imaginary) part, largest(smallest) modulus or several eigenvalues in some special regions and the corresponding eigenvectors.

2.2.1 Projection method

Since the QR Algorithm [33, 17], QZ algorithm [28], power method [127], divide-and-conque method and some other effective methods came, the small and medium dimensional eigenvalue problems are solved precisely by these direct methods, no matter the matrices are symmetric or not. But for the large scale eigenvalue problems, because of the memory storage, processor speed and the influence from some other factors, those effective methods are useless. Since 1950s, the projection method was taken into this field. The projection method is classified into the orthogonal projection method and the inclined projection method.

The orthogonal projection method and the inclined projection method can be easily summarized to find their subspaces, respectively. The subspaces $\mathcal{K}$ and $\mathcal{L}$ whose dimensions are not very large, which are normally called as the right subspace and the left subspace. The inclined projection method is to find $(\tilde{\lambda}_i, \tilde{\mathbf{x}}_i)$,($\|\tilde{\mathbf{x}}_i\| = 1$), and make them to satisfy the equation (2.1).

$$\begin{cases} \tilde{\mathbf{x}}_i \in \mathcal{K} \\ \mathbf{A}\tilde{\mathbf{x}}_i - \tilde{\lambda}_i\tilde{\mathbf{x}}_i \perp \mathcal{L} \end{cases} \tag{2.1}$$

Then $(\tilde{\lambda}_i, \tilde{\mathbf{x}}_i)$ is taken to be the approximate eigenpair $(\lambda_i, \mathbf{x}_i)$ of the matrix $\mathbf{A}$, and it is called as "Ritz Pair", where $\tilde{\lambda}_i$ is the Ritz value and $\tilde{\mathbf{x}}_i$ is the Ritz vector. When $\mathcal{K} = \mathcal{L}$, the equation (2.1) is changed to

$$\begin{cases} \tilde{\mathbf{x}}_i \in \mathcal{K} \\ \mathbf{A}\tilde{\mathbf{x}}_i - \tilde{\lambda}_i\tilde{\mathbf{x}}_i \perp \mathcal{K} \end{cases} \tag{2.2}$$

This is turned to be the orthogonal projection method. Actually, the key impact of the projection method is to reduce the dimensions of the original matrices as small as possible, then by employing a direct method, the Ritz pairs are worked out from the small dimensional eigenvalue problems to approach the solutions of the original eigenvalue problems.

The projection method depends on the subspace very much, so their names always come with the subspaces used. Amongst these, there are several famous methods: the Arnoldi Method, the Block Arnoldi Method, the Lanczos Method, the Block Lanczos Method, the Two-sided Lanczos Method, the ABLE Method, the Subspace Iteration Method, the Davidson Method, the Jacobi-Davision Method and so on. Some of them will be introduced briefly in the following paragraphs.

Arnoldi method and Block Arnoldi method

Both the Arnoldi Method and the Block Arnoldi Method are orthogonal projection methods. When choosing $\mathcal{K} = \mathcal{K}_m(\mathbf{A}, \mathbf{v}_1) = span\{\mathbf{v}_1, \mathbf{A}\mathbf{v}_1, \ldots, \mathbf{A}^{m-1}\mathbf{v}_1\}$ and $\mathbf{V}_m = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m)$ as a group of standard orthogonal basis which comes from the Arnoldi process, the orthogonal projection method is Arnoldi method [2, 127]. The process is to reduce and transform the original matrix to an upper Hessenberg matrix, and can be written in a matrix equation as

$$\mathbf{A}\mathbf{V}_m = \mathbf{V}_m\mathbf{H}_m + \mathbf{h}_{m+1,m}\mathbf{v}_{m+1}\mathbf{e}_m^T = V_{m+1}\tilde{\mathbf{H}}_m, \tag{2.3}$$

where both $\mathbf{H}_m = (\mathbf{h}_{i,j})_{m\times m}$ and $\tilde{\mathbf{H}}_m = (\mathbf{h}_{i,j})_{(m+1)\times m}$ are upper Hessenberg matrices. $\mathbf{H}_m$ is called the projection matrix or the limit matrix of $\mathbf{A}$ under the standard orthogonal basis in the subspace $\mathcal{K}_m(\mathbf{A}, \mathbf{v}_1)$.

The shift and inverse transformation, which is one of the spectral transformations, can be employed in the Arnoldi process, and with this, the Arnoldi method can be used to solve the generalized eigenvalue problem($\mathbf{A}\mathbf{x} = \lambda\mathbf{B}\mathbf{x}$). Sometimes it is called the Shift and Inverse Arnoldi Method, and this method takes shift and inverse operator $(\mathbf{A} - \alpha\mathbf{B})^{-1}\mathbf{B}$ into Arnoldi process. There is another kind of transformation whose name is "Caylay transform"—$(\mathbf{A} - \alpha\mathbf{B})^{-1}(\mathbf{A} - \gamma\mathbf{B})$. But the inversion of the matrix is the drwaback of these two methods. Later then, Meerbergen and Roose [96] took inaccurate Caylay transformation into the Arnoldi process. And Ruhe brought out a generalized shift and inverse transformation $(\alpha_j\mathbf{A} - \beta_j\mathbf{B})^{-1}(\gamma_j\mathbf{A} - \delta_j\mathbf{B})$, which is the most famous Rational Krylov Stratery(RKS) [118, 122, 120, 119, 121]. In that stratery, the parameter $\alpha_j, \beta_j, \gamma_j$ and δ_j can be changed with the iteration process. The last step of this method is to solve a small dimension generalized eigenvalue problem $(\varsigma\mathbf{K}_{m,m} - \eta\mathbf{L}_{m,m})\mathbf{s} = \mathbf{0}$, where $\mathbf{K}_{m,m}$ and $\mathbf{L}_{m,m}$ are $m \times m$ upper Hessenberg matrix, and these problems can be easily solved by the QZ algorithm.

While the dimension of the subspace raising, the operation numbers and memory storage of the Arnoldi process will also be increased. Hence, in the practical computation a restarted technology is employed to reduce the operation numbers. But some useful information(the component of wanted eigenvectors) would be ignored, so whether the restarted technology is good or not, is one of the most significant key factors in the projection method. The explicit restarted technology [101, 150, 140, 126] and Implicit restarted technology [138, 78] are the two main techniques, which are considered to be much better than the others. Amongst these, the Implicit Restarted technology [138] created by Soresen is considered to be the most effective one in the world. And the Thick Restarting technology [101, 150, 140] takes the information obtained from the previous iteration into new solving subspace, so the new subspace is called "augmented" or "compressed" Krylov subspace [128, 103]. In essence, thick restarting is still an explicit restarting method, however, it has some mathematical equivalence [101, 150, 140] with the implicit restarting method. Because there are some unsteady factors [78] in the implicit restarting method, but thick restarting is not affected by them [150], so the thick restarting is not complicated as the implicit restarting method, and it is easy to create new computing method and new software package with it, so it attracts lots of researcher to study.

When choosing $\mathcal{K} = span\{\mathbf{V}_1, \mathbf{AV}_1, \ldots, \mathbf{A}^{m-1}\mathbf{V}_1\}$, where $\mathbf{V}_1 = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_p)$ is $n \times p$ full rank matrix, the orthogonal projection method is the block Arnoldi method. As the subspace method is a generalization of the power method, the block Arnoldi method is a generalization of the Arnoldi method. It has some advantages [5]: 1.suitable for the dense eigenvalues; 2.can obtain the number of the same eigenvalues and the dimension of the subspace; 3.good for parallel computing.

Lanczos method and ABLE method

Double orthogonal Lanczos method is an inclined projection method. After choosing $\mathcal{K} = \mathcal{K}_m(\mathbf{A}, \mathbf{v}_1)$, $\mathcal{L} = \mathcal{K}_m(\mathbf{A}^T, \mathbf{w}_1)$, and their basis $\mathbf{V}_m$ and $\mathbf{W}_m$, when they are a couple of double orthogonal basis($\mathbf{W}_m^T\mathbf{V}_m = \mathbf{I}$), the inclined projection process is same as the double orthogonal Lanczos method, or the double side Lanczos method [127, 73]. This method transforms $\mathbf{A}$ to a tridiagonal matrix with two groups of double orthogonal vectors. After that the eigenvalues of the tridiagonal matrix are computed, and considered as the approximate eigenvalues of the original matrix. When $\mathbf{A}$ is an asymmetric matrix, we will usually face a complicated problem during the double orthogonal Lanczos process, which is called "serious breakdown" [148]. And the reason of the serious breakdown is not from the round-off error and the ill-conditioned matrix. The most effective and famous method to avoid and deal with the serious breakdown is the technique of "look-ahead" [111, 27], with which the Lanczos method is upgraded by choosing the 2×2 pivot element.

Another critical problem in double orthogonal Lanczos process is the loss of double orthogonalization. In order to deal with it, we often use the selective orthogonaiation [110], the partial reorthogonalization [134] or the full orthogonaliation [150]. Another strategy is to "live with" the loss of double orthogonalization [21]. In the case where no serious breakdown takes place, the convergence analysis done by Ye [151] shows that the eigenvalues with largest(smallest) real part are usually calculated from the tridiagonal matrix. The conclusion of Bai [4] declares that under the finite precision, if the condition number of the Ritz value is rational and there is no serious breakdown, the convergence of the Ritz value means the loss of double orthogonalization, while the loss of double orthogonalization also means the convergence of the Ritz value.

When $\mathbf{A}$ is a symmetric matrix, the double orthogonal method becomes the famous Lanczos method [72], which is a orthogonal projection method. Because under the finite precision the orthogonalization of the basis obtained from the Lanczos process will be lost rapidly, this method has not been used for a long time. Until 1971, in the PhD thesis of Paige [107], he indicated that the loss of double orthogonaliztion is not very bad. With the loss of the double orthogonaliztion, more than one Ritz value would converge to the eigenvalues of $\mathbf{A}$; on the contrary it is also right. Now, the Lanczos method is one of the most important techniques to solve several specified eigenvalues and the corresponding eigenvectors of a symmetric matrix.

When choosing $\mathcal{K} = \mathcal{K}_{mp}(\mathbf{A}, \mathbf{Q}_1)$, $\mathcal{L} = \mathcal{K}_{mp}(\mathbf{A}^T, \mathbf{P}_1)$, and their basis $\mathbf{Q}_m$, $\mathbf{P}_m$ as a group of double orthogonal basis ($\mathbf{P}_m^T \mathbf{Q}_m^T = \mathbf{I}$), where $\mathbf{Q}_1$ and $\mathbf{P}_1$ are $n \times p$ full order block matrices, the inclined projection method is the ABLE method [5]. This method is a generalization of the double orthogonal Lanczos method [32] to the block matrix, or a generalization from symmetric matrix to asymmetric matrix. The feature of the ABLE method is that it can change the size of the block dynamically after scanning the numbers of the multiple eigenvalues, and it makes the size of the block to be equal to the number of the multiple eigenvalues or the dense eigenvalues. Hence, it can be used to solve the eigenvalue problems with multiple eigenvalues and the dense eigenvalues. And changing the dynamically changing size of the block can deal with the serious breakdown and nearly breakdown very well.

Subspace Iteration Method

Choosing $\mathcal{K} = span(\mathbf{A}^l \mathbf{X}_0)$, where $\mathbf{X}_0$ is $n \times m$ full order matrix, when l is a positive integer, the orthogonal projection method is the subspace iteration method [18, 49, 142]. Some testing vectors are used in the iteration in this method, and these vectors will converge to the eigenvectors of the original eigenvalue problem. Hence, this method is considered to be a generalization of multiplying power method [17, 127]. The drawback of the subspace iteration method is the velocity of convergence, which might be a little slow [126, 109], so the technique of Chebyshev accelerating [126, 124, 125] is usually combined with it. Another drawback is that it can only calculate the eigenvalues which have largest modulus and the corresponding eigenvectors or Schur vectors. When the eigenvalues with largest(smallest) real(imaginary) parts and the corresponding eigenvecotors are required, the subspace iteration method cannot be used, unless these eigenvalues also have the largest modulus.

Davidson Method and Jacobi-Davidson Method

These two methods belong to the orthogonal projection method. In 1975 the Davidson method [92] was brought forward to calculate several eigenvalues, where the original matrices are from computing quantum chemistry, and these matrices are strongly diagonal. Its main strategy is to use preconditioning technique and Rayleigh-Ritz process to calculate the subspace dynamically. Morgan and Scott [104] analyzed this method and indicated that the key to success is the preconditioning technique, and they generalized the preconditioning strategy to the symmetric Davidson method. In 1993 Sadkane [130] generalized this method to the block form again.

The Jacobi-Davidson method [136] is a combination technique of the Jacobi Method [48] and the Davidsion method. This method expends the space by solving the large preconditioning asymmetric linear system which is deduced by the previous work, and it forces the subspace to approach the required eigenvectors. Because it depends on the solution of the large asymmetric linear equation group, the convergence and stability of this method is so complicated, and it is investigated by the researchers all over the world. Sleijgen et al. have used the Jacobi-Davidson method into the generalized eigenvalue problem [135] and the quadratic eigenvalue problem [137].

Harmonic Projection Method

In the inclined projection method, if the basis of the left space $\mathcal{L}$ and the right space $\mathcal{K}$ is $\mathbf{W}_m$ and $\mathbf{V}_m$, respectively, which satisfy $\mathbf{W}_m = \mathbf{A}\mathbf{V}_m$, the inclined projection method is the harmonic projection method. When choosing $\mathcal{K} = \mathcal{K}_m(\mathbf{A}, \mathbf{v}_1)$ and its basis is a group of standard orthogonal basis, the harmonic projection method is the harmonic Arnoldi method [99, 100, 105, 102, 108]. During the previous years, the harmonic projection method has been one of the most usual and effective method to calculate the inner eigenpairs(eigenvalue and eigenvector). The Ritz value was studied by Freund [26] and Morgan [99], the “harmonic” was taken into the symmetric matrices by Paige [108], and later was taken into the asymmetric matrix by Sleijpen et al. [136].

One of the harmonic projection method’s advantages is that the original dense eigenvalue problem can be changed by choosing a shift point α correctly, and it makes the distribution of the eigenvalues to be located easily. As we know, the standard projection method is not suitable for solving the inner eigenvalues. Apparently, we can transform the inner eigenvalue problem to an outer one by the shift and inverse transformation, but in this case the inverse matrix $(\mathbf{A} - \alpha\mathbf{I})^{-1}$ is required, which would take much more time and require large memory storage. Hence, it is not a good idea when the dimension of the matrix is large or several inversions are needed. Another advantage of harmonic projection method is that the inner eigenvalue problem can be transformed to an outer eigenvalue problem, and the explicit inverse matrix is not required. Consequently, the drawback is that the convergence speed of harmonic projection method is much slower than the shift and inverse Arnoldi method.

Refined Projection Method

Jia and Stewart [63] have proved that, when the the error ε between the expected eigenvector and the subspace approaches zero, the eigenvalue $\tilde{\lambda} \to \lambda$, but we cannot ensure the eigenvectors $\tilde{\mathbf{x}} \to \mathbf{x}$. So in the classical orthogonal projection method, even if the Ritz values converge to the eigenvalues, the Ritz vectors might not converge to the eigenvectors [63, 56]. Due to this viewing, Jia put forth the refined projection method [55], and the refined vectors can be obtained from the small singular value decomposition or by solving a small eigenvalue problem easily. Theoretical analysis indicates that, when $\varepsilon \to 0$, the refined Ritz vectors will converge to the eigenvectors unconditionally [63]. And all the classical projection methods have the corresponding refined methods, such as the refined Arnoldi method [57], the refined harmonic Arnoldi method [61], the refined subspace iteration method [60], the refined double side Lanczos method [149] and so on. And also, Jia has put forward the “refined shift” [59] and the “refined harmonic shift” [58] by researching the implicit restarting technique and the refined Arnoldi method or the refined harmonic method. A great amount numerical experiments have been implemented to prove that the refined method is much better than the classical method [57, 64, 60, 149, 59].

Actually, the refined projection method can be considered as a combination of the two orthogonal projection methods [58], so there are so many relations between the refined projection method and the classical projection method. Recently, the refined projection method which is indicated by Jia has attracted increasing people to study, and is considered as one of the three most significant methods(another two methods are the orthogonal projection method and the harmonic projection method). Later, Jia et al. have taken the refined shift into solving large singular value problems [62].

Eigensolvers in fluid mechanics

Edwards et al. [139] employed Krylov methods for time evolution, steady-state solving and linear stability analysis for the incompressible Navier-Stokes Equations. Similar ideas have been explored by Meerbergen et al. [95, 96], and Lehoucq et al. [77], where Arnoldi method combined iterative solutions of the matrix equations were used for the stability studies of incompressible steady flow. Lehoucq et al. [77] employed the Implicitly Restarted Arnoldi methods with the matrix transformation to solve the eigenvalues of the discretized Navier-Stokes equations. Lehoucq et al. concluded that with careful implementation, the Implicitly Restarted Arnoldi methods are reliable for linear stability analysis. Lehoucq et al. [74] shows that a Cayley transformation leads to a more efficient and robust eigensolver than the usual shift-and-invert transformation when the linear systems are solved inexactly within the Rational Krylov Sequence method. Later Lehoucq et al. also applied the Arnoldi iteration driven by a novel implementation of the Cayley transformation to the stability analysis of three dimensional steady flows on parallel computers [75]. All the aforementioned stability analysis are applied for flow problems, whereas in this paper, we apply the approach to a fluid-structure interaction problem.

2.3 Conclusion

After reviewing the theories and models of flow in a collapsible channel(tube), some popular and effective eigensolvers and the application of some eigensolvers in fluid mechanics or linear stability problem, in order to improve the eigensolver of the linear stability analysis in collapsible channel flows, we decided to solve the generalized eigenvalue problem in collapsible channel flow of Fluid-Beam model with an Arnoldi method. Furthermore, in the next chapter, the full details of FBM, including the configuration, the dimensionless parameters, the governing equations and the boundary conditions. In addition, the Arnodi iteration will be introduced much more clearly as well as a frontal solver, and some simple but necessary examples will be illustrated to make everything understood easily.

Chapter 3

Methodology

3.1 Model and equation

The Fluid-Beam Model was firstly presented by ZX Cai and XY Luo [15] in 2003. Luo et al. [85] indicated the cascade structure of the linear stabilities in the flow-driven collapsible channel flow system. Liu et al. [80] stated the neutral curve of pressure-driven system. During their studies, they met several challenges, which are the refined level of the meshes, memory storage, and computational time as well. In particular, the mesh of pressure-driven system has to be much more refined than that in flow-driven system, because the wall stiffness C_λ is large, the displacement and velocity of the beam oscillation are fairly small. And some neutral points presented by Liu et al. [80] were not neutral enough, due to the traditional QZ algorithm cannot work in the refined mesh, and the unsteady simulation would take quite a long time to locate a neutral point. Therefore, in order to improve the efficiency of research, it is important and necessary to employ a new eigensolver for the project of collapsed channel flows.

3.1.1 Fluid-Beam Model

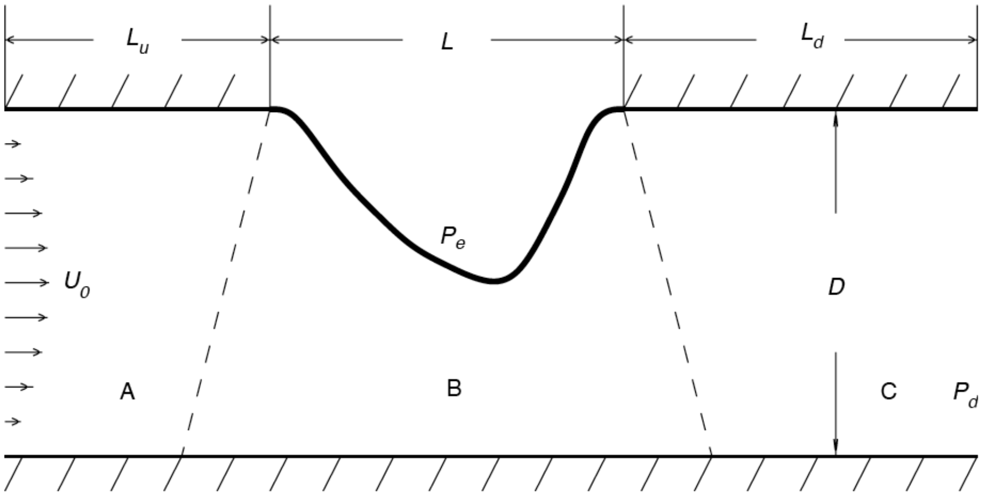


Figure 3-1: The configuration of collapsible channel flow in fluid-beam model(no scaling).

The configuration of two-dimensional collapsible channel flow is shown in Figure 3-1. The width and length of the rigid wall are D and $L_u + L + L_d$, respectively. The subscripts u and d here indicate "upstream" and "downstream", respectively. A part of the upper wall is replaced by an pre-stressed elastic beam with its thickness h, which is subjected to the ends of rigid wall. Pe is the external pressure applied on the beam. A steady flow with average velocity U_0 is assumed at the entrance of the flow-driven system, while a constant pressure gradient is set along the channel in pressure-driven system. The fluid is incompressible and laminar and has the density ρ and viscosity μ. The beam might have large deflection, but still obeys the Hooke's law. The extensional and bending stiffness of the beam are EA and EJ, where E is the Young's modulus, A is the area of the beam cross-section and J is the beam moment of inertia. Both damping and rotational inertia of the beam are neglected, the pre-tension in the beam is T, and beam density is ρ_m.

3.1.2 Dimensionless parameters and governing equations

For convenience, all the non-dimensional variables are marked with stars, and shown in (3.1).

$$\left.\begin{array}{lllll} u_i^* = \dfrac{u_i}{U_0}(i=1,2), & \sigma_{ij}^* = \dfrac{\sigma_{ij}}{\rho U_0^2}, & p^* = \dfrac{p}{\rho U_0^2}, & T^* = \dfrac{T}{\rho U_0^2 D}, & \\ x^* = \dfrac{x}{D}, & y^* = \dfrac{y}{D}, & t^* = \dfrac{tU_0}{D}, & l^* = \dfrac{l}{D}, & L^* = \dfrac{L}{D}, \\ \rho_m^* = \dfrac{h\rho_m}{D\rho}, & c_\lambda^* = \dfrac{EA}{\rho U_0^2 D}, & c_\kappa^* = \dfrac{EJ}{\rho U_0^2 D^3}, & Re^* = \dfrac{\rho U_0 D}{\mu}, & \kappa^* = \kappa D, \end{array}\right\} \tag{3.1}$$

where u is the fluid velocity components, σ_{ij} is the fluid stresses, p is the pressure, t is time, x, y is the Cartesian coordinates, T is the initial tension, κ is the curvature, C_λ and C_κ are the extensional and bending stiffness of the beam, and Re is the Reynolds number. For simplicity, the non-dimensioned sign asterisk (*) are dropped from here. Then we obtain the eight dimensionless governing equations

$$\frac{\partial u_i}{\partial t} + u_j u_{i,j} = -p_{,i} + \frac{1}{Re} u_{i,jj}, (i,j = 1,2) \tag{3.2}$$

$$u_{i,i} = 0, (i,j = 1,2) \tag{3.3}$$

$$\frac{\rho_m}{\lambda}\left(x' \frac{d^2x}{dt^2} + y' \frac{d^2y}{dt^2}\right) = C_\kappa \kappa \kappa' + C_\lambda \lambda' + \lambda \tau_n \tag{3.4}$$

$$\begin{aligned}\frac{\rho_m}{\lambda}\left(y' \frac{d^2x}{dt^2} - x' \frac{d^2y}{dt^2}\right) &= C_\kappa \left(\frac{1}{\lambda}\kappa'\right)' - \lambda\kappa T - \lambda\sigma_n \\ &\quad - C_\lambda \lambda\kappa(\lambda - 1) + \lambda p_e\end{aligned} \tag{3.5}$$

$$x' = \lambda \cos\theta \tag{3.6}$$

$$y' = \lambda \sin\theta \tag{3.7}$$

$$\lambda\kappa = \theta'. \tag{3.8}$$

The principal stretch of the beam is defined as $\lambda = \sqrt{x'^2 + y'^2}$, and the prime denotes the differentiation with respect to l, where l comes from a typical point by $(l, 0, 0)$ in the undeformed configuration.

3.1.3 Boundary conditions

There are two kinds of boundary conditions throughout the whole study, one is common boundary condition, and the other is specified boundary condition. The common boundary condition can be expressed mathematically as:

Table 3.1: Common boundary condition

rigid wall :	$u = v = 0,$	at $y = 0, 0 \leqslant x \leqslant L_0$ at $y = 1, 0 \leqslant x \leqslant L_u$ and $L_u + L \leqslant x \leqslant L_0$
elastic part :	$u(t) = u_w(t),$ $v(t) = v_w(t),$	at $x = x_w$ and $y = y_w$
beam ends :	$\theta = 0,$	at $x_w = L_u$, $y_w = 1$ and $x_w = L_u + L$, $y_w = 1$
outlet flow :	$\sigma_n = \sigma_t = 0,$	at $x = L_0$ and $0 \leqslant y \leqslant 1$

where x_w, y_w, u_w, v_w are the coordinates and velocities of a general point on the beam, σ_n is the normal fluid stress at the channel outlet, and σ_t is the tangential fluid stress.

The specified boundary condition is the way, by which the current system is driven.

Table 3.2: Specified boundary condition

flowrate driven system	inlet flow :	$u = 6y(y-1)$, $v = 0$	at $x = 0$, $0 \leqslant y \leqslant 1$
pressure driven system	inlet flow :	$Pud = constant$,	at $x = 0$, $0 \leqslant y \leqslant 1$

When the specified boundary condition is applied, the mechanism of the systems are totally different. In the flowrate-driven system, a parabolic Poiseuille flow is assumed at the inlet; while in the pressure-driven system, the pressure gradient along the channel is a constant.

3.1.4 Steady solution, unsteady simulation and eigenvalue problem

With the boundary conditions, by employing the weighted residuals method and Petrov-Galerkin method, the eight governing equations can be discretized to a Finite Element Method Equation as:

$$\mathbf{M}\frac{d\mathbf{U}}{dt} + \mathbf{K}(\mathbf{U})\mathbf{U} - \mathbf{F} = \mathbf{R} = \mathbf{0}, \tag{3.9}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{U}$ is the global vector of unknowns$(u_j, p_j, v_j, x_j, y_j, \theta_j, \lambda_j, \kappa_j)^T$, $\mathbf{K}(\mathbf{U})$ is the stiffness matrix determined by a steady solution $\bar{\mathbf{U}}$, $\mathbf{F}$ is the external force vector and $\mathbf{R}$ is the overall residual vector, can be written as $\mathbf{R} = (R_x, R_y, R_c, R_{ex}, R_{ey}, R_{e\theta}, R_{e\lambda}, R_{e\kappa})^T$. R_x, R_y and R_c indicate the residuals of x- and y-momentum and continuity equations of fluid, which are (3.2) and (3.3), and R_{ex}, R_{ey},$R_{e\theta}$,$R_{e\lambda}$ and $R_{e\kappa}$ represent the residuals of the other governing equations.

1. Steady solution: With setting the time derivative $d\mathbf{U}/dt = \mathbf{0}$, a steady solution $\bar{\mathbf{U}}$ can be worked out from the **FEM**(Finite Element Method) equations for a given set of parameters(Re, P_e and so on), which has been done by Cai and Luo [15].

2. Unsteady simulation: After achieving the steady solution of a group of parameters, in order to test the stability of the solution, an inifitesimal perturbation is appied to the steady solution $\bar{\mathbf{U}}$ and the time evolution is calculated. In the unsteady simulation, a slightly different steady solution has to be used as the initial guess, which has a parameter different from the corresponding one of steady solution, such as the value of C_λ. The difference between the steady solution and initial solution should be very small, otherwise, the unsteady solution might not be reliable for that point. Later in Chapter 5, the energy solutions causing by the size of the perturbation to steady solution will be illustrated and discussed. If the unsteady simulation converges to the corresponding steady solution, the

solution is considered to be stable, and the point is denoted as stable point; if not, the unsteady simulation might diverge away from the steady solution or oscillate around the steady solution with time, and now this solution is deemed to be unstable or neutral, and the point is named as unstable point or neutral stability point.

3. Eigenvalue problem: As the numerical perturbations to the steady solution are not strictly infinitesimal, it is not evident that the small-amplitude oscillations definitely correspond to linear instability of the system. To investigate the instability, we solve the eigenvalue problem of the linearized finite element equations, which is essentially the discretized Orr-Sommerfeld eigenvalues system, modified by the elastic beam. The full derivation of eigenvalue problem is in the following paragraph.

After obtaining the steady solution $\bar{\mathbf{U}}$ from the Finite Element Equation, an infinitesimal perturbation $\Delta\mathbf{U}$ is added to it, so that $\mathbf{U} = \bar{\mathbf{U}} + \Delta\mathbf{U}$ is the solution to the current system. If the system is stable, $\mathbf{U}$ will approach the steady solution $\bar{\mathbf{U}}$ at time flies. Writing $\Delta\mathbf{U}$ in the exponential form of $e^{\omega t}\tilde{\mathbf{U}}$, where $\omega(= \omega_r + i\omega_i)$, $\tilde{\mathbf{U}}$ are the complex eigenvalue and eigenvector, respectively. By substituting the exponential form of $\Delta\mathbf{U}$ into (3.9), the matrix eigenvalue equation is

$$\left(\omega\bar{\mathbf{M}} + \bar{\mathbf{K}}\right)\tilde{\mathbf{U}} \equiv \tilde{\mathbf{R}} = \mathbf{0}, \tag{3.10}$$

where $\bar{\mathbf{K}}$ and $\bar{\mathbf{M}}$, are the stiffness matrix and mass matrix, which are determined by the steady solution $\bar{\mathbf{U}}$. $\bar{\mathbf{K}}$ is a sparse, asymmetric and nonsingular matrix, and also not positive definite or semi positive definite; $\bar{\mathbf{M}}$ is a sparse, asymmetric and singular matrix(the reason of singularity is that in the Navie-Stokes Equations, the density of fluid is not in every term), and it is not positive definite or semi positive definite, either. Using the command “spy” in MATLAB, the rough structures of the the stiffness matrix and the mass matrix are shown in the pictures below:

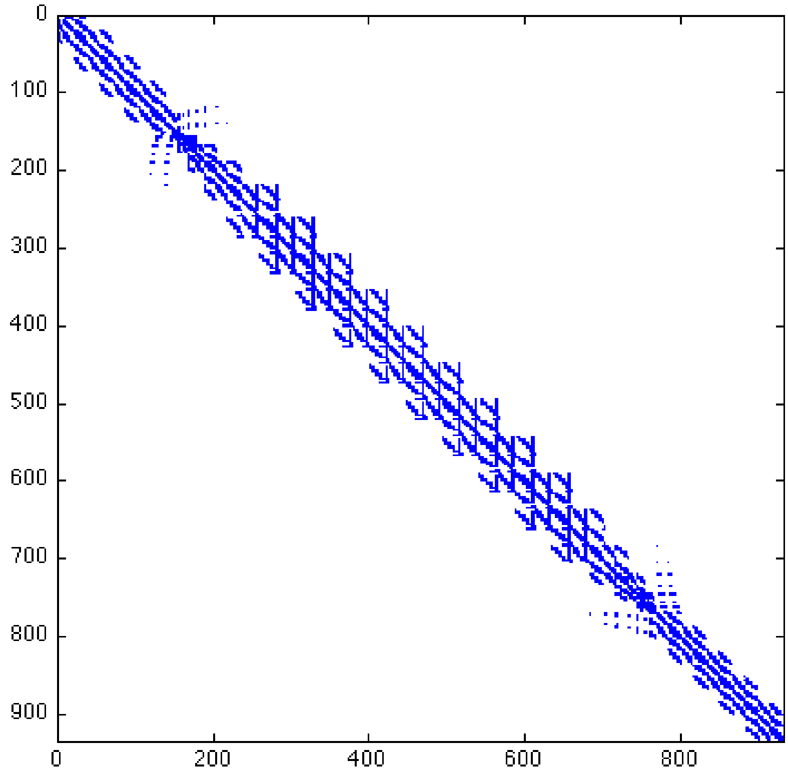


Figure 3-2: The structure of a stiffness matrix **K**(blue only indicates the positions of nonzero entries, while white means zero entries), and this is not a geometrical symmetry.

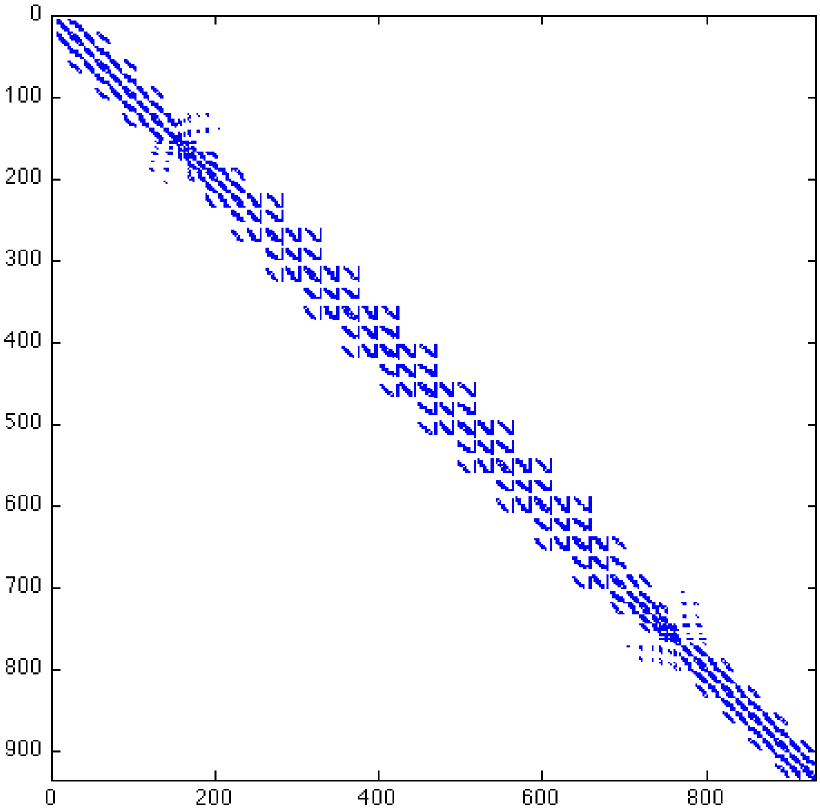


Figure 3-3: The structure of a mass matrix **M**(blue only indicates the positions of nonzero entries, while white means zero entries), and this is not a geometrical symmetry.

From Figure 3-2 and Figure 3-3, it is clear that both of them are sparse matrices, and the generalized eigenvalue problem (3.10) is solved by a QZ Algorithm [28] without taking account of the sparse structures of the two matrices. The QZ Algorithm solves for the complete set of eigensolutions of (3.10), which means it is extremely inefficient for the large dimensional eigenvalue problems. This is one of the most significant reasons that we are going to study a new eigensolver for the project of collapsible channel flows.

3.2 Numerical methods

In the numerical simulation, one of the most important factors, which will determine whether it is good or not is the numerical methods embedded in the project. This might also lead to the efficiency, the accuracy and the truth of the results. Hence, the numerical methods which are going to be used will be presented and discussed here.

3.2.1 Frontal solver

A frontal solver [45] has already been used in the collapsible channel flows for about two decades since Rast [116] firstly employed it in the Fluid-Membrane Model in 1994. Afterwards, Luo et al. carried the frontal solver forward in their further studies [15, 86, 87, 80]. A frontal solver is a very efficient method for solving linear equation groups, because it can save a great deal of computational time with ignoring the useless computations, especially for the sparse matrices.

According to Irons [47], a frontral solver is an approach to solving linear systems, which is used extensively in the finite element analysis. It is a variant of Gauss elimination that automatically avoids a great number of operations involving zero entries. A frontal solver builds a LU or Cholesky decomposition of a sparse matrix given as the assembly of element matrices, by assembling the matrix and eliminating equations only on a subset of elements at a time. The subset is so-called the front and it is essentially the transitional region between the part of the system already finished and the part not touched yet. During the whole process, the fully sparse matrix has never been assembled at all. Only some parts of the matrix are assembled as they enter the front. Processing the front involves dense matrix operations, which use the CPU efficiently. In a typical implementation, only the front is in memory, while the factors in the decomposition are written into files. The element matrices are read from files or created as needed and discarded.

The basic idea of frontal method or band method is the reanalysis of the Gauss Elimination. In the traditional technique the elimination implements after the assembly, while in the frontal method, both the assembly and the elimination implement alternately. The characteristic of frontal method is: All the matrices and vectors do not enter the memory storage in a natural order, but in the order which the computation goes ahead with; In the memory storage, we keep the entries of the matrices and vectors as few as possible. The whole computational process is introduced simply in the following:

1. In the element order, the element matrices and vectors are scanned, then assembled and saved in the memory storage.

2. Examining which degree of freedom has been assembled completely and making it to be the pivot entry. Using the pivot entry, the entries in the other rows or columns are eliminated.

3. After the elimination, the entries in the pivot row or columns are removed out of the memory storage.

4. Repeat 1-3 until all the elements finished.

5. In the order of elimination, from the end to the front, the back substitution are implemented to achieve the original solution.

The frontal method requires less memory storage than the other methods in the whole computational process, and it has been employed by a great many researchers in their studies. However, not only the alternative frontal process is so complicated, but also its adaption is intricate. With the development of the computational hardware in recent years, the frontal method was seldom used in the new computational codes, but it still considered as a very high efficient solver for linear system.

A simple frontal example

This is a simple demonstration which will illustrate the first two stages of a frontal method.

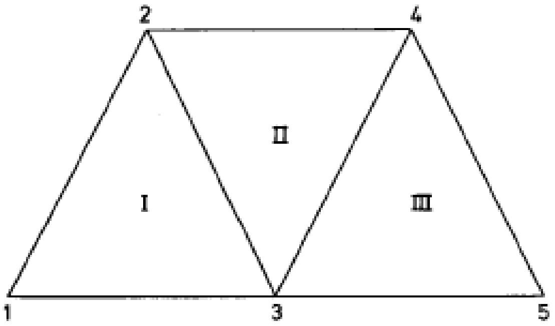


Figure 3-4: A three triangular element mesh for illustrating the process of a frontal solver

The figure above is a three-triangular-element mesh with five nodes. All the equations are stored in the order of ascending node number before solving it. The totality of finite element equations will be as

$$\mathbf{Ax} = \mathbf{b}, \tag{3.11}$$

where $[\mathbf{A}]$ is a linear matrix of coefficients of the vector $\{\mathbf{x}\}$, and $\{\mathbf{b}\}$ is the right hand side vector. This equation is assembled from the first element to the last element.

After assembling the first element, the state of the equation is :

$$\begin{bmatrix} a_{11}^{\mathrm{I}} & a_{12}^{\mathrm{I}} & a_{13}^{\mathrm{I}} \\ a_{21}^{\mathrm{I}} & a_{22}^{\mathrm{I}} & a_{23}^{\mathrm{I}} \\ a_{31}^{\mathrm{I}} & a_{32}^{\mathrm{I}} & a_{33}^{\mathrm{I}} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1^{\mathrm{I}} \\ b_2^{\mathrm{I}} \\ b_3^{\mathrm{I}} \end{bmatrix}, \tag{3.12}$$

where the superscript I denotes the element number from which the matrix entry was derived. The node-1 has the only contribution from element-I, which indicates that x_1 is independent of the other nodes, so this is a independent equations, and x_1 can be solved very easily. At the same time, the node-2 and node-3 have not been fully assembled, so the equation (3.12) is eliminated to

$$\begin{bmatrix} a_{22}^{\mathrm{I}} - \dfrac{a_{21}^{\mathrm{I}}}{a_{11}^{\mathrm{I}}} a_{12}^{\mathrm{I}} & a_{23}^{\mathrm{I}} - \dfrac{a_{21}^{\mathrm{I}}}{a_{11}^{\mathrm{I}}} a_{13}^{\mathrm{I}} \\ a_{32}^{\mathrm{I}} - \dfrac{a_{31}^{\mathrm{I}}}{a_{11}^{\mathrm{I}}} a_{12}^{\mathrm{I}} & a_{33}^{\mathrm{I}} - \dfrac{a_{31}^{\mathrm{I}}}{a_{11}^{\mathrm{I}}} a_{13}^{\mathrm{I}} \end{bmatrix} \begin{bmatrix} x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_2^{\mathrm{I}} - \dfrac{a_{21}^{\mathrm{I}} b_1^{\mathrm{I}}}{a_{11}^{\mathrm{I}}} \\ b_3^{\mathrm{I}} - \dfrac{a_{31}^{\mathrm{I}} b_1^{\mathrm{I}}}{a_{11}^{\mathrm{I}}} \end{bmatrix}, \tag{3.13}$$

and the equation (3.13) is the end of the element-I, afterwards, we are going to element-II. In the node-numbering order, the node-2 is the next assembling node, so the equation has a row and a column coming into as

$$\begin{bmatrix} a_{22}^{\mathrm{I}} - \dfrac{a_{21}^{\mathrm{I}}}{a_{11}^{\mathrm{I}}} a_{12}^{\mathrm{I}} & a_{23}^{\mathrm{I}} - \dfrac{a_{21}^{\mathrm{I}}}{a_{11}^{\mathrm{I}}} a_{13}^{\mathrm{I}} & a_{24}^{\mathrm{II}} \\ a_{32}^{\mathrm{I}} - \dfrac{a_{31}^{\mathrm{I}}}{a_{11}^{\mathrm{I}}} a_{12}^{\mathrm{I}} & a_{33}^{\mathrm{I}} - \dfrac{a_{31}^{\mathrm{I}}}{a_{11}^{\mathrm{I}}} a_{13}^{\mathrm{I}} & a_{34}^{\mathrm{II}} \\ a_{42}^{\mathrm{II}} & a_{43}^{\mathrm{II}} & a_{44}^{\mathrm{II}} \end{bmatrix} \begin{bmatrix} x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} b_2^{\mathrm{I}} - \dfrac{a_{21}^{\mathrm{I}} b_1^{\mathrm{I}}}{a_{11}^{\mathrm{I}}} \\ b_3^{\mathrm{I}} - \dfrac{a_{31}^{\mathrm{I}} b_1^{\mathrm{I}}}{a_{11}^{\mathrm{I}}} \\ b_4^{\mathrm{II}} \end{bmatrix}. \tag{3.14}$$

Then the node-2 is assembled completely, and the elimination can be run again as the previous derivation until all the nodes are finished. This is a simple derivation of the first two stages of a frontal process, the full details are in [45, 47]. Hood mentioned that the pivotal choice is a very important feature in the frontal process. The solution procedure fails if the pivot a_{11}^{I} is zero, so in the practical computation, it is necessary to avoid such an occurrence. The amount of pivotal choice depends on the number of fully assembled rows and columns, thus, the sufficient core has be allocated to the solution routine to allow a reasonable choice of pivots. The frontal solver in collapsible channel flows was written by Rast [116] . Luo and Pedley [86, 87, 85], Cai et al. [15] and Liu et al. [80] have employed the frontal method in their studies and were all satisfied with its high efficiency .

Matrix-vector product in FEM

According to the basic theory of finite element method, the global stiffness matrix is assembled by the element stiffness matrices in an order, and each entry in the element matrix has its own corresponding position in the global matrix. If there is a matrix-vector product going to implement, the traditional strategy is that assembly is implemented first, then the multiplication of the matrix and the vector is executed to find the product. For easy understanding, this whole process is illustrated with a three-element, five-node structure as :

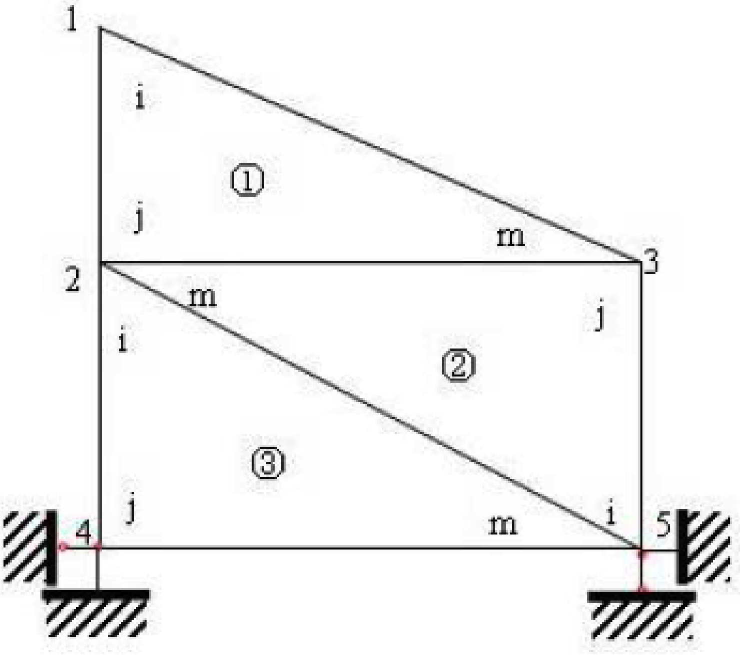


Figure 3-5: A three-element, five-node structure for illustrating matrix-vector product

The global number and the local number are shown in the picture. The global node numbers are 1,2,3,4,5, and the element local node number are i,j,m in the anti-clock wise direction, so the global stiffness matrix or mass matrix should be 10 × 10 matrix. First, all the element stiffness matrices are extended to 10 × 10 matrices, which are called element contributing matrices. The element local number should be same as the number of its position in the global matrix, so the element contributing matrices can be assembled together to form the global matrix. And the three element contributing matrices are as below:

$$\mathbf{K}^1 = \begin{bmatrix} k^1_{ii} & k^1_{ij} & k^1_{im} & 0 & 0 \\ k^1_{ji} & k^1_{jj} & k^1_{jm} & 0 & 0 \\ k^1_{mi} & k^1_{mj} & k^1_{mm} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.15}$$

$$\mathbf{K}^2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & k^2_{mm} & k^2_{mj} & 0 & k^2_{mi} \\ 0 & k^2_{jm} & k^2_{jj} & 0 & k^2_{ji} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & k^2_{im} & k^2_{ij} & 0 & k^2_{ii} \end{bmatrix} \tag{3.16}$$

$$\mathbf{K}^3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & k_{ii}^3 & 0 & k_{ij}^3 & k_{im}^3 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & k_{ji}^3 & 0 & k_{jj}^3 & k_{jm}^3 \\ 0 & k_{mi}^3 & 0 & k_{mj}^3 & k_{mm}^3 \end{bmatrix} \tag{3.17}$$

So the global stiffness matrix is

$$\mathbf{K} = \begin{bmatrix} k_{ii}^1 & k_{ij}^1 & k_{im}^1 & 0 & 0 \\ k_{ji}^1 & k_{jj}^1 + k_{mm}^2 + k_{ii}^3 & k_{jm}^1 + k_{mj}^2 & k_{ij}^3 & k_{mi}^2 + k_{im}^3 \\ k_{mi}^1 & k_{mj}^1 + k_{jm}^2 & k_{mm}^1 + k_{jj}^2 & 0 & k_{ji}^2 \\ 0 & k_{ji}^3 & 0 & k_{jj}^3 & k_{jm}^3 \\ 0 & k_{im}^2 + k_{mi}^3 & k_{ij}^2 & k_{mj}^3 & k_{ii}^2 + k_{mm}^3 \end{bmatrix} \tag{3.18}$$

Afterwards a vector $\mathbf{v}_1 = (v_1, v_2, v_3, v_4, v_5)$ is multiplied to the right of the global stiffness matrix, so the final matrix-vector product is

$$\mathbf{K}\mathbf{v}_1 = \begin{bmatrix} k_{ii}^1 v_1 + k_{ij}^1 v_2 + k_{im}^1 v_3 \\ k_{ji}^1 v_1 + (k_{jj}^1 + k_{mm}^2 + k_{ii}^3) v_2 + (k_{jm}^1 + k_{mj}^2) v_3 + k_{ij}^3 v_4 + (k_{mi}^2 + k_{im}^3) v_5 \\ k_{mi}^1 v_1 + (k_{mj}^1 + k_{jm}^2) v_2 + (k_{mm}^1 + k_{jj}^2) v_3 + k_{ji}^2 v_5 \\ k_{ji}^3 v_2 + k_{jj}^3 v_4 + k_{jm}^3 v_5 \\ (k_{im}^2 + k_{mi}^3) v_2 + k_{ij}^2 v_3 + k_{mj}^3 v_4 + (k_{ii}^2 + k_{mm}^3) v_5 \end{bmatrix} \tag{3.19}$$

The equation (3.19) is the final result of matrix-vector product with the traditional strategy. After this, we will deduce the matrix-vector product with another way which sets off from the element contributing matrix. The multiplications of the element contribution matrix and the vector are

$$\mathbf{K}^1\mathbf{v}_1 = \begin{bmatrix} k_{ii}^1 v_1 + k_{ij}^1 v_2 + k_{im}^1 v_3 \\ k_{ji}^1 v_1 + k_{jj}^1 v_2 + k_{jm}^1 v_3 \\ k_{mi}^1 v_1 + k_{mj}^1 v_2 + k_{mm}^1 v_3 \\ 0 \\ 0 \end{bmatrix} \tag{3.20}$$

$$\mathbf{K}^2\mathbf{v}_1 = \begin{bmatrix} 0 \\ k^2_{mm}v_2 + k^2_{mj}v_3 + k^2_{mi}v_5 \\ k^2_{jm}v_2 + k^2_{jj}v_3 + k^2_{ji}v_5 \\ 0 \\ k^2_{im}v_2 + k^2_{ij}v_3 + k^2_{ii}v_5 \end{bmatrix} \tag{3.21}$$

$$\mathbf{K}^3\mathbf{v}_1 = \begin{bmatrix} 0 \\ k^3_{ii}v_2 + k^3_{ij}v_4 + k^3_{im}v_5 \\ 0 \\ k^3_{ji}v_2 + k^3_{jj}v_4 + k^3_{jm}v_5 \\ k^3_{mi}v_2 + k^3_{mj}v_4 + k^3_{mm}v_5 \end{bmatrix} \tag{3.22}$$

The the assembly process is same the traditional work.

$$\mathbf{K}^*\mathbf{v}_1 = \begin{bmatrix} k^1_{ii}v_1 + k^1_{ij}v_2 + k^1_{im}v_3 \\ k^1_{ji}v_1 + k^1_{jj}v_2 + k^2_{mm}v_2 + k^3_{ii}v_2 + k^1_{jm}v_3 + k^2_{mj}v_3 + k^3_{ij}v_4 + k^2_{mi}v_5 + k^3_{im}v_5 \\ k^1_{mi}v_1 + k^1_{mj}v_2 + k^2_{jm}v_2 + k^1_{mm}v_3 + k^2_{jj}v_3 + k^2_{ji}v_5 \\ k^3_{ji}v_2 + k^3_{jj}v_4 + k^3_{jm}v_5 \\ k^2_{im}v_2 + k^3_{mi}v_2 + k^2_{ij}v_3 + k^3_{mj}v_4 + k^2_{ii}v_5 + k^3_{mm}v_5 \end{bmatrix} \tag{3.23}$$

Compared (3.19) with (3.23), they have the same form of the final results. So no matter which way we choose to implement, the same result of matrix-vector product will be obtained, but what the difference is that the dimension of the global matrix is large and there are so many zero entries in it, while the size of each element matrix is sufficient small. And if the matrix-vector product is implemented with the process of assembly, a frontal solver can be employed, which can save a great deal of memory storage while calculating, however, the process is a bit more complicated than the traditional work.

3.2.2 QR/QZ algorithm

QR algorithm

QR algorithm is an eigenvalue algorithm that is a procedure to calculate the eigenvalues and corresponding eigenvectors of a matrix. The basis of QR algorithm is QR transformation, which was developed in 1950s by Francis [24, 25] and by Kublanovskaya [71], working independently. The basic idea is to perform a QR decomposition, writing the matrix as a product of an orthogonal matrix Q and an upper triangular matrix R, multiply the factors in the reverse order, and iterate.

The practical QR algorithm is shown as :

Let $\mathbf{A}$ be the matrix of the standard eigenvalue problem, and $\mathbf{A}_0 = \mathbf{A}$. Starting from $k = 0$, at the k-th step, the QR decomposition $\mathbf{A}_k = \mathbf{Q}_k\mathbf{R}_k$ is calculated, where $\mathbf{Q}_k$ is the orthogonal matrix and $\mathbf{R}_k$ is the upper triangular matrix. Then $\mathbf{A}_{k+1} = \mathbf{R}_k\mathbf{Q}_k$ is computed.

$$\mathbf{A}_{k+1} = \mathbf{R}_k\mathbf{Q}_k = \mathbf{Q}_k^T\mathbf{Q}_k\mathbf{R}_k\mathbf{Q}_k = \mathbf{Q}_k^T\mathbf{A}_k\mathbf{Q}_k = \mathbf{Q}_k^{-1}\mathbf{A}_k\mathbf{Q}_k \tag{3.24}$$

Because of (3.24), all the $\mathbf{A}_k$ are similar and they have the same eigenvalues. The algorithm is numerically stable because it proceeds by orthogonal similarity transforms.

The matrices $\mathbf{A}_k$ converge to the Schur form of $\mathbf{A}_k$ [33]. The Schur form comes from the Schur decomposition or Schur triangulation. The Schur decomposition is shown as follows: if $\mathbf{A}$ is a $n \times n$ square matrix with complex entries, then $\mathbf{A}$ can be expressed as

$$\mathbf{A} = \mathbf{Q}\mathbf{U}\mathbf{Q}^{-1} \tag{3.25}$$

where $\mathbf{Q}$ is a unitary matrix(the inversion $\mathbf{Q}^{-1}$ is also the conjugate transpose$\mathbf{Q}^T$ of $\mathbf{Q}$), and $\mathbf{U}$ is an upper triangular matrix, which is called a Schur form of $\mathbf{A}$. Since $\mathbf{U}$ is similar to $\mathbf{A}$, it has the same multiset of eigenvalues, and since it is triangular, those eigenvalues are the diagonal entries of $\mathbf{U}$, then the eigenvalue problem is solved easily. In testing for convergence it is imparctical to require exact zeros, but the Gershgorim circle theorem [31, 144] provides a bound on the error.

The iteration equation (3.24) are fairly expensive in the practical computation. This can be mitigated by first bringing matrix $\mathbf{a}$ to the upper Hessenberg form, which has zero entries below the first subdiagonal, and it costs $\frac{10}{3}n^3 + O(n^2)$ arithmetic operations using a technique based on Householder reduction. The rate of convergence depends on the separation between eigenvalues, so a practical algorithm will use shifts, either explicit or implicit to increase separation and accelerate convergence.

In efficient computational codes, the QR algorithm is performed with multiple implicit shifts for several times. This is easy to understand. The first step is to express the $\mathbf{A}_0 = \mathbf{Q}\mathbf{A}\mathbf{Q}^T$; then, at every step, the first column of $\mathbf{A}_k$ is transformed via a small-size Householder similarity transformation to the first column of $p\left(\mathbf{A}\right)$(or $p\left(\mathbf{A}\right)e_1$, where $p\left(\mathbf{A}\right)$, of degree r, is the polynomial that defines the shift strategy (usually $p(x) = (x - \lambda)(x - \bar{\lambda})$, where λ and $\bar{\lambda}$ are the two eigenvalues of the trailing 2×2 principal submatrix $\mathbf{A}_k$, and this is so-called implicit double shift). The successive Householder transformation of size $r + 1$ are performed in order to return the working matrix $\mathbf{A}_k$ to upper Hessenberg form. This operation is known as bulge chasing, due to the peculiar shape of the non-zero entries of the matrix along the steps of the algorithm. The deflation is performed as soon as one of the sub-diagonal entries of $\mathbf{A}_k$ is sufficiently small. The implicit QR algorithm has already been employed in ARPACK, which reduces lots of arithmetic operations.

QZ algorithm

QZ algorithm is one of the direct method, which solves a generalized eigenvalue problems $\mathbf{A}x = \lambda\mathbf{B}x$, and it is the analog of the QR algorithm, so the basis of QZ algorithm is also the Schur decomposition as described in QR algorithm.

The brief idea of QZ algorithm can be illustrated as:
1. Simultaneously reduce **A** and **B** to upper Hessenberg form and upper triangular form(Schur form) by unitary equivalence transformations, respectively. The trick of QZ is the next step which is to reduce **A** also to upper Schur form, whilie keeping **B** in that form in this process.

2. The inverse process $\mathbf{AB}^{-1}$ is simulated by unitary equivalence transformations **Q** and **Z** on the matrix pair **A** and **B**; this is the key of the QZ iteration. If the iteration is successful, it reduces **A** to triangular or quasi-triangular form (that is, with 2 × 2 blocks along the diagonal, in order to avoid complex arithmetic), while preserving the triangular structure of **B**. At convergence, we have the generalized Schur form of **A** and **B**; hence, the orthogonal **Q** and **Z** have been calculated, so that **QAZ** and **QBZ** are upper triangular.

3. Eigenvalue can be computed from the diagonals of the triangular form. Eigenvectors can be computed as the eigenvectors of the triangular problem and then transformed back with **Z** to the eigenvectors of the original problem.

For more details, see the original QZ algorithm [98], and the routine we used in our project is from Garbow [28]. The QZ algorithm leads to the full set of eigenvalues, and it requires $O(n^3)$ floating point operations and $O(n^2)$ memory storages, where n is the order of **A** and **B**. In particular, it requires $30n^3$ floating pointing operations for computing the eigenvalues only, and if eigenvectors are desired, an extra $16n^3$ are located. These estimations are based on the experience about two QZ iterations per eigenvalue are sufficient. Consequently, the computational time of QZ algorithm has the relation with the order of the matrices in the third power n^3.

3.2.3 Arnoldi method

Besides the direct methods of solving eigenvalue problem, there is another kind of solvers which is “projection method”or “projecting method”. Just as its name implies, the projection method is to project the original eigenvalue problem to a subspace, then the dimension of the problem is relatively reduced, but the wanted eigenvalues is still kept in the subspace. The subspace which is often chosen to project is the Krylov subspace, which is

$$\mathcal{K}_m(\mathbf{A},\mathbf{v}_1) = span\{\mathbf{v}_1, \mathbf{A}\mathbf{v}_1, \ldots, \mathbf{A}^{m-1}\mathbf{v}_1\}. \tag{3.26}$$

Arnoldi method is one of the most popular projection method since 1951, and its basic idea is the Arnoldi iteration, which will be introduced in the following paragraphs.

Arnoldi iteration

Definition: If $\mathbf{A} \in \mathbf{C}^{n\times n}$, then a relation of the form

$$\mathbf{A}\mathbf{V}_k = \mathbf{V}_k\mathbf{H}_k + \mathbf{f}_k\mathbf{e}_k^T, \tag{3.27}$$

where $\mathbf{V}_k \in \mathbf{C}^{n\times k}$ and $\mathbf{V}_k$ has orthogonal columns, $\mathbf{V}_k^T\mathbf{V}_k = \mathbf{I}_k$, $\mathbf{H}_k \in \mathbf{C}^{k\times k}$ is an upper Hessenberg matrix with non-negative subdiagonal elements, $\mathbf{f}_k \in \mathbf{C}^n$ and $\mathbf{V}_k^H\mathbf{f}_k = \mathbf{0}$, $\mathbf{e}_k$ is the k-th base vector. The equation (3.27) is the Arnoldi factorization.

A k-step Arnoldi factorization can continue to the (k+1)-step Arnoldi factorization via the iterative formulas below:

$$\beta_{k+1} = \|\mathbf{f}_k\|_2, \mathbf{v}_{k+1} = \frac{1}{\beta_{k+1}} f_{\mathbf{k}}, \tag{3.28}$$

$$\mathbf{V}_{k+1} = [\mathbf{V}_k, \mathbf{v}_{k+1}], \tag{3.29}$$

$$\mathbf{w} = \mathbf{A}\mathbf{v}_{k+1}, \begin{bmatrix} \mathbf{h}_{k+1} \\ \alpha_{k+1} \end{bmatrix} = \mathbf{V}_{k+1}^T\mathbf{w}, \tag{3.30}$$

$$\mathbf{H}_{k+1} = \begin{bmatrix} \mathbf{H}_k & \mathbf{h}_{k+1} \\ \beta_{k+1}\mathbf{e}_k^T & \alpha_{k+1} \end{bmatrix}, \tag{3.31}$$

$$\mathbf{f}_{k+1} = \mathbf{w} - \mathbf{V}_{k+1} \begin{bmatrix} \mathbf{h}_{k+1} \\ \alpha_{k+1} \end{bmatrix} = \left(\mathbf{I} - \mathbf{V}_{k+1}\mathbf{V}_{k+1}^T\right)\mathbf{w}. \tag{3.32}$$

We can deduce that

$$\mathbf{A}\mathbf{V}_{k+1} = \mathbf{V}_{k+1}\mathbf{H}_{k+1} + \mathbf{f}_{k+1}\mathbf{e}_{k+1}^T, \tag{3.33}$$

$$\mathbf{V}_{k+1}^T\mathbf{V}_{k+1} = \mathbf{I}_{k+1}, \tag{3.34}$$

$$\mathbf{V}_{k+1}^T\mathbf{f}_{k+1} = \mathbf{0}. \tag{3.35}$$

Actually, the performance of k-step Arnoldi factorization is to calculate a unitary base $\mathbf{V}_k$, which is a unitary base of Krylov subspace $\mathcal{K}_k(\mathbf{V}_1, \mathbf{A}) = \{\mathbf{v}_1, \mathbf{A}\mathbf{v}_1, \cdots, \mathbf{A}^{k-1}\mathbf{v}_1\}$

The k-step Arnoldi factorization (3.27) can be used to solve the approximate eigenvalues and eigenvectors of a standard eigenvalue problem $\mathbf{Ax} = \lambda\mathbf{x}$ directly. By solving low dimensional eigenvalue problem $\mathbf{H}_k\mathbf{y} = \tilde{\lambda}\mathbf{y}$, $\left(\tilde{\lambda}, \tilde{\mathbf{x}}\right)$ can be obtained from $\left(\tilde{\lambda}, \mathbf{y}\right)$ via $\tilde{\mathbf{x}} = \mathbf{V}_k\mathbf{y}$, $\tilde{\mathbf{x}}$ is usually called Ritz vector, and $\tilde{\lambda}$ is called Ritz value, then the residual norm is defined here

$$\|\mathbf{r}\|_2 = \|\mathbf{A}\tilde{\mathbf{x}} - \tilde{\lambda}\tilde{\mathbf{x}}\|_2 = \|(\mathbf{AV}_k - \mathbf{V}_k\mathbf{H}_k)y\|_2 = |\beta_{k+1}\mathbf{e}_k^T\mathbf{y}|. \tag{3.36}$$

As long as $\|\mathbf{r}\|_2$ in (3.36) is small enough, the Ritz pairs are the best approximations of eigenpairs.

Spectral transformation

Spectral transformations are powerful tools for adjusting the way in which eigensolvers behave when coping with a problem. The general strategy consists in transforming the original problem into a new one in which eigenvalues are mapped to a new position while eigenvectors remain unchanged. There are two popular spectral transformations, which are shift-and-invert transformation and generalized Cayley transformation, and these have been widely used in the research works [74, 75, 95] .

Let us consider a generalized eigenvalue problem $\mathbf{Ax} = \lambda\mathbf{Bx}$, which is as same as our eigenvalue problem of collapsible channel flow. The shift-and-invert spectral transformation is that choosing a proper real pole σ (because both $\mathbf{A}$ and $\mathbf{B}$ are real matrix, we only allow a real σ to keep the computation in real arithmetic) to make $(\mathbf{A} - \sigma\mathbf{B})$ to be invertable, then $\mathbf{Ax} = \lambda\mathbf{Bx}$ is transformed to a new standard eigenvalue problem

$$\mathbf{Sy} = \theta\mathbf{y}, \tag{3.37}$$

where $\mathbf{S} = (\mathbf{A} - \sigma\mathbf{B})^{-1}\mathbf{B}$, and $\theta = \dfrac{1}{\lambda - \sigma}$. Finally, the Arnoldi method is employed to solve the equation (3.37), and the Ritz pairs $\left(\tilde{\lambda}, \tilde{\mathbf{y}}\right)$ obtained from (3.37) is used to approach the eigenpairs $(\lambda, \mathbf{x})$. The computational burden here is in the solution of $(\mathbf{A} - \sigma\mathbf{B})^{-1}\mathbf{B}$. This transformation not only maps the eigenvalues near the pole to largest magnitude one, but also maps the eigenvalues far away from the pole to zero. The spectral condition number of $\mathbf{S}$ could be pretty large, so the linear equation groups would be hard to solve because the rate of convergence of a Krylov-based iterative method [129, 34] depends strongly on the spectral condition number.

The generalized Cayley spectral transformation is considered to be a better one in the spectral condition number. Taking the generalized eigenvalue problem $\mathbf{Ax} = \lambda\mathbf{Bx}$ for example, the generalized Cayley spectral transformation is that taking μ as the zero of the Cayley transformation, the generalized eigenvalue problem $\mathbf{A}\mathbf{x} = \lambda\mathbf{B}\mathbf{x}$ is transformed to

$$\mathbf{C}\mathbf{y} = \gamma\mathbf{y}, \tag{3.38}$$

where $\mathbf{C} = (\mathbf{A} - \sigma\mathbf{B})^{-1}(\mathbf{A} - \mu\mathbf{B})$, and $\gamma = \dfrac{\lambda - \mu}{\lambda - \sigma}$, respectively. In contrast to the shift-and-invert transformation, the generalized Cayley transformation maps eigenvalues far from the pole close to one. If a proper pole σ can be chosen to be right of all the eigenvalues and $\mu > \sigma\lambda$, then the smallest eigenvalue of $\mathbf{C}$ is no smaller than one (in magnitude). Moreover, by judiciously choosing the pole, we can approximately bound the largest eigenvalues of $\mathbf{C}$ (in magnitude) resulting in a small (say order 10) spectral condition number.

We remark that $(\mathbf{A} - \sigma\mathbf{B})^{-1}$ exists in both shift-and-invert and generalized Cayley spectral transformations. In the practical computation, this is so expensive that it requires lots of arithmetic operations and memory storages. Hence, the frontal solver which has been used in our collapsible channel flow is employed here as well.

Implicit restarted Arnoldi method

Lehoucq and Sorensen [78] indicate that the selection of a starting vector will lead to the construction of an approximate basis of the desired invariant subspace of $\mathbf{A}$ in $\mathbf{A}x = \lambda\mathbf{x}$. The best possible starting vector should be a linear combination of a Schur basis for the desired invariant subspace. This can be expressed with

$$\mathbf{v}_1 = \alpha\sum_{j=1}^{r}\left(\|\mathbf{r}_2\|_2\tilde{\mathbf{x}}_j\right) = \alpha\sum_{j=1}^{r}\left(\|\mathbf{r}_2\|_2\mathbf{V}_k\mathbf{y}_j\right) = \alpha\mathbf{V}_k\sum_{j=1}^{r}\left(\|\mathbf{r}_2\|_2\mathbf{y}_j\right). \tag{3.39}$$

During the process of approaching eigenpairs, sometimes the Ritz pairs are so hard to converge to the eigenpairs or take a very long time to converge. Hence, a restarted strategy is very necessary to be used. Among all the Restarted Arnoldi methods [76], the Implicit Retarted Arnolid method(IRA) [79] is concerned to be best one. Its mission is to force the starting vector closer and closer to the desired invariant subspace. The *implicit* implies that the all the updating is completed with an implicitly shifted QR mechanism on $\mathbf{H}_k$. The restarted strategy starts at the k-th step Arnoldi factorization, then it extends the factorization to another p steps. After that, p shifted QR steps are applied on $\mathbf{H}_{k+p}$. The last p columns of the iteration are useless, and the iteration is defined by repeating the implicitly restarted strategy until convergence. An IRA process is illustrated in the following.

Let the starting vector $\mathbf{v}_1 \in \mathbf{C}^n$, and $\|\mathbf{v}_1\| = 1$, $k + p$ is the dimension of the projecting subspace, and p is the number of shifts, so

$$\mathbf{A}\mathbf{V}_{k+p} = \mathbf{V}_{k+p}\mathbf{H}_{k+p} + \mathbf{f}_{k+p}\mathbf{e}_{k+p}^T \tag{3.40}$$

is a $k+p$ Arnoldi factorization. Now we are going to avoid p matrix-vector product by applying implicitly shifted QR mechanism. For simplicity, we take $p = 1$ for example, and μ is a real shift. $\mathbf{H}_{k+1} - \mu\mathbf{I}$ satisfies the QR decomposition, so $\mathbf{H}_{k+1} - \mu\mathbf{I} = \mathbf{QR}$. The first implicitly shift starts from (3.40) with $p = 1$,

$$\mathbf{A}\mathbf{V}_{k+1} = \mathbf{V}_{k+1}\mathbf{H}_{k+1} + \mathbf{f}_{k+1}\mathbf{e}_{k+1}^T. \tag{3.41}$$

Substrate $\mu\mathbf{I}$ on both sides of (3.41), then we obtain that

$$(\mathbf{A} - \mu\mathbf{I})\,\mathbf{V}_{k+1} = \mathbf{V}_{k+1}\,(\mathbf{H}_{k+1} - \mu\mathbf{I}) + \mathbf{f}_{k+1}\mathbf{e}_{k+1}^T. \tag{3.42}$$

Because of $\mathbf{H}_{k+1} - \mu\mathbf{I} = \mathbf{QR}$, (3.42) can be written to

$$(\mathbf{A} - \mu\mathbf{I})\,\mathbf{V}_{k+1} = \mathbf{V}_{k+1}\mathbf{QR} + \mathbf{f}_{k+1}\mathbf{e}_{k+1}^T. \tag{3.43}$$

Multiplying $\mathbf{Q}$ on both sides of (3.43), we obtain

$$(\mathbf{A} - \mu\mathbf{I})\,\mathbf{V}_{k+1}\mathbf{Q} = (\mathbf{V}_{k+1}\mathbf{Q})(\mathbf{RQ}) + \mathbf{f}_{k+1}\mathbf{e}_{k+1}^T\mathbf{Q}. \tag{3.44}$$

$$\mathbf{A}\mathbf{V}_{k+1}\mathbf{Q} = (\mathbf{V}_{k+1}\mathbf{Q})(\mathbf{RQ} + \mu\mathbf{I}) + \mathbf{f}_{k+1}\mathbf{e}_{k+1}^T\mathbf{Q}. \tag{3.45}$$

We let $\mathbf{V}_{k+1}^+ = \mathbf{V}_{k+1}\mathbf{Q}$, $\mathbf{H}_{k+1}^+ = \mathbf{RQ} + \mu\mathbf{I}$, apparently, and $\mathbf{H}_{k+1}^+$ is still in an upper Hessenberg form. As the k-th column of $\mathbf{f}_{k+1}\mathbf{e}_{k+1}^T\mathbf{Q}$ in (3.45) is non-zero, (3.45) is not a normal Arnoldi factorization. However, (3.45) can be rewritten to

$$\mathbf{A}\begin{bmatrix} \mathbf{V}_k^+ & \mathbf{v}_{k+1}^+ \end{bmatrix} = \begin{bmatrix} \mathbf{V}_k^+ & \mathbf{v}_{k+1}^+ \end{bmatrix}\begin{bmatrix} \mathbf{H}_k^+ & h_{k+1}^+ \\ \beta_{k+1}^+\mathbf{e}_k^T & \alpha_{k+1}^+ \end{bmatrix} + \mathbf{f}_{k+1}\begin{bmatrix} \sigma_k\mathbf{e}_k^T & \gamma_k \end{bmatrix}, \tag{3.46}$$

where $\sigma_k = \mathbf{e}_{k+1}^T\mathbf{Q}\mathbf{e}_k$, $\gamma_k = \mathbf{e}_{k+1}^T\mathbf{Q}e_{k+1}$. The first k columns of (3.46) can be expressed as

$$\mathbf{A}\mathbf{V}_k^+ = \mathbf{V}_k^+\mathbf{H}_k^+ + \left(\beta_{k+1}^+\mathbf{v}_{k+1}^+ + \sigma_k\mathbf{f}_{k+1}\right)\mathbf{e}_k^T, \tag{3.47}$$

Let $\mathbf{f}_k^+ = \beta_{k+1}^+\mathbf{v}_{k+1}^+ + \sigma_k\mathbf{f}_{k+1}$, and $\mathbf{f}_k^+$ satisfies $\left(\mathbf{V}_k^+\right)^T\mathbf{f}_k^+ = \mathbf{0}$, hence the equation (3.47) is a k-th Arnoldi factorization, which the next iteration can start from. This shift can save k matrix-vector product in solving process. It is clear that all the derivation can be addressed in the following three figures.

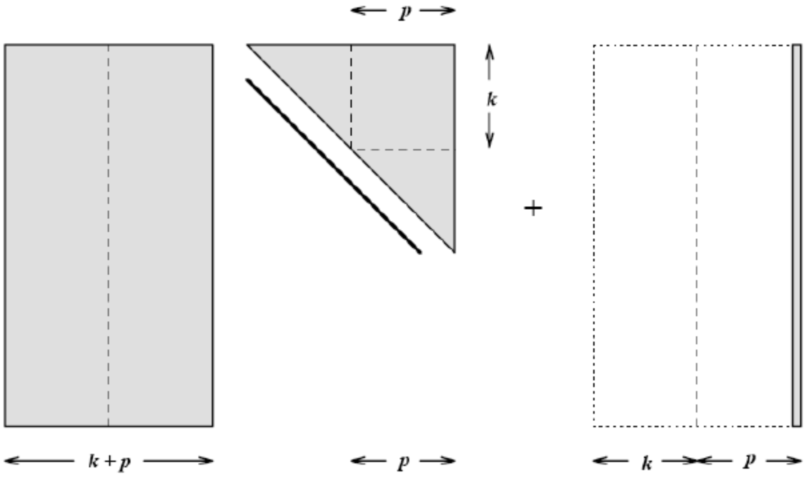


Figure 3-6: $\mathbf{V}_{k+p}\mathbf{H}_{k+p} + \mathbf{f}_{k+p}\mathbf{e}_{k+p}^T$, which is the right hand side of (3.40).

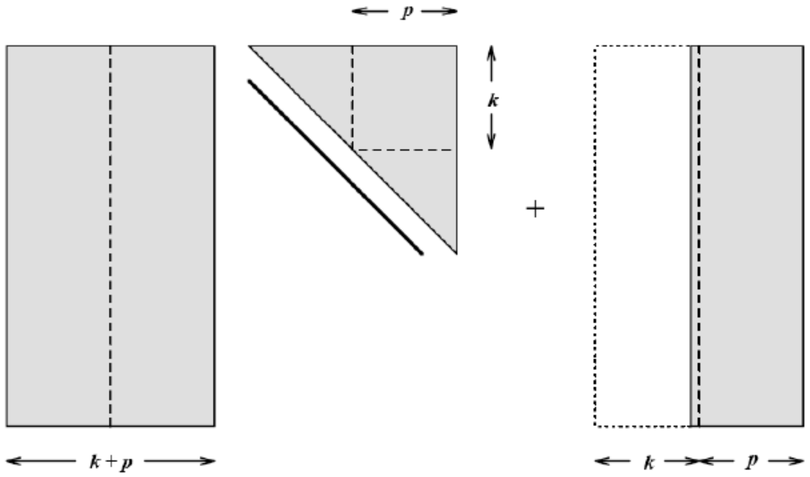


Figure 3-7: $\mathbf{V}_{k+1}^{+}\mathbf{H}_{k+1}^{+} + \mathbf{f}_{k+1}^{+}\mathbf{e}_{k+1}^T$, which shows the right hand side of (3.46).

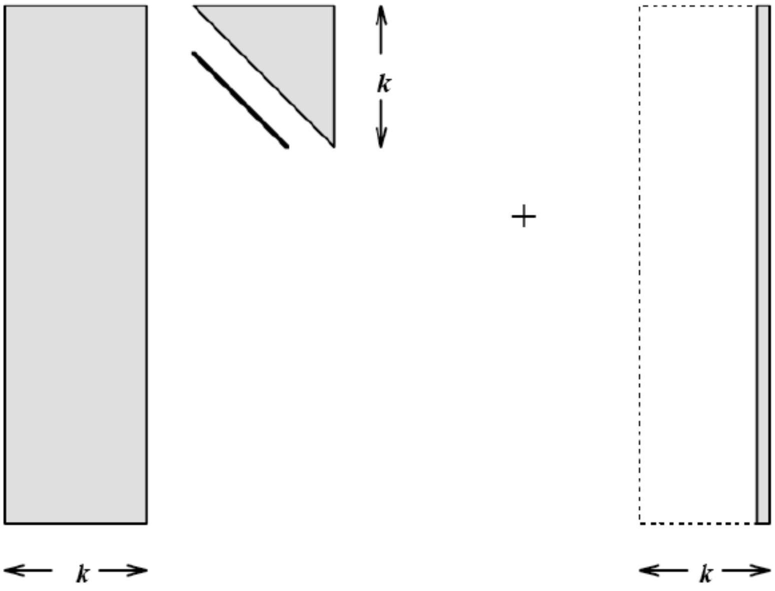


Figure 3-8: $\mathbf{V}_k^+\mathbf{H}_k^+ + (\beta_{k+1}^+\mathbf{v}_{k+1}^+ + \sigma_k\mathbf{f}_{k+1})\mathbf{e}_k^T$, which is the right hand side of (3.47), and the next iteration can start from here.

Multiplying $\mathbf{e}_1$ on both sides of (3.43), we obtain

$$(\mathbf{A}-\mu\mathbf{I})\,\mathbf{V}_{k+1}\mathbf{e}_1 = \mathbf{V}_{k+1}\mathbf{QR}\mathbf{e}_1 + \mathbf{f}_{k+1}\mathbf{e}_{k+1}^T\mathbf{e}_1, \tag{3.48}$$

and the equation (3.48) is simplified to

$$(\mathbf{A}-\mu\mathbf{I})\,\mathbf{v}_{k+1} = \mathbf{V}_{k+1}\mathbf{QR}\mathbf{e}_1 + \mathbf{0}. \tag{3.49}$$

The equation (3.49) can be further deduced to

$$(\mathbf{A}-\mu\mathbf{I})\,\mathbf{v}_{k+1} = \mathbf{V}_{k+1}\mathbf{QR}\mathbf{e}_1 = \mathbf{V}_{k+1}\mathbf{Q}\mathbf{e}_1\mathbf{e}_1^T\mathbf{R}\mathbf{e}_1. \tag{3.50}$$

Letting $\rho_{11} = \mathbf{e}_1^T\mathbf{R}\mathbf{e}_1$, and because of $\mathbf{V}_{k+1}^+ = \mathbf{V}_{k+1}\mathbf{Q}$, the equation (3.50) can be extended to

$$\begin{aligned}(\mathbf{A}-\mu\mathbf{I})\,\mathbf{v}_{k+1} &= \mathbf{V}_{k+1}\mathbf{QR}\mathbf{e}_1 && (3.51)\\ &= \mathbf{V}_{k+1}\mathbf{Q}\mathbf{e}_1\mathbf{e}_1^T\mathbf{R}\mathbf{e}_1\\ &= \mathbf{V}_{k+1}\mathbf{Q}\mathbf{e}_1\rho_{11}\\ &= \mathbf{V}_{k+1}^+\mathbf{e}_1\rho_{11}\\ &= \mathbf{v}_1^+\rho_{11}, && (3.52)\end{aligned}$$

Likewise, for the case $p > 1$, we have

$$\mathbf{v}_1^+ = \Psi(\mathbf{A})\mathbf{v}_1, \tag{3.53}$$

where $\Psi()$ in (3.53) is the matrix polynomial function, the p shifts are the zero point of the function. According to the power method, as long as the shift points are suitably chosen, the component of unwanted eigenvectors in $\mathbf{v}_1$ are discarded, while the part of wanted eigenvectors are kept, and $\Psi()$ is so-called polynomial filter function. Apparently, the dimension of the Krylov subspace depends on the order of $\Psi()$, and $\mathbf{v}_1$ approaches the wanted eigenvector as the increasing of the order of $\Psi()$, and the convergence benefits from this as well. Since the equation (3.53) dose not executes explicitly, but completes with an implicit QR factorization, this algorithm is so-called Implicitly Restarted Arnoldi method.

Lemma 3.10 in [138] proves that, using Ritz value to be the shift point, the $\mathbf{v}_1^+$ in IRA is also the linear combination of the wanted Ritz vectors. Morgan [102] proved that the strategy of IRA is the best one, and this is the most important reason that it converges faster than the other algorithms. Actually, the Ritz pairs obtained from the equation (3.41) and the equation (3.47) are same, which it is reliable to say that all the components of wanted Ritz pairs are kept while restarting, and in the coming iteration, the accuracy of the Ritz pairs will arise increasingly.

3.3 Conclusion

After this chapter, we should understand that the linear stability problem in collapsible channel flow is to solve large sparse asymmetric generalized eigenvalue problems, whose stiffness matrices are asymmetric and nonsingular and mass matrices are also asymmetric but singular, and the traditional solver for these generalized eigenvalue problems — QZ algorithm is so expensive to be employed that it requires quite a large storage memory and takes a fairly long time to finish the whole computation, especially while the dimension of the matrices increases, these are increasingly serious. Hence, it is necessary to find an efficient eigensolver for the project. Arnoldi method combined with a spectral transformation is considered as a much more efficient eigensolver than the traditional solver by some researchers. ARPACK is one of eigensolvers which base on Implicitly Restarted Arnoldi method, and is easy to be employed because of its reverse communication interface. Consequently, we decide to solve these complicated eigenvalue problems in collapsible channel flows with ARPACK, and in order to save the elapsed time in the process of spectral transformation, a frontal solver is employed to get rid of the operation has zero entries in. In next chapter, the new algorithms will be introduced, and after validating the new eigensolvers, some new numerical results are presented, compared and discussed.

Chapter 4

New Arnoldi Eigensolver

In the previous chapter, the generalized eigenvalue problems in collapsible channel flows, the frontal solver, the QR/QZ Algorithm and the Implicitly Restarted Arnoldi method were introduced. Now we are going to solve the eigenvalue problems of the linear stability analysis with Arnoldi method. Two new solvers will be introduced in this chapter, one is Arnoldi method with global matrices(AR-G), and the other is Arnoldi method with a frontal solver(AR-F). And AR-F is the efficient eigensolver we need, because not only the elapsed time is saved, but also the time increment with the matrix dimension is dropped.

4.1 Arnoldi method with global matrices(AR-G)

The linear stability problem in collapsible channel flows is to solve the generalized eigenvalue problem (3.10), and our older eigensolver is QZ algorithm [98], and in the Luo's paper [85], some neutral points and a neutral curve of flowrate-driven system have been published. Due to the memory storage requirement and the quite long computational time, their largest number of degree of freedom is 6152, whose mesh$(16 \times (30+60+60))$ is not enough for the further research work. So, it is necessary to find an much more efficient eigensolver for our project.

Considering the generalized eigenvalue problem (3.10) in chapter 3 and rewriting it to the general form of generalized eigenvalue problem

$$\mathbf{Kx} = \lambda \mathbf{Mx}, \tag{4.1}$$

where $\mathbf{K}$ and $\mathbf{M}$ are stiffness matrix and mass matrix as referred in Chapter 3, $\lambda = -\omega$ and $\mathbf{x} = \tilde{\mathbf{U}}$ are complex eigenvalue and eigenvectors, respectively, and we are trying to solve it with ARPACK [79], and during the tests of Arnoldi method we also have some Arnoldi functions(eig,eigs,sptarn) in MATLAB, which can provide some test work for us. ARPACK is a collection of FORTRAN subroutines, which is to solve large scale eigenvalue problems. In the manual book of ARPACK, it says that it can solve the asymmetric generalized eigenvalue problem by setting Mode=4 in the subroutine. However, the mass matrix can be used in ARPACK has to be symmetric and semi-positive definite. But our mass matrix is not in that form, so (4.1) cannot be worked out by ARPACK directly.

We have tried to solve the eigenvalue problem directly by change the original problem to some new form. The first attempt is to left multiply $\mathbf{M}^T$ on both sides of (4.1)

$$\mathbf{M}^T\mathbf{K}\mathbf{x} = \lambda\mathbf{M}^T\mathbf{M}\mathbf{x}. \tag{4.2}$$

This idea comes from that, after the left multiplication, $\mathbf{M}^T\mathbf{M}$ is a symmetric matrix which satisfies the symmetry of mass matrix in ARPACK. But this cannot make sure that $\mathbf{M}^T\mathbf{M}$ is semi-positive definite(both of ARPACK and MATLAB returns same error information-mass matrix should be semi-positive definite), and also $\mathbf{M}^T\mathbf{K}$ is singular after the left multiplication, which implies we might lose some eigenpairs. Hence, the left multiplication of $\mathbf{M}^T$ fails. The second test is to change (4.1) to the following form

$$\mathbf{K}^T\mathbf{M}\mathbf{x} = \frac{1}{\lambda}\mathbf{K}^T\mathbf{K}\mathbf{x}, \tag{4.3}$$

where $\mathbf{K}^T\mathbf{K}$ is a symmetric matrix, but it fails in the stage of spectral transformation, because all the subroutines return the failure that it cannot find a suitable shift to finish the inversion. At last, we have to give up solving the equation (4.1) by ARPACK directly again.

According to the algorithm [64] for asymmetric generalized eigenproblem, studying some work done by others, and considering that the stiffness matrix in (4.1) is nonsingular, we decide to combine the subroutine(Mode=1,full detail in ARPACK manual) in ARPACK with the shift-and-invert transformation. As the stiffness matrix $\mathbf{K}$ is nonsingular, $\sigma = 0$ can definitely make $(\mathbf{K} - \sigma\mathbf{M}) = \mathbf{K}$ invertable. By choosing $\sigma = 0$ and using the shift-and-invert transformation, the generalized eigenvalue problem (4.1) is converted to a standard eigenvalue problem

$$\mathbf{S}\mathbf{x} = \theta\mathbf{x}, \tag{4.4}$$

where $\mathbf{S} = \mathbf{K}^{-1}\mathbf{M}$, $\theta = \dfrac{1}{\lambda}$. Then we applied IRA(Mode=1 in ARPACK) to (4.4), we are going to find the eigenvalues with the largest real parts and their corresponding eigenvectors. The full IRA algorithm for the linear stability problem of our collapsible channel flows is:

Algorithm AR-G: Determine input parameters: n: the dimension of $\mathbf{K}$ and $\mathbf{M}$, which is also the total degree of freedoms of the FEM equations; $m(m \ll n)$: dimension of projecting subspace; k: the number of wanted eigenpairs; *tol*: the converged tolerance; $\sigma = 0$: the shift of spectral transformation.

1. Shift-and-invert transformation:

$$\mathbf{S} = \mathbf{K}^{-1}\mathbf{M} \tag{4.5}$$

2. Arnoldi iteration: creating a starting vector $\mathbf{v}_1$, and implementing the loop

for j=1,2,$\cdots$,m, **do**

$$\begin{aligned} \mathbf{v}_j &= \frac{\mathbf{v}_j}{\|\mathbf{v}_j\|_2} && (4.6)\\ \mathbf{w} &= \mathbf{S}\mathbf{v}_j && (4.7)\\ \mathbf{V}_j &= [\mathbf{v}_1, \cdots, \mathbf{v}_j] && (4.8)\\ \mathbf{H}_j &= \mathbf{V}_j^T\mathbf{S}\mathbf{V}_j && (4.9)\\ \mathbf{v}_{j+1} &= \left(\mathbf{I} - \mathbf{V}_j^T\mathbf{V}_j\right)\mathbf{w} && (4.10) \end{aligned}$$

endfor

3. Solving the Ritz pairs from the low dimensional eigenvalue problem $\mathbf{H}_j\mathbf{y} = \theta\mathbf{y}$, and obtaining $(\theta, \mathbf{y})$.

4. Calculating the residual norm $\|\mathbf{r}\|_2$ with the equation (3.36), if $\|\mathbf{r}\|_2 < tol$, stop; Otherwise, go back to step 2.

5. If $k < m$, carry out step 2 by creating a new starting vector.

And the whole process can be illustrated in the following flowchart clearly.

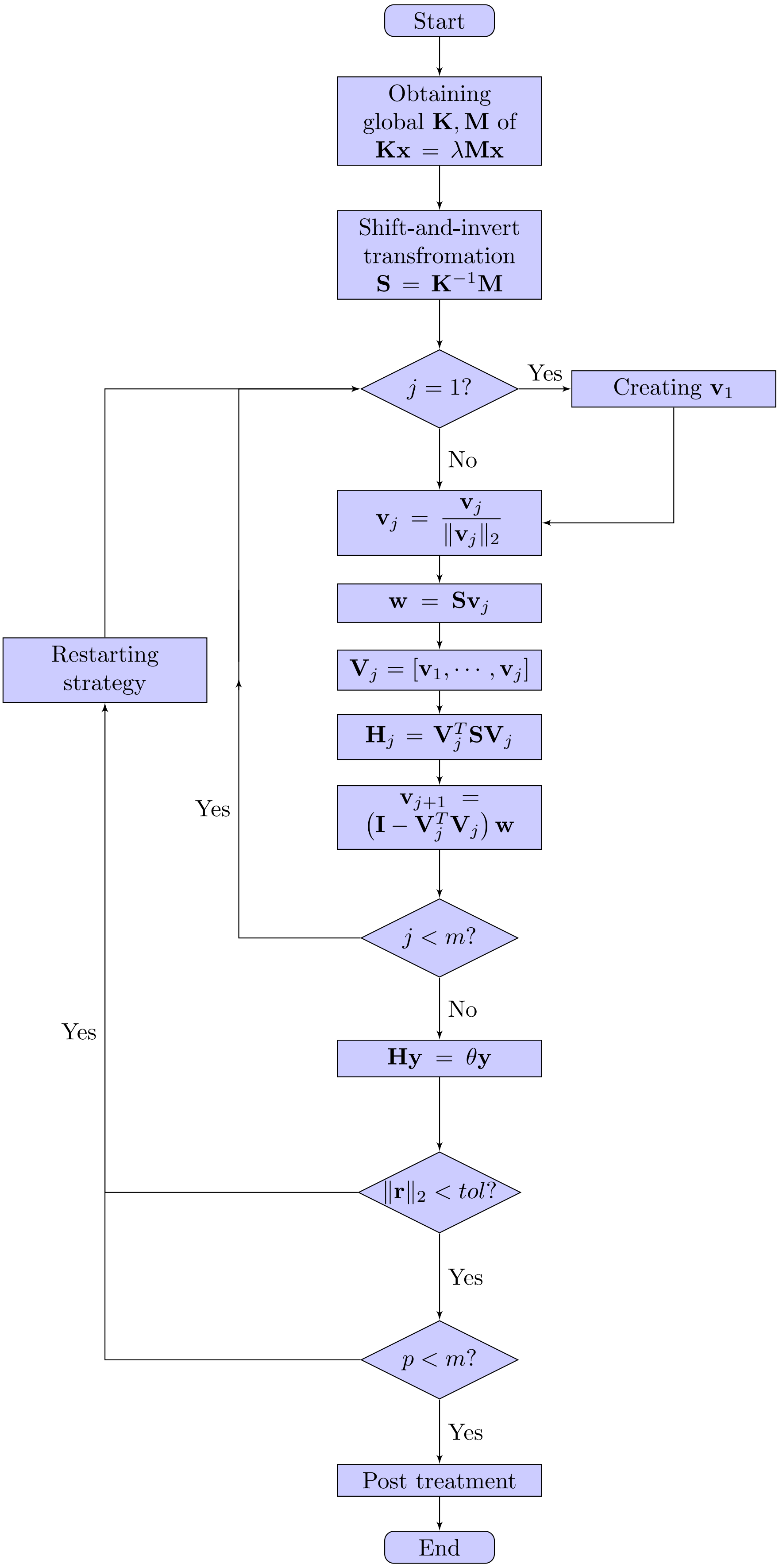

Start
Obtaining global $\mathbf{K}, \mathbf{M}$ of $\mathbf{Kx} = \lambda\mathbf{Mx}$
Shift-and-invert transfromation $\mathbf{S} = \mathbf{K}^{-1}\mathbf{M}$
$j = 1?$
Yes
Creating $\mathbf{v}_1$
No
$\mathbf{v}_j = \frac{\mathbf{v}_j}{\|\mathbf{v}_j\|_2}$
$\mathbf{w} = \mathbf{S}\mathbf{v}_j$
$\mathbf{V}_j = [\mathbf{v}_1, \cdots, \mathbf{v}_j]$
$\mathbf{H}_j = \mathbf{V}_j^T\mathbf{S}\mathbf{V}_j$
$\mathbf{v}_{j+1} = \left(\mathbf{I} - \mathbf{V}_j^T\mathbf{V}_j\right)\mathbf{w}$
Restarting strategy
Yes
$j < m?$
No
Yes
$\mathbf{Hy} = \theta\mathbf{y}$
$\|\mathbf{r}\|_2 < tol?$
Yes
$p < m?$
Yes
Post treatment
End

Figure 4-1: The flowchart of Arnoldi method with Global matrices, where the full expression of the large global matrices is required.

By making a study of the AR-G algorithm, we find that there are still some drawbacks in it. The first one is that during the whole algorithm it needs the explicit expression of **K** and **M**, which require a great much of memory storage; the second is the inverse matrix of stiffness matrix $\mathbf{K}^{-1}$. It destroys the sparsity of the two matrices, and the structure of **S** is shown in the following figure.

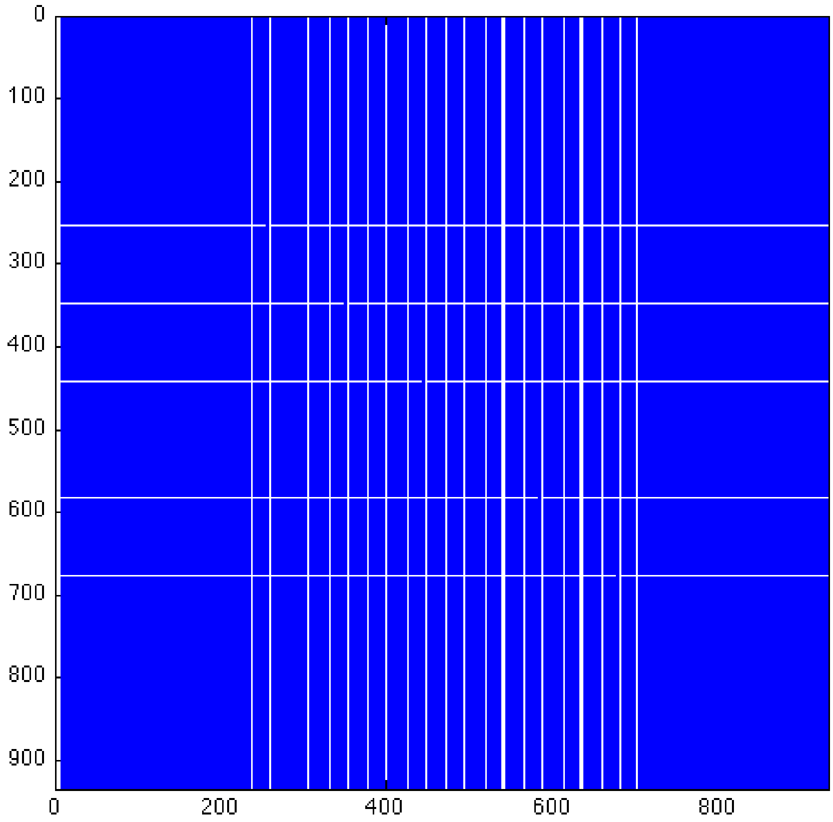


Figure 4-2: The structure of the matrix $\mathbf{S}(=\mathbf{K}^{-1}\mathbf{M})$, which has lost the original sparsity (blue indicates the positions of nonzero entries, while white means those of zero entries)

Some results of the AR-G algorithm are discussed in the section of numerical results, which prove that it can improve the solution of eigenproblem partly, but our worries come true as well, because of the two big matrices.

4.2 Arnoldi method with a frontal solver(AR-F)

In order to get rid of the drawbacks in the previous section, a frontal solver is employed to replace the process of inversion. As we know, the frontal solver is a Gauss elimination which solves a set of linear equation group. What is going to be done is to solve the equations (4.5) and (4.7) with a frontal solver.

By substituting the equations (4.5) into (4.7), we obtain that

$$\mathbf{w} = \mathbf{K}^{-1}\mathbf{M}\mathbf{v}_j. \tag{4.11}$$

Then by left multiplying $\mathbf{K}$ in the both sides of the equation (4.11), the equation (4.7) is changed to

$$\mathbf{K}\mathbf{w} = \mathbf{M}\mathbf{v}_j \tag{4.12}$$

Let $\mathbf{b} = \mathbf{M}\mathbf{v}_j$, and we have the new form of (4.12) as

$$\mathbf{K}\mathbf{w} = \mathbf{b}. \tag{4.13}$$

The equation (4.13) is solved by the frontal solver [45] in the element order, and the global stiffness matrix $\mathbf{K}$ is assembled by the element stiffness matrix $\mathbf{K}_e^l$ with the contributions to the nodes. And the full algorithm is described as

Algorithm AR-F: Determine the input parameters: n: the dimension of $\mathbf{K}$ and $\mathbf{M}$, which is also the number of degrees of freedom in the FEM equations; $m(m \ll n)$: the dimension of projecting subspace; k: the number of wanted eigenpairs; tol: the converged tolerance; $nelem$: the number of elements; $\sigma = 0$: the shift of spectral transformation.

1. Arnoldi iteration: creating a starting vector $\mathbf{v}_1$, and implementing the loop

   for j=1,2,$\cdots$,m, **do**

$$\begin{aligned} \mathbf{v}_j &= \frac{\mathbf{v}_j}{\|\mathbf{v}_j\|_2} & (4.14)\\ \mathbf{b} &= \sum_{l=1}^{nelem} \mathbf{M}_e^l \mathbf{v}_j & (4.15)\\ \sum_{l=1}^{nelem} \mathbf{K}\mathbf{w} &= \mathbf{b} & (4.16)\\ \mathbf{V}_j &= [\mathbf{v}_1, \cdots, \mathbf{v}_j] & (4.17)\\ \mathbf{H}_j &= \mathbf{V}_j^T \mathbf{S} \mathbf{V}_j & (4.18)\\ \mathbf{v}_{j+1} &= \left(\mathbf{I} - \mathbf{V}_j^T \mathbf{V}_j\right) \mathbf{w} & (4.19) \end{aligned}$$

   endfor

2. Solving the Ritz pairs from the low dimensional eigenvalue problem $\mathbf{H}_j\mathbf{y} = \theta\mathbf{y}$, and obtaining $(\theta, \mathbf{y})$.

3. Calculating the residual norm $\|\mathbf{r}\|_2$ with the equation (3.36), if $\|\mathbf{r}\|_2 < tol$, stop; Otherwise, go back to step 1.

4. If $k < m$, carry out step 1 by creating a new starting vector.

The magnificent improvement of AR-F over AR-G is that the shift-and-invert transformation is replaced by the equations (4.15) and (4.16). In AR-F, the two big matrices are not required any more, and the operations where the zero entries are involved will be never implemented. And the full flowchart is shown as

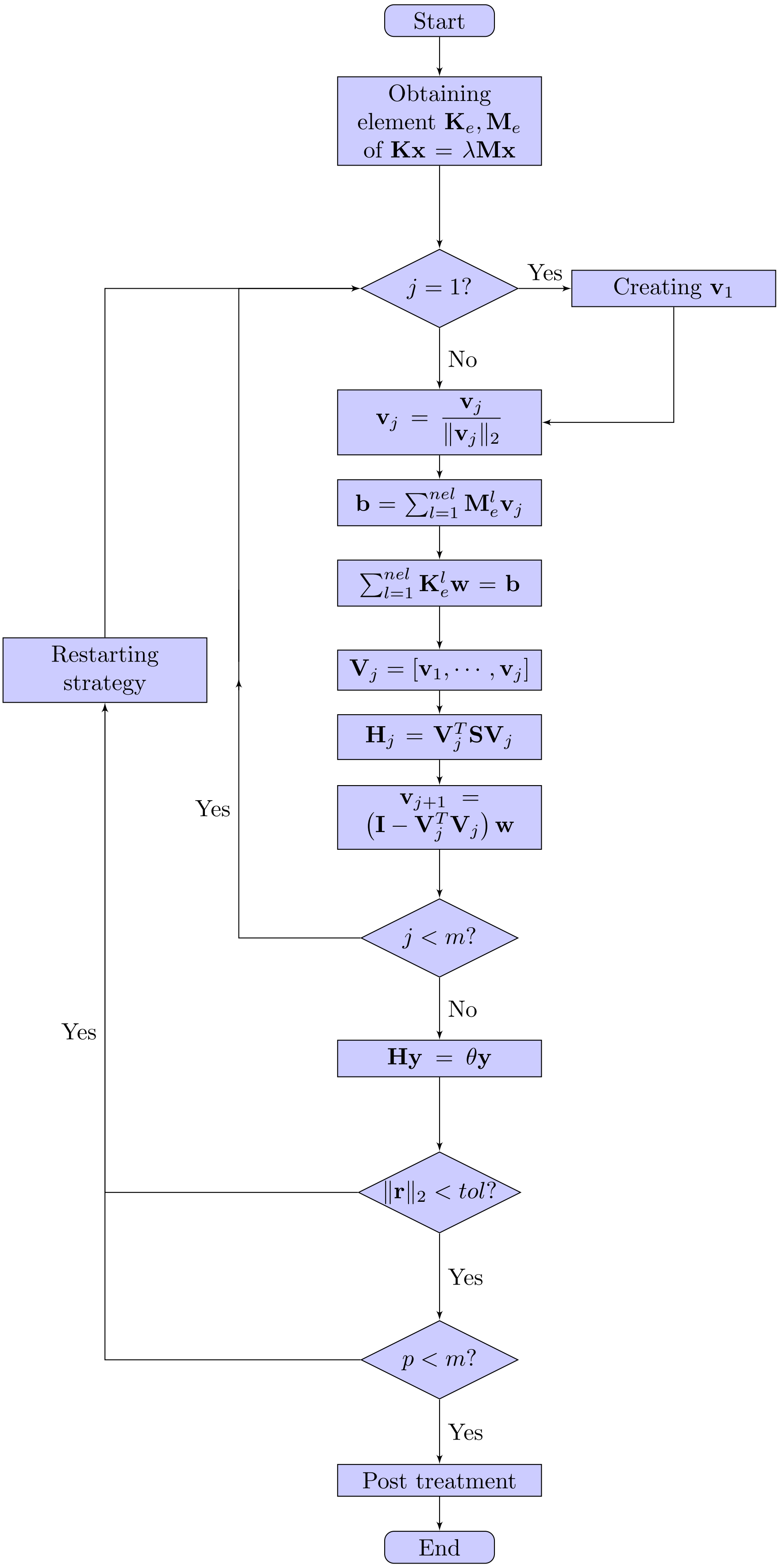

Start
Obtaining element $\mathbf{K}_e, \mathbf{M}_e$ of $\mathbf{Kx} = \lambda\mathbf{Mx}$
$j = 1?$
Yes
Creating $\mathbf{v}_1$
No
$\mathbf{v}_j = \frac{\mathbf{v}_j}{\|\mathbf{v}_j\|_2}$
$\mathbf{b} = \sum_{l=1}^{nel} \mathbf{M}_e^l \mathbf{v}_j$
$\sum_{l=1}^{nel} \mathbf{K}_e^l \mathbf{w} = \mathbf{b}$
Restarting strategy
$\mathbf{V}_j = [\mathbf{v}_1, \cdots, \mathbf{v}_j]$
$\mathbf{H}_j = \mathbf{V}_j^T \mathbf{S} \mathbf{V}_j$
Yes
$\mathbf{v}_{j+1} = \left(\mathbf{I} - \mathbf{V}_j^T \mathbf{V}_j\right) \mathbf{w}$
$j < m?$
No
Yes
$\mathbf{Hy} = \theta\mathbf{y}$
$\|\mathbf{r}\|_2 < tol?$
Yes
$p < m?$
Yes
Post treatment
End

Figure 4-3: The flowchart of Arnoldi method with a Frontal solver, where the full expression of large global matrices is replaced by a frontal solver tactfully.

4.3 Validations

In this section, we are going to validate the AR-G eigensolver and AR-F eigensolver in both flowrate-driven and pressure-driven system. Because QZ algorithm [85] works the completely set of eigenpairs, so their results are considered to the correct solution. If the same solutions can be worked out with AR-G and AR-F, then the new solvers are reliable. Some other results are also illustrated and discussed in the rest of this section.

4.3.1 Flow-driven system

Firstly, a coarse mesh$(8 \times (10 + 20 + 10))$ is structured for the computation. The number of degree of freedom in the problem is 933, and the matrix dimension is 933×933. One of the reason choosing this coarse mesh is that, the results obtained from the coarse mesh might not be reliable, but it takes less time than those refined meshes. About forty neutral points are worked out and listed in the following table.

Table 4.1: Neutral points of $(8 \times (10 + 20 + 10))$ mesh

Re	C_λ	ω_r	ω_i	QZ	ARG	ARF
400	1341.8	-9.26064×10^{-7}	2.02781	$\surd$	$\surd$	$\surd$
390	1329.5	-1.86179×10^{-6}	2.00375	$\surd$	$\surd$	$\surd$
380	1315	5.39827×10^{-6}	1.98462	$\surd$	$\surd$	$\surd$
370	1299	4.86144×10^{-6}	1.96359	$\surd$	$\surd$	$\surd$
360	1280.5	4.84548×10^{-6}	1.94014	$\surd$	$\surd$	$\surd$
350	1259	-4.57793×10^{-6}	1.91395	$\surd$	$\surd$	$\surd$
340	1233	1.47515×10^{-7}	1.88420	$\surd$	$\surd$	$\surd$
330	1201.5	-1.91046×10^{-6}	1.85026	$\surd$	$\surd$	$\surd$
320	1162	-4.19820×10^{-6}	1.81072	$\surd$	$\surd$	$\surd$
310	1110	-2.63270×10^{-6}	1.76312	$\surd$	$\surd$	$\surd$
300	1035	-9.57659×10^{-6}	1.70160	$\surd$	$\surd$	$\surd$
295	978	5.38461×10^{-6}	1.65945	$\surd$	$\surd$	$\surd$
294	963	1.25986×10^{-6}	1.64893	$\surd$	$\surd$	$\surd$
293	946	-1.36736×10^{-6}	1.63722	$\surd$	$\surd$	$\surd$
292	925	5.33728×10^{-6}	1.62317	$\surd$	$\surd$	$\surd$
291	896	3.04976×10^{-6}	1.60437	$\surd$	$\surd$	$\surd$
295	731	5.83500×10^{-6}	1.50809	$\surd$	$\surd$	$\surd$
300	689	2.60925×10^{-6}	1.48589	$\surd$	$\surd$	$\surd$
305	661	-9.47460×10^{-6}	1.47145	$\surd$	$\surd$	$\surd$
310	640.5	6.44089×10^{-6}	1.46119	$\surd$	$\surd$	$\surd$
320	610.25	-3.74760×10^{-6}	1.44581	$\surd$	$\surd$	$\surd$
330	588.4	-1.39879×10^{-6}	1.43408	$\surd$	$\surd$	$\surd$
340	570.8	3.30211×10^{-6}	1.42331	$\surd$	$\surd$	$\surd$
350	554.9	-3.76554×10^{-6}	1.41118	$\surd$	$\surd$	$\surd$
360	533.25	-2.29296×10^{-6}	1.38455	$\surd$	$\surd$	$\surd$
370	478.4	2.99300×10^{-6}	1.89398	$\surd$	$\surd$	$\surd$
370	453.2	3.12238×10^{-6}	1.10969	$\surd$	$\surd$	$\surd$
360	440.68	5.79503×10^{-6}	1.08760	$\surd$	$\surd$	$\surd$
350	425.449	-8.14527×10^{-6}	1.05704	$\surd$	$\surd$	$\surd$
340	409.9	8.21373×10^{-6}	1.01927	$\surd$	$\surd$	$\surd$
330	396.68	-8.46991×10^{-6}	0.97568	$\surd$	$\surd$	$\surd$
320	387.32	-2.27967×10^{-6}	0.93408	$\surd$	$\surd$	$\surd$
310	379.4	-7.55383×10^{-6}	0.90621	$\surd$	$\surd$	$\surd$
300	370.66	1.71856×10^{-6}	0.89341	$\surd$	$\surd$	$\surd$

From the Table 4.1, it is clear that, for the eigenvalues with the largest real parts, the results obtained from QZ, AR-G and AR-F are fully matched. So the AR-G and AR-F can find the same eigenvalues with largest real parts as the QZ algorithm

Luo et al. [85] presented a neutral curve of flow-driven system$(16 \times (30 + 60 + 60))$. In order to validate our new eigensolvers again, some control points from the Table-1 in that paper [85] are chosen for the validation. All the validating results are listed in the following Table 4.2.

Table 4.2: Neutral points of $(16 \times (30 + 60 + 60))$ mesh

		QZ		AR-G		AR-F		
Re	C_λ	ω_r	ω_i	ω_r	ω_i	ω_r	ω_i	Mode
250	56.88	-2.3899×10^{-5}	4.0593	√	√	√	√	Mode-4
230	139.84	-7.6221×10^{-5}	2.5336	√	√	√	√	Mode-3
284.91	310.94	-8.1041×10^{-5}	3.9059	√	√	√	√	Mode-3
485	360.24	-9.2627×10^{-5}	4.5421	√	√	√	√	Mode-3
485	383.79	-2.5951×10^{-5}	1.2277	√	√	√	√	Mode-2
279.62	500	1.7793×10^{-5}	1.2989	√	√	√	√	Mode-2
217	1800	-7.7874×10^{-5}	1.4029	√	√	√	√	Mode-2
220	2250	9.7981×10^{-5}	1.5109	√	√	√	√	Mode-2
273.74	1800	1.8821×10^{-5}	1.9211	√	√	√	√	Mode-2
400	1668.75	-5.8248×10^{-5}	2.1865	√	√	√	√	Mode-2
600	1653.13	-8.5403×10^{-5}	2.3659	√	√	√	√	Mode-2

All the results including the modes are fully matched again, so the new Arnoldi solvers are reliable for the collapsible channel flows in the flow-driven system. In the past paragraphs, we have validated the results of AR-G and AR-F with QZ algorithm in the flow-driven system, whose degrees of freedom might not be enough for further studies. Later we will investigate a refined mesh of the flow-driven system, and compare the results with the cascade structure (6152×6152) obtained in [85].

4.3.2 Pressure-driven system

The mesh and the degree of freedom in the pressure-driven system which Liu et al. used is $50 \times (200 + 140 + 100)$ and 55416, respectively. The eigenpairs cannot be worked out by the QZ algorithm because of the two large matrices. The Intel complier reports the segmentation faults, and even if it allows these two matrices, it would take quite a long time to calculate the full completely set of eigenpairs with QZ algorithm(remember the elapsed time of QZ algorithm is n^3). We take some neutral points(Mode-1) in Liu et al.'s paper [80] as the validating points. All of their results with $C_\lambda > 2000$ are obtained by the transient analysis, which might take one to three days to finish a full computation, and might take much longer time to locate the neutral point. More seriously, these neutral points might not be neutral enough, because it is very hard to compare the results of different steps while the computation is processing.

The explicit expression of the two matrices are also required by the AR-G solver, so it cannot be used in these large eigenvalue problems. Hence, AR-F is the only working eigensolver. With the input parameters(*Re*, *Pud*, *Ped* and C_λ), we recalculated their results and revised, the much more accurate neutral points are illustrated in the following tables, and the corresponding oscillation mode as well.

Table 4.3: The Mode-1 validating points of the pressure-driven system, obtained for $P_e - P_d = 1.95$.

Pud	C_λ	Re	ω_r	ω_i
1.0 (1.0,0%)	308.4 (307.85,0.178%)	115.61 (115.6, 0%)	-9.00310×10^{-6} (N/A)	0.518878 (0.519, 0.023%)
0.7031 (0.7,0.441%)	927 (927,0%)	107.89 (107.4, 0.454%)	-1.60704×10^{-5} (N/A)	0.725966 (0.726, 0.0005%)
0.6872 (0.68,1%)	2000 (2000,0%)	113.83 (112.66, 1%)	-3.31709×10^{-5} (N/A)	0.846492 (0.848, 0.178%)
0.70815 (0.7,1.15%)	5500 (5500,0%)	122.16 (121.90, 0.212%)	$+4.59145 \times 10^{-6}$ (N/A)	1.01443 (1.016, 0.155%)
1.21 (1.2,0.826%)	2.0×10^7 (2.0×10^7,0%)	226.62 (219, 3.36%)	$+7.08475 \times 10^{-6}$ (N/A)	3.61910 (3.623, 0.108%)

From the tables above, it is easy to know that the AR-F eigensolver can work out the correct eigenpairs of large scale eigenvalue problems. Not only we can reproduce all the neutral points by Liu et al. [80], see Figure 3 in [80]. The neutral points above N1-2, where Liu et al. had difficulty to obtain with the QZ solver, are computed using the AR-F solver, and are listed in Table 4.3. Note the top 5 points in Table 4.3 are compared with the ones obtained by [80] using an unsteady solver and manual bi-section in the parameter space, which is extremely laborious and computational intensive (it took days and weeks to identify one point). We remark that the agreement between the unsteady solver and the AR-F solver is excellent with the maximum error of 3.36%, though using the latter requires only a fraction of time ($\approx$ 30 minutes). In addition, we also obtained further eight points using the AR-F solver, for even greater values of C_λ using a much refined mesh($n = 55416$). To solve the eigenvalue problem with matrix of this size is simply out of question using the QZ, AR-G, or using the unsteady solver. Compared with the transient analysis, AR-F is not only faster than it, but also much more accurate than it.

4.4 Numerical results

4.4.1 Computational time

During the process of validating work, the computational time of the three eigensolvers are fairly different, AR-F is the fastest one, AR-G is the second, and QZ is the slowest among the three. So here we are going to investigate the elapsed time, which can help find how efficient the AR-F solver is. In order to make it fair, we use the same Re, P_e, β and C_λ, but the different meshes and number of degree of freedom (from 933×933 to 7325×7325)

Table 4.4: Elapsed time(in seconds) of different meshes at a same group of control parameters

DIM	QZ	AR-G					AR-F				
	time	nev	ncv	maxitr	no.	time(s)	nev	ncv	maxitr	no.	time(s)
933×933	18	2	20	50	2	15	2	40	50	2	7
2063×2063	261	2	20	100	2	151	2	20	100	2	23
		2	40	50	2	144	2	60	50	2	19
2629×2629	577	2	60	50	2	336	2	60	50	2	28
		4	40	150	2	610	4	40	150	4	169
		6	40	150	6	726	6	40	150	6	164
		6	40	200	6	726	6	40	200	6	164
		6	40	250	4	392	6	60	250	6	164
3942×3942	2554	2	40	50	2	1196	2	40	50	2	50
		2	60	50	2	1156	2	60	50	2	55
5117×5117	5116	2	40	50	2	2687	2	40	50	2	78
		2	60	50	2	2670	2	60	50	2	70
6152×6152	9158	2	40	50	2	4879	2	40	50	2	93
		2	40	100	2	4822	2	40	100	2	92
7325×7325	16816	2	80	250	2	8053	2	40	50	2	115
		2	80	300	2	8052	2	40	100	2	116

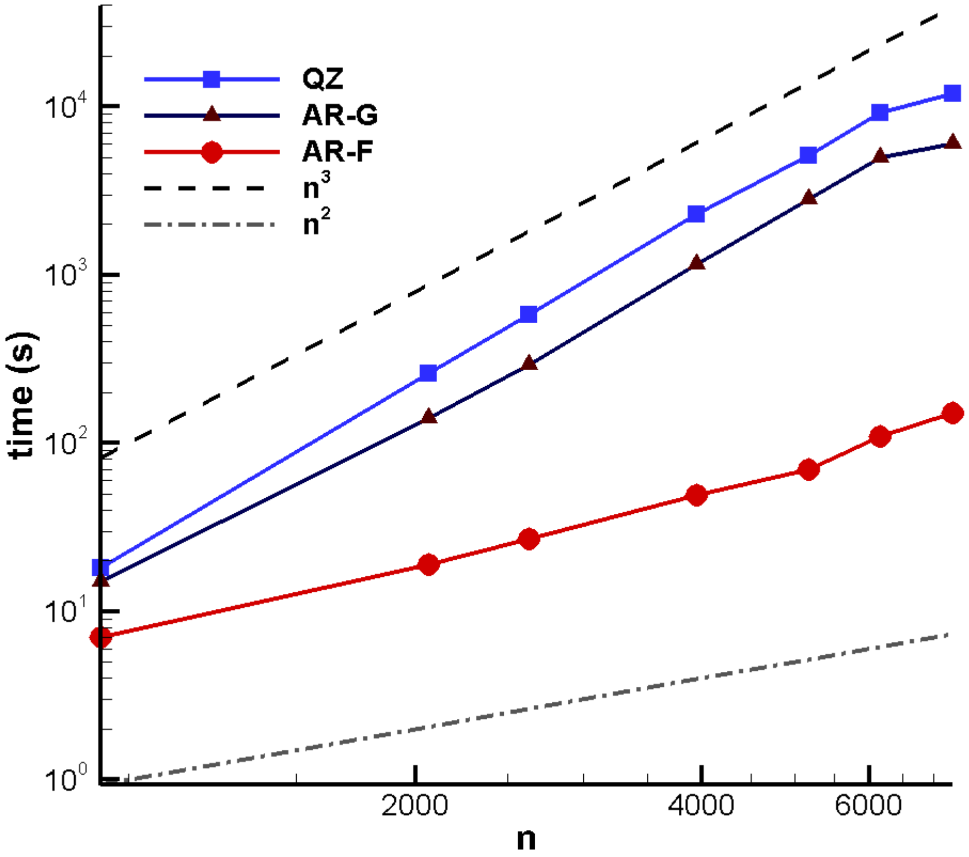


Figure 4-4: Comparison of computational time against matrix size using the QZ, AR-G & AR-F solvers.

Figure 4-4 shows the log-log scale plot of the computing time using the three algorithms versus the dimension size. For comparison purpose, the square and cubic power lines are also shown. The computing time of the QZ algorithm has a cubic relation with the order of matrix(in our research, the order of matrix is merely smaller than the dimension). Though in general requires less time, the AR-G algorithm has more or less the same increasing rate. Clearly, the AR-F algorithm is the fastest with the shortest computing time. It also has approximately the square growth rate with the matrix size.

4.4.2 Neutral stability curves revised

We further exploit the efficiency of the AR-F algorithm in studying the collapsible channel flows. Luo et al. [85] revealed a cascade structure in the flow-driven system, in particular, they obtained a mode-2 neutral stability curve in the C_λ-Re space using the QZ solver, where

C_λ is the scaled wall stiffness. However, the neutral curve was obtained using a relatively courser grid (6152×6152, i.e. with $16 \times (30 + 60 + 60)$ elements), which was the maximum size allowed by their computers then. Recently, Liu et al. analyzed the stability structure and identified a new Mode-1 neutral curve in the C_λ — P_{ud} space [80], where P_{ud} is the pressure drop ($P_{ud} = P_u - P_d$) along the channel, respectively. However, as the system requires a much refined mesh to resolve the thin boundary layer at the higher values of C_λ ($> 10^7$) and P_{ud}, they could not use the QZ solver to find the eigenvalues, but resorted to use an unsteady solver to solve for neutral points iteratively. With belp of the AR-F eigensolver, we can employ much finer mesh to obtain, validate, and extend the neutral curves studied in the previous studies [85, 80].

For the Mode-2 curve in the flow-driven system, we systematically increase the grid size from (933×933) to (7325×7325) and obtained the neutral points using the AR-F solver, the results are then compared to the results by [85], see Figure 3. The results of the denser mesh (7325×7325) agree excellently with the grid used by Luo et al. (6152×6152), therefore for the flow-driven system, the neutral curve obtained by the relatively coarse mesh is accurate.

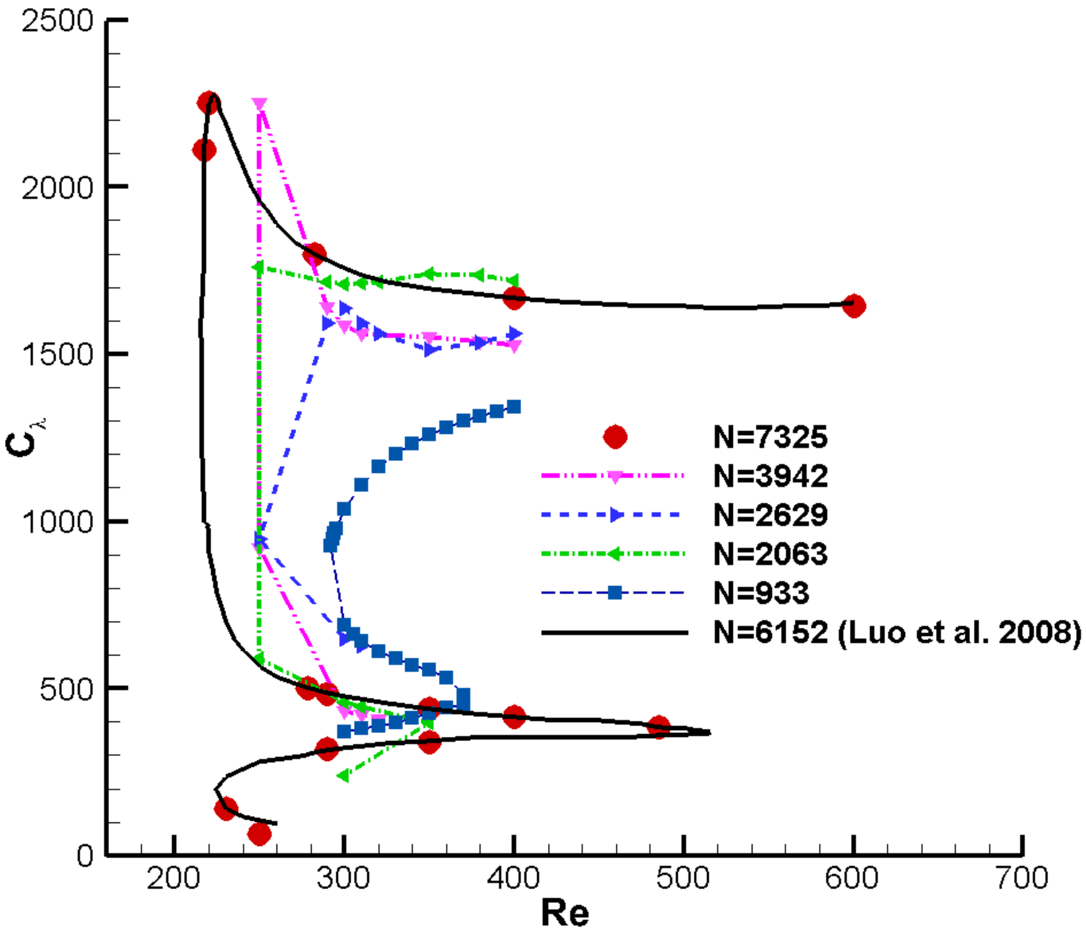


Figure 4-5: Mode-2 neutral curve obtained using the AR-F solver for various numbers of degree of freedom $n \times n$, where n=933, 2063, 2629, 3942, 5117 and 7325. The neutral curve obtained by Luo et al. [85] is shown in the solid curve, with $n = 6152$. In all cases, the non-dimensional transmural pressure $P_e - P_d$ is fixed at 1.95.

4.4.3 Mode-1 in flow-driven system

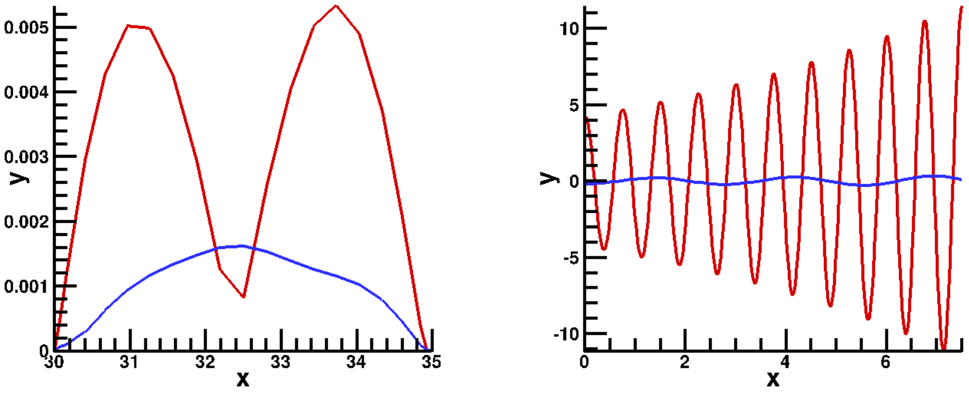
During the studies of flow-driven system in the past years, the neutral point of Mode-1 has never been found in Fluid-Beam Model, even if C_λ is very large. Some stable points of Mode-1 in the case where P_e is so small that the beam goes outwards. And also Luo et al. [85] have showed something about Mode-1in the amplitude plots of Mode-2 neutral points. Recall that in FBM, the fluid is incompressible, the velocity of the flow at the entrance is assumed as a constant u_0, and no reflux is allowed in the model. These three conditions make Mode-1 unstable point cannot be worked out. However, in some physiological applications, the Mode-1 oscillation might be still instructive to be investigated.

According to the three assumptions, we decide to make some changes to help Mode-1 unstable points appear. The first one is about the geometric model. In the previous studies, the geometric model was always kept to be $(1 \times (5+5+30))$, where the Mode-1 unstable point never occurred, so we change it to $(1 \times (25+5+15))$ and $(1 \times (30+5+10))$. The new geometric model with a long entrance might make the upstream flow move a bit more smoothly than before, and this would help the Mode-1 neutral point come out. The second is that the P_e and C_λ are chosen to make the initial deformation of the elastic beam is quite small. Because the fluid is assumed to be incompressible, it is very easy to imagine that if the deformation of the beam is so large that some other parts of the beam must be pushed outwards, this is definitely not Mode-1. A large C_λ means that initial stretching of beam is so large that the mechanism of the beam is same as the rigid wall. Combined with a suitable P_e, the amplitude of the self-excited oscillation can be made to be so small that the flow is fluxing to a channel with a uniform width. The last but necessary one is to employ a coarse mesh. The reason is that when using it, because of the fluid diffusion, the incompressible condition might not be completely satisfied. And in the practical or physiological applications, the fluid is not fully incompressible at all.

Via using the two new geometric models and considering the changes to the model, some new meshes($8 \times (80+20+20)$ and $12 \times (80+20+80)$) are created for the Mode-1 investigation. And in the these two models with a very long entrance, some unstable point of Mode-1 occurred with a group parameters of $Re = 480$, $P_e = 1.95$, $\beta = 1.0 \times 10^{10}$ and $C_\lambda = 2.45 \times 10^5$. The information of some eigenpairs is

Table 4.5: Unstable eigenmodes in flow-driven system

ω_r	ω_i	$Mode$
0.13325	8.36222	$Mode - 2$ unstable
8.17861×10^{-2}	2.26565	$Mode - 1$ unstable



(a) ampl

(b) yt at middle point

Figure 4-6: Unstable Mode-1 at $Re = 480$, $P_e = 1.95$,$\beta = 1.0 \times 10^{10}$, $C_\lambda = 2.45 \times 10^5$

The Figure 4-6 illustrates the eigenmodes of the two eigenvalues in Table 4.5. From the table, the frequency of Mode-1 is about a quarter of Mode-2, which also agrees with that different mode group has their own group of frequencies, and the higher modes have high frequencies. The figure 4-6 shows that the amplitude of Mode-1 is much smaller than Mode-2. Considering the Mode-2 has a higher frequency, the leading mode of this case is Mode-2. And the $y - t$ plot also shows this. It is clear that the Mode-1 unstable points can be obtained in a sparse mesh with a long entering channel, and these might be instructive to investigate the Mode-1 instability.

4.4.4 Neutral stability curves in refined meshes

After validating the AR-G and AR-F eigensolvers, the large scale eigenvalue problems in collapsed channel flows can be solved fast and accurately by them. Hence, in order to go on the study of this project, some large scale problems of the two different boundary conditions are solved, and both of the meshes in the two systems are refined enough.

Flow-driven system

Table 4.6: Neutral points of flow-driven system($50 \times (200 + 140 + 100)$ and number of freedom is 55392)

Point	Re	C_λ	ω_r	ω_i	$Mode$
	720	1649	-5.58304×10^{-5}	2.39606	Mode-2
N2-1	700	1672	$+3.30023\times10^{-5}$	2.40366	Mode-2
	680	1698	$+7.76506\times10^{-6}$	2.41115	Mode-2
	660	1725	$+8.90354\times10^{-6}$	2.41750	Mode-2
	640	1752	-2.00202×10^{-5}	2.42231	Mode-2
	620	1776	$+2.84852\times10^{-5}$	2.42428	Mode-2
N2-2	600	1797.5	-1.14649×10^{-6}	2.42381	Mode-2
	580	1813.3	-2.51121×10^{-6}	2.41955	Mode-2
	560	1821.5	-5.66738×10^{-7}	2.41074	Mode-2
	540	1820	-8.95562×10^{-6}	2.39642	Mode-2
	520	1809	$+1.09727\times10^{-6}$	2.37643	Mode-2
N2-3	500	1800	-6.13079×10^{-6}	2.35544	Mode-2
	480	1800	-4.10984×10^{-7}	2.33639	Mode-2
	460	1807	$+7.60077\times10^{-6}$	2.31823	Mode-2
	440	1817	$+8.08751\times10^{-6}$	2.29886	Mode-2
	420	1827	$+7.39848\times10^{-6}$	2.27648	Mode-2
	400	1836	$+6.30119\times10^{-6}$	2.25002	Mode-2
	380	1845.5	-2.11409×10^{-5}	2.21933	Mode-2
	360	1857.5	-1.66330×10^{-6}	2.18343	Mode-2
	340	1875	-3.07780×10^{-6}	2.14226	Mode-2
	320	1900	-3.49066×10^{-6}	2.09383	Mode-2
N2-4	300	1937	$+1.76015\times10^{-6}$	2.03575	Mode-2
	280	1998	-4.32243×10^{-6}	1.96523	Mode-2
	260	2107	-5.84939×10^{-6}	1.87705	Mode-2
	250	2195	$+6.01826\times10^{-6}$	1.82340	Mode-2
	240	2325	-7.16125×10^{-6}	1.76099	Mode-2
N2-5	235	2410	$+1.93529\times10^{-7}$	1.72480	Mode-2
	230	2512	-6.00555×10^{-6}	1.68425	Mode-2
	225	2627	$+2.92768\times10^{-6}$	1.63661	Mode-2
N2-6	220	2727	$+1.22905\times10^{-6}$	1.57726	Mode-2
	215	2664	-4.15608×10^{-6}	1.48951	Mode-2
	214	2580	$+1.37849\times10^{-6}$	1.46326	Mode-2
N2-7	213	2436	$+6.90523\times10^{-7}$	1.43017	Mode-2
	212	2148	-5.50585×10^{-7}	1.37982	Mode-2
	212	1800	$+3.23933\times10^{-4}$	1.33492	Mode-2
	212	1700	$+3.13894\times10^{-4}$	1.32084	Mode-2
N2-8	212	1600	$+2.39754\times10^{-4}$	1.30612	Mode-2
	212	1500	$+8.73406\times10^{-4}$	1.29071	Mode-2
	212	1400	-1.61622×10^{-4}	1.27450	Mode-2
	215	1000	$+4.33511\times10^{-5}$	1.23297	Mode-2
	219	800	-1.13902×10^{-4}	1.21836	Mode-2
	220	776	-9.24803×10^{-6}	1.21923	Mode-2
N2-9	240	580	-2.62525×10^{-6}	1.24025	Mode-2
	260	498	-4.29251×10^{-6}	1.26025	Mode-2
	280	467	-4.61032×10^{-5}	1.26996	Mode-2
N2-10	300	447	$+3.63519\times10^{-5}$	1.27266	Mode-2
	320	431	-1.34993×10^{-5}	1.26904	Mode-2
	340	418	-3.77324×10^{-5}	1.26212	Mode-2

To be continued

Continued

	360	407	-5.46797×10^{-5}	1.25317	Mode-2
N2-11	380	397.5	-3.62141×10^{-6}	1.24308	Mode-2
	400	388	-7.87792×10^{-5}	1.23041	Mode-2
	420	379	$+9.54023\times10^{-5}$	1.21669	Mode-2
N2-12	440	369	$+9.00410\times10^{-6}$	1.20026	Mode-2
	460	362	$+9.66773\times10^{-5}$	1.18911	Mode-2
	480	359	2.49244×10^{-4}	1.18578	Mode-2
N3-1	300	335.61	-5.67991×10^{-6}	4.07781	Mode-3
	260	316.75	-6.55135×10^{-6}	3.74018	Mode-3
N3-2	250	311	$+7.02530\times10^{-5}$	3.61682	Mode-3
	240	301.4	$+3.37421\times10^{-7}$	3.45648	Mode-3
	231	292.5	$+2.53972\times10^{-5}$	3.28247	Mode-3
	233.4	80	-7.67510×10^{-5}	4.28565	Mode-4
N4-1	231.15	60	$+9.69742\times10^{-6}$	3.85157	Mode-4
	235.32	40	-5.30980×10^{-5}	3.43092	Mode-4

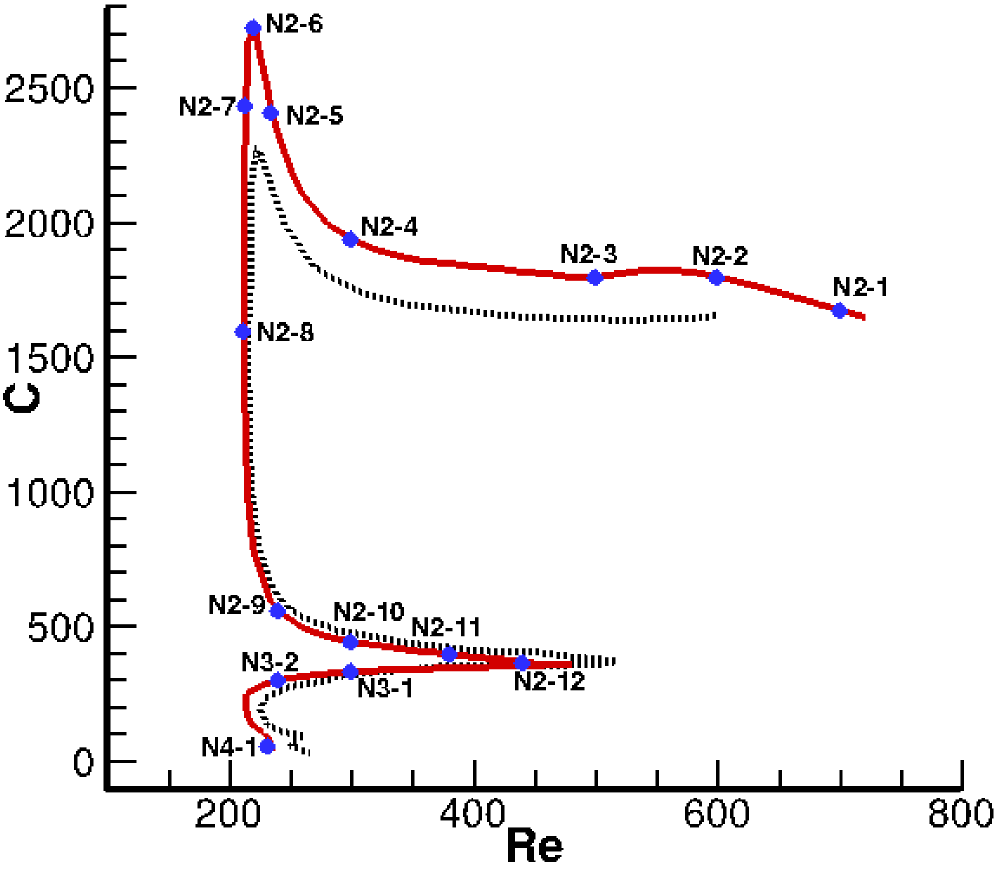


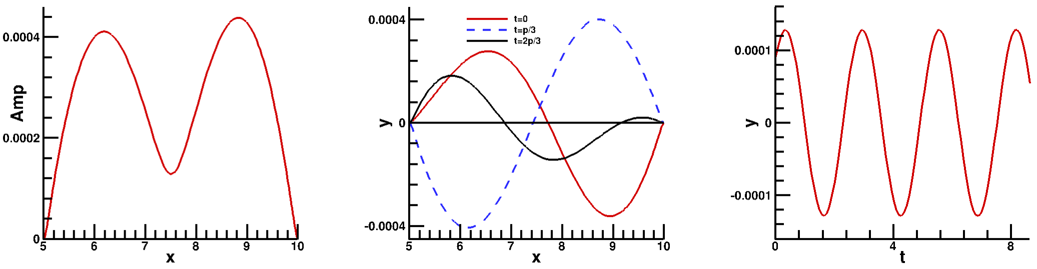
Figure 4-7: Neutral curve of the flow-driven system in a refined mesh(50 × (200 + 140 + 100) and number of degree of freedom is 55392). The dashed line is the neutral curve presented by Luo et al.(see Figure-3 in [85])

Compared with the results obtained by Luo et al., the neutral curve of the refined mesh $50 \times (200 + 140 + 100)$ keeps the cascade structure as the that presented by Luo et al. Both the number of elements in Y-direction and the number of elements in X-direction are increased in the new mesh, this improvement makes all the results much more reliable than before. Via comparing Figure 4-7 with Figure 3 in [85], the left bound of the curve remain in the same position(around at the line of Re=212), which means all the oscillations are damped out by the viscosity of the fluid. The new neutral curve has a shift in Y-direction. This achieves through the increase of element number in Y-direction, which would make the simulation to be much more accurate. As C_λ is the wall stiffness, a high C_λ means that both the velocity and the displacement of the beam are fairly small, so it definitely needs much more elements in Y-direction than lower C_λ. We also extend the branch of Mode-2 to $Re = 720$. From the picture, we find the curve drops slightly after $Re = 600$, but we have to stop the search of the Mode-2 curve at $Re = 720$, because of the assumption of the laminar fluid in our model.

From the Table 4-7, it is easy to find that normally the frequency of a high-order oscillation is larger than low-order ones except $Re > 620$. This agrees with the frequency curve against Re or C_λ in [85]. Considering the Mode-2 neutral curve for example, the lowest frequency might occur in the left bottom of the Mode-2 neutral curve, and if we fix $Re(C_\lambda)$, the frequency will decrease along with the drop of $C_\lambda(Re)$. And in the energy analysis, the direction of $P1 - D$ plot is changed at around the point of the lowest frequency. And it could be implied that these might be same to the Mode-3 and Mode-4 neutral points in the Table 4-7.

Eigenmodes at neutral points

Some operating points are marked in Figure 4-7, and the coresponding plots of amplitude, beamshape and displacement of a point in Y-direction are shown here.

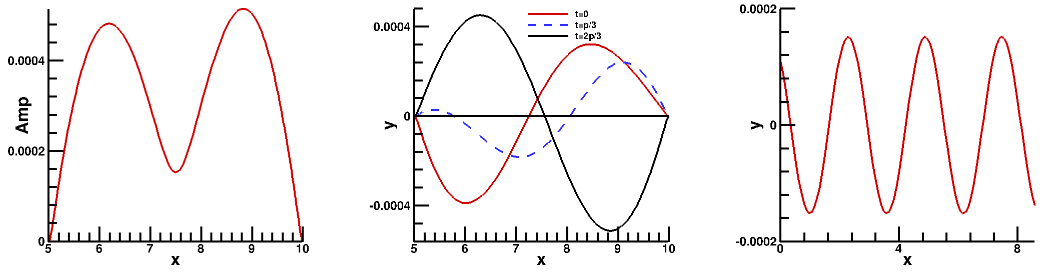


(a) amplitude

(b) beamshape

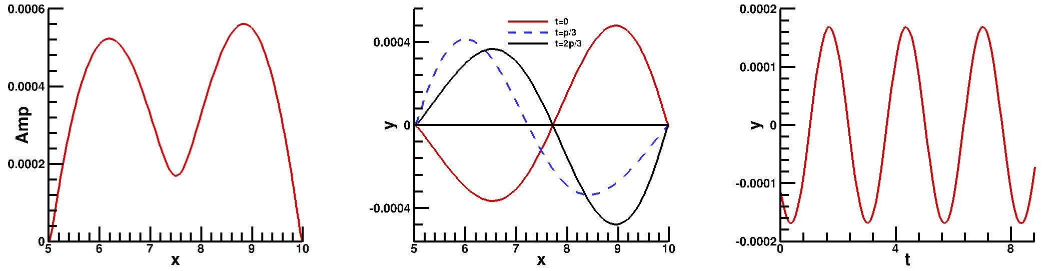
(c) y-t at x=7.5

Figure 4-8: N2-1: $Re = 700$, $C_\lambda = 1672$, $\omega_r = +3.30023 \times 10^{-5}$, $\omega_i = 2.40366$, Mode-2



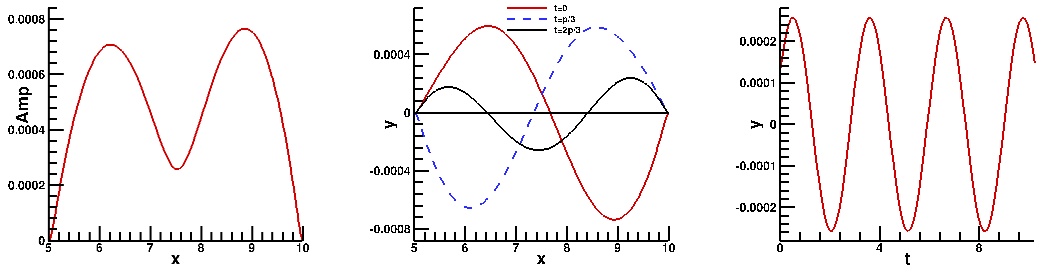
(a) amplitude (b) beamshape (c) y-t at x=7.5

Figure 4-9: N2-2: $Re = 600$, $C_\lambda = 1797.5$, $\omega_r = -1.14649 \times 10^{-6}$, $\omega_i = 2.42381$, Mode-2



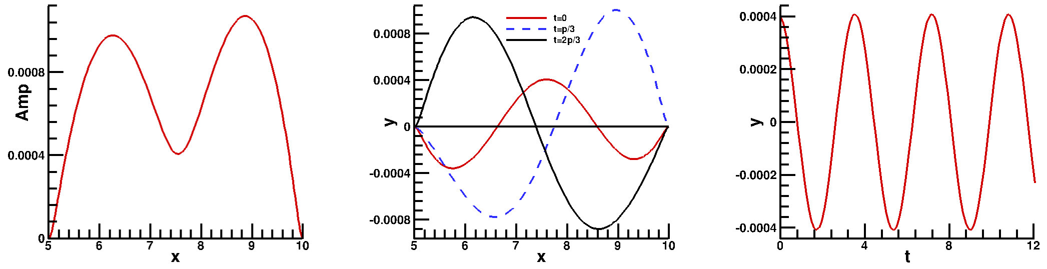
(a) amplitude (b) beamshape (c) y-t at x=7.5

Figure 4-10: N2-3: $Re = 500$, $C_\lambda = 1800$, $\omega_r = -6.13079 \times 10^{-6}$, $\omega_i = 2.35544$, Mode-2



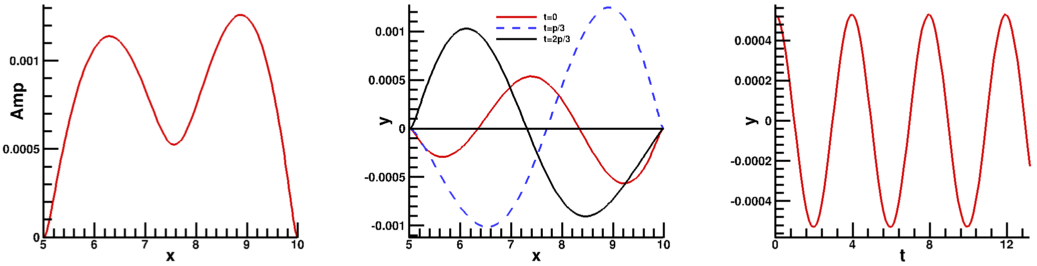
(a) amplitude (b) beamshape (c) y-t at x=7.5

Figure 4-11: N2-4: $Re = 300$, $C_\lambda = 1937$, $\omega_r = 1.76015 \times 10^{-6}$, $\omega_i = 2.03575$, Mode-2



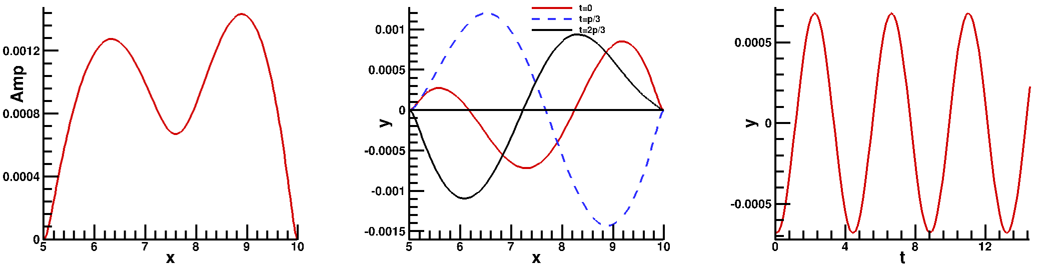
(a) amplitude (b) beamshape (c) y-t at x=7.5

Figure 4-12: N2-5: $Re = 235$, $C_\lambda = 2410$, $\omega_r = 1.93529 \times 10^{-7}$, $\omega_i = 1.72480$, Mode-2



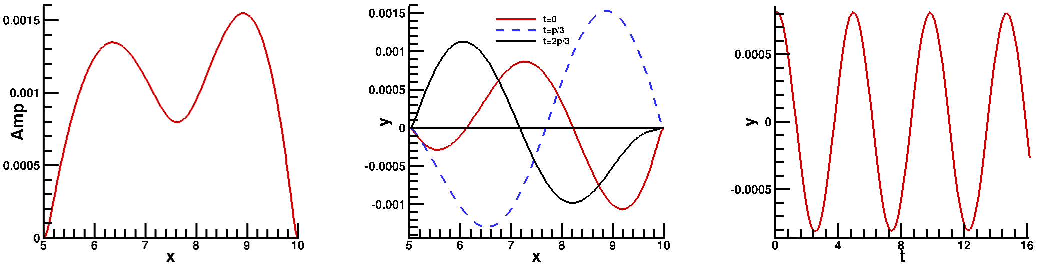
(a) amplitude (b) beamshape (c) y-t at x=7.5

Figure 4-13: N2-6: $Re = 220$, $C_\lambda = 2727$, $\omega_r = 1.22905 \times 10^{-6}$, $\omega_i = 1.57726$, Mode-2



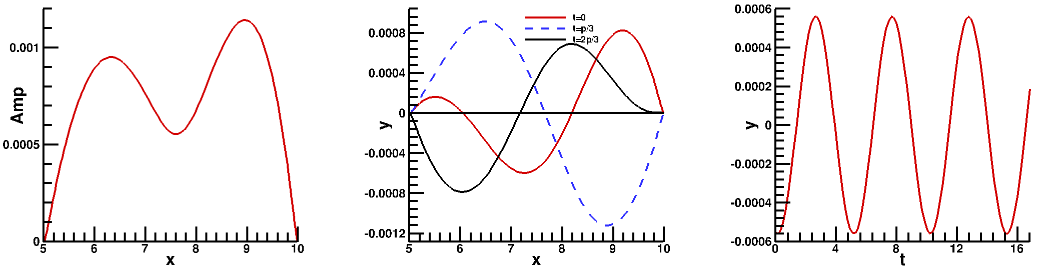
(a) amplitude (b) beamshape (c) y-t at x=7.5

Figure 4-14: N2-7: $Re = 213$, $C_\lambda = 2436$, $\omega_r = 6.90523 \times 10^{-7}$, $\omega_i = 1.43017$, Mode-2



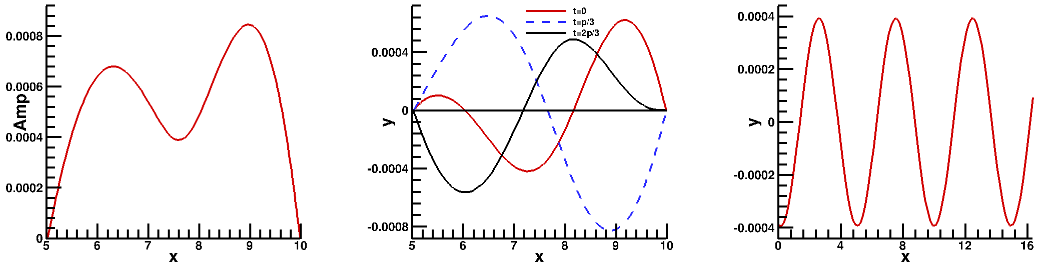
(a) amplitude (b) beamshape (c) y-t at x=7.5

Figure 4-15: N2-8: $Re = 212$, $C_\lambda = 1600$, $\omega_r = 2.39754 \times 10^{-4}$, $\omega_i = 1.30612$, Mode-2



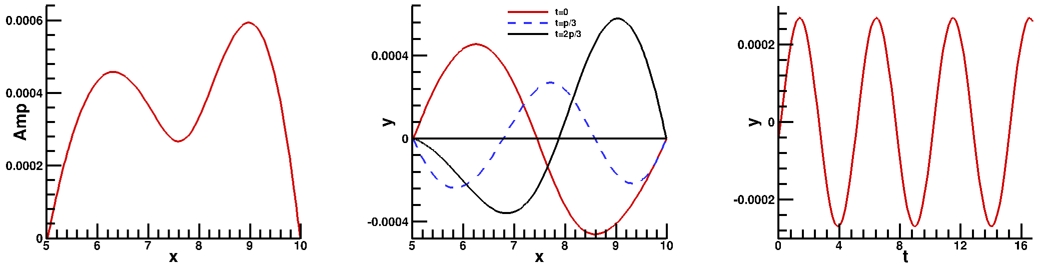
(a) amplitude (b) beamshape (c) y-t at x=7.5

Figure 4-16: N2-9: $Re = 240$, $C_\lambda = 558$, $\omega_r = -2.62525 \times 10^{-6}$, $\omega_i = 1.24025$, Mode-2



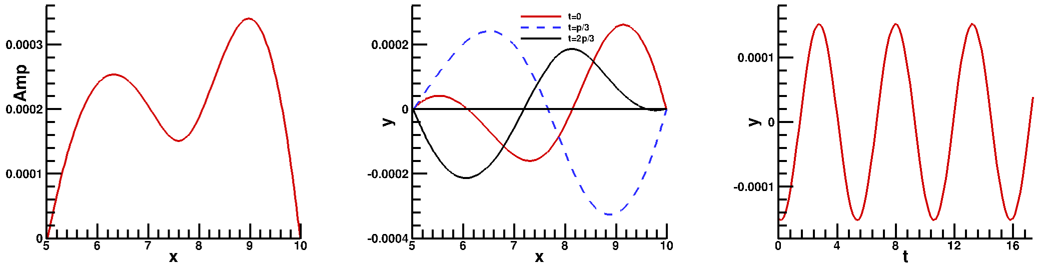
(a) amplitude (b) beamshape (c) y-t at x=7.5

Figure 4-17: N2-10: $Re = 300$, $C_\lambda = 447$, $\omega_r = -3.63519 \times 10^{-5}$, $\omega_i = 1.27266$, Mode-2



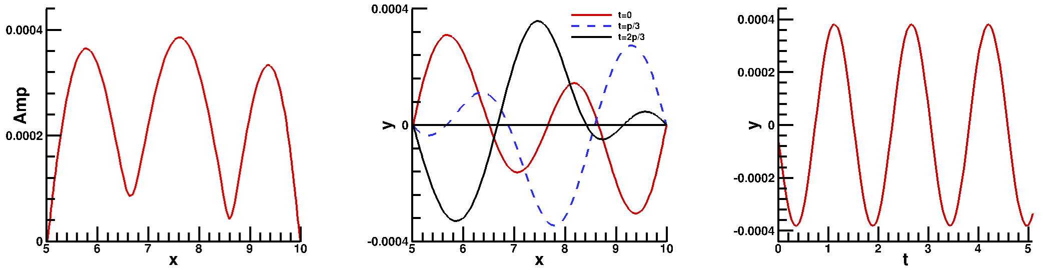
(a) amplitude (b) beamshape (c) y-t at x=7.5

Figure 4-18: N2-11: $Re = 380$, $C_\lambda = 397.5$, $\omega_r = -3.62141 \times 10^{-6}$, $\omega_i = 1.24308$, Mode-2



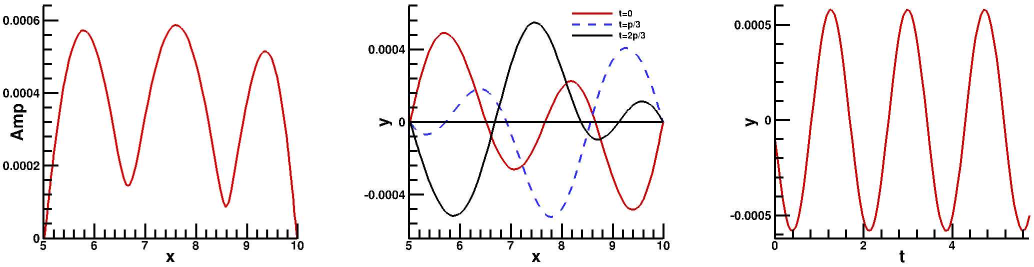
(a) amplitude (b) beamshape (c) y-t at x=7.5

Figure 4-19: N2-12: $Re = 440$, $C_\lambda = 369$, $\omega_r = 9.00410 \times 10^{-6}$, $\omega_i = 1.20026$, Mode-2



(a) amplitude (b) beamshape (c) y-t at x=7.5

Figure 4-20: N3-1: $Re = 300$, $C_\lambda = 335.61$, $\omega_r = -5.67991 \times 10^{-6}$, $\omega_i = 4.07781$, Mode-3

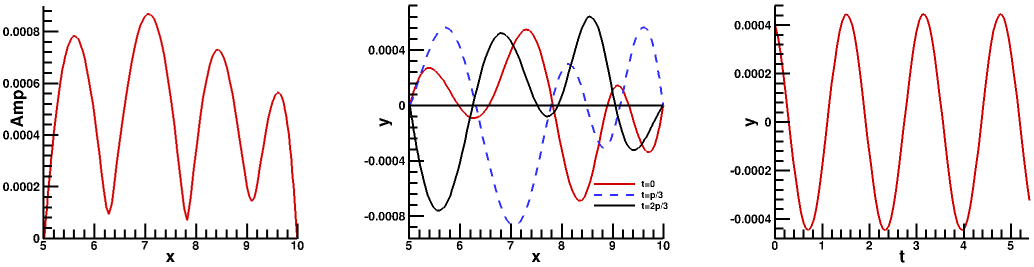


(a) amplitude

(b) beamshape

(c) y-t at x=7.5

Figure 4-21: N3-2: $Re = 250$, $C_\lambda = 311$, $\omega_r = 7.02530 \times 10^{-5}$, $\omega_i = 3.61682$, Mode-3



(a) amplitude

(b) beamshape

(c) y-t at x=7.5

Figure 4-22: N4-1: $Re = 231.15$, $C_\lambda = 60$, $\omega_r = 9.69742 \times 10^{-6}$, $\omega_i = 3.85157$, Mode-4

The sub-figure(a) in the pictures above illustrate the amplitude of each node along the beam. The Mode number depends on the number of peaks in the amplitude plot. Among all the Mode-2 neutral points, the amplitude at N2-8 is larger than the other operating points, so the largest amplitude might happen in the left bound, but not at a small C_λ point or a high Reynolds number. From the dimensionless Reynolds number $Re = \dfrac{\rho U_0 D}{\mu}$, we know that Re depends on the velocity U_0. Because the fluid is incompressible, it is imaginable that the Mode-2 neutral point with the largest amplitude might be at the left bottom of the Mode-2 neutral curve. And a high C_λ means that the beam is too rigid to deform. So the point with the largest amplitude must be at some point down the left bound. Compared N2-2 with N2-3 and N2-5 with N2-7, the same conclusion can be obtained that at a same C_λ, the oscillatory amplitude of a small Re is larger than that of a high Re. Compared N2-4 with N2-10, with the same Re, the largest amplitude takes place at small C_λ, where the beam is in a very soft condition. The sub-figure(b) indicates the beamshape(no scalling) at different time. And the sub-figure(c) shows that the oscillation of the middle point on the beam.

Pressure-driven system

With the AR-F eigensolver, the Mode-1 neutral curve in pressure-driven system is also revised and extended. The full information of the neutral points is in the following table.

Table 4.7: The mode-1 neutral curve of the pressure-driven system, obtained for $P_e - P_d = 1.95$. The neutral points below N1-2 were initially obtained by Liu et al. [80] using the QZ solver, which are identical to the results of the AR-F solver. The points between N1-1 and N1-2 are obtained using the AR-F solver, which agree well with those obtained by Liu et al. using an unsteady simulation, see Table 2. The eight points above N1-1 are the new points obtained using the AR-F solver for a denser mesh.

Pud	C_λ	Re	ω_r	ω_i
1.263	3.0×10^7	237.02	-2.17656×10^{-5}	3.96708
1.2895	3.5×10^7	241.64	$+3.59042 \times 10^{-6}$	4.14419
1.3152	4.0×10^7	253.02	-2.73466×10^{-6}	4.32351
1.4127	6.0×10^7	264.91	-5.71267×10^{-6}	5.03695
1.4989	8.0×10^7	281.09	$+6.63375 \times 10^{-6}$	5.71320
1.6083	1.1×10^8	301.79	-2.08839×10^{-6}	6.63706
1.72775	1.5×10^8	322.68	$+4.88966 \times 10^{-7}$	7.72416
1.849	2.0×10^8	347.15	$+4.53263 \times 10^{-6}$	8.91252

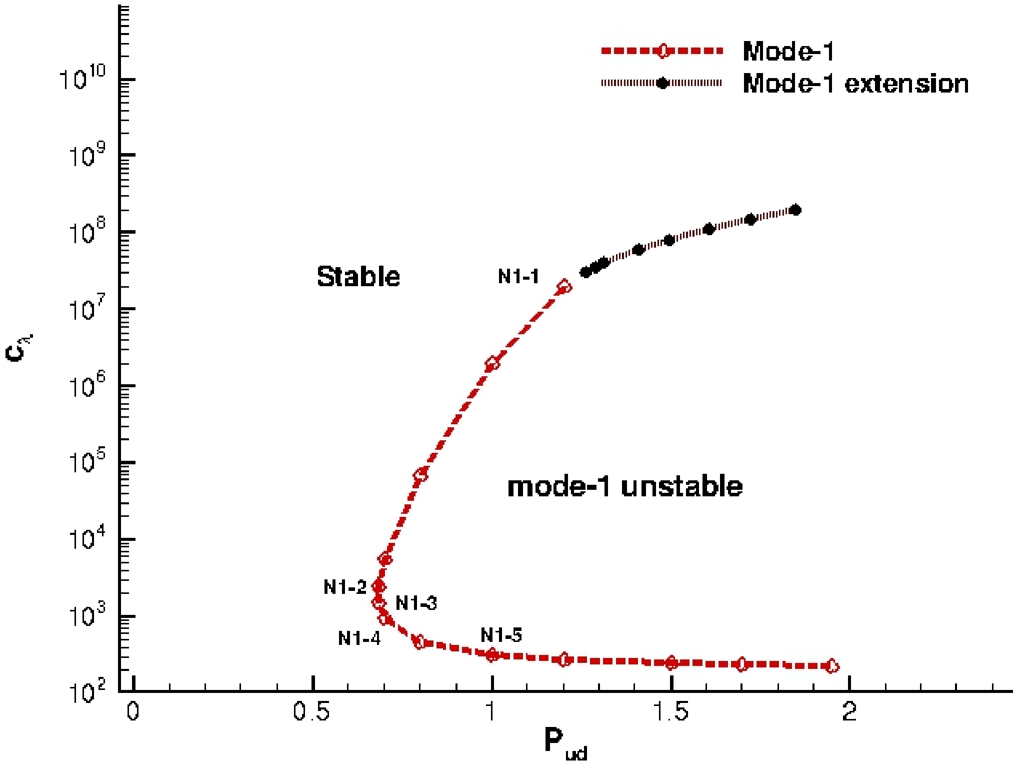
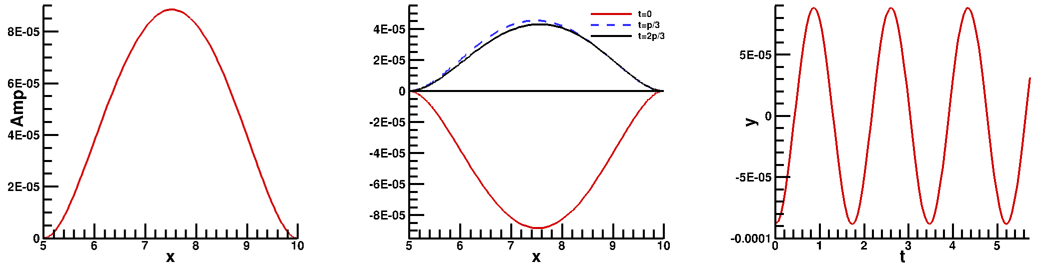
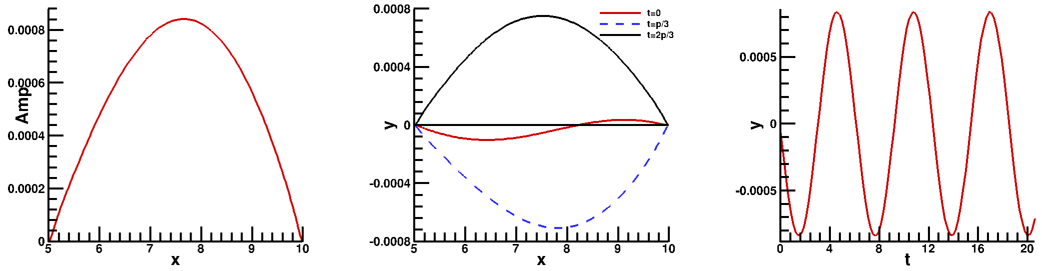


Figure 4-23: The mode-1 neutral curve of the pressure-driven system, obtained for $P_e - P_d = 1.95$. The neutral points below N1-2 were initially obtained by Liu et al. using the QZ solver, which are identical to the results of the AR-F solver. The points between N1-1 and N1-2 are obtained using the AR-F solver, which agree well with those obtained by Liu et al. using an unsteady solver, see Table 2. The eight points above N1-1 are the new points obtained using the AR-F solver for a mesh denser mesh.



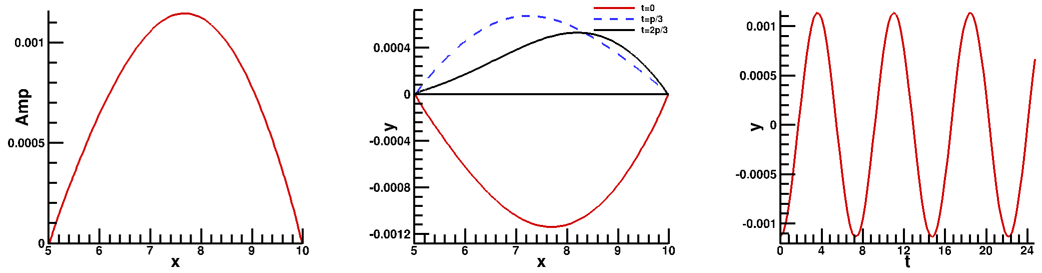
(a) ampl (b) beamshape (c) yt

Figure 4-24: N1-1: $Re = 226.62$, $Pud = 1.21$, $C_\lambda = 2\times 10^7$, $\omega_r = 7.08475\times 10^{-6}$, $\omega_i = 3.61910$, Mode-1



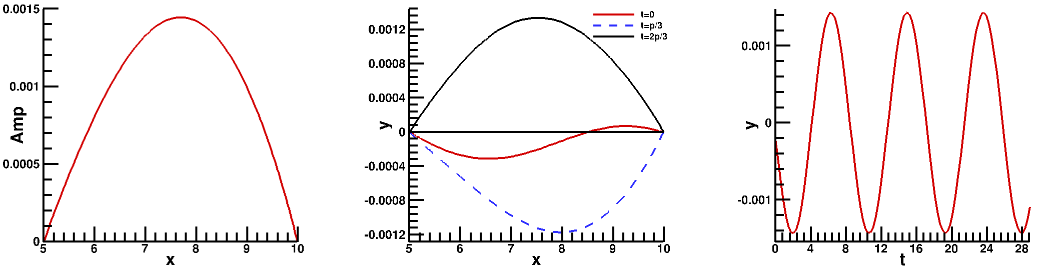
(a) ampl (b) beamshape (c) yt

Figure 4-25: N1-2: $Re = 122.16$, $Pud = 0.70815$, $C_\lambda = 5500$, $\omega_r = 4.59145\times 10^{-6}$, $\omega_i = 1.01443$, Mode-1



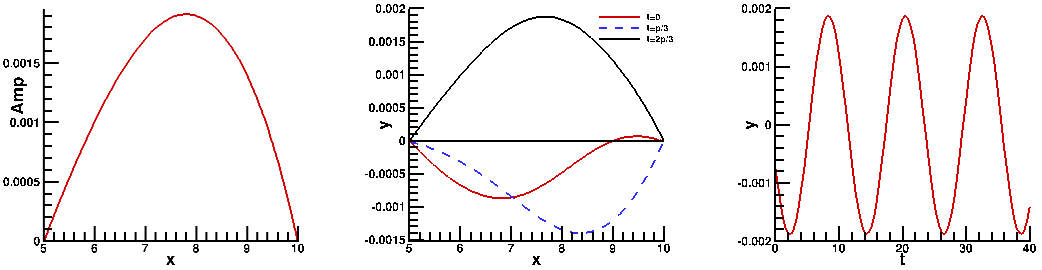
(a) ampl (b) beamshape (c) yt

Figure 4-26: N1-3: $Re = 113.83$, $Pud = 0.6872$, $C_\lambda = 2000$, $\omega_r = -3.31709 \times 10^{-6}$, $\omega_i = 0.84649$, Mode-1



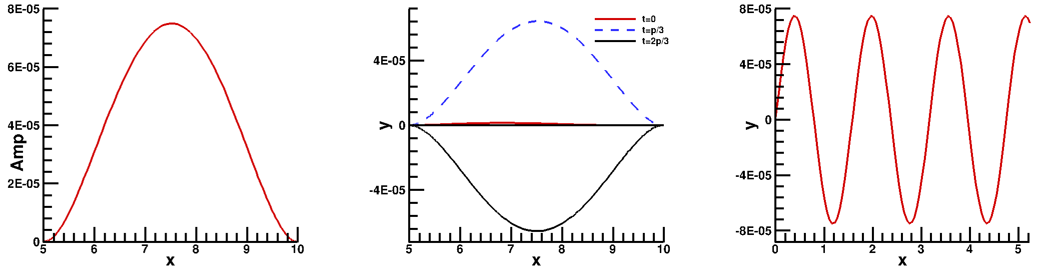
(a) ampl (b) beamshape (c) yt

Figure 4-27: N1-4: $Re = 107.89$, $Pud = 0.7031$, $C_\lambda = 927$, $\omega_r = -1.60704\times10^{-5}$, $\omega_i = 0.72597$, Mode-1



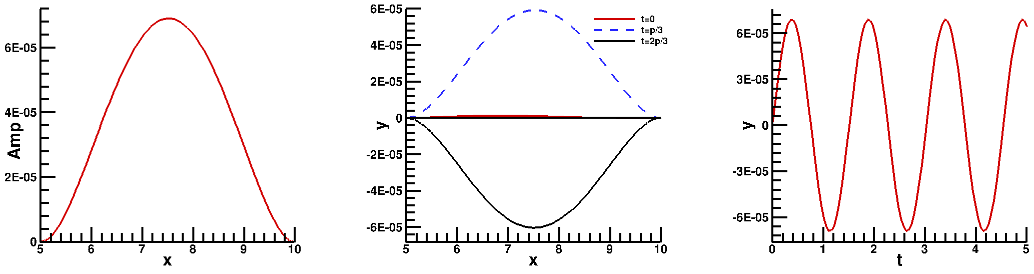
(a) ampl (b) beamshape (c) yt

Figure 4-28: N1-5: $Re = 115.61$, $Pud = 1.0$, $C_\lambda = 308$, $\omega_r = -9.00310 \times 10^{-6}$, $\omega_i = 0.51888$, Mode-1



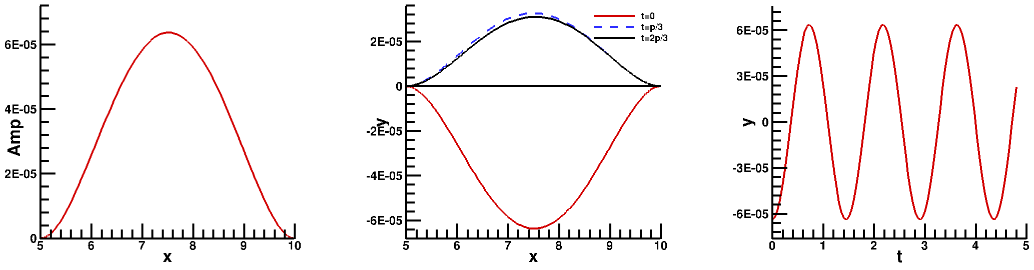
(a) ampl (b) beamshape (c) y-t at x=7.5

Figure 4-29: $Re = 237.02$, $Pud = 1.263$, $C_\lambda = 3 \times 10^7$, $\omega_r = -2.17656 \times 10^{-5}$, $\omega_i = 3.96708$, Mode-1



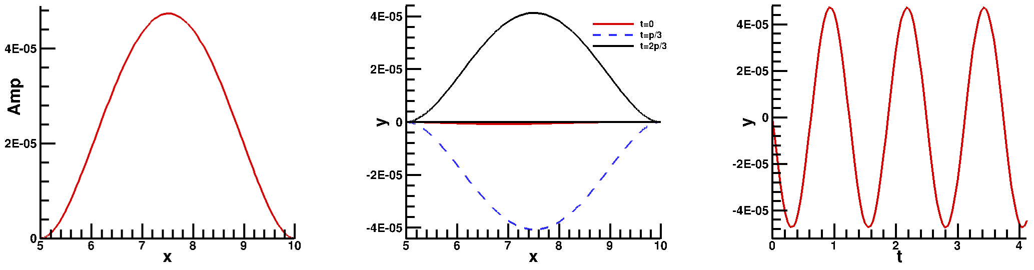
(a) ampl (b) beamshape (c) y-t at x=7.5

Figure 4-30: $Re = 241.64$, $Pud = 1.2895$, $C_\lambda = 3.5 \times 10^7$, $\omega_r = 3.5902 \times 10^{-6}$, $\omega_i = 4.14419$, Mode-1



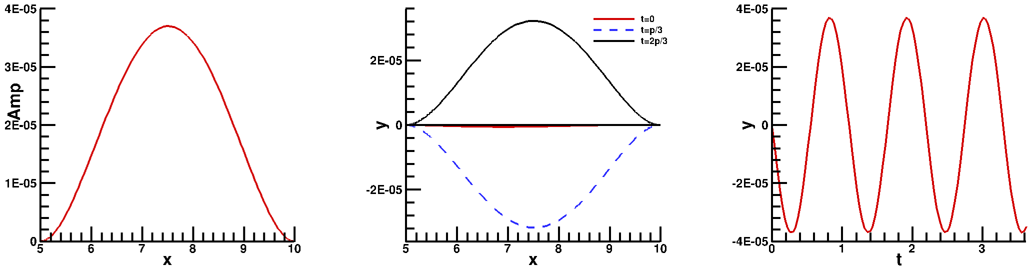
(a) ampl (b) beamshape (c) y-t at x=7.5

Figure 4-31: $Re = 253.02$, $Pud = 1.3152$, $C_\lambda = 4 \times 10^7$, $\omega_r = -2.73466 \times 10^{-6}$, $\omega_i = 4.32351$, Mode-1



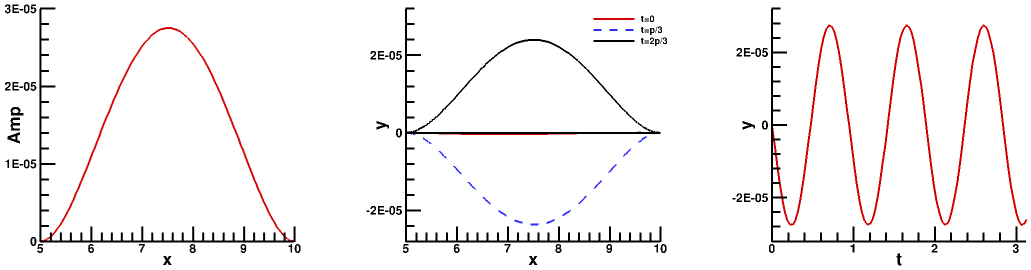
(a) ampl (b) beamshape (c) y-t at x=7.5

Figure 4-32: $Re = 264.91$, $Pud = 1.4127$, $C_\lambda = 6 \times 10^7$, $\omega_r = -5.71267 \times 10^{-6}$, $\omega_i = 5.03695$, Mode-1



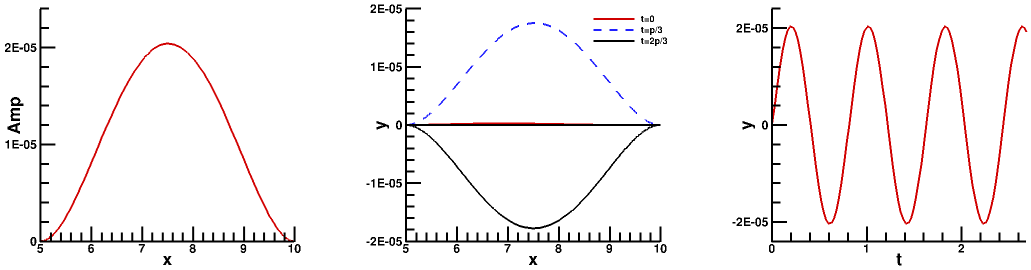
(a) ampl (b) beamshape (c) y-t at x=7.5

Figure 4-33: $Re = 281.09$, $Pud = 1.4989$, $C_\lambda = 8 \times 10^7$, $\omega_r = 6.63375 \times 10^{-6}$, $\omega_i = 5.71320$, Mode-1



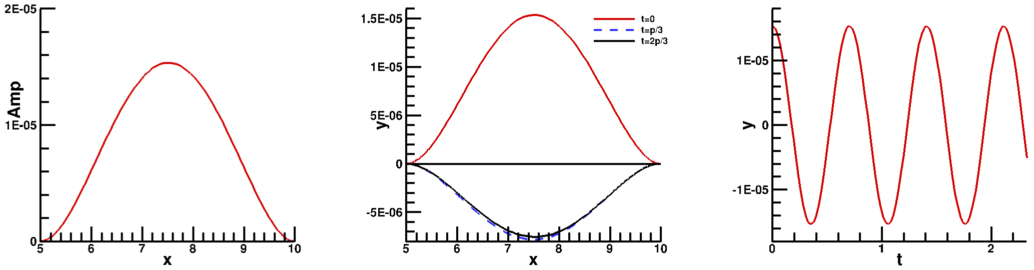
(a) ampl (b) beamshape (c) y-t at x=7.5

Figure 4-34: $Re = 301.79$, $Pud = 1.6083$, $C_\lambda = 1.1 \times 10^8$, $\omega_r = -2.08839 \times 10^{-6}$, $\omega_i = 6.63706$, Mode-1



(a) ampl (b) beamshape (c) y-t at x=7.5

Figure 4-35: $Re = 322.68$, $Pud = 1.72775$, $C_\lambda = 1.5 \times 10^8$, $\omega_r = 4.88966 \times 10^{-7}$, $\omega_i = 7.72416$, Mode-1



(a) ampl

(b) beamshape

(c) y-t at x=7.5

Figure 4-36: $Re = 347.15$, $Pud = 1.849$, $C_\lambda = 2 \times 10^8$, $\omega_r = 4.53263 \times 10^{-6}$, $\omega_i = 8.91252$, Mode-1

The plots of eigenmodes above the N1-1 point are shown in the pictures of past pages. Because the wall stiffness is so huge that the amplitude and the velocity in Y-direction are sufficiently small, there is no significant difference in the shape. And the y-t plots shows the oscillations along with time.

4.5 Conclusion & Further study

4.5.1 Conclusion

With the methodologies in the previous chapter, the new two eigensolvers for collapsible channel flow in FBM are introduced, one is AR-G, and the other is AR-F. The difference between these two eigensolvers is that the global matrices are used in the spectral tranformation(shift-and invert transformation) stage, while in the latter solver a frontal solver is employed instead of the traditional inversion. Hence, the AR-G is the intermediate product procedure of the final efficient eigensolver AR-F, however, it is still used for some testing work now because of its merit, and if we decide to improve the solver with some new software, the global matrices would definitely play its significant role in it.

After derivating the two eigensolvers, a coarse mesh($8 \times (10 + 20 + 10)$) and a much more refined mesh($16 \times (30 + 60 + 60)$ [85]) are used to validate the new solvers with QZ algorithm, because in these size eigenvalue problems, QZ algorithm can still work well. During the validating process, it is found that the computational times of the three solvers are sufficiently different, so a loglog plot of the elapsed times and the dimensional sizes of matrices at a same group of parameters is shown. From that picture, it is clear that AR-F can not only save the computational time, but also drop the increasing rate along the matrix size from n^3 to about n^2, where n is the order of matrix. Especially, AR-F can solve very large eigenvalue problems which QZ algorithm feels helpless to deal with.

With the powerful AR-F solver, the neutral curves of the two specified boundary conditions are revised and extended with a very refined mesh(the number of degree of freedom is more than 50000, which is dramatically large than those in previous studies). Compared with the neutral curve [85] in flow-driven system, the new one keeps the cascade structure as before and the left bound of the curve is still at about $Re = 210$, which means that the viscosity of the fluid is strong enough to damp any disturbance out. In the pressure-driven system, AR-F makes the neutral curve much more accurate than the old one, and extends the curve to very high C_λ zone, where it is hard to locate a point with the transient analysis(unsteady simulation [87, 80]).

4.5.2 Further study

Generalized Cayley transformation

The Generalized Cayley transformation is another kind of spectral transformation which has been used for solving eigenvalue problem [74]. However, it is a little more complicated than the shift-and-invert transformation, and there is very few software package to implement this transformation. The generalized Cayley transformation has been tested to solve the eigenproblems of collapsible channel flows for several times, but its result is not satisfied, so there still needs some research work.

Parallel ARPACK

ARPACK software package also provides the the codes of parallel computation. After the success of AR-F eigensolver, the parallel computation is not far way, and it would improve the eigensolver in collapsible channel flows again.

The eigenvalues with the second or the third largest real parts

In the research of collapsible channel flow, at the beginning of the oscillation, a great many of modes are coupled together, and along with the time, the stable modes would disappear, the unstable modes would increase and the neutral mode would oscillate. So sometimes we would also study the eigenvalues with the second or the third largest real parts. For example, in the neutral curve of the pressure-driven system [80], Liu et al. presented a Mode-2 neutral curve in the Mode-1 unstable zone.

During the testing and validating work of the new two Arnoldi solvers, we also did some research work with the eigenvalues with the second or the third largest real parts. In the Table 4.4, it is known that the eigenvalues with the second and the third largest real parts can be easily worked out with $nev = 4$ or $nev = 6$, and the ncv is not very large. The reason is that the separation and the centrality of the eigenvalues are so fine that it is easy to calculate them out. For the other meshes, we have tried to change the four parameters(*nev*, *ncv*, *tol*, *maxitr*) to search the eigenvalues with the second and the third real parts. Taking 933 × 933 for example, if we choose this group of parameters $nev = 70$, $ncv = 110$, $tol = 1.0 \times 10^{-12}$ and $maxitr = 100$, the eigenvalues with the second or the third real parts can be worked out correctly with a return message " *maximum iteration number reached* " in ARPACK. In this case, it would take a long time to find the eigenvalues with the second and the third largest real parts. The efficiency of the new eigensolvers would decrease. So we also try to orthogonalize the component of the eigenvalues with the largest real parts off from the original matrices.

Take a standard eigenvalue problem $\mathbf{Ax} = \lambda\mathbf{x}$, where $\mathbf{A}$ is a real matrix. And $(\lambda_i, \mathbf{v}_i)$ is a group of real eigenpairs which has been worked out by AR-F eigensolver. Then is used to get rid of the component of $(\lambda_i, \mathbf{v}_i)$ from $\mathbf{A}$.

$$\mathbf{B} = \mathbf{A}(\mathbf{I} - \sum_{i=1}^{n} \mathbf{v}_i \mathbf{v}_i^T) \tag{4.20}$$

$$\mathbf{v}^+ = (\mathbf{I} - \sum_{i=1}^{n} \mathbf{v}_i \mathbf{v}_i^T)\mathbf{v}_s \tag{4.21}$$

If $(\lambda_i, \mathbf{v}_i)$ is complex, after involving the left eigenpairs $(\lambda_i, \mathbf{w}_i)$, the formula is

$$\mathbf{B} = \mathbf{A} - 2\sum_{i=1}^{n} Real\{\lambda_i \mathbf{v}_i \mathbf{w}_i^\dagger\} \tag{4.22}$$

$$\mathbf{v}^+ = (\mathbf{I} - \sum_{i=1}^{n} \mathbf{v}_i \mathbf{w}_i^\dagger)(\mathbf{I} - \sum_{i=1}^{n} \mathbf{v}_i^T \mathbf{w}_i^\dagger)\mathbf{v}_s, \tag{4.23}$$

where "T" is transpose, "†" is transpose and conjugate, $\mathbf{v}^+$ is the new starting vector, which has got rid of the component of $(\lambda_i, \mathbf{v}_i)$, $\mathbf{v}_s$ is the starting vector.

Using the equations above, we tried our orthogonality idea with some computation in both MATLAB and ARPACK. This idea is successful for a 5 × 5 random matrix and a diagonal matrix in both MATLAB and ARPACK, but when we try it with the matrix obtained from the eigenvalue problem of collapsible channel flow, it failed to find any eigenvalues after orthogonalization with both MATLAB and ARPACK. We guess the reason of failure is that the separation of the eigenvalues in ours is not fine enough to implement this idea. In future, we will try some pre-conditioning work or something else to make our eigensolver much more efficient than now.

Chapter 5

Energy analysis with eigenpairs

Following Pedley's theory [113], Stewart et al. [141] presented the energy analysis on 1D model with pressure boundary condition, where the pressure is specified across the channel. Liu et al. [80] introduce and analyze the stability and energy budget of pressure-driven collapsible channel flows in the Fluid-Beam Model, which was constructed by Cai et al. [15]. They obtained all the energy resutls with the transient analysis [86, 87]. Luo et al. presented a much faster, more efficient and more accurate eigenvalue problem [85] for locating neutral points of the linear stability problems. From the Chapter 3, we know that the complex eigenvalues indicate the growth rates and frequencies of the self-excited oscillations, and the complex eigenvectors indicate the magnitudes and the phase angles, which can be considered as the increment of each degree of freedom. As the eigenvalue problem is much more efficient than the transient analysis, it is instructive to carry out the energy analysis combined with the eigenvalue problems.

5.1 Approaches to energy solution with eigenpairs

On the basis of momentum equation of fluid

$$\frac{\partial u_i}{\partial t} + u_j u_{i,j} = \sigma_{ij,j}, \tag{5.1}$$

via multiplying velocity u_i on both sides

$$u_i \frac{\partial u_i}{\partial t} + u_i u_j u_{i,j} = u_i \sigma_{ij,j}, \tag{5.2}$$

and integrating over the whole domain

$$\int_\Omega u_i \frac{\partial u_i}{\partial t} d\Omega + \int_\Omega u_i u_j u_{i,j} d\Omega = \int_\Omega u_i \sigma_{ij,j} d\Omega, \tag{5.3}$$

The integrasl of the energies lead to

$$\int_{\Omega} \dot{e} d\Omega + \frac{1}{2} \int_{\Gamma} q^2 u_j n_j d\Gamma = \int_{\Gamma} \sigma_{ij} n_j u_i d\Gamma - \int_{\Omega} \sigma_{ij} u_{i,j} d\Omega, \tag{5.4}$$

and we write the equation of energy for simple as

$$\dot{E} - F = P - D \tag{5.5}$$

where $\dot{E}$ indicates the rate of change of fluid kinetic energy; $-F$ represents the rate of kinetic energy fluxing out of the channel; P stands for the rate of work done by the external forces to the fluids, and $-D$ is the rate of viscous dissipation of kinetic energy.

Due to the adoption of Arbitrary Lagrangian Eulerian(**ALE**) formulation, the first term $\dot{E}$ in (5.5) can be explained as

$$\dot{E} = \int_{\Omega} \frac{\partial e}{\partial t} d\Omega = \int_{\Omega} \left(\frac{\delta e}{\delta t} - \dot{x_j} e_{,j} \right) d\Omega, \tag{5.6}$$

where $\dot{e} = \dot{\left(\frac{1}{2} q^2\right)} = \frac{\delta e}{\delta t}$ and $q^2 = \sum u_i^2$, $\frac{\delta}{\delta t}$ is the time derivative following a moving node, which is different to $\frac{\delta}{\delta t}$ expressed in the Eulerian frame of reference.

The second term F can be expanded further as:

$$F = \frac{1}{2} \int_0^1 q^2 u|_{inlet} dy - \frac{1}{2} \int_0^1 q^2 u|_{outlet} dy - \frac{1}{2} \int_0^{S_0} q^2 u_n ds, \tag{5.7}$$

where $q^2 = \sum u_i^2$, $u|_{inlet}$ is the fluid velocity at the inlet, $u|_{outlet}$ is the fluid velocity at the outlet, u_n is the beam velocity in the normal direction, and S_0 is the arclength of the beam after deformation.

The third term P is the integral of

$$P = \int_0^1 P_u u dy - \int_0^{S_0} P_e u_n ds - \frac{1}{2} \int_0^L C_\lambda (\lambda - 1)^2 dl - \frac{1}{2} \int_0^{S_0} C_\kappa \kappa^2 ds \tag{5.8}$$

For simplicity, we rewrite (5.8) into $P = P1 - P3 - dU_\lambda - dU_\kappa$, where $P1$ is the rate of work done by upstream pressure, $P3$ is the rate of work done by the beam to resist the external pressure, dU_λ is the rate of stretching strain energy in the beam, dU_κ is the rate of bending strain energy in the beam, respectively.

The last term $-D$ is the rate of viscous dissipation of kinetic energy, and its integration is

$$D = \int_{\Omega} u_{i,j} d\Omega, \tag{5.9}$$

and D can be divided into two parts: D_p, is the rate of viscous dissipation due to the steady motion, and D_s is the rate of viscous dissipation due to oscillatory motion. D_p can be obtained by substituting the steady solution $\bar{\mathbf{U}}$ into (5.9). So

$$D = D_p + D_s \tag{5.10}$$

Hence, by moving the negative terms to the other sides to get rid of the sign of subtraction, the final energy budget equation of the system is :

$$P1 + F = \dot{E} + D + P3 + \dot{U_\lambda} + \dot{U_\kappa} \tag{5.11}$$

Hereby the equations of energy analysis have been reviewed and all the terms are much clearer than before, which are summarized in the following table.

Table 5.1: Definition of the energy analysis

$\dot{E}$	The rate of change of kinetic energy.
$P1$	The rate of work of upstream pressure.
F	The rate of net kinetic energy flux into the system.
D	The rate of viscous dissipation.
D_p	The rate of viscous dissipation due to steady motion.
D_s	The rate of viscous dissipation due to oscillatory motion.
P_3	The rate of word done by the external pressure to the beam.
dU_λ	The rate of stretching strain energy in the beam.
dU_κ	The rate of bending strain energy in the beam.

and now we are going to find the oscillatory solutions from the eigenpairs.

As we know, we have denoted the infinitesimal perturbation to the steady solution $\bar{\mathbf{U}}$ by $\Delta\mathbf{U}$, so that $\mathbf{U} = \bar{\mathbf{U}} + \Delta\mathbf{U}$ is the solution to current system. $\Delta\mathbf{U}$ has been written in the exponential form of $e^{\omega t}\tilde{\mathbf{U}}$, where $\omega(= \omega_r + i\omega_i)$, $\tilde{\mathbf{U}}$ are the complex eigenvalues and eigenvectors solved from the eigenvalue equation (3.10), respectively. The eigenvalue results are presented through

$$\begin{aligned} \Delta\mathbf{U} &= Real\left[e^{(\omega_r + i\omega_i)t}\left(\tilde{\mathbf{U}}_r + i\tilde{\mathbf{U}}_i\right)\right] \\ &= e^{\omega_r t}\left(\tilde{\mathbf{U}}_r \cos(\omega_i t) - \tilde{\mathbf{U}}_i \sin(\omega_i t)\right) \\ &= e^{\omega_r t}\|\tilde{\mathbf{U}}\| \cos(\omega_i t + \phi), \end{aligned} \tag{5.12}$$

where $\|\tilde{\mathbf{U}}\| = \sqrt{\tilde{\mathbf{U}}_r^2 + \tilde{\mathbf{U}}_i^2}$ is the amplitude and $\phi = \arctan \tilde{\mathbf{U}}_i/\tilde{\mathbf{U}}_r$ is the initial phase. Consequently, the current solution **U** can be obtained by distributing its increment to each degree of freedom easily and precisely. Afterwards, by substituting the current solution $\bar{\mathbf{U}}$ into (5.11), the results of energy analysis using eigenpairs can be obtained.

Advantages of the energy approaches using eigenpairs

Compared with the energy analysis of Liu et al. [80], the energy analysis has its own advantages.

1. Time saving: The energy analysis using eigenpairs is much, much faster than energy analysis using unsteady simulation. Liu et al. [80] finish the instability and the energy analysis via solving the linear equation groups by a frontal solver [45, 47, 87, 116], and the whole elapsed time of the unsteady simulation is more than 3 days. However, the energy analysis using eigenpairs only takes a couple of minutes. Obviously, the energy analysis using eigenpairs is better than the unsteady one.

2. Close to neutral points: We have discussed that if the initial solution of the unsteady instability is so larger that the neutral oscillation might be aberrant and might not describe the correct neutral point which we wanted properly. It is hard to guarantee that the neutral point is really neutral as demanded when the perturbation is large. The perturbation adding to steady solution $\bar{\mathbf{U}}$ is infinitesimal, so the eigensolution can show the current neutral point fully.

3. Unique mode: The neutral point means that the beam would oscillate with the same amplitude and frequency along time, and the oscillation would not change and would be kept as the time passing. The unsteady simulation might include some other modes in it, which might take a very long time to disappear.

4. Any point: The eigenvalue energy analysis can be applied to any point, no matter stable or unstable.

5. Separation: Through some derivation, it is very easy to separate the oscillatory energy from the energy of steady flow. The derivations of all the integrating equations are in section 5.4.

6. Accuracy: The imaginary part of the eigenvalue is the frequency of the oscillation, with which the period of the oscillation can be solved by $2\pi/\omega_i$. The increments obtained from eigenpairs are much accurate than the unsteady simulation, so the energy analysis must be better than unsteady simulation.

5.2 Validation of the energy analysis in pressure-driven system

Liu et al. [80] presented the stability problem of pressure-driven system and the corresponding energy analysis to the neutral points. After making the codes of energy from eigenpairs work well, it is necessary to validate the energy analysis of eigenpairs with the energy analysis of transient analysis.

Note that Liu et al. [80] have published five Mode-1 neutral points and the corresponding energy results under the pressure-driven boundary conditions, and these points are much more neutral than before because AR-F has been validated as the fastest solver. Now we are going to validate the energy results obtained from eigenpairs with these results.

During the research of energy analysis with the unsteady simulation, it is found that the different initial solution which is the perturbation to the steady solution would lead to different energy solution. In the other words, the energy results are sensitive to the size of perturbation. So, firstly we are going to figure out the effect of the initial solutions. The neutral point used to test is $Pud = 1.31$, $C_\lambda = 3.5 \times 10^7$. Pud is fixed to 1.31, then C_λ is changed from 1.0×10^7 to 5.0×10^7, especially the values near $C_\lambda = 3.5 \times 10^7$. All the perturbations tested are listed in Table 5.2, and the percentage of perturbation as well.

Table 5.2: The lists of perturbations to a neutral point($Pud = 1.31$, $C_\lambda = 3.5 \times 10^7$)

C_λ	*perturbation*	*percentage*
1.0×10^7	-2.5×10^7	-71.49%
2.0×10^7	-1.5×10^7	-42.88%
3.0×10^7	-5.0×10^6	-14.29%
3.6×10^7	$+1.0 \times 10^6$	+2.88%
3.7×10^7	$+2.0 \times 10^6$	+5.71%
3.8×10^7	$+3.0 \times 10^6$	+8.57%
3.9×10^7	$+4.0 \times 10^6$	+11.43%
4.0×10^7	$+5.0 \times 10^6$	+14.29%
5.0×10^7	$+1.5 \times 10^7$	+42.88%

From the Figure 5-1 and Figure 5-2, it is clear that the positive perturbation and negative perturbation with the same value do not have the same energy results(2.0×10^7 and 5.0×10^7) when the perturbation is a little large, but when the perturbations become small(3.0×10^7 and 4.0×10^7), the results are much closer than large perturbations. This implies that if the perturbation is sufficiently small, for example, the infinitesimal perturbation in the eigenvalue problems, the results would be the same. When the perturbation is from large to small(5.0×10^7 to 3.6×10^7, +42.88% to +2.88%), the energy results are also from large to small. The size of the circle in Figure 5-1 and Figure 5-2 prove this quite well. When the perturbation is 1.0×10^7(-71.49%), a cross part occurs. But we do not know whether this result is reliable or not, because if the perturbation is so large that the oscillation would approach another neutral point and the velocity curve of high C_λ has a great number of "zigzag". On the contrary, we can infer that the energy results obtained from eigenpairs might be reliable, and this is one of the advantages which has been discussed in previous section.

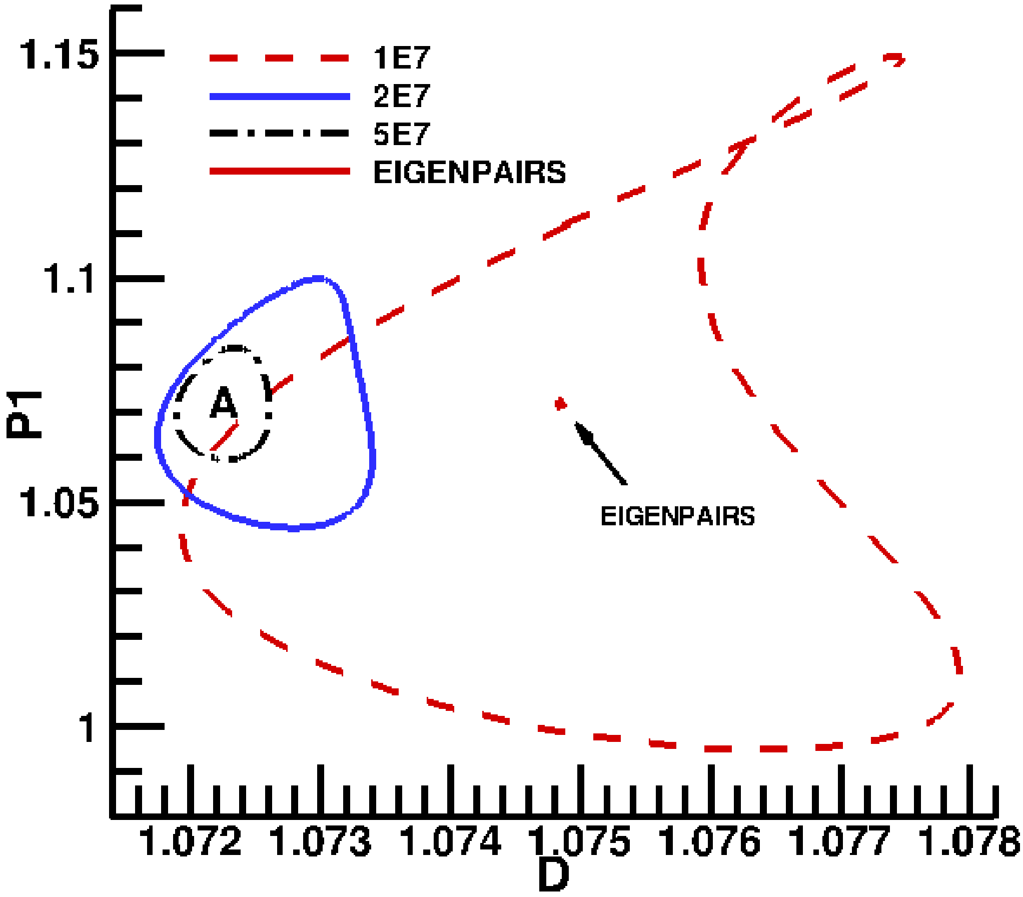


Figure 5-1: $P1-D$ plot of different perturbations to a neutral point ($Pud = 1.31$, $C_\lambda = 3.0\times10^7$)

The red solid line in Figure 5-1 is the energy solution obtained from the eigenpairs, which is the red dot in the centre of the picture because of the infinitesimal perturbation in the problem of eigenvalue. The red dashed line is from the unsteady simulation with a perturbation $C_\lambda = 1 \times 10^7$(-71.49%), and there is a small loop at the top, which agrees with Liu et al. [80]. The blue solid line indicates the energy solution with an initial guess as $C_\lambda = 2 \times 10^7$(-42.88% perturbation to the steady solution). And the black dash-dot line comes from the energy solution of an initial solution $C_\lambda = 5 \times 10^7$(+42.88% perturbation). From this picture, it is clear that the circle of $C_\lambda = 5 \times 10^7$ is inside that of $C_\lambda = 2 \times 10^7$, which shows that a large perturbation would lead to a group of large energy solutions. The dissipation increase along with the rise of perturbing percentage. The energy results of some other perturbations is in the zone of A, which are enlarged in the next picture.

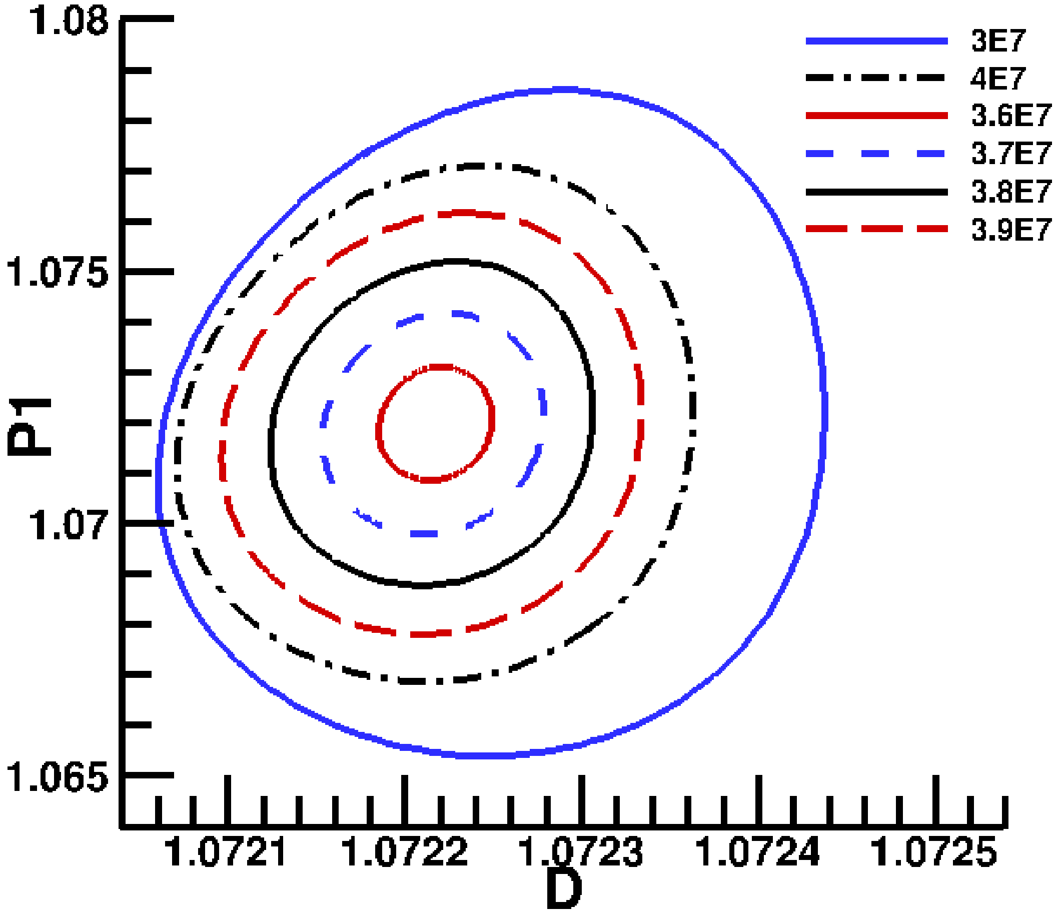
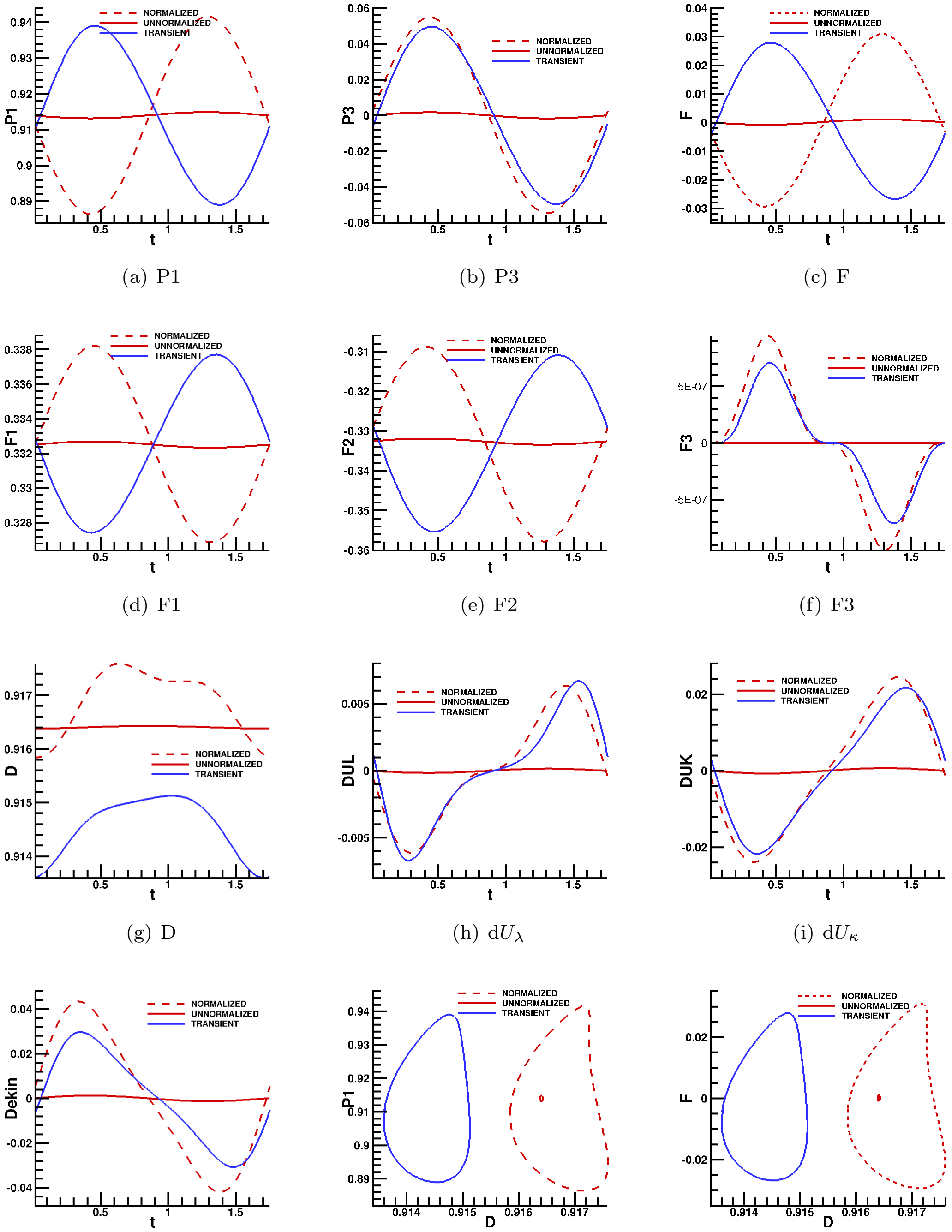


Figure 5-2: The enlargement of zone A in Figure 5-1 of $P1 - D$ plot

In the Figure 5-2, the zone A in the Figure 5-1 is enlarged here. From the outside to the inside, the circles indicates the energy solutions with different initial guesses(from $C_\lambda = 3.6 \times 10^7$(+2.88%) to $C_\lambda = 3 \times 10^7$(-14.29%)). Considering the perturbations of $C_\lambda = 4 \times 10^7$(+14.29%), $C_\lambda = 3.9 \times 10^7$(+11.43%), $C_\lambda = 3.8 \times 10^7$(+8.57%), $C_\lambda = 3.7 \times 10^7$(+5.71%), $C_\lambda = 3.6 \times 10^7$(+2.88%), it is clear that the energy solutions decrease with the drop of perturbation. Then compared the energy solutions of $C_\lambda = 4 \times 10^7$(+14.29%) and $C_\lambda = 3 \times 10^7$(-14.29%), these absolute values of perturbing percentage are same as 14.29%, but the energy solutions are a bit different from each other. The energy solution from a large C_λ is a little smaller than a small C_λ. Finally, the effect of the C_λ perturbing percentage are concluded that a large perturbation leads to a large energy solution, and at the same percentage of perturbation, the low C_λ will obtain a larger energy solution than the high C_λ.



(a) P1

(b) P3

(c) F

(d) F1

(e) F2

(f) F3

(g) D

(h) dU_λ

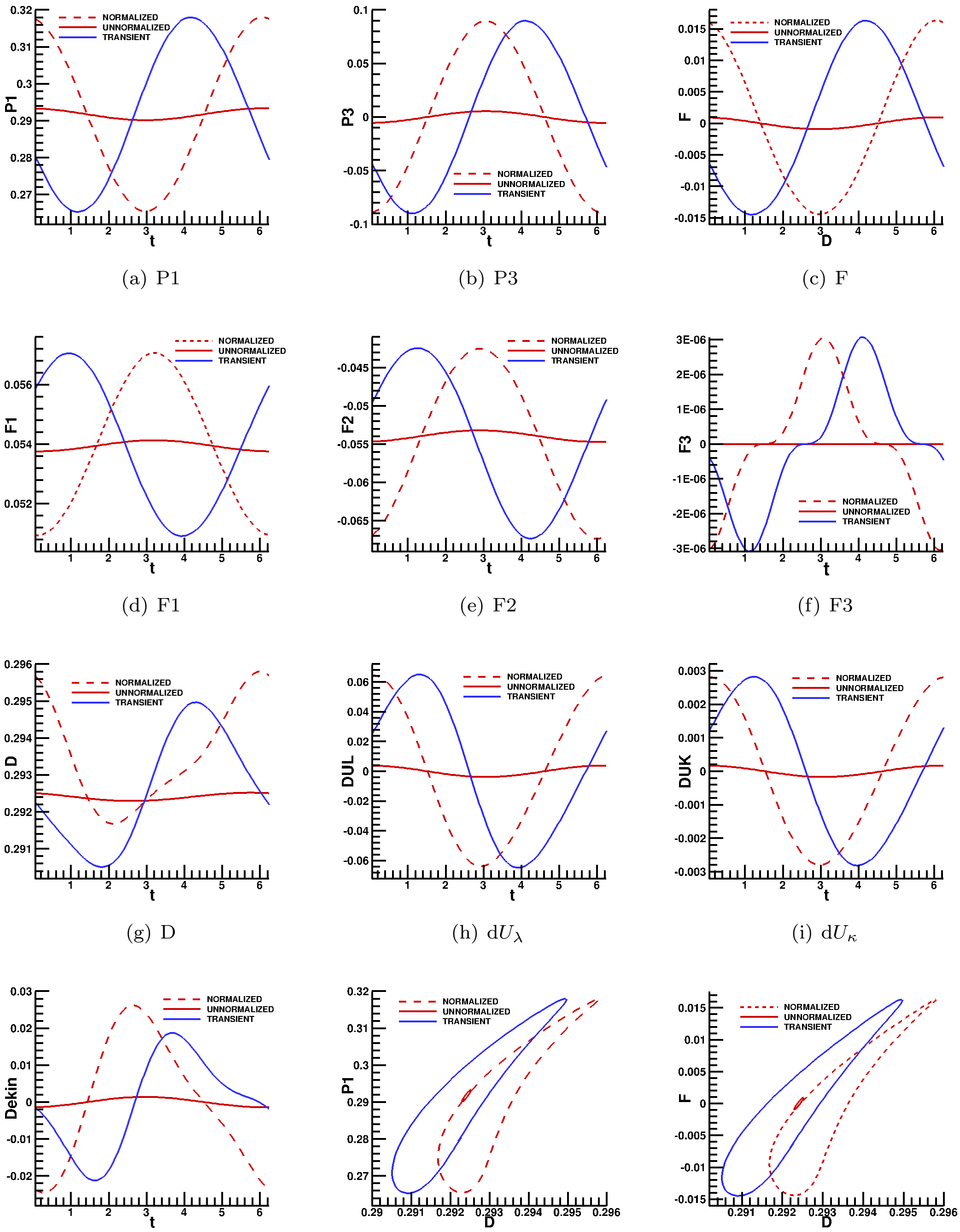
(i) dU_κ

(j) $\dot{E}$

(k) P1-D

(l) F-D

Figure 5-3: N1-1: $Pud = 1.21$, $C_\lambda = 2.0 \times 10^7$, the blue solid curve is from unsteady simulation [80], the red solid curve is from energy solution from eigenpairs, and the dashed curve is the normalized energy solution from eigenpairs.



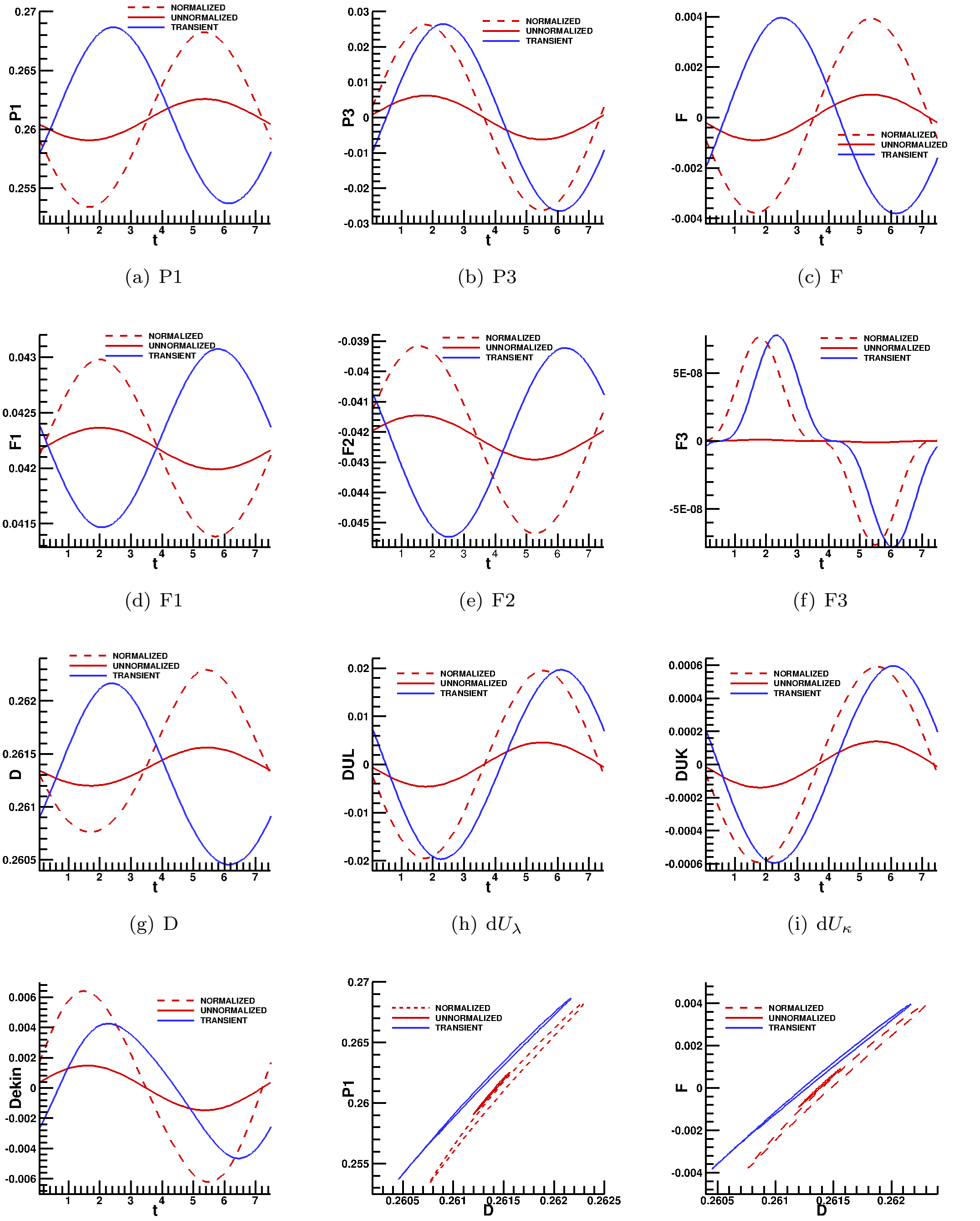
(a) P1 (b) P3 (c) F

(d) F1 (e) F2 (f) F3

(g) D (h) $\mathrm{d}U_{\lambda}$ (i) $\mathrm{d}U_{\kappa}$

(j) $\dot{E}$ (k) P1-D (l) F-D

Figure 5-4: N1-2: $Pud = 0.70815$, $C_{\lambda} = 5500$, the blue solid curve is from unsteady simulation [80], the red solid curve is from energy solution from eigenpairs, and the dashed curve is the normalized energy solution from eigenpairs.



(a) P1

(b) P3

(c) F

(d) F1

(e) F2

(f) F3

(g) D

(h) dU_λ

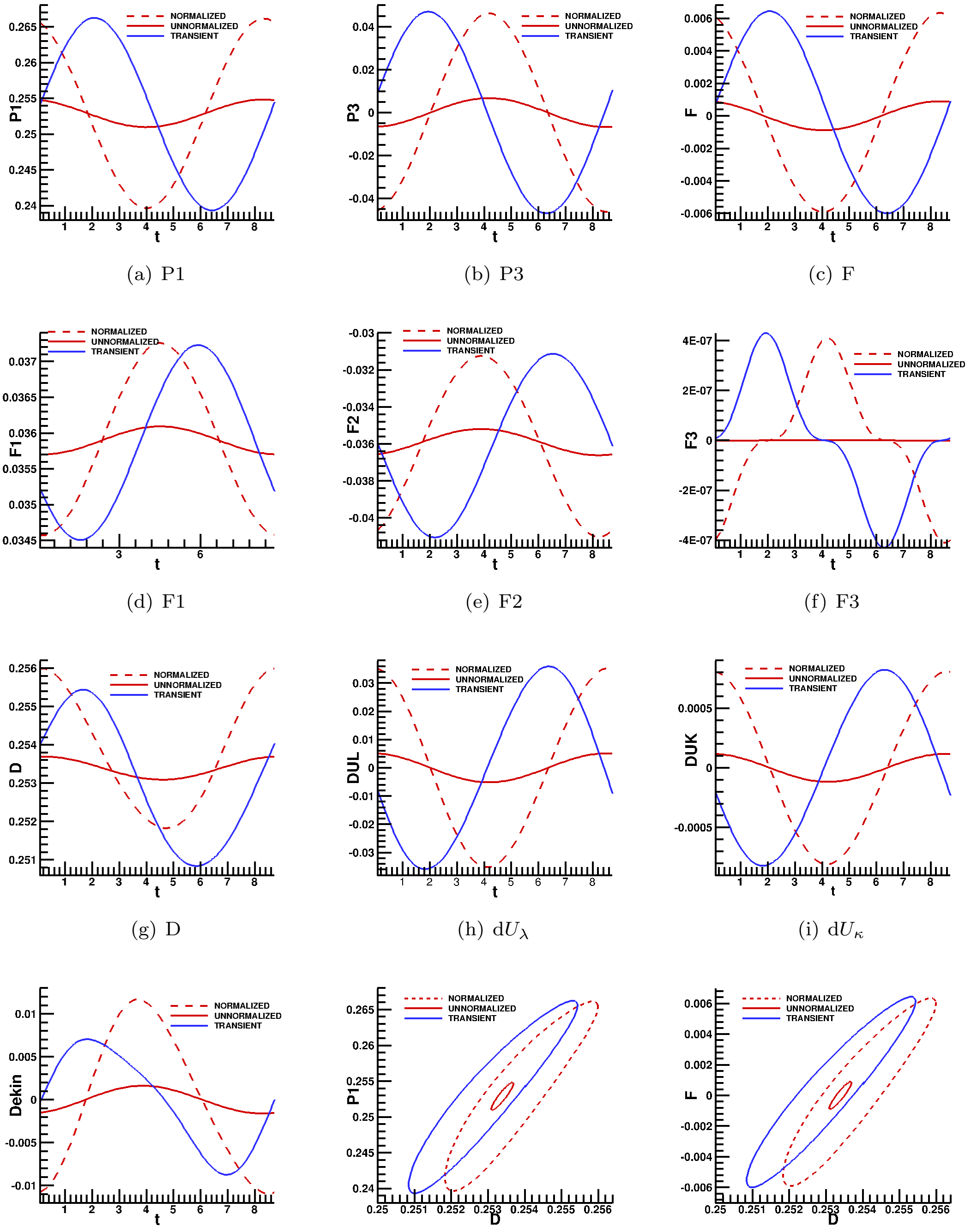
(i) dU_κ

(j) $\dot{E}$

(k) P1-D

(l) F-D

Figure 5-5: N1-3: $Pud = 0.6872$, $C_\lambda = 2000$, the blue solid curve is from unsteady simulation [80], the red solid curve is from energy solution from eigenpairs, and the dashed curve is the normalized energy solution from eigenpairs.



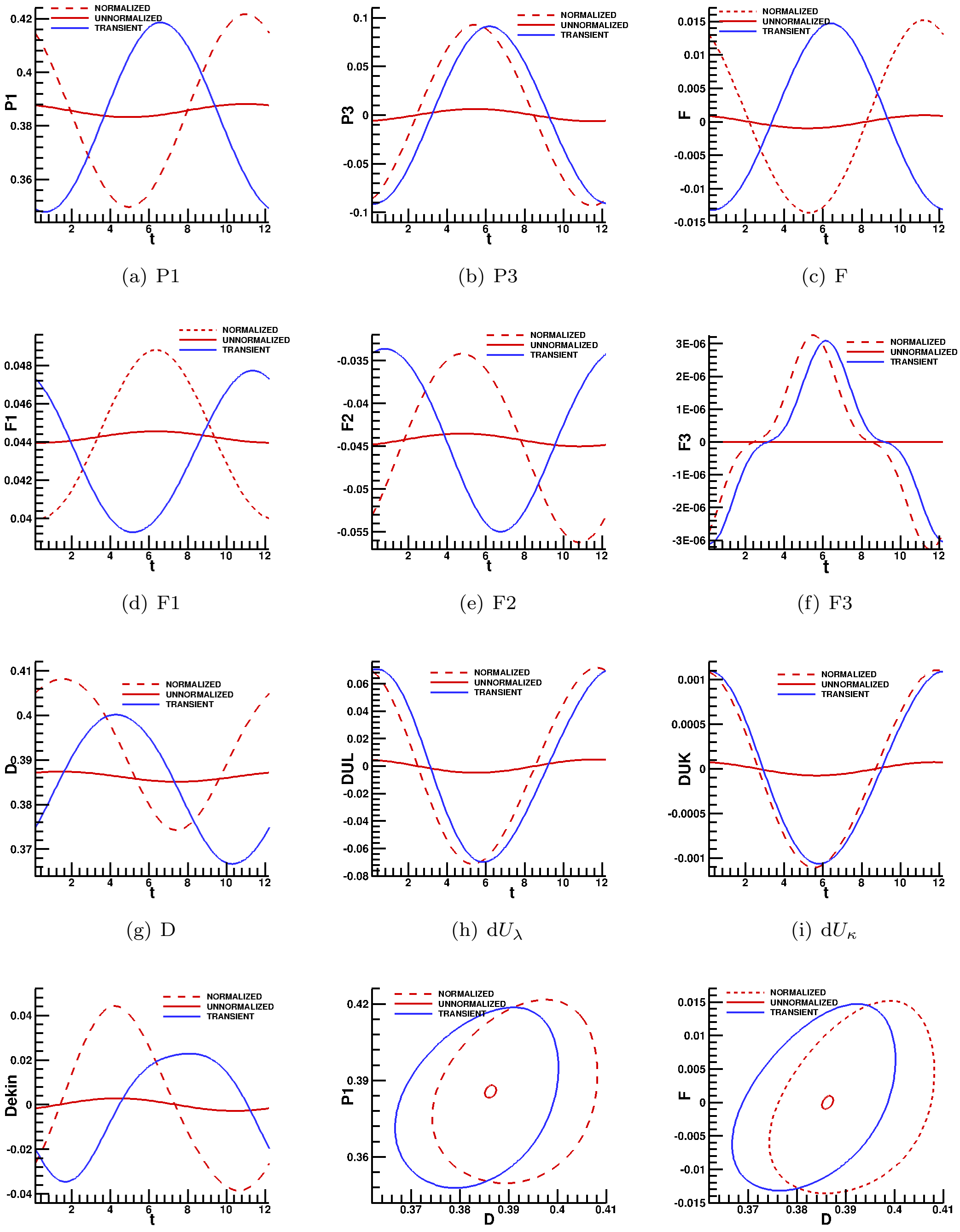
(a) P1 (b) P3 (c) F

(d) F1 (e) F2 (f) F3

(g) D (h) dU_λ (i) dU_κ

(j) $\dot{E}$ (k) P1-D (l) F-D

Figure 5-6: N1-4: $Pud = 0.7031$, $C_\lambda = 927$, the blue solid curve is from unsteady simulation [80], the red solid curve is from energy solution from eigenpairs, and the dashed curve is the normalized energy solution from eigenpairs.



(a) P1 (b) P3 (c) F

(d) F1 (e) F2 (f) F3

(g) D (h) dU_λ (i) dU_κ

(j) $\dot{E}$ (k) P1-D (l) F-D

Figure 5-7: N1-5: $Pud = 1.0$, $C_\lambda = 308.4$, the blue solid curve is from unsteady simulation [80], the red solid curve is from energy solution from eigenpairs, and the dashed curve is the normalized energy solution from eigenpairs.

The figures in the previous five pages shows the validations of energy solutions between eigenpairs and the unsteady simulation. Because the effect of different perturbations has been discussed and the perturbation in eigenvalue problem is infinitesimal. In order to make the results in the same range, the eigenvectors will be normalized. The normalizing strategy of eigenvectors is to compare the range of the eight variables at a same node number, both in the eigensolution and the unsteady simulation, and make the eight variables obtained from eigensolution to be same as the unsteady solution in a period. Let us take the velocity u in X-direction for example, the velocity of transient solution is u_t, and u_e denotes eigensolution. So the normalizing coefficient c_u for u is

$$c_u = \frac{u_{t-max} - u_{t-min}}{u_{e-max} - u_{e-min}}, \tag{5.13}$$

and the velocity of eigensolution after normalization is

$$u_{normal} = c_u u_e \tag{5.14}$$

All the solutions $\mathbf{U}$ contain eight variables as $\mathbf{U} = (u_j, p_j, v_j, x_j, y_j, \theta_j, \lambda_j, \kappa_j)^T$, and we have compared with all the eight normalizing coefficients, which are same for the same neutral point. Hence, the energy from eigensolution can describe the energy distribution very well, while the only difference with the transient energy is that the perturbation of eigensolution is infinitesimal, so the corresponding plot looks a bit smaller than transient analysis. When the wall stiffness C_λ is large, which means both the velocity and Y-displacement of the beam are quite small, the current codes of the transient analysis cannot work the velocities u and v out very well and precisely(the velocity curve contain lots of "zigzag"). And also, both the dissipation D and the rate of change of kinetic energy $\dot{E}$ is fairly sensitive to the velocities, and in the transient analysis, the oscillation of other modes must have some influence to the current neutral mode, so the dissipation D and the rate of change of kinetic energy $\dot{E}$ might be a little different, but their differences are not very large, which can be accepted.

Via the energy analysis by eigenpairs, we continue the energy budget analysis with the new Mode-1 neutral points, and all the results are shown in the following pages, including the pictures and tables.

Table 5.3: The average energy quantities at the operating points in pressure-driven system

C_λ	$P1$	F	D	$\dot{E}$	D_S	D_S/F
2×10^8	2.1385	1.8904×10^{-3}	2.1456	1.3419×10^{-7}	1.1403×10^{-7}	6.0291×10^{-5}
1.5×10^8	1.8667	1.1673×10^{-3}	1.8726	3.5121×10^{-5}	3.2516×10^{-5}	2.7856×10^{-2}
1.1×10^8	1.6172	8.5335×10^{-4}	1.6220	1.8426×10^{-7}	1.8384×10^{-7}	2.1543×10^{-4}
8×10^7	1.4043	5.6743×10^{-4}	1.4083	2.1705×10^{-7}	2.3339×10^{-7}	4.1132×10^{-4}
6×10^7	1.2472	4.0212×10^{-4}	1.2506	2.4981×10^{-7}	2.8545×10^{-7}	7.0984×10^{-4}
4×10^7	1.0806	2.5497×10^{-4}	1.0835	2.9715×10^{-7}	3.6029×10^{-7}	1.4131×10^{-3}
3.5×10^7	1.0387	2.1807×10^{-4}	1.0414	3.0992×10^{-7}	3.9029×10^{-7}	1.7897×10^{-3}
3×10^7	0.9963	1.9120×10^{-4}	0.9989	3.2454×10^{-7}	4.1854×10^{-7}	2.1890×10^{-3}
2×10^7	0.9140	1.5320×10^{-4}	0.9164	3.5763×10^{-7}	4.7864×10^{-7}	3.1243×10^{-3}
5500	0.2917	0.1079	0.2924	1.5751×10^{-6}	4.2077×10^{-6}	3.8998×10^{-5}
2000	0.2608	0.0844	0.2614	2.0561×10^{-6}	6.2096×10^{-6}	7.3612×10^{-5}
927	0.2530	0.0718	0.2534	2.4868×10^{-6}	9.2926×10^{-6}	1.7095×10^{-4}
308.4	0.3857	1.2618×10^{-5}	0.3863	3.7349×10^{-6}	2.8603×10^{-5}	2.2669

The Table 5.3 shows the average energy quantities at the operating points of pressure-driven system in a whole period. All the signs of F in the table are positive, which indicates that the influx at the upstream maintains the oscillation and provides the energy entering the system. This results is same as Liu et al. [80], and suggested by Jensen et al. [53].

Jensen et al. [53] predicated that the ratio of D_S/F would be close to 2/3 for the almost undeformed configuration. And with unsteady simulation, Liu et al. [80] achieved 2/3 at the neutral point $Pud = 1.2$, $C_\lambda = 2.0 \times 10^7$, where the large C_λ indicates the large wall stiffness. If the wall stiffness is so large that the deformation of the beam is small enough, and can be considered as a undeformed configuration. In Table 5.3, the ratios of D_S/F are not close to 2/3, but it is close to zero. The energy solutions of the large C_λ with a small perturbation were also calculated by the unsteady simulation. In these results, the ratios of D_S/F are also close to zero, and there was no small loop either. So the 2/3 ratio might be reached with a large pertrubation to the steady solution. And also the small loop at the top would appear when the perturbation is sufficiently large.

5.3 Energy analysis of flow-driven system

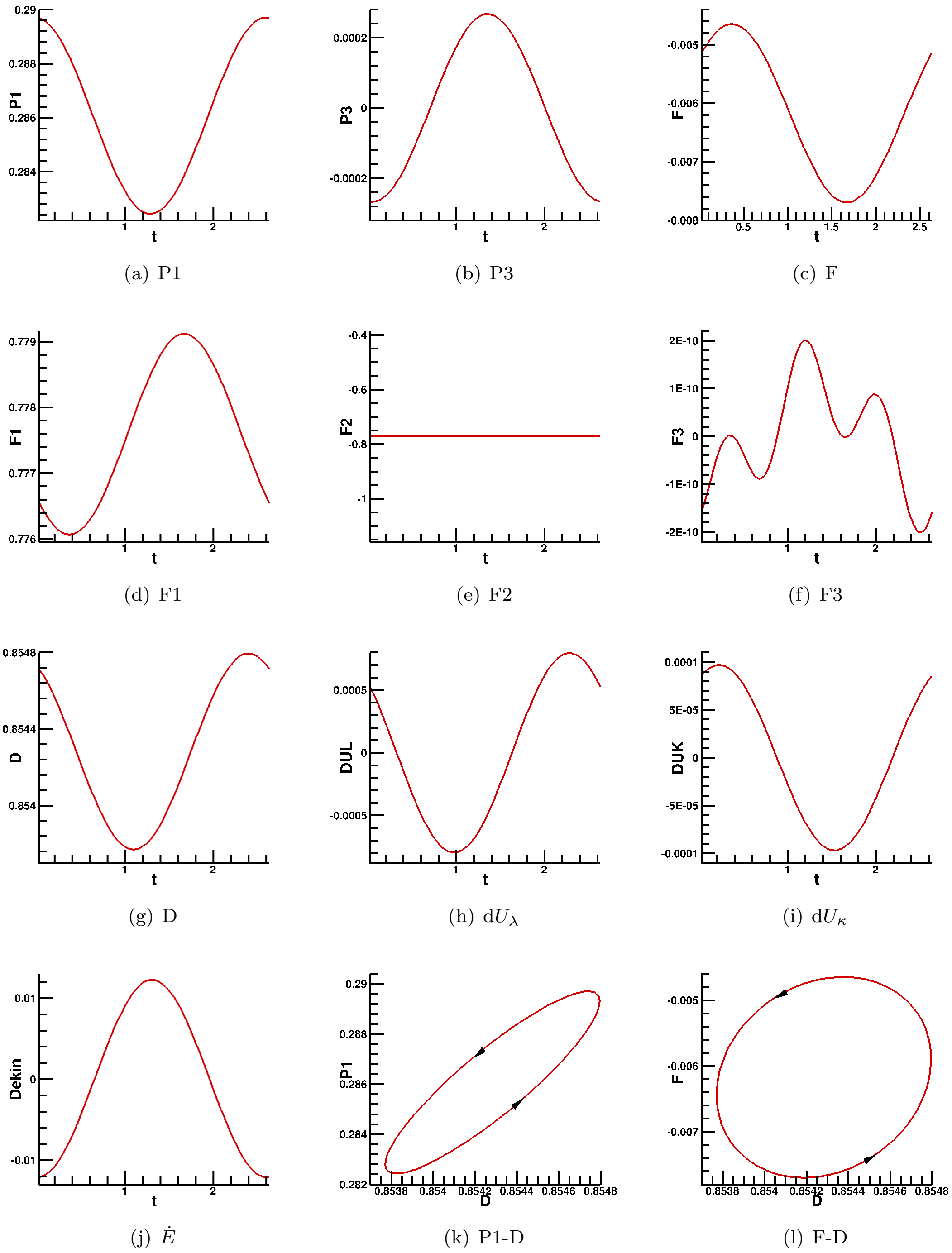
In Chapter 4, we have validated the AR-F eigensolver, and obtained a neutral curve with a refined mesh($50 \times (200 + 140 + 100)$, the number of degree of freedom is 55392). And in the previous section, the energy results from eigenpairs are validated as well. Consequently, we are going to analyze the energy in flow-driven system in this section.

First, some control points are chosen from the refined neutral curve, shown in the following

Table 5.4.

Table 5.4: Operating points from the neutral curve in flow-driven system ($50 \times (200+140+100)$, and number of degree of freedom is 55392)

Point	Re	C_λ	ω_r	ω_i	$Mode$
N2-1	700	1672	$+3.30023\times10^{-5}$	2.40366	Mode-2
N2-2	600	1797.5	-1.14649×10^{-6}	2.42381	Mode-2
N2-3	500	1800	-6.13079×10^{-6}	2.35544	Mode-2
N2-4	300	1937	$+1.76015\times10^{-6}$	2.03575	Mode-2
N2-5	235	2410	$+1.93529\times10^{-7}$	1.72480	Mode-2
N2-6	220	2727	$+1.22905\times10^{-6}$	1.57726	Mode-2
N2-7	213	2436	$+6.90523\times10^{-7}$	1.43017	Mode-2
N2-8	212	1600	$+2.39754\times10^{-4}$	1.30612	Mode-2
N2-9	240	580	-2.62525×10^{-6}	1.24025	Mode-2
N2-10	300	447	$+3.63519\times10^{-7}$	1.27266	Mode-2
N2-11	380	397.5	-3.62141×10^{-6}	1.24308	Mode-2
N2-12	440	369	$+9.00410\times10^{-6}$	1.20026	Mode-2
N3-1	300	335.61	-5.67991×10^{-6}	4.07781	Mode-2
N3-2	250	311	$+7.02530\times10^{-5}$	3.61682	Mode-3
N4-1	231.15	60	$+9.69742\times10^{-6}$	3.85157	Mode-4



(a) P1

(b) P3

(c) F

(d) F1

(e) F2

(f) F3

(g) D

(h) dU_λ

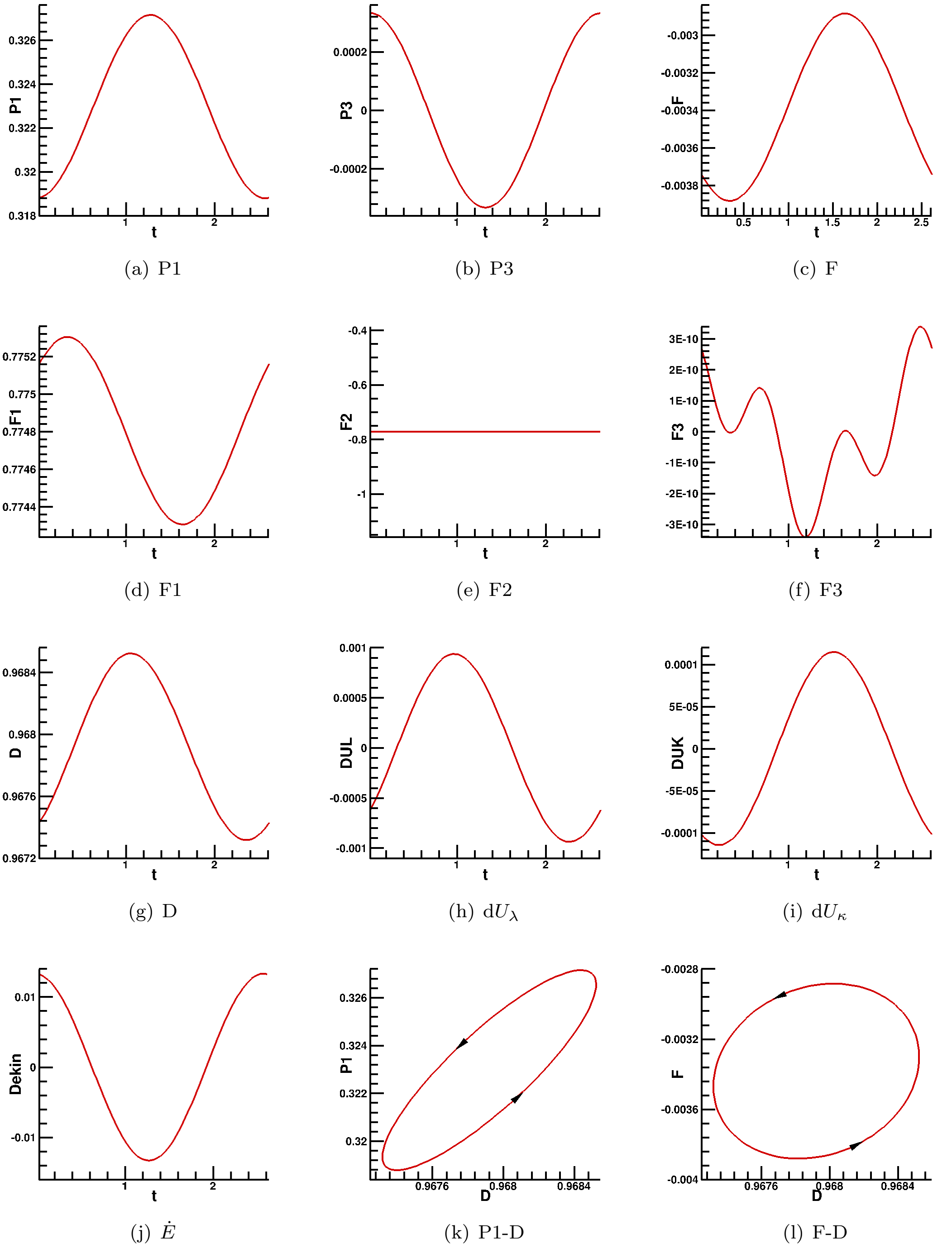
(i) dU_κ

(j) $\dot{E}$

(k) P1-D

(l) F-D

Figure 5-8: N2-1: $Re = 700$, $C_\lambda = 1672$, $\omega_r = +3.30023 \times 10^{-5}$, $\omega_i = 2.40366$, Mode-2



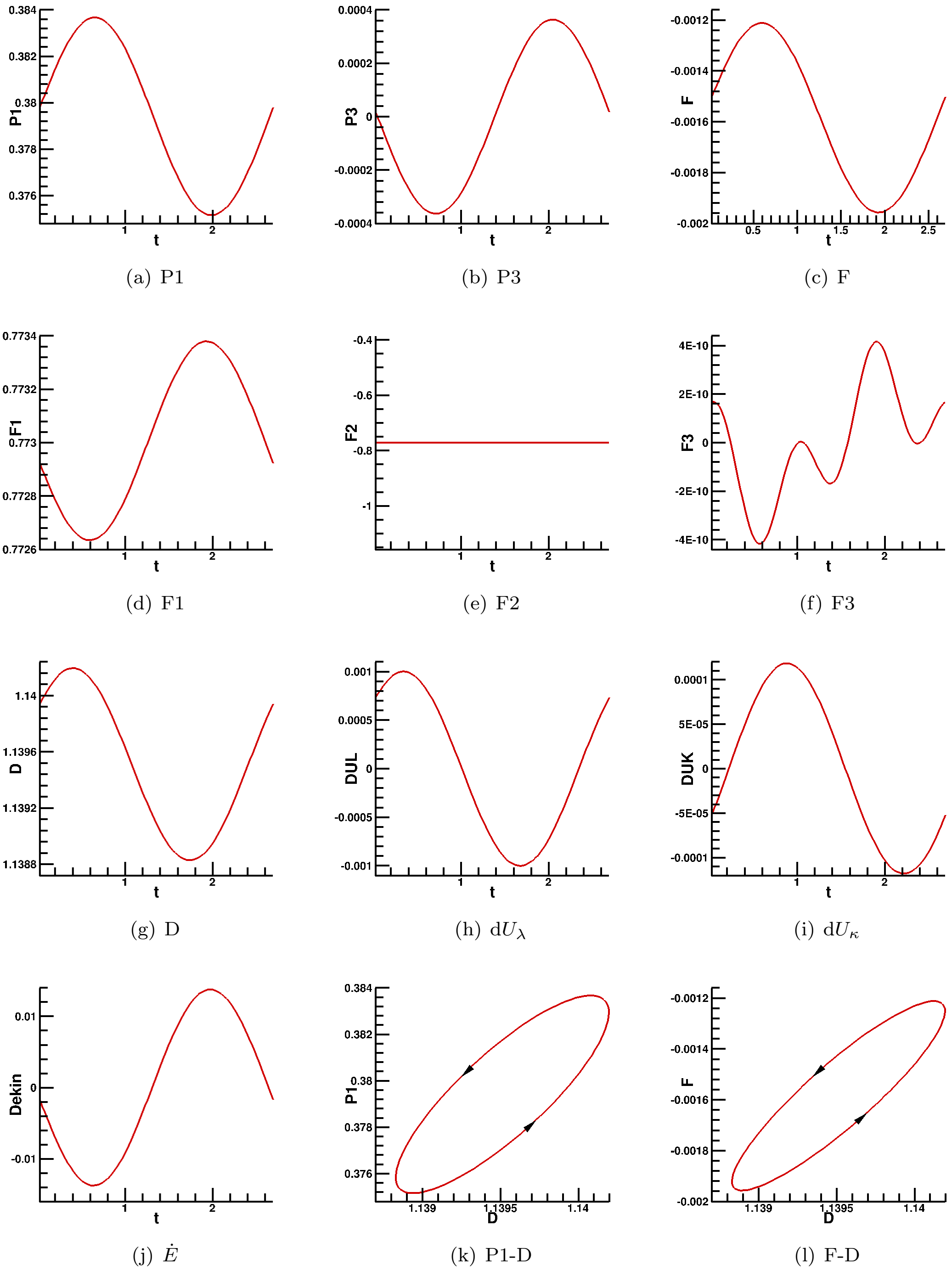
(a) P1 (b) P3 (c) F

(d) F1 (e) F2 (f) F3

(g) D (h) dU_λ (i) dU_κ

(j) $\dot{E}$ (k) P1-D (l) F-D

Figure 5-9: N2-2: $Re = 600$, $C_\lambda = 1797.5$, $\omega_r = -1.14649 \times 10^{-6}$, $\omega_i = 2.42381$, Mode-2



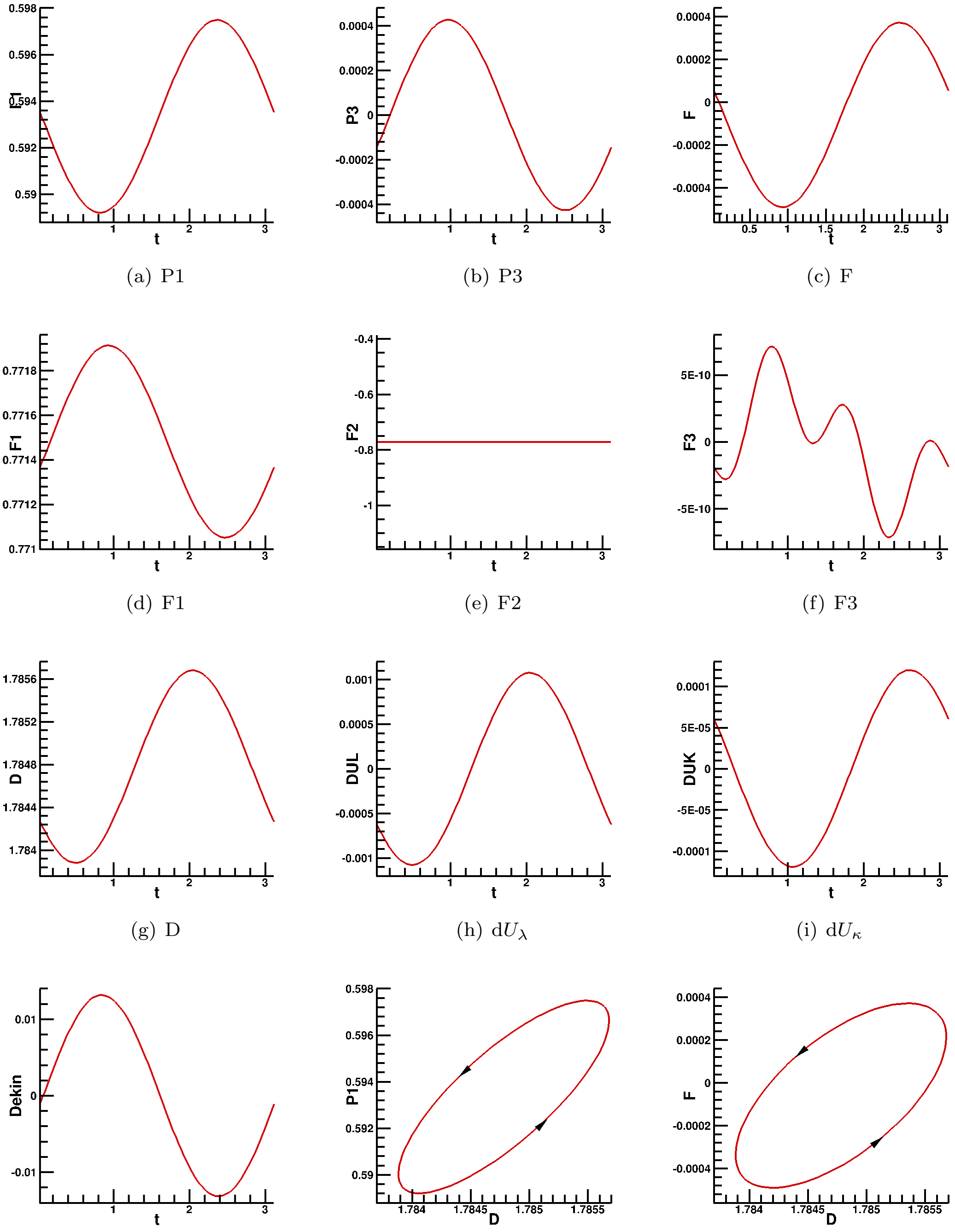
(a) P1 (b) P3 (c) F

(d) F1 (e) F2 (f) F3

(g) D (h) dU_λ (i) dU_κ

(j) $\dot{E}$ (k) P1-D (l) F-D

Figure 5-10: N2-3: $Re = 500$, $C_\lambda = 1800$, $\omega_r = -6.13079 \times 10^{-6}$, $\omega_i = 2.35544$, Mode-2



(a) P1

(b) P3

(c) F

(d) F1

(e) F2

(f) F3

(g) D

(h) dU_λ

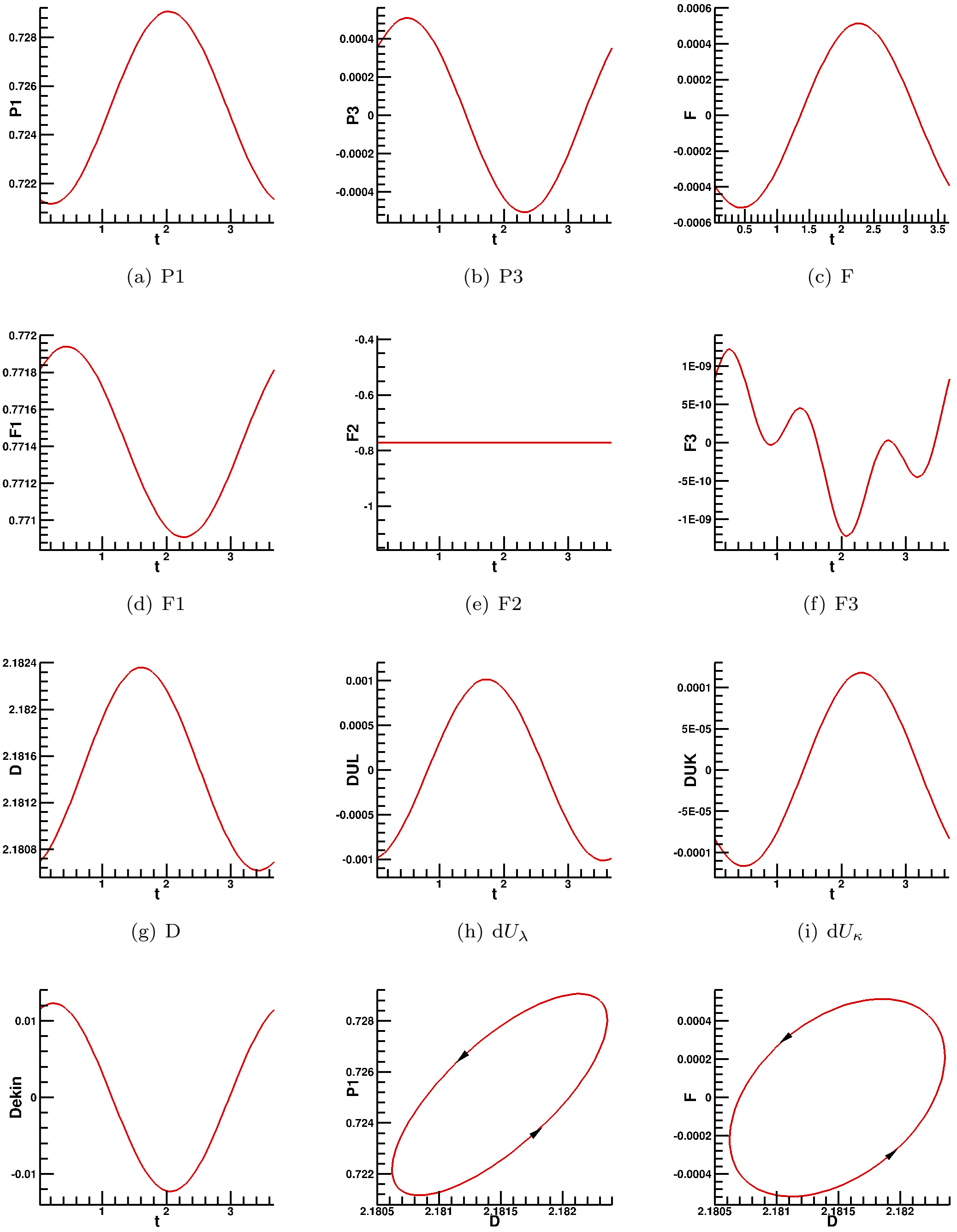
(i) dU_κ

(j) $\dot{E}$

(k) P1-D

(l) F-D

Figure 5-11: N2-4: $Re = 300$, $C_\lambda = 1937$, $\omega_r = 1.76015 \times 10^{-6}$, $\omega_i = 2.03575$, Mode-2



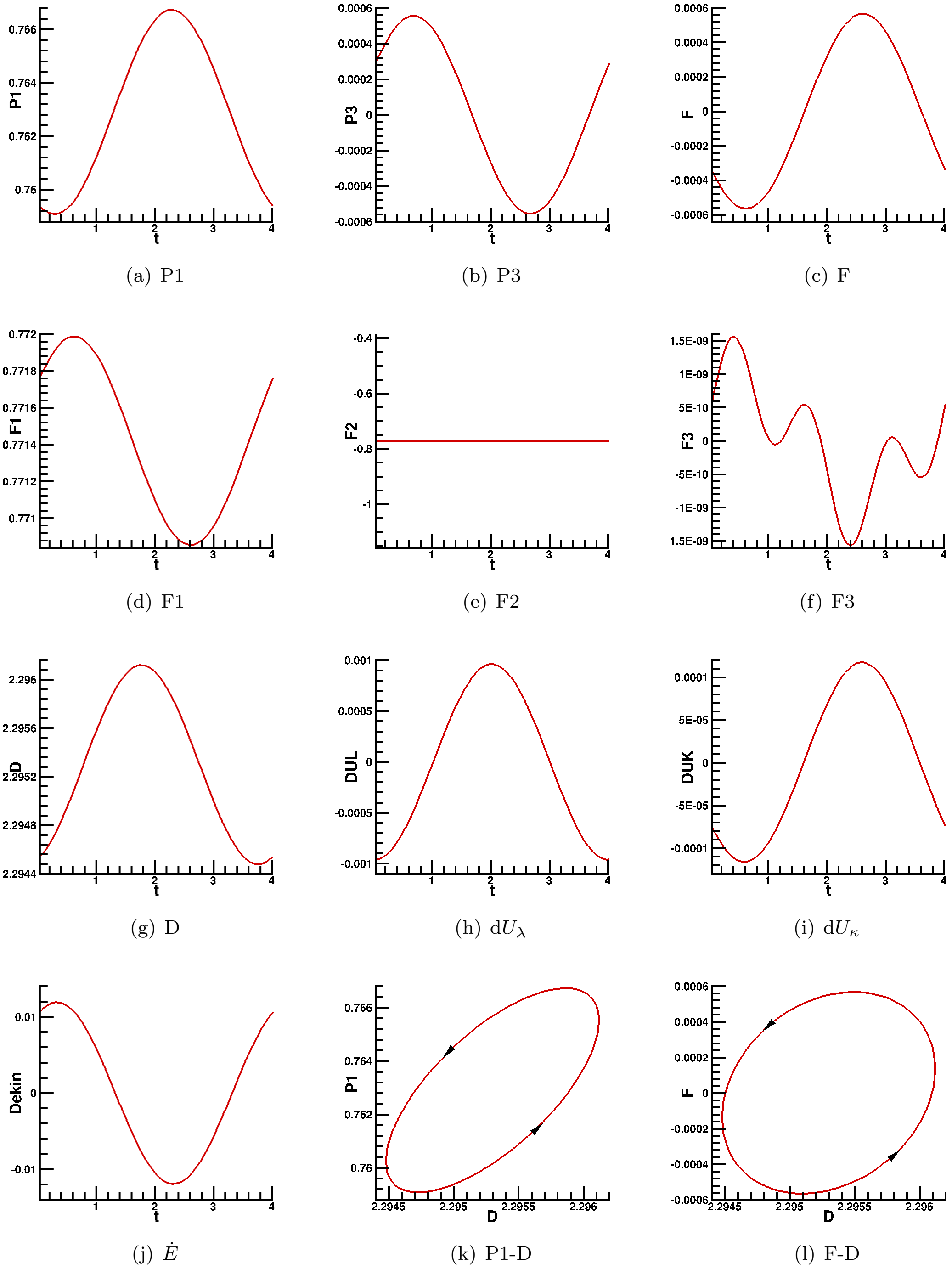
(a) P1 (b) P3 (c) F

(d) F1 (e) F2 (f) F3

(g) D (h) dU_λ (i) dU_κ

(j) $\dot{E}$ (k) P1-D (l) F-D

Figure 5-12: N2-5: $Re = 235$, $C_\lambda = 2410$, $\omega_r = 1.93529 \times 10^{-7}$, $\omega_i = 1.72480$, Mode-2



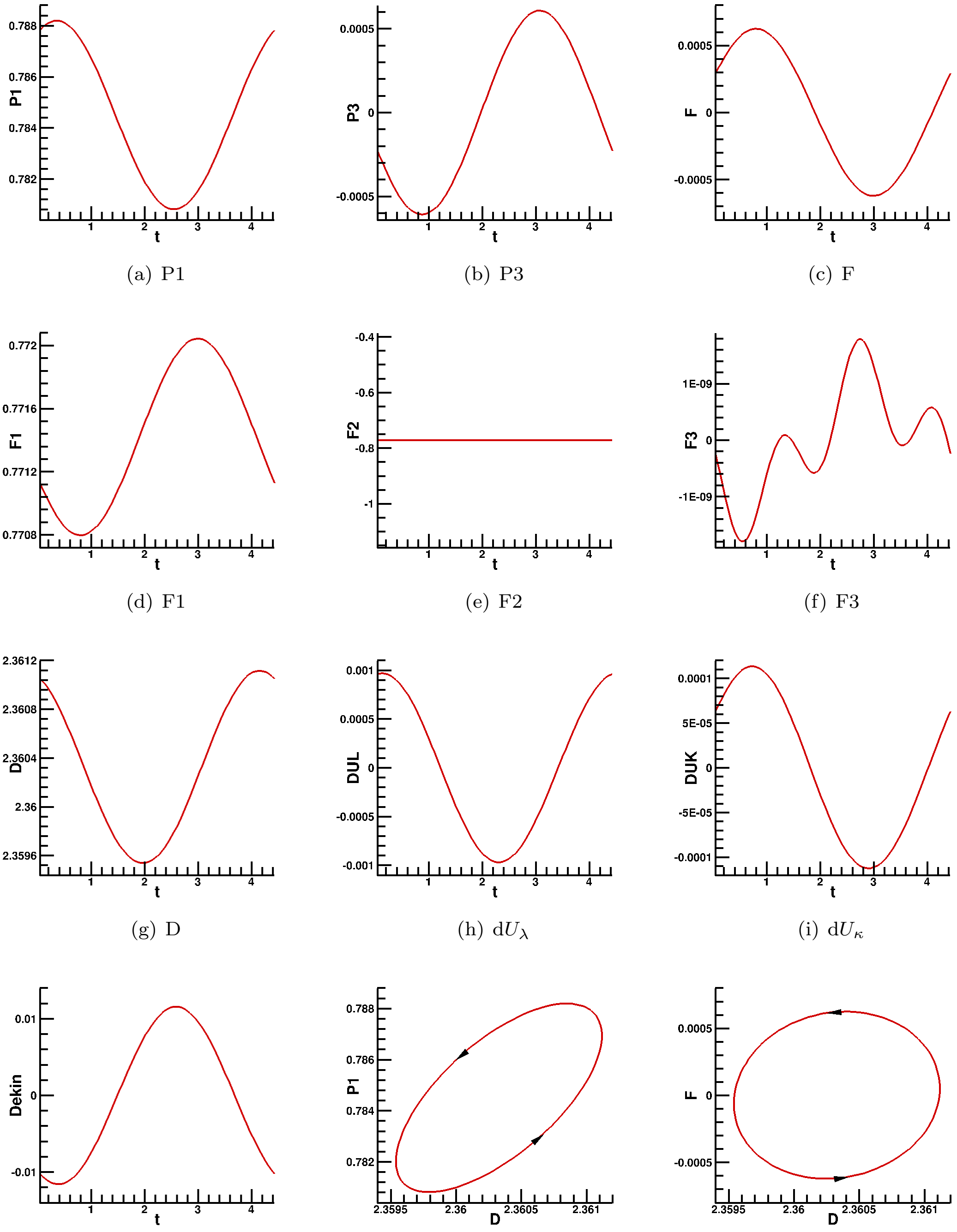
(a) P1 (b) P3 (c) F

(d) F1 (e) F2 (f) F3

(g) D (h) $\mathrm{d}U_\lambda$ (i) $\mathrm{d}U_\kappa$

(j) $\dot{E}$ (k) P1-D (l) F-D

Figure 5-13: N2-6: $Re = 220$, $C_\lambda = 2727$, $\omega_r = 1.22905 \times 10^{-6}$, $\omega_i = 1.57726$, Mode-2



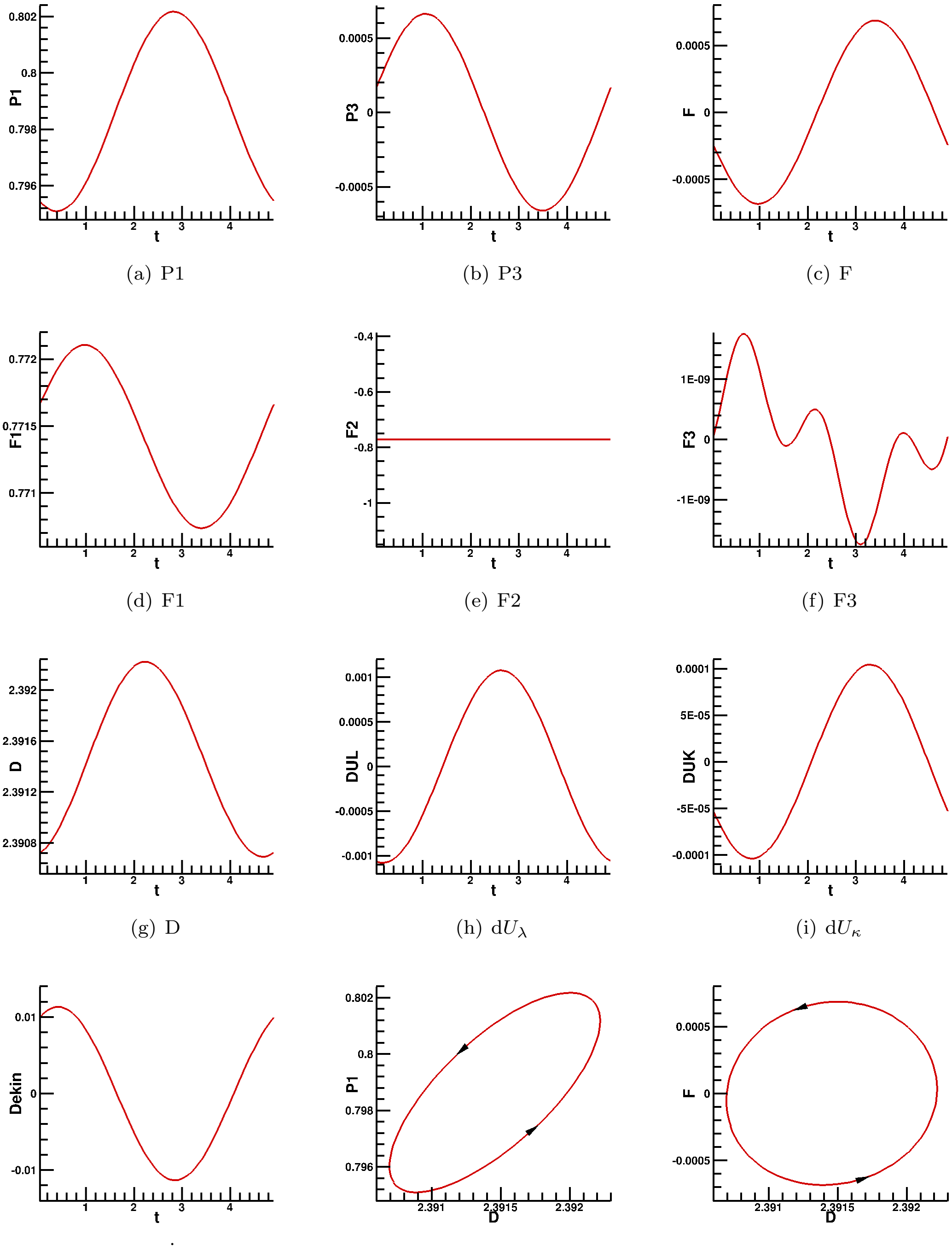
(a) P1 (b) P3 (c) F

(d) F1 (e) F2 (f) F3

(g) D (h) dU_λ (i) dU_κ

(j) $\dot{E}$ (k) P1-D (l) F-D

Figure 5-14: N2-7: $Re = 213$, $C_\lambda = 2436$, $\omega_r = 6.90523 \times 10^{-7}$, $\omega_i = 1.43017$, Mode-2



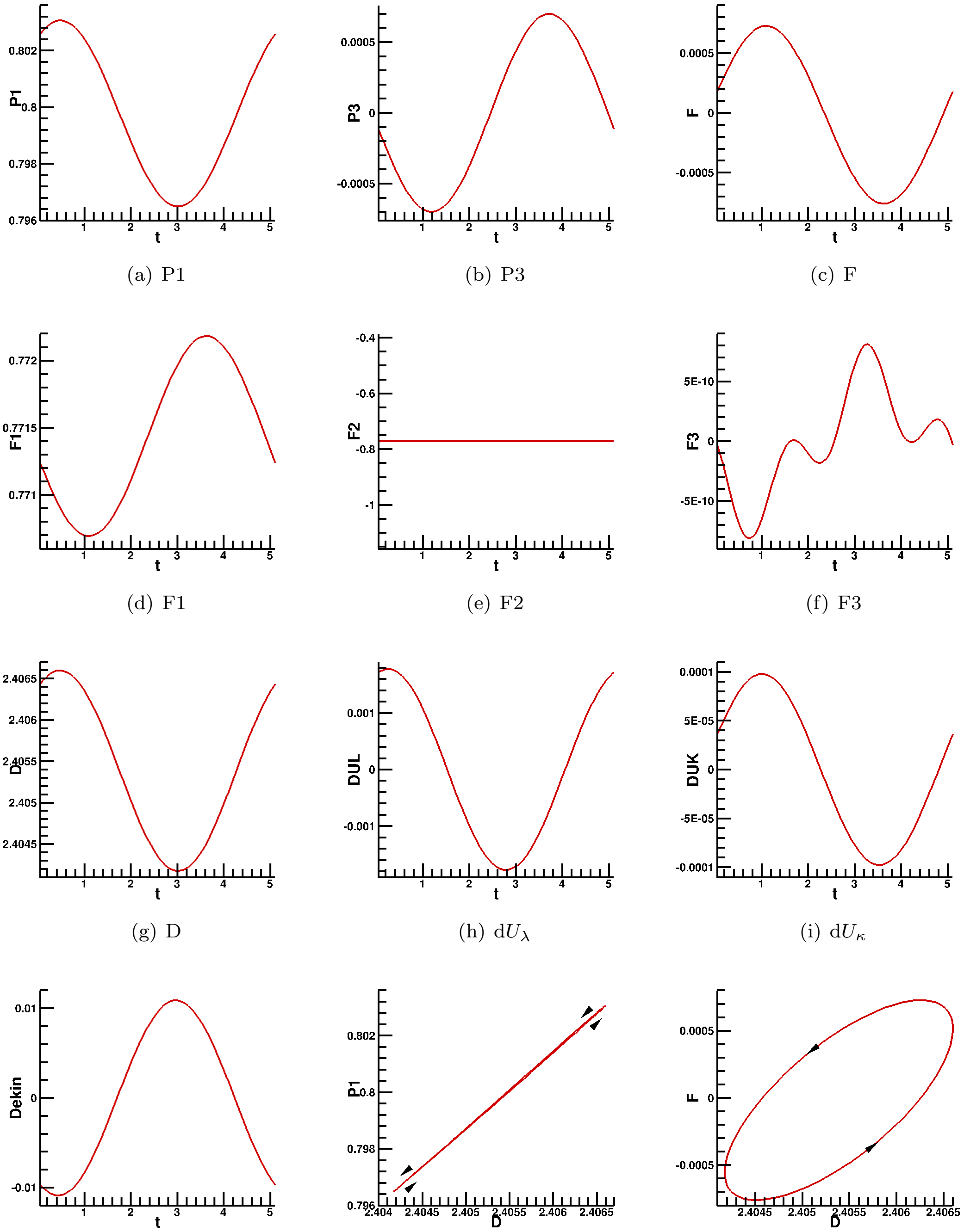
(a) P1 (b) P3 (c) F

(d) F1 (e) F2 (f) F3

(g) D (h) dU_λ (i) dU_κ

(j) $\dot{E}$ (k) P1-D (l) F-D

Figure 5-15: N2-8: $Re = 212$, $C_\lambda = 1600$, $\omega_r = 2.39754 \times 10^{-4}$, $\omega_i = 1.30612$, Mode-2



(a) P1

(b) P3

(c) F

(d) F1

(e) F2

(f) F3

(g) D

(h) dU_λ

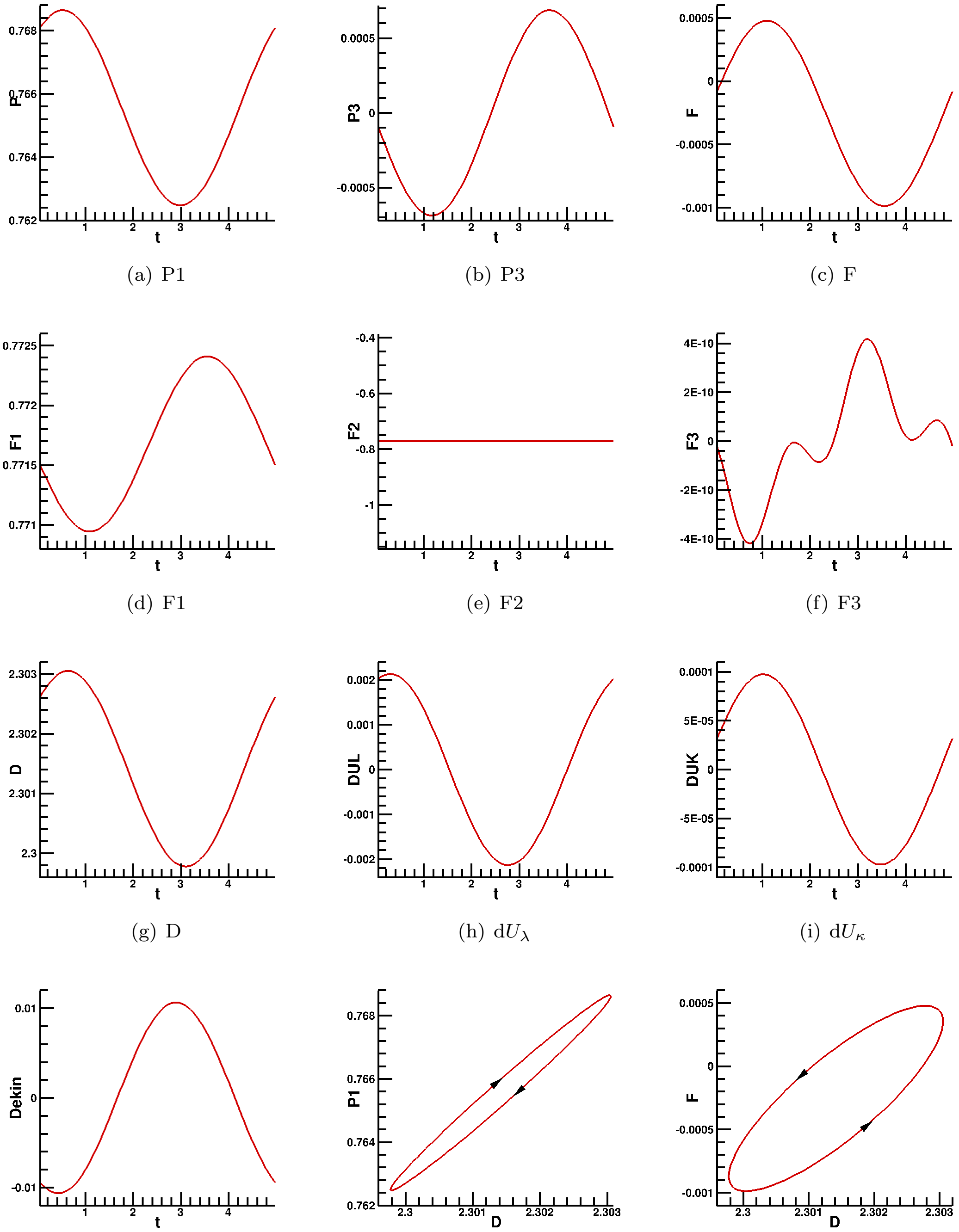
(i) dU_κ

(j) $\dot{E}$

(k) P1-D

(l) F-D

Figure 5-16: N2-9: $Re = 240$, $C_\lambda = 558$, $\omega_r = -2.62525 \times 10^{-6}$, $\omega_i = 1.24025$, Mode-2



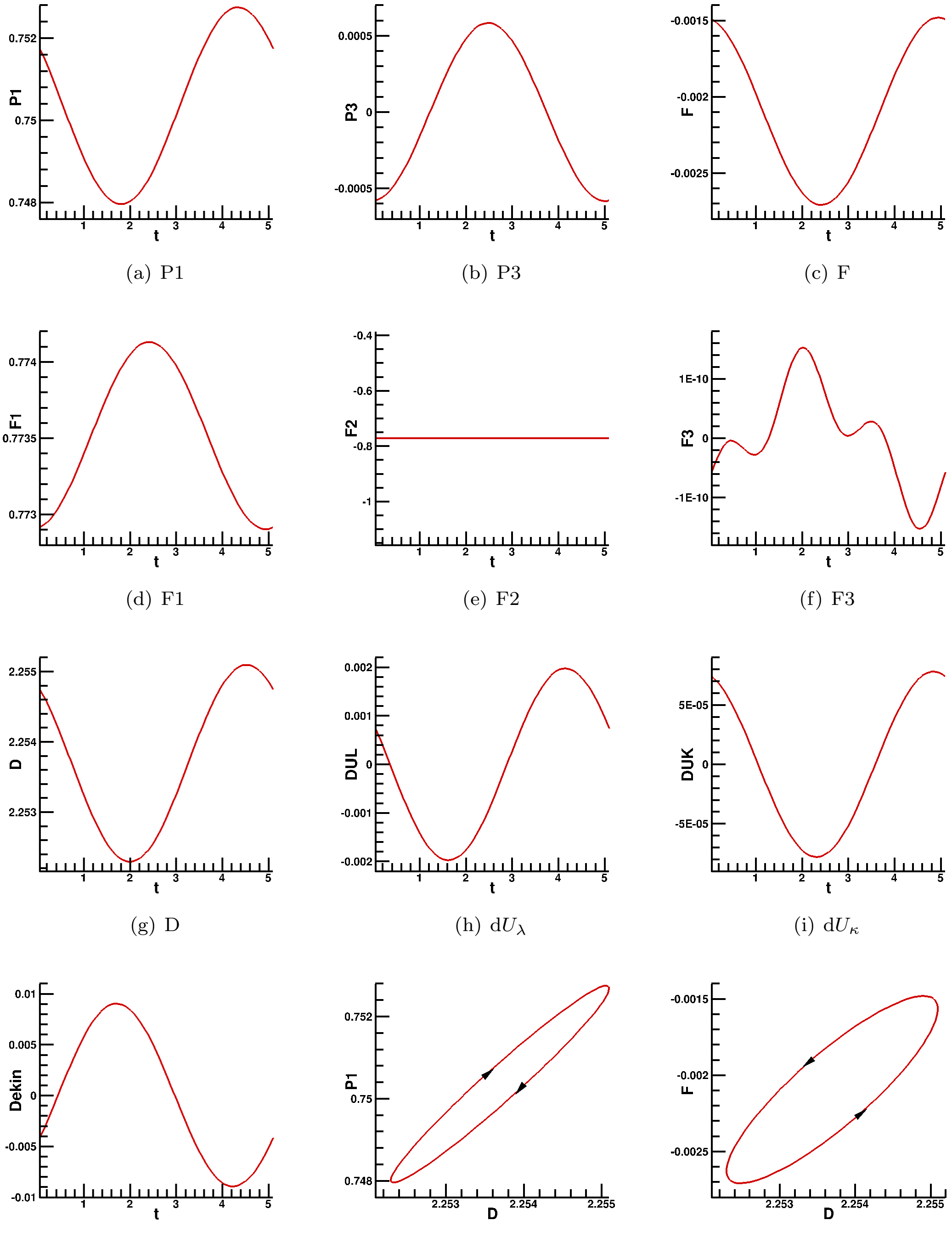
(a) P1 (b) P3 (c) F

(d) F1 (e) F2 (f) F3

(g) D (h) dU_λ (i) dU_κ

(j) $\dot{E}$ (k) P1-D (l) F-D

Figure 5-17: N2-10: $Re = 300$, $C_\lambda = 447$, $\omega_r = 3.63519 \times 10^{-5}$, $\omega_i = 1.27266$, Mode-2



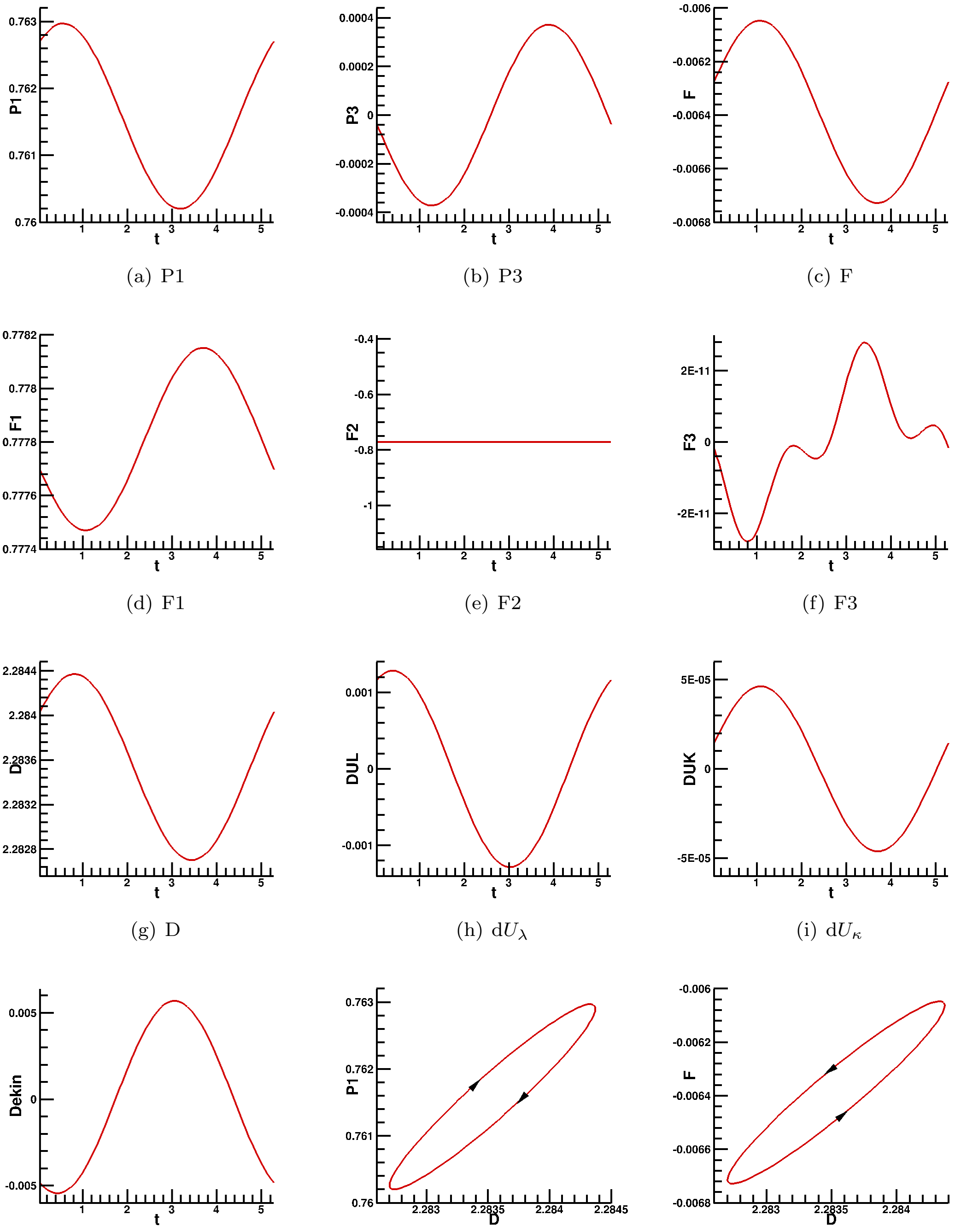
(a) P1 (b) P3 (c) F

(d) F1 (e) F2 (f) F3

(g) D (h) dU_λ (i) dU_κ

(j) $\dot{E}$ (k) P1-D (l) F-D

Figure 5-18: N2-11: $Re = 380$, $C_\lambda = 397.5$, $\omega_r = -3.62141 \times 10^{-6}$, $\omega_i = 1.24308$, Mode-2



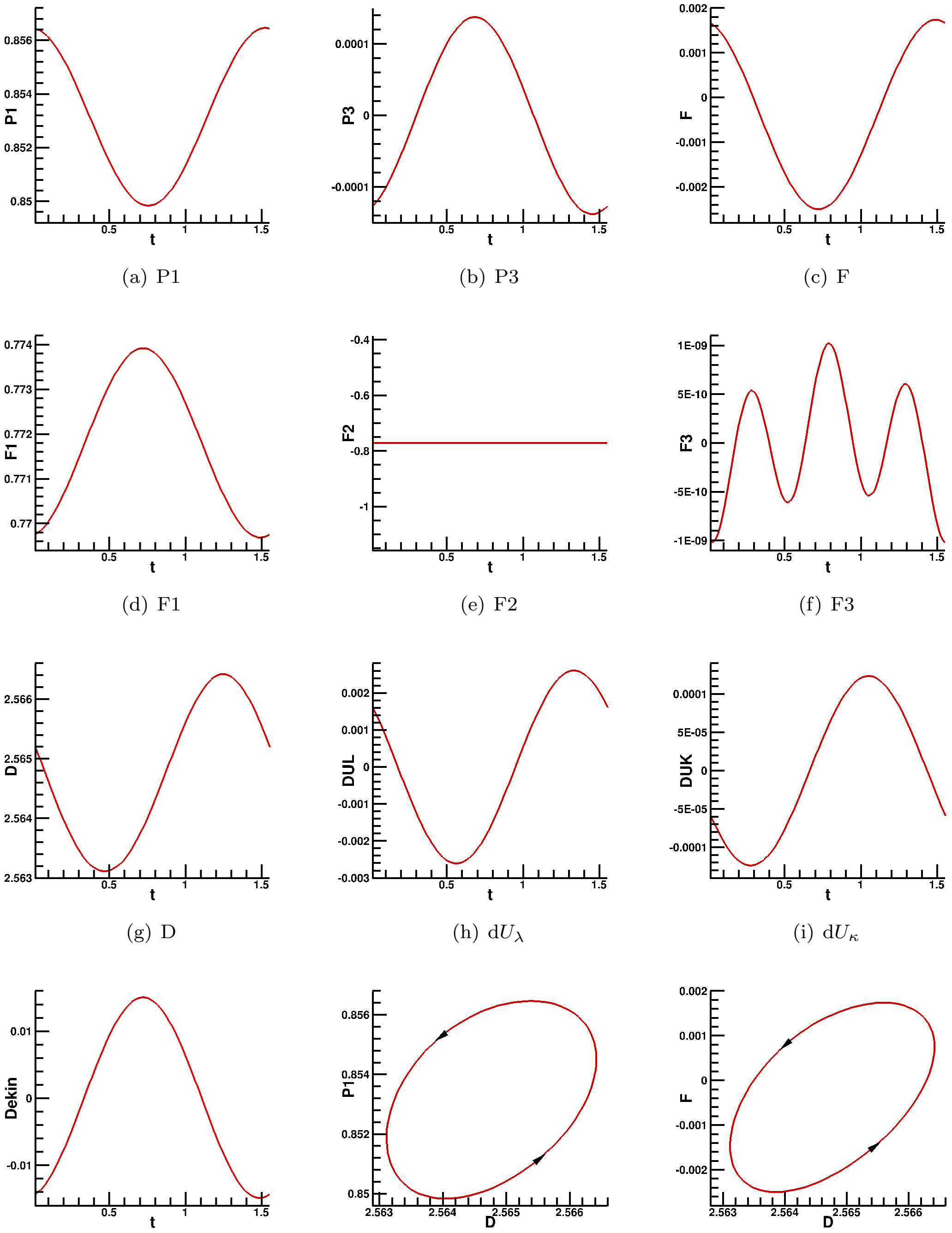
(a) P1 (b) P3 (c) F

(d) F1 (e) F2 (f) F3

(g) D (h) dU_λ (i) dU_κ

(j) $\dot{E}$ (k) P1-D (l) F-D

Figure 5-19: N2-12: $Re = 440$, $C_\lambda = 369$, $\omega_r = 9.00410 \times 10^{-6}$, $\omega_i = 1.20026$, Mode-2



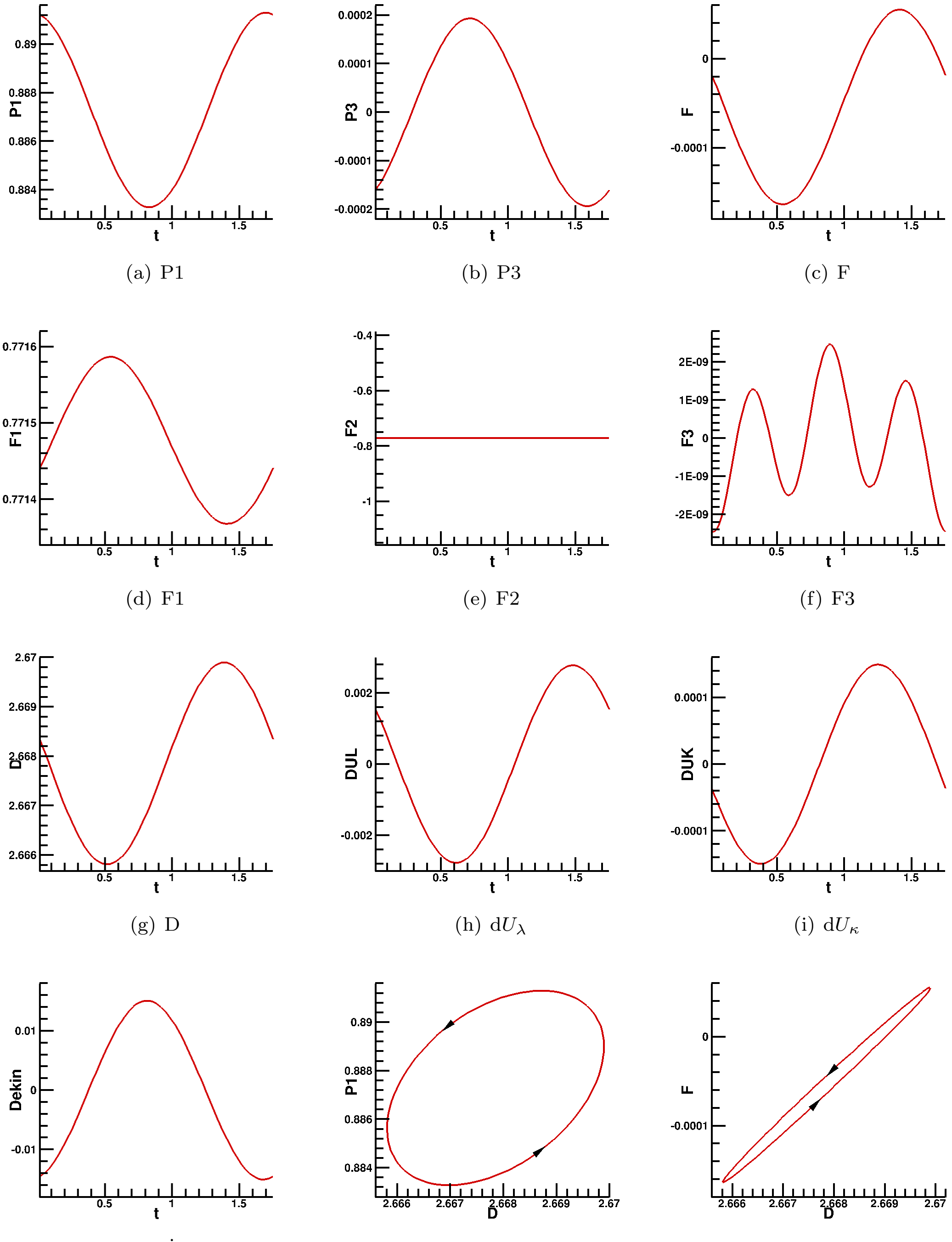
(a) P1 (b) P3 (c) F

(d) F1 (e) F2 (f) F3

(g) D (h) dU_λ (i) dU_κ

(j) $\dot{E}$ (k) P1-D (l) F-D

Figure 5-20: N3-1: $Re = 300$, $C_\lambda = 335.61$, $\omega_r = -5.67991 \times 10^{-6}$, $\omega_i = 4.07781$, Mode-3



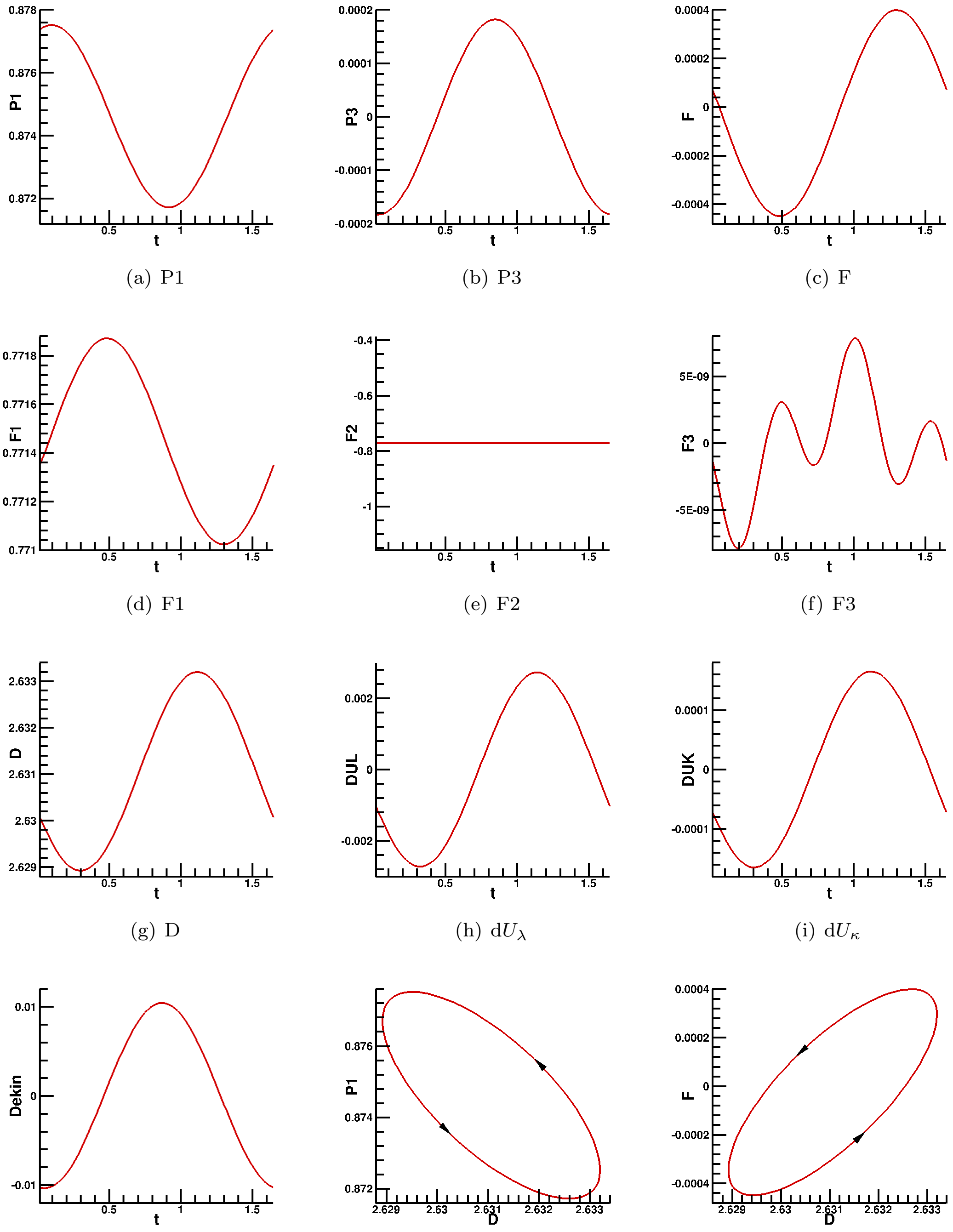
(a) P1 (b) P3 (c) F

(d) F1 (e) F2 (f) F3

(g) D (h) dU_λ (i) dU_κ

(j) $\dot{E}$ (k) P1-D (l) F-D

Figure 5-21: N3-2: $Re = 250$, $C_\lambda = 311$, $\omega_r = 7.02530 \times 10^{-5}$, $\omega_i = 3.61682$, Mode-3



(a) P1 (b) P3 (c) F

(d) F1 (e) F2 (f) F3

(g) D (h) dU_λ (i) dU_κ

(j) $\dot{E}$ (k) P1-D (l) F-D

Figure 5-22: N4-1: $Re = 231.15$, $C_\lambda = 60$, $\omega_r = 9.69742 \times 10^{-6}$, $\omega_i = 3.85157$, Mode-4

The pictures in the past twelve pages illustrate the results of energy integrals, respectively. In the next paragraphs, some description and discussion will be introduced.

The subfigures with the label (a) describe the rate of work of upstream pressure P1. As the velocity of the flow is U_0 at the entrance, P1 is determined by the pressure at the inlet only. According to the dimensionless parameter $p^* = \dfrac{p}{\rho U_0^2}$ and $Re^* = \dfrac{\rho U_0 D}{\mu}$, the pressure of inlet

p can be expressed by the Reynolds number Re as $p = \dfrac{\rho p}{\mu^2 Re^2}$. From this, we know that the pressure of inlet p is inversely related to the Reynolds number Re. Because P1 is a integral of the pressure p. So, if Re is large, the pressure p is small, and P1 is small as well.

The rate of work done by the external pressure to the beam P3 is in the subfigures with label (b). Along with the drop of the Reynolds number Re and the extensional stiffness of the beam C_λ, the term P3 increases. And it is found that the P3 of Mode-2 is much larger than those of Mode-3 and Mode-4.

F(=F1+F2+F3) is the rate of net kinetic energy flux into the system, and they are in the subfigure (c) to (f). From Table 5.5, F is negative when Re is large. And when Re is less than 300, this is close to zero. F1 is the kinetic energy at the outlet, which is dependent on the velocity component of the fluid very much. When Re is large, the velocity is high. For this reason, F1 is large as well. F2 is the kinetic energy at the inlet. Because in the flow-driven system, the velocity of fluid at the entrance is specified as $u = 6y(y-1)$ and $v = 0$ (Table 3.2 in Chapter 3 shows these). Therefore, F2 is a constant after the integration. F3 describes the kinetic energy of the beam. Because both of the velocity and displacement of the beam is fairly small, the result of integral is small as well. This is very different from the unsteady similation, whose perturbation is a bit larger than those in the linear stability analysis.

D is the viscous dissipation, which is drawn in the subfigure (g). The dissipation increases along with the drop of the Reynolds number Re. In the Re-C_λ neutral curve of flow-driven system(Figure 4-7), the stable zone shows that the oscillation or disturbance is damped out by the viscosity of fluid, when Re is less than about 250. This proves that the augment of dissipation is along with the decrease of Re. C_λ also has some slight influence on the dissipation. Compared the neutral point N2-4 with N2-10, whose Re is 300, it is easy to find that the dissipation of N2-10 is much larger than N2-4.

$\mathrm{d}U_\lambda$ and $\mathrm{d}U_\kappa$ are the rate of stretching and bending strain energy in the beam, and are shown in the subfigure (h) and (i), respectively. From the pictures, we find that $\mathrm{d}U_\lambda$ of Mode-3 and Mode-4 are larger than those of Mode-2. And in all the Mode-2 points, when C_λ is small, $\mathrm{d}U_\lambda$ is large. $\mathrm{d}U_\kappa$ of Mode-3 and Mode-4 are larger than those Mode-2. The reason of these must be that the deformation of the beam in the high mode is larger than that in the low mode, so the strain energy is large. The strain energy in low branch is a little larger than that in the high one.

$\dot{E}$ is the rate of change of kinetic energy. This indicates the loss of kinetic energy, and we hope that the average quantities of $\dot{E}$ in one period would be increasingly small. Because of the infinitesimal perturbation in the eigenvalue problems, $\dot{E}$ is made sure to be small enough.

The phase plot of P1-D and F-D are shown in the last two subfigures. The shapes of the P1-D plots before N2-9 looks same, After N2-9, this shape becomes a bit narrow, but they still keep the same direction as before. N2-9 is a critical neutral point, because it has the narrowest P1-D plot. The ellipse in P1-D plot of N4-1 is obviously different from the others, but the reason is not clear now. For the F-D plots, we only find that the ellipse is narrow when the Reynolds number is between 400 and 500, and the ellipse becomes to be a circle in some neutral points. Hence, some more research work is very necessary for these parts.

Table 5.5: The average energy quantities at the operating points in flow-driven system

Point	ω_i	$T = 2\pi/\omega_i$	$Mode$	$P1$	F	D	$\dot{E}$
N2-1	2.4037	2.6127	Mode-2	0.2861	-6.1653×10^{-3}	0.8543	6.0641×10^{-5}
N2-2	2.4238	2.5910	Mode-2	0.3230	-3.3819×10^{-3}	0.9679	1.2206×10^{-5}
N2-3	2.3554	2.6666	Mode-2	0.3794	-1.5844×10^{-3}	1.1395	4.4915×10^{-6}
N2-4	2.0358	3.0848	Mode-2	0.5933	-5.9389×10^{-5}	1.7848	2.1428×10^{-6}
N2-5	1.7248	3.6410	Mode-2	0.7251	-1.6129×10^{-6}	2.1815	2.3585×10^{-6}
N2-6	1.5773	3.9815	Mode-2	0.7629	1.48716×10^{-6}	2.2953	2.5366×10^{-6}
N2-7	1.4302	4.3910	Mode-2	0.7845	2.2173×10^{-6}	2.3603	2.7586×10^{-6}
N2-8	1.3061	4.8082	Mode-2	0.7986	1.7202×10^{-6}	2.3915	5.8959×10^{-6}
N2-9	1.2403	5.0633	Mode-2	0.7998	-1.5965×10^{-5}	2.4054	4.3322×10^{-6}
N2-10	1.2727	4.937	Mode-2	0.7656	-2.5423×10^{-4}	2.3014	1.1584×10^{-5}
N2-11	1.2431	5.0519	Mode-2	0.7504	-2.0941×10^{-3}	2.2537	5.8453×10^{-5}
N2-12	1.2003	5.2320	Mode-2	0.7616	-6.3879×10^{-3}	2.2835	1.1957×10^{-4}
N3-1	4.0778	1.5400	Mode-3	0.8531	-3.7835×10^{-4}	2.5647	1.1296×10^{-4}
N3-2	3.6168	1.7363	Mode-3	0.8873	-5.4416×10^{-5}	2.6678	1.1441×10^{-5}
N4-1	3.8516	1.6305	Mode-4	0.8746	-2.4850×10^{-5}	2.6311	4.5207×10^{-5}

From Table 5.5, it is clear that F is no larger than zero(10^{-6} is sufficiently small and can be considered as zero) for all the operating points. This shows that the oscillatory energy is extracted by the mean flow from the structure, and the oscillations are maintained by the inlet pressure only. This result is same as Liu et al. [80], and agrees with the prediction of Stewart et al. [141]

5.4 Division of mean flow and oscillatory energy

From the first section in chapter 5, all the integrating formulas were reviewed and rederived. Since the eigensolution $\mathbf{U} = \bar{\mathbf{U}} + \Delta\mathbf{U}$ is the current solution of the system and $\bar{\mathbf{U}}$ is the steady solution and $\Delta\mathbf{U}$ is the part causing oscillation, after substituting the eigensolution $\mathbf{U} = \bar{\mathbf{U}} + \Delta\mathbf{U}$ into the integrating formulas, it is so easy to divide the energy into two parts, which are the the energy due to mean flow and the one due to oscillation.

Table 5.1 and the equation (5.11) display each integrating term, via substituting the eigensolution $\mathbf{U} = \bar{\mathbf{U}} + \Delta\mathbf{U}$ into each term, we can obtain some new integrating terms, which divide the two part of energy completely.

5.4.1 Pressure-driven system

First let us consider the equation (5.8), which has been rewritten into $P = P1 - P3 - dU_\lambda - dU_\kappa$. $P1$ is the work done by the upstream pressure, and in the pressure-driven system, P_u at the upstream is set as a constant, so

$$P1 = \int_0^1 P_u u dy \tag{5.15}$$

$$= \int_0^1 P_u(\bar{u} + \Delta u) dy \tag{5.16}$$

$$= \underbrace{\int_0^1 P_u \bar{u} dy}_{meanflow} + \underbrace{\int_0^1 P_u \Delta u dy}_{oscillation}. \tag{5.17}$$

Before continuing the derivation of $P3$, for simplicity, the final form of velocity in the normal direction u_n should be obtained first,

$$u_n = u(-\sin\theta) + v\cos\theta \tag{5.18}$$

$$= (\bar{u} + \Delta u)(-\sin(\bar{\theta} + \Delta\theta) + (\bar{v} + \Delta v)\cos(\bar{\theta} + \Delta\theta) \tag{5.19}$$

$$= (\bar{u} + \Delta u)(-\sin\bar{\theta}\cos\Delta\theta + \cos\bar{\theta}\sin\Delta\theta) \tag{5.20}$$

$$+(\bar{v} + \Delta v)(\cos\bar{\theta}\cos\Delta\theta - \sin\bar{\theta}\sin\Delta\theta) \tag{5.21}$$

$$= (-\bar{u}\sin\bar{\theta}\cancelto{1}{\cos\Delta\theta} + \bar{u}\cos\bar{\theta}\cancelto{\Delta\theta}{\sin\Delta\theta} - \Delta u\sin\bar{\theta}\cancelto{1}{\cos\Delta\theta} + \Delta u\cos\bar{\theta}\cancelto{\Delta\theta}{\sin\Delta\theta}) \tag{5.22}$$

$$+(\bar{v}\cos\bar{\theta}\cancelto{1}{\cos\Delta\theta} - \bar{v}\sin\bar{\theta}\cancelto{\Delta\theta}{\sin\Delta\theta} + \Delta v\cos\bar{\theta}\cancelto{1}{\cos\Delta\theta} - \Delta v\sin\bar{\theta}\cancelto{\Delta\theta}{\sin\Delta\theta}) \tag{5.23}$$

$$= (-\bar{u}\sin\bar{\theta} + \bar{u}\cos\bar{\theta}\Delta\theta - \Delta u\sin\bar{\theta} + \cancelto{0}{\Delta u\cos\bar{\theta}\Delta\theta}) \tag{5.24}$$

$$+(\bar{v}\cos\bar{\theta} - \bar{v}\sin\bar{\theta}\Delta\theta + \Delta v\cos\bar{\theta} - \cancelto{0}{\Delta v\sin\bar{\theta}\Delta\theta}) \tag{5.25}$$

$$= (-\bar{u}\sin\bar{\theta} + \bar{u}\cos\bar{\theta}\Delta\theta - \Delta u\sin\bar{\theta}) + (\bar{v}\cos\bar{\theta} - \bar{v}\sin\bar{\theta}\Delta\theta + \Delta v\cos\bar{\theta}). \tag{5.26}$$

Because $\Delta\theta$ is small enough, $\cos\Delta\theta \approx 1$ and $\sin\Delta\theta \approx \Delta\theta$, and the high order small term(Δ^2) is neglected. The equation (5.26) is the final result and will be useful for the following work. By substituting u_n into the integral of $P3$, we obtain that

$$P3 = \int_0^{S_0} P_e u_n ds \tag{5.27}$$

$$= \int_0^{S_0} P_e((-\bar{u}\sin\bar{\theta} + \bar{u}\cos\bar{\theta}\Delta\theta - \Delta u\sin\bar{\theta}) \tag{5.28}$$

$$+(\bar{v}\cos\bar{\theta} - \bar{v}\sin\bar{\theta}\Delta\theta + \Delta v\cos\bar{\theta}))ds \tag{5.29}$$

$$= \underbrace{\int_0^{S_0} P_e(-\bar{u}\sin\bar{\theta} + \bar{v}\cos\bar{\theta})ds}_{meanflow} \tag{5.30}$$

$$+ \underbrace{\int_0^{S_0} P_e(\bar{u}\cos\bar{\theta}\Delta\theta - \Delta u\sin\bar{\theta} - \bar{v}\sin\bar{\theta}\Delta\theta + \Delta v\cos\bar{\theta})ds}_{oscillation}, \tag{5.31}$$

where p_e is the external pressure to the beam. The integrals of dU_λ and dU_κ are not complicated and can be obtained easily as the following two integrals:

$$\begin{aligned} dU_\lambda &= \frac{1}{2}\int_0^L C_\lambda(\lambda-1)^2 dl && (5.32)\\ &= \int_0^L C_\lambda(\lambda-1)dl && (5.33)\\ &= \int_0^L C_\lambda((\bar{\lambda}+\Delta\lambda)-1)dl && (5.34)\\ &= \underbrace{\int_0^L C_\lambda(\bar{\lambda}-1)dl}_{meanflow} + \underbrace{\int_0^L C_\lambda\Delta\lambda dl}_{oscillation} && (5.35) \end{aligned}$$

$$\begin{aligned} dU_\kappa &= \frac{1}{2}\int_0^{S_0} C_\kappa\kappa^2 ds && (5.36)\\ &= \int_0^{S_0} C_\kappa\kappa ds && (5.37)\\ &= \int_0^{S_0} C_\kappa(\bar{\kappa}+\Delta\kappa)ds && (5.38)\\ &= \underbrace{\int_0^{S_0} C_\kappa\bar{\kappa}ds}_{meanflow} + \underbrace{\int_0^{S_0} C_\kappa\Delta\kappa ds}_{oscillation}. && (5.39) \end{aligned}$$

Then we are going to deduce the integral of F, whose integrating equation is the equation (5.7). For simplicity and clear, the equation (5.7) is written as $F = F2 - F1 - F3$. In all the kinetic terms, the q^2 is required, so the integral of q^2 is firstly derived for convenience.

$$\begin{aligned} q^2 &= u^2 + v^2 && (5.40)\\ &= (\bar{u}+\Delta u)^2 + (\bar{v}+\Delta v)^2 && (5.41)\\ &= \bar{u}^2 + 2\bar{u}\Delta u + \cancelto{0}{\Delta u^2} + \bar{v}^2 + 2\bar{v}\Delta v + \cancelto{0}{\Delta v^2} && (5.42)\\ &= \bar{u}^2 + 2\bar{u}\Delta u + \bar{v}^2 + 2\bar{v}\Delta v && (5.43) \end{aligned}$$

The equation (5.43) is the final expansion of q^2. After this, we continue integrating the term of $F2$.

$$
\begin{aligned}
F2 &= \frac{1}{2}\int_0^1 q^2 u|_{inlet} dy && (5.44)\\
&= \frac{1}{2}\int_0^1 (\bar{u}^2 + 2\bar{u}\Delta u + \bar{v}^2 + 2\bar{v}\Delta v)(\bar{u}|_{inlet} + \Delta u|_{inlet})dy && (5.45)\\
&= \frac{1}{2}\int_0^1 (\bar{u}^2 + \bar{v}^2)\bar{u}|_{inlet} dy + \frac{1}{2}\int_0^1 (\bar{u}^2 + \bar{v}^2)\Delta u|_{inlet} dy && (5.46)\\
&\quad + \frac{1}{2}\int_0^1 (2\bar{u}\Delta u + 2\bar{v}\Delta v)\bar{u}|_{inlet} dy + \cancelto{0}{\frac{1}{2}\int_0^1 (2\bar{u}\Delta u + 2\bar{v}\Delta v)\Delta u|_{inlet})dy} && (5.47)\\
&= \underbrace{\frac{1}{2}\int_0^1 (\bar{u}^2 + \bar{v}^2)\bar{u}|_{inlet} dy}_{meanflow} && (5.48)\\
&\quad \underbrace{+ \frac{1}{2}\int_0^1 (\bar{u}^2 + \bar{v}^2)\Delta u|_{inlet} dy + \frac{1}{2}\int_0^1 (2\bar{u}\Delta u + 2\bar{v}\Delta v)\bar{u}|_{inlet} dy}_{oscillation} && (5.49)
\end{aligned}
$$

The difference between $F1$ and $F2$ is that $F2$ indicates the entrance(upstream) to the channel, while $F1$ denotes the downstream of the channel. Hence, the integrating equation of $F2$ looks same as $F1$, whose difference is the velocity u here should be $u|_{outlet}$ and $u|_{inlet}$, respectively.

$$
\begin{aligned}
F1 &= \underbrace{\frac{1}{2}\int_0^1 (\bar{u}^2 + \bar{v}^2)\bar{u}|_{outlet} dy}_{meanflow} && (5.50)\\
&\quad \underbrace{+ \frac{1}{2}\int_0^1 (\bar{u}^2 + \bar{v}^2)\Delta u|_{outlet} dy + \frac{1}{2}\int_0^1 (2\bar{u}\Delta u + 2\bar{v}\Delta v)\bar{u}|_{outlet} dy}_{oscillation} && (5.51)
\end{aligned}
$$

$F3$ is the most complicated term among these, because both q^2 and u_n are involved in this integration. So

$$
\begin{aligned}
F3 &= \frac{1}{2}\int_0^{S_0} q^2 u_n ds && (5.52)\\
&= \frac{1}{2}\int_0^{S_0} (\bar{u}^2 + 2\bar{u}\Delta u + \bar{v}^2 + 2\bar{v}\Delta v) && (5.53)\\
&\quad ((-\bar{u}\sin\bar{\theta} + \bar{u}\cos\bar{\theta}\Delta\theta - \Delta u\sin\bar{\theta}) + (\bar{v}\cos\bar{\theta} - \bar{v}\sin\bar{\theta}\Delta\theta + \Delta v\cos\bar{\theta}))ds && (5.54)\\
&= \frac{1}{2}\int_0^{S_0} (\bar{u}^2 + \bar{v}^2)(-\bar{u}\sin\bar{\theta} + \bar{v}\cos\bar{\theta})ds && (5.55)\\
&\quad + \frac{1}{2}\int_0^{S_0} (\bar{u}^2 + \bar{v}^2)(\bar{u}\cos\bar{\theta}\Delta\theta - \Delta u\sin\bar{\theta} - \bar{v}\sin\bar{\theta}\Delta\theta + \Delta v\cos\bar{\theta})ds && (5.56)\\
&\quad + \frac{1}{2}\int_0^{S_0} (2\bar{u}\Delta u + 2\bar{v}\Delta v)(-\bar{u}\sin\bar{\theta} + \bar{v}\cos\bar{\theta})ds && (5.57)
\end{aligned}
$$

$$+\frac{1}{2}\int_0^{S_0}\cancelto{0}{(2\bar{u}\Delta u+2\bar{v}\Delta v)(\bar{u}\cos\bar{\theta}\Delta\theta-\Delta u\sin\bar{\theta}-\bar{v}\sin\bar{\theta}\Delta\theta+\Delta v\cos\bar{\theta})ds} \quad (5.58)$$

$$= \underbrace{\frac{1}{2}\int_0^{S_0}(\bar{u}^2+\bar{v}^2)(-\bar{u}\sin\bar{\theta}+\bar{v}\cos\bar{\theta})ds}_{meanflow} \quad (5.59)$$

$$+\underbrace{\frac{1}{2}\int_0^{S_0}(\bar{u}^2+\bar{v}^2)(\bar{u}\cos\bar{\theta}\Delta\theta-\Delta u\sin\bar{\theta}-\bar{v}\sin\bar{\theta}\Delta\theta+\Delta v\cos\bar{\theta})ds}_{oscillation} \quad (5.60)$$

$$+\underbrace{\frac{1}{2}\int_0^{S_0}(2\bar{u}\Delta u+2\bar{v}\Delta v)(-\bar{u}\sin\bar{\theta}+\bar{v}\cos\bar{\theta})ds}_{oscillation} \quad (5.61)$$

The several integrating equations above are the energy integrals of pressure-driven system after separating. After this separation, it might be better for us to investigate the oscillatory energy. Another most significant feature is that the whole energy solving process only takes a couple of minutes to complete.

5.4.2 Flow-driven system

As we have introduced the two specified boundary condition system in chapter 3, it is easy to find all the integrating equation of flow-driven system in accordance with the pressure-driven system.

In the flow-driven system, the velocity at the entrance of channel is fixed. $u|_{inlet}$ is a constant, and $v|_{inlet}$ is zero. Hence, we obtain that

$$P1 = \int_0^1 P_u u dy \quad (5.62)$$

$$= \int_0^1 (\bar{P}_u+\Delta p)\bar{u}dy \quad (5.63)$$

$$= \underbrace{\int_0^1 \bar{P}_u\bar{u}dy}_{meanflow}+\underbrace{\int_0^1 \Delta p\bar{u}dy}_{oscillation} \quad (5.64)$$

By substituting u_n into the integral of $P3$, we obtain that

$$P3 = \int_0^{S_0} P_e u_n ds \quad (5.65)$$

$$
\begin{aligned}
&= \int_0^{S_0} P_e((-\bar{u}\sin\bar{\theta} + \bar{u}\cos\bar{\theta}\Delta\theta - \Delta u\sin\bar{\theta}) && (5.66)\\
&\quad +(\bar{v}\cos\bar{\theta} - \bar{v}\sin\bar{\theta}\Delta\theta + \Delta v\cos\bar{\theta}))ds && (5.67)\\
&= \underbrace{\int_0^{S_0} P_e(-\bar{u}\sin\bar{\theta} + \bar{v}\cos\bar{\theta})ds}_{meanflow} && (5.68)\\
&\quad \underbrace{+\int_0^{S_0} P_e(\bar{u}\cos\bar{\theta}\Delta\theta - \Delta u\sin\bar{\theta} - \bar{v}\sin\bar{\theta}\Delta\theta + \Delta v\cos\bar{\theta})ds}_{oscillation}\,. && (5.69)
\end{aligned}
$$

Both the dU_λ and the dU_κ are as same as the pressure-driven system, so

$$
\begin{aligned}
dU_\lambda &= \frac{1}{2}\int_0^L C_\lambda(\lambda-1)^2 dl && (5.70)\\
&= \int_0^L C_\lambda(\lambda-1)dl && (5.71)\\
&= \int_0^L C_\lambda((\bar{\lambda}+\Delta\lambda)-1)dl && (5.72)\\
&= \underbrace{\int_0^L C_\lambda(\bar{\lambda}-1)dl}_{meanflow} + \underbrace{\int_0^L C_\lambda\Delta\lambda dl}_{oscillation} && (5.73)
\end{aligned}
$$

$$
\begin{aligned}
dU_\kappa &= \frac{1}{2}\int_0^{S_0} C_\kappa\kappa^2 ds && (5.74)\\
&= \int_0^{S_0} C_\kappa\kappa ds && (5.75)\\
&= \int_0^{S_0} C_\kappa(\bar{\kappa}+\Delta\kappa)ds && (5.76)\\
&= \underbrace{\int_0^{S_0} C_\kappa\bar{\kappa}ds}_{meanflow} + \underbrace{\int_0^{S_0} C_\kappa\Delta\kappa ds}_{oscillation} && (5.77)
\end{aligned}
$$

Because of the fixed parabolic flow at the entrance of the channel, there is no oscillatory energy in the $F2$ of flow-driven system.

$$
\begin{aligned}
F2 &= \frac{1}{2}\int_0^1 q^2 u|_{inlet} dy && (5.78)\\
&= \frac{1}{2}\int_0^1 (\bar{u}^2+\bar{v}^2)u|_{inlet} dy && (5.79)\\
&= \frac{1}{2}\int_0^1 (\bar{u}^2+0^2)\bar{u}dy && (5.80)\\
&= \underbrace{\frac{1}{2}\int_0^1 \bar{u}^3 dy}_{meanflow} + \underbrace{0}_{oscillation} && (5.81)
\end{aligned}
$$

Both the $F1$ and the $F3$ are same as pressure-driven system, so

$$F1 = \underbrace{\frac{1}{2}\int_0^1 (\bar{u}^2+\bar{v}^2)\bar{u}|_{outlet}dy}_{meanflow} \tag{5.82}$$

$$+\underbrace{\frac{1}{2}\int_0^1 (\bar{u}^2+\bar{v}^2)\Delta u|_{outlet}dy + \frac{1}{2}\int_0^1 (2\bar{u}\Delta u+2\bar{v}\Delta v)\bar{u}|_{outlet}dy}_{oscillation} \tag{5.83}$$

$$F3 = \frac{1}{2}\int_0^{S_0} q^2 u_n ds \tag{5.84}$$

$$= \frac{1}{2}\int_0^{S_0} (\bar{u}^2+2\bar{u}\Delta u+\bar{v}^2+2\bar{v}\Delta v) \tag{5.85}$$

$$((-\bar{u}\sin\bar{\theta}+\bar{u}\cos\bar{\theta}\Delta\theta-\Delta u\sin\bar{\theta})+(\bar{v}\cos\bar{\theta}-\bar{v}\sin\bar{\theta}\Delta\theta+\Delta v\cos\bar{\theta}))ds \tag{5.86}$$

$$= \frac{1}{2}\int_0^{S_0} (\bar{u}^2+\bar{v}^2)(-\bar{u}\sin\bar{\theta}+\bar{v}\cos\bar{\theta})ds \tag{5.87}$$

$$+\frac{1}{2}\int_0^{S_0} (\bar{u}^2+\bar{v}^2)(\bar{u}\cos\bar{\theta}\Delta\theta-\Delta u\sin\bar{\theta}-\bar{v}\sin\bar{\theta}\Delta\theta+\Delta v\cos\bar{\theta})ds \tag{5.88}$$

$$+\frac{1}{2}\int_0^{S_0} (2\bar{u}\Delta u+2\bar{v}\Delta v)(-\bar{u}\sin\bar{\theta}+\bar{v}\cos\bar{\theta})ds \tag{5.89}$$

$$\cancelto{0}{+\frac{1}{2}\int_0^{S_0} (2\bar{u}\Delta u+2\bar{v}\Delta v)(\bar{u}\cos\bar{\theta}\Delta\theta-\Delta u\sin\bar{\theta}-\bar{v}\sin\bar{\theta}\Delta\theta+\Delta v\cos\bar{\theta})ds} \tag{5.90}$$

$$= \underbrace{\frac{1}{2}\int_0^{S_0} (\bar{u}^2+\bar{v}^2)(-\bar{u}\sin\bar{\theta}+\bar{v}\cos\bar{\theta})ds}_{meanflow} \tag{5.91}$$

$$+\underbrace{\frac{1}{2}\int_0^{S_0} (\bar{u}^2+\bar{v}^2)(\bar{u}\cos\bar{\theta}\Delta\theta-\Delta u\sin\bar{\theta}-\bar{v}\sin\bar{\theta}\Delta\theta+\Delta v\cos\bar{\theta})ds}_{oscillation} \tag{5.92}$$

$$+\underbrace{\frac{1}{2}\int_0^{S_0} (2\bar{u}\Delta u+2\bar{v}\Delta v)(-\bar{u}\sin\bar{\theta}+\bar{v}\cos\bar{\theta})ds}_{oscillation} \tag{5.93}$$

All the integrating equations of flow-driven system are completed here, and these equations might help us achieve more interesting results and analyze the oscillatory energy.

5.5 Conclusion

In this chapter, the energy solutions are achieved by substituting the eigensolutions into the integrating equation of the energy. The energy solution from eigenpairs has five advantages rather than that of unsteady simulation. Afterwards, the effect of the different perturbation at a same neutral point is discussed, and it shows that the a large perturbation leads to a group of large energy solutions. Via a normalizing strategy, the energy solutions from eigenpairs are validated with the unsteady simulation by using the pressure-driven system. Then the energy in the flow-driven system is analyzed by the energy solution from eigenpairs. At last, with the energy solution from eigenpairs, the energy of mean flow and the self-excited oscillation are divided away from each other.

Chapter 6

Discussion & Conclusion

6.1 Achievement

One of the two main achievements in this thesis is the study of the eigensolvers. The Arnoldi iteration is used for solving the eigenvalue problems obtained from the linear stability analysis of the collapsible channel flows for the first time. Afterwards, the AR-G solver and the AR-F solver are studied, created and validated, and with these two, some new results were obtained in both of the systems. The neutral curve in pressure-driven system is verified and extended, while the neutral curves in flow-driven system are revised, and a new neutral curve in flow-driven system is calculated with a refined mesh. We also find that the AR-F solver requires very less CPU time and operation numbers than the other solvers. In addition, with the help of Python script language, we make the whole solving process to be automatic, and this increases the solving efficiency again. The energy analysis from the eigensolution is another important achievement, because it has five advantages over the energy analysis from the unsteady simulation. This can come after the solution of the eigenpairs, which can be made as another part of the automatic codes. The energy results from the unsteady simulation in the pressure-driven system are used to validate the new approach of the energy analysis. Then, the energy distribution of the flow-driven system is analyzed by using the energy solution from eigenpairs. And at the end of the thesis, we also introduce a new strategy to separate the oscillatory energy from energy due to the mean flow. This might be very useful for the future study.

6.2 Discussion

6.2.1 Eigensolvers

In order to investigate the small amplitude oscillations, the linear stability analysis of collapsible channel flows in FBM was first introduced by Luo et al. [84] in 2004. The finalization of the linear stability analysis is to solve a generalized eigenvalue problem, whose dimension of the matrix depends on the number of degrees of freedom in the current mesh. During the investigation of the linear stability in the flow-driven system, the traditional QZ algorithm became increasingly inefficient along with the rising number of degree of freedom. More seriously, the QZ algorithm cannot work at all, when Liu et al. studied the stability of pressure-driven system with a very refined mesh(the number of degree of freedom is 55416). Even if the QZ algorithm can work, it would take several days to finish one computation. Consequently, it is so necessary to employ a much more efficient eigensolver for the further study.

The stability analysis of collapsible channel flows has been a challenge numerically due to the large matrix size and the asymmetric structure. Previous studies used the traditional QZ Algorithm for these problems, and solved for the whole set of eigensolutions. However, this approach required extremely large memory storage and CPU time. The QZ Algorithm also ceases to work when the demand for the mesh size is high. The pressure-driven system, in particular, can generate very thin boundary layers upstream the elastic section. If the flow details are not resolved, then the eigensolutions computed are either inaccurate or the eigensolvers fail to converge. Therefore, the advantages of the AR-F solver is particularly prominent for the pressure-driven system, and we are able to produce new neutral points with the AR-F solver.

The flow-driven system does not require such a refined mesh due to the parabolic entry flow profile used, which does not include very thin boundary layers upstream. Therefore the neutral curve identified by Luo et al. [85] using a relatively coarse mesh seems to be a reasonable approximation. A coarse mesh like this could not even produce converged solutions when used for the pressure-driven system. However, even for the flow-driven system, the neutral curve can have a small shift in the upper branch when a much refined mesh is used.

The AR-F solver is the most efficient and fastest of all the solvers studied, with a rate of CPU time between $O(n)$ and $O(n^2)$, and can solve for eigenvalue problems of very large size. Compared with the AR-F solver, the AR-G solver has a marginal advantages over the QZ Algorithm because it still requires solving the inversion of the global FE matrix. As a consequence, the rate of CPU time is only slightly shorter than $O(n^3)$ of the QZ Algorithm.

The disadvantage of the AR-F solver, however, is that we have to select the optimal group of parameters first in order to locate the eigenvalues required. Since the initial vector is generated randomly, it can only guarantee that the first eigenpair (the most unstable one in this case) is secured. It cannot promise to find other eigenpairs, or indeed if any other eigenpairs are found, these contain all the solutions in the right order. The same disadvantage applies to the AR-G solver. In this sense, the QZ Algorithm is the most reliable one in terms of determining all the required eigenpairs, and hence is recommended for small sized eigenvalue problems. Another issue we should remark is that the process of the frontal solver in AR-F is so complicated that it requires that so much boring work in adapting the codes.

In all the computations, the tolerance for the Arnoldi iteration is set to be 1×10^{-16}, which is the default value used in ARPACK. While this may be unnecessary for the coarse meshes used, we kept this changed in order to make fair comparisons of the computational times used by all the solvers.

Finally, we remark that all computations are performed without using parallel algorithms, therefore the pros and cons of these solvers are only applicable in the series computations. We must also acknowledge that linear stability analysis used here can be misleading in certain fluid dynamics problems, and one gains better insight from also understanding the potential of those eigenvalues as illustrated in [143].

6.2.2 Energy analysis

The energy distribution in collapsible channel flows are the central focus of the researchers [141, 80, 146]. Owing to the increment of each degree of freedom can be obtained by the eigenpairs with the equation (5.12), the seft-excited oscillation can be shown by the eigenpairs, and the energy solutions can be obtained with the eight variables. And the energy solutions obtained from eigenpairs have five advantages than the unsteady simulation. The first one is the computational time. The energy solution worked out by eigenpairs takes very less time than the unsteady simulation. The second is that the oscillation can be guaranteed to be very close to the neutral point. The third is that the energy solution from eigenpairs can get rid of the influence from the other modes. The fourth is that this can be used to analyze any point. At last, with the eigenpairs, it is easy to separate the energy solution due to the oscillation from that due to the mean flow.

The previous studies to the energy distribution in pressure-driven system [80] were based on the unsteady simulation, where large perturbation was applied. However, in the linear stability analysis, an infinitesimal perturbation is applied to the steady solution. These two different perturbations would lead to different energy solution. Hence, a normalization is necessary for the validating work. The direction of the phase plot cannot be matched for some neutral points. The reason might be from the large perturbation and infinitesimal perturbation. Probably, the direction is up to the size of the perturbation.

In the pressure-driven system, a small loop above the main loop was observed by Liu et al., but this loop has never been observed in the results of the energy solution from the eigenpairs. The prominent difference between the unsteady simulation and the linear stability analysis is the size of perturbation. To a same neutral point, the percentage of perturbation will cause different energy results, which we have discussed in Chapter 5. And we have tested that there would be a small loop above the main loop when the coefficient of normalization is large enough, which also shows that the small loop would appear in the case of large perturbation only. Hence, there would be no small loop occurred in the results of energy solution from eigenpairs.

Jensen et al. predicated that the ratio of D_S/F would be close to 2/3 for the undeformed configuration, and Liu et al. observed that ratio once at the top point of his five operating points. But we cannot achieve that 2/3 in the energy results from eigenpairs, even if C_λ is so large that the displacement of elastic section is very small. Probably, the infinitesimal perturbation in the linear stability analysis only leads to the self-excited oscillation with a very small amplitude, which is so different from large perturbation used by Liu et al. in the unsteady simulation.

The energy distribution in the flow-driven system shows great agreement with the results of four Mode-2 points obtained by Liu et al. We also obtained the energy results of some other operating points. There is no direction change in the P1-D phase plot, and this was also observed by Liu et al. in their studies. So we are able to continue the study of energy distribution with the approach of eigensolution. After we analyzed the energy distribution of Mode-3 and Mode-4. In particular, the P1-D plot of Mode-4 is different from the other modes, and we would calculate some neighbor points to confirm it.

The energy result from the eigenpairs depends on the eigensolver very much. If the eigensolver fails to find a converged solution or cannot locate a right result, there would be no reliable energy results. As we have three eigensolvers(QZ, AR-G and AR-F) for the project of the collapsible channel flows, we are able to analyze the energy distribution much more accurately and faster by employing these three solvers properly.

The strategy of the energy separation can make us to know the oscillatory energy much more clearly. For example, in the flow-driven system, the integral of F2 is a constant because of the parabolic entry flow profile used. As we have eight variables in the whole computation, from the integrals of each term, we can indicate which variable would have the main contribution to the energy results.

6.3 Limitation

First, except the AR-G solver (this is an intermediate product to AR-F solver), both the QZ Algorithm and the AR-F solver in collapsible channel flows have their own limitations and disadvantages. The QZ Algorithm is a direct method, which solves the whole set of the eigensolutions. So the QZ Algorithm is considered as the most reliable eigensolver for us now. But the QZ Algorithm fails to work when the size of matrices is large. Even if it can work, it would be very expensive because it would take quite a long time to finish the computation and require a very large amount of memory storage. The AR-F solver is the fastest one here, but at the beginning it needs some extra work for it. We have to select a group of suitable parameters in order to locate the right eigenvalues, because of the sparsity of the matrices and the dimension of the projecting subspace. The AR-F solver in collapsible channel flows can only secure the first eigenpair (the most unstable one in this case) for the majority of meshes, only for some cases, it can make sure the second or the third ones. Hence, the QZ Algorithm is used for the cases, where the size of matrices is small and the second or third eigenpairs are required; while the AR-F solver is for the large dimensional problems, for example, the case where C_λ is very large in the pressure-driven system.

Second, the AR-F solver in collapsible channel flows employs the original ARPACK software package of the Implicitly Restarted Arnoldi method for the Arnoldi solver, which is not a parallel codes. Hence, the current AR-F solver cannot shows the true high performance on the parallel machine completely.

Third, the frontal process is so complicated. Although it is an old solver, there are not many packages for it. Sometimes it requires the researchers to adapt the codes for their own projects, so it needs the patience and the intelligence from the researchers, and some time as well. The subroutines in ARPACK are written in FORTRAN language, which is very popular in the scientific computation, because it is much faster and simper than the others. If the AR-F solver is employed for some other projects, some work on the interface of FORTRAN language is necessary.

Fifth, in order to compare the results with those of unsteady simulation, the energy results from eigenpairs still need a necessary normalization of the eigenpairs in the process. This means that the energy result might not indicate the true results, but the relative results. The idea of the energy results from eigenpairs came out for a very short time, and we do not have enough results for the study. Hence, there is still some work to be done and some results to be analyzed.

6.4 Future Work

As we have discussed in the section of 4.5.2, besides the shift-and-invert transformation, there is another spectral transformation, whose name is the Generalized Cayley transformation, which has already been employed for some eigenvalue problems. In the Generalized Cayley transformation, not only one shift is need, but also another parameter is necessary. This still needs some more research work for using it in the collapsible channel flows. The AR-F solver can work the eigenpairs with the first largest real part very quickly, but sometimes the second or the third ones are required as well. Consequently, an orthogonalization strategy has been applied to the AR-F solver, and it is successful for some matrices. Hence, this strategy might be used for the eigenvalue problems in collapsible channel flows. While studying the eigensolvers for the asymmetric generalized eigenvalue problems, we found that the pre-conditioning work is very necessary for finding the solution. And for our eigenvalue problems, no pre-conditioning work has been used before. Probably, some pre-conditioning work might make our eigenvalue problem to be worked out easily. And the parallel version of ARPACK, whose name is P_ARPACK has already shown its parallel technique in the computational fluid mechanics. So if the parallel technique is employed in the AR-F solver, its efficiency will definitely increase again.

The energetics of the collapsible channel flows attracts the interest of many researchers in the past years. And with the efficient eigensolvers, the energy results can be obtained soon after the eigensolutions. These would be very helpful for us to disclose the energy movement in the collapsible channel flows. We are going to continue analyzing the energy results of Mode-3 and Mode-4 neutral points in the flow-driven system. And for both of the systems, the energy movement between the stable point and the unstable point is another topic of research. And also, after making the strategy of separation work, we might try to find which variable would have the prominent contribution to the energy results in the eight variables.

6.5 Conclusion

The linear stability analysis of the collapsible channel flows in the FBM is to solve large sparse asymmetric generalized eigenvalue problems. The stiffness matrix is sparse, asymmetric, nonsingular, while the mass matrix is sparse, asymmetric and singular. And the eigenvalues with the largest real parts are of our interest. The traditional QZ algorithm takes lots of CPU time to finish, and it cannot work at all when the matrix is so large, because it needs the explicit expression of matrices. Taking account of the structure of the matrix, the Arnoldi method with a frontal solver is studied and created for the project. With the new AR-F eigensolver, the neutral curves of two boundary conditions [85, 80] are revised and extended, respectively. Via changing the length of three sections and setting some specified parameters, the Mode-1 unstable point in the flow-driven system can be observed in a sparse mesh by using the fluid diffusion in the system. After researching the new eigensolvers, the energy solutions can also be achieved by the eigenpairs. This approach is much faster than the unsteady simulation. The energy solutions from the eigenpairs are validated with those from the unsteady simulation in press-driven system after the necessary normalization. Afterwards, this is used to analyze the energy distribution in flow-driven system. At the end, in the energy analysis from the eigensolutions, a new strategy to separate the oscillatory energy from the mean-flow energy is presented and deduced, which would be helpful to disclose the energetics of the system.

Statement

The Chapter 4 is written in a paper submitted to Computers and Structures. The Chapter 5 is being organized in a paper as soon as the result of energy division is finished. A paper about the collapsible channel flows in FBM is submitted to the 2nd Symposium on Fluid-Structure-Sound Interactions and Control, and a talk will be given. And during the PhD study, I attended British Applied Mathematics Colloquium and some other workshops and gave talks.

Bibliography

[1] T. Aoki and D.N. Ku. Collapse of diseased arteries with eccentric cross section. *Journal of biomechanics*, 26(2):133–142, 1993.

[2] W.E. Arnoldi. The principle of minimized iterations in the solution of the matrix eigenvalue problem. *Quart. Appl. Math*, 9(1):17–29, 1951.

[3] T. Auckenthaler, V. Blum, H. J. Bungartz, T. Huckle, R. Johanni, L. Krmer, B. Lang, H. Lederer, and P. R. Willems. Parallel solution of partial symmetric eigenvalue problems from electronic structure calculations. *Parallel Computing*, 37(12):783, 2011.

[4] Z. Bai. Error analysis of the lanczos algorithm for the nonsymmetric eigenvalue problem. *Mathematics of Computation*, 62(205):209–226, 1994.

[5] Z. Bai, D. Day, and Q. Ye. Able: an adaptive block lanczos method for non-hermitian eigenvalue problems. *SIAM Journal on Matrix Analysis and Applications*, 20(4):1060–1082, 1999.

[6] TS Balint and AD Lucey. Instability of a cantilevered flexible plate in viscous channel flow. *Journal of Fluids and Structures*, 20(7):893–912, 2005.

[7] Klaus-Jurgen Bathe and Edward L. Wilson. Solution methods for eigenvalue problems in structural mechanics. *International Journal for Numerical Methods in Engineering*, 6:213–226, 1973.

[8] G.S. Berke, D.C. Green, M.E. Smith, D.P. Arnstein, V. Honrubia, M. Natividad, and W.A. Conrad. Experimental evidence in the in vivo canine for the collapsible tube model of phonation. *The Journal of the Acoustical Society of America*, 89:1358, 1991.

[9] CD Bertram and TJ Pedley. A mathematical model of unsteady collapsible tube behaviour. *Journal of Biomechanics*, 15(1):39–50, 1982.

[10] CD Bertram and TJ Pedley. Steady and unsteady separation in an approximately two-dimensional indented channel. *Journal of Fluid Mechanics*, 130(1):315–345, 1983.

[11] CD Bertram and CJ Raymond. Measurements of wave speed and compliance in a collapsible tube during self-excited oscillations: a test of the choking hypothesis. *Medical and Biological Engineering and Computing*, 29(5):493–500, 1991.

[12] CD Bertram, CJ Raymond, and TJ Pedley. Mapping of instabilities for flow through collapsed tubes of differing length. *Journal of Fluids and Structures*, 4(2):125–153, 1990.

[13] CD Bertram, CJ Raymond, and TJ Pedley. Application of nonlinear dynamics concepts to the analysis of self-excited oscillations of a collapsible tube conveying a fluid. *Journal of Fluids and Structures*, 5(4):391–426, 1991.

[14] R.L. Binns and D.N. Ku. Effect of stenosis on wall motion. a possible mechanism of stroke and transient ischemic attack. *Arteriosclerosis, Thrombosis, and Vascular Biology*, 9(6):842–847, 1989.

[15] ZX Cai and XY Luo. A fluid–beam model for flow in a collapsible channel. *Journal of fluids and structures*, 17(1):125–146, 2003.

[16] C. Cancelli and TJ Pedley. Separated-flow model for collapsible-tube oscillations. *Journal of Fluid Mechanics*, 157:375–404, 1985.

[17] Zhihao Cao. *Numerical Linear Algebra*. Fudan University Press, 1996.

[18] M. Clint and A. Jennings. A simultaneous iteration method for the unsymmetric eigenvalue problem. *IMA Journal of Applied Mathematics*, 8(1):111–121, 1971.

[19] W.A. Conrad. Pressure-flow relationships in collapsible tubes. *Biomedical Engineering, IEEE Transactions on*, (4):284–295, 1969.

[20] J. Cullum and R. A. Willoughby. *Large Scale Eigenvalue Problems*. Elsevier Science., 1986.

[21] J. Cullum and R.A. Willoughby. A practical procedure for computing eigenvalues of large sparse nonsymmetric matrices. *Large Scale Eigenvalue Problems*, 127:193–240, 1986.

[22] M.S. Fee, B. Shraiman, B. Pesaran, and P.P. Mitra. The role of nonlinear dynamics of the syrinx in the vocalizations of a songbird. *Nature*, 395(6697):67–71, 1998.

[23] Paulo J. S. G. Ferreira. The eigenvalues of matirces that occur in certain interpolation problems. *IEEE Transaction on Signal Processing.*, 45:2115–2120, 1997.

[24] J.G.F. Francis. The qr transformation a unitary analogue to the lr transformation-part 1. *The Computer Journal*, 4(3):265–271, 1961.

[25] J.G.F. Francis. The qr transformationpart 2. *The Computer Journal*, 4(4):332–345, 1962.

[26] R.W. Freund. Quasi-kernel polynomials and their use in non-hermitian matrix iterations. *Journal of Computational and Applied Mathematics*, 43(1):135–158, 1992.

[27] R.W. Freund, M.H. Gutknecht, and N.M. Nachtigal. An implementation of the look-ahead lanczos algorithm for non-hermitian matrices. *SIAM Journal on Scientific Computing*, 14(1):137–158, 1993.

[28] B.S. Garbow. Algorithm 535: The qz algorithm to solve the generalized eigenvalue problem for complex matrices [f2]. *ACM Transactions on Mathematical Software (TOMS)*, 4(4):404–410, 1978.

[29] N. Gavriely, Y. Palti, G. Alroy, and JB Grotberg. Measurement and theory of wheezing breath sounds. *Journal of Applied Physiology*, 57(2):481–492, 1984.

[30] N. Gavriely, T.R. Shee, D.W. Cugell, and J.B. Grotberg. Flutter in flow-limited collapsible tubes: a mechanism for generation of wheezes. *Journal of Applied Physiology*, 66(5):2251–2261, 1989.

[31] S.A. Gershgorin. Uber die abgrenzung der eigenwerte einer matrix. *Izv. Akad. Nauk SSSR, Ser. Fiz.-Mat.*, (6):749–754, 1931.

[32] G.H. Golub and R. Underwood. The block lanczos method for computing eigenvalues. *Mathematical software*, 3:361–377, 1977.

[33] G.H. Golub and C.F. Van Loan. *Matrix computations*, volume 3. Johns Hopkins University Press, 1996.

[34] A. Greenbaum. *Iterative methods for solving linear systems*, volume 17. Society for Industrial Mathematics, 1987.

[35] DJ Griffiths. Urethral elasticity and micturition hydrodynamics in females. *Medical and Biological Engineering and Computing*, 7(2):201–215, 1969.

[36] DJ Griffiths. Hydrodynamics of male micturitioni theory of steady flow through elastic-walled tubes. *Medical and Biological Engineering and Computing*, 9(6):581–588, 1971.

[37] J.B. Grotberg and N. Gavriely. Flutter in collapsible tubes: a theoretical model of wheezes. *Journal of Applied Physiology*, 66(5):2262–2273, 1989.

[38] S. Hayashi, T. Hayase, and H. Kawamura. Numerical analysis for stability and self-excited oscillation in collapsible tube flow. *Journal of biomechanical engineering*, 120(4):468–475, 1998.

[39] A.L. Hazel and M. Heil. Steady finite-reynolds-number flows in three-dimensional collapsible tubes. *Journal of Fluid Mechanics*, 486:79–103, 2003.

[40] M. Heil. The stability of cylindrical shells conveying viscous flow. *Journal of Fluids and Structures*, 10(2):173, 1996.

[41] M. Heil. Stokes flow in collapsible tubes: computation and experiment. *Journal of Fluid Mechanics*, 353(1):285–312, 1997.

[42] M. Heil and J. Boyle. Self-excited oscillations in three-dimensional collapsible tubes: simulating their onset and large-amplitude oscillations. *Journal of Fluid Mechanics*, 652:405, 2010.

[43] M. Heil and TJ Pedley. Large post-buckling deformations of cylindrical shells conveying viscous flow. *Journal of Fluids and Structures*, 10(6):565–600, 1996.

[44] M. Heil and S.L. Waters. How rapidly oscillating collapsible tubes extract energy from a viscous mean flow. *Journal of Fluid Mechanics*, 601(1):199–227, 2008.

[45] P. Hood. Frontal solution program for unsymmetric matrices. *International Journal for Numerical Methods in Engineering*, 10(2):379–399, 1976.

[46] T. Ikeda, Y. Matsuzaki, et al. A one-dimensional unsteady separable and reattachable flow model for collapsible tube-flow analysis. *Journal of biomechanical engineering*, 121(2):153, 1999.

[47] B.M. Irons. A frontal solution program for finite element analysis. *International Journal for Numerical Methods in Engineering*, 2(1):5–32, 1970.

[48] C.G.J. Jacobi. Uber ein leichtes verfahren, die in der theorie der s acularst orungen vorkommenden gleichungen numerisch aufzul osen. *J. reine angew. Math*, 30:51–94, 1846.

[49] A. Jennings and WJ Stewart. Simultaneous iteration for partial eigensolution of real matrices. *IMA Journal of Applied Mathematics*, 15(3):351–361, 1975.

[50] OE Jensen. Instabilities of flow in a collapsed tube. *J. Fluid Mech*, 220:623–659, 1990.

[51] OE Jensen. Chaotic oscillations in a simple collapsible-tube model. *Journal of biomechanical engineering*, 114(1):55–59, 1992.

[52] O.E. Jensen. Instabilities of ows through deformable tubes and channels. *Int. Congr. Theor. Applied Mech*, 481:235–268, 2008.

[53] O.E. Jensen and M. Heil. High-frequency self-excited oscillations in a collapsible-channel flow. *Journal of Fluid Mechanics*, 481:235–268, 2003.

[54] OE Jensen and TJ Pedley. The existence of steady flow in a collapsed tube. *J. Fluid Mech*, 206:339–374, 1989.

[55] Z. Jia. *Some numerical methods for large unsymmetric eigenproblems.* PhD thesis, Ph. D. thesis, University of Bielefeld, 1994.

[56] Z. Jia. The convergence of generalized lanczos methods for large unsymmetric eigenproblems. *SIAM journal on matrix analysis and applications*, 16(3):843–862, 1995.

[57] Z. Jia. Refined iterative algorithms based on arnoldi's process for large unsymmetric eigenproblems. *Linear algebra and its applications*, 259:1–23, 1997.

[58] Z Jia. Composite orthogonal projection methods for large matrix eigenproblems. *Science in China Series A: Mathematics*, 42(6):577–585, 1999.

[59] Z. Jia. Polynomial characterizations of the approximate eigenvectors by the refined arnoldi method and an implicitly restarted refined arnoldi algorithm. *Linear algebra and its applications*, 287(1):191–214, 1999.

[60] Z. Jia. A refined subspace iteration algorithm for large sparse eigenproblems. *Applied numerical mathematics*, 32(1):35–52, 2000.

[61] Z. Jia. The refined harmonic arnoldi method and an implicitly restarted refined algorithm for computing interior eigenpairs of large matrices. *Applied numerical mathematics*, 42(4):489–512, 2002.

[62] Z. Jia and D. Niu. An implicitly restarted refined bidiagonalization lanczos method for computing a partial singular value decomposition. *SIAM journal on matrix analysis and applications*, 25(1):246–265, 2003.

[63] Z. Jia and G.W. Stewart. On the convergence of ritz values, ritz vectors, and refined ritz vectors. 1999.

[64] Z. Jia and Y. Zhang. A refined shift-and-invert arnoldi algorithm for large unsymmetric generalized eigenproblems. *Computers & Mathematics with Applications*, 44(8):1117–1127, 2002.

[65] T. J. Garratt K. A. Cliffe and A. Spence. Eigenvalues of block matrices arising from problems in fluid mechanics. *Siam J. Matrix Anal. Appl.*, 15:1310–1318, 1994.

[66] Andy J. Keane and Prasanth B. Nair. *Computational Approaches for Aerospace Design The Pursuit of Excellence.* John Wiley and Sons. Ltd., 2005.

[67] I. Kececioglu, ME McClurken, D. Kamm, and AH Shapiro. Steady supercritical flow in collapsible tubes. part 2 theoretical studies. *Journal of Fluid Mechanics*, 109:391–415, 1981.

[68] I. KECECIOGLU, E. MICHAEL, and R.D. McCLURKEN. Steady, supercritical flow in collapsible tubes. part 1. experimental observations. *J. Fluid Mech*, 109:367–389, 1981.

[69] FP Knowlton and EH Starling. The influence of variations in temperature and blood-pressure on the performance of the isolated mammalian heart. *The Journal of physiology*, 44(3):206–219, 1912.

[70] D.N. Ku. Blood flow in arteries. *Annual Review of Fluid Mechanics*, 29(1):399–434, 1997.

[71] V.N. Kublanovskaya. On some algorithms for the solution of the complete eigenvalue problem. *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*, 1(4):555–570, 1961.

[72] C. Lanczos. *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*. United States Governm. Press Office, 1950.

[73] C. Lanczos. Solution of systems of linear equations by minimized iterations. *J. Res. Nat. Bur. Standards*, 49(1):33–53, 1952.

[74] RB Lehoucq, K. Meerbergen, et al. Using generalized cayley transformations within an inexact rational krylov sequence method. *SIAM journal on matrix analysis and applications*, 20:131–148, 1998.

[75] R.B. Lehoucq and A.G. Salinger. Large-scale eigenvalue calculations for stability analysis of steady flows on massively parallel computers. *International Journal for Numerical Methods in Fluids*, 36(3):309–327, 2001.

[76] RB Lehoucq and JA Scott. An evaluation of software for computing eigenvalues of sparse nonsymmetric matrices. *Preprint MCS-P547*, 1195, 1996.

[77] RB Lehoucq, JA Scott, and Council for the Central Laboratory of the Research Councils (Great Britain). *Implicitly restarted Arnoldi methods and eigenvalues of the discretized Navier Stokes equations*. Citeseer, 1997.

[78] R.B. Lehoucq and D.C. Sorensen. Deflation techniques for an implicitly restarted arnoldi iteration. *SIAM Journal on Matrix Analysis and Applications*, 17(4):789–821, 1996.

[79] R.B. Lehoucq, D.C. Sorensen, and C. Yang. *ARPACK users' guide: solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods*, volume 6. Siam, 1998.

[80] HF Liu, XY Luo, and ZX Cai. Stability and energy budget of pressure-driven collapsible channel flows. *Journal of Fluid Mechanics*, in press:1–23, 2012.

[81] TW Lowe and TJ Pedley. Computation of stokes flow in a channel with a collapsible segment. *Journal of Fluids and Structures*, 9(8):885–905, 1995.

[82] X. Luo, B. Calderhead, H. Liu, and W. Li. On the initial configurations of collapsible channel flow. *Computers & structures*, 85(11):977–987, 2007.

[83] XY Luo. Steady and unsteady flows in collapsible channels. *Advances in Biomechanics*, pages 192–199, 2001.

[84] XY Luo and ZX Cai. Effects of wall stiffness on the linear stability of flow in an elastic channel. In *Proceedings of the eighth international conference on flow-induced vibrations, FIV2004*, volume 2, pages 167–70, 2004.

[85] XY Luo, ZX Cai, WG Li, and TJ Pedley. The cascade structure of linear instability in collapsible channel flows. *Journal of Fluid Mechanics*, 600(1):45–76, 2008.

[86] XY Luo and TJ Pedley. A numerical simulation of steady flow in a 2-d collapsible channel. *Journal of Fluids and Structures*, 9(2):149–174, 1995.

[87] XY Luo and TJ Pedley. A numerical simulation of unsteady flow in a two-dimensional collapsible channel. *Journal of Fluid Mechanics*, 314:191–226, 1996.

[88] XY Luo and TJ Pedley. The effects of wall inertia on flow in a two-dimensional collapsible channel. *Journal of Fluid Mechanics*, 363:253–280, 1998.

[89] XY Luo and TJ Pedley. Multiple solutions and flow limitation in collapsible channel flows. *Journal of Fluid Mechanics*, 420(1):301–324, 2000.

[90] A. Marzo, XY Luo, and CD Bertram. Three-dimensional collapse and steady flow in thick-walled flexible tubes. *Journal of fluids and structures*, 20(6):817–835, 2005.

[91] George Mathew and V. U. Reddy. A quasi-newton adaptive algorithm for generalized symmetric eigenvalues problem. *IEEE Transaction on Signal Processing.*, 44:2413–2422, 1996.

[92] L.R.S. Matrices. Note the iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices. *Journal of Computational Physics*, 17:87–94, 1975.

[93] Y. Matsuzaki, T. Ikeda, T. Kitagawa, and S. Sakata. Analysis of flow in a two-dimensional collapsible channel using universal tube law. *Journal of biomechanical engineering*, 116(4):469–476, 1994.

[94] Y. Matsuzaki, T. Matsumoto, et al. Flow in a two-dimensional collapsible channel with rigid inlet and outlet. *Journal of biomechanical engineering*, 111(3):180, 1989.

[95] K. Meerbergen and D. Roose. Matrix transformations for computing rightmost eigenvalues of large sparse non-symmetric eigenvalue problems. *IMA Journal of Numerical Analysis*, 16(3):297–346, 1996.

[96] K. Meerbergen and D. Roose. The restarted arnoldi method applied to iterative linear system solvers for the computation of rightmost eigenvalues. *SIAM Journal on Matrix Analysis and Applications*, 18(1):1–20, 1997.

[97] M. Sh. Misrikhanov and V. N. Ryabchenko. The quadratic eigenvalue problem in elctric power systems. *Journal of Automation and Remote Control.*, 67:24–47, 2006.

[98] C.B. Moler and G.W. Stewart. An algorithm for generalized matrix eigenvalue problems. *SIAM Journal on Numerical Analysis*, 10(2):241–256, 1973.

[99] R.B. Morgan. Computing interior eigenvalues of large matrices. *Linear Algebra and its Applications*, 154:289–309, 1991.

[100] R.B. Morgan. A restarted gmres method augmented with eigenvectors. *SIAM Journal on Matrix Analysis and Applications*, 16(4):1154–1171, 1995.

[101] R.B. Morgan. On restarting the arnoldi method for large non-symmetric eigenvalue problems. *Mathematics of Computation*, 65(215):1213–1230, 1996.

[102] R.B. Morgan. Implicitly restarted gmres and arnoldi methods for nonsymmetric systems of equations. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1112–1135, 2000.

[103] R.B. Morgan. Gmres with deflated restarting. *SIAM Journal on Scientific Computing*, 24(1):20–37, 2002.

[104] R.B. Morgan and D.S. Scott. Generalizations of davidson's method for computing eigenvalues of sparse symmetric matrices. *SIAM Journal on Scientific and Statistical Computing*, 7(3):817–825, 1986.

[105] R.B. Morgan and M. Zeng. Harmonic projection methods for large non-symmetric eigenvalue problems. *Numerical linear algebra with applications*, 5(1):33–55, 1998.

[106] J. Von Neumann. A model of general equilibrium. *Review of Economic Studies.*, 13:1–9, 1945.

[107] C.C. Paige. *The computation of eigenvalues and eigenvectors of very large sparse matrices.* PhD thesis, University of London, 1971.

[108] C.C. Paige, B.N. Parlett, and H.A. Van der Vorst. Approximate solutions and eigenvalue bounds from krylov subspaces. *Numerical linear algebra with applications*, 2(2):115–133, 1995.

[109] B.N. Parlett. *The symmetric eigenvalue problem*, volume 7. SIAM, 1980.

[110] B.N. Parlett and D.S. Scott. The lanczos algorithm with selective orthogonalization. *Math. Comp*, 33(145):217–238, 1979.

[111] B.N. Parlett, D.R. Taylor, and Z.A. Liu. A look-ahead lanczos algorithm for unsymmetric matrices. *Mathematics of computation*, 44(169):105–124, 1985.

[112] TJ Pedley. The fluid mechanics of large blood vessels. 1980. *CUP, Cambridge*, 1980.

[113] TJ Pedley. Longitudinal tension variation in collapsible channels: a new mechanism for the breakdown of steady flow. *Journal of biomechanical engineering*, 114(1):60, 1992.

[114] S. Pushpavanam. *Mathematical Methods in Chemical Engineering.* PHI Learning Pvt. Ltd., 2004.

[115] T. Rapcsak. Some optimization problems in multivariate statistic. *Journal of Global Optimization*, 28:217–228, 2004.

[116] M.P. Rast. Simultaneous solution of the navier-stokes and elastic membrane equations by a finite element method. *International journal for numerical methods in fluids*, 19(12):1115–1135, 1994.

[117] John G. Lewis Roger G. Grimes and Horst D. Simon. Eigenvalue problems and algorithms in structural engineering. *North-Holland Mathematics Studies*, 127:81–93, 1986.

[118] A. Ruhe. Rational krylov sequence methods for eigenvalue computation. *Linear Algebra and its Applications*, 58:391–405, 1984.

[119] A. Ruhe. The rational krylov algorithm for nonsymmetric eigenvalue problems. iii: Complex shifts for real matrices. *BIT Numerical Mathematics*, 34(1):165–176, 1994.

[120] A. Ruhe. Rational krylov algorithms for nonsymmetric eigenvalue problems. ii. matrix pairs. *Linear algebra and its Applications*, 197:283–295, 1994.

[121] A. Ruhe. Rational krylov: A practical algorithm for large sparse nonsymmetric matrix pencils. *SIAM Journal on Scientific Computing*, 19(5):1535–1551, 1998.

[122] A. Ruhe. Rational Krylov algorithms for nonsymmetric eigenvalue problems. *Recent Advances in Iterative Methods, IMA Volumes in Mathematics and its Applications*, 60:149–164, Springer-Verlag, New York, 1994.

[123] K.J. Ruschak. A method for incorporating free boundaries with surface tension in finite element fluid-flow simulators. *International Journal for Numerical Methods in Engineering*, 15(5):639–648, 1980.

[124] H. Rutishauser. Computational aspects of fl bauer's simultaneous iteration method. *Numerische Mathematik*, 13(1):4–13, 1969.

[125] H. Rutishauser. Simultaneous iteration method for symmetric matrices. *Numerische Mathematik*, 16(3):205–223, 1970.

[126] Y. Saad. Chebyshev acceleration techniques for solving nonsymmetric eigenvalue problems. *Mathematics of Computation*, 42(166):567–588, 1984.

[127] Y. Saad. *Numerical methods for large eigenvalue problems*, volume 158. SIAM, 1992.

[128] Y. Saad. Analysis of augmented krylov subspace methods. *SIAM Journal on Matrix Analysis and Applications*, 18(2):435–449, 1997.

[129] Y. Saad and Y. Saad. *Iterative methods for sparse linear systems*, volume 620. PWS publishing company Boston, 1996.

[130] M. Sadkane. Block-arnoldi and davidson methods for unsymmetric large eigenvalue problems. *Numerische Mathematik*, 64(1):195–211, 1993.

[131] D.W. Schoendorfer and A.H. Shapiro. The collapsible tube as a prosthetic vocal source. In *Proc. San Diego Biomed. Symp*, volume 16, pages 349–356, 1977.

[132] T. C. Scott, R. A. Moore, M. B. Monagan, G. J. Fee, and E. R. Vrscay. Perturbative solutions of quantum mechanical problems by symbolic computation. *Journal of Computational Physics*, 87(2):366–395, 1990.

[133] A.H. Shapiro. Steady flow in collapsible tubes. *Journal of Biomechanical Engineering*, 99:126, 1977.

[134] H.D. Simon. The lanczos algorithm with partial reorthogonalization. *Mathematics of Computation*, 42(165):115–142, 1984.

[135] G.L.G. Sleijpen, A.G.L. Booten, D.R. Fokkema, and H.A. Van der Vorst. Jacobi-davidson type methods for generalized eigenproblems and polynomial eigenproblems. *BIT Numerical Mathematics*, 36(3):595–633, 1996.

[136] G.L.G. Sleijpen and H.A. Van der Vorst. A jacobi–davidson iteration method for linear eigenvalue problems. *SIAM Review*, 42(2):267–293, 2000.

[137] G.L.G. Sleijpen, H.A. Van der Vorst, and M. Van Gijzen. Quadratic eigenproblems are no problem. *SIAM News*, 29(7):8–9, 1996.

[138] D.C. Sorensen. Implicit application of polynomial filters in ak-step arnoldi method. *SIAM Journal on Matrix Analysis and Applications*, 13(1):357–385, 1992.

[139] DC SORENSEN. Krylov methods for the incompressible navier-stokes equations. *Journal of Computational Physics*, 110:82–102, 1994.

[140] A. Stathopoulos, Y. Saad, and K. Wu. Dynamic thick restarting of the davidson, and the implicitly restarted arnoldi methods. *SIAM Journal on Scientific Computing*, 19(1):227–245, 1998.

[141] P.S. Stewart, S.L. Waters, and O.E. Jensen. Local and global instabilities of flow in a flexible-walled channel. *European Journal of Mechanics-B/Fluids*, 28(4):541–557, 2009.

[142] W.J. Stewart and A. Jennings. A simultaneous iteration algorithm for real matrices. *ACM Transactions on Mathematical Software (TOMS)*, 7(2):184–198, 1981.

[143] LloydnL Trefethen, Annee Trefethen, Satishc Reddy, Tobina Driscoll, et al. Hydrodynamic stability without eigenvalues. *Science*, 261(5121):578–584, 1993.

[144] R.S. Varga. *Geršgorin and his circles.* Springer, 2004.

[145] R.J. Whittaker, M. Heil, J. Boyle, O.E. Jensen, and S.L. Waters. The energetics of flow through a rapidly oscillating tube. part 2. application to an elliptical tube. *Journal of Fluid Mechanics*, 648:123, 2010.

[146] R.J. Whittaker, M. Heil, and S.L. Waters. The energetics of flow through a rapidly oscillating tube with slowly varying amplitude. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 369(1947):2989–3006, 2011.

[147] R.J. Whittaker, S.L. Waters, O.E. Jensen, J. Boyle, and M. Heil. The energetics of flow through a rapidly oscillating tube. part 1. general theory. *Journal of Fluid Mechanics*, 648:83, 2010.

[148] J.H. Wilkinson, J.H. Wilkinson, and J.H. Wilkinson. *The algebraic eigenvalue problem*, volume 155. Oxford Univ Press, 1965.

[149] Gang Wu and L. Feng. A quasi-refined iterative algorithm based on the Lanczos biorthogonalization procedure for large unsymmetric eigenproblems. *Numerical Mathematics A Journal of Chinese University English Series*, 13(1):50–63, 2004.

[150] K. Wu and H. Simon. Thick-restart lanczos method for large symmetric eigenvalue problems. *SIAM Journal on Matrix Analysis and Applications*, 22(2):602–616, 2000.

[151] Q. Ye. A convergence analysis for nonsymmetric lanczos algorithms. *Math. Comp*, 56(194):677–691, 1991.