UNIVERSIDADE FEDERAL DO RIO DE JANEIRO

INSTITUTO DE MATEMÁTICA

CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

JOÃO VITOR DE OLIVEIRA SILVA

# GENERALIZED EIGENVALUE PROBLEMS IN LINEAR HYDRODYNAMIC STABILITY

RIO DE JANEIRO

2018

JOÃO VITOR DE OLIVEIRA SILVA

# GENERALIZED EIGENVALUE PROBLEMS IN LINEAR HYDRODYNAMIC STABILITY

Trabalho de conclusão de curso de graduação apresentado ao Departamento de Ciência da Computação da Universidade Federal do Rio de Janeiro como parte dos requisitos para obtenção do grau de Bacharel em Ciência da Computação.

Advisor: Prof. Juliana Vianna Valério, D.Sc.

Coadvisor: Prof. João Antônio Recio da Paixão, D.Sc.

RIO DE JANEIRO

2018

CIP - Catalogação na Publicação

S586g

Silva, João Vitor de Oliveira
Generalized eigenvalue problems in linear hydrodynamic stability / João Vitor de Oliveira Silva. -- Rio de Janeiro, 2018.
53 f.

Orientadora: Juliana Vianna Valério.
Coorientador: João Antônio Recio da Paixão.
Trabalho de conclusão de curso (graduação) - Universidade Federal do Rio de Janeiro, Instituto de Matemática, Bacharel em Ciência da Computação, 2018.

1. Problemas de autovalor generalizado. 2. Estabilidade hidrodinâmica. 3. Métodos numéricos. 4. Subespaço de Kyrlov. 5. Discretização de equações. I. Valério, Juliana Vianna, orient. II. Paixão, João Antônio Recio da, coorient. III. Título.

Elaborado pelo Sistema de Geração Automática da UFRJ com os dados fornecidos pelo(a) autor(a).

JOÃO VITOR DE OLIVEIRA SILVA

# GENERALIZED EIGENVALUE PROBLEMS IN LINEAR HYDRODYNAMIC STABILITY

Trabalho de conclusão de curso de graduação apresentado ao Departamento de Ciência da Computação da Universidade Federal do Rio de Janeiro como parte dos requisitos para obtenção do grau de Bacharel em Ciência da Computação.

Aprovado em _____ de _____ de _____ .

BANCA EXAMINADORA:

_____

Prof. Juliana Vianna Valério, D.Sc.

_____

Prof. João Antônio Recio da Paixão, D.Sc.

_____

Prof. Daniel Gregório Alfaro Vigo, D.Sc.

_____

Prof. Luziane Ferreira de Mendonça, D.Sc.

## AGRADECIMENTOS

Agradeço primeiramente a Deus pela saúde e forças ao longo de toda minha vida, inclusive a graduação.

Agradeço enormemente minha família pelo suporte e incentivo a meus estudos desde sempre, principalmente minha mãe Eunice, meu pai Luiz e meu irmão Pedro.

Agradeço aos amigos que fiz ao longo de todos estes 5 anos de Universidade Federal do Rio de Janeiro pela ajuda mútua e os momentos de descontração ao longo de disciplinas quando fui aluno e monitor e em eventos que ocorreram na instituição.

Por fim, agradeço aos professores da UFRJ pelos preciosos ensinamentos ao longo de toda esta jornada, em especial os professores do Departamento de Ciência da Computação da área de computação científica, visto que é a área com quem tive mais afinidade e espero aprender sobre muito mais nos próximos anos.

When the state of things is such that an infinitely small variation of the present state will alter only by an infinitely small quantity the state at some future time, the condition of the system, whether at rest or in motion, is said to be stable; but when an infinitely small variation in the present state may bring about a finite difference in the state of the system in a finite time, the condition of the system is said to be unstable.

**James Clerk Maxwell**

## RESUMO

Análise de estabilidade hidrodinâmica é um assunto estudado desde o século XIX, dada sua importância em uma quantidade considerável de áreas da ciência e engenharia. Estabilidade linear de soluções já conhecidas são exploradas usando diferentes discretizações e métodos de solução para problemas de autovalores, com o intuito de observar qual estratégia é a mais eficiente em termos de velocidade de execução e acurácia. Um esquema de discretização local e outro global são aplicados as equações do problema, e, após isso, o problema de autovalor generalizado é resolvido usando o algoritmo QZ e a iteração de Arnoldi. Ambos os métodos numéricos anteriores foram aplicados com e sem uma transformação que reduz o tamanho do problema. A análise realizada visa decidir a melhor estratégia de solução, de forma que seja possível aplicar a mesma em problemas com geometrias mais complicadas e escoamentos mais complicados no futuro.



***Palavras-chave:*** Estabilidade hidrodinâmica. Problema de autovalor generalizado. Métodos de discretização. Métodos numéricos. Arnoldi. Subespaços de Krylov. Algoritmo QZ..

## ABSTRACT

*Hydrodynamic stability analysis has been studied since the nineteenth century, due to it's importance on a considerably ammount of fields of science and engineering. Linear stability of well known solutions on the literature are explored using different discretizations and eigensolvers, with the aim to decide which strategy appears to be the most efficient in terms of execution time and accuracy. One local and one global discretization schemes are applied over the problem's equations, and later on the associated generalized eigenvalue problem is solved using the QZ algorithm and the Arnoldi iteration. Both of the previous methods were applied with and without a transform that reduces the problem's size. The analysis done aims to decide which is the best solution strategy, such that it may be employed in more complicated geometries and flows in the future.*



***Keywords:*** Hydrodynamic stability. Generalized eigenvalue problems. Discretization methods. Numerical methods. Arnoldi. Krylov Subspace. QZ algorithm.

## LIST OF FIGURES

## LIST OF ABBREVIATIONS AND ACRONYMS

| | |
|---|---|
| EVP | Eigenvalue problem |
| GEVP | Generalized eigenvalue problem |
| FD | Finite differences |
| FE | Finite elements |
| IRA | Implicitly restarted Arnoldi |
| LHS | Left hand side |
| RHS | Right hand side |
| FLOPS | Floating point operations per second |
| GPU | Graphics processing unit |
| RAM | Random-acess memory |

## LIST OF SYMBOLS

Scalars are on normal lowercase letter

Vectors are on bold lowercase letter

Matrices are on normal uppercase letter

$\mathbf{u}$ is the velocity field, each component described as $(u, v)$

The gradient is defined as $\nabla \mathbf{u} = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} \end{bmatrix}$

$\psi$ is the stream function

$p$ pressure

$U$ charateristic velocity

$L$ characteristic length

$\rho$ density

$\mu$ viscosity

$Re = \frac{\rho UL}{\mu}$ Reynolds number

**CONTENTS**

# 1 INTRODUCTION

Understanding what makes a flow turbulent and predicting when the transition to such state occurs is one of the most important problems in the area of fluid mechanics and aerodynamics. For example: if it is possible to delay the transition of an airplane's fuel maintaining it laminar on its wings, a considerable amount of fuel will end up being saved [10]. Considering that a lot of factors may influence the flow's behavior, guessing if such transition will happen is not an easy task.

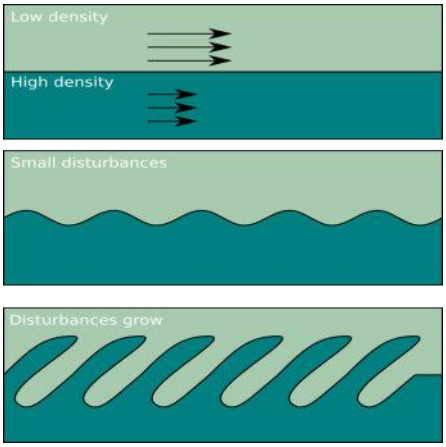
One of the methods used to investigate this phenomenon is linear stability analysis. In order to be applied, a steady state solution of the velocity and pressure fields (in other words, the base flow) is already known of the laminar flow of interest. The instabilities appear because the forces (inertial, viscous and external) equilibrium is perturbed. This perturbation may be written in terms of a normal mode expansion, which results on a continuous (or discrete if the domain is discretized) generalized eigenvalue problem (GEVP) to be solved.

Generalized eigenvalue problems [28], just like the common eigenvalue problems, are of great importance to science and technology. They are found on different fields such as signal/network analysis [32], graph theory and optimization [17], and more recently machine learning [14]. In the context of the hydrodynamic stability, the associated eigenvalues of the linearized perturbed Navier-Stokes equations dictate if the flow is unstable (in other words, if a perturbation will grow indefinitely). Before defining the linear hydrodynamic stability problem and its associated GEVP problem, it is interesting to look a little more at some works that investigated hydrodynamic stability.

## 1.1 HYDRODYNAMIC STABILITY: HISTORY AND MOTIVATION

Studies around hydrodynamic stability have a long history. Lord Kelvin on his studies of instabilities and its relation to vortex flow, along with the Hermann von Helmholtz defined a type of instability that took their name: the *Kelvin-Helmholtz*

*instability* [5]. When two inviscid incompressible fluids flow together with different velocities, the velocity field on the interface between them starts to rotate due to the change between the velocities around it, something that originates vorticity and, as time goes on, turn into a vortex (Figure 1a). This arises in a lot of real life situations, even when we are looking at clouds in a beautiful sunny sky as in Figure 1b.




(a) The velocity field profiles changes due to the *Kelvin-Helmholtz* instability. `https://communitycloudatlas.files.wordpress.com/2015/04/instability.png`

(b) *Kelvin-Helmholtz* instability on clouds. `http://misturaurbana.com/2013/07/o-mar-nas-nuvens/`

Figure 1: *Kelvin-Helmholtz* instability effects.

Other classical situations where such transition occurs was observed by Osborne Reynolds, on a experiment that took his own name: the Reynolds experiment. In [26], he presents an experiment of three different tubes fitted with trumpet mouthpieces (allowing water to enter without disturbance) in a large glass of tank filled with water. A sort of ink was also inserted in the tubes, as a means to highlight the way the flow courses. He observed that:

- When velocities were low, the streak of color extended in a straight line.
- When the water was subjected to perturbations, the streak of ink would shift about the tube, but it was not wave like.
- As the velocity increased more and more, at some point the ink inside would

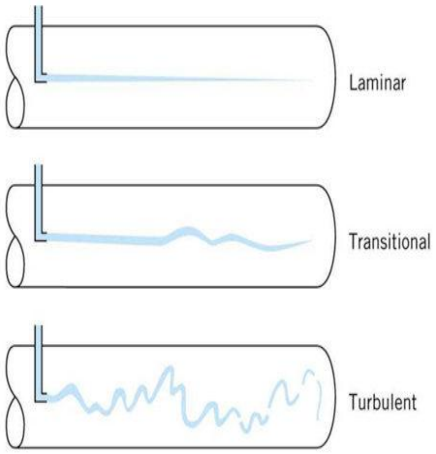



Figure 2: Reynolds experiment. `https://www.learncax.com/component/k2/introduction-to-turbulence-modelling`

become "chaotic", mixing up with the water surrounding it (Figure 2).

The parameter $Re = \frac{\rho U L}{\mu}$, where $\rho$ is the fluid's density (kg/m$^3$), $U$ is the fluid velocity (m/s), $L$ is a characteristic linear dimension (m) is called the Reynolds number and it is related to this experiment. It highlights that, beyond some threshold value of $Re$, the laminar flow in a pipe changes completely to what is known as a turbulent state. This is the essence of someone studying hydrodynamics stability theory: to find if a given laminar flow is stable or unstable, and if so, what are the mechanisms responsible for the shift from the initial behavior and to eventually the turbulent behavior [20].

Flow instabilities is usually unwanted in real life applications. Besides the fuel loss problem on airplanes, coating process that are of great importance to industry require that the flow is laminar, bidimensional, and permanent while the fluid is applied over the substrate [21]. Another different situation is concerning how good blood flows in arteries and observing how different it may be when someone is in a unhealthy condition suffering some disease like Atherosclerosis, where the geometry inside the stenosed artery changes how the blood flows inside it [29]. These situations and a lot of others are the main motivations of the hydrodynamic stability theory.

The present work will be only dedicated to *linear* hydrodynamic stability the-

ory, but there are other ways to analyse. This form of analysis and its associated generalized eigenvalue problem will be discussed in the next chapter. For now, it is important to define what is a GEVP and later on how exactly it appears in this context.

## 1.2 GENERALIZED EIGENVALUE PROBLEM

A generalized eigenvalue problem is a problem described in the following form:

$$A\mathbf{x} = \lambda B\mathbf{x} \tag{1.1}$$

where $A, B \in \mathbb{C}^{m\times m}$, $\lambda \in \mathbb{C}$. Essentially the objective is to find pairs $(\lambda, \mathbf{x})$ such that 1.1 is satisfied. The number $\lambda$ is called a generalized eigenvalue and its associated $\mathbf{x}$ a generalized eigenvector. For simplicity, the term generalized will be dropped when referring to any of those. One can easily notice that when $B = I$, it is essentially an eigenvalue problem. Sometimes GEVP is called a linear matrix pencil in the literature.

Similar to an EVP, by solving the characteristic polynomial $p(\lambda) = \det(A - \lambda B) = 0$ and them solving a system of linear equations for each of its roots (eigenvalues) is a possibility. It is interesting look at an example:

$$\begin{bmatrix} 3 & 4 & -8 \\ 0 & 6 & 6 \\ 0 & 0 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \lambda \begin{bmatrix} 1 & 2 & 3 \\ 0 & 3 & 3 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \tag{1.2}$$

The charateristic polynomial is $p(\lambda) = (3 - \lambda)(6 - 3\lambda)(4 - \lambda)$. The roots (eigenvalues) in this case are easily obtained, being $\lambda_1 = 3$, $\lambda_2 = 2$, $\lambda_3 = 4$. Their (normalized) associated eigenvectors are , $\mathbf{v}_1 = [1\ 0\ 0]^T$, $\mathbf{v}_2 = [0\ 1\ 0]^T$, $\mathbf{v}_3 = \frac{1}{\sqrt{258}}[-16\ \ -1\ \ 1]^T$.

The previous example was a simple one because of the structure behind the matrices: both were upper triangular. An interesting fact is that this is one of

the basic ideas behind how the most used numerical eigensolver for GEVP (the QZ algorithm) finds the eigenvalues. This method iteratively tries to find two unitary transforms $Q, Z$ such that $A\mathbf{x} = \lambda B\mathbf{x}$ is rewritten as

$$QTZ^*\mathbf{x} = QSZ^*\mathbf{x} \tag{1.3}$$

where $T$ and $S$ are upper triangular matrices. In the literature, this is known as the generalized Schur form. It can be shown that the problem $T\mathbf{x} = \lambda S\mathbf{x}$ share the same eigenvalues, something that can be used to find their associated eigenvectors afterwards. This work willl not go into its theory, however [11] provides a good theoretical and pratical overview.

It may seem a little odd not interpreting any GEVP as just a different way of writing a EVP, since it appears that $A\mathbf{x} = \lambda B\mathbf{x}$ could be rewritten as $B^{-1}A\mathbf{x} = \lambda\mathbf{x}$. If this happens to be the case, there would be no need to develop methods specific to it (like the QZ method mentioned). On the previous example, this could be done since $B$ was an invertible matrix, but not on the following example:

$$\begin{bmatrix} 10 & 10 & -10 \\ -10 & 20 & 20 \\ 10 & 20 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \lambda \begin{bmatrix} 2 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \tag{1.4}$$

Since $B$ is not an invertible matrix, it is not possible obtain an EVP in such manner and it may seem that only computing the roots of the charateristic polynomial is a possibility to find the eigenvalues. There are indeed ways of obtaining a corresponding EVP, but it requires certain transforms on the problem. As for now, try to guess what would happen to 1.1 when a $\tilde{\mathbf{x}} \in \text{Nullspace}(B)$ and $\tilde{\mathbf{x}} \notin \text{Nullspace}(A)$ is taked into account on the previous example. There would a non null vector on the LHS of the equality and a null vector on the RHS of the equality multiplied by a scalar $\lambda$. Interestingly, $\lambda$ "tries" to maintain this equality by going towards infinity, something that can be checked by doing the following:

- Shift the Nullspace($B$) by $\epsilon$ :

$$(B + \epsilon I)$$

- Assuming $\tilde{\mathbf{x}} \in$ Nullspace($B$), it can be verified that :

$(B + \epsilon I)\tilde{\mathbf{x}} = B\tilde{\mathbf{x}} + \epsilon I\tilde{\mathbf{x}} = \mathbf{0} + \epsilon\tilde{\mathbf{x}} = \epsilon\tilde{\mathbf{x}}$, not a null vector anymore.

- Now, write the associated characteristic polynomial

$p(\lambda) = \det(A - \lambda(B + \epsilon I)) = 1000\lambda + 12\lambda^3\epsilon - 180\lambda^2\epsilon + 600\lambda\epsilon + 2000$

- The (incredibly scary) solutions for $p(\lambda) = 0$ are:

$$\begin{aligned}
\lambda_1 =& \frac{15\epsilon - 50}{3\sqrt[3]{\sqrt{\epsilon^3\left(-27\epsilon^3 + 270\epsilon^2 + 3069\epsilon + 1000\right)} - 63\epsilon^2}} \\
&+ \frac{5\sqrt[3]{\sqrt{\epsilon^3\left(-27\epsilon^3 + 270\epsilon^2 + 3069\epsilon + 1000\right)} - 63\epsilon^2}}{3\epsilon} + 5 \\
\lambda_2 =& -\frac{5i\left(\sqrt{3} - i\right)(3\epsilon - 10)}{6\sqrt[3]{\sqrt{\epsilon^3\left(-27\epsilon^3 + 270\epsilon^2 + 3069\epsilon + 1000\right)} - 63\epsilon^2}} \\
&+ \frac{5i\left(\sqrt{3} + i\right)\sqrt[3]{\sqrt{\epsilon^3\left(-27\epsilon^3 + 270\epsilon^2 + 3069\epsilon + 1000\right)} - 63\epsilon^2}}{6\epsilon} + 5 \\
\lambda_3 =& \frac{5i\left(\sqrt{3} + i\right)(3\epsilon - 10)}{6\sqrt[3]{\sqrt{\epsilon^3\left(-27\epsilon^3 + 270\epsilon^2 + 3069\epsilon + 1000\right)} - 63\epsilon^2}} \\
&- \frac{5i\left(\sqrt{3} - i\right)\sqrt[3]{\sqrt{\epsilon^3\left(-27\epsilon^3 + 270\epsilon^2 + 3069\epsilon + 1000\right)} - 63\epsilon^2}}{6\epsilon} + 5
\end{aligned}$$

Finally, by taking the following limits:

$$\lim_{\epsilon\to 0}\lambda_1 = -2, \quad \lim_{\epsilon\to 0}\lambda_2 = +\infty\, i, \quad \lim_{\epsilon\to 0}\lambda_3 = -\infty\, i \tag{1.5}$$

This shows that on the "limit of invertibility", some of the eigenvalues are heading towards infinity. These are the so called infinite eigenvalues. They are defined as:

**Definition 1.1.** *Denoting $\lambda(A, B + \epsilon I)$ as a eigenvalue of $A\mathbf{x} = \lambda(B + \epsilon I)\mathbf{x}$, if $\lim_{\epsilon\to 0}\lambda(A, B + \epsilon I) = \pm\infty$ or $\lambda(A, B + \epsilon I) = \pm\infty\, i$ then $\lambda$ is an infinite eigenvalue of $A\mathbf{x} = \lambda B\mathbf{x}$.*

For pratical purposes they are usually not important, but their presence poses a

problem to eigensolvers due to how their iterations work. On chapter 4, computational methods to solve a GEVP with the presence of these are discussed.

## 1.3 OBJECTIVE, TECHNOLOGIES AND OUTLINES

The present work produces results on linear hydrodynamic stability already present on the literature, but trying different strategies (some to our knowledge have not been tried before) concerning the discretization and methods to solve the GEVP. The aim is to try to refine what is the most coherent way of solving the problem in known situations in order to be applied on more difficult and advanced flows that arise in applications.

According to [20], the state of art methods are matrix-free methods that uses an exponential time stepper (after removing the pressure field after some sort of projection method), with spectral elements discretization frameworks such as NEK5000 [9]. This strategy is not discussed in this work, the GEVP matrices are still used.

The used technologies were the following:

- Wolfram Mathematica 11[25] for symbolic computation and plotting of the eigenspectrums.
- GNU Octave 4.0.1[8] for all numerical computations.
- LaTeX for the present text format and TikZ for some graphical representations.

This introduction chapter exposed some of the ideas behind both the linear stability and the GEVP. Chapter 2 is concerned with presenting the base flows and how to construct the stability equations. Chapter 3 presents methods to discretize the domain, followed by chapter 4 focusing on strategies do solve the GEVP obtained. Finally, chapter 5 presents the obtained results and chapter 6 discusses the conclusions and possible future works. See Figure 3 for a diagram representation of the described outline.

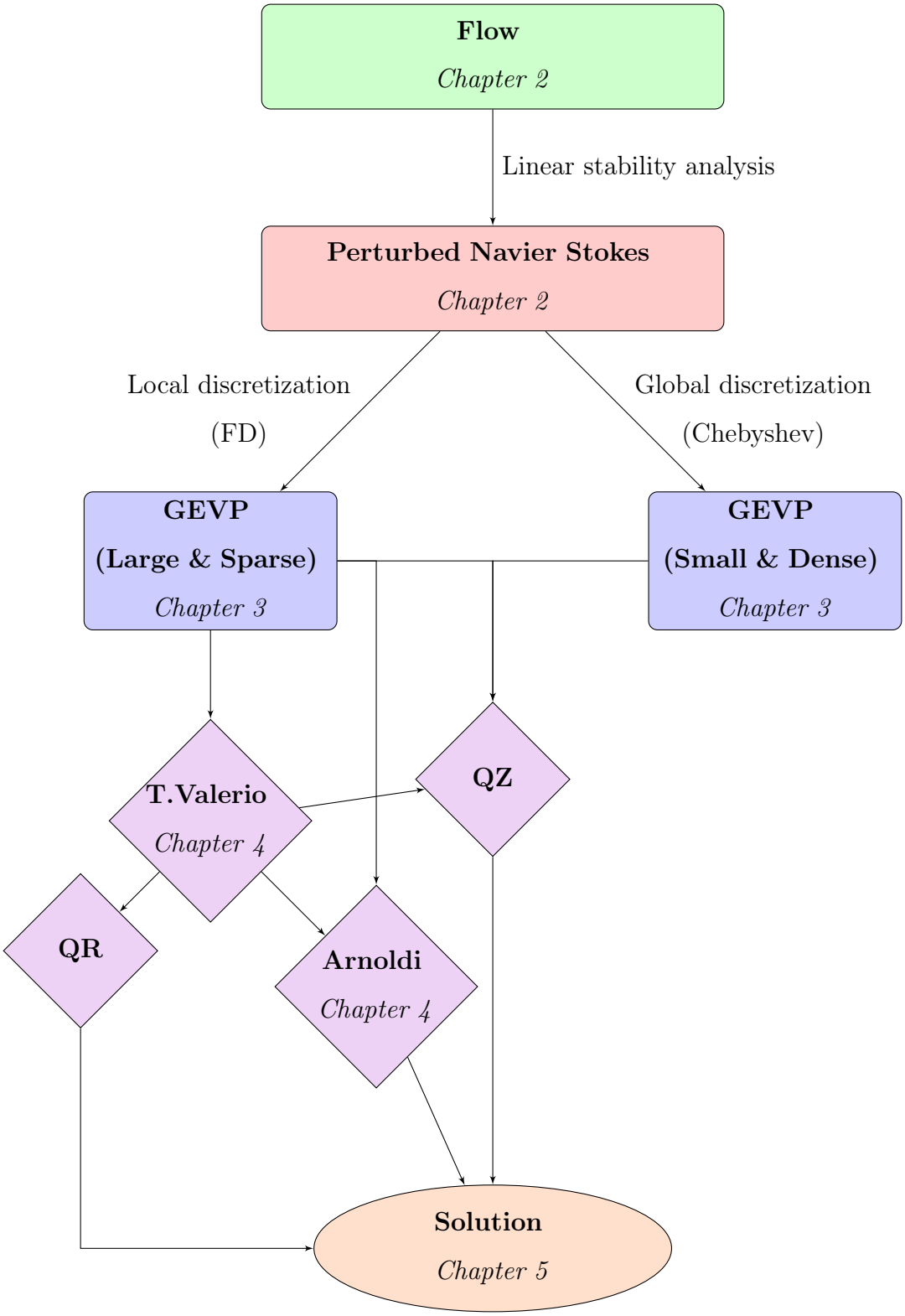



Figure 3: Diagram describing the outline of the present work.

# 2 LINEAR STABILITY ANALYSIS AND BASE FLOWS

According to [20], hydrodynamic stability analysis can be divided in 4 groups. Two of those concern instabilities that evolve according to time evolution: temporal stability analysis and transient growth analysis. The former is dedicated to investigating the asymptotic time behaviour of perturbations, while the latter observes in short-time dynamics how perturbations are amplified. Both theories have their importance and may give insights about the underlying stability of the system. An example of work that does both is [2]. The present work only concerns the *linear* case of the temporal stability analysis.

## 2.1 PLANAR COUETTE AND POISEUILLE FLOWS

Simply put, base flow is a steady-state or periodic solution of the Navier-Stokes equations. Steady-state solutions can be obtained by "dropping" the time derivative terms of said equations, in other words, the flow is assumed to be permanent. Assuming the fluid is incrompressible, the continuity and conservation of momentum are respectively (on nondimensional form):

$$\begin{cases} \nabla.\mathbf{u} = 0 & (2.1) \\ Re\left(\cancel{\frac{\partial \mathbf{u}}{\partial t}} + \mathbf{u}.\nabla\mathbf{u}\right) = -\nabla p + \nabla^2\mathbf{u}, & (2.2) \end{cases}$$

where $\mathbf{u}$, $p$ represent the velocity and pressurre fields respectively. The dimensionless parameter $Re$ is called the Reynolds number. It represents the ratio between inertial forces and viscous forces. Flows with high velocity have preominant inertial forces, and these are the ones which may cause a significant change on the velocity field's profile.

There are bidimensional flows that the velocity field essentially depends of only of the spatial coordinates, these are called parallel flows. There are two famous flows that results on linear stability are already present on the literature: the planar Couette and Poiseuille flows.

The planar bidimensional Couette flow happens when there is not a pressure gradient across the boundary layer ($\frac{\partial p}{\partial y} = \frac{\partial p}{\partial x} = 0$) , each velocity component does not depend on the horizontal spatial coordinate ($\frac{\partial u}{\partial x} = \frac{\partial v}{\partial x} = 0$)
and between two paralell plates, the velocity has the same magnitude and direction, but oposite orientation. The following boundary conditions are then met:

$$u(1) = 1, u(-1) = -1, v(\pm 1) = 0, \quad \text{where } \mathbf{u} = [u, v]^T.$$

The analytical solution can be easily computed in this simple case. It is described by $\mathbf{u}_{\text{base}} = [U = y, v = 0]^T, \quad p = p_{\text{base}} = 0$ (even though any constant could be used, this form will be assumed for simplicity). The graphical representation is in Figure 4.

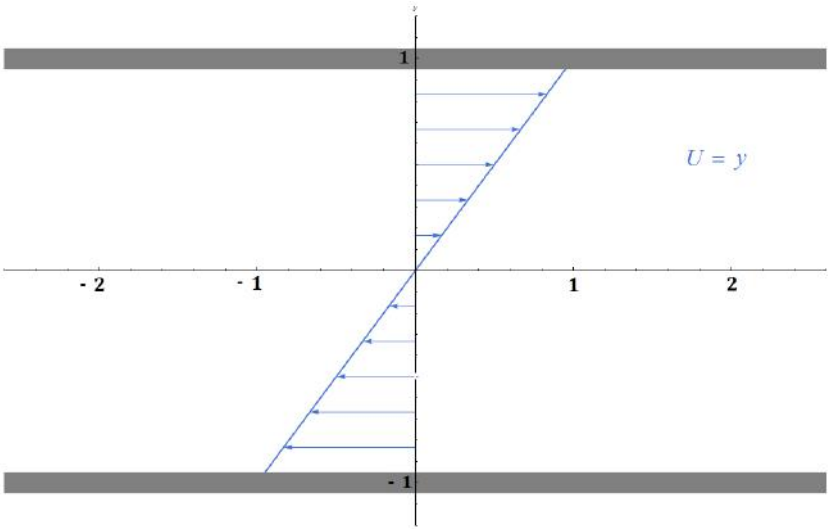



Figure 4: Planar Couette flow.

Similarly, the planar Poiseuille flow shares the same geometry and it is also assumed that each velocity component depends only on the vertical position. On the other hand, there happens to be a pressure gradient parallel to the flow itself, one that is constant ($\frac{\partial p}{\partial x} = -G$, $\frac{\partial p}{\partial y} = 0$). Also, the velocity is null on both parallel plates. The following boundary conditions are then met:

$$u(\pm 1) = 0, v(\pm 1) = 0, \quad \text{where } \mathbf{u} = [u, v]^T.$$

The analytical solution can also be easily computed in this simple case. It is described by $\mathbf{u}_{\text{base}} = [U = 1 - y^2, 0]^T, \quad p = p_{\text{base}} = -Gx$. Figure 6 shows the parabolic

shape of this velocity field.

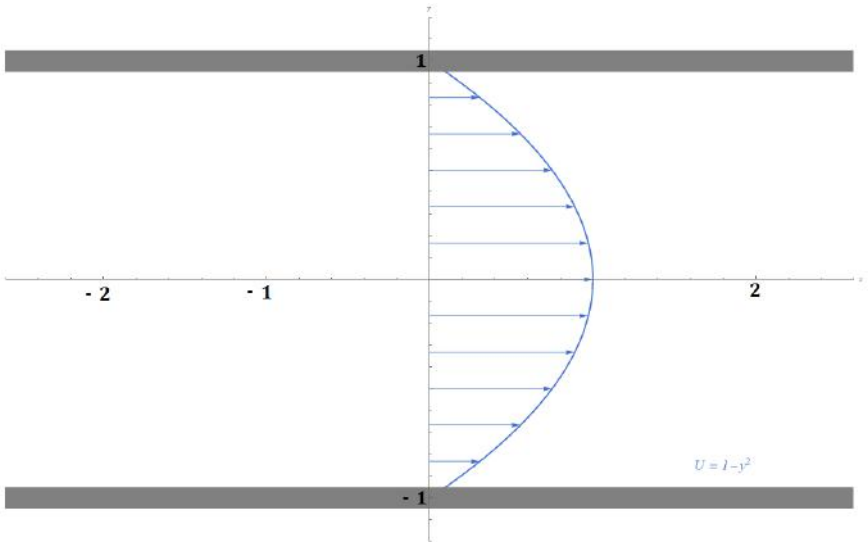



Figure 5: Planar Poiseuille flow.

Unlike the general case, these two flows are so simple that even have analytical solutions, something that could indicate that they both do not hold much importance. This is not case because the ideas behind the linear hydrodynamics analysis remain the same on more complex problems. Also there are some flows that may be approximately written as one of these two [35].

## 2.2 INTRODUCING A PERTURBATION

In order to decide if the previous base flow is stable, a known perturbation is introduced in the previous steady state solution. It is assumed that the perturbed solution are of the form:

$$\begin{aligned} &\mathbf{u} = \mathbf{u}_{\text{base}} + \epsilon\tilde{\mathbf{u}}, \quad \tilde{\mathbf{u}} = [\hat{u}, \hat{v}]^T \\ &p = p_{\text{base}} + \epsilon\tilde{p} \\ &0 < \epsilon << 1, \text{ small perturbation} \end{aligned} \tag{2.3}$$

where $\tilde{\mathbf{u}} = \hat{\mathbf{u}}(y)e^{i\alpha x}e^{\lambda t}$ and $\tilde{p} = \hat{p}(y)e^{i\alpha x}e^{\lambda t}$. The fields $\tilde{\mathbf{u}}$ and $\tilde{p}$ are called perturbed fields, both representing a wave (each having an amplitude, as well as spatial and temporal dependency). An imposing condition to these perturbed fields

is that $\hat{u}(\pm 1) = 0, \hat{v}(\pm 1) = 0$, since the previously boundary conditions should still be satisfied by the new solution. The expressions themselves suggest that if the real part of the scalar $\lambda$ (denoted by $\Re\{\lambda\}$) is larger than 0, the perturbation will evolve indefinitely. On the contrary, if it is less than 0 then the perturbation will disappear as time passes, meaning that the flow will recover its base flow behavior. Note that the imaginary part of $\lambda$ (denoted by $\mathrm{Im}\{\lambda\}$) does not matter for this form of analysis (it only represents the shape of the wave's oscillations, not if the perturbation grows or decays).

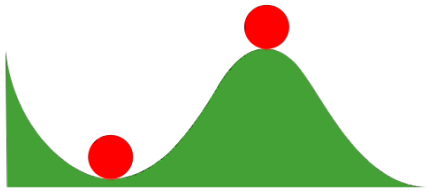


Figure 6: Ball on a hill. The left ball represents a $\Re\{\lambda\} < 0$ and the right one a $\Re\{\lambda\} > 0$ situation. `https://commons.wikimedia.org/wiki/File:Stable-unstable1.svg`

At first, the new velocity and pressure fields are substituted back to the Navier-Stokes equations (the full one, now accounting time derivatives as well). By using the assumption that the perturbation's magnitude is tiny, it seems reasonable to drop terms of order $O(\epsilon^2)$. Essentially this removes the non linear terms of the system and results in the following:

$$-\left[Re(i\alpha U)\right]\hat{u} - \left(Re\frac{dU}{dy}\right)\hat{v} - i\alpha\hat{p} + \frac{d^2\hat{u}}{dy^2} - \alpha^2\hat{u} = (Re)\ \ \lambda\hat{u} \tag{2.4}$$

$$-\left[Re(i\alpha U)\right]\hat{v} - \frac{d\hat{p}}{dy} + \frac{d^2\hat{v}}{dy^2} - \alpha^2\hat{v} = (Re)\ \ \lambda\hat{v} \tag{2.5}$$

$$i\alpha\hat{u} + \frac{d\hat{v}}{dy} = 0 \tag{2.6}$$

The above continous problem has the following unknowns: $\hat{u}$, $\hat{v}$, $\hat{p}$ (amplitudes), and most importantly, $\lambda$. The parameters $Re$ and $\alpha$ (wavenumber, the ratio of frequency and speed of propagation) are defined previously in this form of analysis. In

other words, for each pair of $(Re, \alpha)$, a different solution of the mentioned unknowns is obtained. Due to the difficulty at solving this continous generalized eigenvalue problem, a discretization method may be applied.

Historically, to study the stability of parallel flows was use the fact that every bidimensional flow may be written in terms of a scalar streamfunction [18]:

$$\begin{gathered} \psi(y) = \phi(y) e^{i\alpha x + \lambda t} \\ \tilde{u} = \frac{\partial \psi}{\partial y}, \quad \tilde{v} = -\frac{\partial \psi}{\partial x} \quad \rightarrow \quad \hat{u} = \frac{\partial \phi}{\partial y}, \quad \hat{v} = i\alpha\phi(y) \end{gathered} \tag{2.7}$$

The way $\hat{u}$ and $\hat{v}$ are defined in 2.7 already satisfies 2.6. Using this fact, by differentiating against $y$ both sides of 2.4 and summing with equation 2.5 multiplied by the factor $-\alpha\, i$ , the following equation is obtained:

$$\begin{gathered} \left(\frac{d^2}{dy^2} - \alpha^2\right)^2 \phi - (Re\; i\alpha U \left[\frac{d^2\phi}{dy^2} - \alpha^2\phi\right]) + (Re\; i\alpha)\frac{d^2U}{dy^2} = \lambda Re \left(\frac{d^2\phi}{dy^2} - \alpha^2\phi\right) \\ \phi(-1) = 0, \phi(1) = 0, \phi'(-1) = 0, \phi'(1) = 0 \end{gathered} \tag{2.8}$$

This eigenvalue problem is named the Orr-Sommerfeld equation. Unlike the primitive variable formulation (the problem written in terms of the velocity and pressure fields), there is only one unknown field $\phi$ along with the eigenvalues. Also, unlike the primitive variable formulation, the infinite eigenvalues are not present. The focus of this work is not to work with such equation, since a streamfunction does not even exist for complex flows and geometries. Considering that most works dedicated to solving the linear stability for parallel flows end up using it, results from [23] will be reproduced for benchmarking and comparison purposes.

# 3 DISCRETIZATION METHODS

There is a diversity of methods to discretize system of differential equations. Some of these can be considered a local discretization method or a global discretization method (these are called spectral methods). The first group of methods approximates a function derivative locally using low order polynomials, some famous examples are finite differences and finite element. The second group of methods approximates a function using a high order polynomial that interpolates the whole discretized domain, some famous examples are Chebyshev collocation method and discrete Fourier transform. Regardless of the method, a resultant GEVP has to be solved aftewards.

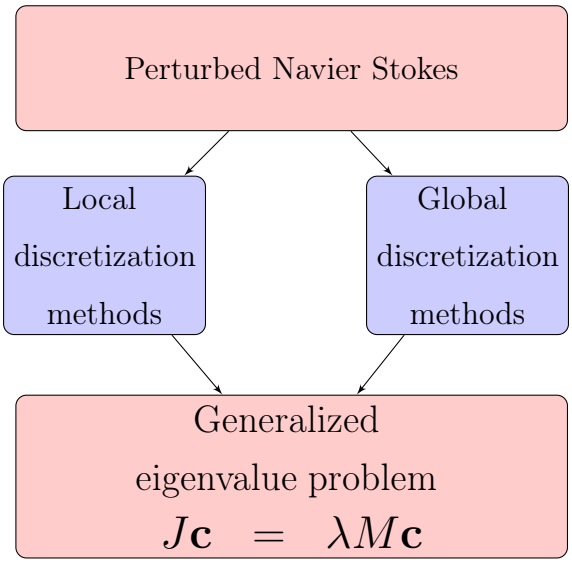



Figure 7: Diagram describing the GEVP matrices construction.

The simple geometry on parallel flows allows methods of both groups to be employed, but the presence of an unknown scalar field (pressure) and a vector filed (velocity) has to be taken with some care depending on the chosen method. This work uses a local discretization method (finite differences) and a global discretization method (Chebyshev collocation method).

## 3.1 FINITE DIFFERENCES METHOD

The finite differences method is usually presented as truncated formulas of the Taylor series expansion of some unknown function $f(x)$. Another interpretation is the one previously mentioned: a local interpolation is performed using a low order polynomial $L(x)$. The second order central difference (assuming an uniformly distributed grid) can be obtained at some point $x_j$ by using a polynomial that interpolates the points $(x_{j-1}, f(x_{j-1})), (x_j, f(x_j))$, and $(x_{j+1}, f(x_{j+1}))$:

Denoting $f(x_j)$ as $f_j$ and $\Delta x = x_{j+1} - x_j$ :

$$L(x) = f_{j-1}\, l_{j-1}(x) + f_j\, l_j(x) + f_{j+1}\, l_{j+1}(x), \text{ where } l_j(x) = \prod_{k \neq j} \frac{x - x_k}{x_j - x_k}$$

$$L'(x) = f_{j-1} l'_{j-1}(x) + f_j l'_j(x) + f_j l'_{j+1}(x), \text{ where } l'_j(x) = \sum_{k \neq j} \frac{1}{x_j - x_k} \prod_{m \neq (j,k)} \frac{x - x_m}{x_j - x_m}$$

$$L''(x) = f_{j-1} l''_{j-1}(x) + f_j l''_j(x) + f_j l''_{j+1}(x), \text{ where } l''_j(x) = \sum_{l \neq j} \frac{1}{x_j - x_l} \sum_{k \neq (j,l)} \frac{1}{x_j - x_k} \prod_{m \neq (j,l,k)} \frac{x - x_m}{x_j - x_m}$$

$$f'_j \approx L'(x_i) = \frac{f_{j+1} - f_{j-1}}{2\Delta x} \qquad \rightarrow \text{ central difference (1st derivative)}$$

$$f''_j \approx L''(x_i) = \frac{f_{j-1} - 2f_j + f_{j+1}}{\Delta x^2} \rightarrow \text{ central difference (2nd derivative)}$$

The theory assures an error of $O(\Delta x^2)$ for both approximations [4]. This local approximations assures a sparse matrix to describe the derivative of every node.

Back to the discretization of the primitive variables, one could assume that both fields could be discretized in the same nodes (this is called a collocated grid). Surprisingly, this is not the case. When such thing is done, non-physical pressure oscilations may occur because the energy of the system is not conserved as it should be. More details of the issues that arise by using a collocated grid are discussed in [6]. One of the most common ways to solve this issue is to use a staggered grid. A staggered grid is a discretization scheme such that pressure is discretized in between two velocity nodes, as illustrated in Figure 8.

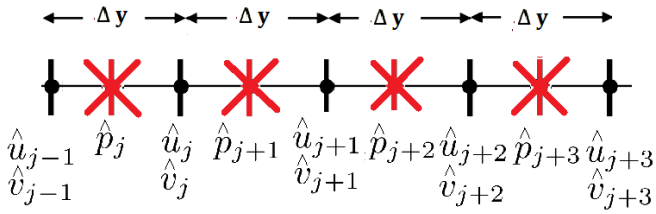



Figure 8: Staggered grid: black nodes contains velocities and red (crossed) nodes contains pressures.

Using a staggered grid, the momentum conservation equation 2.4 (horizontal component) and 2.5 (vertical component) are discretized in the black nodes as

$$\hat{p}(y_j^{(black)}) \approx \frac{p_j + p_{j+1}}{2}$$
$$\left.\frac{d^2\hat{u}}{dy^2}\right|_{y_j^{(black)}} \approx \frac{u_{j-1} - 2u_j + u_{j+1}}{\Delta y^2}, \quad \left.\frac{d^2\hat{v}}{dy^2}\right|_{y_j^{(black)}} \approx \frac{v_{j-1} - 2v_j + v_{j+1}}{\Delta y^2}, \quad \left.\frac{d\hat{p}}{dy}\right|_{y_j^{(black)}} \approx \frac{p_{j+1} - p_j}{\Delta y} \tag{3.1}$$

while the continuity equation 2.6 is discretized as

$$\begin{aligned} \hat{u}(y_j^{(red)}) &\approx \frac{u_j + u_{j+1}}{2} \\ \left.\frac{d\hat{v}}{dy}\right|_{y_j^{(red)}} &\approx \frac{v_{j+1} - v_j}{\Delta y} \end{aligned} \tag{3.2}$$

## 3.2 CHEBYSHEV COLLOCATION METHOD

Chebyshev collocation method is based on the idea of approximating the unknown function $f$, but using a high order polynomial. One issue that arises by using a uniformly distributed grid is the Runge-Phenomenon [27]. Fortunately, there is way to circunvent this problem by doing the interpolation at another distribution of nodes, one possibility being the Chebyshev nodes [30]. The distribution of the Chebyshev nodes is the horizontal coordinate of equally spaced points in the unit circle:

$$ChebyshevPoint_j = \cos\left(\frac{j\pi}{N}\right), \text{ where } N \text{ is the grid size.} \tag{3.3}$$

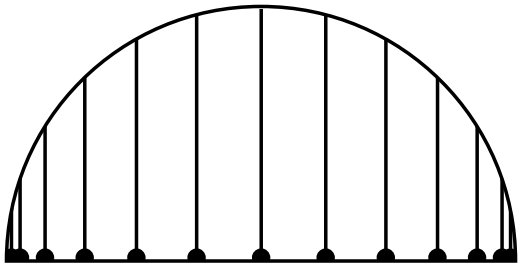


Figure 9: Chebyshev points. In constrast to an equidistant grid, more points are near the boundaries.

The following theorem provides error bounds for the interpolation error in these nodes, as well as the corresponding derivative approximations. The proof and more details can be obtained in [30].

**Theorem 3.1.** *Given a function $f$ and a set of Chebyshev points $\{x_j\}_N$, define $\phi(z) = \frac{1}{N}\sum_{k=1}^{N} \log|z - z_k|$ ($z_k$ being the roots of the interpolant $p_N(x)$) and $\tilde{\phi} = \sup_{x\in[-1,1]} \phi$. If there's a constant $\phi_f > \tilde{\phi}$ such that $f$ is analytic throughout the closed region $\{z \in \mathbb{C} : \phi(z) \leq \phi_f\}$, there exists a constant $C > 0$ such that for all $x \in [-1, 1]$ and all $N$:*

$$|f(x) - p_N(x)| \leq C \exp\left(-N\left(\phi_f - \tilde{\phi}\right)\right) \tag{3.4}$$

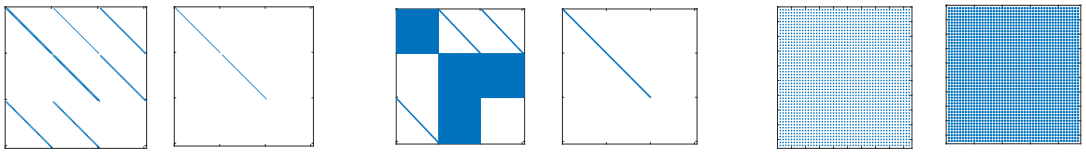
The same result holds for any *kth* derivative, only the constant $C$ changes. This means that Chebyshev interpolation (and the corresponding spectral derivative) converges geometrically (in exact arithmetic). Such nice convergence properties may suggest that global discretization methods are always superior to local ones, but if the base flow solution has finite regularity or if the domain is irregular (a possibility considering more complicated flows) the fast convergence is lost and may end up being inferior than a local discretization. A combination of both strategies (only assuming regularity locally and using a high order interpolant) is used in the spectral elements method [15].

With this in mind, denoting $\mathbf{f}$ as some $f(x)$ of interest evaluated at each Chebyshev node, it's possible to define a spectral derivative matrix $D$ such that $\mathbf{f}' = D\mathbf{f}$. The $(N+1) \times (N+1)$ matrix is defined in [30] as:

$$
\begin{aligned}
D(1,1) &= \frac{2N^2+1}{6}, \quad D(N+1,N+1) = \frac{-2N^2+1}{6} \\
D(j,j) &= \frac{-x_j}{2(1-x_j^2)}, \quad j = 2,...,N \\
D(i,j) &= \frac{\theta_i}{\theta_j}\frac{(-1)^{i+j}}{(x_i - x_j)}, \quad i,j = 1,...,N+1, \quad i \neq j, \\
&\text{where} \begin{cases} \theta_i = 1, & \text{if} \quad 2 \leq i \leq N \\ \theta_i = 2, & \text{otherwise} \end{cases}
\end{aligned} \tag{3.5}
$$

By definition, this makes the matrix $D$ dense. This is clearly something that makes a global discretization method distinct from a local one (which are characterized by sparse matrices).

Back to the eigenvalue problem, the discretization of the system composed of the perturbed Navier-Stokes equations (equations 2.4, 2.5 and 2.6) is done in a collocated grid, which results in the issue previously mentioned, although it appears to be less of an issue in global discretization methods. Even so, it's possible to use a staggered grid as detailed in [16].
By using the described schemes, as well the Chebyshev collocation discretization over equation 2.8, the corresponding GEVP has the following structure (10):



(a) $J$ VP, Finite Differences (b) $M$ VP, Finite Differences (c) $J$ VP, Chebyshev (d) $M$ VP, Chebyshev (e) $J$ Orr-Sommerfeld, Chebyshev (f) $M$ Orr-Sommerfeld, Chebyshev

Figure 10: Matrices of the resulting generalized eigenvalue problem $J\mathbf{c} = \lambda M\mathbf{c}$ using different discretizations. The existence of blocks is present on the primitive variables formulation (VP) and matrices densities are associated with using a global or local discretization method. Also, for a certain estabilished accuracy, local methods have notably larger matrices compared to the global ones (e.g. 400 nodes on FD and 51 nodes on Chebyshev as in Figure 13).

Solving the problem directly with the QZ algorithm without exploiting any of the problem properties, using the same perturbation as in [33] ($Re = 500$ and $\alpha = 1.5$) results in the following Couette spectrum (Figure 13, points are eigenvalues with the horizontal and vertical coordinates being the real and imaginary parts respectively):

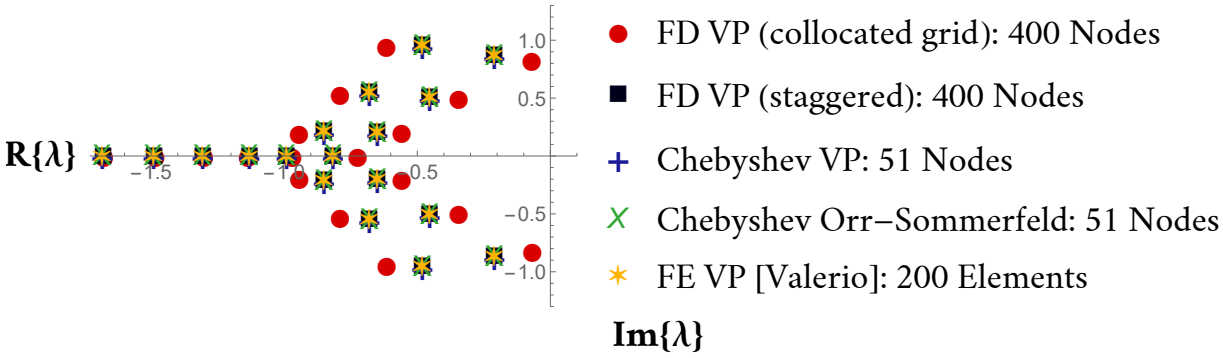



Figure 11: Couette spectrum (18 eigenvalues) for different discretizations. $Re = 500$ and $\alpha = 1.5$

Some noteworthy observations can be drawn from the results: first of all, all methods result on the same spectrum as the one in [33] (the only exception being the spectrum resulted by using finite differences with a collocated grid). Also, by looking at $\Re\{\lambda\}$ for all the computed eigenvalues, it is correct to say that the flow is linearly stable (all eigenvalues have negative real part).

As for the Poiseuille flow, [7] informs that the flow is linearly unstable for $Re > 5772$, this value receiving the name of critical Reynolds number. The results for $Re = 5772, \alpha = 1.02$ and $Re = 10^4, \alpha = 1.02$ are as follows:

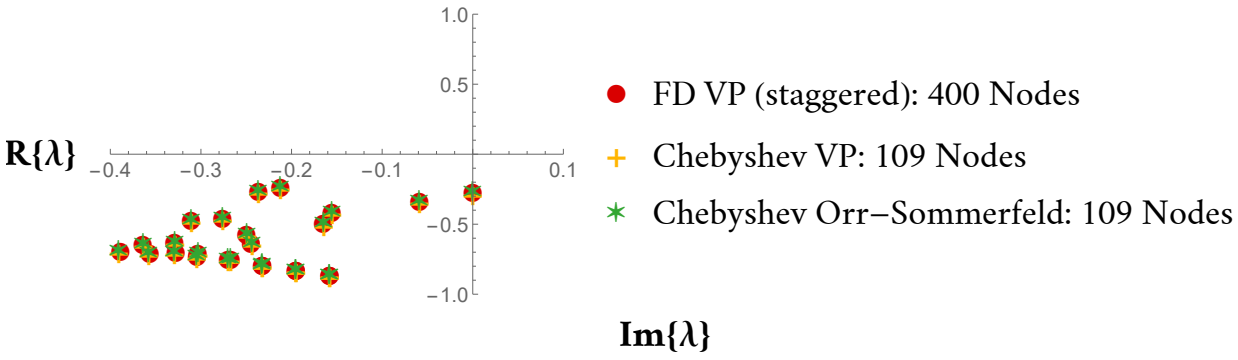



Figure 12: Poiseulli spectrum (18 eigenvalues) for different discretizations. $Re = 5772$ and $\alpha = 1.02$. The leading eigenvalue is $\lambda_{FD} = -0.00017757 - 0.269285i$, $\lambda_{Chebyshev} = -4.30121 \times 10^{-7} - 0.269217i$.

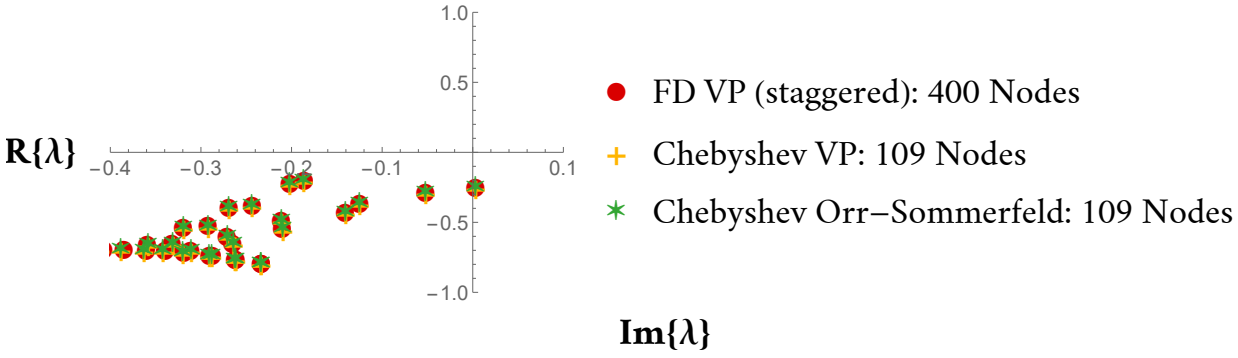



Figure 13: Poiseulli spectrum for different discretizations. $Re = 10^4$ and $\alpha = 1.02$. The leading eigenvalue is $\lambda_{FD} = 0.00304019 - 0.244499i$, $\lambda_{Chebyshev} = 0.00328321 - 0.244388i$.

The results are in accordance to the literature, as the leading eigenvalue is present in the fourth quadrant (positive real part), in others words, the unstable region $\Re\{\lambda\} > 0$.

With the discretization methods resulting in coherent eigenvalues, the question returns to how to solve the GEVP efficiently. "Blindly" applying the QZ is not the best strategy, since as described in [3] it is costly and does not explore any underlying special structure the problem may have (such as sparsity when a local discretization

method is used). The following chapter is dedicated to discussing strategies to solve it in different ways.

# 4 GENERALIZED EIGENVALUE PROBLEM: CONSIDERATIONS AND METHODS

After the discretization process of the primitive variables formulation, a generalized eigenvalue problem of the form $\boxed{J\mathbf{c} = \lambda M\mathbf{c}}$, where $J, M \in \mathbb{C}^{(2N_v+N_p)\times(2N_v+N_p)}$ ($N_v$, $N_p$ are the number of velocity and pressure nodes respectively), $\lambda \in \mathbb{C}$ and $\mathbf{c} \in \mathbb{C}^{(2N_v+N_p)}$, $\mathbf{c}$ being a vector containing both the grid's unknown amplitudes of the perturbed velocities and pressure fields. The scalar $\lambda$ is associated with the magnitude of the perturbation as already mentioned. The associated matrices have the following properties:

- The matrix $J$ will be a sparse matrix if a local discretization method was employed, and $J$ will be a dense matrix by using a global discretization method. $M$ is a diagonal matrix.
- The matrix $J$ is ill-conditioned and non-hermitian, while $M$ is a singular matrix, in other words, there is a generalized non-hermitian eigenvalue problem (GNHEP). The second matrix is responsible for the presence of infinite eigenvalues, where the associated eigenvectors are the vectors that belong to Nullspace($M$). Physically, they are related to the incompressibility of the flow.
- Since a system of three equations was discretized, the matrices $J$ and $M$ can be divided in blocks accordingly:

$$
\begin{array}{c} \\ N_v \\ N_v \\ N_p \end{array}
\begin{array}{c}
\begin{array}{ccc} N_v & N_v & N_p \end{array} \\
\left[\begin{array}{c|c|c} J_{11} & J_{12} & J_{13} \\ \hline J_{21} & J_{22} & J_{23} \\ \hline J_{31} & J_{32} & 0 \end{array}\right]
\end{array}
\left[\begin{array}{c} \hat{u} \\ \hline \hat{v} \\ \hline \hat{p} \end{array}\right]
= \lambda
\begin{array}{c} \\ N_v \\ N_v \\ N_p \end{array}
\begin{array}{c}
\begin{array}{ccc} N_v & N_v & N_p \end{array} \\
\left[\begin{array}{c|c|c} M_{11} & 0 & 0 \\ \hline 0 & M_{22} & 0 \\ \hline 0 & 0 & 0 \end{array}\right]
\end{array}
\left[\begin{array}{c} \hat{u} \\ \hline \hat{v} \\ \hline \hat{p} \end{array}\right]
$$

As a means to solve the GEVP efficiently, the following techniques take into account these characteristics. It is worth noting that they may be even used together, possibly making the calculations even faster.

## 4.1 SHIFT - INVERT

The GEVP of interest cannot be solved with a EVP solver directly. If $M$ was not singular, its inverse could be used to obtain a EVP with same eigenvalues and eigenvectors as described in the introduction. An easy way to overcome this situation is to introduce an arbitrary shift $\sigma \in \mathbb{C}$ in the spectrum. Assuming $\sigma \neq \lambda_j \forall j$ (it is not equal to any of the unknown eigenvalues), the steps below may be employed:

$$\begin{aligned} J\mathbf{c} &= \lambda M\mathbf{c} \\ J\mathbf{c} - \sigma M\mathbf{c} &= \lambda M\mathbf{c} - \sigma M\mathbf{c} \\ (J - \sigma M)\mathbf{c} &= (\lambda - \sigma)M\mathbf{c} \\ B\mathbf{c} &= \beta\mathbf{c} \end{aligned}$$

where $B = (J - \sigma M)^{-1}M$ and $\beta = (\lambda - \sigma)^{-1}$. Since $J$ is invertible, $\sigma = 0$ is a consistent choice. This new EVP has a shifted version of the previous spectrum:

- $\lambda_j$ such that $|\lambda_j - \sigma| >> 1$ are mapped near the origin, in other words, $|\beta_j| \approx 0$. The infinite eigenvalues are turned into exactly zeros.
- $\lambda_j$ such that $|\lambda_j - \sigma| << 1$ are mapped to a $\beta_j$ with high absolute value.

This new transformed problem can now be solved with EVP solvers, and later on the relation between each $\beta_j$ with $\lambda_j$ and $\sigma$ may be used to obtain the eigenvalues. Most importantly, the shifted spectrum favors the inner workings of iterative eigensolvers such as the Arnoldi iteration. The latter is often used to compute certain regions of the spectrum, these regions being related to the maginitude of the eigenvalues. More about this method is presented subsequently.

## 4.2 ARNOLDI ITERATION

Since the objective is to detect if a there is a single $\lambda_j$ such that $\Re\{\lambda_j\} > 0$, not all of the eigenvalues are necessary, which in turn means that only a portion of the

eigenvalues could be computed. Moreover, the idea of shifting the eigenvalues is a interesting one when iterative methods such as the Arnoldi iteration are used. This method constructs at each step a Krylov subspace. The latter is defined as:

$$\mathcal{K}_s := span\left\{\mathbf{c}_0, B\mathbf{c}_0, B^2\mathbf{c}_0, ..., B^{s-1}\mathbf{c}_0\right\} \tag{4.1}$$

where $\mathbf{c}_0 \in \mathbb{C}^{(r\times r)}$ (for the sake of simplicity, the size will be denoted by $r$) is an arbitrary initial vector. This subspace's basis is composed of the powers of the matrix $B$, similar to the Power iteration and Simultaneous iteration/QR iteration. The underlying idea of using matrix powers is that the invariant subspace in each iteration remains the same, while the eigenvalues are raised to a $k-th$ power (at the $k-th$ step), highlighting the eigenvalues with the largest modulus. More details are present in [11].

The way Arnoldi iteration constructs $\mathcal{K}_s$ is not as straightforward as the definition above suggests. The set of vectors of later iterations have nearly converged to the eigenvector of the eigenvalue with largest modulus (essentially the same idea as the Power method), which numerically poses ill conditioning caused by the floating point arithmetic failing to preserve the linear indepedence in the set. Given this issue, the following preposition from [34] provides a path to construct a well-behaved basis for $\mathcal{K}_s$:

**Proposition 4.1.** *For $B \in \mathbb{C}^{r\times r}$ and nonzero $\mathbf{c} \in \mathbb{C}^r$, there is an unitary matrix $Q \in \mathbb{C}^{r\times r}$ such that $Q\mathbf{e}_1 = \gamma\mathbf{c}$ for some $\gamma \neq 0$ and $H = Q^*BQ$ is upper Hessenberg. A matrix is defined as upper Hessenberg if $h_{ij} = 0 \quad \forall i, j$ such that $i > j + 1$ (a structure almost identical to that of an upper triangular matrix).*

The proposition above states that every matrix has a Hessenberg matrix with the exactly same eigenvalues, with the corresponding eigenvectors related by a linear transform (one that essentially changes the coordinate system). This can be checked by doing:

$$B = QHQ^*$$

Supposing $\mathbf{v}_j$ is an eigenvector of$B$ associated with eigenvalue $\lambda_j$

$$B\mathbf{v}_j = QHQ^*\mathbf{v}_j$$

$$\lambda_j\mathbf{v}_j = QHQ^*\mathbf{v}$$

$$\lambda_j Q^*\mathbf{v}_j = HQ^*\mathbf{v}$$

$$\text{Denoting } \mathbf{w}_j = Q^*\mathbf{v}_j$$

$$\lambda_j\mathbf{w}_j = H\mathbf{w}_j$$

same eigenvalues, $\mathbf{w}_j$ and $\mathbf{v}_j$ are related.

(4.2)

Once again using the proposition 4.1, $B = QHQ^* \rightarrow \boxed{BQ = HQ}$ for some upper Hessenberg matrix $H \in \mathbb{C}^{r\times r}$. Considering only the first $s$ columns ($s < r$), this equation is obtained:

$$BQ_s = Q_{s+1}\tilde{H}_s \tag{4.3}$$

where $Q_s \in \mathbb{C}^{m\times s}$, $Q_{s+1} \in \mathbb{C}^{m\times(s+1)}$ and $\tilde{H}_s \in \mathbb{C}^{(s+1)\times s}$. A recursion process is defined based on the last column of equation 4.3:

$$B\begin{bmatrix} | & \dots & | \\ \mathbf{q}_1 & \dots & \mathbf{q}_s \\ | & \dots & | \end{bmatrix} = \begin{bmatrix} | & \dots & | & | \\ \mathbf{q}_1 & \dots & \mathbf{q}_s & \mathbf{q}_{s+1} \\ | & \dots & | & | \end{bmatrix} \begin{bmatrix} \times & \times & \times & \dots & \times & h_{1s} \\ \times & \times & \times & \dots & \times & h_{2s} \\ 0 & \times & \times & \dots & \times & h_{3s} \\ 0 & 0 & \times & \dots & \times & h_{4s} \\ \dots & \dots & \dots & \dots & & \dots \\ 0 & 0 & 0 & \dots & \times & h_{ss} \\ 0 & 0 & 0 & \dots & 0 & h_{(s+1)s} \end{bmatrix},$$

$$B\mathbf{q}_s = h_{1s}\mathbf{q_1} + h_{2s}\mathbf{q_2} + \dots + h_{ss}\mathbf{q_s} + h_{(s+1)s}\mathbf{q_{s+1}}$$

$$\mathbf{q_{s+1}} = \frac{B\mathbf{q_s} - \sum_{j=1}^{s} h_{js}\mathbf{q_j}}{h_{(s+1)s}}, \quad \text{where } h_{(s+1)s} = ||B\mathbf{q_s} - \sum_{j=1}^{s} h_{js}\mathbf{q_j}||$$

$\Rightarrow$ The recursion starts with an arbitrary unit vector $\mathbf{q}_1$.

The subspace $span\,\{\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3, ..., \mathbf{q}_{s+1}\}$ is the same as $\mathcal{K}_s$, only written in terms of an orthonormal set of vectors. This new basis's construction is an iterative process, depending only of the matrix-vector product $B\mathbf{q}_j$ and a Gram-Schmidt orthogonalizaton process. Compared to the previous basis, it is more numerically stable because of the orthogonality between the $\mathbf{q}_j$ as well as their normalization as unit vectors (in floating arithmetic, loss of significance could happen).

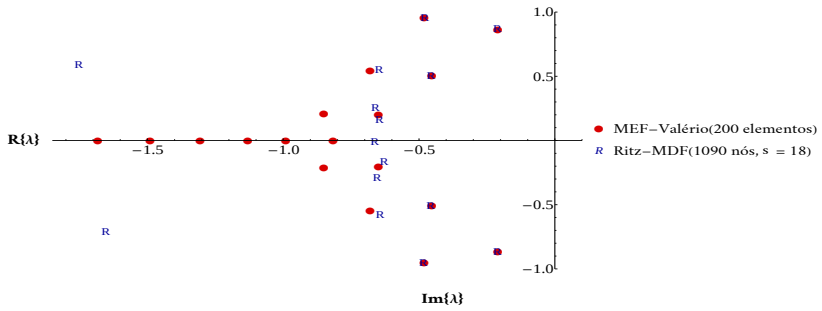
How the eigenvalues are computed still needs to be discussed. As described in equation 4.2, the full matrices $B$ and $H$ share the same spectrum, while the same cannot be said for $B$ and $\tilde{H}$ (the latter is not even a square matrix). By doing analogous steps, it is possible to construct a square matrix $H_s$ as outlined below:

$$\begin{aligned} BQ_s &= Q_{s+1}\tilde{H}_s \\ Q_s^*BQ_s &= Q_s^*Q_{s+1}\tilde{H}_s \\ Q_s^*BQ_s &= Q_s^*Q_{s+1}\tilde{H}_s \\ Q_s^*BQ_s &= \tilde{I}\ \ \tilde{H}_s \\ &\downarrow \\ H_s &= Q_s^*BQ_s \end{aligned} \tag{4.4}$$

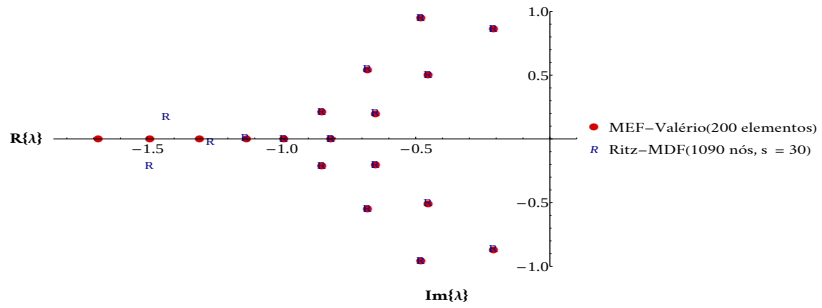
which is basically $\tilde{H}_s$ without its last line. This matrix can be interpreted as the orthogonal projection of $B$ in the Krylov subspace $\mathcal{K}_s$ with respect to the basis $\{\mathbf{q}_1, ..., \mathbf{q}_s\}$ [31]. The eigenvalues of $H$ are estimates of some of the eigenvalues of $B$ (these are named the Ritz values). They can be obtained with the QR algorithm and using that $H$ is upper Hessenberg to obtain $O(s^2)$ FLOPS per iteration (employing Givens rotations in each QR factorization and noting that the upper Hessenberg structure is preserved along the iterations [34]).
If necessary, it is also possible to obtain eigenvector estimates (Ritz vectors) after computing the eigenvectors $\mathbf{t}_j$ of $H_s$. Using the same arguments as in equation 4.2, the approximated eigenvector $\mathbf{v}_j$ of $B$ is obtained by doing $\mathbf{v}_j = Q_s\mathbf{t}_j$.

Using a simple implementation over the previous Coeutte problem for Krylov subspaces sizes $s = 18, 30, 47$ and $\sigma = 0.1$ (shift), it is possible to visualize the

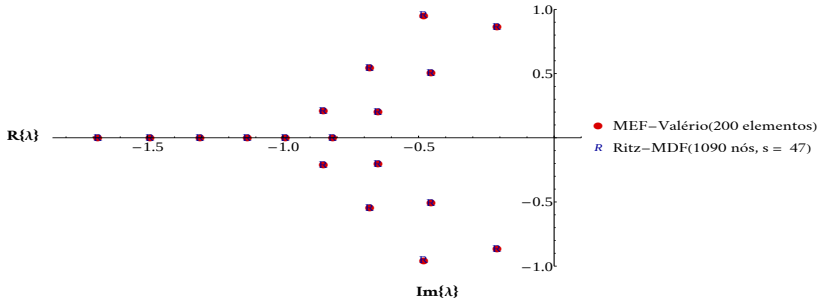
convergence of some Ritz value:




(a) Arnoldi (FD) $s = 18$ × Valerio [33] (FE)




(b) Arnoldi (FD) $s = 30$ × Valerio [33] (FE)




(c) Arnoldi (FD) $s = 47$ × Valerio [33] (FE)

Figure 14: Comparison of the spectrum using Arnoldi iteration (with shift invert transform after FD discretization) and Valerio [33] (QZ after FE discretization).

Even with different discretizations and eigensolvers, visually both are in accordance when $s = 47$. The largest absolute difference between the eigenvalues on Figure 14 was exactly 0.000392586. It is worth noting that instead of solving an EVP for a matrix of size $3269 \times 3269$, a much smaller matrix of size $47 \times 47$ was constructed and then its eigenvalues were computed, on the other hand the shift $\sigma = 0.1$ was chosen based on previous knowledge of this particular flow's spectrum.

The convergence properties of the method is not yet fully understood, in [13] some of the recent results are summarized. On the other hand, it is known that the method solves at each iteration a minimization problem (for more details see appendix A).

## 4.3 VALÉRIO TRANSFORM

Another method that aims to reduce the GEVP problem size removing the infinite eigenvalues from the GEVP is presented in [33]. A transformation was developed by looking at the observable blocked structure of the matrices, as well as assuming that $N_p = N_v - 1$ (such as the case of FD with a staggered grid).

$$\begin{array}{c} \\ N_v \\ N_v \\ N_v-1 \end{array}\!\!\begin{array}{c} \begin{array}{ccc} N_v & N_v & N_v-1 \end{array} \\ \left[\begin{array}{c|c|c} J_{11} & J_{12} & J_{13} \\ \hline J_{21} & J_{22} & J_{23} \\ \hline J_{31} & J_{32} & 0 \end{array}\right] \end{array}\!\!\begin{bmatrix} u' \\ — \\ v' \\ — \\ p' \end{bmatrix} = \lambda \begin{array}{c} \\ N_v \\ N_v \\ N_v-1 \end{array}\!\!\begin{array}{c} \begin{array}{ccc} N_v & N_v & N_v-1 \end{array} \\ \left[\begin{array}{c|c|c} M_{11} & 0 & 0 \\ \hline 0 & M_{22} & 0 \\ \hline 0 & 0 & 0 \end{array}\right] \end{array}\!\!\begin{bmatrix} u' \\ — \\ v' \\ — \\ p' \end{bmatrix}$$

$$\Downarrow$$

Removing the 4 boundary conditions, this results in

$$\begin{array}{c} \\ N_v-1 \\ N_v-3 \\ N_v-1 \end{array}\!\!\begin{array}{c} \begin{array}{ccc} N_v-1 & N_v-3 & N_v-1 \end{array} \\ \left[\begin{array}{c|c|c} J_{11} & J_{12} & J_{13} \\ \hline J_{21} & J_{22} & J_{23} \\ \hline J_{31} & J_{32} & 0 \end{array}\right] \end{array}\!\!\begin{bmatrix} u' \\ — \\ v' \\ — \\ p' \end{bmatrix} = \lambda \begin{array}{c} \\ N_v-1 \\ N_v-3 \\ N_v-1 \end{array}\!\!\begin{array}{c} \begin{array}{ccc} N-1 & N_v-3 & N-1 \end{array} \\ \left[\begin{array}{c|c|c} M_{11} & 0 & 0 \\ \hline 0 & M_{22} & 0 \\ \hline 0 & 0 & 0 \end{array}\right] \end{array}\!\!\begin{bmatrix} u' \\ — \\ v' \\ — \\ p' \end{bmatrix} \tag{4.5}$$

The removal of the boundary counditions not only slightly reduces the GEVP size, but also makes the blocks $J_{31}$ and $J_{13}$ square matrices. This will be necessary for the transformation matrices construction afterwards.

Even though it is not a algorithmic strategy to solve the GEVP, computing

the roots of the charateristic polynomial defined by the derterminant of the matrix $A(\lambda) = J - \lambda M$ is a possibility:

$$\begin{array}{c} \\ N_v{-}1 \\ N_v{-}3 \\ N_v{-}1 \end{array} \begin{array}{c} \overset{N_v-1}{} \\ \left[\begin{array}{c|c|c} A_{11}(\lambda) = J_{11} - \lambda M_{11} & A_{12} = J_{12} & A_{13} = J_{13} \\ \hline A_{21} = J_{21} & A_{22}(\lambda) = J_{22} - \lambda M_{22} & A_{23} = J_{23} \\ \hline A_{31} = J_{31} & A_{32} = J_{32} & 0 \end{array}\right] \end{array} \overset{\begin{array}{c} N_v-3 \\ \\ N_v-1 \end{array}}{} \begin{bmatrix} u' \\ \hline v' \\ \hline p' \end{bmatrix} = \begin{bmatrix} 0 \\ \hline 0 \\ \hline 0 \end{bmatrix} \tag{4.6}$$

$$A(\lambda) \qquad \mathbf{c} \quad = \quad 0 \tag{4.7}$$

The idea is to apply two elementary transforms which are essentially a blocked two sided Gaussian elimination. Assuming $J_{31}$ and $J_{13}$ are invertible, the transform $T_L$ is applied from the left to introduce zeros in $A_{23}$ and $T_R$ from the right to introduce zeros in $A_{32}$:

$$A(\lambda)\ \mathbf{c} = 0$$

$$T_L\ A(\lambda)\ T_R\ \mathbf{d} = 0, \quad \text{where } \mathbf{d} = {T_R}^{-1}\mathbf{c}, \forall \mathbf{c} \tag{4.8}$$

$$\tilde{A}(\lambda)\ \mathbf{d} = 0 \tag{4.9}$$

$$\text{where} \quad T_L = \left[\begin{array}{c|c|c} I_{[m]} & 0 & 0 \\ \hline -J_{23}J_{13}^{-1} & I_{[2n-m-b]} & 0 \\ \hline 0 & 0 & I_{[m]} \end{array}\right], \quad T_R = \left[\begin{array}{c|c|c} I_{[m]} & -J_{31}^{-1}J_{32} & 0 \\ \hline 0 & I_{[2n-m-b]} & 0 \\ \hline 0 & 0 & I_{[m]} \end{array}\right]$$

$$\Downarrow$$

$$\tilde{A}(\lambda) = \left[\begin{array}{c|c|c} \tilde{A}_{11}(\lambda) & \tilde{A}_{12}(\lambda) & \tilde{A}_{13} \\ \hline \tilde{A}_{21}(\lambda) & \tilde{A}_{22}(\lambda) & 0 \\ \hline \tilde{A}_{31} & 0 & 0 \end{array}\right] \tag{4.10}$$

The reason of the previous two steps still may not be clear, but the calculation of the eigenvalues just turned out to be considerably easier. As previously mentioned,

these are the roots of the charateristic polynomial $p(\lambda) = \det\left(\tilde{A}(\lambda)\right)$. Considering that permutations only change the determinant's sign and that the determinant of a triangular block matrix is the product of the determinants of each block along the diagonal [24]:

$$
\begin{aligned}
F\tilde{A}(\lambda) &= \left[\begin{array}{c|c|c} \tilde{A}_{31} & 0 & 0 \\ \hline \tilde{A}_{21}(\lambda) & \tilde{A}_{22}(\lambda) & 0 \\ \hline \tilde{A}_{11}(\lambda) & \tilde{A}_{12}(\lambda) & \tilde{A}_{13} \end{array}\right] \\
p(\lambda) &= \pm \det\left(F\tilde{A}(\lambda)\right) \\
&= \pm \det(\tilde{A}_{31}) \times \det\left(\tilde{A}_{22}(\lambda)\right) \times \det(\tilde{A}_{13}) \\
&= \tau \det\left(\tilde{A}_{22}(\lambda)\right), \quad \tau \in \mathbb{C}
\end{aligned} \tag{4.11}
$$

Surprisingly the expression shows that only the block $\tilde{A}_{22}(\lambda)$ is associated with the polynomial roots, that is, it is the only block that is related to the finite eigenvalues. This results shows that a reduction from $3N_v - 5$ to the problem size to $N_v - 3$. In pratice, the matrix $\tilde{A}(\lambda)$ and its characteristic polynomial are not constructed. Instead, the elementary transforms $T_L$ and $T_R$ are applied to $J$ and $M$ (after the boundary conditions are removed from the system).

$$
\begin{aligned}
\tilde{A}(\lambda)\mathbf{d} &= 0 \\
T_L \ A(\lambda) \ T_R \, \mathbf{d} &= 0 \\
T_L \ (J - \lambda M) \ T_R \, \mathbf{d} &= 0 \\
T_L \ J \ T_R \, \mathbf{d} &= T_L \ \lambda M \ T_R \, \mathbf{d} \\
\tilde{J}\mathbf{d} &= \lambda \tilde{M}\mathbf{d}
\end{aligned} \tag{4.12}
$$

$$\Downarrow$$

$$\tilde{J}_{22}\mathbf{d}_2 = \lambda \tilde{M}_{22}\mathbf{d}_2 \tag{4.13}$$

$$\tilde{J}_{22} = J_{22} + (-J_{23}J_{13}^{-1}J_{12}) + (-J_{23}J_{13}^{-1}J_{11} + J_{21})(J_{31}^{-1}J_{32}) \tag{4.14}$$

$$\tilde{M}_{22} = M_{22} + (-J_{23}J_{13}^{-1}M_{11} + J_{21})(J_{31}^{-1}J_{32}) \tag{4.15}$$

The matrices $\tilde{J}_{22}$ and $\tilde{M}_{22}$ are now dense and inversible matrices. It is now possible to solve an EVP $\boxed{\tilde{M}_{22}^{-1}\tilde{J}_{22}\mathbf{d}_2 = \lambda\mathbf{d}_2}$ or $\boxed{\tilde{J}_{22}^{-1}\tilde{M}_{22}\mathbf{d}_2 = \lambda^{-1}\mathbf{d}_2}$ and then use QR iteration (as done in [33]) or even use Arnoldi on the reduced problem (considering that certain regions of the finite spectrum are not very interesting to be computed) as this work does. This makes Valerio transform quite similar to the Orr-Sommerfeld operator, but while the former assumes certain properties of the flow, the latter requires that certain matrix sizes inside each $3 \times 3$ block are met.

Independent of the method of choice, the eigenvectors of the original problem can be obtained from equation 4.9:

$$\begin{aligned} \tilde{A}(\lambda)\mathbf{d} &= 0 \\ &\downarrow \\ \tilde{A}_{11}(\lambda)\mathbf{d}_1 + \tilde{A}_{12}(\lambda)\mathbf{d}_2 + \tilde{A}_{13}\mathbf{d}_3 &= 0 \\ \tilde{A}_{21}(\lambda)\mathbf{d}_1 + \tilde{A}_{22}(\lambda)\mathbf{d}_2 &= 0 \\ \tilde{A}_{31}\mathbf{d}_1 &= 0 \end{aligned} \tag{4.16}$$

Essentially the linear system 4.16 has to be solved. As $\tilde{A}_{31}$ is invertible, $\mathbf{d}_1 = 0$. From that, $\tilde{A}_{22}(\lambda)\mathbf{d}_2 = 0$ is obtained after substitution on the above equation. This is essentially the new reduced GEVP problem, and $\mathbf{d}_2$ will be the eigenvector associated with the $\lambda$ in question. Finally, it is possible to conclude that $\mathbf{d}_3 = -\tilde{A}_{13}^{-1}\tilde{A}_{12}(\lambda)\mathbf{d}_2$ by substitution of both $\mathbf{d}_2$ and $\mathbf{d}_1$. The full eigenvector $\mathbf{d}$ is then equal to

$$\mathbf{d} = \begin{bmatrix} 0 \\ \mathbf{d}_2 \\ \mathbf{d}_3 \;=\; -\tilde{A}_{13}^{-1}\tilde{A}_{12}(\lambda)\mathbf{d}_2 \end{bmatrix} \begin{matrix} N_v - 1 \\ N_v - 3 \\ N_v - 1 \end{matrix} \tag{4.17}$$

Even so, $\mathbf{d}$ is not an eigenvector of the original problem. This is quickly handled by using the relation 4.8 previously set, requiring only the following matrix-vector product: $\mathbf{c} = T_R\,\mathbf{d}$.

After discussing the strategies alternatives to the GEVP, their results (separately and even together) to both Couette and Poiseulli are discussed in the next chapter.

# 5 RESULTS

All the tests were done using Octave 4.0.1, with a Intel Core 2 Quad Q9400(6M Cache, 2.66 GHz, 1333 MHz FSB) and 4GB RAM memory desktop. The methods comparison was only done with FD discretization, due to the fact that $N_p = N_v - 1$ (a requirement of Valério transform) and the sparsity structure can be explored.

The results were obtained by the following 5 procedures:

- Execute the QZ algorithm to the original GEVP $J\mathbf{c} = \lambda M\mathbf{c}$.
- Use Valerio transform and then execute QZ algorithm to the reduced problem $\tilde{J}_{22}\mathbf{d} = \lambda\tilde{M}_{22}\mathbf{d}$.
- Use Valerio transform and then execute QR algorithm over the EVP $\tilde{J}_{22}^{-1}\tilde{M}_{22}\mathbf{d} = \lambda\mathbf{d}$.
- Use Valerio transform and then realize an Arnoldi iteration on the EVP $\tilde{J}_{22}^{-1}\tilde{M}_{22}\mathbf{d} = \lambda\mathbf{d}$. It is worth mentioning that to our knowledge these steps were not emplyoed together before.
- Apply shift-invert over the original GEVP, using the ARPACK [19]. It is a highly optimized variant of the implicitly restarted Arnoldi iteration (IRA), detailed in [1].

## 5.1 COUETTE FLOW

The first 18 eigenvalues computed are ilustrated in Figure 15. There are noticeable differences in eigenvalues near $\Re\{\lambda_j\} = -1$. This error is solved in [33] by permuting the lines of $J$ and $M$ before applying the transform. This was not done in this work because while both discretizations are valid as Figure 13 depicted, the FD did not appear to have an obvious permutation in the same vein as the FE formulation.

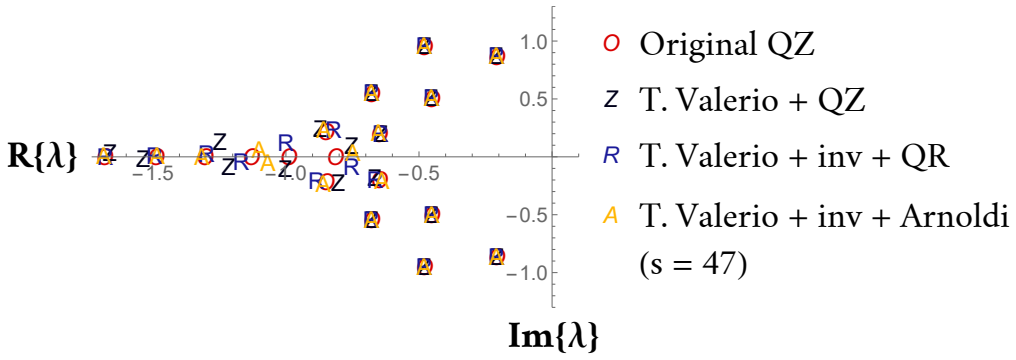



Figure 15: Comparison between all procedures in Couette flow (except ARPACK's IRA that is virtually the same as QZ).

All strategies' execution time were measured, to quantify if there was an advantage of avoiding the use of QZ. The entire solution process (matrices construction, transform and GEVP/EVP solution) was done for different node sizes so as to perceive differences as the problem dimension grows. The obtained results are presented in Figure 16.

As expected, QZ is worst one in total running time. The new proposed strategy (first using Valério transform and then Arnoldi iteration) had the second best running time, losing only to ARPACK (IRA). After the 9 runs, it appeared that the bottleneck lied in the transform (accounting for approximately 90% of the total time). This suggests that if the linear systems in 4.14, 4.15 are solved more efficiently, the total running time could rival ARPACK IRA.

## 5.2 POISEULLI FLOW

The first 18 eigenvalues were computed with all procedures once again. Curiously, the reduced problem did not contain any noisy error in a region of the spectrum like Couette. What is interesting in this case was the difference between Arnoldi and other procedures. The motive behind the discrepancy can be explained by noting that the eigenvalues are more clustered, a fact that slows Arnoldi (too many eigenvalues of similar large magnitudes). Increasing the Krylov subspace size was a

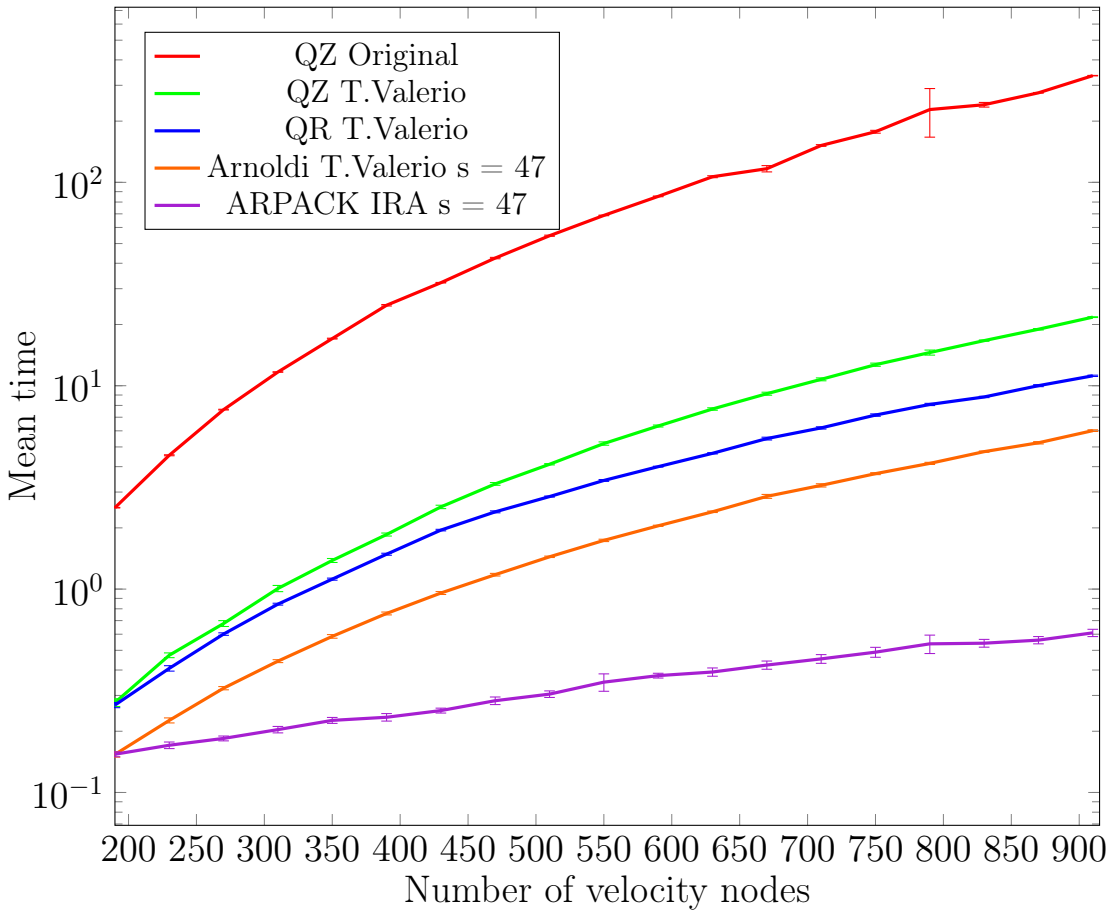



Figure 16: Mean execution time (in seconds) for each strategy present in Figure 15 and ARPACK IRA after 9 runs. The t-value confidence interval has 98% of probability.

way to treat this issue. The results are presented below from Figure 17 to Figure 20:

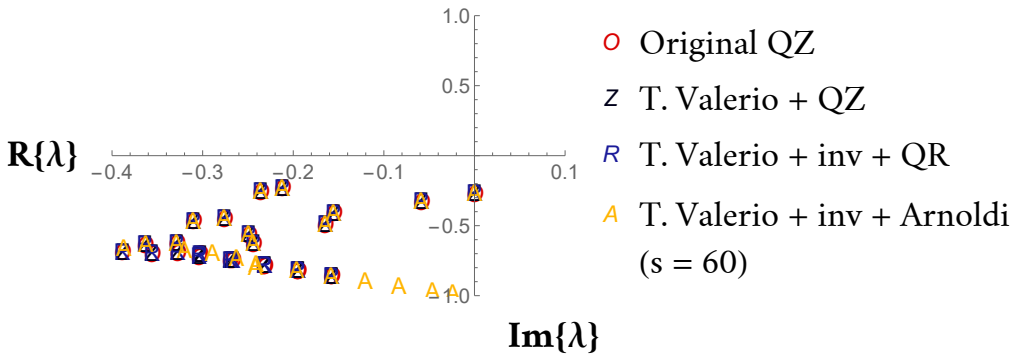



Figure 17: Comparison using $Re = 5772$, $\alpha = 1.02$ and Arnoldi with $s = 60$

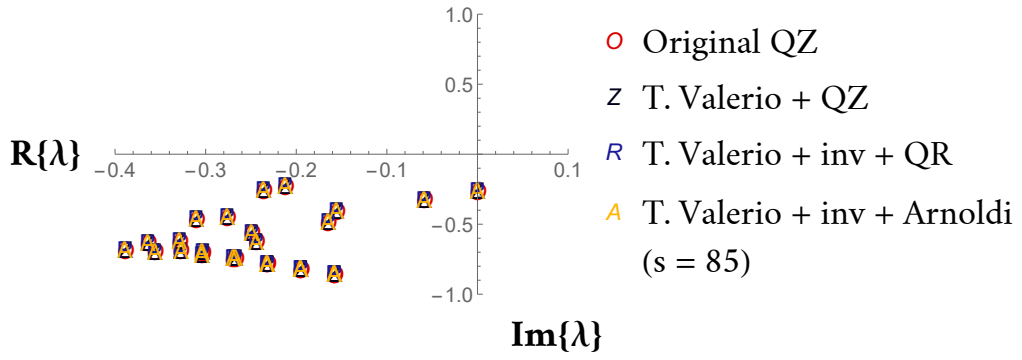



Figure 18: Comparison using $Re = 5772$, $\alpha = 1.02$ and Arnoldi with $s = 85$ (better Ritz values convergence)

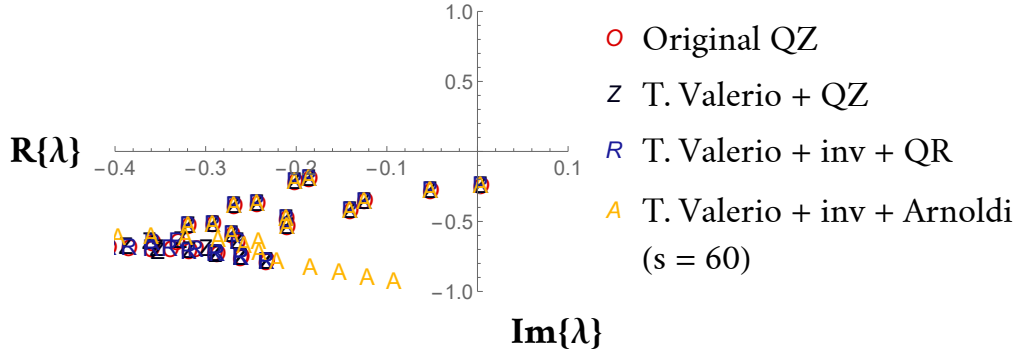



Figure 19: Comparison using $Re = 10^4$, $\alpha = 1.02$ and Arnoldi with $s = 60$

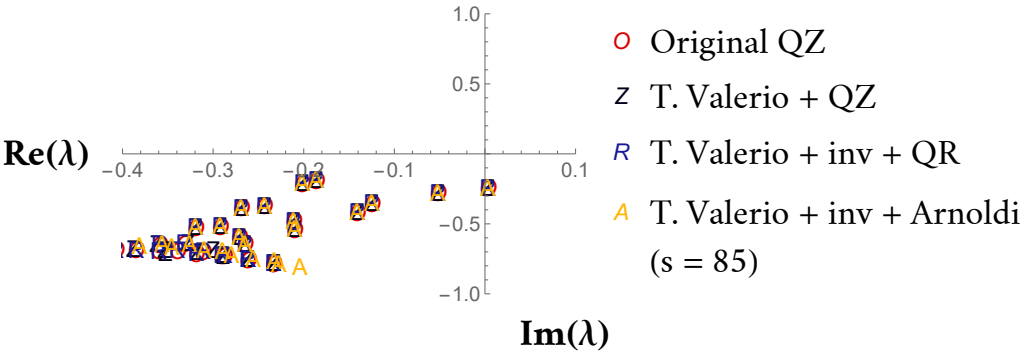



Figure 20: Comparison $Re = 10^4$, $\alpha = 1.02$ and Arnoldi with $s = 85$ (better Ritz values convergence).

Since the subspace size increase was not considerable, the running times were pretty much the same as the Couette results.

# 6 CONCLUSIONS AND FUTURE WORKS

Over the development of this work it was possible to observe that the approach used to solve the linear stability of flows implies in how much time and accurate the final solution will be. Concepts of how perturbing a base flow and solving the GEVP were applied successfuly for both paralell flows, where the literature results were met even with the new tried strategies. The most important observations that could be drawn over the development were:

- If a bad discretization scheme is employed over a system of differential equations, it will not matter if an efficient solver is used in sequence, as the results are already compromised by the incoherence between the differential and the difference equations.
- The way a problem is discretized guides which method to be used to numerically solve a continuous problem, as the corresponding matrices have properties associated with the discretization used and a computational linear algebra method depends on these properties to work properly.

For the future, there is still a lot to explore as answers to the questions below:

1. How hard will it be to replicate the procedure in bidimensional (non paralell) and tridimensional flows? Is Valerio transform adaptable to this situation?
2. What if a global state of art discretization such as Nek5000 spectral elements was used? If the blocked dense matrices are difficult to represent in memory (eg. in three dimensional flows), is the sparse nature of larger finite differences/elements discretizations a beter choice?
3. How does this rival against the recent matrix-free/exponential time stepper methods?
4. Are there ways to speed up the process by paralelizing it with CUDA (GPU)[22]?

5. In Valério transform, is it possible to avoid the actual construction of $\tilde{J}_{22}$ and $\tilde{M}_{22}$, such that a representation (a form of matrix decomposition that can be used at linear systems solving later on) in an attempt to preserve sparsity?

6. Is is possible to "zero out" the $J_{23}$ and $J_{32}$ blocks using orthogonal transforms such as reflections or rotations (Householder or Givens Rotations [31])?

In the future, the answers to these questions should help in a deeper understanding of linear hydrodynamic stability problems, discretization of differential operators and generalized eigenvalues problems and obtain the respective solution in a fast and reliable manner.

# REFERÊNCIAS

[1] B. Lehoucq, R., C. Sorensen, D., e Yang, C. Arpack users' guide: Solution of large scale eigenvalue problems with implicitly restarted arnoldi methods.

[2] Bazzi, M. S. Breakup dynamics of non-newtonian thin liquid sheets. Tese de Mestrado, Pontíficia Universidade Católica do Rio de Janeiro, 2018.

[3] Blackford, S. Direct Methods. Disponível em `http://www.netlib.org/utk/people/JackDongarra/etemplates/node282.html`. Acesso em: 2 de setembro de 2018.

[4] Burden, R., e Faires, J. *Numerical Analysis*. Cengage Learning, 2010.

[5] Craik, A. D. Lord kelvin on fluid mechanics. *The European Physical Journal H 37*, 1 (Jun 2012), 75–114.

[6] Date, A. Fluid dynamical view of pressure checkerboarding problem and smoothing pressure correction on meshes with colocated variables. *International Journal of Heat and Mass Transfer 46*, 25 (2003), 4885 – 4898.

[7] Drazin, P., Reid, W., e Books24x7, I. *Hydrodynamic Stability*. Cambridge Mathematical Library. Cambridge University Press, 2004.

[8] et al, J. W. E. GNU Octave 4.0.1. Disponível em `https://www.gnu.org/software/octave/`. Acesso em: 2 de setembro de 2018.

[9] Fischer, P. F., Lottes, J. W., e Kerkemeier, S. G. NEK5000. Disponível em `https://nek5000.mcs.anl.gov/index.php/Main_Page`. Acesso em: 2 de setembro de 2018.

[10] Fujino, M., Yoshizaki, Y., e Kawamura, Y. Natural-laminar-flow airfoil development for a lightweight business jet.

[11] Golub, G. H., e Van Loan, C. F. *Matrix Computations*, third ed. The Johns Hopkins University Press, 1996, pages 330-335; 375-387.

[12] Householder, A. *The Theory of Matrices in Numerical Analysis*. Dover Books on Mathematics. Dover Publications, 2013.

[13] Jarlebring, E. Lecture notes in matrix computations for large-scale systems, Autumn 2015.

[14] Karampatziakis, N., e Mineiro, P. Discriminative features via generalized eigenvectors. In *Proceedings of the 31st International Conference on Machine Learning* (Bejing, China, 22–24 Jun 2014), E. P. Xing e T. Jebara, Eds., vol. 32 of *Proceedings of Machine Learning Research*, PMLR, pp. 494–502.

[15] Karniadakis, G., e Sherwin, S. *Spectral/hp Element Methods for Computational Fluid Dynamics: Second Edition*. Numerical Mathematics and Scientific Computation. OUP Oxford, 2013.

[16] Khorrami, M. R. A chebyshev spectral collocation method using a staggered grid for the stability of cylindrical flows. *International Journal for Numerical Methods in Fluids 12*, 9, 825–833.

[17] Koren, Y., Carmel, L., e Harel, D. Ace: a fast multiscale eigenvectors computation for drawing huge graphs. In *IEEE Symposium on Information Visualization, 2002. INFOVIS 2002.* (Oct 2002), pp. 137–144.

[18] Kreith, F. *Fluid Mechanics*. Mechanical/Chemical engineering. Taylor & Francis, 1999.

[19] Lehoucq, R., Maschhoff, K., Sorensen, D., e Yang, C. ARPACK. Disponível em `https://www.caam.rice.edu/software/ARPACK/`. Acesso em: 2 de setembro de 2018.

[20] LOISEAU, J.-C. *Dynamics and global stability analysis of three-dimensional flows*. Theses, Ecole nationale supérieure d'arts et métiers - ENSAM, maio de 2014.

[21] Nam, J., e Carvalho, M. Linear stability analysis of two-layer rectilinear flow in slot coating. 2503 – 2512.

[22] NVIDIA. Cuda. Disponível em `https://developer.nvidia.com/cuda-zone`, 6. Acesso em: 2 de setembro de 2018.

[23] Orszag, S. A. Accurate solution of the orr–sommerfeld stability equation. *Journal of Fluid Mechanics 50*, 4 (1971), 689–703.

[24] Powell, P. D. Calculating Determinants of Block Matrices. *ArXiv e-prints* (dezembro de 2011).

[25] Research, W. Wolfram Mathematica 11. Disponível em `https://www.wolfram.com/mathematica/`. Acesso em: 2 de setembro de 2018.

[26] Reynolds, O. An experimental investigation of the circumstances which determine whether the motion of water shall be direct or sinuous, and of the law of resistance in parallel channels. *Philosophical Transactions of the Royal Society of London 174* (1883), 935–982.

[27] Runge, C. Uber empirische Funktionen und die Interpolation zwischen aquidistanten Ordinaten. *Zeitschrift fur Mathematik und Physik 46* (1901), 224–243.

[28] Saad, Y. Numerical methods for large eigenvalue problems, pages 220-229.

[29] SHERWIN, S. J., e BLACKBURN, H. M. Three-dimensional instabilities and transition of steady and pulsatile axisymmetric stenotic flows. *Journal of Fluid Mechanics 533* (2005), 297–327.

[30] Trefethen, L. N. *Spectral Methods in MATLAB, pages 1-8,41-49.* 2000.

[31] Trefethen, L. N., e Bau, D. *Numerical Linear Algebra, pages 250-264.* SIAM, 1997.

[32] Turkboylari, M., e Stuber, G. L. Eigen-matrix pencil method-based velocity estimation for mobile cellular radio systems. In *2000 IEEE International Conference on Communications. ICC 2000. Global Convergence Through Communications. Conference Record* (June 2000), vol. 2, pp. 690–694 vol.2.

[33] Valério, J., Carvalho, M., e Tomei, C. Filtering the eigenvalues at infinite from the linear stability analysis of incompressible flows. *Journal of Computational Physics 227*, 1 (2007), 229 – 243.

[34] Watkins, D. S. The qr algorithm revisited. *SIAM review 50*, 1 (2008), 133–145.

[35] Whitby, M., e Quirke, N. Fluid flow in carbon nanotubes and nanopipes. *Nature Nanotechnology 2*, 2 (2007), 87.

# APPENDICES

## APPENDIX A – The minimization problem behind Arnoldi iteration

Before enunciating the problem, it seems reasonable to look at the relation established between the characteristic polynomial $p(t)$ and $p(A)$, where $A \in \mathbb{C}^{r\times r}$ and $t$ are it's eigenvalues. According to the Cayley-Hamilton theorem [12],

$$\begin{aligned} p(t) &= \det(A - tI) = \gamma_0 t^0 + \gamma_1 t + ... + \gamma_{r-1} t^{r-1} + t^r = 0 \\ p(A) &= \gamma_0 I + \gamma_1 A + ... + \gamma_{r-1} A^{r-1} + A^r = 0_{r\times r} \end{aligned} \tag{6.1}$$

Which in turn means that

$$A^r = \gamma_0 I + \gamma_1 A + ... + \gamma_{r-1} A^{r-1} = q(A)$$

Informally, as its eigenvalues are the roots of the charateristic polynomial, the matrix $A$ is also a "root" of the polynomial $p$ (not exactly because it results on a null matrix and not a scalar). Also, the matrix $A^r$ is a linear combination of the lower powers of $A$ as denoted by the polynomial $q(A)$. Back to Krylov subspaces, there is a subspace $\mathcal{K}_s$ such that every one of its vector $\mathbf{v}$ may be written as some linear combination

$$\mathbf{v} = \theta_0 \mathbf{c}_0 + \theta_1 A\mathbf{c}_0 + ... + \theta_{s-1} A^{s-1}\mathbf{c}_0$$

Defining the following polynomials :

$$\begin{aligned} \tilde{q}(A) &= \theta_0 I + \theta_1 A + ... + \theta_{s-1} A^{s-1} \\ \tilde{p}(A) &= \tilde{q}(A) - A^s \end{aligned} \tag{6.2}$$

The Arnoldi iteration solves at each iteration the following optimization problem:

| **Find** $\tilde{q}$ in the space of polynomials of degree $s-1$ such that |
|---|
| $\|\|\tilde{p}(A)\mathbf{c_0} = \tilde{q}(A)\mathbf{c}_0 - A^s\mathbf{c}_0\|\| =$ minimum |

The interpretation is that, while $A^s\mathbf{c}_0$ cannot be written in terms of $\tilde{q}(A)$, Arnoldi tries to find the polynomial that in a least squares sense makes the distance between the subspace and $A^s\mathbf{c}_0$ minimum. If the Arnoldi iteration goes further until the $r-1th$ power (and assuming it does not break down), the solution would be the $q(A)$ as defined in 6.1. The Ritz values (eigenvalues of $H_s$) are the roots of the polynomial $\tilde{p}(A)$ [31].