Fast Ion Confinement in The Reversed-Field Pinch

by

Benjamin F. Hudson II

A dissertation submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

(Physics)

at the

University of Wisconsin – Madison

2006

Fast Ion Confinement in The Reversed-Field Pinch

Benjamin F. Hudson II

Under the supervision of Professor Stewart C. Prager

At the University of Wisconsin – Madison

The physics questions this thesis addresses are: Are fast ions confined in a stochastic magnetic field? Is their confinement different from the bulk plasma and why? What is the impact of varying levels of magnetic stochasticity on fast ion confinement? How does the presence of a fast ion population affect the bulk plasma?

Performing Neutral Beam Injection (NBI) with deuterium creates a population of 20 keV ions in our deuterium plasma. These fast ions collide and fuse with the background at a rate far greater than the background reaction rate. The resultant neutron flux is detected and provides a method of estimating the confinement of the fast ion population. The observed tens of milliseconds confinement time show that the fast ions are not being lost stochastically.

Extensive simulation was done to understand both the character of magnetic stochasticity in Madison Symmetric Torus (MST) and also the confinement of fast ions in a stochastic magnetic field. Large subsets of ion orbits were found to be not stochastic.

Analytically, the mechanism for the confinement of fast ions was found to be that the guiding center drifts, which can be substantial, decouple the guiding center motion from the background stochastic field and result in good confinement.

In addition, under certain plasma conditions, NBI current drive was measured and found to agree with analytical predictions. One effect of the current drive was to alter the plasma equilibrium magnetic field such that an internally resonant tearing mode became non-resonant. This beginning work illustrates the potential for NBI current drive in the Reversed-Field Pinch (RFP) as well as to study other aspects of plasma physics.

Acknowledgements

There is no question that my time in graduate school here has been the most exciting and rewarding of my life. The opportunity to work and learn with the wonderful and brilliant people of the MST group, and also the other plasma groups on campus, have helped me grow as a person and as a new scientist.

The greatest debt of gratitude that I owe is of course to my mentor, Gennady Fiksel. Gennady encouraged the utmost independence, while always being there to provide guidance. In addition to the physics knowledge and other assistance that Gennady passed on to me, I recognized early on that one of the ways to better myself as a scientist would be to not just learn what Gennady had to teach, but to emulate and understand the keen and clear way in which he thinks about problems. To the extent that I've been successful at this, the rewards in my work have been apparent, and it is something I hope to continue.

I was fortunate enough to come to a program that had Prof. Stewart Prager as a leader, and have him as my academic advisor. He is one of the most open, patient and fair people here, and the success of the MST group under his stewardship reflects that.

Many others deserve more appreciation than I can offer here; my good friends Mike Davis, Fatima Ebrahimi, Yesenia Pumarada-Cruz, Mirella Cengher, for being loyal friends and making me laugh even when things were tough. My office-mates Adam Bayliss and Dave Albers, who showed me it is possible to do physics even over loud discussions about dogs with gangsta rap playing in the background. Long live the "MST Fencing Crew"; Max Wyman, Art Blair and myself. Other scientists and friends that I've

had the pleasure to laugh with and learn from, Jay Anderson, Ted Biewer, Rob O'Connell, Diane Demmers, John Sarff and John Wright.

I also want to thank Bill Zimmerman, without whom sometimes I believe MST would come to a grinding halt, a true teacher in the best sense of the word, and Peter Weix, whose job as safety officer is performed diligently and helped me to be more responsible, as well as physically intact at the end of my studies here.

Finally I want to dedicate my thesis to my mother, Mary, without whose love and support over the years I couldn't have begun to achieve what I have. She is the strongest person that I know, and I try to emulate her perseverance in all that I do.

To all of the above, and to the countless others deserving of thanks and recognition, I can only once again say thank you, and that I hope that life returns to you in good measure the kindness and gifts that you have given me.

Benjamin F. Hudson II

Table of Contents

List of Tables

List Of Figures

1 Introduction

The physics questions that motivate this work are: Are the fast ions from neutral beam injection confined in the reversed field pinch (RFP) magnetic field? Is their confinement different from the bulk plasma and why? What is the impact of varying levels of magnetic stochasticity on fast ion confinement? How does the presence of a fast ion population affect the bulk plasma?

While these are important goals for their own sakes, taken together they also address the question of feasibility of neutral beam injection (NBI) in the RFP. This is as yet a quite open field of investigation and with the importance of neutral beams to tokamak physics and reactor scenarios; the issue of confinement of NBI ions is key.

1.1 Using Neutral Beam Injection to Study Fast Ion Physics

A brief review of selected work in the study of fast ion confinement is appropriate to present the context in which the work contained in this thesis was performed. Some of the early theoretical work examined the role of particle drifts and finite Larmor radius effects for fast ions[1,2] and they were found to usually improve their confinement relative to the bulk plasma.

Neutral beam injection, in addition to bootstrap current, is a source of non-inductive current drive that was successfully measured[3] in DIII-D. While the two sources for a time were indistinguishable, later experiments[4] were able to isolate the fast ion driven current profile. The dependence of the NBI current profile on tearing modes was

also measured and the loss mechanism explained by a resonant interaction of the fast ion orbits and the magnetic perturbations.

The confinement of fast ions, in this case alpha particles that resulted from NBI, was studied extensively in TFTR.[5] They were found to be confined even better than the bulk plasma over most of the plasma radius, but did have a layer of stochasticity where the ripple from the toroidal field coils would resonate with the particles motion.

In NSTX, the confinement of fast ions from NBI was studied[6] for both short pulses of NBI, as we have on MST, and for a long injection period. They employed three diagnostics; neutron detection, Faraday cup fast ion edge probe, and a neutral particle analyzer. They found that the neutron flux was consistent with classical slowing down, and that confinement was good. During sawtooth crashes, the neutron flux dropped rapidly, attributed to a loss of fast ions. Fast ion confinement during sawtooth events is also considered in this thesis.

Also in the spherical tokamak a theoretical study was done[7] that showed that fast ion orbits can be stochastic despite the underlying field having well-ordered flux surfaces. This orbit stochasticity was due to a resonant interaction between bounce motion and gyro-motion. This is one example of the ion orbit having a stark difference from the background field trajectories and the small gyro-radius plasma ions as well. This issue is central to this thesis, but explored in a stochastic magnetic field as opposed to an ordered one.

Experiments performed[8] in the gas dynamic trap (GDT) found that the fast ion energy was also determined solely by classical slowing down, and that despite the

presence of micro-instabilities and the possibility of anomalous cross-field transport and pitch angle scattering into the loss cone, none of these losses were observed.

Other experiments, mostly in tokamaks, have shown sawtooth stabilization,[9] driving of Alfvén waves[10] and driving of instabilities (bump on tail).[11] For performing these physics experiments, understanding fast ion confinement is important and good fast ion confinement is helpful if one is interested in more than transient effects.

In the reversed field pinch, we wish to add to the body of knowledge, but expand it to understand the effect of a stochastic magnetic field on fast ion confinement.

The basic process of neutral beam injection is as follows. We begin with a plasma source, held at a high potential. Ions are drawn from the plasma and accelerated by electrically biased grids at a lower potential than the plasma source. The shapes of these grids also serve to focus the ion current into a beam. The ion beam becomes a neutral beam by passing through a neutral gas that provides the beam ions with electrons. The neutral beam then enters the plasma, unaffected by the strong magnetic fields until it again becomes ionized by losing electrons through atomic processes or collisions. Through Coulomb collisions the fast ion then slows down and transfers energy to the background plasma electrons and ions. Fast ions can also be lost by interacting with background neutrals, becoming neutral themselves, which can result in them reaching the device wall or being re-ionized in a different part of the plasma.

There are many things to study about fast ions; this thesis is primarily concerned with confinement. In addition, neutral beams are an intense source of directed momentum, and this could affect the plasma current and fluid flow. Also, the high

energies of the fast ions from NBI are a source of heat to raise the plasmas temperature, which is key to fusion reactor scenarios.

1.2 Importance of Fast Ion Confinement

If NBI is a source of fast particles that are responsible for the many effects listed above, then it stands to reason that good confinement is necessary to have anything other than a transient and minimal effect. The most obvious way to characterize the role of confinement is to compare the timescales of the effects we wish to accomplish using NBI. Using NBI to heat a plasma, one compares the energy confinement time of the bulk plasma to the fast ion energy loss time. This will be discussed in more detail in Section 2.2, 3.2 and 7.5 but here we can state that the energy confinement time for a typical MST plasma is ~1 ms., whereas the energy loss time of NBI ions (E_{fi} = 20 keV, T_e = 400 eV, n_e = $1x10^{19}$ m^{-3}) is around 17 ms. This means the energy that the fast ions can transfer will leave much faster than it is put in. When MST runs in what we call PPCD (Pulsed Parallel Current Drive),[12] where the plasma current has been externally shaped, magnetic fluctuations are reduced a factor of 3 or more. Consequently, the energy confinement time goes up a factor of ten (~10 ms) and the electron temperature increases to around 1 keV. At this electron temperature the fast ion energy loss time is around 50 ms. This would give ~7% of the fast ions energy to the plasma in one energy confinement time, which is a small, but non-negligible fraction.

Next we can consider the issue of particle confinement. If the ions escape quickly then they won't have time to deposit energy. The particle confinement time of electrons and ions is also around 1 ms in MST. The reason for this is that the magnetic fields in

MST are believed to be stochastic, so particle and energy diffusion are coupled strongly, and more importantly, velocity dependent. For stochastically streaming fast ions, with large parallel velocities, the confinement time would be substantially shorter. The particle diffusion for fast ions (that have about 100x the energy of plasma ions) would be 10x faster than the bulk ions.

This question of fast ion confinement in the RFP has not been addressed, and is the main topic of this thesis. What is the fast ion confinement? If different than expected, why is it different? How does it depend on fluctuations? How does it depend on injection methods and other plasma parameters?

1.3 Overview of Thesis

In Chapter 2 we will discuss magnetic stochasticity, which needs to be understood before even attempting to quantify the behavior of fast ions in the RFP. The chapter begins with the results of prior work that determined the particle confinement time in MST. It is then followed with an overview of the magnetic island structure in the RFP and how it leads to stochasticity, which explains the measured values of the particle confinement times in the MST core.

Chapter 3 describes modeling of the stochastic magnetic field for the express reason of providing an accurate structure to do particle tracing in, but the analysis allows other useful results such as an estimate of the magnetic diffusion coefficient and electron heat transport.

With a solid understanding of the test-bed for fast ion confinement studies, we turn to Chapter 4, which describes the experiment performed to measure the fast ion

confinement time. An overview of the neutral beam is presented where we discuss the components of the injector and the sequencing of an injection pulse. Independent measurements of key beam parameters, ion focusing, and neutral current into MST are discussed, as well as a description of the diagnostics and techniques used. The chapter concludes with the experiment performed, the construction of a neutron detector (whose purpose will be explained), the results from the experiment, and an estimate of the confinement time of NBI fast ions. We then go on to discuss experimental results for other initial conditions of fast ions, including low fluctuations (PPCD), high fluctuations (sawteeth), injection opposite (counter) to the plasma current and radial injection.

We return to simulation in Chapter 5 in order to understand the observed experimental results detailed in Chapter 4. To do this we perform simulation of fast ion trajectories in a stochastic magnetic field, and insofar as possible attempt to recreate the conditions of NBI. Differences between particle trajectories and magnetic field line trajectories are discussed prior to presenting the simulation results. The confinement of fast ions is characterized in a variety of injection scenarios and plasma conditions.

Chapter 6 discusses experimental results of looking at the effects of our neutral beam on other plasma features, in particular tearing mode amplitude and mode rotation and possible evidence of on-axis current drive. Also, a computer model of heating from NBI was created that offers predictions of the degree of heating under varying plasma conditions and beam parameters.

References

1) H. E. Mynick and J. A. Krommes Phys. Rev. Lett. **43** 1506 (1979)

2) H. Naitou, T. Karnimura and J. M. Dawson J. Phys. Soc. Japan 258-265 (1979)

3) C. B. Forest, et. al. Phys. Rev. Lett. **73**, 18, (1994)

4) C. B. Forest, et. al. Phys. Rev. Lett. **79,** 3, (1997)

5) E. Ruskov, et. al. Plasma Phys. Control. Fusion 38, 389-403 (1998)

6) D. S. Darrow, et. al. PPPL-3754

7) V. A. Yavorksij, et. al. PPPL-3733

8) A. V. Arukeev, et. al. Nucl. Fusion, **40** 4 (2000)

9) C.Angoni et. al. Plasma Phys. Control Fusion **44** pp. 205-222

10) E.D. Fredricson et. al. Phys. Rev. Lett. **87**, 14, 145001

11) H.L. Berk, B.N. Breizman and M.S. Pekker Plasma Phys. Rep. 23 p.778 (1998)

12) J.S. Sarff, N. E. Lanier, S. C. Prager, M. R. Stoneking Phys. Rev. Lett. **78** 62 (1997)

2 Review of Magnetic Stochasticity

Before addressing the problem of fast ion confinement in a magnetic field (stochastic or otherwise) we must first have a proper description of the magnetic field. We will begin in Section 2.1 with a brief description of the RFP with particular emphasis on the magnetic field structure. In Section 2.2 we present a brief overview of magnetic stochasticity in the RFP as it is of central importance in understanding the confinement of particles of all gyro-radii.

2.1 Introduction to The Reversed Field Pinch

The RFP is a toroidally axisymmetric plasma where a significant portion of the plasma current is self-generated, the so-called "RFP dynamo". The plasma is formed in the usual way, applying an electric field to a gas inside the vacuum vessel and introducing a source of startup electrons to assist in the ionization. The poloidal and toroidal fields in this configuration are of the same order throughout most of the plasma volume, unlike the tokamak where the toroidal field is much larger than the poloidal field. Also the toroidal field reverses in the plasma edge thus giving the RFP the "reversed" part of its name. Fig. 2.1 shows the RFP equilibrium magnetic configuration.

We express the field line pitch as the ratio of the toroidal to poloidal transits. This ratio is refered to as the "safety factor" or "q". Although referred to hereafter as the

"safety factor", it is not used in the context of plasma stability, merely a convention to describe the field line pitch. The expression for q, in the unshifted cylindrical approximation, is

$$q(r) = \frac{rB_\phi(r)}{R_0 B_\theta(r)}. \tag{2.1}$$

Here r is the distance from the magnetic axis, B_ϕ is the toroidal magnetic field, R_0 is the major radius and B_θ is the poloidal magnetic field. The safety factor is monotonically decreasing over the plasma volume and passes through zero where B_ϕ reverses direction at the reversal surface.

Magnetic islands can grow at locations where a radial magnetic perturbation characterized by a poloidal mode number, m, and a toroidal mode number, n, is perpendicular to the wave vector of the equilibrium magnetic field. Mathematically we write this as $\vec{k} \bullet \vec{B} = 0$. These resonant surfaces occur where $q(r)$ is the ratio of those two mode numbers. We can express this as

$$q(r) = \frac{m}{n} \tag{2.2}$$

where m and n are again the integer mode numbers. If the islands are sufficiently large, they can overlap and render the magnetic field strongly stochastic in the overlapped region. Particles can then follow the broken and diffusive field lines quickly to the wall. This is the most serious problem with the RFP configuration and is a key concern when considering the effect of this stochastic field on NBI ions.

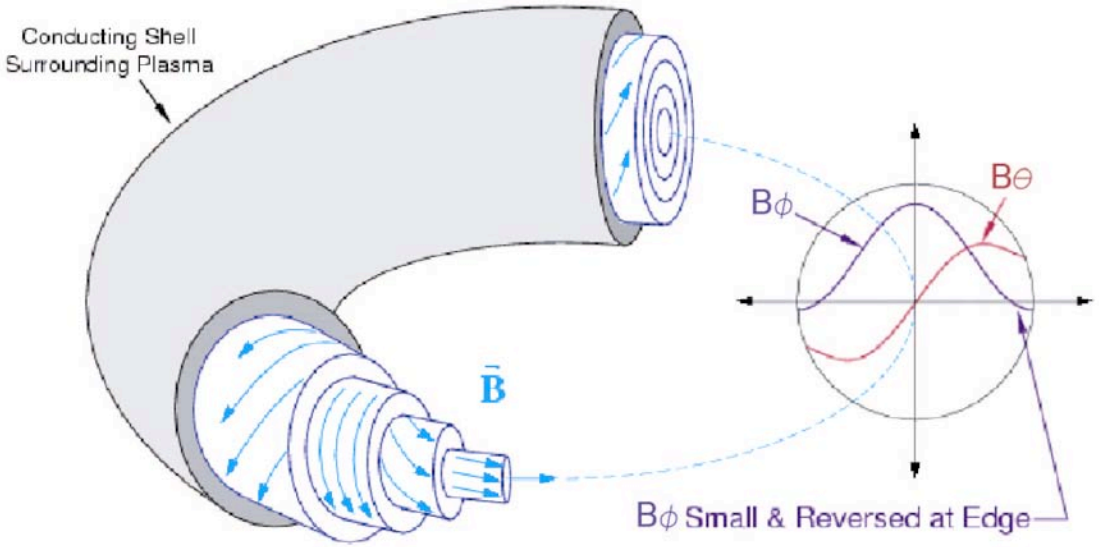


Figure 2.1 Equilibrium magnetic fields in the RFP. The toroidal, B_ϕ and poloidal, B_θ components are approximately equal in magnitude, and the toroidal field reverses near the wall.

2.2 Magnetic Island Formation and Stochasticity

It is understood that in the RFP, electrons follow magnetic field lines closely and these field lines are rarely on closed, nested flux surfaces. Instead they follow stochastic paths over a large extent of the plasma radius. This allows rapid transport of particles and energy out of the system. In this section we will focus on what gives rise to a stochastic magnetic field and the quantification of stochasticity in terms of a diffusion coefficient. This description provides background for upcoming field line simulations that are covered in Chapter 3.

An equilibrium magnetic field profile is shown in Figure 2.1 and the q-profile is in Figure 2.2. However, the RFP is susceptible to unstable tearing modes, which form

magnetic islands at the mode rational surfaces. The locations where q is a rational number are resonant with radial magnetic fluctuations with mode numbers m and n, and at these places, field lines reconnect and can form magnetic islands. The size of the magnetic islands in a sheared-slab approximation (See Appendix A) is given as

$$W_{1,n} = 4\sqrt{\frac{\tilde{b}_{1,n}}{B_{\grave{e}}(r_s)}\frac{r_s}{n\left|q'(r_s)\right|}} \tag{2.3}$$

where $\tilde{b}$ is the radial magnetic field fluctuation of a given mode number (m = 1, n) and r_s is the location of the associated rational surface. Figure 2.3 shows the q-profile with the island widths for each resonant mode (m = 1, n = 6-32) overplotted.

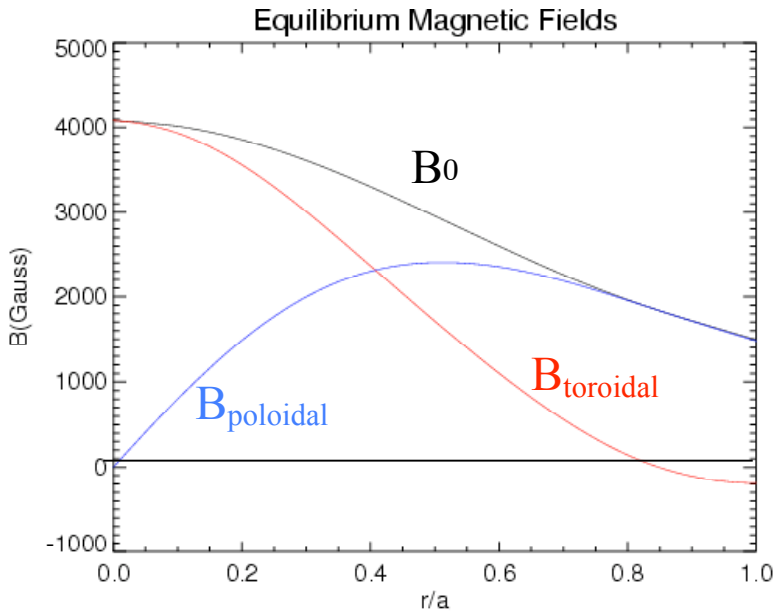


Figure 2.2 Equilibrium magnetic fields for a typical MST plasma. The reversal surface occurs at around r/a = 0.82.

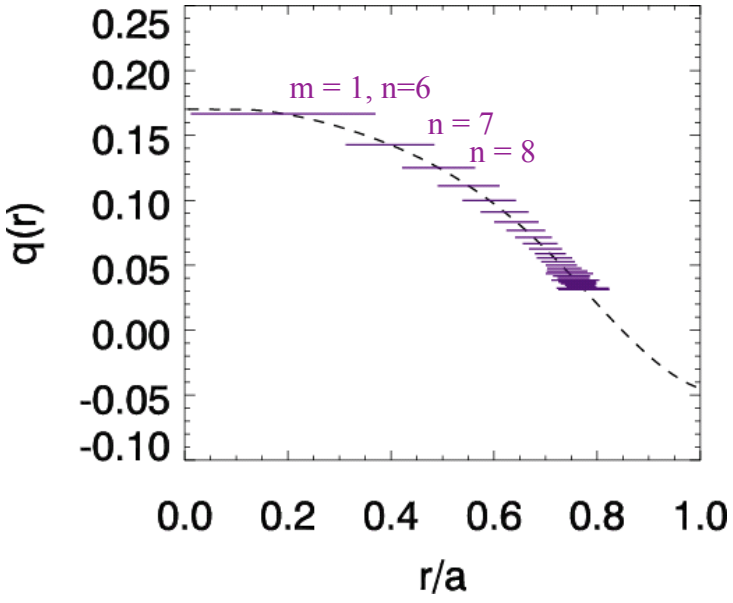


Figure 2.3 – Safety factor (dashed) with magnetic island widths (solid horizontal lines) shown at the location of the rational surfaces. Note the density of islands is high as $q = 0$ is approached.

When magnetic islands overlap, the orbits of field lines in the overlapped region become stochastic and field lines can wander randomly throughout the region. A field line can be assumed to be getting a random "kick" as it tries to continue along it's original trajectory. This suggests a random walk model for which one can define[5] a diffusion coefficient, D, according to

$$D = \lim_{l \to \infty} \frac{\left\langle (\Delta r)^2 \right\rangle}{2l}, \tag{2.4}$$

where Δr is a radial step size or excursion, and l is the length along the field line.

Cursory examination of this expression reveals that it is only valid if the radial excursion is allowed to increase without bound, with the additional constraint that $\Delta r << l$. In our confined plasma devices, this is not the case and so a meaningful limit must be placed on the value of *l*. This will be explored further in Section 3.1.4 when we discuss the results of simulation of a stochastic magnetic field and the criteria for arriving at the value of the magnetic diffusion coefficient.

Rechester and Rosenbluth[6] applied the idea of magnetic island overlap and stochasticity to describe electron heat transport. The electron transport observed in the core of the RFP is well described by their formulation and provides a good picture of the effect of magnetic island overlap on plasma confinement.

References

1) J.B. Taylor, Phys. Rev. Lett. **33**, 1139 (1974)

2) T.D. Rempel et. al., Phys. Rev. Lett. **57**, 1438 (1991)

3) N. Lanier, PhD Thesis, Dec. 1999

4) T. M. Biewer, et. al., Phys. Rev. Lett. 91, 4 45004 (2003)

5) Jokipii, J. R. ApJ **146**, 480 (1966)

6) A. B. Rechester and M. N. Rosenbluth, Phys. Rev. Lett. **40**, 1 (1978)

3 Simulation of Magnetic Field Line Diffusion in MST

In Section 3.1 we discuss a field line tracing code that was developed for modeling the stochastic magnetic field in MST. By using experimentally obtained data where possible, we attempt to closely approximate the real magnetic field. The equilibrium fields can be reconstructed accurately by using an in-house code MSTFIT.[1] This code generates a plasma equilibrium that best satisfies constraints imposed on it by diagnostic data such as density, temperature, and edge magnetic fields. A 3-D resistive MHD code DEBS[2] is used to evolve a cylindrical plasma and determine the radial structure of the tearing modes that give rise to magnetic islands. The eigenmodes are normalized to experimentally measured values at the plasma edge and used as input to the field line tracing code. After integrating the resultant field line trajectories, we use the prescription in Eq. 2.4 and calculate the magnetic diffusion coefficient for various levels of magnetic perturbations.

Section 3.2 compares the result to experimental inferences of the magnetic field diffusion coefficient. Lastly we relate magnetic field line diffusion to our ultimate goal, describing the diffusion of fast ions.

3.1 Field Line Tracing Code (MAL) [MAgnetic Lines]

MAL is a FORTRAN code which integrates along a given magnetic field line to solve for its trajectory. The profiles for the equilibrium field are given at discrete points as a function of radius only. The fluctuating fields are given as a spectrum in wavelength space poloidally and in real space radially and axially. To smoothly integrate along a trajectory, an analytical expression is required. This is obtained by doing a 9-term

Chebyshev fit to the equilibrium field and a 16-term Chebyshev fit to the perturbation modes. Other techniques are used to handle the fits near the origin, but the overall agreement is within 2% of the input fields. Sections 3.1.1 – 3.1.2 discuss specifically how the magnetic fields are computed.

3.1.1 Equilibrium Reconstruction

To have the field line tracing represent the physical situation as closely as possible, we begin with a formulation for the toroidal and poloidal magnetic fields in a typical MST discharge. Before proceeding it should be noted that unless otherwise stated, the plasma and machine parameters discussed and used in the simulation are:

Major radius: 1.5m

Minor radius: 0.5m

Plasma Current: 385kA

Edge poloidal magnetic field: 0.15T

Axis magnetic field: 0.4T

The MSTFIT equilibrium reconstruction code uses available diagnostics as constraints to generate a best-fit equilibrium magnetic profile. MST has a wide variety of diagnostics available to probe the plasma conditions and the magnetic field. We employ Far InfraRed (FIR) interferometry[3] to obtain information about the radial electron density profile, and FIR polarimetry to get the poloidal magnetic field. A Motional Stark Effect (MSE) diagnostic[4] is used to determine the on-axis toroidal field. Multi-point Thomson scattering[5] is used to determine the electron temperature. Charge Exchange

Recombination Spectroscopy (CHERS)[6] is used to determine impurity ion temperature, which when combined with the electron density measurements and a estimate[7] of Z_{eff} ($Z_{eff} \sim 2$), yields the plasma ion density. The resultant radial profiles over a large ensemble of plasma shots result in the equilibrium fields shown in Fig. 2.2 with the safety factor shown in Fig. 2.3.

3.1.2 Radial Profiles of Tearing Modes

As we have seen, the overlap of islands is dependent on the size of the fluctuations that define the island size. This required an estimate of resonant perturbation at each rational surface. Additionally, the entire radial profile of each mode is needed to accurately represent the overall magnetic field.

This is accomplished by making use of the DEBS code, which is a cylindrical 3-D resistive MHD solver. The experimental plasma parameters are used as inputs and the system is allowed to evolve self-consistently. The most serious constraint to this modeling is the question of the Lundquist number, which in experiment is,

$$S = \frac{\mu_0 \mathrm{v}_a}{\eta} = 6.3 \times 10^6 \tag{3.1}$$

where v_a is the Alfvén speed,

$$\mathrm{v}_\mathrm{a} = \frac{B}{\sqrt{n_i m_i \mu_0}} \cong 1.3 \times 10^6 \, m/s \tag{3.2}$$

a is the minor radius, taken to be a characteristic length for the system, and η is the plasma resistivity, given by

$$\eta = 5 \times 10^{-5} \frac{\sqrt{Z_{eff}} \ln \Lambda}{T_e^{3/2}} \cong 1.3 \times 10^{-7} \, \Omega m \tag{3.3}$$

In the simulation, as one tries to resolve ever-larger S, it requires increasing spatial and temporal resolution to capture the energy evolution of the magnetic field. In practice, this means that a value of $S = 10^4$ takes a few hours to execute, while $S = 10^6$ takes about 1.5 months. We have one run at $S = 10^6$, and believe this to be a valid representation of the magnetic structure in a typical RFP discharge. It captures experimentally observed behavior such as the dominance of the $m = 1$, $n = 6$ mode and more importantly sawtooth behavior. These features are only resolved at large S, but are a ubiquitous feature of MST plasmas.

The key results of the DEBS run are shown in Fig. 3.1 and Fig. 3.2. Fig. 3.1 shows the spectrum of magnetic modes and their amplitudes. The convention in Fig. 3.1 is that the core-resonant modes (inside the reversal surface) have negative n values. The modes resonant outside the reversal surface have positive n values and are typically very small. These modes are not included in the field line or particle tracing simulations as they would contribute very little and greatly increase the simulation time. Fig. 3.2 shows the radial structure of the $m = 1$, $n = 6$-8 modes. These are global internally resonant tearing modes and their values at the rational surfaces partially determine the magnetic island width. There are also $m = 0$ modes which are resonant at the reversal surface. These do not contribute much to the magnetic topology over most of the plasma radius; they render very few orbits stochastic, but are included in the simulation because they act as a non-negligible perturbation to the overall field line. The $m = 0$ modes also couple to the $m = 1$ modes, but the localized nature of the eigenmode amplitudes tends to make this effect negligible.

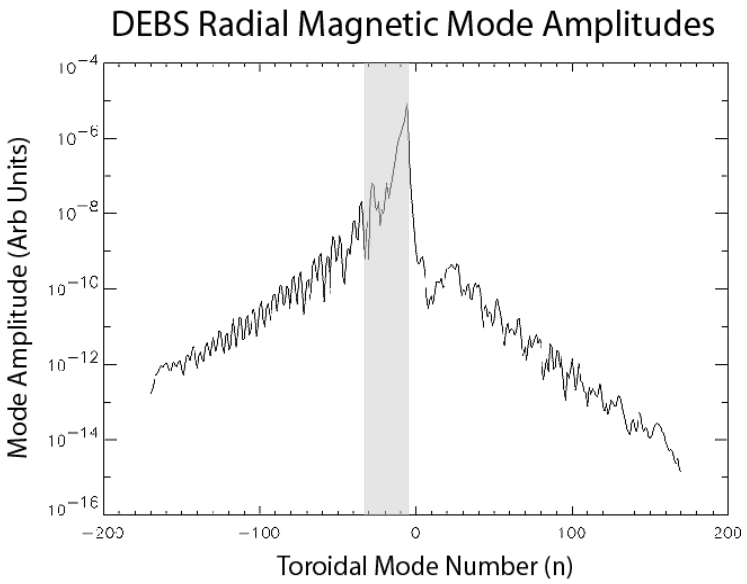


Figure 3.1 Radial magnetic mode amplitudes from DEBS. The internally resonant modes -6 to -32 are denoted by the shaded region. The amplitude drops rapidly with increasing $|n|$. The modes resonant outside reversal (positive n on this plot as DEBS takes negative values to be internally resonant modes) are even smaller and so can be safely neglected.

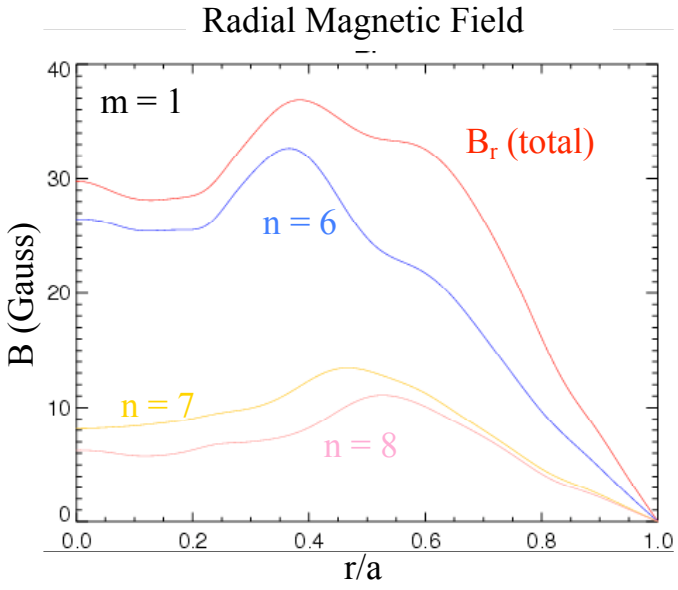


Figure 3.2 Radial magnetic perturbation eigenmodes from DEBS. The $m = 1$ modes are global, with the $n = 6$ dominant in typical MST discharges.

3.1.3 Scaling to Experimental Quantities

The magnetic profiles mentioned previously come from two sources, MSTFIT for the equilibrium fields, and DEBS for the radial eigenmodes. The amplitude of the radial eigenmodes from DEBS are not necessarily representative of a typical magnetic field structure. The modes in DEBS change in time as energy is transferred back and forth between them. We do our best to pick a time in the simulation that most accurately represents a typical discharge, i.e., between sawtooth crashes. To be more precise, we would like to take experimentally obtained fluctuation levels and scale the radial eigenmodes to those. The problem is that we don't measure $\tilde{b}_r(r,m,n)$ so another method must be found.

MST contains a toroidal array of 64 magnetic pickup coils that record the values of the edge magnetic fluctuations. These allow Fourier decomposition into its respective modes as shown in Fig. 3.3. These coils measure the toroidal (n) spectrum of the poloidal and toroidal magnetic fluctuations at the wall. What remains is to determine which component to use for scaling the $m = 0$ and $m = 1$ modes in the MAL code.

We use Ampere's law, at the wall of MST, to analyze the spectral components of the measured magnetic fluctuations. The radial component of Ampere's law at the wall is given by

$$\hat{r} \bullet (\nabla \times \tilde{b})\,|_{r=a} = \mu_0 \tilde{j}_r\,|_{r=a} \cong 0 \tag{3.4}$$

which becomes

$$\frac{1}{\mu_0}\left(k_\theta \tilde{b}_\phi - k_\phi \tilde{b}_\theta\right)_{r=a} = \frac{1}{\mu_0}\left(\frac{m}{a}\tilde{b}_\phi - \frac{n}{R}\tilde{b}_\theta\right) \cong 0 \tag{3.5}$$

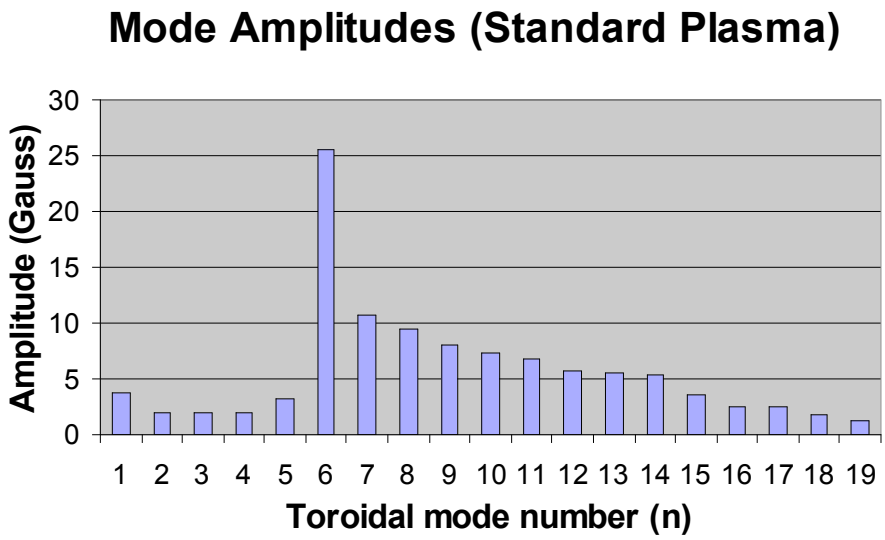


Figure 3.3 Spectrum of magnetic fluctuation amplitudes measured at the vessel wall. The poloidal component is dominantly $m = 1$ so we use this for scaling to the internal tearing modes.

The poloidal fluctuation at the wall is then

$$\tilde{b}_\theta(m,n) = \frac{m}{n}\frac{R}{a}\tilde{b}_\phi(m,n) \tag{3.6}$$

If we set $m = 0$ in the above expression, we see that $\tilde{b}_\theta(0,n) = 0$. This means that the $m = 0$ component is purely represented by $\tilde{b}_\phi$. For the $m = 1$ modes, we note that since $\tilde{b}_\theta(0,n) = 0$, $\tilde{b}_\theta(m,n) = \tilde{b}_\theta(1+,n)$. Given that $m = 0, 1$ are the dominant poloidal mode numbers in MST, then we can use $\tilde{b}_\theta(m,n)$ to be our values for the $m = 1$ modes.

In the field line tracing code then, we scale the $m = 1$ modes to the experimentally measured poloidal fluctuations and the $m = 0$ modes to the toroidal component. The final

assumption made is that the radial magnetic field scales consistently with this analysis according to the relative values that were provided by the DEBS code.

Referring back to Fig. 3.3 we note that the $n = 6$, which is the innermost resonant mode, dominates. However, modes 1-5 are not resonant, but still present as perturbation components. This will be very important when we examine fast ion orbits in Chapter 5.

3.1.4 Magnetic Diffusion Coefficient for Standard Plasmas

MAL solves the field line equations,

$$\frac{dr}{B_r} = \frac{rd\theta}{B_\theta} = \frac{Rd\phi}{B_\phi} \tag{3.7}$$

where θ is the poloidal angle, ϕ is the toroidal angle, r is the minor radial variable, R is the major radius. B_r is the radial magnetic field (in our case just the perturbation) while B_θ and B_ϕ are the poloidal and toroidal magnetic fields (both equilibrium and perturbation), respectively. The resultant trajectories can be visualized as Poincaré plots (Fig. 3.4). These allow a visual assessment of the magnetic topology and clearly show the presence of good flux surfaces, magnetic islands, and regions of stochasticity. Quantitatively, the code also evaluates the following to arrive at a value for the magnetic diffusion coefficient (Fig. 3.5), D_m.

$$D_m = \frac{\left\langle (\Delta r)^2 \right\rangle}{2\Delta l}, \tag{3.8}$$

where Δr is the radial excursion, Δl is the field line length, and the brackets denote an average over a large number of realizations.

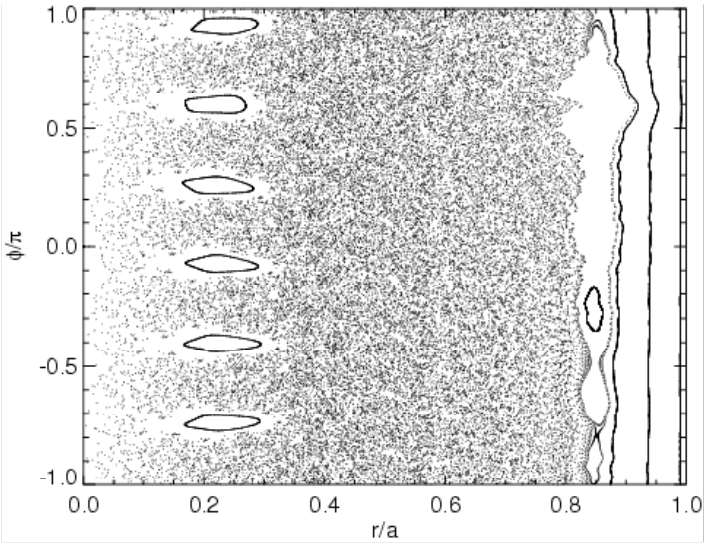


Figure 3.4 Field line puncture plot by MAL. The field lines are stochastic over most of the plasma radius, as expected. The $n = 6$ islands are only partially overlapped, as well as the $m = 0$ islands (resonant at r/a = 0.84, $q = 0$) which remain due to truncation of the high-n $m = 1$ mode spectrum.

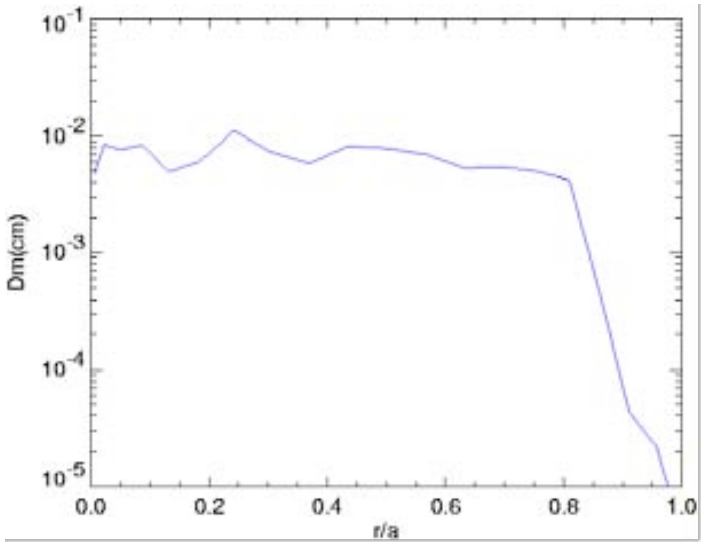


Figure 3.5 MAL calculation of the magnetic diffusion coefficient. The diffusion coefficient is rather constant over most of the plasma radius, but the values at r/a < 0.2 and r/a > 0.8 are not valid there since the system is not diffusive in those regions.

It is challenging to define a meaningful diffusion coefficient in such a system where the degree of stochasticity (randomness), which is inherently a function of the perturbation strength, varies with radius, but also where the very idea of diffusion is invalid (e.g. at the boundaries r = 0 (where we have an imposed mathematical regularity condition) and near the reversal surface). While such a system defies any simple description, we can remove one aspect of uncertainty, namely the varying degree of stochasticity, and create a model that illustrates the effects of a bounded system on the calculation of a diffusion coefficient. This is a fruitful analysis because as we will see, the model will replicate the full field line tracing simulation quite well.

We consider a 1-D random walk model in a bounded domain. The trajectory that is followed is a small sinusoidal perturbation, in the $x - y$ plane (representative of the spatial trajectory of a magnetic field line),

$$y = \delta \sin\left(2\pi n \frac{x}{L}\right) \tag{3.9}$$

δ is the amplitude of the perturbation, taken to be 0.01 m, n is the number of wavelengths in the system, akin to the toroidal mode number of a radial magnetic eigenmode and L is the length of the system, taken to be the toroidal circumference of MST or about 9.4 m.

In addition, we impose upon the trajectory a probability of getting a random "kick" in the y (radial) direction (either inward or outward, equally likely) to simulate the stochasticity. The probability was chosen such that after each step in the x direction, it would have been kicked a radial distance of

$$\Delta y = \sqrt{2 D \, \Delta x} \tag{3.10}$$

with D being a diffusion coefficient of value 10^{-4} m (taken from the MAL result) and Δx a step size in the x direction. A reflective boundary was placed such that if a radial step were to take it past the boundary it would instead be kicked in the opposite direction to the intended excursion. A plot of one trajectory under these conditions is shown in Fig. 3.6. As can be clearly seen the particle wanders in radius (only weakly affected by the underlying sinusoidal trajectory) and reflects off the boundaries as described above.

We now ensemble many trajectories this way and look at how the calculated diffusion coefficient varies with length and compare it to the specified diffusion coefficient which we used to determine the amplitude of the radial step. The horizontal distance was expanded and the ensemble was performed over 1000 trajectories.

The result is shown in Fig. 3.7. The sinusoidal field is shown in the red dash-dot line, and we can see that for short line lengths, the line closely follows the perturbation,

$$\frac{(\Delta y)^2}{2\Delta x} \sim \frac{1}{2}\left(\frac{dy(x)}{dx}\right)^2 \Delta x \sim \frac{1}{2}\delta^2 \Delta x\,. \quad (\text{short lengths}) \tag{3.11}$$

What this means is that the motion is considered to be ballistic, or linear with the imposed perturbation. As additional kicks take the trajectory away from the original line, the calculated value for the diffusion coefficient saturates at the value shown as the straight blue dashed line in the figure.

$$\frac{(\Delta y)^2}{2\Delta x} \cong D \quad (\text{Diffusive regime}) \tag{3.12}$$

In this range the motion is considered to be purely diffusive as it has been altered from its ballistic path, but has not yet (on average) interacted with the boundary. Finally, as the distance becomes even longer, additional reflections are included in determining the trajectories position and as such the radial excursion approaches some finite average,

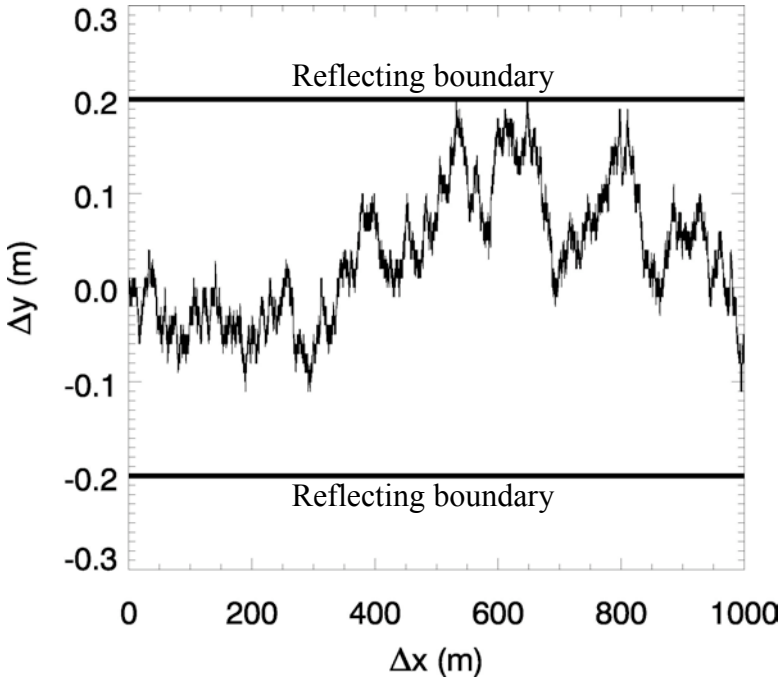


Figure 3.6 A single trajectory in the 1-D diffusion model is shown. The solid boundaries represent a reflecting wall that limits the vertical range of diffusion.

whereas the length has no such bound. This results in the decrease of the calculated diffusion coefficient that goes approximately as $1/\Delta x$.

$$\lim_{\Delta x \to \infty} \frac{\left\langle (\Delta y)^2 \right\rangle}{2\Delta x} = 0 \tag{3.13}$$

Similarly, we can consider the calculated value of the diffusion coefficient from MAL and see how it varies with length. This is shown in Fig. 3.8 and is perhaps surprisingly close to the result from the simple model shown in Fig. 3.7, though the benefit of the agreement means that the understanding of the bounded diffusion and its role in the saturation of the value of the diffusion coefficient is sound.

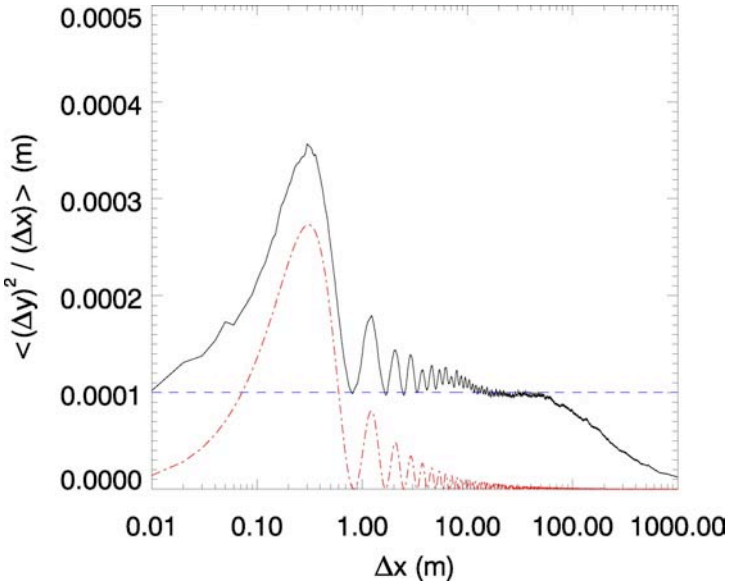


Figure 3.7 Dependence of D vs. Δx in the 1-D diffusion model. Initially the displacement is highly tied to the perturbation (shown in the red dash-dot line), then saturates to the imposed diffusion coefficient (shown in the blue dashed line), and finally decreases as $1 / \Delta x$ due to being bounded.

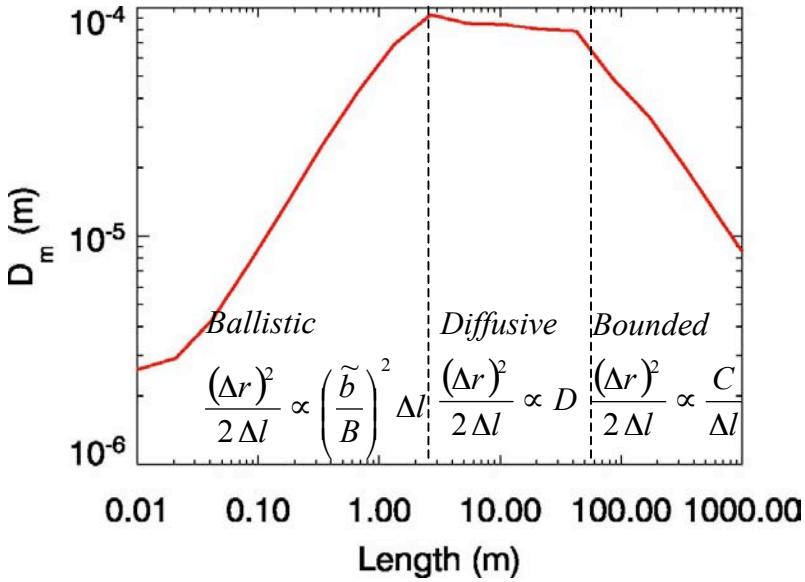


Figure 3.8 Diffusion coefficient vs. length from MAL. Just like in the simple model we see a linear (ballistic) region, a diffusive region and a bounded region. The length scales at which the regions transition are quite close to what was shown in Fig. 3.7.

If the step size that we use to define our diffusion coefficient is much less than the electron mean free path, we can say that the diffusive transport is governed by the field line stochasticity rather than the Brownian motion that would result from particle collisions. Rechester & Rosenbluth[8] analyzed the heat transport resulting from electron diffusion along a stochoastic magnetic field in the collisionless regime and it can also be shown to be valid for our analysis as well. When we estimate the electron mean free path in MST,

$$\lambda_{MFP} = \frac{v_{T,e}}{\nu} = \frac{1.2\times10^{7}\ m/s}{5.6\times10^{4}\ /s} = 214\,m \tag{3.14}$$

where $\nu = 2.91x10^{-6} n_e \ln\Lambda T_e^{-3/2}$ is the electron collision frequency, we see that λ_{MFP} exceeds the distance at which the field line diffusion saturates to D_M, which occurs after about 1 m. Physically this means that transport due to following a diffusive magnetic field dominates over collisional Brownian motion.

We can also compare the calculated diffusion coefficient vs. that predicted by Rechester & Rosenbluth. Rechester and Rosenbluth derive

$$D_m\big|_{r=r_s} = L_c\left(\frac{B_r(m,n)}{|B|}\right)^2\Bigg|_{r=r_s} \tag{3.15}$$

with L_c being an "auto-correlation" length. Physically, L_c represents the length over which two exponentially diverging field lines become decorrelated from each other and has been calculated[9] to be

$$L_c = \frac{\pi R}{\ln\left(\frac{\pi s}{2}\right)} \tag{3.16}$$

where

$$S = \frac{1}{2}\left(\frac{W_{m,n} + W_{m',n'}}{\left| r_{m,n} - r_{m',n'} \right|} \right) \tag{3.17}$$

is the "overlap parameter" that characterizes how overlapped two adjacent islands are.

For MST, where the islands are sufficiently overlapped, a reasonable value of S might be around 5, which would give an L_c of approximately 1 m. Interestingly enough, this is approximately the step size, in the 1-D model of diffusion presented previously, where the field line motion transitions from ballistic to diffusive, which is what one would expect given the definition of the correlation length.

The formula in Eqn. 3.15 can be better understood by considering the field line equation

$$\Delta x \sim \frac{\tilde{b}_x}{B} \Delta l \,. \tag{3.18}$$

If this is to describe a diffusive system then

$$D = \frac{(\Delta x)^2}{2\Delta l} = \left(\frac{\tilde{b}}{B} \right)^2 \frac{\Delta l}{2} \tag{3.19}$$

and then we identify $\Delta l / 2 \sim L_c$. This analogy is only intended to elucidate how the radial magnetic perturbation plays the role of a random displacement.

We use our estimates of the island size and positions to get a value for the diffusion coefficient by Rechester and Rosenbluth's method, and compare with the field line tracing code. The results are shown in Fig. 3.9. We see that it overestimates the

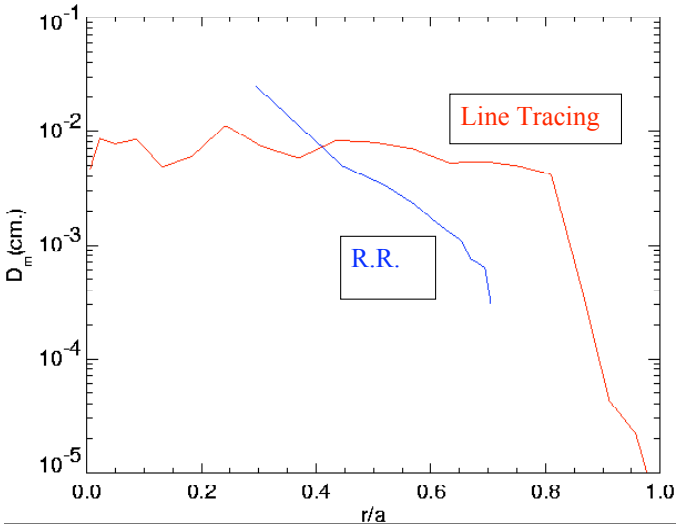


Figure 3.9 Comparison of Rechester-Rosenbluth diffusion to MAL results. Agreement is fair in the mid radius, but R.R. overestimates where overlapping is weak, and underestimates where overlapping is strong.

value where the field is not strongly overlapped, as would be expected, and underestimates it where are field is most stochastic. Rechester and Rosenbluth formulate this expression in the limit of a large number of densely packed resonances. In the mid radius, the island are overlapped, but not densely so; perhaps 2-3 islands at a given radial location. Where the islands are more strongly overlapped, we have a truncated mode spectrum; we only go to $n = 32$. There is also an $m = 0$ perturbation that can influence the trajectory. The approximate agreement is encouraging though in light of the limitations due to the varying stochasticity and boundedness of the diffusive region.

3.1.5 Magnetic Diffusion Coefficient in Reduced Fluctuations "PPCD"

Since we went to great lengths to use realistic fluctuation levels for our standard plasma, we can examine the case where fluctuations are reduced. In MST we inductively

drive current through application of a poloidal electric field to flatten the parallel current profile, and lessen the source of free energy available to drive instabilities. This is known as Pulsed Parallel Current Drive "PPCD." By measuring the fluctuation levels during PPCD we can use different factors to scale the fields we use in the simulation, relative to the values we used in the standard discharge.

We observe experimentally that the fluctuations decrease for many of the internally resonant modes. The difference between standard fluctuations and PPCD is shown in Fig. 3.10. For all but the most core-resonant modes, the reduction is a factor of about 4, and in some instances even greater.

With smaller magnetic perturbations we expect less island overlap and a reduction of magnetic stochasticity in our simulations. The q-profile with the islands overplotted is shown in Fig. 3.11, and indeed from this we expect much of the core to no longer be stochastic. The q-profile is slightly different from the standard case; the application of the poloidal electric field drives the toroidal magnetic field more negative, resulting in lower edge value of q and an inward shift of the reversal surface. The Poincaré plot for PPCD is shown in Fig. 3.12 and we see that the n = 6, 7 and 8 islands are now not overlapped and flux surfaces in the core have been restored.

Before discussing the results from MAL regarding the D_m for PPCD plasmas, we must make a couple of caveats. First, the DEBS eigenmodes are only for "standard"

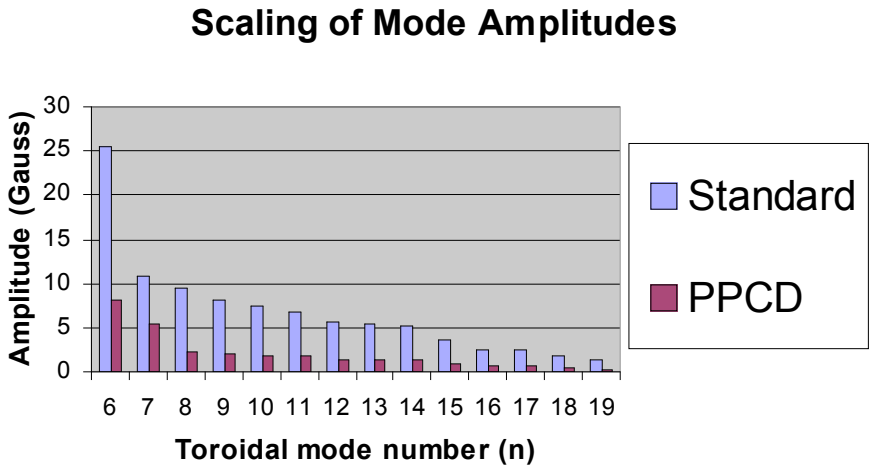


Figure 3.10 Magnetic mode amplitude reduction during PPCD. The mode amplitudes are observed to decrease up to a factor of 4 from the standard case.

plasmas; there is currently no DEBS simulation of PPCD. Secondly, all the caution that had to be applied to the estimate of D_m in the standard case is even more necessary here. As can be seen from the Poincaré section, the radial range of stochasticity is reduced, so the excursions are more tightly bounded. Nevertheless, we can examine the code's results in the same method as before and in Section 3.2 compare with experiment.

The profile for the diffusion coefficient is shown in Figure 3.13. We can see that it is small near the axis where we have restored flux surfaces. While the field lines are highly perturbed due to the presence of islands, because the lines are unbroken the mean square deviation along a field line quickly approaches to zero; an perturbed but non-stochastic field line is very much bounded, thus the "diffusion coefficient" goes from reflecting the linear motion quickly to the $1/l$ behavior, with no diffusive regime. The value of D_m is highest where expected, and more importantly, where the field lines are stochastic. The

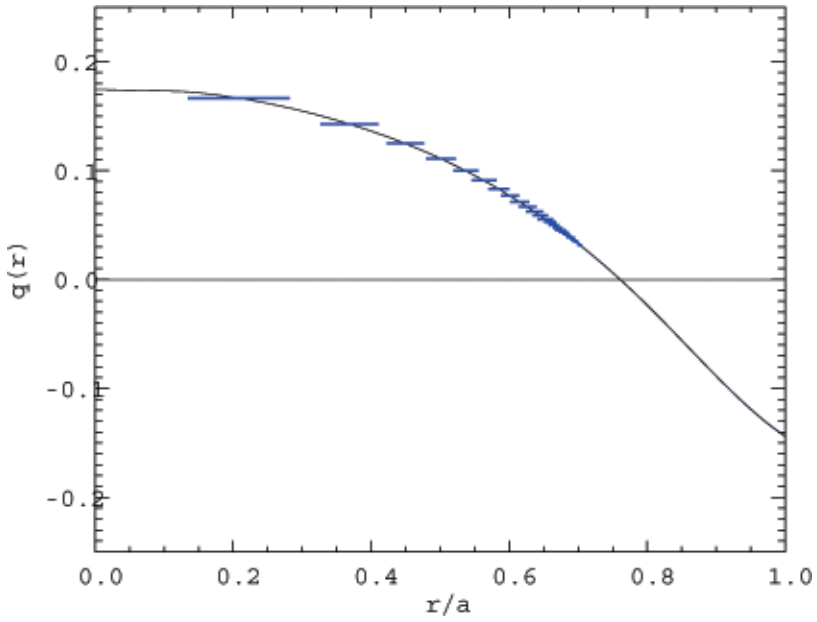


Figure 3.11 Safety factor for PPCD. The increased reversal during PPCD gets pushes the resonances inward, and lower levels of fluctuations reduce the stochasticity.

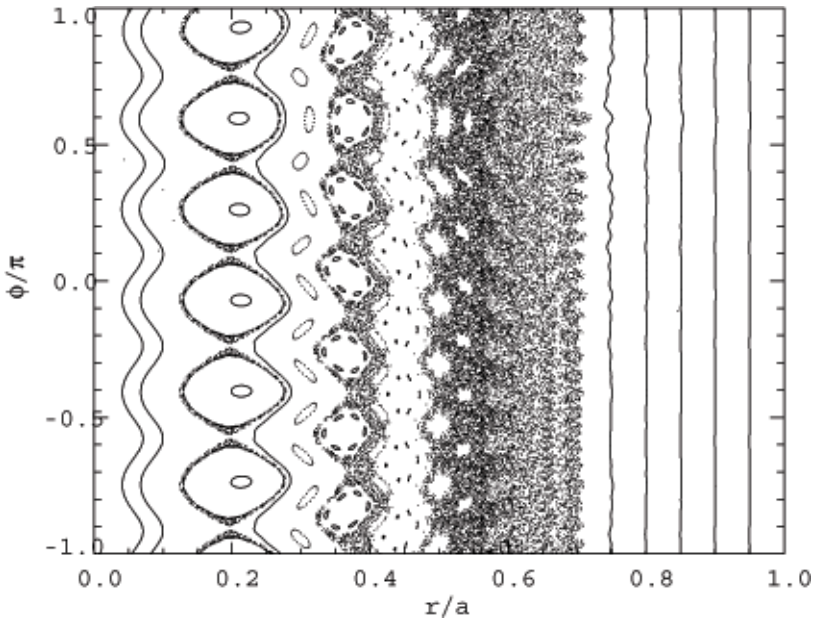


Figure 3.12 Puncture plot for "PPCD". Intact islands appear at the n = 6,7,8 resonant surfaces. Only a thin band of stochasticity remains between $0.5 < r/a < 0.7$.

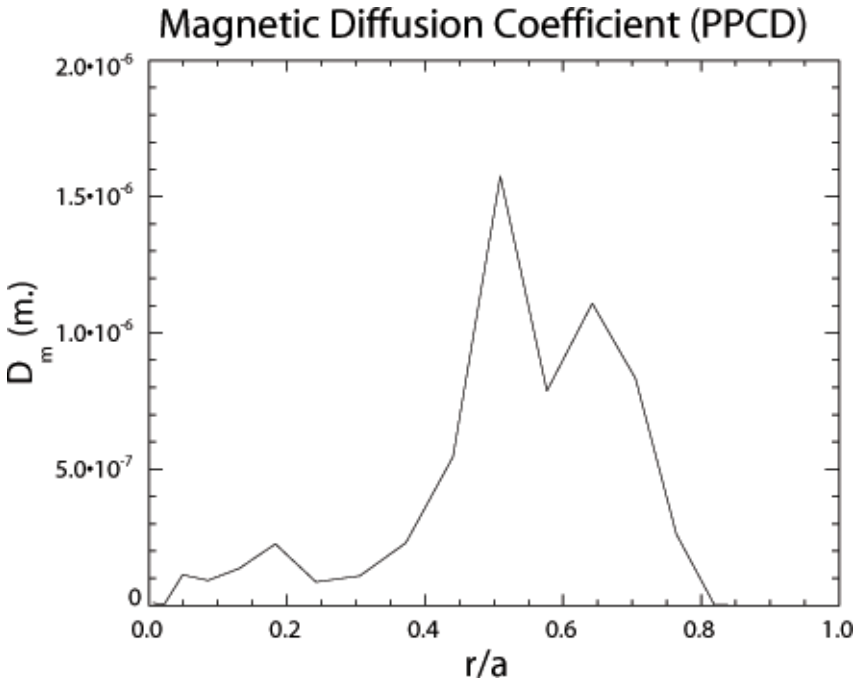


Figure 3.13 Magnetic diffusion coefficient in PPCD (from MAL). Diffusion is reduced about a factor of 100 where it would still be valid $0.5 < r/a < 0.7$.

value of D_m is around 10^{-6} m., which is a factor of about 100 lower than the standard case. The R.R. estimate, to the degree that it would be valid for PPCD, would predict only a factor of 10 decrease, since the diffusion coefficient roughly scales as the square of the fluctuation amplitudes.

3.2 Comparison To Electron Heat Transport Experiment

Electrons have very small gyro-orbits because of their small mass, so we expect them to closely follow the field lines, stochastic or not. Rechester and Rosenbluth further postulated that electron heat transport would be described by a diffusion coefficient

$$\chi_e(r) = D_m(r)\mathrm{v}_{te} \tag{3.20}$$

with v_{te} being the electron thermal velocity. This can be seen by considering that

$D = \frac{(\Delta x)^2}{\Delta l}$ and $\mathrm{v} = \frac{\Delta l}{\Delta t}$, therefore $\frac{(\Delta x)^2}{\Delta l}\frac{\Delta l}{\Delta t} = \frac{(\Delta x)^2}{\Delta t} = \chi = D\,\mathrm{v}$, which we can immediately identify as being a particle diffusion coefficient. Since the electrons are assumed to be in a Maxwellian distribution, their parallel velocity along the field line is the thermal velocity. In this analysis, the role of an ambipolar electric field, which restricts particle diffusion, but not heat diffusion, is ignored.

If true, then the electron heat flux would be described by Fourier's heat law, giving

$$Q_r(r) = -n_e(r)\chi_e(r)\frac{\partial T_e(r)}{\partial r} \tag{3.21}$$

where Q is the electron heat flux, n_e is the electron density, χ_e is the electron diffusion coefficient, and T_e is the electron temperature.

This was measured previously[10] in MST and used to get a value for the electron heat diffusion coefficient. Multiplying the results from MAL by the electron thermal velocity, we compare the simulation directly with experiment. For standard plasmas we have good agreement as shown in Fig. 3.14. The profile is flat in both cases with transport decreasing markedly near the reversal surface.

The PPCD case (Fig. 3.15) is interesting as agreement is only reached in the region where the comparison would be valid; where the magnetic field is stochastic. The electron temperature profile is shown in Fig. 3.16. Note that it is different than the standard T_e profile as the reduced fluctuations result in hotter plasmas, and also the ability to support temperature gradients appears due to the loss of a fast radial heat equilibration mechanism.

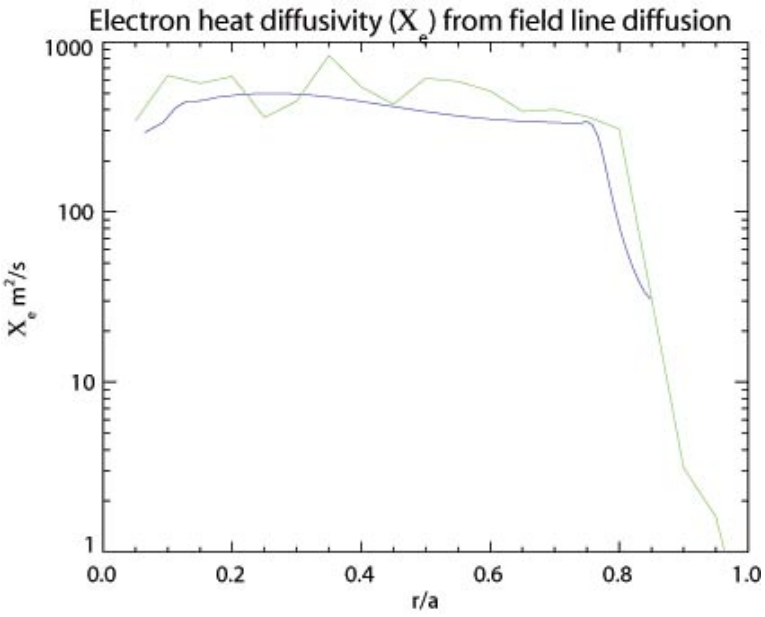


Figure 3.14 Comparison of electron heat diffusion between experiment (blue) and computation (green). Good agreement as found, including the presence of a transport barrier at the reversal surface.

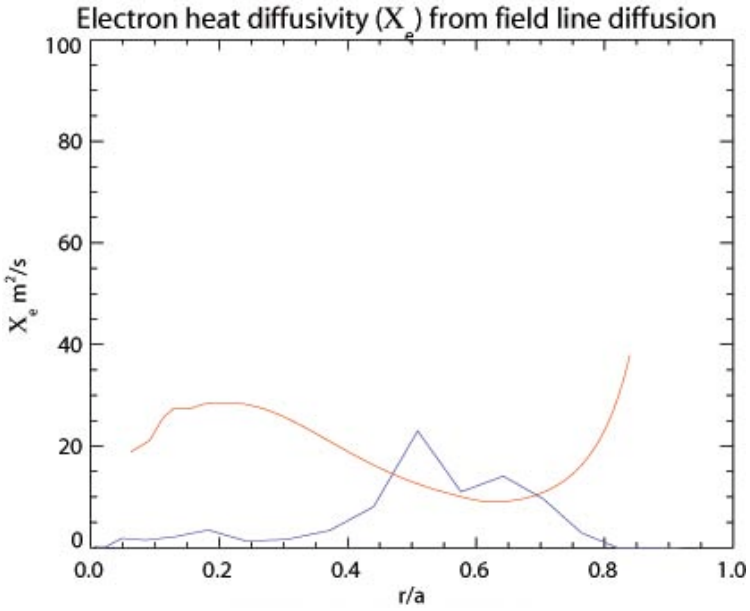


Figure 3.15 Comparison of electron heat diffusion between computation and experiment in PPCD. The experiment (red) agrees well with the numerical value (blue) only in the region where the comparison would be valid.

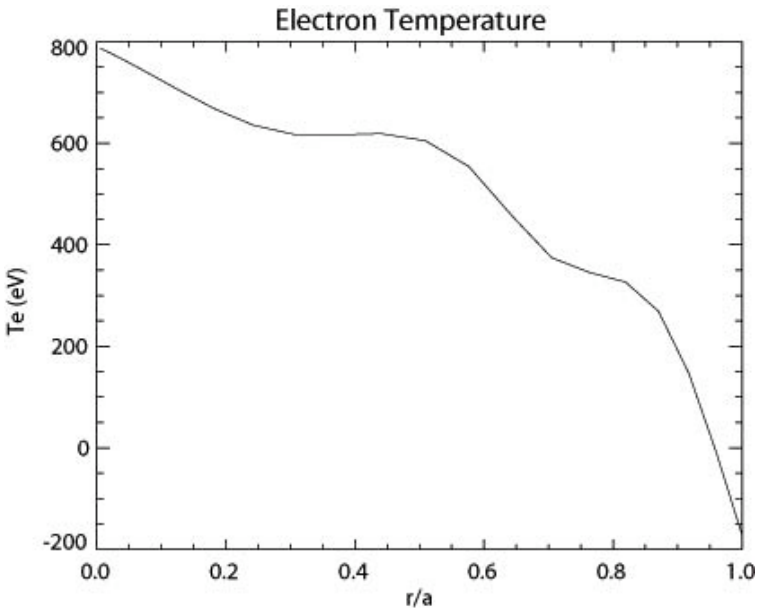


Figure 3.16 Electron temperature during PPCD as measured with Thomson scattering. The improved confinement of PPCD plasmas results in a higher electron temperature.

3.3 Connection between Field Line Diffusion and Fast Ion Diffusion

An interesting picture begins to emerge in that, thus far, having considered field line diffusion and electron diffusion, we have in essence studied the variation of particle confinement in the RFP as a function of gyro-radius. Magnetic field lines represent the zero energy limit of particle motion and were shown to be stochastic and poorly confined, which was quantified by evaluating the magnetic diffusion coefficient.

The confinement of the smallest gyro-radius particles, electrons, was reviewed from past measurements and compared to recent simulation of magnetic field line diffusion. The transport of electrons, again assuming no ambipolar electric field, was well described by flow along stochastic field lines.

By simply scaling upward in energy, one might expect to get the answer for the degree of confinement of faster particles, however this may not be so. The role of

particle drifts that are proportional to gyro-radius becomes important (Section 5.2). As we will see, this confinement dependence on gyro-radius is an important physics result from this thesis. Following chapters address the confinement of large gyro-radius particles both experimentally and theoretically.

3.4 Summary

The magnetic field is of central importance in understanding the orbits and subsequently the confinement of fast ions. In the RFP, magnetic fields are not typically on nested surfaces, but rather, fill stochastic volumes, which makes an analytic description of particle orbits impossible. A numerical approach was taken that captures important physics of the RFP magnetic field. Field line tracing was done to characterize its diffusive nature and allow quantitative comparison with experiment. A simple model was discussed that illustrated the consequences of a bounded stochastic system. The heat transport of electrons in a stochastic magnetic field was previously measured, and the inferred magnetic diffusion was close to the field line simulation results. Under reduced magnetic fluctuations, simulation and experiment were again compared and found to be in good agreement. With a sufficient description of the magnetic fields, we can use the same environment as a foundation to simulate the motion of fast ions, which as we will see, have greatly different dynamics than the electrons whose transport was previously discussed.

References

1) J. K. Anderson et. al., Nuclear Fusion **44**, 162 (2004)

2) D. Schnack, JCP **70**, 339 (1987)

3) Y. Jiang, N. E. Lanier, D. L. Brower, Rev. Sci. Instrum. **68**, 703 (1999)

4) D. Craig et. al., Rev. Sci. Instrum. **72,** 1008 (2001)

5) T. M. Biewer et. al., Rev. Sci. Instrum. **74,** 1649 (2003)

6) H. T. Evensen et. al., Rev. Sci. Instrum. **66,** 845 (1995)

7) J. C. Reardon et. al., Rev. Sci. Instrum. **72**, 598 (2001)

8) A. B. Rechester, M.N. Rosenbluth, Phys. Rev. Lett. **40** 38 (1978)

9) G. M. Zaslavsky and B. V . Chirikov, Usp. Fiz. Nauk **14**, 195 (1972) [Sov. Phys. Usp. **14**, 549 (1972)]

10) T. M. Biewer et. al., **91**, 4, 045004 (2003)

4 Fast Ion Confinement Experiment

To study the question of fast ion confinement in the RFP, we used a tangential (to the magnetic axis) neutral beam injector, described in Section 4.1, as a tool to introduce a population of fast ions into the RFP plasma. Typical waveforms and values of the beam current are discussed in Section 4.2. These are the first in a series of diagnostics needed to quantify the number of beam ions we are introducing in the plasma. In Section 4.3 we present a measurment of the divergence of the beam as it is focused to enter the vessel through a small diameter vacuum port. Section 4.4 describes a calorimeter that was used successfully to measure the total beam current entering MST. The neutral to ion fraction of the beam was measured with an experiment utilizing charge exchange emission and is presented in Section 4.5.

The results of the fast ion confinement experiment begin in Section 4.6 where the neutron emission from deuterium beam ions fusing with deuterium bulk plasma ions is measured. Section 4.7 shows the procedure used to calculate the beam particle confinement time in a 400 kA standard RFP discharge, and is one of the key results in this thesis. The confinement under varying levels of magnetic fluctuations is studied in Section 4.8 (increased magnetic fluctuations during a sawtooth crash) and Section 4.9 (decreased magnetic fluctuations during PPCD). The role of the beam injection vector relative to the background magnetic field is explored in Section 4.10 (poloidal magnetic field direction reversed) and Section 4.11 (radial injection).

4.1 Injector Construction

The neutral beam injector used on MST has three basic components, the plasma source (Section 4.1.1), ion accelerating and focusing grids (Section 4.1.2) and a neutralizing chamber (Section 4.1.3). Each of these will be discussed in turn to give the reader an understanding of the neutral beam formation process.

4.1.1 Plasma Source

Fig. 4.1 shows a schematic view of the plasma chamber[1]. First we have a plasma emitter consisting of an arc source that creates a low-temperature plasma. The plasma then expands into a small vacuum chamber, which is a cylindrical chamber of approximately 28 cm in diameter by 24 cm deep. Along the perimeter of the expansion chamber is a multipole cusp field provided by an array of permanent magnets that provide additional focusing of the expanding plasma. The expanding plasma acts as a divergent ion beam (charge balanced by electrons) that strikes a spherical electrode. The electrode is gridded by the presense of small apertures that allow ions to pass through. Successive grids are at lower potentials, stripping the ions of their electrons and focusing the ions into a converging beam.

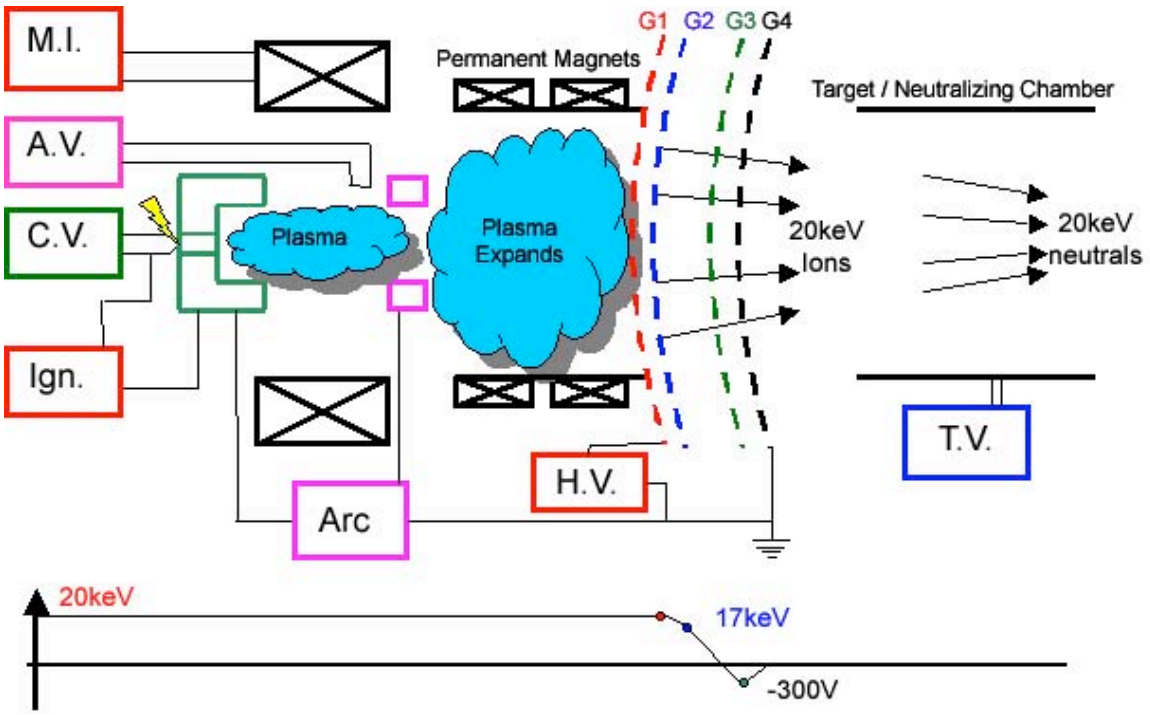


Figure 4.1 Layout of neutral beam injector. The lower portion denotes what potential the components are held at to accelerate the ions and suppress electron current.

An optical trigger starts the process

At –7.50 ms the magnetic isolation current turns on

At –650 μs target valve and anode valve open

At –450 μs the cathode valve opens

At –400 μs the arc supply comes on

At –100 μs the high voltage turns on

At 0 μs the ignition spark occurs and the plasma forms

Magnetic isolation – In the plasma emitter, to restrict the flow to the axial direction, an axial magnetic field is needed. This is provided by a solenoidal current loop around the plasma chamber. The current rise rate is about 1.3 A/ms, which for 7.5 ms of rise gives about 10 A in the coil.

Target Valve – Gas is puffed into the target chamber to provide sufficient neutral density for the accelerated ions to neutralize and become fast neutrals, which enter the plasma.

Anode Valve – Provides even fueling for the initial plasma formation

Cathode Valve – Provides fueling near the ignition spark where the plasma first forms.

Arc Supply – 300 V applied across the plasma chamber, draws a current of 960 A.

High Voltage – 20 kV Used to draw ion current from the plasma and into the acceleration grids.

Ignition – A 1 kV filament that emits electrons to start the initial cascade for plasma formation.

4.1.2 Ion Optics

The expanding plasma source reaches a spherical electrode, which is perforated by small apertures to allow transmission. There are three other grids, ~20 cm in diameter, at successively lower potentials, as shown in the electric potential plot in the lower section of Fig. 4.1, that draw ions off and focus them. We have four grids; Source (G1), Gradient (G2), Suppression (G3), Reference (G4). Between grids G1 and G2, there is a 3 kV drop that separates the ion and electrons from the plasma source. Between grids

G2 and G3, the ion experiences an energy gain of approximately 17 kV and this is where it is most strongly focused and accelerated. Between grids G3 and G4, we see from Fig. 4.1 that the potential is negative, and this is to provide a potential hill to electrons further down the beam line. Any free electrons would be accelerated towards the rear grids and interfere with the ion beam by either combining with a passing ion or affecting the space charge distribution between grids G1 and G2.

4.1.3 Neutralization Chamber

The focused ion beam then passes through a chamber that has been puff filled with neutral gas. The ion beam charge exchanges with the neutral gas and becomes the neutral beam that enters the plasma. One important question is the efficiency of the process. If there were too little gas in the chamber, the beam would be mostly ions entering MST. If too much gas is used, scattering would become more pronounced and affect the focusing of the beam.

While an accelerated ion may become neutral, it may also once again be ionized through collisions and thus enter MST as an ion. The equations that describe the processes in the target chamber are

$$\begin{aligned} \frac{dn_f^+}{dl} &= -\sigma_{CX} n_f^+ n_T^0 + \sigma_{EXC} n_f^0 n_T^0 \\ \frac{dn_f^0}{dl} &= \sigma_{CX} n_f^+ n_T^0 - \sigma_{EXC} n_f^0 n_T^0 \end{aligned} \tag{4.1}$$

$$\frac{n_f^0}{n_f^0 + n_f^+} \xrightarrow{l=\infty} \frac{\sigma_{CX}}{\sigma_{CX} + \sigma_{EXC}} \tag{4.2}$$

where n_f is the fast particle density, n_T is the density of gas in the target chamber, the sigmas are the cross-sections for charge-exchange and impact ionization. The superscripts (+ and 0) denote charge state. The first term is the charge exchange between the fast particles and the background. The second term is the contribution from impact ionization of fast neutrals. Since the scattering cross section is much lower than the charge exchange or excitation cross sections, we can assume that for a modest sized target chamber that scattering will not play a strong role.

This system of equations has an equilibrium value, which is determined experimentally to be the "equilibrium fraction" and is given by Eqn. 4.2. The real value is a state where the two competing effects have balanced each other, and since the cross sections are velocity dependent, the equilbrium fraction is as well. This is the steady state population ratio between the number of ions and neutrals. It is often tabulated in atomic processes databases[1] and is shown in Fig. 4.2. For a deuterium beam passing through neutral deuterium, the value is given as ~87% neutral. We measure this experimentally (Section 4.5) and attain very close agreement.

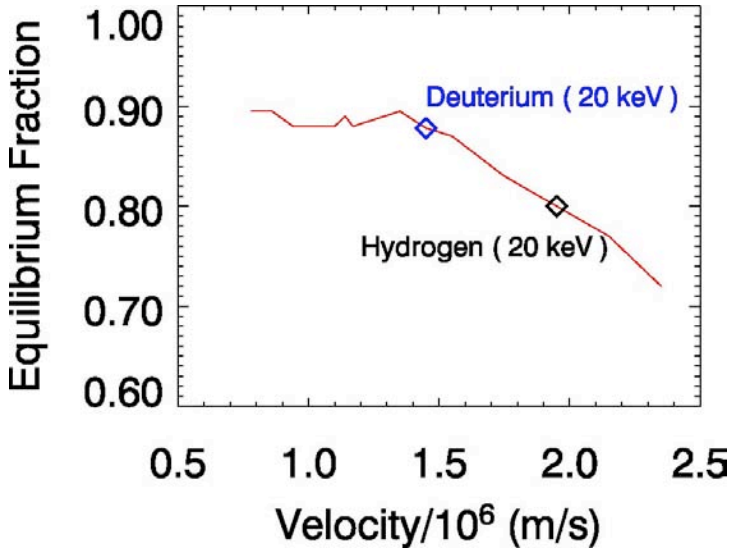


Figure 4.2 Equilibrium fraction. The value for the deuterium beam at 20 keV is shown by the blue diamond (~87%) and the value for the hydrogen beam at 20 keV is shown by the black diamond (~80%).

4.2 Neutral Beam Voltage and Current Diagnostics

Fig. 4.3 shows a typical time trace for the accelerating voltage on the source grid. The plasma current is shown in Fig. 4.4 and is approximately 850 A. The ion current is shown in Fig. 4.5 with the dashed line a typical value when using deuterium instead of hydrogen. If we consider the Child-Langmuir law,

$$j = \frac{4}{9}\varepsilon_0 \sqrt{\frac{2e}{m}} \frac{V^{3/2}}{d^2} \tag{4.3}$$

which governs the space-charge limited current (or the ion current flowing from the plasma to the source grid), we see that the current is inversely proportional to the square-root of the particle mass, so would be about 40% higher for hydrogen vs. deuterium.

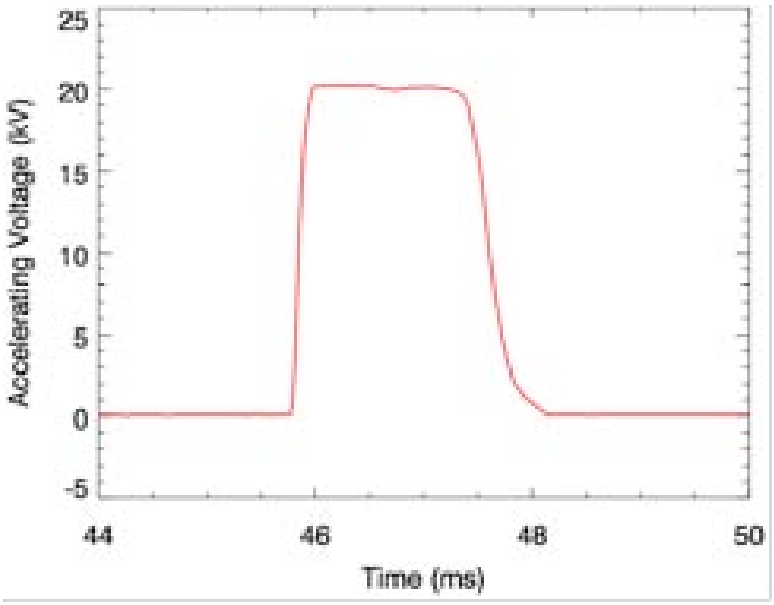


Figure 4.3 A typical time trace of the voltage on the Source grid (G1).

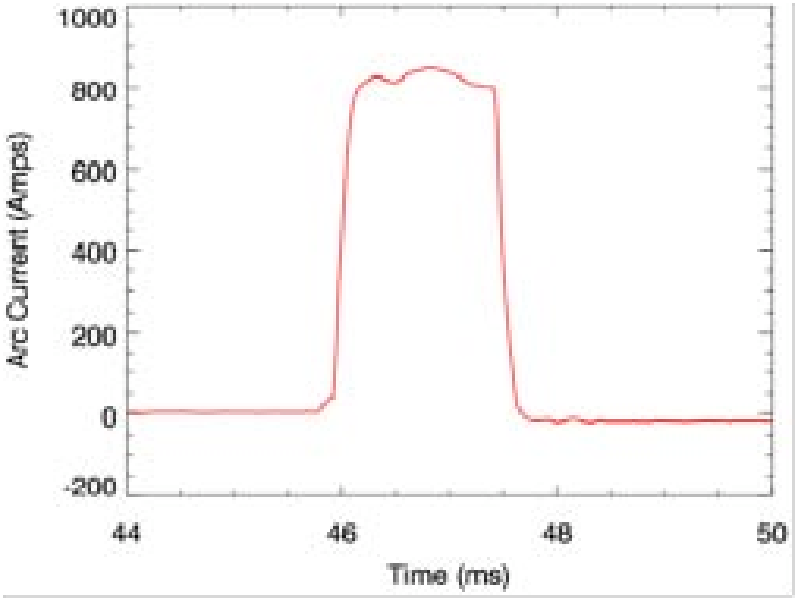


Figure 4.4 The plasma current inside the arc discharge chamber.

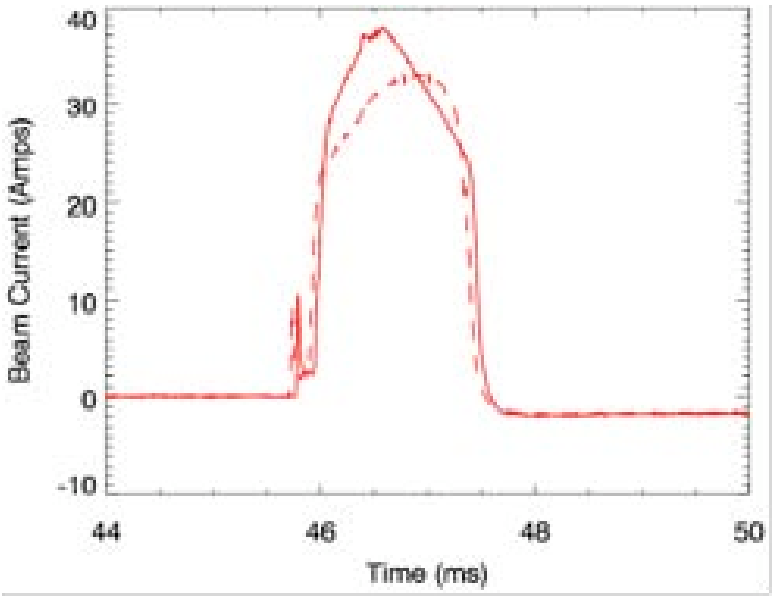


Figure 4.5 Ion beam current as measured by a shunt resistor between G1 and G4. The solid curve is for hydrogen and the dashed is deuterium. The small initial spike is electrical noise.

There is a difference in the shape of the beam current traces, which we believe is due to the ion optics being optimized for hydrogen, although of course the heavier deuterium will affect the time response of the beam as well.

4.3 Optical Measurement Of Beam Divergence

To quantify the degree of focusing of the neutral beam, which is needed to estimate the amount of neutral current into MST, we use a perpendicular port between the target chamber and the vacuum vessel as shown in Fig. 4.6.

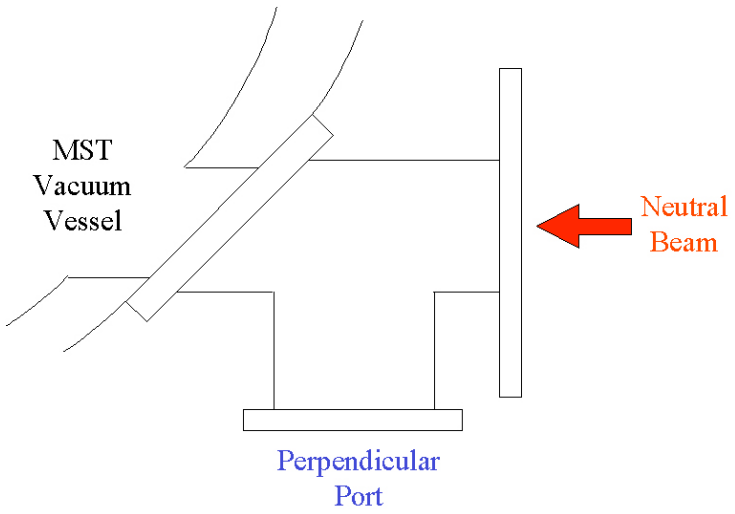


Figure 4.6 Perpendicular port, 3.375" ID. Used to view/access the neutral beam before it enters MST.

H-alpha (or D-alpha) light has a wavelength of ~ 656 nm and this is in the visible spectrum. A digital camera with a slow shutter speed was used to view the light profile across the beam cross section (Fig. 4.7 Left). Using standard photo-analysis software we reconstructed the density profile of the beam. The Gaussian shape of the profile is shown in Fig. 4.7 (Right) and by measuring the changes in the width of beam, we estimate the divergence to be about 0.02 radians. This is close to the manufactured specifications.

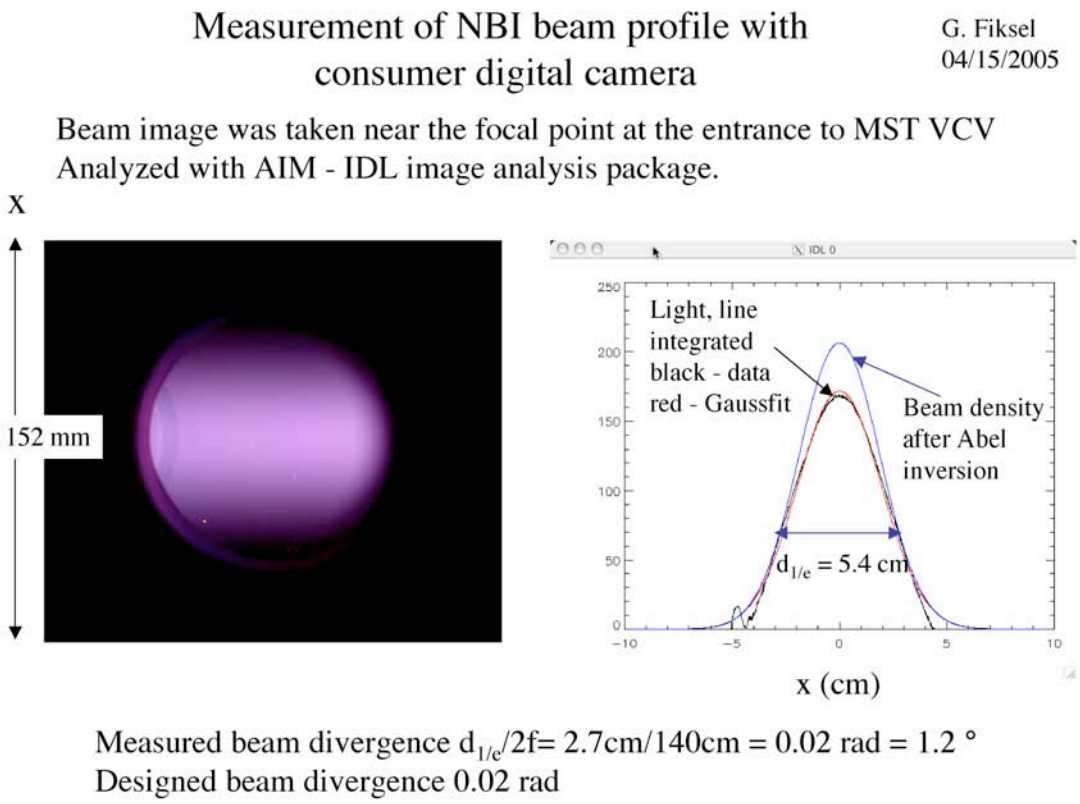


Figure 4.7 Image of NBI taken with digital camera. Abel inversion of light profile provides estimate of beam divergence.

4.4 Calorimeter Measurement Of Neutral Particle Flux Into MST

While we can estimate the neutral current into MST from the ion current diagnostic (Fig. 4.5) and the equilibrium fraction (Fig. 4.2), the perpendicular port gives the opportunity to take a direct measurement of the beam properties just before it enters the vacuum vessel. To do this, a calorimeter was constructed and is shown in Figs. 4.8 and 4.9. The basic principle of operation is that the particle flux (neutrals + ions) strike three copper fins, the kinetic energy being transferred to heat in the copper. Each fin then experiences a temperature increase as it equilibrates through the relatively thin

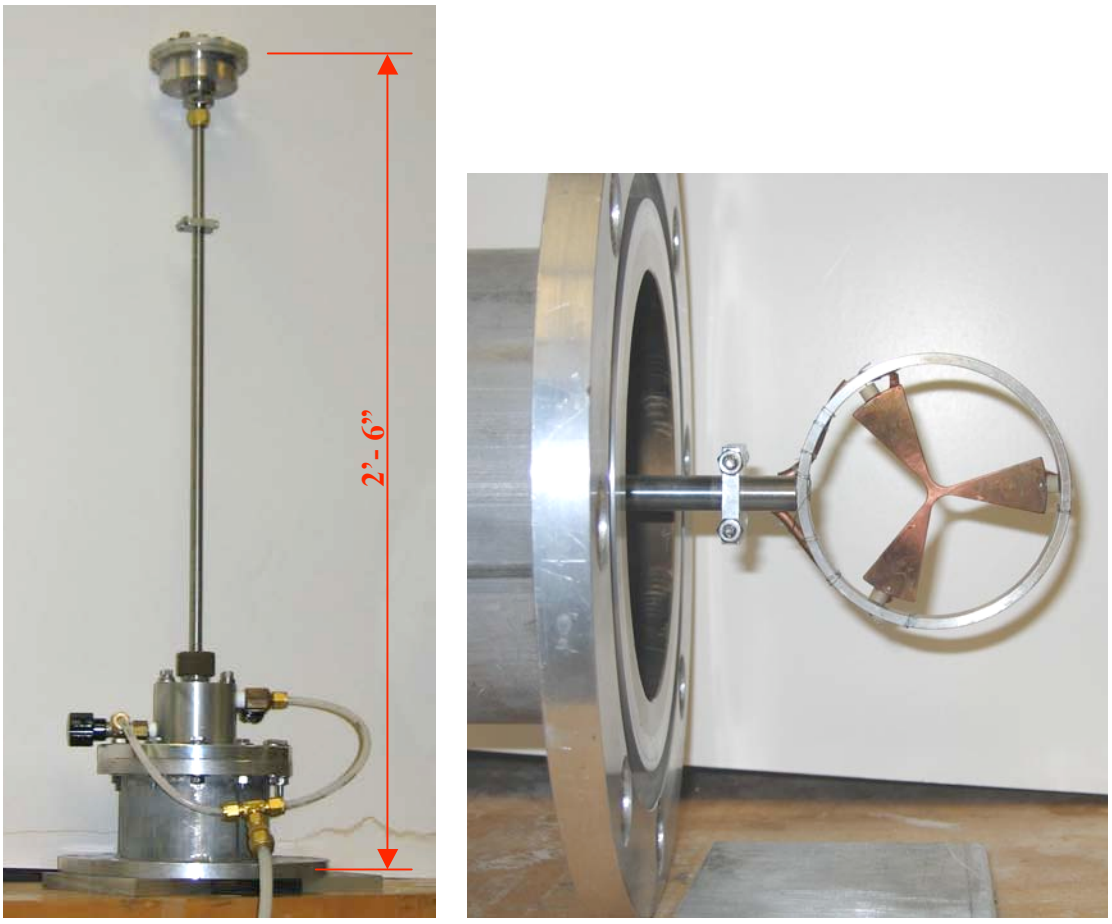


Figure 4.8 Calorimeter probe. Elevation (left) shows the probe housing, neck and cup. Two-valve assembly to roughing line minimizes gas leakage into MST. Beam-facing view (right) shows copper fins and support ring (stainless steel)

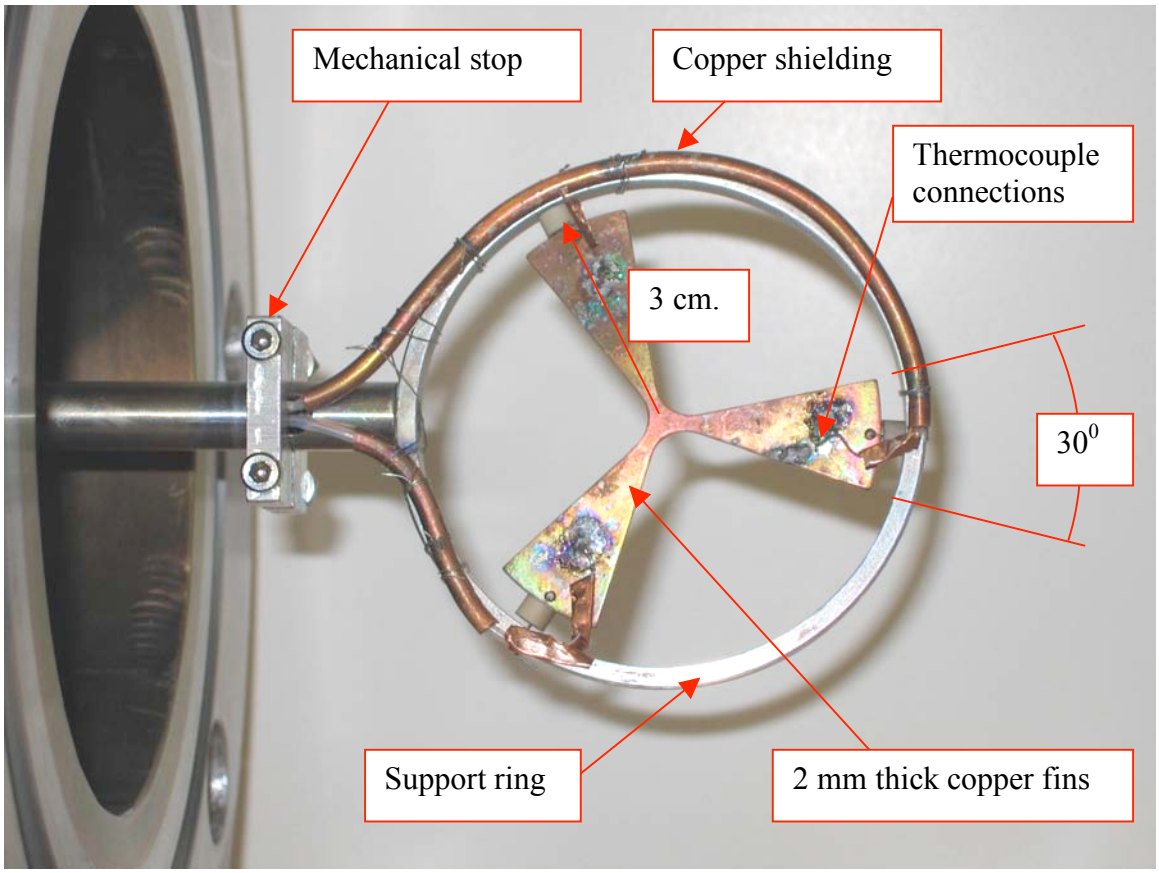


Figure 4.9 Lee side of calorimeter probe. Shows solder positions of thermocouples and copper shielding of thermocouple wires.

thickness of the fin. The temperature is monitored as a voltage by thermocouples on the lee side of the copper fins.

We first check the beam alignment by means of looking for symmetrical heating in all three fins. Once we are convinced the beam is aligned, we consider the interpretation of the recorded temperature increase. The physical dimensions and geometry of the copper plate were chosen to have the plates mostly isolated from each other thermodynamically. The thinness of the copper was selected to give a reasonable temperature rise, as well as a fast time response as compared to any parallel heat

conduction. The thin junction point resists heat flowing from one fin to the other, in the event of asymmetrical heating. As stated earlier, if we first align the beam, then there would be no heat transport from one fin to the other as there would be no temperature gradient between fins.

The temperature rise can be simply expressed as

$$\Delta T = \frac{\Delta Q}{c_v m_c} \tag{4.4}$$

where c_v is the specific heat of copper (0.383J/g*c) and m_c is the mass of the copper fins (10.1g). In SI units, the heat into the copper is,

$$\Delta Q = N\frac{1}{2}m\mathrm{v}^2 = \frac{I\tau}{e}eV_G . \tag{4.5}$$

where I is the beam current we want to measure, τ is the beam on time, and V_g is the accelerating voltage of the beam.

The response of the thermocouple (Type J Iron/Constantan) is very linear over the ranges of temperatures that we limited ourselves to (20 C to 175 C). The voltage/temperature ratio in this range is

$$\sigma = 0.05mV/C^{\circ} . \tag{4.6}$$

Combining Eqns. 4.4 - 4.6, we have a formula for the beam current as a function of the measured voltage response of the thermocouples,

$$I_b = 4\frac{c_v m_c}{V_g \tau\sigma}\Delta V . \tag{4.7}$$

In Eqn. 4.7 ΔV is in millivolts and the leading factor of 4 comes from the fact that the fins only cover ¼ of total beam area.

For 19 kV, and a beam time of 1.42 ms (see Section 4.5), we get I_b = 19 A through the radius encompassed by the fins. Assuming a Gaussian profile,

$$N(r) = N(0)e^{-r^2/r_0^2}. \tag{4.8}$$

With width r_0 = 2.7 cm, (from light measurement) the total beam current would be

$$I_{Total} = \frac{1}{\left(1 - e^{-(a/r_0)^2}\right)} \times \frac{1}{0.9} \times I_{meas} = 28\,A, \tag{4.9}$$

using

$$\int_0^a e^{-r^2/r_0^2} 2\pi r dr = \pi r_0^2 \left(1 - e^{-(a/r_0)^2}\right), \tag{4.10}$$

where *a* is the radius of the fins (3 cm). The 0.9 in Eqn. 4.9 is because we only have 90% of the beam at full energy, the rest we neglect for simplicity.

However, current through the port is less than the total, so we also would like to know how much current is going into MST. To calculate that, we use 4.9 again, but the leading factors becomes $I_{MST} = \frac{\left(1 - e^{-(b/r_0)^2}\right)}{\left(1 - e^{-(a/r_0)^2}\right)} \times I_{meas} = 25\,A$,

where again *a* is the radius of the fins and *b* is the radius of the port (4.3 cm.). As seen in Fig. 4.5, a rough average beam current (total) is around 28 A, which is in agreement with the previous measurement from the calorimeter (Eqn. 4.9).

4.5 Estimate of Neutral Fraction Via Charge-Exchange Light

In Section 4.4 we established that about 25 A of total current is entering MST. We also have from Section 4.1.3 that the equilibrium fraction is 87% so we predict that there is 22 A of neutral current entering MST. One way to check this is to use the fact

that the light emission is proportional to the ion + neutral density. If we could separate the ions from the neutrals and take light measurements based on this then the ratio of the two light signals would indicate the neutral fraction.

In MST we can pulse the toroidal magnetic field with no plasma. The ion component of the beam has a total Larmor radius of about 0.65 m in a 450 G field. This is sufficient for the beam ions entering MST to be bent away from beam neutrals and strike the vessel wall. We have vertical viewing ports that look through the poloidal cross section where the beam is approximately tangent to the magnetic axis. By using photo-diodes on the central port to intercept the light emission from the beam, we can compare the beam density with and without ions and thus get the neutral fraction. The setup is shown in Fig. 4.10.

A typical trace of the strongest signal is shown in Fig. 4.11. It roughly conforms to the neutral beam current but if comparing the amplitudes of the beam off/on traces, one finds a ratio of 0.88. This is in good agreement with the cited value of 0.87 for the equilibrium fraction of a hydrogen beam in neutral hydrogen.

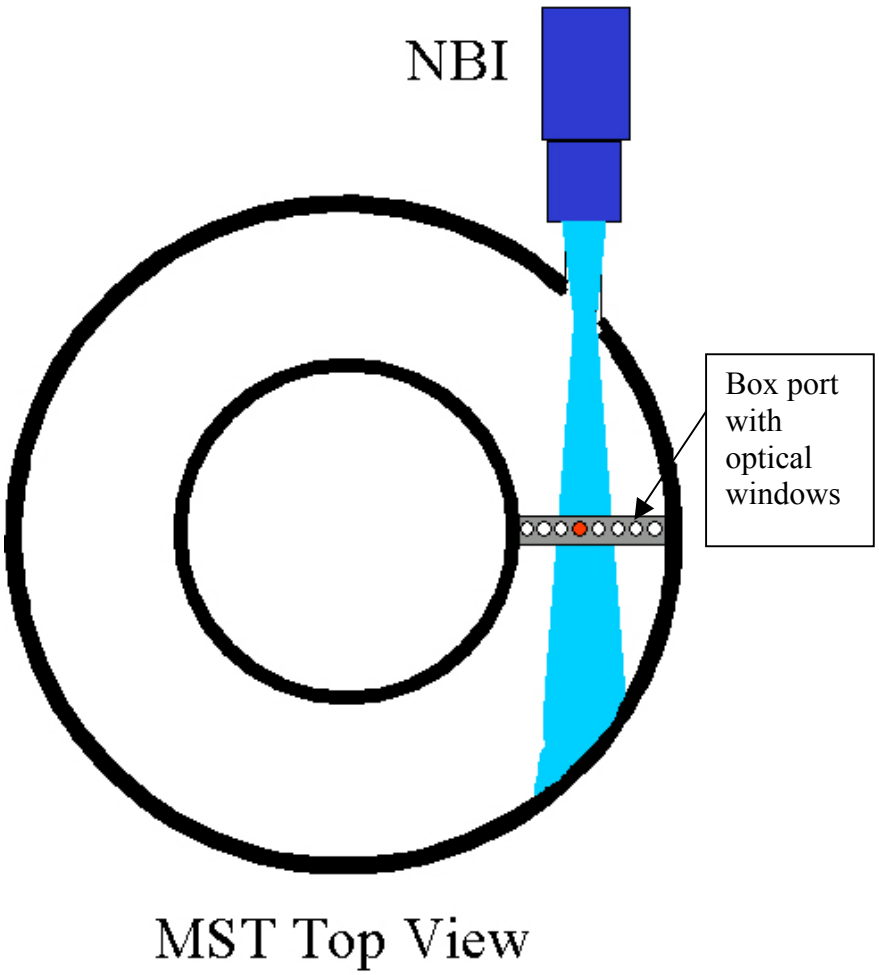


Figure 4.10 Setup of beam/gas emission experiment. The light from the beam is observed through a viewing window perpendicular to the beam chord. The red circle on the box port denotes where the signal was maximum.

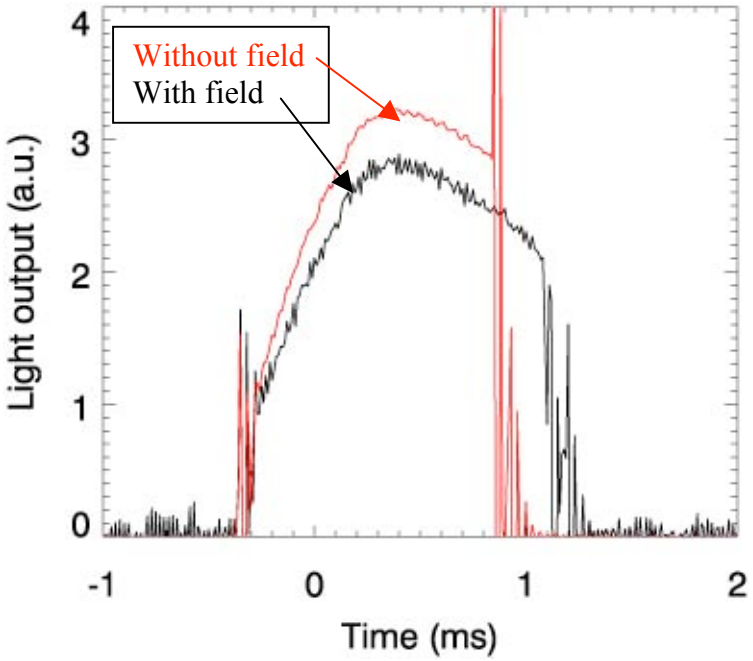


Figure 4.11 Light signal from photodiode during NBI (D_2) into gas (no plasma). The red trace is with no vacuum magnetic field, the black solid trace is with a background toroidal field of 450 G at the wall. The red trace was cut short by a beam arc, but prior to that the ratio of the signals was steady.

4.6 Neutron flux during NBI

When we inject the beam, the neutral atoms ionize and then interact with the background plasma through Coulomb collisions. With injected deuterium however, fusion reactions between the beam and the background ions can occur. This results in a significant neutron flux that can be directly related to the fast ion population. The temporal behavior of the neutron flux is then related to the confinement of the fast ion population.

4.6.1 Neutron Detector Construction

A drawing of the experimental setup is shown in Fig. 4.12. The neutrons from beam/plasma interaction pass quite readily through the aluminum shell of MST. They then enter the plastic scintillator within the neutron detector. The neutrons undergo elastic collisions with hydrogen atoms, transferring energy to them. The recoil protons then excite the phosphoring and scintillating material, which then radiate their energy in the form of photons. The design of the detector was to have approximately 1 mean free path for the neutrons to interact in, and a reflective layer was placed around four sides of the 5" square scintillator, which had the effect of doubling the captured photon output.

This photon is captured by the photomultiplier tube (PMT) and converts the incoming light to a current pulse. The amplification of the electron current by the photomultiplier tube then produces a voltage that is proportional to the incident neutron flux. The scintillator is also sensitive to hard x-rays, which we shield out by putting a layer of lead around the scintillator and PMT.

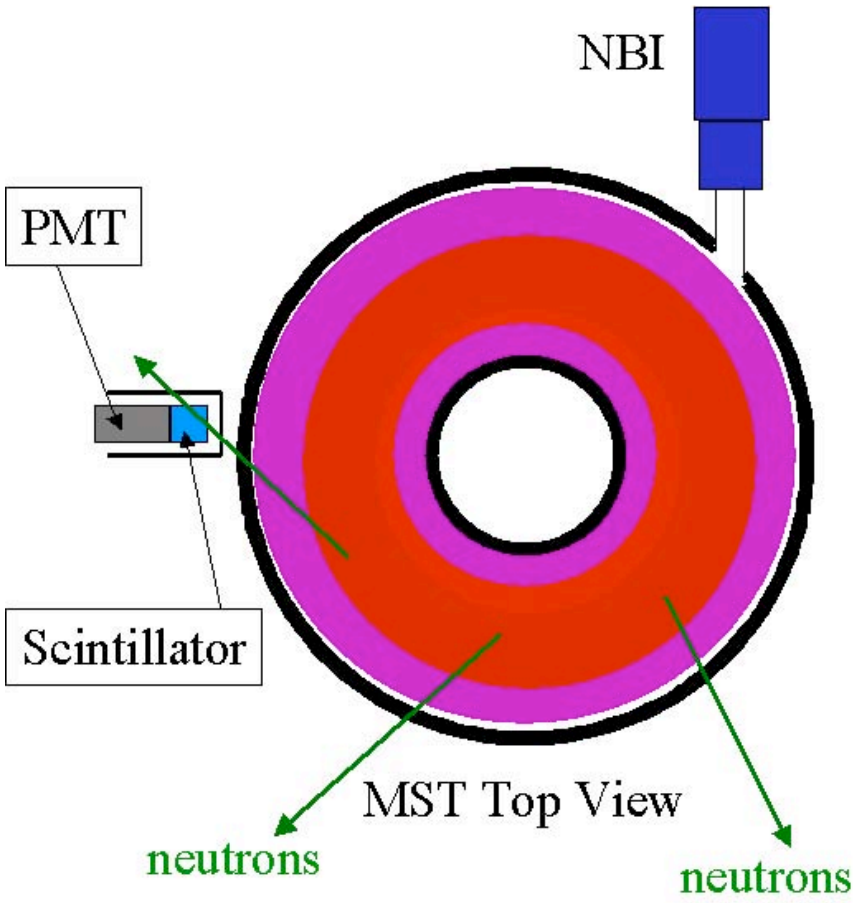


Figure 4.12 Setup of neutron detector. Fast ions fuse with the background plasma and the product neutrons are detected outside the vessel.

The neutron detector was calibrated with a small neutron source, giving a response of approximately 7×10^9 neutrons / second / Volt when the detector is in current mode. The calibrated value also agreed well with the expected signal from the calculated fast ion fraction, taking into account estimates of ionization and the observable fraction of the total neutron flux out of the vacuum vessel. By adjusting the gain in the apparatus, we can increase the sensitivity to measure neutron flux during normal plasma operations and thus get an indication of the plasma ion temperature. This is beyond the scope of this

thesis, but illustrates that the neutron detector is a useful plasma diagnostic even beyond its use during NBI.

4.6.2 Measured Neutron Flux

The D-D fusion reaction,

$$\begin{aligned} D + D &\Rightarrow He^3(0.82MeV) + n(2.45MeV) \quad 50\% \\ D + D &\Rightarrow T(1.01MeV) + p(3.02MeV) \quad 50\% \end{aligned} \tag{4.11}$$

has a cross section[2] shown in Fig. 4.13 which is extremely sensitive to the fast ions energy. Also in Fig. 4.13 we contrast the reaction rate of the fast ions (20 keV) with the background plasma, and the plasma fusing with itself. For a typical plasma temperature of 400eV, the reaction rate is approximately 10^{-31} m^3/s, whereas the fast ion rate is about 10^{-26} m^3/s. The expressions for the total neutron flux are given by

$$\begin{aligned} \Phi_{plasma} &= \frac{1}{4}\int_V < \sigma v >_{D-D} \; n_i^2(r)dV \\ \Phi_{fi} &= \frac{1}{2}\int_V \sigma_{D-D} v_{fi} n_{fi}(r) n_i(r) dV \end{aligned} \tag{4.12}$$

where in the top equation $< \sigma v >_{D-D}$ is the maxwellian averaged reaction rate for an ion species with a thermal velocity of v and the plasma ion density is n_i. In the lower equation $\sigma_{D-D} v_{fi}$ is the reaction rate for a fast ion to fuse with an essentially stationary background ion and the fast ion density is written as n_{fi}. Even though the fast ion density is (assuming all ions are confined perfectly) a factor of 100 lower than the background plasma, the disparity in the reaction rates more than compensates for this and as a result the neutron flux from fast ions fusing with the background is dominant over the background alone.

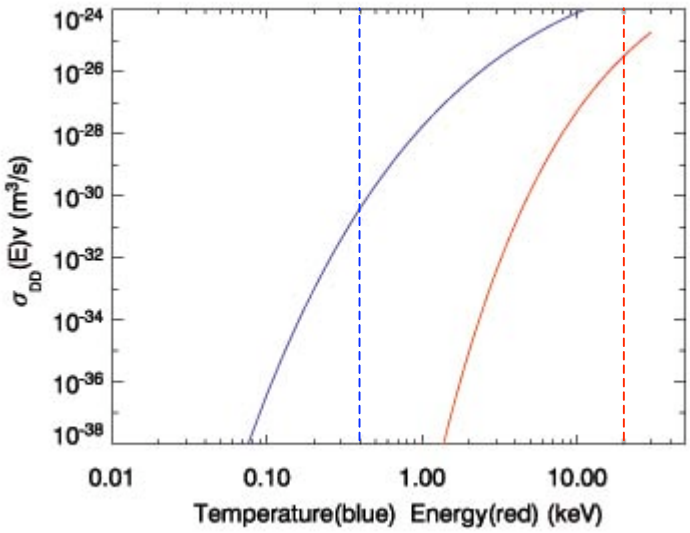


Figure 4.13 D-D fusion reaction rates. The blue curve is the Maxwellian averaged rate, with the dashed line indicating our typical plasma temperature. The red curve is for a test particle of the specified energy colliding with a deuterium at rest. The disparity in the rates suggests a high signal to noise ratio for NBI neutrons.

We observe that the neutron signal persists for tens of milliseconds after the injector is turned off. Without doing any analysis, we see that this is a favorable result for fast ion confinement, but we will now model the neutron signal and get a quantitative estimate of the fast ion's confinement time.

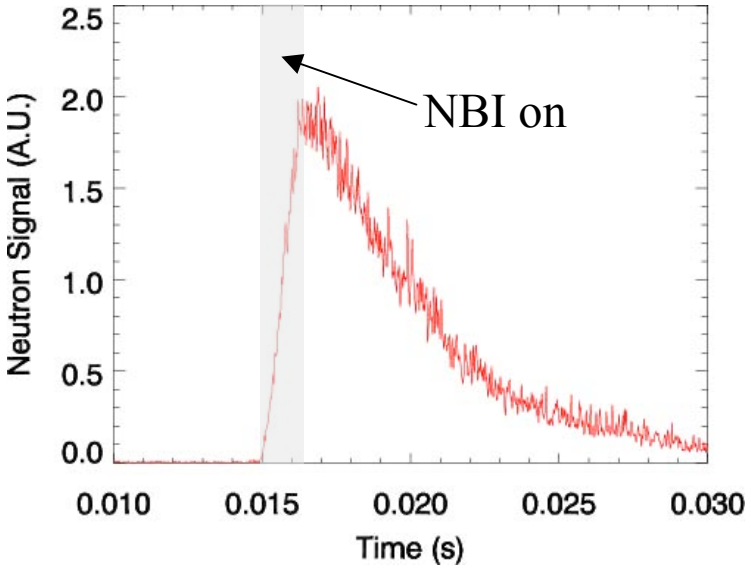


Figure 4.14 Neutron flux observed during NBI. A linear growth during injection is followed by a slow decay. The long timescale suggests favorable ion confinement.

The neutron flux during NBI into a 400 kA standard plasma is shown in Fig. 4.14. We see that there is a linear rise during the beam on phase, followed by a decay of the neutron signal after the beam has turned off. We should also note that the neutron flux seen before the signal is within the noise of the detector, as expected given the reaction rates seen in Fig. 4.13.

4.7 Estimate Of Fast Ion Confinement Time

To extract the fast ion confinement time, we begin by writing the neutron flux from MST as

$$\Gamma(t) \propto \int_V \sigma_{D-D}(\mathrm{v}_{fi})\mathrm{v}_{fi}(t)n_{fi}(r,t)n_i(r)dV \tag{4.13}$$

and also the evolution of the fast ion population as

$$\frac{dn_{fi}(r,t)}{dt} = S(r,t) - \frac{n_{fi}(r,t)}{\tau_{fi}} \tag{4.14}$$

where the fast ion quantities are denoted by '*fi*' and the plasma ion density is given as n_i. The proportionality is due to the reduction of the neutron signal by scattering from the vessel and its various components in the process of arriving at the detector, as well as the solid angle subtended by the detector itself.

In the fast ion population equation (Eqn. 4.14), the source term represents the linear increase in time of the fast ion population during the injection phase, but the radial dependence is a function of the injection geometry and the ionization of the beam along the injection chord. A loss term is included and characterized by τ_{fi}. This loss time is an aggregate of all loss mechanisms in the system, of which the primary ones are CX (fast ions becoming neutrals), and stochastic losses. All particle profiles are considered static, no particle diffusion is allowed. For this analysis; the time dependence only comes in when considering the rate at which ions enter or leave the system. The assumption of a static plasma ion profile is justified from the steady state condition. The assumption of negligible fast ion radial evolution is addressed below.

The energy dependence of a fast ion due to slowing on a background plasma[3] is

$$\frac{1}{E_{fi}}\frac{dE_{fi}}{dt} = -\frac{Z^2 e^4 n_e m_{fi}^{1/2} \ln(\Lambda)}{4\sqrt{2}\pi\varepsilon_0^2 m_e E_{fi}^{3/2}}\left(\frac{4}{3\pi^{1/2}}\left(\frac{m_e}{m_{fi}}\frac{E_{fi}}{T_e}\right)^{\frac{3}{2}} + \sum_i \frac{m_e}{m_i}\frac{n_i Z_i^2}{n_e}\right). \qquad (4.15)$$

The first term is energy loss to electrons and the second term is energy loss to ions of varying species. For particles with energy greater than ~15 T_e, the majority of the energy is given to electrons. This is very much the case with our 20 keV beam going into a 400 eV plasma. In this case, the energy loss rate is inversely proportional to the fast ions mass, which is why hydrogen should be used for heating.

To avoid confusion, a few remarks about the radial dependence in Eqns 4.13 and 4.14 are in order. While there is a definite radial structure to the fast ion population and the plasma ion population, we can assume that they do not change in time. The relatively flat plasma ion density in the core (Section 5.7) mitigates the effect of any small-scale changes in the fast ion distribution. A change in the radial distribution of the fast ions would in principle affect the intercepted neutron flux. However, such a correction is very small considering that the distances involved render the change in solid angle to the detector to also be very small. We then included the radial dependence in the equations for completeness, but it can easily be taken to be separable from the time variable, which is our interest.

Fig. 4.15 shows this modeling applied to the experimental curve of Fig. 4.14. Here we have three loss times; infinity (no losses), 20 ms, and 1 ms. The 20 ms curve is the best agreement to the experimental curve. The 1 ms curve greatly underestimates the observed signal[4]. Furthermore, comparing the experimental curve to the infinite confinement time curve, we see that they are very close. The decay in the signal is not due to particle loss, but rather the change in the D-D cross section as described by the fast ions classical slowing down rate given in Eqn. 4.15. This is also important because it places an upper bound on the core neutral density. This will be discussed more in Section 4.10.

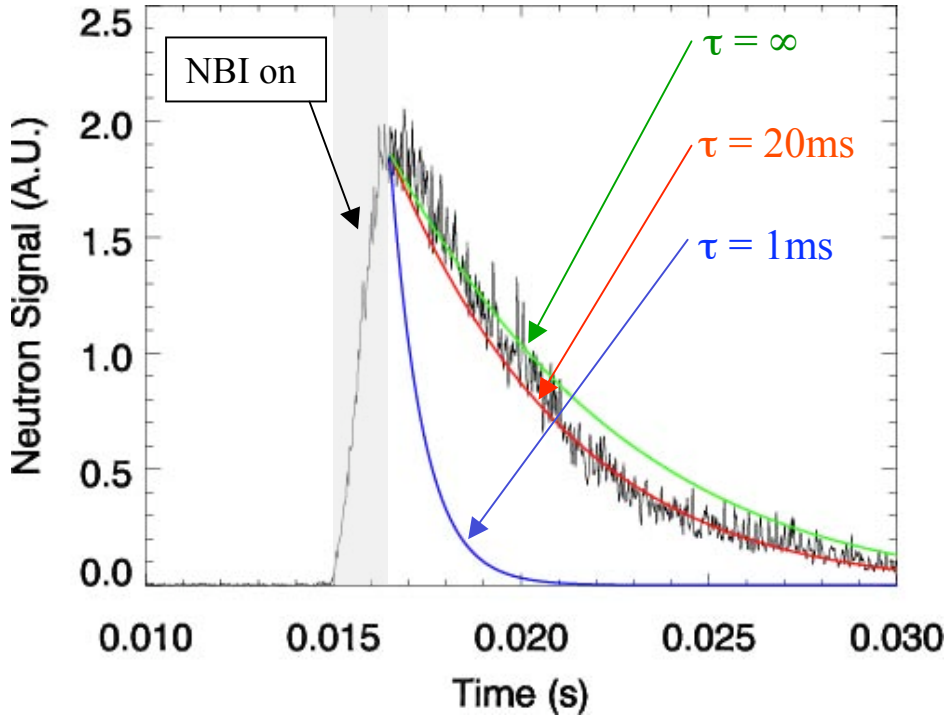


Figure 4.15 Calculation of fast ion confinement time. Perfect confinement (green) and slightly overestimates the experimental signal. The 20 ms curve (red) is a good fit but slightly underestimates, and the simple stochastic ion estimate (blue) of 1 ms is clearly not applicable.

4.8 Fast Ion Confinement During A Sawtooth Crash

We know that field line stochasticity, which scales with the magnetic fluctuations, impacts particle transport. To examine the role of fluctuations on fast ion confinement, we can consider what happens during a "sawtooth crash". The temporal behavior of the magnetic fluctuations for a single mode is shown in Fig. 4.16. Sawteeth, being ubiquitous in the experiment are easy to see in terms of the neutron signal. Fig. 4.17 shows a single beam shot, but this time a sawtooth crash occurs 2 ms after the beam is

off. There is the characteristic slow decay then at the time of the crash there is a 50% drop in the signal. After the sawtooth event, the slow decay again resumes.

The short duration of the sawtooth crash allows us to treat the fast ions energy as constant. Therefore, the decrease in the neutron signal must be due to particle losses, rather than the reaction rate changing as a result of the ion slowing down. The steepness of the drop makes a quantitative estimate of the confinement uncertain, but a best fit curve to the experimental signal results in a confinement time of ~0.4 ms.

In Fig. 4.15 we show the neutron signal for a five shot ensemble. Approximately 5 ms after the beam a small decrease in the signal is observed. This is a sawtooth crash averaged into the plotted signal. We cannot fit a curve to a portion of the signal that has sawteeth in them, because it would underestimate the confinement time.

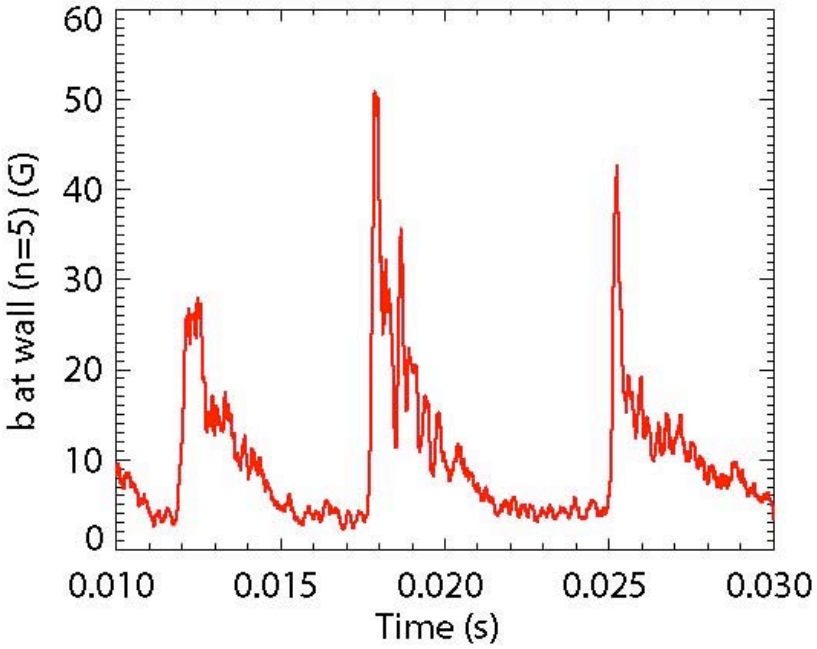


Figure 4.16 Magnetic fluctuation amplitudes increase by a factor of three (or more) during a sawtooth crash.

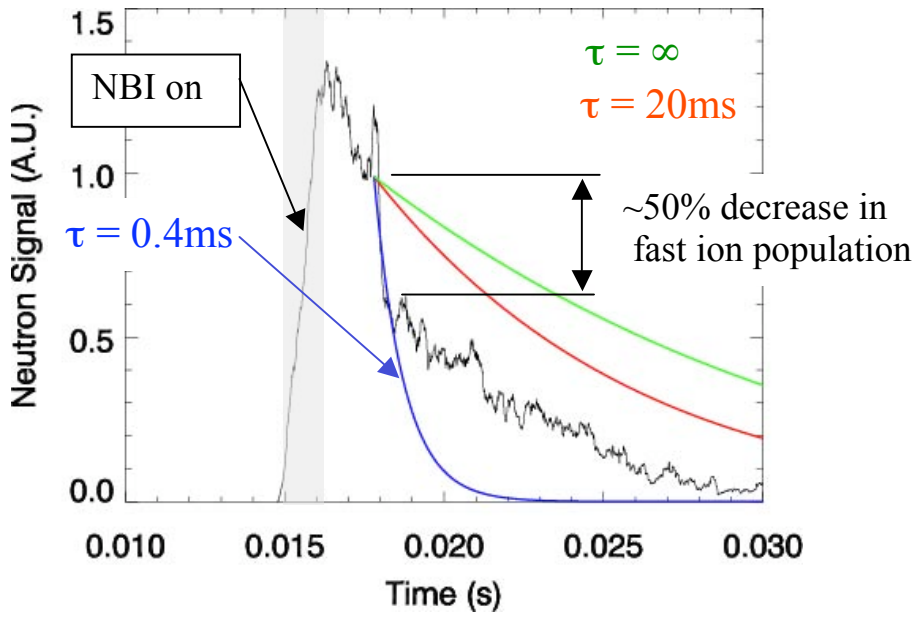


Figure 4.17 Neutron signal during a sawtooth crash. The neutron flux decrease is commensurate with stochastic diffusion of fast ions in a high-fluctuation plasma.

4.9 Fast Ion Confinement During PPCD

We can also consider the effect of reduced fluctuations on fast ion confinement. The effect of PPCD on the magnetic fluctuations (as measured at the plasma edge) is shown in Fig. 4.18 Fig. 4.19 shows the neutron signal taken during a 400 kA PPCD shot. We first note the decrease in neutron flux is slower than in the standard case, and could be due to good confinement. However, the electron temperature in the standard case is about 350 eV whereas in the PPCD case it is about 800 eV. The 3/2 dependence on the electron temperature in the slowing down time (Eqn. 4.15) shows that the fast ion loses energy at a much slower rate in the hotter plasma. Using the temperature of 800 eV yields a minimum 30 ms confinement time as shown in Fig. 4.19. The modest confinement improvement over the standard case suggests that further reducing fluctuations has a limited effect on confinement once the fast ions are largely non-stochastic.

Even though the fast ion confinement time is not markedly improved, if NBI heating is desired, then the energy confinement time of the plasma must be as large as possible. PPCD is very useful as it results in upwards of a ten-fold increase in energy confinement time[5] and so would be integral in any heating experiment.

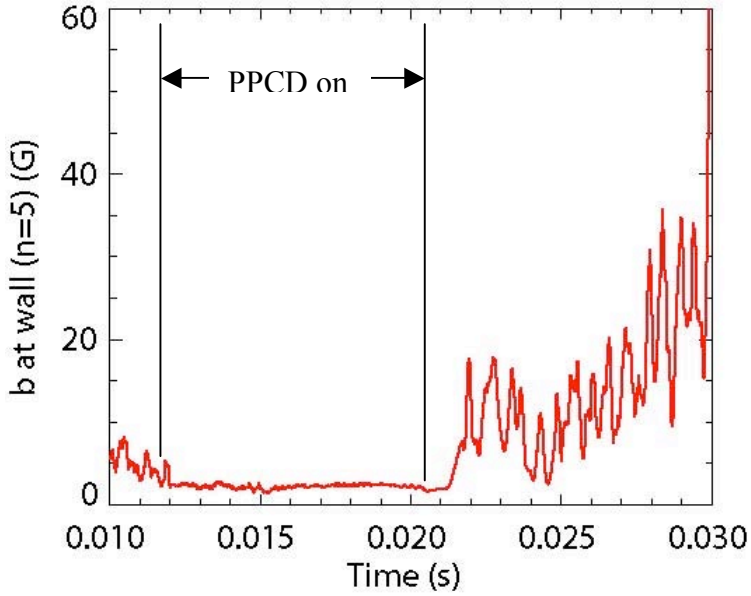


Figure 4.18 Magnetic fluctuation reduction during PPCD. The fluctuations are reduced at least a factor of 3 during PPCD.

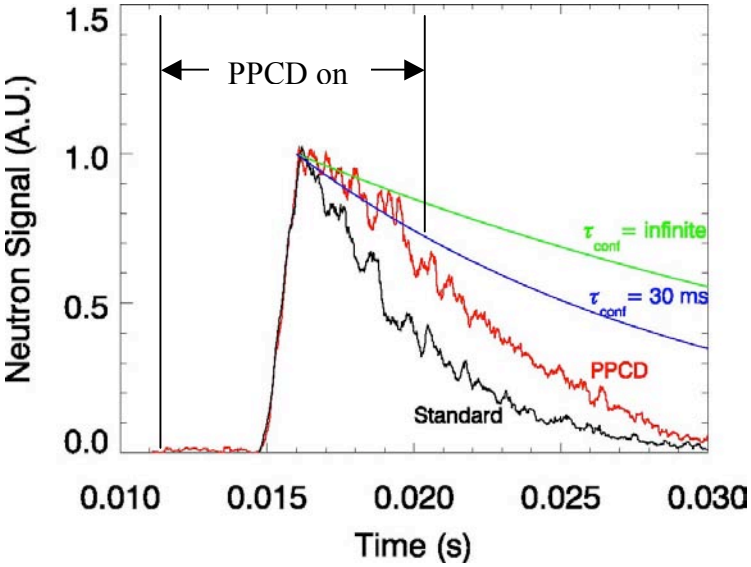


Figure 4.19 NBI into PPCD. The neutron signal during injection into PPCD (red) is shown with two decay curves for different values of the fast ion confinement time. A 30 ms confinement time (blue) makes a good lower bound. The neutron signal for the standard case (black) is shown for comparison. Infinite confinement is shown in green.

4.10 Fast Ion Confinement During Counter-Injection

Next we turn to confinement during NBI counter-injection. To achieve counter-injection, we reverse the polarity of the primary transformer windings. This reverses the direction of the electric field and hence the plasma current. We begin by considering the equilibrium changes that result from the reversal.

One motivation of doing co-injection is that it is favorable in terms of first-orbit losses. Ions that are born near the edge are bent inward in the poloidal magnetic field. With counter-injection, the opposite is true; ions are bent towards the wall. The effect (see Section 5.1) as shown in Fig. 5.3 is that the estimated ionization fraction goes from 85% down to about 45%.

We can once again look at the neutron signal and compare directly with the co-injection case. Fig. 4.20 shows two curves overlaid, the neutron flux during co-injection (red) (from Fig. 4.15) and the neutron flux for counter-injection (blue). The amplitude has the expected drop based on first orbit losses. When we normalize the amplitudes, as shown in Fig. 6.21, we can directly compare of the confinement times. We see that a 4 ms confinement time best fits the counter-injection neutron signal. This is interesting because it is a timescale significantly less than the co-injection case of 20 ms and also notably larger than the diffusive timescale of 1 ms.

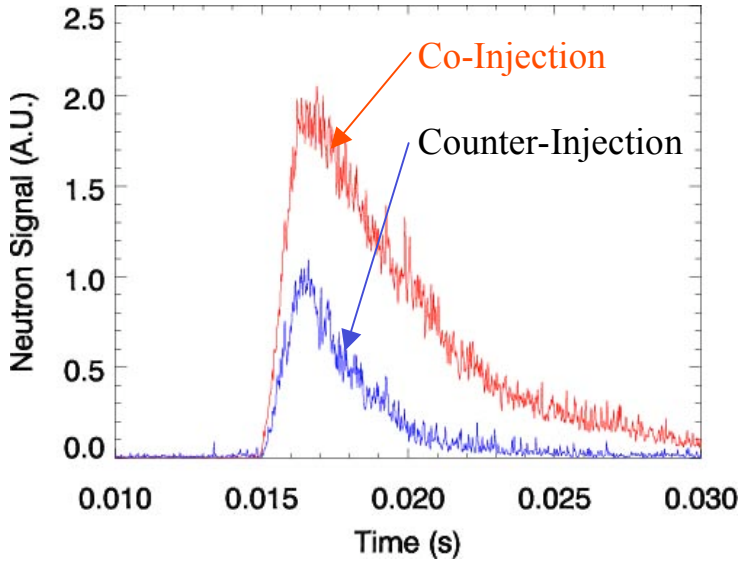


Figure 4.20 NBI during counter-injection. The resultant maximal neutron flux is decreased by about the expected amount based on first orbit losses.

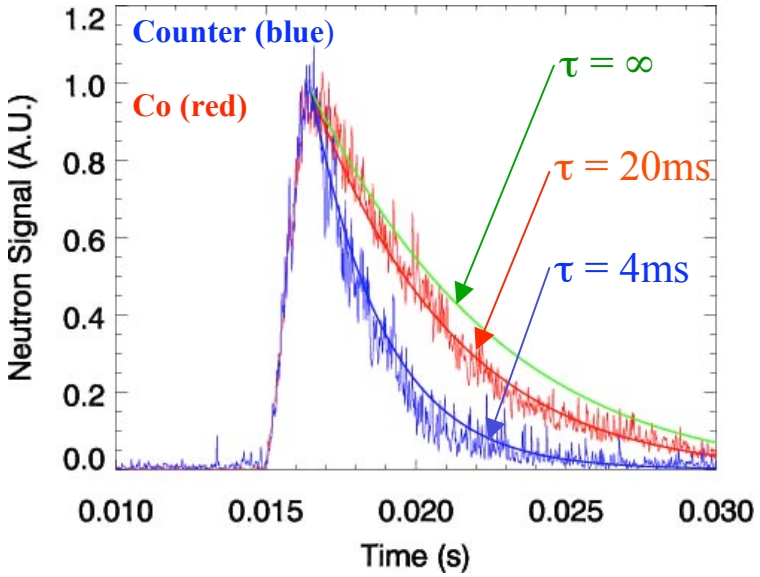


Figure 4.21 The same signals from Fig. 4.20 are normalized for comparison. Loss times show that the counter-injected ions are less confined than the co-injection case. The 4 ms confinement time is approaching (though not at) the estimate for stochastic diffusion of fast ions.

One possibility to explain this timescale, which is less than the 20 ms co-injection confinement time, is that the fast ion is diffusing from near-axis through most of the plasma radius, though not diffusive to the wall (See Section 5.9 for the simulation of counter-injection). The ion encounters a larger neutral density when it is further out in radius, which increases the rate of CX, and directly translates to a loss of particles. To calcuate the CX time, we must first know the neutral density throughout the plasma cross-section. A previous measurment[6] of the neutral density profile is shown in Fig. 4.22.

We can also consider the charge exchange losses during NBI into a standard plasma. The 20 ms confinement time for co-injection implies a minimum CX time

$$\tau_{CX} = \left(n_0 \left\langle \sigma_{CX}(\mathrm{v}_{fi})\mathrm{v}_{fi} \right\rangle\right)^{-1} \tag{4.16}$$

of 20 ms (if all losses were through that mechanism alone). We can choose on-axis values from Fig. 4.22 within the experimental uncertainty. Next, we utilize Eqn. 4.16 to calculate the charge-exchange time. This puts an upper bound on the on-axis neutral density of $5\text{x}10^{8}\ \text{cm}^{-3}$, which is an important restriction for this hard to measure quantity.

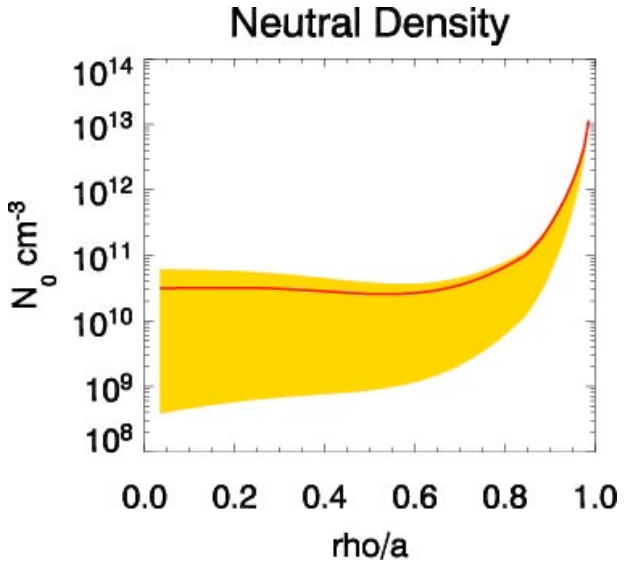


Figure 4.22 Neutral density in standard 400kA discharges. Measured using FIR, Thomson scattering and MSTFIT reconstruction. The nature of the Abel inversion of an edge-peaked profile leads to large uncertainty in core values.

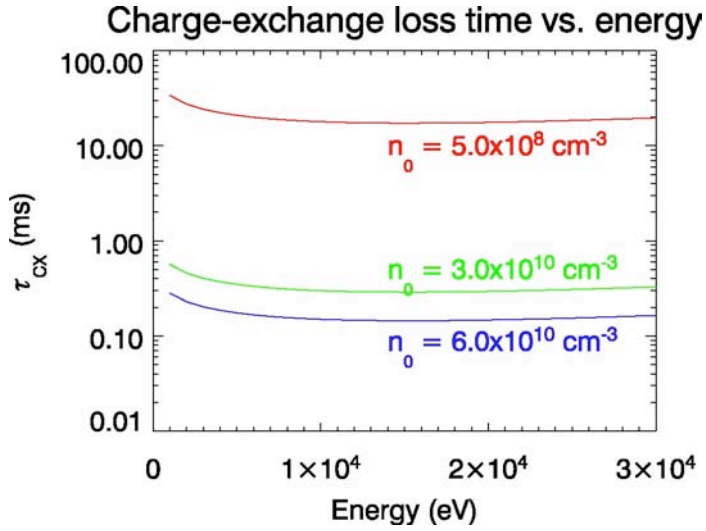


Figure 4.23 The charge-exchange loss times are plotted vs. energy for three different densities that correspond to the lower bound (red), best agreement (green) and upper bound (blue) of the axis value of the experimental curve shown in Fig. 4.22. At the fast ion energy of 20 keV, the charge exchange time is nearly the same as the fast ion confinement time. This puts an upper limit on the core neutral density of 5×10^8 cm^{-3}.

4.11 Fast Ion Confinement During Radial Injection

Considering radial injection as opposed to tangential, the most obvious difference is the initial angle of the fast ions relative to the magnetic field. With tangential injection, for most of the fast ions, there is a large component of the velocity parallel to the field, whereas with radial injection the initial velocity is perpendicular to the magnetic field. To achieve this, we make use of a diagnostic neutral beam that is part of the CHERS setup. The beam's parameters are: E = 20 keV, beam time = 3 ms, beam current = 1 A.

Again we see a neutron signal very close to that observed with the tangential beam, yielding a confinement time of 18 ms. This suggests that despite the fact that the ions are not born dominantly on-axis, a significant population of them must be well confined. There is a small sawtooth that occurs at about 19 ms. This reduces the fast ion population and so makes it difficult to compare with the confinement curve, but prior to the crash, the curve represented the data well. The simulation for radial injection is discussed in Section 5.11.

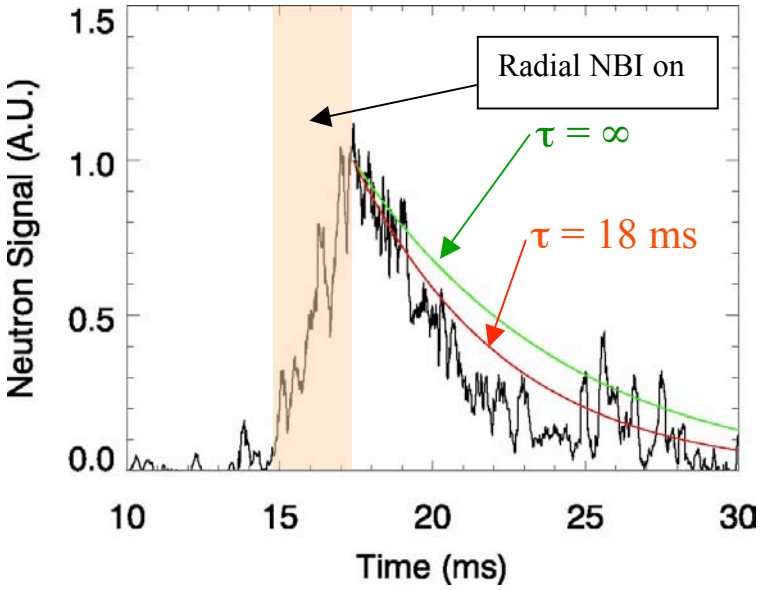


Figure 4.24 Neutron flux during radial injection. For a short pulse (~ 2.5 ms), 20 keV beam we again see the nearly the same confinement as with tangential injection (~18 ms). The drop in confinement at 19 ms and 21 ms are likely due to small internal sawteeth. The burst after 25 ms is from a sawtooth crash, and is likely a combination of plasma-plasma (from higher T_i) and beam-plasma (increase in plasma ion density) interaction.

4.12 Summary

The hardware components of the neutral beam have been presented and their functions and sequence of operation explained. This laid the groundwork for understanding the injector and the physics of how a neutral beam injector works. Various diagnostics are used to monitor beam performance during use. Additional diagnostics and experimentals were performed to verify the theoretical assumptions about the beam current, beam profile, and neutralization fraction. The actual experiment consisted of injecting 20 keV deuterium neutrals into a standard MST discharge and detecting the resultant neutron flux out of the vacuum vessel. By analyzing the rate of change of the

neutron signal, a fast ion confinement time of at least 20 ms was inferred. This demonstrates that the fast ion population is weakly affected by the underlying stochastic magnetic field. This represents the first measurement of fast ion confinement in the RFP.

Repeating the experiment under reduced magnetic fluctuations showed that confinement is marginally improved (30 ms confinement time). During a sawtooth crash, when magnetic fluctuations are large, we estimated a ~0.4 ms confinement time. This agrees with estimates of stochastic losses of fast ions. Counter-injection showed increased first-orbit losses as well as reduced confinement (4 ms confinement time). This could be explained through increased charge exchange losses, though by no means is a definitive argument. Radial injection showed good confinement (18 ms confinement time), similar to co-injection. Finally, the fast ion confinement time sets an upper bound on the core neutral density, which due to the method of measurement is inherently very inaccurate in that region.

References

1) ORNL 3113 p. 53 (Stier et. al., Phys. Rev. **96** 973 (1954))

2) S. Gladstone and R. H. Lovberg, *Controlled Thermonuclear Reactions* (Van Nostrand, New York, 1960), Chapt. 2

3) K. Miyamoto, *Plasma Physics for Nuclear Fusion* (MIT Press, Cambridge, MA 1989)

4) G. Fiksel, B. Hudson, et. al. Phys. Rev. Lett. 95, 125001 (2005)

5) J. S. Sarff et. al. Phys. Rev. Lett. **78**, 62 (1997)

6) N. E. Lanier, PhD Thesis (1999)

5 Simulation Of Fast Ion Motion

In Chapter 2 we established an accurate description of a typical magnetic fields configuration in the RFP, and MST in particular. The diffusion of the stochastic magnetic field lines and small gyro-radius particles, specificly electrons, was discussed. Now we are prepared to proceed with analyzing the orbits of large gyro-radius particles in this magnetic field in order to ascertain their expected degree of confinement or other important behavior. This thesis is concerned with fast ions, such as those born of neutral beam injection, though in principle the analysis could apply equally to ions in the tail of a plasma ions distribution function.

We begin the chapter in Section 5.1 by discussing prompt losses, which are ions that strike the vessel wall within a gyro-orbit.

In Section 5.2 we begin with a discussion of guiding center drifts, which is important for large gyro-radius particles.

Section 5.3 develops an analytical expression for the fast ions guiding center safety factor, which is very different than the thermal ions due to the presence of drifts.

Fluctuations in the ion's guiding center orbit lead to guiding center islands, in analogy to magnetic islands, and are the topic of Section 5.4. The overlapping of these islands determines the confinement of the fast ion.

Section 5.5 offers an explanation of the good confinement observed experimentally in Chapter 4, and is based on the analytical formulation given in the previous section.

Section 5.6 discusses the effect of toroidicity and its role in fast ion motion.

Section 5.7 concerns the actual simulation of fast particles in a stochastic magnetic field as done by an exact motion code. The code and plasma parameters used are discussed to set the background for the simulations in the following section.

In Section 5.8 the simulation results are presented and discussed in the context of representative descriptions of the confinement of neutral beam ions. A significant range of ions is shown to have regular orbits for typical injection energies. The transition from regular orbits to stochastic orbits is presented and compared with the analytical treatment formulated in section 5.4.

Section 5.9 examines simulation done in a counter-injection scenario. The results are compared with the reduced confinement seen in experiment.

The effects of varying levels of magnetic fluctuations are described in Section 5.10, as these provide an approximation of fast ion confinement in PPCD and during a sawtooth crash.

Radial injection is discussed in Section 5.11. This is not done with the heating beam on MST, but is of some importance because it is done on other machines, notably TPE-RX. There is some relevance for MST because we do have diagnostic neutral beams, which are oriented radially. They serve another purpose besides the inclusion of fast ions into the system, but nevertheless the physics is worth a cursory examination.

5.1 Prompt Losses

The neutral beam atoms entering the plasma are ionized by the plasma ions, primarily through charge-exchange, and also through impact ionization with the plasma

ions and electrons. The absorption profile of fast neutrals becoming fast ions as it passes through the plasma is given by

$$\frac{N_{fi}(l)}{N_{fi}(0)} = \left(1 - \exp\left(-\int_0^l \sigma_{CX}\, n_i(l') + \sigma_{Ii}\, n_i(l') + \sigma_{Ie}\, n_e(l')\, dl'\right)\right) \tag{5.1}$$

where σ_{CX}, σ_{Ii} and σ_{Ie} are the cross-sections for ionization due to charge-exchange, electron impact and ion impact, and the integral is done along an injection chord.

The injection chord is approximated to be on the midplane and tangent to the magnetic axis. A plot of the normalized beam neutral population along the injection chord is shown in Fig. 5.1. The total ionization of the fast ions is around 95%. Because of the injection geometry and the relatively low edge plasma ion density, most of the beam ionization occurs near the magnetic axis. Doing particle tracing in the absense of fluctuations, and weighting the density of the fast ions according to the ionization rate results in the distribution of fast ions in a poloidal cross-section and is shown in Fig. 5.2. We will use this peaked distribution to justify near-axis ions as being the most typical fast ion orbit to analyze, though all radii are of interest.

We first consider prompt losses of ions born near the vessel wall, where the magnetic field is mostly in the poloidal direction. Depending on the ions path during the first gyro-orbit, it may or may not strike the vessel wall and be lost. For co-injection the magnetic field tends to bend the ions inward and confine them; counter-injection does the opposite.

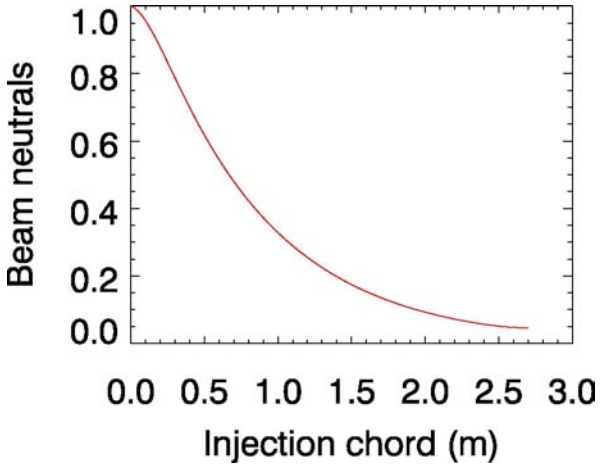


Figure 5.1 Fraction of beam neutrals along the injection chord. The beam neutrals are ionized (primarily through charge-exchange) by the plasma ions. At a plasma ion density of ~$8x10^{18}$ m^{-3} this gives 95% ionization of the injected neutral beam atoms.

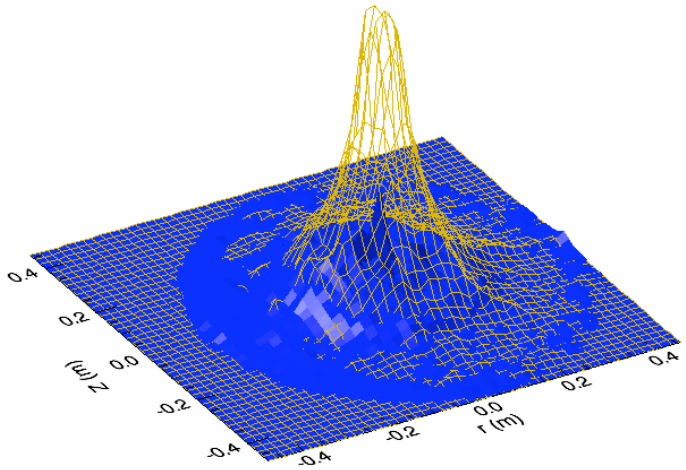


Figure 5.2 Distribution of co-injected fast ions in a toroidal system. The population is strongly peaked about the magnetic axis. From this we will often take a "typical" orbit to begin at r/a = 0.2.

These prompt losses reduce the total number of fast ions that are within the plasma. Again, particle-tracing code easily determines if a fast ion will strike the vessel wall promptly. The results of this calculation are shown in Fig. 5.3 (left) for co-injection and Fig. 5.3 (right) for counter-injection. The attenuation of the beam neutrals is the same in both cases. The fast ions are bent inward away from the wall by the poloidal magnetic field for the co-injection case. Counter-injection causes the ions to be bent toward the wall by the poloidal magnetic field. As a result, the total number of fast ions in the system is about half for counter-injection. This provides us with a preferential injection scenario with respect to the background magnetic field.

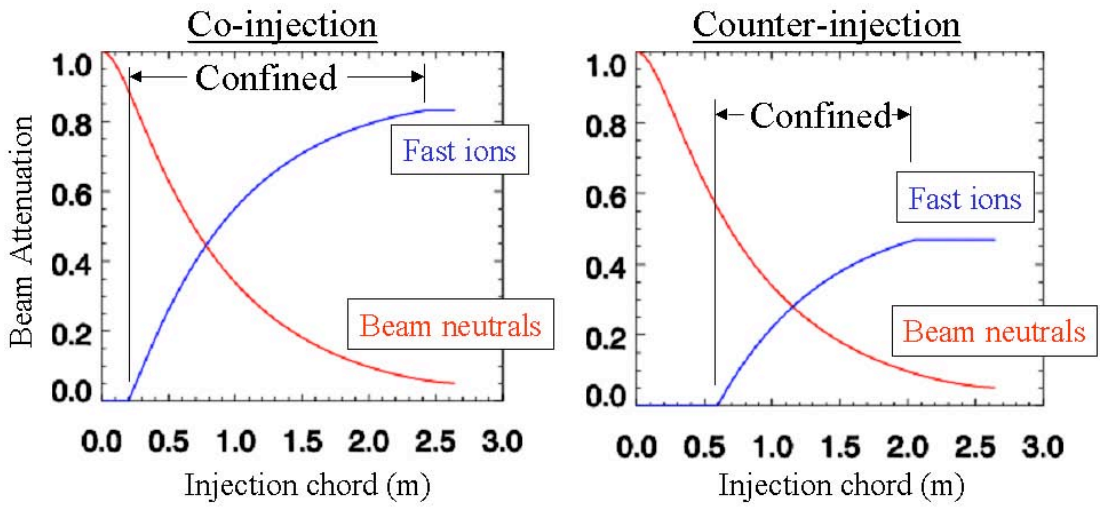


Figure 5.3 Prompt losses for co-injection and counter-injection. The ionization of the beam neutrals (red) results in fast ions (blue). If an ion is born too close to the wall, its gyro-orbit may strike the wall and be lost, thus not contributing to the overall number of beam ions; the flat regions on the blue curves reflect this. During counter-injection, NBI ions are bent outward in the poloidal magnetic field in the edge of MST and are more likely to be lost. The result is that the total number of fast ions that can be confined is reduced by about a factor of two.

5.2 Guiding Center Drifts

While our simulations follow the exact motion of the particle, it is more illuminating to consider instead the motion of the ions guiding center. We will begin with a discussion of the ions guiding center (IGC) motion in a general magnetic field.

The equation describing fast ion guiding center motion is given[1] by

$$\vec{v}_{GC} = v_{\parallel}\vec{b} + \frac{v_{\perp}^2}{2\omega_c}\frac{\vec{B}\times\vec{\nabla}B}{|B|^2} + \frac{v_{\parallel}^2}{\omega_c}\frac{\vec{B}\times\vec{\kappa}}{|B|} + \frac{\vec{E}\times\vec{B}}{|B^2|} \qquad (5.2)$$

with $\vec{E}, \frac{d\vec{E}}{dt} = 0$. The first term in Eqn 5.2 is the motion of the guiding center parallel to the magnetic field, the second term is the gradient drift and the third term is the curvature drift. We can calculate the velocity contribution from drifts and compare it to the total particle velocity. In addition, we can also estimate the magnitude of the $\vec{E} \times \vec{B}$ drift for a typical ion to see if it is acceptable to neglect the role of the electric field and still have a suitable description of the guiding center motion. First we have the particle velocity, which for a 20 keV deuterium ion is

$$\mathrm{v} = \sqrt{\frac{2E(eV)e}{m}} = 1.4x10^{6}\, m/s\,. \tag{5.3}$$

The gradient and curvature drifts are of similar magnitude, so a typical value might be

$$\mathrm{v}_{curv} = \frac{\mathrm{v}_{\|}^{2}}{\omega_c}\frac{B \times \nabla B}{B^2} \cong \frac{\left(1.4x10^{6}\, m/s\right)^2}{7x10^{6}\, s^{-1}} 0.2\ m \cong 5x10^{4}\, m/s\,. \tag{5.4}$$

The E X B drift velocity at r/a < 0.5 is no greater than

$$\mathrm{v}_{\phi}^{ExB} = \frac{E_r(V/m)\,B_\theta(T)}{\left|B(T)\right|^2} = \frac{2000 \times 0.2}{(0.3)^2} = 4x10^{3}\, m/s\,. \tag{5.5}$$

Therefore, we can neglect the effects of electric fields because the magnitude of the $\vec{E} \times \vec{B}$ drift is at least a factor of 10 lower than the drifts from the magnetic field.

5.3 Fast Ion Guiding Center Safety Factor

We assume that the guiding center motion is confined to an unperturbed flux surface, which is true to $\vartheta(1)$ in $\rho_\perp$, where $\left(\rho_\perp \equiv \frac{m\mathrm{v}_\perp}{eB}\right)$. Analogous to the expression for the safety factor of the magnetic field lines, we have

$$q_{IGC}(r) = \frac{r v_{\phi}^{GC}}{R v_{\theta}^{GC}}, \tag{5.6}$$

where again r is the minor radius variable and R is the dimension of the major radius; v_{ϕ}^{GC} and v_{θ}^{GC} are the toroidal and poloidal guiding center velocities.

Combining Eqns. 5.2 and 5.6 (see Appendix B) we arrive at the following expression for the guiding center safety factor

$$q \cong q_M + \frac{s_{\|}}{b_{\theta}^2} \rho \frac{2(1-\mu\Omega)b_{\theta}^2 - r\mu\Omega'}{2R\sqrt{1-\mu\Omega}}, \tag{5.7}$$

where q_M is the magnetic field line safety factor. In Eqn. 5.7 $s_{\|} = \frac{\vec{v} \bullet \vec{B}}{|\vec{v} \bullet \vec{B}|}$ is the direction of the guiding centers velocity parallel or perpendicular to the magnetic field, and for our case has a value of 1.0 for co-injection and –1.0 for counter-injection. The poloidal component of the magnetic field, normalized to the total field is given by b_{θ}. The gyro-radius for the total energy is ρ. The ratio of the perpendicular energy to the total energy divided by the cyclotron frequency is, $\mu = \frac{E_{\perp}}{E_0 \Omega} = \frac{v_{\perp}^2}{v_0^2 \Omega}$, where $\Omega = \frac{eB}{m}$ and $\Omega' = \frac{e}{m}\frac{dB}{dr}$ are the gyro-frequency and the radial derivative of the gyro-frequency, respectively.

The key thing to note about Eqn. 5.7 is that the guiding center does not have the same safety factor as the magnetic field lines. The presence of gradient and curvature drifts, which are substantial for fast ions due to their gyro-radius, change the safety factor.

Another observation is that the denominator in Eqn. 5.7 is resonant: the value of q would go to +/- infinity if the parallel velocity went to zero. This is simply the magnetic mirroring that arises self-consistently from the transformation of the guiding center velocity equations.

We now consider a typical ion that might arise from neutral beam injection. Fig. 5.4 compares the safety factor for an ion born at r/a = 0.2, with 90% of its velocity directed parallel to the magnetic field. The ions safety factor is significantly different from that of the field lines, and for the ion that is born at r/a = 0.2, it is about 1/5, as compared to the magnetic field, which is about 1/6. While a seemingly unimportant fact, it is critical when one then considers the effect of resonances.

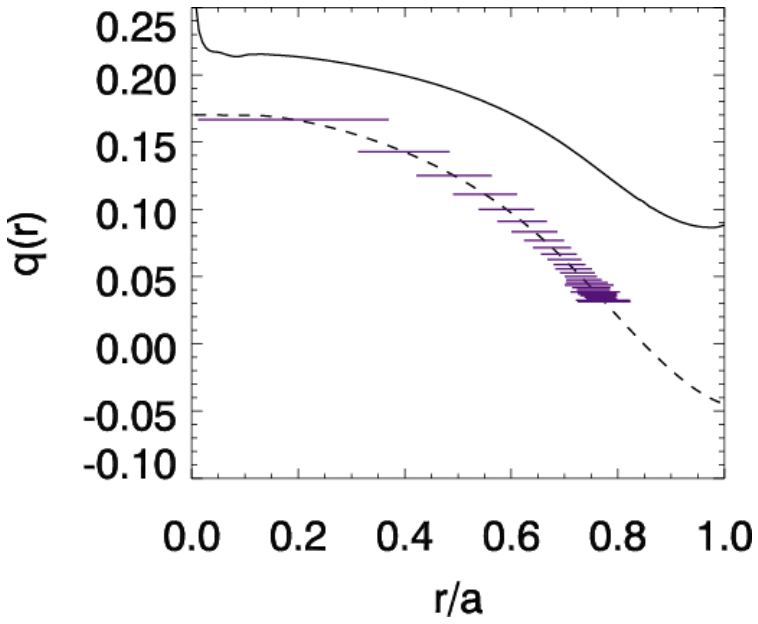


Figure 5.4 Comparison of ion guiding center safety factor (solid) with magnetic safety factor (dashed). The effect of drifts on the ions guiding center shifts the safety factor away from the field lines (or low energy limit) case.

5.4 Ion Guiding Center Islands

Unless otherwise stated, we are considering orbits in a cylindrical geometry. As a result, in the cylindrical approximation, the guiding center drifts have only poloidal and toroidal components to them, not radial. This is simply because both the magnetic gradient and curvature are only in the radial direction. By ordering of the fast ion's guiding center motion in Larmor radius,

$$\begin{aligned} &\vartheta(0) \quad V = V_{\|}; \quad V_r = 0 \\ &\vartheta(1) \quad V = V_{\|} + drifts; \quad V_r = 0 + \propto \tilde{b}_r(r,m,n) \end{aligned}, \tag{5.8}$$

we see that the ion's guiding center does not follow the toroidal or poloidal components of the magnetic field, the deviation from the radial component is small. The guiding center drifts due to the radial magnetic field are second order and are neglected.

The guiding center trajectory has a different helical pitch (as seen in the change of q) from the magnetic field lines, but we assume it responds to the same radial perturbations, since in the absence of drifts the particle would follow the field lines, and in the radial direction there are no guiding center drifts. Therefore we expect that the orbit also has an analogous resonance condition. Referring back to Fig. 5.4, the upward shift in the ions q-profile brings the 1/5 surface into resonance, whereas with the magnetic field, at about that location the 1/6 surface is in resonance. While the field then responds to the 1/6 magnetic mode to form an island, the guiding center responds instead to the 1/5 radial velocity perturbation induced by the 1/5 magnetic mode. We stated earlier that the parallel motion of the guiding center responds to the magnetic fluctuations in the radial direction. Therefore, the relative radial perturbation in velocity is defined by the strength of the local magnetic modes. The effect of a resonance though is key, and it is the 1/5 radial magnetic mode that is most responsible for the radial perturbation in the ion's guiding center velocity.

In direct analogy with magnetic fields lines then, we can write the expression for the width of the resultant "ion guiding center islands" (see Appendix B for details). We recall the magnetic island formula

$$W = 4\sqrt{\frac{b_r r_s}{B_\theta n|q'|}}, \tag{5.9}$$

and also note that

$$\frac{\mathrm{v}_r}{\mathrm{V}_\theta} \cong \frac{b_r}{B_\theta}\left(\frac{B_\theta}{B}\frac{\mathrm{V}}{\mathrm{V}_\theta}\right). \tag{5.10}$$

Since the correction to poloidal velocity of the ions guiding center is a few percent (Eqn. 5.4), the part of Eqn. 5.10 in parentheses is of order unity. Substituting this into the expression for the width of the ion guiding center islands we have

$$W_{IGC} = 4\sqrt{\frac{\mathrm{v}_r r_s}{\mathrm{V}_\theta n\left|q'_{IGC}\right|}} \cong 4\sqrt{\frac{b_r r_s}{B_\theta n\left|q'_{IGC}\right|}} \tag{5.11}$$

where all the relevant quantities are taken at the radial location where the *guiding center* motion is resonant. This allows us to vastly simply calculations for estimates of the guiding center island size by not needing to calculate any perturbed velocity components, which are in fact an infinite sum over modes, even though "only" the leading dozen or so would be required to get an approximate expression.

The positions and widths of the ion guiding center islands are shown in Fig. 5.5. Here we see that the $n = 5$ island is present, but not overlapped with the $n = 6$ IGC island. There are regions of stochasticity in the phase space of the orbit, as seen in the overlapped areas at r/a > 0.4. It should be stressed though that the IGC q-profile is for a specific value of initial radius and velocity, and that an ion starting further out in radius will have a very different q-profile and island overlap configuration.

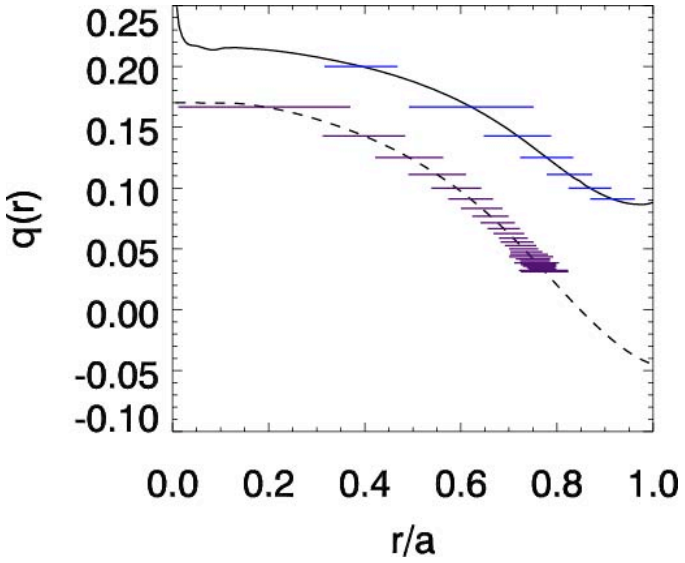


Figure 5.5 Ion guiding center islands. The solid *q*-profile and blue islands are for a 20 keV deuterium ion starting at r/a = 0.2. The field line *q* and associated islands below are shown for comparison. There are no overlapped resonances at the ions start point so good confinement can be expected.

5.5 Explanation of Improved Fast Ion Confinement

We saw that the fast ion's confinement is weakly affected very much by the stochastic magnetic field. We calculated a confinement time far in excess of what it would have been were the orbits stochastic. From the previous sections in this chapter, without doing any particle tracing simulations at all, we can apply the Chirikov overlap criterion to the guiding center islands. For our test 20 keV ion, we see in Fig. 5.5 that the 1/5 guiding center island, calculated from the above expressions, is not overlapped with the 1/6 guiding center island. In this particular case then, we expect the fast ion to not be stochastic, even though the magnetic field, *in the exact same location*, is. Given that the vast majority of our fast ions conform to this description, that is they are near the axis, we

expect that they would not be stochastic, and hence well confined. This formulation can be tested numerically by doing exact motion simulation and is the subject of Sections 5.7 - 5.11, but already agrees in a qualitative sense with the observed experimental results.

5.6 Effect of Toroidicity

We described at length the effects of drifts, gradient and curvature on the fast ions guiding center orbit. In these formulations, we justified neglecting the effect of toroidicity, and using a cylindrical representation, but is that valid?

Toroidal curvature creates variation of the curvature vector of the magnetic field lines in radius and poloidal angle, which introduces a poloidal component to the curvature vector. This means that there is a small radial guiding center drift. However, in practice, this is not important as it affects mainly ions that begin at a large value of the minor radius. In any case, we can note that $\left|\vec{R}_C\right|_{Toroidal} \geq 3\left|\vec{R}_C\right|_{Poloidal}$, which indicates that the poloidal curvature dominates the contribution to the drift velocities. Some edge ions may drift into the wall before completing a few toroidal transits, but the non-stochastic orbits that don't drift immediately into the wall remain well confined.

The decrease in the toroidal field strength away from the toroidal axis also introduces another gradient drift. For passing particles, which would be typical of near-axis fast ions from NBI, the drift shifts the particles orbit radially from the magnetic axis. Using MST and typical beam ion parameters, we find a shift of $2q(r)\rho_{\|} = 0.06a$. For trapped particles, typical of edge-born fast ions, they have banana orbits with a width of $\frac{2\sqrt{2}\,q(r)\rho}{\sqrt{\varepsilon}} = 0.1a$, where $\varepsilon \equiv r/R_0 \cong a/R$ for this case. It is important to note that the

orbit widths are functions of q, and in the RFP q is a factor of 5 lower than in tokamaks in the core, and goes to zero at the reversal surface. In tokamaks, the banana widths are typically larger than the ion gyro-radius and so govern neo-classical transport. In the RFP, the reverse is true; the gyro-radius of 20 keV D ions at the edge is about $0.1a$ and the banana width is $0.1a$ or smaller. We expect then that these toroidal effects have minimal impact on our orbit formulation.

5.7 Test Particle Orbit Simulation

We use a particle tracing code (RIO) to simulate the orbit of a fast ion in a stochastic magnetic field to ascertain and understand the degree and nature of the ion's confinement. The code simulates a time-independent background plasma and evolves the velocity of the fast ion as it loses energy to the plasma. The setup of the code and description of the background plasma are also briefly discussed.

After having a description of the stochastic magnetic field of the RFP and an analytical approximation of the guiding center motion of fast ions, we have the tools to understand the results of doing exact particle tracing to arrive at a more exact resolution of the confinement problem. In addition to MAL, a code RIO was developed at the Budker Institute that solves the ion's equation of motion in the same field geometry used as RIO. Specifically, it solves the coupled differential equations:

$$\begin{aligned} \frac{d}{dt}\vec{\mathrm{v}}_{fi}(\vec{x},t) &= \frac{Z_{fi}e}{m_{fi}}\left(\vec{\mathrm{v}}_{fi}(\vec{x},t)\times\vec{B}(\vec{x})\right) - \nu_s\,\vec{\mathrm{v}}_{fi}\left(\vec{x},t\right) \\ \frac{d\vec{x}}{dt} &= \vec{\mathrm{v}}_{fi}\left(\vec{x},t\right) \end{aligned}, \qquad (5.12)$$

where $\acute{\imath}_s$, the slowing down frequency, is given[2] by,

$$\nu_s^{\alpha/\beta} = \left(1 + \frac{m_\alpha}{m_\beta}\right)\psi\left(x^{\alpha/\beta}\right)\nu_0^{\alpha/\beta} \tag{5.13}$$

$$\nu_0^{\alpha/\beta} = 4\pi e_\alpha^2 e_\beta^2 \lambda_{\alpha\beta} n_\beta / m_\alpha^2 \mathrm{v}_\alpha^3 \tag{5.14}$$

$$x^{\alpha/\beta} = m_\beta^2 \mathrm{v}_\alpha^2 / 2kT_\beta \tag{5.15}$$

$$\psi(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t} \sqrt{t}\ dt \tag{5.16}$$

The code calculates both the ions motion and drag on the background plasma for electrons and an arbitrary number of ion species. For our simulations, we use deuterium plasma ions and fully stripped carbon, which is the dominant impurity in MST. We approximate all experimental profiles with the functional form:

$$A(r) = \left(A|_{r=0} - A|_{r=a}\right)\left(1 - \left(\frac{r}{a}\right)^\alpha\right) + A|_{r=a} \tag{5.17}$$

where the values at $r = 0$ and $r = a$ specify axis and wall quantities, respectively and alpha determines the flatness of the profile. The electron temperature has been measured with Thomson scattering, electron density profile obtained with FIR interferometry. The plots of the electron and deuterium ion densities and temperatures are shown in Fig. 5.6 and 5.7 respectively. To get the plasma ion density we use the previous estimate of Z_{eff} ($Z_{eff} \sim 2$) and the impurity (Carbon V) density.

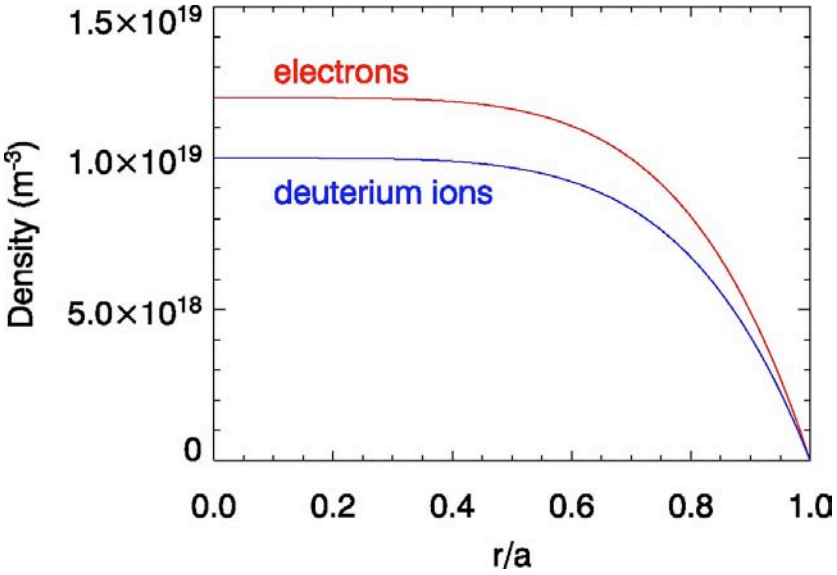


Figure 5.6 Density of plasma components used in simulation. Electron density (red) and deuterium ion density (blue).

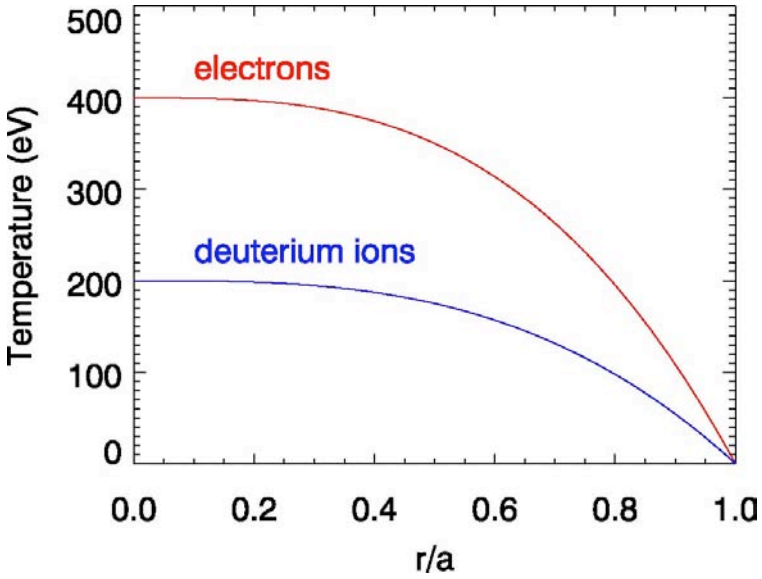


Figure 5.7 Temperature profiles of plasma components used in simulation. Electron temperature (red) and deuterium ion temperature (blue).

While we can't directly simulate tangential injection in a cylindrical geometry, we can approximate it by calculating the pitch angle of the fast ions relative to the injection chord. This is shown schematically in Fig. 5.8. The injector itself has an 8 degree downward tilt, but is neglected.

5.8 Confinement Of Fast Ions And Onset Of Stochasticity

This section shows the results of the particle tracing simulation. We begin by considering the most typical fast ion, that is, one near the axis. Next we look at the confinement of ions born further out in radius. Finally we conclude with a brief section on confinement of an ion born inside an intact ion guiding center island.

5.8.1 Near-Axis Co-Injected Ions

We begin this section by describing a near-axis, co-injected deuterium ion in the absence of magnetic fluctuations and no background plasma for the ion to slow down on. The purpose is to illustrate, in this simple case, the expected effect of the drifts discussed in section 5.2.

Fig. 5.9 shows a plot of the guiding center position of a 20 keV deuterium ion injected nearly parallel to the magnetic field and in the same direction as the plasma current. We can see clearly the periodicity of the orbit; ~5 poloidal transits for every toroidal transit, as expected from the q profile (Fig. 5.5).

Next we include magnetic perturbations. Fig. 5.10 shows a plot of the ion's guiding center position overlaying a puncture plot of the background magnetic field. While the field is clearly stochastic, the ion shows a confined 1/5 character to its orbit.

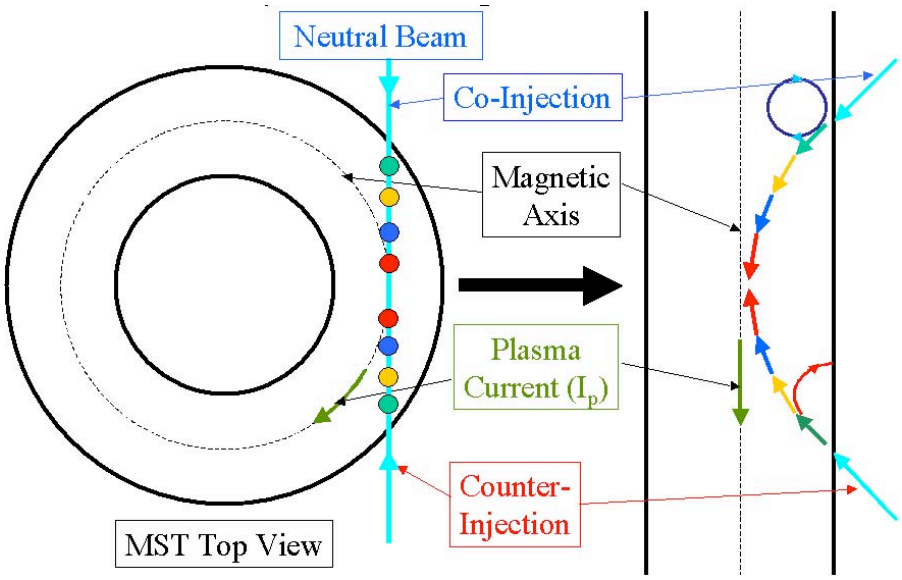


Figure 5.8 Approximating toroidal injection into a cylindrical system. The pitch angle of the ion with the equilibrium field is retained by varying the direction of injection relative to the axis of the cylinder.

The fast ion is responding to the presence of an $n = 5$ perturbation, which forms an island in its phase space. The ion's guiding center is outside the separatrix of the $n = 5$ ion island and has a regular orbit, albeit highly perturbed by the nearby resonance. By choosing a different toroidal angle for the ion start position with respect to the ion guiding center island we can place the guiding center inside the ion guiding center island.

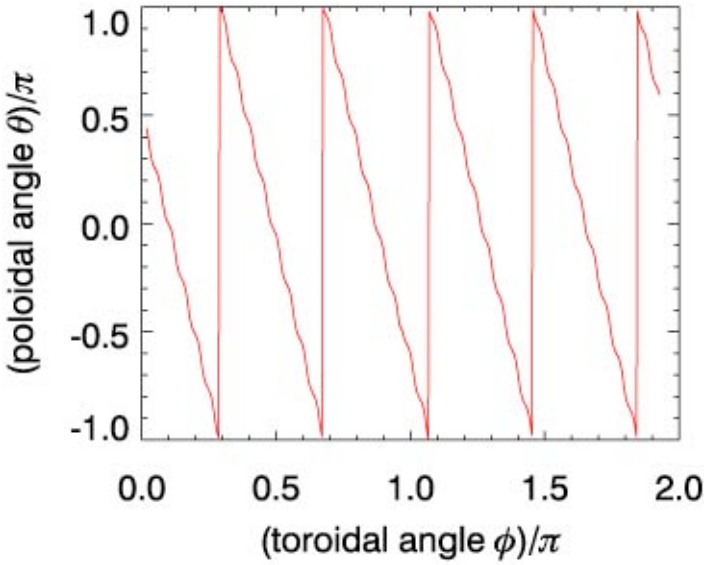


Figure 5.9 Ion guiding center motion in absence of perturbations. An ion born at r/a = 0.2 and plotted with poloidal angle vs. toroidal angle clearly shows the $n = 5$ character to be apparent despite the magnetic fields $n = 6$ character at the same location.

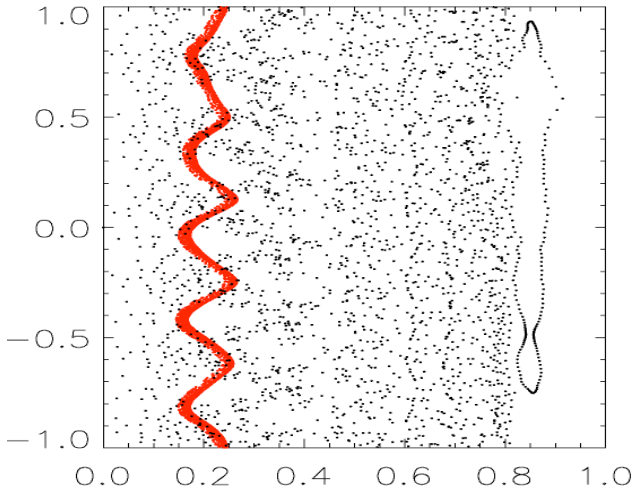


Figure 5.10 Ion guiding center motion with perturbations superimposed on magnetic Poincaré map. The guiding center feels a strong perturbation from the $n = 5$ ion island, but as expected, since the resonances are not overlapped in the location of the orbit, the trajectory is confined to a perturbed but unbroken surface.

Just like a magnetic field line inside the separatrix would trace out a closed orbit around the island's O-point, a fast ion's guiding center can also be constrained inside an ion island. Fig. 5.11 shows an example of such an "island trapped" trajectory.

We have now established a very important physics point. There are at least some fast ions that will not be stochastic even when in a stochastic magnetic field. However, if this is because of the guiding center drifts taking the ion trajectory away from the stochastic field lines, we might expect the ion orbit to only be regular for certain energies. Refering to Eqn. 5.7, the leading factor is the total energy gyro-radius, $\rho = \frac{m\mathrm{v}}{qB}$. If we allow the ion to slow down on the prescribed plasma as per Eqn. 4.15, then at some energy, the orbit should become stochastic.

Fig. 5.12 shows the same initial conditions as Fig. 5.10 but now with Coulomb collisions allowed to drag on the ion. What is interesting about this situation is now the ion's safety factor is explicitly a function of time, as the ion's energy and gyro-radius are functions of time. The guiding center safety factor is also shown for three different times in Figs. 5.12 - 5.14. Phase one is the ions starting condition. As the ion loses energy, its guiding center safety factor decreases, approaching that of the magnetic field lines, which is the zero energy limit of Eqn. 5.7. At a certain energy (about 18 keV in this case), the q-profile has lowered enough that the $n = 5$ resonant surface moves inward past the position of the ions guiding center. This is seen to result in a discrete jump in the guiding center position from one side of the $n = 5$ island to the other, whereupon the ion resumes its prior character of good confinement. We do note that the $n = 6$ ion island is resonant, but initially much further out in radius. However, as the ion loses energy, this

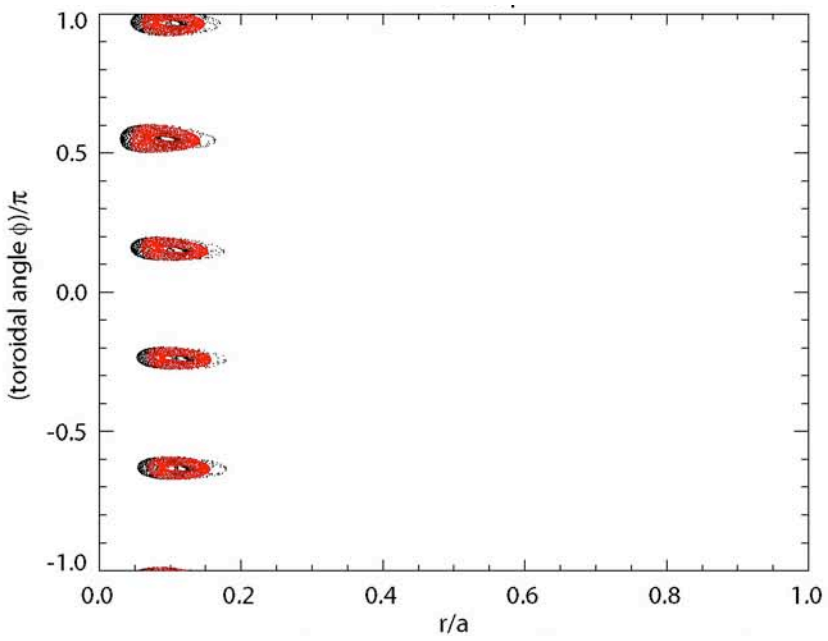


Figure 5.11 Island-trapped fast ion. The phase of an ions starting location determines whether it is inside an island or outside.

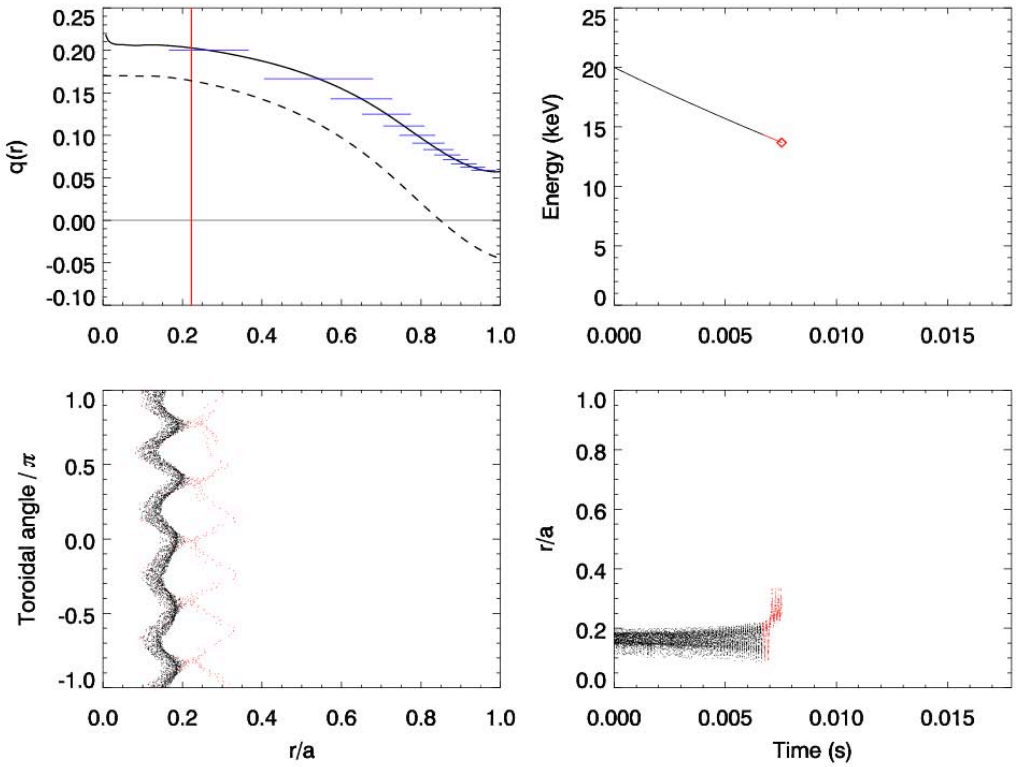


Figure 5.12 A single fast ion slowing from 20 keV to 13 keV. A 20 keV ion experiences good confinement in the vicinity of the $n = 5$ resonance, but makes a discrete jump when the ion island separatrix passes the guiding center position. The upper left plot shows the magnetic q-profile (dashed), the fast ion guiding center q-profile (solid) and island widths for the ion guiding center (blue). During the time the simulation spans in the plot the average guiding center position is shown with the vertical red line, and is in the neighborhood of the non-overlapped $n = 5$ island. The upper right plot is the fast ion energy. A puncture plot of the guiding center position is shown in the lower left, clearly revealing the $n = 5$ ion guiding center island. The lower right plot shows the radial position of the ion guiding center vs. time, with a jump made at ~13 keV (7 ms) due to the $n = 5$ resonance moving inward past the guiding center position.

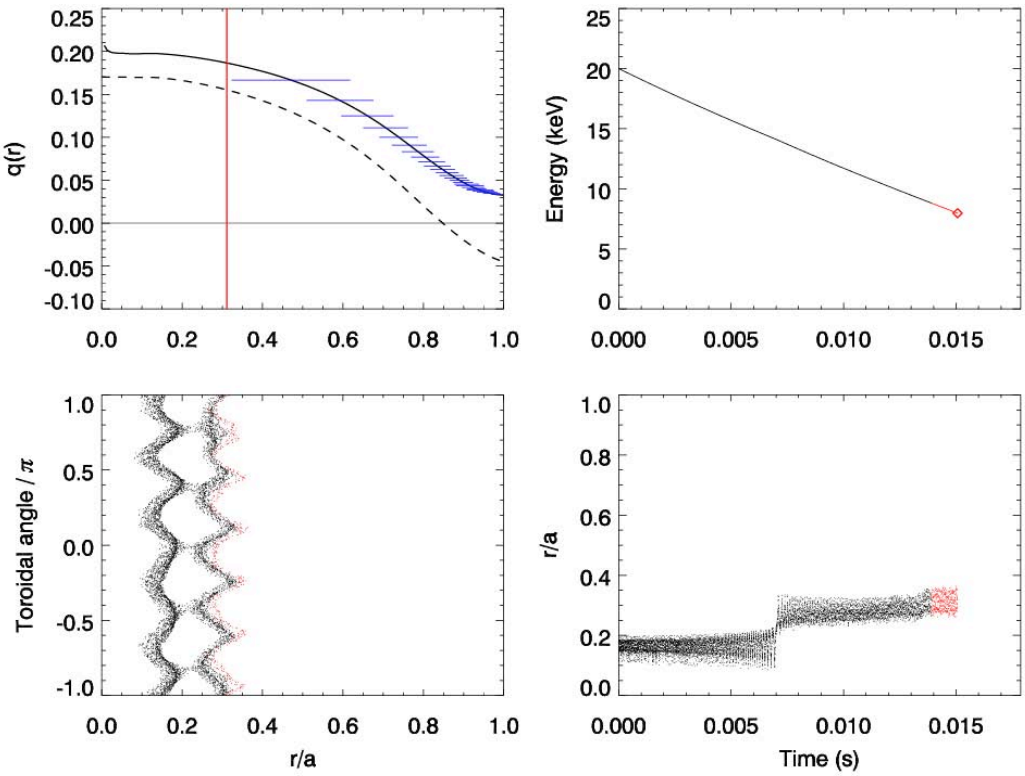


Figure 5.13 A single fast ion slowing from 20 keV to 7.5 keV. As seen in the upper left and lower right plots, the $n = 5$ goes out of resonance and confinement is still good since the overlapped resonances, $n = 6+$, have not yet reached the guiding center position. Also note the $n = 6$ character of the guiding center punctures (lower left) that agree with a strong $n = 6$ resonance perturbing the orbit.

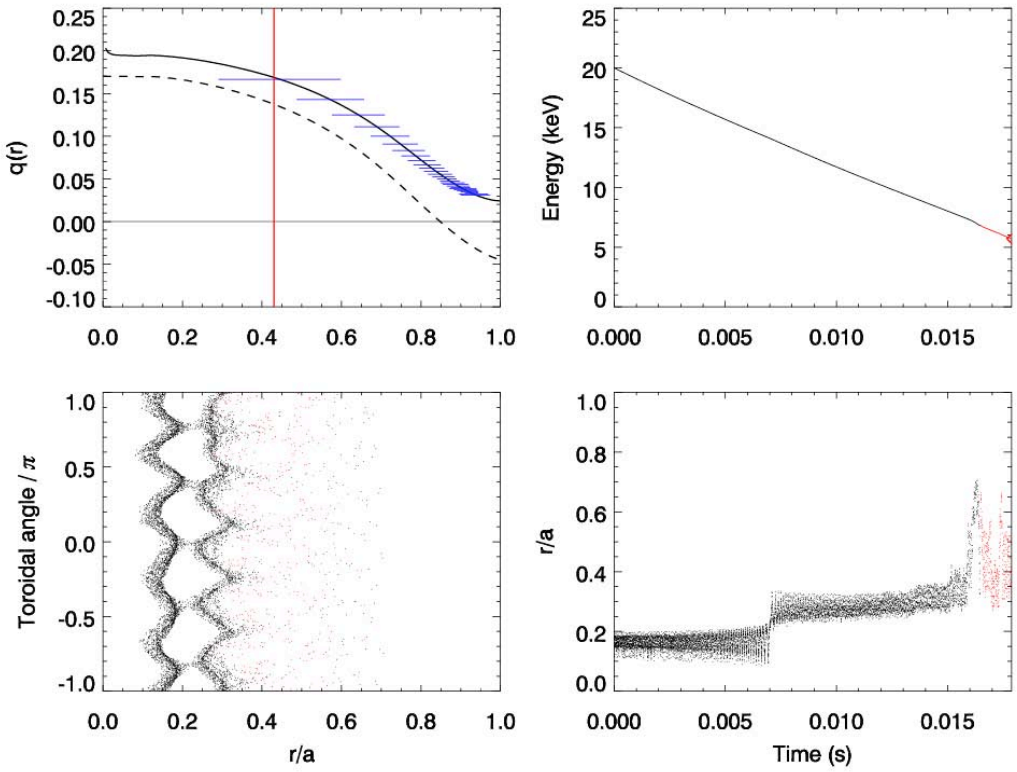


Figure 5.14 A single fast ion slowing from 20 keV to 6 keV. Once the overlapped ion islands reach the guiding center position, the orbit becomes stochastic. The short diffusion time (lower right plot) of < 1 ms to go from r/a = 0.3 to r/a = 0.7 was predicted (Section 1.2) under the assumption that stochastic fast ions diffusing would be lost quickly due to their high parallel velocity.

resonance moves inward as q decreases. We see in Fig. 5.13 that the $n = 6$ island is partially overlapped, but the ion is not at the overlapped location, and so is not stochastic. Advancing in time, Fig. 5.14 shows that the ion's guiding center position coincides with the edge of the overlapped $n = 6$ island and consequently, the orbit is stochastic, as easily seen in the large and random radial excursions within the area expected to be stochastic.

This sequence of plots shows that our assumptions about the ion's motion were correct qualitatively and to a large extent, quantitatively. More importantly, since the transition to stochasticity occurred after the ion lost 13 keV of its energy, good confinement can be expected over a robust range of energies and this is very promising when considering the effectiveness of NBI.

5.8.2 Confinement Dependence on Ionization Position

We perform simulations for ions that start at a variety of radii, varying the initial pitch angle in accordance with Fig. 5.8. This allows us to examine the confinement of ions that would be born at different positions along the NBI injection chord. The *q*-profile for an ion starting at r/a = 0.4 is shown in Fig. 5.15 and the radial position shown in Fig. 5.16. The orbits remains significantly confined, and when the energy reaches 11 keV (as opposed to 7 keV in the r/a = 0.2 case) we see that the ion becomes stochastic.

Considering one other initial position, r/a = 0.8, we see that the guiding center *q*-profile in Fig. 5.17 is very different than the magnetic field *q*-profile. This is due to magnetic mirroring. Recall that the magnetic field is strongest at the axis, so if an ion is born with a small

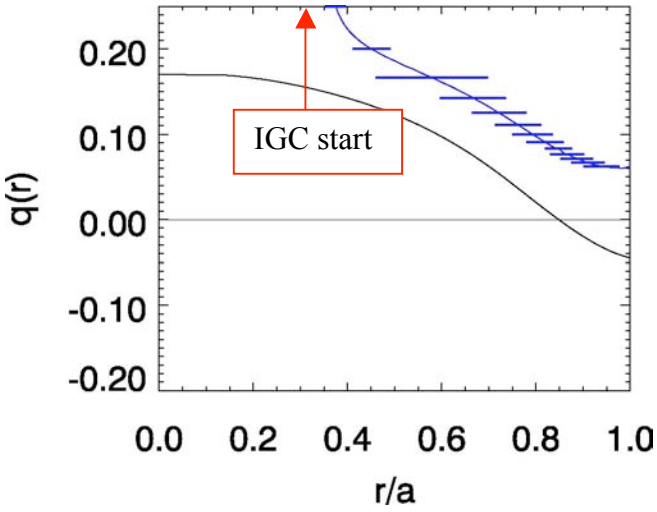


Figure 5.15 Ion starting at r/a = 0.4. Because the orbits are bent inward in co-injection, the IGC starts at r/a ~ 0.35. Also note the discontinuity because of the mirroring condition.

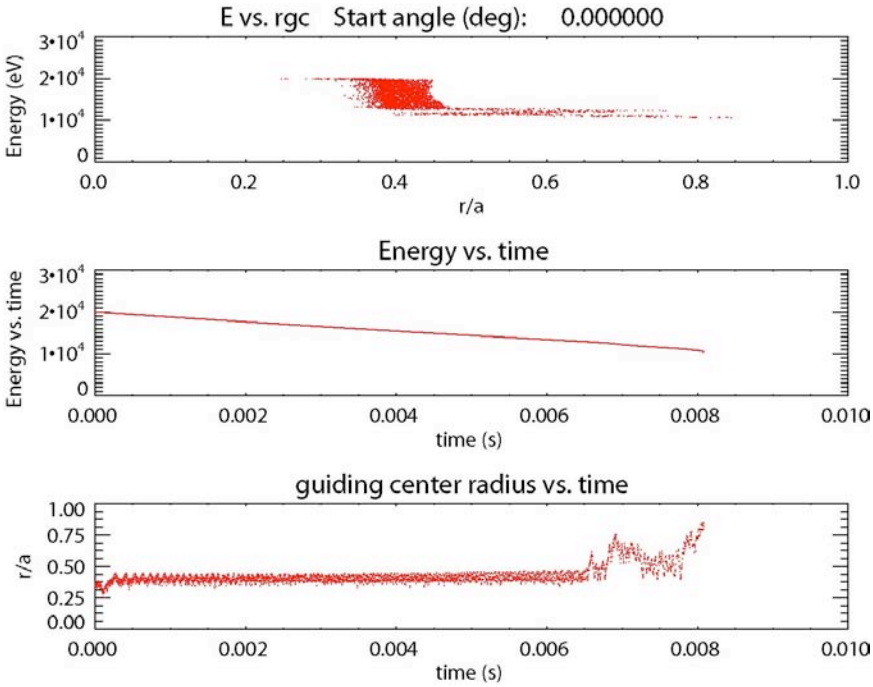


Figure 5.16 The ion is briefly confined, then becomes stochastic much earlier than the r/a = 0.2 case. Less energy is needed to be lost for the overlapped guiding center islands to reach the ion guiding center position.

enough parallel energy, in theory, it would mirror if it had the ability to go from the weak field position to a stronger field position. Orbit stochasticity provides a mechanism for transport, but the mirroring condition remains. Mathematically, this can be seen in the denominator of Eqn. 5.17. Since $\mu\Omega$ represents the perpendicular energy fraction, if it is 1, then the denominator goes to zero. This leads to an infinite IGC safety factor.

Fig. 5.18 shows the radius vs. time plot for the r/a = 0.8 starting position. The orbit is actually confined quite well, and this is due to the fact that the shear in the *q*-profile due to mirror has rendered the island size very small. It is worth noting that very few fast ions are born in these outer radii, and there are also prompt losses to contend with.

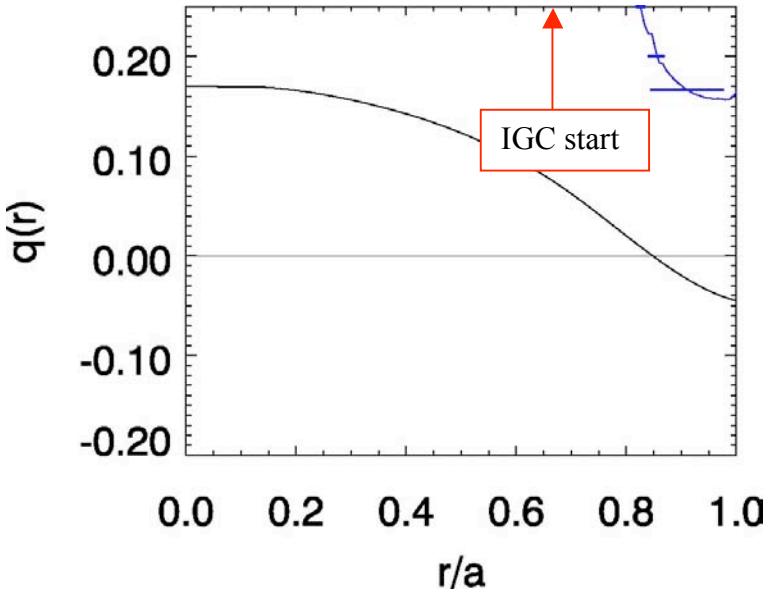


Figure 5.17 Ion starting at r/a = 0.8. The very large perpendicular velocity makes the mirroring effect dominant, however, the extreme *q*-shear results in very small islands.

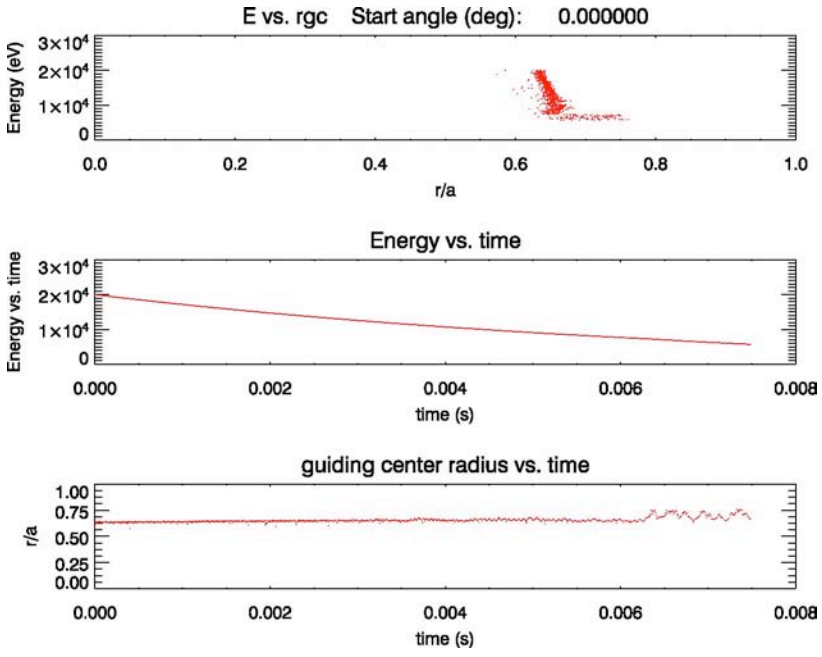


Figure 5.18 The r/a = 0.8 ion displays good confinement. Also the strong mirroring effect restricts the orbit to the low-field side of its initial position.

As a final note, the perpendicular gyro-radius, $\rho_\perp = \frac{m\mathrm{v}_\perp}{eB}$, (Fig. 5.19) has to be much smaller than the gradient scale length, $k^{-1} = \left(\frac{\nabla B}{B}\right)^{-1}$, to have the first order approximation of the ions guiding center motion, that we used to get the fast ion safety factor, q_{IGC}, be valid. However, $|k\rho_\perp| << 1$ (Fig. 5.20) is not strongly satisfied at large initial radius; k and $\rho_\perp$ both increase with r. Also, mirroring invalidates the definition of q here since there are no more "poloidal" transits. In simulation, the average magnetic moment is conserved, but the large gyro-radii at these locations and the effect of mirroring, result in highly irregular orbits that defy any simple analytical treatment.

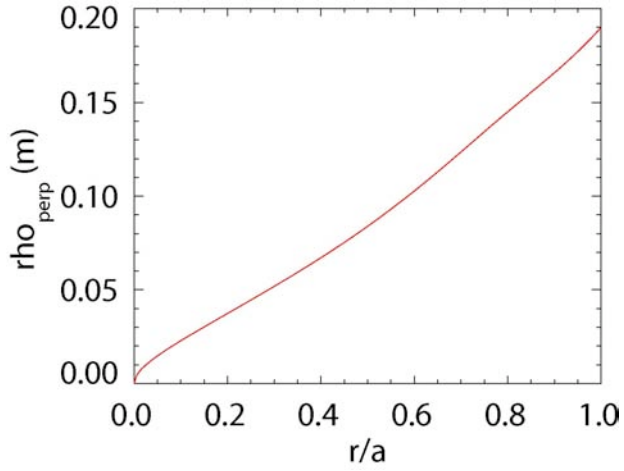


Figure 5.19 Perpendicular gyro-radius for 20 keV deuterium ions in tangential injection. On-axis, where injection is parallel, the value is zero and increases at larger radii due to more perpendicular velocity relative to the magnetic field as well as decreasing field strength.

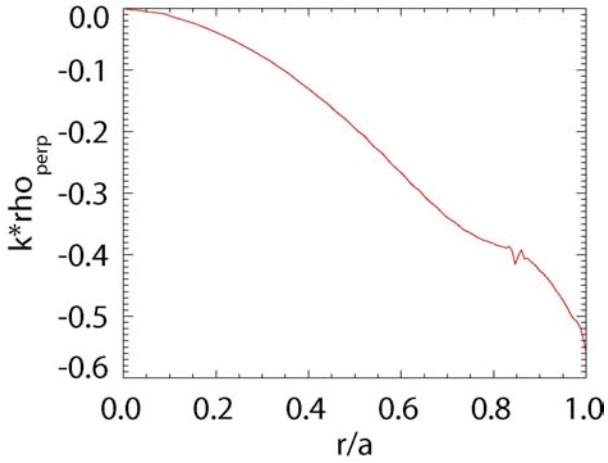


Figure 5.20 In the first order in gyro-radius expansion of guiding center velocity, the condition $|k\rho_{\perp}| << 1$ must be satisfied. We see that for on-axis ions where the variation in field strength in the radial direction, k, and the perpendicular gyro-radius, $\rho_{\perp}$, are both small, this condition is satisfied. Further out in radius, where both quantities increase, the validity of the expansion becomes less valid.

5.8.3 Island Trapping of Fast Ions

Before leaving the topic of co-injected fast ions into a standard plasma discharge, a few remarks about the interesting case of ions that are trapped in their own guiding center islands are in order. When one considers any island, magnetic or ion, the island may be non-overlapped, partially overlapped, or completely overlapped. If there is a portion that is not overlapped, the field lines (or trajectories) should experience good confinement. If outside the separatrix, we say that it is on a flux surface, whereas if inside the separatrix it could be thought of as being on a flux tube with the winding number of the appropriate island.

Fig. 5.21 shows the puncture plot of a fast ions guiding center that started inside one of the daisy-chains of small islands that form naturally in any nonlinear resonant system. As the position of the resonance moves in (from the ion losing energy) the islands move with it and drag the ion's guiding center along. Since it is inside the $n = 6$ island, this island, as we can see from Fig. 3.4 and Fig. 5.5, is never completely overlapped.

Future work could entail looking for island-trapped ions, as they would be very localized. Recent experiments with pellet injection suggest particles may be getting trapped in island structures in MST, as well as locally high temperatures routinely observed in islands via SXR tomography.

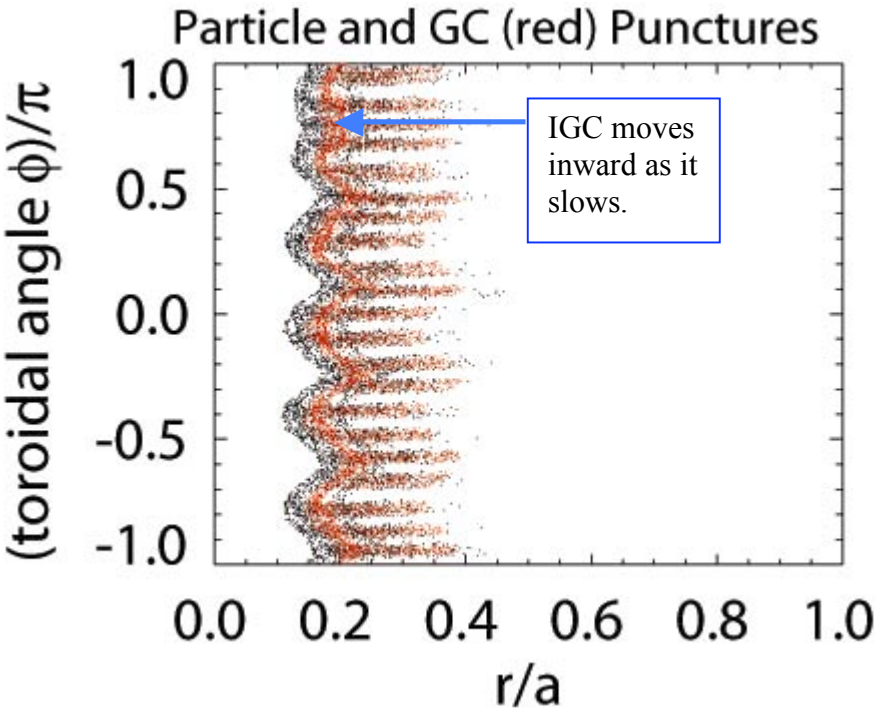


Figure 5.21 Fast ions can be dragged along in space with the changing positions of resonances. If an island retains its structure as the ion loses energy and the ion is trapped inside it, confinement can be quite good.

5.9 Confinement of Counter-Injected Ions

If we examine test particle confinement in counter-injection, we again first start with considering the q-profile. In this case $s_{||}$ is negative and so the shift in the ions q-profile is *down* relative to the magnetic field. Plotting the islands resulting from the standard perturbations, we see in Fig. 5.22 that the overlap is much more pronounced. This is because the radial density of the $m = 1$ resonant surfaces increases as q approaches zero.

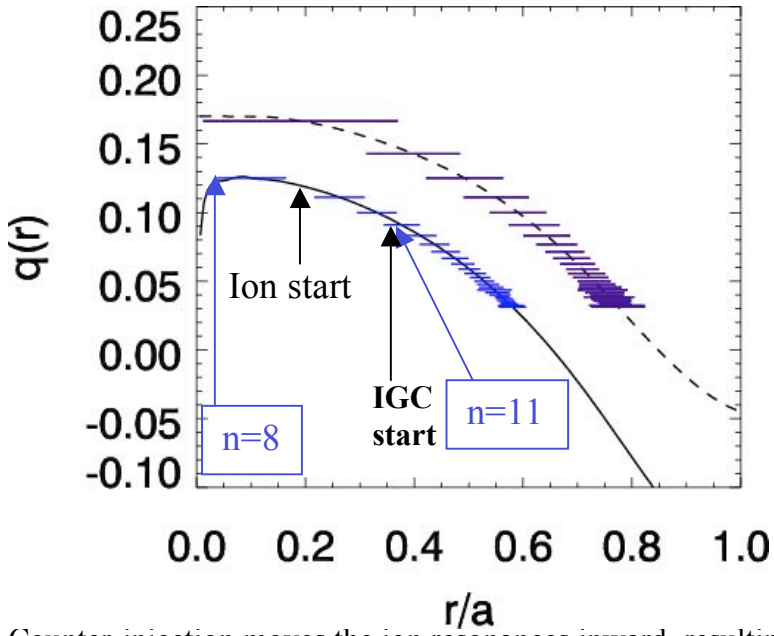


Figure 5.22 Counter-injection moves the ion resonances inward, resulting in more dense rational surfaces. Note that the overlapped region is further from the wall.

The observant reader would also notice that the ion islands are closer, and in general smaller, than their magnetic counterparts. With co-injection the ion islands are in general smaller, but spaced further apart (Fig. 5.5). While the island width is a function of several variables, all of which change from one radial location to the other, it is instructive to note the differences. In co-injection, the resonant surfaces move out, and $\frac{dq_{IGC}(r)}{dr}$ generally increase in magnitude. These effects tend to counteract each other. However, a fact that co-injection and counter-injection share is that the value of the magnetic mode at the ion resonance is less than the value at the magnetic resonance. For example, in co-injection, the $n = 6$ magnetic mode is about 25 Gauss where it is resonant, but it is only 20 Gauss at the ions resonance. The reason for this is that magnetic modes tend to peak at their resonant surfaces, which to state yet one more time, are not the same

as where the ions are resonant. With counter-injection the location of the ion resonant surfaces also move inward, which further decreases the island size. However, as one can see from Fig. 5.22, the islands are still significantly overlapped and one would expect stochastic ion trajectories.

It should also be stated that since the orbits are bent outward in the poloidal magnetic field, that if the ion is born at r/a = 0.2, its guiding center is at a greater radius, where the overlap tends to be stronger. Co-injection has the benefit of the opposite situation.

Once again, for our typical ion that starts at r/a = 0.2, we see in Fig. 5.23 that the orbit is indeed stochastic, as expected. However, one small benefit of counter-injection is that since the *q*-profile is shifted down, the resonances move inward. From the simulation it can be seen that while the orbit is stochastic, it does not get as close to the wall. The limit on the radial motion of the stochastic orbit would reduce the possible charge-exchange losses since the neutral density is edge-peaked. The simulation supports the hypothesis offered in Section 4.9, where the confinement was shown to be between the degree seen during co-injection and a sub-millisecond stochastic confinement time.

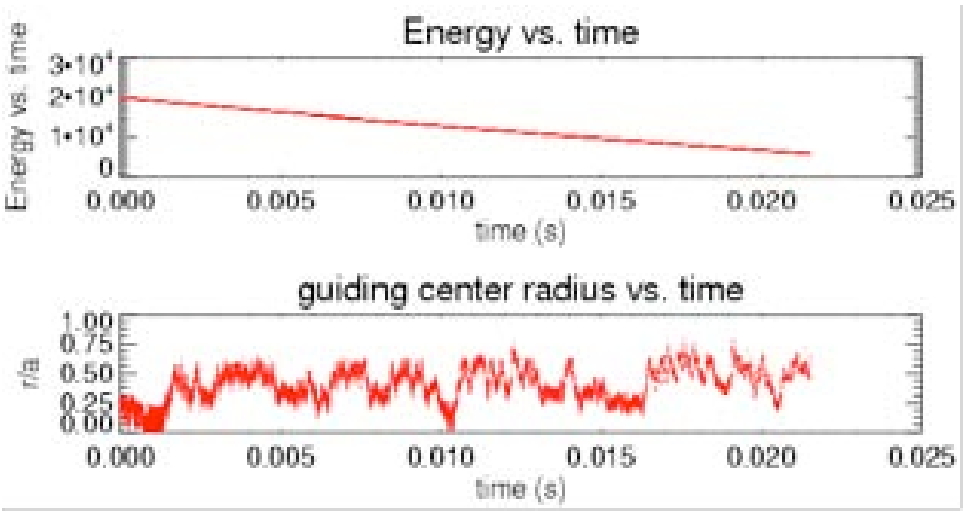


Figure 5.23 Position of IGC for counter-injection. The ion starts at r/a = 0.2 and is almost immediately stochastic.

5.10 Confinement Dependence On Magnetic Fluctuation Level

MST provides a test-bed for understanding the role of magnetic fluctuations because of the sensitivity of the magnetic topology on them. There are three basic levels of magnetic fluctuations available in common MST discharges. Aside from the standard level, we can use current profile control "PPCD" which has the effect of reducing fluctuation levels roughly a factor of 3-5. This was discussed in Section 3.1.5 when modeling magnetic stochasticity during PPCD.

We also have MHD relaxation events, or "sawtooth crashes" where the magnetic mode activity increases typically about a factor of 3. While PPCD is something we can control, the sawtooth crashes occur quasi-periodically and are natural phenomena for our RFP discharges. As such, any NBI injection scenario will have to deal with them and so the confinement picture under these levels of fluctuations should be considered.

5.10.1 Confinement in High Fluctuations (Sawtooth Crashes)

In the code, we can easily increase the amplitude of the magnetic modes by a factor of 3 to simulate a sawtooth crash. Avoiding the repetition that would come from analyzing fast ions starting at every radii, we can just examine our 'typical' ion again at r/a = 0.2. We see in Fig. 5.24 that it is confined for a short time, then immediately becomes stochastic and diffuses rapidly to the machine boundary. This is not unexpected from the size of the $n = 5$ islands and agrees with our established picture of island overlap (Fig. 5.25) and the temporal dependence on the islands position. The bottom frame in Fig. 5.24 shows the ripple in radius before the ion is lost. It is significantly larger than in the standard perturbation case due to the larger $n = 5$ ion island.

The island overlap picture shown in Fig. 5.25 suggests that even higher energy ions might have robust good confinement across a sawtooth crash. For ions at energies such that q would be slightly less than 1/4, there would be a large radial range of regular orbits as the overlapped $n = 5$ resonance would be further out in radius. The increase in island size during increased fluctuations would have to be larger to effect stochasticity on the centrally peaked fast ion population.

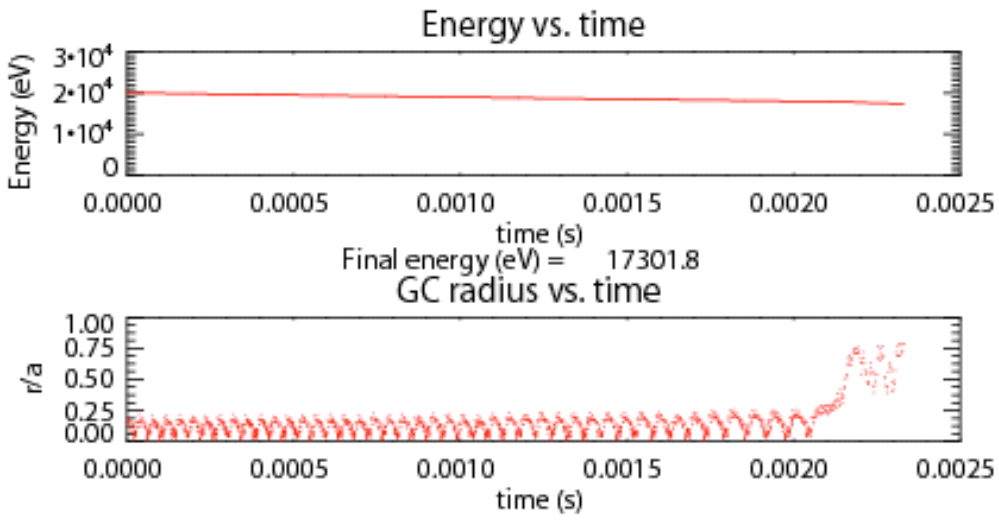


Figure 5.24 A 20 keV deuterium ion born at r/a = 0.2, but with magnetic fluctuations increased by a factor of three. Initially the orbit is not stochastic, then at about 18 keV becomes stochastic and diffuses out to the wall in a few tenths of milliseconds. This confinement time, once stochastic, is approximately what we see experimentally. As can be seen in Fig. 5.25, initially, at r/a = 0.2, there is no overlapped ion island, and hence we expect the orbit in the simulation not to be stochastic until it loses some energy.

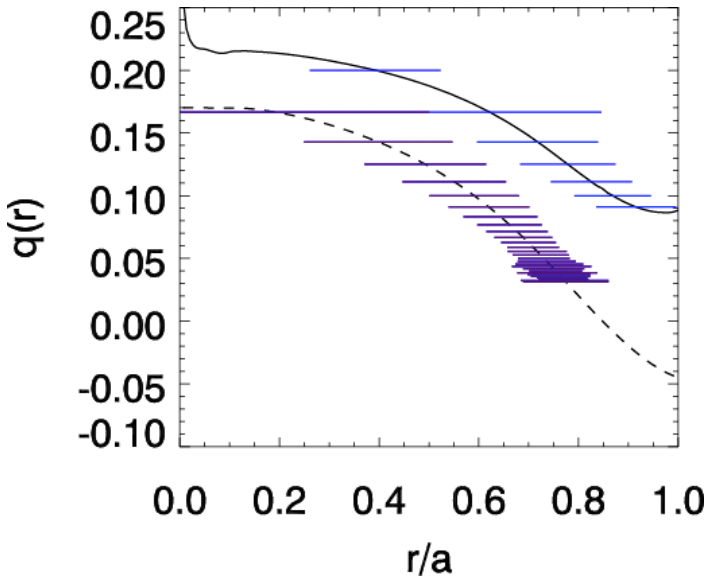


Figure 5.25 20 keV ion(solid black) (r/a = 0.2) and magnetic (dashed black) *q*-profiles and resonances with magnetic fluctuations increased x 3. The $n = 5$ and $n = 6$ ion islands are overlapped and result in orbit stochasticity when the islands reach the position of the guiding center.

5.11.2 Confinement in Low Fluctuations (PPCD)

We can also lower the level of fluctuations to mimic PPCD. Fig. 5.26 shows the ion *q*-profile and Fig. 5.27 shows the radial position vs. time of our standard ion in PPCD-like fluctuations. Not surprisingly, the ion retains its excellent confinement.

To get a rough handle on the impact of PPCD on fast ion confinement, we can characterize the confinement by looking at how much energy an ion deposits before becoming stochastic. For this we simulated different cases of ions under PPCD and standard conditions and compared the average amount of energy deposited as a function of starting

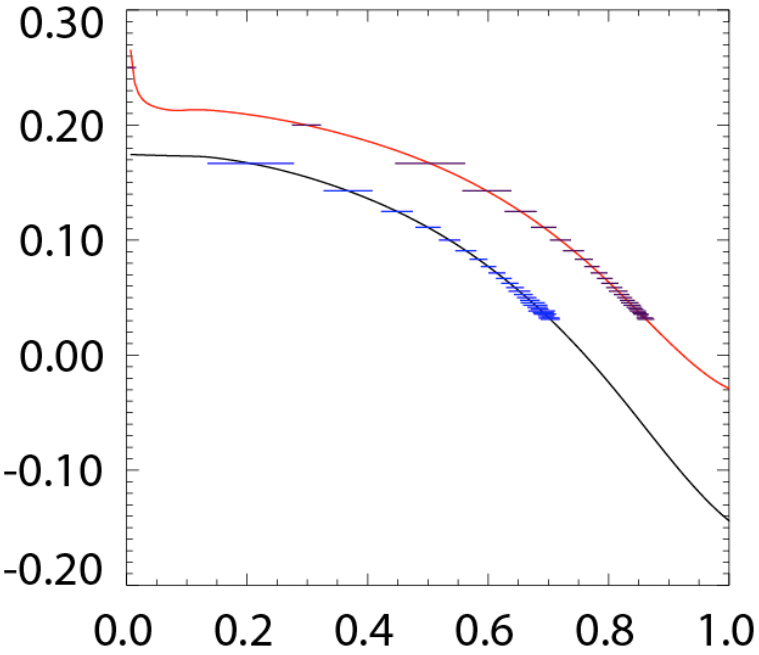


Figure 5.26 Island widths for magnetic lines (lower) and IGC's (upper) Fluctuation levels reduced a factor of 3 to simulate PPCD-like conditions.

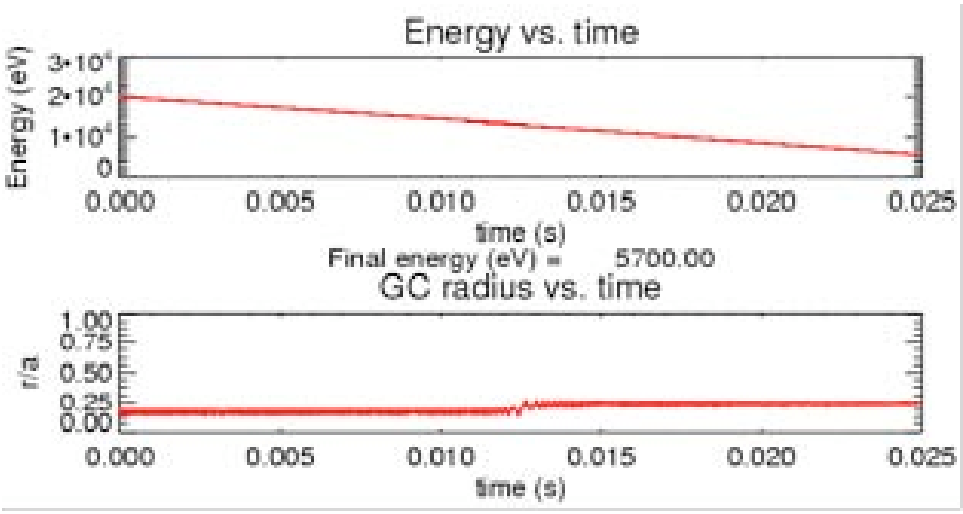


Figure 5.27 Improved confinement at lower fluctuation levels. The simulation ends before the ion is able to become stochastic.

radius. Fig. 5.28 shows that on-axis ions under standard conditions deposited about 70% of their energy, while PPCD ions deposited just over 80% as seen in Fig. 5.29. While fluctuations play the main role in when an ion becomes stochastic there are also subtle differences due to the fact that the equilibrium fields in PPCD are also different than in the standard case.

To avoid leaving the reader with the idea that PPCD is unimportant for NBI that is absolutely not the case. While the fast ions may be confined nearly as well, the bulk plasma, which in any NBI heating scenario must be well confined, benefits enormously from the confinement improvement from PPCD. This is discussed in detail in Section 6.4.

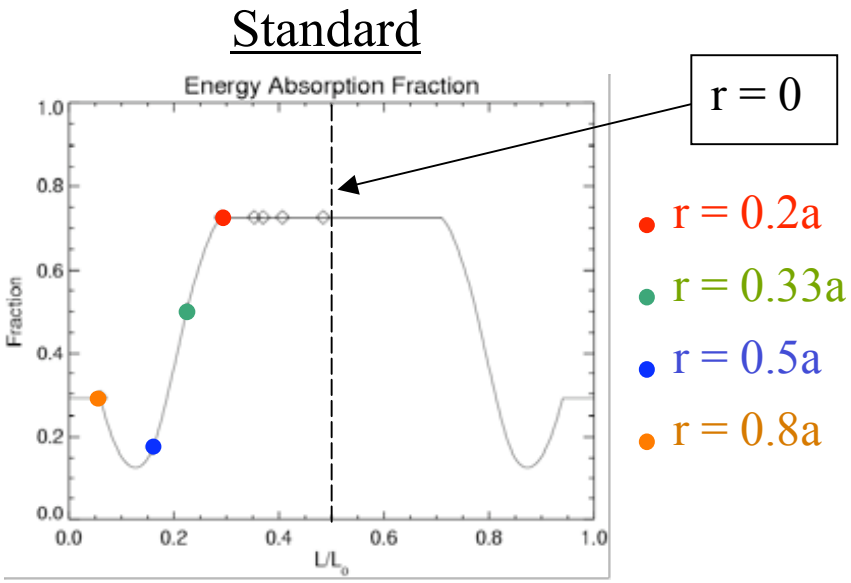


Figure 5.28 Energy lost before becoming stochastic as a function of starting radius. The x-axis is distance along a central injection chord to show the strong localization of particles at low r/a.

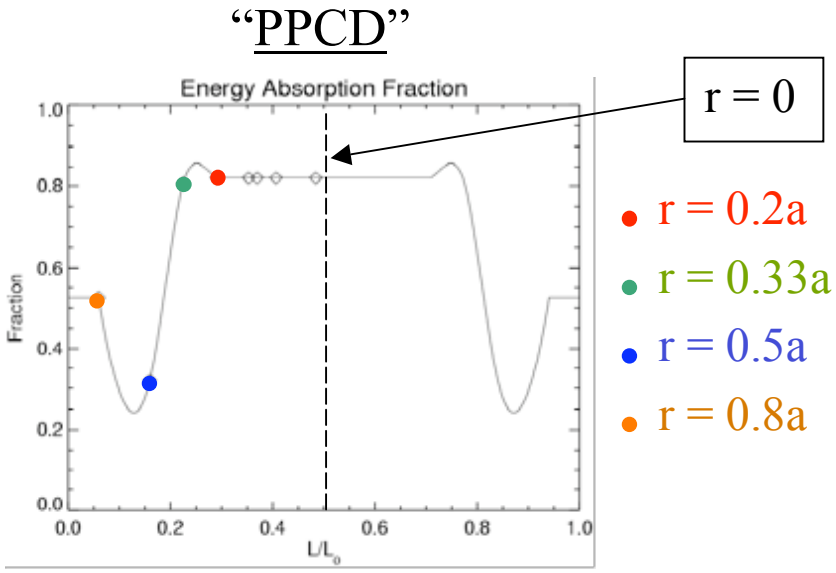


Figure 5.29 Energy absorption in "PPCD". The already good confinement of fast ions is little improved by going to lower fluctuations.

5.11 Radial Injection

A heating beam certainly would not be used for radial injection for the simple reason that the chord through the plasma would be minimized. This would result in poor absorption and quite possibly a problematic heat load at the beam footprint. Current drive would be minimal because it would be reliant solely on drifts as the ions have no inherent parallel velocity. Also the large perpendicular velocity, at all radii of ionization, would make first orbit losses severe, possibly causing localized heat load on the beam side of the vessel. Nevertheless, we can query about the confinement properties because some RFP devices[3] do have radial neutral beam injectors. Also diagnostic neutral beams, while not used as a source of fast ions as a primary goal, do introduce them into the plasma.

Our r/a = 0.2 ion, radially injected now, is shown in Fig. 5.30. We can see that the ion has superior confinement properties. The ion q-profile shown in Fig. 5.31 gives us indication as to why this may be. The large perpendicular velocity introduces mirroring, due to the radial gradient of the magnetic field. As can be seen in Eqn. 5.7 and mentioned in Section 5.3, a large perpendicular velocity drives q singular and very far away from the magnetic profile. Since the resonance widths are a function of q', and q' gets exceedingly large, the associated islands can be very small. This reasoning has its limitations because it's quite likely that the notion of rotational transforms in the vicinity of a mirroring point is ill posed, due to the interference of the ion motion from the mirroring condition, nevertheless, it seems a plausible explanation.

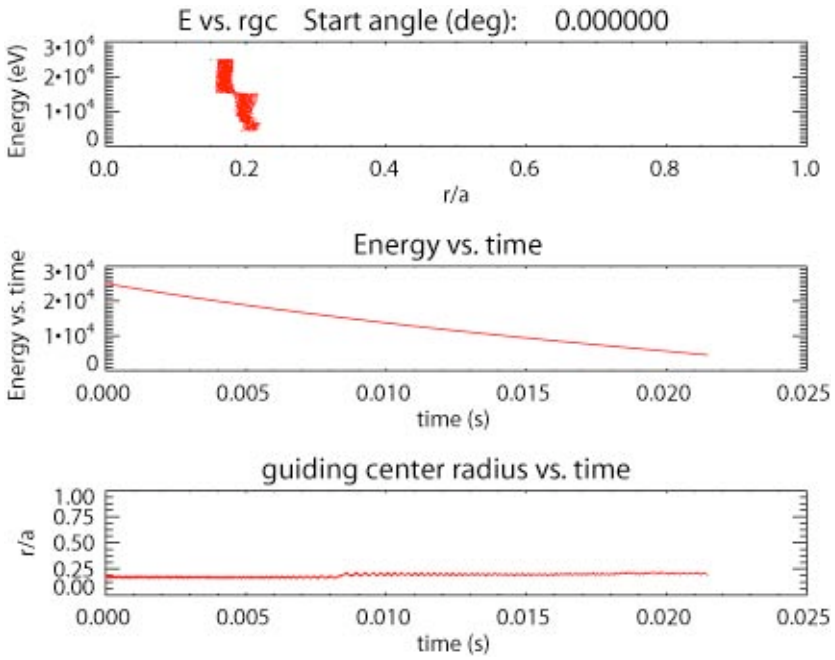


Figure 5.30 A radially injected fast ion can experience good confinement similar to co-injection.

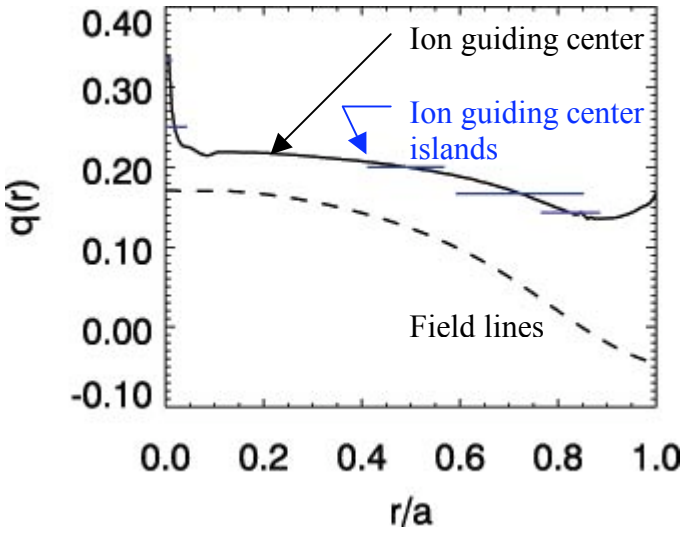


Figure 5.31 There is a significant radial range where radially injected ions are not stochastic. The lack of a curvature drift is offset by the increased gradient drift to maintain the shift in the safety factor.

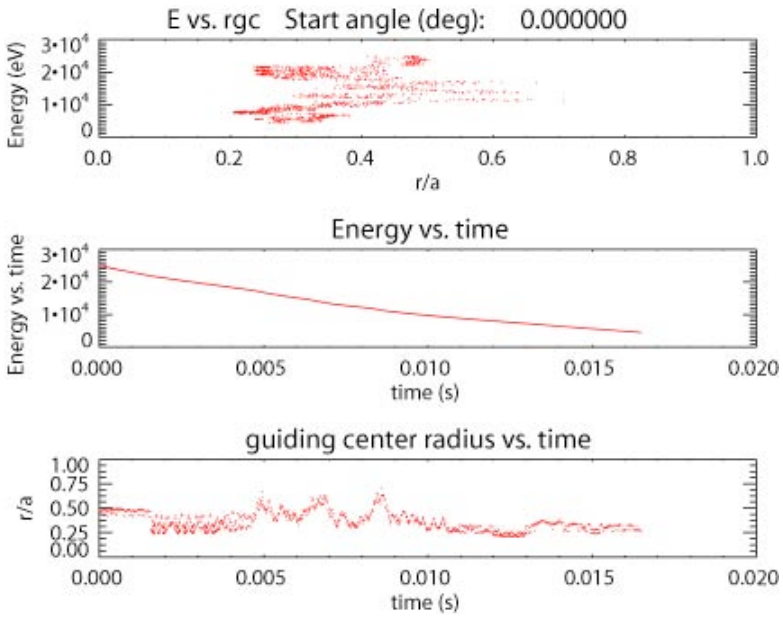


Figure 5.32 Radial injection with the ion starting at r/a = 0.4 (outboard). We see that it is stochastic right away. Note that the guiding center can't go further in than r/a = 0.2, this is due to mirroring.

Radial injection may benefit in many cases from q being large, but in reality the estimate of q from Eqn. 5.7 has a faulty assumption in it. We assumed that the perpendicular energy was very nearly 100% because injection was perpendicular to the equilibrium field. For the initial moment of ionization, this is the case, but the average perpendicular energy can be significantly less due to drifts, which will give the radially injected ion a parallel component to its velocity. It should be noted that the shift in the ion q-profile in the simulation of radial injection is to increase q, thus giving some approximation of co-injection.

Since we've discussed co, counter, and now radial injection, it is worth asking what the effect of treating the injection angle as a continuum rather than a fixed value would be. In other words, at what point does co-injection become counter-injection? A

naïve answer would be just on the 'counter-injection side' of radial injection, however, we saw that pure radial injection looks very similar to co-injection, so this must not be the answer. We can look at the value of q vs. injection angle, recalling that q_{co} is greater than $q_{magnetic}$ and $q_{counter}$ is less than $q_{magnetic}$. This experiment requires a judicious choice of starting radius. If we choose a start too close to the axis, the ion will cross the geometric axis with its gyro-orbits and this would highly distort the value of q. Also an orbit far out in radius would hit the wall as we approach perpendicular injection. We also want to do this in the absence of perturbations to be able to average the value of q over a long timescale without having to worry about stochasticity. Finally we will do this in a toroidal geometry, to have it be more representative. Toroidal simulations of ions in a perturbed field are discussed in the next section.

Fig. 5.33 shows q vs. injection angle for a near-axis ion. At the extremes, we see the typical values of q for co and counter injection. As we approach radial injection, angle = 0.0 on the plot, there is a discontinuity. The discontinuity occurs when the poloidal motion of the guiding center goes to zero, which happens when the poloidal component of the ion parallel motion is counteracted by the poloidal component of the guiding center drifts. In these instances, the ion has only toroidal motion and this is seen clearly in simulation.

If indeed there are good orbits when an ion is off-resonance from the magnetic field, as seen often in radial injection, perhaps the injection angle of –0.2 would result in many confined, albeit pathological, fast ion orbits. If one had some flexibility in the horizontal tilt in a radial beam, this experiment could be performed.

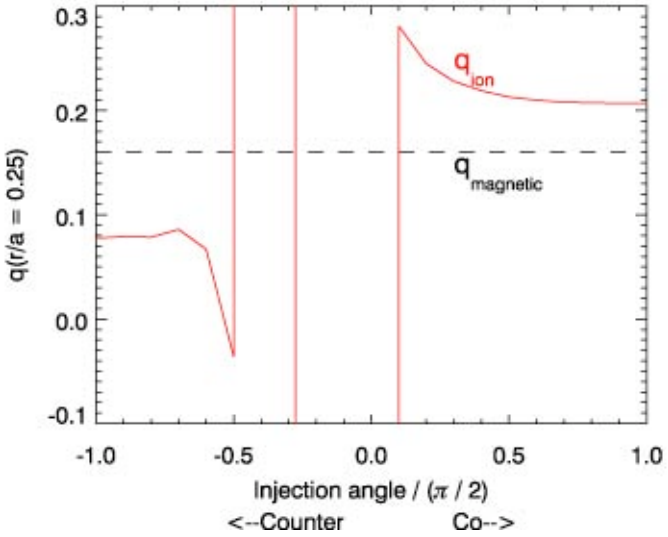


Figure 5.33 Dependence of q on injection angle. We start an ion at r/a =0.25 and vary the angle to go from co-injection to counter-injection. There are large discontinuities where the injection approaches radial. This is due to mirroring and q becoming ill-defined.

5.12 Toroidal Simulation

Here we wish to introduce a brief section on toroidal injection. This is not given a full treatment for the following reasons. First, the magnetic fields in the simulation are only quasi-toroidal. They take the cylindrical magnetic fields and introduce a Shafranov shift, which then re-expresses the flux surfaces as nested circles. The code does not solve the Grad-Shafranov equation self-consistently. Secondly, the DEBS simulation is done in a cylindrical geometry, so the question of the radial eigenmode structure in a torus becomes pertinent. Third, our analytical treatment of q, and the derivation of such from guiding center motion, is restricted to a cylindrical geometry. Also, a toroidal field, with

the appropriate variation in field strength vs. poloidal angle, would necessitate taking a proper average of the ion guiding center motion and could not be expressed as an instantaneous value as we are able to do in a cylindrical system. Nevertheless, armed with what we know from the cylindrical analysis, we can proceed.

Fig. 5.34 shows our standard 20 keV ion, r/a = 0.1, co-injected in a toroidal system. Similar to the cylindrical case, the orbit is well confined, with a discrete jump where the dominant perturbation goes out of resonance as the ion loses energy. In Figure 5.35 we see an ion which starts at r/a = 0.5 and has a brief period of confinement, but then becomes stochastic.

Fig. 5.36 is interesting as it is the same simulation as Fig. 5.35 except that the starting position is moved 11.7 degrees toroidally. This had the effect of putting the ion guiding center inside the $n = 5$ ion island. As the ion loses energy, the resonance moves inward and the ion goes along with it. At about 10 keV, the resonance that trapped the ion disappears, and the confinement improves further, shown by the tightness of the guiding center excursions. It is now only feeling a perturbation from the next dominant mode, presumably $n = 6$, which is some radial distance away from the guiding centers location. Finally Fig. 5.37 shows an example of counter-injection, r/a = 0.1, and as expected, the orbit is much more stochastic than the co-injection case, as previously shown in a cylindrical system.

The strong qualitative agreement between a cylindrical and a quasi-toroidal simulation allows one to trust the results of the cylindrical model as approximating the experiment, despite its limitations. The most significant divergence between the cylindrical and toroidal orbits has to do with the mirroring condition. With tangential

injection, this only occurs for particles that start closer to the edge of the device where their parallel velocity is quite small. There is also the note that because of the toroidal curvature, it is possible to have radial drifts. These lead to losses within a toroidal transit that didn't occur in the cylindrical case; however, these are only for ions born near the edge and are expected to have little impact on the overall fast ion population, as most ions are bore close to the axis.

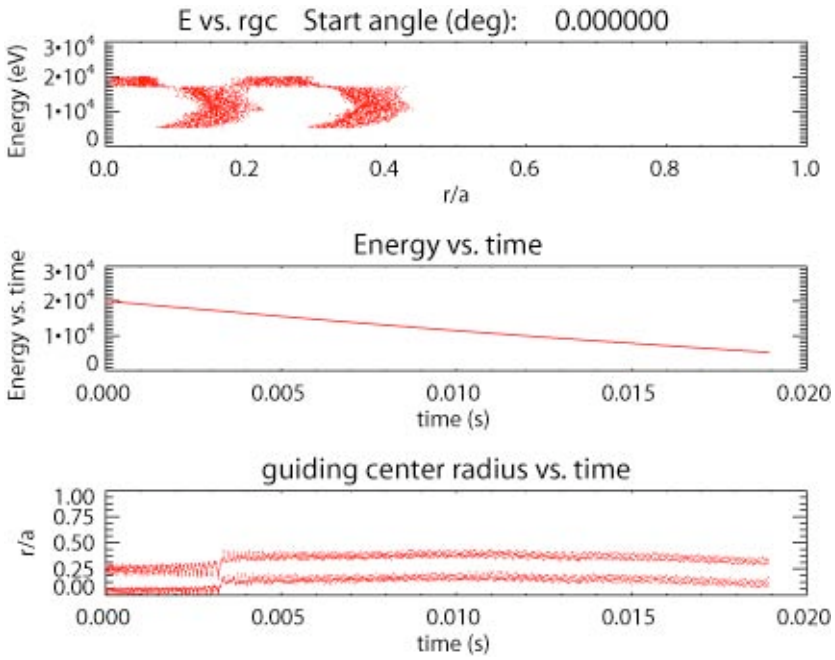


Figure 5.34 Ion at r/a = 0.1 in a toroidal simulation. Good confinement as in the cylindrical case. The double trace is from the Shafranov shift giving the orbit an asymmetrical location relative to the geometric axis.

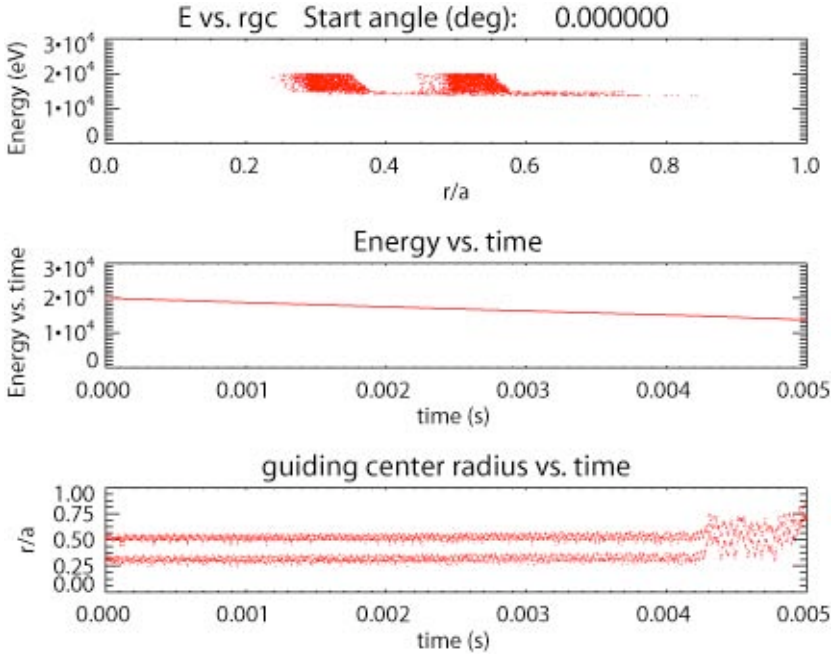


Figure 5.35 Ion at r/a = 0.5. Brief period of confinement then stochasticity similar to the cylindrical case.

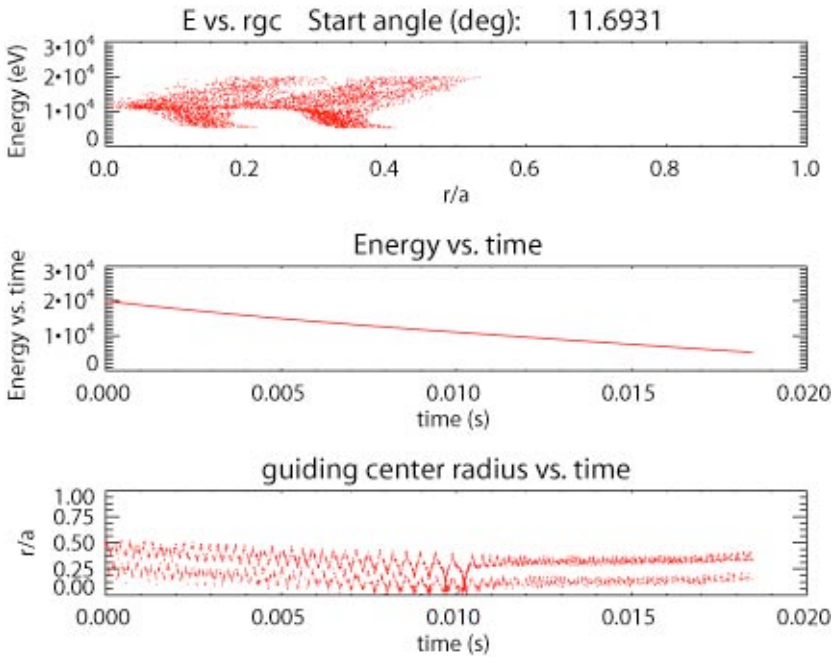


Figure 5.36 Example of ion trapping. The same simulation as Fig. 5.29 except that the ion started at 11.7 degrees toroidally vs. 0. This change in the phase relative to the perturbations put the guiding center inside an ion island.

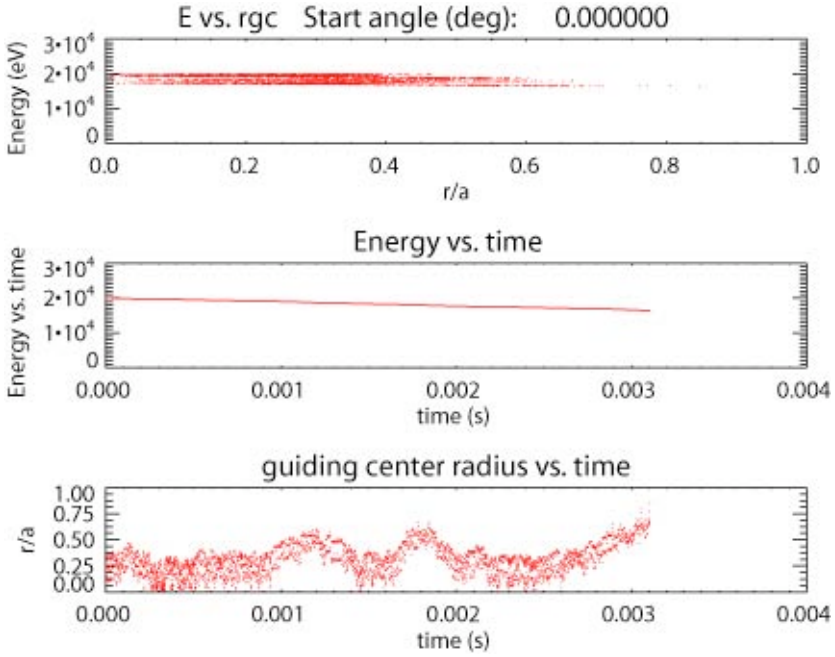


Figure 5.37 Guiding center orbit in counter-injection. The orbit is immediately stochastic and quickly reaches the wall, as expected from cylindrical counter-injection.

5.13 Summary

We have utilized an exact motion particle tracing code to predict and characterize fast ion confinement in a stochastic magnetic field. By considering the effect of curvature and gradient drifts on the ion guiding center, we saw analytically how this decouples the ion guiding motion from that of the underlying magnetic field. This was firstly described in terms of the ion safety factor. An analogy with magnetic islands was made to describe ion islands, and draw conclusions about its stochasticity based on overlap arguments.

This formulation was tested numerically and found to be valid, resulting in a broad parameter space of fast ion confinement. The time-dependence of the ion guiding center motion was explicitly considered, and as expected after losing energy the ion approaches the safety factor of the magnetic field and becomes similarly stochastic.

The dependence of confinement on magnetic fluctuation levels was discussed with the expected degradation in a high fluctuation case, but little change in the reduced fluctuation scenario due to the already superior confinement.

Counter-injection was found to have poor confinement properties overall because of the increased density of rational surfaces over most of the plasma radius.

Finally a brief discussion of radial injection showed that in general the confinement is similar to co-injection, with some instances of excellent confinement due to the effect of mirroring. The limitations of the concept of radial injection were also discussed as drift motions provide an average parallel component of velocity, which distorts the idealized picture.

References

1) A. I. Morozov and L. S. Solov'ev, *Reviews of Plasma Physics 2*, Consultants Bureau, 1966

2) K. Miyamoto, *Plasma Physics for Nuclear Fusion* (MIT Press, Cambridge, MA, 1989)

3) H. Sakakita, 30th Annual EPS Conference on Controlled Fusion and Plasma Phys.

6 NBI Effect on Plasma

Aside from the study of fast ion confinement, the main reason for performing NBI is to use it as a method of plasma control, be it for heating, current drive, instabilities, etc… This chapter examines effects that have been observed on the bulk plasma, what can be predicted, and some extrapolative modeling with an eye towards a larger beam system sometime in the future.

6.1 NBI and Tearing Modes

We can predict that given the short pulse and comparatively low total energy (relative to the Ohmic input power of MST) any changes from our particular neutral beam will be small. To attempt to maximize the beam's relative effect, we inject into low current plasma. However, the high frequency (~ 4 ms) of sawtooth crashes at low plasma currents (~250 kA) makes the fast ion confinement very poor. Also, with very short quiescent periods, it becomes more difficult to observe the NBI effect on a "steady-state" plasma condition. As a compromise, injection was performed just after the plasma begins its ramp-down. A typical plasma current waveform is shown in Fig. 6.1a as well as a typical beam window. Figure 6.1b shows the loop voltage across the toroidal gap, which is one of the signals that are commonly used to diagnose sawteeth. As can be seen, the sawtooth crashes are quite common but during the ramp-down, when the plasma is allowed to relax and there is no energy source to drive instabilities, we see that there are no crashes. The trade-off that must be accepted is that the equilibrium is changing, as opposed to an ideal case that would be during the plasmas flat-top with no sawteeth, but the changes are small and on a much longer timescale than our NBI phase.

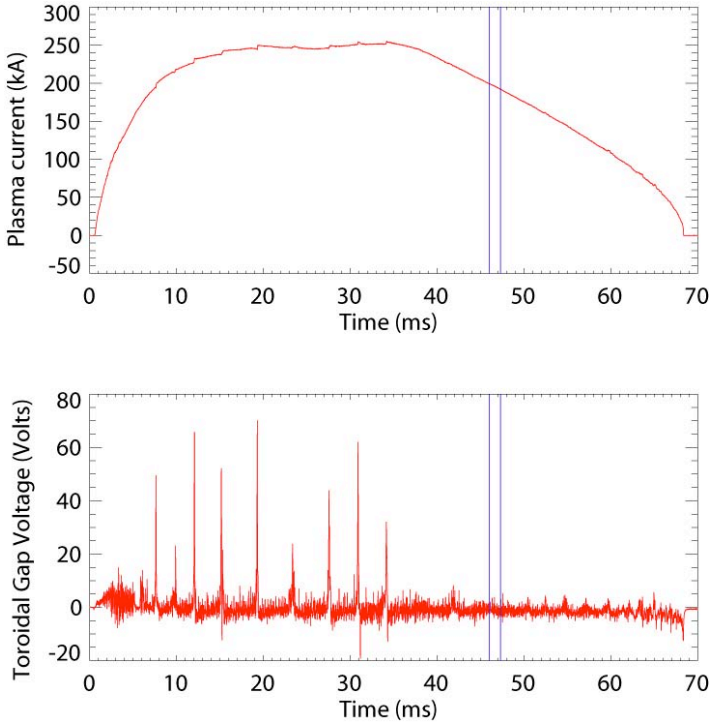


Figure 6.1 a) Top – Typical plasma current, the blue vertical lines denote the time when NBI is on. b) Bottom – Toroidal gap voltage, clear sawtooth crashes during flat-top, but none during ramp-down. The fluctuations at 46 ms are especially low.

We also start with injecting neutral hydrogen into the plasma as opposed to deuterium as before. The reason for this is that the slowing down time is half of what it is for deuterium (because of the mass ratio) and also because the fast ion current is a factor of $\sqrt{2}$ larger for the same beam current (because of the speed of hydrogen vs. deuterium at the same energy).

The equilibrium magnetic field profile can be estimated from taking the measured fields at the edge and assuming a force-free, cylindrical model. This is known as the

"alpha-model". In this simple reconstruction, a parallel current profile is assumed to be of a functional form

$$J_{\|} \propto \lambda\left(1-(r/a)^{\alpha}\right), \tag{6.1}$$

where λ and α are free parameters that are determined when one specifies the edge fields. Also, after having determined the magnetic field profiles, this defines the *q*-profile as well.

The reconstructed λ and α are shown in Fig. 6.2. We see that λ increases, as one might expect if adding to the current, which we are ostensibly doing during co-injection. In addition, α decreases, steepening the current profile, as one might expect if we are only increasing current near the axis, again as we expect based on the fast ion deposition profile. The parallel current profile according to Eqn. 6.1 is shown in Fig. 6.3, as well as the on-axis value of *q*, which will be important to the discussion shortly. One simple way to see why *q* decreases is that if we are driving toroidal current, that would serve to increase B_{θ}, and hence drive *q* lower.

Another observation was that the magnetic fluctuations at the edge changed in amplitude and rotation velocity during NBI and for some time thereafter. In Chapter 3 we used the toroidal magnetic pickup array to get the amplitudes of the fluctuations mode by mode. The same array gives information on the rotation speed.

In Fig. 6.4, an ensemble (~ 45 shots) taken with NBI (red) and without NBI (black) are shown for comparison. The strongest change is in the poloidal component of the $m = 1$, $n = 5$ magnetic mode. The amplitude (Fig. 6.4a) decreases about 2 Gauss, or 10%, and most notably the rotation (Fig. 6.4c) decreases from 8 km/s to around 2 km/s.

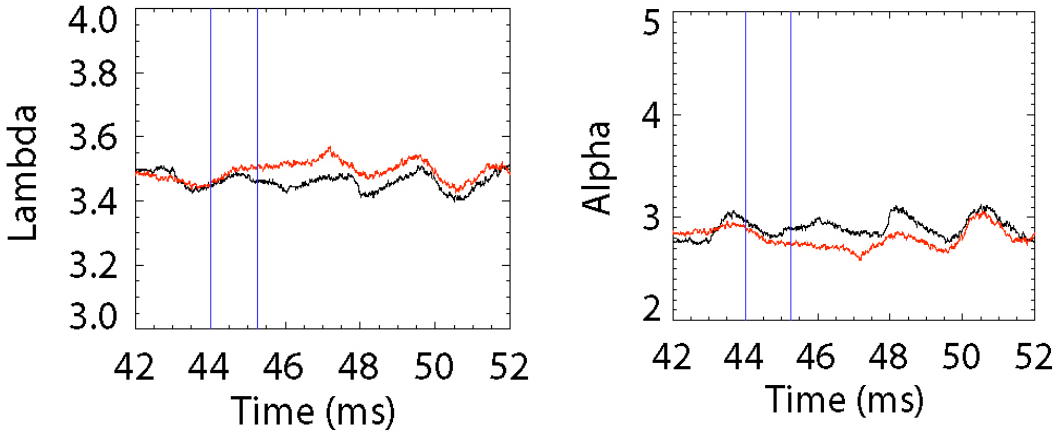


Figure 6.2 Parallel current profile parameters during NBI (red – beam on, black – beam off). The approximate profile becomes steeper and larger on-axis, consistent with our expectations with counter-NBI.

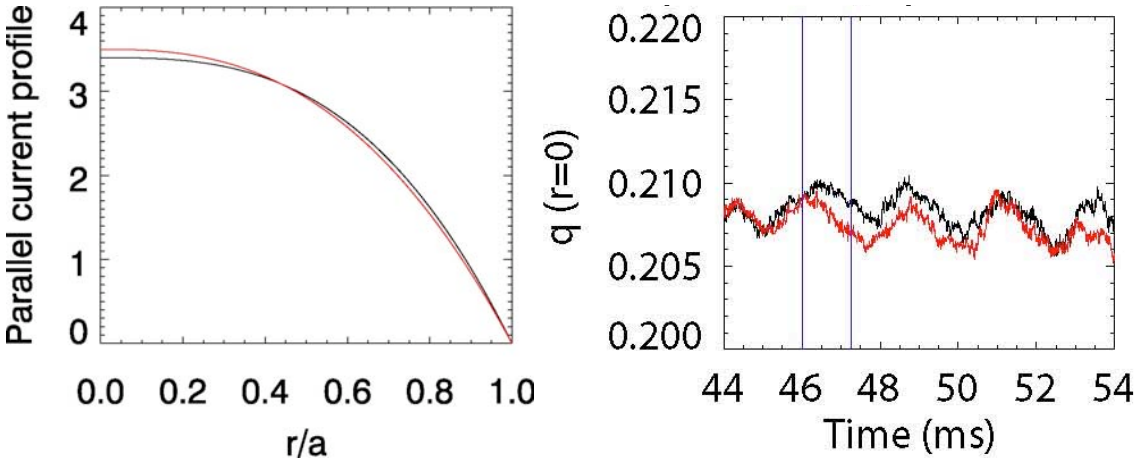


Figure 6.3 The reconstructed current profile (left) (red – beam on, black – beam off) results in a value of the on-axis *q* (right) that clearly decreases during NBI.

The toroidal component, Figs. 6.4b and 6.4d, has no obvious change in amplitude, but the rotation, from being nominally locked, decreases by about 5 km/s. This effect is not seen on any other modes in this type of shot. We can also note that the effect is a maximum at the end of NBI and then persists for several milliseconds.

Under slightly different conditions (q_0 ~0.19, beam on at 44 ms vs 46 ms), the effect is even more pronounced, as shown in Fig. 6.5. We see that the mode rotation quickly drops to zero, and additional beam ions do not further decrease it. This strongly suggests that it is not a simple torque that is being applied to the plasma flow responsible for the mode rotation frequency. The fact that the mode amplitude of the poloidal component decreases during the entire beam length, then gradually returns to its pre-beam value, wheras the rotation responded much faster, is not understood. Contrast this with the toroidal component that appears to have roughly the same temporal response as the poloidal component of the rotation.

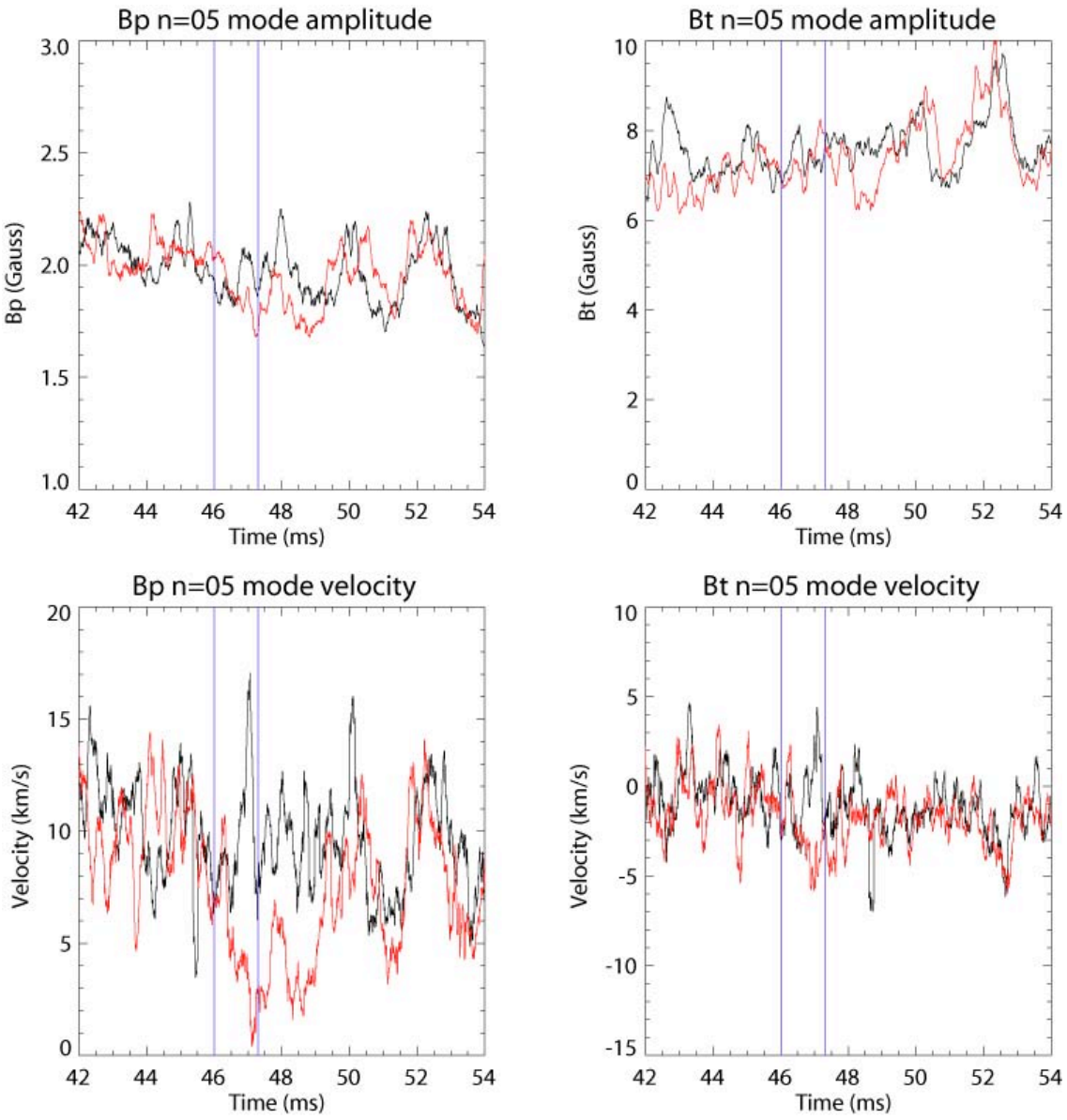


Fig. 6.4 Plasma rotation and mode amplitude response to NBI. During the NBI on (red) shots, the poloidal component of the $m = 1$, $n = 5$ fluctuation amplitude (Fig. 6.4a, upper left) decreases. The toroidal component amplitude (Fig. 6.4b, upper right) appears unchanged. The poloidal mode velocity (Fig. 6.4c, lower left) decreases sharply, and the toroidal rotation (Fig. 6.4d, lower right) also decreases. The difference in component velocities is because both $m = 0$ and $m = 1$ contributions are rotating, presumably at different rates.

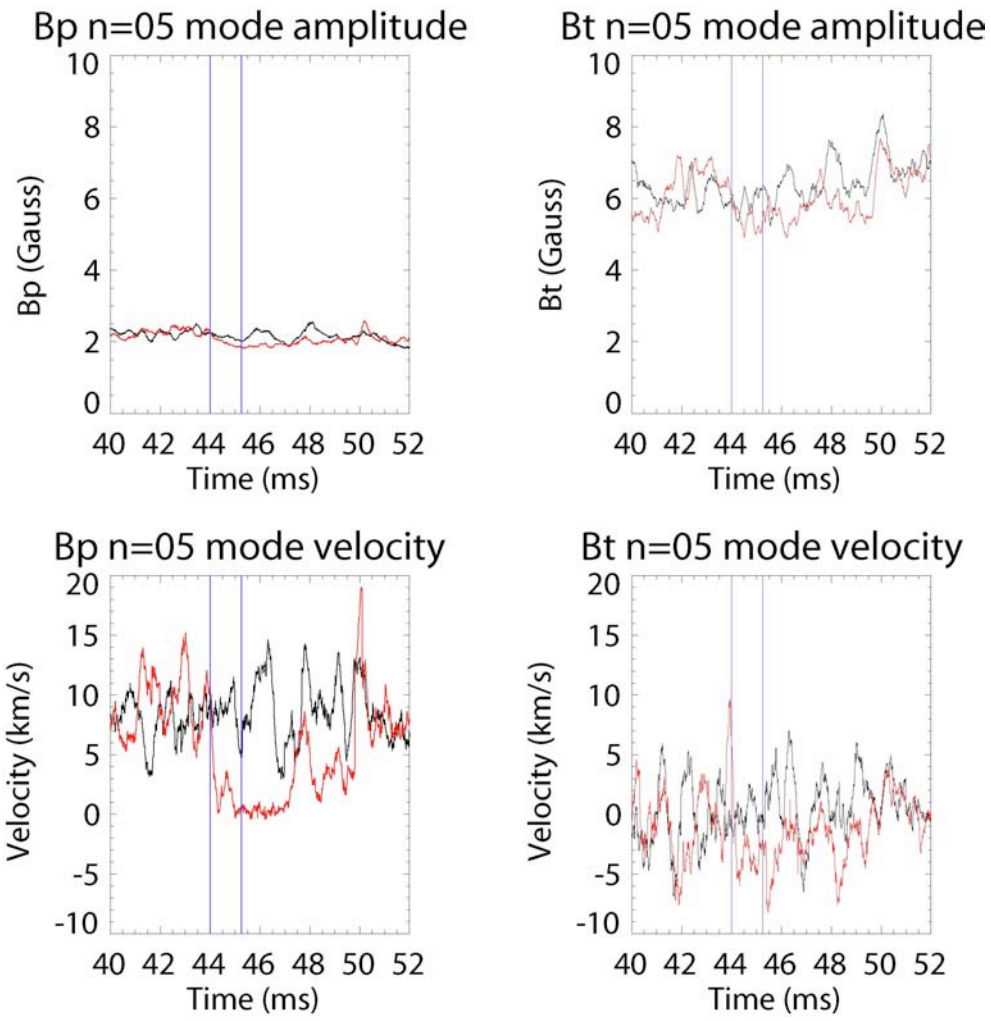


Figure 6.5 Another example under slightly different plasma conditions. Here the mode quickly locks, well before the beam is done firing. However, note that the amplitude of the poloidal component (upper left) does decrease until the beam goes off, then shows a gradual rise back to the pre-beam value.

Fig. 6.6 shows the magnetic amplitude and rotation of the $n = 5$ mode, for a single plasma discharge instead of an ensemble. We see that the rotation of the poloidal component intermittently jumps to ~20 km/s, and an associated increase in the mode

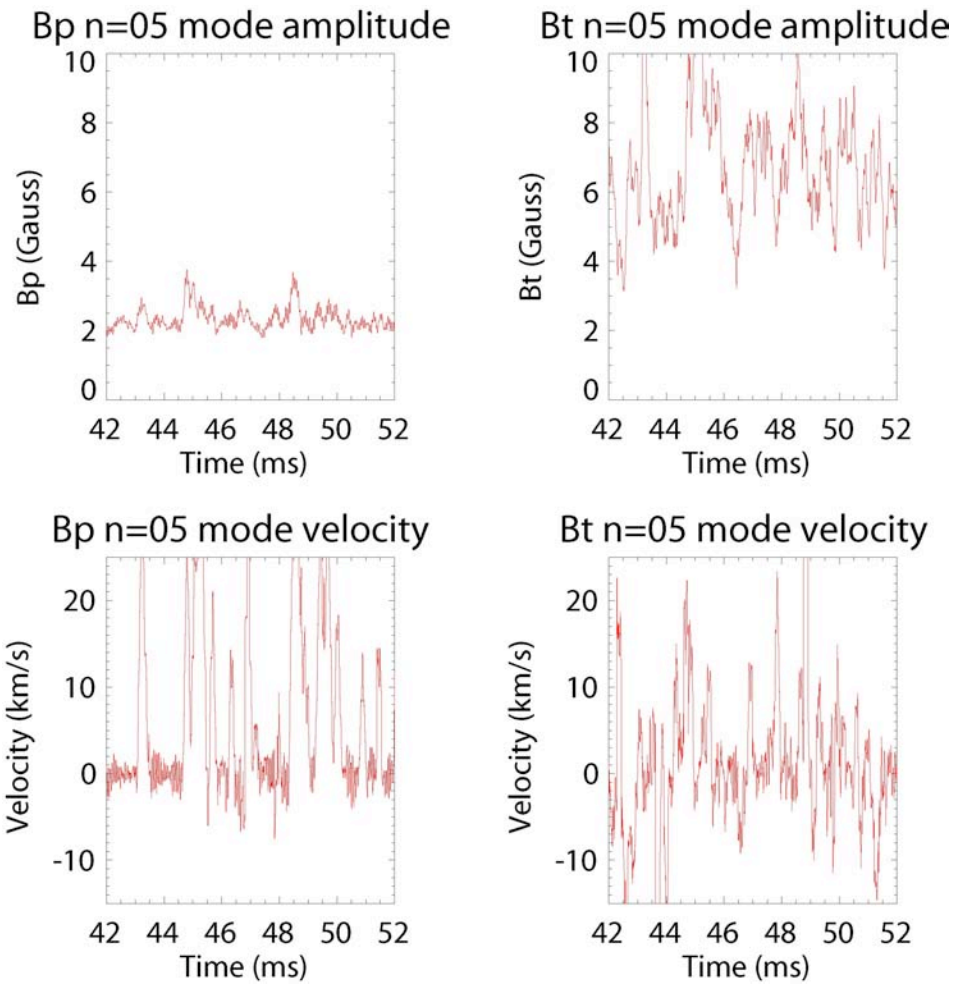


Figure 6.6 The amplitude and rotation signals for a single shot show that the mode is locking and unlocking, and when rotating the mode amplitude increase. Remember that the amplitude contains all m values and so gives a clear contrast between the $m = 0$, $n = 5$ and the $m = 1$, $n = 5$ components. It also demonstrates that in an ensemble, the mode amplitude is the statistical average of the $m = 0$, $n = 5$ and $m = 1$, $n = 5$ contributions.

amplitude is also observed. The $m = 0$ modes are locked to the plasma wall, and hence their contribution to the magnetic signal results in a zero velocity. This indicates that the $m = 1$, $n = 5$ mode is coming in and out of resonance, and when it is resonant, rotates at around 20 km/s, which is typical of the $n = 6, 7, ..$ mode rotation. This intermittance,

when averaged over an ensemble as previously shown, results in an average value of approximately 10 km/s. This average rotation speed then is the probability weighted contribution of the $m = 0$, $n = 5$ and $m = 1$, $n = 5$ modes. This point is important because it shows that the rotation and amplitude of the $m = 1$, $n = 5$ mode is not actually decreasing, it is simply less prevalent in the ensemble and thus the ensembled average rotation and amplitude approaches the non-resonant value.

This is very important; because in Fig 6.4 there is a linear decrease in rotation as the beam comes on, wheras in Fig 6.5 there is an immediate locking of the observed rotation. The first conclusion is that the $m = 1$, $n = 5$ mode is marginally resonant. When the mode is growing, it rotates at speed typical of the overall plasma rotation ~20 km/s, and the amplitude of the mode increases as well. Secondly, the beam ions are affecting this resonance condition. In the first case, Fig 6.4, a small number of beam ions aren't enough to affect the probability of it being locked (non-resonant) but as the number grows with time it becomes more likely to lock, hence the average value of the rotation speed approaches zero. In Fig. 6.5, where it locks immediately, the conclusion is that the mode is so close to being out of resonance, that very few beam ions are nessecary to take the mode out of resonance; hence we see the average rotation go quickly to zero.

However, this is predicated on the assumption that the $m = 1$, $n = 5$ mode is indeed marginally resonant. To get an idea if this is the case, we can consider Ampere's Law, evaluated at the vessel wall. By performing the manipulation shown in Eqn. 6.2, we are able to evaluate what the "average m" of a given n number is, based on the polarization of the measured magnetic fields at the wall. The values for several modes are shown in Table 6.1. While the low-n modes, 1,2,3 are clearly $m = 0$, and the high-n

modes, 6,7,8.., are clearly $m = 1$, we see that the $n = 5$ is somewhere in between, which also suggests marginal resonance.

$$\left(\nabla \times \tilde{b}\right)_r = k_\phi \tilde{b}_\theta - k_\theta \tilde{b}_\phi \cong 0 \quad \Rightarrow \quad \frac{n}{R}\tilde{b}_\theta = \frac{m}{a}\tilde{b}_\phi \quad \Rightarrow \quad < m >= n\frac{a}{R}\frac{\tilde{b}_\theta}{\tilde{b}_\phi} \tag{6.2}$$

The rotation and mode amplitudes shown in Fig. 6.4 and Fig 6.5 were taken at the edge, but as we have seen, $\tilde{b}_r$ has a radial structure to it. The fast ions are also dominantly in the core, so we wish to know what the mode amplitude is at that radius. We don't have experimental radial profiles of $\tilde{b}_r$ yet, but using FIR polarimetry we can measure the line-intergrated $\tilde{b}_r$ that we can take to be indicative of a core measurement (recall $\tilde{b}_r$ is nearly zero at the wall). Fig. 6.7 shows this measurement and reveals that indeed the $n = 5$ mode has decreased in amplitude. Additionally the $n = 6$ and possibly $n = 4$ modes have also decreased. This is in contrast to what we see in the edge signals, and may be the result of nonlinear coupling with the $m = 0$ modes.

n	$V(b_\theta)$(km/s)	$V(b_\phi)$(km/s)	b_θ	b_ϕ	<m>	m=1 resonant
1	0	-12	1.5	6.5	0.08	No
2	0	-12	1.0	7	0.10	No
3	0	-10	0.6	6.5	0.09	No
4	0	-10	1.8	6.5	0.35	~No
5	+10	0	2	6	0.55	Marginal
6	+25	+20	5	12	0.83	Yes
7	+15	+15	3.5	8.5	0.96	Yes
8	+20	+12	2	7	0.76	Yes
9	+15	+9	1.5	6	0.75	Yes
10	+15	+8	1.3	5	0.86	Yes

Table 6.1 Degree of resonance of the $m = 1$, $n = 5$ mode according to polarization of the measured magnetic fluctuations. The first column is the toroidal mode number. The second and third columns are the velocities of the toroidal and poloidal components of the mode. The third and fourth columns are the amplitudes, and the fifth column is the value of the "average *m*" as computed according to Eq. 6.2. The value of its "average m" does indeed lie between 0 (not resonant, most fluctuation due to $m = 0$ component) and 1 (resonant, most fluctuation due to $m = 1$ component)

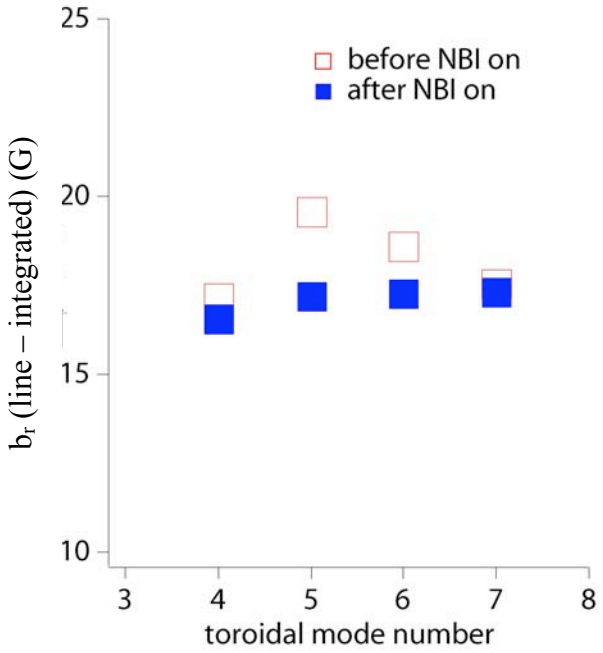


Figure 6.7 FIR polarimetry is used to get a line-integrated value of $\tilde{b}_r$. We see that there is a significant drop in the $n = 5$ amplitude, and interestingly also in the $n = 6$ and possibly $n = 4$. No decrease in amplitude in any mode besides the $n = 5$ is observed on the edge signals.

Next we can consider the observed timescale of the observed changes in rotation and amplitude. The change in rotation persists from 46 ms (when the beam comes on) to approximately 50 ms, where the beam on/off cases have nearly become identical. If this effect is due to modifying the current profile, then we should consider the confinement of the beam ions (which determines the number of charge carriers) and the velocity of beam ions (which changes through slowing down on the background plasma). We have no way of measuring the confinement of the hydrogen beam ions, but if we assume that it is close to deuterium beam ions, we can utilize the neutron detection method to estimate a beam particle confinement time. As shown in Fig. 6.8, the confinement is rather poor

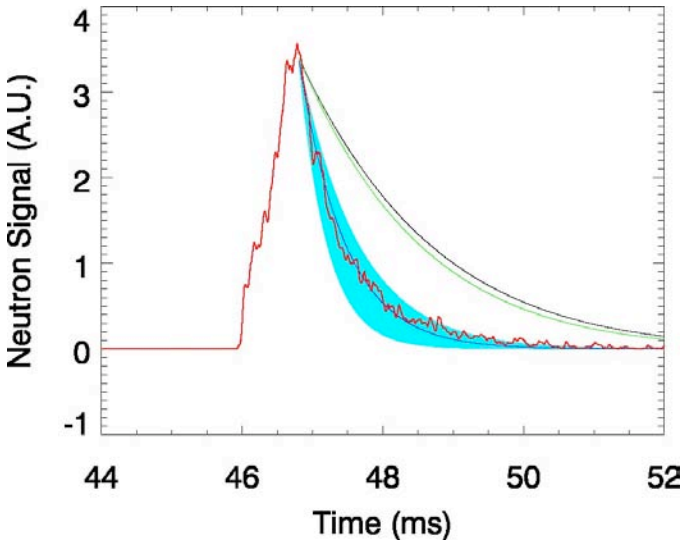


Figure 6.8 Neutron flux during low-current ramp-down. An electron temperature of 200 eV is assumed. The black curve (perfect confinement) and green curve (20 ms confinement time) are clearly not applicable. The dark blue curve is a 1 ms confinement time, with the light blue range surrounding it corresponding to a 0.5 ms (lower) and 1.5 ms (upper) confinement time.

and modeling of the neutron signal results in a confinement time of around 1 ms. This is of a reasonable timescale to explain the observed effect. We can also note that the slowing down time for hydrogen ions at an electron temperature of 200 eV (the assumed electron temperature since we have no measurement for this data) is around 4 ms.

If we believe that the raising of the on-axis current during co-injection is responsible for making the $m = 1$, $n = 5$ mode less likely to be resonant, then performing counter-injection would be an interesting test. In that case we would expect the parallel current to decrease on-axis, q to increase, and the mode more likely to be resonant and rotating, thus increasing the average value of the rotation speed. The experiment was

performed, and there was no change in either the magnetic signals (fluctuation amplitude or rotation) or the alpha model signals, which would reflect a change in the axis current.

Why did we see nothing with counter-injection? The fast ion density is approximately the same during counter-injection as co-injection, especially near the core where first-orbit losses are not occurring. However, we know that the confinement in counter-injection is not as good as co-injection as shown in Section 4.10 and discussed in Section 5.9. Again, the neutron flux for deuterium injection in co-injection was measured and the confinement was found to be extraordinarily bad. As shown in Fig. 6.9 the neutron flux mimics the beam ion input current, which indicates that the loss time is exceedingly fast, much less than a millisecond, which would be indiciative of constant stochastic loss of the beam ions. Neutrons can be produced by the beam striking the deuterium gas (deuterium ions in plasma + neutrals in plasma + layer of deuterium on the wall). The poor confinement is likely why we saw no effect from the injection whatsoever.

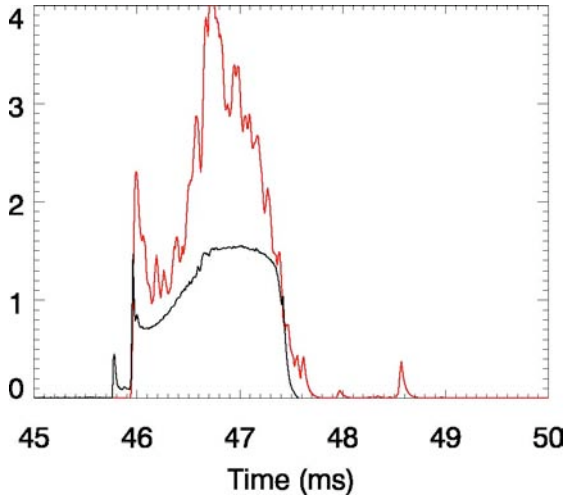


Figure 6.9 Neutron flux during counter injection into low-current plasma ramp-down. The neutron signal (red) has nearly the same shape as the beam current (black). This means the fast ions are being lost nearly as fast as they are being born. Fast ion stochastisity is likely, but also very high first orbit losses.

6.4 NBI Heating Model

During NBI, the electron temperature was measured by a multi-point Thomson scattering diagnostic and there was no measurable change. We can then consider what kind of particle confinement (both plasma and fast ion) and beam parameters would be needed for measurable plasma heating.

First we address why we saw nothing with the beam we have. The beam's input power, 400 kW, is a small, but non-negligible fraction of the Ohmic input power of about 2 MW. However, the energy transfer between the beam ions and the plasma is not instantaneous. We take the fast ion energy loss frequency (from Eqn. 4.15) to be ~ 60/s for deuterium (double that for hydrogen) and compare the maximum instantaneous heating from NBI vs. Ohmic power and find

$$\begin{aligned} P_{Ohmic} &\sim 2\times 10^{6} W \\ P_{NBI} &\sim \nu E = (60/s)\,(4\times 10^{5} W)\,(1.5\times 10^{-3} s) = 3.6\times 10^{4} W << P_{Ohmic} \end{aligned}, \tag{6.3}$$

therefore, we see no heating.

If we were to have a longer beam pulse, say 15 ms, and a larger beam current, say 100 A, this would bring us to within a factor of ten of the Ohmic power, under the above estimate. However, the long beam pulse means that all the energy is not instantly in the plasma, as one can assume so with the short pulse beam. Also, the question of plasma confinement arises to complicate a heating estimate.

Since we have a stochastic magnetic field, the electron heat diffusion is proportional to the thermal velocity. Moreover, since our fast ions preferentially heat the electrons, these heated electrons are able to diffuse more rapidly and become lost before equilibrating with the background ions. Therefore, the question of NBI isn't just a matter of the beam power, and fast ion confinement, but also of the plasma confinement itself. While a 0-D model (time only) of NBI heating is simple to implement, we note one other thing that is important. The fast ion distribution isn't global, it is very localized to the core region. This means that the density of fast ions can be quite high and the heating very localized. This could in theory lead to electron (or ion) temperature gradients that are assumed for our purposes here, to diffuse according to the heat equation. The following model was written by the author and is intended to be a simple 1-D model to predict and understand the requirements for significant heating with neutral beams in a diffusive plasma.

To model these processes we need to solve the system of differential equations

$$\frac{dn_b}{dt} = S_b \tag{6.4}$$

$$\frac{\partial n_b}{\partial r} = \frac{\partial n_i}{\partial r} = \frac{\partial n_e}{\partial r} = 0 \tag{6.5}$$

$$\frac{d(n_b T_b)}{dt} = D_b \nabla^2 (n_b T_b) - \nu^E_{e/b} n_b T_b - \nu^E_{i/b} n_b T_b \tag{6.6}$$

$$\frac{d(n_e T_e)}{dt} = D_e \nabla^2 (n_e T_e) + \nu^E_{e/b} n_b T_b - \nu^{Equil}_{e/i} (T_e - T_i) n_e + SS_e \tag{6.7}$$

$$\frac{d(n_i T_i)}{dt} = D_i \nabla^2 (n_i T_i) + \nu^E_{i/b} n_b T_b + \nu^{Equil}_{e/i} (T_e - T_i) n_e + SS_i \tag{6.8}$$

where

n_b, T_b Beam density and energy

n_e, T_e Electron density and temperature

n_i, T_i Ion density and temperature

$\nu^E_{\alpha/b}$ Energy loss rate of test particle (subscript *b*) with species alpha

$\nu^{Equil}_{\alpha/\beta}$ Energy equilibration rate of species alpha and beta

D_α Radial heat diffusion coefficient of species alpha

S_b Beam source term

SS_α Steady-state heating term for species alpha

As fast ions are born in the plasma, they of course begin to transfer energy to other particles. However, as we know, the rate of energy transfer is an exponential process. Therefore, especially for a long pulse neutral beam, the ions that came in at the end of the pulse are depositing energy at a different rate than the ones that came in at the beginning. While we could ignore this for our present beam because of its short duration, in general this should be taken into account. The entire fast ion population then is made of *n* different groups, which we call beamlets. These are followed independently and

evolved in the system of equations, but do not interact with each other. For most simulations we take $n = 100$ as being sufficient, less if considering a short pulse beam.

To advance the diffusive part of the equation we use the Crank-Nicholson method[4] that is second-order accurate and numerically stable for an arbitrary timestep. In radial coordinates then we have the following finite difference equation for the evolution of the temperature profile

$$\frac{T_i^{n+1} - T_i^n}{\Delta t} = \frac{D}{2}\left(\begin{array}{l} \dfrac{T_{i+1}^{n+1} - 2T_i^{n+1} + T_{i-1}^{n+1}}{(\Delta r)^2} + \dfrac{T_{i+1}^n - 2T_i^n + u_{i-1}^n}{(\Delta r)^2} \\ + \dfrac{T_{i+1}^{n+1} - u_{i-1}^{n+1}}{2r_i(\Delta r)} + \dfrac{T_{i+1}^n - T_{i-1}^n}{2r_i(\Delta r)} \end{array} \right). \tag{6.9}$$

where i is a radial position index, n is a time index, Δt is the timestep, r is the actual radial position and Δr is a radial step size.

The Dirchelet boundary condition

$$T_{Wall} = 0 \tag{6.10}$$

and the regularity condition at the axis

$$\frac{dT_{Axis}(r)}{dr} = 0 \tag{6.11}$$

are sufficient to solve the diffusion equation across one timestep. A full derivation of the finite difference equation and its solution method can be found in Appendix C.

Before going to the results of the code, some thought must be put towards what value of D to use. For electrons, we can use the electron heat diffusion stated in Chapter 2 of $D_e \cong 400\ m^2/s$. For thermal ions, whose drift motion is not appreciable at their energies, we can again assume they move along stochastic field lines giving $D_i \cong D_e\sqrt{(m_e T_i)/(T_e m_i)}$. Ambipolar effects on particle transport are neglected.

We can begin first with what the model predicts for a short pulse neutral beam. Figure 7.8 shows the on-axis electron temperature vs. time. The sharp rise during the beam on time is due to the fact that the source of heat for the electrons is increasing linearly, and the electron heat hasn't had time to diffuse. When the beam turns off, both the energy source from the NBI decreases (due to slowing down of fast ions) and also the electron heat has time to diffuse outward, lowering the axis temperature. The sharp drop at around 18 ms is where the fast ions are taken out of the equation as they reach the lower energy limit for the simulation. While one might say that the model is at that point invalid, an interesting consideration is that if the fast ions were to become stochastic and lost (perhaps due to slowing as we have previously seen), the heating from them might have a very similar profile. Of course, one would have to properly account for electron and ion heating during the sawtooth as well.

Heat diffusion is a very strong effect, and by improving the plasma confinement, we expect to be able to attain higher temperatures. In Fig. 6.11 the diffusion coefficient for the electrons was lowered from 400 m^2/s to 10 m^2/s.

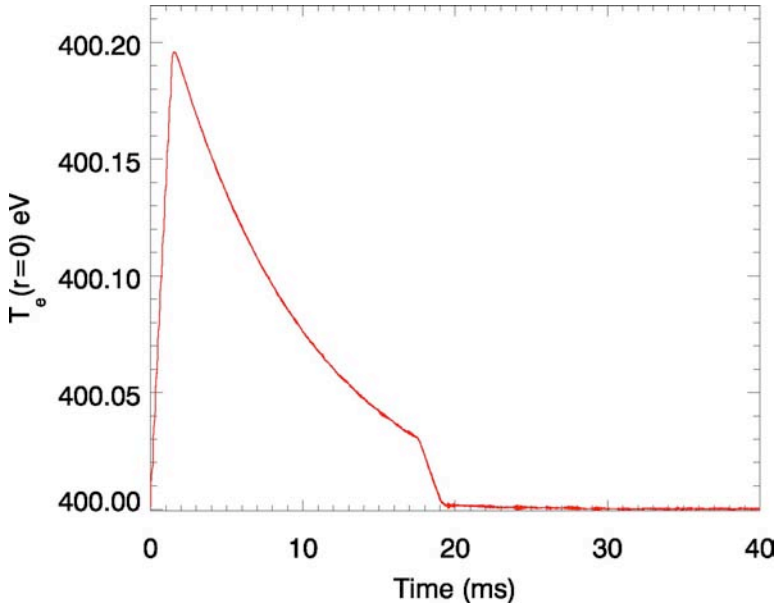


Figure 6.10 Electron temperature during NBI. 20 A, 20 kV, 1.5 ms of NBI results in very little heating. The sharp drop at 18 ms is from beamlets being removed from the simulation due to thermalization.

If we were to have a more powerful heating beam, such as 50 kV, 100 A, 15 ms, then we can determine what sort of temperatures to expect in that case. Figure 6.12 shows the on-axis electron temperature during high-power NBI. While possibly even measurable, the very modest heating is rather small considering the amount of NBI power being used. The situation changes when "PPCD" plasmas are considered. Again, we lower the electron heat diffusion coefficient to 10 m^2/s. The resultant profiles are shown in Fig. 6.13, with the electron (blue), plasma ion (green) vs. radius, at two points in time. The dashed lines are the initial temperature profiles, the solid lines are the profiles taken when the axis electron temperature is at a maximum.

Now the electron temperature has an increase of just over 200 eV, with the ions having a 20 eV change. At these fast ion energies the vast majority of the

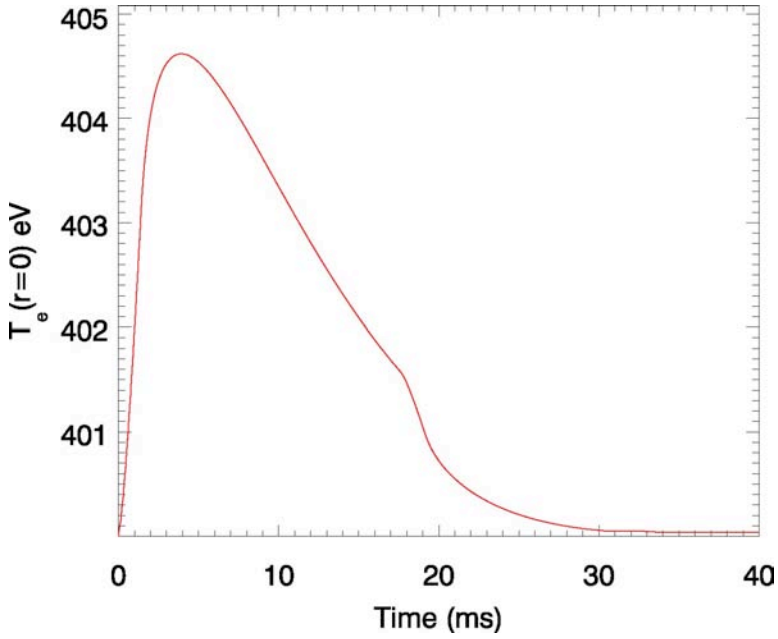


Figure 6.11 For "PPCD-like" confinement (electron heat diffusion a factor of 40 lower than standard, there is a significant improvement in the maximum temperature, but still overall very small.

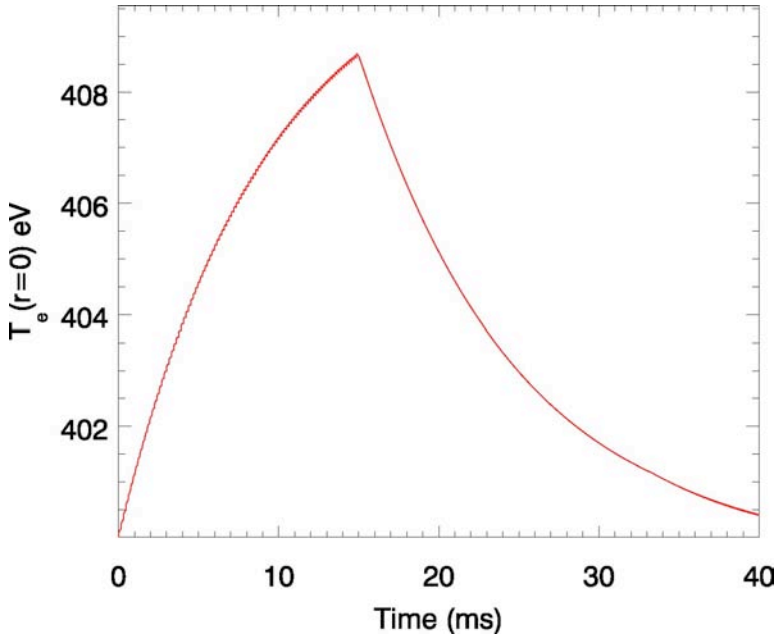


Figure 6.12 For a high power (50 keV, 100 A) long-pulse (15 ms) beam, under standard confinement, there is still only very small central heating.

energy is going into the electrons. The initial electron temperature corresponds to a low-current PPCD plasma, as the good confinement properties of PPCD result in higher plasma temperatures. The spread of the electron (and ion) heating is plainly from the radial diffusion of the NBI deposited energy. Fig. 6.14 and Fig. 6.15 show the evolution of the on-axis electron and ion temperature for this case.

Finally in Fig. 6.16 we show the result of increasing the electron temperature to 1 keV and the ion temperature to 500 eV before NBI. This approximates high-current PPCD. Again, PPCD allows for measurable heating (~90 eV for the electrons), but the differential heating is smaller due to the fact that the higher electron temperature lowers the collision rate between the fast ions and the electrons.

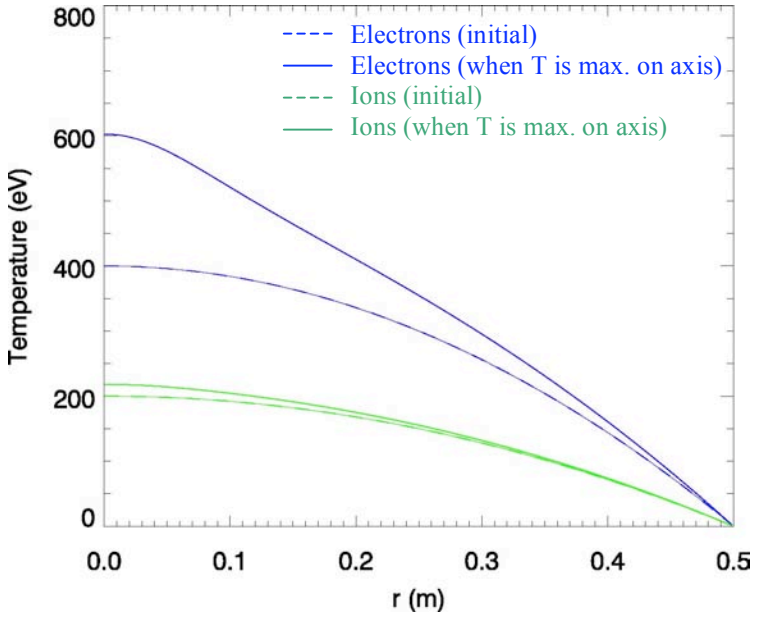


Figure 6.13 Temperature profiles for high-power NBI into low current PPCD. With good plasma confinement, significant heating is possible. The effect of heat diffusion can be seen, as the electron heat has diffused out to where there is no direct heating of the electrons and ions from fast ions.

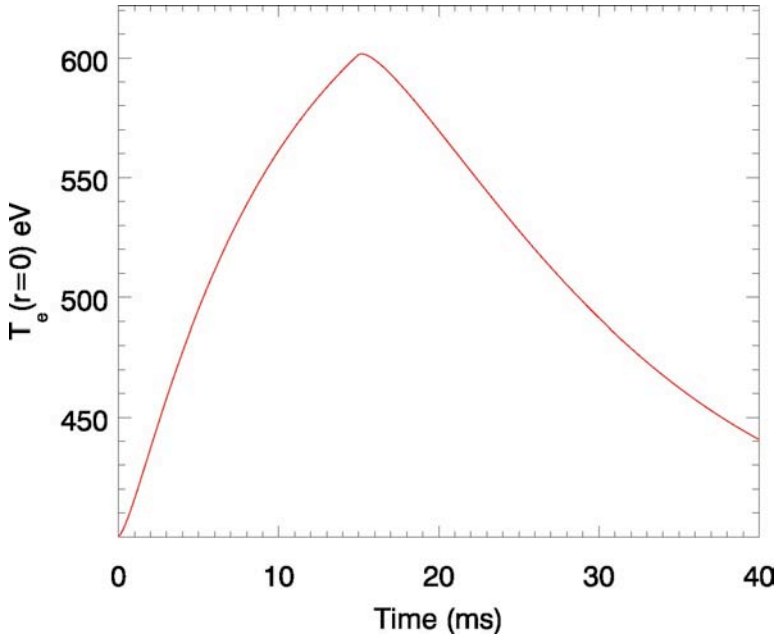


Figure 6.14 High-power NBI into low-current PPCD. The axis electron heating is a significant increase over the steady-state value.

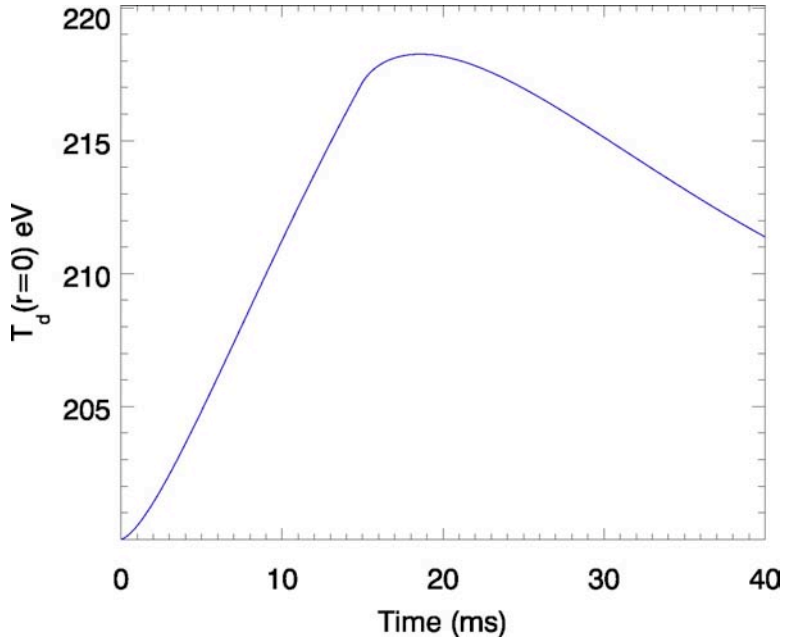


Figure 6.15 The axis ion temperature profile for high power NBI into low-current PPCD.

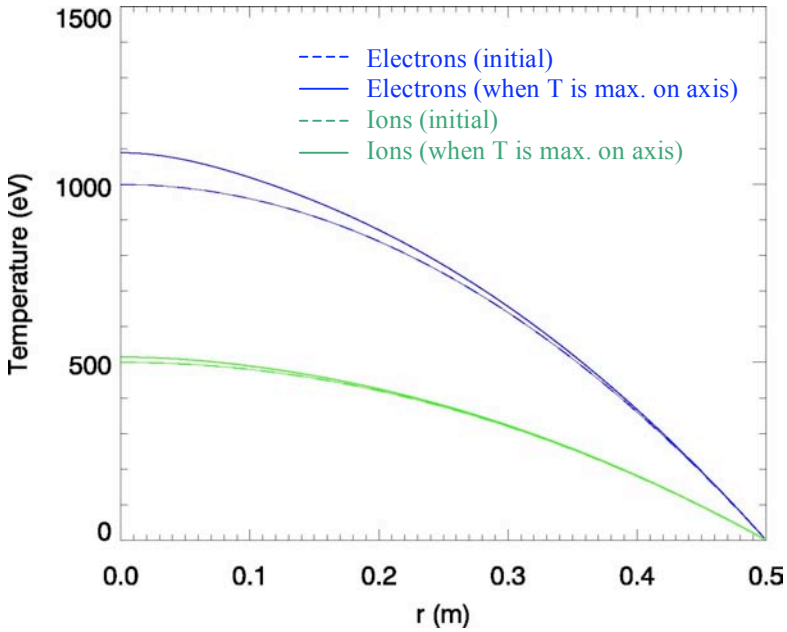


Figure 6.16 The profiles for high power NBI into high-current PPCD. The heating is less than the low-current PPCD case because of the smaller collision rate of fast ions with electrons at these higher plasma temperatures.

6.3 Summary

Neutral beam injection into a low-current plasma during the ramp-down phase was performed to look for global changes in the plasma due to fast ions. Evidence of current drive was observed, and likely restricted to the core. Also measured was a change in the $m = 1$, $n = 5$ tearing mode amplitude and rotation. Analysis of the mode's resonant condition suggests that the fast ion current modifies the local current profile in such a way as to make the marginally resonant mode more likely to be non-resonant. This explains the observed decrease in amplitude and rotation. The reverse case was not seen during a counter-injection experiment, and neutron flux measurements show that the confinement of the fast ions is likely far too poor for any significant effect.

A 1-D model of NBI heating has been written and results for many injection scenarios and levels of injection power were shown. With a high-power heating beam into PPCD-like plasmas, increases of central electron temperatures of 100-200 eV are possible. While a stronger beam source is necessary, the study shows that good plasma confinement remains the strongest requirement for efficient use of NBI from a heating point of view.

References

1) P. Gohil, K. H. Burrell, A. B. Hassam, T. H. Osborne, Plasma Phys. Control. Fusion **38** 1243-1247 (1996)

2) R. E. Bell et. al., IAEA-CN-69 / EXP1 / 13 (R)

3) R. W. Harvey, M. G. McCoy, J. Y. Hsu, A. A. Mirin, Phys. Rev. Lett. **47** 2 (1981)

4) Numerical Recipes (Fortran Edition) (1990), p. 638

7 Conclusions

The foremost physics result presented in this thesis is that fast ions in a stochastic magnetic field can have confinement superior to that of the background plasma. Small gyro-radius particle transport in a stochastic magnetic field is understood through the well-known Rechester-Rosenbluth model. The substantial drift velocities of larger gyro-radius particles decouple the orbits from the stochastic magnetic field and can result in good confinement.

The experiment performed used NBI as a source of fast ions and measured the resultant neutron flux from the fast ions fusing with the background plasma ions. The neutron flux provides a measure of the fast ion population and as such the temporal response of the neutron signal allowed us to infer a confinement time of the fast ions. The model used to describe the neutron flux assumed that the energy loss of the fast ions was purely due to Coulomb collisions with the background plasma. A fast ion confinement time of 20 ms was found to describe the observed neutron flux. This suggests that the fast ion orbits are not stochastic even though the background plasma is believed to be so.

Further experiments were performed to study the confinement of fast ions under varying conditions. Under low magnetic fluctuations, present during PPCD, the confinement time improved to 30 ms. During a sawtooth crash, when magnetic fluctuations are larger, a confinement time of 0.4 ms was estimated, consistent with stochastic losses of fast ions. Reversing the direction of injection relative to the plasma current, or counter-injection, was performed and the confinement time in this case was 4 ms. It is believed that the counter-injection confinement time is due to a hybrid of good

confinement of fast ions and increased charge-exchange losses. Finally radial injection was performed and the confinement time found to be 18 ms, similar to that of co-injection.

Numerical simulation of particle orbits in a stochastic magnetic field strongly supported the observed experimental results. By simulating the magnetic field, both equilibrium and perturbation, we quantified the magnetic stochasticity in terms of a magnetic diffusion coefficient, whose value was found to be ~10^{-4} m. The magnetic diffusion was consistent with previous measurements of electron heat transport and agreed with the Rechester-Rosenbluth formulation. However, for fast ions in the same stochastic magnetic field, their orbits were not stochastic under the conditions that approximate the experiment. By varying the magnetic fluctuation and injection geometry as described previously, we observed confinement properties that agreed with the experimental results.

Analytical study of fast ion orbits was done by solving the guiding center equation of motion in the RFP magnetic field. The significant drifts of large gyro-radius particles were found to take the guiding center motion out of resonance with the velocity fluctuations imposed by the background magnetic field. In analogy to the overlap of magnetic islands, the lack of ion guiding center island overlap was found to be the explanation of the improved confinement observed both experimentally and numerically.

NBI current drive was measured in some low-current plasmas and agreed with analytical predictions. Observed changes in magnetic mode rotation and amplitude were also measured, and the presence of NBI current drive under these conditions serves as a possible explanation.

Appendix A

Derivation Of Magnetic And Guiding Center Islands

This Appendix describes how the presence of a perturbation in a trajectory's phase space can lead to what are typically known as islands. These islands, which when converted to real space structures, when overlapped lead to stochastic orbits. This is important for magnetic field lines because particles moving along field lines are poorly confined in a stochastic magnetic field. It is also important for particles in and of themselves, which can exhibit resonances with regard to a perturbing force, and can become stochastic.

We will begin with a treatment of magnetic field lines, in a sheared slab approximation, and then repeat the treatment for ion guiding centers, noting differences where they occur. The formulation in the context of field lines is quite general as it is a mapping, and so is as good as any other mapping one might choose for illustration. The drift approximation for guiding centers is another example of a mapping, with the low-energy limit recovering the field line result. When going to higher energies, the guiding center motion and the field line motion become decoupled, but it is still a mapping, undergoing a transformation via drift contributions.

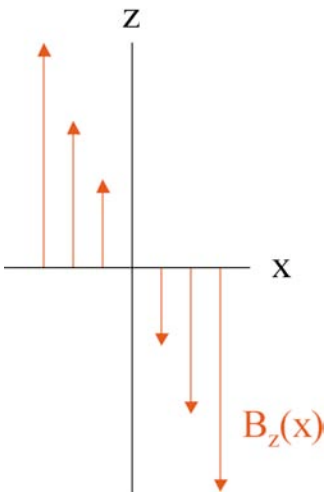


Figure A.1 Sheared slab magnetic field. We let the resonant surface be at $x = 0$, and the strength of the toroidal field be zero at $x = 0$. The gradient of the axial field is then shown schematically by the length of the arrows.

The sheared field is the toroidal component, which we will take to be in the axial or z, direction. We will also impose a perturbation in the radial direction. The field in this 'sheared' cylindrical geometry is then

$$B(x,z) = B_x(z)\hat{x} + B_y\hat{y} + B_z(x)\hat{z} \tag{A.1}$$

and the field line equation (in the x,z plane) are

$$\frac{dz}{dx} = \frac{B_z(x)}{B_x(z)}. \tag{A.2}$$

Assume the x-component is the result of a magnetic perturbation of the form

$$B_x(z) = A\sin\left(\frac{nz}{R}\right) \tag{A.3}$$

To have the same form as the sheared slab model we let $x = r - r_s$ such that $x = 0$ at our resonant surface and z is now the axial direction. We will also Taylor expand B_z since this assumes we are close to the resonant surface $\left(\vec{k} \bullet \vec{b} = 0\right)$

$$B_z(x=0) \cong 0 + x\frac{dB_z(x)}{dx}\bigg|_{x=0} + \ldots \tag{A.4}$$

Rearranging the field line equation (A.2) we have

$$B_z^{/} x dx = A\sin\left(\frac{nz}{R}\right)dz \tag{A.5}$$

Integrating (A.5) we get

$$\frac{1}{2}B_z^{/}x^2 = \frac{AR}{n}\cos\left(\frac{nz}{R}\right) + Const. \tag{A.6}$$

There are three classes of lines that this equation can trace out. Open lines which do not cross $x = 0$, but are perturbed. Closed field lines which are not allowed to make periodic transits due to being turned back quickly by the perturbation. Finally, there is the special field line that separates these, which is known as the separatrix and defines a "magnetic island".

Evaluating at $x = 0$, $z = 0$, which is on the separatrix line, we can solve for the constant, giving

$$\frac{1}{2}B_z^{/}(x)x^2 = \frac{AR}{n}\left(1 - \cos\left(\frac{nz}{R}\right)\right) \tag{A.7}$$

This gives the radial displacement of the separatrix as a function of z.

The maximum excursion of this line in x is the half width of the magnetic island.

$$W_{\frac{1}{2}} = 2\sqrt{\frac{AR}{nB_z^{/}(x)\big|_{x=0}}} \tag{A.9}$$

To write (Eqn. A.9) in the usual form we take the radial (x in the sheared-slab model) derivative of Eqn. 2.1 at the rational surface,

$$q'(x) = \left.\frac{rB_z'(x) + B_z(x)}{RB_\theta}\right|_{x=0} \overset{x=0, r=r_s}{\Rightarrow} \frac{r_s B_z'}{RB_\theta(r_s)} \tag{A.10}$$

Substituting this into (A.9) we find the well-known island width formula in a cylindrical screw-pinch.

$$W = 4\sqrt{\frac{A_{1,n}}{B_{\grave{e}}(r_s)}\frac{r_s}{n\left|q'(r_s)\right|}} \tag{A.11}$$

In the event that adjacent magnetic islands overlap, there is a bifurcation in the allowed solutions for the field line. The overlapped resonances lead to what is known as "strong stochasticity", c.f. Chirkov, which is of a global character (the combined width of the overlapped resonances) and is what is responsible for the commonly known stochasticity in many magnetic confinement configurations.

At this point we turn our attention to the resonances for ion guiding centers. In Appendix B we present the equation of motion of the ion guiding center, to first order in Larmor radius. The components of this motion are used below in the derivation, so it is implicitly a first order calculation. As above, the magnetic field perturbation is taken to be first order in some smallness parameter, which we commonly take to be the experimentally realized magnetic fluctuation level.

So now we begin again with the equations of motion, which for an ion guiding center, in the x-z plane, can be written as:

$$\frac{dx}{dt} = v_x, \frac{dz}{dt} = v_z \Rightarrow \frac{dx}{dz} = \frac{v_x}{v_z} \tag{A.12}$$

where $\vec{v} = (v_x, v_y, v_z)$ and the subscript *x* has replaced *r* in the cylindrical representation to connect with the sheared slab formulation.

In a cylindrical geometry, there are no radial drifts. This is because the curvature vector is purely in the radial direction, and the gradients in magnetic field strength are also just in the radial direction. This permits the approximation that the guiding centers radial velocity is just its parallel velocity along the radially perturbed magnetic field.

$$\frac{v_x}{V} \cong \frac{b_x}{B} \tag{A.13}$$

where $b_x = A\sin\left(\frac{nz}{R}\right)$, as before.

Expand around the resonant surface ($\vec{k} \bullet \vec{v} = 0$);

$$v_z(r_s + x) \cong 0 + xv_z'(x)\Big|_{r_s} + \ldots \tag{A.14}$$

We then have for the equation of motion in the x-z plane

$$v_z'(r_s)xdx = \frac{V}{B}b_x dz = A\sin\left(\frac{nz}{R}\right)\frac{V}{B}dz \tag{A.15}$$

Again integrating and taking the maximum value, we find

$$\frac{1}{2}\left|v_z'(r_s)\right|x^2 = 2A\frac{R}{n}\frac{V}{B} \tag{A.16}$$

So the half-width of the ion guiding center island is given by:

$$W_{1/2} = 2\sqrt{\frac{AVR}{B\left|v_z'(r_s)\right|n}} \tag{A.17}$$

This is the half-width for the size of the ion guiding center island separatrix. To further refine the expression and put it into quantities we are already familiar with, we begin by writing the derivative of *q*, for the ion guiding center, evaluated at the resonant surface, as

$$q'_{IGC}(x)\Big|_{x=r_x} = \frac{d}{dx}\left(\frac{x\mathrm{v}_z}{Rv_\theta}\right) = \frac{r_s \mathrm{v}'_z(r_s)}{R\mathrm{v}_\theta(r_s)} \tag{A.18}$$

which when put into Eqn. A.17, gives

$$W_{1/2} = 2\sqrt{\frac{A}{B}\frac{V}{\mathrm{v}_\theta}\frac{r_s}{n\left|q'_{IGC}(r_s)\right|}} \tag{A.19}$$

We could stop at this point and have a very good estimation of the width of the ion guiding center resonances, however, in practice, one can make another approximation that puts Eqn. A.19 into a much more useful form.

Under the assumption that the contribution to the guiding centers velocity is small relative to the total velocity, we approximate

$$\frac{\mathrm{v}_\theta}{V} \cong \frac{b_\theta}{B} \tag{A.20}$$

then we arrive at

$$W = 4\sqrt{\frac{Ar_s}{b_\theta n\left|q'_{IGC}(r_s)\right|}} \tag{A.21}$$

this expression for the half width of the ion guiding is convenient because it is in terms of quantities that are well known. The field amplitudes are known quantities, and the radial derivative of the ion guiding center safety factor can be calculated from the expression given in 5.7. The last assumption introduces only a slight error into the estimate, of which a typical value for an r/a = 0.2 fast ion is

$$\sqrt{\frac{\mathrm{v}_\theta}{V}} \cong \sqrt{1+\frac{\mathrm{v}_\theta^{drift}}{V}} \cong \sqrt{1+\frac{0.05}{1}} \cong 1.02\,. \tag{A.22}$$

It is also important to note that the ion motion is acting in accordance with the *magnetic* perturbation but taken at the resonant drift surfaces defined by the *guiding*

center's rotational transform. Intuitively this makes sense as the location of the resonances are determined by the guiding centers motion, and the presence of a perturbation is a general condition. Another thing to note is that a given magnetic mode has a radial structure to it, which for $\widetilde{b}_r$, tends to be peaked near its resonant surface. The corresponding resonant drift surface is in a different location, so the strength of the perturbation is subsequently less. This has the general effect of an ion island being smaller than their magnetic counterparts, though this is not a rigid statement because of other contributions, such as resonant surface location, and *q*-shear.

For both field lines and ion guiding centers, the approximations made in the various quantities fall well within the error in using the sheared slab formulation itself, and have been found to be sufficiently accurate for describing the degree of resonance overlap in either the magnetic or ion guiding center system. This approach has been verified by comparing Poincaré sections of the ion guiding centers to the analytical result, and using the island overlap criterion to predict and explain the onset of fast ion stochasticity.

Appendix B

Derivation of Ion Guiding Center Safety Factor

The derivation of the safety factor in 5.7 is not a straightforward calculation, even when done in a cylindrical geometry. The equation is of central importance in understanding both qualitatively and quantitatively, the results of this thesis; that fast ions have different resonant conditions than the background magnetic field.

The result was first worked out by Y. Tsidulko in a 2002 internal draft report, though presented only in the final form shown in Eqn. 5.7. The full derivation is worked out below, including the expansion of the guiding center safety factor to first order in Larmor radius.

We begin by writing down the equation of the ion guiding center motion

$$\frac{\vec{\mathrm{v}}}{\mathrm{V}} = \left[s_{\|} \sqrt{1-\mu\Omega}\, \vec{b} \right] + \vec{b} \times \left[\frac{\mu}{2} \nabla\Omega + \left(1-\mu\Omega\right)\left(\vec{b}\bullet\nabla\right)\vec{b} \right] \rho \tag{B.1}$$

where,

$$\mathrm{V} = \sqrt{\frac{2E}{m}},\ \ \mu = \frac{E_{\perp}}{E\Omega} = \frac{\mathrm{v}_{\perp}^2}{\mathrm{v}^2}\frac{m}{eB},\ \ \Omega = \frac{eB}{m},\ \ \rho = \frac{\mathrm{V}}{\Omega},\ \ s_{\|} = \frac{\vec{\mathrm{v}}\bullet\vec{b}}{\left|\vec{\mathrm{v}}\bullet\vec{b}\right|} = \pm 1 \tag{B.2}$$

Additional useful expressions

$$\mathrm{V}^2 = \mathrm{v}_{\|}^2 + \mathrm{v}_{\perp}^2,\ \ 1 = \frac{\mathrm{v}_{\|}^2}{\mathrm{V}^2} + \mu\Omega,\ \ \mathrm{v}_{\|} = \sqrt{1-\mu\Omega}\,\mathrm{V} \tag{B.3}$$

The guiding center rotational transform, in a cylindrical geometry with no Shafranfov shift, can be written as:

$$q^{GC}(r) = \frac{r\mathrm{v}_{\phi}^{GC}}{R\mathrm{v}_{\theta}^{GC}} \tag{B.4}$$

We can now take the toroidal and poloidal components of the given velocity expression in Eqn. B.10. We do this term-by-term and find for v_{ϕ}^{GC} :

$$\frac{\mathrm{v}_{\phi}^{GC}}{\mathrm{V}} = s_{\|}\sqrt{1-\mu\Omega}b_{\phi} + \frac{b_{\theta}^{3}\rho}{r}(1-\mu\Omega) - \frac{b_{\theta}\rho\mu\Omega'}{2} \tag{B.5}$$

Similarly for the theta component we find:

$$\frac{\mathrm{v}_{\theta}^{GC}}{\mathrm{V}} = s_{\|}\sqrt{1-\mu\Omega}b_{\theta} - \frac{b_{\theta}^{2}b_{\phi}\rho}{r}(1-\mu\Omega) + \frac{b_{\phi}\rho\mu\Omega'}{2} \tag{B.6}$$

Now we can use these expressions to solve for the guiding center q_{GC}. Using Eqn. B.18 and Eqn. B.19, we do a Taylor expansion in gyro-radius, about the guiding center.

$$q_{GC} = \frac{rB_{\phi}}{RB_{\theta}} + \frac{r}{R}\rho\frac{\partial}{\partial\rho}\left(\frac{\mathrm{v}_{\phi}}{\mathrm{v}_{\theta}}\right) + \ldots \cong \frac{rB_{\phi}}{RB_{\theta}} + \frac{r}{R}\rho\frac{\mathrm{v}_{\theta}\mathrm{v}'_{\phi} - \mathrm{v}_{\phi}\mathrm{v}'_{\theta}}{\mathrm{v}_{\theta}^{2}} \tag{B.7}$$

Where the primes denote first derivatives in ρ. q_0 is the magnetic q which is valid for infinitely small Larmor radii.

For brevity we write $\alpha = \sqrt{1-\mu\Omega}$.

The numerator is

$\mathrm{v}_{\theta}\mathrm{v}'_{\phi} - \mathrm{v}_{\phi}\mathrm{v}'_{\theta} =$

$$\left(\overbrace{s_{\|}\alpha b_{\theta}}^{(1)} - \overbrace{\frac{b_{\theta}^{2}b_{\phi}\rho\alpha^{2}}{r}}^{(2)} + \overbrace{\frac{\rho\mu b_{\phi}\Omega'}{2}}^{(3)}\right)\left(\overbrace{\frac{b_{\theta}^{3}\alpha^{2}}{r}}^{(a)} - \overbrace{\frac{\mu b_{\theta}\Omega'}{2}}^{(b)}\right) - \left(s_{\|}\alpha b_{\phi} + \frac{b_{\theta}^{3}\rho\alpha^{2}}{r} - \frac{\rho\mu b_{\theta}\Omega'}{2}\right)\left(-\frac{b_{\theta}^{2}b_{\phi}\alpha^{2}}{r} + \frac{\mu b_{\phi}\Omega'}{2}\right) \tag{B.8}$$

Multiplying out term by term we find a great deal of cancellation

$$1a = s_{\|}\alpha^3\left(\frac{b_\theta^4 + b_\theta^2 b_\phi^2}{r}\right) \quad 1b = \frac{s_{\|}\alpha\mu\Omega'}{2}\left(-b_\theta^2 - b_\phi^2\right)$$

$$2a = \left(\frac{\rho\alpha^4}{r^2}\right)\left(-b_\theta^5 b_\phi + b_\theta^5 b_\phi\right) = 0 \quad 2b = \left(\frac{\rho\alpha^2\mu\Omega'}{2r}\right)\left(b_\theta^3 b_\phi - b_\theta^3 b_\phi\right) = 0$$

$$3a = \left(\frac{\rho\alpha^2\mu\Omega'}{2r}\right)\left(b_\theta^3 b_\phi - b_\theta^3 b_\phi\right) = 0 \quad 3b = \left(\frac{\mu^2\rho\Omega\Omega'}{4}\right)\left(-b_\phi b_\theta + b_\phi b_\theta\right) = 0$$

Also note that $b_\theta^2 + b_\phi^2 = 1$

We drop terms with ρ, because that will lead to 2nd order terms in ρ when put back into the Taylor expansion. This gives $s_{\|}\left(\frac{\alpha^3 b_\theta^2}{r} - \frac{\alpha\mu\Omega'}{2}\right)$ for the numerator.

For the denominator, we drop all terms with ρ and find $\left(s_{\|}\alpha b_\theta\right)^2$

We have

$$\frac{s_{\|}\left(\frac{\alpha^3 b_\theta^2}{r} - \frac{\alpha\mu\Omega'}{2}\right)}{\left(\alpha b_\theta\right)^2} = \frac{s_{\|}\left(2\alpha^2 b_\theta^2 - r\mu\Omega'\right)}{2rb_\theta^2\alpha} \tag{B.9}$$

which when replaced back into the expansion we finally arrive at

$$q \cong \frac{rB_\phi}{RB_\theta} + \frac{r}{R}\rho\frac{s_{\|}\left(2\alpha^2 b_\theta^2 - r\mu\Omega'\right)}{2rb_\theta^2\alpha} = \frac{rB_\phi}{RB_\theta} + \frac{s_{\|}}{b_\theta^2}\rho\frac{2\left(1-\mu\Omega\right)b_\theta^2 - r\mu\Omega'}{2R\sqrt{1-\mu\Omega}} \tag{B.10}$$

Appendix C

Finite Difference Solution To The Radial Heat Equation

If we wish to account for stochastic heat diffusion during neutral beam injection, we need to solve the radial heat equation. The method of finite differences is a natural way to do this, as our advancement of the energy profiles due to collisions between species (electrons, thermal ions and fast ions) is done explicitly and is discretized by using numerical loops.

First we can write down a differential equation for T

$$\frac{dT(r,t)}{dt} = D\nabla^2 T(r,t) \tag{C.1}$$

where D is the diffusion coefficient (units l^2 / t) and T is the temperature. The radial component of the Laplacian is simply

$$\nabla^2 = \frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r}. \tag{C.2}$$

Therefore, we begin by writing the finite-difference expressions for the two derivative terms. We also state that the notation used in this Appendix is that time indices will be given as n and spatial indices as j.

The first derivative in time can be written as

$$\frac{dT}{dt} = \frac{T_j^{n+1} - T_j^{n-1}}{2\,\Delta t} or \frac{T_j^{n+1} - T_j^{n}}{\Delta t} \tag{C.3}$$

While either is valid for small enough timestep Δt, we will use second form later on.

Similarly, the spatial first derivative term can be written as

$$\frac{dT}{dr} = \frac{T_{j+1}^{n} - T_{j-1}^{n}}{2\,\Delta x} \tag{C.4}$$

where Δx is the discreteness of the spatial grid.

This leaves only one term (the second derivative term in the Laplacian) to be evaluated. To determine the finite difference expression for that, we do a Taylor expansion of *T(r)* about a small spatial displacement.

$$T_{j+1}^n \cong T_j^n + \Delta r \frac{\partial T}{\partial r}\bigg|_j + \frac{(\Delta r)^2}{2}\frac{\partial^2 T_j^n}{\partial r^2}\bigg|_j + \ldots \tag{C.5}$$

Solving for the second derivative term, and inserting the expression for the first derivative (C.2) we have

$$\frac{\partial^2 T_j^n}{\partial r^2} \cong \left(T_{j+1}^n - T_j^n - \Delta r \frac{T_{j+1}^n - T_{j-1}^n}{2\Delta r} \right)\frac{2}{\Delta r^2} = \left(\frac{T_{j+1}^n}{2} - T_j^n + \frac{T_{j-1}^n}{2} \right)\frac{2}{\Delta r^2} = \frac{T_{j+1}^n - 2T_j^n + T_{j-1}^n}{\Delta r^2} \tag{C.6}$$

Putting Eqn. C.4 and Eqn. C6 together we can write the heat equation in both explicit (depends only on the current or previous time or spatial steps) and implicit (depends on future time or spatial steps) form.

$$\frac{T_j^{n+1} - T_j^n}{\Delta t} = D\left(\frac{T_{j+1}^n - 2T_j^n + T_{j-1}^n}{\Delta r^2} + \frac{1}{r_j}\frac{T_{j+1}^n - T_{j-1}^n}{2\Delta r} \right) \text{ (Explicit)} \tag{C.7}$$

$$\frac{T_j^{n+1} - T_j^n}{\Delta t} = D\left(\frac{T_{j+1}^{n+1} - 2T_j^{n+1} + T_{j-1}^{n+1}}{\Delta r^2} + \frac{1}{r_j}\frac{T_{j+1}^{n+1} - T_{j-1}^{n+1}}{2\Delta r} \right) \text{ (Implicit)} \tag{C.8}$$

At this point, we combine Eqn. C.7 and Eqn. C.8 and take the average. This is the Crank-Nicholson method. Doing so, we arrive at

$$\frac{T_j^{n+1} - T_j^n}{\Delta t} = \frac{D}{2}\left(\begin{array}{l} \dfrac{T_{j+1}^{n+1} - 2T_j^{n+1} + T_{j-1}^{n+1}}{\Delta r^2} + \dfrac{T_{j+1}^n - 2T_j^n + T_{j-1}^n}{\Delta r^2} \\ + \dfrac{T_{j+1}^{n+1} - T_{j-1}^{n+1}}{2r_j\Delta r} + \dfrac{T_{j+1}^n - T_{j-1}^n}{2r_j\Delta r} \end{array} \right) \tag{C.9}$$

This gives a system of equations and we want to solve for the future time (n+1) and three spatial coordinates (j-1, j, j+1). We will use matrix methods to solve it, but in order to put into a useful form we do the following.

We need to organize Eqn. C.9 into present and future timesteps. Start by defining $\alpha = \frac{D\Delta t}{\Delta r^2}$ and write

$$\begin{aligned} &T_j^{n+1} - \frac{\alpha}{2}T_{j+1}^{n+1} + \alpha T_j^{n+1} - \frac{\alpha}{2}T_{j-1}^{n+1} - \frac{\alpha}{4}\frac{\Delta r}{r_j}\left(T_{j+1}^{n+1} - T_{j-1}^{n+1}\right) = \\ &T_j^{n} + \frac{\alpha}{2}T_{j+1}^{n} - \alpha T_j^{n} + \frac{\alpha}{2}T_{j-1}^{n} + \frac{\alpha}{4}\frac{\Delta r}{r_j}\left(T_{j+1}^{n} - T_{j-1}^{n}\right) \end{aligned} \tag{C.10}$$

Now we can group Eqn. C.10 according to j-elements

$$\begin{aligned} &\overbrace{\left(-\frac{\alpha}{2} + \frac{\alpha}{4}\frac{\Delta r}{r_j}\right)}^{a}T_{j-1}^{n+1} + \overbrace{(1+\alpha)}^{b}T_j^{n+1} + \overbrace{\left(-\frac{\alpha}{2} - \frac{\alpha}{4}\frac{\Delta r}{r_j}\right)}^{c}T_{j+1}^{n+1} = \\ &\overbrace{\left(\frac{\alpha}{2} - \frac{\alpha}{4}\frac{\Delta r}{r_j}\right)}^{d}T_{j-1}^{n} + \overbrace{(1-\alpha)}^{e}T_j^{n} + \overbrace{\left(\frac{\alpha}{2} + \frac{\alpha}{4}\frac{\Delta r}{r_j}\right)}^{f}T_{j+1}^{n} \end{aligned} \tag{C.11}$$

Notice the coefficients of each term are labeled a, b, c, etc… This shorthand will come in very handy soon. Now finally we can write the matrix that we want to solve.

$$\begin{pmatrix} b & c & & & \\ a & b & c & & \\ & \ddots & \ddots & \ddots & \\ & & a & b & c \\ & & & a & b \end{pmatrix}\begin{pmatrix} T_0^{n+1} \\ T_1^{n+1} \\ \vdots \\ T_{J-1}^{n+1} \\ T_J^{n+1} \end{pmatrix} = \begin{pmatrix} e & f & & & \\ d & e & f & & \\ & \ddots & \ddots & \ddots & \\ & & d & e & f \\ & & & d & e \end{pmatrix}\begin{pmatrix} T_0^{n} \\ T_1^{n} \\ \vdots \\ T_{J-1}^{n} \\ T_J^{n} \end{pmatrix} \tag{C.12}$$

where J represents the last index of the spatial index j.

We can now address the boundary conditions. The first boundary condition is a Dirchelet condition that the T is zero at the wall.

$$T_J^{n+1} = T_J^n = 0 \tag{C.13}$$

It should be noted that even though we specify the temperature to be zero at the wall, in practice the model doesn't evolve this point to avoid "divide by zero" errors. The radial grid runs from r = 0+Δx to a - Δx.

The second boundary condition is the regularity condition that the derivative across the origin is zero

$$\frac{dT_0^n}{dr} = \frac{dT_0^{n+1}}{dr} = 0\,. \tag{C.14}$$

This boundary condition is tough to implement into the matrix equation because the matrix is expressed in terms of the actual values, not their derivatives.

The regularity condition written in the form of finite differences (at $j = 0$) is

$$\frac{\partial T(0,t)}{\partial r} = 0 = \frac{1}{2}\left(\frac{T_1^{n+1} - T_{-1}^{n+1}}{2\Delta r} + \frac{T_1^n - T_{-1}^n}{2\Delta r}\right) \Rightarrow T_{-1}^{n+1} - T_{-1}^n = T_1^{n+1} - T_1^n \tag{C.15}$$

This equation is unsettling because it uses an index of (-1) in order to evaluate the boundary condition. It needs to know what the point on the 'other side' of r = 0 is doing. Since we don't have that point, we have to eliminate those two indices from our matrix equation. Equation C.11 with $j = 0$ is

$$aT_{-1}^{n+1} + bT_0^{n+1} + cT_1^{n+1} = dT_{-1}^n + eT_0^n + fT_1^n \tag{C.16}$$

but we note that $c = -f$ and $a = -d$.

Eliminating d and f, we find

$$a(T_1^{n+1} + T_1^n - T_{-1}^n) + bT_0^{n+1} + cT_1^{n+1} = -aT_{-1}^n + eT_0^n - cT_1^n \tag{C.17}$$

where $T_{-1}^{n+1} = T_{-1}^{n+1} + T_1^n - T_{-1}^n$ from Eqn. C.15.

In Eqn. C.17 we see that aT_{-1}^{n} will cancel out and we can solve for T_0^{n+1} which will allow us to evolve the axis point at the future time in terms of known spatial quantities.

$$T_0^{n+1} = -\left(\frac{c+a}{b}\right)(T_1^{n+1} + T_1^{n}) + \frac{e}{b}T_0^{n} \tag{C.18}$$

Or in terms of our original constant

$$T_0^{n+1} = \left(\frac{\alpha}{1+\alpha}\right)(T_1^{n+1} + T_1^{n}) + \frac{(1-\alpha)}{(1+\alpha)}T_0^{n} \tag{C.19}$$

and regrouping according to future and present times

$$(1+\alpha)T_0^{n+1} - \alpha T_1^{n+1} = (1-\alpha)T_0^{n} + \alpha T_1^{n} \tag{C.20}$$

Now we can specify the first elements of the diagonal and superdiagonal (the diagonal to the right of the main diagonal). This gives the full matrix as

$$\overbrace{\begin{pmatrix} b & -g & & & \\ a & b & c & & \\ & \ddots & \ddots & \ddots & \\ & & a & b & c \\ & & & a & b \end{pmatrix}}^{A} \overbrace{\begin{pmatrix} T_0^{n+1} \\ T_1^{n+1} \\ \vdots \\ T_{J-1}^{n+1} \\ T_J^{n+1} \end{pmatrix}}^{x} = \overbrace{\begin{pmatrix} e & g & & & \\ d & e & f & & \\ & \ddots & \ddots & \ddots & \\ & & d & e & f \\ & & & d & e \end{pmatrix}}^{B} \begin{pmatrix} T_0^{n} \\ T_1^{n} \\ \vdots \\ T_{J-1}^{n} \\ 0 \end{pmatrix} \tag{C.21}$$

where $g = \alpha$.

To solve this matrix one employs the Thomas algorithm which solves a matrix of the form: $A\ \mathbf{x} = B$, if A is tridiagonal. We note that B is a fully determined matrix; it is based on the known temperature profile.

The solution to C.21 at each timestep advances the diffusion of the temperature profile, which is then modified at each radial location according to energy exchange with the background plasma and fast ions.

Additional notes on heating model

1) Because the fast ion density is modeled as an exponential, at mid-radii and beyond the density quickly approaches zero. This is a source of instability and is addressed by setting a cutoff radius at which the fast ions do not interact with the plasma. This radius is typically taken to be where the density has fallen by a factor of 100 from the axis density.

2) The Coulomb drag model used is only valid when the fast ions are much more energetic than the background plasma. For a long simulation they are capable of thermalizing, which results in extremely high collision frequencies and is also a source of numerical instability.

3) The code also calculates the neutron flux from both fast ions fusing with thermal ions, and the thermal ions fusing with each other.

The entire NBI heating code, written in IDL 6.2, is below.

```
;*******************************************************
;*
;*  NBI Heating Model  nbi_heating_model.pro
;*
;*******************************************************
;Written: 6/16/2006     Ben Hudson
;
;Purpose:
;   Description of arbitrary length neutral beam
;injection effects on plasma temperature.
;
;Features:
;   Arbitary radial profiles for plasma density and temperature
;   Beam ions treated as short-pulse beamlets that evolve independently.
;   Radial diffusion of temperature profiles.
;
;Method:
;   *The steady-state source terms to the plasma are found that would
;offset diffusion and equilibration between speices, thus sustaining
;the specified profile in the absence of NBI.
;   *Diffusion of the electron and ion temperature profiles are
;calculated (not applied).
;   *The beamlets are introduced as a source term and energy transfer
;is calculated (not applied) for NBI->electrons and NBI->ions.
;   *Energy transfer is calculated (not applied) for electrons->ions
;   *Energy transfer of all components are applied.
;   *Additional calculates made for evolved profile (neutron flux, etc.)
;   *Advance to next time-step.
;
;Advancement scheme:
;   The diffusion calculations are done with the radial heat equation
;written in finite differences.  The Crank-Nicholson method is used
;for stability of the algorithm.
;   The advancement of the energy transfer equations is by definition
;explicit, so a simple time-step advancement is used for that.
;
;Full documentation available in Appendix C of Hudson thesis, 2006
;*******************************************************************
*

pro nbi_heating_model
```

```
;Graphical interface
set_plot,'ps'
!p.font = 1.0
device,/color
device,xoffset = 0.5,/inches
device,yoffset = 0.5,/inches
device,xsize = 5,/inches
device,ysize = 4,/inches
loadct,39

;**** System and simulation dimensions ****

   a = 0.5  ; radius (m)
   rmajor = 1.5    ;Major radius of MST

   t_max = 10.0e-3  ;Max time (s)

   dt = 1.0e-6       ;Time step (s)  (10-6 optimum)
   dr = 1.0e-3       ;Radial grid resolution  (10-3 optimum)

   nt = t_max/dt    ;Number of time steps in simulation
   nr = a/dr   ;Number of radial positions

   t = findgen(nt)*dt  ;Array for time values
   r = (findgen(nr)+.5)*dr ;Array for radial values

   nobeam = 0.0   ;set to 0 to have a beam
   stepbeam = 0.0   ;set to 0 to have realistic beam shape
   nodiffusion = 0.0 ;set to 0 to enable radial diffusion
   noions = 0.0    ;set to 0 to have ions as well as electrons
   noelectrons = 0.0   ;set to 0 to have electrons
   k_loss = 0.0  ;set to 0 for no loss times, set to 1 for loss times. Either use this or the
diffusion

   plotprofiles = 1
   n_profiles = 10     ;Number of radial profiles to plot. Split evenly among simulation
time.


;****************************************

;*** Physical constants (SI)
ec = 1.6e-19    ;Electron charge
m_p = 1.67e-27  ;Proton mass
m_e = 9.1e-31   ;Electron mass
```

```
;*** Define plasma and beam constants ***

;---- Diffusion coefficients
d_m = 8e-4  ;m
chi_e = make_array(nr,value = 400.0,/double)     ;m^2/s  400m^2/s (Ted)  10m^2/s PPCD
chi_d = make_array(nr,value = 6.0,/double)

;---- Beam parameters ----
z_b = 1.0     ;Beam ion charge
mu_b = 1.0      ;beam ion mass / proton mass   2.0 for deuterium
m_b = mu_b*m_p  ;Mass of beam ion

beam_time = 1.5e-3   ;Beam on duration (s)
beam_current = 20.0     ;Beam current (Amps)

n_beamlets = 20.0   ;Number of small beam pulses that compose the entire injection
period.


alpha = 3.7379  ;D2 400ka
;alpha = 3.59    ;H 200ka
t_b0 = 2e4   ;Beam ion energy

   ;Final beam axis density based on 30 amps of beam current for 1.5ms
n_b0 = 3.168e17*beam_current/30.0*beam_time/1.5e-3  ;20kev
;n_b0 = 3.168e17*beam_current/30.0*beam_time/1.5e-3*.53/.80  ;50kev


n_b = n_b0*exp(-(2.0*r/a*alpha)^2)
num_fi = int_tabulated(r,2.0*!pi*1.5*2*!pi*r*n_b)

; We don't need (and maybe shouldn't have) fast ions in the mid to outer radii so
;set those to zero.  Also this helps prevent the low plasma edge temperatures from
;thermalizing the fast ions and invalidating the model.
res = where(n_b le .01*n_b0)
cutoff = res[0]

if stepbeam eq 1 then n_b = make_array(nr,value = n_b0)


w_b = t_b0*n_b
t_b = w_b/n_b

tau_b = 20e-3   ;Fast ion confinement time in seconds
```

```
;---- Electrons ----
z_e = 1.0      ;Electron z (duh)
mu_e = 1.0       ;Electron mass to electron mass (mega-duh, but these keeps the notation
consistent)
m_e = mu_e*m_e

t_e0 = 400.0       ;Axis electron temperature (eV)
n_e0 = 1.0e19     ;Axis electron density (m^-3)

n_e = n_e0*(1.0-(r/a)^2)    ;Electron density profile
t_e = t_e0*(1.0-(r/a)^2)    ;Electron temperature profile
w_e = 1.5*t_e*n_e      ;Value of Electron energy density

tau_e = .001    ; Electron energy loss time (not used in diffusion model)

;-------------------

;---- Ions (see electrons above for description of parameters) ----
z_d = 1.0
mu_d = 2.0
m_d = mu_d*m_p

t_d0 = 200.0
n_d0 = 1.0e19

n_d = n_d0*(1.0-(r/a)^2)
t_d = t_d0*(1.0-(r/a)^2)

w_d = 1.5*t_d*n_d

tau_d = .001
;-------------------

;- Other constants -
lambda_be = 15.0
lambda_bd = 15.0
lambda_de = 15.0
;-------------------

;*********************************************
;
; Preliminary calculations
;
;*********************************************

;Differential volume elements (each radius is a toroidal shell)
```

```
;Used for neutron flux calculation
d_volume = r*0.0
for i = 0, nr-1 do begin
   if i eq 0 then d_volume[i] = 2.0*!pi^2*rmajor*r[0]^2
   if i gt 0 then d_volume[i] = 2.0*!pi^2*rmajor*(r[i]^2 - r[i-1]^2)
endfor

total_neutron_flux = make_array(nt,value = 0.0,/double)
thermal_neutron_flux = make_array(nt,value = 0.0,/double)


;**** Initial profiles (used to find steady state heat sources) ****


n_b_initial = n_b
w_b_initial = w_b
t_b_initial = w_b/n_b   ;The beam doesn't have a 'Temperature'. This refers to the mean
energy.

w_e_initial = w_e
w_d_initial = w_d

;*********************************


;***** Time arrays for data *****
beam_str = {t_b:dblarr(nr),n_b:dblarr(nr)}
beam = replicate(beam_str,nt+1)

electrons_str = {t_e:dblarr(nr),n_e:dblarr(nr)}
electrons = replicate(electrons_str,nt+1)

deuterium_str = {t_d:dblarr(nr),n_d:dblarr(nr)}
deuterium = replicate(deuterium_str,nt+1)

neutrons_str = {n:dblarr(nr)}
neutrons = replicate(neutrons_str,nt+1)

;*********************************

;*** Structures to hold current beamlet information
beamlet_str = {t_b:t_b,n_b:n_b/n_beamlets,w_b:w_b/n_beamlets}
beamlet = replicate(beamlet_str,n_beamlets)

;*******************************
```

```
;Initial conditions
beam[0].t_b = t_b
beam[0].n_b = n_b
electrons[0].t_e = t_e
electrons[0].n_e = n_e
deuterium[0].t_d = t_d
deuterium[0].n_d = n_d


if plotprofiles eq 1 then begin
   plot,r,t_b,title ='Temperature profiles (eV)',yr = [0,6e2],/nodata
   oplot,r,t_b,color = 0,thick = 2
   oplot,r,t_e,color = 54,thick = 2
   oplot,r,t_d,color = 150,thick = 2
endif

t_e_axis = dblarr(nt)
t_e_axis[0] = w_e[0]/n_e[0]/1.5

; Define or reset diffusion arrays
s_e_diffusion = r*0.0
s_d_diffusion = r*0.0


w_b_axis = make_array(nt)
t_d_axis = make_array(nt)

;Arrays to hold contributions to the energy profile, used to diagnose.
s0 = make_array(nt)
s1 = s0
s2 = s0
s3 = s0
s4 = s0
s5 = s0


;***********************************************
;
;   Beginning of time loop:  index = k
;
;-------------------------------------
;
;   Important note: We start at t=0, but we assume that the first beamlet is alread in
;the plasma.  If we started at t=0 with no beam and evolved the profiles, there would
;be no change anyway because the steady-state terms would prevent it.  This saves one
step,
```

```
;but also allows the use of the beam equations without having to build in an extra check
;just for t=0.
;
;***********************************************
   for k = 0l,nt -1 do begin
   if ((k*dt mod 1e-3) eq 0) then print,string(k*dt*1e3)+'ms'

   if nodiffusion eq 1 then goto,skipdiffusion

   ;  Calculate the advancement of the electron and ion energy profiles due to radial
   ;diffusion.
   ;  Sol_w_X is the advanced solution, which is stored for later use, but also for t=0
   ;is used to find the steady-state source term for this mechanism.
   sol_w_e = diffuse_profile(r,t_e,chi_e,dt,dr)*n_e*1.5
   sol_w_d = diffuse_profile(r,t_d,chi_d,dt,dr)*n_d*1.5


   ;Calculate steady state heat terms to cancel diffusion
   if k eq 0 then begin

     print,'steady state heat found'
     ; If it was allowed to diffuse away, what would it look like after 1 time step?


     oplot,r,sol_w_e/n_e/1.5,color = 54,line = 4  ;electrons
     oplot,r,sol_w_d/n_d/1.5,color = 153,line = 4 ;Deuterium

     ;steady state diffusion source term
      s_e_diffusion = w_e_initial - sol_w_e
      s_d_diffusion = w_d_initial - sol_w_d

   endif

   skipdiffusion:


   ;Initialize and reset arrays for the energy transfer from beamlets to electrons and ions
   d_w_e_beamlets = dblarr(nr)*0.0
   d_w_d_beamlets = dblarr(nr)*0.0

   ;Calculate the number of beamlets currently being evolved.
   current_beamlets = fix(t[k]/beam_time*n_beamlets)+1.0

   ;If past the beam-on time, do not add anymore beamlets
   if current_beamlets ge n_beamlets then current_beamlets = n_beamlets
```

```
   flux = 0.0 ;Initialize the value of the neutron flux for this timestep.

   ;---------------------------------
   ;  Beamlet loop.  index: j
   ;  -----------------
   ;  Each beamlet will have a different energy. This evolves the energy transfer from
each
   ;beamlet to the background plasma.
   ;---------------------------------
   for j = 0,current_beamlets-1 do begin
      d_w_b = dblarr(nr)*0.0 ;Initialize the change in the beamlets energy

   ;Load the information for the current beamlet into convienient variable names.
   w_b = beamlet[j].w_b
   n_b = beamlet[j].n_b
   t_b = beamlet[j].t_b

   ;Calculate velocities
   v_b = sqrt(2.0*t_b*ec/m_b)
   v_e = sqrt(2.0*t_e*ec/m_e)
   v_d = sqrt(2.0*t_d*ec/m_d)

   t_e = w_e/n_e/1.5
   t_d = w_d/n_e/1.5

   ;Beam loss to electrons
   nu_b_e = nu_e(m_b,m_e,v_b,t_e,z_b,z_e,lambda_be,n_e)
   d_w_b_e = -nu_b_e*w_b*dt

   if nobeam eq 1 then d_w_b_e[*] = 0.0
   if noelectrons eq 1 then d_w_b_e[*] = 0.0


   ;Beam loss to deuterium
   nu_b_d = nu_e(m_b,m_d,v_b,t_d,z_b,z_d,lambda_bd,n_d)
   d_w_b_d = -nu_b_d*w_b*dt

   ;Apply radial cutoff for fast ions
   ;Set values outside the cutoff to zero.  This automatically sets the beam energy
   ;loss at these points to zero.
   d_w_b_e[cutoff:*] = 0.0
   d_w_b_d[cutoff:*] = 0.0

   ;We don't want to evolve the beam energy below 15*Te.  This would require a kinetic
   ;description, and also ion collision frequencies blow up as energy gets very small.
   count = 0.0
```

```
res = where(t_b le 15.0*t_e,count)
if count gt 0 then begin
  ;print,'Fast ion approaching thermal, simulation halted'
  d_w_b_e[res] = 0.0
  d_w_b_d[res] = 0.0
  ;stop
endif

;-----------------------------------------------
; Checks for keys that will turn off certain effects of beamlets, electrons or ions.
;-----------------------------------------------
if nobeam eq 1 then d_w_b_d[*] = 0.0
if noions eq 1 then d_w_b_d[*] = 0.0


if nobeam eq 1 then d_w_b_d[*] = 0.0d
if nobeam eq 1 then d_w_b_e[*] = 0.0d


if noions eq 1 then d_w_b_d[*] = 0.0d
if noions eq 1 then d_w_e_d[*] = 0.0d


if noelectrons eq 1 then d_w_b_e[*] = 0.0d
if noelectrons eq 1 then d_w_e_d[*] = 0.0d

;The change in the energy density of the beamlet
d_w_b = d_w_b_e + d_w_b_d

;Advance beamlets energy density.
beamlet[j].w_b += d_w_b


; Checks for the integrity of the run.  If an unphysical result is being
;advanced this will stop the code.
count = 0.0
res = where(beamlet[j].w_b le 0.0,count)
if count gt 0 then stop

res = where(nu_b_e le 0.0,count)
if count gt 0 then stop

res = where(nu_b_d le 0.0,count)
if count gt 0 then stop
```

```
    d_w_e_beamlets += -d_w_b_e ;Energy to electrons from beamlets
    d_w_d_beamlets += -d_w_b_d  ;Energy to ions from beamlets

    ; handle loss of beamlet particles
    beamlet[j].n_b -= beamlet[j].n_b/tau_b*dt
    beamlet[j].w_b -= beamlet[j].w_b/tau_b*dt

    ;Note: this recovers the beam 'energy', since it has no 'temperature'
    beamlet[j].t_b = beamlet[j].w_b/beamlet[j].n_b

    ;Build up the on-axis value for the beams energy density
    w_b_axis[k] += beamlet[j].w_b[0]

    flux += neutron_flux(beamlet[j].n_b,beamlet[j].t_b,m_b,n_d,d_volume)
    endfor ;j loop (beamlet loop)

    total_neutron_flux[k] = flux

    ;Electron and ion energy transfer. The change in the electrons energy density is
transfered to the
    ;ions.

    nu_e_d = 1.8e-19*sqrt(m_e*1e3)*z_e^2*z_d^2*(n_d*1e-
6)*lambda_de*sqrt(m_d*1e3)/((m_e*1e3)*(t_d) + (m_d*1e3)*(t_e))^1.5
    d_w_e_d = 2.0*nu_e_d*(t_d - t_e)*n_e*dt

    ;----------------------------------------
    ; Check for additional keys that eliminate energy transfers.
    ;----------------------------------------
    if noions eq 1 then d_w_e_d[*] = 0.0
    if noelectrons eq 1 then d_w_e_d[*] = 0.0


;Diagnostics
goto,skipshit
    print,'beam terms'
    print,'dwbe',d_w_b_e[0]
    print,'dwbd',d_w_b_d[0]
    ;print,'dweb',d_w_e_b[0]
    ;print,'dwdb',d_w_d_b[0]
    ;print,'beam total',d_w_b_e[0]+d_w_b_d[0]-d_w_e_b[0]-d_w_d_b[0]
    print,''
    print,'electron terms'
    ;print,'dweb',d_w_e_b[0]
    print,'dwed',d_w_e_d[0]
    ;print,'dwbe',d_w_b_e[0]
```

```
    ;print,'dwde',d_w_d_e[0]
    ;print,'beam total',d_w_e_b[0]+d_w_e_d[0]-d_w_b_e[0]-d_w_d_e[0]
    print,''
;    print,ggg
skipshit:


    ; Assigns the steady state heat source for electrons and deuterium
    if k eq 0 then begin
      s_e = -d_w_e_d + 1.0/tau_e*w_e*dt*k_loss
      s_d = d_w_e_d + 1.0/tau_d*w_d*dt*k_loss
    endif


    ;Energy transfer change to electrons
    ;  k_loss enables a constant loss rate instead of diffusion.
    d_w_e = d_w_e_beamlets + d_w_e_d + s_e - 1.0/tau_e*w_e*dt*k_loss
    if noelectrons eq 1 then d_w_e[*] = 0.0

    ; Source terms for diagnostic purposes.
    s0[k] = d_w_e_beamlets[0]/n_e[0]/1.5
    s1[k] = d_w_e_d[0]/n_e[0]/1.5
    s2[k] = s_e[0]/n_e[0]/1.5
    if nodiffusion eq 0 then s3[k] = s_e_diffusion[0]/n_e[0]/1.5
    if nodiffusion eq 0 then s4[k] = (sol_w_e[0]-w_e[0])/n_e[0]/1.5
    s5[k] = 1.0/tau_e*w_e[0]*dt*k_loss/n_e[0]/1.5


    ;Energy transfer change to deuterium
    d_w_d = d_w_d_beamlets - d_w_e_d + s_d - 1.0/tau_d*w_d*dt*k_loss
    if noions eq 1 then d_w_d[*] = 0.0

;************************************************
;
; Apply calculated changes (source + diffusion + transfer) to profiles
;
;************************************************


    ; Adjust electron profile
    if nodiffusion eq 1.0 then w_e = w_e + d_w_e
    if nodiffusion eq 0.0 then w_e = sol_w_e + d_w_e + s_e_diffusion

    ; Adjust deuterium profile
    if nodiffusion eq 1.0 then w_d = w_d + d_w_d
    if nodiffusion eq 0.0 then w_d = sol_w_d + d_w_d + s_d_diffusion
```

```
    ; Calculate temperatures based on new profiles
    t_e = w_e/n_e/1.5
    t_d = w_d/n_d/1.5

    ;Store present profiles in structures.
    if k lt nt-1 then begin
       electrons[k+1].t_e = t_e
       electrons[k+1].n_e = n_e
       deuterium[k+1].t_d = t_d
       deuterium[k+1].n_d = n_d
    endif


    ;Plotting routines
    inds = fix(findgen(n_profiles)/(n_profiles-1.0)*n_elements(t))
    inds[n_profiles-1] = nt-1
      res = where(k eq inds,count)
      ;print,inds,k
      ;print,res,count
      ;print,jlkj

    if ((count gt 0) and (plotprofiles eq 1)) then begin

       oplot,r,t_e,color = 50
       oplot,r,t_d,color = 150

       t_bb = t_b*0.0
       for i = 0,current_beamlets-1 do begin
         t_bb += beamlet[i].t_b/current_beamlets
       endfor

      mycolor = 0  ;Black for beam traces when beam is on, red if beam is off
       if t[k] gt beam_time then mycolor = 254

       oplot,r,t_bb/100.0,color = mycolor
;print,'printed'
    endif

    ;Diagnostics
    goto,skip_diag
    if k eq 0 then plot,r,nu_e_b,yr = [-20,200],/nodata,title = 'collision frequencies (red =
beam,blue = e-,green = d+)
    oplot,r,nu_b_e,color = 254
    oplot,r,nu_b_d,color = 254,line = 3
    oplot,r,nu_e_b,color = 50
```

```
    oplot,r,nu_e_d,color = 50,line = 3
    oplot,r,nu_d_b,color = 150
    oplot,r,nu_d_e,color = 150,line = 3


    ;skip_diag:
    print,'Axis colision frequencies,energy transfer rates'
    print,'nu_b_e',nu_b_e[0],d_w_b_e[0]
    print,'nu_b_d',nu_b_d[0],d_w_b_d[0]
    ;print,'nu_e_b',nu_e_b[0],d_w_e_b[0]
    print,'nu_e_d',nu_e_d[0],d_w_e_d[0]
    ;print,'nu_d_b',nu_d_b[0],d_w_d_b[0]
    ;print,'nu_d_e',nu_d_e[0],d_w_d_e[0]
    print,'**************************'
    print,ggg
    skip_diag:

    ;Save axis values for plotting
    t_e_axis[k] = t_e[0]
    t_d_axis[k] = t_d[0]

    endfor  ;of time loop k

    ;*****************************************
    ;
    ; Output and plotting routines
    ;
    ;*****************************************

    plot,t,w_b_axis,title = 'Axis beam ion energy density'

    print,'Te,Td,Eb (r=0) final=',t_e[0],t_d[0],t_b[0]
    print,'Axis diffusion coefficients (e-,d+):',chi_e[0],chi_d[0]

    t_max = max(t_e_axis)
    ind = where(t_e_axis eq max(t_e_axis))
    inde = ind
    time_max = dt*ind[0]
    mytitle = 'Max Te(r=0) = '+string(t_max)+ '  at t='+string(time_max)
    mytitle = ''
    myr = [min(t_e_axis),(max(t_e_axis)-min(t_e_axis))*1.1 + min(t_e_axis)]
    plot,t*1e3,t_e_axis,title = mytitle,xtitle = 'Time (ms)',ytitle = 'T!Le!N(r=0) eV',yr =
myr,ystyle = 1,/nodata,charsize = 1.3
    oplot,t*1e3,t_e_axis,thick = 2,color = 254

    t_max = max(t_d_axis)
```

```
   ind = where(t_d_axis eq max(t_d_axis))
   indd = ind
   time_max = dt*ind[0]
   mytitle = 'Max Td(r=0) = '+string(max(t_d_axis))+ '  at t='+string(time_max)
   mytitle = ''
   myr = [min(t_d_axis),(max(t_d_axis)-min(t_d_axis))*1.1 + min(t_d_axis)]
   plot,t*1e3,t_d_axis,title = mytitle,xtitle = 'Time (ms)',ytitle = 'T!Ld!N(r=0) eV',yr =
myr,ystyle = 1,/nodata,charsize = 1.3
   oplot,t*1e3,t_d_axis,thick = 2,color = 54


   ;plot,t,t_e_axis,ytitle = 'Axis Temperature (eV)',xtitle = 'Time(ms)',sub = 'Electrons
(red), Ions (blue)',thick = 2, charsize = 1.1,/nodata
   ;oplot,t,t_d_axis,color = 54


   plot,r,electrons[inde].t_e-electrons[0].t_e,title = 'delta t_e (where t_e(r=0) is
max)',xtitle = 'r(m)'
   plot,r,deuterium[indd].t_d-deuterium[0].t_d,title = 'delta t_d (where t_d(r=0) is
max)',xtitle = 'r(m)'

   print,'radial cutoff=',r[cutoff]

   goto,skip_temps
   plot,r,beam[0].t_b, title = 'temps',/nodata,/ylog,yrange = [50,2.5e4],ystyle = 1
   oplot,r,beam[0].t_b,color = 254
   oplot,r,electrons[0].t_e,color = 50
   oplot,r,deuterium[0].t_d,color = 150
   for i = 0,nt-1 do begin
     oplot,r,beam[i].t_b,color = 254
      oplot,r,electrons[i].t_e,color = 50
      oplot,r,deuterium[i].t_d,color = 150
      print,i,beam[i].t_b[0]
   endfor
   skip_temps:

   goto,skip_density
   plot,r,beam[0].n_b, title = 'density',/nodata,/ylog,yr = [1e15,1e20],ystyle = 1
   oplot,r,beam[0].n_b,color = 254
   oplot,r,electrons[0].n_e,color = 50
   oplot,r,deuterium[0].n_d,color = 150
   for i = 0,nt-1 do begin
      oplot,r,beam[i].n_b,color = 254
      oplot,r,electrons[i].n_e,color = 50
      oplot,r,deuterium[i].n_d,color = 150
   endfor
```

```
   skip_density:


   for i = 0l,nt-1 do begin


      sigma_v = 2.33e-14/(deuterium[i].t_d/1e3)^(.66)*exp(-
18.76/(deuterium[i].t_d/1e3)^(.33))*1e-6  ;m^3/s

      dd_rate = 0.5*sigma_v*(deuterium[i].n_d)^2*d_volume
      dd_total = total(dd_rate)
      thermal_neutron_flux[i] = dd_total

   endfor

   plot,t,thermal_neutron_flux,xtitle = 'Time (s)',title = 'Thermal neutron flux per second'

   plot,t,total_neutron_flux,xtitle = 'Time (s)',title = 'Total neutron flux per second'

   plot,t,s0,title = 'dTe from beamlets s0',xtitle = 'Time (s)'
   plot,t,s1,title = 'dTe from ions s1',xtitle = 'Time (s)'
   plot,t,s2,title = 'dTe thermal source s2',xtitle = 'Time (s)'
   if nodiffusion eq 0 then plot,t,s3,title = 'dTe diffusion source s3',xtitle = 'Time (s)'
   if nodiffusion eq 0 then plot,t,s4,title = 'dTefrom diffusion s4',xtitle = 'Time (s)'
   if k_loss eq 1 then plot,t,s5,title = 'dTe from losses s5 (not used if diffusion
present)',xtitle = 'Time (s)'
   plot,t,s0+s1+s2+s3+s4-s5,title = 'dTe sum',xtitle = 'Time (s)'


   myend:
   device,/close
end



;************************************
;
; Function get_x
;
; ---------------------
;
; Purpose:
;Compute the value of 'x' used in NRL p. 32
;
; Inputs:
;m_beta - mass of background particles (grams)
```

```
;v_alpha - velocity of test particle (cm/s)
;t_beta - temperature of background particles (eV)
;*************************************
function get_x,m_beta,v_alpha,t_beta

   return,m_beta*v_alpha^2/(2.0*1.6e-12*t_beta)

end


;*************************************
;
; Function get_psi
;
; --------------------
;
; Purpose:
;Compute the value of 'psi' used in NRL p. 32
;
; Inputs:
;x - return value from function get_x
;*************************************
function get_psi,x

   psi = dblarr(n_elements(x))
   for i = 0,n_elements(x)-1 do begin
      if x[i] ge 10.0 then psi[i] = 1.0 else $
        psi[i] = -2.0/sqrt(!pi)*sqrt(x[i])*exp(-x[i]) + Errorf(sqrt(x[i]))
   endfor

   return,psi
end

;*************************************
;
; Function get_dpsi
;
; --------------------
;
; Purpose:
;Compute the value of dpsi/dx used in NRL p. 32
;
; Inputs:
;x - return value from function get_x
;*************************************
function get_dpsi,x
```

```
   dpsi = dblarr(n_elements(x))
   for i = 0,n_elements(x)-1 do begin
      if x[i] ge 10.0 then dpsi[i] = 0.0 else $
         dpsi[i] = 2.0/sqrt(!pi)*exp(-x[i])*sqrt(x[i])
   endfor

   return,dpsi
end


;*************************************
;
; Function nu_e
;
; ---------------------
;
; Purpose:
;Compute the energy loss frequency for a test particle.
;From NRL p. 32
;
; Inputs:
;m_alpha_in - mass of test particle (kg)
;m_beta_in - mass of field particle (kg)
;v_alpha_in - velocity of test particle (m/s)
;t_beta_in - temperature of field particles (eV)
;z_alpha_in - charge of test particle (in units of e)
;z_beta_in - charge of field particle (in units of e)
;lambda_ab_in - lambda of the field and test particles
;n_beta_in - density of the field particles (m^-3)
;
;Most of the program is done in SI units, but the formulas
;for NRL are in CGS, so it is easier to used converted values
;to plug into the formulas directly.
;
;IDL is 'pass by value' so the values sent to the function remain
;unchanged; they don't need to be converted back to SI.
;*************************************
function
nu_e,m_alpha_in,m_beta_in,v_alpha_in,t_beta_in,z_alpha_in,z_beta_in,lambda_ab_in,n_
beta_in

   ; Convert to CGS
   m_alpha = m_alpha_in*1e3
   m_beta = m_beta_in*1e3
   v_alpha = v_alpha_in*1e2
```

```
    z_alpha = z_alpha_in
    z_beta = z_beta_in
    lambda_ab = lambda_ab_in
    n_beta = n_beta_in/1e6
    m_p_cgs = 1.67e-24
    a_alpha = m_alpha/m_p_cgs

    ec_cgs = 4.8032e-10     ;Electron charge in cgs units

    f = ec_cgs/m_p_cgs

    ;nu0_old = 4.0*!pi*z_alpha^2*z_beta^2*lambda_ab*n_beta/((m_alpha*v_alpha)^2 *
v_alpha)*ec_cgs^4

nu0 = 4.0*!pi*z_alpha^2*z_beta^2*lambda_ab*n_beta/((a_alpha*v_alpha)^2 *
v_alpha)*ec_cgs^2*f^2
;print,a_alpha
;print,nu0_old
;print,nu0
;print,ggg
    t_beta = t_beta_in

    x = get_x(m_beta,v_alpha,t_beta)

    psi = get_psi(x)
    dpsi = get_dpsi(x)

    nue = 2.0*(m_alpha/m_beta*psi - dpsi)*nu0

if finite(nue[0]) eq 0 then begin
print,'malpha',m_alpha
print,'mbeta',m_beta
print,'valpha',v_alpha[0]
;print,'vbeta',v_beta[0]
print,'zalpha',z_alpha
print,'zbeta',z_beta
print,'lambda',lambda_ab[0]
print,'nbeta',n_beta[0]
print,'tbeta',t_beta[0]
print,'nu0',nu0[0]
print,'x',x[0]
print,'psi',psi[0]
print,'dpsi',dpsi[0]
print,'nue',nue[0]

print,ggg
```

```
endif

   return,nue
end

;*************************************
;
; Function neutron_flux
;
; --------------------
;
; Purpose:
;Compute the total neutron flux from D-D reactions of
;fast ions with the background plasma from the vessel.
;
; Inputs:
;n_fast - fast ion density (m^-3)
;e_fast - energy of fast ion (eV)
;m_fast - fast ion mass (kg)
;n_plasma_ions - density of plasma ions in d_vol
;d_vol - the volume sent to the function (m^-3), typically a differential shell
;
;*************************************
function neutron_flux,n_fast,e_fast,m_fast,n_plasma_ions,d_vol

   ; Constants
   a1 = 47.88
   a2 = 482.0
   a3 = 3.08e-4
   a4 = 1.177
   a5 = 0.0

   ;Electron charge (just so it doesn't have to be passed)
   ec = 1.67e-19
   energy = e_fast

   velocity = sqrt(2.0*energy*ec/m_fast)


   ;Cross section in m^2
   sigma = (a5 + ((a4 - a3*(energy/1e3))^2 + 1.0)^(-1.0) *
a2)/((energy/1e3)*(exp(a1*(energy/1e3)^(-0.5)) - 1.0)) * 1e-28

   dd_rate = 0.5*sigma*velocity*n_fast*n_plasma_ions*d_vol
   dd_total = total(dd_rate)
```

```
   return,dd_total
end


;*******************
;diffuse_profile
;
;Purpose: given a 1-D profile this uses the Crank-Nickolsen method
;to advance the solution to the radial diffusion equation:
;du/dt = D(1/r*du/dr + d2u/dr2)
;
;The heat equation can be written in finite elements as:
;  Ax = b
;
;where the LHS is the time-step n+1 and the RHS is time-step n (or the current step).
;A is tridiagonal and consists of points in space around point i (i-1,i,i+1).
;
;Special care is required for the origin where the BC is that du/dr = 0
;
;Refer to hudson thesis for full documentation and derivation of the finite
;element representation.
;
;inputs:
;   r     -- radial array variable
;   u     -- profile
;   d     -- diffusion coefficient
;   dt    -- time step
;   dr    -- radial step
;*************************
function diffuse_profile,r,u,d,dt,dr

nr = n_elements(r)
rhs = make_array(nr)    ;Right hand side of matrix equation


alpha = d*dt/dr^2


sub = make_array(nr)
main = make_array(nr)
super = make_array(nr)

for j = 0,nr-1 do begin
   sub[j] = (-alpha[j]/2.0 + alpha[j]/4.0*dr/r[j])

   main[j] = (1.0 + alpha[j])
```

```
   super[j] = (-alpha[j]/2.0 - alpha[j]/4.0*dr/r[j])

endfor


;Advance the solution one time-step
   for i = 1, nr - 2 do begin

     rhs[i] = (alpha[i]/2.0 - alpha[i]*dr/4.0/r[i])*u[i-1] + $
        (1.0 - alpha[i])* u[i] + $
        (alpha[i]/2.0 + alpha[i]*dr/4.0/r[i])*u[i+1]  ;original
   endfor

   ;Boundary condition that the profiles are zero at the final radius point
   ;not strictly correct.  Needs to be made generalized.
   rhs[nr-1] = 0.0


   ;The boundary condition that the derivative of the profile is zero
   ;at r=0 can be written as:
   rhs[0] = (1.0 - alpha[0])*u[0] + alpha[0]*u[1]    ;Origin point

   ;Coefficients for the initial conditions
   super[0] = -alpha[0]
   main[0] = (1.0 + alpha[0])

   u = trisol(sub,main,super,rhs)


return,u

end
```

Appendix D A Users Guide To MAL / RIO

The FORTRAN code MAL (MAgnetic Lines) and RIO (Random Ion Orbits) was written in 2002 by Yuri Tsidulko at the Budker Institute of Nuclear Physics. It's purpose is to be able to track particles in the stochastic magnetic field of MST in order to determine the magnetic confinement properties of neutral beam injected ions. The code makes use of the output of the DEBS MHD code, which provides the magnetic perturbation eigenmodes in a periodic cylinder. We will go through the steps done in setting up MAL for use in tracing the magnetic field lines and then expand into RIO which one can consider an enhancement of the basic MAL code.

The DEBS modes are given in a 3-D k-space array, which is then FFT'd to become an array in periodic cylindrical coordinates (r, θ , z). Once this is done, in order to have an analytical expression for the equations of the magnetic field lines, a polynomial fit is done (16 term Chebyshev fit for the perturbation modes). The equilibrium field is also fit the same way (typically 9 terms).

The code that does the expansion and fitting is called 'fex.exe'. The file 'fexin.fex' is a text file that specifies the fitting parameters (terms in Chebyshev polynomial, how many perturbation modes to include, etc.) There is also a file 'config.fex' that is used to specify filenames for the DEBS data, typically 'bzmean.dat', 'btmean.dat', 'brturb.dat', 'btturb.dat', 'bzturb.dat', as well as the output filenames for the fex code itself. When 'fex.exe' is run, it generates a file; typically named 'fex64.fex' (fex = Fourier EXpansion) that contains the polynomial fits to the DEBS data.

The next step is to go to the MAL code. Here we begin by editing 'config.mal' which is another text file that specifies which '.fex' file to use, this would be the

'fex64.fex' file. It also specifies filenames for the output of the MAL code. MAL can generate the following files:

mal_##.rep (A report file that contains final values of the field line tracing)

mal_##_**.res (An output file with the actual data from the field line tracing)

The ## specifies a run number, typically 01, 02, etc.. and the ** specifies the mapping of the field line data desired. These will be explained in more detail in the following section. The code below is an actual config file, which is identical for both MAL and RIO. The code itself is in bold, with comments in normal text.

```
$$$$$$$$$$$$$$$$$$ Description of CONFIG.***  $$$$$$$$$$$$$$$$$$$$$$$$
       Program ***.   Version 4.02
  BACKSPASE for addit. write: 1            Position of rewrite index  !
 Value of the rewrite index is important for output files only.
 Rewrite index = 0 means open output file as "NEW",
   in VMS it results increasing file version number.
 Rewrite index = 1 means open output file as "UNKNOWN",
   in VMS it results overwriting the existent file.

Field_modes:  mean and perturbation modes expansion:  field64.fex     0

Map##_in:  input parameters for prog. RIO     :rio##in.rio            0
Map##_rep: report                             :rio##.rep              1
Map##_***: output results for sections        :rio##***.res           1
 ## means automatically generated number from 01 to 99
*** means automatically suffix: Tim, The, Rad, Phi, ThP, RaP, PhP.
 They can be indicated explicitly, for example:
Map01_in:  input parameters for prog. RIO    :rioAAin.rio             0
Map01_rep: report                            :rioAA.rep               1
Map01_The: output results for theta sections :rio_The.res             1
Map01_Phi: output results for phi sections   :rio_Phi.res             1
```

The main file to edit is the input file, ‘mal_in.mal’. The best way to explain how to make use of the code is to go through the input file line by line and notate what each line does. So as to be as clear as possible, the text of the actual file will be in bold.

```
      Programm MAL.   Version 4.02
This is first position for value of parameter =>!

Nfil: number of input files              = 1.000000000000000D+000
   Number of input files to use in run.
If multiple runs are desired, an input file must be made for each.
This parameter must be set to the number of input files in each input
file.

K_def_rep: key for report default parameters   = 1.000000000000000D+000
   If set to 0, some parameters will not be shown in the .rep file.
Excludes some values from the final report.

****************** Geometry and magnetic field parameters *****
Ktor: geometry key                             = 0.000000000000000D+000
   KTOR=0 means cylindrical, KTOR=1 means toroidal
K_geom:  key for use "fex" geometry parameters = 0.000000000000000D+000
   0 = Use Ktor, Rad_a, Rad_R, Shaf_shift from this file
   1 = use default parameters

eps:  perturbation factor(epsilon)             = 0.000000000000000D-001
   Coefficient which multiplies the radial field.  Can be replaced with
   gauge values below
```

This factor multiplies the DEBS eigenmodes by a specific number. If the modes were pre-scaled to the desired absolute values, it could be used as an overall amplification factor.

```
Rad_a: minor radius a [cm]                     = 5.200000000000000D+001
Rad_R: major radius R [cm]                     = 1.500000000000000D+002
Shaf_shift: Shafranov shift [cm]               = 0.000000000000000D+000
```

The Shafranov shift assumes a shifted circle model. The Grad-Shafranov equation is not actually solved as there is no plasma pressure in this code.

```
Time_s: time when field is calculated [s]      = 1.000000000000000D-003
```

This parameter doesn’t do anything. It is here to specify a start time in the event that the code eventually includes time-varying fields.

```
================ Gauge magnetic field ====================
K_dim_ga: key what component should be gauged  = 1.000000000000000D+000
      K_dim_ga < 1 means no renormalization
        K_dim_ga = 1,2,3 means B_the,rad,phi
        K_dim_ga > 3 means |B|
The gauge key specifies what component of the magnetic field to use to
specify the actual value of the field magnitude.  We typically scale it
to the poloidal magnetic field as that defines a plasma current.
Gauge_Tim: time of gauge                       = 0.000000000000000D+000
Gauge_The: theta/pi position of gauge point    = 0.000000000000000D+000
```

```
Gauge_Rad: radial position/a of gauge point    = 1.000000000000000D+000
Gauge_Phi: phi/pi position of gauge point      = 0.000000000000000D+000
```
These specify the location of the gauge point. We typically use the wall value.

```
B_The_ga_mean: mean B_the [G] at gauge point   = 1.503000000000000D+003
```
The value in Gauss of the field at the gauge point. It scales both the poloidal and toroidal fields accordingly.

```
;K_All_pert_ga: Component to gauge pert. fields= 3.000000000000000D+000
```
This key can be used to scale the perturbation fields to a given component. We don't use this because we scale the m = 0 and m = 1 modes to different components.

```
**************** Parameters for map sections definition ***************
```
Here we define where to record the field line position. It can be done at a given poloidal angle (this is what we use for a typical puncture plot, to get an r-z) or less commonly a toroidal angle or radial value.

A length mapping records the position at a given length along a field lines trajectory. This can be useful for seeing how a field line diffuses.

```
Len_sect:  step for length/a map               = 5.000000000000000D+006
```
Length at which the coordinate of the field line is recorded. This is ALWAYS calculated so if it isn't desired, set it to a large value.

```
Max_Len_step: max intermediate length/a step   = 1.000000000000000D-000
```
A value used for precision calculations. Only important if B is nearly a straight line.

```
K_Len_sect: key for map usage                  = 0.000000000000000D+000
```
K_Len_sect=0 means don't save the map
K_Len_sect=1 means use the map, don't demand precise hitting at grid coordinates
K_Len_sect=2 means use the map, demand precise hitting at grid coordinates

```
The_sect_b: begin theta/pi (poloidal angle) for= 0.000000000000000D-000
    map on toroidal plane.
The_sect_e: end theta/pi position for section  = 2.000000000000000D-000
N_The_sect: # of subdivisions between _b & _e  = 1.000000000000000D+000
K_The_sect: key for map usage                  = 2.000000000000000D+000

Rad_sect_b: begin r/a for radial sectioning    = 0.000000000000000D-000
Rad_sect_e: end r/a for radial sectioning      = 1.000000000000000D-000
N_Rad_sect: # of subdivisions between _b & _e  = 1.000000000000000D+000
K_Rad_sect: key for section edge hitting       = 0.000000000000000D+000

Phi_sect_b: begin phi/pi (toroidal angle) for  = 0.000000000000000D-000
          map on poloidal plane
Phi_sect_e: end phi/pi for phi map             = 2.000000000000000D-000
N_Phi_sect: # of subdivisions between _b & _e  = 1.000000000000000D+000
K_Phi_sect: key for map usage                  = 0.000000000000000D+000

**** Edge usages create stopping conditions for each line ***********
```

As the comments say, these create stopping conditions for each line, however more often we simply don't use these, and set a number of data points to record instead. This is to avoid getting extremely large files and having some lines be sparsely represented.

```
K_Len_edg: key for length edges usage          = 0.000000000000000D+000
   0 = don't use edge limits, 1 = do use edge limits
Len_min: min length edge                       = 0.000000000000000D+000
Len_max: max length edge                       = 6.190000000000000D+005

K_The_edg: key for theta edges usage           = 0.000000000000000D+000
The_min: min theta/pi edge                     = 0.000000000000000D+000
The_max: max theta/pi edge                     = 1.000000000000000D+010

K_Rad_edg: key for radial edges usage          = 0.000000000000000D+000
Rad_min: min radial edge                       = 0.000000000000000D+000
Rad_max: max radial edge                       = 6.000000000000000D-001

K_Phi_edg: key for phi edges usage             = 0.000000000000000D+000
Phi_min: min phi/pi edge                       = 0.000000000000000D+000
Phi_max: max phi/pi edge                       = 1.000000000000000D+010

****************** Parameters for start points definition *****
```

For multiple trajectories (see Ntraj below) we need to define the starting points for each. The values are evenly divided between starting and ending points, inclusive. Ex. 1 traj = start point, 2 trajs = start point, end point, 3 trajs = start, mid, end, and so on…

```
The_f: theta/pi (poloid.angle) for first traj. = 0.000000000000000D-000
The_l: theta/pi    for last trajectory         = 2.000000000000000D+000

Rad_f: r/a for first trajectory                = 1.000000000000000D-002
Rad_l: r/a for last trajectory                 = 9.900000000000000D-001

Phi_f: phi/pi (toroid.angle) for first traj.   = 0.000000000000000D-000
Phi_l: phi/pi     for last trajectory          = 2.000000000000000D-000

****************** Number of trajectories and points  *****
Ntraj: number of trajectories                  = 1.000000000000000D+002
```

These specify the number of points on a giving mapping. This is what is typically used to get an equal representation of each field line.

```
Npo_Len: max number of points on length map    = 0.000000000000000D+008
```
creates *len.res, map at values of l

```
Npo_The: max number of points on toroid.map    = 4.000000000000000D+003
```
creates *the.res, map on toroidal plane

```
Npo_Rad: max number of points on radial map    = 0.000000000000000D+001
```
creates *rad.res, map on r=constant surface

```
Npo_Phi: max number of points on poloid.map    = 0.000000000000000D+002
```
creates *phi.res, map on poloidal plane

```
K_sum: K_sum=1 means save sum map              = 0.000000000000000D+000
```
creates *sum.res, has all mapping coordinates in it

```
****************** Parameters for line diffusion calculation *****
```

MAL calculates the magnetic field line diffusion coefficient. It can be done vs. radius or normalized poloidal flux, but the report file shows the result vs. both area measures so it is not important which one is chosen. It should be noted that D is given as $(\Delta r)^2/(\Delta l)$. We typically consider it to be $D = (\Delta r)^2/(2\Delta l)$ so we divide the code result by 2.

```
N_rad_diff: number of gradations in radial      = 5.000000000000000D+001
```
section for diff.coef. calculation. Shows D in different radial sections

```
T_diff_min: min delta length/a                  = 1.000000000000000D+000
```
If D is calculated with r, not flux, then $D(N) = (\Delta r)^2/(a * \text{T\_diff\_min} * 2^N)$

N is given in .rep files where D is reported.

```
K_rA_diff: key of radial measure for diffusion = 1.000000000000000D+000
```
KRADIF=0 means measure is \t rad

KRADIF=1 means measure is equilibrium A_ph (poloidal flux)

```
****************** Result parameters ****************
```
The following data is saved in the .res files for each mapping. If fewer parameters are desired they can be excluded here. The calculation of the Lyapanov exponent can also be disabled, though in practice calculating it doesn't greatly slow the execution time.

```
Ncolumn: number of columns in result files      = 6.00000000000000D+000
```
colunmns:
1 - point number
2 - x/a line position
3 - y/a line position
4 - z/a line position
5 - l/a line length
6 - sigma *a Lyapunov exponent

```
K_Lyap: 1 means calc. Lyapunov exp.             = 1.000000000000000D+000

****************** Debuging and scheme parameters ***************
K_out: 1 means output numer.results on screen   = 0.000000000000000D+000

RelPre: relative precision for equ.solv.        = 1.000000000000000D-008
AbsPre: absolute precision for equ.solv.        = 2.000000000000000D-012
```
The precision has been benchmarked on speed vs. accuracy and these seem to be optimal settings.

```
tr_min: minimal radius value                    = 4.000000000000000D-010
```
Prevents calculations at r = 0

```
tau_min: minimal value of tau                   = 1.000000000000000D-001
Kj_r: radial mean current key                   = 1.000000000000000D+000
```
KJR=0 means no radial current, KJR=1 means non-zero j_r

The magnetic field should be divergence free, however, numerical error makes this impossible. If this key is set to 0, it forces the field to have $J = 0$ at the wall, and consequently everywhere else to have a small divergence. If the wall current is allowed

to vary, the field can be made divergence free everywhere except at the wall. This is the preferred state.

```
k_A_B_R: key to choose field                    = 0.000000000000000D+000
```

KABR=0 means calculate magnetic field lines

KABR<0 means calculate vector potential lines

KABR>0 means calculate current density lines

```
--------------------- Gauge Amplitudes by mode ------------------------
               ---------------- mode #1 --------------------
N_The_gaug_01: theta number of the mode         = 0.000000000000000D+000
N_Phi_gaug_01: phi number of the mode           = 1.000000000000000D+000
```

Each perturbation mode (only one shown here) is specified by its *m* (theta) and *n* (phi) mode number.

```
B_The_gaug_01: B_the [G] of mode at ga.point    = 7.937000000000000D+000
```

The value of the equilibrium field was specified above, so scaling here with the B_The_gaug_01 key sets the absolute perturbation strength. The specification of the component is misleading. The actual component is specified below, this is just the amplitude of that component.

```
K_ampli_ga_01: key for gauge amplitude          = 1.000000000000000D+000
```

K_ampli_ga_##<0 means gauge mode amplitude in the gauge point

K_ampli_ga_##>0 means gauge max mode amplitude on the surface

K_ampli_ga_##=0 means scale it with epsilon instead

|K_..._ga_##|=1 means gauge if the mode is in *.fex file

|K_..._ga_##|=2 means gauge, stop code if mode is not in *.fex

|K_..._ga_##|=3 means multiply gauged amplitude by epsilon

```
K__pert_ga_01: key what comp. should be gauged = 3.000000000000000D+000
```

The components are specified the same as the equilibrium field gauge, here it is set to be scaled to phi.

The two output files ‘mal_##.rep’ and ‘mal_##_***.res’ are rather self explanatory, the ‘rep’ (report) file contains summaries of the input values as well as the data for the final values of the field line (length, angle traversed, final coordinates, etc..) and the diffusion coefficient profiles. The ‘res’ (results) file has the data from the mapping in columns specified by *Ncolumns* (point, *x, y, z*, etc…).

Now we move onto the RIO code. The file ‘config.rio’ is the same as ‘config.mal’; it specifies the perturbation file for the magnetic fields (usually ‘fex64.fex’) and the desired output filenames.

The input file ‘rio##in.rio’ is mostly the same as the MAL input file, but with the following additions. Again, we will write a whole input file in and add comments to the bold text.

```
       Program RIO.   Version 4.02

First position for the value of the parameter=> !

Nfil: number of input files                     = 1.000000000000000D+000

K_def_rep: key for report default parameters    = 1.000000000000000D+000
```
If set to 0, the omitted parameters will not be shown in the *.rep file.

```
****************** Geometry and magnetic field parameters *****
K_geom: key for take geometry param.from file   = 1.000000000000000D+000
```
1 = use parameters Ktor, Rad_r and Shaf_shift from *.fex file

0 = use these parameters from this file

```
Ktor: geometry key                              = 1.000000000000000D+000
```
KTOR=0 means cylindrical, KTOR=1 means toroidal

```
eps:  perturbation factor  epsilon              = 3.000000000000000D-001
Rad_a: minor radius a [cm]                      = 5.200000000000000D+001
Rad_R: major radius [cm]                        = 1.500000000000000D+002
Shaf_shift: Shafranfov shift [cm]               = 5.000000000000000D+000

================ Gauge magnetic field ====================
K_dim_ga: key what component should be gauged   = 1.000000000000000D+000
```
K_dim_ga < 1 means no renormalization

K_dim_ga = 1,2,3 means B_the,rad,phi

K_dim_ga > 3 means |B|

```
Gauge_Tim: time of gauge point                  = 0.000000000000000D+000
Gauge_The: theta/pi position of gauge point     = 0.000000000000000D+000
Gauge_Rad: radial position/a of gauge point     = 1.000000000000000D+000
Gauge_Phi: phi/pi position of gauge point       = 0.000000000000000D+000

B__ga_mean: mean B__ [G] in the gauge point     = 1.500000000000000D+003
```
;B_The_ga_mean: (synonym to "B__ga_mean")

;B_theta_wall: (synonym to "B__ga_mean")

```
------------- gauge perturbations ---------------------
K_All_pert_ga: key what comp. should be gauged = 1.000000000000000D+000
```
for all perturbations except those, which have K__pert_ga_## indicated.

K_All_pert_ga < 1 means no renormalization

K_All_pert_ga = 1,2,3 means B_the,rad,phi

K_All_pert_ga > 3 means |B|

If "K_All_pert_ga" is omitted then "K_dim_ga" works for perturbations.

```
****************** Parameters for sections definition *****
Tim_sect:  step for time map [s]                = 1.000000000000000D-000
```
Time at which the coordinate of ion is recorded in the corresponding

*.res file. This is ALWAYS calculated so if it isn't desired,

set it to a large value.

```
Max_Tim_step: max internal time step * Om_B0    = 2.000000000000000D+000
```
A value used for precision calculations. Only important if the motion nearly a straight line.

```
K_Tim_sect: key for map usage                   = 2.000000000000000D+000
```
K_Tim_sect=0 means don't use the map

K_Tim_sect=1 means use the map, don't demand precise hitting

K_Tim_sect=2 means demand precise hitting

These keys are the same for all maps

```
The_sect_b: begin angle/pi (pol.) for theta map= 0.000000000000000D-000

The_sect_e: end theta/pi pos. for sectioning    = 2.000000000000000D-000
N_The_sect: # of subdivisions between _b & _e   = 2.000000000000000D+000
K_The_sect: key for map usage                   = 2.000000000000000D+000

Rad_sect_b:  begin r/a for radial sectioning    = 0.000000000000000D-000
Rad_sect_e:  end r/a for radial sectioning      = 8.000000000000000D-001
N_Rad_sect: # of subdivisions between _b & _e   = 1.000000000000000D+000
K_Rad_sect: key for section edge hitting        = 2.000000000000000D+000

Phi_sect_b: begin angle/pi (tor.) for phi map   = 0.000000000000000D-000
Phi_sect_e: end phi/pi for phi map              = 2.000000000000000D-000
N_Phi_sect: # of subdivisions between _b & _e   = 2.000000000000000D+000
K_Phi_sect: key for map usage                   = 2.000000000000000D+000
```

The momentum maps below are the same as any other map, they specify start and end values for recording the coordinates at a given ion momentum.

```
ThP_sect_b: begin for theta momentum sectioning= 0.000000000000000D-000
```

```
ThP_sect_e: end   for theta momentum sectioning= 1.000000000000000D-000
N_ThP_sect: # of subdivisions between _b & _e  = 0.000000000000000D+006
K_ThP_sect: key for map usage                  = 0.000000000000000D+000

RaP_sect_b: begin for radial momentum section  = 0.000000000000000D-000
RaP_sect_e: end for radial momentum section    = 1.000000000000000D-000
N_RaP_sect: # of subdivisions between _b & _e  = 0.000000000000000D+006
K_RaP_sect: key for map usage                  = 0.000000000000000D+000

PhP_sect_b: begin for phi momentum             = 0.000000000000000D-000
PhP_sect_e: end for phi momentum               = 1.000000000000000D-000
N_PhP_sect: # of subdivisions between _b & _e  = 0.000000000000000D+006
K_PhP_sect: key for map usage                  = 0.000000000000000D+000

K_Tim_edg: key for time edges usage            = 2.000000000000000D+000
```

Tim_edg=0 no use min,max

Tim_edg=1 use min

Tim_edg=2 use max

Tim_edg=3 use min and max

```
Tim_min: min time [s] edge                     = 0.000000000000000D+000
Tim_max: max time [s] edge                     = 2.000000000000000D-005

K_The_edg: key for theta edges usage           = 0.000000000000000D+000
The_min: min theta/pi edge                     = 0.000000000000000D+000
The_max: max theta/pi edge                     = 1.000000000000000D+010
K_The_f_edg: key for theta edges first usage   = 0.000000000000000D+000
The_f_min: min theta/pi edge before first entry= 0.000000000000000D+000
The_f_max: max theta/pi edge before first entry= 1.000000000000000D+010
The_f_min and The_f_max have to be in the {The_min,The_max} range

K_Rad_edg: key for radial edges usage          = 2.000000000000000D+000
Rad_min: min radial edge/a                     = 0.000000000000000D+000
Rad_max: max radial edge/a                     = 1.000000000000000D-000
K_Rad_f_edg: key for radial edges first usage  = 2.000000000000000D+000
Rad_f_min: min radial edge/a before first entry= 0.000000000000000D+000
Rad_f_max: max radial edge/a before first entry= 1.000000000000000D+000

K_Phi_edg: key for phi edges usage             = 0.000000000000000D+000
Phi_min: min phi/pi edge                       = 0.000000000000000D+000
Phi_max: max phi/pi edge                       = 1.000000000000000D+010
K_Phi_f_edg: key for phi edges first usage     = 0.000000000000000D+000
Phi_f_min: min phi/pi edge before first entry  = 0.000000000000000D+000
Phi_f_max: max phi/pi edge before first entry  = 1.000000000000000D+010

K_ThP_edg: key for theta momentum edges usage  = 0.000000000000000D+000
ThP_min: min theta momentum edge               = 0.000000000000000D+000
ThP_max: max theta momentum edge               = 1.000000000000000D+010
K_ThP_f_edg: key for theta mom.edges 1st use   = 0.000000000000000D+000
ThP_f_min: min theta mom.edge before 1st entry = 0.000000000000000D+000
ThP_f_max: max theta mom.edge before 1st entry = 1.000000000000000D+010

K_RaP_edg: key for radial momentum edges usage = 0.000000000000000D+000
RaP_min: min radial momentum edge              = 0.000000000000000D+000
RaP_max: max radial momentum edge              = 1.000000000000000D+010
```

```
K_RaP_f_edg: key for rad. mom.edges 1st usage  = 0.000000000000000D+000
RaP_f_min: min rad. mom.edge before 1st entry  = 0.000000000000000D+000
RaP_f_max: max rad. mom.edge before 1st entry  = 1.000000000000000D+010

K_PhP_edg: key for phi momentum edges usage    = 0.000000000000000D+000
PhP_min: min phi momentum edge                 = 0.000000000000000D+000
PhP_max: max phi momentum edge                 = 1.000000000000000D+010
K_PhP_f_edg: key for phi mom.edges 1st usage   = 0.000000000000000D+000
PhP_f_min: min phi mom.edge before 1st entry   = 0.000000000000000D+000
PhP_f_max: max phi mom.edge before 1st entry   = 1.000000000000000D+010

*****************************************************
K_rio/ion: key for rio/ion regime switching    = 1.000000000000000D+000
```

0 = Uses rio settings -- Beam injection

1 = Uses ion settings -- Single particle tracing

```
****************** Parameters for start points position (RIO/ION) *****
****************** Injector geometry parameters ************
```

The code can simulate a beam port and inject a stream of particles according to a specified distribution. See comments after the file description for discussion on this.

```
W_The_Pl: inject. plate width[cm] in theta dir.= 3.000000000000000D+000
W_Phi_Pl: inject. plate width [cm] in phi dir. = 3.000000000000000D+000
K_edge_Pl: injection plate edge key            = 0.000000000000000D+000
```

K_edge_Pl=0 means Gaussian distribution

K_edge_Pl=1 means homogeneous distribution with sharp edge

```
K_form_Pl: injection plate form key            = 0.000000000000000D+000
```

K_form_Pl=0 means elliptical form

K_form_Pl=1 means rectangular form

Only used if K_edge_Pl = 1

```
Rot_Pl: inject. plate rotation angle/pi        = 0.000000000000000D+000
Pit_Pl: inject. plate pitch angle/pi           = 1.000000000000000D-003
Azi_Pl: inject. plate azimuthal angle/pi       = 0.000000000000000D-000

The_Pl: inject. plate center theta/pi position = 0.000000000000000D-000
Phi_Pl: inject. plate center phi/pi position   = 0.000000000000000D-000
Rad_Pl: inject. plate center radial/a position = 1.000000000000000D-001

D_The_V: beam veloc. angle/pi width(theta dir.)= 5.300000000000000D-002
D_Phi_V: beam veloc. angle/pi width(phi dir.)  = 5.300000000000000D-002
Rot_V: beam rotation angle/pi                  = 0.000000000000000D+000
Pit_V: beam pitch angle/pi                     = 1.000000000000000D-003
Azi_V: beam azimuthal angle/pi                 = 0.000000000000000D-000

****************** Fast ion parameters  *****
E_ev: ion mean energy [ev]                     = 1.200000000000000D+004
```

For single particle mode all ions are at this energy. For beam mode this is the mean energy

```
DE_ev: ion energy dispersion [ev]              = 5.000000000000000D+000
```

Dispersion only used in beam mode

```
R_E_fin: ratio of final en. to mean/start ene. = 9.700000000000000D-001
```

An important limit to stop the simulation at a certain energy. We typically stop a trajectory at 15*Te

```
A___F: atom weight number of Fast ion          = 1.000000000000000D+000
Znu_F: nuclear charge number of Fast ion       = 1.000000000000000D+000
Z___F: start charge number of Fast ion         = 1.000000000000000D+000
Exc_F: start excitation index of Fast ion      = 0.000000000000000D+000
K_trans: key allows ion transformation         = 1.000000000000000D+000
```

Charge exchange

K_trans=0 means no transformation is allowed

K_trans=1 means neutral => ion transformation is allowed

K_trans=2 means neutral <= ion transformation is allowed

K_trans=3 means neutral <=> ion transformations are allowed

```
K_drag: key allows ion drag                    = 1.000000000000000D+000

****************** Parameters for ION regime *****
Tim_f:  start time for first.traj. [s]         = 0.000000000000000D-000
Tim_l:  start time for last trajectory [s]     = 0.000000000000000D-000

The_f:  theta/pi (poloid.angle) for first.t.   = 0.000000000000000D-000
The_l:   theta/pi    for last trajectory       = 0.000000000000000D-000

Rad_f: r/a for first trajectory                = 3.000000000000000D-001
Rad_l: r/a for last trajectory                 = 3.000000000000000D-001

Phi_f: phi/pi (toroid.angle) for first traj.   = 0.000000000000000D-001
Phi_l: phi/pi     for last trajectory          = 1.600000000000000D-000

********** Fast ion parameters for ION regime *****
E_ev_f: ion start energy [ev] for first traj.  = 2.000000000000000D+004
E_ev_l: ion start energy [ev] for last traj.   = 2.000000000000000D+004

Pitch_f: start pitch angle/pi for first traj.  = 0.000000000000000D-001
Pitch_l: start pitch angle/pi for last traj.   = 0.000000000000000D-001

Azimu_f: start azimuth. angle/pi for 1st traj. = 0.000000000000000D-001
Azimu_l: start azimuth. angle/pi for last traj.= 0.000000000000000D-001

****************** Plasma parameters  ******************
T_e_axi: axis electron temperature [ev]        = 3.000000000000000D+002
T_e_wal: wall electron temperature [ev]        = 0.000000000000000D+002
s_T_e: electron temperature exponent           = 5.000000000000000D+000

N_sort_i: number of plasma ion sorts           = 2.000000000000000D+000

  Hydrogen ions:
A_i_____01: atom weight number of plasma ion   = 1.000000000000000D+000
Znu_i___01: nuclear charge number of plasma ion= 1.000000000000000D+000
Z_i_____01: charge number of plasma ion        = 1.000000000000000D+000
Exc_i___01: excitation index of plasma ion     = 0.000000000000000D+000
```

```
T_i_axi_01: axis ion temperature [ev]          = 2.000000000000000D+002
T_i_wal_01: wall ion temperature [ev]          = 0.000000000000000D+002
s_T_i___01: ion temperature exponent           = 5.000000000000000D+000
n_i_axi_01: axis plasma ion density [cm^{-3}]  = 2.000000000000000D+013
n_i_wal_01: wall plasma ion density [cm^{-3}]  = 0.000000000000000D+013
s_n_i___01: ion density exponent               = 1.400000000000000D+001

  Hydrogen neutrals:
A_i_____02: atom weight number of plasma ion   = 1.000000000000000D+000
Znu_i___02: nuclear charge number of plasma ion= 1.000000000000000D+000
Z_i_____02: charge number of plasma ion        = 0.000000000000000D+000
Exc_i___02: excitation index of plasma ion     = 0.000000000000000D+000
T_i_axi_02: axis ion temperature [ev]          = 2.000000000000000D+002
T_i_wal_02: wall ion temperature [ev]          = 0.000000000000000D+002
s_T_i___02: ion temperature exponent           = 5.000000000000000D+000
n_i_axi_02: axis plasma ion density [cm^{-3}]  = 1.000000000000000D+010
n_i_wal_02: wall plasma ion density [cm^{-3}]  = 2.000000000000000D+013
s_n_i___02: ion density exponent               =-5.000000000000000D+000
```

Radial electric field parameters (not implemented yet)

```
U_axis: potential at axis [ev]                 = 0.000000000000000D-000
s_U: potential exponent                        = 5.000000000000000D+000
dA_ph_min: min.rel.scale of depen. on A_ph     = 1.000000000000000D-002
           for time dependent case
K_eq_surf: key for surface measure value       = 1.000000000000000D+000
```

K_eq_surf=0 means measure is radii

K_eq_surf=1 means measure is Aph*rho

```
****************** Number of trajectories and points  *****
Ntraj: number of trajectories                  = 1.000000000000000D+000

Npo_Tim: max number of points on time map      = 2.000000000000000D+001
Npo_The: max number of points on theta map     = 2.000000000000000D+002
Npo_Rad: max number of points on radial map    = 2.000000000000000D+001
Npo_Phi: max number of points on phi map       = 3.000000000000000D+000
Npo_ThP: max number of points on theta mom. map= 1.000000000000000D+006
Npo_RaP: max num. of points on radial mom. map = 1.000000000000000D+006
Npo_PhP: max number of points on phi mom. map  = 1.000000000000000D+006
K_sum: K_sum=1 means save sum map              = 1.000000000000000D+000

****** Parameters for ion diffusion and current drive calculation *****
```

The diffusion coefficient for both the particles and their guiding centers are calculated according to the parameters below. As with the magnetic field line diffusion coefficient they are saved in the .rep file.

```
N_rad_diff: number of gradations in radial     = 4.000000000000000D+000
```

Section for diff.coef. calculation. Breaks the cross section into

N sections and calculates diff.coef. within each.

```
Rad_diff_02: is 2d  position r/a of gradation  = 1.000000000000000D-001
Rad_diff_06: is 6th position r/a of gradation  = 4.000000000000000D-001
```

Omitted boundaries are set automatically proportional to the area.

```
T_diff_min: min delta time*om_ci               = 1.000000000000000D-000
```

If D is calculated with time, not theta, then $D(N) = (\Delta r)^2 / (a * \text{T\_diff\_min} * 2^N)$

N is given in .rep files where D is reported.

The_diff_min: min.Theta/pi step for diff. calc.= 1.000000000000000D-001

Sets a reasonable value of an angular 'step' for diffusion

There are two ways of calculating diffusion coefficients.

1) D = ("delta r")^2 / "delta t"

2) D = ("delta flux")^2 / "delta theta"

K_rA_diff: key of radial measure for diffusion = 1.000000000000000D+000

KrAdiff=0 means the numerator is measured in r

KrAdiff=1 means the numerator is measured in normalized poloidal flux (A_ph*rho)

K_TThe_diff: key of "time" measure for diff. = 1.000000000000000D+000

KTThediff=0 means the denominator is measured in time

KTThediff=1 means the denominator is measured in Theta

K_cur_drive: key for current drive calculation = 1.000000000000000D+000

****************** Result parameters ****************
Ncolumn: number of columns in result files = 1.30000000000000D+001**

columns:
1 - point number
2 - x/a ion position
3 - y/a ion position
4 - z/a ion position
5 - t time [s]
6 - sigma *a Lyapanov exponent
7 - ion energy [ev]
8 - ion velocity pitch angle/pi
9 - ion velocity azimuth angle/pi
10 - x/a guiding center position
11 - y/a guiding center position
12 - z/a guiding center position
13 - mu adiabatic invariant

K_Lyap: 1 means calc. Lyapanov exp. = 0.000000000000000D+000

```
****************** Debugging and scheme parameters ****************
K_out: 1 means output numer.results on screen  = 1.000000000000000D+000

RelPre: relative precision for equ.solv.       = 2.000000000000000D-009
AbsPre: absolute precision for equ.solv.       = 4.000000000000000D-012

tr_min: minimal radius value                   = 2.000000000000000D-008
tau_min: minimal value of tau                  = 1.000000000000000D-001

Kj_r: radial mean current key                  = 1.000000000000000D+000
   KJR=0 means no radial current, KJR=1 means non-zero j_r


------- mode #1 -------------------
K__pert_ga_01: key what comp. should be gauged = 1.000000000000000D+000
```

If "K__pert_ga_##" is omitted then "K_All_pert_ga" is used.

K__pert_ga_## < 1 means no renormalization

K__pert_ga_## = 1,2,3 means B_the,rad,phi

K__pert_ga_## > 3 means |B|

```
N_The_gaug_01: theta number of the mode        = 1.000000000000000D+000
N_Phi_gaug_01: phi number of the mode          = 7.000000000000000D+000
B__pert_ga_01: B__ [G] of the mode in ga.point = 3.300000000000000D+001
```

;B_The_gaug_01: (synonym to "B__pert_ga_01")

```
K_ampli_ga_01: key for gauge amplitude         = 1.000000000000000D+000
```

K_ampli_ga_##<0 means gauge mode amplitude in the gauge point

K_ampli_ga_##>0 means gauge maximal mode amplitude on the surface

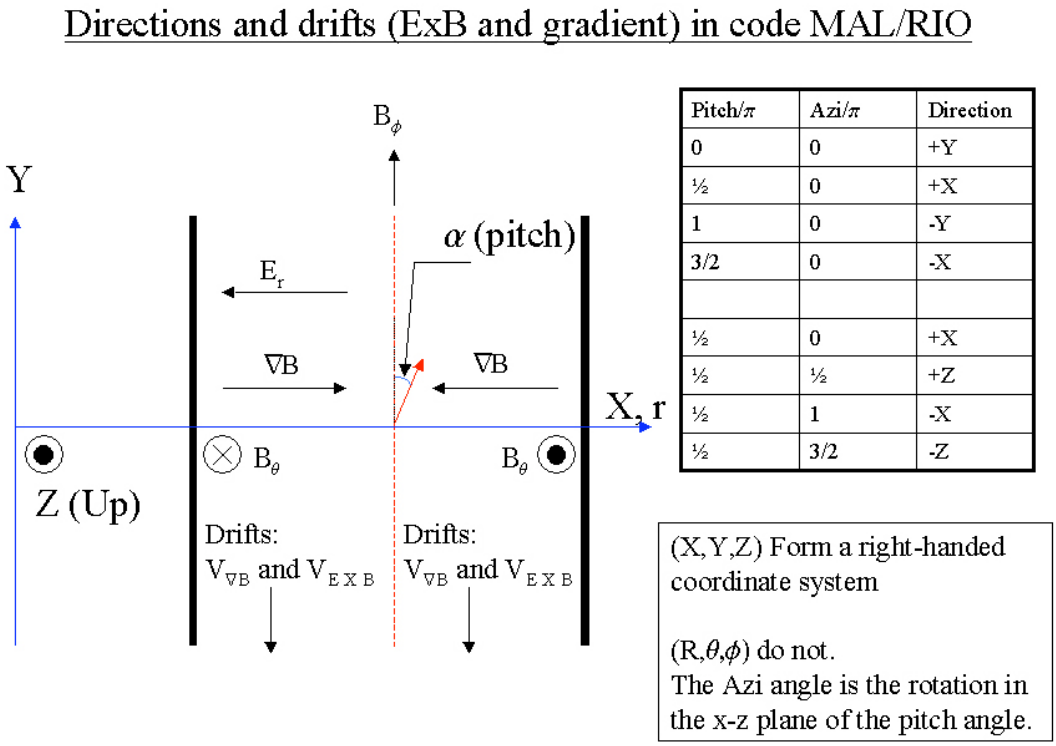
K_ampli_ga_##=0 means scale it with epsilon instead

|K_..._ga_##|=1 means gauge if the mode is present in *.fex file

|K_..._ga_##|=2 means gauge, stop code if the mode not present

```
K_phase_ga_01: key for gauge phase             = 0.000000000000000D+000
The_max_ga_01: theta/pi pos. of max of the mode= 0.000000000000000D+000
Phi_max_ga_01: phi/pi pos. of max of the mode  = 0.000000000000000D+000
```

(maximum of the same sign as "B_The_ga_##")



Pitch/π	Azi/π	Direction
0	0	+Y
½	0	+X
1	0	-Y
3/2	0	-X
½	0	+X
½	½	+Z
½	1	-X
½	3/2	-Z

Figure D.1 Coorindates in MAL/RIO. An important thing to note is that "*y*" and "ϕ" are oppositely directed.

A final word about the "beam mode" of the above input file is in order. What it does is after specifying a port location, beam density distribution, port width, etc. It randomly assigns a value of starting position, direction and energy to each particle according to what was specified, and each particle is then tracked individually. While a very powerful and accurate tool for specifying a full injection scheme, unfortunately its use is almost always prohibitive. A single ion orbit can take several days to run on a fast server. To do a full beam simulation, with enough single orbits to be meaningful would take months. This is why we concentrated on single particle orbits at representative positions and injection vectors. Also, with the beam simulation option, given the inherent

randomness of a distribution, a large number of trajectories would have to be followed in order to be sure that the result wasn't anomalous.

Finally I'd like to extend a special thank you to Yuri Tsidulko, who without developing MAL / RIO, his patience in teaching me how to use this code, and his support throughout the debugging process, my thesis would have taken on a much different character.