# FYS-3900
# Master's thesis in physics

## Simulations of plasma filament dynamics in toroidal geometry

Petter Rønningen

May 16, 2010

FACULTY OF SCIENCE AND TECHNOLOGY
Department of Physics and Technology
University of Tromsø

# FYS-3900
# Master's thesis in physics

## Simulations of plasma filament dynamics in toroidal geometry

Petter Rønningen

May 16, 2010

# Abstract

Interchange driven propagation of so-called "blobs" arises in basic laboratory plasmas, in the scrape-off layer of fusion plasmas and as equatorial spread F (ESF) phenomena in the Earth's ionosphere. The propagation of such blobs has been demonstrated in the VTF experiment at MIT and in numerical simulations. Experimental results show that the velocity of these blobs is inversely proportional to the neutral gas pressure. These blobs have also shown the tendency to form mushroom-like shapes.

We here develop a two-field model for the interchange motions in magnetised plasma in toroidal geometry. This model is then used to simulate the dynamics of propagating blobs using a two dimensional advection-diffusion solver. Studying the effects of ion-neutral collisions and large blob amplitudes, we observe the separation between an inertia dominated and a collisionally dominated regime. We also observe that this separation persists for all simulated amplitudes.

We further perform simulations with parameters set to closely resemble the VTF experiments. These simulations show that the experimental result of the velocity being inversely proportional to the neutral gas pressure is in a qualitative match with the simulated results for the collisionally dominant regime.

# Preface

Seeing this thesis finally come together marks the end of a five year journey here in Tromsø. This last year has been a busy but remarkable experience.

I would like to thank my supervisor Odd Erik Garcia for his valubable feedback and the insights he has provided during this year. His lectures on plasma physics in my third year sparked an interest in both plasma physics and physics in general that has since then only grown.

I would also like to thank Noam Katz at MIT for providing the data from the VTF experiments as well as providing me with additional details on the experiments.

Finally, I would like to thank my dearest Victoria for being so patient, supporting and motivating during this year.

I will end this with the words of the Irish poet and Nobel laureate William Butler Yeats which reflects well, even after long hours of work, on the end of this year:

> *"Education is not the filling of a pail, but the lighting of a fire."*

Petter Rønningen
Tromsø, May 2010

# Contents

# 1 Introduction

Interchange driven propagation of so-called "blobs", localised perturbations or field aligned filaments on a background plasma [9], arises in basic laboratory plasmas, in the scrape-off layer of fusion plasmas as well as in space plasma such as the so-called Equatorial Spread-F (ESF) phenomena [11, 15].

In fusion plasmas, a scrape-off layer using magnetic fields intersecting at designated collectors is commonly used in an attempt to control the exhaust of plasma and heat by leading it along the magnetic field to target plates. However, blobs has been observed to quickly propagate across the field lines of this scrape-off layers, effectively escaping the scrape-off layer and hitting the chamber walls of the fusion tokamaks. This poses a problem both in term of loss of heat and energy from the fusion plasma itself, as well as the release of polluting particles into the fusion plasma.

The ESF phenomena, arising on the night side plasma in the F-layer for low lattitudes, is a large scale "localised" depletion of plasma relative to the background plasma density propagating upwards in the radial direction across the magnetic field of the Earth. The size of these depletions often range from centimeters to several hundred kilometers [11, 15]. This phenomenon cause disruption in communications and navigation as they give localised changes in the way communication signals interact with the ionsphere, for example posing difficulties when attempting to send communication signals over the horizon using refraction in the ionsphere.

As the phenomenas are driven by an interchange mechanism, either through the curvature and radial dependence of the magnetic field for fusion plasmas or the gravitation for ESF, we will in this thesis develop and implement a cylindrical advection-diffusion model for the simulation of transport of plasma perturbations across the magnetic field based on existing two-field models [8]. We will further use this to simulate propagating perturbations in an attempt to investigate how the perturbation amplitude and the neutral background gas affects the propagation and view this in light of the ESF propagation speed found in [16].

Furthermore, we shall use this simulation code in an attempt to investigate experiments on the Versatile Toroidal Facility (VTF) at MIT [13], where a relation between propagation speed of blobs and the background neutral density was found. We shall also investigate how the "mushroom"-like shape observed in these experiments and predicted in earlier simulations using a similar model [9] relate to our numerical experiments.

# 2 Transport in magnetised plasmas

In this chapter we seek to derive a model for fluctuation driven transport in magnetised plasmas. We will first briefly describe the physical mechanics behind the dynamics in section 2.1. Motivated by this, we will seek to describe the plasma dynamics by a simple two-field model in section 2.2.

## 2.1 Fluctuation-induced transport

Plasmas in a non-uniform magnetic field are subject to mechanisms resulting in transport of particles perpendicular to the magnetic field. We will here map out some of the important physical mechanisms behind this phenomena.

We will consider a confined homogeneous background plasma subject to a radially dependent magnetic field. This magnetic field is curved and depends on the radial distance from the center as in a toroidal plasma chamber (such as the VTF or Blåmann device). We will show that this curvature of the magnetic field and its radial dependency give rise to a charge polarisation on any perturbation to the homogeneous background plasma. Furthermore, we will see that this charge polarisation will drive the plasma perturbation across the magnetic field lines.

It is worth noting that we are looking for slowly varying effects on convection of plasma resulting in transport of entire filament structures. Due to this, we wish to disregard high frequency variations such as magnetic field waves and sound waves.

Let us now consider the inhomogeneous but azimuthal symmetric and radially dependent magnetic field described as

$$\mathbf{B} = -\frac{B_0 R_0}{R}\mathbf{b} \tag{1}$$

This magnetic field is considered to be parallel with the azimuthal direction, and uniform in the vertical direction. The local background plasma is initially considered homogeneous and we will assume quasineutrality, with the electron density being equal to the ion density. We will let the parallel and perpendicular motion relate to the direction of motion relative to the magnetic field direction. Furthermore, we will in this thesis assume no or negible variations along the magnetic field.

In this view, the fluid model of plasma is a good approximation as we are only concerned with motion perpendicular to the magnetic field.

For simplicity of argument we take the momentum equation for a plasma species, disregarding collisions and viscosity

$$mn\frac{d\mathbf{u}}{dt} = qn\left(\mathbf{E} + \mathbf{u}\times\mathbf{B}\right) - \nabla p \tag{2}$$

where $m$ is the mass of the plasma particle, $n$ is the particle density, $\mathbf{u}$ is the fluid element velocity, $q$ is the plasma particle charge, $\mathbf{E}$ is the electric field and $p$ is the pressure. As we are describing fluid motion, note that the time differential describes the convective derivative $d_t = (\partial_t + \mathbf{u} \cdot \nabla)$. Taking the vector product with the magnetic field ($\mathbf{B}\times$Eq. (2)) gives us the perpendicular drifts

$$\begin{aligned} \mathbf{u}_\perp &= -\frac{1}{B^2}\mathbf{B}\times\mathbf{E} + \frac{1}{qnB^2}\mathbf{B}\times\nabla p + \frac{m}{qB^2}\mathbf{B}\times\frac{d\mathbf{u}}{dt} \\ &= \mathbf{u}_\mathrm{E} + \mathbf{u}_\mathrm{d} + \mathbf{u}_\mathrm{p} \end{aligned} \tag{3}$$

where the first term describes the electric drift $\mathbf{u}_\mathrm{E}$, an uniform plasma drift transporting plasma perpendicular to both the magnetic and electric field in bulk regardless of particle charge. The second term describe the diamagnetic drift $\mathbf{u}_\mathrm{d}$ driven by pressure gradients in the plasma, perpendicular to both the magnetic field and the pressure gradient. The third term describes the polarisation drift $\mathbf{u}_\mathrm{p}$ which in this simple picture can be thought of as a balance drift to conserve quasineutrality as the diamagnetic drift is charge separating.

It is worth noting that the diamagnetic drift in the fluid picture for this curved magnetic field gives a contribution equal to the contribution of gradient and curvature drifts in the guiding center picture [9].

We shall see that the charge separating diamagnetic drift current gives, for a perturbation on the background plasma, to a vertical electric field. This is the driving mechanism behind the perpendicular motion: The resulting vertical electric field gives rise to electric drift $\mathbf{u}_\mathrm{E}$ (or $E \times B$-drift as it is also called) perpendicular to both the magnetic field and the electric field. This drift is not charge separating, and consequently a plasma bulk motion directed outwards in the radial direction at the center of the perturbation.

Furthermore, by virtue of quasineutrality, the charge polarisation of the perturbation "blob" is balanced by the polarisation current $\mathbf{u}_\mathrm{p}$ due to the time evolution of the electric field as well as the advection of the electric drift. For the positive perturbation this results in a vorticity dipole with counter-clockwise rotation around the top of the blob and clock around the bottom.

Figure 1 illustrates the transport mechanism of a typical blob, with the vertical electric field from polarisation giving us the electric outward drift. $\kappa$ is here the curvature and gradient vector for the magnetic field.

## 2.2 Modelling the transport: The two field model

We now seek to find a simple model for the evolution of the particle density and vorticity in a magnetised plasma that accounts for the transport phenomenas described above. This model will be based on the fluid picture of plasma due to the assumption of no or little variations or drift along the magnetic field, which makes this a good approximation.

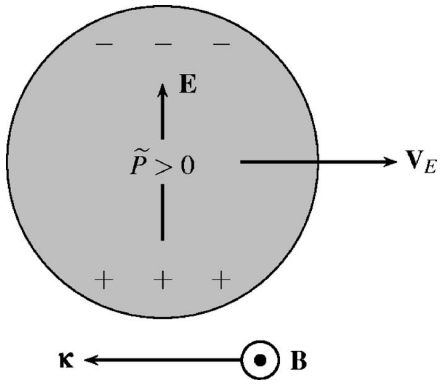


Figure 1: Illustration of the ideal interchange mechanism for a blob structure. We note the vertical electric field giving rise to the electric drift opposite to the curvature vector $\kappa$. Figure taken from [9].

### 2.2.1 Plasma motion

For any magnetised plasma, we have the momentum equation for particle species $\alpha$, assuming electrostatic motions and an isothermal fluid, given as

$$m_\alpha n_\alpha \frac{d\mathbf{u}_\alpha}{dt} = q_\alpha n_\alpha \left(-\nabla\phi + \mathbf{u}_\alpha \times \mathbf{B}\right) - T_\alpha \nabla n_\alpha - \nabla\cdot\mathbf{\Pi}_\alpha - \sum_\beta m_\alpha n_\alpha \nu_{\alpha\beta} \left(\mathbf{u}_\alpha - \mathbf{u}_\beta\right) \tag{4}$$

(as described in for example [6, 8]) where $m_\alpha$ is the particle mass, $n_\alpha$ is the particle density, $\mathbf{u}_\alpha$ is the plasma species fluid velocity, $q_\alpha$ is the plasma species particle charge, $\phi$ is the electrostatic potential and $T_\alpha$ is the species' temperature, $\mathbf{\Pi}_\alpha$ is the viscous stress tensor, $\mathbf{B}$ is the magnetic field vector and $\nu_{\alpha\beta}$ is the collision frequency between species $\alpha$ and $\beta$. Note that the differential we use here is the convective derivative $d_t = (\partial_t + \mathbf{u}_\alpha \cdot \nabla)$ describing advection by the fluid velocity.

We first get the projected plasma fluid velocity perpendicular to $\mathbf{B}$ by taking the vector product of the magnetic field and momentum equation, $\mathbf{B}\times$Eq. (4). We choose to rewrite the magnetic field as $\mathbf{B} = B\mathbf{b}$ where $\mathbf{b}$ is the unit vector in the magnetic field direction. We note that $\mathbf{B} \times (\mathbf{u}_\alpha \times \mathbf{B}) = B^2\mathbf{u}_{\perp\alpha}$ for the second term on the right hand side allows us to solve for $\mathbf{u}_{\perp\alpha}$:

$$\begin{aligned} \mathbf{u}_{\perp\alpha} &= \frac{1}{B}\mathbf{b}\times\nabla\phi + \frac{T_\alpha}{m_\alpha B}\mathbf{b}\times\nabla\ln n_\alpha + \frac{m_\alpha}{q_\alpha B}\mathbf{b}\times\frac{d\mathbf{u}_\alpha}{dt} \\ &+ \sum_\beta \frac{m_\alpha \nu_{\alpha\beta}}{q_\alpha B}\mathbf{b}\times(\mathbf{u}_\alpha - \mathbf{u}_\beta) + \frac{1}{q_\alpha n_\alpha B}\mathbf{b}\times(\nabla\cdot\mathbf{\Pi}_\alpha) \end{aligned} \tag{5}$$

The first two terms describes the electric (or $\mathbf{E}\times\mathbf{B}$) drift and diamagnetic

drift respectively. The third term describes polarisation drift, while the two remaining terms describes momentum loss due to collisions and due to viscosity. While the electric and diamagnetic drifts are independent of the plasma velocity, the polarisation drift and the collision term do depend on the velocity $\mathbf{u}_\alpha$.

As we are assuming no or negible drift or variations along the magnetic field, the perpendicular picture gives for our use the full picture.

When then assume low frequency plasma variations ($d\mathbf{u}_\alpha/dt \ll \omega_{c\alpha}$) and a low frequency particle collision rate ($\nu_{\alpha\beta} \ll \omega_{c\alpha}$). The electron motion $\mathbf{u}_e$ can to lowest order be decomposed into

$$\mathbf{u}_e = \mathbf{u}_E + \mathbf{u}_{de} + O\left(\frac{\omega}{\omega_{ce}}\right) \tag{6}$$

where $\mathbf{u}_E$ is the electric drift and $\mathbf{u}_{de}$ is the electron diamagnetic drift. We can then iterate on Eq. (5) using this lowest order approximation for the electron fluid drift motion $\mathbf{u}_e$.

We can find the perpendicular motion using the first order approximated electron motion from Eq. (6) in Eq. (5), neglecting the electron inertia and assuming only collisions with a stationary neutral gas ($\mathbf{u}_n \approx 0$) we get

$$\mathbf{u}_{\perp e} = \frac{1}{B}\mathbf{b}\times\nabla\phi - \frac{T_e}{eB}\mathbf{b}\times\nabla\ln n_e + \frac{m_e\nu_{en}}{eB}\mathbf{b}\times\left(\frac{1}{B}\mathbf{b}\times\nabla\phi - \frac{T_e}{eB}\mathbf{b}\times\nabla\ln n_e\right)$$

which with the electron cyclotron frequency given as $\omega_{ce} = eB/m_e$ becomes

$$\mathbf{u}_e = \mathbf{u}_E + \mathbf{u}_{de} + \frac{\nu_{en}}{\omega_{ce}}\frac{T_e}{eB}\nabla_\perp\left(\frac{e\phi}{T_e} - \ln n_e\right) \tag{7}$$

For the ion motion we will assume no ion drift along the magnetic field lines due to high inertia, so-called flute modes, as well as the ions being cold, resulting in no diamagnetic drift for the ions. The first order approximation of the ion motion then becomes

$$\mathbf{u}_i = \mathbf{u}_E + O\left(\frac{\omega}{\omega_{ci}}\right)$$

which iterated with Eq.(5) and assuming only collisions with a stationary neutral gas, we get

$$\begin{aligned}\mathbf{u}_i &= \mathbf{u}_E + \frac{m_i}{q_iB}\mathbf{b}\times\frac{d\mathbf{u}_E}{dt} + \frac{m_i\nu_{in}}{q_iB}\mathbf{b}\times\mathbf{u}_E \\ &= \mathbf{u}_E - \frac{1}{\omega_{ci}B}\frac{d\nabla_\perp\phi}{dt} - \frac{\nu_{in}}{\omega_{ci}}\frac{\nabla_\perp\phi}{B}\end{aligned} \tag{8}$$

where we have the ion cyclotron frequency $\omega_{ci} = eB/m_i$. We now have expressions for the ion and electron drift which we will further use in the particle continuity and charge continuity equations in the following sections.

### 2.2.2 Particle continuity

Introducing the particle continuity equation

$$\frac{\partial n_\alpha}{\partial t} + \nabla \cdot (n_\alpha \mathbf{u}_\alpha) = 0$$

for species $\alpha$, assuming quasineutrality ($n_\mathrm{i} \approx n_\mathrm{e} = n$) we can write the equation for electron and ion continuity as

$$\frac{\partial n}{\partial t} + \nabla \cdot (n\mathbf{u}_\mathrm{E} + n\mathbf{u}_\mathrm{de} + n\mathbf{u}_\mathrm{ce}) = 0 \tag{9}$$

$$\frac{\partial n}{\partial t} + \nabla \cdot (n\mathbf{u}_\mathrm{E} + n\mathbf{u}_\mathrm{pi} + n\mathbf{u}_\Pi + n\mathbf{u}_\mathrm{ci}) = 0 \tag{10}$$

where $\mathbf{u}_\mathrm{ce}$ is the electron collisional drift from Eq. (7), $\mathbf{u}_\mathrm{pi}$ is the ion polarisation drift and $\mathbf{u}_\mathrm{ci}$ is the ion collisional drift from Eq. (8). Subtracting Eq. (9) from Eq. (10) gives us the charge continuity equation

$$\nabla \cdot (n\mathbf{u}_\mathrm{pi} + n\mathbf{u}_\Pi + n\mathbf{u}_\mathrm{ci} - n\mathbf{u}_\mathrm{de} - n\mathbf{u}_\mathrm{ce}) = 0 \tag{11}$$

These two equations, Eq.(9) and Eq.(11) gives us a closed two-field model in terms of the particle density $n$ and potential $\phi$.

### 2.2.3 The cylindrical coordinates

We will now specify the coordinates used. Let us assume the cylindrical coordinates $(R, \Theta, Z)$ where $\hat{\mathbf{R}}$ is in the outward radial direction, $\hat{\mathbf{Z}}$ is in the upward vertical direction and $\hat{\mathbf{\Theta}}$ is in the azimuthal direction. We let the magnetic field be defined as

$$\mathbf{B} = \frac{B_0 R_0}{R}\mathbf{b}$$

where $B_0$ is the magnetic field strength at the radial position $R_0$ and $\mathbf{b} = -\hat{\mathbf{\Theta}}$. With $\mathbf{a} = (a_R, 0, a_Z)$, we introduce the divergence operator with components only perpendicular to the magnetic field

$$\nabla \cdot \mathbf{a} = \frac{1}{R}\frac{\partial}{\partial R}(R\, a_R) + \frac{\partial a_Z}{\partial Z} \tag{12}$$

and the gradient operator

$$\nabla a = \left(\frac{\partial a}{\partial R}, 0, \frac{\partial a}{\partial Z}\right)$$

which gives us the perpendicular Laplace operator

$$\nabla_\perp^2 a = \frac{1}{R}\frac{\partial}{\partial R}\left(R\,\frac{\partial a}{\partial R}\right) + \frac{\partial^2 a}{\partial Z^2}$$

Finally, we have the curl operator

$$\nabla \times \mathbf{a} = \left(0, \frac{\partial a_R}{\partial Z} - \frac{\partial a_Z}{\partial R}, 0\right)$$

### 2.2.4 Compression and advection of plasma

We can in the particle and charge continuity equations write for each of the drift terms

$$\nabla \cdot (n\mathbf{u}) = \nabla n \cdot \mathbf{u} + n\nabla \cdot \mathbf{u} = (\nabla n + n\nabla) \cdot \mathbf{u}$$

where the first term in the last expression gives advection and the second term gives compression of the plasma. Using this gives us the divergence of the ion polarisation drift and ion collisional drift as

$$\begin{aligned} \nabla \cdot (n\mathbf{u}_{\mathrm{pi}}) &= (\nabla n + n\nabla) \cdot \left( \frac{m_{\mathrm{i}}}{q_{\mathrm{i}} B^2} \frac{d\nabla_\perp \phi}{dt} \right) \\ \nabla \cdot (n\mathbf{u}_{\mathrm{ci}}) &= (\nabla n + n\nabla) \cdot \left( \frac{m_{\mathrm{i}} \nu_{\mathrm{in}}}{q_{\mathrm{i}} B^2} \nabla_\perp \phi \right) \end{aligned}$$

It can be shown (given in [7]) that the contributing drift term from the viscous stress tensor takes the form

$$\nabla \cdot (n\mathbf{u}_\Pi) = \mu \nabla_\perp^4 \phi$$

where $\mu$, given in Eq. (16) the following section, is a constant dependent on several factors such as the ion temperature. The electron collision drift becomes

$$\nabla \cdot (n\mathbf{u}_{\mathrm{ce}}) = (\nabla n + n\nabla) \cdot \left[ \frac{\nu_{\mathrm{en}}}{\omega_{\mathrm{ce}}} \frac{nT_{\mathrm{e}}}{q_{\mathrm{e}} B} \nabla_\perp \left( \frac{e\phi}{T_{\mathrm{e}}} - \ln n_{\mathrm{e}} \right) \right]$$

While the electric and diamagnetic drifts are incompressible for a homogeneous magnetic field ($\nabla \cdot \mathbf{u}_{\mathrm{E}} = 0$ and $\nabla \cdot \mathbf{u}_{\mathrm{d}\alpha} = 0$), an inhomogeneous magnetic field as given in Eq. (1) makes these drifts compressible. Using the cylindrical divergence operator (Eq. 12) on the electric drift gives us

$$\begin{aligned} \nabla \cdot \mathbf{u}_{\mathrm{E}} &= \nabla \cdot \left( \frac{1}{B} \mathbf{b} \times \nabla \phi \right) \\ &= -\frac{1}{R} \frac{\partial}{\partial R} \left( \frac{R}{B} \frac{\partial \phi}{\partial Z} \right) + \frac{\partial}{\partial Z} \left( \frac{1}{B} \frac{\partial \phi}{\partial R} \right) \\ &= -\frac{2}{BR} \frac{\partial \phi}{\partial Z} \end{aligned}$$

and similar for the diamagnetic drift

$$\nabla \cdot \mathbf{u}_{\mathrm{d}\alpha} = \nabla \cdot \left( \frac{T_\alpha}{q_\alpha B} \mathbf{b} \times \nabla \ln n \right) = -\frac{2T_\alpha}{n q_\alpha BR} \frac{\partial n}{\partial Z}$$

We note that in the limiting regime of a uniformly magnetised plasma, $R \to \infty$, these drifts become incompressible. Using the above expressions we find the terms in the particle continuity equation

$$\begin{aligned}\nabla\cdot(n\mathbf{u}_{\mathrm{E}}) &= n\,(\nabla\cdot\mathbf{u}_{\mathrm{E}}) - \frac{2n}{BR}\frac{\partial\phi}{\partial Z}\\ \nabla\cdot(n\mathbf{u}_{\mathrm{de}}) &= -\frac{2T_{\mathrm{e}}}{q_{\mathrm{e}}BR}\frac{\partial n}{\partial Z}\end{aligned}$$

Using these above expressions in Eq. (11) and dividing by $n$, we can write it as

$$\begin{aligned}(\nabla\ln n+\nabla)\cdot\left(\frac{m_{\mathrm{i}}}{q_{\mathrm{i}}B^2}\frac{d\nabla_\perp\phi}{dt}+\frac{m_{\mathrm{i}}\nu_{\mathrm{in}}}{q_{\mathrm{i}}B^2}\nabla_\perp\phi\right)+\frac{2T_{\mathrm{e}}}{q_{\mathrm{e}}BR}\frac{\partial\ln n}{\partial Z}\\ =\mu\nabla_\perp^4\phi-(\nabla\ln n+\nabla)\cdot\left[\frac{\nu_{\mathrm{en}}}{\omega_{\mathrm{ce}}}\frac{T_{\mathrm{e}}}{eB}\nabla_\perp\left(\frac{e\phi}{T_{\mathrm{e}}}-\ln n\right)\right]\end{aligned}\quad(13)$$

and we can write Eq. (9) as

$$\begin{aligned}(\partial_t+\nabla\cdot\mathbf{u}_{\mathrm{E}})\ln n-\frac{2}{BR}\frac{\partial\phi}{\partial Z}-\frac{2T_{\mathrm{e}}}{q_{\mathrm{e}}BR}\frac{\partial\ln n}{\partial Z}\\ =-(\nabla\ln n+\nabla)\cdot\left[\frac{\nu_{\mathrm{en}}}{\omega_{\mathrm{ce}}}\frac{T_{\mathrm{e}}}{eB}\nabla_\perp\left(\frac{e\phi}{T_{\mathrm{e}}}-\ln n\right)\right]\end{aligned}\quad(14)$$

This are our model equations in their dimensional form.

### 2.2.5 Normalisation by perturbation scale length

We will now turn to simplify and normalise Eq. (13) and Eq. (14) to non-dimensional equations. The normalisation carried out in will be to the characteristic blob size $l$ and characteristic time $\gamma$: $\gamma t\rightarrow t'$ and $\mathbf{R}/l\rightarrow\mathbf{R}'$ , thus giving us

$$\frac{d}{dt}\rightarrow\gamma\frac{d}{dt'}$$

for the temporal scale, and

$$\nabla\rightarrow\frac{1}{l}\nabla'$$

for the spatial scales. To keep non-dimensionality, the scaling of the potential (since velocities are normalised by $\gamma l$) then becomes

$$\phi\rightarrow B_0l^2\gamma\phi'$$

We introduce the dimensionless vorticity as $\Omega'=\nabla_\perp'^2\phi'$. We will for simplicity here assume that the convective term $\nabla\ln n$ is small compared to the compression term. The treatment of this term is included in section 2.4.

Using this in Eq. (13) gives us, dropping the primes for simplicity of notation, the equation

$$\frac{d\Omega}{dt}+\frac{c_{\mathrm{s}}^2}{\gamma^2}\frac{2}{R_0l}\frac{\partial\ln n}{\partial Z}=\frac{Be}{l^2\gamma m_{\mathrm{i}}}\mu\nabla_\perp^2\Omega-\frac{\nu_{\mathrm{in}}}{\gamma}\Omega$$

where we have defined the sound speed $c_\mathrm{s}$ as

$$c_\mathrm{s} = \sqrt{\frac{T_\mathrm{e}}{m_\mathrm{i}}}$$

We can then choose to set the characteristic time $\gamma$ to the ideal interchange time

$$\gamma^2 = c_\mathrm{s}^2 \frac{2}{R_0 l} = \frac{g}{l} \tag{15}$$

where we define the effective gravity $g = 2c_\mathrm{s}^2/l$. Using $\nu_\mathrm{i} = \nu_\mathrm{in}/\gamma$ and the (rather lengthy) expression given for $\mu$ in [7]:

$$\mu_\Omega = \frac{\mu}{\gamma l^2} = \frac{3}{96} \frac{u_\mathrm{ti}^2 n Z_\mathrm{i}^4 e^4 \ln\Lambda}{\omega_\mathrm{ci}^2 \gamma l^2 \pi^{3/2} \varepsilon_0^2 m_\mathrm{i}^{1/2} T_\mathrm{i}^{3/2}} \tag{16}$$

with the ion gyration frequency $\omega_\mathrm{ci} = eB/m_\mathrm{i}$, $u_\mathrm{ti}$ is the ion thermal velocity, $Z_\mathrm{i}$ is the ion charge state, $\ln\Lambda$ is the Colomb logarithm and $\varepsilon_0$ is the permitivity in vacuum, we get the non-dimensional equation for vorticity $\Omega$:

$$\frac{d\Omega}{dt} + \frac{\partial \ln n}{\partial Z} = \mu_\Omega \nabla_\perp^2 \Omega - \nu_\mathrm{i} \Omega \tag{17}$$

Let us summarise the terms in this equation in order, for clarity: First term represents the polarisation drift, while the second term is the interchange term from the diamagnetic drift advection ($\nabla B$ and curvature drift advection in the guiding center picture). The right hand side holds the diffusive viscosity and the ion-neutral collisional drag. Noe that the term $\partial_Z \ln n$ in the vorticity equation is the interchange term which is the driving mechanism for the propagation of perturbations as mentioned in section 2.1.

The continuity equation (14) becomes with our normalisation

$$\frac{d \ln n}{dt} - \frac{2}{R_0}\frac{\partial \phi}{\partial Z} - \frac{1}{\gamma l^2}\frac{2T_\mathrm{e}}{eB_0 R_0}\frac{\partial \ln n}{\partial Z} = \frac{1}{\gamma l^2}\frac{\nu_\mathrm{en}}{\omega_\mathrm{ce}}\frac{T_\mathrm{e}}{eB}\left[\nabla_\perp^2 \ln n + (\nabla_\perp \ln n)^2\right]$$

where we by using using

$$\xi = \frac{1}{\gamma l^2}\frac{2T_\mathrm{e}}{eB_0 R_0} \qquad \mu_\mathrm{n} = \frac{1}{\gamma l^2}\frac{\nu_\mathrm{en}}{\omega_\mathrm{ce}}\frac{T_\mathrm{e}}{eB}$$

get the simplified non-dimensional equation for the particle density $n$:

$$\frac{d \ln n}{dt} - \frac{2}{R_0}\frac{\partial \phi}{\partial Z} - \xi \frac{\partial \ln n}{\partial Z} = \mu_\mathrm{n}\left[\nabla_\perp^2 \ln n + (\nabla_\perp \ln n)^2\right] \tag{18}$$

Summarising the terms in this equation in order: The first two terms represents the electric drift advection and compression, the third term represent the diamagnetic drift advection ($\nabla B$ and curvature drift advection in the guiding center picture) and the right hand side represent collisional diffusion. For the

case of the MHD ordering, we have no diamagnetic advection and thus neglect the third term here. This also have the advantage of reducing the parameter space.

Keep in mind that we here have used the convective derivative $d_t = \partial_t + \mathbf{u}_\mathrm{E} \cdot \nabla$. We can then combine this with Eq. (17) as our model equations on the form of inhomogeneous diffusion equations

$$\frac{d\Omega}{dt} = \mu_\Omega \nabla_\perp^2 \Omega - \nu_\mathrm{i} \Omega - \frac{\partial \ln n}{\partial Z} \tag{19}$$

$$\frac{d \ln n}{dt} = \mu_\mathrm{n} \nabla_\perp^2 \ln n + \mu_\mathrm{n} \left(\nabla_\perp \ln n\right)^2 + \frac{2}{R_0} \frac{\partial \phi}{\partial Z} + \xi \frac{\partial \ln n}{\partial Z} \tag{20}$$

We here identify the Rayleigh and Prandtl number expressed by the effective diffusive viscosity $\mu_\Omega$ and diffusion $\mu_\mathrm{n}$ as

$$\mathrm{Ra} = \frac{1}{\mu_\Omega \mu_\mathrm{n}} \qquad \mathrm{Pr} = \frac{\mu_\Omega}{\mu_\mathrm{n}}$$

where the Rayleigh number is the ratio of effective buoyancy to dissipative forces and the Prandtl number is the the ratio of viscosity to diffusion.

### 2.2.6 Small perturbation amplitude

Noting that for small perturbations $\widetilde{n}$ with an amplitude $\Delta n$ on a background particle density $n_0$: $n = n_0 + \Delta n\, \widetilde{n}$, (that is, $\widetilde{n}$ is normalised to an amplitude of 1) we can write the driving term in the vorticity equation as

$$\frac{\partial \ln n}{\partial Z} = \frac{\partial}{\partial Z} \ln \left(1 + \frac{\Delta n\, \widetilde{n}}{n_0}\right) = \frac{\Delta n}{n_0} \frac{1}{1 + \frac{\Delta n\, \widetilde{n}}{n_0}} \frac{\partial \widetilde{n}}{\partial Z}$$

which for $\Delta n \ll n_0$ gives us

$$\frac{\partial \ln n}{\partial Z} \approx \frac{\Delta n}{n_0} \frac{\partial \widetilde{n}}{\partial Z}$$

and similar for $\partial_t \ln n$. Using this, we can define the characteristic inverse time $\gamma_{\Delta \mathrm{n}}$ as

$$\gamma_{\Delta \mathrm{n}}^2 = \gamma^2 \left(\frac{\Delta n}{n_0}\right) = \frac{2 c_\mathrm{s}^2}{Rl} \frac{\Delta n}{n_0}$$

to reduce the number of parameters in the model. The resulting equations can then be expressed as an evolution of $\Omega$ and $\widetilde{n}$ instead of $\Omega$ and $\ln n$, neglecting the quadratic term in $\Delta n / n_0$:

$$\frac{d\widetilde{n}}{dt} - \frac{2}{R_0} \frac{\partial \phi}{\partial Z} - \xi \frac{\partial \widetilde{n}}{\partial Z} = \mu_\mathrm{n} \nabla_\perp^2 \widetilde{n}$$

This results in the set of inhomogeneous diffusion equations

$$\frac{d\Omega}{dt} + \frac{\partial \widetilde{n}}{\partial Z} = \mu_\Omega \nabla_\perp^2 \Omega - \nu_\mathrm{i} \Omega \tag{21}$$

$$\frac{d\widetilde{n}}{dt} = \mu_\mathrm{n} \nabla_\perp^2 \widetilde{n} + \frac{2\widetilde{n}}{R_0} \frac{\partial \phi}{\partial Z} + \xi \frac{\partial \widetilde{n}}{\partial Z} \tag{22}$$

modeling the evolution of the independent variables $\Omega$ and $\widetilde{n}$ for small perturbation amplitudes on a background plasma.

We notice that this gives an important prediction of the scaling for the characteristic velocity $u_\mathrm{c}$ for small perturbation amplitudes as

$$\frac{u_\mathrm{c}}{c_\mathrm{s}} = \left( \frac{2l}{R} \frac{\Delta n}{n_0} \right)^{\frac{1}{2}}$$

which dictates that the velocity depends on the size $l$ of the perturbation as well as the perturbation amplitude $\Delta n$.

### 2.2.7 Velocities for high collision frequencies

Using Eq. (13) for the vorticity and considering the limiting regime of a high ion-neutral collision frequency $\nu_\mathrm{in}$, we note that we (assuming only the collisional term and interchange term to be the two major contributing terms)

$$\frac{2T_\mathrm{e}}{m_\mathrm{i} R} \frac{\partial \ln n}{\partial Z} \sim \nu_\mathrm{in} \nabla_\perp^2 \phi = \nu_\mathrm{in} \nabla \times \mathbf{u}_\mathrm{E}$$

As the first term goes as

$$\frac{2T_\mathrm{e}}{m_\mathrm{i} R} \frac{\partial \ln n}{\partial Z} \sim \frac{g}{l}$$

where $g$ is the effective gravitation described earlier, and the second term as

$$\nu_\mathrm{in} \nabla \times \mathbf{u}_\mathrm{E} \sim \frac{\nu_\mathrm{in}}{l} u_\mathrm{c}$$

with $u_\mathrm{c}$ being the characteristic velocity, we get the scaling for $u_\mathrm{c}$:

$$u_\mathrm{c} \sim \frac{g}{\nu_\mathrm{in}}$$

indicating that for high ion-neutral collision frequencies the velocity should depend inversely proportional to the ion-neutral collision frequency.

### 2.2.8 Radial transport of plasma

We will here replicate an argument given in [9] that the properties of the radial convective flux are given by the center of mass velocity, which in turn makes the

center of mass velocity an important quantity in order to study and understand the radial transport of plasma.

We define the integrated particle density

$$Q = \int d\mathbf{R}\, n$$

where $n$ is the particle density and the integral is taken over the entire plasma. The center of mass position and the center of mass velocity can then be found as

$$R_{\mathrm{COM}} = \frac{1}{Q}\int d\mathbf{R}\, Rn \qquad u_{\mathrm{COM}} = \frac{dR_{\mathrm{COM}}}{dt}$$

Now, noting that the convective drift resulting in plasma bulk motion is the electric drift, we see that the radial and vertical components of the electric drift can be expressed as

$$\mathbf{u}_{\mathrm{E}} = \frac{1}{B}\mathbf{b}\times\nabla\phi = \frac{1}{B}\left(-\frac{\partial\phi}{\partial Z}, 0, \frac{\partial\phi}{\partial R}\right)$$

Taking the mass conservation equation assuming the electric drift to be incompressible

$$\frac{\partial n}{\partial t} + \mathbf{u}_{\mathrm{E}}\cdot\nabla n = 0$$

and multiplying each term by $R$ and integrating over the entire domain gives us

$$\int d\mathbf{R}\, R\frac{\partial n}{\partial t} = \frac{\partial}{\partial t}\int d\mathbf{R}\, Rn = u_{\mathrm{COM}}$$

and

$$\int d\mathbf{R}\, \mathbf{u}_{\mathrm{E}}\cdot\nabla n = \frac{1}{B}\int d\mathbf{R}\, R\left(\frac{\partial}{\partial Z}\left(n\frac{\partial\phi}{\partial R}\right) - \frac{\partial}{\partial R}\left(n\frac{\partial\phi}{\partial Z}\right)\right)$$

where the first term vanishes if assuming periodic boundary conditions in the $Z$-direction, and using integration by parts gives us

$$\int d\mathbf{R}\, \mathbf{u}_{\mathrm{E}}\cdot\nabla n = \frac{1}{B}\int d\mathbf{R}\, n\frac{\partial\phi}{\partial Z}$$

which gives us a center of mass velocity proportional to the total radial convective flux

$$u_{\mathrm{COM}} = -\frac{1}{B}\int d\mathbf{R}\, n\frac{\partial\phi}{\partial Z}$$

It follows that all the properties of the convective flux due to an isolated plasma filament are given by the radial center of mass velocity.

## 2.3 Interchange instabilities

For a magnetised plasma with a particle density perturbation $\widetilde{n}$ on a non-uniform background particle density $n_0$ depending on the radial distance $R$ described as

$$n = n_0(R) + \widetilde{n}$$

the perturbations may give rise to linear instabilities often referred to as interchange due to both positive and negative perturbations being indefinitely magnified on certain background densities $n_0$. To investigate this, let us first consider $n_0$ as a declining particle density profile along the outward radial axis.

With a positive density perturbation $\widetilde{n} > 0$ (a blob), the electric drift described in the previous section will transport plasma radially outwards. However, as the density gradient is outward declining, the density of the plasma transported out of the local region of the perturbation will be less than of the plasma transported into the region. This will in turn cause the perturbation amplitude to grow. For a negative density perturbation $\widetilde{n} < 0$ (a depletion), the plasma transport will instead be directed inwards. This will cause a local transport of low density plasma into the perturbation and a transport of higher density plasma out of the perturbation. In both cases the initial perturbation is continouosly enhanced in the absence of any additional damping effect applied, causing an interchange instability of the perturbation.

For the opposite case of a increasing particle density profile with the radial coordinate, the perturbations will however be damped: For the positive perturbation higher density plasma will be transported out of the local region while lower density plasma will be transported in, reducing the initial perturbation amplitude. In a similar manner, a negative perturbation will cause lower density plasma to be transported out while higher density plasma is transported in.

Let us now explore the stability of this system by performing linearisation around an equilibrium state and look at the plane wave solution for the system.

We will assume the particle density $n$ and the electrostatic potential $\phi$ to be on the forms

$$\begin{aligned} n &= n_0 + \widetilde{n} \\ \phi &= \phi_0 + \widetilde{\phi} \end{aligned}$$

where $n_0$ and $\phi_0$ are the background particle density and potential, while $\widetilde{n}$ and $\widetilde{\phi}$ are the small perturbation on these backgrounds. We further assume that $n_0$ takes the form of a large scale exponential profile in the radial direction, with no variations in the vertical direction

$$n_0 = N \exp\left(-\frac{R}{L_\mathrm{n}}\right)$$

and that the background potential is zero ($\phi_0 = 0$). Using the background particle density profile, we can write

$$\ln n = \ln N + \ln\left[\exp\left(-\frac{R}{L_\mathrm{n}}\right) + \frac{\widetilde{n}}{N}\right]$$

which with the assumption that $|\tilde{n}/N| \ll 1$ allows us to approximate the second term as

$$\ln\left[\exp\left(-\frac{R}{L_\mathrm{n}}\right) + \frac{\widetilde{n}}{N}\right] \approx -\frac{R}{L_\mathrm{n}} + \ln\left(\frac{\widetilde{n}}{N}\right) \approx -\frac{R}{L_\mathrm{n}} + \frac{\widetilde{n}}{N}$$

We will also ignore any collisional or viscous effects, as these will merely provide a dampening effects and thus not contribute to the mechanisms causing the instabilities.

Linearising the equations (17) and (18) around $n_0$ and $\phi_0$, with the second order terms vanishing, gives us

$$\frac{\partial\widetilde{n}}{\partial t} - \zeta\frac{\partial\widetilde{\phi}}{\partial Z} - \xi\frac{\partial\widetilde{n}}{\partial Z} = 0$$

$$\frac{\partial\nabla_\perp^2\widetilde{\phi}}{\partial t} + \frac{\partial\widetilde{n}}{\partial Z} = 0$$

where $\zeta = l/L_\mathrm{n}$.

Introducing the plane wave solution with the wave number $\mathbf{k}$ and frequency $\omega$, and assuming local cartesian coordinates,

$$\begin{pmatrix} \tilde{n} \\ \tilde{\phi} \end{pmatrix} = \begin{pmatrix} \hat{n} \\ \hat{\phi} \end{pmatrix} \exp\left[i\left(\mathbf{k}\cdot\mathbf{R} - \omega t\right)\right]$$

this then becomes

$$\begin{aligned} -i\omega\widehat{n} + \zeta i k_Z\widehat{\phi} - \xi i k_Z\widehat{n} &= 0 \\ i k_\perp^2\omega\widehat{\phi} + i k_Z\widehat{n} &= 0 \end{aligned}$$

where $k_\perp^2 = k_R^2 + k_Z^2$, or on matrix form

$$\begin{pmatrix} -i\omega - i\xi k_Z & i\zeta k_Z \\ i k_Z & i k_\perp^2\omega \end{pmatrix}\begin{pmatrix} \widehat{n} \\ \widehat{\phi} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

which by requiring the determinant to be zero for a non-trivial solution, gives us the dispersion relation

$$k_\perp^2\omega^2 + k_\perp^2 k_Z\xi\omega + \zeta k_Z^2 = 0$$

or solved for $\omega$

$$\omega = -\frac{1}{2}k_Z\left(\xi \pm \sqrt{\xi^2 - \zeta\frac{4}{k_\perp^2}}\right)$$

For the limiting case of $\xi \to 0$ in the MHD ordering, this reduces to the well known growth rate for ideal interchange instability

$$\omega = i\sqrt{\mu}\frac{k_Z}{k_\perp}$$

Inserted into the plane wave solution, this then gives

$$\begin{pmatrix} \widetilde{n} \\ \widetilde{\phi} \end{pmatrix} = \begin{pmatrix} \widehat{n} \\ \widehat{\phi} \end{pmatrix} \psi(t) \exp\left[i\left(\mathbf{k}\cdot\mathbf{R}\right)\right]$$

where $\psi(t)$ is an exponential growth

$$\psi(t) = \exp\left[\sqrt{\kappa}\frac{k_Z}{k_\perp}t\right]$$

which only vanishes when $k_Z \ll k_\perp$, that is in the case when the plane wave has no vertical gradients in $n$ and thus no charge or vorticity polarisation due to $\nabla B$ and curvature drifts.

Restoring the dimensional units for the growth rate we get

$$\omega \sim \gamma\left(\frac{l}{L_\mathrm{n}}\right)^{\frac{1}{2}} = \left(\frac{g}{L_\mathrm{n}}\right)^{\frac{1}{2}}$$

which is the well known growth rate for interchange instabilities.

## 2.4 The convective term in the vorticity equation

We have so far neglected the non-linear convective term $\nabla \ln n$ in the vorticity equation. Although this term is in many cases dropped as it is small in magnitude (as in [8]), we will here explore the conditions where we can exclude this term. The numerical implications of including this term is covered in section 3.4.

We will explore the case of a homogeneous background plasma $N$ and a blob-like structure of the form

$$n = N + \Delta n \exp\left(-\frac{R^2}{2l}\right)$$

where$\Delta n$ is the blob amplitude. We rewrite the convective and compressional terms as

$$\left(\nabla \ln n + \nabla\right)\cdot\nabla_\perp\phi = \frac{1}{n}\left(\nabla n + n\nabla\right)\cdot\nabla_\perp\phi$$

and note that we can write each term in order of magnitude scales as

$$\nabla n \sim \frac{\Delta n}{l} \qquad n\nabla \sim \frac{N}{L}$$

where $L$ is here the scale length of the potential perturbations. Formally comparing the ratio of these two terms

$$\frac{\nabla n}{n\nabla} \sim \frac{\Delta n}{N}\frac{L}{l}$$

we see that the convective term is small and can be disregarded when the ratio of the perturbation amplitude $\Delta n$ and the background plasma $N$ is small, that is

$$\left|\frac{\Delta n}{N}\right| \ll 1$$

We are however still left with the case where the blob amplitude is actually not much smaller than the background particle density. This raises the question of the effect this term has for large blob amplitudes.

## 2.5 On quasineutrality

We will here argue for the quasineutral property of plasmas in general. This property allow us to equate the electron particle density and the ion density through charge neutrality and is thus commonly used to close equation sets. This property is driven by the shielding of charges over small length scales.

For the sake of argument, we will assume a uniform magnetic field and no plasma drift along the magnetic field. Further, we take into account that the electron mass is nonzero.

Starting with the particle continuity equation for the electrons and ions, where the last term is the polarisation drift of electrons and ions respectively

$$\frac{\partial n_e}{\partial t} + u_E \cdot \nabla n_e + \frac{m_e n_e}{eB}\frac{d\nabla_\perp^2 \phi}{dt} = 0 \tag{23}$$

$$\frac{\partial n_i}{\partial t} + u_E \cdot \nabla n_i - \frac{m_i n_i}{eB}\frac{d\nabla_\perp^2 \phi}{dt} = 0 \tag{24}$$

where we have neglected collisional dissipation. Introducing the space charge density $\rho$ through Gauss' law

$$\rho = en_i - en_e = \epsilon_0 \nabla \cdot E = -\epsilon_0 \nabla^2 \phi$$

Subtracting equation (23) from (24) we get

$$\left(1 + \frac{\rho_m}{\epsilon_0 B^2}\right)\frac{d\rho}{dt} = 0$$

where $\rho_m = m_\mathrm{e} n_\mathrm{e} + m_\mathrm{i} n_\mathrm{i}$ is the plasma mass density. The first term here describes the electric drift advection of space charge, while the second term describes the polarisation drift. In many situations the second term is of several orders of magnitude larger than the first term. This implies that we may neglect the presence of space charges for the cross field plasma dynamics. The space charge density can at any time be calculated from the divergence of the electric field.

## 2.6 Conservation of particles in cylindrical coordinates

In a domain where the boundaries are far away from the initial perturbation, no loss of particles should be expected until the perturbation reach the boundaries. Taking the equation for the particle density Eq. 22, we can find the particle evolution by integrating over the entire domain $D$ in $\mathbf{R} = (R, \Theta, Z)$, giving us

$$\frac{\partial N}{\partial t} = -\int_D d\mathbf{R} \left[\frac{1}{B}\frac{\partial}{\partial R}\left(n\frac{\partial \phi}{\partial Z}\right) - \frac{1}{B}\frac{\partial}{\partial Z}\left(n\frac{\partial \phi}{\partial R}\right)\right] + \int_D d\mathbf{R} \left(\frac{2n}{BR}\frac{\partial \phi}{\partial Z}\right)$$

where $\partial_t N$ is the time evolution of the integrated particle density, the first integral on the right hand side is the Poisson bracket and the second integral is the compression of electric drift. As we assume the advection process to take place far from the boundaries, the diffusion will not give rise to any particle loss. We have here written the Poisson bracket for the advection term $\mathbf{u}_\mathrm{E} \cdot \nabla n$ as

$$\{\phi, n\} = \frac{\partial}{\partial Z}\left(n\frac{\partial \phi}{\partial R}\right) - \frac{\partial}{\partial R}\left(n\frac{\partial \phi}{\partial Z}\right)$$

As the magnetic field is radially dependent, we express it as

$$\frac{1}{B} = \frac{R}{B_0 R_0}$$

Integrating the second term in the Poisson bracket over $dZ$ gives us, with $Z_0$ and $Z_1$ being the boundaries in the vertical direction

$$\int_D d\mathbf{R} \left[\frac{R}{B_0}\frac{\partial}{\partial Z}\left(n\frac{\partial \phi}{\partial R}\right)\right] = \int_{D_{R\Theta}} dR\, d\Theta \left(\frac{R^2}{B_0}\left[n\frac{\partial \phi}{\partial R}\right]_{Z=Z_0}^{Z=Z_1}\right) = 0$$

if we have periodicity in $Z$. (Note that we will through the numerical considerations have periodicity in $Z$, which will be elaborated in chapter 3.)

We then perform integration by parts over $dR$ on the first term in the Poisson bracket, where $R_0$ and $R_1$ are the radial boundaries:

$$\begin{aligned} -\int_D d\mathbf{R} \left[\frac{R}{B_0 R_0}\frac{\partial}{\partial R}\left(n\frac{\partial \phi}{\partial Z}\right)\right] &= -\frac{2}{B_0 R_0}\int_D d\mathbf{R} \left(n\frac{\partial \phi}{\partial Z}\right) \\ &\quad + \int_{D_{Z\Theta}} \left(\left[R^2 n\frac{\partial \phi}{\partial Z}\right]_{R=R_0}^{R=R_1}\right) dZ\, d\Theta \end{aligned}$$

Note that the second term on the right hand side vanish if we assume either the particle density $n$ or the potential derivative $\partial_Z \phi$ to vanish at the boundaries. This leaves us with a term for the integral over the Poisson bracket which is exactly cancelled by the integral over the compression of electric drift:

$$\int_D d\mathbf{R} \left(\frac{2n}{BR}\frac{\partial \phi}{\partial Z}\right) = \frac{2}{B_0 R_0}\int_D \left(n\frac{\partial \phi}{\partial Z}\right) d\mathbf{R}$$

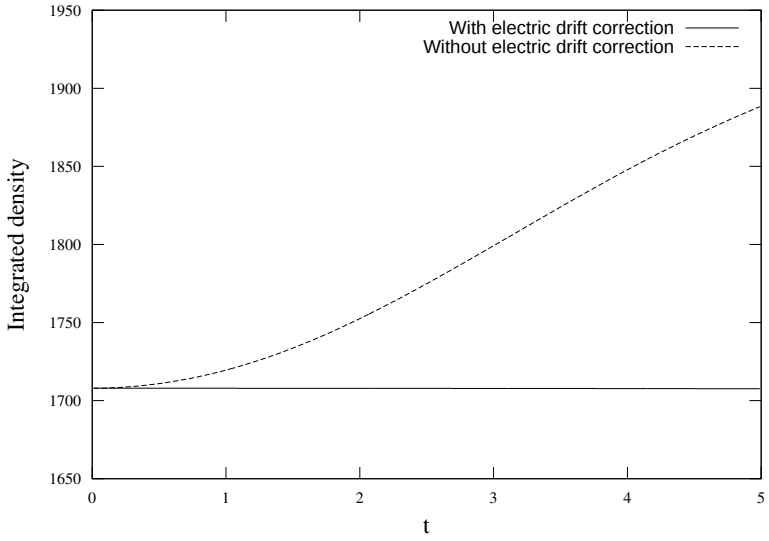


Figure 2: Integrated particle density for two simulations in cylindrical coordinates: One using the electric drift correction (continuous), the other in the absence of this correction (dashed). Note that the integrated particle density increases with time steps.

leaving us with the desired particle conservation, $\partial_t N = 0$.

This result poses a requirement for us to include the electric drift compression in the evolution equation for the particle density when solving in cylindrical coordinates to ensure particle conservation. This is in contrast to when solving in the rectangular coordinate system, where the electric drift is incompressible and the electric drift advection conserves mass.

Figure 2 shows the integrated particle density for two simulations in cylindrical coordinates: One with the compression of the electric drift included and one without it. We see that omitting this compressing term does indeed lead to an increased amount of particles while including it conserves it.

# 3 Numerical approaches

We will in this chapter describe the numerical approaches used in the two dimensional advection/diffusion solver (2DADS) code. This code is also further described in chapter 4. The discretisation used throughout this chapter is described in section 3.1. Using this, we shall develop a combined spectral/finite difference time splitting scheme for solving the model equations in cylindrical coordinates.

## 3.1 Discretisation

For the spatial discretisation, we will be using cell centered equidistant grid points on our solution domain. With a domain length of $L_R$ and $L_Z$, the distance between these $N_R$ and $M_Z$ cells are given by

$$\Delta_R = \frac{L_R}{N_R} \qquad \Delta_Z = \frac{L_Z}{M_Z}$$

With $n = \{0, \ldots, N_R + 1\}$ and $m = \{0, \ldots, M_Z + 1\}$, the grid points in the radial direction $R_n$ are taken at the center of these cells, placing the boundaries at $R_{1/2}$ and $R_{N_Z+1/2}$, and similar for the vertical direction $Z_m$ with the boundaries at $Z_{1/2}$ and $Z_{M_Z+1/2}$. The points at $R_0$, $R_{N_R+1}$, $Z_0$ and $Z_{M_Z+1}$ are so-called ghost points. The discretisation of $R$ and $Z$ takes the form

$$\begin{aligned} Z_m &= Z_{\min} + \left(m - \frac{1}{2}\right)\Delta_Z \\ R_n &= R_{\min} + \left(n - \frac{1}{2}\right)\Delta_R \end{aligned}$$

where $Z_{\min}$ and $R_{\min}$ is the position of the lower vertical and innermost radial boundary.

The discretisation vector $\mathbf{u}$ for a solution $U$ on the radial domain becomes

$$\mathbf{u} = \{u_1, u_2, \cdots, u_n, \cdots, u_{N-1}, u_N\}$$

where the approximated discretised solution is related to the exact solution by $u_n \approx U\,(R = R_n)$. Using the ghost points $u_0$ and $u_{N+1}$ in the radial direction, we can represent the boundary conditions in the following way: Consider the Dirichlet boundary conditions where the values on the boundary are known

$$U_{1/2} = U(R = R_{1/2}) \qquad U_{N+1/2} = U(R = R_{N+1/2})$$

or the Neumann boundary conditions where the gradient (in this case the radial derivative) is known

$$U'_{1/2} = \partial_R U(R = R_{1/2}) \qquad U'_{N+1/2} = \partial_R U(R = R_{N+1/2})$$

We then find the approximate ghost point values for the Dirichlet condition using a centered finite difference on the boundary as

$$u_0 = 2U_{1/2} - u_1 + \mathcal{O}\left(\Delta_R^2\right) \quad (25)$$

$$u_{N+1} = 2U_{N+1/2} - u_N + \mathcal{O}\left(\Delta_R^2\right) \quad (26)$$

and for the Neumann condition, using interpolation and the derivative on the boundary, as

$$u_0 = u_1 - \Delta_R U'_{1/2} + \mathcal{O}\left(\Delta_R^2\right) \quad (27)$$

$$u_{N+1} = u_N + \Delta_R U'_{N+1/2} + \mathcal{O}\left(\Delta_R^2\right) \quad (28)$$

We seek to introduce ghost points in the vertical direction as well, although the assumed periodicity makes the point around the boundaries take another form than for the radial direction. Periodicity ensures that

$$U(Z = Z_m) = U(Z = Z_m + L_Z)$$

so that using the vertical discretisation vector

$$\mathbf{u} = \{u_1, u_2, \ldots, u_m, \ldots, u_{M-1}, u_M\}$$

where the approximated discretised solution is related to the exact solution by $u_m \approx U\left(Z = Z_m\right)$, the ghost points on the exterior of a lower boundary must be equal to the interior points on the upper boundary, and vica verca:

$$u_0 = u_M \qquad u_{M+1} = u_1 \quad (29)$$

Our temporal discretisation will be by a fixed time step $\Delta_t$, where the solution at a given time step $i$ will be noted with a superscript: $u^i = U(t = i\Delta_t)$.

Combined, our discretised approximation $u$ to the exact solution $U$ on a two-dimensional geometry is then given as

$$u^i_{m,n} \approx U(R = R_n, Z = Z_m, t = i\Delta_t)$$

which is the combined spatial and temporal discretisation we will use throughout this chapter. Figure 3 illustrates the discretisation, with the ghost points in gray and the boundary in the halfway between the ghost points and the interior points.

## 3.2 Spectral transformation

The assumption of a periodic boundary condition in the vertical $Z$-direction allows us to use the discrete Fourier transformation in this specific spatial direction. The discrete Fourier transform can be defined, with with $m = 0, \ldots, M-1$ as

$$\widehat{u}_m = \sum_{l=0}^{M-1} u_l \exp\left(\frac{i2\pi lm}{M_Z}\right)$$

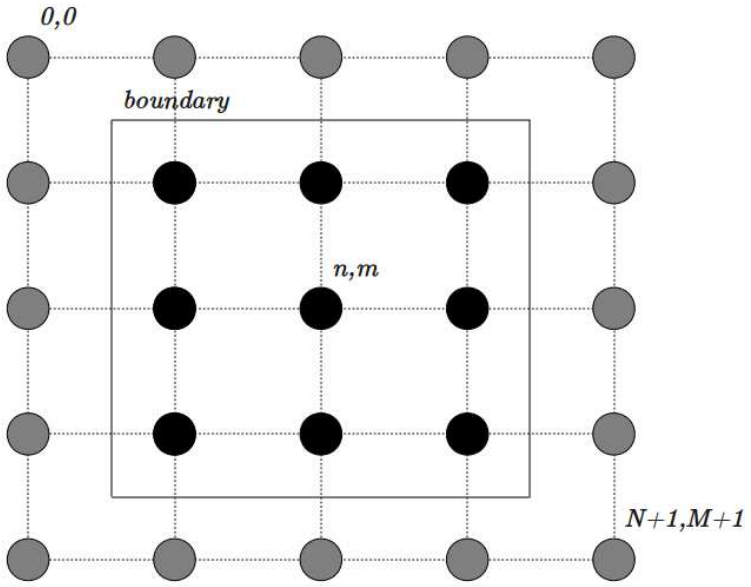


Figure 3: Illustration of the discretisation used. Ghost points are marked in grey.

and the inverse transform given as

$$u_m = \frac{1}{M} \sum_{l=0}^{M-1} \widehat{u}_l \exp\left(-\frac{i2\pi lm}{M_Z}\right)$$

where $u_m$ is the discretisation of the solution in the vertical direction as described in section 3.1. This transform and inverse transform are implemented by using the Fast Fourier Transform (FFT) algorithm.

It is well known that for the Fourier transform, the differential in regular space becomes a multiplication in Fourier space and is exact:

$$\frac{\partial}{\partial Z} \rightarrow ik_Z$$

where $k_Z = 2\pi nm/M_Z$, so that the Laplace operator in cylindrical coordinates for a spectral transformation in the $Z$-direction for each spectral component $k_Z$ becomes

$$\nabla_\perp^2 \rightarrow \frac{\partial^2}{\partial R^2} + \frac{1}{R}\frac{\partial}{\partial R} - k_Z^2 \tag{30}$$

Using periodic boundary conditions in the vertical direction then lets us use the Fourier transform and exactly represent the derivatives in the $Z$-direction for accuracy and further allows us to solve the model equations for each Fourier mode separately, simplifying the solving of the elliptic and diffusion equations.

## 3.3 Solving the elliptic equation

With the connection between the potential and the vorticity expressed as the elliptic equation $\Omega = \nabla^2\phi$, a method for solving for the electrostatic potential $\phi$ is required. Using the discretisation specified in section 3.1, we write the discretisations for $\Omega$ and $\phi$ as

$$\mathbf{\Omega} = \{\omega_1, \ldots, \omega_n, \ldots, \omega_N\} \qquad \mathbf{\Phi} = \{\phi_1, \ldots, \phi_n, \ldots, \phi_N\}$$

with the corresponding Fourier transformed variables

$$\widehat{\mathbf{\Omega}} = \{\widehat{\omega}_1, \ldots, \widehat{\omega}_n, \ldots, \widehat{\omega}_N\} \qquad \widehat{\mathbf{\Phi}} = \left\{\widehat{\phi}_1, \ldots, \widehat{\phi}_n, \ldots, \widehat{\phi}_N\right\}$$

Using the Laplace operator in Eq. (30) and using a centered finite difference scheme gives us for each spectral component $k_Z$

$$\frac{\widehat{\phi}_{n-1} - 2\widehat{\phi}_n + \widehat{\phi}_{n+1}}{\Delta_R^2} + \frac{\widehat{\phi}_{n+1} - \widehat{\phi}_{n-1}}{2R_n\Delta_R} - k_Z^2\widehat{\phi}_n = \widehat{\omega}_n$$

or

$$(1-\lambda_n)\,\widehat{\phi}_{n-1} - \left(2 + \Delta_R^2 k_Z^2\right)\widehat{\phi}_n + (1+\lambda_n)\,\widehat{\phi}_{n+1} = \Delta_R^2\widehat{\omega}_n$$

which rewritten on a tri-diagonal matrix form becomes

$$\begin{pmatrix} \ddots & \ddots & \ddots & & \\ & (1-\lambda_n) & -\left(2+\Delta_R^2k_Z^2\right) & (1+\lambda_n) & \\ & & \ddots & \ddots & \ddots \end{pmatrix} \widehat{\mathbf{\Phi}} = \Delta_R^2\widehat{\mathbf{\Omega}} \qquad (31)$$

using $\lambda_n = \Delta_R/(2R_n)$. Note that $\lambda_n$ is here dependent on $R_n$ and thus vary for each row $n$ of the matrix.

Applying boundary conditions alters the first and last row in the matrix and the first and last vector elements on the right hand side. For the Dirichlet boundary condition, the potential $\phi$ assumes the values $\phi_{1/2}$ and $\phi_{N+1/2}$ on the boundaries. Using Eq. (25) and Eq. (26) we then get

$$\widehat{\phi}_0 = 2\widehat{\phi}_{1/2} - \widehat{\phi}_1 \qquad \widehat{\phi}_{N+1} = 2\widehat{\phi}_{N+1/2} - \widehat{\phi}_N$$

which introduced for $n = 1$ gives

$$-\left(3 - \lambda_n + \Delta_R^2k_Z^2\right)\widehat{\phi}_1 + (1+\lambda_n)\,\widehat{\phi}_2 = \Delta_R^2\widehat{\omega}_1 - 2\,(1-\lambda_n)\,\widehat{\phi}_{1/2}$$

and for $n = N$

$$(1-\lambda_n)\,\widehat{\phi}_{N-1} - \left(3 + \lambda_n + \Delta_R^2k_Z^2\right)\widehat{\phi}_N = \Delta_R^2\widehat{\omega}_n - 2\,(1+\lambda_n)\,\widehat{\phi}_{N+1/2}$$

For the Neumann boundary condition, the radial differential of the potential $\partial_R\phi$ assumes the values $\phi'_{1/2}$ and $\phi'_{N+1/2}$ on the boundaries. Using Eq. (27) and Eq. (28) we then get

$$\widehat{\phi}_0 = \widehat{\phi}_1 - \Delta_R\widehat{\phi}'_{1/2} \qquad \widehat{\phi}_{N+1} = \widehat{\phi}_{N+1/2} + \Delta_R\widehat{\phi}'_{N+1/2}$$

| Coefficient | $K=1$ | $K=2$ | $K=3$ |
|---|---|---|---|
| $\alpha_0$ | 1 | 3/2 | 11/6 |
| $\alpha_1$ | 1 | 2 | 3 |
| $\alpha_2$ | 0 | $-1/2$ | $-3/2$ |
| $\alpha_3$ | 0 | 0 | 1/3 |
| $\beta_1$ | 1 | 2 | 3 |
| $\beta_2$ | 0 | $-1$ | $-3$ |
| $\beta_3$ | 0 | 0 | 1 |

Table 1: The coefficients $\alpha_k$ and $\beta_k$ for the stiffly stable time integrator scheme.

which introduced for $n=1$ gives

$$\left(1+\lambda_n+\Delta_R^2 k_Z^2\right)\widehat{\phi}_1+\left(1+\lambda_n\right)\widehat{\phi}_2=\Delta_R^2\widehat{\omega}_1+\Delta_R\left(1-\lambda_n\right)\widehat{\phi}'_{1/2}$$

and for $n=N$

$$\left(1-\lambda_n\right)\widehat{\phi}_{N-1}-\left(1-\lambda_n+\Delta_R^2 k_Z^2\right)\widehat{\phi}_N=\Delta_R^2\widehat{\omega}_n-\Delta_R\left(1+\lambda_n\right)\widehat{\phi}'_{N+1/2}$$

These dependencies on the boundary conditions will alter the matrix elements $(1,1)$ and $(N,N)$ in Eq. (31) as well as include the boundary condition values on the right hand side vector $\widehat{\mathbf{\Omega}}$.

## 3.4 Stiffly stable time integrator for the diffusion equations

With the evolution equation for the particle density and vorticity taking the form of the inhomogeneous diffusion equation

$$\partial_t U=\mu\nabla_\perp^2 U+\mathcal{L}U$$

where $U$ represents either of the dependent variables and $\mathcal{L}$ is a differential operator on $U$, we can use the $K$-th order stiffly stable implicit/explicit scheme on the discretisation $u$ of $U$, as described in [10, 12]:

$$\frac{1}{\Delta_t}\left(\alpha_0 u^i-\sum_{k=1}^{K}\alpha_k u^{i-k}\right)=\kappa\nabla_\perp^2 u^i+\sum_{k=1}^{K}\beta_k\mathcal{L}u^{i-k}+\mathcal{O}\left(\Delta_t^K\right)$$

where the superscript on $u$ denotes the time step. We will for simplicity of notation assume that $\mathcal{L}$ is a linear operator. The coefficients $\alpha_k$ and $\beta_k$ are given in table 1. We can rearrange this to the implicit part on the left hand side and the explicit part on the right hand side:

$$\left(\alpha_0-\kappa\Delta_t\nabla_\perp^2\right)u^i=\sum_{k=1}^{K}\left(\alpha_k+\Delta_t\beta_k\mathcal{L}\right)u^{i-k}+\mathcal{O}\left(\Delta_t^K\right)\tag{32}$$

Using Eq. (30) for the discretisation of $\nabla_\perp^2 U^i$ and approximating the differentials by centered finite differences gives us for each spectral component $k_Z$:

$$\nabla_\perp^2 \widehat{u}_n = \frac{\widehat{u}_{n-1} - 2\widehat{u}_n + \widehat{u}_{n+1}}{\Delta_R^2} + \frac{\widehat{u}_{n+1} - \widehat{u}_{n-1}}{2R_n\Delta_R} - k_Z^2 \widehat{u}_n + \mathcal{O}\left(\Delta_R^2\right)$$

With this, Eq. (32) then reads

$$-r\left(1-\lambda_n\right)\widehat{u}_{n-1}^i + \left(\alpha_0 + 2r + k_Z^2\right)\widehat{u}_n^i - r\left(1+\lambda_n\right)\widehat{u}_{n+1}^i = \sum_{k=1}^{K}\left(\alpha_k \widehat{u}_n^{i-k} + \beta_k \mathcal{L}\widehat{u}_n^{i-k}\right) \tag{33}$$

using $r = \kappa\Delta_t/\Delta_R^2$ and $\lambda_n = \Delta_R/\left(2R_n\right)$. Writing this on a tri-diagonal matrix form gives us

$$\begin{pmatrix} \ddots & \ddots & \ddots & & \\ & -r\left(1-\lambda_n\right) & \alpha_0 + 2r + k_Z^2 & -r\left(1+\lambda_n\right) & \\ & & \ddots & \ddots & \ddots \end{pmatrix} \widehat{\mathbf{u}}^i = \sum_{k=1}^{K}\left(\alpha_k + \beta_k\mathcal{L}\right)\widehat{\mathbf{u}}^{i-k} \tag{34}$$

Note that $\gamma$ is dependent on $R$, and thus changes for each row in the matrix, similar to the elliptic equation (31). Applying boundary conditions alters the first and last row in the matrix and the first and last vector elements on the right hand side, as described in the following.

For Dirichlet boundary conditions, we assume the function values to stay fixed on the boundaries. Using the ghost point values on $u_0$ and $u_{N+1}$ from Eq. (25) and Eq. (26), we can write Eq. (33) for $n = 1$ as

$$\left[\alpha_0 + \left(3 - \lambda_n\right)r + k_Z^2\right]\widehat{u}_1^i - r\left(1+\lambda_n\right)\widehat{u}_2^i = \mathcal{K} + 2r(1-\lambda_n)U_{1/2} \tag{35}$$

and $n = N$ as

$$\left[\alpha_0 + \left(3 + \lambda_n\right)r + k_Z^2\right]\widehat{u}_N^i - r\left(1-\lambda_n\right)\widehat{u}_{N-1}^i = \mathcal{K} + 2r(1+\lambda_n)U_{N+1/2} \tag{36}$$

where we have used the short hand notion

$$\mathcal{K} = \sum_{k=1}^{K}\left(\alpha_k \widehat{u}_n^{i-k} + \beta_k \mathcal{L}\widehat{u}_n^{i-k}\right)$$

with $n = 1$ for Eq. (35) and $n = N$ for Eq. (36).

For Neumann boundary conditions, we assume the gradients on the boundaries to stay fixed. Using the ghost point values on $u_0$ and $u_{N+1}$ from Eq. (27) and Eq. (28), we can write Eq. (33) for $n = 1$ as

$$\left[\alpha_0 + \left(1 + \lambda_n\right)r + k_Z^2\right]\widehat{u}_1^i - r\left(1+\lambda_n\right)\widehat{u}_2^i = \mathcal{K} - r\left(1-\lambda_n\right)\Delta_R U'_{1/2}$$

and $n = N$ as

$$\left[\alpha_0 + \left(1 - \lambda_n\right)r + k_Z^2\right]\widehat{u}_N^i - r\left(1-\lambda_n\right)\widehat{u}_{N-1}^i = \mathcal{K} + r(1+\lambda_n)\Delta_R U'_{N+1/2}$$

where $\mathcal{K}$ is given as above.

As with the elliptic equation, the boundary conditions here give rise to a change of the elements $(1,1)$ and $(N,N)$ in the matrix in Eq. (34) as well as including the boundary values on the right hand side.

## 3.5 The rectangular coordinate limit

When solving the elliptic equation in section 3.3 and the time splitting scheme in section 3.4 we note that taking the limit $R_0 \to \infty$ gives us the rectangular coordinate system solution. The radially dependent term $\lambda_n$ used in our matrix equations becomes

$$\lim_{R_0 \to \infty} \lambda_n = 0$$

Using this, the matrix equations become simple tridiagonal matrices, easily stored and solved as the terms do not vary for each row (and column).

## 3.6 Initial conditions

We will assume the initial conditions to be stationary solutions of the particle density, vorticity and the potential, satisfied by the Dirichlet or Neumann boundary conditions at the inner and outer radial edge, along with an added perturbation on the particle density $\widetilde{n}$:

$$\begin{aligned} n(t=0) &= n_0 + \widetilde{n} \\ \Omega(t=0) &= \Omega_0 \\ \phi(t=0) &= \phi_0 \end{aligned}$$

For our simulations, this perturbation will take the form of a gaussian of width $l$ and amplitude $\Delta n$ around the position $\mathbf{R}_0 = (R_0, Z_0)$:

$$\widetilde{n}(\mathbf{r}) = \Delta n \, \exp\left[-\frac{(\mathbf{R}-\mathbf{R}_0)^2}{2l^2}\right]$$

The initial vorticity field and potential field are considered set to zero: $\Omega_0 = 0$ and $\phi_0 = 0$.

## 3.7 The algorithm

Using the time splitting scheme presented in 3.4, we can now present the algorithm used to find the time evolution of the particle density and vorticity field for each time step:

1. For time step $i$, the particle density $n^i$, the vorticity $\Omega^i$ and the electrostatic potential $\phi^i$ are given, including boundary conditions, from either initial conditions or the previous time step.

2. Ghost point values are constructed as described in section 3.1.
3. As described in section 3.4, the explicit parts of the evolution equation for $n^{i+1}$ and $\Omega^{i+1}$, such as the Poisson bracket and electric drift compression, are calculated using $n^i$, $\Omega^i$ and $\phi^i$ and the solutions from the $K$ previous time steps.
4. The variables $n^i$, $\Omega^i$ and $\phi^i$ are transformed to Fourier space, $\widehat{n}^i$, $\widehat{\Omega}^i$ and $\widehat{\phi}^i$, using FFT as described in section 3.2.
5. The time splitting scheme is then used to solve for $\widehat{n}^{i+1}$ and $\widehat{\Omega}^{i+1}$ for each spectral mode $k_Z$ by solving the matrix equation (34).
6. $\widehat{\phi}^{i+1}$ is found from $\widehat{\Omega}^{i+1}$ by solving the matrix equation (31) for each spectral mode $k_Z$.
7. The inverse Fourier transform is then performed on these fields to yield the fields for time step $i+1$: $n^{i+1}$, $\Omega^{i+1}$ and $\phi^{i+1}$.

For the initial time steps, we have a reduced number of previous time steps in order to construct the explicit part. For $K = 1$, the time splitting scheme takes the form of the simple forward difference scheme, followed by the second order stiffly stable scheme for $K = 2$ and then a third order stiffly stable scheme throughout the simulation with $K = 3$.

It is worth noting that for this first time step from an initial condition of $\Omega_0 = 0$, the vorticity equation takes the simple form, using forward difference for the time differential of $\Omega$:

$$\Omega^1 = -\Delta_t \partial_Z n^0 = \Delta_t \Delta n (Z - Z_0) \exp\left[-\frac{(\mathbf{R} - \mathbf{R}_0)^2}{2}\right]$$

(with time step in superscript) resulting in a dipole vorticity field after the first time step if the initial perturbation $\widetilde{n}$ is a "blob"-like structure.

## 3.8 Arakawa scheme for the Poisson bracket

The non-linear advective term used for the advection with the electric drift for the particle density or vorticity $U$, here noted as the Poisson bracket

$$\{\phi, U\} = \widehat{\mathbf{Z}} \cdot \nabla\phi \times \nabla U$$

or on the very handy form

$$\{\phi, U\} = \frac{\partial}{\partial Z}\left(U\frac{\partial \phi}{\partial R}\right) - \frac{\partial}{\partial R}\left(U\frac{\partial \phi}{\partial Z}\right)$$

has to be treated with care in the numerical picture.

| Quantity | Expression |
|---|---|
| Center of mass position | $\frac{\int d\mathbf{R}\, Rn}{\int d\mathbf{R}\, n}$ |
| Center of mass velocity | $\frac{d}{dt}\frac{\int d\mathbf{R}\, Rn}{\int d\mathbf{R}\, n}$ |
| Integrated particle density | $\int d\mathbf{R}\, n$ |

Table 2: A selection of the diagnostics output from the simulation code.

The traditional scheme with centered finite differences does not account for conservation of quantities such as mean vorticity, mean kinetic energy and mean square vorticity, and give rise to nonlinear computational instabilities unless the scheme is restricted to a specific form [5].

The "Arakawa scheme" proposed by Akio Arakawa [5] does ensure the conservation of these quantities: The mean kinetic energy $\int d\mathbf{R}\ (\nabla\phi)^2/2$, the mean vorticity $\int d\mathbf{R}\Omega$ and the mean square vorticity $\int d\mathbf{R}\Omega^2$. Consequently, expanding the potential $\phi$ into orthogonal harmonic functions, the mean wave number $\overline{k^2}$ is conserved since the mean kinetic energy $E$ and mean square vorticity $W$ can be expressed as

$$E = \frac{1}{2}\sum_n k_n^2 \widehat{E}_n \qquad W = \frac{1}{2}\sum_n k_n^4 \widehat{W}_n$$

giving us

$$\overline{k^2} = \frac{\sum_n k_n^2 \widehat{W}_n}{\sum_n \widehat{E}_n}$$

which is a conserved quantity if $E$ and $W$ are conserved, thus severely restricting the length scales the energy distributes to.

In addition, the Arakawa scheme does exhibit good properties regarding accuracy compared to other methods for solving two-dimensional advection problems, as demonstrated in [14].

The actual scheme has here been omitted for brevity but can be found in [5, 14]. Figure 4 shows a stencil diagram over the scheme for the point $(n, m)$. Note that the usage of ghost points greatly aid the computation of this term near the border.

## 3.9 Simulation diagnostics

For each time step the simulation program outputs various diagnostic quantities, used to both quantify the behavior of the physical system as well as troubleshooting and finding any numerical issues. While the list of quantities output by the program is extensive, we will here list some of the most important quantities from this thesis' perspective in table 2.

The integrated quantities are here taken as sums over the entire simulation domain, accounting for the cylindrical coordinate system.

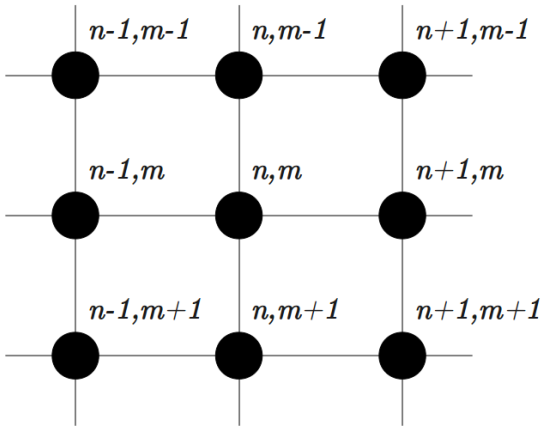


Figure 4: Stencil diagram over the Arakawa scheme for the point $(n, m)$.

## 3.10 Validation of the diffusion equation

Taking the equation for the particle density (22) and disregarding the advection with and compression of the electric drift, we end up with the diffusion equation

$$\frac{\partial n}{\partial t} = \mu \nabla^2 n$$

We can now look for a steady state solution $\partial_t n = 0$ of this equation in cylindrical coordinates $(R, Z, \Theta)$. Setting $\mu = 1$ and assuming the solution to be azimuthally symmetric as well as invariant along the $Z$-axis, the solution $n(R)$ will depend on $R$ only. We also assume the inner and outer boundaries, at $R_{1/2}$ and $R_{n+1/2}$ respectively, to be defined by

$$n(R_{1/2}) = n_{1/2} \qquad n(R_{n+1/2}) = n_{n+1/2}$$

Using the Laplace operator for cylindrical coordinates, we then get

$$0 = \frac{1}{R}\frac{\partial}{\partial R}\left(R\frac{\partial n}{\partial R}\right) = \frac{\partial^2 n}{\partial R^2} + \frac{1}{R}\frac{\partial n}{\partial R}$$

This has the solution

$$n = C_{n+1/2} \log R + C_{1/2}$$

where the constants $C_0$ and $C_1$ are defined by the boundary conditions:

$$\begin{aligned} C_0 &= \frac{n_{1/2} - n_{n+1/2}}{\log R_{1/2} - \log R_{n+1/2}} \\ C_{n+1/2} &= n_{n+1/2} - C_{1/2} \log R_{n+1/2} \end{aligned}$$

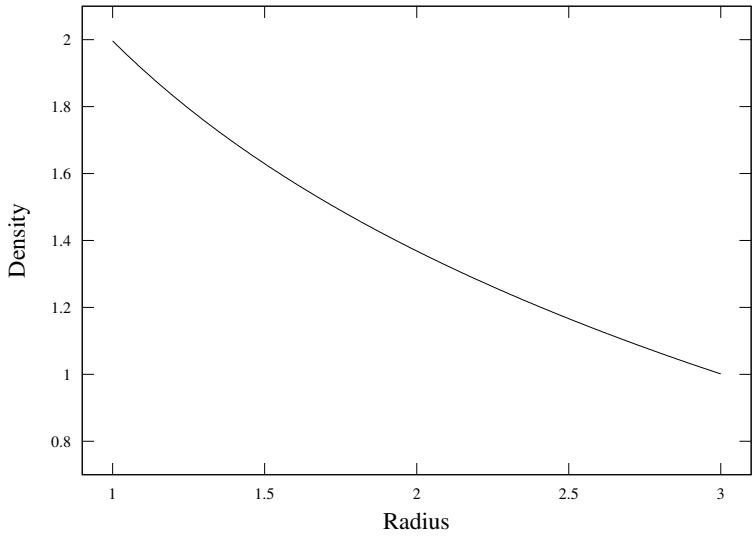


Figure 5: The steady state solution of particle density in a cylindrical coordinate system, matching the expected steady state density profile.

Simulating the diffusion equation with $R_{1/2} = 1$, $R_{n+1/2} = 3$, $n_{1/2} = 2$ and $n_{n+1/2} = 1$ after the system has reached a steady state, as shown in figure 5, we get an exact match with this analytical result validating the diffusion process and elliptic solver in our simulation code for cylindrical coordinates.

## 3.11 Including the convective term

There are, for the convective term described in section 2.4, two concerns. First, perhaps the easiest solved, is the added computational cost. Second, it is the accuracy of including this term and its added time splitting of the numerical scheme. Let us explore this further by attempting to add a similar term to the diffusion equation and find a scheme for it.

The diffusion equation used in our simulations, with the added term, has the form

$$(\nabla \ln n + \nabla) \cdot \frac{\partial \nabla_\perp \phi}{\partial t} = \mu \nabla_\perp^2 \Omega$$

where $\Omega$ is the vorticity, $\phi$ is the potential, $n$ is the particle density and $\mu$ is a constant. In 2DADS this is simplified by neglecting the first term and using $\nabla_\perp^2 \phi = \Omega$ to get

$$\frac{\partial \Omega}{\partial t} = \mu \nabla_\perp^2 \Omega$$

which is easily solved. However, as we shall see, by not neglecting the term we will get additional dependencies on the particle density and the potential which are not as easy to solve directly. We see that we get

$$\nabla \ln n \cdot \partial_t \nabla_\perp \phi + \partial_t \Omega = \mu \nabla_\perp^2 \Omega$$

and can no longer use the relation between the potential and the vorticity to eliminate the potential as a dependence in the evolution term. In addition, the particle density and its spatial differentials are included. This suggest that there is no good implicit way of including this term, and we will have to include it explicitly by using the calculated values of the particle density and its spatial differentials and the potential for previous time steps. The question of the solution's stability and accuracy then arises. Moreover, this term give a cubic nonlinearity associated with the electric drift advection.

Simulations including this term explicitly are described in 5.7.

# 4 Modifications to 2DADS

This chapter will describe the major bulk of modifications performed on the original 2DADS code developed and provided by Odd Erik Garcia.

The original 2DADS Fortran code was originally written in FORTRAN 77. Although fully functional, this old dialect of the Fortran language puts some severe restrictions on the form of the syntax. With the bulk of the programming languages available being more relaxed in this respect, it feels natural to move this code to a more modern base. Due to this, the code was modified into obeying the Fortran 95 standard. As all commonly used Fortran compilers support both versions, this poses no threat to the cross platform nature of the original program.

In addition, Fortran modules has been introduced and used to encapsulate certain parts of the code in a step towards a more extensible format and obeying more modern programming paradigms.

All of the modifications carried through is done with regard to further expansion of the programming code to a bi-directional finite difference method and other coordinate systems.

## 4.1 Introduction of cylindrical coordinates

The main modification of the 2DADS program is the extension of model and parameters to support the cylindrical coordinate system $(R, Z)$, assuming invariance in the azimuthal direction. After these modifications, 2DADS is capable of performing simulations in both cylindrical and rectangular coordinates.

### 4.1.1 Extending the matrix solver system to account for cylindrical coordinates

2DADS uses the assumption that the matrix formulation of the time evolution for the particle density and vorticity, as well as the elliptic Poisson equation relating the electrostatic potential to the vorticity, being tridiagonal with the elements being constant for all rows apart from the first or last row. These matrices can easily be stored in a $3 \times 3$ matrix containing the lower, upper and center diagonal elements:

$$\begin{pmatrix} D_1 & U & & & & \\ L & D & U & & & \\ & \ddots & \ddots & \ddots & & \\ & & L & D & U & \\ & & & L & D_2 & \end{pmatrix} \rightarrow \begin{pmatrix} & D_1 & U \\ L & D & U \\ L & D_2 & \end{pmatrix}$$

Solving for such a matrix can be performed by a simple LU-factorisation and back-substitution, as described in [17].

In the case of a cylindrical (or other) coordinate system, these matrices change structure, as shown in Eq. (31) for the elliptic equation and Eq. (34) for the time evolution of the particle density and vorticity. These matrices have elements that vary with the radius $R$, and thus also with the row number in the matrix. A simple $3 \times 3$ storage matrix is not possible for this case.

Although these matrices do keep their tri-diagonal form, it is desirable to device a system that does not assume this matrix form, keeping further extensions in mind.

The simplest flexible solution fell on using the LAPACK linear algebra package for factorising these matrices and then solving them through back-substitution. This allows the system to solve for any arbitrarily sized matrix with any arbitrary structure, not only tri-diagonal.

With this in place, the program is able to solve for the matrices defined in Eq. (31) and Eq. (34).

## 4.2 Precalculating and storing of the LU-factorisation

Motivated by the execution time and memory requirement posed by arbitatily sized simulation domains and geometry, we look towards making the matrix solver system more efficient.

This is primarily achieved by making a system for prefactorisation of the matrices. For each time step, $3 \times M_Z$ LU-factorisations of $N_R \times N_R$ sized matrices are being performed, one elliptic equation and two evolution equations. ($M_Z$ is here the resolution in the vertical direction and thus the number of vertical spectral components, $N_R$ is the resolution in the radial direction as described in 3.1.) However, as these matrices are constant and not time-varying, these matrices can be pre-factorised and stored for retrieval. This requires the program to store the LU-factorisation matrices throughout the simulation. The LU factorisation algorithm provided by LAPACK stores both the lower matrix $L$ and the upper matrix $U$ in the same $N_R \times N_R$ matrix. These are then retrieved for each time step, allowing us to solve the systems by simple back-substitution.

As the original matrices are tri-diagonal, the resulting LU-factorisation matrix is also tri-diagonal. The majority of the matrix entries are zero. To account for large matrices and a potential upper memory limit, a system for storing these sparse matrices is deviced.

Each of the matrices are serialised as a two-dimensional array containing the position and the value of each element in the matrix. We here provide an

example tri-diagonal matrix and it's accompanying serialisation:

$$\begin{pmatrix} a & b & 0 & 0 & 0 \\ c & d & e & 0 & 0 \\ 0 & f & g & h & 0 \\ 0 & 0 & i & j & k \\ 0 & 0 & 0 & l & m \end{pmatrix} \rightarrow \begin{pmatrix} a & b & c & d & e & f & g & \cdots \\ 1 & 2 & 6 & 7 & 8 & 12 & 13 & \cdots \end{pmatrix}$$

For a $1024 \times 1024$ solution grid, with a double precision data format (8 bytes), this reduces the storage need of the prefactorised LU matrices from 26GB to less than 10MB.

## 4.3 Data output as HDF5

2DADS outputs a large set of data, including diagnostics such as energy and density integrals, power spectral densities and the variable fields (particle density, vorticity and potential fields). Motivated to make this set of data smaller and easier to handle, the major part of output has been changed to the HDF5 format.

HDF5 (Hierarchical Data Format 5) is an open format created and maintained by the HDF Group [1] aimed at producing a compact and flexible data format for complex data structures in science and engineering.

The existing output format of 2DADS is a text-based format. Introducing HDF5 as 2DADS main output format has the benefits of reducing the output processing overhead as well as the output data file sizes. The HDF5 API also ensures that the data output will be compressed, further reducing the storage requirements. The largest gain is experienced for the output of the particle density, vorticity and potential fields, which for resolutions of $1024 \times 1024$ reduces the storage requirements by a factor of 20.

Since the format has such widespread support in program packages such as MATLAB, Mathematica and Octave (to mention a few), an additional motivation factor is the increased ease of import and manipulation of data using these packages.

### 4.3.1 The hierarchical organisation of the data

Many output files for each time step has been merged and contained in one individual file using an internal hierarchy. As an example, the three fields output at regular intervals as well as their derivatives is now output in one single file `fields.nnn.h5`, where *nnn* indicates the output time step, instead of individual files for each field and its derivatives.

We here illustrate the structure of this file:

```
fields.nnn.h5
         TIME     (1)
         OMEGA    (Nx+2, My+2)
         OMEGAX   (Nx+2, My+2)
         OMEGAY   (Nx+2, My+2)
         STRMF    (Nx+2, My+2)
         STRMFX   (Nx+2, My+2)
         STRMFY   (Nx+2, My+2)
         THETA    (Nx+2, My+2)
         THETAX   (Nx+2, My+2)
         THETAY   (Nx+2, My+2)
```

`TIME` is here the output time for this field, while `OMEGA`, `OMEGAX` and `OMEGAY` contains the vorticity field $\Omega$ and its differentials $\partial_x\Omega$ and $\partial_y\Omega$ respectively. `STRMF` and `THETA` contains the potential $\phi$ and particle density $n$. Their differentials are contained in a similar manner as for the vorticity.

## 4.4 Parallel computing using OpenMP

Motivated by reducing the computing time spent on simulations using 2DADS, it has been desirable to implement a parallel computing scheme as the simulations are to be executed on the Stallo cluster. The cluster consists of 704 servers, each using two quad-core processors[4]. The two parallel schemes possible were MPI and OpenMP.

MPI (Message Passing Interface) is an API designed for parallel or distributed computing on multiple servers[2]. OpenMP on the other hand is a compiler-level implementation aimed at utilising multiple processors or cores on a single server[3].

The choice fell on OpenMP as the parallel computing scheme used for 2DADS, mainly because of its ease of implementation compared to MPI. The drawback is that the OpenMP implementation can only utilise up to the eight cores of a single Stallo server for each simulation. Although this puts an upper limit on the performance gain through the use of OpenMP, it still provides a significant reduction of execution time as many of the individual tasks in 2DADS are easily parallelised.

The libraries available on Stallo, such as LAPACK, BLAS and HDF5, are compiled with support for parallel computing.

### 4.4.1 Implementing OpenMP

Most commonly used Fortran 90 compilers such as GNU Fortran and Intel Fortran do support the OpenMP implementation. In addition, OpenMP works by utilising compiler directives in the source code masked as comments for compilers without OpenMP support. This maintains the portability between platforms.

The compiler directives are used around blocks of parallelisable code, such as loops or array operations and assignments. The compiler then uses this to generate the parallelised code, using the maximum number of available cores or processors available. Fortran in particular is well suited for such automated parallelism, as the language has support for array and matrix operations which often easily can be automatically parallelised.

OpenMP uses a shared memory model where the executed threads can access to the same memory spaces. Although the parallelisation itself is automated by the compiler, care has to be taken when specifying which variables that are shared or to be kept isolated for each thread. To ensure that no variables remain unspecified in behavior between threads which easily can (and did) give elusive program bugs, a compiler directive to force the behavior of each variable to be specified has been used.

An example of the implementation can be seen in the file `stiff.f90`:

```
!$OMP PARALLEL DEFAULT(NONE) SHARED(R, Nx, My, A, F, eqStiffAlpha, &
!$OMP   eqStiffBeta, deltat, itl, bndvl) PRIVATE(m, n)

!$OMP WORKSHARE
R = 0D0
!$OMP END WORKSHARE
!$OMP DO SCHEDULE(DYNAMIC)
do n=1,Nx
        do m=1,My
                do k=1,itl-1
                        R(m,n) = R(m,n) + eqStiffAlpha(k,itl-1) &
* A(m,n,k) + deltat * eqStiffBeta(k,itl-1) * F(m,n,k)
                enddo
        enddo
enddo
!$OMP END DO

! fill in ghost points since spectral transform start at zero index
!$OMP WORKSHARE
R(0, 1:Nx) = R(My, 1:Nx)
R(My+1, 1:Nx) = R(My+1, 1:Nx)

! add the contribution of boundary conditions to the explicit part
R(0:My+1, 1) = R(0:My+1, 1) + bndvl(1)
R(0:My+1, Nx) = R(0:My+1, Nx) + bndvl(2)
!$OMP END WORKSHARE

! make spectral transformation of the array
!$OMP DO SCHEDULE(DYNAMIC)
do n=1,Nx
        call rlfft(R(0:My+1, n),My,+1)
enddo
!$OMP END DO
!$OMP END PARALLEL
```

While this part of the program does not give any large increase of performance due to the small number of operations that are parallelised, it still provides an

illustration on the usage of the various compiled directives and how they tie in with the program code.

We see here how the entire parallel blocks is encapsulated by starting and ending directives in the form `!$OMP`. The variables are then specified either as shared between threads or private for each thread, using the `SHARED()` and `PRIVATE()` directive. The contained blocks are then specified as either do-loops with the `DO SCHEDULE(DYNAMIC)` directive, allowing the compiler to allocate the do-iterations to the available processors on a first-come-first-serve basis, or as `WORKSHARE` blocks, which automatically split the vector or matrix operations between the available processors.

## 4.5 Notes on computational complexity

The computational complexity of the algorithms directly affect the execution time for simulations. Due to this we will here give a brief overview of the core algorithms and the governing complexities in big-O notation as described in literature such as [17].

The spectral methods (using FFT) used in 2DADS is, although a core mechanism in the computation, computationally inexpensive with a cost of $\mathcal{O}(n \ln n)$. This low cost, combined with providing a high accuracy for derivatives, makes spectral methods desirable for this kind of simulations. This does however require periodic boundary conditions to avoid ringing effects such as Gibbs phenomena on the boundaries. In 2DADS this periodic direction is the vertical $Z$-direction.

The main bulk of the computation time is spent on the rather expensive solving of $n$ matrices of dimension $n \times n$ for the finite difference scheme of the inhomogeneous diffusion equation. Although we do employ a strategy of prefactorisation the matrices into LU (Lower/Upper) matrices and solving through back-substitution, this backsubstitution has a computational complexity of $\mathcal{O}(n^2)$ for each matrix, resulting in $\mathcal{O}(n^3)$ for each time step. The advantage is that we can employ boundary conditions in the direction finite difference is used. In 2DADS we employ this scheme for the radial direction.

## 4.6 Notes on 2DADS

The modified 2DADS code used in this thesis is available for download at

`http://www.student.uit.no/~per024/2dads.per024.tgz`

The package contains the source code and makefiles for compiling and running the simulation code on Stallo, but can easily be modified for any Linux distribution or other platform as the Fortran 90 code used is a well-known and well-supported standard.

When compiling the simulation code, the pre-compiled libraries for HDF5 support, LAPACK and BLAS must be available for linking, either dynamically during runtime or during compile-time. This is controlled in the makefile.

The parameters are controlled by the file `input.ini` and the output is put in a subdirectory called `run/` under the folder the program is executed from.

Although the code is commented, a few notes on the syntax might be brought to attention. First, the code is written for both rectangular and cylindrical coordinates. Due to this the radial direction is denoted `X` and the vertical direction is denoted as `Y`. Second, the particle density (or more generally the particle density) n is in the code represented as `THETA`. The vorticity is represented as `OMEGA`, and the potential (or stream function) is `STRMF`. Every differential is noted as a suffix on the independent variable, for example `THETAX` and `THETAXX` for $\partial_R n$ and $\partial_{RR} n$.

The core routines of the program is located in the files `2dads.f90` and `stiff.f90`, containing the main time step algorithm and the solving of the stiffly stable scheme respectably. The right hand side contributions for the diffusion equations are calculated in `theta.f90` and `omega.f90` for $n$ and $\Omega$, and the Poisson bracket is represented using the Arakawa scheme in `arakawa.f90`.

# 5 Simulations

We will in this chapter re-create previous simulations of plasma blob dynamics to validate the code, as well as explore variations of parameters such as blob amplitude and ion-neutral collision frequency.

## 5.1 Common parameters

All simulations presented in this chapter use the same set of parameters given in this section. Any deviations from these are stated explicitly.

We initially use the model equations with linearised interchange term from 2.2.5

$$\begin{aligned} \frac{d\Omega}{dt} + \frac{\partial \widetilde{n}}{\partial Z} &= \mu_\Omega \nabla_\perp^2 \Omega - \nu_{\mathrm{i}} \Omega \\ \frac{d\widetilde{n}}{dt} &= \mu_{\mathrm{n}} \nabla_\perp^2 n \end{aligned}$$

in the rectangular coordinate limit of $\mathbf{R} = (R, Z) \approx \mathbf{x} = (x, y)$. The boundary conditions in the radial direction for the particle density $\widetilde{n}$, vorticity $\Omega$ and the electrostatic potential $\phi$ are all set to the Dirichlet conditions

$$\begin{aligned} \widetilde{n}(x = x_{\min}) = \widetilde{n}(x = x_{\max}) &= 0 \\ \Omega(x = x_{\min}) = \Omega(x = x_{\max}) &= 0 \\ \phi(x = x_{\min}) = \phi(x = x_{\max}) &= 0 \end{aligned}$$

The initial conditions are then specified as

$$\begin{aligned} \widetilde{n}(t = 0) &= \exp\left[ -(\mathbf{x} - \mathbf{x}_0)^2 \right] \\ \Omega(t = 0) &= 0 \\ \phi(t = 0) &= 0 \end{aligned}$$

where $\widetilde{n}$ take the blob-like shape around an initial position $\mathbf{x}_0$ with a perturbation amplitude $\Delta n = 1$. We use the diffusion coefficient $\mu_n = 10^{-2}$ and viscousity coefficients $\mu_\Omega = 10^{-2}$ (which gives us the default Rayleigh number $\mathrm{Ra} = (\mu_{\mathrm{n}} \mu_\Omega)^{-1} = 10^4$ and Prandtl number $\mathrm{Pr} = \mu_\Omega / \mu_{\mathrm{n}} = 1$). The choice on these parameters will give behavior near the ideal regime with a close to maximum center of mass velocity, as demonstrated in [9]. In addition, we specify the ion-neutral collision frequency to $\nu_{\mathrm{i}} = 0$.

The dimensions of the domain default to a $50 \times 50$ in size, with the center in origo for rectangular coordinates and $R_0$ in cylindrical coordinates, giving us

$$\begin{aligned} x_{\min} = y_{\min} &= -25 \\ x_{\max} = y_{\max} &= 25 \end{aligned}$$

The spatial resolution defaults to $N_x = M_y = 1024$ and the time step size to $\Delta t = 5 \times 10^{-2}$. Recall that the velocity scaling with the normalisation of perturbation length $l$ and ideal interchange time $\gamma_{\Delta n}$ goes as

$$u_{\mathrm{c}} \sim \gamma_{\Delta n} l = \left(\frac{g}{l}\right)^{\frac{1}{2}} = c_{\mathrm{s}} \left(\frac{2l}{R_0}\right)$$

as described in section 2.2.6.

## 5.2 Re-creating previous simulations for verification

In [9] a series of simulations using the same two-field model in rectangular coordinates are carried through using a simulation code that is periodic in both directions. We will here present the re-creation of these simulations to verify the behavior of the simulation code used in this thesis after modifying it to accommodate for cylindrical geometry. The simulations of interest are variations of the diffusion and viscousity coefficients (and consequently Rayleigh and Prandtl numbers), given in table 3.

| Simulation | $\mu_\Omega$ | $\mu_n$ | Ra | Pr |
|---|---|---|---|---|
| 1 | $10^{-2}$ | $10^{-2}$ | $10^4$ | 1 |
| 2 | $5 \times 10^{-3}$ | 2 | $10^2$ | $4 \times 10^2$ |
| 3 | $5 \times 10^{-1}$ | $2 \times 10^{-2}$ | $10^2$ | $4 \times 10^{-2}$ |

Table 3: Simulations of varying Ra and Pr with spatial and temporal resolutions.

The simulations are shown in figures 6 to 8. Compared to figure 5, 12 and 13 in [9], these are near identical apart from simulation 3 presented in figure 8. We note that while the bi-periodic simulation code allows the diffused blob to propagate past the outer rightmost border, the boundary conditions in the simulation code used here force the particle density and vorticity to zero at this outer edge, effectively transporting plasma through the border.

## 5.3 Variation of major radius in $(R, Z)$

Interested in seeing the effects of the transition from a rectangular coordinate system $(x, y)$ to the cylindrical coordinate system $(R, Z)$, we here present simulations carried through in $(R, Z)$ with a variation of major radius $R_0$, using otherwise default parameters. The variations in $R_0$ (normalised by the initial blob size $l$) for each of the simulations is given in table 4. Note that for simulation 4 we let $R_0 \to \infty$, accomplished by using the rectangular coordinate system $(x, y)$.

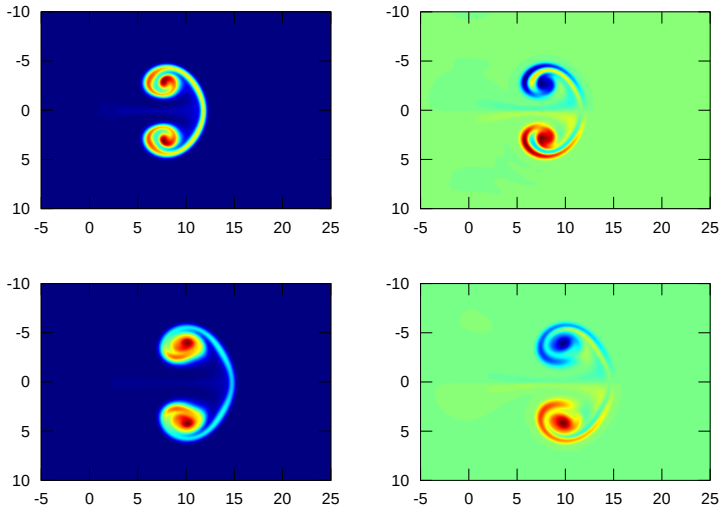


Figure 6: Evolution of simulation with $\mathrm{Ra} = 10^4$, $\mathrm{Pr} = 1$ at time $t = 15$ in the top row and $t = 20$ in the bottom row. Left column displays the particle density $n$, right column displays the vorticity $\Omega$.

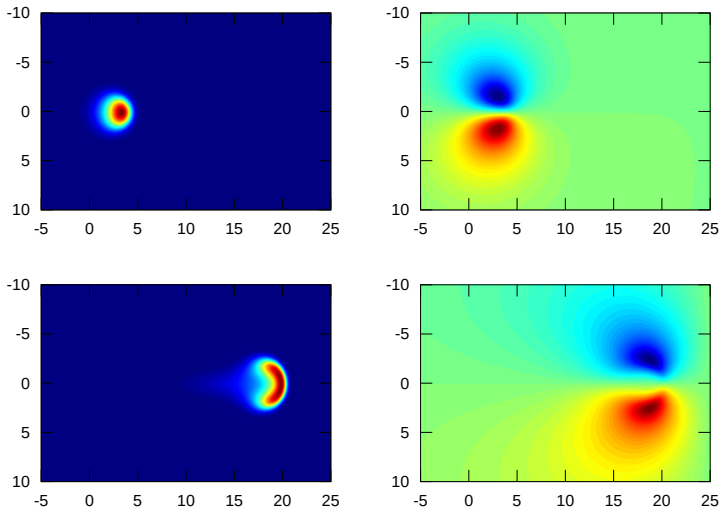


Figure 7: Evolution of simulation with $\mathrm{Ra} = 10^2$, $\mathrm{Pr} = 4 \times 10^2$ at time $t = 10$ in the top row and $t = 50$ in the bottom row. Left column displays the particle density $n$, right column displays the vorticity $\Omega$.

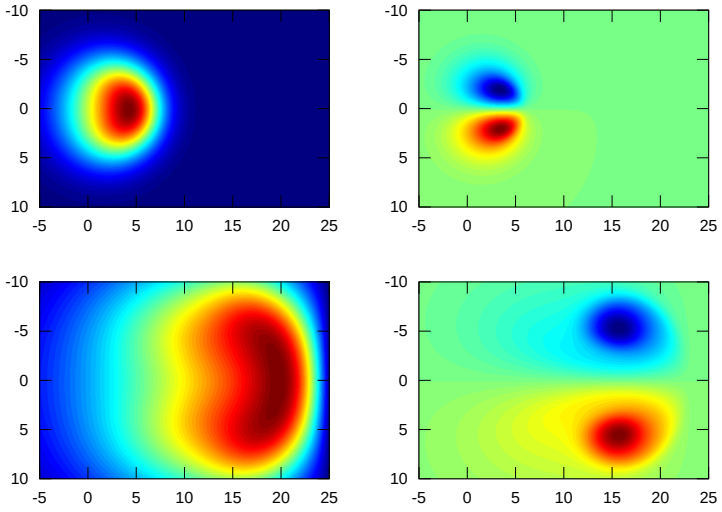


Figure 8: Evolution of simulation with $\mathrm{Ra} = 10^2$, $\mathrm{Pr} = 4 \times 10^{-2}$ at time $t = 10$ in the top row and $t = 50$ in the bottom row. Left column displays the particle density $n$, right column displays the vorticity $\Omega$.

| Simulation | $R_0$ |
|---|---|
| 1 | 25 |
| 2 | 50 |
| 3 | 100 |
| 4 | $\infty$ |

Table 4: Simulations of varying major radius $R_0$ in $(R, Z)$.

In figure 9 we see the evolution of the center of mass velocity normalised by the major radius to the left, as the velocity scales as

$$u_{\mathrm{c}} \sim \left(\frac{1}{R_0}\right)^{\frac{1}{2}}$$

To the right is the evolution of center of mass velocity kept in the same scale, effectively keeping the effective gravity $g$ constant. The rectangular coordinate limit is included for reference. We here observe that the velocities converge toward the velocities in rectangular coordinates as $R_0$ increases. We also note that the center of mass velocity has trailing effects after the maximum that are higher for small values of $R_0$. These are essentially the effects of the electric drift compression which diminishes as $R_0$ increases..

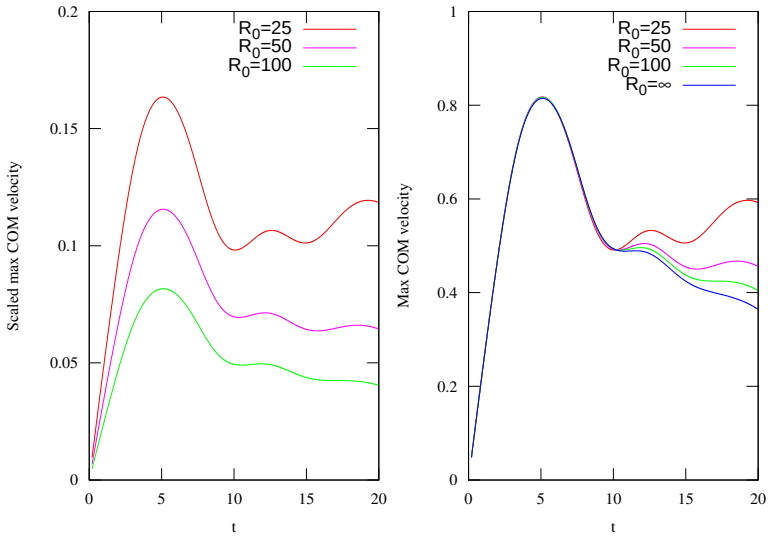


Figure 9: Scaled center of mass velocity scaled (left) and center of mass velocity with a constant effective gravity (right) for simulations with varying major radius $R_0$. We here note how a lower $R_0$ leads to a higher center of mass velocity as well as a higher trailing velocity after the maximum has been reached.

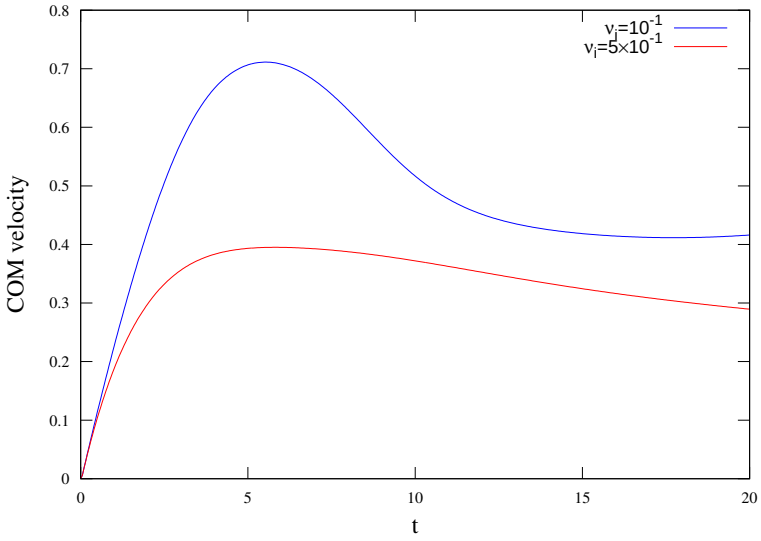


Figure 10: Center of mass for two selected simulations of $\nu_i = 10^{-1}$ and $\nu_i = 5 \times 10^{-1}$.

## 5.4 Varying ion-neutral collision frequency $\nu_{\text{i}}$

We move on to study how the systems behaves under a variation of the ion-neutral collision frequency $\nu_{\text{i}}$. This appears in the vorticity equation (19) as a friction term for vorticity and is treated implicitly in the simulation code.

Using otherwise default parameters, we here vary the collision frequency $\nu_{\text{i}}$ in a range from $10^{-2}$ to $10^{1}$ in a rectangular coordinate system $(x, y)$.

The center of mass evolution for two selected collision frequencies is shown in figure 10. In figure 11 the maximum center of mass velocity for each of the simulations is plotted as a function of $\nu_{\text{i}}$. We note that the center of mass velocity tends towards the dimensionless characteristic speed $u_{\text{c}} = 1$ for the collisionless limit, while we see the velocity approaching a velocity scaling as $1/\nu_{\text{i}}$ for the collisionally dominant regime of $\nu_{\text{i}} > 1$.

Also of interest is noting that, apart from the impact on velocity, $\nu_{\text{i}}$ also has an impact on the so called mushroom-like shape of the propagating blob structure. Figure 12 shows snapshots of two simulations for $\nu_{\text{i}} = 0.1$ and $\nu_{\text{i}} = 0.5$ for $t = 20$, both the particle density $n$ and vorticity $\Omega$. The here see that a higher $\nu_{\text{i}} = 0.5$ leads to a more simple blob-like structure while $\nu_{\text{i}} = 0.1$ retains several characteristic traits of the mushroom structure. This transition from mushroom-like to circular blob structure is equally visible in the entire range of $\nu_{\text{i}}$ but have here been omitted for brevity.

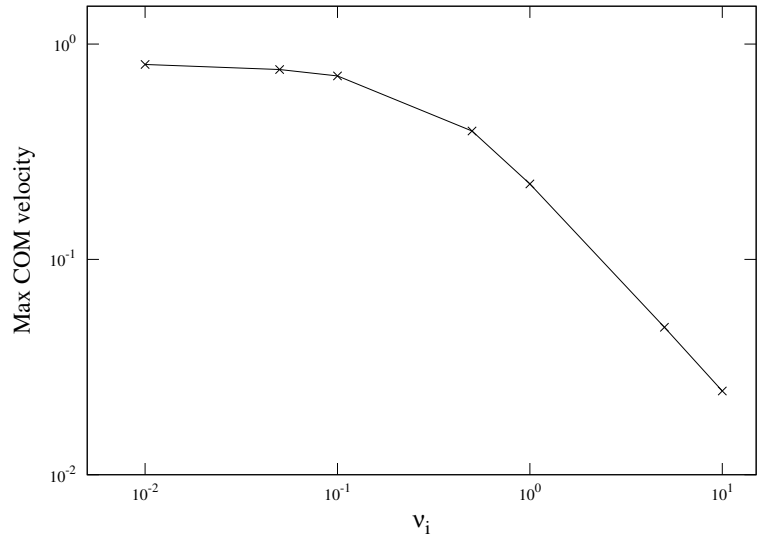


Figure 11: logarithmic plot of maximum radial center of mass velocity for varying ion-neutral collision frequency $\nu_i$ in rectangular geometry $(x, y)$. Note that the maximum center of mass velocity approaches the characteristic speed $u_c$ as the friction takes on lower values.

We shall briefly set this simulation in a context to emphasise it: The velocity of propagating depletions is described in [16] with the expression

$$u = \frac{1}{8} R_{\text{dep}} \left[ \left( \nu_{\text{in}}^2 + \frac{16 g_{\text{E}}}{R_{\text{dep}}} \right)^{\frac{1}{2}} - \nu_{\text{in}} \right] \tag{37}$$

where $R_{\text{dep}}$ in this case is the radius of the depletion front, $\nu_{\text{in}}$ is the ion-neutral collision rate and $g_{\text{E}}$ is the Earth's gravity (analogue to the effective gravity in the 2-field model), we note that the results is a qualitative match with the simulation results. By fitting the parameters $R_{\text{dep}}$ and $g_{\text{E}}$ we can get a matching curve, as demonstrated in figure 13. This is further discussed in chapter 8.

## 5.5 Variation of blob centering

So far we have used a centered perturbation in a large box where the effects of the borders have been negible in the time frames used. We will here compare how blobs placed at the innermost wall will propagate and evolve.

Using otherwise default parameters, we compare three simulations with blob centering $\mathbf{x}_0 = (x_0, 0)$, where $x_0 = 0$, $x_0 = -24$ and $x_0 = -23$.

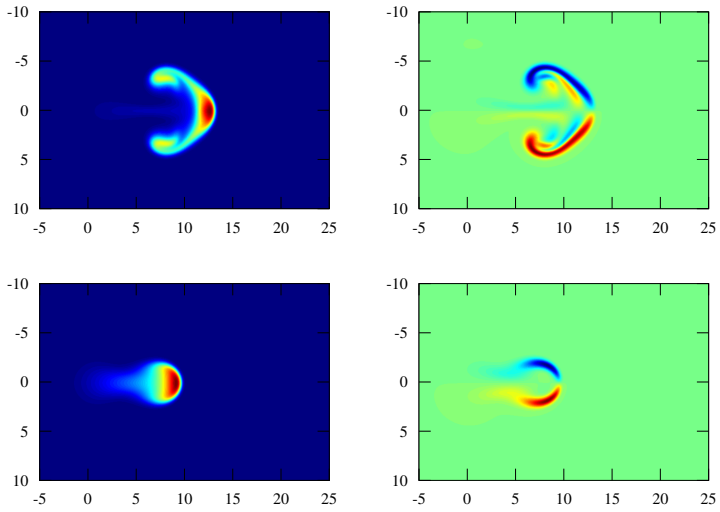


Figure 12: The particle density (left) and vorticity (right) for two simulations at $\nu_{\mathrm{i}} = 0.1$ (top) and $\nu_{\mathrm{i}} = 0.5$ (bottom), time $t = 20$. Note here that the simulation for the lower $\nu_{\mathrm{i}}$ yields a structure resembling more of a mushroom shape, while for the simulation with higher $\nu_{\mathrm{i}}$ there is a more coherent blob-like structure.

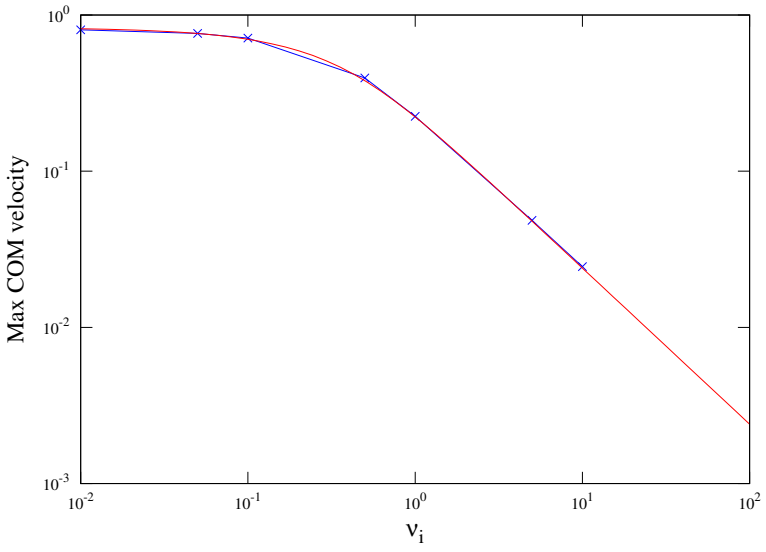


Figure 13: The radial center of mass velocity for varying ion-neutral collision frequency $\nu_{\mathrm{i}}$, as in figure 11, plotted against a fit for an expression (Eq. (37)) for the velocity.

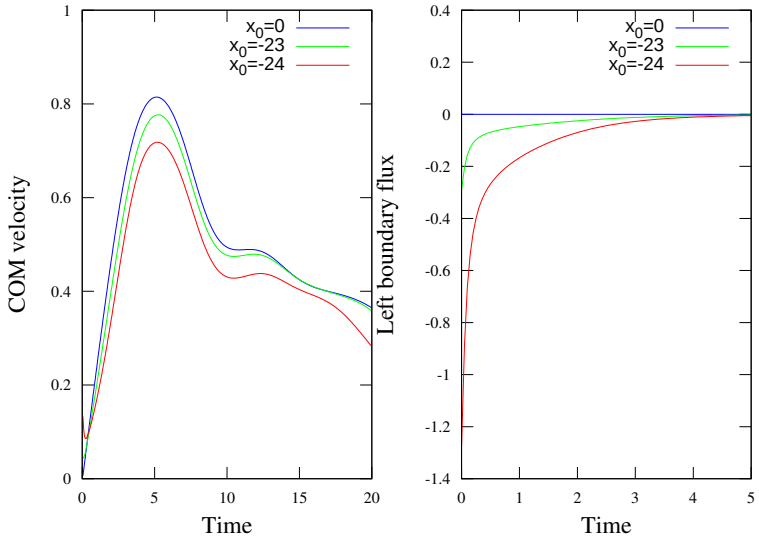


Figure 14: Center of mass velocity (left) and left boundary flux of the particle density $n$ (right) for varying blob centerings $x_0$. We note the loss of $n$ through the left boundary for blobs centered close to the border, causing the overall reduced center of mass velocity.

Figure 14 shows how the center of mass velocity is affected by this difference in centering and the boundary flux of the particle density on the innermost boundary. We see that the initial center of mass velocity is much higher for a blob close to the boundary than for a centered blob, but quickly drops below the velocity of the centered blob before $t = 1$. The continued propagation of this blob is then lower for blobs closer to the boundary for the duration of the simulation. Looking at the boundary flux, we see that for the blobs near the boundary there is a large loss of the particle density $n$, causing the amplitude of the blob structure to decrease and possibly account for the then lower center of mass velocity.

Looking at the particle density and the potential field for $x_0 = 0$ and $x_0 = 24$ at $t = 5$ in figure 15 we see that the blobs are similar in structure, although for $x_0 = 24$ we see that the potential field is clearly influenced by the boundary conditions of the electrostatic potential.

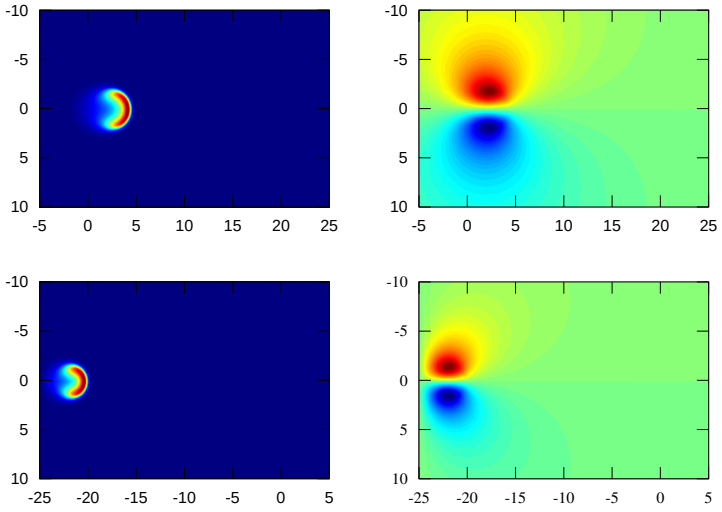


Figure 15: The particle density $n$ (left) and velocity potential $\phi$ (right) for a blob centered around $x_0 = 0$ (top) and $x_0 = -24$ (bottom). We see here the tilted potential for when the blob is positioned close to the border.

## 5.6 Variation in ion-neutral collisions and blob amplitude

Using the logarithmic formulation of the particle density, described in section 2.2.5,

$$\frac{d\Omega}{dt} = \mu_\Omega \nabla_\perp^2 \Omega - \nu_\mathrm{i}\Omega - \frac{\partial \ln n}{\partial Z}$$
$$\frac{d \ln n}{dt} = \mu_\mathrm{n} \nabla_\perp^2 \ln n + \mu_\mathrm{n} \left(\nabla_\perp \ln n\right)^2 + \frac{2}{R_0}\frac{\partial \phi}{\partial Z}$$

we can perform simulations over a range of perturbation amplitudes as well as ion-neutral collisions in order to investigate how these parameters affect the transport of plasma. Note that the normalisation here is given by the blob size $l$ and the inverse interchange time $\gamma = (g/l)^{1/2}$.

Figure 16 shows the dependence of the maximum radial center of mass velocity on both the relative perturbation amplitude $\eta = \Delta n/n_0$ and ion-neutral collision frequency $\nu_\mathrm{i}$, with $\eta$ ranging from 3 to 12 and $\nu_\mathrm{i}$ ranging from $10^{-3}$ to 10 and otherwise default parameters. It is clear that the maximum center of mass velocity exhibits a similar profile with regard to its dependence on the ion-neutral collision frequency regardless of perturbation amplitude. We also see that the center of mass velocity is also dependent on the perturbation amplitude, where a higher perturbation amplitude yields a higher velocity, close to proportional to the relative perturbation amplitude $\eta$.

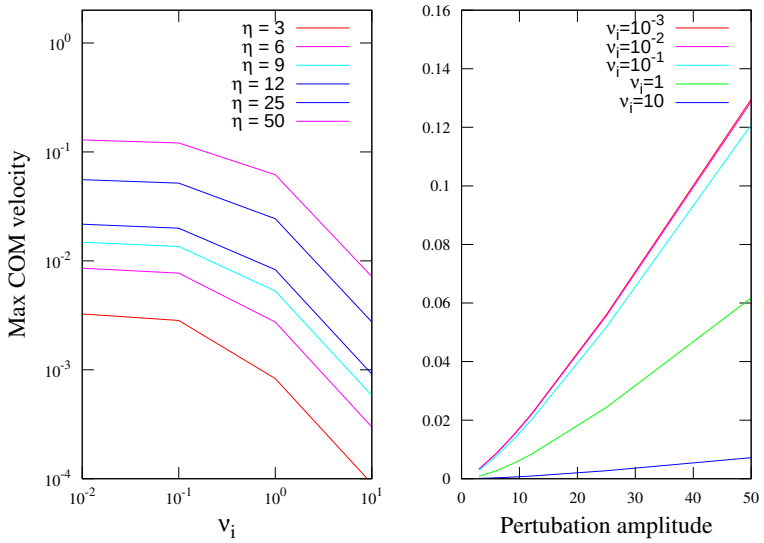


Figure 16: Maximum center of mass velocity as a function of ion-neutral collision frequency (logarithmic, left) and perturbation amplitude (right), clearly showing the dependence of $\nu_{\mathrm{i}}$ and $\eta$ for the center of mass velocity.

## 5.7 The convective term

Although the convective term in Eq. (13) is assumed to be negible for small perturbations as described in section 2.4, the effects it has on larger perturbations is not known. This term has here been implemented into the time splitting scheme as an explicit term to quantify the effects it has on the center of mass velocity.

Figure 17 shows a simulation with a relative perturbation amplitude of $\eta = 12$ with and without the convective term. We see that although the maximum velocity is higher for simulations including the convective term, it has a small effect on the center of maximum mass velocity.

## 5.8 On numerical instabilities

The simulations carried out may become, unless parameters are chosen carefully, numerically unstable. We will here present conditions where this might happen and how these numerical instabilities manifest themselves.

A stability constraint for the diffusion equation is

$$\frac{c\Delta_t}{\Delta_x^2} < 1$$

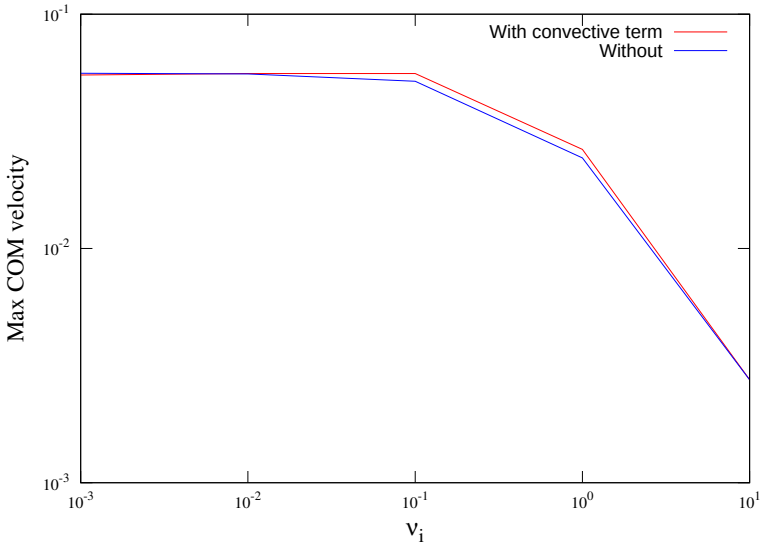


Figure 17: Simulations with and without the convective term, for $\eta = 25$ for varying $\nu_{\mathrm{i}}$. We note the slightly higher radial center of mass velocity when including the convective term.

where $c$ is a constant. Although this is a necessary condition, it might not be sufficient for numerical stability. The value of $c$ can be thought as the speed of propagation through the system, effectively setting a constraint on the relation between the spatial and temporal discretisation as well as relating it to the velocity. [17, 18]

In our simulations, the value of $c$ will for diagnostic purposes be the maximum velocity in the system $u_{\max}$, and we will refer to this as the CFL number:

$$\text{CFL number} = \frac{u_{\max}\Delta_t}{\Delta_x^2}$$

Providing an example is a simulation for a box size of $8 \times 10$ with a resolution $N_x = N_y = 1024$ and a time step size of $\Delta t = 10^{-2}$, using the logarithmic formulation of the particle density and a perturbation amplitude of $\Delta n = 16$. For this specific example, $\mathrm{Ra} = 10^4$ and $\mathrm{Pr} = 1$.

Figure 18 demonstrates one such numerical instability forming between the time step $t = 2.20$ and $t = 2.30$. We clearly see the jagged gradient profile giving a non-physical oscillation which grows quickly. As seen in figure 19 the CFL number, indicating that the system is indeed becoming numerically unstable, as well as the physical quantities such as the integrated particle density, mean energy and fluctuation energy instantly growing towards unrealistic numbers. The mean energy and fluctuating energy are here the defined as the respective

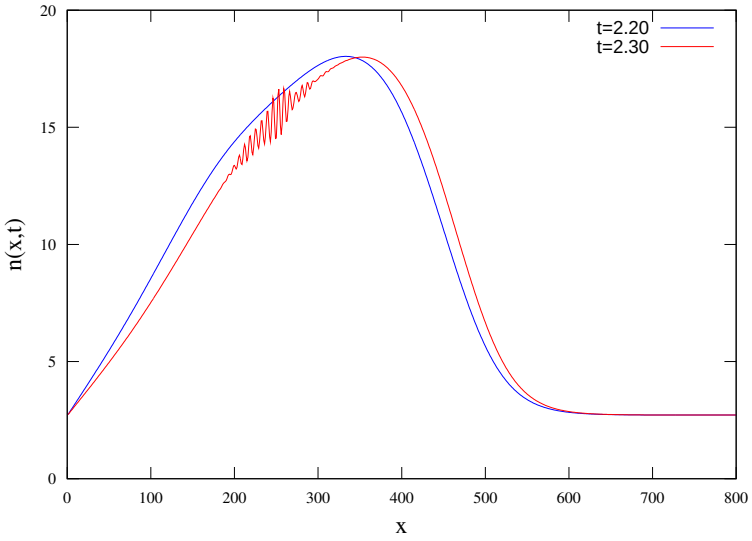


Figure 18: Profile cut of the particle density at the center line $y_0 = 0$ for a simulation becoming numerically unstable between time steps $t = 2.20$ and $t = 2.30$. Note the formation of non-physical oscillations.

integrals

$$\int d\mathbf{R}\,\frac{1}{2}\left[\left(\frac{\partial\phi}{\partial R}-\overline{\frac{\partial\phi}{\partial R}}\right)^2+\frac{\partial\phi}{\partial Z}\right] \qquad \int d\mathbf{R}\,\frac{1}{2}\left(\overline{\frac{\partial\phi}{\partial R}}\right)^2$$

Note that the growth of these physical quantities lag behind the actual formation of the instability.

Parameters that often cause numerical instability in our system includes large relative perturbation amplitudes $\eta$ and high Rayleigh numbers ($10^6$ and above) that leads to steep gradients in particle density. Care must be taken when simulating such conditions to accommodate for them with high enough spatial and temporal resolutions.

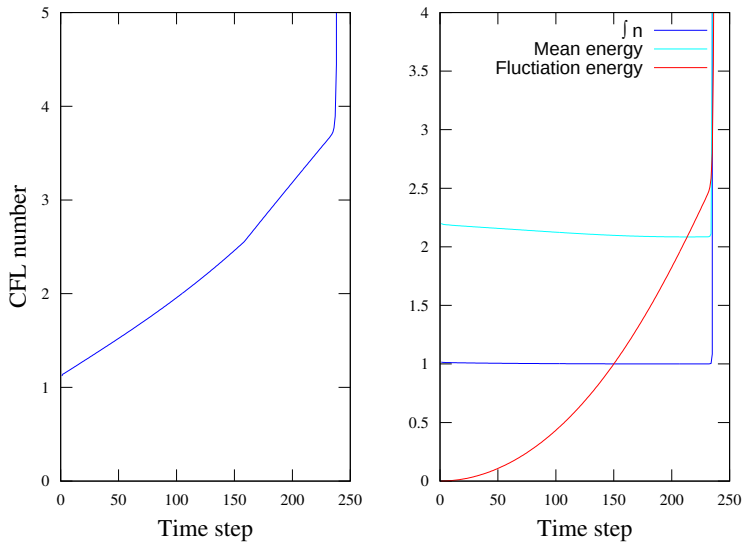


Figure 19: The CFL number for the simulation becoming unstable (left) as well as physical quantities integrated particle density, mean and fluctuation energy (right). These quantities are normalised with an arbitrary quantity to fit within the same figure, as their actual values are not for this case important. Note the instant growth right before $t = 2.40$, as an effect of the earlier created instability.

# 6 Experiments on the VTF

We will in this first section present the paper "Experiment on propagation of plasma filaments" [13] and its findings. In the second section we take a closer look at the data sets from these experiments.

## 6.1 Experiments on the propagation of plasma filaments

The contents of the following section is taken from [13] in its entirety unless otherwise noted. This includes figures 20, 21 and 22 which has been reproduced by kind permission of Noam Katz. Note that the numbering of figures is altered to match the numbering in this thesis.

The experiment is described as follows:

> The basic experimental setup in Fig. 20 shows how a reproducible blob is created. The imposed toroidal magnetic field varies as $1/R$ and has the value $B_0 = 40$ mT at $R_0 = 1$ m (where $R$ is the major radius). The breakdown is driven by a 60 $\mu$s pulse of microwaves at a frequency of 2.45 GHz and power of 15 kW. Near the inner wall, the $1/R$ main field is enhanced by a toroidal solenoid (Fig. 20 at left) such that the field strength at this location reaches the 87 mT required for electron-cyclotron resonant breakdown. Even though the rf power is injected by a single horn antenna, the microwave reflections of the chamber walls give axially symmetric breakdown. Since the field is purely toroidal, there are no field lines terminating at the wall and therefore no sheaths to drain current along the field.
>
> The plasma [...] consists mainly of singly charged argon ions and is created in a chamber filled with argon gas at $10^{-5}$–$10^{-4}$ torr. Since the plasma density is typically $2 \times 10^{16}$ m$^{-3}$, the ionization fraction is only about 1%, and there is a constant background of neutrals even after breakdown.
>
> The plasma is tracked by an array of 200 Langmuir probes. The tip spacing is 7 cm horizontally and 7 cm vertically, with triple resolution (horizontally) near the center. The main Langmuir probe array is located at a single toroidal angle, but other Langmuir probes are used to verify the azimuthal symmetry of the blobs.

Further, it is stated that:

> We observe experimentally for the first time the mushroom blob shape, which has been seen in many simulations [...]. This shape is displayed in Fig. 21, which shows the propagation of a typical blob in poloidal cross section. The time step between adjacent density

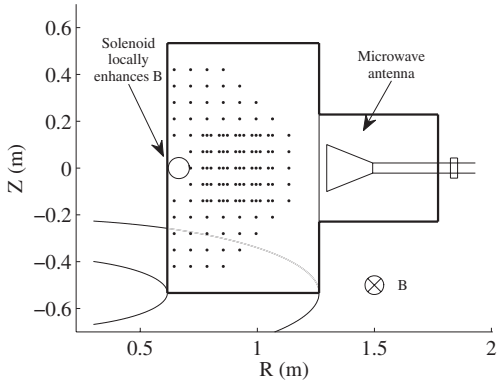


Figure 20: A poloidal cross section of the experimental setup. At right is the rf horn. At left is the solenoid (circle), with toroidal magnetic field into the plane. The dots represent Langmuir probes. There is also a background toroidal magnetic field (into the page). Note that the horn and Langmuir probes are not at the same toroidal angle. [13]

> plots is 100 $\mu$s, and the first plot occurs 25 $\mu$s after the microwaves are turned off. [...]
>
> The propagation seen in Fig. 21 can be quantified and it is found to depend on the neutral pressure in the chamber. This dependence is explored in separate plasma discharges covering a range of neutral pressures. As Fig. 22 shows, we find that the blob's center-of-mass speed is inversely proportional to the neutral pressure. [...]

## 6.2 Further observations

The following section is based on findings from the actual data sets from these experiments presented in [13], kindly provided by Noam Katz.

Data from fifteen experiments were provided, where a plasma blob was recorded propagating in a toroidal magnetic field at neutral pressures $P_{\mathrm{n}}$ ranging $0.412 \times 10^{-4} - 4.54 \times 10^{-4}$ torr, as described in the previous section.

The data sets consists, for each experiment, of a grid of measured ion saturation currents $j_{\mathrm{sat}}$ on the Langmuir probes for a single azimuthal angle and a time record of the microwave pulse used to create the blob structure. The floating potential measurements described in [13] has not been provided to us in these data sets.

The relation between electron density $n_{\mathrm{e}}$ and $j_{\mathrm{sat}}$ is, by assuming cold ions as

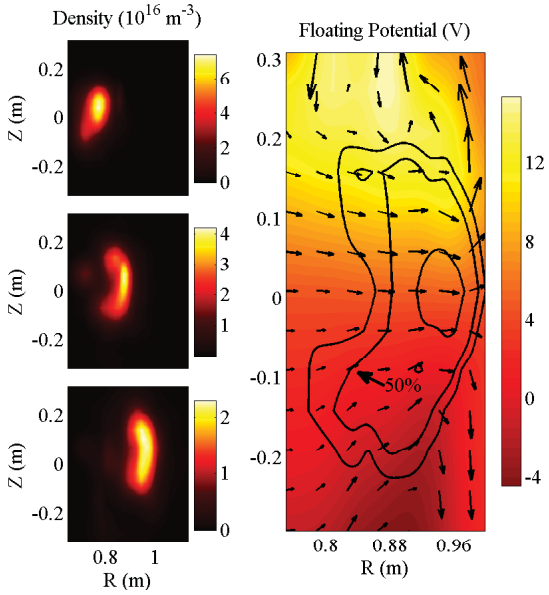


Figure 21: Poloidal cross section of typical blob at 3 different times ($\Delta t = 100$ $\mu$s), showing characteristic mushroom shape. The density is calculated from ion saturation current; its decrease is consistent with the expansion of the blob. The blob propagation is consistent with the vertical electric field, which is reflected in the potential structure at the right. The overlaid $\mathbf{E} \times \mathbf{B}$ velocity arrows show the velocity field of a vortex pair. [13]

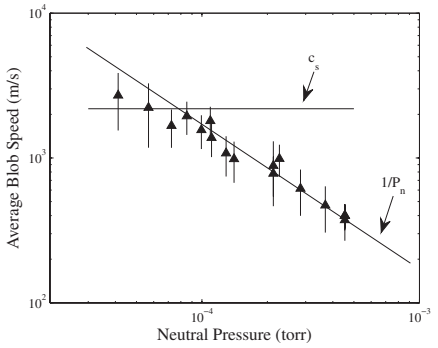


Figure 22: Blob center-of-mass speed versus neutral pressure ($P_{\mathrm{n}}$). The speed scales inversely with the pressure, but this scaling appears to break down at low pressure. The error in speed is approximated by the standard deviation of the inferred blob speed as it fluctuates in time. [13]

described in [6], given by

$$j_{\text{sat}} = -e \left( \frac{T_e}{m_i} \right)^{\frac{1}{2}} n_e \tag{38}$$

where $e$ is the electron charge, $T_{\text{e}}$ is the electron temperature and $m_{\text{i}}$ is the ion mass. Assuming quasineutrality ($n = n_e \approx n_i$) and ignoring temperature fluctuations, the negative current is then proportional with the particle density $n$. According to personal communication with Katz, the maximum particle density in each experiment is assumed to be $n_{\text{max}} = 10^{16}$ m$^{-3}$.

Positive records of $j_{\text{sat}}$ is contained in the data sets which through Eq. (38) would yield a un-physical negative plasma densities. The ion current density measured by the Langmuir probes is assumed to be inaccurate for low plasma densities. Due to this, we assume these measurements to indicate no plasma or a near-zero particle density.

### 6.2.1 Inspecting the experiments

The quality of the experiments with respect to how well they generate and lead to propagation of one single blob varies. We will in this section present considerations from inspecting the experiments individually. Several of the figures presented here have not been published previously.

The microwave pulses used to generate a plasma are typically 60 $\mu$s in length. The pulse starts and ends at the approximate same time steps for all experiments except at the upper neutral pressure range where variations in pulse length are seen.

For the lower range of $P_{\text{n}}$ from $0.412 \times 10^{-4}$ to $0.724 \times 10^{-4}$ torr, multiple or complex structures are being formed and propagated, as shown in figure 23. We note strong radial dispersion as the structures are propagated outwards.

For the medium pressure range from $0.852 \times 10^{-4}$ to $2.84 \times 10^{-4}$ torr we see one clear blob-like structure forming and propagating, as shown in figure 24. Only for the higher range from $P_{\text{n}}$, $2.13 \times 10^{-4}$ to $2.84 \times 10^{-4}$ torr do we see a mushroom-like shape being formed and propagated, shown in figure 25. Figure 27 shows the early formation of these mushroom structures at $P_{\text{n}} = 2.13 \times 10^{-4}$ torr in a higher temporal sampling.

For the highest neutral pressure range from $3.69 \times 10^{-4}$ to $4.54 \times 10^{-4}$ torr we see two blob-like structures forming and propagating outwards. The experiments are summarised in table 5.

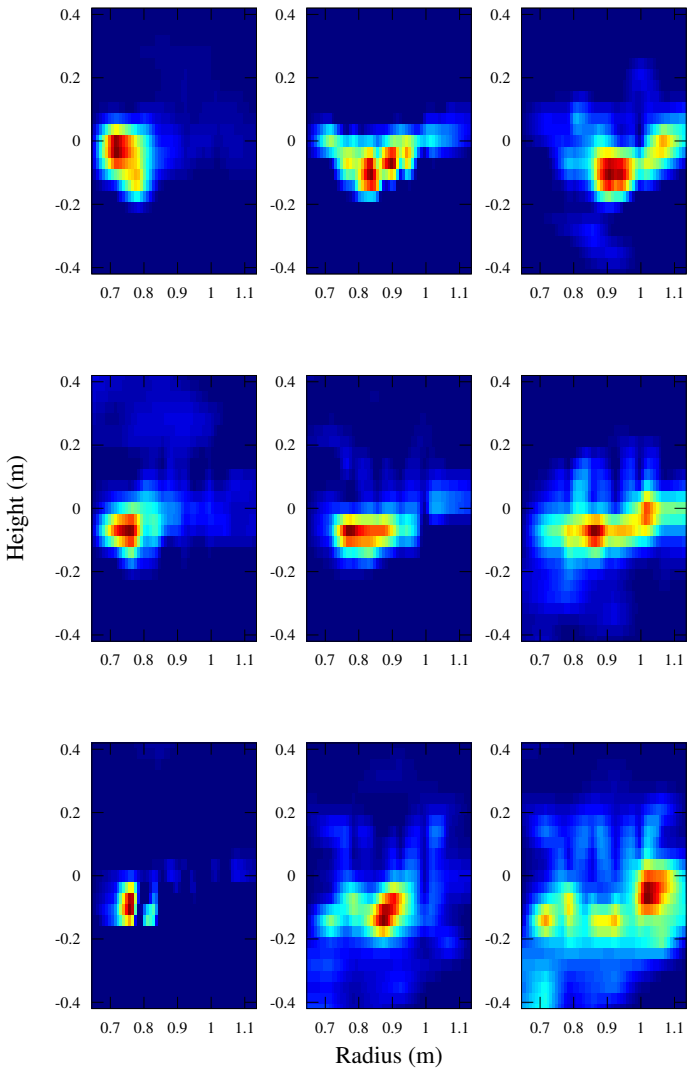


Figure 23: Experiments at neutral pressure $P_{\mathrm{n}}$ being $0.412 \times 10^{-4}$, $0.568 \times 10^{-4}$ and $0.724 \times 10^{-4}$ torr (vertically). The snapshots are taken at $t = 20\mu$s after the end of the microwave pulse, and in increments of $50\mu$s (horisontally). We note the formation of complex structures and strong dispersion. Not previously published.

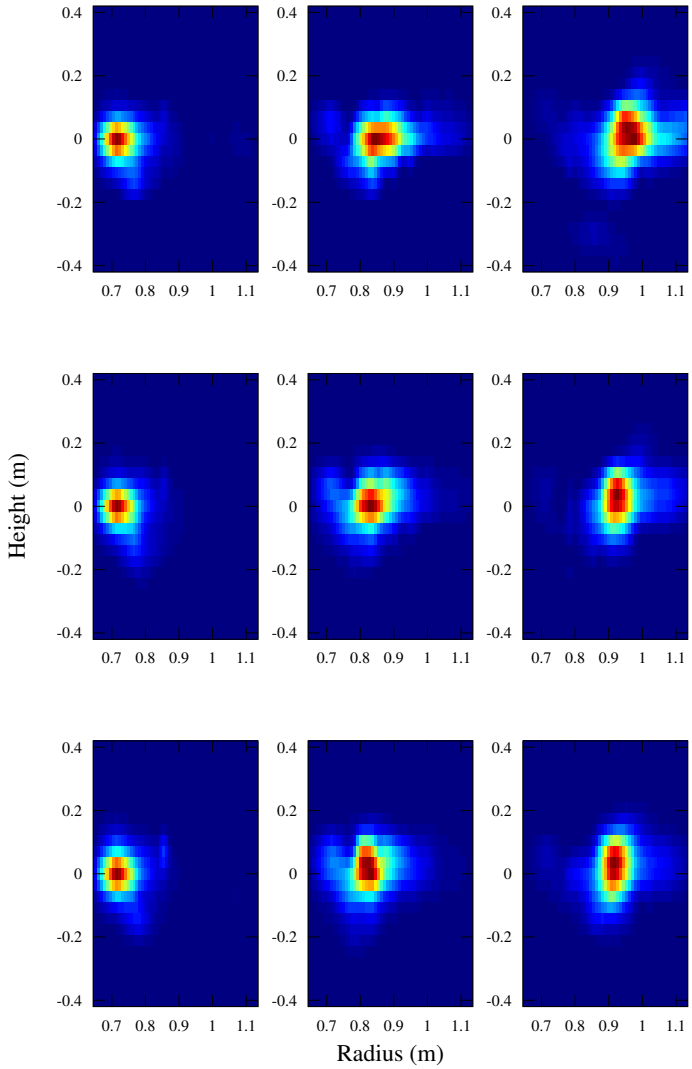


Figure 24: Experiments at neutral pressure $P_{\mathrm{n}}$ being $0.852 \times 10^{-4}$, $0.994 \times 10^{-4}$ and $1.11 \times 10^{-4}$ torr (vertically). The snapshots are taken at $t = 20\mu$s after the end of the microwave pulse, and in increments of $50\mu$s (horisontally). We note a single blob structure forming and propagating outwards. Not previously published.

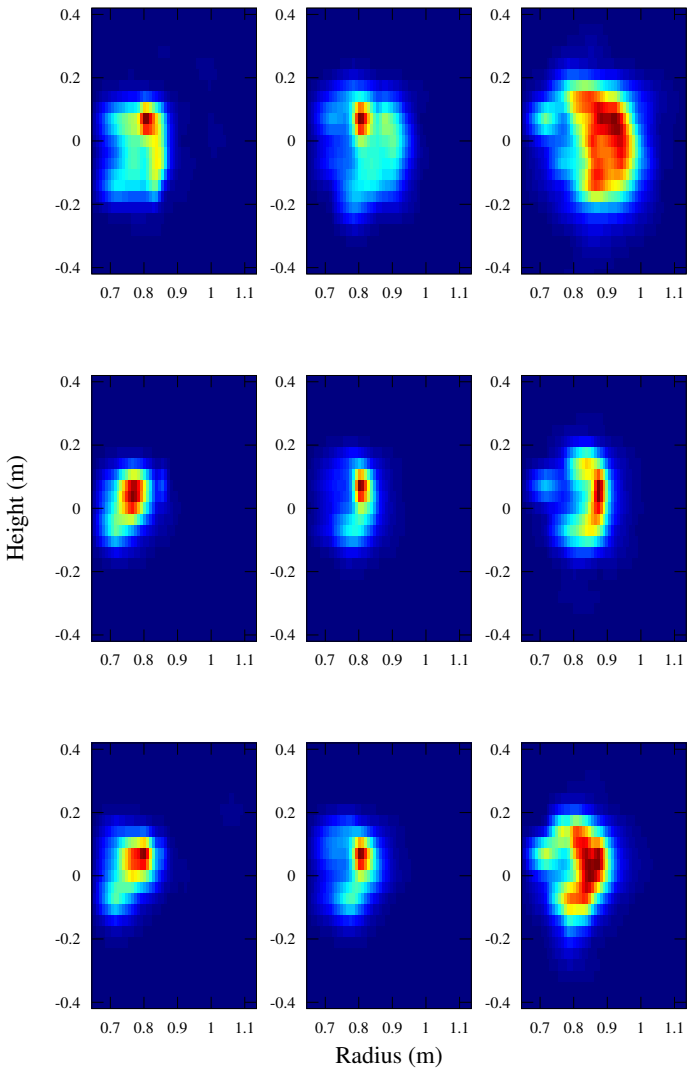


Figure 25: Experiments at neutral pressure $P_{\mathrm{n}}$ being $2.13 \times 10^{-4}$, $2.13 \times 10^{-4}$ and $2.84 \times 10^{-4}$ torr (vertically). The snapshots are taken at $t = 20\mu$s after the end of the microwave pulse, and in increments of $50\mu$s (horisontally). We note a single blob structure forming and propagating outwards while giving rise to a mushroom shape. Not previously published.

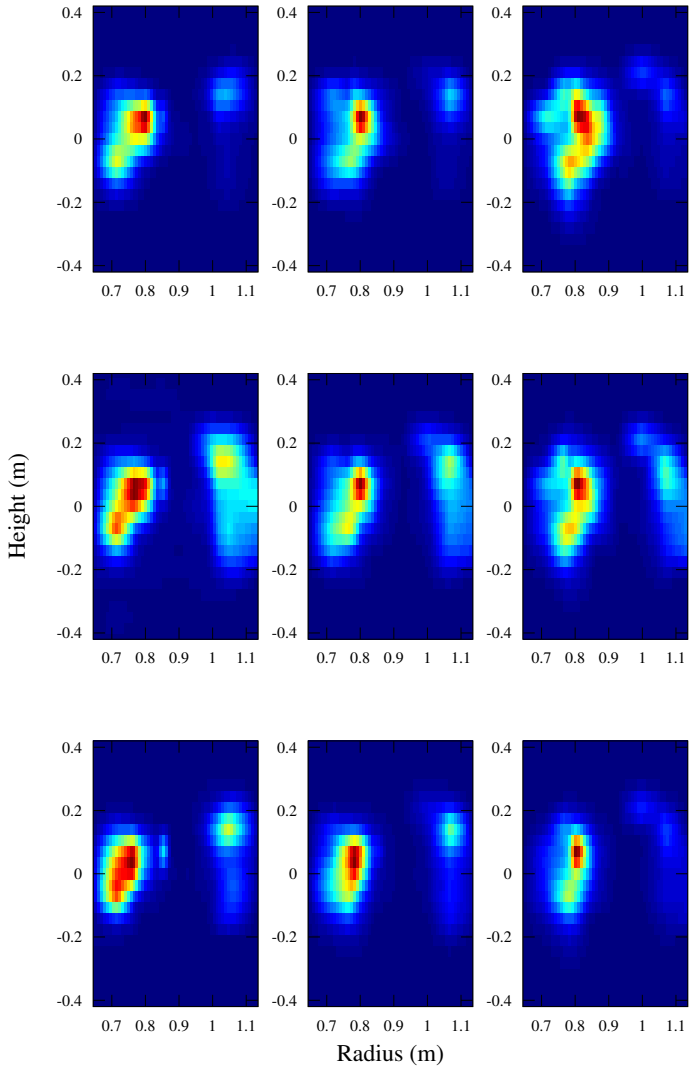


Figure 26: Experiments at neutral pressure $P_{\mathrm{n}}$ being $3.69 \times 10^{-4}$, $4.54 \times 10^{-4}$ and $4.54 \times 10^{-4}$torr (vertically). The snapshots are taken at $t = 20\mu$s after the end of the microwave pulse, and in increments of $50\mu$s (horisontally). We note two blob structures forming and propagating outwards. Not previously published.

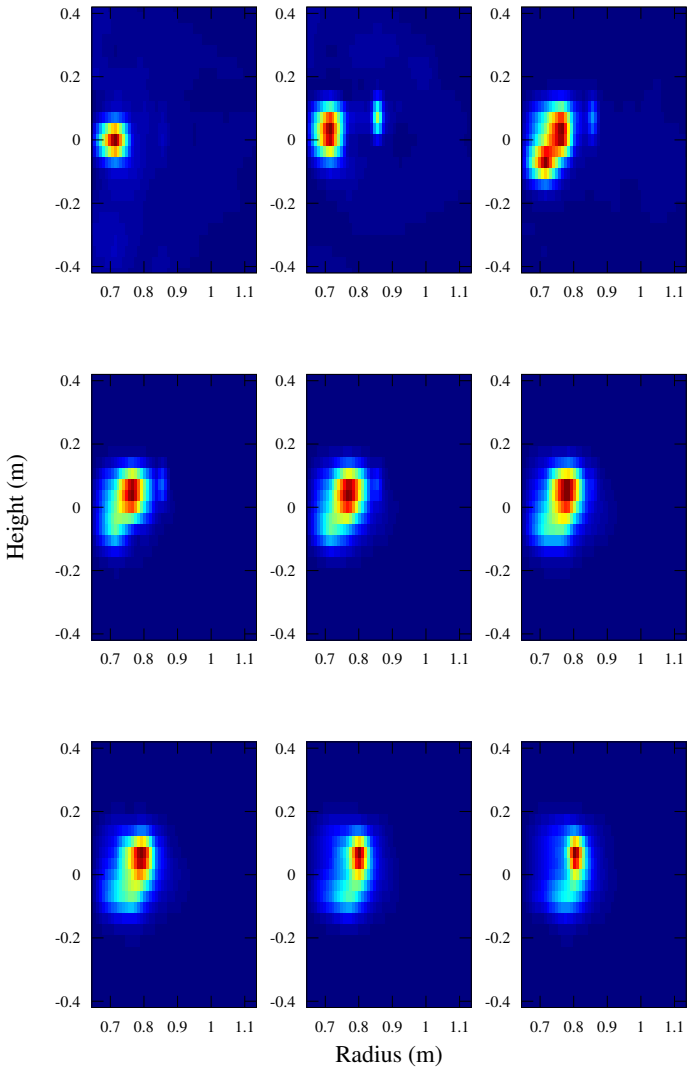


Figure 27: Initial times of experiment with neutral pressure $P_{\mathrm{n}} = 2.13 \times 10^{-4}$ torr at time $t = -20\mu$s before the microwave pulse end and in increments of $10\mu$s, showing the early formation of the mushroom structure. Not previously published.

| $P_{\mathrm{n}}(10^{-4}$ torr) | Experiments | Description |
|---|---|---|
| 0.412–0.724 | 3 | Multiple blobs, strong dispersion |
| 0.852–1.41 | 5 | Single blob |
| 2.13–2.84 | 3 | Single blob, mushroom shape |
| 3.69–4.54 | 4 | Multiple blobs |

Table 5: Neutral pressure range ($P_{\mathrm{n}}$), number of experiments and structure description for the experiments.

### 6.2.2 Center-of-mass velocity for the experiments

The radial center of mass coordinate $R_{\mathrm{COM}}$ for each experiment is calculated by the integrals in cylindrical geometry

$$R_{\mathrm{COM}} = \frac{\int n\, R d\mathbf{R}}{\int n\, d\mathbf{R}} = \frac{\int j_{\mathrm{sat}}\, R d\mathbf{R}}{\int j_{\mathrm{sat}}\, d\mathbf{R}}$$

for each time step. We can find the center of mass velocity

$$V_{\mathrm{COM}} = \frac{dR_{\mathrm{COM}}}{dt}$$

by fitting the center of mass position to a straight line from when the perturbation starts propagating after the microwave pulse has ended to the maximum radial position, which can be when the blob structure hits the chamber wall. The average center of mass velocity can then be taken as the slope of this line. This is demonstrated in figure 28 where we see the fitting curve on the center of mass position after the microwave pulse has ended. This evolution of center of mass coordinates is typical for all experiments.

This approach is in agreement to the method employed by [13]. The resulting fig. 29 shows the average center of mass velocity as a function of the neutral gas pressure. A fitted curve proportional to $1/P_{\mathrm{n}}$ is also plotted, along with the estimated sound speed $c_{\mathrm{s}}$.

Note that in this plot we have not included the average center of mass velocities for very low neutral pressures ($0.412 \times 10^{-4}$ to $0.724 \times 10^{-4}$ torr) as these are not coherent blob structures.

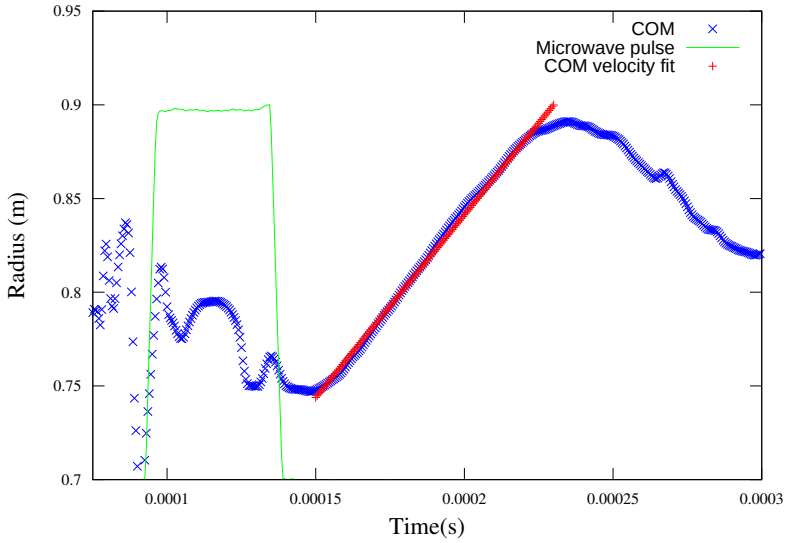


Figure 28: The center of mass coordinate $R_{\mathrm{COM}}$ for the experiments (blue) fitted against a center of mass velocity (red) for the average velocity between start position and end position at the neutral gas pressure $P_{\mathrm{n}} = 8.5 \times 10^{-5}$ torr. The microwave pulse effect (without units) is plotted in green for reference.

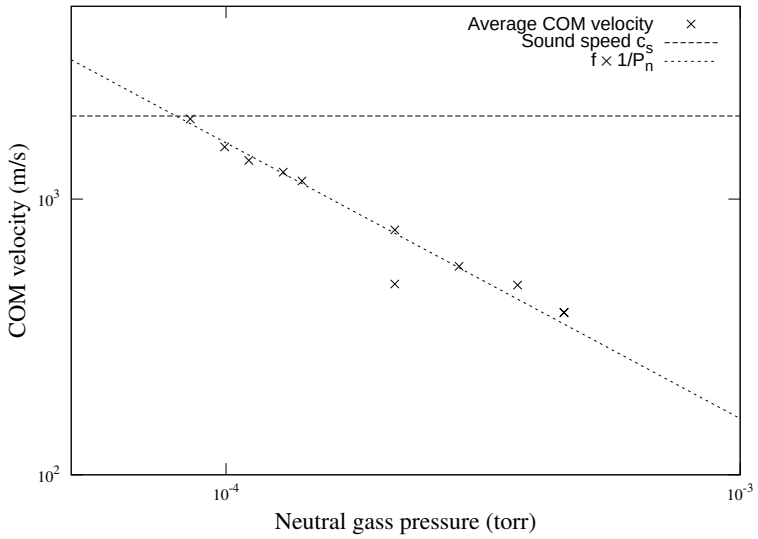


Figure 29: Average center of mass velocities as a function of neutral pressures $P_{\mathrm{n}}$ from the VTF experiments. The horisontal line indicates the estimated sound speed $c_{\mathrm{s}} = 2 \times 10^3 \mathrm{m/s}$. The diagonal line is proportional to $1/P_{\mathrm{n}}$.

# 7 Simulations of plasma filament structures

We will now turn to numerical simulations of the plasma dynamics in the experiment described in chapter 6. We will find the simulation parameters and the scaling that fits best with the physical environment. By doing this we will test the measured relation between the neutral gas density and the outward blob propagation velocity. The qualitative behavior of reduced propagation speed for increased ion neutral collisions in the vorticity equation is already demonstrated to hold for the simulations in section 5.

## 7.1 Setting the parameters

The experiment has a machine cross section of $60 \times 80$ cm with a blob size of $l = 0.075$ m. The major radius of the torus is $R_0 = 0.90$ m. Normalising this to the blob size we get the geometrical parameters for our simulation; box size in the radial and vertical direction as $L_R = 8$, $L_Z = 5$ and major radius to the center of the box as $R_0 = 12$. This also places the starting position of the blob at the innermost edge; using the position vector $\mathbf{r} = \mathbf{R} - \mathbf{R}_0$ where $\mathbf{R} = (R, Z)$ and $\mathbf{R}_0 = (R_0, 0)$ is the center of the cross section, we have the blob at the initial radial starting point $r_0$ relative to $R_0$ as $\mathbf{r}_0 = (R_0 - r_0, 0) = (-3, 0)$.

The ion-neutral collision frequency depends on the neutral gas particle density $n_\mathrm{g}$ as well as the thermal speed $u_\mathrm{ti}$ of the ions, as

$$\nu_\mathrm{in} = u_\mathrm{ti} \sigma n_\mathrm{g} \tag{39}$$

where $\sigma$ is the collisional cross section of ions, set to $\sigma = 58 \times 10^{-20}$ m$^2$ [13]. Referring to [13], the temperature of the ions $T_\mathrm{i}$ is estimated to 1 eV, giving us the thermal speed for ions as

$$u_\mathrm{ti} = \sqrt{\frac{T_\mathrm{i}}{m_\mathrm{i}}} \approx 2.5 \times 10^3 \text{ m/s}$$

where $m_\mathrm{i}$ is the argon ion mass. We can express the neutral gas density by the neutral pressure $p_\mathrm{n}$, assuming the ideal gas law on the form $p_\mathrm{n} = n_\mathrm{g} T$ with $T$ being the neutral gas temperature. We assume the neutral gas temperature to be $T \approx 300$ K. With this, the neutral gas density becomes

$$n_\mathrm{g} = 3.2 \times 10^{18} \text{ m}^{-3}$$

assuming the pressure to be $p_\mathrm{n} = 10^{-4}$ torr, characteristic for the experiments. The conversion factor between torr and Pascal is 1 Pa $\approx 7.5 \times 10^{-3}$ torr.

For the model interchange time $\gamma$, with the sound speed as $c_\mathrm{s} = 2.2 \times 10^3$ m/s for the electron temperature $T_\mathrm{e} = 2$ eV, we get

$$\gamma = c_s \left( \frac{2}{Rl} \right)^{\frac{1}{2}} \approx 1.2 \times 10^4 \text{ s}^{-1}$$

which gives us the simulation model parameter $\nu_{\mathrm{i}}$ by

$$\nu_{\mathrm{i}} = \frac{\nu_{\mathrm{in}}}{\gamma} \approx 4 \times 10^{-1}$$

As this is the parameter for a characteristic neutral gas pressure, our parameter scan in the ion collisional parameter ranges from $10^{-3}$ to 10. For the pressure ranges used in the VTF experiments ($4.1 \times 10^{-5}$ torr to $4.5 \times 10^{-4}$ torr) the range of $\nu_{\mathrm{i}}$ is 0.16 to 1.8.

We will now seek to find an estimate for the collisional diffusion. With the normalised collisional diffusion for the particle density given as

$$\mu_{\mathrm{n}} = \frac{\nu_{\mathrm{en}}}{\omega_{\mathrm{ce}}} \frac{T_{\mathrm{e}}}{eB} \frac{1}{\gamma l^2}$$

as shown in section 2.2.5 we assume the electron-neutral collision frequency to take the same form as the ion-neutral collision frequency in Eq. (39), with the thermal velocity of electrons as $u_{\mathrm{te}} = (T_{\mathrm{e}}/m_{\mathrm{e}})^{\frac{1}{2}}$. With $B \approx B_0 = 40$ mT this then gives us the approximate value of $\mu_{\mathrm{n}} \approx 10^{-6}$. Using the expression for the viscosity Eq. (16), with $\ln \Lambda \approx 10$, we find that the diffusive viscosity coefficient $\mu_\Omega \approx 2 \times 10^{-2}$. These gives us a Rayleigh number of $\mathrm{Ra} \approx 5 \times 10^7$. Due to numerical considerations, discussed further in section 7.4, we choose a lower $\mathrm{Ra} = 10^4$, with the corresponding $\mathrm{Pr} = 1$, as these are easier to simulate and provide results similar to higher values of Ra as shown in [9]. This gives us the same parameters in diffusion and viscosity that is mapped out in chapter 5. We have shown that perturbations with these parameters will develop mushroom-like shapes.

The background particle density is unknown, which motivates a scan of a given range for the initial perturbation amplitude parameter $\eta = \Delta n / n_0$ in a range from 3 to 50. A parameter scan for higher amplitudes would also be desirable, although this gives numerical limitations due to high computational cost as described in section 7.4.

On the inner and outer boundary conditions in the radial direction for the particle density $n$, vorticity $\Omega$ and the electrostatic potential $\phi$ are all set to the Dirichlet conditions, assuming the chamber walls of the box to be electrically grounded:

$$\begin{aligned} n(R = R_{\mathrm{min}}) = n(R = R_{\mathrm{max}}) &= n_0 \\ \Omega(R = R_{\mathrm{min}}) = \Omega(R = R_{\mathrm{max}}) &= 0 \\ \phi(R = R_{\mathrm{min}}) = \phi(R = R_{\mathrm{max}}) &= 0 \end{aligned}$$

The initial conditions are taken from a system initially at rest, and are specified as

$$\begin{aligned} n(t = 0) &= n_0 + \widetilde{n} \\ \Omega(t = 0) &= 0 \\ \phi(t = 0) &= 0 \end{aligned}$$

where $n_0 = 1$ is the unknown reference background particle density and $\widetilde{n}$ takes the blob-like shape around the initial position $\mathbf{r}_0$, given by

$$\widetilde{n} = \frac{\Delta n}{n_0} \exp\left[-\frac{(\mathbf{r} - \mathbf{r}_0)^2}{2}\right]$$

The spatial resolution for these simulations are chosen as $N_x = N_y = 1024$ and the time step size to $\Delta t = 10^{-3}$.

## 7.2 On the accuracy of simulations

Positioning the initial blob perturbation at the innermost chamber wall in our simulations can give rise to inaccuracies, as we force the Dirichlet boundary condition $n = 0$ on the boundaries. With a high relative amplitude of the blob, strong gradients are formed from the boundary to the center of the perturbation. The accuracy of the solver depends on the chosen spatial and temporal resolution.

With precalculation of the LU factorisation, the complexity of the algorithm is $\mathcal{O}(n^3)$, easily raising the computational cost above practical limitations. In addition, an increased spatial resolution also dictates higher requirements on the temporal resolution through the CFL condition described in section 5.8.

With this in mind, we seek to quantify the errors in a range of temporal resolutions with the spatial resolution set to $L_R = L_Z = 1024$, as the computational time scales linearly with the temporal resolution as it merely increases the number of time steps to reach a given time.

Simulations are performed with the temporal resolution $\Delta_t$ ranging from $10^{-2}$ to $10^{-4}$ separated by a factor of 5, and the relative amplitude of the perturbation to $\Delta n/n_0 = 16$, otherwise default parameters are as described in Section 7.1.

Figure 30 shows a particle density profile cut of $n(R, Z)$ at $t = 5$ and $Z = 0$ for $\Delta_t = 10^{-4}$ as well as the relative difference between this profile and the profiles at $\Delta_t = 10^{-3}$ and $\Delta_t = 5 \times 10^{-4}$. We see that the maximum relative difference between the higher resolution ($\Delta_t = 10^{-4}$) and the lower ($\Delta_t = 10^{-3}$) in the perturbation amplitude here is 0.04. The lower resolutions ($\Delta_t = 10^{-2}$, $\Delta_t = 5 \times 10^{-3}$) are not shown here but their relative difference are much higher. We also see that the solutions converge as the temporal resolution increases.

Figure 31 shows the relative difference between a reference $\Delta_t = 10^{-3}$ and $\Delta_t = 5 \times 10^{-4}$, $\Delta_t = 10^{-4}$ for the maximum velocity and the maximum center of mass velocity. While the relative difference here is smaller than for the particle density profile, we see that for a higher temporal resolution the initial velocity as well as center of mass velocity is lower than for lower resolutions. This is due to boundary effects where the particle density is forced to zero. We see again that the increased temporal resolution cause the solutions to converge.

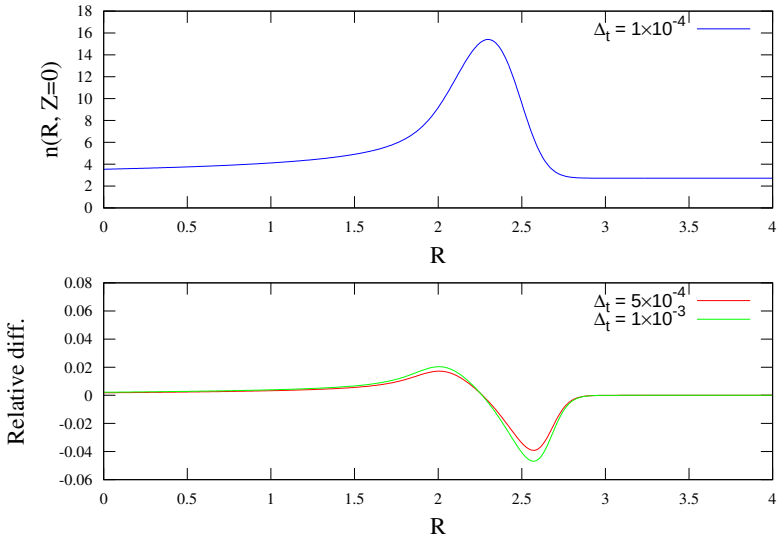


Figure 30: perturbation amplitude profile $n$ at $Z = 0$ for $\Delta_t = 10^{-4}$ at time $t = 5$ (top) and the relative difference between $\Delta_t = 10^{-4}$ and the same profiles for $\Delta_t = 10^{-3}$ and $\Delta_t = 5 \times 10^{-4}$ (bottom).

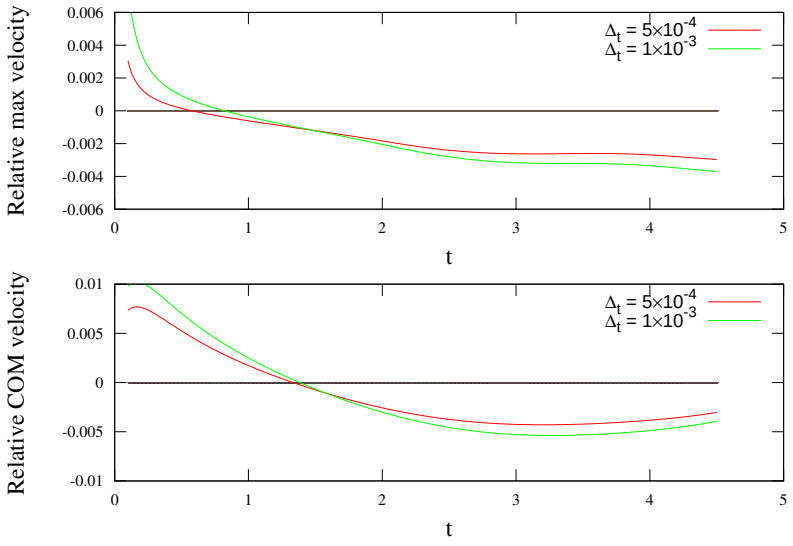


Figure 31: Relative difference in maximum velocity between $\Delta_t = 10^{-3}$,$\Delta_t = 5 \times 10^{-4}$ and a reference $\Delta_t = 10^{-4}$ as a function of time (top) difference in maximum center of mass velocity between $\Delta_t = 10^{-3}$,$\Delta_t = 5 \times 10^{-4}$ and a reference $\Delta_t = 10^{-4}$ as a function of time (bottom).

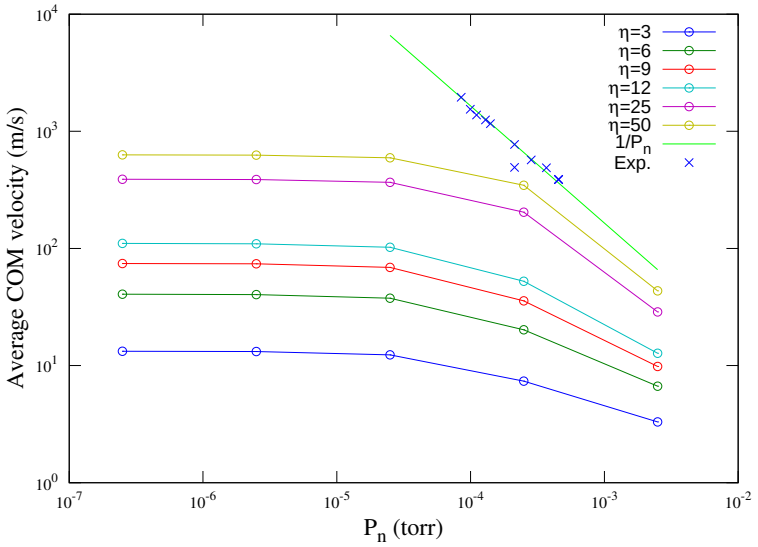


Figure 32: Average center of mass velocity for varying neutral pressure $P_{\mathrm{n}}$ and perturbation amplitude $\eta = \Delta n/n_0$. The experimental results are also plotted along with a fitted line proportional to $1/P_{\mathrm{n}}$. We see the dependence of perturbation amplitude for the center of mass velocity as well as its dependence on the slope in the collisionally dominated regime.

Accounting for these relative differences, the computational cost of choosing the lower range of resolutions for our simulations is considered a practical solution. Based on this, we consider the choice of $\Delta_t = 10^{-3}$ a good balance between computational cost and accuracy.

## 7.3 Simulation results

Figure 32 shows the average center of mass velocity as a function of the neutral gas pressure for the simulations at relative perturbation amplitudes $\eta = \Delta n/n_0$ between 3 and 50. We see the "bend" on the curve for the average center of mass velocities at approximately $10^{-4}$ torr, and a function line going as $1/P_{\mathrm{n}}$ has been drawn to use as a comparison for the declining slopes for neutral pressures above $10^{-4}$ torr.

We see the clear dependence on perturbation amplitude for the center of mass velocity as well as its dependence on the neutral gas pressure $P_{\mathrm{n}}$. Although the perturbation amplitudes are not high enough to reach the average center of mass velocities in the experiments, we note that the fitted line matches the slope of the simulations for higher perturbation amplitudes.

## 7.4 Numerical issues

Simulating higher relative perturbation amplitudes poses a challenge. For our standard spatial resolution of $1024 \times 1024$, amplitudes above 50 tend to quickly become numerically unstable and within a few time steps give infinite values. This can partially be remedied by increasing the temporal resolution, but only to a certain extent. However, increasing the spatial resolution has a large computational cost attached to it as the algorithm, by solving $n$ pre-factorised LU-factorisations, scales as $\mathcal{O}(n^3)$. The increased spatial resolution also requires a higher temporal resolution by the CFL requirement (section 5.8), further increasing the computational time.

The increased perturbation amplitude in our case also has a problem with breaking the initially assumed periodic boundary conditions in the vertical direction. As we use a gaussian to represent our plasma blob, for high enough amplitudes the derivative of particle density on the vertical boundaries is discontinuous. The solution to this is to either use another shape of with stronger gradients as the initial perturbation so that the particle density goes to zero in the vertical direction or to introduce finite differences in two dimensions allowing us to set non-periodic boundary conditions in the vertical direction. The problems with using stronger gradients in the initial perturbation is that the spatial resolution has to be increased to acomodate for this, further increasing the computational cost. On the other hand a bi-directional finite difference method would, using prefactorisation into a LU matrix problem, be an algorithm of $\mathcal{O}(n^4)$ as the single matrix would be $n^2 \times n^2$ in size.

Although the simulations show a similar bend and ramping of the maximum and average center of mass velocities, we have in our simulations not been able to reach the center of mass velocities described experimentally. This is due to the numerical challenge of simulating perturbations with a very high relative blob amplitude. If the assumption of a near zero background plasma is to hold, the relative blob amplitude must be very high to replicate the same velocities. This is not without numerical issues: A high relative blob amplitude will generate strong gradients on the particle density, which dictates higher requirements on the spatial and temporal resolution to keep numerical stability, as described in section 7.4. Using the observation from section 16 where we see that the blob velocity is proportional to the relative blob amplitude we can conclude that the required relative amplitude to reach velocities comparable to the measured velocities is around 500.

High values for the Rayleigh number also pose problems as they require higher spatial and temporal resolutions. Increasing the Rayleigh number from $10^4$ to $10^6$ requires a doubling the resolution and consequently increasing the temporal resolution. However, for $\mathrm{Ra} = 10^4$ we are nearing the ideal diffusion-less regime for $\nu_\mathrm{i} = 0$, with the evolution of the center of mass velocities being very similar to the evolutions of center of mass velocities for higher Rayleigh numbers, as demonstrated in the parameter study performed in [9]. Note that for larger

Rayleigh numbers we would stay in the ideal regime for higher values of $\nu_i$, thus larger Ra should in principle be explored.

# 8 Discussion

We have in this thesis seen that the $\nabla B$ and curvature drifts give rise to charge polarisation on perturbations on a background plasma, further resulting in a vertical electric field and thus electric drift causing the perturbation to propagate across the magnetic field lines. The dynamic balancing of charge by the polarisation drifts further cause vortices to be created on these perturbations as described in 2.1.

For small perturbation amplitudes, we have an ideal scaling for the velocity $u_{\mathrm{c}}$ as

$$\frac{u_{\mathrm{c}}}{c_{\mathrm{s}}} = \left(\frac{2l}{R}\frac{\Delta n}{n_0}\right)^{\frac{1}{2}}$$

where $c_{\mathrm{s}}$ is the sound speed, $l$ is the blob size, $R$ is the radius and $\Delta n/n_0$ is the relative blob amplitude. This gives that the blob velocity increases with blob size and the perturbation amplitude, as described in section 2.2.6. For larger perturbation amplitudes, this relation takes instead the form

$$\frac{u_{\mathrm{c}}}{c_{\mathrm{s}}} = \left(\frac{2l}{R}\right)^{\frac{1}{2}}$$

where the perturbation amplitude does not enter the scaling. This introduces the perturbation amplitude as an additional parameter that affects the velocities in the simulations.

The experiments presented in chapter 6, demonstrating the propagation of blobs perpendicular to the magnetic field lines in a toroidal chamber, show that the center of mass velocity for a propagating blob is inversely proportional to the neutral pressure $P_{\mathrm{n}}$

$$u_{\mathrm{COM}} \sim \frac{1}{P_{\mathrm{n}}}$$

Indeed, from the model equations we have seen, in section 2.2.7, that the dependence of ion-neutral collisions for the velocity (for regimes of high ion-neutral collision frequencies) can be described as inversely proportional:

$$u_{\mathrm{c}} \sim \frac{g}{\nu_{\mathrm{in}}}$$

From this we have motivated the simulation of propagating perturbations, or "blobs", for a range of collisional frequencies (proportional to the neutral gas pressure) and perturbation amplitudes. We will now turn to comparing the VTF experiments with our simulated experiments more in-depth.

## 8.1 Comparison with VTF blob experiments

Since the outward particle flux is proportional to the center of mass velocity, as described in section 2.2.8, quantifying the general transport of plasma can

be achieved by measuring the center of mass velocity. This velocity is demonstrated to be dependent on the Rayleigh and Prandtl numbers [9] as well as the perturbation amplitude and ion-neutral collision rate as described in section 5.6. We will here discuss the effects of these last two parameters. It is worth mentioning that this velocity also depends on the domain size, although this is fixed in the experiments.

As a result of the experiments in [13] it is stated that the average center of mass velocity for a perturbation propagating across the magnetic field is scaling inversely with the neutral gas pressure: $u_{\mathrm{COM}} \sim 1/P_{\mathrm{n}}$.

In section 5.4 we show how the ion-neutral collision rates affects the maximum center of mass velocity: A higher ion-neutral collision rate $\nu_{\mathrm{i}}$ decreases the velocity $u_{\mathrm{COM}}$. Figure 11 shows the maximum center of mass velocity as a function of the ion-neutral collision rate in a logarithmic plot. With a "bend" around $\nu_{\mathrm{i}} = 10^{-1}$, we see that in the inertia dominated regime (for lower values of $\nu_{\mathrm{i}}$) the maximum center of mass velocity is near constant and thus not affected by the ion-neutral collision rate. However, for the collisionally dominated regime (values of $\nu_{\mathrm{i}} > 10^{-1}$) we see the maximum center of mass velocities decreasing, very similar to the measured $u_{\mathrm{COM}} \sim 1/P_{\mathrm{n}}$ (since $P_{\mathrm{n}}$ is proportional to $\nu_{\mathrm{i}}$).

In [16] an expression for the propagation velocities $u$ of the depletion bubbles in ESF phenomenas was presented, here given for reference:

$$u = \frac{1}{8} R_{\mathrm{dep}} \left[ \left( \nu_{\mathrm{in}}^2 + \frac{16 g_{\mathrm{E}}}{R_{\mathrm{dep}}} \right)^{\frac{1}{2}} - \nu_{\mathrm{in}} \right] \tag{40}$$

where $R_{\mathrm{dep}}$ is the radius of the depletion front, $\nu_{\mathrm{in}}$ is the ion-neutral collision rate and $g_{\mathrm{E}}$ is the Earth's gravity. As the interchange mechanisms driving these phenomenas are the same as used in the 2-field model, the close fit between this expression and the simulated results given in 11 is of high interest. The fitted curve is for $R_{\mathrm{dep}} = 11.5$ and $g_{\mathrm{E}} = 0.24$. We remind the reader that the plasma perturbations show varying degrees of mushroom-like shapes, which may serve as analogues to the compressed depletion bubble fronts and thus gives us an $R_{\mathrm{dep}}$ much larger than the actual blob (or bubble) size.

However, although the expression gives a qualitative match, the methods used in [16] might not be applicable for plasma perturbations as several of the assumptions break with the dynamics observed in the simulating perturbations such as assuming the stream function (and consequently the potential) to be zero on the edges of the perturbation. As observed, this is indeed not the case for the perturbation dynamics we have simulated where charge build up on both sides of the perturbation, driving the electric drift.

Nevertheless, it serves a good match with respect to demonstrating how the velocity of perturbations are divided into inertia dominated and collisionally dominated regimes characteristic for diffusion processes. Indeed, we can easily see that for the collisionless limit $\nu_{\mathrm{in}} \to 0$ Eq. (40) gives the constant solution

$u = (R_{\mathrm{dep}} g)^{1/2}/2$ while for the collisionally dominant limit $\nu_{\mathrm{in}} \gg 1$ we have the inversely proportional solution $u = g/\nu_{\mathrm{in}}$ in accordance to the simulated results.

This is further shown in section 5.6 where simulations varying $\nu_{\mathrm{i}}$ and the relative blob amplitude showing that this bend from $\nu_{\mathrm{i}} = 10^{-1}$ to $\nu_{\mathrm{i}} = 1$ remains for the different simulated relative blob amplitude variations in the range between from 3 to 50. This is an indication that this "bend" separating the collisionally and inertia dominated regimes persist also for higher relative amplitudes and its relation to $\nu_{\mathrm{i}}$ is independent on the relative blob amplitude.

In section 7.3 where we have simulated conditions close to the experiments presented in [13], we see in figure 32 the variation of maximum and average center of mass velocities with $\nu_{\mathrm{i}}$ and the relative blob amplitude. We see in this case the same indications of a "bend" around the neutral pressure $P_{\mathrm{n}} \approx 10^{-4}$ torr (corresponding to $\nu_{\mathrm{i}} \approx 4 \times 10^{-1}$), while for higher neutral pressures ($P_{\mathrm{n}} > 10^{-4}$) the center of mass velocity declines as the neutral pressure increases.

Let us now address the proposed dependency $u_{\mathrm{COM}} \sim 1/P_{\mathrm{n}}$. For higher relative blob amplitudes ($\eta = 25$ and $\eta = 50$) in the range from $P_{\mathrm{n}} = 4 \times 10^{-4}$ torr to $P_{\mathrm{n}} = 4 \times 10^{-3}$ torr the slope of the declining average center of mass velocity $u_{\mathrm{COM}}$ does indeed fit the description of being proportional to $1/P_{\mathrm{n}}$. For lower relative blob amplitudes this is not the case, as can be easily seen in figure 32.

For lower neutral pressures $u_{\mathrm{COM}}$ reaches the "ramp" at a maximum velocity in the inertia dominated regime. This ramp is suggested in [13] as the sound speed $c_{\mathrm{s}}$, although for our simulations this ramp value is clearly dependent on the relative blob amplitude $\eta$. While the sound speed $c_{\mathrm{s}}$ is the same for all relative blob amplitude simulations, the maximum velocity is not. This does not exclude that the sound speed may be an upper bound on $u_{\mathrm{COM}}$ for higher perturbation amplitudes (indeed, through the scaling $\gamma \sim c_{\mathrm{s}}$ it follows that the a higher sound speed in our model parameters should give higher velocities) although it is not the governing restriction when simulating lower perturbation amplitudes.

We will now turn to discussing the disparity between the experimental and simulated results shown in figure 32 where we have the average center of mass velocity for the perturbation amplitudes $\eta = 25$ and $\eta = 50$ plotted against the experimental average center of mass velocities. We notice that the experimental results are not perfectly matched to the simulations but is slightly shifted, both upwards on the velocity scale and left on the collision scale. The functional dependence is very similar. This shift can be explained by inaccuracies in parameters we will map out in the following two paragraphs.

The velocity scale is dependent on the sound speed as well as the scaling parameter between blob amplitude and the major radius as we have $u_{\mathrm{c}}/c_{\mathrm{s}} \sim (2l/R)^{1/2}$ (Eq. (15)), which is further dependent on the electron temperature. This indicates that we would get higher $u_{\mathrm{COM}}$ for higher electron temperatures (and consequently $c_{\mathrm{s}}$) as well as larger blob sizes $l$. Inaccuracies in both these parameters will therefore yield a different velocity scaling. Indeed, we have here

assumed a constant electron temperature although in [13] it is stated that the electron temperatures in these experiments vary in the range between $1-3$ eV. We have also shown that $u_{\mathrm{COM}}$ depends on the initial perturbation amplitude $\eta = \Delta n/n_0$, although increasing the amplitude in our simulations yields numerical challenges as described in section 7.4.

As for the scaling along the ion-neutral collision scale, the scale is primarily dependent on the ion temperature, neutral gas temperature (relating the neutral pressure to neutral density as well as $\nu_{\mathrm{i}}$) and the sound speed $c_{\mathrm{s}}$ as described in section 7.1. An increased ion temperature would shift the simulated velocity curves in 32 to the left towards lower $\nu_{\mathrm{i}}$, while a change in electron temperature (and consequently the sound speed) would shift the velocity curves to the right. It is worth noting that the neutral gas temperature is not specified in the experiments although assumed to hold a room temperature. There is also a large experimental uncertainty for the ion temperature. In addition several dominant collisional cross sections are possible such as elastic collisions or charge exchange.

In the simulations the Rayleigh number $\mathrm{Ra} = 10^4$ was used since higher values are harder to simulate without encountering numerical instabilities. According to [9], the center of mass velocity evolution for $\mathrm{Ra} > 10^4$ are very similar to $\mathrm{Ra} = 10^4$ and thus well suited as an approximated parameter for our simulations to account for the numerical challenges of higher values of Ra, as described in 7.4. However, an increased Ra would likely increase the velocity for higher $\nu_{\mathrm{i}}$ until the ideal regime is recovered.

Summarising the points of importance, we have seen that the simulated results give propagation speeds in two regimes: One inertia dominated and one collisionally dominated where the average center of mass velocity scales as $1/P_{\mathrm{n}}$. These regimes are independent of the perturbation amplitude. Further we have seen the qualitative agreement with the experimental results and mapped out which parameters that can be used to provide a closer match between simulations and experiments.

## 8.2 The "mushroom" shapes

In [13] the propagating blobs are reported to take on "mushroom" shapes similar to the shapes described in simulations presented in [9]. The dipole vorticity field, as a consequence of the dipole electrostatic potential field, gives rise to the mechanism that drives the center of the perturbation faster than the top and bottom edges, as well as curling the edges backward. This has been demonstrated for a range of varying Rayleigh and Prandtl numbers in the simulations.

For our chosen parameters of $\mathrm{Ra} = 10^4$ and $\mathrm{Pr} = 1$, the perturbation takes the form previously illustrated in figure 6, a definite mushroom-like shape with vortices at the top and bottom. For a higher Prandtl number $\mathrm{Pr} = 4\times10^2$ where

the viscosity is dominating over the diffusion we see, in figure 7, a mushroom shape without the vortices and less dispersion.

On the other hand we are aware of effects that will reduce this mushroom form: In figure 8 we see a simulation for where the diffusion dominates over viscousity, resulting in a more circular shape. The inclusion of ion-neutral collisions also reduces this shape, as described in section 5.4, where a high neutral pressure leads to a high ion-neutral collision frequency which in turn leads to a more coherent blob-structure.

Through inspection of the data sets from the experiments performed in [13], it is apparent that this mushroom shape is not characteristic for the majority of the experiments. Only for the three experiments in the neutral pressure range $2.13\times10^{-4}$ to $2.84\times10^{-4}$ torr is an actual mushroom shape observed. The other experiments, either below or above this neutral pressure range, exhibit no such apparent behavior. For the lower pressure range $0.412 \times 10^{-4}$ to $1.41 \times 10^{-4}$ torr circular or radially elongated dispersed structures are observed, while for the high pressure range $3.69 \times 10^{-4}$ to $4.54 \times 10^{-4}$ torr multiple circular blobs are observed.

Noting that an increased ion-neutral collision rate should reduce this mushroom-like shape, it is expected to be more prominent in the lower neutral gas ranges. This does not agree with experimental measurements.

Examining the experiments for where the perturbation assumes a mushroom-like shape in a higher temporal sampling (figure 27, for neutral pressure $n_{\mathrm{g}} = 2.13 \times 10^{-4}$) we note that in the early creation of the perturbation there is indeed several "blobs" created: One ahead of the main perturbation and one down to the left. As the amplitude of the main perturbation increases it clearly dominates the overall structure. However, the remnants from the lower left blob is present, lagging behind the main structure and possibly contributing to the observed mushroom-like shape.

Along with the lack of mushroom shapes at lower neutral pressures this suggests that, while it still may be a contributing mechanism, the mushroom shapes observed might not originate from the same dynamics in the two-field model.

# 9 Conclusion

We have in this thesis explored the propagation of coherent structures in magnetised plasmas, so-called "blobs", in toroidal chambers such as presented in the VTF experiments [13].

The relation between the propagation velocity and the perturbation amplitude as well as the neutral gas pressure has been explored through simulations. We have seen that the velocity increases near proportionally with the perturbation amplitude for large amplitudes, while the dependency on the neutral gas pressure separates into two regimes: One inertia dominated and one collisionally dominated where the average center of mass velocity scales as $1/P_{\mathrm{n}}$. These regimes have been shown to be independent of the perturbation amplitude.

Furthermore, results from the experiments at the VTF has been reviewed. The average velocities measured in these experiments has been shown to qualitatively match simulations performed to match these experiments. Experimental parameters affecting the the match between simulated results and experimental results have also been mapped out.

In addition, the formation of mushroom-like structures observed in [13] has been found not to be primarily generated by the dipole vorticity field around the perturbation. This is supported by observations made on variation of parameters in simulations and their effect on the creation of mushroom-like structures.

## 9.1 Future work

We propose further work by performing simulations on the the perturbation amplitudes and the required higher Rayleigh number to represent the experiment more acurately. The perturbation should if so be modelled to fit the experimental perturbation profile more acurately.

It is also desirable to further develop the simulation code to be able to account for fixed boundary conditions in the vertical direction in addition to the current periodic condition to accommodate for the higher perturbation amplitudes needed. Doing this requires the development of a two dimensional finite difference solver.

In both these future work cases mentioned above, it is greatly encouraged to implement a more versatile parallelism, like MPI, in the simulation code to take advantage of being executed on several nodes on high performance computing systems such as Stallo. Although the current code is parallelised using OpenMP, this is restricted to using the processors or cores on a single node only.

We also propose that the observed separation of inertia and collisionally dominated regimes independent of perturbation amplitudes should be tested experimentally as well as the proportional scaling for perturbation velocity to perturbation amplitude for high amplitudes.

# References

[1] *HDF5 – http://www.hdfgroup.org/HDF5/*, 2010.

[2] *MPI – http://www.mpi-forum.org/*, 2010.

[3] *OpenMP – http://openmp.org/wp/resources/*, 2010.

[4] *Stallo hardware – http://www.notur.no/hardware/stallo/*, 2010.

[5] Akio Arakawa. Computational design for long-term numerical integration of the equations of fluid motion: Two-dimensional incompressible flow. part i. *Journal of Computational Physics*, 1(1):119 – 143, 1966.

[6] Francis F. Chen. *Plasma physics and controlled fusion, Volume 1: Plasma physics*. Springer, 2006.

[7] W. Fundamenski, O.E. Garcia, V. Naulin, R.A. Pitts, A.H. Nielsen, J. Juul Rasmussen, J. Horacek, J.P. Graves, and JET EFDA contributors. Dissipative processes in interchange driven scrape-off layer turbulence. *Nuclear Fusion*, 47(5):417, 2007.

[8] O. E. Garcia. Two-field transport models for magnetized plasmas. *Journal of Plasma Physics*, 65(02):81–96, 2001.

[9] O. E. Garcia, N. H. Bian, and W. Fundamenski. Radial interchange motions of plasma filaments. *Physics of Plasmas*, 13(8):082309, 2006.

[10] Martien A. Hulsen. Stability of the implicit/explicit extension of the stiffly-stable schemes of gear. *MEAH-138*, 1996.

[11] D. L. Hysell. An overview and synthesis of plasma irregularities in equatorial spread f. *Journal of Atmospheric and Solar-Terrestrial Physics*, 62(12):1037 – 1056, 2000.

[12] George Em Karniadakis, Moshe Israeli, and Steven A Orszag. High-order splitting methods for the incompressible navier-stokes equations. *Journal of Computational Physics*, 97(2):414 – 443, 1991.

[13] Noam Katz, Jan Egedal, Will Fox, Ari Le, and Miklos Porkolab. Experiments on the propagation of plasma filaments. *Phys. Rev. Lett.*, 101(1):015003, Jul 2008.

[14] Volker Naulin and Anders H. Nielsen. Accuracy of spectral and finite difference schemes in 2d advection problems. *SIAM J. Sci. Comput.*, 25(1):104–126, 2003.

[15] Sidney L Ossakow. Spread-f theories–a review. *Journal of Atmospheric and Terrestrial Physics*, 43(5-6):437 – 452, 1981. Equatorial Aeronomy - I.

[16] E. Ott. Theory of rayleigh-taylor bubbles in the equatorial ionosphere. *Journal of Geophysical Research*, 83:2066–2070, May 1978.

[17] Timothy Sauer. *Numerical Analysis*. Pearson Addison-Wesley, 2006.

[18] Walter A. Strauss. *Partial Differential Equations*. Wiley, 2008.

# List of Figures

# List of Tables