UNIVERSITY OF CALIFORNIA
Los Angeles

A Numerical Study of the Tokamak Edge Fluid Equations

A dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy
in Nuclear Engineering

by

Marcus Steven Day

1995

The thesis of Marcus Steven Day is approved.

Russell Caflisch

Farrokh Najmabadi

Gerald C. Pomraning

Xiaolin Zhong

Robert W. Conn, Committee Chair

University of California, Los Angeles
1995

To my wife and my parents...

Contents

List of Figures

ACKNOWLEDGMENTS

Some old adage says that you can learn the most when "sure-fire" methods refuse to work as advertized. I certainly faced more of that than I ever intended, and then the ensuing self-doubts as I tried to understand why. Throughout though, my wife Michele gave me the confidence and the reasons to challenge the frustration. My parents seeded and constantly reinforced my desire for quality and purpose; I hope their influence makes it through to this paper. Barry Merriman was my teacher, my conscience, my friend, and maybe the only one who really knows how hard we tried to get everything to work out pretty. Dean Robert Conn laid out a work environment consisting always of both freedom and support. Professor Farrokh Najmabadi provided thoughtful input, and defended my work throughout the years to ensure steady funding. Adrian Lenardic, my grad-school roommate introduced me to, and is a critical part of, a circle of eternal companions too numerous to list—my most fond and fun memories of graduate school will always involve this group, and the relaxing weekends we spent together. In addition to being a true friend, Neil Morley has always provided advice, dicussusion, questions and answers involving every aspect of Unix, graphics, LaTeX, Fortran, fluids, physics and math. The administrative staff at UCLA, including Haya Leiner, Charlotte Carter, Richard Kruse and Concha Soto helped to resolve all those funny little problems that arise along the way. In all, I will surely miss the environment of the UCLA campus that I came to know as home. I leave hoping that I have contributed in some way to UCLA's atmosphere of fun, unity and scientific purpose.

In addition, I wish also to acknowledge NeXT, Visual Numerics, Lighthouse, Stone Design, the makers of LaTeX, and Folger's for selling products that work. I thank the San Diego Supercomputer Center, especially Anke Kamarath, for providing painless access to, and unlimited use of the DEC Alpha Farm. This research was sponsored via the Department of Energy research contract number DE-FG03-86ER52130; I thank the powers that rule the DOE for letting our group convince them of what we felt were some important issue to be resolved. Finally, I thank the trailblazers in my field, particularly Bas Braams, Tom Rognlien and Erik Vold for fielding a constant stream of stupid questions, and giving me hope, support and direction. I truly have been blessed with a perfect surrounding.

VITA

December 12,1965	Born, Richmond, CA
1988	B.S. Nuclear Engineering UC Berkeley
1992	M.S. Applied Plasma Physics & Fusion Engineering UC Los Angeles
1988-1995	Graduate Student Researcher (supported by DOE Research Grant)
1994	Married a beautiful Italian girl who moved my life into a dream

PUBLICATIONS AND PRESENTATIONS

1. *ITER Divertor Modeling, Experimental Simulations and Advanced Materials Studies at UCLA; Results from the PPPL Linear Plasma Simulator*, Talk given by Lothar Schmitz at the Tech. Mtg & Wkshp on ITER Divertor Phys., Garching, Germany, Fefruary 21-25, 1995
2. *FMS: A Tool for Fluid Modeling of Plasmas*, APS Division of Plasma Physics Conference, Minneapolis, Minnesota, November 7-11, 1994
3. *Tokamak Edge Plasma Modeling and Code Development Issues*, presentation to the LLNL, Livermore, CA, October 14, 1994
4. *Discretization Issues for Highly Anisotropic Fluids*, APS Division of Plasma Physics Conference, St. Louis, MO, November 1-5, 1993
5. *Modeling of the Plasma Edge in Fusion Tokamaks*, Master's Thesis, 1992
6. *A Second Generation Code for Modeling Tokamak Edge Plasmas*, M. Day, B. Merriman, F. Najmabadi and R. W. Conn, Journal of Nuclear Materials, 196-198 (1992) 785-788

- continued next page-

PUBLICATIONS AND PRESENTATIONS - continued

7. *Applications of a Non-Orthogonal Edge Fluid Code for the ITER divertor*, APS Division of Plasma Physics Conference, Seattle, WA, November 16-20, 1992
8. *A Second Generation Code for Modeling Tokamak Edge Plasmas*, Tenth PSI Conference, Monterey, CA, March 30 - April 3, 1992
9. *Fully Implicit Fluid Plasma Simulation with a Generalized Flow Solver*, AGU Chapman Conference on Micro and Meso Scale Phenomena in Space Plasmas (funded by the American Geophysical Union), Kauai, HI, February 17-22, 1992
10. *Status of UCLA Edge Modeling*, Tokamak Edge Modeling Workshop, Livermore, CA, January 14-16, 1992
11. *Plasma/Neutral Fluid Simulations on a Non-Orthogonal Finite Volume Mesh*, APS Division of Plasma Physics Conference, Tampa, FL, November 4-8, 1991

ABSTRACT OF THE DISSERTATION

A Numerical Study of the Tokamak Edge Fluid Equations

by

Marcus Steven Day
Doctor of Philosophy in Nuclear Engineering
University of California, Los Angeles, 1995
Professor Robert W. Conn, Chair

A detailed fluid model is developed to describe the plasma in the regime found at the periphery of a tokamak fusion device. The model is geared particularly to investigating volume power dispersion methods recently proposed for high-power fusion reactor plasmas, and is evaluated in one and two dimensions for comparisons with data from the gas-target experiments at the PISCES-A facility at UCLA. Our model provides rather good agreement with experimental data, but requires excessive computational resources due to the low-order numerical discretizations employed to handle plasma convection. Development of the model is carefully documented, with emphasis on physical interpretation of *ad hoc* extensions commonly employed in the edge modeling community. The resulting equations are written in the two-dimensional poloidal plane of a tokamak in coordinates that align with the local projection of the magnetic field. There are a number of unique characteristics that require specialized numerical treatment for solution of this system, including strong directional anisotropy, stiff nonlinear coupling of the source terms, and detailed internal structure that requires fine grid resolution.

We solve the strongly nonlinear coupled set of modeling equations using a fully-implicit time evolution to steady state via Newton-Raphson iterations at each adaptively chosen time step. A very general damping strategy is implemented, along with standard convergence acceleration methods, such as mesh sequencing. The linear systems for each Newton iteration are solved via the BiCGSTAB conjugate gradient-based algorithm, which is preconditioned with line-by-line grid sweeps. Our fully-implicit discretization limits the class of available solution techniques to first order classical and exponential (streamline-weighted) dissipation. Because of this, and the profiles computed, excessive resolution requirements often preclude spatially converged steady solutions to our model description.

As a significant, but somewhat tangent project, we investigated a novel technique for treating the strongly anisotropic dissipation using a numerical grid not aligned to the principal diffusion axes, despite alignment of the underlying coordinates of the model. We rigorously demonstrate that no discrete algorithm can exist for consistently approximating such a scenario unless the grid is (nearly) aligned. The strength of this condition is a function of both the anisotropy and the rotation angle of the principal axes of dissipation. The results underscore the essential 1D nature of the fluid transport in the poloidal plane of the tokamak edge, while the big-picture results underscore the influence of numerical computing limitations on the development and understanding of edge plasma models.

Chapter 1

Introduction

Nuclear fusion offers one particularly attractive energy source to be harnessed for long-term national production of electricity. But a myriad of technological issues stand in the way of its viable commercial implementation. At the root of the problem, a burning fusion plasma must operate at temperatures in excess of 250 million degrees! When this plasma comes into contact with a confining material vessel, the surface is subject to huge particle and energy fluxes, which erode and heat all the plasma facing components. The exhaust region in a tokamak reactor must accommodate the intense power and particle loads from a burning plasma, while minimizing the flow of impurity ions from the surfaces to the core region. Robust and versatile solutions for this difficult problem have yet to be verified, primarily due to the lack of high-powered simulation experiments. Experimental equipment is often prohibitively expensive, so much of the preliminary concept evaluation and design will have to be carried out via computer simulation.

In this work, we explore computer models of the exhaust region in a tokamak where plasma comes in contact with material surfaces. We develop a precise set of modeling equations, and then proceed to solve them for a number of different scenarios, with the intention of evaluating specific design concepts. Our objective has been to study the gas-target scenario, a promising concept where tokamak exhaust would be diverted into a remote handling chamber that is filled with neutral gas. If the concept is successful, a large fraction of the plasma energy will be uniformly dispersed over a the wall of the divertor chamber via atomic radiation processes.

The gas-target concept relies heavily on the interaction of the charged plasma fluid that exhausts from the core, with neutral and impurity species puffed in through the boundary. The relevant atomic processes include ionization, charge exchange and recombination. The modeling equations for this system involve highly anisotropic diffusion, temperature dependent sources and intricate two-dimensional domain configurations. A large portion of this work has been devoted towards developing sufficiently general numerical methods to deal with such complexities, and a significant portion of this dissertation describes both the detailed model and the solution techniques under development.

In Chapter 2, we provide a short review of the tokamak configuration, followed by a summary of processes that are important to the edge plasma region. There, we define the necessary terminology and motivate developing our advanced modeling and numerical computing machinery. In Chapter 3, a literature review is presented, both to describe previous works in this area and to highlight some of the shortcomings. We then attempt to address these shortcomings in later sections. A specific set of modeling equations is formulated subsequently in Chapter 4 for the special plasma of the tokamak edge. We show how this model is to be interpreted with respect to standard Newtonian mechanics, and that our results appear to disagree with other models that have been published. Numerical computing tools, designed specifically for this problem, are then detailed in Chapter 5. We motivate our unique numerical treatment for strongly anisotropic dissipation, and in Chapter 6, proceed to demonstrate its usefulness and limitations. In Chapter 7, our resulting model and solution techniques are applied to standard benchmark scenarios and in Chapter 8, to one and two dimensional descriptions of simple plasma configurations. The model is found to be in fairly good agreement, though indicates that the model is still missing some key ingredients. A list of conclusions from the work follows in Chapter 9, along with a list of suggested follow-on efforts and unresolved issues.

The text of the dissertation body is then followed by a number of smaller Appendices. Appendix A specifies the fundamental theoretical model from which we derive some of our transport coefficients. In Appendix B, we detail the formulation of the source term expressions employed in the model. In Appendix C, we describe a particularly difficult issue encountered in the numerical modeling of the plasma heat transport equation, stripping away all other processes for simplicity. This difficulty 'hides' itself in underresolved plasma simulations to give poor initial convergence results. Appendix D contains details involved in computing the rate of numerical convergence for the computer code that implements the numerical schemes described as part of the thesis, while Appendix E gives an example model specification file used to match experimental data on the PISCES-A facility. The purpose is to convey clearly the actual model implemented, while demonstrating simultaneously the power of our software design philosophy.

While this research is motivated by the need to find solutions to this tokamak edge problem, we end up often with a focus on the numerical and theoretical issues that arise during the modeling process. In fact, we have identified a universal limitation on available techniques to approximate numerically the strongly anisotropic plasma dissipations. We implemented a particularly useful coding technique that eases incorporation of our complex models. Then, through cursory applications of our methods, we found that our models predict fairly accurately, the one-dimensional behavior of the gas-target experiments in the PISCES-A facility at UCLA. When trying to extend the model to two dimensions we quickly ran into computer hardware limitations, and thus obtained only minimal results. The gas-target regime, in general, was found to contain internal features that are beyond reasonable numerical resolution capabilities, even in one dimension. Perhaps most importantly though, we found that future understanding of the detached plasma state is wholly dependent on quantum improvements to present numerical solution capabilities, above and beyond that presented in this work.

1.1 Summary of Results

1. Physical models were assembled to produce precise equations for the tokamak edge plasma. Our results differ significantly from a number of similar models that have been published in the literature. A simple picture is given to assure that our model is consistent with fundamental Newtonian dynamics, and to outline a process for adding new physical processes to the equation set.

2. To a good approximation, the edge plasma obeys strongly anisotropic fluid conservation equations. Numerical solutions proceed typically on a numerical grid that is aligned with the anisotropy. But there are many practical reasons why this situation is not optimal. A generalized criteria is developed for

whether strongly anisotropic flows can be consistently discretized along mesh coordinates skewed to the anisotropy, and an optimal finite difference stencil is presented. The conditions allowing misaligned discretizations are satisfied only near divertor X-points; we discuss this situation in detail.

3. A representative edge model was constructed to illustrate the gross influence of key edge plasma transport parameters. Simple pictures are given that quantitatively explain the results of a number of parameter variation studies. The most dominating effect seems to arise from methods used to artificially limit the electron conductive heat flux. Such "flux limiters" typically restrict heat fluxes to within physical limits; however our experiments indicate that the accepted algebraic form is unduly sensitive to undetermined parameters. There has been recent work to extend these models; implications are discussed.

4. A steady detached plasma, produced in the UCLA PISCES-A facility, was modeled in one and two dimensions. The one-dimensional model consists of fluid equations for the background Hydrogen plasma (ions and electrons) in addition to the reaction-diffusion equations for atomic and molecular Hydrogen. The problem was posed along the central axis of the cylindrical plasma device. Preliminary results are presented for the extension of this model to include the radial dimension. The problem is difficult to model due to the small system size with respect to collision mean free paths, and the inefficiencies of the underlying numerical schemes used to solve the modeling equations.

5. To carry out the research described in this dissertation, a flexible programming model was developed that allows easy incorporation of complex physics modules in a programming-language-independent way. We implemented this model via a Fortran prototype computer code, FMS2D (Fluid Modeling System in Two Dimensions)[1]. We present details of the code, in terms of the major algorithm selections and an example model specification.

---

[1] It is customary to "personalize" computer codes of such magnitude with a catchy name before it is passed on to future researchers. However, we will show that algorithms in the code suffer from serious efficiency issues. The code will certainly be rewritten before it can be useful for routine two-dimensional production modeling runs.

Chapter 2

Background

In a fusion reactor, light atoms (usually the isotopes of hydrogen, namely deuterium and tritium) are heated to high enough temperatures that there is a sufficient rate of fusion. For many fuels, this process changes the total mass of the particles involved, and releases a large amount of energy equivalent to the mass discrepancy (according to the Einstein formula, $E = mc^2$ where m = mass change during the reaction, and c is the speed of light). The energy is carried by the kinetic energy by the products of the reaction. For deuterium and tritium, the products are generally helium and neutrons. The helium is retained in the mixture to heat the remaining fuel, while the neutrons are removed, and used to drive a standard heat conversion cycle. The following lists the most promising nuclear reactions for a first generation fusion power reactor [1]:

$$\begin{aligned}
\mathrm{d} + \mathrm{t} &= \alpha + \mathrm{n} + 17.56\ \mathrm{MeV} \\
\mathrm{d} + \mathrm{d} &= \mathrm{t} + \mathrm{H} + 4.03\ \mathrm{MeV} \\
\mathrm{d} + \mathrm{d} &= {}^3\mathrm{He} + \mathrm{n} + 3.27\ \mathrm{MeV} \\
\mathrm{d} + {}^3\mathrm{He} &= \alpha + \mathrm{H} + 18.30\ \mathrm{MeV}
\end{aligned}$$

The numbers on the product sides are the amount of energy released per reaction. In a mixture at any temperature, all of these reactions will occur to some extent, but they are dominated by thermalizing coulomb elastic and inelastic col-

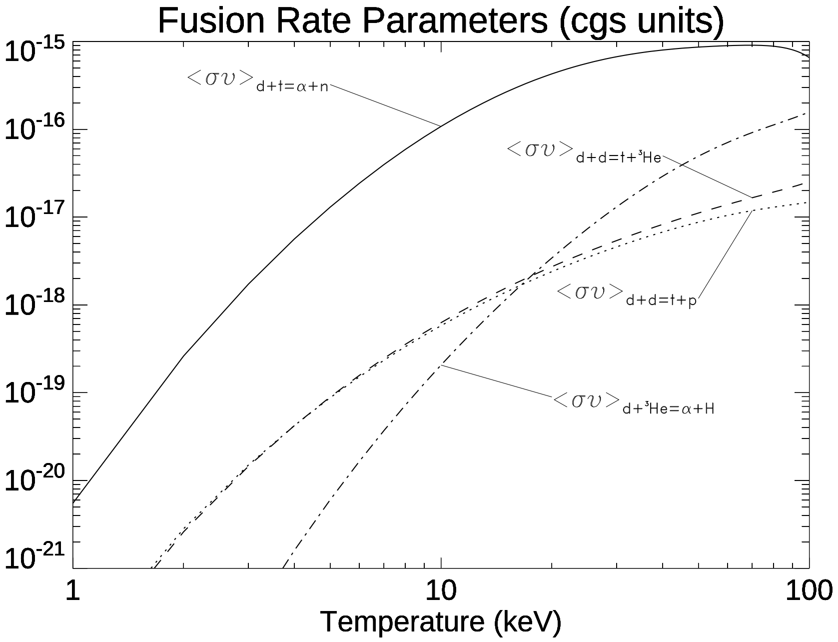

Figure 2.1. Fusion reaction rate parameters. Constructed from the fit table, 2C2, of the reference.

lisions. With increasing temperature however, the coulomb interaction rate falls off sharply, while the fusion rates increase. The energy emitted from the fusion reactions eventually becomes a significant fraction of that required to heat and maintain the system. The condition, 'break-even', where the fusion energy exceeds the energy input into the system, is a minimum step in achieving an economical power producing facility[2]. Figure 2.1[1] shows the fusion rate parameters[1] for the four candidate reactions as a function of temperature. Fusion rates are very low for temperatures below about 1 keV (1 keV≈10,000,000C°); loosely speaking, break-even is expected at temperatures near 15 keV. Development of fusion-based electricity generating devices centers around producing a method to contain and heat the reacting mixtures to the extremely high temperatures where fusion rates

---

[1] The fusion rate per unit volume is proportional to the particle densities of the reactants. The plotted rate parameter is the constant of this proportionality.

become large enough to produce net power.

The confinement properties of charged particles in a magnetic field suggest one method of containing the burning fusion fuel. When heated to these extreme conditions, electrons are stripped from the fuel atoms, and the mixture becomes a plasma. Charged particles of a plasma are known to be confined, via the Lorentz force, to helical orbits about a strong magnetic field. However, since there will be no applied force *along* the field line, motion of the particles in that direction will not be not impeded. One can build a sufficiently long, straight system (greater than 1 km), or the magnetic field that confines the charged particles can be bent into a closed configuration. Examples of plasma devices that employ these closed magnetic structures are tokamaks, stellarators and reverse-field pinches[3]. The work here is primarily concerned with the tokamak geometry, the leading design concept, where the confining magnetic field is shaped into a doughnut-like configuration using large external magnetic coils.

In a tokamak, there is a strong magnetic field in the toroidal direction (the long way around the doughnut), along which a large plasma electric current is driven. This toroidal current induces a poloidal (the short way around) magnetic field, and results in a corkscrew or helical total magnetic field structure. The helixes of the tokamak do not, in general, meet on themselves each time around the torus. Rather, with successive toroidal passes they ergodically trace out tubes that are (nearly) circular in the poloidal plane, as illustrated in Figure 2.2 (it turns out that elongated D-shaped cross sections may provide better confinement properties for a tokamak plasma; cf. [4] for further discussion on this matter). A tokamak configuration is a collection of many such surfaces, nested one inside the other,

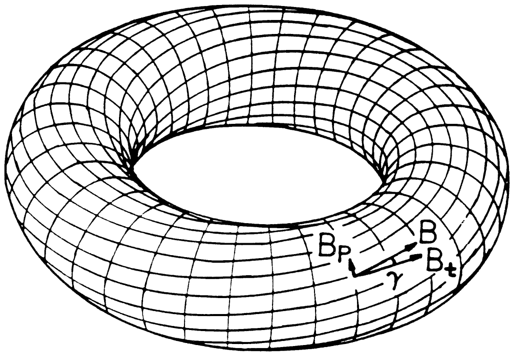

Figure 2.2. Ergodic surfaces in a toroidal geometry. Note that the magnetic field lines run primarily in the toroidal direction. The twist in the structure is such that a cut through minor axis (the short way around the doughnut) will trace a circular outline of the surfaces.

with successive outward levels surrounding a larger number of magnetic field lines, or equivalently, a greater magnetic *flux*. An expansion force arises from the closed field lines and will tend to drive the configuration unstable (i.e. to straighten the field lines), so that a third, vertical, field is necessary. These vertical field lines are constructed to bow toward the tokamak major axis, providing a potential energy minimum to prevent vertical *and* radial shifts of the tokamak plasma/field structure.

Since plasma motion is impeded across the lines of magnetic force, the magnetic surfaces form a rather convenient natural coordinate system in which to formulate the physical transport properties of the system[5]. The surfaces are those of constant magnetic flux, or *flux surfaces*, and the specialized orthogonal magnetic coordinates are aligned locally along the gradient vector of the magnetic flux surfaces and parallel to the magnetic field lines[2].

---

[2] The third vector vector in this 3D orthogonal coordinate system, the “drift direction”, is mutually orthogonal to the first two coordinates

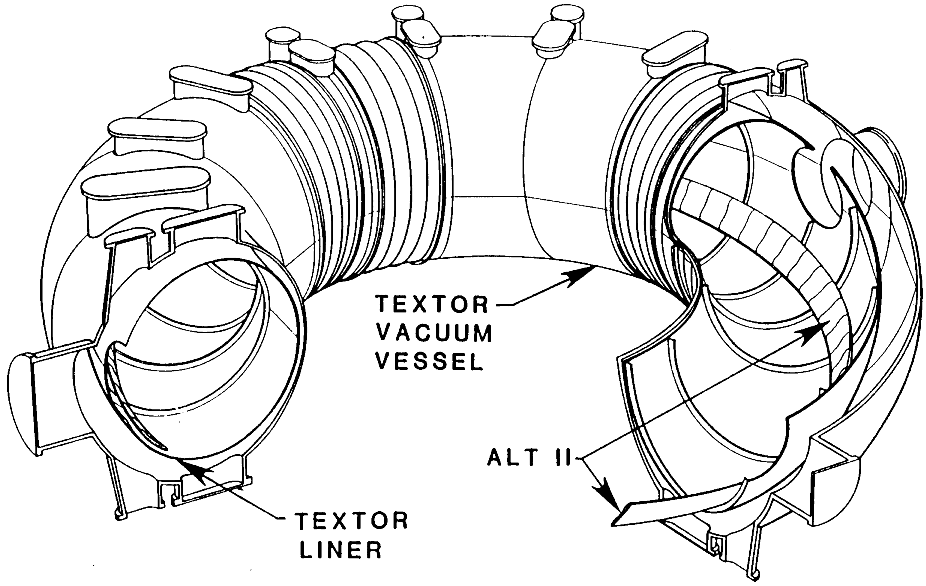

Figure 2.3. Typical toroidal plasma vessel to contain the doughnut-shaped magnetic field structure, shown here in the poloidal plane. Note the irregularly shaped Plasma Facing Components (PFCs) along the inner lining of the vessel.

2.1 The Edge Plasma

The core plasma of a tokamak envelops a set of nested magnetic flux surfaces as described earlier. But for a number of practical reasons, components inside the reactor vessel designed to receive the boundary plasmas are not generally aligned with these flux surfaces (a typical toroidal plasma vessel is pictured in Figure 2.3). The shape discrepancies give rise to a set of field lines that terminate on the walls and internal structures of the vessel, and thus trace out *open flux surfaces.* In contrast to the core particles moving on closed surfaces, those in the edge come in direct contact with structures along the vessel wall, i.e. the plasma facing components (PFCs). Outward flow (exhaust) of the confined plasma to the open surfaces results from collisional and turbulent diffusion[6] processes. The *edge plasma* (also, the boundary or scrape-off-layer [SOL] plasma) is the zone of transition between

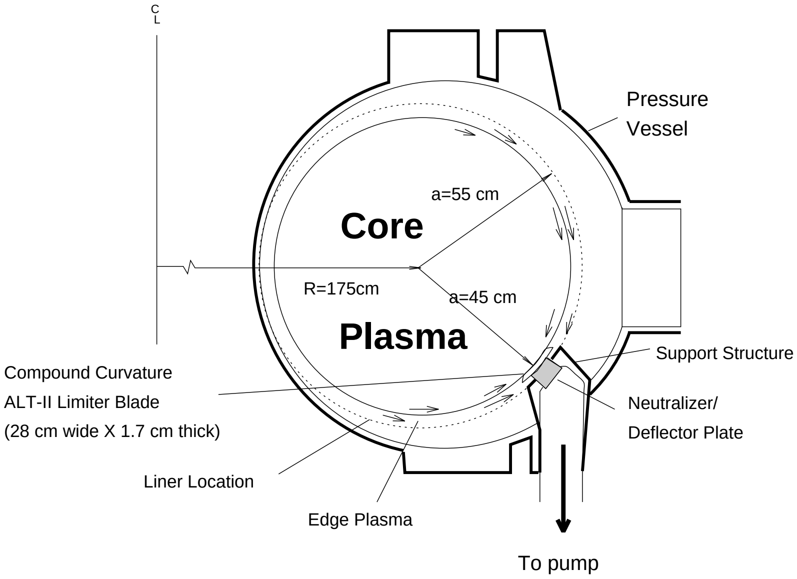

Figure 2.4. Typical limited plasma configuration. The pressure vessel pictured is toroidally symmetric, as is the ALT-II limiter blade. In this plane, particles flow primarily in the direction of the arrows, and into the scoops containing neutralizer plates and pumping ducts.

the confined closed surface region, to the open flux surface boundaries. Particles and energy diffusing from the core will erode PFCs and produce sputtered impurities that contaminate the plasma. Control of the particle and power flux through the edge to the material surfaces of PFCs is a critical design issue for high power fusion devices, such as ITER, and is fundamental to the ultimate success of a burning plasma in any fusion reactor[7].

In a tokamak, two types of systems have been employed with success to control the exhaust flows in the edge. A *limiter* is an object placed in the path of the flowing boundary plasma to impede direct access to material surfaces "downstream", while a *divertor* reroutes the flow of the plasma by altering the shape of the confining magnetic field. Figure 2.4 displays schematically a typical limiter configuration in the poloidal plane. Note that in the edge, the bulk flow of plasma is along the magnetic field. In the plane shown, the magnetic field direction has a small component tracing out the projection of the flux surface. In the poloidal plane, flow along the field line appears to lay along the projected flux surface, as indicated by the arrows in the figure. A limiter "blade" is often used used to scrape off boundary plasma before it can reach more sensitive PFCs lined along the the inside edge of the vessel. The ALT-II limiter(c.f. [8]), for example, also utilizes a pump behind the scrape-off blade to exhaust some of the recycled and sputtered gas created near the blade's leading edge. This particular experiment was designed to show that helium can be exhausted from a fusion plasma, and to control generated impurities from streaming back towards the core plasma. Limiters may take on a variety of geometries, from fully toroidal belt-type limiters[9, 10] to local limiters placed for protection around sensitive equipment, to the more intricate shaping of the ALT-II blades.

A typical magnetic divertor configuration is shown in Figure 2.5. A divertor uses shaped magnetic topologies, instead of a physical component, to scrape off the edge region of a tokamak plasma and direct it to a remote region where the plasma-materials interaction takes place. In the remote chamber, increased impurity, neutral and plasma densities can be tolerated without as seriously degrading the quality of the core plasma. A wider array of schemes are thus available to reduce the intense particle and heat fluxes. With a magnetically diverted plasma, the increased flexibility in flux-reducing schemes are available at the cost of creating and maintaining the complex magnetic field structure. This additional cost may be substantial or prohibitive.

Although these two designs have served low power experiments well, the peak

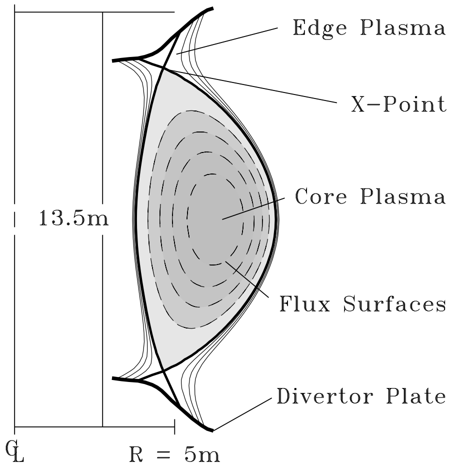

Figure 2.5. Typical diverted plasma configuration. ITER flux surface data provided by B. Lee, 11/93. Note that there are open and closed flux surfaces, and the the divertor plates are usually designed to accept the open flux surfaces at a grazing angle. It should also be pointed out that field lines intercept the plane of this figure nearly perpendicular; the plate to field pitch angle is usually of the order of 1 degree

heat loads for the next generation of tokamak experiments, such as ITER, will be considerably higher. The nominal power during burn for ITER is 1.5 GW and 20% of this power is carried in the form of the helium ions (alpha particles). It is expected that 80% of this 300MW is intercepted by the divertor. For a typical power decay length the resultant power flux to the target plates would be roughly 35 MW m^{-2}[11]. Aggressively engineered heat panels can presently accept fluxes on the order of only 5-8 MW m^{-2} in steady state. Simple procedures for merely terminating the plasma on divertor or limiter target surfaces appear not to be adequate because the effective "wetted area" of the plates will be too small to lead to acceptably low heat fluxes.

The above requirements have led to consideration of various *volume energy dissipation* schemes, such as the gas-target, radiative and volume recombination divertor scenarios. In all these cases, a large fraction of the power into the divertor region would be distributed uniformly over the divertor walls by atomic radiation processes. But such regimes remain largely uncharacterized because they are difficult to achieve in the low-power experimental devices available today. Computational tools will fill this gap, helping to provide an understanding of the stability and accessibility of these regimes.

2.2 Tokamak Edge Plasma Processes

Important physical processes occurring in the edge plasma of a tokamak device are detailed below. From this list, we extract a fundamental set to include in our model, but also see how our approximations fit in with the bigger picture. Details are given regarding the transport of plasma, neutrals, and impurities, as well as how the plasma-facing surfaces are affected. Also included is a summary of diagnostic techniques so we can touch base with the experimentally measured quantities.

2.2.1 Plasma Transport in the SOL

- The advection of plasma particles is limited to only one direction because they are bound to the local magnetic field by the Lorentz force. However, collisions combined with turbulent convection may move a particle from one field line to the next, and lead to macroscopic cross-field transport[12, 13]. The time scale for cross field transport is typically much longer than parallel advection times, but the domain length in the parallel direction is a great deal longer. As a result, the edge problem is indeed two-dimensional. These strongly anisotropic flow characteristics lead to significant numerical difficulties[14], which will be discussed in subsequent sections.

- The electrons in a plasma possess far greater mobility along the field lines[15] than across. As a result, they would tend to leave the plasma along any open field lines much faster than the ions. However, an electric sheath potential is set up automatically to retard electrons, and enhance the flow of ions, to preserve overall charge neutrality[16], net current. This parallel electric field, referred to as the sheath/presheath structure, extends from the material walls some distance into the plasma, and dramatically influences the transport throughout the edge region.

- The presence of gradients in magnetic and electric fields lead to distortions in the gyro orbits of particles about the field lines[17]. When averaged over a large number of particles, these distortions appear as apparent cross field drift flows, primarily within the flux surfaces. Drift flows will significantly affect the edge profiles (c.f. [18]), and may be manipulated directly or indirectly via externally controllable parameters, such as electrical biasing of the SOL (c.f. [19]).

- Turbulent fluctuations in the edge quantities[20], driven by profile gradients[21] and velocity shearing[22], lead to outward transport not easily modeled by the fundamental fluid conservation equations. This anomalous (i.e. phenomenological) transport is modeled heuristically[23]—typically including terms for radial diffusion and an inward "pinch" advection[24]. The models are not well established or validated, but have been necessary to match experimental results, as classical theory grossly under-predicts the observed cross-field transport[25, 26].

2.2.2 Neutral Gas Transport

- Neutral particles are introduced into the edges of plasma devices as a result of plasma neutralization and recycling, thermal desorption, and direct gas puffing or volume recombination—a primary source being the the material walls of the vacuum vessel. Because neutral particles are not influenced by the shape of the magnetic field, vessel geometries can play a key role in determining neutral profiles. And because volumetric ionization sources strongly impact the plasma profiles, the vessel geometry can in turn have large effects on the plasma.

- Recombination and inelastic collision events generally produce plasma and neutrals in an excited state. Line radiation is then emitted as the atoms relax to ground state, and the radiation is transported through the edge region to the PFCs. The plasma additionally emits Bremsstrahlung and cyclotron radiation due to collisions and confinement by the magnetic field. As a whole, the radiation generally aids to diffuse the intense power loads to in-vessel components, and therefore should be included as a volumetric energy sink to the edge plasma.

- Neutrals are actively pumped from the plasma devices. In steady state, the neutral pumping rate balances the total flow of particles from the core region, in addition to those neutrals introduced through puffing and wall desorption. Neutrals introduced into the system diffuse into the plasma and become ionized via collisions with energetic electrons. Likewise, the plasma particles may recombine, either in the volume or at the material surface. *Recycling* is the balance of these phenomena, and may play an important role in the redistribution of exhaust heat flux in the edge.

2.2.3 Impurity Transport

- Impurities may be introduced into the edge region via sputtering from the in-vessel components or by chemical reactivity between Hydrogen and the wall material. If the emitted impurities are electrically charged, they will be accelerated back toward the surface by the sheath potential, and may cause additional sputtering. For some wall materials (such as Carbon and Tungsten), self-sputtering may become a runaway process under the proper conditions[27]. For most cases, sputtering or chemical reactivity leads to a neutral impurity atom or molecule reaching the plasma, where it is subsequently ionized and participates in the recycling process.

- If the impurities escape the sheath potential, they too will become involved in the recycling and radiation processes. Most often, the impurity species radiate energy a great deal more efficiently than the background hydrogen species[28] (e.g. via Bremsstrahlung radiation), and represent a very effective power redistribution process. However, if the impurities are transported into the core region, this increased radiation will tend to extinguish the burning plasma[2, 29]. Impurities can reach the core by charge exchange, diffusion, entrainment (if the bulk plasma flow is inward) and anomalous "inward pinch" convection processes.

2.2.4 Surface Processes

- Material surfaces in contact with the plasma act as neutralizer plates for the exiting charged species[30]. Ions and electrons recombine at the surface to form neutral gas molecules which then diffuse back into the plasma volume from the surface. In the bulk edge plasma, the molecules may be dissociated into neutral atoms, and therefore contribute to volumetric sources of both neutrals and ions, enhancing local particle recycling. This process plays an key role in providing a volumetric source of plasma in front on divertor strike points that lowers plasma temperatures near the target plates.

- Secondary (and tertiary) electron emission from the plasma-facing surface may be caused by electron, photon or ion impact, and will tend to modify the sheath potential[31, 32] by altering the effective number of charge carriers to the wall. Neutral collisions in the sheath will also modify the sheath potential by impeding direct flows, and providing ionization sources. These complex effects remain to be effectively modeled, or even parameterized for application in fluid models.

- Sputtering yields may be complicated by chemical sputtering and radiation enhanced sublimation[33] in lower temperature plasmas (e.g. divertors, SOL plasma near limiters, etc). The effect is of particular concern for Carbon-based PFCs, where the sputtered yield includes chemical processes and yields a myriad of hydrocarbon species that radiate and dissociate throughout the system[34].

2.2.5 Other Processes

- In most tokamaks, the edge is truly three-dimensional in nature (i.e. the toroidal coordinate may not be reasonably ignored), due to toroidal asymmetries in the PFCs or magnetic field. Examples include the presence at discrete toroidal locations of neutral pumping ducts, diagnostic and beam ports, antennae, bundle divertors, and bumper limiters. Additionally, imperfections in the magnetic coil set may result in toroidal field ripple and uncertain separatrix and field null locations.

- Transient effects are observed under typical experimental conditions. While operating tokamaks in H-mode[35], edge localized modes[36] (ELMs) occur frequently, providing transient fluxes of particles and heat across the core/edge interface. ELMs account for a majority of the transport during

H-mode operations on DIII-D[37].

- Oscillatory fluctuations of plasma density and electric potential in the edge indicate the presence of electrostatic turbulence and the possibility of convection cells in the poloidal plane[38]. The cells have been observed on CCT[6] and TEXTOR[39], and appear to account for a large fraction of the overall particle transport. The presence of these cells requires the existence of pressure-driven convective radial flow not present in current theories of anomalous transport.

2.2.6 Experimental Data

A key aspect of modeling plasma systems is direct comparisons to experimentally observed data. It is instructive to list the types of measurements currently available. There are several tokamak devices in operation world wide that are used in the study of plasma exhaust systems. A partial list includes ASDEX[40] (Garching, FRG), DIII-D[41] (San Diego,CA), JET[42, 43] (Culham, UK), TFTR[44] (Princeton, NJ), TEXTOR[45, 46] (Julich, FRG), TEXT[47, 48] (UT, TX), TdeV[49] (Varenne, Canada), CCT[6] (UCLA, CA), ALCATOR C-MOD[50] (MIT) and others. Additionally, there are smaller, specialized plasma devices used to simulate parts of a tokamak plasma under more controlled (and less expensive) conditions, such as PISCES[51] (UCLA, CA). Data is retrieved as follows:

- Langmuir probes obtain average and fluctuating electric potential, electron density and temperature profiles as a function of space and time.

- Lithium beams measure density profiles–usually deeper into the plasma than standard Langmuir probes.

- Mach probes are used to measure plasma velocity profiles.

- CCD cameras measure the D_α and H_α light emitted by inelastic electron scattering with neutral hydrogen.

- Impurity spectroscopy is used to measure impurity concentrations and can provide ion temperature profiles and plasma flow velocities.

- Bolometric panels placed on the vessel wall permit determination of frequency-integrated power deposition rates.

- IR TV determines power deposition on strike point locations.

- Pressure gauges measure the background neutral gas density.

- Miscellaneous devices monitor impurity concentrations.

- Ragowski coil systems and Mirnov loops measure the magnetic background and fluctuating fields–allowing, for example, real-time corrections to the detailed shape of the confining field in a divertor channel.

- Thermocouples, photo-diodes, tile current gauges as well as other miscellaneous devices are employed to measure the non-plasma conditions that will help determine vessel interactions with the plasma.

Experimental data in the edge is generally available at only a small number of locations in the tokamak. Most of these measurement techniques require significant interpretation to arrive at two dimensional profiles for the edge plasma state properties. In fact, interpretation of diagnostic measurements is probably the largest area of research in the fusion program. As a result, benchmarking a two-dimensional predictive computer model is a difficult task at best—the uncertainties in the models are at least as great as many of the uncertainties in the data. Regardless, we aim to reproduce gross features of the edge plasmas, such as density, temperature and flow profiles. To this end, earlier efforts in the area, to be described in the next section, have been reasonably successful, and provide inspiration and direction for our efforts.

Chapter 3

Literature Review

Present models of the edge plasma contain a wide range of sophistication, depending strongly on their implementation. Accordingly, no attempt will be made here to survey the vast number of independent modeling techniques and efforts. We will instead try to point to methods that have been successfully applied to the SOL for our purposes, and relay the general context of their application. A general review of the present state of research in this field is available in References [52] and [53]. We emphasize arbitrarily the specific modeling efforts that have had particular influence over the solution techniques we have chosen to implement.

3.1 Specific Problem Definition

First, however, it is a good idea to be very specific about the problem we are interested in solving. Computational solutions of the nonlinear transport equations are sought for the fluid regime of plasma (these equations will be presented in detail in Chapter 4). The tokamak poloidal plane (cf. Figure 2.4) is the two dimensional domain of interest, and we limit the analysis to regions where magnetohydrodynamic (collective) effects are negligible (see the following note). Further, the detailed surface effects described in earlier sections are neglected for the most part except as they influence boundary conditions to our simplified system. Similarly, drift flows, turbulent fluctuations and time dependence are ignored. The core plasma is external, fixed information and an electrostatic sheath is assumed to exist at all non-parallel (to the magnetic field) interfaces between plasma and material. The goal is to understand the behavior of the complex edge plasma, as a separate entity, in terms of the influence of geometry and various physical processes internal to the edge plasma fluid. Models of this genre have been proposed to provide guidance to the ITER design team for selection of divertor concepts, and have been applied with some success to the data measured on all current tokamak experiments.

Collective Effects

A significant feature of plasma is *collective behavior*, where mutual electric and magnetic fields created by the perturbation of a small part of the system, have significant effects on the overall system. The feature gives rise to a rich variety of electromagnetic wave propagation characteristics, but also to great difficulty in formulating a tractable model of a plasma system that is physically realistic. Additionally, it is unclear at this time, which of the vast extant wave propagation/excitation modes are predominantly responsible for the particle and energy transport. The time-dependent collective behavior phenomena (i.e. electromagnetic wave propagation) is generally ignored in edge plasma modeling efforts, mostly because they present far too great a computational burden at this time.

3.2 Fluid Model Validity

Since the model to be employed is based on the "fluid approximation", some comments seem appropriate regarding its validity for tokamak edge plasmas. Formally, a fluid model is valid if the mean free path for randomizing collisions is somewhat less than the scale lengths characterizing the profile gradients or the overall system size. For a plasma, the mean free path parallel to the magnetic field is approximately τv_{th}[12] (v_{th} is the thermal velocity, $v_{th} = \sqrt{kT/2m}$, and τ is the mean time between collisions). In MKS units (T in eV), the parallel mean free path between collisions for ions and electrons, λ^e_{mfp} and λ^i_{mfp} respectively, are then

$$\lambda^e_{mfp} = 3.44 \cdot 10^3 \frac{T_e^{\frac{3}{2}}}{\ln \Lambda} \sqrt{\frac{kT_e}{2m_e}} \qquad \lambda^i_{mfp} = 2.08 \cdot 10^5 \frac{T_i^{\frac{3}{2}}}{Z^4 \sqrt{m/m_p} \ln \Lambda} \sqrt{\frac{kT_i}{2m}}$$

(all the variables have their usual definitions[54]) The particle masses cancel from these formula, and for singly charged ions, the mean free path is approximately the same for either species ($\lambda^e_{mfp} \sim \lambda^i_{mfp} \sim \lambda_{mfp}$).

"Mild" edge plasmas have density and temperature, (n, T), on the order of (10^{20}m^{-3}, 20 eV), while "high-power" plasmas occur with corresponding values, (10^{19}m^{-3}, 200 eV). Thus

$$\lambda^{\text{low T}}_{mfp} \sim .034\text{m} \qquad \lambda^{\text{high T}}_{mfp} \sim 34\text{m}$$

For comparison, a typical field line connecting the core to the divertor target for ITER, say, has length on the order of $2\pi R q \sim 160$m (where R is the tokamak major radius and q is the field rotational transform). In either case, it would seem that there are only marginally enough collisions to warrant fluid approximations. However, processes near the material surfaces lead to strong energy sinks, and subsequently to local regions of short temperature gradient scale lengths. It happens that in typical tokamak plasmas, this electron temperature drop is often steep enough to invalidate fluid models. Under these conditions, heat flow is dominated by *non-local* transport[1] (i.e. energy flows into a region from locations having significantly different equilibrium characteristics). Some caution therefore should be exercised when interpreting fluid results in this region of the modeled domain.

Perpendicular to the magnetic field, where anomalous transport (primarily diffusive) is assumed to dominate flows, the corresponding λ_{mfp} is D/v_{th} ($D(\sim \chi \sim \eta)$ is the (anomalous) cross field particle diffusivity, on the order of 1 m^2/s, χ the anomalous heat diffusivity and η the anomalous viscosity). The perpendicular λ_{mfp} is on the order of 10^{-6} m, and well within the range of fluid validity.

When the fluid model is strictly invalid over a significant portion of the domain to be modeled, there exists a hierarchy of alternative levels of approximation. If the system is collisional enough to thermally equilibrate, but the particles traverse the system before they can communicate parallel and perpendicular energies, then the equations of magneto*gyro*dynamics[55], which maintain an anisotropic hydrostatic pressure tensor, may apply. If the parallel transport is fast enough that particles move through scale lengths even before establishing thermal equilibrium along the field line, then a gyro-kinetic[56] approach is more appropriate. Finally, if the system is small or structured enough that particles cannot complete uninterrupted orbits about the local magnetic field, a particle approach[57] is the only practical alternative. Computationally, the major distinction between the various levels of approximation is the amount of information to be stored; fluids require thermodynamic quantities to be stored at sparse locations, while particle models

---

[1] This phenomena leads to so-called *flux-limited diffusion* models. We discuss some features of this type of model in later sections

essentially require the position and velocity of each member in the system. The fluid approximations therefore allow a much greater range external information to be incorporated, such as detailed geometry, at the expense of losing detailed information about the individual particle distribution.

The same type of hierarchy exists for modeling the transport of neutral particles. The coarsest level of approximation is the single group diffusion model, employing all the standard diffusion approximations[58] (no self-collisions, frequent background collisions, short mean free path). If the neutrals are in a regime where there are enough neutral-neutral collisions to establish thermal equilibrium, then the full navier-Stokes equations of gas dynamics are required. If there is little communication between neutral particles of various energies, a multi-group approach may be more appropriate[59]. If most of the neutrals travel through the system before they can establish a thermal equilibrium, even within their group, or if a large number of neutral species exist that need to be tracked, a Monte Carlo approach (c.f. [60]) (with a fixed background) is often employed. Some of the divertor concepts under current investigation (notably, the ultra-high recycling gaseous divertor [61]) operate in regimes where neutral-neutral collisions are dominant, but also where charge-exchanged neutrals may penetrate fully through the plasma. Here, all the above models fail on at least one count, and efforts then seek to uncover trends in the data that will be useful in predicting the behavior of the more complex system.

3.3 Literature Survey

With the above limitations at hand, we now describe the plasma models commonly used in studies of the edge plasma, which stem from the pioneering research of Grad[62]. Grad's contribution was a reformulation, for charged species, of the Chapman-Enskog statistical thermodynamics[63] originally applied only to neutral particles. Grad's transport coefficients were numerically evaluated by Braginskii[64] for the case of one charged ion species, and one electron species; the resulting conservation set are commonly referred to as the *Braginskii equations* in the edge modeling community. The *modified* Braginskii equations are formed by replacing the classical transport of plasma perpendicular to the magnetic field with arbitrary anomalous transport. The modified Braginskii equations are combined with a separate model for the neutral species and form a coupled set of nonlinear partial differential equations to be solved self-consistently.

The steady modified Braginskii equations, which will be presented in § 4, are quite complex, have no general analytic solutions, and must be solved numerically. Moreover, for a tractable numerical problem, the full set of modified Braginskii equations must be simplified in one and two dimensional analyses by invoking some judgment about the importance of the various edge effects. Often the number, and types, of these effects that can be successfully included in a solution technique, becomes a measure of a code's usefulness to the general community. As such, we concentrate here on the solution algorithms employed in the codes used by the various groups, pointing only to a *typical* example of the complexity achievable as a rough figure of merit. In doing so, we underscore that the purpose of our work is to improve the analysis tools available for the community, so that the community can contribute to improving the models. We outline three significant one-dimensional fluid models before comparing the present status of two-dimensional efforts.

One-dimensional models are often used for scoping studies (i.e. to determine the influence of various aspects of a model). A classic example was introduced by Harrison, et. al.[65, 66] to investigate recycling physics in the INTOR[67] reactor concept. In that study, transport is modeled only *along* the open magnetic field lines of the SOL, in "flux-tubes" between the magnetic flux surfaces. Radial transport of plasma affects only upstream power and particle flux boundary conditions. Recycled neutral particles are allowed to escape the system only if they traverse the plasma between the target strike point and an adjacent neutral pump throat. The neutrals are emitted with a cosine distribution from a grazing-incidence divertor strike plate so that a significant fraction are initially pointing toward the pump. By carefully accounting for duct and field line shapes, Harrison, et. al. were able to quantify how recycling and scrape-off lengths influenced overall plasma temperatures and heat flux to the divertor plates—ultimately leading to estimates of divertor material lifetimes as a function of particle exhaust rates. Such one-dimensional estimates guide later, more detailed investigations.

The state-of-the-art one dimensional fluid model was developed by Campbell[68], and is employed in the NEWT-1D code to model the SOL of the DIII-D tokamak and certain scenarios of the proposed ITER divertor. In the NEWT-1D code[2] Hydrogen plasma is modeled with the Braginskii set along the magnetic field. Radial transport is modeled as a volumetric sinks along the field, and drift flows are neglected entirely. The flux tube geometry of Harrison, et. al.[65] is used, but only

---

[2] The NEWT-1D code is apparently the outgrowth of research carried out by Knoll towards his doctoral dissertation[69]

to model the flux tube adjacent to the separatrix. Impurities (e.g. Argon, Beryllium etc.) are included in the system via corresponding continuity and momentum equations, under the assumption of thermal equilibrium with the hydrogen ions. Campbell uses a damped Newton algorithm (c.f [70]) to find the steady state solution to the large set of nonlinear algebraic equations arising from discretizing the fluid equations with a standard finite volume method[71]. Even in one dimension, the internal profiles of these plasma require excessive resolution, often preventing numerical convergence of the solution.

The primary goal of Campbell's work is to arrive at global estimates for the puffing/pumping rates and entrainment characteristics, and to estimate whether significant power reduction through recycling-related radiation would lead to core directed convection (i.e. *flow reversal*) of the hydrogen plasma (flow reversal would likely lead to contamination of the core plasma). Campbell's results thus far have shown that the power flux on the divertor target may indeed be reduced by puffing a large amount of neutral gas into the system near the target, attaining the so-called "gas-target" plasma regime[72]. However, if the gas is Hydrogen to avoid intentional contamination of the plasma, the puffing process tends always to lead to flow reversal. More recently, Campbell examined the introduction of Argon, Neon and Beryllium impurities to attain the benefit of the heavier elements' stronger atomic radiation rates and reduce the gas puffing requirements. Most of the solution profiles computed thus far again show the strong reversal features unless a large amount of impurities are introduced, which may contaminate the core plasma. At the levels that Campbell demonstrated necessary to significantly reduce the heat loads and temperatures at the divertor plate, the impurity species begin to affect the transport of the background plasma, invalidating the approximations used (cf. [73]) used in the underlying models.

Campbell's early 1D results were confirmed independently by Merriman, et.al.[74] in simple extensions to a model that successfully characterized high recycling regimes in the PISCES experiment[75]. Merriman et. al. have pointed out that the more demanding edge plasma conditions of ITER lead to small scale internal features that defy numerical resolution for the computational resources typically available. We discuss this point in Chapter 8. The fine-scale features are expected to persist in two-dimensional models. Merriman et. al. are presently implementing more accurate numerical techniques, and a more complete description of the neutral momentum transport to alleviate this problem. The issue of achievable accuracy under the more extreme divertor scenarios is critical in two spatial dimensions, where resolution is considerably more prohibitive than in one dimension. The work presented by Merriman et.al. has been the testbed for the dimension-independent components of the physics and numerical algorithm development of the present work.

The primary two-dimensional fluid edge plasma model in use today was developed by Braams[76], and is implemented in the code, b2. Features of b2 (in its most widely distributed form) include orthogonal curvilinear meshes, multiple ion species, diffusive cross field transport, and an "analytic" form for the neutral transport. The discretization is based on the SIMPLE method of Patanker[71], modified to allow compressibility. The allowable grids follow the multiply connected poloidal magnetic flux surfaces[77], possibly extending a small distance into the core region. More recent modifications include two-dimensional equations for the electric potential, the drift motion, a fluid neutral model that includes momentum effects[78], and a crude radiation transport module[79]. The newer version of the code is named b2.5. A still newer version, b3 is forthcoming, and will include self-consistent fluid neutral transport[80]. The b2/2.5 code is routinely used to analyze tokamak edge plasmas in the high-collisionality regime, and has successfully modeled flow reversal and impurity transport. The number of groups in the fusion community actively using the b2 code in their own research is overwhelming, and indicates that improvements to the usability, applicability or robustness of the algorithms would be gratefully received by a large established user base.

Success of the b2/2.5 model has encouraged development of a large number of similar efforts, each aimed at improving particular numerical or physical aspects of the code.

- EPIC, by Vold et. al.[18], implemented the b2 discretization, but generalized the allowable domain geometry by "stepping" the orthogonal grid to approximate misaligned boundaries. EPIC also approximated the electric field and drift flow component of velocity before such work was implemented in b2 and NEWEDGE–discussed below. In the reference, Vold, et.al. demonstrated strong effects of domain geometry and predicted equally strong influence of the electric potential in providing fluid drifts that alter the plasma solution.

- LEDGE[81] implemented a fully implicit time-stepping algorithm to solve the b2/2.5 equations more efficiently and robustly. The matrix-free Krylov-subspace iteration algorithm implemented in LEDGE allows the fully implicit inclusion of the electric field and drift flows, in addition to time accurate simulations. Fully implicit solution procedures dramatically improve the stability of time-marching algorithms[82]—this is discussed briefly in Chapter 9 as a possible fix to some of the numerical problems we experienced on a day-to-day basis in obtaining results for this dissertation.

- NEWEDGE[69] implemented a damped Newton's method to the set of equations that result from a mixed discretization containing terms from the Patanker scheme, as well as central and upwind contributions (all of them are first order convergent). More recently, the code was upgraded to include various impurity transport models akin to the that employed by Campbell. NEWEDGE runs efficiently on the CRAY vector computers and is robust enough to allow inclusion of a large number of additional transport equations. Knoll, et al.[83] have explored the two dimensional behavior of Argon, Beryllium and Neon injected edge plasmas also studied by Campbell. A significant limitation of NEWEDGE is that the Newton algorithm requires the storage and inversion of a large Jacobian matrix. The upper ceiling on problem size is set by the available memory of the computer. Knoll has, however, implemented a wide variety of efficiency-improving modifications to the standard Newton algorithm to greatly speed the solution on moderate sized problems[84]. Also, in contrast to the Krylov iteration methods, a wide variety of preconditioning methods are available.

- NEWEDGE-MG[85] implemented the damped Newton's method of NEWEDGE for only the coarsest allowable domain mesh. A simplified Multi-Grid approach[86] is then used to resolve the computation without forming or solving a Jacobian matrix, allowing a order of magnitude increase in resolution. Homotopy is used in nonlinear continuation from mild to difficult plasma regimes. Significantly though, the method requires a great deal of problem-specific fine tuning of the solution procedure. Because the various levels within the multigrid algorithm are solved only partially, the method is prone to divergence on strongly coupled plasmas. Also, the simplified implementation of multi-grid techniques prohibits accurate solutions on anything other than rectangular grids.

- UEDGE[87] is a recent project to concentrate the US modeling efforts. In UEDGE, the b2 discretization can currently be solved via the matrix-free Krylov time marching of LEDGE or the damped Newton's method of NEWEDGE. This large code is maintained and upgraded regularly for public access on the NERSC CRAY supercomputers. UEDGE is used as a test bed for new physics modules, such as a nonlocal electron heat flux[88, 89], and the coupling to one-dimensional core models, as well as for more routine analysis of experimental data on the D-III-D tokamak experiment[90]. The code provides a good central repository for proven schemes because the new design is geared around modularity of the important algorithmic steps. UEDGE presently implements the majority of the the physics modules and efficiency improvements developed for NEWEDGE.

- There are also a number of independent efforts to couple the semi-implicit algorithm of b2.5 to the DEGAS[60] and EIRENE[91] Monte-Carlo neutral transport codes (c.f. [92]). Complete Monte-Carlo particle trajectories are computed after a fixed number semi-implicit time steps of the b2.5 code, and the process is iterated to self-consistent steady state. While generally accepted as the most accurate of available schemes, this external coupling ap-

> proach is computationally very expensive and prone to temporal divergence—particularly when used in cases with high neutral particle densities, as expected in the ITER gas divertor concept. The methods have been applied with limited success to more mild plasma conditions, as reported in Ref. [93]. This too is an area of active research.

There are a number of alternative edge fluid codes available that are not built on the b2/2.5 discretization. Gerhauser, et. al.[94] use a code, SOLXY, to model the SOL of the TEXTOR experiment, a limited plasma. While the code assumes rectangular geometry, great emphasis was placed on the analysis of the coupling of $E \times B$ flows. A model was proposed therein for the viscous coupling of the drift velocity in the core plasma to the parallel flow at the limiter surface. However, a recent paper[95] casts some doubt on part of the detailed treatment of the boundary condition in SOLXY (and NEWEDGE). SOLXY is one of the two primary edge fluid modeling codes extensively used within the European community (the other is b2).

In the PLANET code[96], Petravic has implemented a standard alternating direction implicit scheme (ADI) to semi-implicitly advance the *modified* Braginskii equations in time to second order accuracy. Petravic has used PLANET to investigate a wide range of physical phenomena including drift flows[97] and ultra-high recycling divertors[98].

Ueda[99] reports a Particle-In-Cell (PIC) discretization that, in principle, is not limited to orthogonal coordinate systems. In the reference, the model has been applied to the DIII-D tokamak and employs the *modified* Braginskii equation set without electric potential or drift flow effects. Ueda has demonstrated the functionality of the PIC method for edge plasma, but has not yet applied the scheme to grids that do not align with the poloidal flux surfaces.

Vesey[100] presented a finite element (FE) approach to modeling the SOL. Vesey employed the streamline-upwind Petrov-Galerkin (SU/PG) method[101] to treat advection dominated flows while in a generalized geometry setting. A fully implicit Newton algorithm, similar to that employed in NEWEDGE, is used to solve the resulting nonlinear set of nonlinear algebraic equations. This FE implementation allows arbitrary trapezoidal domain decompositions, and therefore nonorthogonal domains of solution. Vesey has presented initial solutions that indicate accurate boundary treatments will result in significant dependence of plasma parameters on target plate angles. While Vold, et. al.[18], and Knoll[102] propose various "fixes" to an inherently orthogonal discretization, Vesey's FE implementation is not at all so restricted. Expansion of the work to reactor relevant plasmas is forthcoming.

Kuprat[103] employs a moving finite-element (MFE) [104] method to automatically refine regions of the domain that lack resolution. The scheme employs triangular elements, general connectivity, and uses a Riemannian metric to generate a modified Dulauney triangulation tailored for highly anisotropic edge plasma flow. Further, "graph-massage" is used to insert or delete nodes to maintain a constant resolution accuracy when measured on the solution manifold in the Riemannian units. Further work in this area will be necessary before benchmark plasmas can be modeled, however the scheme shows a great deal of promise in providing a general way to maximize solution accuracy given a fixed number of nodes (from memory limits, say). This appears to be the only current work utilizing dynamically adaptive gridding methods.

Traditionally have been four annual meetings held on a national level, where researchers in the area of edge physics and modeling present periodic summaries of the status of their work: “Plasma-Surface Interactions in Controlled Fusion Devices”, the “Sherwood Conference on Plasma Theory and Applications” and the “Tokamak Edge Modelling Workshop”, and the somewhat more broad annual meeting of the American Physical Society, Division of Plasma Physics.

Presently, the major impetus for further development in this area has been the uncertainties surrounding the ITER divertor design, and its impact over the entire ITER device[7]. The nonlinearities of plasma fluid equations are exacerbated in the ITER parameter regime—so much so, in fact, that the simple equation solvers present all the presently available codes routinely fail to find solutions in many of the interesting design ranges. In the fusion community, there is thus a large demand for improved numerical techniques, both in accuracy and solution method robustness, to enable further probing of new divertor plasma operating regimes.

Conclusions

Though the above summary surely neglects a few important efforts worldwide; the message is that there appears to be a great many *independent* efforts devoted toward investigating the tokamak edge divertor problem. It is valid to ask whether an additional effort is likely to make a significant contribution.

There are a few common threads to the research described above that should not be duplicated once again. The first regards the discretizations; relatively little effort has been devoted to the discretization of the modeling equations, despite their rather unique and complex structure (e.g. *strongly* anisotropic diffusion). Some effort should be devoted toward studying the various resulting implications. A second common thread is that there has been no attempt to develop a computer modeling code that is accessible to a wide variety of physicists and engineers, in the sense of modern distributable analysis codes. Most codes listed in the previous section, and in the review references cited at the beginning of the chapter, were developed as a single part of one student's doctoral dissertation, and subsequently extended over some years of research. The situation give rise to specialized codes not generally intended for the research community at large; this seems a great waste of effort. Finally, there has been considerable variation in the actual equations used to represent the fluid plasma edge, even under identical problem descriptions.

The present studies were aimed at these three issues issues, rather than at the development of a particular model or solution technique.

Chapter 4

Fluid Plasma Equations

In this chapter, we attempt to remedy a problem that exists in the tokamak edge modeling community regarding determination of the 'correct' equations that express energy and momentum transport. As an example, the plasma fluid heat equation (to be derived) is written

$$\frac{\partial}{\partial t}\left(\frac{3}{2}p\right) + \nabla \cdot \left(\frac{5}{2}p\vec{u} + \vec{q}\right) = \mathbf{\Pi} : \nabla\vec{u} + \vec{u} \cdot \nabla p + \frac{1}{2}m\vec{u}^2 S_n - \vec{u} \cdot \vec{S}_p + S_E$$

where the terms have their usual definitions—the S terms are sources of particles momentum and energy due to non-Coulomb processes, $\mathbf{\Pi}$ is the viscous stress tensor, p is the scalar pressure, $\vec{q}$ is the heat flux and $\vec{u}$ is the velocity. This equation, and its total-energy equivalent, appears in many references, such as References [61, 18, 76, 93, 69, 100, 105] as the basis for predictive tokamak edge plasma computations. Most often, this equation will appear in coordinate-dependent form, expressed in terms of the poloidal and radial components in the poloidal plane.

In some cases found in the references, one or more of the last three terms appears in a slightly modified form or is dropped from the equation entirely, while in other cases, some of the coordinate transformation terms appear to be missing. In fact, most of the discrepancies between our work and that of those cited result from application of additional uncited or unexplained physical constraints. The purpose of this chapter is to provide a complete derivation of the applicable equations, including a description of the assumptions that are involved in the process. A general principle is then offered to use in inserting new 'anomalous' processes in an unambiguous way. In § 4.6, we discuss a more subtle point in resolving various models.

We begin this chapter by developing the conservation equations for a standard Newtonian fluid with volumetric sources. We use the insight gained along the way to ascribe physical significance to the various terms arising from the kinetic approach to deriving the fluid equations. We obtain the complete set of equations used for plasma and neutral transport, appropriately transformed into the poloidal plane.

4.1 Newtonian Kinematics

In order to identify the precise physical significance of the various source terms appearing in the edge plasma fluid equation set, we can relate them to those appearing in Newton's laws of motion. Note however, that most textbooks on elementary fluid dynamics ignore the case with volumetric fluid sources. The extension for the fundamental equations is trivial, but will lead to not-so-trivial modifications for derived conservation laws, such as the often-used pressure (or, heat) equation.

In the following, assume we have a small blob of fluid, Ω_t with surface, $\partial\Omega_t$ moving in time at the prevailing fluid velocity, $\vec{u}$. We would like to write the conservation laws for this blob (i.e. the expressions for conservation of mass, momentum and total energy).

"First Principles" for our system:

1. The mass of the blob will change in response to a volumetric mass source rate. The volume of the blob is defined by its surface, which moves at the prevailing velocity.
2. The total linear momentum of the blob will change in response to applied forces, as well as to any internal momentum generation.
3. The total energy increases due to work done by the external system on the blob and by internal heat generation. Energy is lost via conduction through its surface.

These statements are relevant only in the frame moving with the mass, i.e. Newtonian kinematics are valid only in a Lagrangian coordinate system. Mathematical representation of these expressions will involve the fluid velocity and spatial gradients, which we can only obtain in the Eulerian (fixed) frame. The rate of change of the fundamental quantities while moving with the fluid, measured in fixed coordinates, is expressed via the *convective derivative*, $D/Dt = \partial/\partial t + \vec{u} \cdot \nabla$. The operator $\mathrm{d}/\mathrm{d}t$ conversely, measures the rate of change measured in the moving frame.

A well known result in fluid dynamics is "Reynold's Transport Theorem"[106] (or the convection theorem[107]), which expresses the rate of change of a scalar or vector quantity, f, associated with a fluid blob as its surface moves through space with velocity $\vec{u}$,

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega_t} f dV = \int_{\Omega_t} \left(\frac{Df}{Dt} + f \nabla \cdot \vec{u} \right) dV$$

dV is a differential parcel of the small blob volume such that $\int_{\Omega_t} dV = \Omega_t$. The time rate of change of f in Ω_t should be equal to its volumetric source rate.

In terms of the mass density, ρ, the first principle above can be expressed as

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega_t} \rho dV = \int_{\Omega_t} S_m dV,$$

where the mass of the blob is $m = \int_{\Omega_t} \rho dV$ and S_m is the volumetric mass source rate. Applying Reynold's Transport theorem,

$$\int_{\Omega_t} \left(\frac{D\rho}{Dt} + \rho\nabla\cdot\vec{u}\right) dV = \int_{\Omega_t} S_m dV$$

And since this must hold for any volume, the integrands will be equal if appropriate smoothness conditions apply[1], yielding

$$\frac{D\rho}{Dt} + \rho\nabla\cdot\vec{u} = S_m$$

By expanding the convective derivative, the left hand side can be written in *conservation form*,

$$\frac{\partial\rho}{\partial t} + \nabla\cdot\rho\vec{u} = S_m \tag{4.1}$$

This and the previous equation represent alternative differential forms of the equation of *continuity* for the case where volumetric mass sources are present.

The total linear momentum in the i-direction is computed as the integral of the momentum density, ρu_i, over the volume of the blob. Principle 2 can therefore be expressed as

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega_t} \rho u_i dV = \int_{\Omega_t} \rho f_i dV + \int_{\partial\Omega_t} P_{ij} dS_j + \int_{\Omega_t} S_{pi} dV \tag{4.2}$$

---

[1] For Reynold's transport theorem to hold, the transported quantity must be C^1 differentiable[107]. Also, $\partial\Omega_t$ is to be an orientable union of *regular* surfaces, as defined by Kellogg[108]

where dS_j is the j-component of the outward pointing normal to a differential section of the blob boundary, $\partial\Omega_t$. P_{ij} is the stress tensor, f_i is the i-component of an external force, $\vec{F}$, and S_{pi} is the i-component of the volumetric generation rate of momentum, $\vec{S}_p$. Note that force may be exerted on the blob both by a stress acting on its surface and by a body force.

The momentum source term, $\vec{S}_p$, deserves special comment. In the tokamak edge plasma, $\vec{S}_p$ arises typically from atomic processes, such as charge exchange, which deposit a local parcel of momentum in the fluid. The processes do not contribute necessarily to a corresponding change in the *bulk* momentum density. The component of $\vec{S}_p$ that does act as a body force, will appear in the momentum balance formally analogous to the other body forces, such as gravity or electric field, and will lead to a component of the energy source that looks formally identical to the other "work"-related energy source terms. These effects will be captured self-consistently with an appropriate kinetic collision term in Boltzmann-type analyses, as discussed in the next section (an example is the frictional force and heating terms, in the Braginskii equations, that are due to collisional Coulomb interactions between ions and electrons). For now, it suffices to think of $\vec{S}_p$ as an *unspecified* momentum source which may or may not contain a component which acts on the fluid as a body force.

Proceeding, the divergence theorem can be used to convert the surface integral in Equation 4.2 to a volume integral. Noting again that the expression must hold for all parts of the volume, the integral signs can be removed formally. Switching to vector notation

$$\frac{D\rho\vec{u}}{Dt} + \rho\vec{u}\nabla\cdot\vec{u} = \rho\vec{F} + \nabla\cdot\mathbf{P} + \vec{S}_p$$

Expanding the convective derivative, this can be written as

$$\frac{\partial \rho \vec{u}}{\partial t} + \nabla \cdot (\rho \vec{u}\vec{u} - \mathbf{P}) = \rho \vec{F} + \vec{S}_p \tag{4.3}$$

These are equivalent differential forms to express *conservation of momentum.*

Finally, the total energy of the blob per unit mass, E, for a simple fluid is (to an additive constant) composed of kinetic and internal components, $E = \frac{1}{2}\vec{u}^2 + e$. This quantity will increase due to work done on the blob on by external system via surface and body forces. Additionally, the total energy will increase due to any volumetric heat generation, S_E (which, again, may contain contributions in the form of work done by a body-type force). Subtracting out the integrated heat flux, $\vec{q}$, that passes through the surface, the time rate of change of total energy contained in the blob is

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega_t} \rho E dV = \int_{\Omega_t} \rho \mathbf{F} \cdot \vec{u} dV + \int_{\partial\Omega_t} P_{ij} u_j dS_j + \int_{\Omega_t} S_E dV - \int_{\partial\Omega_t} q_j dS_j$$

in accordance with our third fundamental principle. As before, the surface integrals may be transformed via the divergence theorem, and the integral signs removed formally, to give

$$\frac{D\rho E}{Dt} + \rho E \nabla \cdot \vec{u} + \nabla \cdot \vec{q} = \rho \vec{F} \cdot \vec{u} + \nabla \cdot (\vec{u} \cdot \mathbf{P}) + S_E$$

or

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot (\rho E \vec{u} + \vec{q} - \vec{u} \cdot \mathbf{P}) = \rho \vec{F} \cdot \vec{u} + S_E \tag{4.4}$$

The last two equations are alternate forms of the differential expression of *conservation of energy.*

With the above equations, we therefore identify S_m to be the volumetric mass production rate, as seen by a volume of fluid moving with the prevailing velocity. $\vec{S}_p$

and S_E are respectively the volumetric momentum and energy production rates. In the edge plasma, $S_m = mS_n$, where m is the total mass of an ion-electron pair, and S_n is the net volumetric production rate of ion-electron pairs due to ionization and recombination processes. $\vec{S}_p$ consists of any momentum brought into the system during the creation of the ion-electron pairs, along with net momentum transferred in or out of the ion channel via charge exchange. Finally, S_E consists of any direct energy loss or gain due to the atomic processes, along with a gain of any energy brought in with the creation of new ion-electron pairs (i.e. if the new particles are brought in with a nonzero average velocity or temperature). Due to the complex nature of the atomic processes, the majority of these source terms are evaluated empirically, using data compiled from a great many experiments and tabulated into numerical tables.

4.1.1 Pressure/Heat Equation

In practice, the force terms arising in the momentum and energy equations are difficult to obtain explicitly. The electric field is affected by non-ambipolar currents in the plasma, and therefore may become part of the unknown state vector. We can simplify the energy equation, using continuity and momentum conservation to eliminate the "work" term. As a side-effect, the kinetic energy term will be eliminated from the time derivative, and the resulting equation will be for the evolution of the *internal* energy density. We carry out this procedure explicitly by adding the momentum equation scaled by $-u_i$, and the continuity equation scaled by $+\frac{1}{2}\vec{u}^2$ to the total energy equation. Referring to the convective forms of

Equations 4.1 and 4.3, one can easily verify that

$$\frac{D\rho e}{Dt} + e\rho\nabla \cdot \vec{u} + \nabla \cdot \vec{q} = \frac{1}{2}\vec{u}^2 S_m - \vec{S}_p \cdot \vec{u} + S_E + \mathbf{P} : \nabla\vec{u} \tag{4.5}$$

provided use is made of the following three identities

$$\frac{1}{2}u_i^2 \frac{D\rho}{Dt} - u_i \frac{D\left(\rho u_i\right)}{Dt} = -\frac{D\left(\frac{1}{2}\rho u_i^2\right)}{Dt}; \qquad e = E - \frac{1}{2}\vec{u}^2$$

$$\nabla \cdot (\vec{u} \cdot \mathbf{P}) - \vec{u} \cdot \nabla \cdot \mathbf{P} = \mathbf{P} : \nabla\vec{u}$$

Note that the last identity follows only if the stress tensor is symmetric. There has been some confusion in the literature related to this point which is discussed in the the final section of this chapter. In the above fluid heat equation, e is the specific internal energy. Equation 4.5 is the so-called pressure, or heat equation. A key feature is the inclusion of the mass and momentum source rates, which appear from the algebraic manipulations that removed the kinetic energy from the left hand side.

We proceed to evaluate the same set of equations from the point of view of kinetic theory. The end result will be quantitatively similar to the above conservation set, Equations 4.1, 4.3 and 4.4, so we should be able to determine exactly the physical nature of the source terms that arise.

4.2 Moment Approach to Fluid Equations

As a first approximation, the neutral and impurity processes mentioned earlier will play a negligible role on the transport characteristics of the background plasma. In this case, the stress tensor, heat flux, volumetric friction forces and viscous energy dissipation terms can be evaluated by accounting only for the interactions between each of the plasma species and between the plasma and ambient magnetic field. However, even in this approximation, the actual evaluation of the collisional transport terms is a tedious calculation. The procedure involves computing velocity moments of the Coulomb collision integral, and was first carried out for a rarefied plasma by S. I. Braginskii[64] using the expansion techniques that Chapman and Cowling[63] applied to neutral gases. Recently, the Braginskii analysis was extended by Helander, et. al.[80] to additionally include transport effects due to charge exchange with a background neutral gas. Ionization and recombination have not yet been included consistently in the transport calculations, and so are presently added to the existing model via *ad hoc* source terms.

Here, we sketch the starting point for the calculations, review the published results, and then discuss the primary assumptions invoked throughout the derivation. A simplified set of modeling equations is presented subsequently that accounts, at some level, for the neglected source terms and violated physics. We use the above Newtonian kinematical arguments to guide us in adding the applicable source terms to the derived plasma conservation set so that we may implement correctly a given physical model of the atomic physics processes.

Proceeding, the "*stosszahlansatz*" used by Boltzmann (cf. [109]), is that the states of any two molecules in our system can be de-correlated statistically. This allows one to write an expression for the time rate of change of $f(\vec{x}, \vec{u}, t)$, the probability density for any molecule in the system to have its the state lay within the region $[\vec{x} + d^3x] \times [\vec{v} + d^3\vec{v}]$ in phase space at time t in terms of known forces. Here, the relevant forces are quasistatic electromagnetic fields; the appropriate kinetic equation is

$$\frac{\mathrm{d}f}{\mathrm{d}t} = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial \vec{x}} \cdot \vec{v} + \frac{q}{m}\left[\vec{E} + \frac{1}{c}\vec{v} \times \vec{B}\right] \cdot \frac{\partial f}{\partial \vec{v}} = C$$

where C, the collision integral, represents the probability rate density that particles will enter that state in phase space, including simple scattering and all other sources. Because the acceleration (or force) is divergence-free, this equation can be written in conservation form

$$\frac{\partial f}{\partial t} + \frac{\partial}{\partial \vec{x}} \cdot \vec{v} f + \frac{\partial}{\partial \vec{v}} \cdot \frac{q}{m} \left[\vec{E} + \frac{1}{c} \vec{v} \times \vec{B} \right] f = C$$

Next, velocity-space moments of this kinetic equation are taken to obtain moment conservation equations analogous to those derived in the preceding section[2].

The lowest order velocity moment, $\int d^3\vec{v}$, is most simply computed using the conservative form of the kinetic equation. We define $n = \int d^3\vec{v} f$ (the particle density) and $n\vec{u} = \int d^3\vec{v}\vec{v} f$ (the particle flux, $\vec{u}$ the fluid velocity) to help simplify the equation

$$\int d^3\vec{v} \left(\frac{\partial f}{\partial t} + \frac{\partial}{\partial \vec{x}} \cdot \vec{v} f + \frac{\partial}{\partial \vec{v}} \cdot \frac{q}{m} \left[\vec{E} + \frac{1}{c} \vec{v} \times \vec{B} \right] f = C \right)$$

Since $f(v \to \infty) \Rightarrow 0$ (i.e. there is zero probability that any particles exist at the extreme values of velocity) and $\int d^3\vec{v}\, \partial/\partial\vec{v} \cdot q/m \left[\vec{E} + \frac{1}{c}\vec{v} \times \vec{B} \right] f = 0$ by the divergence theorem, the last two terms on the left hand side vanish, and our definitions give

$$\frac{\partial n}{\partial t} + \nabla \cdot n\vec{u} = \int d^3\vec{v} C \tag{4.6}$$

For constant mass, this would be the conservative form of our continuity equation, Equation 4.1, if we identify

$$S_m = m \int d^3\vec{v} C$$

---

[2] Evaluation of the transport coefficients is a complex issue not presented here; there are ample references on the matter, including the seminal paper by Braginskii[64].

m is the mass of the individual particles described by the distribution function, f.

We can similarly calculate the momentum and energy moments of the kinetic equation in conservative form. But first, to simplify the notation, we will identify the "random" component of velocity, $\vec{v}\,' = \vec{u} - \vec{v}$, such that $\int d^3\vec{v}\,'\vec{v}\,'f = 0$, $d\vec{v} = d\vec{v}\,'$. Additionally, the heat flux, $\vec{q}$, scalar pressure, p, total pressure tensor, $\mathbf{P}$, and viscous stress tensor, $\mathbf{\Pi}$, are defined as follows

$$\vec{q} = \int d^3\vec{v}\,'m\frac{\vec{v}\,'^2}{2}\vec{v}\,'f$$

$$p = \int d^3\vec{v}\,'m\frac{\vec{v}\,'^2}{3}f = \int d^3\vec{v}\,'m\frac{(\vec{v}-\vec{u})^2}{3}f$$

$$\mathbf{P} = -\int d^3\vec{v}\,'m\vec{v}\,'\vec{v}\,'f = -p\mathbf{I} + \mathbf{\Pi}$$

$$\mathbf{\Pi} = -\int d^3\vec{v}\,'m\left[\vec{v}\,'\vec{v}\,' - \frac{\vec{v}\,'^2}{3}\mathbf{I}\right]f$$

where $\mathbf{I}$ is the identity tensor. Note that the sign convention used here for the total pressure (stress tensor) is opposite to that quoted in the NRL Formulary, but is instead consistent with the bulk of literature in mathematical fluid dynamics. Proceeding, the first moment, $\int d^3\vec{v}m\vec{v}$, of the kinetic equation will lead to the differential form of the equation for conservation of momentum. The time derivative term

$$\int d^3\vec{v}m\vec{v}\frac{\partial f}{\partial t} = \frac{\partial}{\partial t}\int d^3\vec{v}m\vec{v} = \frac{\partial(mn\vec{u})}{\partial t}$$

by virtue of the independence of t and $\vec{v}$, the definition of $\vec{u}$, and that f is zero at the limits of integration. The spatial gradient term is rewritten in terms of the flow and random velocity components, the integrated variable is changed from $d^3\vec{v}$ to $d^3\vec{v}\,'$, and the result is simplified via substitution of the definitions for $\mathbf{P}$ and $\vec{u}$:

$$\begin{aligned}\int d^3\vec{v}m\vec{v}\frac{\partial}{\partial\vec{x}}\cdot\vec{v}f &= \frac{\partial}{\partial\vec{x}}\cdot\int d^3\vec{v}m\vec{v}\vec{v}f = \frac{\partial}{\partial\vec{x}}\cdot\int d^3\vec{v}\,'m\left(\vec{u}\vec{u} + 2\vec{u}\vec{v}\,' + \vec{v}\,'\vec{v}\,'\right)\\ &= \nabla\cdot\left(mn\vec{u}\vec{u} - \mathbf{P}\right)\end{aligned}$$

The final step is allowed because the flow velocity, $\vec{u}$, is independent of the random velocity, $\vec{v}\,'$, so it can be pulled from under the integral. The acceleration term is simplified as follows

$$\begin{aligned}
&\int d^3\vec{v} m\vec{v}\frac{\partial}{\partial \vec{v}}\cdot\frac{q}{m}\left[\vec{E}+\frac{1}{c}\vec{v}\times\vec{B}\right]f \\
&\quad=\int d^3\vec{v}\left(\frac{\partial}{\partial \vec{v}}\cdot m\vec{v}\frac{q}{m}\left[\vec{E}+\frac{1}{c}\vec{v}\times\vec{B}\right]f-\left[\frac{\partial}{\partial \vec{v}}\cdot m\vec{v}\right]\cdot\frac{q}{m}\left[\vec{E}+\frac{1}{c}\vec{v}\times\vec{B}\right]f\right) \\
&\quad=-q\int d^3\vec{v}\left[\vec{E}+\frac{1}{c}\vec{v}\times\vec{B}\right]f=-nq\left[\vec{E}+\frac{1}{c}\vec{u}\times\vec{B}\right]
\end{aligned}$$

Where the identity, $\partial/\partial\vec{v}\cdot\vec{v}=\mathbf{I}$, has been observed. Combining terms, we obtain for the momentum balance

$$\frac{\partial\left(mn\vec{u}\right)}{\partial t}+\nabla\cdot\left(mn\vec{u}\vec{u}-\mathbf{P}\right)=nq\left[\vec{E}+\frac{1}{c}\vec{v}\times\vec{B}\right]+\int d^3\vec{v}m\vec{v}C \tag{4.7}$$

and therefore, owing to Equation 4.3

$$\vec{S}_p=\int d^3\vec{v}m\vec{v}C$$

In a similar fashion, the time derivative term of the energy moment equation is

$$\int d^3\vec{v}\frac{m\vec{v}^2}{2}\frac{\partial f}{\partial t}=\frac{\partial}{\partial t}\int d^3\vec{v}\frac{m\vec{v}^2}{2}f=\frac{\partial}{\partial t}\int d^3\vec{v}\,'\frac{m}{2}\left[\left(\vec{v}\,'+\vec{u}\right)\cdot\left(\vec{v}\,'+\vec{u}\right)\right]f$$

After changing the variable of integration to $\vec{v}\,'$, and expanding the dot product,

$$\int d^3\vec{v}\frac{m\vec{v}^2}{2}\frac{\partial f}{\partial t}=\frac{\partial}{\partial t}\int d^3\vec{v}\frac{m\vec{v}^2}{2}f=\frac{\partial}{\partial t}\int d^3\vec{v}\,'\frac{m}{2}\left[\frac{m\vec{v}\,'^2}{2}f+m\vec{v}\,'\cdot\vec{u}f+\frac{1}{2}m\vec{u}^2\right]$$

The first term is essentially the time derivative the scalar pressure, p. The second integral vanishes identically, while the third is just the time rate of change of the bulk kinetic energy, $\frac{1}{2}nm\vec{u}^2$. Thus

$$\int d^3\vec{v}\frac{m\vec{v}^2}{2}\frac{\partial f}{\partial t}=\frac{\partial}{\partial t}\left(\frac{3}{2}p+\frac{1}{2}nm\vec{u}^2\right)$$

Similarly, the velocity in the *spatial* derivative term is split into its bulk flow and random components. Using the definition for scalar pressure and manipulations similar those used in the continuity and momentum equations, the spatial derivative term of the energy moment of the kinetic equation can be expressed as

$$
\begin{aligned}
&\int d^3\vec{v}\frac{m\vec{v}^2}{2}\frac{\partial}{\partial\vec{x}}\cdot\vec{v}f \\
&\quad= \int d^3\vec{v}\,'\frac{1}{2}m\left[\vec{u}^2+2\vec{u}\cdot\vec{v}\,'+\vec{v}\,'^2\right]\nabla\cdot\left[\left(\vec{u}+\vec{v}\,'\right)f\right] \\
&\quad= \nabla\cdot\int d^3\vec{v}\,'m\left[\frac{1}{2}\vec{u}^2\vec{u}f+\frac{1}{2}\vec{u}^2\vec{v}\,'f+\vec{u}\cdot\vec{v}\,'\vec{u}f+\vec{u}\cdot\vec{v}\,'\vec{v}\,'f+\frac{1}{2}\vec{v}\,'^2\vec{u}f+\frac{1}{2}\vec{v}\,'^2\vec{v}\,'f\right] \\
&\quad= \nabla\cdot\left(\frac{1}{2}mn\vec{u}^2\vec{u}-\vec{u}\cdot\mathbf{P}+\frac{3}{2}p\vec{u}+\vec{q}\right)
\end{aligned}
$$

The initial manipulation and subsequent dropping of the first and third terms that come from expanding the vector products, are due to the integration over velocity space. Finally, noting that $\vec{u}\cdot\vec{u}\times\vec{B}=0$, the acceleration term is simplified

$$\int d^3\vec{v}\frac{m\vec{v}}{2}\frac{\partial}{\partial\vec{v}}\cdot\frac{q}{m}\left[\vec{E}+\frac{1}{c}\vec{v}\times\vec{B}\right]f=-nq\vec{E}\cdot\vec{u}$$

Putting the terms together, the energy moment of the kinetic equation is

$$
\begin{aligned}
&\frac{\partial}{\partial t}\left(\frac{3}{2}p+\frac{1}{2}mn\vec{u}^2\right)+ \\
&\qquad\nabla\cdot\left[\left(\frac{3}{2}p+\frac{1}{2}nm\vec{u}^2\right)\vec{u}+\vec{q}-\vec{u}\cdot\mathbf{P}\right]=nq\vec{E}\cdot\vec{u}+\int d^3\vec{v}\frac{m\vec{v}^2}{2}C
\end{aligned}
\tag{4.8}
$$

Now, since the Lorentz force is orthogonal to the velocity, and therefore does no work on our fluid, we can compare directly these results to those of the generalized derivation presented earlier, Equation 4.4. Specifically, we identify the total energy source rate

$$S_E=\int d^3\vec{v}\frac{m\vec{v}^2}{2}C$$

provided $\rho E=\frac{3}{2}p$. We obtain the heat equation here in the same way as before, by adding $\frac{1}{2}m\vec{u}^2$ times the continuity equation, and $-\vec{u}\cdot$ with the momentum balance, Equations 4.6 and 4.7 respectively, to the total energy, Equation 4.9:

$$\begin{aligned} &\frac{\partial}{\partial t}\left(\frac{3}{2}p\right) + \nabla\cdot\left(\frac{3}{2}p\vec{u} + \vec{q}\right) \\ &\quad = \mathbf{P} : \nabla\vec{u} + \frac{1}{2}m\vec{u}\int d^3\vec{v}C - \vec{u}\cdot\int d^3\vec{v}m\vec{v}C + \int d^3\vec{v}\frac{m\vec{v}^2}{2}C \end{aligned} \tag{4.9}$$

The heat equation obtained from kinetic theory agrees with the one we derived from “first principles”. Our interpretation of the source terms arising from the kinetic analyses are therefore consistent with the picture developed from Newtonian kinematics. In particular, the heat equation should include explicit continuity and momentum sources, if they exist, analogous to those appearing in Equation 4.5. Further, the energy source, S_E, will contain all forms of energy brought into the system by our atomic processes, including *both* kinetic and internal energy a new particle may bring in upon creation. The “extra” terms in the heat equation arise from algebraic manipulations done to remove the kinetic energy and work terms from the balance. A final, slightly modified form of Equation 4.10, appears more often in the literature, and is obtained by splitting out the scalar pressure and viscous stress. If, additionally, the source functions are written in the simpler notation, we obtain

$$\frac{\partial}{\partial t}\left(\frac{3}{2}p\right) + \nabla\cdot\left(\frac{5}{2}p\vec{u} + \vec{q}\right) = \mathbf{\Pi} : \nabla\vec{u} + \vec{u}\cdot\nabla p + \frac{1}{2}m\vec{u}^2 S_n - \vec{u}\cdot\vec{S}_p + S_E \tag{4.10}$$

4.3 The Braginskii Equations

In a plasma, we may assume that the dominant collisions occurring are due to the Coulomb electrostatic force, and that the plasma is relaxing to a drifting Maxwellian velocity distribution. Applying these assumptions, one can evaluate the integrals in Equation 4.10. The procedure is a bit tedious, but was carried out by Braginskii[64].

The Braginskii expressions are valid formally under the following conditions:

1. One species of singly charged ions and one group of electrons occupy the system. There are no volumetric particle or energy sources, and the only momentum imparted to the system is via the electric and Lorentz forces,
2. The distribution functions change slowly with respect to the the Coulomb collision rate,
3. The parallel scale lengths of profile inhomogeneities are much longer than the collisional mean free path,
4. The perpendicular scale length is much longer than perpendicular diffusion step sizes,
5. The Coulomb logarithm is large, or equivalently, "hard" collisions resulting in $> 90°$collisions are rare,
6. The electron gyro-radius is much longer than the Debye length,
7. The flow and stress distortions on the distribution functions are small, which is assured usually if the flow velocity, $\|\vec{u}\| \leq v_{th}/3$, where v_{th} is the thermal velocity, and
8. Anomalous transport is negligible

Unfortunately, the conditions listed above for validity of the Braginskii equations are almost always violated in tokamak edge plasmas to some extent. Typically then, an *ad hoc* model, if one exists, replaces the terms in the equations that are known to have problems. Computationally tractable simplifications are often proposed to model the edge plasma phenomena, and are then evaluated numerically for comparison to experiment. The following is a list of the most significant of the violations. In the next section, the resulting simplified model is subsequently presented.

- There is a strong influence by neutral and impurity particles in the system. These particles will contribute source terms to each of the three conserved quantities, in addition to influencing the plasma transport coefficients. The form of the new source terms can be guided by the physical insight gained from comparing the kinetic and Newtonian derivations.

- Time dependence of our system is herein ignored for numerical tractability, and we search "directly" for steady state solution profiles (time scales do however influence solution methods used for the numerical system of equations that represent discrete forms of the flow laws).

- The parallel scale lengths often are substantially longer than the local collision mean free path. For the heat equation, this condition leads ultimately to heat transport not well modeled by the small number of terms retained in the Laguerre expansion. In practice, the Braginskii form, which was derived under assumptions to the contrary, often leads to computed heat fluxes that violate causality by being greater than what can be carried physically by the underlying particles of the system. The subject is explored in further depth within results-oriented sections of this dissertation.

- The condition of small bulk flow velocity is always violated at the locations where the plasma contacts a material surface. There, an electrostatic "Bohm" sheath is formed automatically to preserve plasma quasineutrality. The situation is summarized elsewhere, and it suffices to mention that the fluid equations can be suitably altered to simulate such a boundary, in terms of its influence on the bulk plasma.

- Finally, laboratory plasmas are almost always dominated by anomalous transport processes. Modification of the fluid equations for this situation was proposed by Vold, et. al.[18]. In the present work, a greatly reduced set of equations has been employed.

The results of the Braginskii analysis are summarized in Appendix A, and appear in the standard reference NRL Plasma Formulary[54]. The analysis neglects collisions with neutral particles, and needs to be extended for our use based on results of the previous sections. For direct comparison, Equations 4.6, 4.7 and 4.10 are rewritten in terms of the convective derivative, explicit Coulomb source terms, and any remaining unspecified sources.

$$\begin{aligned}
\frac{Dn}{Dt} &+ n\nabla\cdot\vec{u} = S_n && (4.11)\\
mn\frac{D\vec{u}}{Dt} &= -\nabla p + \nabla\cdot\mathbf{\Pi} + nq\left[\vec{E} + \frac{1}{c}\vec{u}\times\vec{B}\right] && (4.12)\\
&\quad +\vec{R} + \vec{S}_p - m\vec{u}S_n\\
\frac{3}{2}n\frac{DkT}{Dt} &+ p\nabla\cdot\vec{u} = -\nabla\cdot\vec{q} + \mathbf{\Pi} : \nabla\vec{u} + Q && (4.13)\\
&\quad + \left(\frac{1}{2}m\vec{u}^2 - \frac{3}{2}kT\right)S_n - \vec{u}\cdot\vec{S}_p + S_E
\end{aligned}$$

In our notation, $\mathbf{\Pi} = -\mathbf{P}$ of Braginskii. The particle, momentum and energy source terms on the right hand sides of these equations are due usually to plasma collisions with neutral particles and impurity species. Notice that the form of these terms is not obvious without the analysis of the preceding sections.

4.4 Simplifying Assumptions

The full Braginskii model is posed in three spatial dimensions and contains at least 10 independent variables for three velocity components, density and temperature (if $p = nkT$ is applied) for both of the two charged species present. For more than one species of ions, the NRL equations would need to be replaced by a more general (and larger) set. Additionally, important effects, such as recycling and charge exchange depend critically on background neutral gases. In fact the neutrals are themselves strongly affected by these interaction processes, and need to be included consistently with their own set of conservation equations (Reference [18] provides discussion on this point). As a result, the complete set of classical modeling equations, which contains all these processes is quite large and complex.

In this section, we give a list of assumptions that are used to strip down the full Braginskii model to arrive at a simpler set of equations. The result is representative of typical coupled edge fluid models to date. To these equations, we need to add the appropriate neutral processes in accordance with the intuition developed in the preceding sections. We note that this entire process certainly is not unique by any means. The key challenge in tokamak edge fluid plasma modeling has been to select a model for the situation that is sufficiently accessible for analysis, yet reproduces experimentally measured results to a satisfactory extent.

Parallel Braginskii Equations and Anomalous Fluxes

The basic experimental observation for edge plasmas is that the Braginskii model of transport in the both radial and diamagnetic directions under-predicts the outward heat and mass fluxes observed in existing plasma devices. Typically, the radial particle and heat fluxes are replaced by empirical relations incorporating experimental observation or theoretical expressions based on turbulent diffusion due to various microinstabilities. Most of these can be expressed in the following form[76, 110]

$$\begin{aligned}
\Gamma_{\perp i} &= n^i v_\perp - D_\perp \nabla_\perp n^i - D_\perp^{T^e} \frac{n^i}{n^e} \nabla_\perp T^e \\
\mathbf{r} \cdot \mathbf{\Pi}_{\|\perp} &= m^i n^i v_\| v_{\perp v} - m^i n^i \eta_\perp^{ei} \nabla_\perp v_\| \\
q_{e\perp} &= v_{\perp T^e} n^e T^e - \chi_\perp^e n^e \nabla_\perp \left(kT^e\right) - \chi_\perp^{en} T^e \nabla_\perp n^e \\
q_{i\perp} &= v_{\perp T^i} n^i T^i - \chi_\perp^i n^i \nabla_\perp \left(kT^i\right) - \chi_\perp^{in} T^i \nabla_\perp n^i
\end{aligned} \tag{4.14}$$

We found no significant experimental or theoretical base for the anomalous momentum flux; it is merely postulated in Reference [76], presumably by analogy with the other expressions. In the usual edge modeling process, various components of the Braginskii equations are replaced with subsets of the Equations 4.14. It should be clear that such a process is rather arbitrary, but some guidelines do apply.

We apply the general principle that if the anomalous process is presupposed to "act" like the underlying physical process, then by definition, one should be careful to carry its effects through the full set of equations.[3] Strictly speaking, for example, if an anomalous radial *advection* of particles, $v_\perp$, is posited, then there should be a

---

[3] Note that this rule has more philosophical implications—that our goal is to model the system by mimicking known fluid processes preferentially over arbitrary empirical data fits, etc. The approach is defensible, but not absolute.

corresponding convection of heat, $v_\perp nkT$, and momentum flux, $mn\vec{u}v_\perp$. Further, that velocity will contribute to the viscous stress tensor in the way in which we would expect convection to act. On the other hand, if the assumption is that the bulk radial velocity is due to some other process, say Fick's law particle diffusion (scaling as the radial density gradient), the process may enter a bit differently. In the diffusion case, momentum and energy are transported with the particles as before, but no momentum is imparted by radial gradients of this velocity, by assumption. Diffusive particle transport will *not* contribute to the Braginskii viscous stress tensor. The work term in the energy balance for this latter case, $(\vec{u} \cdot \mathbf{P})$, is computed with the *total* velocity (including diffusive components) acting on the stress tensor, which has no effects from the diffusion velocity. This is rather a subtle issue, but gives rise to a number of difficulties in sorting out experimental observations[93, 110, 111]. See § 4.6 for a detailed resolution of this issue.

We take the following subset of all possible anomalous transport models (assuming $n^e = n^i = n$:

$$\begin{aligned}
\Gamma_{\perp i} &= nv_\perp - D_\perp \nabla_\perp n \\
\mathbf{r} \cdot \mathbf{\Pi}_{||\perp} &= \mathbf{r} \cdot \nabla \cdot \left(m^i n \vec{u}\vec{u} - \mathbf{\Pi}^a \right) \\
q_{e\perp} &= \frac{5}{2} v_\perp n T^e - \chi^e_\perp n \nabla_\perp \left(kT^e \right) - \vec{u} \cdot \mathbf{\Pi}^{ae} \\
q_{i\perp} &= \frac{5}{2} v_\perp n T^i - \chi^i_\perp n \nabla_\perp \left(kT^i \right) - \vec{u} \cdot \mathbf{\Pi}^{ai}
\end{aligned}$$

This model implies that there is an anomalous convection speed, $v_\perp$, and an anomalous diffusive flux of particles driven by the gradient in n. There is likewise a diffusive component of energy, driven by the radial temperature gradient. The anomalous convection leads to terms in the stress tensor, while both lead to viscous heating.

In the Braginskii stress tensor, we take the strong magnetic field case, where the gyro-viscosities and perpendicular viscosities are negligible compared to the parallel term, η_0 (see [112] for a more careful treatment). In some models, an anomalous radial viscous flux, of the form $m^i n \nu_\perp \nabla_\perp v_{||}$, is added to the parallel momentum equation. It happens that the value used typically for $\nu_\perp$ leads to viscous transport of the order the gyro-viscous terms. Presumably, the flux was added to simplify the numerical schemes employed, but we ignore the perpendicular effects completely. The resulting form for the viscous stress tensor is

$$\mathbf{\Pi} = \Pi_{ss}\mathbf{bb} + \Pi_{rr}\left(\mathbf{I} - \mathbf{bb}\right)$$

where $\mathbf{b}$ is the unit vector in the direction of the magnetic field, and $\mathbf{I}$ is the identity tensor.

Flux Limited Parallel Thermal Diffusion

The heat flux vector is simplified, at least formally, by retaining only the parallel term in the classical description, i.e. the Spitzer[15] conductivity. However, it is often necessary to limit the conducted heat flux artificially because the Spitzer form leads to values greater than what can be carried by the fundamental particles of the system. The root of the breakdown lays in the inadequate description of the local state via the truncated Laguerre expansion in the kinetic analysis outlined above. The artificial limiting of the parallel diffusive flux typically takes the form

$$\kappa_{||}^e \Rightarrow \frac{\kappa_{||}^e}{\left(1 + \left|\frac{\kappa_{||}^e \nabla_{||} kT^e}{\alpha_e v_{\mathrm{th}} nkT^e}\right|\right)}$$

and is routinely used in the present calculations. Here α_e is an arbitrary constant parameter. The effect of this artificial limiting is discussed in later sections. We found computed results to be quite sensitive to this aspect of the model, marking it as a good candidate for further exploration.

Total Momentum Equation

To simplify the system, we apply the external constraint that the advective flow velocity for the electrons is everywhere identical to that of the ions (i.e. that frictional and electric field forces are negligible). The plasma approximation[17] is also invoked, equating the electron and ion particle densities. These assumptions preclude plasma currents and self-generated electrostatic electric fields, as discussed in Reference [18]. The significance of the neglected effects arising has been verified both experimentally and computationally (see Reference [93] for a summary of the present status of related experiments and computational physics investigations). But we arbitrarily ignore plasma currents because they have not been indicated as essential for establishing or enhancing the volumetric power dispersion regimes under study.

The foregoing assumptions reduce our system from 10 plasma unknowns to just 4, for the plasma density, parallel velocity, and ion and electron temperature. But the plasma velocity must satisfy both the electron and ion momentum equations which are much simpler when added together:

$$\frac{\partial \left(\left[m^i + m^e \right] n\vec{u} \right)}{\partial t} + \nabla \cdot \left(\left[m^i + m^e \right] n\vec{u}\vec{u} - \left[\mathbf{\Pi}^i + \mathbf{\Pi}^e \right] \right) + \nabla \left(p^i + p^e \right) = \vec{S}_p^i + \vec{S}_p^e$$

Since the mass of the electrons is very much smaller than that of the ions, $m^i + m^e \approx m^i$ (for Hydrogen $m^i/m^e > 10^3$). From the above arguments, the classical viscous stress tensor has only terms in $\eta_0 \sim nkT\tau$. Since $\tau \sim T^{5/2}\sqrt{m}$, $\eta_0^i/\eta_0^e \approx \left(T^i/T^e\right)^{5/2}\sqrt{m^i/m^e}$, the electron viscosity is comparable to the ion viscosity when

$T^e/T^i \approx (m^i/m^e)^{1/5} \approx 4.5$, and therefore not generally negligible. The simplified equation for the sum of ion and electron momentum now reads

$$\frac{\partial\left(m^i n\vec{u}\right)}{\partial t} + \nabla\cdot\left(m^i n\vec{u}\vec{u} - \mathbf{\Pi}\right) + \nabla p = \vec{S}_p$$

where $\mathbf{\Pi} = \mathbf{\Pi}^i + \mathbf{\Pi}^e$, $\vec{S}_p = \vec{S}^i_p + \vec{S}^e_p$, and $p = p^i + p^e$. There is no electron-to-ion momentum transfer terms left in the total momentum equation. Heating from ions to electrons is now

$$Q_{ei} = -Q_{ie} = -\frac{3m^e n}{m^i}\frac{k}{\tau_e}\left(T^e - T^i\right)$$

The flux-limited electron heat flux, $\vec{q}_e$, is computed as described above, while the ion heat flux is just the classical parallel expression, $\vec{q}_i = -\kappa^i_{\|}\mathbf{b}\cdot\nabla\left(kT^i\right)$ (where $\mathbf{b}$ is the unit vector along the magnetic field).

We have now defined explicitly the standard model for fluid *plasma* transport for tokamak edge plasma. In Appendix B, we detail the source terms necessary to couple the plasma to the background neutral particles (one specific model of the processes anyway). All that remains is to collect up these assumptions into the model equations, and transform these magnetic-field-oriented expressions into the natural coordinates of our discretization.

Resulting Equations

Collecting all the approximations, we can now write out explicitly the model equations for transport in a tokamak edge. These include a diffusion equation for atomic Hydrogen plus the Braginskii equations, simplified and augmented with anomalous radial transport and source terms (also, the electron mass was considered small compared that of the ions).

Thermal neutral gas diffusion:

$$\frac{\partial n_0}{\partial t} + \nabla \cdot \left(-D_0 \nabla n_0 \right) = -S_n \tag{4.15}$$

Plasma continuity

$$\frac{\partial n}{\partial t} + \nabla \cdot \left(n\vec{u} \right) = S_n \tag{4.16}$$

Total momentum

$$\frac{\partial \left(m^i n \vec{u} \right)}{\partial t} + \nabla \cdot \left(m^i n \vec{u}\vec{u} - \mathbf{\Pi} \right) + \nabla \left(p^i + p^e \right) = \vec{S}_p \tag{4.17}$$

Electron heat equation

$$\frac{3}{2}\frac{\partial p^e}{\partial t} + \nabla \cdot \left(\frac{5}{2} p^e \vec{u} + \vec{q}_e \right) = S_E^e + \vec{u} \cdot \nabla p^e \tag{4.18}$$

Ion heat equation

$$\frac{3}{2}\frac{\partial p^i}{\partial t} + \nabla \cdot \left(\frac{5}{2} p^i \vec{u} + \vec{q}_i \right) = \mathbf{\Pi}^i : \nabla \vec{u} + \frac{1}{2} m^i \vec{u}^2 S_n - \vec{u} \cdot \vec{S}_p^i + S_E^i + \vec{u} \cdot \nabla p^i \tag{4.19}$$

Source Terms

$$\begin{aligned}
S_n &= n n_0 \langle \sigma v \rangle^{\text{eii}} - n n \langle \sigma v \rangle^{\text{rec}} \\
\vec{S}_p &= m^i \vec{u}_0 n n_0 \langle \sigma v \rangle^{\text{eii}} + m^i \left(\vec{u}_0 - \vec{u} \right) n n_0 \langle \sigma v \rangle^{\text{cx}} \\
&\quad - m^i \vec{u} n n \langle \sigma v \rangle^{\text{rec}} \\
S_E^e &= -\epsilon^{\text{eii}}\, n n_0 \langle \sigma v \rangle^{\text{eii}} - \frac{3}{2} k T^e n n \langle \sigma v \rangle^{\text{rec}} - \frac{3 m^e n}{m^i} \frac{k}{\tau_e} \left(T^e - T^i \right) \\
S_E^i &= \left(5\text{eV} + \frac{1}{2} m^i \vec{u}_0^2 \right) n n_0 \langle \sigma v \rangle^{\text{eii}} \\
&\quad - \left(\frac{1}{2} m^i \vec{u}^2 + \frac{3}{2} k T^i \right) n n \langle \sigma v \rangle^{\text{rec}} + \frac{3 m^e n}{m^i} \frac{k}{\tau_e} \left(T^e - T^i \right)
\end{aligned}$$

4.5 Poloidal Plane Projection

Next, we write down these equations in terms of the parallel and transverse components of the derivatives and velocities. As above, we define the unit vectors in the parallel and radial direction, $\mathbf{b}$ and $\mathbf{r}$ respectively. Also, the unit vector in the flux surface parallel to the poloidal plane is $\mathbf{x}$ (note that $\mathbf{r}$ is $\perp$ to $\mathbf{x}$). The parallel component of the divergence is $\partial/\partial s = \mathbf{b}\cdot\nabla$. We define the projection factor, $\mathcal{P}$, so that $\mathbf{b}\cdot\nabla = \mathcal{P}\mathbf{x}\cdot\nabla$. Note that because the parallel direction changes with respect to the poloidal coordinate frame, the divergence and gradient operators will lead to terms involving $\nabla\cdot\mathbf{b}$ and $\nabla\cdot\mathbf{r}$. Since they are arbitrary spatial functions of the geometry at hand, they must be retained in the equations for generality (note that for fixed projection, as in many test problems, $\nabla\cdot\mathbf{b} = \nabla\cdot\mathbf{r} = 0$). The term, $\nabla\cdot\mathbf{b}$ is easily computable by noting that $\nabla\cdot\vec{B} = \nabla\cdot|\vec{B}|\mathbf{b} = 0$, so that

$$\nabla\cdot\mathbf{b} = \frac{\mathbf{b}\cdot\nabla|\vec{B}|}{|\vec{B}|} = \frac{\mathcal{P}}{|\vec{B}|}\frac{\partial|\vec{B}|}{\partial x} \tag{4.20}$$

For any vector, the transformation gives $\vec{a} = a_{||}\mathbf{b} + a_{\perp}\mathbf{r}$, we can write

$$\begin{aligned}\nabla\cdot\vec{a} &= \mathbf{b}\cdot\nabla a_{||} + \mathbf{r}\cdot\nabla a_{\perp} + a_{||}\nabla\cdot\mathbf{b} + a_{\perp}\nabla\cdot\mathbf{r}\\ &= \mathcal{P}\mathbf{x}\cdot\nabla a_{||} + \mathbf{r}\cdot\nabla a_{\perp} + a_{||}\nabla\cdot\mathbf{b} + a_{\perp}\nabla\cdot\mathbf{r}\end{aligned}$$

Common implementations of the finite volume approach (cf. [69]) are written for a constant transformation projection, $\mathcal{P}$, and capture correctly the coordinate effects in the first two terms. The latter two terms represent an effective flux lost from a coordinate due to the changing orientation with respect to the fixed volume walls.

The velocity, $\vec{u}$ is written in terms of the total anomalous radial velocity, $v_{\perp}$, the part that "acts" like convection, $V_{\perp}$ (as discussed above), and other usually diffusive contribution, v_D

$$\vec{u} = v_{\|}\mathbf{b} + v_{\perp}\mathbf{r} = v_{\|}\mathbf{b} + (V_{\perp} + v_D)\,\mathbf{r}$$

In the momentum equation, we encounter the product $\mathbf{b}\cdot\nabla\cdot(\rho\vec{u}\vec{u} - \mathbf{\Pi})$. In the strong magnetic field limit,

$$\begin{aligned}
\mathbf{b}\cdot\nabla\cdot\mathbf{\Pi} &= \mathbf{b}\cdot\nabla\cdot\left[(\Pi_{ss} - \Pi_{rr})\,\mathbf{bb} + \mathbf{I}\,\Pi_{rr}\right] \\
&= (\Pi_{ss} - \Pi_{rr})\,\nabla\cdot\mathbf{b} + \mathbf{b}\cdot\nabla\Pi_{ss}
\end{aligned}$$

The relevant stress components are

$$\begin{aligned}
\Pi_{ss} &= 2\eta_0\frac{\partial v_{\|}}{\partial s} - \eta_0\frac{2}{3}\nabla\cdot\vec{u} \\
\Pi_{rr} &= 2\eta_0\frac{\partial v_{\perp}}{\partial r} - \eta_0\frac{2}{3}\nabla\cdot\vec{u}
\end{aligned}$$

Substituting these relations into the expression we obtain

$$\begin{aligned}
\mathbf{b}\cdot\nabla\cdot\mathbf{\Pi} &= \frac{2}{3}\frac{\partial}{\partial s}\left[\eta_0\left(2\frac{\partial v_{\|}}{\partial s} - \frac{\partial V_{\perp}}{\partial r}\right) - \eta_0\left(v_{\|}\nabla\cdot\mathbf{b} + V_{\perp}\nabla\cdot\mathbf{r}\right)\right] \\
&\quad +2\eta_0\left(\frac{\partial v_{\|}}{\partial s} - \frac{\partial V_{\perp}}{\partial r}\right)\nabla\cdot\mathbf{b}
\end{aligned}$$

Similarly, we obtain for the convective term, $\mathbf{b}\cdot\nabla\cdot(\rho\vec{u}\vec{u})$,

$$\begin{aligned}
\mathbf{b}\cdot\nabla\cdot(\rho\vec{u}\vec{u}) &= \mathbf{b}\cdot\nabla\cdot\left(\rho v_{\|}^2\mathbf{bb} + \rho v_{\|}v_{\perp}(\mathbf{br} + \mathbf{rb}) + \rho v_{\perp}^2\mathbf{rr}\right) \\
&= \frac{\partial}{\partial s}\left(\rho v_{\|}^2\right) + \frac{\partial}{\partial r}\left(\rho v_{\|}v_{\perp}\right) + \rho v_{\|}^2\nabla\cdot\mathbf{b} + \rho v_{\|}v_{\perp}\nabla\cdot\mathbf{r}
\end{aligned}$$

Combining these expressions,

$$\begin{aligned}
\mathbf{b}\cdot\nabla\cdot(\rho\vec{u}\vec{u} - \mathbf{\Pi}) &= \frac{\partial}{\partial s}\left(\rho v_{\|}^2\right) + \frac{\partial}{\partial r}\left(\rho v_{\|}v_{\perp}\right) - \frac{2}{3}\frac{\partial}{\partial s}\left[\eta_0\left(2\frac{\partial v_{\|}}{\partial s} - \frac{\partial V_{\perp}}{\partial r}\right)\right] \\
&+ \left[\rho v_{\|}^2 - 2\eta_0\left(\frac{\partial v_{\|}}{\partial s} - \frac{\partial V_{\perp}}{\partial r}\right)\right]\nabla\cdot\mathbf{b} + \rho v_{\|}v_{\perp}\nabla\cdot\mathbf{r} \\
&- \frac{2}{3}\frac{\partial}{\partial s}\left[\eta_0\left(v_{\|}\nabla\cdot\mathbf{b} + V_{\perp}\nabla\cdot\mathbf{r}\right)\right]
\end{aligned}$$

Final Model Equations

Combining all the model and geometry information of this section (in addition to a couple of Appendices), we can finally write the model equations to be solved in FMS2D. The set is written in terms of gradients in the poloidal plane, and includes the effects of a spatially varying transformation to parallel coordinates. One species of neutrals (atomic Hydrogen) is included—more neutral species are input in an analogous way (see the PISCES model for an example where molecular Hydrogen was added).

Atomic Hydrogen (thermalized with ions):

$$\frac{\partial n_0}{\partial t} + \frac{\partial}{\partial x}\left(-D_0\frac{\partial n_0}{\partial x}\right) + \frac{\partial}{\partial y}\left(-D_0\frac{\partial n_0}{\partial y}\right) = n^2\langle\sigma v\rangle^{\mathrm{rec}} - nn_0\langle\sigma v\rangle^{\mathrm{eii}} \tag{4.21}$$

Plasma Density (continuity):

$$\begin{aligned}
&\frac{\partial n}{\partial t} + \mathcal{P}\frac{\partial}{\partial x}\left(nv_{\|}\right) + \frac{\partial}{\partial y}\left(nv_{\perp}\right) \\
&\qquad\qquad = nn_0\langle\sigma v\rangle^{\mathrm{eii}} - n^2\langle\sigma v\rangle^{\mathrm{rec}} - nv_{\|}\nabla\cdot\mathbf{b} - nv_{\perp}\nabla\cdot\mathbf{r}
\end{aligned} \tag{4.22}$$

Parallel Momentum (total):

$$\begin{aligned}
&\frac{\partial}{\partial t}\left(\rho v_{\|}\right) + \frac{\partial}{\partial y}\left[\rho v_{\|}v_{\perp} - \eta_{\perp}\frac{\partial v_{\|}}{\partial y}\right] \\
&\quad \mathcal{P}\frac{\partial}{\partial x}\left[\rho v_{\|}^2 + p - \frac{2}{3}\eta_0\left(2\mathcal{P}\frac{\partial v_{\|}}{\partial x} - \frac{\partial V_{\perp}}{\partial y}\right) - \frac{2}{3}\eta_0\left(v_{\|}\nabla\cdot\mathbf{b} + V_{\perp}\nabla\cdot\mathbf{r}\right)\right] \\
&= \; -\rho v_{\|}\left(nn_0\langle\sigma v\rangle^{\mathrm{cx}} + n^2\langle\sigma v\rangle^{\mathrm{rec}}\right) \\
&\qquad - \left[\rho v_{\|}^2 + p - 2\eta_0\left(\mathcal{P}\frac{\partial v_{\|}}{\partial x} - \frac{\partial V_{\perp}}{\partial y}\right)\right]\nabla\cdot\mathbf{b} - \rho v_{\|}v_{\perp}\nabla\cdot\mathbf{r}
\end{aligned} \tag{4.23}$$

Electron Heat:

$$\begin{aligned}
&\frac{\partial}{\partial t}\left(\frac{3}{2}p^e\right) + \mathcal{P}\frac{\partial}{\partial x}\left(\frac{5}{2}p^e v_\parallel - \mathcal{P}\kappa_\parallel^e \frac{\partial kT^e}{\partial x}\right) + \frac{\partial}{\partial y}\left(\frac{5}{2}p^e v_\perp - \kappa_\perp^e \frac{\partial kT^e}{\partial y}\right) \\
&\quad = \; -nn_0\langle\sigma v\rangle^{\mathrm{eii}}\epsilon_e^{\mathrm{eii}} - n^2\langle\sigma v\rangle^{\mathrm{rec}}\frac{3}{2}kT^e - \frac{3m^e n}{m^i}\frac{k}{\tau_e}\left(T^e - T^i\right) \\
&\qquad - \left(\frac{5}{2}p^e v_\parallel - \mathcal{P}\kappa_\parallel^e \frac{\partial kT^e}{\partial x}\right)\nabla\cdot\mathbf{b} - \left(\frac{5}{2}p^e v_\perp - \kappa_\perp^e \frac{\partial kT^e}{\partial y}\right)\nabla\cdot\mathbf{r} \\
&\qquad + \mathcal{P}v_\parallel \frac{\partial p^e}{\partial x} + v_\perp \frac{\partial p^e}{\partial y}
\end{aligned} \tag{4.24}$$

Ion Heat:

$$\begin{aligned}
&\frac{\partial}{\partial t}\left(\frac{3}{2}p^i\right) + \mathcal{P}\frac{\partial}{\partial x}\left[\frac{5}{2}p^i v_\parallel - \mathcal{P}\kappa_\parallel^i \frac{\partial kT^i}{\partial x} + \frac{2}{3}v_\parallel \eta_0\left(2\mathcal{P}\frac{\partial v_\parallel}{\partial x} - \frac{\partial V_\perp}{\partial r}\right)\right] \\
&\qquad\qquad + \frac{\partial}{\partial y}\left[\frac{5}{2}p^i v_\perp - \kappa_\perp^i \frac{\partial kT^i}{\partial y} + \frac{2}{3}v_\perp \eta_0\left(2\frac{\partial V_\perp}{\partial y} - \mathcal{P}\frac{\partial v_\parallel}{\partial x}\right)\right] \\
&= \; nn_0\langle\sigma v\rangle^{\mathrm{eii}}\left(\frac{1}{2}m^i\vec{u}^2 + 5\mathrm{eV}\right) - n^2\langle\sigma v\rangle^{\mathrm{rec}}\frac{3}{2}kT^i \\
&\quad + nn_0\langle\sigma v\rangle^{\mathrm{cx}}m^i\vec{u}^2 - \frac{3m^e n}{m^i}\frac{k}{\tau_e}\left(T^e - T^i\right) \\
&\qquad\qquad - \left[\frac{5}{2}p^i v_\parallel - \mathcal{P}\kappa_\parallel^i \frac{\partial kT^i}{\partial x} + \frac{2}{3}v_\parallel \eta_0\left(2\mathcal{P}\frac{\partial v_\parallel}{\partial x} - \frac{\partial V_\perp}{\partial y}\right)\right]\nabla\cdot\mathbf{b} \\
&\qquad\qquad - \left[\frac{5}{2}p^i v_\perp - \kappa_\perp^i \frac{\partial kT^i}{\partial y} + \frac{2}{3}v_\perp \eta_0\left(2\frac{\partial V_\perp}{\partial y} - \mathcal{P}\frac{\partial v_\parallel}{\partial x}\right)\right]\nabla\cdot\mathbf{r} \\
&\quad + \mathcal{P}v_\parallel \frac{\partial p^i}{\partial x} + v_\perp \frac{\partial p^i}{\partial y}
\end{aligned} \tag{4.25}$$

Pressure and Mass Density:

$$p = p^i + p^e \qquad p^e = nkT^e \qquad p^i = nkT^i \qquad \rho = m^i n \tag{4.26}$$

4.6 Inconsistencies and Anomalous Transport

- In this section we resolve a well-documented inconsistency with the plasma modeling equations in the presence of anomalous transport.

As pointed out in Appendix B of M. Baelmans' doctoral dissertation[93], the equations for conservation of energy can be derived by computing the velocity integral of the kinetic equation weighted against $\frac{1}{2}m\vec{u}^2$, obtaining Equation 4.9

$$\begin{aligned}\frac{\partial}{\partial t}\left(\frac{3}{2}p+\frac{1}{2}mn\vec{u}^2\right)&+\\ \nabla\cdot\left[\left(\frac{3}{2}p+\frac{1}{2}nm\vec{u}^2\right)\vec{u}+\vec{q}-\vec{u}\cdot\mathbf{P}\right]&=nq\vec{E}\cdot\vec{u}+\int d^3\vec{v}\frac{m\vec{v}^2}{2}C\end{aligned}$$

or against $\frac{1}{2}m\left(\vec{v}-\vec{u}\right)^2$ obtaining the form appearing in the Braginskii analysis,

$$\begin{aligned}\frac{3}{2}n\frac{DkT}{Dt} \quad + \quad & p\nabla\cdot\vec{u}=-\nabla\cdot\vec{q}+\mathbf{\Pi}:\nabla\vec{u}+Q\\ &+\left(\frac{1}{2}m\vec{u}^2-\frac{3}{2}kT\right)S_n-\vec{u}\cdot\vec{S}_p+S_E\end{aligned}$$

In earlier sections, we showed that the continuity and momentum equations can be used to convert between the two. But when anomalous fluxes are inserted into the equations in favor of the classical expressions, the two energy equation forms are no longer equivalent. It was concluded that there is an arbitrary choice regarding which of the two results are more correct (see References [110, 111] for similar examples).

The resolution of this issue lies in the way that the anomalous transport is assigned. The validity of the manipulations performed to compare the two energy equations is predicated on whether the Braginskii momentum equations are satisfied identically. However, a typical edge model assumes that the radial velocity is determined by particle diffusion

$$v_{\perp}\mathbf{r}=-\frac{D_{\perp}}{n}\nabla_{\perp}n \tag{4.27}$$

which surely is not consistent with the Braginskii set. Arbitarily assuming a form for one of the dependent variables over-specifies the system, and will give either singular or consistent equations. On the other hand, is seems perfectly valid to discard one of the Braginskii equations in favor of an anomalous expression, provided the modeler is aware that in doing so, he/she has rendered the self-consistent picture of the Braginskii model invalid.

Another example is the one-dimensional model presented for the PISCES-A modeling in Chapter 8, which is predicated on the assumption that the state profiles are described via a set of self-similar Bessel function profiles. To reconcile this assumption with the classical fluid picture, a source term is added to the continuity equation of the form

$$-n\frac{D_\perp}{\lambda_\perp^2}$$

with similar expressions added to the momentum and energy equations. If we want to specify that the radial velocity given by the Expression 4.27, does satisfy classical conservation of momentum and *total* energy, we are free to likewise adjust the source term, $\vec{S}_p$, in the heat equation appropriately to ensure this condition.

The issues boil down to the more philosophical question about the real intention of modeling experimental plasma devices—to match data and predict results, or to understand the process in terms of known fluid behavior. In this sense, it would be perfectly valid to add arbitrary processes into the model wherever necessary to suit data-fitting purposes, without carrying the term throughout out the model "self-consistently". However, in doing so, we loose the aid of our intuition in extending the model to unexplored new systems.

Ultimately, the way to reconcile these two treatments is to compute them both and compare the results obtained. In the typical example of added anomalous radial velocity, it turns out that the two derived energy equations differ by exactly the $\vec{F} \cdot \vec{u}$ work required to sustain the assumed velocity against radial pressure and viscous forces. The force is included in the total energy equation implicitly, while it is absent by default from the heat equation. With this information, we can reconcile completely the expressions employed in the References [76, 81] (the substitution is carried out rather mysteriously via substitution of the electron momentum balance into the ion and electron total energy equations). The neglected cooling term in question is approximately

$$v_\perp \nabla_\perp p = \frac{D_\perp}{n} \nabla_\perp n \nabla_\perp p$$

Inserting typical tokamak parameters, $\nabla_\perp n \approx n/L_n$, $\nabla_\perp T \approx T/L_T, L_T \approx L_n/2 \approx$ 5cm, and $n, T, D_\perp \approx 10^{20}, 50, 10^4$. The radial pressure gradient is then $\approx 10^3 erg/cm^3$, so the cooling is on the order of .1MW/m^3, and fortunately thus make no *practical* difference in the computations, despite the confusion! For simplicity then, we have dropped this discrepancy term from our heat equation, 4.26.

4.7 Conclusions

We have derived a set of transport equations rigorously which are to be used for modeling the edge plasma in tokamak divertor-related experiments (see Equations 4.5). This set assumes that one group of background atomic Hydrogen neutral gas resides in the domain of interest, along with one species each of electrons and singly charged ions. The atomic Hydrogen is modeled with the OGITE[18] approximation, where the fluid has equilibrated everywhere with the ions. The plasma equations are constructed based on the 10 equation Braginskii model[64], except that auxiliary conditions are specified on the radial fluxes of particles, momentum and energy (see Equations 4.15), in accordance with standard edge modeling practice. Additionally, we assume quasineutrality, and ignore fluid drifts and plasma currents to simplify the physical model for analysis.

The result of the analysis is a set of 4 equations for the plasma velocity, total parallel momentum, electron and ion temperatures (Equations 4.16 – 4.19). These four are coupled to the diffusing background neutral species via particle, momentum and energy exchange terms that were outlined carefully in Appendix B.

The reduced set of equations five equations are written out by component for the two-dimensional poloidal plane to be modeled, and are written out as Equations 4.23 – 4.26. This process illustrates that additional terms arise due to the expansion of the parallel coordinate system in projecting to the poloidal plane.

Chapter 5

General Description of FMS2D

The computer can help us to understand various aspects of the tokamak edge plasma, but only if we design and implement sufficiently general and efficient techniques to obtain numerical solutions of our models. As part of the research, we have designed and implemented the FMS2D (Fluid modeling system in two dimensions) computer program with exceptionally general design criteria:

- We require the ability to specify and solve diverse models;
- We should be able to alter the models without undue difficulty;
- To the extent possible, our model specification should be devoid of artifacts relating to the specific numerical scheme or programming language used to implement the model.

Current indicators show that understanding the fluid behavior of the tokamak plasma edge, especially in the divertor regions, will be critical to successful design of future plasma devices. The above criteria were designed to enhance the usefulness of the code for follow-on research on tokamak edge-related plasmas. Implementation is outlined in the subsequent section.

Before proceeding however, note that practical experience with the code on complex applications has brought about modifications in the detailed implementation strategy of the solution methods. As of this writing, the present organizational structure is in use by at least four researchers over a wide range of application. Several fundamental changes have already been incorporated giving rise to the form to be described herein. With continued success, we expect the number of users, and thus practical experiences to increase over the coming years. Logically, the next change to FMS2D should be to improve overall computational efficiency.

5.1 Design Philosophy of FMS2D

Fluid models of plasma can be quite complex, particularly those that include the self-consistent effects of neutral particles. The plasma models contain anisotropic state dependent diffusivities and source functions from tabulated data. Neutral particles often exist in regimes where fluid model validity is questionable at best. Models describing the plasma and neutral transport are thus under intense scrutiny in the fusion community; new augmentation or modifications are proposed at nearly every fusion-related physics conference or workshop. A computer code for study of the edge plasma research must be able to handle a diverse set of fluid models.

To facilitate such diversity, we designed FMS2D using an exceptionally modular approach. The starting point was to recognize that the models of our interest (by definition) are expressible as vector fluid conservation laws. We developed a language in which to easily express complex coupled conservation law models. We implemented general techniques to discretize these equations, and then algorithms to solve the resulting discrete equation systems. The initial incarnation of this effort was carried out in Fortran, and without particular regard to computational efficiency.

5.1.1 The Model Specification

The code is designed to accept a system of time dependent fluid–type equations in two spatial dimensions. We are careful to include sufficient generality to cover cases of immediate interest for tokamak edge modeling, as well as likely future extensions. Typical examples of the equations encountered consist of a diffusion model for the neutral density coupled to the modified Braginskii fluid equations for plasma density (as detailed in earlier sections). The range of boundary conditions includes specified value (i.e. Dirichlet), specified flux, specified normal derivatives (i.e. Neumann), and specified (possibly nonlinear) relations between unknowns (interdependent Robbin-type; such as that occurring at Bohm sheaths).

To accommodate these and other expected forms, we chose a numerical scheme accepting an arbitrary system of equations of the form

$$\frac{\partial C(\vec{x},t,\vec{s})}{\partial t} + \nabla \cdot \vec{\Gamma}_c - \nabla \cdot (D \cdot \nabla s) = S \tag{5.1}$$

with boundary conditions specified in the form

$$G(\vec{x},t,\vec{s},\nabla\vec{s}) = 0$$

where $\vec{s} = [s_1, \ldots, s_n]$ is the state vector of unknowns, C is the concentration of some physically conserved property, $\vec{x}$ is the spatial coordinate and D is the diffusivity tensor (e.g. $\vec{s} = [n_H, n, v, T_i, T_e]$ in the Asdex model, Chapter 7, while $\vec{s} = [n_{H_2}, n_H, n, v, T_e]$ in the PISCES model, see Chapter 8). The flux term, $\vec{\Gamma}_c$, includes convection and other nonlinear or coupling fluxes not explicitly expressible in diffusion form (see the comment section regarding computation of artificial dissipation). S is the volumetric source rate of C. $\vec{\Gamma}_c$, D and S generally are functions of the state, position, time and state gradient.

In FMS2D, a FORTRAN77 compliant[1] source code subroutine serves as the interface to pass the complete model, consisting of C, $\vec{\Gamma}$, D and S, for a given $\vec{s}$, $\nabla\vec{s}$, $\vec{x}$ and t (in the implementation, we found it more convenient to express the diffusivity as a diffusive flux; internal parts of the code then unravel D as required). This implies, for example, that complete replacement of the physical model requires only the generation of one new interface file. Moreover, we have provided a number of debugging and variation control implementation tools are to ease this part of the specification. Appendix E gives an example one such specification file, the PISCES model—this complex example was chosen because it demonstrates the power of such abstractions. Details of the auxiliary tools available will not be given here. It suffices to point out that once the model has been expressed in this fashion, facilities exist to parse the source code into any language/format that is ultimately convenient (an X-Window interface for model input and debugging is clearly possible, for example).

5.1.2 Computational Domains

In addition to model definition, a two-dimensional domain must be specified on which FMS2D is to compute a steady state solution. The issue is a bit more complicated than a cursory glance may indicate. The Scrape-Off-Layer (SOL)

---

[1] One design selection, apparent in Appendix E, was to violate the deplorable six character naming convention and case insensitivity enforced by the FORTRAN77 standard. The additional freedom drastically improves the readability of the source code, and thus is supported by nearly every modern FORTRAN77 compiler.

and edge of a tokamak are defined and illustrated in the introductory chapter of this document to include the outer periphery of the tokamak plasma *and* a layer into the core. In the most general case then, our domain should include regions both inside and outside the closed magnetic flux surface structures. And since the medium we wish to investigate is highly anisotropic with respect to these surfaces, a discretization of the modeling equations will be aligned to this complex field structure (a significant portion of the work presented in this thesis pertains to proving that this must be so). As a result, our total domain of interest is often *multiply connected.*

A typical domain is shown in Figure 6.2 of Chapter 6. Again, it is noted that *toroidal* symmetry is assumed so that the 3D geometry of this region is obtained by rotating the plane “the long way around a doughnut”. In Chapter 7, we use a simpler edge plasma domain, that we will invoke as a subset of the full problem when convenient for benchmarking the code and studying our plasma model. For simplicity, this region excludes a core-like portion from the computation, but includes all the other necessary boundaries. This logically square domain is a great deal easier to deal with; it provides a fair testbed to analyze the physical coupling processes, yet admits profile solutions that are easier to plot and understand. Another simplified domain is illustrated with the cylindrical PISCES geometry (as shown in Figure 8.8 of Chapter 8). PISCES is designed to operate under plasma conditions similar to those expected in a true divertor, but the simple geometry eases understanding of the physical processes involved without the complicating effects of difficult geometries and intricate magnetic field shaping.

5.2 Discretization of the General Equation

Given the general model specification, we are free to develop any suitable discretization of the individual terms that is convenient and useful. In Chapter 6, we look in detail at various methods to discretize the anisotropic diffusion. The discretization of convective fluxes and volumetric source terms are detailed below. We also discuss how to handle the time derivative and numerical boundary conditions. The prescriptions supplied herein, like the models presented later, are merely example applications of the FMS2D methodology, and are quite easily interchangeable within the structure of FMS2D (in most cases, via modification or replacement of a single FORTRAN subroutine).

5.2.1 Convection Discretization

The numerical operator employed in FMS2D for convective transport consists of two parts, a central-type difference and then a subsequent addition of artificial dissipation required for monotonicity. The former is performed in terms of a finite-volume techniques, while the latter is more appropriately handled with finite differences, as discussed in the section regarding discretization of anisotropic dissipation physics.

The convection flux term, $\nabla \cdot \vec{\Gamma}_c$, in the general conservation equation can be integrated over a small discrete volume in the usual finite volume method for arbitrary cell shape and orientation,

$$\int_{\Omega} \nabla \cdot \vec{\Gamma}_c \, dV = \oint_{\partial\Omega} \vec{\Gamma}_c \cdot dA \approx \sum_i F^i_{jc} \cdot dA^i_j \tag{5.2}$$

where F^i_{jc} is the ith component of the flux through the jth wall of the finite volume. Thus, the volume integral of the divergence of the convective flux can be computed as the sum of convective fluxes through each wall.

The convective fluxes are computed at the center of each wall of the finite volumes. For this, state values at the walls are linearly interpolated from neighboring nodes on either side of each wall. The wall state values are passed to the model interface file where the user routine forms the two orthogonal components of the local convective flux and passes them back to a control volume operator routine. The control volume operator routine then proceeds to form the inner product of the wall fluxes with the corresponding directed cell wall area vectors to determine the total flux divergence via Equation 5.2.

A couple of points are worth noting:

1. For edge problems, the model terms are most easily expressed in terms of the locally flux-aligned coordinate system. However, the coordinates of the discrete nodes in the 2D space are more easily dealt with in cartesian or cylindrical geometry. In FMS2D, we chose to follow this mixed specification by storing grid point locations in either cartesian or cylindrical coordinates. Numerical gradients (required at cell walls, for example) are computed in the grid coordinates, and then transformed using the local magnetic field geometry. The resulting flux-surface-based gradients are passed into the model specification routine for the use in forming the physical conservation law terms. This step requires that the transformation from lab coordinates to poloidal coordinates be 'smooth'. The transformation cannot be smooth near the X-point of a diverted tokamak or when the underlying mesh is too coarse. This issue is elaborated on in the section regarding anisotropic diffusion.

2. The error generated in approximating the wall state value with a linear in-

terpolation will scale as the square of the local mesh spacing (i.e. linear interpolation is second order accurate). Likewise, the error committed in approximating convective fluxes with these interpolated states has leading error of the same scaling. Therefore, the flux divergence computed from Equation 5.2 can easily be shown to scale *linearly* with the local mesh spacing, i.e. the flux divergence will be computed to first order accuracy.

3. The error committed in adding simple artificial dissipation is well-known to also scale linearly with the mesh spacing. Any higher order convective flux calculation will be dominated by the error term of the artificial diffusion; more accurate schemes than that above would not be worth the additional effort.

4. The classification of "convective fluxes" and "source terms" is somewhat arbitrary in our setting. Notice, for example, that the pressure term in the momentum balance equation can appear as a pressure gradient source term, or a divergence of momentum density in the fluxes. To simplify instructions for the users of FMS2D, we needed to ensure that either specification was treated equivalently in the numerics. In general then, the convective fluxes can depend on gradients as well as state values at the cell wall. However, a linear interpolation can only represent a gradient with first order accuracy unless the wall is midway between neighboring cells. The divergence of such quantities on an irregular mesh will be zero order accurate, and therefore will *not*, in general, be convergent with resolution!

With regard to final point above, we have partially resolved the problem by computing state gradients at the wall with a *quadratic* polynomial interpolation.

For a rectangular grid, this involves three nodal values, and can thus be constructed without increasing the overall stencil size. For a non-rectangular grid, quadratic interpolation is unduly complex; it seems better to clearly specify to users that convective fluxes cannot contain terms involving state gradients.

In all of our rectangular exemplary cases, the three nodes surrounding a wall are in line, and a convenient formula for the quadratically interpolated state gradient is generated by differentiating the Newton formula:

$$y(x) = y[x_1] + y[x_2,x_1](x - x_1) + y[x_3,x_2,x_1](x - x_1)(x - x_2)$$

$$y[x_i] = y_i$$

$$y\left[x_i,x_j\right] = \left(y_i - y_j\right) / \left(x_i - x_j\right)$$

$$y\left[x_i,x_j,x_k\right] = \left(y\left[x_i,x_j\right] - y\left[x_j,x_k\right]\right) / (x_i - x_k)$$

The gradient computed from this formula is then transformed per the above discussion, and passed to the model interface file, along with the wall states; the resulting convective fluxes are treated as usual.

5.2.2 Characteristic Speeds and Artificial Dissipation

The maximum characteristic velocity, a, determines the amount of numerical flux required to ensure non-oscillatory solution profiles. The actual relation is easy to derive heuristically. And since the FMS2D code is written to accept a generalized set of conservation laws, we also provide general method of computing the appropriate a values.

For the special case of a uniform cartesian grid, our discretization scheme reduces to centered finite differences. It is well-known that centered differences generate nonphysical extrema in the presence of the anticipated physics (strong advection), and require a corrective stabilization technique. There are several general candidates: slope limiters, upwind differencing, and artificial diffusion; all of which can be applied in our general setting. Initially, we incorporated a streamline–weighted artificial diffusion, as it is the most robust and simple option. It has the drawback of a rather low accuracy, so that we will need to replace this rudimentary advection method with a more appropriate scheme in the final stages of development. Prior to conclusion of the present work however, we attempted to use a modified form of the high-order family of methods employed by Jameson (cf. [113]). The methods use third-order profile interpolants unless a discontinuous solution is detected, where a first-order scheme is then used. In our implementation, the switch changing the method from third to first order needed to be 'smooth' so that the Jacobian matrix remain well-behaved. However, our limited experimentation never resulted in a satisfactory switch that was general enough for the general edge problem. We opted therefore to rely on the lower order results for discussions.

The general form of the first-order accurate artificial dissipation term can be extracted from the 1D advection equation (for constant positive velocity, a)

$$\frac{\partial u}{\partial t} = -\frac{\partial (au)}{\partial x} \tag{5.3}$$

Because there is no physical diffusion in this ideal case, the 1D flux of u on a discrete grid towards node i, $\vec{\Gamma}_{i-\frac{1}{2}} = au_{i-1}$. On the other side of point i, $\vec{\Gamma}_{i+\frac{1}{2}} = au_i$. From Equation 5.3, the time rate of change of u is the divergence of these fluxes, or (for uniform grid spacing, Δ_x)

$$\frac{\partial u}{\partial t} = \frac{u_i - u_{i-1}}{\Delta_x} \tag{5.4}$$

This expression is the standard, first order accurate upwind difference approximation to the first order convection operator. The expression is to be compared to the second order accurate central difference, $\partial u/\partial t = (u_{i-1} - u_{i+1})/(2\,\Delta_x)$. Our finite volume treatment of convective fluxes outlined earlier reduces to the latter form in the case of a uniformly spaced mesh and constant speed.

It is well-known that Equation 5.4 produces a stable oscillation-free solution to the convection equation, and that central schemes do not (not without significant modifications). To obtain Equation 5.4 from our central scheme, we add

$$a\frac{u_i - u_{i-1}}{\Delta_x} - a\frac{u_{i+1} - u_{i-1}}{2\,\Delta_x} = -\left(\frac{a\Delta_x}{2}\right)\frac{u_{i-1} - 2u_i + u_{i+1}}{(\Delta_x)^2} \approx -D_A\nabla^2 u$$

Here, we recognized the difference operator that approximates the second derivative with respect to x. Since the term to add has a diffusion-like form, it generally is called artificial diffusion, where D_A is the artificial diffusion coefficient. It is important to note that D_A is proportional Δ_x, so that the purely numerical flux from D_A diminishes as the grid is refined. Also note that the required artificial diffusivity scales linearly with the characteristic information propagation velocity of the system.

The above derivation is rather heuristic; more rigorous proofs abound in the literature on basic numerical schemes. Two important results arise from the rigorous derivations: the truncation error of this scheme decreases as the first power of the mesh spacing, Δ_x, and the errors are diffusive in nature. The latter result is beneficial as it implies that underresolution of fine-scale physical convection phenomena will not lead to numerically generated oscillations in solution profiles. Unfortunately, we'll see in later sections that the former leads to excessive grid resolution requirements. However, the scheme is robust, well understood and has been useful in developing the FMS2D code logic and understanding the plasma edge system.

To apply the scheme for a given model, we need to choose the speed, a, greater than any information propagation speed. In order to identify the required directions and speeds, the (linearized) conservation equations should be transformed to eliminate cross-dependent fluxes (i.e. to produce a set of scalar equations such as Equation 5.3). The procedure is well documented in the literature (see e.g. [114]) and amounts to computing the eigenvalues and eigenvectors of the flux Jacobian obtained by linearizing the flux operator with respect to the conserved quantities. Once transformed, the equations describe linearly independent advection along the eigenvectors, each with velocity equal to the corresponding eigenvalue. Thus, $a \geq \lambda_i$, where λ_i are the eigenvalues of the flux jacobian.

As a simple practical example, we consider 1D convection of two species of particles sharing the same momentum and density, but having distinct temperatures. Here, the states are $u = (n,v,T_e,T_i)$, our conserved quantities are $C = \left(n,Mnv,\frac{3}{2}nT_e,\frac{3}{2}nT_i\right)$, and our fluxes are $F = \left(nv,Mnv^2 + p,\frac{5}{2}nT_e v,\frac{5}{2}nT_i v\right)$, where $p = n(Te + Ti)$. The linearized flux Jacobian is:

$$J = \begin{pmatrix} 0 & 1/M & 0 & 0 \\ -Mv^2 & 2v & 2/3 & 2/3 \\ -5T_e v/2 & 5T_e/2M & 5v/3 & 0 \\ -5T_i v/2 & 5T_i/2M & 0 & 5v/3 \end{pmatrix}$$

The eigenvalues of this matrix are:

$$\vec{\lambda} = \left(5v/3, v, \frac{4v}{3} + \sqrt{\frac{15p/n + Mv^2}{9M}}, \frac{4v}{3} - \sqrt{\frac{15p/n + Mv^2}{9M}}\right) \tag{5.5}$$

If the convection flow is everywhere subsonic, our maximum characteristic velocity (a in D_A above) is

$$a = \frac{4v}{3} + \sqrt{\frac{15p/n + Mv^2}{9M}}$$

$a = 5v/3$ otherwise. In this case, 1D pressure and advection waves are allowed to travel only along the x axis. In a 2D case, the characteristic directions are generally skewed from the coordinate and/or mesh axis. Whether aligned or misaligned, we shall apply the artificial dissipation using the optimal anisotropic diffusion operator developed elsewhere in this document.

It is important to recognize from Equation 5.5 that our dissipation apparently allows for acoustic propagations as well as convection. Often, upwind-type schemes (see e.g. the schemes described in Refs. [71] and [115]) are based only on *convection*. If care is not taken to include sound wave propagation, numerical oscillations may creep back into the solution despite using the strong first-order dissipation. This rather subtle point could be neglected in some cases, but is vitally damaging in others. We discuss the conditions for this, along with examples, in Chapter 8.

Also note that diffusion terms are omitted from the above analysis. The characteristic propagation velocity for diffusional transport is infinite (i.e. the eigenvalues of the flux Jacobian matrix are complex), and so too would be our numerical diffusion. Fortunately, the diffusion operator cannot itself generate internal numerical oscillations (see discussions elsewhere in this document regarding the so-called *maximum principal*). It is sufficient to include only the non-diffusive flux terms in the characteristic analysis.

Finally, note that the above analysis could be automated. As part of the user-model interface, the FMS2D user supplies conserved quantities and convective fluxes formed using the current state values. In principal, the resulting flux vector could be numerically linearized to form the flux Jacobian and eigenvalues could be numerically extracted as required. One might suggest this as a way to completely isolate the physics model specification from the numerical solution algorithms. However, the linearization and eigenvalue decomposition algorithms are computationally expensive, especially given the frequency they would be required and the accuracy of the the method. Fortunately, the analytic procedure above has only to be carried out once per system, and that could be accomplished using modern symbolic manipulation software packages.

5.2.3 Source Terms

The source terms can be integrated to second order over the cells by accounting only for the linear variation of the unknown state quantities. This can be accomplished simply be multiplying the source term evaluated with the state quantities at the center of each cell, with the respective cell's volume.

The source term plays a dominant role in the dynamics of the edge plasma, particularly in the high, and ultra-high recycling regimes. We therefore experimented with various higher accuracy methods to treat the nonlinear sources separately, e.g. 1–100 point Gaussian quadrature methods. We found that any more than single point integrations were inconsequential to the results obtained. Presumably, any benefit we gained by integrating the sources more accurately was swamped by the the large errors of the low-order convection operator described above. Additional testing of the source term discretization should follow implementation of higher order convection operators.

5.2.4 Numerical Boundary Conditions

Boundary conditions for our system are expressed as

$$G\left(\vec{s}, \nabla s, \vec{x}, t\right) = 0, \vec{x} \in \partial\Omega$$

where $\partial\Omega$ is the boundary of the computational domain, and $\vec{s}$ is the state. For the tokamak edge fluid models, these equations are nonlinear, and couple the various conserved quantities. Consider a Hydrogen plasma in contact with material surfaces. Here, Bohm sheath conditions apply such that $\vec{v} \cdot \vec{n} = C_s = \sqrt{k(T_e + T_i)/M}$ ($\vec{n}$ is the outward normal vector). The electron power flux to the surface is expressed in terms of a sheath transmission factor, δ, times the convected internal energy

$$P = \delta k T_e J^{+}_{sat},$$

$$\delta \cong 2/(1-\gamma_e) + \left(\frac{1}{2} M C_s^2 + kT_i + e\left[kT_e \frac{1}{2} \ln\left(\frac{M/2\pi m}{1 + T_i/T_e}\right)\right]\right)(1 - R_{iE}) \cong 7$$

for a pure Hydrogen plasma, where J_{sat} is the current of exiting plasma particles, γ_e is the secondary electron coefficient, and R_{iE} is the ion energy reflection fraction. Moreover, some quantities may be completely outward flowing at a boundary. In such cases, an explicit boundary condition is inappropriate, yet a boundary node may need to be present for other conserved quantities in the system. Numerical implications of both these features are discussed.

Since the steady state physical boundary equations contain no explicit relaxation parameter, the conditions must be satisfied at each time step. Such *instantaneous* application leads to numerical problems owing to two distinct phenomena. First, the nonlinear equations may be too difficult to solve without continuation or relaxation methods. Second, internal conservation for diffusion dominated systems leads to oscillatory internal profiles during system relaxation. Both problems are remedied by evolving the boundary condition equations with a pseudo time relaxation.

To illustrate, consider linear diffusion with a fixed boundary and flat initial conditions displaced from the solution. We wish to evolve the system to steady state using time relaxation of the internal profile. However, the initial time step is often considerably smaller than the characteristic diffusion time. Boundary conditions are satisfied exactly (without time relaxation), and the discrete cell next to the boundary will see a diffusion flux due to the the resulting gradient at the boundary. Since the time step is too small to allow diffusional transport, an opposing gradient will be set up internally to counter the boundary flux. Oscillating internal values are subsequently required to minimize net fluxes into each discrete cell. As the system evolves, the internal extrema will be diffused from the system, but strongly coupled plasma profiles will often evolve subsequently to conditions where the Newton algorithm can no longer solve the nonlinear equation for the next time step.

To remedy the situation, we evolve the boundary conditions according to a user-specified characteristic relaxation time, from the initial condition to a final time-independent steady value. The equation of interest is

$$\frac{dG}{dt} = -\frac{G(t)}{\tau_{BC}}$$

which becomes $G = 0$ in steady state. A backward Euler time discretization is applied, giving (after manipulations to cast the expression in terms of a residual function appropriate for our nonlinear solution routine)

$$R^{k+1} = G^{k+1} - G^{k}\left(\frac{\tau_{BC}}{dt + \tau_{BC}}\right)$$

where superscripts refer to discrete time level. An appropriate τ_{BC} is closely related to physical diffusion times and the exact form of the boundary condition. In FMS2D, τ_{BC} is therefore specified by the user for each of the states and independently along each of the boundary segments.

The outflow condition can be expressed in terms of an equation for the boundary node if we provide a consistent extrapolation expression of the internal profile.

Numerically, interior accuracy is maintained if such extrapolations are carried out to one order below that of the numerical scheme applied to the internal nodes. For our purposes, second order accuracy will suffice, so we use linear extrapolation, i. e.

$$G^k = s^k_{BC} - \left[s^k_2 + \frac{s^k_1 - s^k_2}{x^k_1 - x^k_2}(x_{BC} - x_2) \right]$$

at time k, where BC, 1, 2 refer to values on the boundary, and one and two nodes into the domain, respectively. Here, x_A is the *distance* to the boundary node from point A. This expression is inserted into the discretized time relaxation given above, from the solver's point of view, the outflow boundaries are treated uniformly with the rest of the equations.

Corner points exist so that every internal node looks identical in logical space. No conservation equation exists for the corner point, and it straddles physically two boundaries that may have different associated boundary conditions. We chose to use a simple planar extrapolation to compute the corner point state values, and have not seen any negative repercussion from the choice.

5.2.5 Time evolution

In these early stages of code development, we sought only steady solutions to the general equation, Equation 5.1. But strong nonlinearities often prevented iterative nonlinear solution approaches from converging–our implementation of the Newton-Raphson technique is discussed in a subsequent section. In the literature, a successful method relies on 'parameter continuation'. For our system of conservation laws, a quite natural continuation parameter is pseudo-time, so-called because we are disinterested in the details of the solution profiles as they evolve toward a steady equilibrium.

In FMS2D, the backward Euler method is the implemented time marching scheme;

$$\frac{\partial C}{\partial t} \sim \frac{C^{k+1} - C^k}{\Delta t} = -\nabla \cdot \mathbf{F}^{k+1} + \nabla \cdot \left[\mathbf{D^{k+1}} \cdot \nabla(\mathbf{s^{k+1}})\right] + S^{k+1}, \tag{5.6}$$

where superscripts denote time level, and $k+1$ is at a time Δt past the time at k. This equation is nonlinear, and becomes easier to solve as Δt approaches zero (where $\mathbf{s}^{k+1} = \mathbf{s}^k$). The easier equations are solved (using methods detailed in the following section), marching the profiles from initial guesses to steady state consistent solutions.

Near the completion of this work, we began to suspect that the foregoing method may not be appropriate for strongly coupled systems, because the nonlinear solver must evolve all states at a large time step. This may lead to non-physical damping of mode propagations that are necessary to keep the solution physical. In practice, we found with this adaptive time stepping, plasmas with significant internal structure evolving naturally to non-physical states. Often, the time stepping subsequently failed. Additional research is required here to determine if time-accuate solution methods would behave more appropriately.

5.2.6 Equation Solvers

The time and space discretized fluid equations, along with the discretized boundary conditions, form a nonlinear system of N equations, where N is the total number of unknowns—the number of states per cell ($n_0, n, u_{\|}$, and T, for example) times the number of grid cells—typically as large as $10,000$. These equations can be written as

$$R(\mathbf{s}) = 0. \tag{5.7}$$

This system may be strongly nonlinear, due to state dependent transport coefficients and source terms, for example. To solve the system, we use the multi-dimensional Newton's method [116]. Apparently, this method was first applied to the tokamak edge plasma problem by Knoll[69]. In this approach, starting from a present best guess, $\mathbf{s}_0$, we look for a small correction $\delta\mathbf{s}$ that solves

$$R(\mathbf{s}_0 + \delta\mathbf{s}) = 0. \tag{5.8}$$

By linearizing this equation, $R(\mathbf{s}_0 + \delta\mathbf{s}) \approx R(\mathbf{s}_0) + \delta\mathbf{s} \cdot J$, the correction can be obtained from

$$\delta\mathbf{s} \cdot J = -R(\mathbf{s}_0), \tag{5.9}$$

where $J(\mathbf{s}_0)$ is the "Jacobian matrix" of the nonlinear system. The linear system solution process is discussed in a later section.

Even if the linear problems can be solved adequately, Newton's method is often still unable to find a solution to the equilibrium equations directly. The initial guess may be too far from the equilibrium or the linearization used to derive the method may be a poor approximation to the full nonlinear problem. We guard against divergent Newton steps by damping undesirably large values (i.e. $\delta\mathbf{s} \rightarrow \lambda\delta\mathbf{s}$, where $\lambda \leq 1$). Our initial implementation was that of [117], but we found serious convergence problems when trying to solve difficult plasmas in the ITER parameter regime. These difficulties have not been reported in the literature. We therefore do not rule out that our implementation faithfully adhered to the specifications. In the most recent incarnation of the FMS2D code, the damping factor, λ, is set to alternatively unity, and then modified as follows:

$$\lambda = MIN\left(\lambda, \left|\frac{f_{max} s_i^{scale}}{ds_i}\right|\right) \tag{5.10}$$

where i runs over all unknowns, ds_i is the full Newton update for the ith component of the unknown state vector, s_i^{scale} is a characteristic scale value for each state, and f_{max} is a fixed number. The scale quantities are specified (also via the user interface file) functions of the local state, gradient, position and time. If, as is typical, s_i^{scale} is simply the local value of the state, Equation 5.10 just ensures that each value of the updated state will not differ from previous values by more than a factor of λ (and positive definite quantities, such as temperature and density will not be allowed to acquire negative values). This is not adequate, however for velocity components which will, in general, cross zero, and may coincidently take on small values. Since Equation 5.10 depends on state dependent scales, we can choose that the scale of the velocity components to vary as the local sound speed. In practice, we dynamically set the velocity scale function to

$$s_u^{scale} = MAX(\{u\} + f_u C_s)$$

where $f \sim 0.3$, and C_s is the local sound speed.

In addition to the above criterion, it seemed prudent to enforce that each state be limited ultimately to never leave a user-specified range. The following is therefore also applied:

$$\begin{aligned} if\ \left(s_i - \lambda ds_i < s_i^{min}\right)\ then\, \lambda &= \frac{s_i - s_i^{min}}{ds_i} \\ if\ (s_i - \lambda ds_i > s_i^{max})\ then\, \lambda &= \frac{s_i - s_i^{max}}{ds_i} \end{aligned}$$

In practice, if the latter criteria were met so as to reduce the damping parameter, the Newton iteration procedure often ceases to converge. The reason is that the linearized equation may admit solution update vectors that continue to point out of the "box" specified by $[s_i^{min}, s_i^{max}]$. The Newton iteration process then 'beats against the box walls' indefinitely. In FMS2D, this situation will eventually be flagged as a unconverged termination of the Newton solver.

The foregoing damping procedure, combined with the adaptive time evolution procedure has resulted in a marginally robust set of nonlinear solution tools, that have performed satisfactorily on *most* problems of interest. We find rather poor performance when applied to very difficult plasmas, such as those in the volume-recombination or gas-target regimes.

5.3 Linear System Construction/Solution

Repetitive construction of the $N \times N$ Jacobian system and its solution dominates the computational effort required to solve our set of nonlinear equations (here $N =$ *number of grid points (M)* $\times$ *number of unknowns (K)*). Practical solutions are obtained only by minimizing the storage requirements and optimizing the solution process for these linear systems. We detail the storage and linear solution schemes available in the FMS2D code, because both have a very significant impact on the maximum accuracy and speed for which a given plasma model can ultimately be computed. We find that the Jacobian matrix must be stored in a sparse format, and that an approximation to the matrix must also be systematically constructed, decomposed and stored in order to carry out efficient linear solution techniques.

The values stored as entries in the Jacobian have significant physical interpretation. An entry quantifies the degree of influence one unknown has over another in the domain, i.e. Jacobian rows are the set of linearized stencil entries about each node in the system. Knowing that we have constructed a *local* discretization of the conservation partial differential equations, we have determined in advance that many of these elements are zero by design. In fact, owing to the 9-point stencil we have incorporated, there are, at most, $9K$ nonzero entries per row. Since this is considerably less than the total number of positions in the $N \times N$ matrix, we can drastically reduce the storage requirements of our code.

A logically rectangular 2D domain is arbitrarily divided into cells. We arbitrarily number these cells via a single index starting in the "lower-left corner", and proceeding left-to-right first, then bottom-to-top. Indexing within the Jacobian is further chosen to vary fastest on the individual states at each node. If there are five states in the system of interest, the first five rows of the Jacobian will specify the linearized discretization stencil for these five states at the node in the lower left corner of the grid (grid-major ordering–see figure 5.3). Locations of the nonzero entries in the Jacobian matrix are depicted graphically in Figure 5.3 (ignoring boundary nodes). The nonzero entries can be stored in a rectangular array that is $N \times 9K$ provided we carry along an integer pointer array of the same size to select the column occupied by each nonzero element. Storing and manipulating even these reduced matrices accounts for a majority of the memory requirements of FMS2D (limiting run sizes to $(M, K) \leq (10^4, 10)$).

Sparse linear solution algorithms are abundant in the literature, and are the subject of intense active research. However, many ready-made state-of-the-art routines are available from, e.g. netlib[118], such as those included in the *SLAP*[119], *SPARSEPAK*[120], *LAPACK*[121] and *ITPACK*[122] packages. We made heavy use of both of these public domain packages early in the development of FMS2D, but ultimately created routines that were tailored for our specific needs, as described below.

Sparse methods are generally of two types: optimized Gaussian elimination, and Krylov iteration techniques. The standard Gaussian elimination algorithm is detailed in any introductory matrix algebra textbook (cf. [123]). With proper precautions, the method is very general and robust. However, it is also prohibitively slow and memory intensive. The forward substitution phase of the solution process produces a dense lower triangular matrix (where all elements must be stored), while the entire process requires approximately $2N^3/3$ arithmetic operations. Thus, doubling a 2D grid would lead to 64 times the computational work.

More useful Gaussian elimination-based solvers apply various element reordering algorithms to minimize the bandwidth about the diagonal of the Jacobian matrix[124, 125]. This in turn minimizes the degree of fill-in encountered during the forward decomposition phase of the standard Gaussian elimination procedure. Minimizing the fill-in leads to less arithmetic operations (by a factor of N^2/W^2, where W is the full band width[126]), and reduced storage requirements. The main drawback to these schemes for 2D PDE's is that the minimum achievable bandwidth scales like the width of the grid, ($\sim \sqrt{N}$). Here the computational work to solve the linear systems will scale as $\mathcal{O}N^2$, and for large systems, is beyond current hardware capabilities. Note however that for 1D problems (and for preconditioning to be described later), the grid width is unity, and banded-GE becomes quite efficient (banded-GE is presently used in most 1D edge plasma fluid codes).

Krylov-based methods appear to be much more attractive for large systems of linear equations arising from 2D conservation PDE's. The general idea is to cast the linear matrix equation into the form of a functional minimization. Any linear system, $Ax = b$ can be written as a quadratic functional, $\phi(x)$, defined by

$$\phi(x) = \frac{1}{2}x^T Ax - x^T b$$

where $b \in \mathcal{R}^N$ and $A \in \mathcal{R}^{N\times N}$. The minimum value of ϕ (if one exists) is $-b^T A^{-1}b/2$, achieved by setting $x = A^{-1}b$. Minimizing ϕ and solving $Ax = b$ are equivalent problems.

Why would we want to cast our linear equations into a form that looks harder to solve? The minimum of this functional can be found geometrically by sliding down the gradient of the surface in the $N-$dimensional subspace of the quadratic functional. The gradient is spatially dependent, so if we move some distance towards the solution, and then recalculate a new gradient direction, the procedure makes up an iterative technique that leads to the solution of our linear system. In a Conjugate Gradient (CG) method, the iterations are constructed via a simple three-term recurrence[2], which is most effective if the level sets of $\phi(x)$ are nearly circular in the *Krylov* subspace. The relevant parameter, $\kappa =$ condition number, is a measure of the ellipticity of these surfaces.

The method is optimal for elliptic problems, where $\kappa \approx 1$. A preconditioner can always be applied to transform the original general (non-elliptic) problem into one more suitable. An optimal preconditioner is constructed by approximating the original linear system with one much easier to solve, say $Kx = b$. for convection/diffusion problems in 2D, a reasonable K operator approximates A as the sum of 1D "slices". A set if block tridiagonal linear systems result which are easily solved using banded-GE techniques. Other possibilities involve partial decomposition of the matrix into Choelesky or LU type pieces (see any standard introductory

---

[2] For an excellent review of how this is done, the reader is referred to [127].

matrix analysis course textbooks), or even just simple Jacobi iterations.

For completeness, the BiCGSTAB algorithm is outlined below, as implemented in FMS2D. Here K_1 is a left preconditioner, K_2 is a right preconditioner, $K = K_1 K_2$ so that the modified linear system being solved is $\left(K_1^{-1} A K_2^{-1}\right)(K_2 x) = K_1^{-1} b$.

$$
\begin{aligned}
&r_0 = b - Ax_0, \bar{r}_0 = r_0 \\
&\rho_0 = \alpha = \omega_0 = 1, \\
&v_0 = p_0 = 0, \\
&\quad \rho_i = \bar{r}_0 r_{i-1}; \beta = (\rho_i/\rho_{i-1})(\alpha/\omega_{i-1}) \\
&\quad y = K^{-1} p_i; v_i = Ay, \\
&\quad \alpha = \rho_i/(\bar{r}_0 v_i); \\
&\quad s = r_{i-1} - \alpha v_i; \\
&\quad z = K^{-1} s; t = Az; \\
&\quad \omega_i = \left(K_1^{-1} t\right)\left(K_1^{-1} s\right) / \left(\left(K_1^{-1} t\right)\left(K_1^{-1} t\right)\right); \\
&\quad x_i = x_{i-1} + \alpha y + \omega_i z; \\
&\quad r_i = s - \omega_i t
\end{aligned}
$$

In typical applications, we set K_1 to the line-by-line sweep, once along each of the two coordinates, and set K_2 to just the identity matrix.

Matrix Scaling

Numerical accuracy and speed of the solution process is strongly governed by how well J is computed, and how well the algebraic system is iteratively solved. In our case, the un-scaled modified Braginskii set given earlier consists of variables that range in magnitude from 10^{-3} to 10^{20}. Elements of the residual vector can also have arbitrary magnitudes that depend entirely on the discretization technique.

The most general way to optimize the condition of the linear systems is to employ both row *and* column scaling before solution. Column scaling effectively equalizes the influence of the various states, according to a user specified normalization. The row scaling is carried out subsequently, by simply re-scaling each row to unity. The algebraic system solved then becomes

$$\delta \mathbf{s} \cdot J_{\text{new}} = -R_{\text{new}}$$

where $J_{\text{new}} = \mathbf{R}\left(\mathbf{C}\left(J\right)\right)$, and $R_{\text{new}} = \mathbf{R}\left(\mathbf{C}\left(R\right)\right)$. The scaling process is general, and equivalent in some sense, to non-dimensionalizing the equation set. The practical difference is that the user need not be concerned with scaling the input equations (which may be quite complex) or converting the results to useful quantities upon output from the code. There is however, a certain degree of flexibility in choosing the column scaling matrix that ultimately has a very large impact on the solution efficiency of the Newton algorithm. In fact, poor choice may itself cause the Newton iterations to diverge in the more strongly coupled plasma scenarios.

5.4 Detecting Convergence

The convergence tests applied to the nonlinear solution algorithm must not only be designed to detect a reasonable completion of the calculation but also when an iteration cycle gets "hung up" (as occurs near local minima on the solution manifold) and also when equations in the system are solved identically at the outset, so that further residual reduction is not possible. Additionally, we found it convenient to recognize when the iterations fail to make reasonable progress after some fixed number of iterations.

Convergence at a particular time step is obtained when any of the following three events occurs for each of the unknowns:

1. the Newton update, dx_i is small compared to a typical scale value of the

state, $\langle x \rangle_i$, or

2. the Newton update, is considerably smaller than the largest update in the present iteration cycle, dx_i^{max}, or
3. the residual has been significantly reduced from its maximum, r_i^{max}.

The scale values are returned as part of the model interface file, and therefore can depend dynamically on the state, state gradient, position and time (this scale value is also used to column-scale the Jacobian matrix). A vector of maximum values for the nodal residuals and iteration updates is maintained for the current iteration cycle to make these comparisons. The pseudo-code form of the test is as follows:

```
do $i = 1,N$
  $if\ [\ \ (r_i > f_1\, r_i^{max})$
    $\cap\, (dx_i > f_2\, dx_i^{max})$
    $\cap\, (dx_i > f_3\, \langle x \rangle_i)]\ \ then$ not converged
```

A flag is set, $converged\ =\ .TRUE.$, prior to entering this loop, and is reset if the condition is true for any component.

We also want to stop the iterations if the iteration number, $k > k_{max}$ the maximum allowable, or if

```
$if\ \ \ k = k_{check}$
  do $i = 1,N$
    $if\ [\,(r_i > f_4\, r_i^{max}) \cap (dx_i > f_5\, \langle x \rangle_i)\,]\ \ then$ stop iteration
```

This enforces that the iterations will be terminated if progress towards solution is too slow. f_i, k_{max} and k_{check} are arbitrary parameters, and the iteration is stopped if either of the these or the first set of situations are detected. For all calculations presented in this thesis, the values used were $f_1 = f_2 = f_3 = 10^{-6}, f_4 = f_5 = .125, k_{max} = 8$ and $k_{check} = 4$. No studies were performed to determine the relation these values had to the efficiency of the computations.

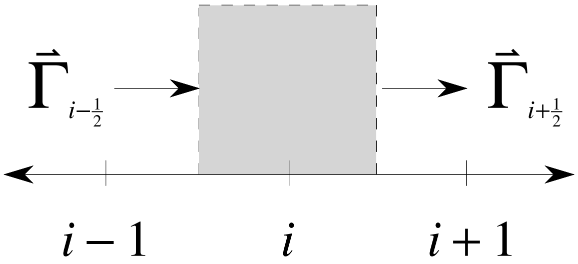

Figure 5.1. Advection on a discrete 1D grid

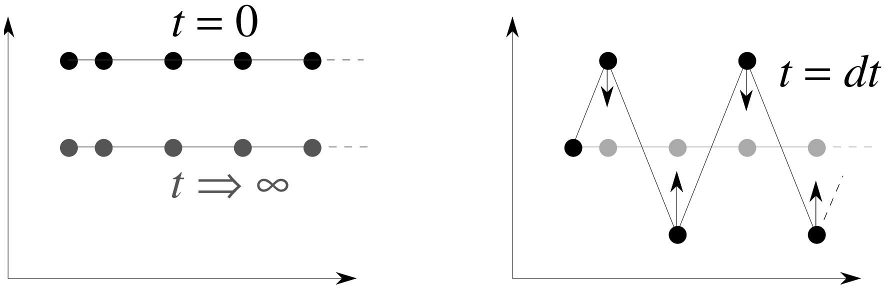

Figure 5.2. Transient profile oscillations due to boundary conditions

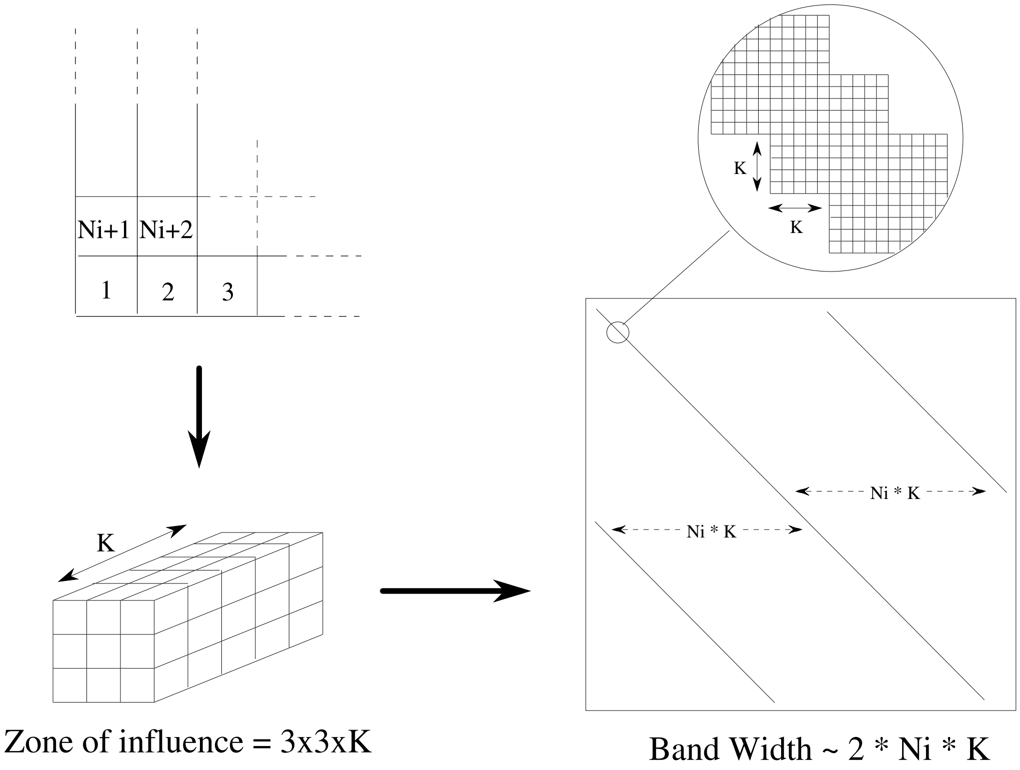

Figure 5.3. Local stencil and element numbering leading a to sparse Jacobian matrix. The picture depicts the locations of nonzero entries in the matrix.

Chapter 6

Anisotropic Dissipation

The behavior of numerical methods in the solution of diffusion problems is quite well understood in one space dimension, and even in two dimensions, provided the diffusivity tensor is isotropic. However, in our 2D application, diffusion flux along the poloidal flux surface (e.g. parallel electron heat conduction) can be orders of magnitude stronger than its cross field counterpart. Discretizing the resulting anisotropic diffusion flux can often lead to unexpected, non-physical results.

The numerical difficulties associated with anisotropic dissipation have been noted in many scientific fields. Applications include analysis of X-ray and MRI scans of bone tissue[128], edge and velocity field detection for the analysis of heart motion[129], determination of the principal stress axis orientations in composite material formation[130], earth mantle convection models[131, 132], polymer forming processes[133] and anisotropic mixing-length theories for fluid calculations involving turbulent eddies[134, 135]. In recent applications, researchers also apply anisotropic dissipation along flow streamlines to stabilize otherwise unphysically oscillatory numerical schemes(c.f. [136, 137]).

In order to discuss the problem, we make a couple of definitions. Consider the steady diffusion equation:

$$\nabla \cdot (\mathbf{D} \cdot \nabla u) = 0 \tag{6.1}$$

D is the diffusivity tensor, and u is the diffused quantity. We can write the differential operator of Eq. 6.1 in an arbitrary curvilinear coordinate system:

$$\frac{1}{h_x h_y} \frac{\partial}{\partial x_i} \left(D_{ij} \frac{h_x h_y}{h_i} \frac{\partial u}{\partial x_j} \right) = 0, \; i,j \in \{x,y\}$$

The tensor elements, D_{ij}, here are with respect to the x, y-coordinate system, and h_i are the corresponding metric coefficients. We are concerned primarily with elliptic operators such that there exists a rotation transformation that annihilates the off-diagonal diffusion terms, D_{ij}, where $i \neq j$ (see Figure 6.1). Often the physical diffusion properties of a system are most easily expressed in this rotated coordinate frame.

$$\begin{aligned} &\{x,y\} \Rightarrow \{s,r\} \ni \\ &\nabla \cdot (D \cdot \nabla u) = \frac{\partial}{\partial s}\left(D_{ss} \frac{\partial u}{\partial s} \right) + \frac{\partial}{\partial r}\left(D_{rr} \frac{\partial u}{\partial r} \right) = 0 \end{aligned} \tag{6.2}$$

Under these conditions, the diffusion operator is separable into two 1D-like problems (due to the lack of cross-diffusion). Numerical discretizations of this type of operator consists of simple extensions of the standard one-dimensional diffusion differences extensively studied in most introductory books on numerical analysis (c.f. [82]). Such a transformation is possible in models of the tokamak edge, and in fact is used exclusively in modern edge plasma fluid calculations. Here, numerical grids are generated to follow the principal axes of diffusion (i.e. along the poloidal flux surfaces), and the diffusive flux divergence is computed as a simple linear sum of the two contributing terms in Equation 6.2.

However, in view of the complex magnetic field structure and intricate bounding vessel designs of a tokamak edge, it is not always convenient to construct such numerical grids. We have motivation to investigate the feasibility of a more general scheme. In particular, we want that the scheme allow discrete cells to follow the shapes of bounding structures, which often are *not* aligned with the principal axes.

In the following chapter, we develop these techniques. We begin by describing important limitations of existing methods, and then give an example that demonstrates some pitfalls of a naively designed solution to the problem. After studying the pitfalls, a general solution becomes self-evident, and results in an optimized stencil for misaligned dissipation. We explore the limitations of this new stencil for extreme test cases, and then in situations closer to that expected in a tokamak edge plasma.

6.1 Motivation to Generalize for Edge Plasma

In previous computational edge modeling efforts, the same poloidal flux surface coordinates used to describe the physical model of the system, were also used to guide the location of the discrete domain sub-volumes. See Figure 6.2 for an example of one such grid, based on the flux surfaces of the single-null configuration of the current ITER design. As noted earlier, the choice clearly seems to be the most convenient because the edge plasma has distinct physical behavior along each of the two coordinates (i.e. the principal axes of diffusion lay along the poloidal coordinates). Unfortunately however, the poloidal flux surface direction rarely aligns with the bounding vessel in experimental and fusion grade reactors. Rather, the flux surfaces intersect such structures at grazing angles ($\leq 10°$) to minimize engineering difficulties caused by heat and particle fluxes (compare the "lower legs" of the left plot in Figure 6.2 with the target regions of Figure 2.5). Despite that the vessel geometry often has minimal direct impact on the magnetized plasma species, the neutral particles in the edge move independent of the field, and are emitted from the shaped vessel walls. The plasma/neutral coupling gives rise to an indirect geometry dependence of the charged species that has a *significant* impact over resulting plasma profiles. See [18, 100] for initial studies of this effect.

Poloidal flux surface gridding is problematic additionally near the X-point (see Figure 2.5), where the coordinates become singular and the computational domains become multiply connected. Multiply connected computational domains lose the inherent efficiency and simplicity associated with structured grids. They require complex mapping procedures and intricate techniques for marching through the system—making basic tasks, such as linear solutions and visualizing results much more complicated, and in this case inhibiting the use of more modern techniques.

Finally, we note that a great many numerical schemes exist for handling multidimensional convection problems (c.f. Reference [138] for a review, and Reference [139] for example applications). We will demonstrate later in the work, that such 'higher-order' convection schemes are indeed necessary in the fluid edge problem, yet few are easily extensible to complex multiply connected problem domains. Therefore, both to overcome practical complexity as well as fundamental limitations of present discretization schemes available for the tokamak edge fluid problem, we investigated methods for highly anisotropic dissipation that do not require an aligned computational mesh.

For simplicity, we construct a model problem wherein dissipation acts only along one of two locally orthogonal (curvilinear) coordinates ($\hat{r}$ and $\hat{s}$, where $\hat{s} = \vec{B}/\|\vec{B}\|$, and dissipation is solely along $\hat{s}$). Extension to anisotropic diffusion in multiple directions is straight forward, and will be included where appropriate for later applications.

We make the following definitions and assumptions. The orientation angle, θ, of the anisotropy field is specified as a function of x and y (the fundamental spatial coordinates, usually cartesian or cylindrical), measured counter-clockwise from the positive x-axis. x and y are specified with ξ and η (the grid coordinates), see Figure 6.3. The grid is fixed in time, and may be irregular, nonorthogonal, and/or not aligned with the poloidal flux surfaces. It is assumed that the grid varies smoothly in space so that grid transformation relations are smoothly differentiable[1]. We proceed to derive an optimization principal that will guide the development, and define the applicability range, of our new misaligned diffusion operator.

6.2 A Simple System with Problems

Note first, however, that a great deal of guidance can be obtained without the general machinery. To motivate a more complete analysis, we take $(\hat{r}, \hat{s}) := (\xi, \eta)$, the orientation angle, $\theta = 0$, $D_{\xi\xi} = 1$, and $D_{\eta\eta} = 0$. Here, the Equation 6.1 takes a rather simple form, namely

$$\frac{\partial^2 u}{\partial \xi^2} = 0 \qquad (6.3)$$

Using standard numerical techniques, the domain is split into finite trapezoidal zones, and the conserved states are monitored at the centers of these cells. We denote u^* the set of values of u at the (equal spaced) mesh locations. The discrete

---

[1] Rigorously, “smooth” grid spacings change by somewhat less than factors of two over the width of the stencil. This requirement is especially important in maintaining accuracy levels in profiles with significant structure, and can be relaxed in regions of smooth behavior. Finite element schemes do not have this requirement because adjacent regions are never “sampled”, but they have their own pitfalls and requirements. The reader is referred to [14] for a short review.

operator Υ will operate on u^* consistent with Equation 6.3. A minimum practical set of neighbors in two dimensions that allows direct communication of information from all directions includes eight surrounding nodes. If we expect the numerical and differential operators to possess identical symmetries across the coordinate axes, the difference stencil approximating Equation 6.3 must be of the form:

$$\Upsilon\left(u^*_{\xi,\eta}\right) = \frac{\partial^2 u^*_{\xi,\eta}}{\partial \xi^2} = \frac{\partial^2 u^*_{\xi,\eta}}{\partial x^2} = \frac{\partial^2 u^*_{\xi,\eta}}{\partial \hat{s}^2}$$

$$= \begin{bmatrix} a & -2a & a \\ b & -2b & b \\ a & -2a & a \end{bmatrix} : \begin{bmatrix} u_{\xi-1,\eta+1} & u_{\xi,\eta+1} & u_{\xi+1,\eta+1} \\ u_{\xi-1,\eta} & u_{\xi,\eta} & u_{\xi+1,\eta} \\ u_{\xi-1,\eta-1} & u_{\xi,\eta-1} & u_{\xi+1,\eta-1} \end{bmatrix}$$

where a and b are free (positive) parameters, but are scaled to create an operator consistent with Equation 6.3. The discrete state matrix above is formed at all internal nodes, (ξ, η) ($u_{\xi,\eta}$ is the value of the discrete state function, u^*, at the $(x_{\xi,\eta}, y_{\xi,\eta})$ specified by (ξ, η)) and the matrix operation is the scalar product function, $A : B = A_{ij}B_{ij}$.

In this very simple system, we find that any values for (a, b) other than $(0, 1)$ lead to results that violate The Maximum Principle (see [140] for example). The Maximum Principle states that profiles that obey elliptic differential equations will have extrema only on the boundary. Unwise choices of (a, b) lead to internal oscillations that satisfy the difference equations, but violate the dissipation physics being modeled.

For example, if we construct the operator, Υ, by centrally differencing a bilinear profile, as suggested in [14], a simple analysis for a uniform mesh yields $(a, b) = (\frac{1}{4}, \frac{1}{2})$. This discretization was applied to the Dirichlet case, where the boundary conditions had discontinuities along the sides parallel to the anisotropy—the two boundary values are set arbitrarily to .55 and 1. The exact internal solution is a plane, $u^* = x$, because there is no y-directed diffusivity to communicate the $y = 0$ or $y = 1$ boundary information into the domain. Along the boundaries, $u^* = .55$ if $x < .5$, and $u^* = 1$ if $x \geq .5$. The solution to this discrete system using the bilinear stencil is shown as the left plot in Figure 6.4. We obtain a more suitable discretization setting $(a, b) = (0, 1)$. The resulting profile is shown on the right of the figure, and produces the exact (i.e. nonoscillatory) solution. The disparity in results for two convergent schemes indicates the importance of proper diffusion scheme design.

6.3 General Stencil Anaysis

That the bilinear stencil produces internal oscillations is not a surprise if one studies the individual stencil elements. Since negative weight is given to some of the neighboring cells in the stencil, there will always be the possibility that an internal state value will not have a value between the extrema of its surrounding neighbors (i.e. potentially violate The Maximum Principle). Oscillations are particularly likely if the boundary conditions contain a discontinuity.

We set out then to design a difference scheme without such negative external stencil elements, and begin with a few additional required definitions. For the moment we retain the simplicity of a uniform mesh $(x, y) := (\xi, \eta)$, though we accommodate a (constant) direction, $\theta \neq 0$ of rotation of the principal diffusion axes from the x-coordinate. We also define $D_{ss} := D$, and $D_{rr} := d$ $(\leq D)$, respectively the diffusivities along the *major* and *minor* principal axes. The diffusivity tensor, $\mathbf{D}$, is rotated from the field-aligned coordinates to (x, y), following the rules for rank-2 tensors:

$$\mathbf{D} = \begin{bmatrix} D & 0 \\ 0 & d \end{bmatrix}_{sr} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}_{xy} = \begin{bmatrix} D\cos^2\theta + d\sin^2\theta & (D-d)\sin\theta\cos\theta \\ (D-d)\sin\theta\cos\theta & D\sin^2\theta + d\cos^2\theta \end{bmatrix}_{xy}$$

Note that for elliptic (i.e. *positive definite*) problems, $Eu := Au_{xx} + 2Bu_{xy} + Cu_{yy}$,

$$B^2 \leq AC. \tag{6.4}$$

(An operator is positive definite if it has all positive eigenvalues). We seek a *positive-type*[141] scheme

$$\Upsilon\left(u^*_{\xi,\eta}\right) = \begin{bmatrix} a_{nw} & a_n & a_{ne} \\ a_w & a_c & a_e \\ a_{sw} & a_s & a_{se} \end{bmatrix} : \begin{bmatrix} u_{\xi-1,\eta+1} & u_{\xi,\eta+1} & u_{\xi+1,\eta+1} \\ u_{\xi-1,\eta} & u_{\xi,\eta} & u_{\xi+1,\eta} \\ u_{\xi-1,\eta-1} & u_{\xi,\eta-1} & u_{\xi+1,\eta-1} \end{bmatrix}$$

such that the *external* stencil elements, $a_{nw}, a_n, a_{ne}, a_w, a_e, a_{sw}, a_s, a_{se}$ all have a sign opposite to that of the *central* stencil element, a_c. If such a scheme exists, then the numerical approximation to the elliptic operator will obey a discrete form of The Maximum Principal obeyed by the continuum elliptic operator. Local minima and maxima will not be allowed to exist in solution profiles within the grid length of the discrete stencil. On scales larger than the stencil, the continuum Maximum Principal holds, so that a global Maximum Principal will exist that precludes the non-physical, oscillatory solution profiles to the discrete equation, $\Upsilon = 0$. We present the following proposition:

> **Proposition:** Suppose $\Upsilon(u^*)$ is a *positive-type* approximation to the elliptic equation, $Eu := Au_{xx} + 2Bu_{xy} + Cu_{yy}$, and is represented by the above definition for the central and internal coefficients a_i on a uniform grid in x and y. Then the discrete form of Equation 6.4 holds. In fact, $B \leq \min(A, C)$.

To prove this proposition, we realize that the difference approximating the Laplacian must be symmetric about the x and y axes,

$$\Upsilon(x) = \Upsilon(-x), \ \ , \Upsilon(y) = \Upsilon(-y)$$

This reduces the possible number of entries so that the operator must have the form

$$\begin{bmatrix} a & b & c \\ d & e & d \\ c & b & a \end{bmatrix} : \begin{bmatrix} u_{\xi-1,\eta+1} & u_{\xi,\eta+1} & u_{\xi+1,\eta+1} \\ u_{\xi-1,\eta} & u_{\xi,\eta} & u_{\xi+1,\eta} \\ u_{\xi-1,\eta-1} & u_{\xi,\eta-1} & u_{\xi+1,\eta-1} \end{bmatrix}$$

Further, the operator must satisfy

$$\Upsilon\left(x^2\right) = 2A$$

$$\Upsilon(xy) = 2B$$

$$\Upsilon\left(y^2\right) = 2C$$

where

$$A = D\cos^2\theta + d\,\sin^2\theta$$

$$B = (D-d)\,\sin\theta\,\cos\theta$$

$$C = D\sin^2\theta + d\,\cos^2\theta$$

Using that the mesh is has $\Delta x = \Delta y = 1$ evaluate Υ at the origin for the 3 profiles to obtain

$$2A = 2(a+b+c)$$

$$2B = 2(c-a)$$

$$2C = 2(a+b+c)$$

Now, since all stencil elements a, b, c, d must be positive, $c - a$ *must* be less than $c + a$, and is certainly less than $a + b + c$ if b is any positive constant. The proposition follows.

It is useful to determine what these conditions imply, with respect to the allowable anisotropy. Applying the proposition to the definition of A, B, C, we find

$$(D-d)\,\sin\theta\,\cos\theta \le D\cos^2\theta + d\sin^2\theta$$

$$(D-d)\,\sin\theta\,\cos\theta \le D\sin^2\theta + d\cos^2\theta$$

or, defining the ratio, $r = d/D \le 1$ (remember that $d \le D$) and rearranging,

$$\begin{aligned} r_{max} &\ge \frac{(\sin\theta - \cos\theta)}{(\sin\theta + \cos\theta)}\cot\theta, \text{and} \\ r_{max} &\ge \frac{(\cos\theta - \sin\theta)}{(\cos\theta + \sin\theta)}\tan\theta \end{aligned} \tag{6.5}$$

In order that both constraints, Equation 6.5, be satisfied, the diffusivity ratio, r, must lay within the shaded zone of Figure 6.5, which extends to $r = 1$ (isotropic dissipation). Notice that along grid directions, any ratio can be accommodated, as $r = 0$ approaches a one-dimensional problem in aligned coordinates. The most restrictive case occurs when the principal directions cut between the grid points, where the maximum dissipation ratio is approximately 5.8:1.

The results of the analysis are that in order to retain a 9-point *positive-type* numerical scheme when discretizing anisotropic dissipation arbitrarily oriented to the discrete grid, we cannot have principal diffusivities with a ratio of greater that about 5.8:1. In fact, the following shows that there will be a finite maximum representable anisotropy ratio no matter how large the discrete stencil, unless the anisotropy is aligned in one of the directions of the discrete grid:

> **General Discrete Anisotropic Dissipation:** Assume a local perturbation from equilibrium in a anisotropically diffusing medium. The potential energy of this disturbance is to be dissipated along the *principal* dissipation axes. Since a discrete diffusing system can only dissipate perturbations along the *mesh* directions, profile adjustment must affect both components of the gradient when the mesh is not aligned. However, the split of energy transport, forced by the orientation of the numerical grid, does not generally agree with the ratio of physical diffusivities. In general then, there will be an imbalance of potential energy with the available channels for dissipation, leading to residual energy in the system at steady state (in the form of internal maxima that violate the continuum Maximum Principle).

For our application, this is a rather bleak state of affairs; we must either align our grid to the anisotropy (which was shown earlier to fail some basic modeling requirements), or enlarge the stencil size to increase to number of directions available for alignment. The latter is suggestion not practical; numerical solution of the plasma models on five and nine point grids already consume the presently available computer CPU and memory limits.

In the next section, we design an alternative diffusion operator that aligns optimally with the discrete grid whenever possible. And despite that in some regions the resulting grid may be misaligned, we find the resulting operator quite useful for arbitrarily oriented anisotropic dissipation, even for diffusivity ratios of the order 1000:1, and greater. As a side note, we point out that the technique is useful for applying the first order streamwise artificial dissipation used to stabilize the convection terms in our model. Application of the artificial dissipation is covered separately elsewhere.

6.4 Fixed Constant Misaligned Diffusion

To develop a stencil-optimizing procedure, we continue with the assumption that the principal dissipation axes, (s, r), are rotated a fixed angle, θ, from the (x, y) system. Here, the diffusion equation is written

$$\nabla \cdot (\mathbf{D} \cdot \nabla u) = D_{xx}\frac{\partial^2 u}{\partial x^2} + 2D_{xy}\frac{\partial^2 u}{\partial x \partial y} + D_{yy}\frac{\partial^2 u}{\partial y^2} = 0 \tag{6.6}$$

Applying the lessons learned in the previous section, the two second derivatives are discretized with the following operators:

$$\frac{\partial^2}{\partial x^2} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix} \qquad \frac{\partial^2}{\partial y^2} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{6.7}$$

The mixed derivative may in general, be written as the following linear combination:

$$\frac{\partial^2}{\partial x \partial y} = \frac{c_1}{2}\begin{bmatrix} 0 & -1 & 1 \\ -1 & 2 & -1 \\ 1 & -1 & 0 \end{bmatrix} + \frac{c_2}{2}\begin{bmatrix} -1 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -1 \end{bmatrix} \tag{6.8}$$

where $c_1 + c_2 = 1$.

Since the mesh directions may also be constructed diagonally, we compute c_1 and c_2 to give the correct limiting results for each of the four grid directions ($\theta = 0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}$. Using simple trigonometric relations, the results are:

$$\Upsilon\left(u^*\right) = \left(D\cos^2\theta + d\sin^2\theta\right)\begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix} + \left(D\sin^2\theta + d\cos^2\theta\right)\begin{bmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$+2\left(D - d\right)\sin\theta\cos\theta\left(\frac{\cos^2\left(\theta - \frac{\pi}{4}\right)}{2}\begin{bmatrix} 0 & -1 & 1 \\ -1 & 2 & -1 \\ 1 & -1 & 0 \end{bmatrix} + \frac{\sin^2\left(\theta - \frac{\pi}{4}\right)}{2}\begin{bmatrix} -1 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -1 \end{bmatrix}\right)$$

As required, this combination reduces to the central aligned *positive-type* stencils such as:

$$\Upsilon\left(u^{*}\right)=\left[\begin{array}{rrr} 0 & 0 & \frac{1}{2} \\ 0 & -1 & 0 \\ \frac{1}{2} & 0 & 0 \end{array}\right]$$

when $\theta = n\frac{\pi}{4}$, $n = 1, 5, 9 \ldots$ (and the intuitive extension for $\theta = n\frac{\pi}{4}$, $n = 3, 7, 11 \ldots$)

To test the foregoing, we use a setup similar to the one used before. Since we designed the scheme to give the "correct" stencil when $\theta = 0$ and $\frac{\pi}{2}$ there is no need to test these cases (see previous sections for such results). The anisotropy is instead set skew to the mesh at $\theta = \frac{\pi}{8}$ and $\frac{\pi}{4}$ (and the boundary values are adjusted, as illustrated in the figure). Results are shown in Figure 6.6. Note that, as advertised, the directions that align with the mesh give a nodally exact solution, while those in between show the characteristic oscillations that result from negative external stencil weight values. Also note that $\theta = -\frac{\pi}{8}$ is the *worst* direction possible—the oscillations have maximum amplitude when the anisotropy cuts between representable directions. Finally note that these two, along with the $\frac{\pi}{2}$ case from the previous experiment, span the entire range of (constant) angles possible.

For comparison, we repeated the experiment using the simpler cross difference expression (in place of the complex term in brackets of the third term in (6.4)):

$$\frac{\partial^2}{\partial x \partial y} = \frac{1}{4}\left[\begin{array}{rrr} -1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & -1 \end{array}\right]$$

obtained by setting $c_1 = c_2 = \frac{1}{2}$ (and thereby precluding angle dependence), and also by switching the two mixed differences in the full expression (effectively giving the "worst" consistent, second order mixed difference). Because the rotating stencil ideas are employed, the discontinuity on the boundary does not lead to point-to-point oscillations. However, the lack of optimization leads to less than ideal results for the $-\frac{\pi}{4}$ cases, as shown in Figure 6.7. Note that both give comparable results. The corresponding $-\frac{\pi}{8}$ cases are shown in Figure 6.8. Interestingly, both give results that are comparable to those of the optimized stencil for the same case.

Finally, we notice that if we require a *positive-type* scheme, and the physical dissipation is unidirectional (diffusivity given by the scalar, D), then our derived constraints, Equation 6.5, give that an *artificial anisotropic dissipation* of magnitude $d = rD$ must be added to the system. The plot on the left in Figure 6.9 shows the standard result at $\theta = -\frac{\pi}{8}$ exhibiting the characteristic oscillations. The figure on the right shows the result when the minimal transverse diffusion is added. Note that all the solutions presented at $-\frac{\pi}{8}$ exhibit excessive smearing of the interface; adding artificial cross diffusion keeps the operator positive, preventing internal extrema, but at the cost of even further smearing of the solution. Based on these results, there seems little hope that the operator will fulfill our goals. In the next section, however, we present results to the contrary. We'll see that this section focused on the worse case ($\theta = n\frac{\pi}{8}, n = 1, 3\ldots$), and that the scheme gives quite acceptable results for general $\theta = \theta(x, y)$.

6.5 Error Analysis for Rotating Stencil

It is informative to compute the order of convergence of this scheme using standard Taylor-series analysis methods. Here, we write the state values at neighboring points in the stencil as an expansion about the central point:

$$
\begin{aligned}
u_{i,j} &= f_1 u_{0,0} + f_2 \Delta_x u_x + f_3 \Delta_y u_y + \frac{\Delta_x^2}{2} u_{xx} + f_4 \frac{\Delta_x \Delta_y}{2} u_{xy} + f_5 \frac{\Delta_y^2}{2} u_{yy} \\
&+ f_7 \frac{\Delta_x^3}{6} u_{xxx} + f_8 \frac{\Delta_x^2 \Delta_y}{6} u_{xxy} + f_9 \frac{\Delta_x \Delta_y^2}{6} u_{xyy} + f_{10} \frac{\Delta_x^3}{6} u_{yyy} \\
&+ f_{11} \frac{\Delta_x^4}{24} u_{xxxx} + f_{12} \frac{\Delta_x^3 \Delta_y}{24} u_{xxxy} + f_{13} \frac{\Delta_x^2 \Delta_y^2}{24} u_{xxyy} + f_{14} \frac{\Delta_x \Delta_y^3}{24} u_{xyyy} + f_{15} \frac{\Delta_y^4}{24} u_{yyyy}
\end{aligned}
$$

The coefficients, f_k, are given in the following table for the column i, and row j (the centroid lays at (row,column)=$(0,0)$):

i,j	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}
$-1,-1$	$+1$	-1	-1	$+1$	$+1$	$+1$	-1	-1	-1	-1	$+1$	$+1$	$+1$	$+1$	$+1$
$-1,0$	$+1$	-1	0	$+1$	0	0	-1	0	0	0	$+1$	0	0	0	0
$-1,+1$	$+1$	-1	$+1$	$+1$	-1	$+1$	-1	$+1$	-1	$+1$	$+1$	-1	$+1$	-1	$+1$
$0,-1$	$+1$	0	-1	0	0	$+1$	0	0	0	-1	0	0	0	0	$+1$
$0,0$	$+1$	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$0,+1$	$+1$	0	$+1$	0	0	$+1$	0	0	0	$+1$	0	0	0	0	$+1$
$+1,-1$	$+1$	$+1$	-1	$+1$	-1	$+1$	$+1$	-1	$+1$	-1	$+1$	-1	$+1$	-1	$+1$
$+1,0$	$+1$	$+1$	0	$+1$	0	0	$+1$	0	0	0	$+1$	0	0	0	0
$+1,+1$	$+1$	$+1$	$+1$	$+1$	$+1$	$+1$	$+1$	$+1$	$+1$	$+1$	$+1$	$+1$	$+1$	$+1$	$+1$

From this table, we deduce the following:

$$
u_{xy} = \frac{1}{\Delta_x \Delta_y} \begin{bmatrix} 0 & -1 & +1 \\ -1 & +2 & -1 \\ +1 & -1 & 0 \end{bmatrix} - \frac{\Delta_x^2}{12} u_{xxxy} - \frac{\Delta_x \Delta_y}{12} u_{xxyy} - \frac{\Delta_y^2}{12} u_{xyyy}
$$

$$
u_{xy} = \frac{1}{\Delta_x \Delta_y} \begin{bmatrix} -1 & +1 & 0 \\ +1 & -2 & +1 \\ 0 & +1 & -1 \end{bmatrix} - \frac{\Delta_x^2}{12} u_{xxxy} + \frac{\Delta_x \Delta_y}{12} u_{xxyy} - \frac{\Delta_y^2}{12} u_{xyyy}
$$

$$
u_{xx} = \frac{1}{\Delta_x^2} \begin{bmatrix} 0 & 0 & 0 \\ +1 & -2 & +1 \\ 0 & 0 & 0 \end{bmatrix} - \frac{\Delta_x^2}{12} u_{xxxx}
$$

$$u_{yy} = \frac{1}{\Delta_y^2} \begin{bmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{bmatrix} - \frac{\Delta_y^2}{12} u_{yyyy}$$

These expressions are inserted into Equation 6.4, and after a bit of tedious algebra, we obtain (for constant $\mathbf{D}$)

$$\begin{aligned} \nabla \cdot (\mathbf{D} \cdot \nabla u) &= 0 = \Upsilon(u^*) \qquad (6.9) \\ & -\frac{1}{24} \Delta_x \Delta_y (D-d) u_{xxyy} \sin^2 2\theta \\ & -\frac{1}{24} \Delta_x^2 \left[2 \left(D \cos^2\theta + d \sin^2\theta \right) u_{xxxx} + (D-d) \sin 2\theta u_{xxxy} \right] \\ & -\frac{1}{24} \Delta_y^2 \left[2 \left(D \sin^2\theta + d \cos^2\theta \right) u_{yyyy} + (D-d) \sin 2\theta u_{xyyy} \right] \end{aligned}$$

There are several things to note about the behavior of the error term in Equation 6.10. When $d = 0, D = 1$ and $\theta = 0$, we recover the the error expression given above for $\partial^2 u / \partial x^2$. Also note that the leading error term scales as Δ_x, i.e. the method is first order accurate unless $\theta = n\frac{\pi}{2}$, $n = 0, 1, 2 \ldots$ (anisotropy along the grid), in which case it is second order accurate, as expected. If $\theta = \frac{\pi}{4}$, the error term is first order in both grid directions, but if $\Delta_x = \Delta_y$, then the method is second order in the diagonal mesh spacing, $\Delta_x\sqrt{2}$. Finally, we note that if the diffusion is isotropic (i.e. $d = D$) then the second order error scales as the *fourth* derivative of the solution, and is therefore quite small in magnitude for most physical profiles.

6.6 Convergence Order Verification

With these limitations at hand, we proceed to evaluate the quality of the convergence when just part of the domain contains anisotropy that is not representable on a discrete grid. We expect this situation when the orientation is specified as a general function in space. First, qualitative results are shown, then rigorous numerical accuracy is checked, and found to agree with the derivation.

We look first at extreme, realistic scenarios, where profiles often may not be fully resolved. We model these scenarios by setting the axis of anisotropy to trace out quarter-circles about the origin, through a domain, $[0,1]^2$, so that all angles are sampled. Essential (i.e. Dirichlet) boundary conditions are specified along the unit square, and the discretized diffusion equation is solved in steady state. Unresolved solution profiles are simulated as discontinuities in the zeroth and first derivative along the boundary of our domain. We can identify this particular setup as one to approximate the singly connected gridding of an X-point region in a tokamak plasma (see Figure 6.10), where the discontinuities represent strong radial gradients in the temperature profiles near the separatrix of a diverted tokamak plasma. The two extreme cases will help to give us an indication of how the scheme will perform in realistic cases.

Along the back and end of our unit box, $(y = \eta = 1)$, and $(x = \xi = 1)$, we fix u to unity. Along the front $(y = \eta = 0)$, and start $(x = \xi = 0)$, we connect boundary values at the first two walls and linearly or discontinuously bring the BC values toward 0.5 at the front corner. We assume a constant, unit-valued parallel diffusivity and the exact solutions are quarter circle contours of constant u. The optimized stencil treats the boundary exactly, since the anisotropy is orthogonal to the walls.

The results for a coarse mesh (12x12) are plotted in Figure 6.11. Qualitatively, boundary irregularities are propagated into the domain with considerable integrity, especially in view of the coarse mesh (or, lack or resolution). Internal representation of the boundary discontinuities is oscillatory, but is well contained within three grid points of the position of the exact solution, regardless of resolution (these cases were run to 321x321, though results are omitted–the coarse results are easier to analyze). We take this as a good indication of the capabilities of the our discretization—the discontinuous profiles represent the most extreme circumstances we expect to encounter.

We proceed to compute numerically the convergence order throughout the domain for this spatially dependent anisotropy. If the result is everywhere greater than one, the method is said to be converging to the continuum solution of the associated partial differential equation The order of convergence is found from a set of discrete solutions via:

$$\text{ConvergenceOrder} = \frac{\log\left(\frac{u_3^* - u_2^*}{u_2^* - u_1^*}\right)}{\log(2)} \tag{6.10}$$

where u_n^* is the solution to the discrete form of equation(6.1) on a uniform grid level, n. Each successive level in the sequence of grids is formed by equally subdividing the coarser level (i.e. the finer grid is partially coincident with the coarser grid, and the order is computed wherever the three grids coincide).

We must use smooth profiles for these calculations, to ensure that the higher order derivatives in 6.10 exist. Instead of the discontinuous profiles used above, we fix the boundary conditions to e^r, $r = \sqrt{x^2 + y^2}$, where $x = (i-1)/(N_i-1), y = (j-1)/(N_j-1)$. The steady solution on square grids, $N_i (= N_j) = (11, 21, 41, 81, 161)$, and we compute the convergence using sets of 3 grids, (e.g. 41x41, 81x81 and 161x161). Only results on the 41x41 grid are shown. (Note the convergence analysis is invalid at the boundary, as the exact solution is always obtained–the order at those points is arbitrarily set to zero in the plots). The convergence order is computed for $D = 1$ and $d = (0, 0.2, 0.8, 1.0)$, and shown in Figure 6.12. Since we refine both in x and y, we expect from Equation 6.10 to obtain second order convergence. The accuracy order is plotted over all the mesh points, and in all cases averages very close to 2, as expected. The strange spikes and undulations in the surfaces are not yet understood, and may merely be the results of finite precision arithmetic used in the graphics package to compute Equation 6.10.

6.7 X-Point Simulation

In the previous section, we have found our discretization to be second order convergent on general anisotropy orientations, regardless of the diffusivity ratios–a result not expected after the cursory analysis of fixed anisotropy directions. But before proposing that this method be used in a full tokamak edge modeling code, we present a final diffusion test contrived to simulate the X-point region of a diverted tokamak plasma (see Figure 6.13). Typically, this problem would be discretized along the field using *aligned* meshes (such as the one in Figure 6.2) to minimize non-physical cross dissipation generated by the numerics. We try our new misalignment capabilities to alternatively solve the anisotropic diffusion problem on a cartesian, equal-spaced grid. We then judge the usefulness of our scheme in this situation by comparing the computed results to known solutions.

We are free to construct any representative problem for illustrative purposes. But notice that for purely anisotropic dissipation, if a flux surface line connects two points along the boundary having the same value, the solution profile will be constant along that line. The quality can easily be judged by comparing solution profile contours with the underlying flux surfaces.

We work on the unit square, and set the dissipation orientation angle to trace out a simulated X-point by following levels sets of a hyperbolic paraboloid (see Figure 6.14). Linear Dirichlet conditions are set so that each field line is to have its own value. A contour plot of our solution should agree with the imposed field lines.

Qualitatively, the scheme performs well over most the domain, as evidenced by the closely similar line shapes of the two plots. However, near the separatrix at the X-point, the field lines change direction abruptly, so that the $(s, r) \Rightarrow (x, y)$ transformation is not smooth enough for our discretization to be consistent. As a result, the profile is smeared near the X-point, and there is non-physical communication across the separatrix. Figure 6.15 shows the resulting order of convergence. Several interesting features are apparent. The order of convergence in the zone of rapid field direction change is close to one, down from the value of two most everywhere else. Also, the errors seem to propagate along the field separatrix, reducing order there too. It appears that convergence order reductions can be propagated along the primary axes if there is significant corresponding diffusion. Also, the spikes in this order computation are not yet understood. Likely, they are due to a spurious cancellation of higher order derivatives of the solution function. In that case the leading error terms may be missing in those areas, and the spikes result from machine precision limitations.

The smearing across the X-point is of concern to us. Without maintaining the integrity of anisotropic character of our equations, the scheme will not be usual in edge fluid computations. However, at the X-point in a tokamak the magnetic field line is purely toroidal. In the poloidal plane, flow there may be stochastic(e.g. [142]) and may be describable by isotropic diffusion[143]. In fact, the magnetic field *pitch* $(= \frac{B_x}{B})$ is a spatially varying function, ranging in value from $\frac{B_x}{B} = 0$ at the X-point to values near $\frac{B_x}{B} = 0.1$ at the horizontal midplane of the tokamak[2]. See Figure 6.16 for a schematic picture of this geometry.

We simulate this additional effect by contriving a *pitch* function, $\frac{B_x}{B} = r_X/r_M$, where r_X is the distance in the poloidal plane to the X-point, and r_M is a scale distance, to the horizontal midplane, for example. The resulting diffusivity tensor is

$$\begin{bmatrix} D\frac{B_p}{B_T} + d\frac{B_D}{B_T} & 0 \\ 0 & d \end{bmatrix}_{sr} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{yx} & D_{yy} \end{bmatrix}_{xy}$$

When $\frac{B_x}{B} = 0$, the diffusion is isotropic $(D_{ss} = D_{rr})$. We set $D = 1$ and $d = .01$ for this test problem. Note that this process provides a physical mechanism to communicate diffusive information across the X-point, but is not the rigorous result that one obtains by carrying our the correct coordinate tranformation. In adding this feature however, we lose the simple contour line approach to checking our solution. We therefore can only show that the scheme produces a convergent result with the correct qualitative features. Notice that the convergence order near the X-point (and along the separatrices) is significantly higher with this addition. This is expected; the first attempt lost accuracy trying to discretize a rapidly changing anisotropy. The new version approaches isotropy when the direction begins to change too rapidly. Notice also that the mysterious spikes remain because the form of $\frac{B_x}{B}$ is such as to not effect the profiles in these regions.

---

[2] The actual magnetic field structure used in a typical tokamak edge plasma calculation is obtained from MHD field reconstruction codes based on experimental or design data from the magnetic coils

6.8 General Grids

For completeness, our optimized operator, 6.4 is expanded to accommodate nonuniform and/or nonorthogonal gridding. We can then satisfy all the minimum requirements set forth at the beginning of this chapter.

The transformation equations result directly from application of the chain rule to (6.1). Formally (here subscripts denote differentiation),

$$\frac{\partial}{\partial x} = \xi_x \frac{\partial}{\partial \xi} + \eta_x \frac{\partial}{\partial \eta} \qquad \text{and} \qquad \frac{\partial}{\partial y} = \xi_y \frac{\partial}{\partial \xi} + \eta_y \frac{\partial}{\partial \eta} \tag{6.11}$$

while the inverse transformation equations are,

$$\frac{\partial}{\partial \xi} = x_\xi \frac{\partial}{\partial x} + y_\xi \frac{\partial}{\partial y} \qquad \text{and} \qquad \frac{\partial}{\partial \eta} = x_\eta \frac{\partial}{\partial x} + y_\eta \frac{\partial}{\partial y}$$

and if $J = x_\xi y_\eta - x_\eta y_\xi$, these may may be combined to give,

$$\begin{array}{ll} \xi_x = y_\eta / J & jx = -y_\xi / J \\ \xi_y = -x_\eta / J & jy = x_\xi / J \end{array}$$

Putting these into (6.11), we get the following operators,

$$\begin{aligned}
\frac{\partial}{\partial x} &= \frac{y_\eta}{J}\frac{\partial}{\partial \xi} - \frac{y_\xi}{J}\frac{\partial}{\partial \eta} \\
\frac{\partial}{\partial y} &= -\frac{x_\eta}{J}\frac{\partial}{\partial \xi} + \frac{x_\xi}{J}\frac{\partial}{\partial \eta} \\
\frac{\partial^2}{\partial x^2} &= \frac{y_\eta}{J}\frac{\partial}{\partial \xi}\left(\frac{y_\eta}{J}\frac{\partial}{\partial \xi} - \frac{y_\xi}{J}\frac{\partial}{\partial \eta}\right) - \frac{y_\xi}{J}\frac{\partial}{\partial \eta}\left(\frac{y_\eta}{J}\frac{\partial}{\partial \xi} - \frac{y_\xi}{J}\frac{\partial}{\partial \eta}\right) \\
\frac{\partial^2}{\partial x \partial y} &= -\frac{y_\eta}{J}\frac{\partial}{\partial \xi}\left(\frac{x_\eta}{J}\frac{\partial}{\partial \xi} - \frac{x_\xi}{J}\frac{\partial}{\partial \eta}\right) + \frac{y_\xi}{J}\frac{\partial}{\partial \eta}\left(\frac{x_\eta}{J}\frac{\partial}{\partial \xi} - \frac{x_\xi}{J}\frac{\partial}{\partial \eta}\right) \\
\frac{\partial^2}{\partial y^2} &= \frac{x_\eta}{J}\frac{\partial}{\partial \xi}\left(\frac{x_\eta}{J}\frac{\partial}{\partial \xi} - \frac{x_\xi}{J}\frac{\partial}{\partial \eta}\right) - \frac{x_\xi}{J}\frac{\partial}{\partial \eta}\left(\frac{x_\eta}{J}\frac{\partial}{\partial \xi} - \frac{x_\xi}{J}\frac{\partial}{\partial \eta}\right)
\end{aligned} \tag{6.12}$$

The diffusion equation (6.1), is then rewritten in terms of operations on ξ and η:

$$\frac{1}{J^2}\left[y_\eta^2 D_{xx} + x_\eta^2 D_{yy} - 2x_\eta y_\eta D_{xy}\right]\frac{\partial^2 u}{\partial \xi^2} \tag{6.13}$$

$$
\begin{aligned}
&+ \quad \frac{1}{J^2}\left[y_\xi^2 D_{xx} + x_\xi^2 D_{yy} - 2x_\xi y_\xi D_{xy}\right]\frac{\partial^2 u}{\partial \eta^2} \\
&- \quad 2\frac{1}{J^2}\left[y_\xi y_\eta D_{xx} + x_\xi x_\eta D_{yy} - \left(x_\eta y_\xi + x_\xi y_\eta\right) D_{xy}\right]\frac{\partial^2 u}{\partial \xi \partial \eta} \\
&+ \quad \frac{1}{J}\left[y_\eta\left(\frac{\partial D_{xx}}{\partial x} + \frac{\partial D_{xy}}{\partial y}\right) - x_\eta\left(\frac{\partial D_{yy}}{\partial y} + \frac{\partial D_{xy}}{\partial x}\right)\right. \\
&\qquad + D_{xx}\left[y_\eta\frac{\partial}{\partial \xi}\left(\frac{y_\eta}{J}\right) - y_\xi\frac{\partial}{\partial \eta}\left(\frac{y_\eta}{J}\right)\right] + D_{yy}\left[x_\eta\frac{\partial}{\partial \xi}\left(\frac{x_\eta}{J}\right) - x_\xi\frac{\partial}{\partial \eta}\left(\frac{x_\eta}{J}\right)\right] \\
&\qquad \left. + 2D_{xy}\left[y_\xi\frac{\partial}{\partial \eta}\left(\frac{x_\eta}{J}\right) - y_\eta\frac{\partial}{\partial \xi}\left(\frac{x_\eta}{J}\right)\right]\right]\frac{\partial u}{\partial \xi} \\
&+ \quad \frac{1}{J}\left[x_\xi\left(\frac{\partial D_{yy}}{\partial y} + \frac{\partial D_{xy}}{\partial x}\right) - y_\xi\left(\frac{\partial D_{xx}}{\partial x} + \frac{\partial D_{xy}}{\partial y}\right)\right. \\
&\qquad + D_{xx}\left[y_\xi\frac{\partial}{\partial \eta}\left(\frac{y_\xi}{J}\right) - y_\eta\frac{\partial}{\partial \xi}\left(\frac{y_\xi}{J}\right)\right] + D_{yy}\left[x_\xi\frac{\partial}{\partial \eta}\left(\frac{x_\xi}{J}\right) - x_\eta\frac{\partial}{\partial \xi}\left(\frac{x_\xi}{J}\right)\right] \\
&\qquad \left. + 2D_{xy}\left[y_\eta\frac{\partial}{\partial \xi}\left(\frac{x_\xi}{J}\right) - y_\xi\frac{\partial}{\partial \eta}\left(\frac{x_\xi}{J}\right)\right]\right]\frac{\partial u}{\partial \eta}
\end{aligned}
$$

Here, $D_{xx} = D\cos^2\theta + d\sin^2\theta$, $D_{yy} = D\sin^2\theta + d\cos^2\theta$, and $D_{xx} = (D - d)\cos\theta\sin\theta$. For a fixed mesh, all derivatives of the grid locations and field orientation with respect to ξ, η are constant. To evaluate the above coefficients, the first derivative with respect to ξ, η are centrally approximated, while the second derivatives are approximated via Equation 6.4.

Equation 6.13 has been programmed into the FMS code, and used to generate the plasma-related results of this thesis document. Aside from the alignment procedures, the transformations carried out are standard, and subject to the usual restrictions (e.g. smooth and continuously differentiable).

6.9 Conclusions

To summarize, we require a rotating stencil to handle discontinuous boundary conditions. A rotating stencil is constructed by transforming the diffusion equation into the grid coordinates, and then combining the allowable grid discretizations, or stencils. Consistency with the differential operator places constraints on how these grid stencils are to be combined, but we are afforded one free parameter, which we optimize to suit our purposes. Our optimality condition is operator alignment, when possible. Based on this, we developed a scheme for discretizing misaligned anisotropic diffusion that reproduces expected 1D-like results when the dissipation is aligned with any of the available mesh directions. However, we found inconsistent discretizations when the operator doesn't align, if the physical major-to-minor diffusivity ratio is greater than about 5.8 : 1.

Despite the formal inconsistency for misaligned grids, the scheme performs very well for generally oriented strongly anisotropic dissipation. To verify applicability to our tokamak edge problem, we contrived a pseudo X-point problem and analyzed the resulting profiles, both qualitatively and numerically for good convergence characteristics. The scheme passes all minimum qualifications for providing a useful discretization of the strongly anisotropic dissipation physics of a tokamak edge fluid plasma problem, including allowing misalignment of the numerical grid with respect to the principal axes of diffusion. This property has been verified in particular for the case of computing profiles across a magnetic X-point *without* using multiply connected gridding structures.

The drawbacks to using this scheme are that it is very expensive to compute, and that the fast changes in magnetic field direction cannot be captured in the small number of sample points. In the following chapters on exercising the code, we use a simple version of this algorithm and aligned grid.

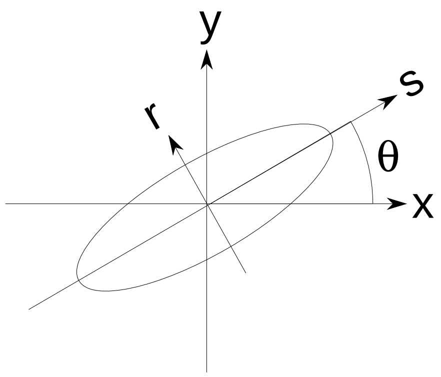

Figure 6.1. The principal axes of diffusion in an elliptic problem are generally rotated from the reference coordinate axes. The physical model is often much simpler when expressed in the coordinates of the principal axes.

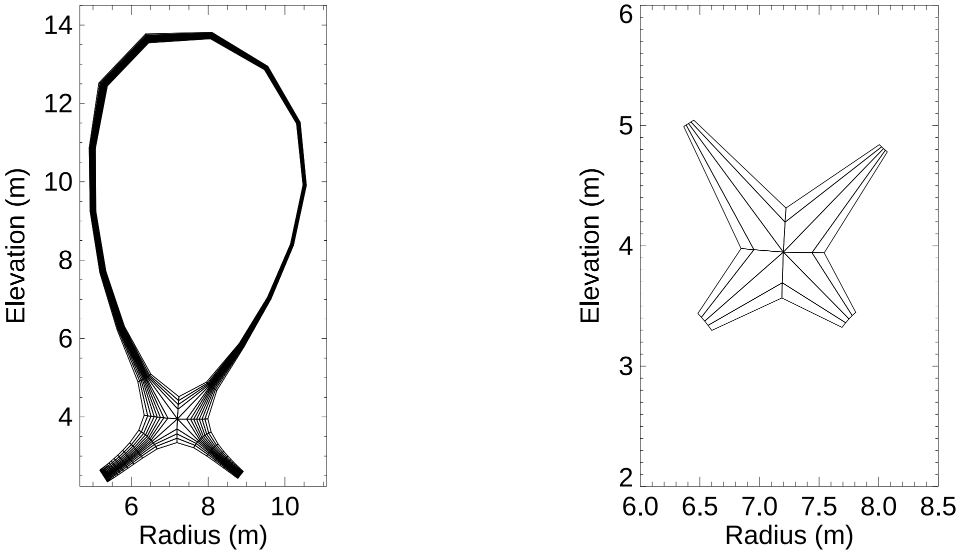

Figure 6.2. Example flux-surface-aligned grid for the single null configuration of the ITER reactor design. The figure on the right is a detail of the grid near the X-point, showing the local non-orthogonality and increased number of neighbor nodes for cells in this region.

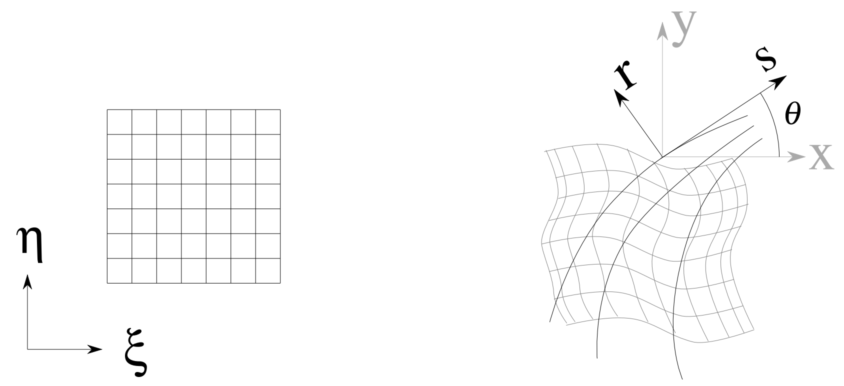

Figure 6.3. Coordinates used for dissipation stencil analysis. The field (specified by "$\hat{s}$") is spatially varying and given with respect to the absolute coordinates (x, y). The computational grid point locations are also given with respect to (x, y), and indexed with (ξ, η).

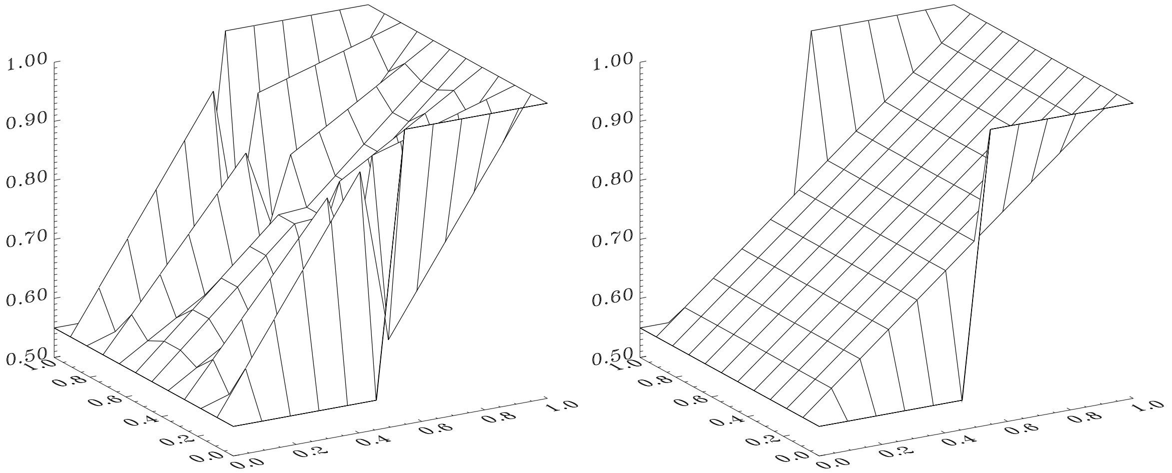

Figure 6.4. Oscillatory dissipation discretization on the left, constructed from bilinear interpolations of the state quantity. The discretization is convergent and consistent with the diffusion operator, but not accurate. The exact solution is reproduced with the simple operator that avoids negative stencil elements.

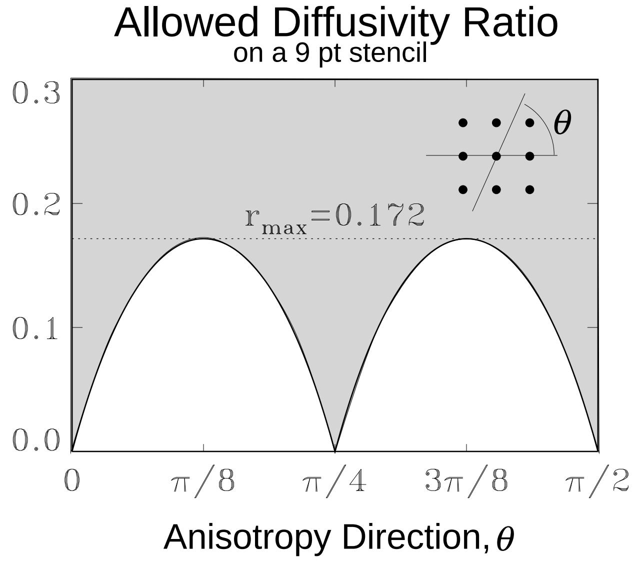

Figure 6.5. Maximum allowable ratio of principal diffusivities based on requiring that the numerical operator on a 9-point stencil remain of *positive-type*. Note that along grid directions, any ratio can be accommodated, and that the most restrictive case occurs when the principal directions cut between the grid points.

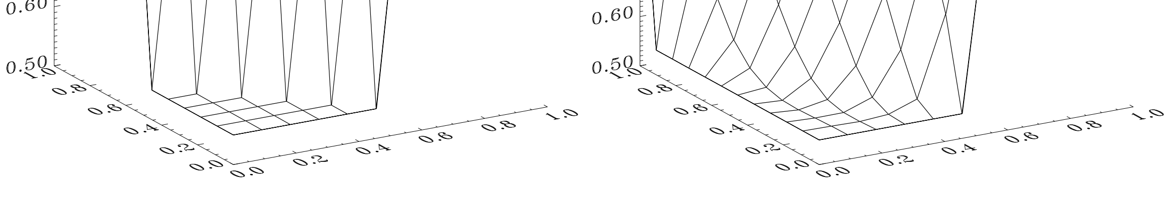

Figure 6.6. An improved operator for anisotropic dissipation at $\theta = -\frac{\pi}{4}$ (on the left), and $-\frac{\pi}{8}$ (on the right). The $\frac{\pi}{4}$ case is nodally exact, $\frac{\pi}{8}$ is the worst case possible, and exhibits the expected oscillations.

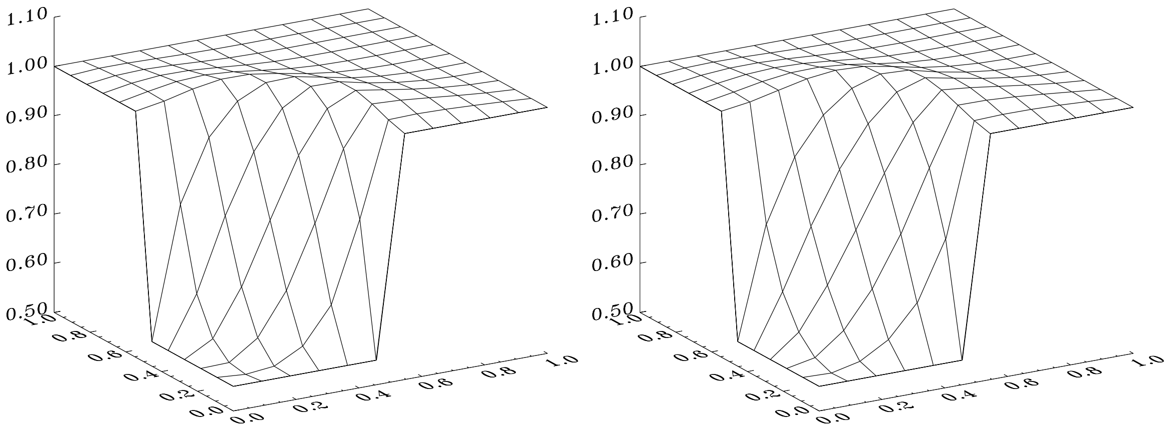

Figure 6.7. Alternative rotating stencil results at $-\frac{\pi}{4}$. The plot on the left results from the angle independent stencil, while the plot on the right uses the operator from Equation 6.4, but with the two cross derivative stencils switched.

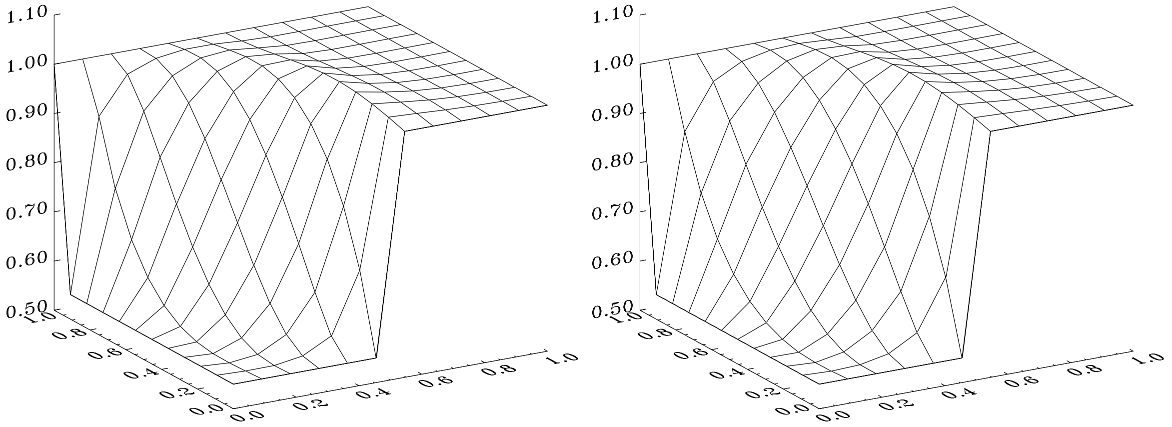

Figure 6.8. Alternative rotating stencil results at $-\frac{\pi}{8}$. The plot on the left results from the angle independent stencil, while the plot on the right uses the operator from Equation 6.4, but with the two cross derivative stencils switched.

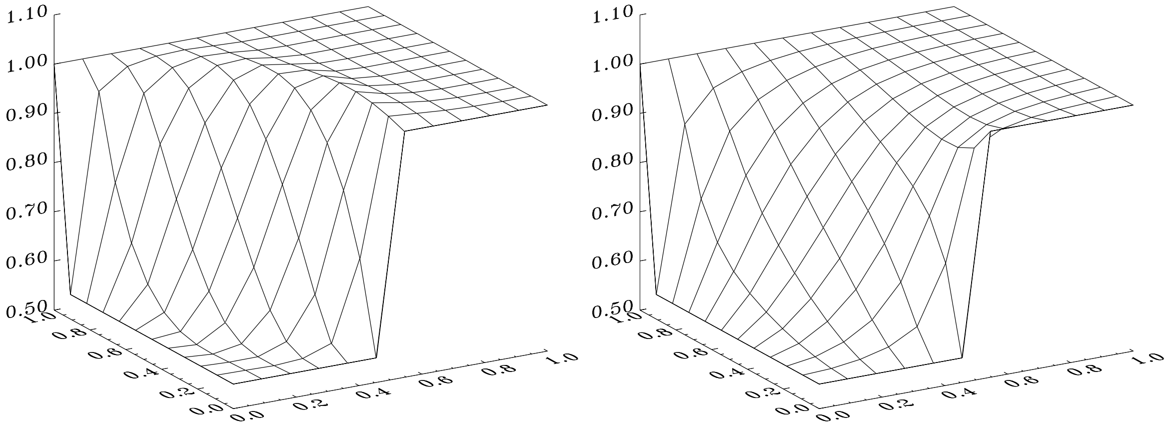

Figure 6.9. Adding artificial anisotropic dissipation at $-\frac{\pi}{8}$ to stabilize the difference operator. The transverse dissipation $d = rD$ (r from Equation 6.5) is required to keep the operator of *positive-type*, but does not diminish with resolution.

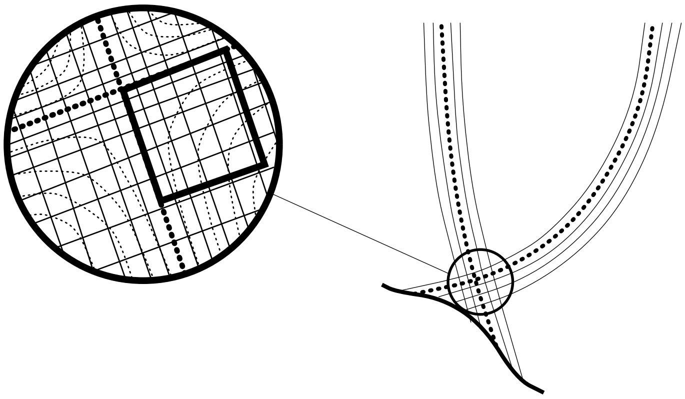

Figure 6.10. Region of X-point test problem, where flux surfaces (dotted lines) are skew to the grid. The test region is the box outlined in the figure inset.

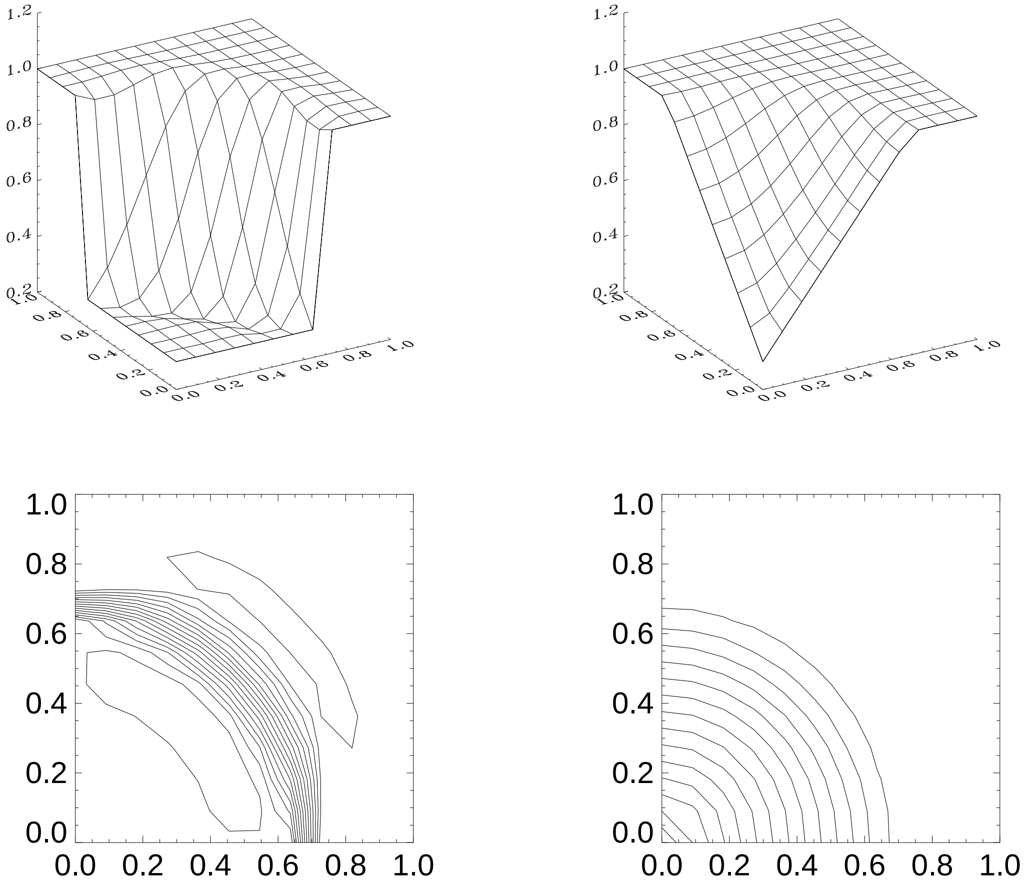

Figure 6.11. Computed solutions to the circular model using the optimized stencil. The left figures show results with the discontinuous boundary conditions that represent steep profile gradients along the separatrix. The right figures show the model of more mild profiles (or, equivalently, more resolution).

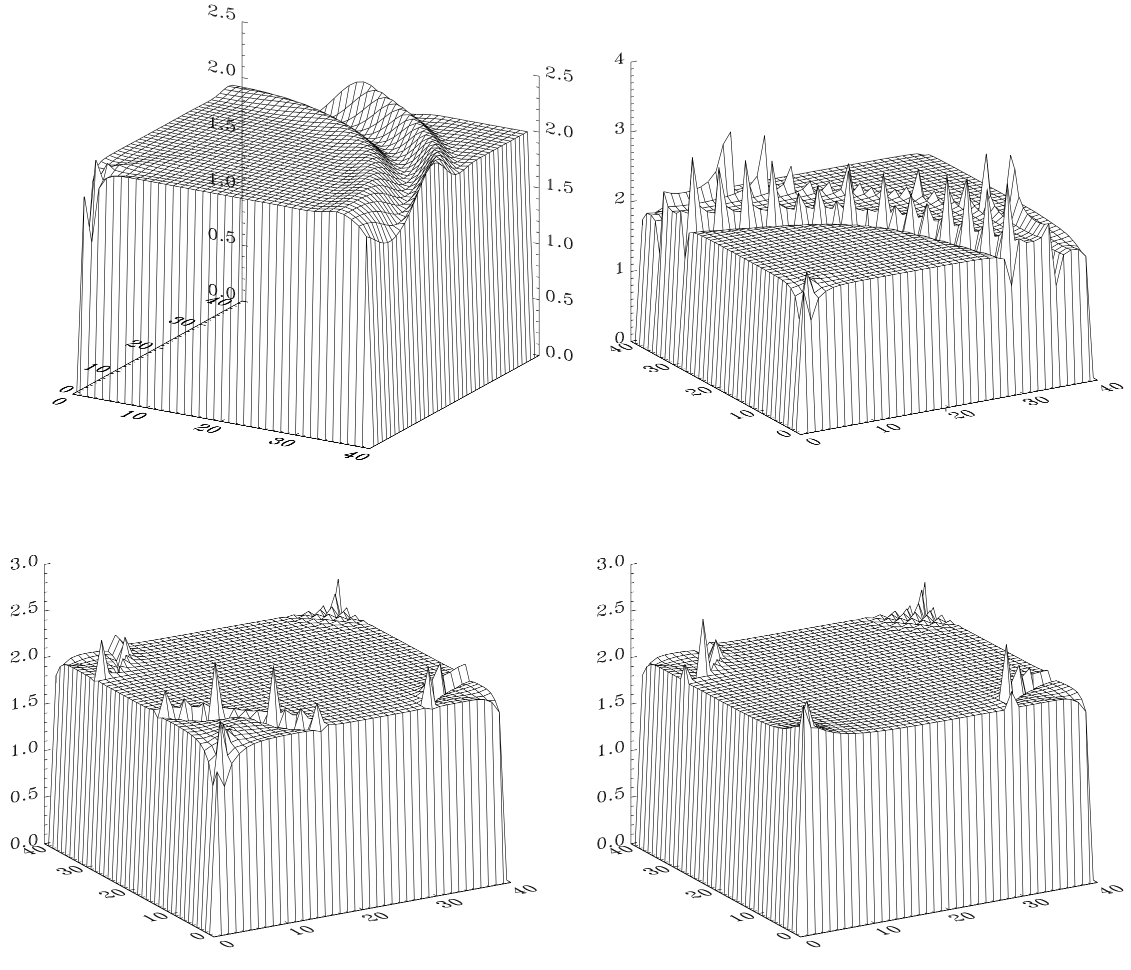

Figure 6.12. Numerical convergence order computed from 161x161 solutions using (clockwise from upper left $d = 0.0, 0.2, 1.0, 0.8$). Since each step resolves equally on Δ_x and Δ_y we expect (and obtain) a second order convergence rate. Note that the spurious spikes in the domain are probably due to finite precision of the graphics package used to compute the rate.

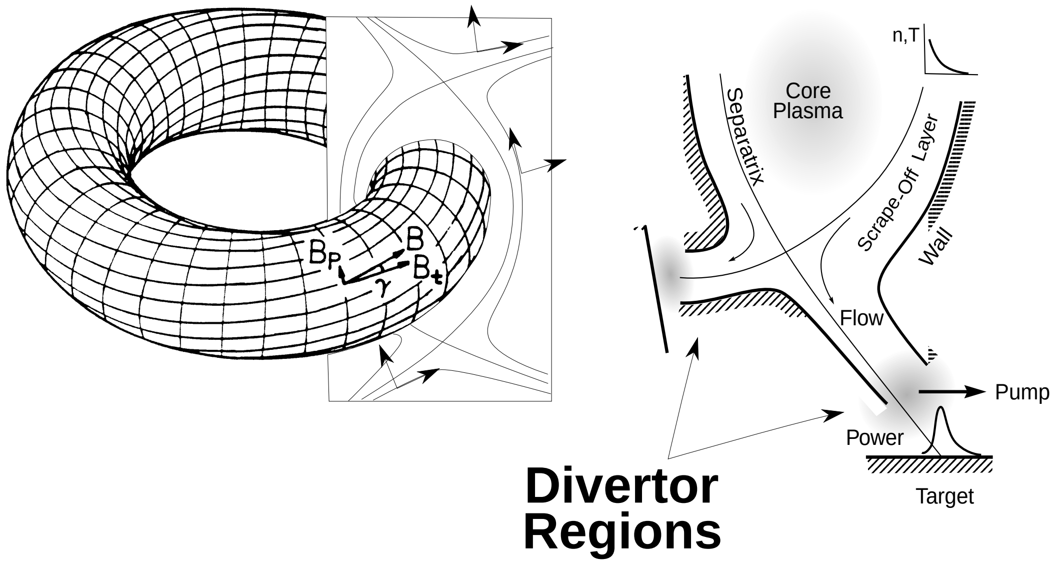

Figure 6.13. Geometry of a tokamak edge plasma divertor.

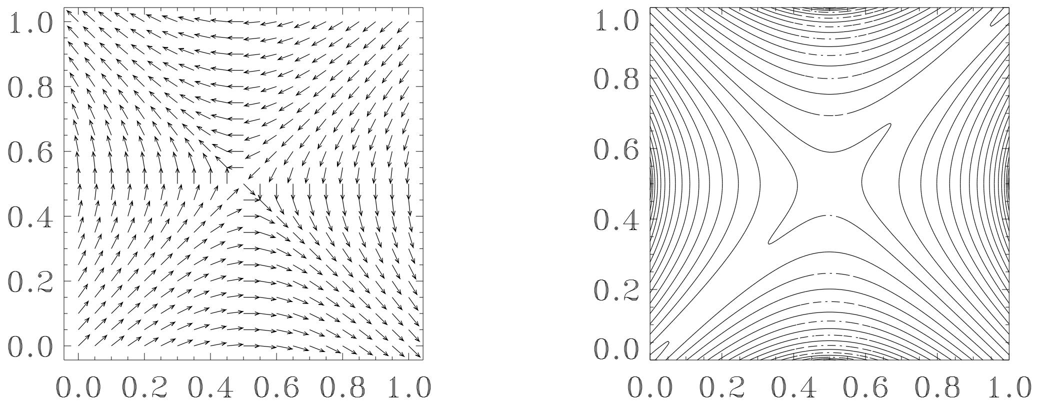

Figure 6.14. Anisotropic diffusion ($r = 1000 : 1$) in X-point geometry. The left plot is input field shape, the right plot is contours of of the solution data.

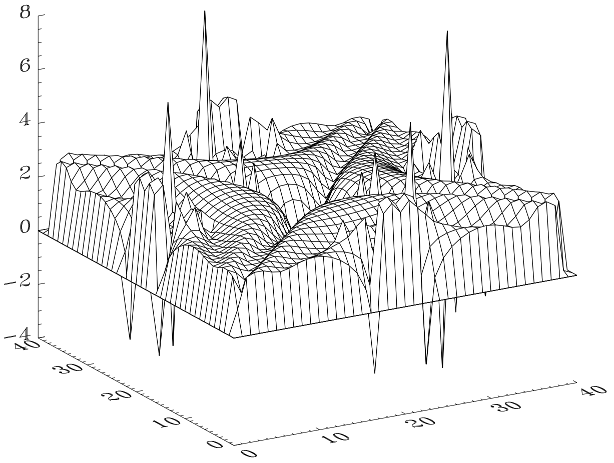

Figure 6.15. Convergence order of the X-point problem. Errors created near the X-point are diffused out along the separatrix, lowering the order of convergence there. The localized spikes in the convergence are still not yet understood.

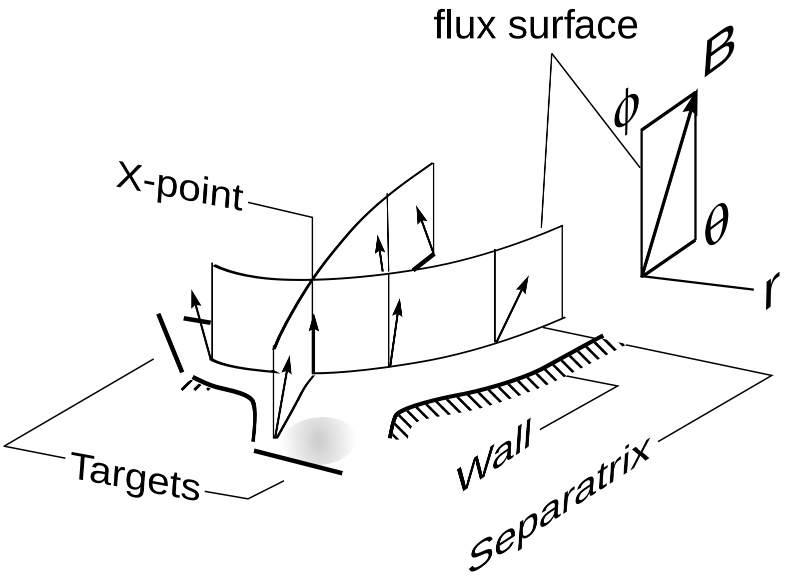

Figure 6.16. Magnetic field pitch angle near a divertor X-point. At the X-point the field is *exactly* perpendicular to the poloidal plane, and the transport may become isotropic diffusion.

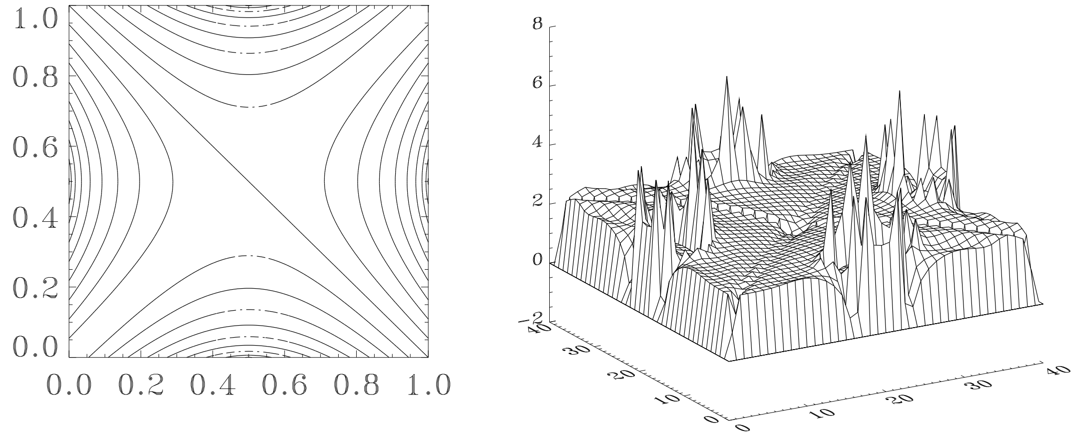

Figure 6.17. Anisotropic diffusion ($r = 1000 : 1$) in X-point geometry with isotropy at the X. The left plot shows the contours of the results, the right plot shows the calculated convergence rate.

Chapter 7

Benchmark Plasma Solutions

We present solutions for standard set of plasma/neutral equations that were originally developed to approximate the ASDEX diverted plasma (cf. References [144, 145, 146, 147, 148]). This special model has been employed to benchmark a number of edge code development projects (e.g. [18, 76, 69, 85, 100]), and is the basis of a recent proposed test problem to benchmark codes for use on the ITER divertor problem[149]. The model has a relatively simple form yet captures the essence of much of the physics relevant to the edge problem, including the generation of a high-recycling region for lowering target temperatures. Since the model problem is specified on a rectangular domain, geometry issues do not arise, and the diffusion operator is constructed to be always aligned to the discrete grid *and* poloidal flux surfaces. It should be pointed out that the boundary conditions and neutral source details are not compatible for producing a so-called "detached" or thermally collapsed divertor plasmas of current interest in for the ITER design. There is considerable disagreement between the ASDEX benchmark model, and the equations we derived in Chapter 4. Our purpose here is to present results of the benchmark as posed to allow direct comparisons to previous results, and to get a feel for the influence of the many "free parameters" in the edge fluid equations.

In particular, parameter variation studies were performed to assess the sensitivity of the model to perpendicular diffusivities of particle, momentum and heat, thermal conductive flux limit values, ionization and charge exchange rate data, wall boundary conditions for the energy equations and the magnetic field projection into the poloidal plane. We find that the computed profiles are sensitive to most of these parameters, and is influenced *significantly* by minor variations in the heat flux limit. Due to large uncertainties in the form and value of the flux limiter, and its strong impact on solution profiles, we discuss the strategies and current status of improved models.

7.1 ASDEX Model Equations

The plasma model equations parallel to the magnetic field are derived from the Braginskii analysis[64], and then projected into the poloidal plane with a constant pitch, $\mathcal{P}$. The *drift* component perpendicular to the magnetic field, but still within the flux surfaces, is ignored by setting the corresponding convective velocity and spatial gradients identically to zero. In the radial direction, perpendicular to the field and flux surfaces, the classical Braginskii fluxes are replaced by *anomalous* expressions—as a result, the radial velocity, v, is no longer an independent variable, and the continuity equation has both parabolic and hyperbolic characteristics. The momentum and energy equations are simplified by ignoring viscous heat generation and cross dissipation terms. There are singly charged ions and neutrals, each with atomic mass, $m^i = 1.5$. The neutral species is everywhere in thermal equilibrium with the ions, consistent with the OGITE[18] approximation.

In the following x denotes length in the poloidal plane along the flux surface, and y denotes the radial length, locally perpendicular to the flux surfaces. The velocity $\vec{u} = u\mathbf{x} + v_\perp\mathbf{r}$, where $\mathbf{x}$ and $\mathbf{r}$ are unit vectors in the poloidal and radial directions, respectively.

Plasma continuity:

$$\frac{\partial n}{\partial t} + \frac{\partial}{\partial x}\left(nu\right) + \frac{\partial}{\partial y}\left(nv_\perp\right) = nn_0\langle\sigma v\rangle^{\mathrm{eii}} \tag{7.1}$$

Neutral continuity:

$$\frac{\partial n_0}{\partial t} - \frac{\partial}{\partial x}\left(D_0\frac{\partial n_0}{\partial x}\right) - \frac{\partial}{\partial y}\left(D_0\frac{\partial n_0}{\partial y}\right) = -nn_0\langle\sigma v\rangle^{\mathrm{eii}} \tag{7.2}$$

Parallel momentum:

$$\begin{aligned}\frac{\partial\left(mnv_\parallel\right)}{\partial t} &+ \frac{\partial}{\partial x}\left(m^i nuv_\parallel + \mathcal{P}p - \mathcal{P}^2\eta_\parallel\frac{\partial v_\parallel}{\partial x}\right) \\ &+ \frac{\partial}{\partial y}\left(m^i nv_\perp v_\parallel - \eta_\perp\frac{\partial v_\parallel}{\partial y}\right) = 0\end{aligned} \tag{7.3}$$

Electron energy:

$$\begin{aligned}\frac{\partial}{\partial t}\left(\frac{3}{2}p^e\right) &+ \frac{\partial}{\partial x}\left(\frac{5}{2}p^e u - \mathcal{P}^2\kappa^e_\parallel\frac{\partial kT^e}{\partial x}\right) \\ &+ \frac{\partial}{\partial y}\left(\frac{5}{2}p^e v_\perp - \kappa^e_\perp\frac{\partial kT^e}{\partial y}\right) \\ &= p^e\nabla\cdot\vec{u} - \kappa_{eq}k\left(T^e - T^i\right) - nn_0\langle\sigma v\rangle^{\mathrm{eii}}\epsilon_e\end{aligned} \tag{7.4}$$

Ion heat equation:

$$\begin{aligned}\frac{\partial}{\partial t}\left(\frac{3}{2}p^i\right) &+ \frac{\partial}{\partial x}\left(\frac{5}{2}p^i u - \mathcal{P}^2\kappa^i_\parallel\frac{\partial kT^i}{\partial x}\right) \\ &+ \frac{\partial}{\partial y}\left(\frac{5}{2}p^i v_\perp - \kappa^i_\perp\frac{\partial kT^i}{\partial y}\right) \\ &= p^i\nabla\cdot\vec{u} + \kappa_{eq}k\left(T_e - T_i\right) + nn_0\langle\sigma v\rangle^{\mathrm{eii}}\epsilon_i\end{aligned} \tag{7.5}$$

Equations of state:

$$p = p^e + p^i \qquad p^e = nkT^e \qquad p^i = nkT^i \tag{7.6}$$

Poloidal and Radial Velocities:

$$u = \mathcal{P} v_{\|} \qquad\qquad v_{\perp} = -\frac{D_{\perp}}{n}\frac{\partial n}{\partial y}$$

Flux limited electron conductivity and Spitzer-Härm conductivities:

$$\kappa_x^e \;=\; \frac{\kappa_{\mathrm{SH}}^e}{\left(1 + \left|\frac{\kappa_{\mathrm{SH}}^e \mathcal{P} \frac{\partial}{\partial x}(kT^e)}{\alpha^e \sqrt{\frac{kT^e}{2m_e}} nkT^e}\right|\right)} \qquad \begin{array}{l} \kappa_{\mathrm{SH}}^e = 3.2nkT^e\tau^e/m^e \\ \kappa_x^i = \kappa_{\mathrm{SH}}^i = 3.9nkT^i\tau^i/m^i \end{array}$$

Braginskii viscosities:

$$\eta_x = \frac{4}{3}\left(\eta_x^i + \eta_x^e\right), \qquad \eta_x^i = 0.96nkT^i\tau^i, \qquad \eta_x^e = 0.73nkT^e\tau^e$$

Neutral diffusion coefficient and energy relaxation:

$$D_{n_0} = \frac{kT^i}{3m^e n\left(\langle\sigma v\rangle^{\mathrm{cx}} + \langle\sigma v\rangle^{\mathrm{cx}}\right)} \qquad\qquad \kappa_{\mathrm{eq}} = 3\frac{m^e n}{m\tau^e}$$

Miscellaneous constants:

$$\alpha^e = 0.12, \qquad k = 1.602 \times 10^{-12}\left(\frac{\mathrm{erg}}{\mathrm{eV}}\right), \qquad m^i = 1.5\, m^p, \qquad Z = 1$$

$$\mathcal{P} = 0.06, \qquad m^p = 1.6726 \times 10^{-24}, \qquad m^e = 9.1094 \times 10^{-28}$$

Anomalous transport coefficients:

$$\kappa_y^i = n\chi^i \qquad \kappa_y^e = n\chi^e \qquad \eta_{\perp} = mn\nu_{\perp},$$

$$D_{\perp} = 2.0 \times 10^4, \quad \chi^i = 0.2 \times 10^4, \quad \chi^e = 4.0 \times 10^4, \quad \nu_{\perp} = 0.2 \times 10^4$$

Rate coefficients

$$\langle \sigma v \rangle^{\mathrm{eii}} = \frac{3 \times 10^{-8} T^{e2}}{300 + T^{e2}} \qquad \langle \sigma v \rangle^{\mathrm{cx}} = 10^{-8} \tag{7.7}$$

Ionization Energy Transfer

$$\epsilon_e = 25\mathrm{eV}, \qquad \epsilon_i = 5\mathrm{eV}$$

The collision times, τ^e and τ^i are given in Appendix A. The model domain is bounded by sections representing the core, upstream, wall, private flux and pump regions (see Figure 7.1). The rectangular domain is 100 cm along the poloidal flux surfaces and 4 cm across. The magnetic field intercepts the poloidal plane at a constant projection angle such that $\frac{B_x}{B} = \mathcal{P} = 0.06$.

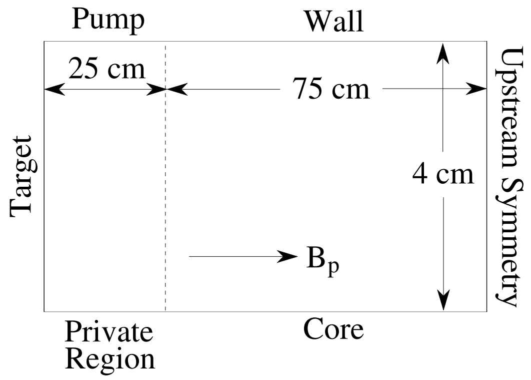

Figure 7.1. Geometry of ASDEX test problem. This topologically square region approximates the scrape-off layer of the high-recycling ASDEX divertor plasma. Similar domains are used throughout this work to simplify the computation setup and graphical output.

Boundary Conditions

The *physical* boundary of a general “edge” or “scrape–off–layer (SOL)” region will consist of walls, target plates and pumps. Additionally, there will be arbitrary boundaries that delimit the edge from the full plasma, such as symmetry planes, core boundary and private flux regions. The "wall" boundaries are generally near material surfaces; the field is often parallel to the walls, and the plasma density and temperatures are low enough that sheath effects are neglected. In the ASDEX model, no net particle flow is allowed to the wall and the electron and and ion temperatures are specified there. On the target plate, we enforce Bohm sheath conditions, as outlined in earlier sections of this document. At pump entrances through the wall near the boundary, only neutrals are allowed to cross the boundary plane. Neutral pumping can be expressed in terms of "albedos"[18, 59]. There is no convective flow of heat or particles through the symmetry plane. The external source of plasma and heat is realized through the core boundary—specified primarily as fixed (Dirichlet) state conditions. The private region (see figure 2.5) is often ignored in edge computations, though it borders on the divertor area along the separatrix. If the private region is neglected, it must appear as an interface boundary condition for a diverted plasma. Generally, particle and heat fluxes are specified at the private area interface, and no convective flow is allowed across the bounding plane.

The following is a summary the default ASDEX model problem boundary conditions. The vector of unknowns in the system is (in order) $\mathbf{s} = (n, n_0, u, T^i, T^e)$; also $\Gamma = nv_\perp$, $\Gamma_0 = -D_0 \nabla_\perp n_0$ and $\perp$ is the outward normal direction at the boundary.

$$\text{Core:}\left[\begin{array}{rcl} n & = & 1.8\times 10^{13} \\ \nabla_\perp n_0 & = & 0 \\ u & = & 0 \\ T^i & = & 80 \\ T^e & = & 80 \end{array}\right] \quad \text{Private:}\left[\begin{array}{rcl} \Gamma & = & 0 \\ \Gamma_0 & = & 0 \\ u & = & 0 \\ \nabla_\perp T^i & = & 0 \\ \nabla_\perp T^e & = & 0 \end{array}\right]$$

$$\text{Symmetry:}\left[\begin{array}{rcl} \Gamma & = & 0 \\ \Gamma_0 & = & 0 \\ u & = & 0 \\ \nabla_\perp T^i & = & 0 \\ \nabla_\perp T^e & = & 0 \end{array}\right] \quad \text{Target:}\left[\begin{array}{rcl} \text{outflow:} & & \text{no BC} \\ \Gamma_0 & = & -\Gamma \\ u & = & \mathcal{P} m^i C_s \\ \nabla_\perp T^i & = & 0 \\ -\mathcal{P}^2 \kappa^e_{\text{SH}} \nabla_\perp \left(kT^e\right) & = & \frac{3}{2}\mathcal{P} C_s n k T^e \end{array}\right]$$

$$\text{Wall:}\left[\begin{array}{rcl} \Gamma & = & 0 \\ \Gamma_0 & = & 0 \\ \nabla_\perp u & = & 0 \\ T^i & = & 2 \\ T^e & = & 2 \end{array}\right] \quad \text{Pump:}\left[\begin{array}{rcl} \Gamma & = & 0 \\ \Gamma_0 & = & 0.5\frac{(1-\gamma)}{(1+\gamma)} n_0 \sqrt{\frac{kT^i}{m^i}} \\ \nabla_\perp u & = & 0 \\ \nabla_\perp T^i & = & 0 \\ \nabla_\perp T^e & = & 0 \end{array}\right]$$

$$C_s = \sqrt{\frac{k\left(T^i + T^e\right)}{m^i}} \qquad M = -1 \qquad \gamma = 0.95(\text{albedo})$$

7.2 Solution Procedure

A well-known characteristic of the Newton method used in FMS2D is that it requires a very good initial guess to the solution profiles before the iteration procedure converges quadratically. It is not clear how to generate such data, especially when working on an unfamiliar system or in complex geometries.

In this work, we relied on the robust relaxation techniques described elsewhere in this document. We generated simple analytic initial data, and time-marched the conservation equations to steady state with the fully-implicit Euler time discretization. At each time step, the resulting nonlinear system was solved iteratively using the full Newton machinery. We damp undesirable large Newton updates, and adaptively choose the time increment commensurate with the difficulty of the Newton solve. In this way, we can start the solution process with $dt = 10^{-9}s$, and end up with $dt \sim 10^4 s$ or so. When the solution ceases to change appreciably from time step to time step (on the order of $t \geq 10^4 s$), we stop the code, declaring that a steady solution has been found.

When the Newton iterates require damping, convergence of the scheme is difficult to characterize. Often, a given time step-size will be too large for the scheme to be convergent at all—once that situation is detected, the current is step abandoned, and restarted at a smaller time step (the equations reduce to the identity function for $dt \to 0$). During successful steps, we do in fact observe the quadratic convergence characteristic of the Newton scheme. This model case is fairly well behaved, and the solver proceeds quite well—the time increment is adjusted from initial to final values in about 30 time steps, each taking 4-5 Newton iterations[1]

Time relaxation is not always required. For the parameter variation and grid refinement studies, a steady solution was interpolated (using isoparametric bilinear interpolation shape functions) onto the new grid and used as initial conditions. The *direct* Newton method was used to find the new solution (in practice, we had merely to set $dt_{initial} = t_{final} = 10^4$). A new equilibrium was generally found after 3-9 Newton iterations. As the solution moves into spatial convergence, the sequencing steps are solved easier and easier by the Newton algorithm.

---

[1] The algorithm for increasing/decreasing the time increment is arbitrary and based on practical experience. The chosen scheme has worked well on a variety of problems, but is certainly not optimal.

Most of the calculations presented in the following sections were performed on a DEC Alpha or DECstation 5000/200 ULTRIX machine. On 16x32 grids, initial solves (from scratch) generally took on the order of a couple hours to find steady profiles. Direct solves from good initial data usually required a few minutes. Resolved calculations on grids of 128x256 usually took about an hour per Jacobian inversion, far too slow for routine production—we made no attempt to optimize the code for performance. The initial, model code verification runs were computed on grids up to 128x256; the parameter variation studies were all performed on 32x64 grids.

7.3 Default Model Results

Steady solutions to the discretized model equations were found on a coarse grid having 8 grid cells across the radial length of 4 cm, and 16 cells along the poloidal length of 100 cm. The grid was exponentially stretched to concentrate cells near the target and core boundaries. A new grid, 16x32, was generated by equally subdividing the cells of the 8x16 grid, so that one quarter of the new cells centers coincide with the cell centers of the coarser grid. The 8x16 result was then linearly interpolated to the 16x32 grid, and inserted into FMS2D as the initial conditions. The solution to the 16x32 problem was similarly interpolated to a 32x64 case, and its solution interpolated to a 64x128 grid. The figures display the final solution, on the grid of 64x128 cells (64 cells in the radial direction). It is apparent that the resulting cells are not particularly well distributed, but the solution is nearly resolved (except for near the wall/pump interface). There are 5 unknowns per node, and boundary nodes exist all around the domain giving

$5 \cdot 66 \cdot 130 = 42,900$ unknowns for a grid of 64x128 cells. The Jacobian matrix, required for computing with the Newton algorithm, has with $386,100$ nonzero entries in this case, representing about the maximum problem size one would attempt without optimizing the code. The results are shown in Figure 7.2. These solutions are to be compared quantatively with those presented in Reference [76], using the contour plots shown in Figures 7.3 and 7.4. Our contour plots display the same data levels as those in the reference. In all parts of the domain, our computed result is within 5 percent of the published results for this problem, and the solution profile features are identical.

There are several interesting features of these solutions. First, the close agreement of our solutions to the published data is rather surprising, given that we used a completely different computational grid *and* neutral transport model. This would seem to indicate that in this relatively mild plasma scenario, the detailed treatment of neutral transport is not particularly important. In fact however, we shall find in subsequent sections that this agreement is largely coincidental because the various model parameters do have significant effect on the solution, yet were set rather arbitrarily.

It should be pointed out that this test model does contain severe faults, in terms of providing a good benchmark for edge plasma codes. The boundary condition along the wall ($y = 4$) for the electron and ion energy equations is clearly inappropriate. As we resolve the grid, an internal discontinuity persists at the pump/wall boundary ($x = 75, y = 4$). The discontinuity in temperature leads to sharp changes in pressure (see Equation 7.6), which is balanced by a corresponding sharp change in poloidal velocity due to Equation 7.4. This model was actually

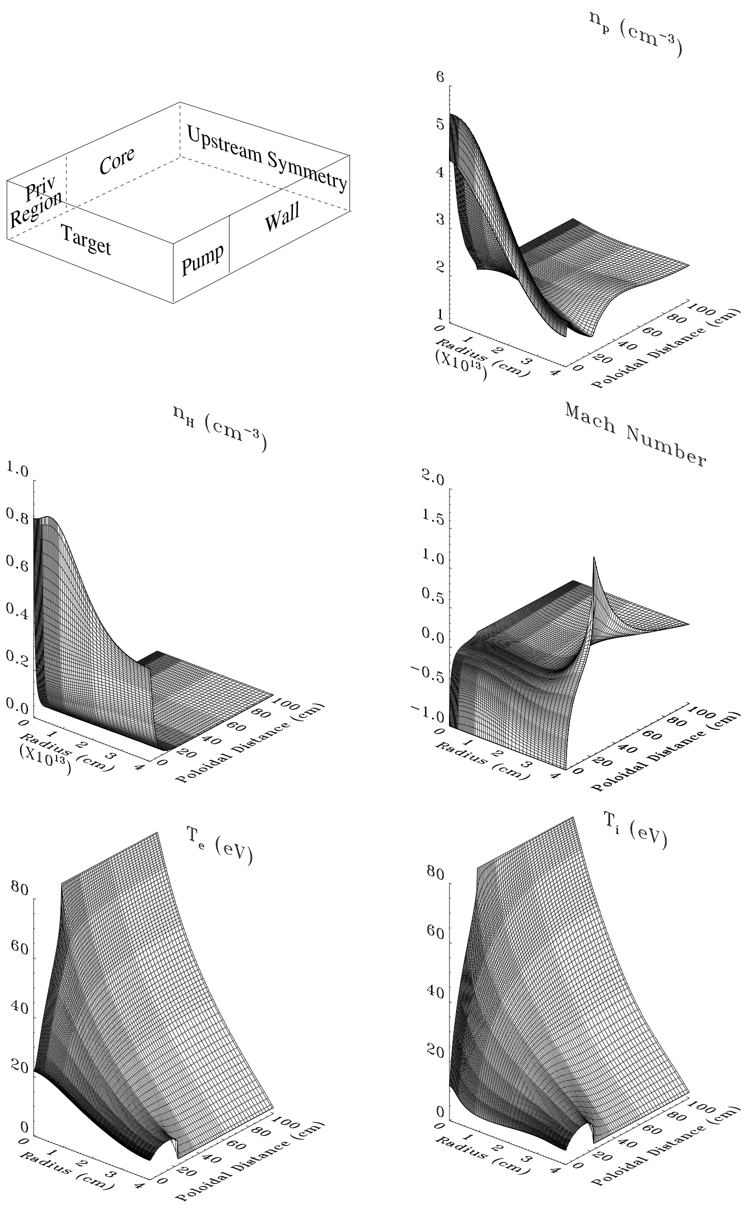

Figure 7.2. Solutions for the default ASDEX case. The plasma and neutral densities (n and n_0, resp.) are plotted in units of $10^{13} cm^{-3}$, parallel mach number is computed locally, and the electron and ion temperatures (T_e and T_i, resp.) are plotted in electron volts. Notice that the temperatures fall rapidly away from the core. Temperature discontinuities along the wall boundary at the pump lead to large pressure gradients, and subsequent velocity spikes, as evident in the figure. This spike grows with mesh refinement.

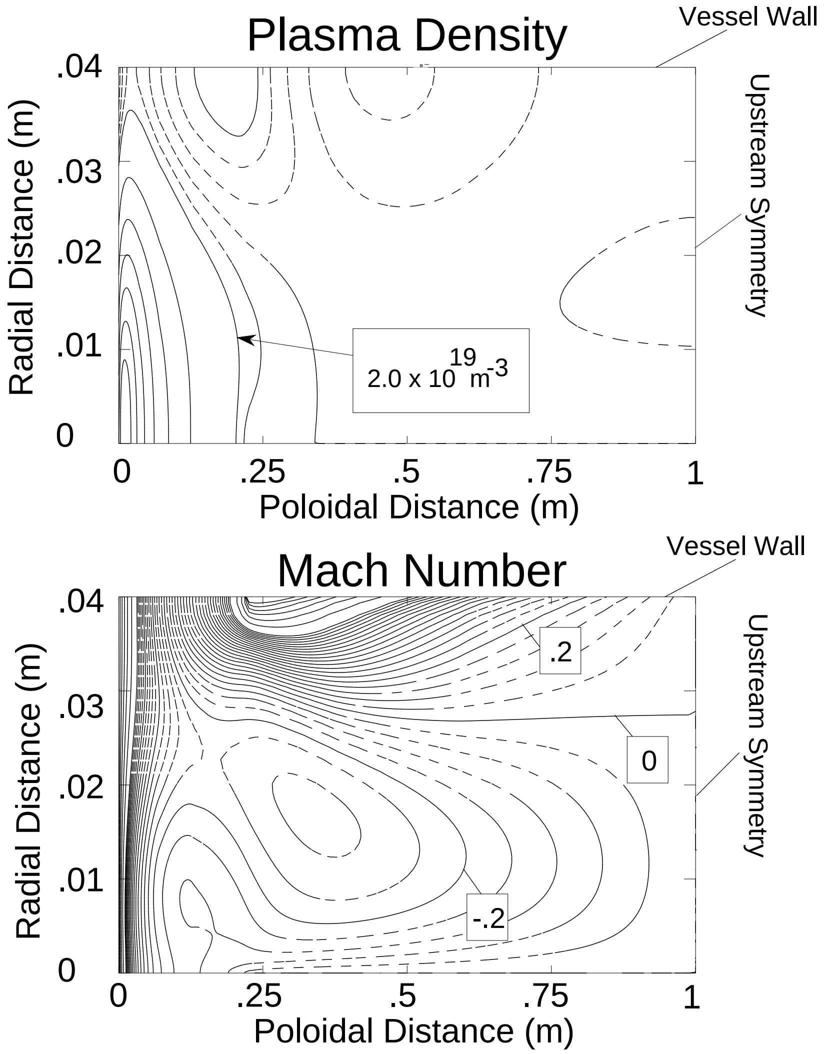

Figure 7.3. Contours of n and Mach number for default ASDEX model, computed on 64x128 grids. In the top plot, the increment between dashed contours is $10^{12} cm^{-3}$, and between solid contours is $5 \cdot 10^{12} cm^{-3}$. In the lower plot, the increment between dashed contours is 0.02, and between solid contours is 0.10 (no units). The bottom of the plots bound the core and private interfaces, the right side represents upstream. The top bounds the wall and pump while the target is to the right. Note the prominent plasma density buildup just in front of the target, and the corresponding zone of flow reversal driven by the local pressure gradient.

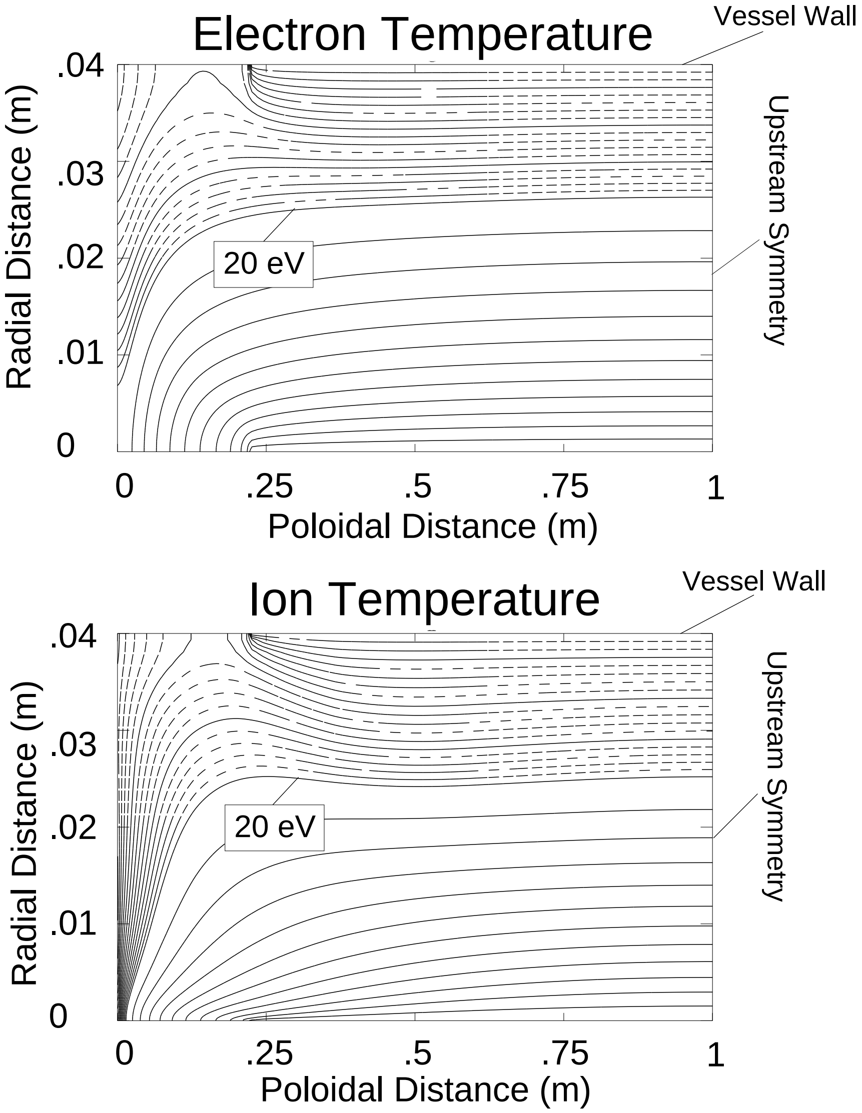

Figure 7.4. Contours of T_e and T_i for default ASDEX model, computed on 64x128 grids. The increment between dashed contours is 1 eV, and between solid contours is 5 eV. Plot orientation is identical to that of the previous page. Note that the two profiles differ; the strong electron parallel conductivity leads to stronger alignment of the temperature contours with the ambient magnetic field (which runs horizontal in these figures).

computed on a grid of 128x256 cells with the similar, but exacerbated solution features. Apparently, resolution of the outer boundary condition requires a great deal more cells than we are presently capable of applying—there may not even be a computable solution for this case.

7.4 Parameter Variations

The ASDEX benchmark model (hereafter, the *test model*) was the basis for a series of parameter variation studies. These studies provide a way to assess the sensitivity of the basic edge plasma model to the uncertain parameters used in its specification. For each parameter, we will identify the aspect of the solution most directly affected, and discuss the associated engineering or physics related issues.

7.4.1 Wall Boundary Conditions

First, we consider a simple method to remove the aforementioned difficulty in the model problem. We follow an earlier suggestion by Vold[18] and Vesey[100] and modify the wall boundary conditions to be of Neumann form, setting the outward normal gradient of electron and ion temperatures to zero. Since the radial convection and conduction are effectively shut off at the wall, all the heat put into the system through the core boundary would now be removed along the target.

The corresponding solution on a 32x64 grid is shown in Figure 7.5, where the desired features are apparently achieved. The kinks in the electron and ion temperature profiles at the pump/wall interface have been removed from the solution, and as a result, the the velocity and plasma density profiles are considerably smoother. Since this change has affected all the solution profiles, it is interesting to look at the total energy flux to the target plate.

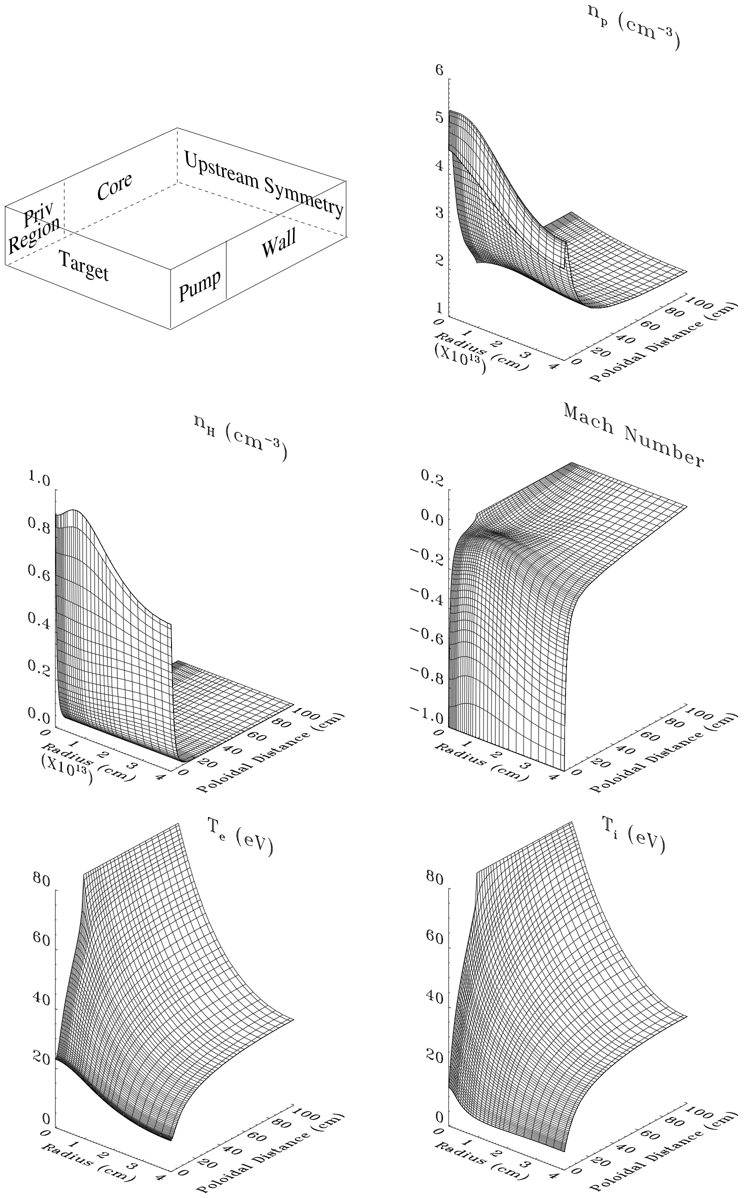

Figure 7.5. Solutions for the Neumann ASDEX case. Notice that the kink in the temperature profiles at the wall/pump interface has been removed, leading to smoother velocity and density solutions. The plasma density peak just in front of the target has also increased slightly due to stronger flow reversal apparent in the velocity profile. This case is generally more suitable to benchmarking studies.

The total heat deposited on the target plate is computed from

$$\begin{aligned} q_{\text{target}} &= \left(\delta_e n C_s k T_e + \delta_i n C_s k T_i + \frac{1}{2} M n C_s^3 + 13.6 k n C_s\right) B_p/B \\ &= (T_e(\delta_e + \gamma_e) + T_i(\delta_i + \gamma_i) + 13.6) 10^{-9} n k C_s \frac{B_p}{B} \left(MW/m^2\right) \end{aligned} \tag{7.8}$$

which represents the sum of electron and ion heat, and ion kinetic energy transmitted through the sheath plus the surface recombination energy. The formula ignores any volumetric radiation that impinges on the plate. For this model, $\delta_e = 4, \delta_i = 2.5, \epsilon_e = 25$ eV, $\gamma_e = \gamma_i = 1$ and $\mathcal{P} = .06$.

A series of runs were performed to vary the boundary conditions smoothly between each of these two regimes. Here, since the two cases are of different type (Neumann vs. Dirichlet), we needed to convert one of them to the other type; we chose to work with the boundary radial gradients. The default model resulted in boundary gradients in both T_e and T_i of approximately 15 eV/cm. Three intermediate cases were run with $\nabla_\perp T_e = \nabla_\perp T_i = a, a \in (4, 8, 12)$. The total heat flux is plotted in Figure 7.6 for each of the five cases as a function of position along the target plate ($x = 0$). The target plasma density, and electron and ion temperatures are likewise plotted in Figures 7.7, 7.8 and 7.9 respectively.

As expected, the total heat flux increased with decreasing radial heat conduction to the wall boundary. The Neumann condition has resulted in considerably higher wall temperatures, now on the order of 25 eV. Interestingly, both the temperature profiles at the target plate change very little in spite of the 22 eV difference in wall temperature.

It is interesting to note how the plasma has adjusted to exhaust the increased heat load, as a number of phenomena are simultaneously incorporated. The positive velocity spike due to the Dirichlet wall boundary conditions on the tempera-

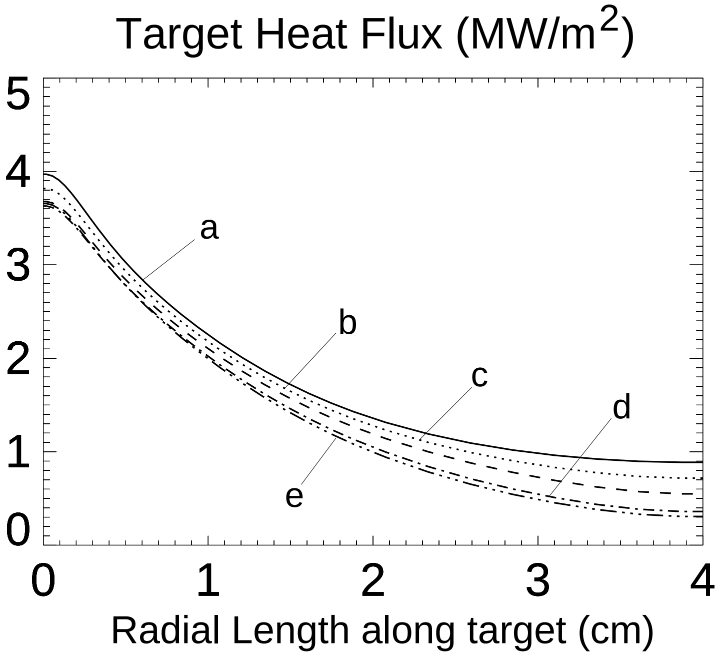

Figure 7.6. Target heat flux profiles vs. $(\nabla_\perp T)_{wall}$ at the first wall. The cases a-e represent $(\nabla_\perp T)_{wall} = (0, 4, 8, 12)$ and $T_{wall} = 2$, respectively. The heat flux on the plate increases with decreasing gradient at the side wall, but not a significant amount.

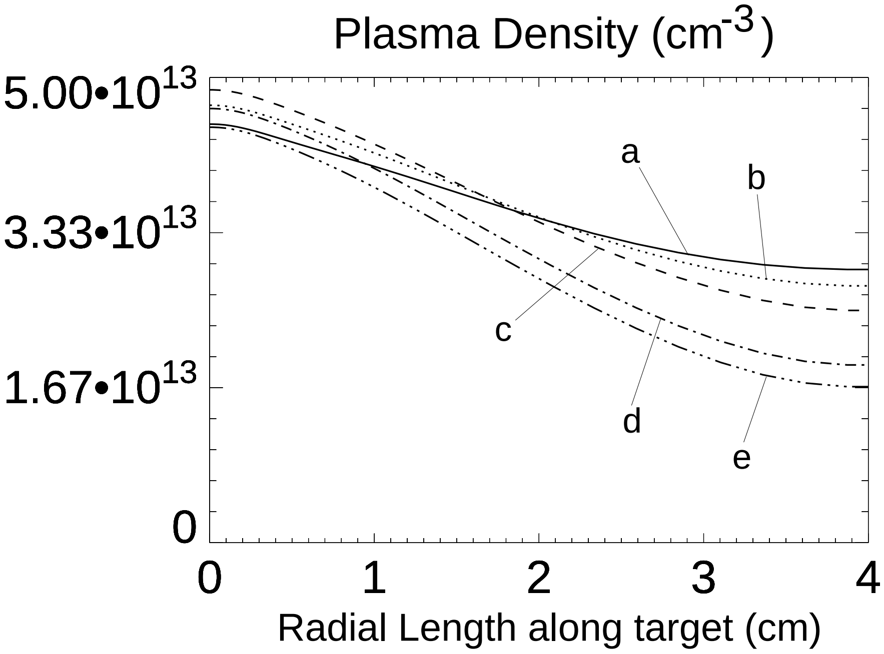

Figure 7.7. Target plasma density profiles vs. $(\nabla_{\perp}T)_{wall}$ at the first wall. The cases a-e represent $(\nabla_{\perp}T)_{wall} = (0, 4, 8, 12)$ and $T_{wall} = 2$, respectively. When less heat is allowed to leave through the side walls of the system, the plasma density increases to improve power outflow at the target.

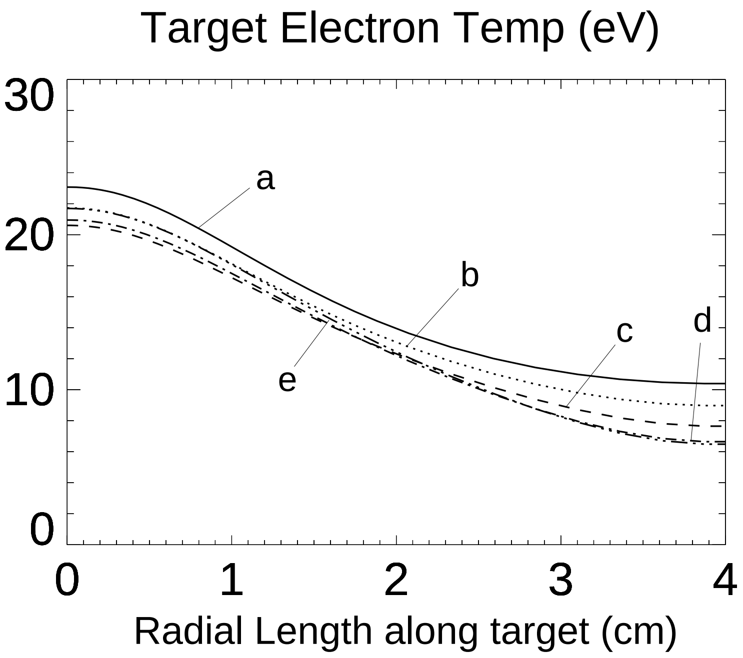

Figure 7.8. Target electron temperature profiles vs. $(\nabla_\perp T)_{wall}$ at the first wall. The cases a-e represent $(\nabla_\perp T)_{wall} = (0, 4, 8, 12)$ and $T_{wall} = 2$, respectively. With less heat loss from the the side wall, the target electron temperature increases to exhaust the excess heat through the plasma sheath.

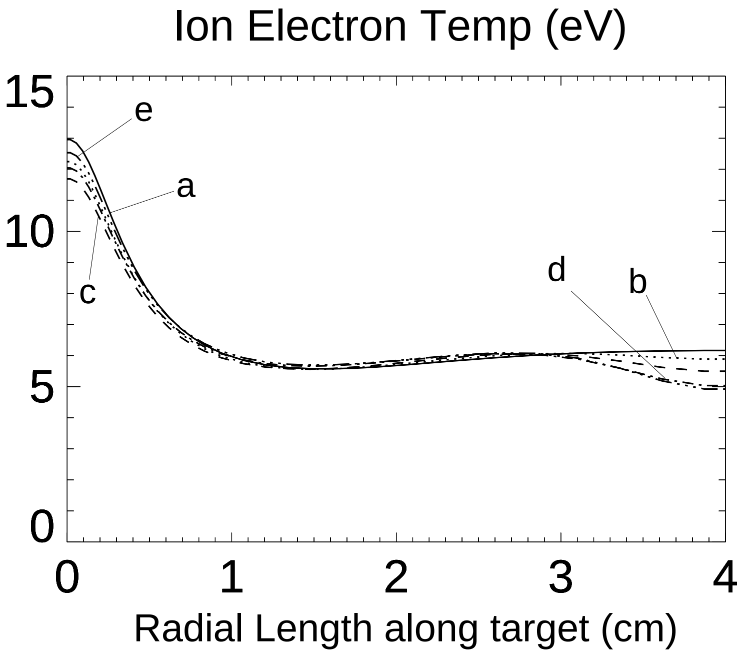

Figure 7.9. Target ion temperature profiles vs. $(\nabla_\perp T)_{wall}$ at the first wall. The cases a-e represent $(\nabla_\perp T)_{wall} = (0, 4, 8, 12)$ and $T_{wall} = 2$, respectively. Since very little heat is removed through the plasma sheath by the ions, small increases in the target heat load have relatively little effects on the ion temperature profile. Alternatively, since the radial transport is more comparable to the parallel flows in the ion channel, radial perturbations extend less into the plasma than they do for the electrons.

ture equations leads to a local depletion of plasma density (apparent in Figure 7.2). The Neumann condition removes this depletion zone, which ultimately increases the target plasma density values near the outboard side of the target. The density increase accounts for only a portion of the additional required heat exhaust, and so is accompanied by a corresponding increase in electron temperature. However, the internal electron energy transport split (parallel vs. radial) was not affected by the change in boundary conditions; the electrons maintain a constant profile shape leading to larger T_e values on the inboard side of the target. In the next section, we discuss why the latter increase ultimately leads to a *decrease* in plasma density on the inboard side of the target.

7.4.2 Anomalous Particle Diffusivity

In order to assess the influence of the particle diffusivity, four variations were created from the default case, setting $D_\perp$ to $(1,4,8,10)\cdot 10^4 m^2/s$. These values represent some of the extreme values reported in the literature, and we'd like to investigate the influence this range has on the solution of our model system.

The resulting plasma density, electron and ion temperatures along the plate are plotted in Figures 7.10, 7.11 and 7.12, respectively. The peak of plasma density occurs in the higher temperature regions where there is considerable neutral densities (just in front of the divertor target on the inboard hot side of the plate). Particles diffuse away from this region to balance the source. Figure 7.10 clearly shows how the radial gradient of plasma density increases with decreasing $D_\perp$ to expel plasma from the particle source region. However, locally we see that the temperature profiles drop where the plasma density increases.

We can use these results to verify a rather simple scaling. If we ignore the

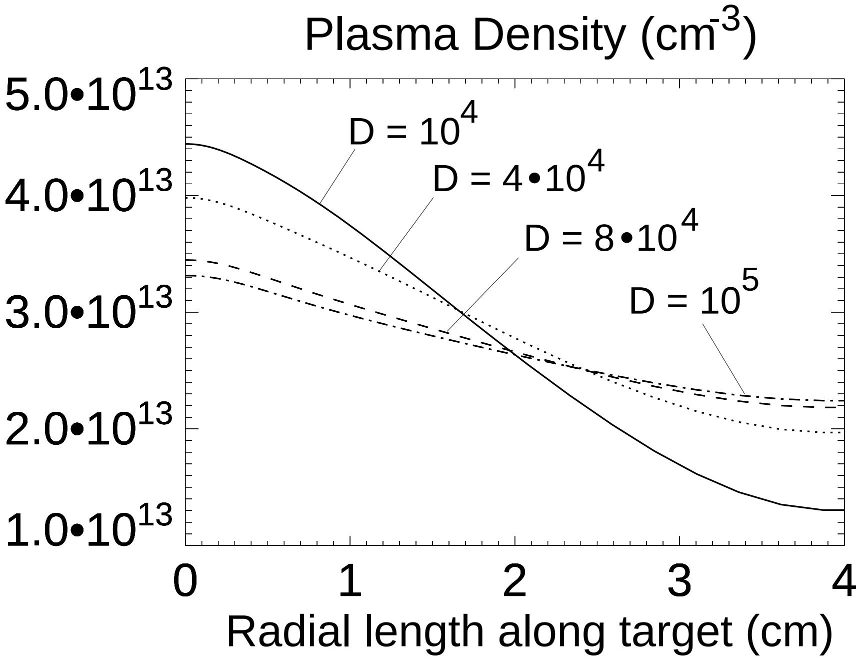

Figure 7.10. Target plasma density profiles vs. D_n. Increasing the particle diffusivity flattens the plasma density profile along the target plate as expected.

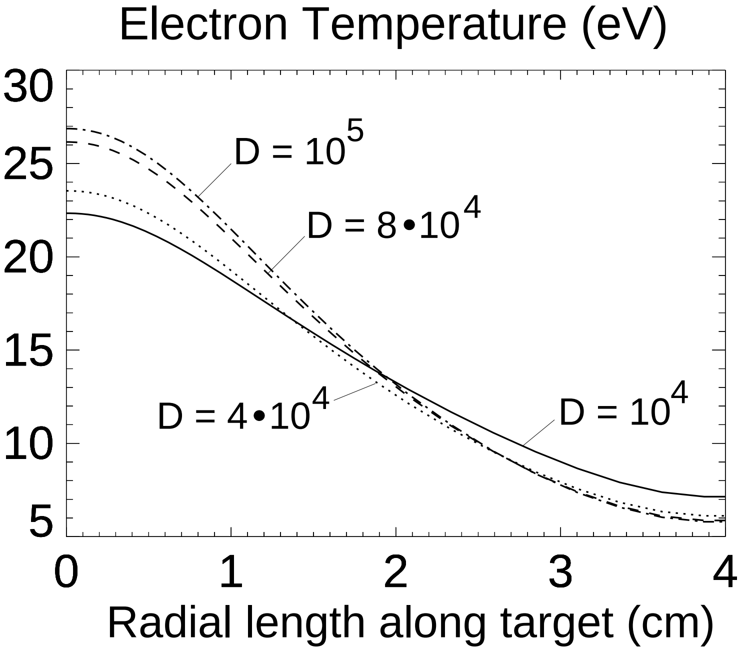

Figure 7.11. Target electron temperature profiles vs. D_n. The electron temperature along the plate adjusts with the changing plasma density to preserve approximate pressure balance along the field lines.

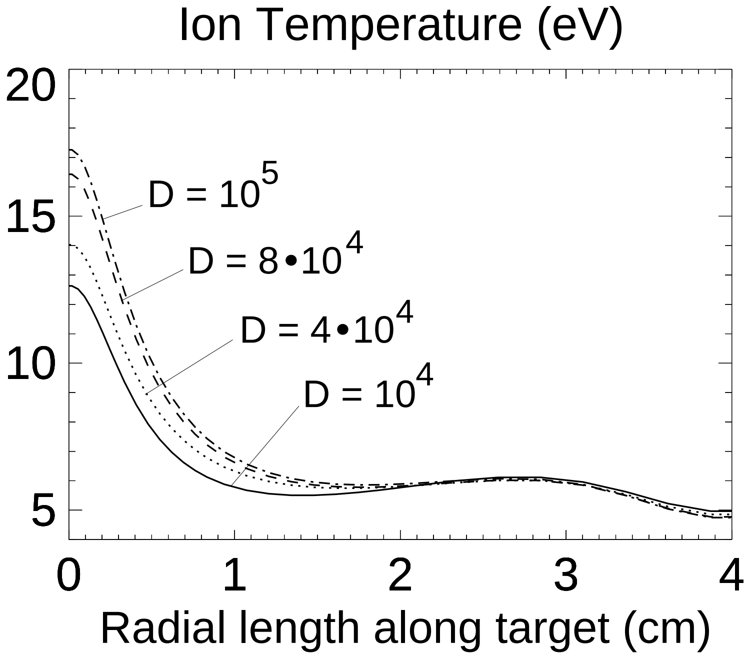

Figure 7.12. Target ion temperature profiles vs. D_n. The ion temperature profile remains mostly unaffected by the significant density modifications. The primary effect with increasing D_n is to increase the radial convective heat flux, thus spreading the profile slightly

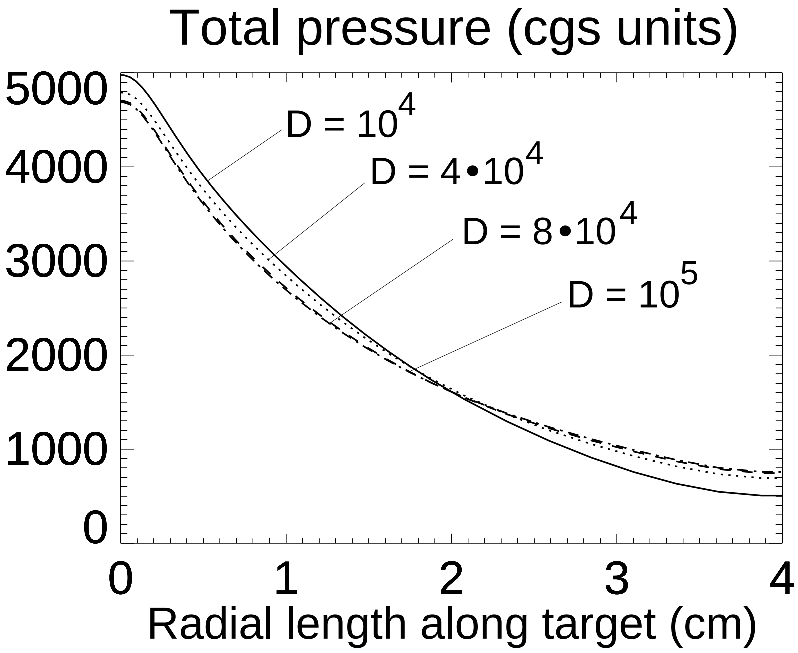

Figure 7.13. Target pressure profiles vs. D_n. The pressure profiles remain essentially unchanged under this profile modification, verifying the expression derived in the text.

viscosity terms and assume constant radial transport through a flux tube, the momentum conservation equation, Equation 7.4, can be integrated over a field line from the stagnation point to the target. The divergence theorem leads to the following

$$\begin{aligned} \left(mnu_{\|}^2 + P\right)_{inlet} &= \left(mnu_{\|}^2 + P\right)_{target} \\ P_{inlet} &\approx mnC_s^2 + P_{target} = 2P_{target} \end{aligned} \tag{7.9}$$

where P is the total pressure (ion plus electron) and mach number = -1 at the target, zero at the inlet. Thus, if viscous dissipation is ignorable, the total pressure should be roughly constant regardless of $D_\perp$.

In Figure 7.13, the target pressure is plotted in erg/cm^3. Aside from small deviations, attributable to parallel viscosity and numerical error (with our nonconservative difference), it is clear that the pressure is indeed unchanged by the particle diffusivity.

This feature provides a good rule of thumb for understanding divertor plasma. When some process in the edge plasma leads to increases plasma density, the ionization source rate increases. Since each ionization results in a loss from the electron energy channel, the electron temperature decreases accordingly. Roughly speaking, the ion temperature will then adjust to ensure pressure balance (also, to account for the volume source term in Equation 7.6). It was noted earlier that the overall heat load to the target plate is rather unaffected by such temperature/density trade-off. Power flux reduction must employ methods to disperse the plasma momentum ahead of the target—precisely the idea behind the gas-target and radiative divertor concepts proposed for the ITER divertor design.

7.4.3 Anomalous Electron Heat Diffusivity

We repeated the foregoing analysis, this time manipulating the electron radial heat diffusivity, $\chi^e \in (1, 2, 4, 8, 10) \cdot 10^4 m^2/s$ (and again using the default settings for all the other model parameters). The resulting plasma density, electron temperature and ion temperature profiles along the target plate are shown in Figures 7.14, 7.15 and 7.16, respectively.

In this case, we see the expected broadening of the electron temperature profile with increasing χ^e, and also the fall in plasma density that accompanies the rise in target temperatures. However, notice also that the magnitude of the electron temperature has increased significantly along the entire plate. The fixed wall boundary condition places an overall constraint on the allowable temperature gradients at the core. Increasing the radial conductivity effectively increases the heat flux into

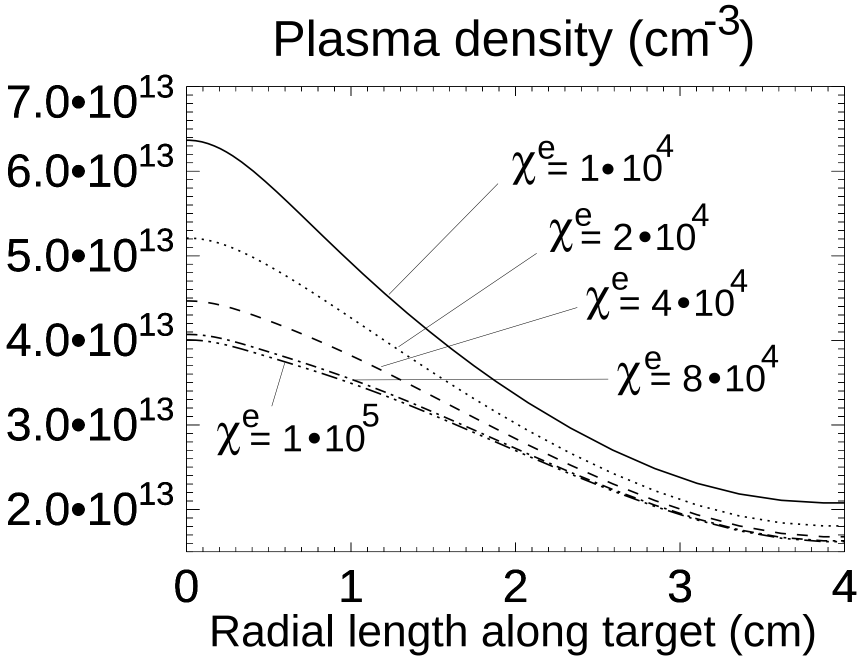

Figure 7.14. Plasma density vs. χ^e along the target plate. Despite that more particles are created through the source term due to elevated temperatures with increasing χ^e, the plasma density decreases to preserve pressure balance.

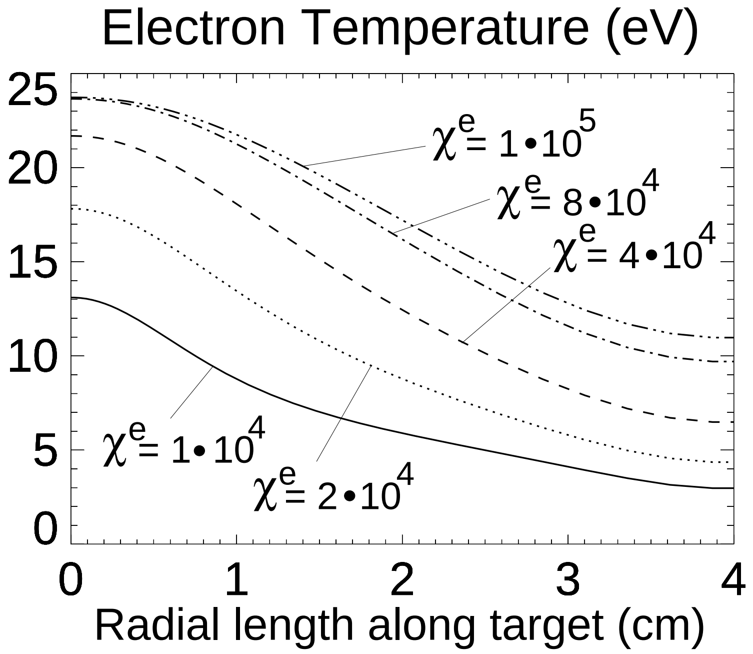

Figure 7.15. Electron temperature vs. χ^e along the target plate. Since the increased diffusivity leads to an increase in total energy input to the SOL (see the text), the result is an increase in overall temperature.

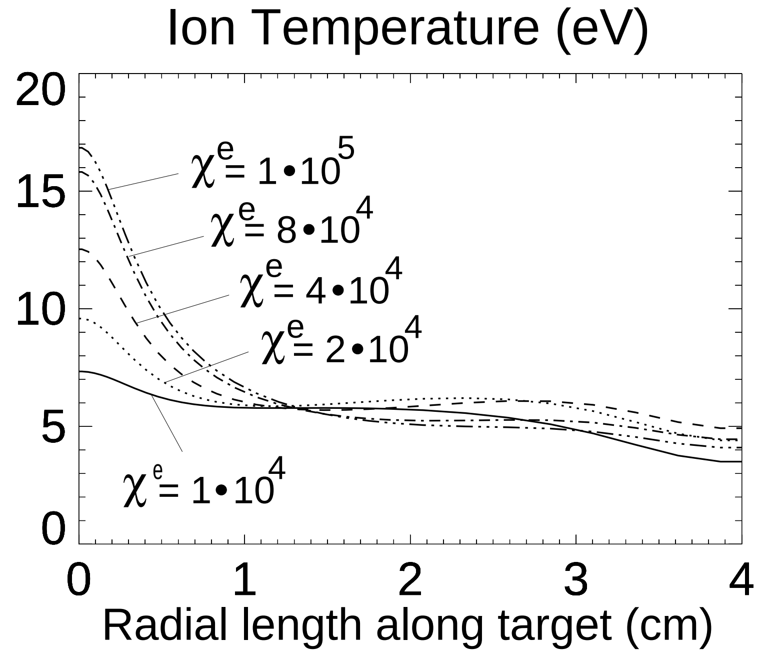

Figure 7.16. Ion temperature vs. χ^e along the target plate. The ion temperature increases due to the volumetric energy source terms arising from the ionization process (see text).

the system at the core boundary (for these two cases, the radial gradients at the core and wall are essentially unchanged). This additional heat is conducted to the target and is responsible for the rise in temperature.

Finally, notice that the ion temperature profile is modified, primarily towards the inboard side of the target plate. This change is due to three processes: the equilibration energy source term, $\kappa_{\text{eq}} k \left(T_e - T_i\right)$, maintenance of overall pressure balance as discussed earlier, and ionization energy *input* (i.e. the term involving ϵ_i in Equation 7.6). Since the net volumetric sources are considerably lower in the outboard regions, there is significantly less ionization energy input. The pressure balance makes up the remainder.

7.4.4 Anomalous Ion Diffusivity

Finally, in the spirit of the previous two sections, we vary the ion diffusivity through the values, $\chi^i = \in (0.1, 0.2, 0.4, 0.8, 1.0) \cdot 10^4$. The resulting plasma density, and electron and ion temperatures are shown in Figures 7.17, 7.18 and 7.19, respectively.

In this case we get the expected broadening of the ion profiles at the target, but the plasma density and electron temperatures remain essentially unchanged. And notice that, contrary to the previous case, the absolute magnitude of the ion target temperature remains unchanged. Since the parallel ion conduction is an order of magnitude less than that of the electron conduction, much smaller values of the radial heat diffusivity produce comparable broadening of the ion profile. Our parameter variation here results in far smaller perturbations to the overall energy put into the system. As a result, the ion profile can be broadened without significantly affecting the other variables.

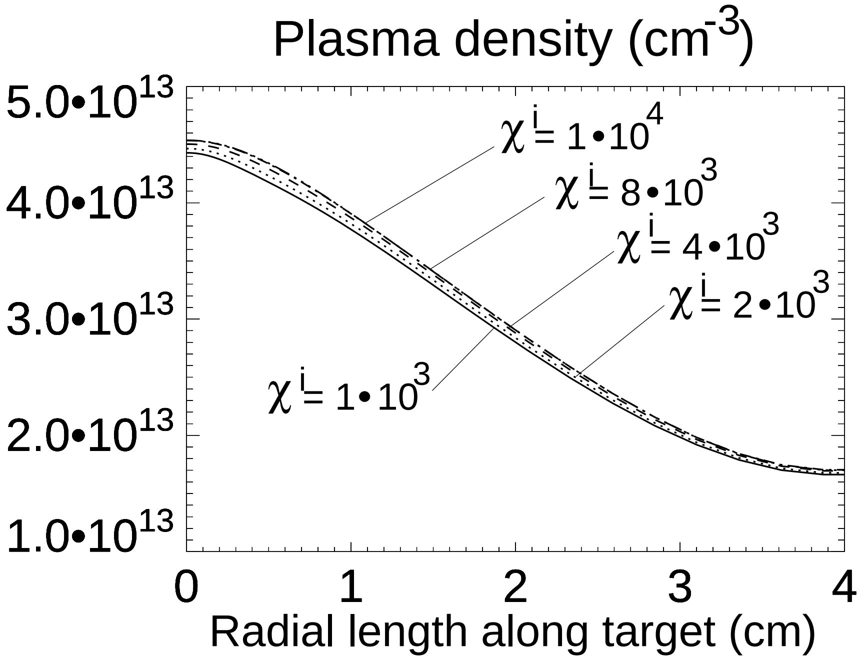

Figure 7.17. Plasma density vs. χ^i along the target plate. The density profile is unaffected by changes to the ion radial transport.

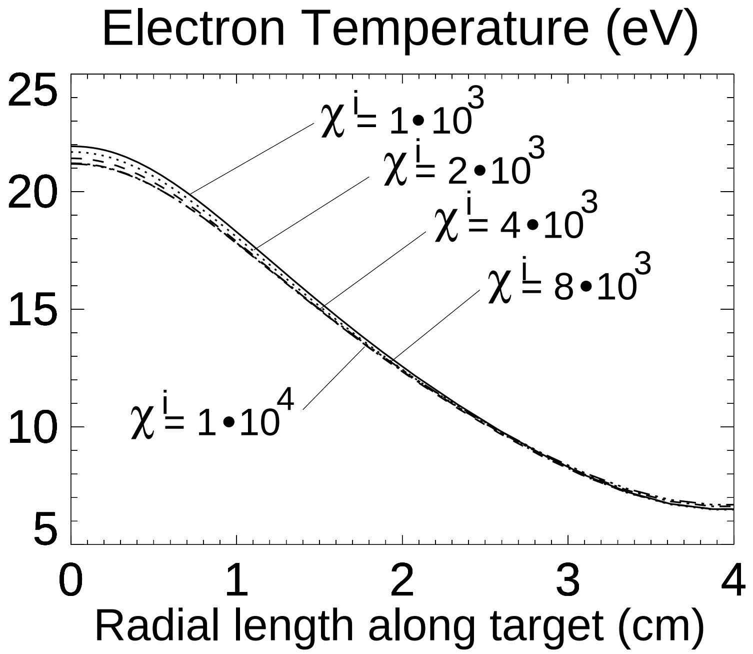

Figure 7.18. Electron temperature vs. χ^i along the target plate. Like the particle density, the electron temperature is insensitive to changes in the structure of the ion profile along the target plate.

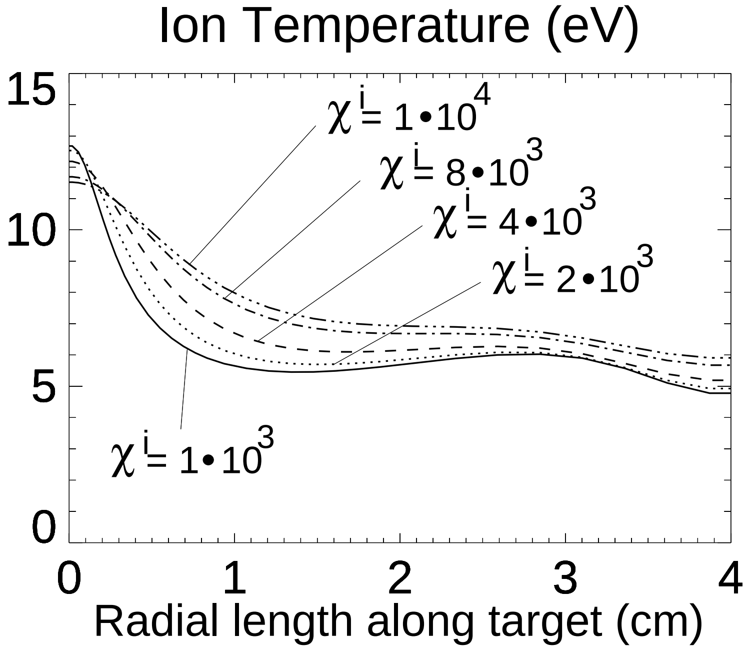

Figure 7.19. Ion temperature vs. χ^i along the target plate. The expected broadening of the ion temperature profile is apparent, along with a slight increase in total ion energy to the plate that corresponds with the input increase.

7.4.5 Ionization/Charge-Exchange Rate Data

Recently, Post[150] provided an atomic reaction data set which augments the compilation of Janev, et. al.[151], specifically in the very low electron temperature regime (i.e. $T_e \leq 5\text{eV}$). Dynamics of the low temperature divertor regimes (e.g. gas-target and volumetric recombination regimes) are dominated by atomic processes so that incorporation of this data is essential for evaluating "physics-based" solutions to the high heat flux problem in the current ITER divertor design.

For our model case here however, temperatures are considerably higher than the those required in the gas-target and volumetric recombination scenarios, and the changes in the data base at higher temperatures are less significant. It is nevertheless useful to compare the results obtained using this new data to those obtained using the simple analytic formula, Equation 7.7. We compare the default results with those obtained using $\langle \sigma v \rangle^{\text{ie}}$ from data table lookup functions that were constructed from bilinear interpolations (on the log-log coordinates) of the Post data. Also, we look at the effect of the charge-exchange rate, given in the model problem as a constant. We use $\langle \sigma v \rangle^{\text{cx}} = 10^{-8}, 5 \cdot 10^{-8}$ and the fit proposed in [151], which lays in all cases between these values.

The analytic electron impact ionization rate parameter function is plotted in Figure 7.20, while the Post data was plotted in Figure B.2. The resulting plasma density, and electron and ion temperatures are plotted in Figures 7.21, 7.22 and 7.23, respectively. The particle source region in front of the divertor plate is roughly where $15 \leq T_e \leq 25$, but from the rate parameter plots, we see that the Post data is larger than the Braams fit in this range. Thus the particle source is stronger in the Post case, and the density near the target increases. The electron temperature

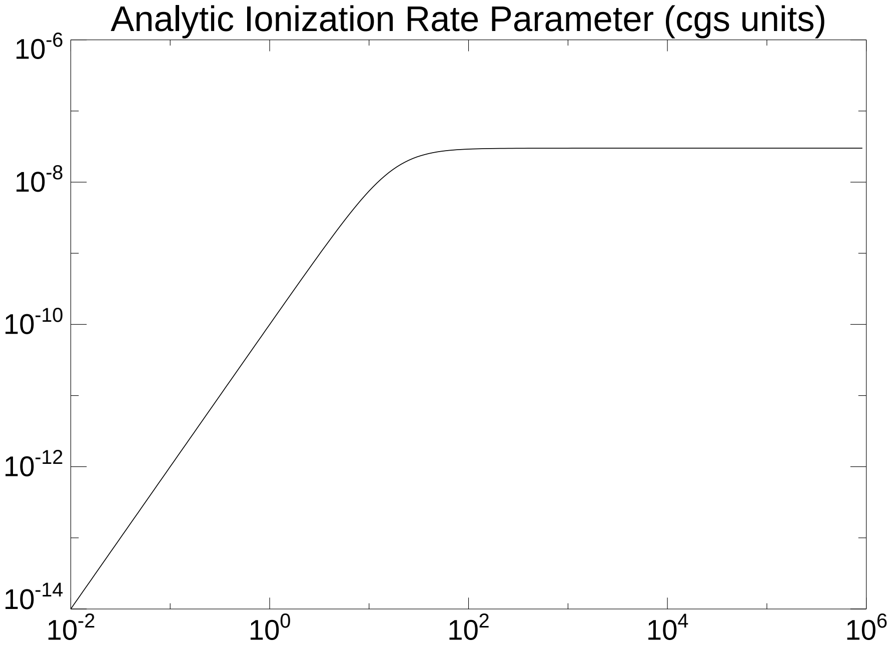

Figure 7.20. Analytic approximation to electron impact ionization rate parameter. Note that this parameter is independent of plasma density (see the following figure). Also, this approximation is considerably larger than than the more accurate data.

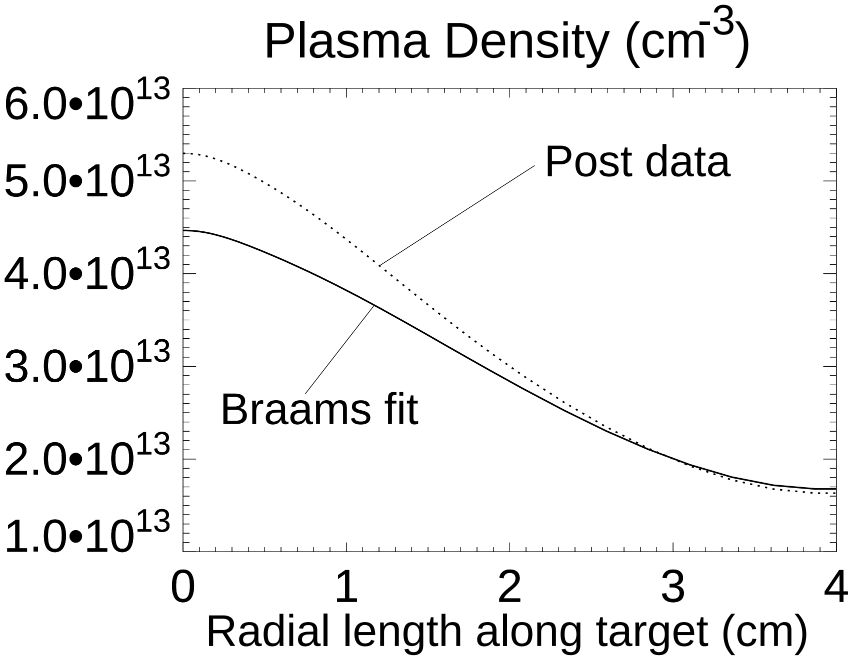

Figure 7.21. Plasma density vs. ionization rate parameter along the target plate. Even though the results are significantly different for these mild cases, the results would be much more dramatic if the low temperature regimes, $T_e < 3$ eV, were accessed.

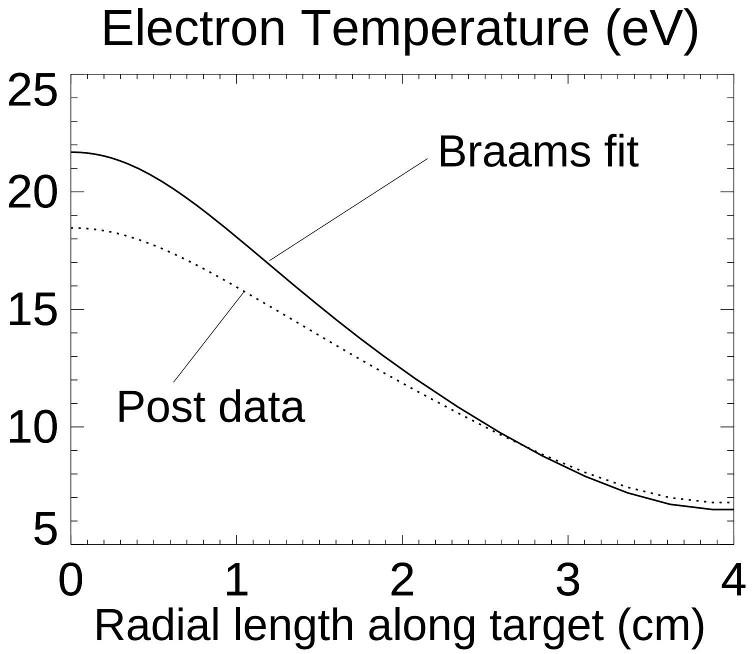

Figure 7.22. Electron temperature vs. ionization rate parameter along the target plate. This plot is just the inverse of the previous plasma density result, as increased ionizations lower the temperature and increase the local plasma density.

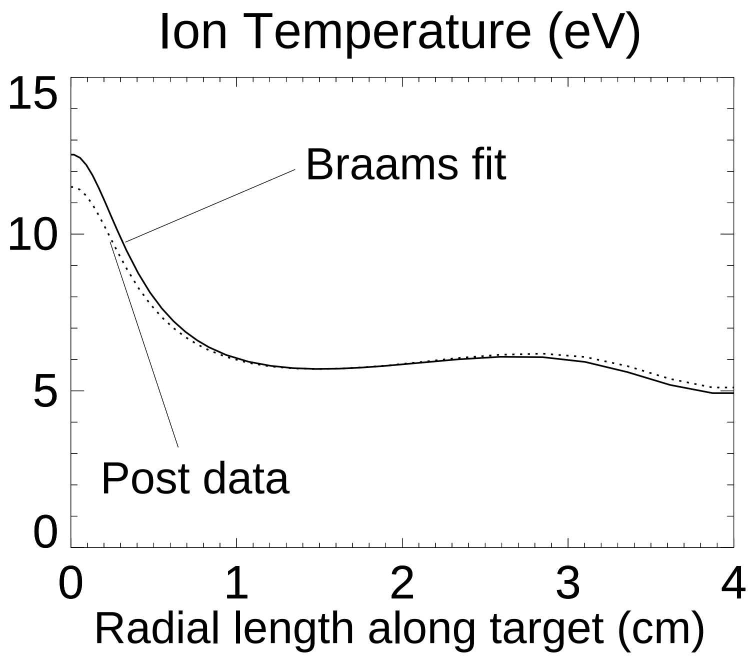

Figure 7.23. Ion temperature vs. ionization rate parameter along the target plate. The ion temperature is again rather unaffected by these rate coefficients.

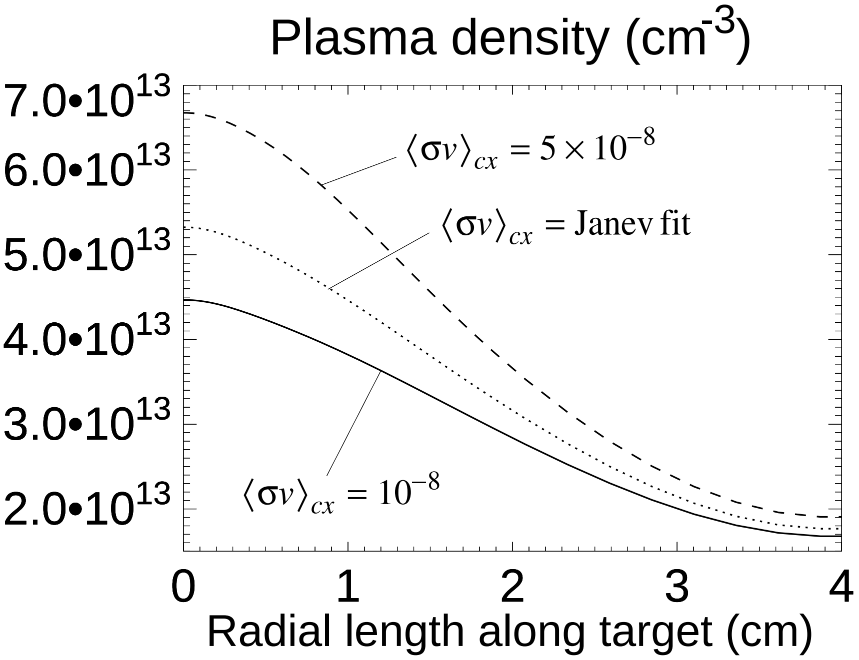

Figure 7.24. Plasma density vs. charge exchange rate parameter along the target plate. Increasing the charge exchange shortens the neutral particle mean free path. For a given production, the neutral density increasing, providing a greater source for ionization.

drops mainly due to the increased ionization events. The ion temperature increases, slightly, to make up the pressure balance along the field lines.

Next, the charge exchange rate parameter was varied as described above. The resulting plasma density, and electron and ion temperatures are plotted in Figures 7.24, 7.25 and 7.26, respectively. The neutral diffusion coefficient is inversely related to the charge exchange rate parameter so that an increase in the frequency of charge exchange events effectively decreases the diffusion step length, and the neutrals penetrate less into the plasma. But since neutrals are created at a constant rate (target plate recycling), the neutral density will increase just in front of the target (i.e. in the particle source region). The increase raises the plasma

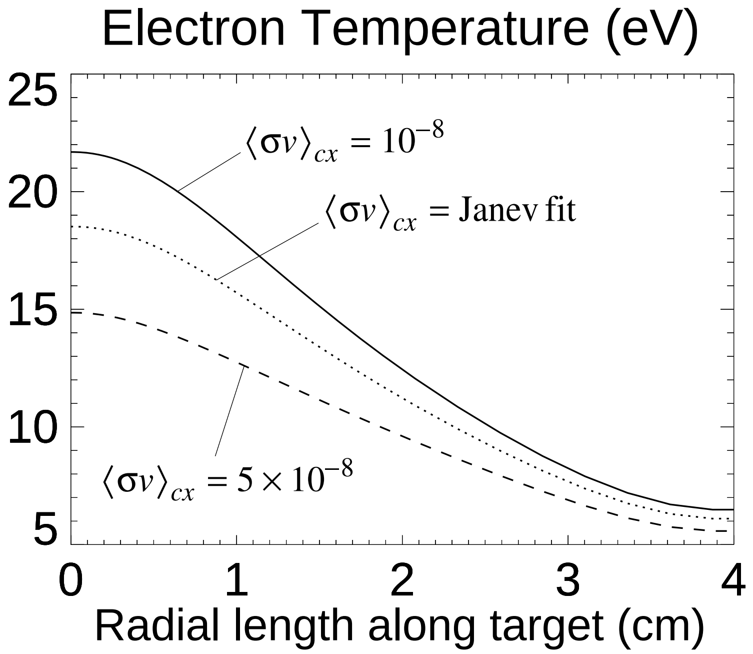

Figure 7.25. Electron temperature vs. charge exchange rate parameter along the target plate. The increase in ionization events removes more heat from the electron channel, reducing the target temperatures.

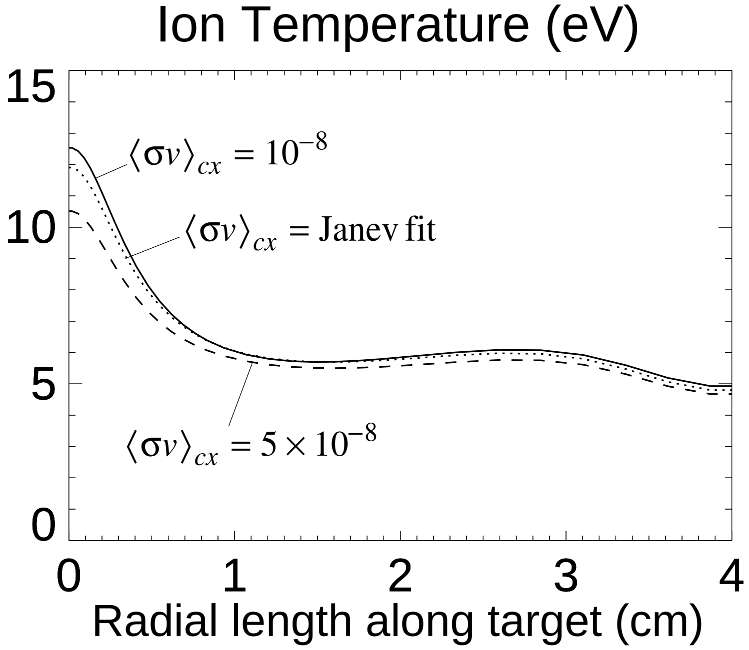

Figure 7.26. Ion temperature vs. charge exchange rate parameter along the target plate. The charge exchange process conserves mass, momentum and energy, leading therefore to minimal effects on the ion energy distribution.

particle source rate, and subsequently, the plasma density at the target—note that there is an inherent amplification of events. We can easily verify from the plots, that increasing the charge exchange reaction rate coefficient indeed increases the plasma density at the target, and decreases the electron temperature. Again, the ion profile adjusts only slightly to reestablish pressure balance in the field lines. Apparently, the correct atomic physics data is vital for obtaining accurate results. In fact, owing to the sheath dynamics, the factor-of-two change in electron temperature would lead to corresponding increases in heat flux and sputtering at the target.

7.4.6 Magnetic Field Projection

For the last in this series of parameter variations, we compute the effects of the magnetic field projection factor, $\mathcal{P}$. With decreasing projection, the field line connecting the main source of particles and heat (core) is effectively lengthened, and radial transport becomes increasingly competitive with transport along the field lines. A shorter connection length will therefore result in less radial spreading of the temperature profiles, and higher peak temperatures at the target. The plasma density will decrease accordingly.

Results are displayed in Figures 7.27, 7.28 and 7.29, respectively. Here, $\mathcal{P} \in (0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09)$, representing a full range of expected values. It is apparent that the system responds according to prediction.

7.5 Heat Flux Limiter

From the results of our test model problem, the ratio, r_T, of the separatrix electron temperature to that of the divertor is typically $80/15 \approx 5$. Measurements

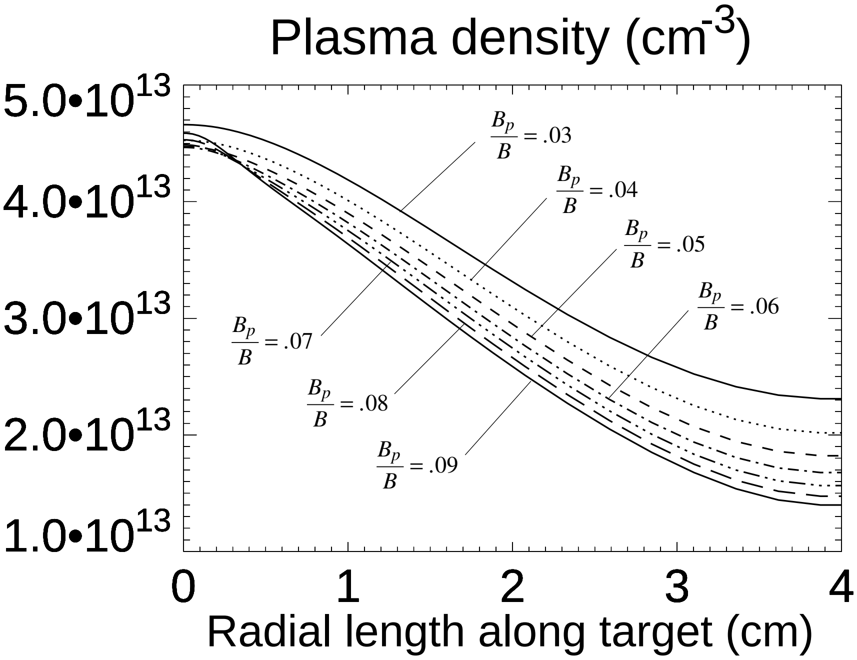

Figure 7.27. Plasma density vs. magnetic field projection along the target plate. As the projection increases, particle are swept though the systems much faster, decreasing their density.

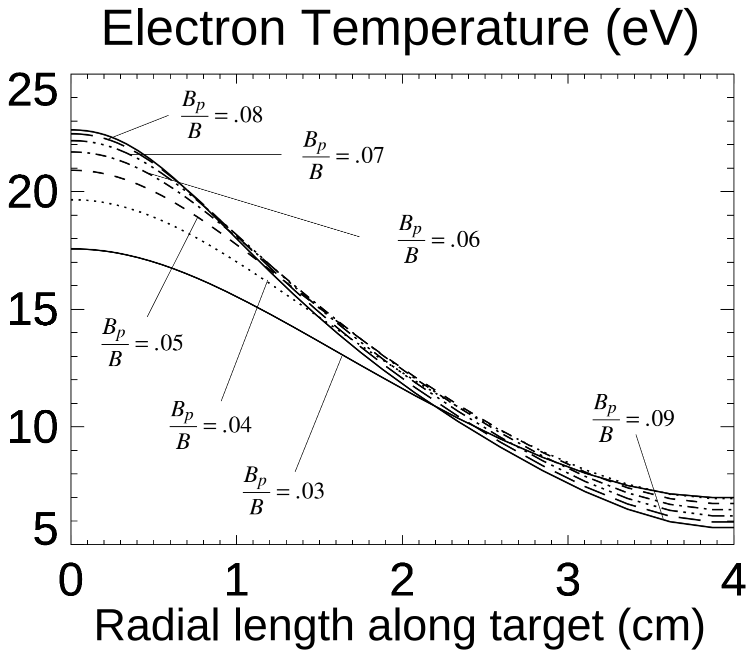

Figure 7.28. Electron temperature vs. magnetic field projection along the target plate. A shortened connection length with increasing projection leads to higher target temperatures.

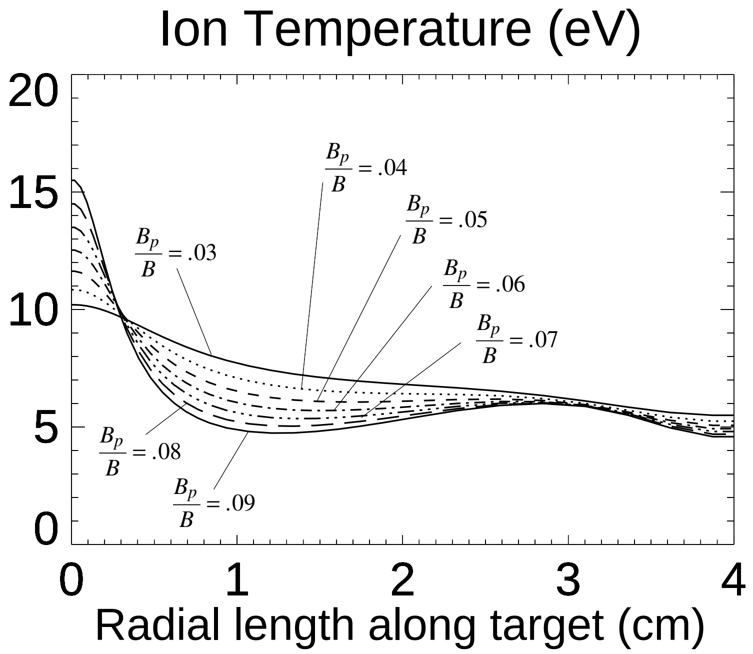

Figure 7.29. Ion temperature vs. magnetic field projection along the target plate. With an increased magnetic field connection length, the ions are allowed to diffuse radially.

from experiments (see e.g. [152]) likewise indicate values like $r_T \sim 4-6$. However, based on the classical parallel electron thermal conductivity, one would expect $r_T \sim 1-2$. The reason the test model agrees favorably with the experimental data is somewhat hidden in the expression for parallel electron heat conductivity (see Equation **??**), where classical Spitzer conductivity has been modified so that the computed heat flux is never greater than $\alpha_e n v_{th} T_e$ (there, $\alpha = 0.12$). This ansatz implies that the maximum heat flow attainable is some fraction of the average internal energy of the electrons carried at their thermal velocity.

The form of the model electron conduction flux limiter is common in tokamak edge modeling and has been suggested in the modeling of other physical systems as well, including laser fusion (see e.g. [153, 154, 155, 156]), radiation hydrodynamics (see e.g. [157]) and plasma processing (see e.g. [158]). However, the exact constant, α, is the subject of some debate and depends strongly on the application. For example, in a cursory literature search, we found values ranging from .03 to $\sqrt{3}$ (see Ref. [159], and citations therein). The purpose here is to point out how sensitive this benchmark case is to the coefficient used in this flux limited model, suggesting that perhaps the form of the limiter does not properly account for the dominant physical processes involved. We then present an overview of models proposed to correct these deficiencies, and discuss their implications to the numerical methods that would be required to include such models into codes like FMS2D.

Consider the plots of plasma density and electron temperature, shown in Figures 7.30 and 7.31, respectively. There, the default model was solved using a flux limit parameter, $\alpha^e \in (0.1, 0.2, 0.3, 0.4, 0.5)$. Note the huge variations that result, particularly as the flux limiter factor changes from 0.1 to 0.2. The "accepted" value

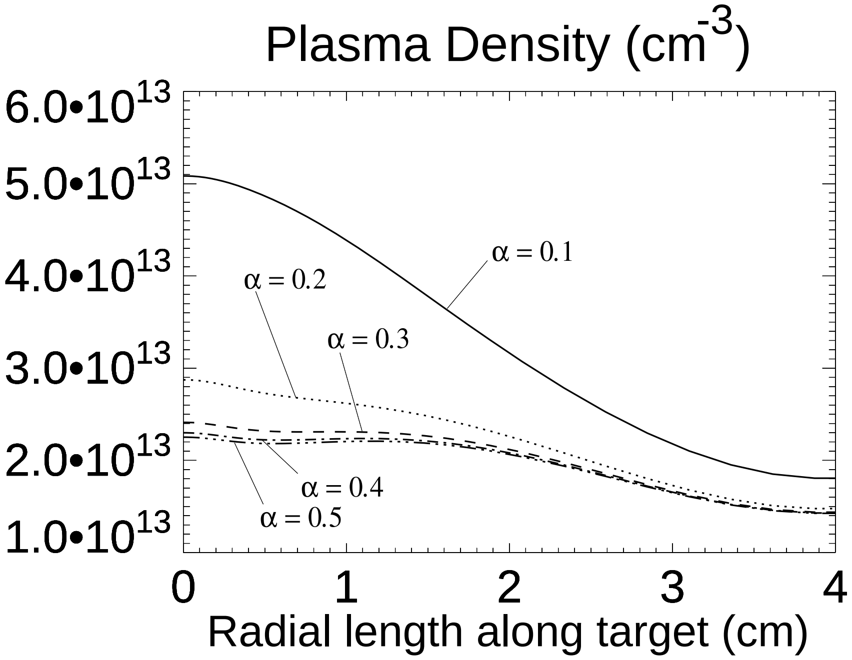

Figure 7.30. Plasma density vs. flux limit factor, α^e, along the target plate

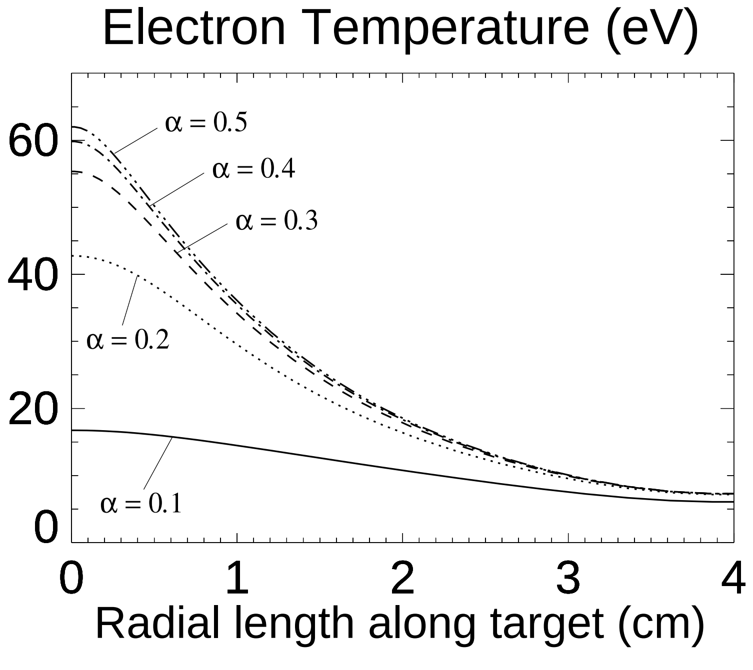

Figure 7.31. Electron temperature vs. flux limit factor, α^e, along the target plate

in present edge modeling apparently is 0.12, which lays in the midst of this zone of strong dependence. The ion temperature is not plotted because of its rather weak variation under these conditions.

From an engineering point of view, the electron temperatures resulting from the model using $\alpha^e = 0.2$ will give rise to a large electrostatic sheath potential at the target, which accelerates heavy impurities towards the surface, generating intolerable erosion and impurity generation problems. But the plasma computed with $\alpha^e = 0.1$ is suitable for experimental operation. We therefore should be gravely concerned with the details of the heat flux limiter formulation if we are to confidently predict within which of the two regimes a given divertor scenario will operate.

7.5.1 Discussion: Flux Limited Diffusion

The limited research in this area applied to the tokamak edge problem (see Ref. [160, 88]) is rather heuristically developed, due to the complexity of plasma transport. The procedure for deriving an improved heat flux-limiter begins by recognizing that the particles primarily responsible for heat flow are at energies of $4 - 6T_e$, i.e. in the high energy tail of the Maxwellian distribution. Since the collisional mean free path for a charged particle *increases* with energy, the most of the heat flux at any location in the plasma has arrived there from some distance away. These conditions violate a fundamental assumption in the derivation of Spitzer conduction–namely, that of slowly varying profiles with respect to the collisional mean free path.

In Ref. [160], Krasheninnikov separates out the electrons within the energy range responsible for most of the heat flow. An approximate form for the high-energy electron distribution function was proposed which enabled derivation of *non-local* analytic solutions to the Fokker-Planck equation for the resulting heat and particle flow. The proposed distribution was valid only for high-Z plasmas. To complete the problem specification, an absorbing wall was assumed for electrons of higher energy than the confining sheath potential. This solution was compared against a full numerical Fokker-Planck calculation for the the case of thermal decay of the temperature perturbation in a uniform plasma. The results of the approximation agreed very well to the "exact" Fokker-Planck calculation. However, it was subsequently pointed out in Ref. [88] that the validity of Krasheninnikov's approximate form breaks down at energies near the loss cone in the long mean free path limit. Since the validation of the model was performed for a uniform plasma, these limitations had minimal effects on Krasheninnikov's benchmark problem. It was suggested that this limitation severely handicaps the model when boundary effects are important.

A new model was subsequently presented in Ref. [88] that was based on properly accounting for the effects of a nearby wall while still allowing short gradient scale lengths (measured with respect to the collision mean free path). The authors point out that for DIII-D parameters, $\lambda_{mfp}/L \sim 0.1 - 0.2$ (L = electron temperature parallel gradient scale length) for thermal particles, but ~ 1 for particles at energies comparable to the confining electric potential energy, $e\Phi$ of the combined sheath and presheath—and still larger for electrons in the very high energy tail of the distribution function. Closed form expressions are derived for four energy groups of electrons: fully collisional, marginally collisional yet confined to the electrostatic sheath potential, marginally collisional electrons that escape the sheath, and collisionless high-energy particles that exit the system at the boundary directly. Four corresponding heat flux expressions are constructed, and the total heat flux is computed as the sum of each of the contributions. Division among the groups involves the distance to the nearest boundary along the field and the local collision mean free path.

In both the previous efforts, the resulting transport equations contain a significant *non-local* component. Physically, this arises because heat is transported through regions of low collisionality. Since the models are built from expansions of the particle distribution functions, accounting for energy transported via less collisional parts of the distribution function must integrate over larger domains.

However, non-local dependence is difficult to handle numerically. In terms of discretizations, the stencil approximating a nonlocal differential operator involves a large number of grid points. The numerical operations required to compute the discretized equation becomes excessive for explicit schemes that compute over many time steps. For implicit schemes, a linearized operator (e.g. the Jacobian operator in a Newton iteration scheme) becomes less sparse and less diagonally dominant, increasing the computational work required to obtain solutions. From a gridding standpoint, nonlocal plasma parallel heat conductivity models also require that the domain be split exactly along the magnetic flux surfaces in order to enable numerical integration along magnetic field lines.

Alternatively, in the fields of radiation hydrodynamics and neutral particle diffusion, local expressions have been derived for diffusion coefficients that preserve causality (i.e. a Fick's law diffusion that never allows more flux than is physically attainable)[161, 162]. Ref. [163] in particular, describes a diffusion treatment that additionally is “naturally” flux-limiting and thus relaxes many of the restrictions usually necessary for valid application of diffusion theory. In the reference, expressions for diffusivity were derived by employing an approach similar to the standard Chapman-Enskog development in the kinetic theory of gases[63] (on which the Braginskii model was developed). Knowing *a priori* that ones requires a flux limited expression for heat flux, a suitable closure is suggested.

Application to a collisional plasma will be considerably more complex than the procedure used in Reference [163] for an isotropic collision operator. The Braginskii analysis without this added restriction is already very, very complex. However, work in this area is obviously required if we wish to remove this large uncertainty in the model. Presently, we can conclude that the effect of the flux limit is strong enough even in the range 0.1 - 0.2 to cast significant doubt on the edge modeling currently published.

7.6 Conclusions

The ASDEX benchmark problem was defined and then solved using the FMS2D code. We found performance of the code a bit unsatisfactory, but able to complete a limited amount of testing runs. We found the model to contain rather un-resolvable features. If the wall boundary condition in that model is modified slightly, resulting profiles are much better behaved, and a converged steady profile is achievable.

The ASDEX model was then employed in the code to ascertain the nature of influences of the various model parameters. Using a simple pressure balance, many of the results can be interpreted easily. We verified in general that decreasing a divertor target temperature does not necessarily decrease the heat load on the material surface there. If there is no dissipation of momentum in the SOL plasma, a temperature decrease will lead to a rise in particle density and increased surface recombination. The two effects balance roughly.

The *ad hoc* method of limiting electron heat flux is seen to have a significant effect on the computed temperature profiles along the divertor target. This effect has been known for several years now, and some research exists to suggest other simple models. The results of this study are presented to underscore the urgency of those efforts. Apparently, uncertain values of parameters in the limiter form lead to order-of-magnitude changes in the solution profiles—making the difference between viable and impossible tokamak scenarios.

Chapter 8

PISCES Modeling

This chapter is devoted to a study of the numerics involved in computing gas-target plasma conditions created in the PISCES facility at UCLA[51]. A model of this plasma has been under development for more than 1.5 years, primarily by L. Schmitz and B. Merriman for use in a 1-1/2 dimensional code that is closely related to FMS2D[164]. We have inserted that model into the FMS2D code to analyze the numerical convergence characteristics, and have begun two-dimensional extensions of the model for future use. The following were found:

1. Results computed with FMS2D employing the classical artificial dissipation scheme agree with those published in Reference [165]. Additionally, we found those results to be somewhat underresolved.

2. The SUPG scheme outlined in Appendix A performs significantly better on the problem than the classical scheme. However, for a strongly collapsed plasma, such as that expected in ITER scenarios, the SUPG scheme leads to node-to-node oscillations due to insufficient numerical damping. The classical scheme we employ by default in FMS2D is therefore more appropriate than the SUPG-type schemes for these types of problems.

3. Two dimensional extensions of the 1-1/2 D model are possible but prohibit resolved calculations. The difficulty is primarily due to inappropriate radial boundary conditions for the heat equation. Internal boundary layers form that cannot be resolved with our low-order numerical schemes.

8.1 Gas Target Plasma Regimes

In contrast to the high-recycling plasma scenarios discussed in previous chapters, the *gas target* regime exhibits decreases in electron temperature *and* plasma density[166, 167, 72, 165] along the magnetic field lines. This apparent violation of Equation 7.9 is possible if the plasma momentum is dissipated radially before reaching the target. Momentum damping occurs via charge exchange and inelastic collisional processes with the background neutrals, and radiation due impurity gases in the divertor region; increasing the neutral or impurity concentrations would therefore enhance this effect. The drawback of the scheme is that gas puffed into the system to enhance the effect often leads to flow reversal, where the bulk convection of plasma particles is directed away from the target. Impurity atoms may be entrained in the reversed flow, and carried into the core where they displace fuel atoms, radiate core energy and extinguish the fusion reactions. The gas target regime represents a potential solution for the high heat flux and boundary temperatures predicted for ITER, provided that stable, robust scenarios are identified where impurity entrainment is not an issue.

Initial results have been demonstrated in experiments on the linear PISCES-A divertor simulation facility at UCLA[165]. Reduced target temperatures and heat fluxes have additionally been observed during divertor gas puffing and/or impurity gas injection in the DIII-D[41], Alcator C-Mod[168] and JET[169] tokamaks.

8.2 PISCES-A Gas Target Experiments

PISCES-A[51] is a linear device wherein plasma electrons are created using a reflex arc configuration with a hot LaB_6 cathode. The electrons are confined by the Lorentz force to a steady state magnetic field, which is sustained by an array of solenoidal magnets. The cylindrical vacuum vessel is approximately 1.5m in length and 10cm in diameter. Along the field, electrons reflex between electrostatic ("sheath") potentials that exist naturally in the plasma to preserve ambipolar flows. Ions are produced via electron collisions with background neutral particles in the vessel. The ions are not confined by the applied field because of their large mass, nor are they appreciably heated by the electrons. The ion temperature in PISCES-A is approximately 2 eV due to Franck-Condon dissociation processes. The atomic and molecular Hydrogen temperatures are approximately 0.5 eV and 0.1eV, respectively. The electron temperature and plasma densities attainable in the PISCES-A are characteristic of divertor-type plasmas. Some effects of finite ion temperature, notably energetic ion fluxes to the target, can be simulated in the device, by electrical biasing of the collector plates.

In the gas target experiment, a circular tube, made of anodized aluminum (length = 0.9m, inner radius Δr = 2.25 cm) is located 0.55 m from the source, and placed coaxial to the 1700 G magnetic field. Hydrogen gas can be fed into the system either near the source, or near the center of the target plate at the back end of the tube (opposite the cathode source). Neutral pressure is monitored at 3 axial locations, and a spatially fixed Langmuir probe is used to measure the electron temperature and plasma density along the centerline of the machine; the probe is moved between measurements to obtain axial profiles. A second, fast-scanning probe is employed at discrete axial locations to obtain radial scans of plasma density and electron temperature. H_α light is used to distinguish atomic and molecular Hydrogen contributions to the total neutral pressure.

Recent data from the PISCES gas target experiment[170] indicates that a stable gas target plasma is achieved when the total neutral particle density, $n_0 \approx 2 \times 10^{21}$. Here, the plasma ($n \leq 3 \times 10^{19} m^{-3}, T_e \leq 20$eV) is observed to detach from the end plate, and up to 97% of the electron heat flux (≤ 7 MW/m^2) is dissipated by Hydrogen line radiation, dissociation and ionization losses. This is the condition we would like to investigate.

8.3 1-1/2 D PISCES Model

From the point of view the tokamak edge fluid-type descriptions, only a limited number of models exist at present for the gas target regime (cf. [61, 98, 170, 171]). The models are still under development, and are particularly challenging numerically to solve for converged steady solutions. In this dissertation, we study the model developed in the UCLA PMI group[165].

The one-and-a-half dimensional model for the PISCES experiment is discussed in detail in the applicable reference. Summarizing, FMS2D solves 5 conservation equations, for the plasma density, temperature and parallel velocity, as well as the diffusion of atomic and molecular Hydrogen, along the center axis of the PISCES machine (see Figure [?]). The "1/2" dimension comes from assuming self-similar radial profiles shapes parameterized by the value along the axis of the device (due to the small machine size, Bessel profiles are assumed), and then using that shape to estimate radial transport. The radial losses are then included as “sinks” of particles, momentum and energy. A simplified set of Braginskii[64] equations are applied parallel to field to model the flows of plasma along the machine axis. The relevant equations are

$$
\begin{aligned}
\frac{\partial n}{\partial t} + \frac{\partial \left(nv\right)}{\partial z} &= n_H n \langle \sigma v \rangle^{\mathrm{eii}} + 2 f^{+} n n_{H2} \langle \sigma v \rangle^{\mathrm{Diss}} - n \frac{D_\perp}{\lambda_\perp^2} && (8.1) \\
\frac{\partial m^i n v}{\partial t} + \frac{\partial}{\partial z} \left(m^i n v^2 n k T_i + n k T_e - \Pi_i \right) & && (8.2) \\
&= -m^i n v \left(\langle \sigma v \rangle^{cx} + \langle \sigma v \rangle^{in} \right) - m^i n v \frac{D_\perp}{\lambda_\perp^2} && \\
\frac{\partial}{\partial t} \left(\frac{3}{2} n k T_e \right) + \frac{\partial}{\partial x} \left(\frac{5}{2} n k T_e v + q_z^e \right) &= -\epsilon^{\mathrm{eii}} n n_H \langle \sigma v \rangle^{\mathrm{eii}} && (8.3) \\
&- \sum_{j}^{N_{diss}} \epsilon^{\mathrm{j}} n n_{H2} \langle \sigma v \rangle^{\mathrm{diss}j} - \sum_{j}^{H} 3 \frac{m^e n}{m^j \tau_e^j} \left(k T_e - k T_j \right) - Q_0^e && \\
\frac{\partial n_H}{\partial t} - \frac{\partial}{\partial z} \left(D_H \frac{\partial n_H}{\partial z} \right) &= 2 \left(1 + f^{+} \right) n_{H2} n \langle \sigma v \rangle^{\mathrm{diss}} && (8.4) \\
&- n_H n \langle \sigma v \rangle^{\mathrm{eii}} + R_H n \frac{Dperp}{\lambda_\perp^2} - C \cdot 0.3 \frac{4}{6} \frac{n_H v_{th}^H}{\Delta r} && \\
\frac{\partial n_{H2}}{\partial t} - \frac{\partial}{\partial z} \left(D_{H2} \frac{\partial n_{H2}}{\partial z} \right) &= \left(1 - f_{\mathrm{targ}} \right) \bar{x} S_{\mathrm{puff}} && (8.5) \\
&- n_{H2} n \langle \sigma v \rangle^{\mathrm{diss}} + R_{H2} n \frac{Dperp}{\lambda_\perp^2} + C \frac{1}{2} \cdot 0.3 \frac{4}{6} \frac{n_H v_{th}^H}{\Delta r} &&
\end{aligned}
$$

Here, all the variables have their usual definitions (cf. Chapter 3). The additional variables are: f^{+}, the fraction of H_2 dissociation leading to H atoms; the radial gradient scale length, $\lambda_\perp = \Delta r / 2.405$, where Δr is the radius of the PISCES device ($\Delta r = 2.25\mathrm{cm}$); $D_\perp$, the plasma radial diffusivity is the sum of Bohm (cf. Reference [17]) and classical, computed with a collision frequency that includes the presence of neutral particles; R_H and R_{H2} are the reflection coefficients for plasma returning from the side walls as atomic and molecular Hydrogen, respectively; v_{th}^H

is the atomic Hydrogen thermal velocity, and Q_0^e is internal energy gained/lost due the particle and momentum sources (i.e. the terms that arise from algebraic manipulations of the total energy equation to obtain the heat balance equation). q_z^e is the flux-limited electron conductivity, where the limit factor set to $\alpha_e = 0.5$. The first sum in the heat equation above is over exothermic reactions, the second is over thermal equilibration contributions. S_{puff} is a specified feed rate of H_2 gas—a fraction, f_{targ}, of which is inserted only at the target; the puffing profile, $\bar{x}$, is designed to simulate the finite conductance of the tube—the integral of $\bar{x}$ is normalized to unity. The last terms in the two neutral equations represent the free flight loss of atomic Hydrogen to the cylinder wall, and its subsequent return as H_2. This effect arises due to the long neutral mean free paths, when compared with the diameter of the machine (approximately 30% of the atomic Hydrogen hitting the wall is assumed to recombine to H_2 gas, while the other 70% just reflects back into the system–molecular Hydrogen is assumed to always reflect as H_2 so there are no corresponding H_2 loss terms). All the atomic physics data (reaction cross sections and exothermic releases) are obtained from References [150] and [172].

Boundary Conditions

At the target ($z = 0$), the plasma density is computed with an 'outflow' (extrapolated) boundary condition (as discussed in Section [?]), whereas the velocity and electron temperatures are consistent with the Bohm sheath criterion

$$v = C_s = \sqrt{\frac{k\left(T_e + T_i\right)}{m^i}}$$

$$\frac{5}{2} n k T_e v + q_z^e = \frac{1}{2} \gamma C_s n k T_e$$

$\gamma = 6$ is the electron sheath transmission factor. The neutral fluxes at the target are prescribed via recycling conditions, adding the remaining portion of the gas feed to the molecular Hydrogen through the target

$$\begin{aligned} D_H \frac{\partial n_H}{\partial z} &= R_T f_H \frac{1}{2} n C_s \\ D_{H2} \frac{\partial n_{H2}}{\partial z} &= \frac{1}{2} R_T \left(1 - f_H\right) \frac{1}{2} n C_s + f_{\text{targ}} S_{\text{puff}} \end{aligned} \tag{8.6}$$

The leading factors of one half indicate the normalization of f_H, the fraction of ions coming in to the target that desorb as H_2. Also, R_T is the target recycling coefficient. Upstream ($z = 1m$), the plasma density is fixed to 3.7 $10^{19} m^{-3}$, and $T_e \equiv 18.9$eV (these are the experimentally observed values for the conditions we intend to model). The neutral species and plasma velocity are assumed to be symmetric (zero axial gradient).

8.3.1 1-1/2 D Results

As the above expressions indicate, this model is fairly complex (the model has been developed over approximately 1.5 years). See Appendix E for the implementation details of this model. The only obvious "adjustment knobs" available to match the experimental data are the overall puffing rate, the gas feed profile, $\bar{x}$, and a scale factor in front of the two contributions to the perpendicular plasma diffusivity. Also, of course there may be physical processes that have not been included. The equation set was solved with FMS2D using all the parameters specified in Reference [165]. The physics implications of the model are carefully considered in the reference, and are not repeated here. We are concerned in this chapter with the accuracy of the numerical representation of this model, particularly as it pertains to the derived physics such as the required puffing rate to sustain the stable front, and the profile of radiation heat deposition that results.

The calculation was performed first on a 128-cell grid, which was constructed following the description in Chapter C. The solution was recomputed on a "doubled" grid, containing 256-cells. The resulting profiles are overlaid in Figures 8.1 and 8.2, along with a numerically resolved calculation, and experimentally measured data. In these plots, the dotted lines are the underresolved results, the heavy solid line is the spatially converged result. The plotted points with error bars are the experimentally measured data appearing in Reference [170] The changes with resolution are indeed significant, even for this very mild set of profiles. For example, if impurity gases were present, line radiation scales roughly as the square of electron density. 30% changes to the height of the plasma density peak would lead to order-of-magnitude errors in the computed radiation heat deposition along the walls of the divertor. Indeed, the resolved calculation shows that the gas puffing and radial losses used here do not seem appropriate. The electron temperature is too low, and the plasma density peak is too high. A logical line of follow-on modeling would adjust the associated parameters to search a more suitable fit. Notice that the underresolved cases consistently error on the side of optimism.

8.3.2 Modifying Classical Artificial Dissipation

Based on the results of Appendix D, we might have expected the observed convergence behavior. We might also expect then that the SUPG scheme described there would perform much better than the classical scheme. FMS2D was modified to apply the SUPG method, and the model was re-evaluated on grids of resolution $16, 32, 64, 128, 256$. The results are presented in Figure 8.3, where marked improvements are obvious. The solution computed at the very coarse level (16

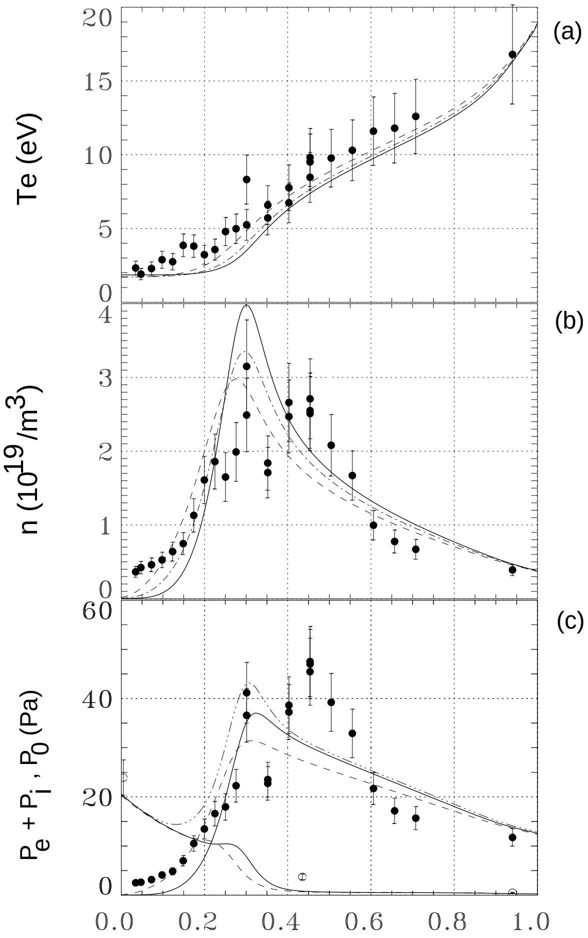

Figure 8.1. Model (a) electron temperature T_e, (b) plasma density n and (c) total pressure $p = p^e + p^i$ results vs. experimental data from PISCES. The solid line is the numerically converged result. The coarse solution fits the data quite well, but resolved calculation indicate a poorer fit. Some adjustment of the model still appears to be necessary.

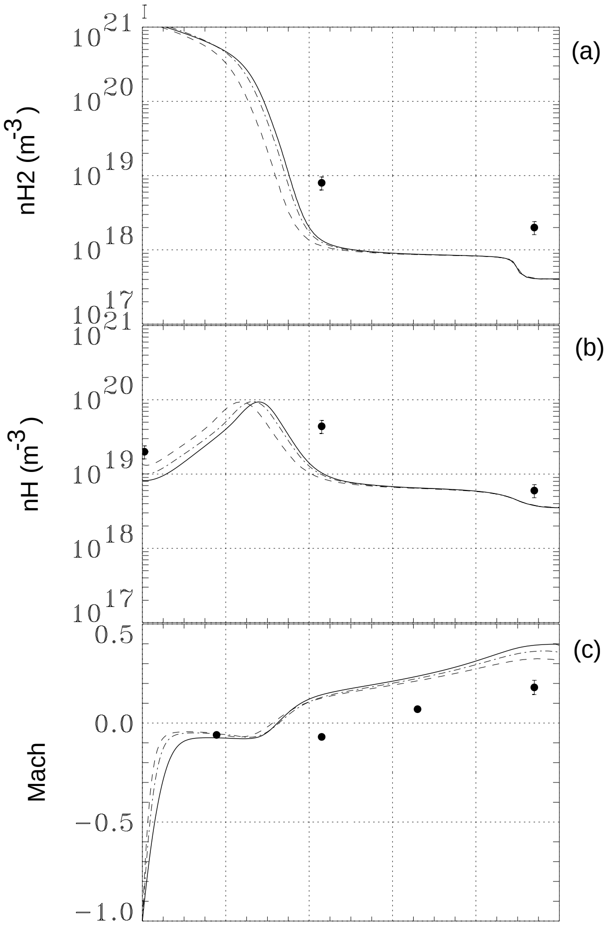

Figure 8.2. Model (a) Molecular Hydrogen density, n_{H2}, (b) atomic Hydrogen density, n_H and (c) parallel mach number results vs. experimental data from PISCES. The solid line is the numerically converged result. Experimental data for neutral density and Mach number are quite sparse. These quantities have strong a strong impact on the plasma profiles.

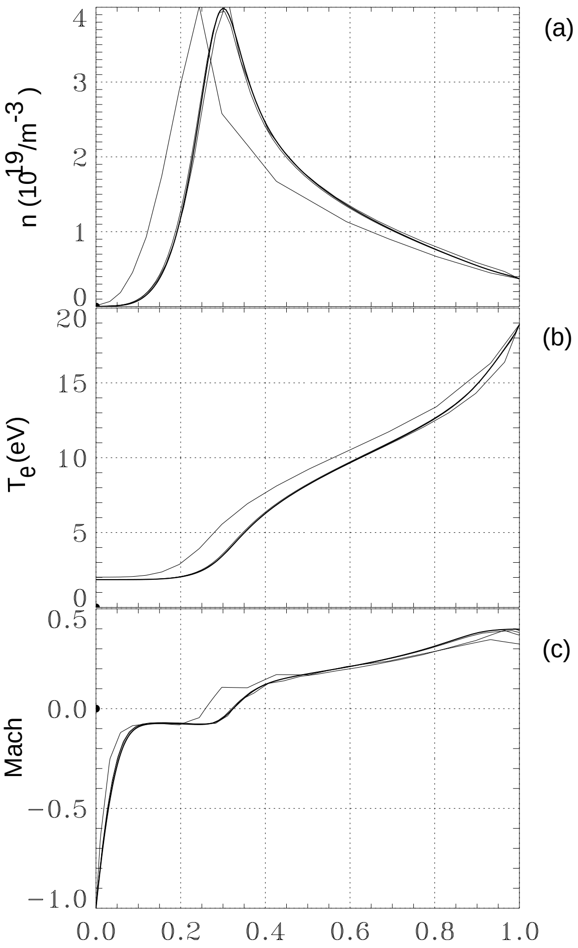

Figure 8.3. PISCES model convergence results using the SUPG discretization. Results are shown for (a) plasma density, (b) electron temperature, and (c) parallel Mach number, using 16, 32, 64, 128, 256-cell grids. Only the 16-cell case differs significantly from the converged result, which is plotted with heavy solid lines.

cells) already predicts the gross features of the solution, while the next level (32 cells) does a reasonable job approximating fine features, including the peak plasma density height and location. The levels beyond that are virtually indistinguishable from the converged solution. This implies that for these types of problems the SUPG method can be used on 64 point grids for analyzing the physics of the PISCES machine with this model.

It is instructive to repeat the tests using a "more difficult" plasma. For illustrative purposes only, we doubled the momentum dissipation losses in the model, and re-computed the solution. This is not unreasonable on physical grounds, actually. The model assumes a constant charge-exchange rate parameter equal to $10^8 \text{cm}^{-3}\text{s}^{-1}$ (see Appendix E), but Reference [172] indicates that this may be low over a significant portion of the domain of the domain. We run a test like this to judge the influence of this parameter before complicating the model unnecessarily. The results of doubling the momentum dissipation due to friction with molecular and atomic Hydrogen are plotted in Figure 8.4 This result contains a number of interesting features. The plasma computed is more extreme, in the sense that the peak in n is 30% higher, somewhat more narrow, and falls more sharply into the collapsed region in front of the target. The temperature is likewise more extremely collapsed. The cause of this behavior is apparent from the Mach number plot, where the momentum boundary layer at the target has moved into the plasma, sweeping more particles and heat from the detached zone into the target. Numerically, we note that the SUPG method computes with a grid of 64-cells, a solution that is virtually indistinguishable from the converged solution, while the solution computed with straight artificial dissipation remains underresolved, even with four

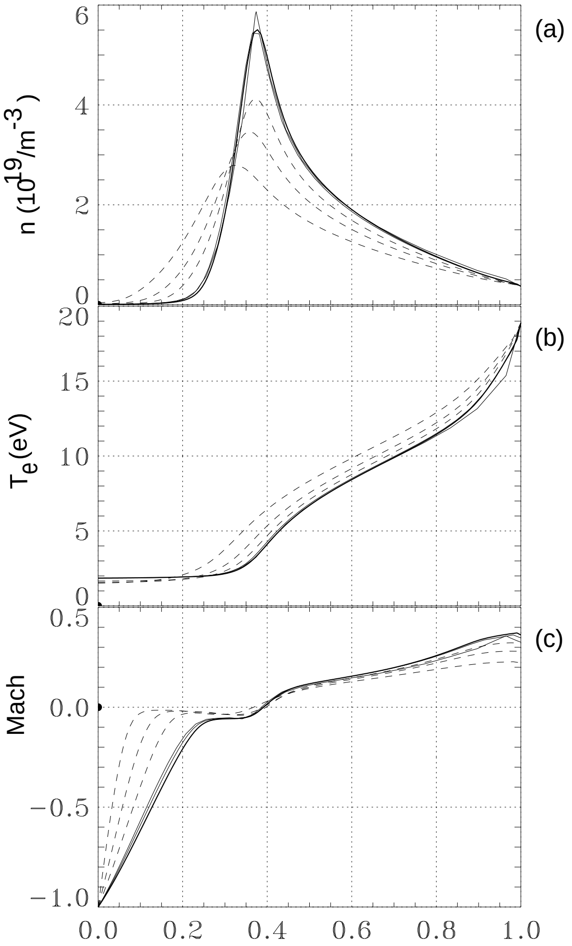

Figure 8.4. Modified PISCES model with enhanced momentum damping. Here dashed lines are the result computed with classical artificial dissipation on $64, 128, 256$-cell grids, while solid lines were computed with the SUPG scheme on $32, 64, 128$-cell grids. The heavy solid line is the converged solution for (a) plasma density, (b) electron temperature, and (c) parallel mach number.

times the number of grid points.

As an aside, the increased accuracy also had significant *practical* implications—since coarse solutions provide better initial guesses to the next finer solution, we were able to run the Newton algorithm in "direct-solve" mode, requiring only 5 or 6 iterations per solve which speeded up the computations immensely. The final mesh sequence, 128→256, converged in one Newton iteration, confirming that even at the detailed level, the profiles are essentially unchanging with mesh resolution.

Since the SUPG method depends on the *convective* flow speeds (vs. characteristic speeds used in our classical scheme), the artificial dissipation applied was effectively reduced by the factor C_s/v from that of the classical scheme. We can therefore reasonably suspect that the SUPG results might be matched simply by computing the standard artificial dissipation using the convection flow speeds instead. The case above was solved again, but this time with the classical artificial dissipation thus modified; the results are omitted because they are indeed indistinguishable from the SUPG results. We conclude that the classical scheme, as derived, was applying excessive dissipation for this problem, and we are tempted to change our definition of "characteristic velocity" given in § 5.2.2 to be just the convection speed.

Such modifications are not performed in general (i.e. "hardwired" in the FMS2D discretization) however because acoustic transport is negligible only when energy flux is dominated by conduction. In the PISCES model, there is no ion thermal transport, and electron heat flux is almost always conductive. The PISCES model has been extended to handle ITER-like parameters, and an equation, analogous to the electron equation, was added for ion heat transport. The 1-1/2 D ITER model was evaluated using the above suggested modifications to the classical artificial dissipation scheme (Results employing the classical dissipation were presented in Reference [164]). The ITER-case solutions are shown in Figure 8.5, where the velocity profiles exhibit very clearly the node-to-node oscillations characteristic of under-diffused central differencing.

The oscillations appearing in the solutions shown above decrease with mesh spacing, and the profiles will eventually converge to the correct solution. However, in the presence of strong nonlinearities, the oscillatory profiles leads to difficulty in sequencing to finer and finer grids (i.e. the coarse solutions do not provide a particularly good guess for the solution on the net finer grid). We conclude then that in problems dominated by dissipation, the SUPG method may be superior to the classical upwind scheme; but if, significant energy is transported via acoustic or convective disturbances, the SUPG will fail miserably, and we would then prefer to use the classical scheme. Interestingly, most large scale computer codes used for modern tokamak edge modeling employ the Patanker discretization[71], which is similar in spirit to the SUPG scheme. The oscillatory results have not generally been reported as a problem in these codes, but only a limited number of converged solutions for detached, collapsed plasmas have appeared in the literature. In any case, both discretizations are accessible in the FMS2D code.

8.3.3 Parametric Studies

Parametric studies are the primary method for using numerical model as an aid to understand the behavior of the plasma system. As an example, we use FMS2D to verify that the level of the gas feed plays a decisive role in detaching the plasma from the back wall of the gas target tube in the PISCES-A experiment.

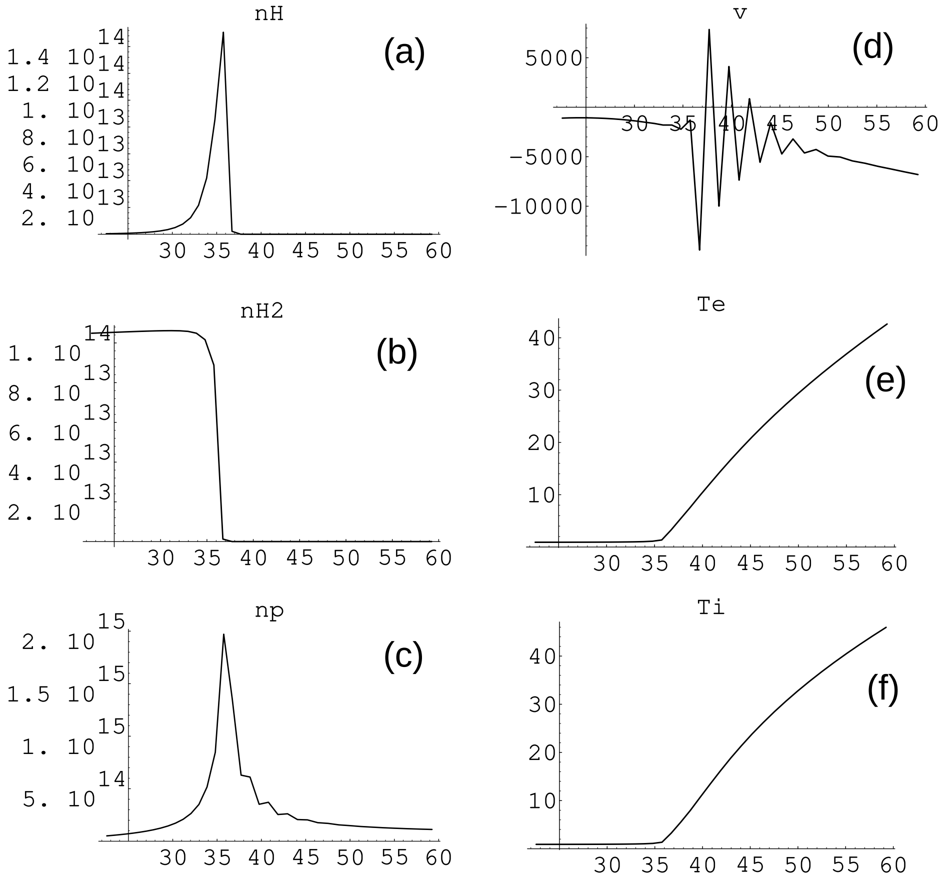

Figure 8.5. ITER impurity radiation model using *modified* classical artificial dissipation. The parallel velocity plotted in (d) displays the characteristic node-to-node oscillations which then show up in (c) the plasma density (work-in-progress plots courtesy of B. Merriman, 2/14/95).

Several solutions are shown in intermediate levels of detachment and were created by decreasing the particle source rate from the default model. The perpendicular plasma particle transport coefficient is also an interesting parameter. We look at the effects of minor changes to the anomalous contribution of to this parameter. The following solutions are found for a 128-cell grid where FMS2D was used with the SUPG "trick" above–allowed because the heat transport in the PISCES case is dominated by electron conduction. The results plotted are steady-state spatially converged solutions.

We first look at the effects of varying the molecular Hydrogen source rate. This is the primary "adjustment valve" in experiment for achieving detached plasma. We expect some of the behavior of the previous case, except that now we should have gross control of the position of the detachment front. In the reference model, H_2 atoms are puffed into the system (see the variable `TargetPuffFlux` in the model description in Appendix E) at a flux of $5.2 \times 10^{18}\mathrm{m}^{-2}\mathrm{s}^{-1}$. We adjust this value to learn how the plasma moves from attached, high-recycling cases to the detached case seen in the experiment. The results are plotted in Figure 8.6, where the puff rate was set to $(2.5, 3, 3.5, 4, 4.5) \times 10^{18}\mathrm{m}^{-2}\mathrm{s}^{-1}$, and the model of arbitrarily enhanced momentum dissipation (described above) was used. We expect that the length of the detached region will increase with the gas feed, and that the overall plasma density peak should increase. The figure verifies this intuition. The plasma density peak increases nearly linearly with gas puff rate, while the position of the peak shifts very quickly once full detachment is reached. The momentum boundary layer and zone of thermal collapse are evident.

Finally, we vary the perpendicular particle diffusivity to understand its effect on

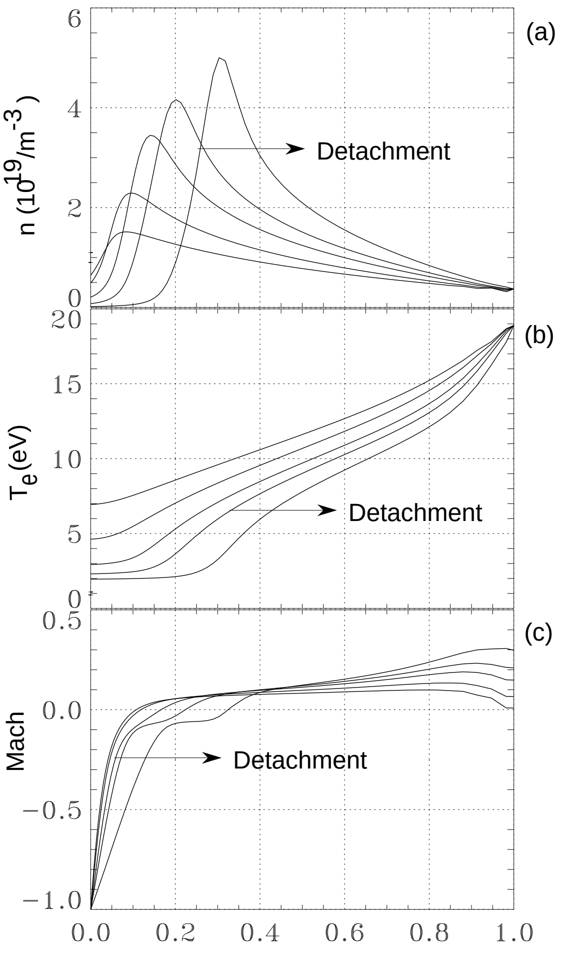

Figure 8.6. PISCES model showing detachment process with gas feed. The plasma detaches with increasing H_2 puffing. The plasma (a) is swept from the end region due to the increasing momentum boundary layer (c). The electron temperature (b) collapses correspondingly

the system. The particle diffusivity, $D_\perp$, is the sum of two terms $D_\perp = D_{\mathrm{BOHM}} + D_{CLAS}$, where D_{BOHM} is the anomalous Bohm coefficient, and D_{CLAS} is the classical value. The Bohm coefficient dominates classical transport where there is a low level of neutral particles. However, the ion-neutral collisionality increases with gas pressure near the target, and ultimately, the classical term can dominate by a factor of 20 or more. We concentrate on the anomalous component, and look at how this part affects the plasma state. We re-ran the PISCES model with enhanced momentum dissipation, modify the scaling constant in front of the Bohm contribution to $D_\perp$ (i.e. `DBohm_coeff` in the model file, see Appendix E) from 1.1 to 1.7. The results are plotted in Figure 8.7

The behavior of the plasma in response to changing the anomalous perpendicular particle diffusivity is consistent with the the fact that $D_\perp$ does not affect overall particle balance. The peak density value is constant, and only the location of the peak changes–the plasma particles are diffusing to the wall faster, and returning as neutral gas. The neutral gas then contributes to the background momentum sink.

Conclusions

In Appendix E, we make evident the complexity of the PISCES model. Through parameter studies like those summarized above, we have gained an understanding of the dynamics of the detached plasma state. Obviously, the development of the complete PISCES model involved countless such studies, and the above is intended to give just a flavor or the process.

In its present state, we find overall that the model responds in a physically

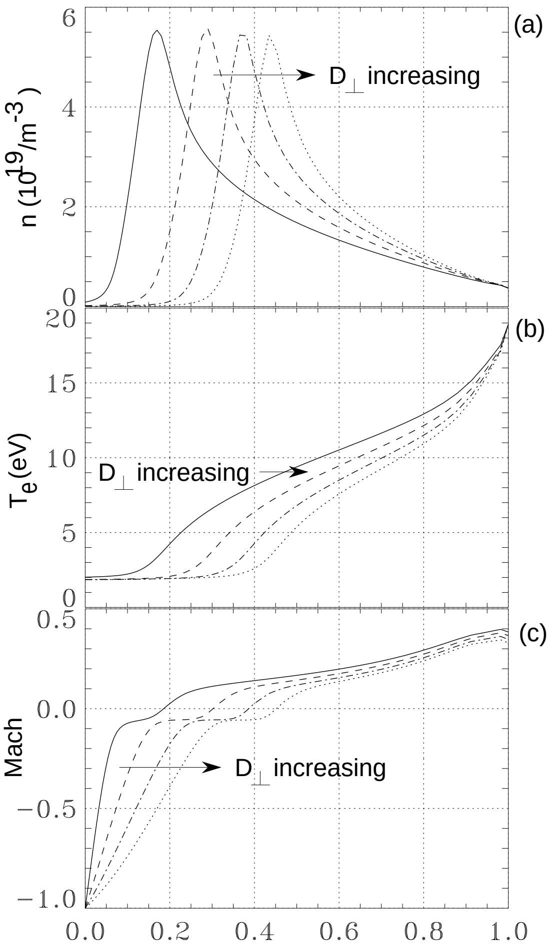

Figure 8.7. PISCES model showing ionization front motion with $D_\perp$. The detachment zone shifts essentially away from the target with increasing $D_\perp$. The peak plasma density in (a) remains constant; the electron temperature (b) and Mach number profile (c) maintain shape and values, but shift in position.

meaningful way, but apparently still lacks the correct profile control to match the measured plasma density and temperature. The most obvious shortcoming is that the neutrals do not obtain organized momentum from the collisional friction force. The topic is discussed in Reference [80], and is under investigation in our group.

8.4 2D PISCES Modeling

In this section, we take a cursory look at two-dimensional effects in the PISCES machine. Two-dimensional structure arises in the plasma, due primarily to the interaction with neutrals. Without gas injection, anomalous diffusion processes in a finite cylindrical system lead to Bessel-like solutions for the plasma density and electron temperature profiles. As these profiles impinge on a neutral gas layer, the temperature dependent ionization rates, and subsequent ionization radiation losses lead to more radially uniform profiles downstream. Additionally, in the regions of high neutral density, the electron temperature is low and the neutral particle mean free paths are considerably larger than the diameter of the device. Where the electron temperature is substantial, the mean path drops rapidly, and the neutral particles cannot penetrate into the plasma. We wish to extend the 1-1/2 dimensional model of the previous section to a 2D model, so that we might remove the restriction imposed by assuming Bessel function solution profiles.

Only a rudimentary two dimensional model was constructed before we quickly ran into computational limitations. We did however encounter some interesting cursory results that merit discussion. The model was constructed from rather simple extensions of the 1-1/2 D model presented in the previous section. The geometry is shown in Figure 8.8. The following is a list of the applicable changes:

1. The equations are posed on a two dimensional cylinder so that discrete finite volumes contain metric coefficients due to radial expansion of the coordinates. The radial extent of the domain is changed to $r \in [0, 2.25]$cm.
2. Radial dissipation of particles, parallel momentum and electron energy are specified via anomalous transport coefficients, using the methods standard to edge modeling (detailed in Chapter 3). The corresponding transport coefficients are initially set to the equal to the computed $D_\perp$.
3. The source terms related to radial transport (i.e. those$\sim n/\lambda_\perp^2$) are dropped

The upstream and downstream boundaries change to span the radius, $0 \leq r \leq \Delta r$. On the upstream boundary, we impose experimentally observed profiles for the temperature and plasma density (essentially normalized Bessel functions, $J_0(r/\Delta r)$). The flux conditions applied to the parallel velocity and neutral particles in the 1-1/2D model are simply extended across the upstream boundary. Downstream, all the flux and outflow conditions translate directly to the 2D geometry.

Additionally, two new boundaries are added to the system, at $r = 0$ and $r = \Delta_r = 2.25$cm. We assume that the machine is symmetric about its axis, so that the radial gradients at $r = 0$ of all states quantities are zero. Along the wall ($r = 2.25$cm), the plasma density is set to a "low" value, say $n = 10^{19}\text{m}^{-3}$, and the shear in parallel velocity is negligible ($\partial v/\partial r = 0$). The neutrals recycle plasma particles from the wall, so expressions similar to Equation 8.7 apply

$$\begin{aligned} D_H \frac{\partial n_H}{\partial r} &= -R_H f_H \frac{1}{2} D_\perp \frac{\partial n}{\partial r} \qquad (8.7) \\ D_{H2} \frac{\partial n_{H2}}{\partial r} &= -\frac{1}{2} R_{H2} (1 - f_H) \frac{1}{2} D_\perp \frac{\partial n}{\partial r} \end{aligned}$$

Proper radial boundary conditions along the cylinder wall for the electron temperature are not obvious. Axial Langmuir probes exist only along the central axis, and heat flux measurements are not available. Radially scanned fast probe data is available at discrete axial locations that suggests $T^e(r = 2.25) \approx 0.5T^e(r = 0)$ over the axial length of the device. This would lead to a nonlocal model in FMS2D, so we considered three alternative forms:

- *Fixed T^e.* The observed profiles suggest that at the tube entrance, $T^{we} >$ 5eV, and that at the target, $T^{er} \approx 3$. For a first try, we set the inlet wall T^{we} to 10eV, the target to 3eV, and connect the two with a linear profile;
- *Neumann conditions,* $\nabla_\perp = 0$. This implies no radial heat conduction to the wall;
- *Outflow* Neglect anomalous radial conductivity compared to the strong Spitzer heat flux. In this case, no wall condition is necessary, as the resulting equation is first order in r.

The solution profiles that results from each of these are discussed in the following section.

8.4.1 2D Results

FMS2D was used to compute solutions for the three scenarios outlined above, where we have used the classical artificial dissipation scheme to stabilize convection along and across the magnetic field. In the first set of plots (see Figures 8.9, 8.10, 8.11 and 8.12), the linear Dirichlet conditions are applied to the wall boundary. The grid is exponentially stretched in order to resolve the boundary layer that develops in the electron temperature along the cylinder wall. Presently, available computer memory put an upper limit on our runs sizes to 32x128 cell grids. 128 cells are used along the field, and 32 cells span the radius of the gas target cylinder. In the results, we see the same general features along the axis of the machine as we did with the 1-1/2 D model, except that the plasma density and electron temperatures seem to be very underresolved (apparent because the peak n value is only 10^{13}cm^{-3}, and changes significantly with grid resolution). In conservation terms, the heat and particle fluxes into the domain are being countered here by large numerical fluxes (the artificial dissipation flux). The presence of the radial boundary layer in the T_e profile indicates that the plasma near the cylinder wall is tending to relax to a profile considerably different than the linear boundary condition would imply. In fact, judging by the thickness of the thermal layer (see the contours of T^e plotted in Figure 8.13), we postulate that the radial heat conduction is negligible here with respect to the parallel flows.

If radial heat conduction to the the wall may be ignored, the linear Dirichlet boundary conditions are not appropriate (equivalently, the process occurs on a scale below our resolution ability). A modified problem is constructed by shutting off entirely the radial conduction to the cylinder wall—this is the Neumann condition.

When applied with FMS2D using currently available solution routines and computer hardware, the Neumann models converged only for grids smaller than 32x64. Beyond that, the time evolution process ceases to step forward in time to steady state, even after we allowed FMS2D to work on the problem for 7 days using a DECstation 5000/200(!). It appears that the 32x32 solution is not a good guess for the more resolved calculations; at this level, we must consider that test a failure.

Arguably, a more appropriate method to deal with the disparate transport would ignore the radial heat conduction altogether. Note that radial heat flow still still occurs in this case, because the diffusing plasma particles themselves carry their energy to the wall. In this case there no longer are any radial second-order differential operators, so the radial heat balance is transformed into a *hyperbolic* equation. This situation is exactly the opposite of that occurring in the plasma particle balance, which admits diffusive flux *across* the field and convective flux along. The hyperbolic radial operator implies that no boundary conditions on $r = \Delta r = 2.25$cm should be applied, but rather extrapolation should be used, analogous to the method used for the plasma density at the target. The results of the model with this treatment of the radial heat transport are shown in Figures 8.14, 8.15, 8.16 and 8.17. Aside from the absent boundary layer, the results look similar to the Dirichlet case above. However due to hardware limitations, this is the extent that we could explore alternative two-dimensional models. This is a whole new area needing attention in follow-on research. Figure 8.18.

8.4.2 Conclusions of Initial 2D Work

We cannot yet conclude on the correctness of the simple extension of the 1-1/2 D model into two dimensions. The boundary layer computed in the first case above may indeed be physically extant, and observable in PISCES. We found, though, that the internal features leading to the boundary layer are very robust; we obtained similar results from a simpler model that excludes the molecular Hydrogen altogether.

Though only rudimentary 2D models were analyzed in the previous section, we showed that

1. Radial conduction probably does not contribute much to the heat balance in PISCES. Thus, Dirichlet or Neumann boundary conditions do not seem appropriate along the cylinder wall of the gas target experiment.
2. The low accuracy first-order discretization has prevented us from obtaining any *quantitative* 2D modeling results to compare with data from the experiment

8.5 Conclusions of the PISCES Modeling

The focus of this chapter has been to demonstrate how robust, low-accuracy numerical schemes can be used to get a "feel" for how well a model describes a given physical system. An underlying observation is that numerical schemes used for model evaluation may play a decisive role in whether converged results are meaningful or even attainable. In its present state, the FMS2D code has been very useful for analyzing complex plasma models in one dimension. The two-dimensional models do not show quantitative agreement with the measured data, however. This is clearly the result of excessive resolution requirements that arise from our low-accuracy numerical scheme, coupled with our lack of present understanding of the radial heat flows in the PISCES experiment.

The 1-1/2 D model described in Reference [165] changes somewhat with resolution beyond that used for their study. The results of the modeling (e.g. prediction of gas puffing rates, radial transport required or profiles of radiation power deposition for a given scenario, say) may be inaccurate by up to 30% or more. We showed how the underlying numerical discretization used to solve the model can be modified slightly to obtain better accuracy for this case. But the "numerical trick" seems to work only because the heat flow in that system is dominated by conductive transport. In systems where ion heat flow is not negligible, such tricks fail because the classic convection-related numerical oscillations (point-to-point fluctuations) prevent converged solutions. Physically, this occurs because ion heat flow is convection-dominated, and the modified numerical scheme then has insufficient artificial dissipation.

Given the finite computing resources, and the above results, we conclude that robust high-order discretizations are absolutely necessary in order to understand the two dimensional flows in detached gas target divertor experiments. Using present methods, robust schemes grossly underresolve the profiles, and more accurate methods admit the classic oscillatory solutions. Physical models of the gas target regime can be developed and tested only with numerical results that are physically realistic and dependable.

We conclude with by noting that our various discretization methods span the algorithms employed in nearly all "off-the-shelf" edge plasma fluid codes in one or two dimensions. The results we find, regarding the need for quantum improvements in the numerical algorithms, should apply across the board. While the current popular focus in the field is on exploring new physics regimes, our experience indicates that the priority should be to bring the numerical techniques at least up to a marginally acceptable level.

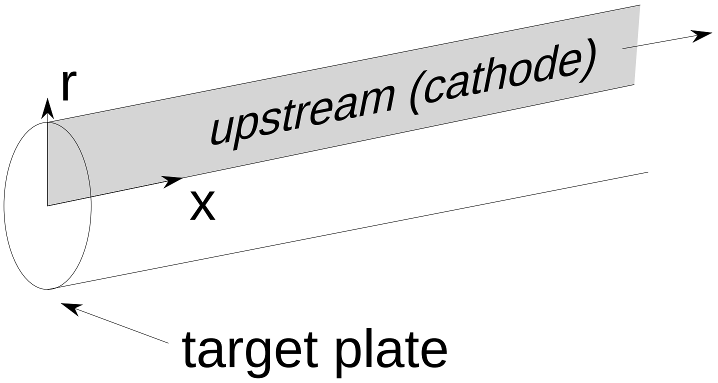

Figure 8.8. 2D PISCES geometry and coordinates.

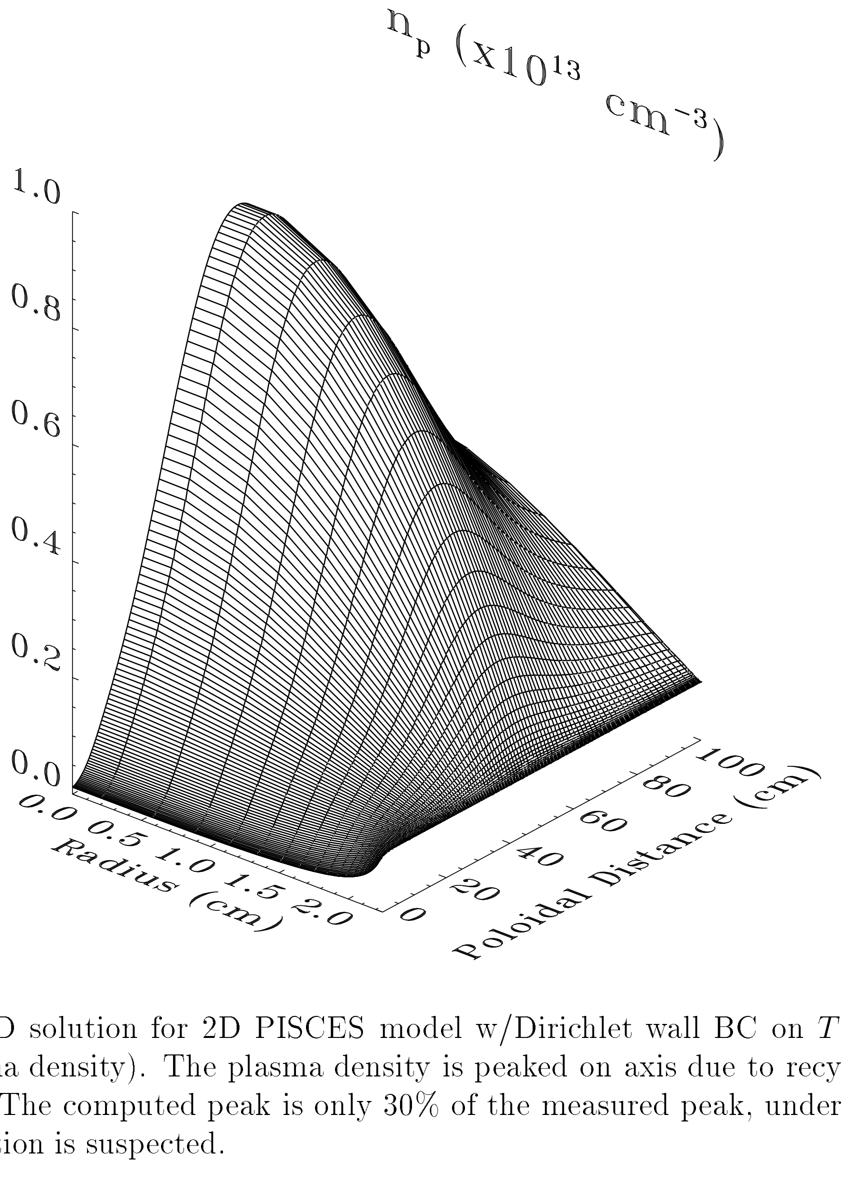

Figure 8.9. FMS2D solution for 2D PISCES model w/Dirichlet wall BC on T^e (plasma density). The plasma density is peaked on axis due to recycling. The computed peak is only 30% of the measured peak, under-resolution is suspected.

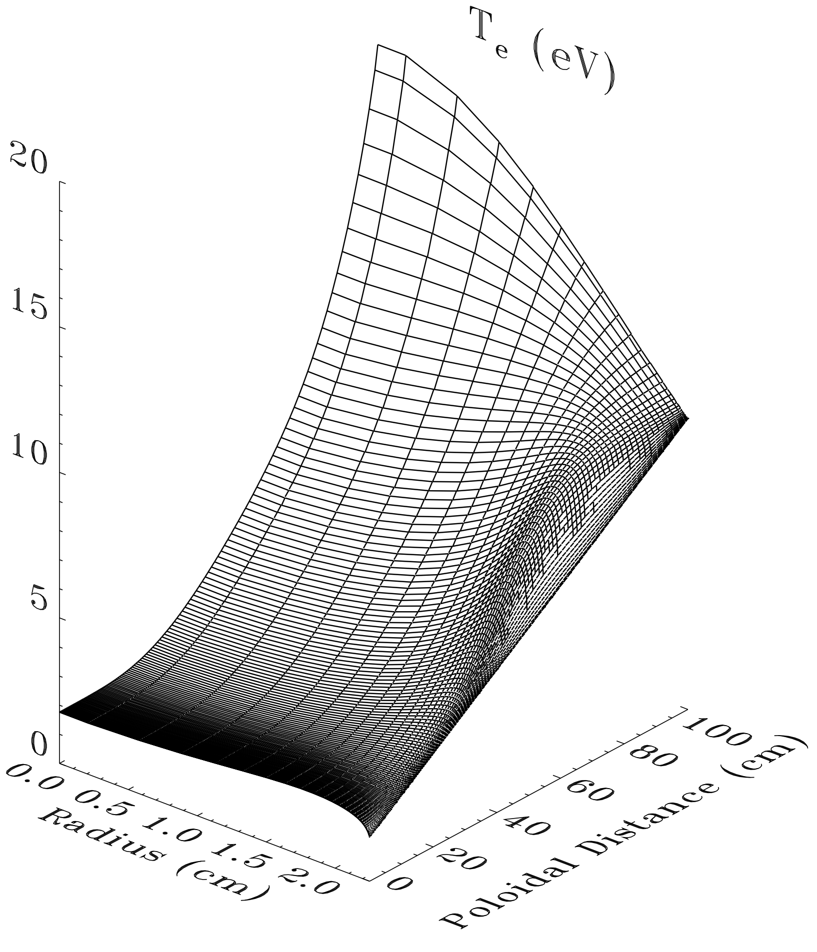

Figure 8.10. FMS2D solution for 2D PISCES model w/Dirichlet wall BC on T^e (electron temperature). The boundary layer along the wall for $x < 50$ is clear. The temperature profile does not appear to have the correct shape.

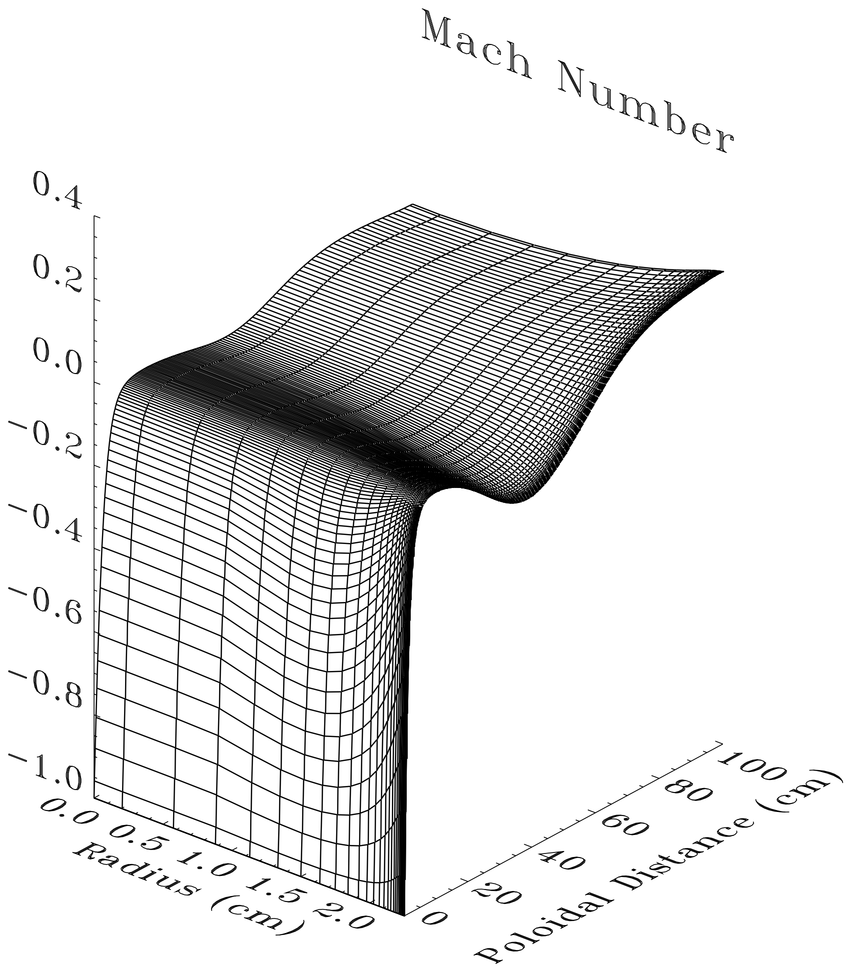

Figure 8.11. FMS2D solution for 2D PISCES model w/Dirichlet wall BC on T^e (Mach number). The Mach number shows reversed flow upstream from the target which is stronger along the cylinder wall.

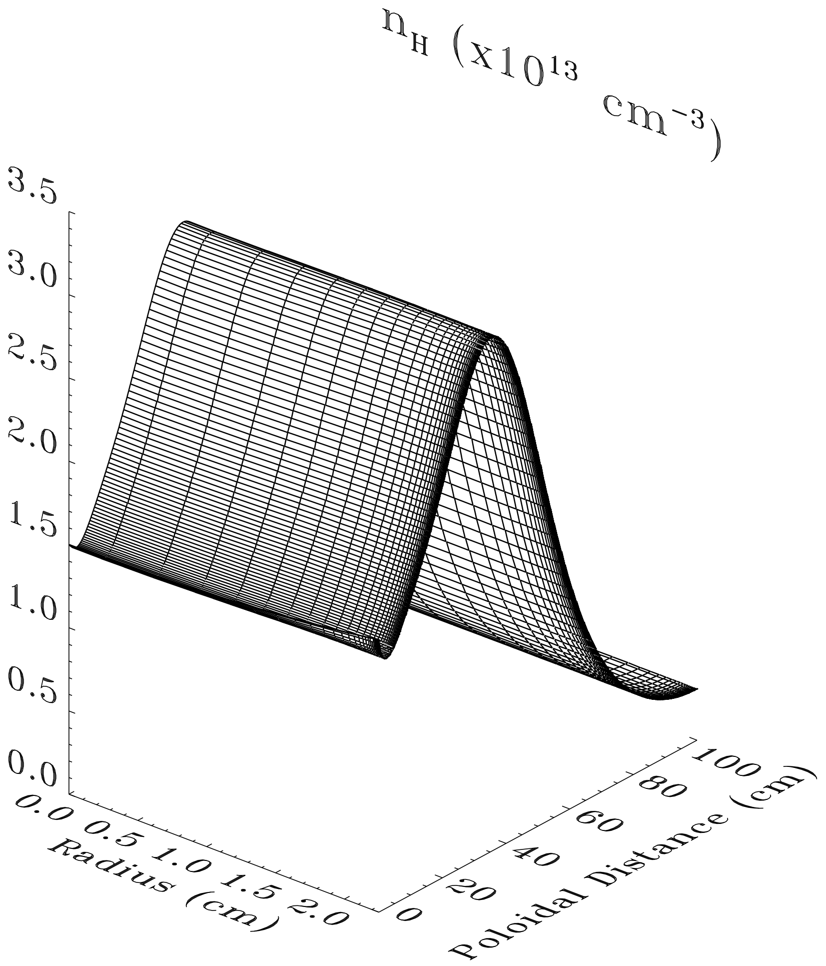

Figure 8.12. FMS2D solution for 2D PISCES model w/Dirichlet wall BC on T^e (atomic Hydrogen). There does not appear to any radial structure to this profile. Since no heat is entering the system, neutral particle mean free paths are much longer than the diameter of the machine.

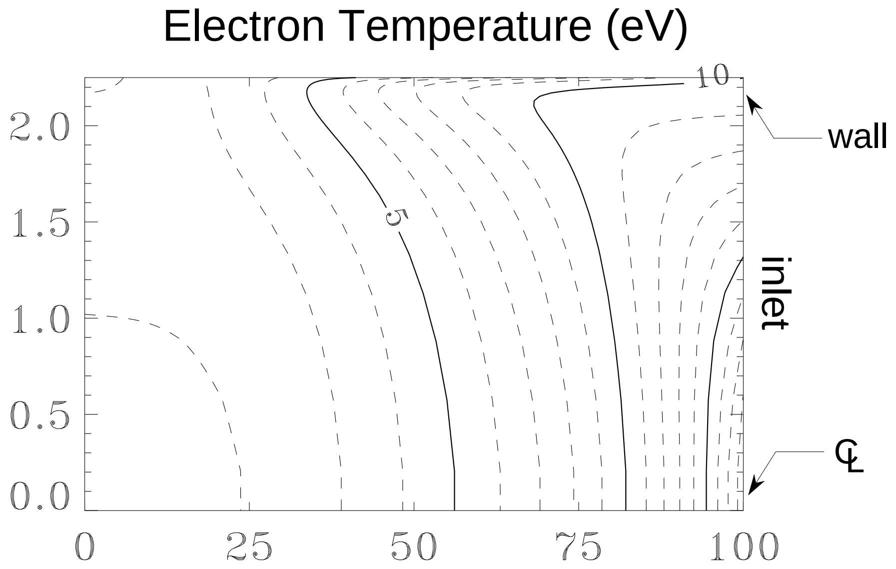

Figure 8.13. Contours of the T^e profile from the 2D PISCES model using Dirichlet wall BC on T^e. The boundary layer observed near the wall (upper boundary in this figure) may be an artifact of improper boundary conditions).

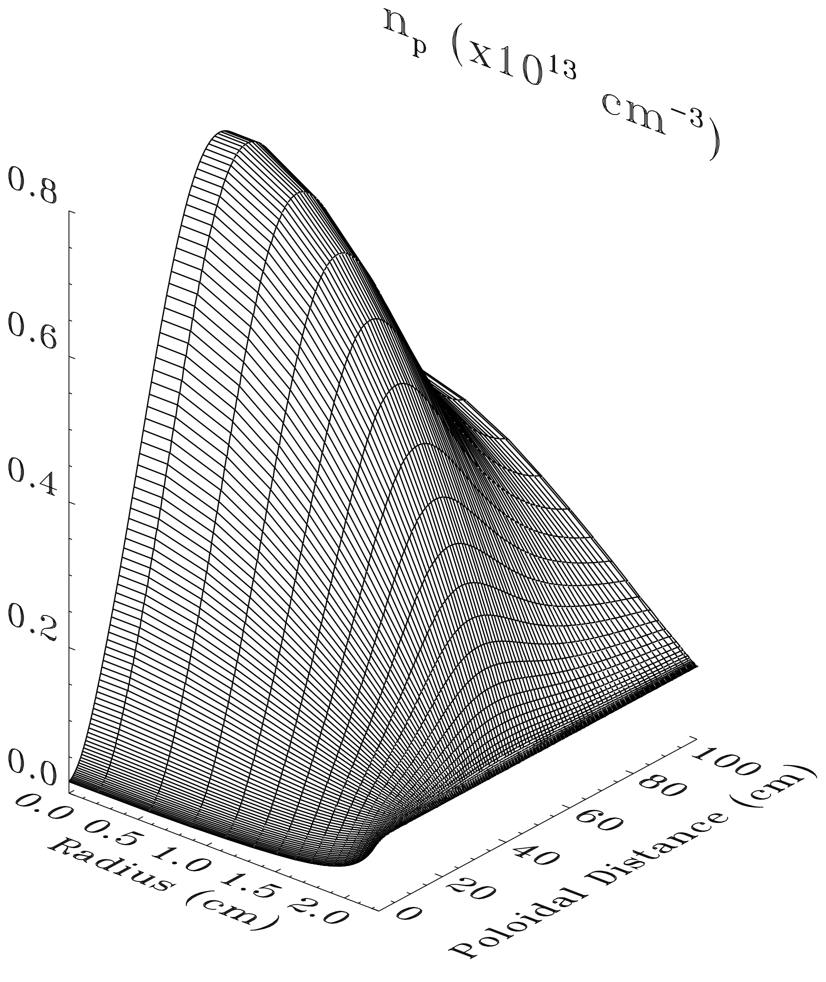

Figure 8.14. 2D PISCES model n results for *outflow* wall BC. This profile is slightly lower valued, but otherwise identical to the Dirichlet case.

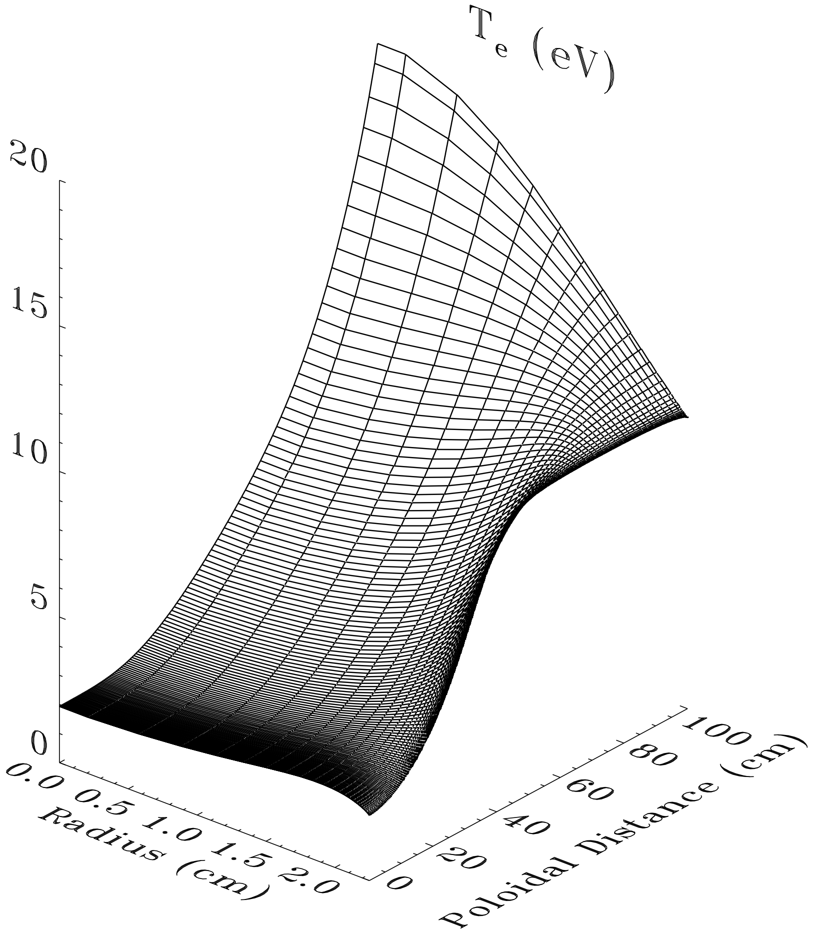

Figure 8.15. 2D PISCES model T^e results for *outflow* wall BC. The boundary layer along the cylinder was has been completely removed. Though the profile is smoother than the in the Dirichlet case, it is no better resolved.

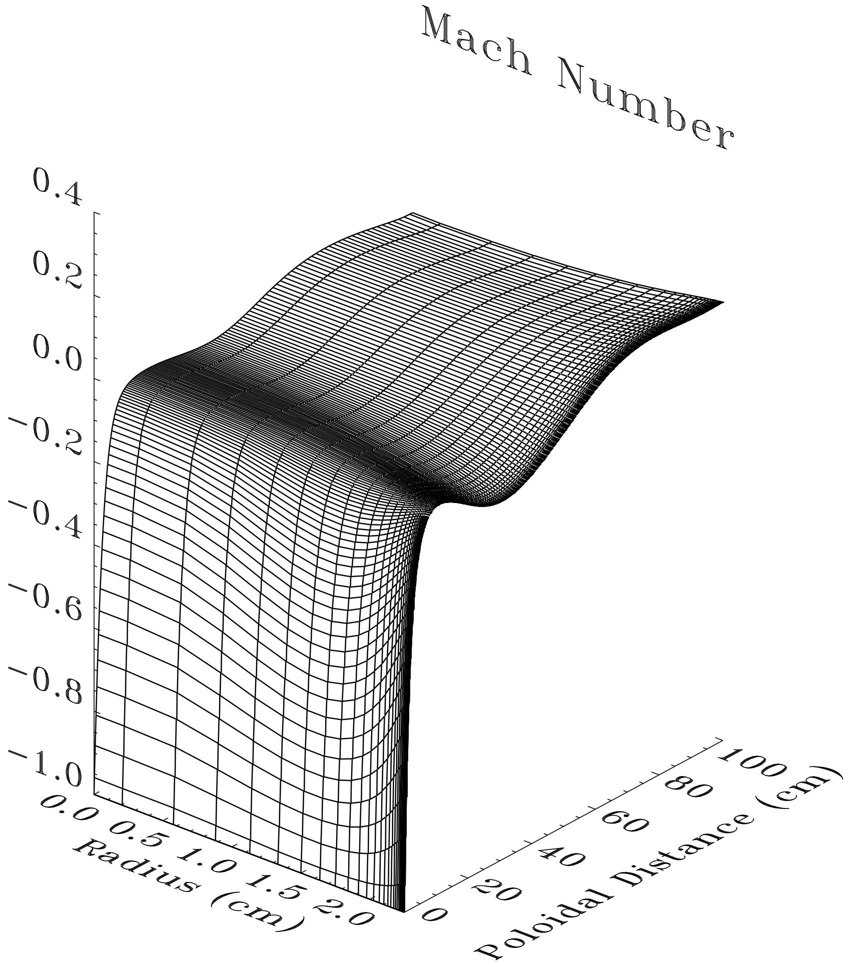

Figure 8.16. 2D PISCES model Mach number results for *outflow* wall BC. The plasma is slightly less flow-reversed in this configuration, particularly near the cylinder wall at the inlet. But again, these results are underresolved, so minute discrepancies may not be of physical interest.

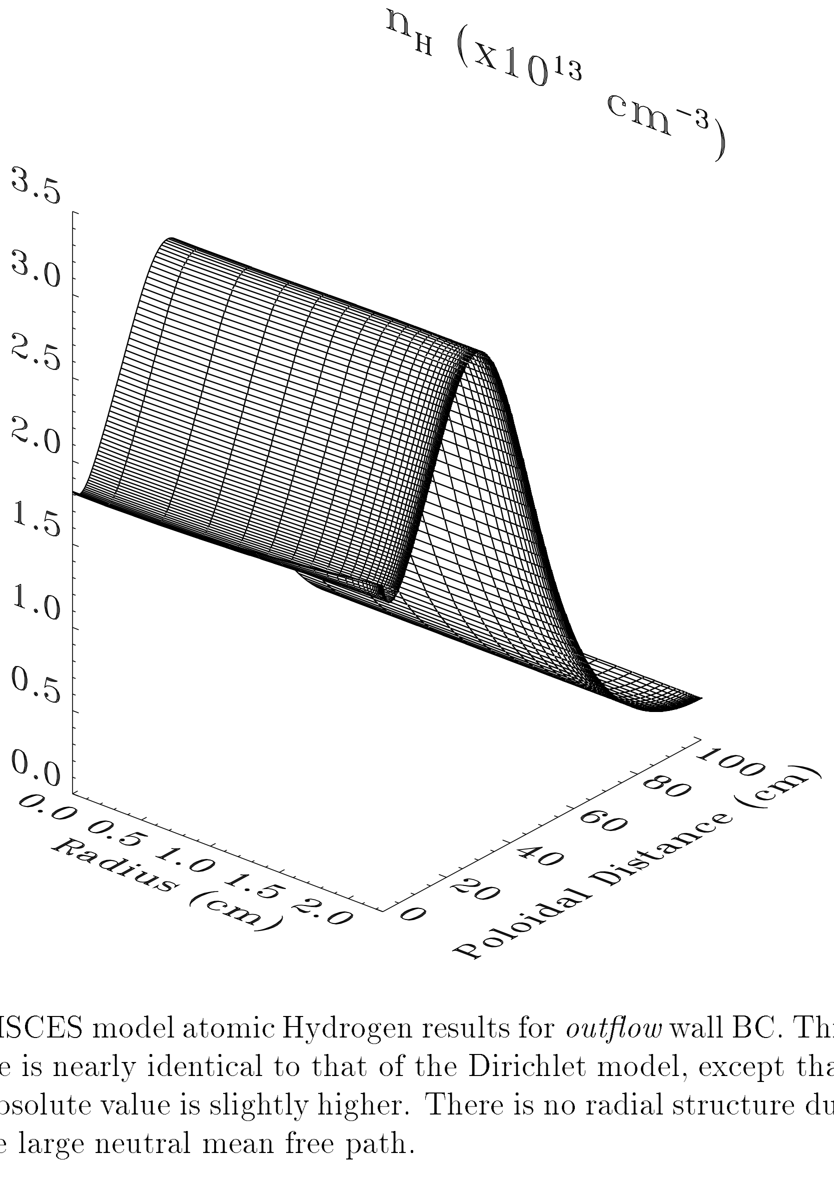

Figure 8.17. 2D PISCES model atomic Hydrogen results for *outflow* wall BC. This profile is nearly identical to that of the Dirichlet model, except that the absolute value is slightly higher. There is no radial structure due to the large neutral mean free path.

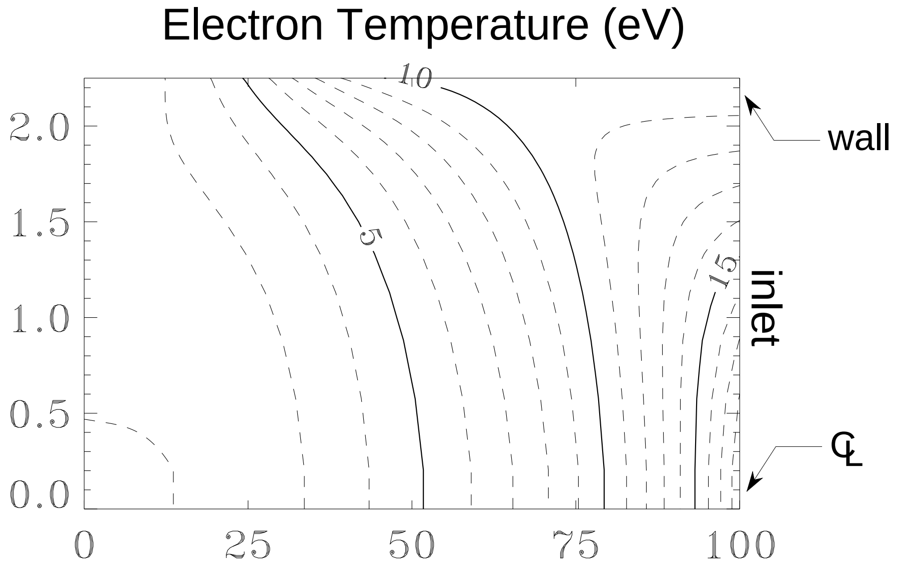

Figure 8.18. Contours of 2D PISCES results for *outflow* wall BC on T^e. The boundary layer observed near the wall (upper boundary in this figure) in the Dirichlet case is completely absent here, and a profile consistent with the *flux-limited* diffusion model is apparent

Chapter 9

Conclusions and Future Efforts

9.1 Summary of Results

In Chapter 1, a description is given of the many detailed processes that arise when considering models of the tokamak edge plasma. In Chapter 4, we first developed a general set of modeling equations. The purpose was to provide a framework for simplifying and tailoring the general theory to our specific problem. We carefully identified the role of source terms in the model, and showed how to consistently add new atomic processes into the model in an *ad hoc* fashion. The atomic source terms have significant influence on the profiles established by the edge plasma. From this general set, we then list all the assumptions used to actually obtain a specific model for our regime of interest—the gas-target divertor. Due to the care in derivation, certain anomalies noted in the literature were easily resolved—we found that most were attributable to lack of specification or communication among researchers. For example, including anomalous particle fluxes leads to a set of equations involving total energy conservation that appears to violate an equivalently valid set of equations involving just the internal energy.

We then detailed the design of the computer code, FMS2D, used to implement the various complex models described in the rest of this document. With an eye for the shortcomings of the many existing edge modeling codes, FMS2D was designed to minimize the difficulty in adjusting and sharing the physical model of a given system. An example complete model specification is given in Appendix E to demonstrate the usefulness of the design. The code employs Newton-Raphson iterations to adaptively time-step toward steady solution profiles on an arbitrary number of coupled conservation equations. The nonlinear iterations needed to be damped to avoid temporal divergence, mesh-sequencing was used to accelerate convergence properties, and iterative linear solvers were employed to aid in the Newton-Raphson solution technique. The convection discretizations employed were those of the standard edge modeling tools available today, including classical and streamline-upwind artificial dissipation. The FMS2D code was *not* developed for runtime efficiency, and is therefore quite slow in running modest-sized two-dimensional plasma problems of interest. The purpose however was to evaluate the discretizations and the model specification logic to determine if a focus on efficiency (through vectorization/parallelization, for example) would be profitable.

As significant related effort, we sought to improve on the geometric flexibility afforded by the typical edge modeling code. Presently, simple numerical discretizations are based on aligning the discrete nodes of the domain with the direction of the principal axes of anisotropic dissipation. Although such aligned schemes seem obviously superior in potential accuracy over a scheme that is arbitrarily oriented, it fails some fundamental modeling criteria. First, the grids are orthogonal to the magnetic flux surfaces, which themselves are grazing to the surfaces that they intercept. It is therefore impossible to accurately represent any physical boundary.

Second, the flux surfaces form a *multiply-connected* domain, requiring considerable logic in numerical solvers and limiting discretizations to first or second order accuracy. Third, the grids are difficult to generate and plot graphically, and therefore do not promote rigorous numerical practices. Routine accuracy estimation techniques are difficult, as is the application of any modern high-resolution numerical techniques. Unfortunately, as detailed in Chapter 6 we developed a fundamental principle stating, in short, that strongly anisotropic dissipation in fact cannot be consistently represented on anything other than an aligned grid. As a result, the optimal procedure that we developed was found to be essentially of no additional benefit, except when modeling the region near X-points in the poloidal plane of tokamak devices. But detailed modeling of the X-point region is currently beyond available computer hardware resources. We thus completed the remainder of the dissertation work using *aligned* grids, but ensuring that our discretization obeyed the required discrete minimum principle developed in Chapter 6.

Solver and simplified discretizations at hand, we proceeded to apply a one-dimensional plasma model to the PISCES-A gas-target experiment at UCLA. A key feature of the experimentally measured plasma profiles is that up to 97% of the heat flux impinging on a gas layer down the center of a cylindrical tube is dissipated by volume power dispersion as a result of plasma-neutral interactions (such as line radiation due to electron impact ionization events). In the experiment, reactions favoring the transfer of momentum from the plasma to the neutrals, is enhanced by puffing gas into the back end of the tube. In the momentum-dissipation regimes, the models are particularly difficult to resolve numerically as they contain significant internal structure. They therefore provided a good test of the FMS2D code's abilities. A model of the system included the diffusive transport of atomic and molecular Hydrogen coupled to a simplified set of the Braginskii transport equations for the plasma. The ion and neutral gases have fixed low temperatures, and Bohm sheath conditions are applied.

The one-dimensional modeling results were quite satisfactory. The profiles determined from Langmuir probe measurements were matched well by the model which had very few modifiable parameters. We were able to use the model to verify our intuition about the nature of the collapsing plasma ionization front. Successes in 1D prompted us to extend the model into two dimensions. Despite the rather successful experiences noted in Chapter 7 during benchmarking and testing stages, the FMS2D code suffered severely in 2D from inadequate numerical resolution. The convection discretization schemes hindered complete two-dimensional analysis of the PISCES-A experiment, despite using up to 256x64 grid points. Insufficient results were obtained to make significant comment on whether the extended model would have worked appropriately, but the types of code modifications required to remedy the situation are such as to move that work from the scope of this thesis. We are content with the lessons learned.

9.2 Reflections and Suggestions for Follow-on Research

At the outset of this research project, the goal of the dissertation was to develop and apply the FMS2D computer code to models of fully two-dimensional edge plasma devices for investigating gas-target divertor scenarios. Such a code was constructed and tested to support application on fully two dimensional tokamak plasmas. We opted to test the capabilities of the FMS2D code using pessimistically designed test problems and a well diagnosed experiment with simplified geometries.

We apparently over-estimated the robustness and efficiency of the methods chosen (nonlinear, linear solvers, discretizations) when they are applied to the collapsed divertor plasma scenario. We rigorously proved that strongly anisotropic dissipation can only be discretized consistently if node locations are very nearly aligned with the principal axes of diffusion. First-order artificial dissipation was found to be grossly inadequate for resolving the collapsed plasma regimes of interest for volume power dispersion divertors. At a more fundamental level, we even found that specification of the physical processes was somewhat arbitrary, and that there was significant disagreement among the researchers in the field, even on the precise equations to employ, due to the overall complexity of the models.

During the development of the FMS2D code, we included all the logic necessary for handling multiple grids, arbitrary boundaries, arbitrary equations and grids, changeble discretizations and portability across numerous Unix computing platforms. Additionally, models were developed to describe high-recycling DII-D single-null diverted plasmas, high-power impurity radiation and volume recombination dominated plasmas for the ITER divertor studies. But the fundamental inefficiency of first-order prevents FMS2D from finding converged steady results for these two-dimensional models that are of use to the community. For brevity and clarity we opted to leave them out of this document.

As a result of the aforementioned deffiencies, we conclude that present modeling techniques, as reflected in this document, are still far from production-level model eveluations required in designing tokamak divertor components. Improvements are necessary at all levels of the process, not only in the theoretical models that describe the underlying flow transport, but in the algorithms used to descretize those models. Significantly, the defficiencies we encountered, which are reported herein, are generally applicable to all of the leading edge plasma fluid modeling codes, because we have used methods which approximate or generalize the underlying algorithms used in every large-scale edge code to date. It is important to recognize that the difficulties are tightly coupled to the nature of the plasma under investigation. Simple, high-recycling plasma, such as those of the ASDEX test case, are well-handled by present-day codes, including our FMS2D. But the internal features that arise during plasma detachment and thermal collapse (as well as important time-dependent features, such as MARFEs and ELMs ignored in our analyses) are well beyond the capabilities of all present codes.

Modern, high-resolution numerical methods will be required to resolve the detailed nonlinear profiles of the collapsed plasma regimes, and provide design information to the ITER and TPX efforts. Certainly, this will involve solution adaptive gridding techniques and explicit time evolution methods. Production software will need to be developed so that a larger number of scientists can become involved in the study of applied edge physics without having to invest the efforts to develop the work from scratch.

We were, however, quite satified with the software engineering contained in the FMS2D code. And since a great deal more physics can be learned from the PISCES experiments, as well as mild conditions during DIII-D runs, a reasonable task requiring minimal effort might be to rewrite the core algorithms of FMS2D in a form that vectorizes and multitasks well on a supercomputer. We could not carry this out due to time constraints, but recommend this route as the fastest way to increase the number of competent edge modelers—and the pool of new ideas that such a situation would encourage. Barring this rather small effort, we feel that it would not be particularly productive for the fusion community in general to continue to support the extension and application of present-day fluid edge modeling codes into the volume power-redistribution regimes. The fundamental ineficiencies that result from applying first-order discretizations are not surmountable by any forseeable computer hardware advances. The community must make the commitment to support advanced numerics development, perhaps at the expense of delaying gas-target and radiative-divertor exhaust scenarios for another couple of years. Perhaps that would buy enough time that the theorists could develop a better model of electron conduction!

Appendix A

Transport Coefficients

To arrive at the actual set of conservation equations describing the plasma fluid system, we must assign applicable forms to the integrals that define the stress tensor and heat flux, as well as carry out the source term integrals. The former can be accomplished by assuming a form for the distribution functions, while the latter requires detailed information about the nature of collisional communication. The former is discussed in this note, while the latter is discussed in Appendix B

It is well known that in physical systems without any applied forces or spatial gradients in the characteristic thermodynamic quantities, the particle distribution functions tends toward the Maxwell-Boltzmann form,

$$f = n\left(\frac{m}{2\pi T}\right)^{\frac{3}{2}} \exp\left(-\left[\frac{m\left(\vec{v}-\vec{u}\right)^2}{2kT}\right]\right)$$

(k is Boltzmann's constant $\approx 1.6022\times10^{-12}\,\mathrm{erg/eV}$). When gradients do exist, the distribution function can be expanded in Laguerre polynomials about the above equilibrium state. Subsequent moments of the kinetic equation will then lead to a hierarchy of equations for the expansion coefficients[1].

---

[1] Laguerre polynomials are chosen because they are a complete orthonormal basis whose expansion coefficients possess the correct weighting to describe interpretable physical quantities, such as the heat flow. For related reasons, the Laguerre series is rapidly convergent for most physical situations (cf. [173], p.1037 for a listing of the properties of a Laguerre expansion)

Expanding to high enough order in Laguerre polynomials to include the temperature gradient forces, heat flux and viscous stress tensor and using the three moments of the kinetic equation described earlier, the following expressions can be obtained for a collisional plasma. Here, the plasma consists of electrons and a single species of ions with electric charge, $Z = 1$ (the system is further subject to a rather restrictive set of assumptions to be discussed). Relaxation and communication between the charged species occurs via electrostatic Coulomb collisions, and no explicit account is taken for the the effect of charge exchange, and particle or energy sources. The relations are summarized in the NRL Plasma Formulary. In these expressions, the unit vector, $\mathbf{b} = \vec{B}/\|\vec{B}\|$, and the perpendicular direction (denoted with subscript, $\perp$) is defined orthogonal to $\vec{B}$ and the *diamagnetic*, or $\vec{E} \times \vec{B}$, drift flow direction (resp. denoted with subscript, $\wedge$).

The body force in this system arises from the usual $\left[\vec{E} + \frac{1}{c}\vec{v} \times \vec{B}\right]$ expression. The momentum source due to Coulomb interactions between the electrons and ions is evaluated to produce effectively a body force, $\vec{R}$, due to the sum of thermal, $\vec{R}_T$, and friction, $\vec{R}_u$

$$\vec{R} = \vec{R}_{ei} = -\vec{R}_{ie} = \vec{R}_T + \vec{R}_u$$

where

$$\begin{aligned} \vec{R}_T &= -0.71 n \nabla_{\|} \left(kT^e\right) - \frac{3n}{2\omega_{ce}\tau_e} \mathbf{b} \times \nabla_{\perp} \left(kT^e\right) \\ \vec{R}_u &= ne \left(\frac{\vec{j}_{\|}}{\sigma_{\|}} + \frac{\vec{j}_{\perp}}{\sigma_{\perp}} \right); \quad \sigma_{\|} = 2.0\sigma_{\perp} = 2.0 \frac{ne^2\tau_e}{m^e} \end{aligned}$$

Electron and ion heating are, respectively

$$Q_{ei} = -Q_{ie} - \vec{R} \cdot \Delta\vec{u}; \qquad Q_{ie} = \frac{3m^e}{m^i} \frac{nk}{\tau} \left(T^e - T^i\right)$$

where $\Delta\vec{u}$ is the relative drift, $\vec{u}_e - \vec{u}_i$. The electron heat flux, $\vec{q}_e$, is due to both friction, and thermal gradients, $\vec{q}_e = \vec{q}_u^{\,e} + \vec{q}_T^{\,e}$

$$\begin{aligned}
\vec{q}_u^{\,e} &= 0.71 nkT^e \vec{u}_{\|} + \frac{3nkT^e}{2\omega_{ce}\tau_e}\mathbf{b}\times\vec{u}_{\perp} \\
\vec{q}_T^{\,e} &= -\kappa_{\|}^e \nabla_{\|}\left(kT^e\right) - \kappa_{\perp}^e \nabla_{\perp}\left(kT^e\right) - \kappa_{\wedge}^e \mathbf{b}\times\nabla_{\perp}\left(kT^e\right)
\end{aligned}$$

The ion heat flux is due only to the thermal gradient

$$\vec{q}_i = -\kappa_{\|}^i \nabla_{\|}\left(kT^i\right) - \kappa_{\perp}^i \nabla_{\perp}\left(kT^i\right) - \kappa_{\wedge}^i \mathbf{b}\times\nabla_{\perp}\left(kT^i\right)$$

The corresponding electron and ion thermal conductivities are

$$\begin{aligned}
\kappa_{\|}^e &= 3.2\frac{nkT^e\tau_e}{m^e}; \quad \kappa_{\perp}^e = 4.7\frac{nkT^e}{m^e\omega_{ce}^2\tau_e}; \quad \kappa_{\wedge}^e = \frac{5nkT^e}{2m^e\omega_{ce}}; \\
\kappa_{\|}^i &= 3.9\frac{nkT^i\tau_i}{m^i}; \quad \kappa_{\perp}^i = \frac{2nkT^i}{m^i\omega_{ci}^2\tau_i}; \quad \kappa_{\wedge}^i = \frac{5nkT^i}{2m^i\omega_{ci}};
\end{aligned}$$

The stress tensor components for either species are

$$\begin{aligned}
\Pi_{xx} &= \frac{\eta_0}{2}\left(W_{xx}+W_{yy}\right) + \frac{\eta_1}{2}\left(W_{xx}-W_{yy}\right) + \eta_3 W_{xy}; \\
\Pi_{yy} &= \frac{\eta_0}{2}\left(W_{xx}+W_{yy}\right) - \frac{\eta_1}{2}\left(W_{xx}-W_{yy}\right) - \eta_3 W_{xy}; \\
\Pi_{xy} &= \Pi_{yx} = \eta_1 W_{xy} - \frac{\eta_3}{2}\left(W_{xx}-W_{yy}\right); \\
\Pi_{xz} &= \Pi_{zx} = \eta_2 W_{xz} + \eta_4 W_{yz}; \\
\Pi_{yz} &= \Pi_{zy} = \eta_2 W_{yz} - \eta_4 W_{xz}; \\
\Pi_{zz} &= \eta_0 W_{zz}
\end{aligned}$$

where the $\hat{z}$ coordinate is along $\mathbf{b}$. The rate-of-strain tensor, $\mathbf{W}$ is

$$W_{jk} = \frac{\partial u_j}{\partial x_k} + \frac{\partial u_k}{\partial x_j} - \frac{2}{3}\delta_{jk}\nabla\cdot\vec{u}$$

the ion viscosity coefficients are

$$\begin{aligned}\eta_0^i &= 0.96nkT^i\tau_i; \quad \eta_1^i = \frac{3nkT^i}{10\omega_{ci}^2\tau_i}; \quad \eta_2^i = \frac{6nkT^i}{5\omega_{ci}^2\tau_i}; \\ \eta_3^i &= \frac{nkT^i}{2\omega_{ci}}; \quad \eta_4^i = \frac{nkT^i}{\omega_{ci}};\end{aligned}$$

and the electron viscosity coefficients are

$$\begin{aligned}\eta_0^e &= 0.73nkT^e\tau_e; \quad \eta_1^e = 0.51\frac{nkT^e}{\omega_{ce}^2\tau_e}; \quad \eta_2^e = 2.0\frac{nkT^e}{\omega_{ce}^2\tau_e}; \\ \eta_3^e &= -\frac{nkT^e}{2\omega_{ce}}; \quad \eta_4^e = -\frac{nkT^e}{\omega_{ce}};\end{aligned}$$

Finally, the collision times are

$$\tau_e = \tau_{ei} = \frac{3\sqrt{m^e}\,(kT^e)^{3/2}}{4\sqrt{2\pi}n\lambda_{ei}e^4}$$

$$\tau_i = \sqrt{2}\tau_{ii} = \frac{3\sqrt{m^i}\,(kT^i)^{3/2}}{4\sqrt{\pi}n\lambda_{ii}e^4}$$

and λ is the Coulomb logarithm for the applicable transport process

$$\lambda_{ei} = \begin{cases} 23 - \frac{1}{2}\log\left(\frac{n}{T_e^3}\right), & T^i m^e/m^i < T^e < 10\text{eV} \\ 24 - \frac{1}{2}\log\left(\frac{n}{T_e^2}\right), & T^i m^e/m^i < 10\text{eV} < T^e \\ 30 - \frac{1}{2}\log\left(\frac{n}{T_e^2(m^i/m_p)}\right), & T^e < T^i m^e/m^i \end{cases}$$

$$\lambda_{ii} = 23 - \frac{1}{2}\log\left(\frac{2n}{T_i^3}\right),$$

Notice that **q**, the unit vector in the so-called 'diamagnetic' direction, actually points in the direction perpendicular to both the magnetic field and the plane of the temperature gradient.

Appendix B

Plasma Source Terms

A background of neutral particles is assumed to exist in the plasma edge system, as atomic physics processes play a major role in the tokamak plasma edge. In the simplest models, the neutral gas is diffusive against the plasma, making statistically rare collisions with other neutral particles. The collisions produce ionization and recombination events which change the identity of the involved particles and provide sources of particles and energy to the plasma (and lead to corresponding neutral particle sinks). Additionally, charge-exchange events transfer momentum between the plasma ions and the neutral particles (see Figure B.1). The need for consistent coupling, and the method for arriving at a reasonable approximation of the neutral transport characteristics, is discussed in Reference [18]. There, the neutral gas density, n_H, is inserted into the model as an additional unknown state variable, and the diffusion approximation is applied[59] to describe the particle flux vector as $\vec{\Gamma} = -D_0 \nabla n_H$. The neutral diffusion coefficient, D_0, is computed from an OGITE-like (OGITE = One Group Ion Thermal Equilibrium) approximation[18], so that

$$D_0 = \frac{\lambda_{tr}}{3} u_0; \quad u_0 = \sqrt{\frac{8kT_0}{\pi M}}$$

where the neutral gas temperature, T_0, and λ_{tr} is the transport mean free path of the neutral against collisions with the background plasma (the form arises from the P1 approximation to the scattering integral in the neutral particle kinetic equation; cf. Reference [174]). In the OGITE approximation, $T_0 = T^i$, but more generally, we may specify this as part of the system we are modeling.

The particle source S_n^j due a particular atomic event, j, may be expressed conveniently in terms of Maxwellian-averaged collision rate parameters $\langle\sigma v\rangle^j$, which are tabulated functions available for most relevant atomic process (cf. [151], and more recent additions, such as [150]). The source rate density (occurrences per unit time per unit volume) for a process involving collisions between species a and b is defined as $\left(n_a n_b \langle\sigma v\rangle_{ab}^j\right) / \left(1 + \delta(ab)\right)$, where n_α is the particle density of species α, and $\delta(ab) = 1$ if the particles a and b are of the same species[1]. Thus, the energy transfer rate (corresponding to the S_E term in Equations 4.5 and 4.10) is the sum of terms, $\mathcal{E}^j \left(n_a n_b \langle\sigma v\rangle_{ab}^j\right) / \left(1 + \delta(ab)\right)$, where $\mathcal{E}^j$ includes all energy transferred in the reaction, including exothermic energy, ϵ^j, and internal and kinetic energy of the reactants, $\mathcal{I}^j$. A similar expression arises for the momentum source, $\vec{S}_p$, in terms of the total momentum exchanged per event, $\mathcal{P}^j$. The net sources are the linear sum over all the N_{react} relevant processes

$$S_n = \sum_j^{N_{\text{react}}} S_n^j; \quad \vec{S}_p = \sum_j^{N_{\text{react}}} S_n^j \mathcal{P}^j; \quad S_E = \sum_j^{N_{\text{react}}} S_n^j \mathcal{E}^j$$

Ionization

Hydrogen ionization is described via reactions 2.1.5, 2.1.6 and 2.1.7 in Reference [151]

$$e + H(1s, 2s, ns) + \epsilon^{\text{eii}} \rightarrow e + H^+ + e$$

where "$\#s$" refers to the excitation level of the neutral atom, and ϵ^{eii} is the endothermic energy of the reaction. A modern set of rate coefficients, that combines the three reactions based on experimental data, may be obtained (cf. Reference [150]). These processes create an electron/ion pair and remove one neutral atom from the system. No momentum is transferred. The reaction consumes energy from the electron channel, both due to needing to impart kinetic energy above the electron binding energy, plus an additional amount, due to unsuccessful attempts that result in inelastic collisions (see Figure B.1). Both $\langle\sigma v\rangle^{\text{eii}}$, and ϵ^{eii} are dependent on electron temperature and plasma density, and are plotted in Figures B.2 and B.3, respectively.

In the absence of a model for molecular Hydrogen, we construct a simple physical picture to obtain the exothermic energy added to the ion channel per ionization event. All ions are assumed to flow toward a material surface, where they recombine with electrons, and are thermally desorbed from the surface as molecular Hydrogen. Some time later, the molecular Hydrogen dissociates into atomic Hydrogen via the Franck-Condon process, transferring the 3-5 eV/particle molecular bond energy to the resultant particles. Since the atomic Hydrogen has no fluid flow velocity, by the diffusion assumption, and is in thermodynamic equilibrium with the ions, a reasonable assumption is that this 3-5 eV dissociation energy appears ultimately in the plasma as incoming thermal energy associated with an ionization event. The neutral splits into an ion and an electron having nominally equal speeds, however the ion carries most of the energy due to the large mass ratio, i.e. $S_E^{\text{eii}i} = \left(\epsilon^{\text{eii}} + \mathcal{I}^{\text{eii}}\right) \cdot S_n^{\text{eii}} \approx E_0 \cdot S_n^{\text{eii}}$, where E_0 is the total energy of the neutrals that are ionized (5 eV from the discussion above). Combining all the associated exchanges,

$$\begin{aligned} S_n^{\text{eii}} &= n n_0 \langle \sigma v \rangle^{\text{eii}}; \quad \vec{S}_p^{\text{eii}} = m^i \vec{u}_0 S_n^{\text{eii}} \\ S_E^{\text{eii}e} &= -\epsilon^{\text{eii}} \cdot S_n^{\text{eii}}; \quad S_E^{\text{eii}i} = 5\text{eV} S_n^{\text{eii}} \end{aligned} \tag{B.1}$$

The neutral particle source magnitude is just equal and opposite in sign to the plasma creation rate. Notice that this model implies a deposition of energy at the target plate due to surface recombination of the exiting plasma particles. This component, and that due to the radiation of the associated inelastic collisions, is often neglected when quoting resultant target power fluxes, but is significant, and often dominant in high recycling divertors with large particle fluxes.

Recombination

Recombination occurs at material surfaces and within the volume of the plasma. The portion occurring at the material surfaces is treated as outlined above, while the volumetric recycling is treated as follows. The volumetric recombination rate is $n_e n_i \langle \sigma v \rangle^{\text{rec}} = n^2 \langle \sigma v \rangle^{\text{eii}}$. The rate coefficient includes the normal reaction, 2.1.8 in Reference [151],

$$e + H^+ \rightarrow H(n) + h\nu$$

plus three body-enhanced reactions. The radiation, $h\nu$ emitted equals the binding energy of the electron. The total rate coefficient is also available via Reference [150] as a function of electron temperature and plasma density. This data is plotted in Figure B.4 Note that appreciable recombination rates are realized only for $T^e <$ .5eV or so. However, if a divertor can be operated in this regime, the volumetric recombination reduces the target surface heat load significantly by disposing of the ion/electron pairs before they can convect to the Bohm sheath to recombine on the surface. When recombination dominates, the plasma is said to be "detached" from the target, and the plasma temperatures are said to be "collapsed", forming an ideal set of conditions for tokamak divertor operation (heat load dispersed throughout divertor chamber via radiation).

The recombination process is described as a loss of one ion/electron pair to the neutrals. If the neutrals carry no net momentum, none is lost by the plasma in this process (i.e. there must be some conservative process to slow the new neutral particle down; we include only its net effects). Also, there is no associated loss of plasma energy. Recombination therefore is described by the following set of source terms

$$\begin{aligned} S_n^{\text{rec}} &= -nn\langle\sigma v\rangle^{\text{rec}}; \qquad \vec{S}_p^{\text{rec}} = m^i \vec{u} S_n^{\text{rec}} \\ S_E^{\text{rece}} &= \frac{3}{2} k T^e S_n^{\text{rec}}; \qquad S_E^{\text{reci}} = \left(\frac{1}{2} m^i \vec{u}^2 + \frac{3}{2} k T^i\right) S_n^{\text{rec}} \end{aligned} \tag{B.2}$$

The neutral particle sink magnitude is just equal and opposite in sign to the plasma loss rate rate.

Charge-Exchange

In a charge-exchange event, a bound electron on a neutral atom jumps to a neighboring singly charged plasma ion. In the simple case, both the ion and neutral particles are Hydrogen, and the dominant reaction is

$$H^+ + H(1s) \rightarrow H(1s) + H^+$$

reaction 3.1.8 in Reference [151] (several other reactions occur, but involve various excited levels of neutrals, and are statistically less probable). The net result of this reaction is that the fast moving ions impart momentum to otherwise motionless neutrals, without affecting the overall number density or energy content of either species. Until recently, source functions for the conservation equations due to charge exchange were developed using arguments analogous to those for ionization and recombination. That is, the source functions corresponding to charge exchange are

$$\begin{aligned} S_n^{\mathrm{cx}} &= 0; \quad \vec{S}_p^{\mathrm{cx}} = m^i\left(\vec{u}_0 - \vec{u}\right) n n_0 \langle \sigma v \rangle^{\mathrm{cx}} \\ S_E^{\mathrm{cx}e} &= 0; \quad S_E^{\mathrm{cx}i} = 0 \end{aligned} \tag{B.3}$$

However, recent analytical work[80] suggests that when the neutrals are in thermal equilibrium with the ions, the charge exchange can be seen to create a situation where ions spend a fraction of their existence as neutral particles, unaffected by the magnetic field and Coulomb interactions. The Braginskii transport coefficients for the plasma are apparently therefore enhanced by the factor, $\left(n_0 + n\right)/n_0$ due to this "identity crisis". The charge exchange rate density is given by $n n_0 \langle \sigma v \rangle^{\mathrm{cx}} \approx 2.93\sigma_x n_0 (T^i/m^i)$, so that the neutrals feel an effective body force in a temperature gradient, and are thus heated. The modified source terms are

$$\begin{aligned} \vec{S}_p^{\mathrm{cx}new} &= -M\left(\vec{u}_0 - \vec{u}\right) n n_0 \langle \sigma v \rangle^{\mathrm{cx}} + 0.24 n \nabla T \\ S_E^{\mathrm{cx}new} &= 1.47 n n_0 \langle \sigma v \rangle^{\mathrm{cx}} \left(T^i - T_0\right) \end{aligned}$$

However, the newer model depends on the neutral particles satisfying a momentum balance of their own, leading to another set of equations (the compressible gas dynamics equations). Such models are currently under development[175].

For simplicity, we use the older model, Equations B.4, where $\langle \sigma v \rangle^{\mathrm{cx}} \approx 2 \cdot 10^{-8}$ (in Reference [151], we see that the temperature and plasma density dependence for this rate parameter are negligible).

Additional Processes

More source terms will arise if we wish to include the transport of impurities and additional neutral species. The following lists some common extensions:

- *Trace amount of hi-Z impurities exist in the plasma.* If an atom contains many bound electrons, there are a large number of excited states which can potentially radiate in quantum electronic transitions. Thus, hi-Z impurities radiate energy much more efficiently than Hydrogen, and may be useful to enhance the power dispersion in the divertor. Initial studies have been performed to quantify this behavior, by inserting a small amount of impurities in the system via a fixed fraction of the ion density[90] (on the order of 1-5% or so). The effects on the overall plasma transport are assumed negligible, except for the enhanced energy radiation, which enters only via the energy source, S_E. More complete models of impurity radiation exist[176, 177, 76, 178, 179, 83], but involve adding additional unknowns to the system to track impurity particle densities and/or momentum.

- *Molecular Hydrogen is modeled.* Since the recombined Hydrogen is desorbed from the target surface (and puffed into some experiments) as diatomic molecular Hydrogen, it would seem more appropriate to track H_2 diffusion through the plasma, in addition to that of the atomic Hydrogen. As a result, the atomic Hydrogen source is due to volumetric recombination *and* the Franck-Condon dissociation process from a diffusive molecular Hydrogen fluid, a more dispersed process. This adds H_2 gas to the model, which is often modeled by diffusion equations, and assumed to exist at the desorption energy, ~ 0.1eV. Also, there are a large number of intermediate reactions involving the dissociation and ionization of the H_2 molecule. This process was included in the PISCES model presented in this work, but greatly complicated the source term computation.

- *There are "Charge-Exchange" and "Franck-Condon" groups for neutrals.* The UEDGE code[87] employs a model wherein two species of neutral gas exist, one having energies associated with the Franck-Condon process ($T_g \sim 3$eV) and a second in thermal equilibrium with the plasma ions via charge exchange ($T_{cx} \sim T^i$). The source terms will then reflect these assumptions, and the above plasma source terms suitably modified.

- *Fluid neutrals are included.* Owing to the above discussions regarding charge exchange transfer of momentum to the neutrals, the source terms due to ionization and recombination should be altered to reflect the nonzero convective velocity of the atomic Hydrogen species. Additionally then, the momentum (and possibly, energy) conservation equation should be added to the system, increasing the number of unknowns in the system.

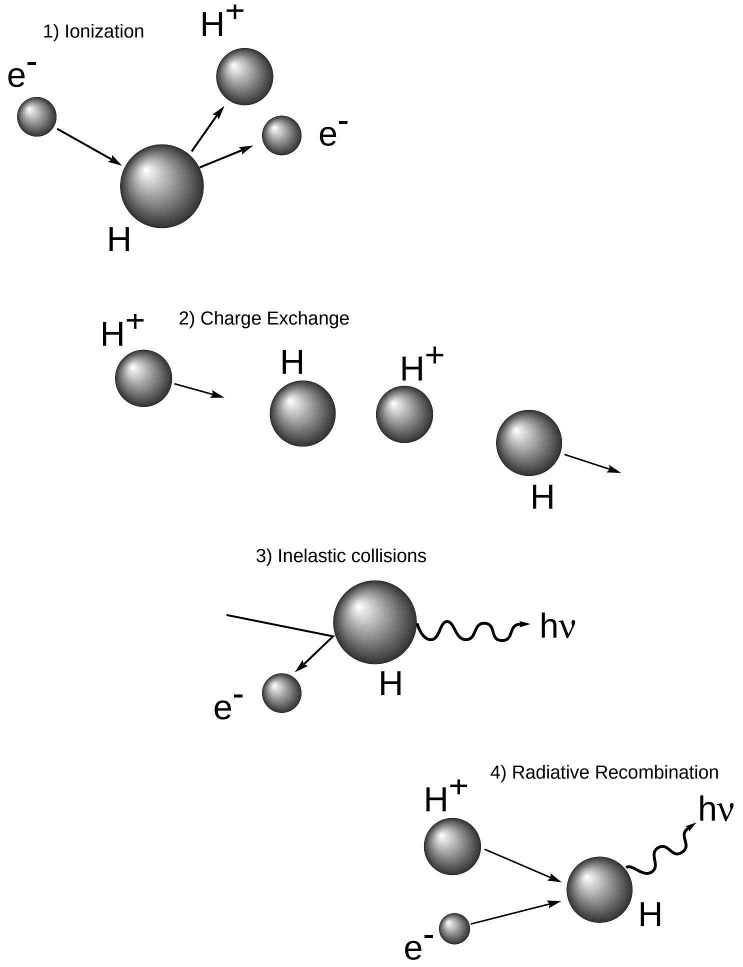

Figure B.1. Example Hydrogenic collision processes occurring in the tokamak edge plasma as a result of collisions between the plasma and neutral fluids. Ionization creates and ion/electron pair from a neutral, charge exchange transfers and electron from a neutral to an ion, effectively resulting in momentum transfer. Inelastic collisions excite neutral atoms and remove energy from the electrons. The excitation energy is subsequently radiated. Radiative recombination removes an ion/electron pair and forms a neutral. The binding energy is radiated (cf. [151] for descriptions)

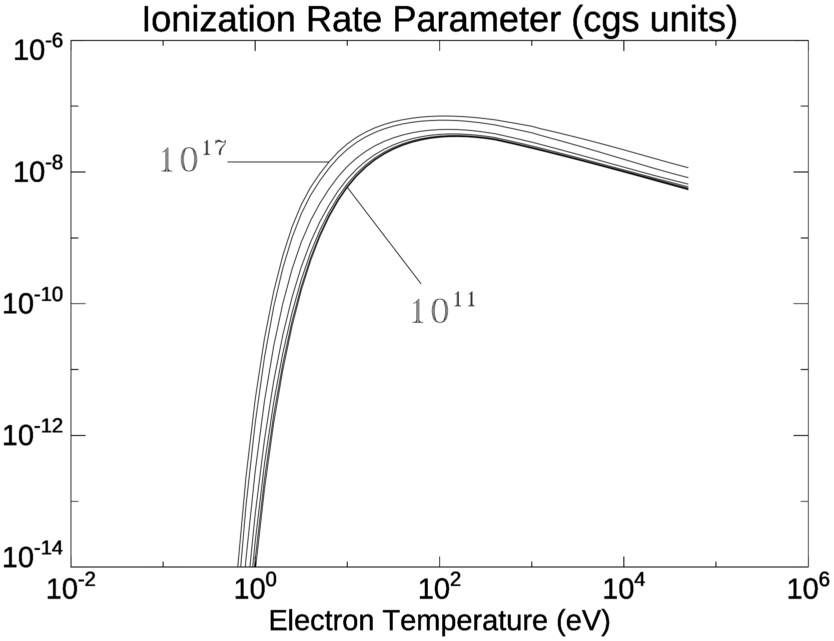

Figure B.2. Electron impact ionization rate parameter, $\langle\sigma v\rangle^{\rm eii}$. A curve is drawn for values of electron density, $n_e = 10^m cm^{-3}$, m= (11-17). There is a significant density and temperature dependence of this rate parameter for low temperatures.

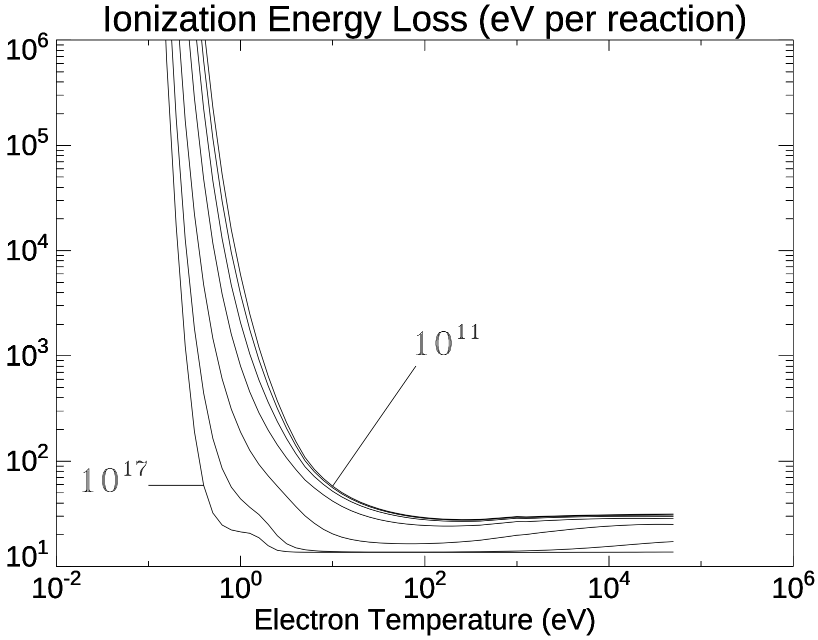

Figure B.3. Average energy lost, ϵ^{eii} from the electron channel for each ionization event. The function rises sharply at low electron temperatures. Below 13.6 eV, an electron cannot impart enough energy to dislodge the bound electron; however in the Maxwellian distribution, there may be a small fraction of particles that are energetic enough.

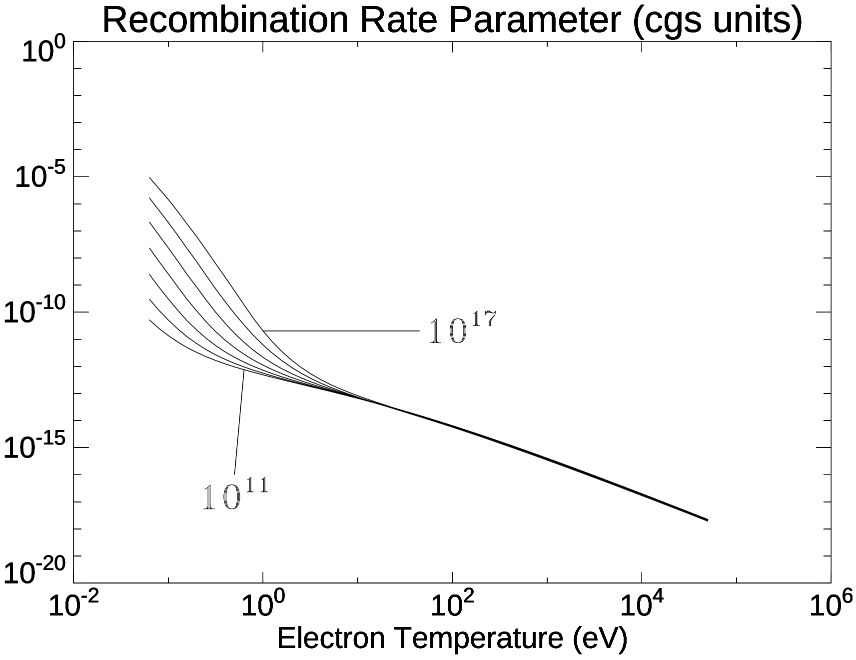

Figure B.4. Recombination rate parameter. Note that appreciable recombination rates are realized only for $T^e < .5$eV or so. If a divertor can be operated at low temperature, the volumetric recombination disposes of the ion/electron pairs before they can convect to the Bohm sheath to recombine on the surface.

Appendix C

Discretized Spitzer Conductivity

In the development of diffusion discretization schemes, the dissipation is usually assumed not to vary a great deal over the domain of interest. The Spitzer[15] conductivity we encounter in fluid plasmas does not meet this condition, particularly in the case of "collapsed" plasmas envisioned for divertor tokamaks. In this Appendix, we show that standard difference techniques may give unexpected results when plasma temperatures are very low and the transport is dominated by Spitzer diffusion. We offer no solution to the problem, but suggest that it may be responsible for the difficulties we observe in computing steady collapsed plasmas when we do so by starting from similar high recycling solutions.

C.1 Analytic Solution

The classical Spitzer heat flux, q^e_{SPITZER}, is given by the expression

$$q^e_{\mathrm{SPITZER}} = -\kappa^e_{\|} \nabla_{\|} \left(kT_e\right) \tag{C.1}$$

where $\kappa^e_{\|} = CT^{5/2}$[15], and C is a constant with respect to T_e (C is very weakly dependent on plasma density via the Coulomb logarithm). We ignore all other processes, and seek a steady solution of $\nabla_{\|}\, q^e_{\mathrm{SPITZER}} = 0$ over a unit length domain

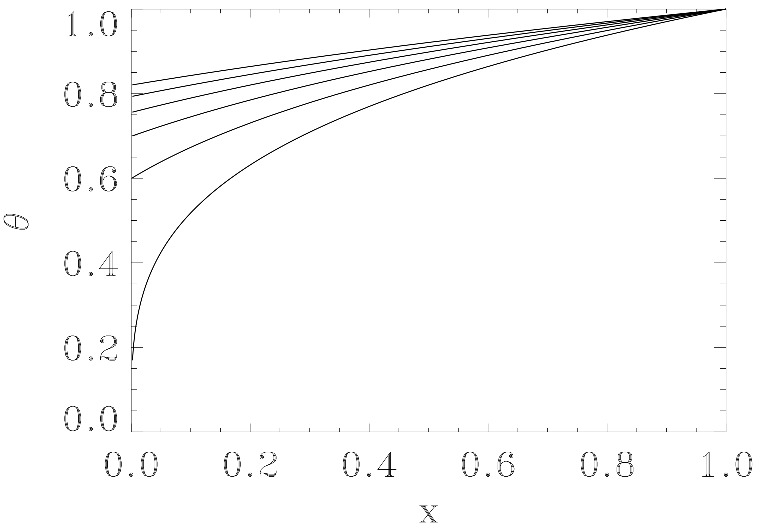

Figure C.1. Analytic solutions to the non-dimensionalized Spitzer problem. For decreasing target temperatures (at the left boundary), the boundary gradient increases. This leads to resolution difficulties.

in x. Inserting the expressions, integrating twice, and rearranging, we find

$$T = C\left(x + C_1\right)^{\frac{2}{7}} \tag{C.2}$$

where we have normalized the domain length, non-dimensionalized the temperature, and absorbed any constants of integration into C and C_1. The constants, C and C_1, are determined by the boundary conditions at $x = 0, 1$. In Figure C.1, we plot this solution for upstream $T(1) = 1$, and various “target” values. It is clear that as the target temperatures drops, a boundary layer is formed that requires additional resolution. This low-temperature situation occurs at the onset of thermal collapse when modeling a “detached plasma”.

It is interesting to see how this situation affects numerical approximations to Spitzer conduction. We set up FMS2D to solve the one-dimensional thermal conduction problem using the grid and boundary conditions shown in Figure C.2. For points internal to the domain, FMS2D stores state values at the cell centers.

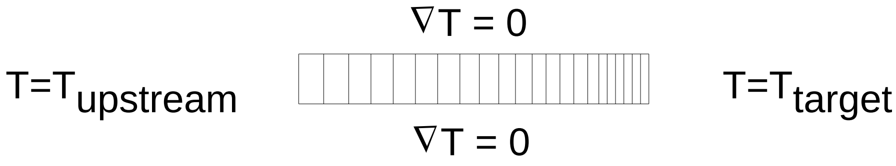

Figure C.2. One dimensional grid used by FMS2D for Spitzer conduction study. The upstream and target boundaries enforce Dirichlet conditions, while the side walls enforce Neumann conditions.

Unknowns also lie at positions along the boundary at the center of cell walls, and at the domain corners (a scalar equation solved on a 16-cell grid → $18 \times 3 = 54$ unknowns in the system). The solution is computed on exponentially stretched grids having $16, 32, 64, 128$ and 256 cells along the magnetic field (in x). We employ the usual 3-point central scheme for irregular grids (cf. [82], p. 197), where the diffusive fluxes are evaluated at the cell walls. The upstream temperature is set arbitrarily to $T_e = 100$eV, and the target temperature is set to $T_e = 6$eV (a "low" target value in the regime expected which was chosen *a priori* to illustrate the poor numerical behavior).

The results are shown in Figure C.3, where the profiles are all overlaid on the same plot. Vertices on the surfaces indicate the locations of the unknowns. The discretization is converging to the correct solution, but note how the shape of the profile varies with resolution. The coarse grid solution admits a prominent "kink" in the computed solution. The remainder of the profile is flat essentially. With increased grid-size, the kink is reduced, and the profile begins to converge formally the true solution—but not until well beyond the 256-cell grid shown (at the bottom of the pile in the figure).

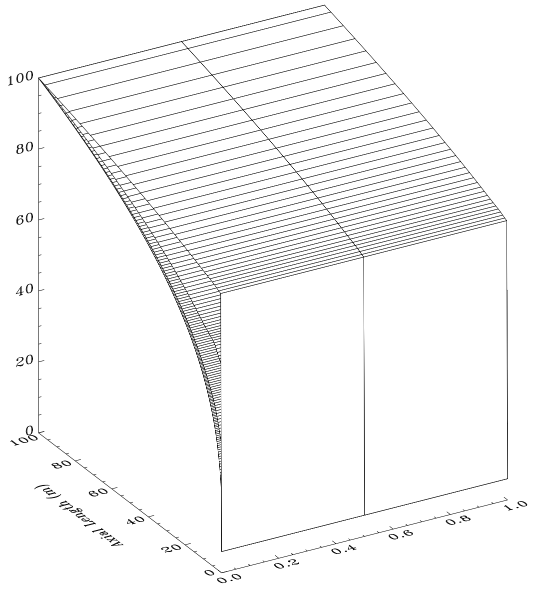

Figure C.3. Numerical solution to the Spitzer problem. The bottom surface was computed on a 256-cell grid. Due to the strong temperature dependence in the thermal conductivity, underresolved calculations are very far from the true solution. Prominent in the figure is the large drop in temperature over the last cell of the coarse grid.

C.2 Discussion and Implications

These results have been observed during routine plasma modeling calculations when the electron temperature profile is dominated at the target by conduction, particularly at the onset of thermal collapse (see Chapter [?] for discussion and example calculations of detached, thermally collapsed plasmas). Such behavior has consistently inhibited our solution of difficult tightly-coupled plasmas, which we compute normally by using various "continuation" methods that rely on smooth refinement characteristics. We would, for example, wish to solve a marginally detached state, yet begin with a similar high-recycling solution computed previously. After a few step of evolution, we slip into one of these modes, where the boundary temperature drops sharply, yet just inside the temperature rises abruptly to non-physical values; the Newton algorithms employed in FMS2D cannot easily progress through such transitions, yet carrying out the complete relaxation on finer grids is impractical.

We have seen that this behavior is the direct result of the rapid variations of the Spitzer conductivity with T_e. Since the target temperature is low, the conductivity is small, and continuity of thermal flux requires a correspondingly large $\nabla_{||} T_e$. Flux balance (heat conservation), at the last cell in front of the target, requires that incoming fluxes be equal those exiting the cell at the boundary. Since the conductivity changes rapidly over the last cell, the required gradient on the *upstream* side of the cell is small, and we end up with the observed profile kinks unless the grid is small enough that the diffusivity is nearly constant throughout the cell.

Because this behavior is connected with the cell size and state dependence of the transport coefficients, it will be observed for any numerical algorithm not specifically tuned for the Spitzer case (this includes schemes such as "harmonic averaging" [71] designed for computing transport through heterogeneous materials). However, if we tune the discrete method for the Spitzer case, than we will certainly not do well for handling, say, temperature dependence in the neutral diffusion equations. For the present, we opted for the rather extreme resolution requirements in exchange for the flexibility of a discretization that is independent of the detailed physical processes.

Appendix D

Convergence Order Verification

A convergence analysis was carried out on FMS2D to ensure that the various discretizations available in the code performed as expected[1]. In the results, we verified that the classical artificial dissipation converged as the grid spacing, h, to the first power (i.e. *first-order* convergence). A second discretization, based loosely on the Streamline-Upwind Petrov-Galerkin (SUPG) approach[101] performed better—in some cases achieving roughly second order accuracy.

We describe the convergence test that was applied, and give the results obtained for the two discretizations. And, since the SUPG formulation seems to produce better results, we give the details of our simple extension of the scheme that was derived originally for finite-element discretizations of incompressible flows.

D.1 1D Test Equation

Since the plasma equations admit only a single direction of convective particle transport, and we intend to discretize the physical domain along this direction, a one-dimensional convergence analysis will be sufficient to verify the convection

---

[1] The tests have the added benefit of helping to verify that the code is free of silly programming errors by allowing direct comparison of FMS2D results to known solutions

discretization. A generalized scalar equation is constructed to contain convection, diffusion and source terms so that all parts of the discretization are included in the test. Methods for discretizing the diffusion terms were studied in detail in Chapter [?] A "solution" function, s, is set arbitrarily to the exponential expression, $s = e^{2-x}$, $x \in \{0,1\}$. The Taylor series expansion of this solution will have well-behaved higher derivative contributions making it suitable for convergence analysis. The general equation is constructed for s to be an exact solution, therefore

$$\frac{\partial}{\partial x}\left(Ps - \frac{\partial s}{\partial x}\right) = -\left(P+1\right)e^{2-x} \tag{D.1}$$

where

$$P = uh/2\kappa$$

The non-dimensional Peclet number, P, characterizes the relative strength of convection over diffusion. Here, u represents the convection speed along the coordinate of interest, and κ is the corresponding physical diffusivity.

Solutions were computed on a set of grids constructed of 10,20,40,80 and 160 uniform cells dividing the x coordinate over the range, $[0,1]$. The *error* in numerical solution s_i is $\epsilon_i = \left(s - s_i\right)/h^n$, where h is the grid spacing and n is the *rate of convergence* of the scheme. Given a solution, and its approximation on two grid-levels, s_{i-1} and s_i, where $h_i = \frac{1}{2}h_{i-1}$, the rate of convergence is computed using the ratio

$$R = \frac{s - s_{i-1}}{s - s_i} \equiv \frac{\epsilon h^n}{\epsilon\left(\frac{h}{2}\right)} = 2^n$$

so that

$$n = \frac{\log R}{\log 2} \tag{D.2}$$

The grid is constructed so that the vertices of the mesh at level $i-1$ coincide exactly with alternate vertices of the next finer grid, i (see Figure D.1). The FMS2D discretizations are cell-centered, so the the values are interpolated to the vertices using second-order bilinear interpolation (valid because our schemes will never produce accuracy order greater than two). The error is computed at every vertex on the $i-1$ level, and the overall rate of convergence is taken as the minimum of n computed over the domain at that level (the Dirichlet problem is solved exactly on the boundaries; the boundary points do not have definable convergence rates).

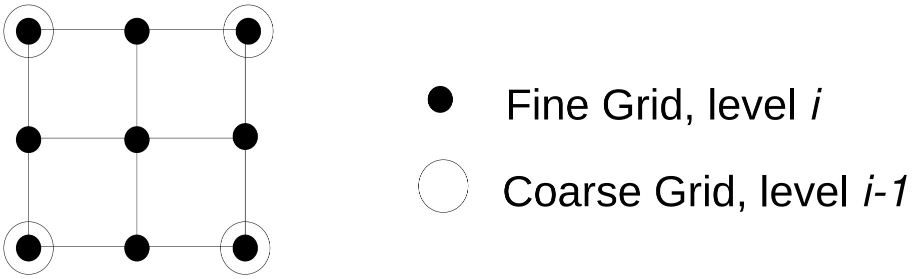

Figure D.1. Nested grids used for convergence study. The FMS2D discretization is cell-centered, so vertex values used in convergence study are bilinearly interpolated.

We present results of the FMS2D default, classical artificial diffusion, which was detailed in Section[?], and also a modified streamline upwind artificial dissipation, which was adapted from Reference [101] for finite differences. Since the solutions over the domain are not particularly exciting, they are not plotted. The order of accuracy was computed via Equation D.2 for four refinement stages (10:20, 20:40, 40:80, 80:160) using three Peclet numbers ($P = 1, 10, 10^6$). P=0 represents pure diffusion transport, and since central differencing is trivially second order accurate in both cases we omit this result. Also note that the dissipation schemes are symmetric in x, so we needed only to look at positive values of P to cover the entire range. The results are listed below

Order of convergence						
	SUPG			Classical		
Stage	$P = 1$	$P = 50$	$P = 10^6$	$P = 1$	$P = 50$	$P = 10^6$
10:20	1.879	1.300	0.897	0.935	0.904	0.897
20:40	1.924	1.655	0.953	0.994	0.947	0.952
40:80	1.967	1.884	0.977	0.999	0.973	0.977
80:160	1.945	1.967	0.989	0.999	0.986	0.989

The SUPG scheme is considerably more accurate than the classical scheme, except when convection is very strong ($P \to \infty$). Also, the improved accuracy is significant only for high resolution cases. The solution here is smoother than most interesting results encountered in practice, yet it takes nearly our limit of resolution to achieve second order accuracy in the moderate convection case that is representative of most of our situations (remember that we are concerned ultimately with two-dimensional problems, where 160x160 problems with multiple states solved fully-implicitly, would require considerable computational effort and storage).

D.2 Finite Difference SUPG

A linear convection-diffusion problem (with no sources) having fixed boundary conditions at $x = 0, L$, has the exact solution

$$s(x) = \frac{e^{Px/L} - 1}{e^P - 1} \left(s(L) - s(0) \right)$$

The Streamline-Upwind Petrov-Galerkin method is based on reproducing this particular situation exactly at the discrete nodes. It is possible to construct such a scheme if we assign an explicit form to the calculation of spatial derivatives, particularly at cell walls. From Reference [101], we infer that this is accomplished in our case by adding the artificial dissipation flux

$$\vec{\Gamma}_{\mathrm{ART}} = -D_{\mathrm{SUPG}} \nabla s$$

where D_{SUPG} is the (signed) classical artificial diffusivity, $D_{\mathrm{CLASS}} = uh/2$, scaled by a factor, ξ where

$$\xi = \coth(\alpha) - 1/\alpha$$

$$\alpha = \frac{uh}{2\kappa} \tag{D.3}$$

and κ is the physical dissipation of the species modeled. The function α is plotted in Figure D.2. Since $|\alpha| < 1$, this scheme always applies less dissipation than the classical scheme. Note that in the special case that $\kappa = 0$ (e.g. the continuity equation), $P \to \infty$, and $\xi \to \operatorname{sign}(u)$. Note also that the function ξ is smooth, yet the expression D.3 would fail in computations as $\alpha \to 0$. For $0 < |\alpha| < \pi$, we can formally expand the hyperbolic cotangent in a Taylor series

$$\coth\alpha = \frac{1}{\alpha} + \frac{\alpha}{3} - \frac{\alpha^3}{45} + 2\frac{\alpha^5}{945} - \ldots$$

and the first term cancels directly. In routine computation, one can truncate this series at the third term, and use it for $|\alpha| < 1.d-3$. (maximum error $\sim 2 \times 10^{-25}$).

D.3 Discussion

We have seen that the accuracy of our modified scheme depends strongly on the strength of the flow field, and is therefore not easy to predict in advance. We can expect that in the worst of cases, the scheme should perform at least as well as

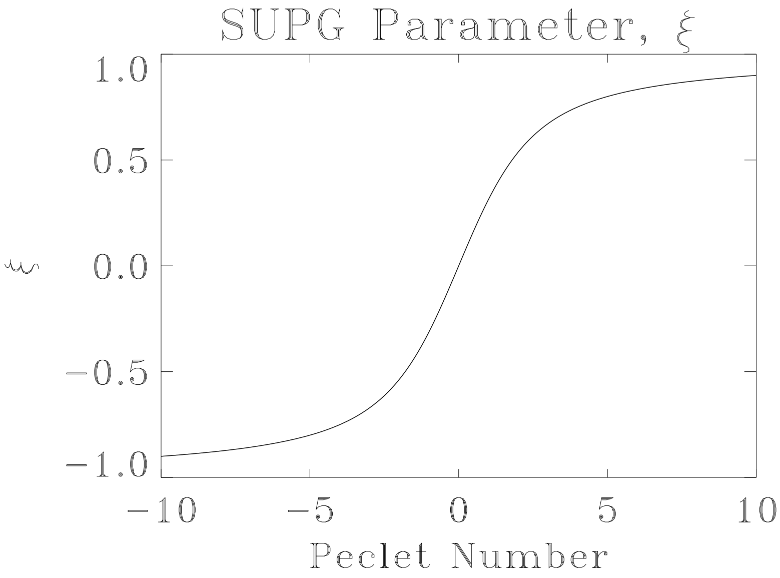

Figure D.2. SUPG parameter, ξ, as a function of Peclet number. The SUPG artificial dissipation is $\xi uh/2$, where u and h are the local flow speed and grid spacing, respectively. For a small number Peclet number, this parameter essentially converts classical, first-order artificial dissipation into a second order scheme.

classical first-order accurate artificial dissipation. The SUPG treatment is tested in the modeling of a laboratory plasma in Chapter 7, where it is found to work "suspiciously well".

We do however have reason to suspect the general appropriateness of the SUPG dissipation to our plasma fluid problem. In the above definitions, u was assumed to be the convection velocity, consistent with Reference [101]. But we showed in section ... that the information in our system will actually flow along *characteristics*, including acoustic propagations. Sound waves can often be considerably faster than convection, and would then lead to oscillatory solutions due to insufficient artificial dissipation. We can remedy the situation, if it does occur, by computing the above SUPG parameter, ξ, using the characteristic speeds ($\lambda_i \geq u$). However, in fluid plasmas, scheme will revert back to the classical diffusion method in that case, because where flow speeds are small, sound speeds are often large.

As a final note, observe that the numerical techniques of Patanker[71] are applied in the majority of modern tokamak edge fluid codes. Patanker's method is a staggered grid, finite-volume application of the SUPG methodology. In this sense, the improved discretization methods outlined here, roughly bring FMS2D in line with the existing literature, albeit in a more general and flexible way.

Appendix E

Model Specification

As discussed in Chapter 8, the equations that describe a typical realistic plasma model can be very complex, even for "simple" experiments such as PISCES. In this Appendix, we give a listing of the exact code used to specify the PISCES model (barring the atomic data table interpolation formulae available from References [150, 172], and transport coefficients computed using standard formulae from the NRL Formulary).

The purpose of showing the model specification code is two-fold. There are a large number of details that go into making a model "work" for a given plasma. We would like to have the model completely available for follow-on researchers to critique; all too often a particular model is buried deep in a messy Fortran implementation and must be re-written from scratch to be evaluated or modified. In the pages that follow, we include the actual source code that was used to compute the solution in Chapter 8. This avoids the efforts of formatting all the expressions in LaTeX, while also preventing the possibility of any errors in transcription.

The second, perhaps more important reason to display the Fortran model file is that *we can*—in the sense that we have structured the FMS2D code to accept the entire model specification via a single Fortran subroutine. This is possible because the model was suitably parameterized, as described in Chapter 5.

According to our generalized equation, the problem is fully specified by giving the concentration, flux, source and boundary functions. In the applications, there is actually a little more information that is required. The model implementation specification file must return the following, given a valid set of data describing the state, $\vec{s}$, state gradient, $\vec{\nabla} s$, position, $\vec{x}$, and time, t (or, using the Courier font and the 1D vector notation in the source code, the state, `s`, gradient, `grad_s` and `grad_s_r`, the position, `x` and `r`, and the time, `t`):

- The number of state profiles in the system, `N`: The number of states is required purely for bookkeeping purposes since the entire model is specified to FMS2D through this interface.

- Concentration functions, `C`: Formally, the concentrations can be arbitrary, or even zero, because when the steady equations are solved, $-\partial C/\partial t = 0$. However, the artificial flux used to stabilize the convection operator is applied as $D_{\mathrm{ART}} \nabla C$. Additionally successful time evolution to steady profiles usually requires the appropriate conserved quantity concentrations.

- Convective, `F_c, F_c_r`, and diffusive, `F_d, F_d_r`, flux functions in the direction along and across the local flux surface: The diffusive flux is used to compute an effective diffusivity for use in the SUPG artificial diffusivity and misaligned diffusion schemes (the computed diffusivity may not correspond to the physical diffusivity, as in the case of *flux-limited* diffusion). A null diffusion flux component, such as that occurring in the continuity equation, is properly treated.

- Source function `S`

- Boundary condition type, `bc_type`: ('local' and 'none' supported currently). The boundary condition type is a character string identifier, where 'null' specifies that internal procedures should be used to generate extrapolated states at the boundary node (see § 5.2.4). If 'local' is specified, the expression returned as the boundary residual will be minimized in conjunction with the other internal conservation equations.

- Boundary condition relaxation time, `tau` (see § 5.2.4).

- Boundary residual function, `B`: On solution→0.

- Initial conditions `s_0`: Initial conditions are specified usually in terms of spatial location only, but can in general be a function of initial values of the other variables or gradients.

- Characteristic propagation velocities, along `max_characteristic_speed`, across `max_characteristic_speed_r` the flux surfaces in the poloidal plane: Characteristic velocities are, nominally, the magnitude of the largest eigenvalue of the convective flux Jacobian (as detailed in § 5.2.2). If the SUPG discretization is being used (see Appendix D for discussion), this speed is set to the magnitude of the *convective* velocity.

- Characteristic state scale value, `s_scale`: The state scale should be the change in the state that is expected to significantly alter the physics of the models.

- Sample values, `sample_s`: Sample state values are used only upon initial start up, and should be set to values that allow a safe call of the full routine. Sample values are loaded using the Fortran statement "ENTRY" so that the bulk of the model file is skipped during the first pass until these sample values can be loaded.

Aside from this set of required data, the user is free (and encouraged) to construct whatever temporary variables that will clarify the model description. Template subroutines are available to aid in simple efficient set up of the model files, and various debugging tools are available to verify the correct physics implementation BEFORE insertion into the FMS2D structure for solution.

Considerable effort was devoted towards this setup, and we have paid a heavy price in terms of runtime efficiency. But changeout of an entire seven equation impurity transport model, say, to a simple benchmark test problem requires only the replacement of this file, and no changes to the discretization or solver parts of the code. We view this setup as uniquely required in the field of edge modeling, and thus an ideal way to investigate the rapidly evolving fluid models for tokamak edge calculations.

The PISCES model referred to as the "default" model employed for the Reference [170] is given as a typical example of a complex model specification. This code implements the 1-1/2 D model, so radial fluxes, characteristics and boundaries are shut off (actually, since they are not explicitly set, they default to the Neumann condition perpendicular to boundary).

```
c FMS:tag                    FMS-1D MODEL
c
c FMS:begin{}{ FMS MODEL NAME: }
c
c PISCES
c
c FMS:end{}
c
c FMS:begin{}{ OVERVIEW: }
c
c This is a model of the plasma and neutral gas transport
c in the UCLA PISCES experimental device. It is particularly
c intended to model gas target experiments, in the steady state.
c PISCES is a linear plasma device, with a target plate at one end
c and a plasma inlet at the other, and a linear, cyclindrically
c symetric magnetic field. The first 94 cm in from of the target
c is enclosed by a metallic tube.
c
c FMS:end{}
c
c FMS:begin{}{ DOMAIN: }
c
c The model describes the quantities
c along the central magnetic field line in PISCES, with
c the x = 0 coordinate corresponding to the target
c plate, and x increasing toward the inlet end (where the
c plasma source is located in the machine).  The boundaries
c are at the target (x = 0) and the inlet (x = Ldevice),
c and the problem is posed on the interval [0,Ldevice].
c
c FMS:end{}
c
c FMS:begin{}{ STATES: }
c
c (1) atomic hydrogen density (nH)
c (2) plasma density (np)
c (3) plasma velocity (v)
c (4) plasma electron temperature (Te)
c (5) modelcular hydrogen density (nH2)
c
c FMS:end{}
c
c FMS:begin{}{ EQUATIONS: }
c
c The model consists of 5 conservation equations:
c
c (1) diffusive transport equation for neutral atomic hydrogen
c (2) continuity equation for plasma density
c (3) Braginskii momentum equation for plasma momentum
c (4) Braginskii energy equation for electron energy
c (5) diffusive transport equation for neutral molecular hydrogen
c
c FMS:end{}
c
c FMS:begin{}{ BOUNDARY CONDITIONS: }
c
c Typical boundary conditions corresponding to the equations:
c
c (1) specified flux or density at target, zero flux at inlet
c (2) no condition at target (outflow), specified value at inlet
c (3) sheath velocity condition at target, zero velocity gradient
c     at inlet
c (4) sheath electron heat flux condition at target, fixed
c     electron temperature at inlet
c (5) specified flux or density at target, zero flux at inlet
c
c FMS:end{}
c
c FMS:begin{}{  INITIAL CONDITIONS: }
c
c Typical initial conditions are either a previously
c computed equilibrium, or analytic formulas with
c
c (1) nH: exponentially decreasing from given target to inlet values
c (2) np: constant at specified inlet value
c (3) v : a power law fit to specified Mach number at inlet and target
c (4) Te: linear fit to specified boundary values
c (5) nH2: exponentially decreasing from given target to inlet values
c
c FMS:end{}
```

```
c
c FMS:begin{}{  FEATURES AND COMMENTS: }
c
c All units are CGS, with gauss for magnetic field and electron-volts
c for temperature unless otherwise noted.
c
c The model includes the effects of neutral gas puffing
c and pumping, sidewall and target recycling and recombination,
c volume recombination, ionization, dissociation, charge exchange,
c elastic collisions, and perpendicular diffusive losses of plasma
c resulting from Bohm and classical transport.
c
c Other features: the heat flux for the electrons is flux limited.
c The plasma ions and neutral species have specified temperatures.
c
c Note positive flow velocity or flux corresponds to flow away from
c the target, towards the inlet.
c
c FMS:end{}
c
c FMS:begin{}{  GLOSSARY OF MODEL TERMS: }
c
c B...............B field strength on axis in PISCES
c BesselZero......first zero of the Bessel function J0.
c Bohm_coeff......factor to modulate the amount of Bohm diffusion
c Bohm_coeff_in...factor multiplying the Bohm diffusion rate inside the
c                 PISCES tube
c Bohm_coeff_out..factor multiplying the Bohm diffusion rate outside
c                 the PISCES tube
c BohrRadius......Bohr atomic radius
c Chi_perp_e......perpendicular thermal conductivity for electrons
c Chi_perp_i......perpendicular thermal conductivity for ions
c Cs..............sound speed
c dEe_SV2_2_10_tot..energy released ions from Janev reaction 2.2.10
c dEe_SV2_2_12....energy released ions from Janev reaction 2.2.12
c dEe_SV2_2_5.....energy released ions from Janev reaction 2.2.5
c dEe_SV2_2_6.....energy released ions from Janev reaction 2.2.6
c dEe_SV2_2_7.....energy released ions from Janev reaction 2.2.7
c dEe_SV2_2_8.....energy released ions from Janev reaction 2.2.8
c dEe_SV2_2_9_tot..energy released ions from Janev reaction 2.2.9
c Dn0.............general neutral diffusivity function
c Dperpn0_coeff...factor multiplying the classical diffusion rate
c DuctWidth.......half width of the PISCES tube
c D_classical_H...classical perpendicular diffusivity for plasma due to
c                 collisions with H
c D_classical_H2..classical perpendicular diffusivity for plasma due to
c                 collisions with H2
c D_nH............H diffusivity
c D_nH2...........H2 diffusivity
c D_perp..........total perpendicular diffusivity of the plasma
c D_perp_bohm.....Bohm diffusivity
c D_perp_neutral..total perpendicular plasma diffusivity for plasma due
c                 to collisions with neutrals
c D_Spitzer_e.....Spitzer parallel conductivity
c Ei..............electron energy loss per ionization
c Eifnc...........electron energy loss per ionization function
c fluxlimit_e.....flux limit fractional multiplier for electrons
c f_Hp............probability that an H from an H2 turns into an H+
c                 upon dissociation
c gas_source......total spatially distributed puffing rate (per unit
c                 cross sectional area) for H2
c grad_nH.........gradient of nH
c grad_nH2........gradient of nH2
c grad_nH2_inlet..inlet boundary value for grad_nH2
c grad_nH_inlet...inlet boundary value for grad_nH
c grad_np.........gradient of np
c grad_p..........gradient of plasma pressure
c grad_p_e........gradient of electron pressure
c grad_p_i........gradient of ion pressure
c grad_Te.........gradient of Te
c grad_v..........gradient of v
c grad_v_inlet....inlet boundary value for grad_v
c H2toHpFrac......function that returns the probability that an H from
c                 an H2 turns into an H+ upon dissociation
c H2_radius.......approximate radius of H2 molecule
c H2_TargInFlux...flux of H2 into device from the target
c H2_TargOutFlux..flux of H2 out of device from the target
c H2_TargRecycle..recycling flux of H2 from the target
c H2_targ_albedo..albedo of target for H2 reflection
c heatfluxbc_Te...set to .true. if heat flux boundary condition for
```

```
c                 electron energy equation is in use, .false. if not.
c Hp__H2_targfrac..fraction of H+ that recycles as H2 off target plate
c Hp__H2_wallfrac..fraction of H+ that recycles as H2 off sidewall
c Hp__H_targfrac...fraction of H+ that recycles as H off target plate
c Hp__H_wallfrac...fraction of H+ that recycles as H off sidewall
c H_radius........approximate radius of H atom
c H_TargInFlux....flux of H into device from the target
c H_TargOutFlux...flux of H out of device from the target
c H_TargRecycle...recycling flux of H from the target
c H_targ_albedo...albedo of target for H reflection
c junkcut.........a temporary cutoff factor
c junkmult........a temporary cut off parameter
c k...............boltzman constant (in ergs/eV)
c Ldevice.........length of PISCES device
c LimitedIdentity..function that is the identity, but with limited
c                 extreme values. Used to prevent evaluating terms
c                 outside their domain of validity. For example, used
c                 to prevent putting 0 or negative temperatures into
c                 rate functions.
c Lperp...........perpendicular length scale in PISCES device
c Lslot...........length of tube in PISCES
c L_mfp_H.........mean free path for H before ionization
c L_mfp_H2........mean free path for H2 before dissociation
c M...............ion mass
c max_speed.......maximum characteristic speed in the system
c Mach_inlet......boundary value for mach number at inlet
c Mach_target.....boundary value for mach number at target
c mu_H............H reduced mass
c mu_H2 ..........H2 reduced mass
c m_e.............electron mass
c m_electron......electron mass
c m_proton........proton mass
c NBohmDiffusionCoefficient..Bohm diffusivity function
c NElectronViscosity0..electron viscocity function
c nH..............atomic hydrogen (H) density
c nH_limited......limited value of nH
c nH_0............analytic initial data for H
c nH2.............molecular hydrogen (H2) density
c nH2_0...........analytic initial data for H2
c nH2_inlet.......inlet boundary value for nH
c nH2_limited.....limited value of nH2
c nH2_target......target boundary value for nH2
c nH_cutoff.......parameter used to limit very low nH values
c nH_inlet........inlet boundary value for nH
c nH_startcut.....parameter used to limit very low nH values
c nH_target.......target boundary value for nH
c NIonViscosity0..ion viscosity function
c Nin.............number of states in IC data file
c Nnue............electron coulomb collision frequency function
c Nnun0...........neutral plasma collision frequency
c np..............plasma density
c np_0............analytic initial data for np
c np_inlet........inlet boundary value for np
c np_limited......limited value of np
c nu_e............electron coulomb collision frequency
c nu_H............total H collision frequency
c nu_H2...........total H2 collision frequency
c Nwci............ion cyclotron frequency function
c pi..............value of the constant Pi
c ProjFactor......poloidal projection factor (cosine of angle between B
c                 filed and the poloidal plane; cos(0) = 1 for PISCES).
c puff_profile....distributed gas puffing profile
c qe..............electron thermal conductivity
c qe_c............shorthand for qe_clssical
c qe_classical....classical electron thermal conductivity
c qe_l............shorthand for qe_limited
c qe_limit........electron heat flux convective limit for flux limiter
c qe_target.......electron heat flux at the target
c RecombRate......recombination rate
c Rrecombine......recombination rate function
c R_H.............H sidewall recycling coefficient
c R_H2............H2 sidewall recycling coefficient
c R_inout_asym....factor used to modulate sidewall recycling
c                 coefficient
c R_in_factor.....modulating factor for sidewall recycling inside the
c                 PISCES tube
c R_in_factor_target...R_in_factor at start of PISCES tube
c R_in_factor_tubeend..R_in_factor at end of PISCES tube
c R_out_factor....factor multiplying sidewall recycling rates outside
c                 the PISCES tube
```

```
c R_targetH.......target recycling factor for H
c R_targetH2......target recycling factor for H2
c R_wallH.........wall recycling factor for H
c R_wallH2........wall recycling factor for H2
c r...............coordinate along PISCES radius
c scale_nH........typical scale for value of nH
c scale_nH2.......typical scale for value of nH2
c scale_np........typical scale for value of np
c scale_Te........typical scale for value of Te
c scale_v.........typical scale for value of v
c SigmaVcx........<sigma v> for charge exchange
c SigmaVeifnc.....<sigma v> function for electron impact ionization,
c                 this approximation is a function of Te only
c SigmaVeii.......<sigma v> used for electron impact ionization
c SigmaVin........<sigma v> for ion-H neutral elastic collisions
c SigmaVin_H2.....<sigma v> for ion-H2 neutral elastic collisions
c SigmaV_eonH2....<sigma v> for e against H2 collisions
c SigmaV_H2onH....<sigma v> for H2 against H collisions
c SigmaV_H2onH2...<sigma v> for H2 against H2 collisions
c SigmaV_HonH.....<sigma v> for H against H collisions
c Sigma_H2_H2.....cross section for H2 against H2 collisions
c Sigma_H_H.......cross section for H against H collisions
c Sigma_H_H2......cross section for H against H2 collisions
c SV2_2_10........<sigma v> for Janev reaction 2.2.10
c SV2_2_10__e_H2__e_e_Hp_H...<sigma v> function 2.2.10 (Janev tables)
c SV2_2_5.........<sigma v> for Janev reaction 2.2.5   (Janev tables)
c SV2_2_5__e_H2__e_H_H.......<sigma v> function 2.2.5  (Janev tables)
c SV2_2_6.........<sigma v> for Janev reaction 2.2.6   (Janev tables)
c SV2_2_6__e_H2__e_H_H.......<sigma v> function 2.2.6  (Janev tables)
c SV2_2_7.........<sigma v> for Janev reaction 2.2.7   (Janev tables)
c SV2_2_7__e_H2__e_H_H.......<sigma v> function 2.2.7  (Janev tables)
c SV2_2_8.........<sigma v> for Janev reaction 2.2.8   (Janev tables)
c SV2_2_8__e_H2__e_H_H.......<sigma v> function 2.2.8  (Janev tables)
c SV2_2_9.........<sigma v> for Janev reaction 2.2.9   (Janev tables)
c SV2_2_9__e_H2__e_e_H2p.....<sigma v> function 2.2.9  (Janev tables)
c SV_coll_H.......<sigma v> for all H  collisions combined
c SV_coll_H2......<sigma v> for all H2 collisions combined
c t...............time coordinate
c TargetPuffFlux..total H2 puffing flux through the target
c Te..............electron temperature
c Te_0............analytic initial data for Te
c Te_inlet .......inlet boundary value for Te
c Te_limited......limited value for Te
c Te_target.......target boundary value for Te
c TH..............H temperature
c TH2.............H2 temperature
c THmin...........min allowed value for H temperature in PISCES
c Ti..............background ion temperature
c v...............plasma velocity along PISCES axis
c v_0.............analytic initial data for v
c viscosity.......total plasma viscosity
c Vth_H...........H thermal speed
c Vth_H2..........H2 thermal speed
c v_target........plasma velocity at the target
c wc_H............H ion cyclotron frequency
c wc_H2...........H2 ion cyclotron frequency
c x...............coordinate along PISCES centerline
c x_pump_end......end coordinate of tent shaped H2 puffing profile
c x_pump_peak.....coordinate of peak of tent shaped H2 puffing profile
c x_pump_start....starting coordinate for tent shaped H2 puffing
c                 profile
c
c FMS:end{}
c
c FMS:begin{}{  APPENDIX: SUMMARY OF THE GENERAL FMS-1D MODEL }
c
c (terms inside [] are vectors with N components)
c The FMS-1D model problem is to solve for the unknown
c states [s](x,t), which are determined by a system of
c N equations, of the form
c
c     d[C]/dt  + d( [F_c] + [F_d] )/dx = [S]
c
c with N boundary conditions at each endpoint, of the form
c
c     d[B]/dt = -[B]/[tau] ( => [B] = [0] at steady state)
c
c and N initial data conditions, of the form
c
c     [s](x,t=0) = [s_0](x).
```

```
c
c Here [C], [F_c], [F_d], [S], [B], [tau] can be functions
c of x,t,[s],d[s]/dx, and represent the concentration,
c convective  flux, diffusive flux, source term, steady
c state boundary condition and boundary condition time
c constants in the model. In addition, these terms can
c depend on a continuation parameter, lambda, which
c varies from 0 to 1 when continuation methods are in use.
c
c FMS:end{}
c
c FMS:begin{}{  APPENDIX: SUMMARY OF THE FMS-1D MODEL IMPLEMENTATION }
c
c The model is implemented as a FORTRAN subroutine which returns
c values for N, [C], [F_c], [F_d], [S], [B], [tau], [s_0],
c given values for x,t,[s],d[s]/dx, lambda, and boundary index
c (the boundary index specifies for which part of the boundary, i.e.
c for which endpoint, to construct a boundary condition; the
c continuation parameter lambda has the default value 0 when
c continuation methods are not in use). In addition the routine must
c also return values for quantities required by the numerical solution
c methods: the max characteristic speeds, the boundary condition types,
c scale values for the states, and a sample numerical value for each
c state.
c
c
c The formal specification for this routine is:
c
c SYNTAX: subroutine FMS1DModel(
c        x, t, [s], [grad_s], lambda, boundary_index,
c        N, [C], [F_c], [F_d], [S], [B], [tau], [s_0],
c        [bc_type], [max_characteristic_speed], [s_scale], [s_sample])
c
c DATA TYPES: [] denote vector types; [bc_type] are string variables,
c boundary_index and N are integer, and the rest are double
c precision.
c
c EFFECTS: returns appropriate values for quantities in the last
c two rows of arguments, given values for the quantities in the
c first row of arguments. It must not modify the values of the
c first row quantities. Refer to above for meaning of the terms.
c
c NAMING: In the implementation, all these standard argument terms are
c given names starting with "FMS_", so that they don't interfere
c with natural choice of names inside the routine. Also, d[s]/dx
c is referred to as [grad_s], in order to use the FORTRAN name
c character set.
c
c NOTE: The comments for the FORTRAN code can also contain commands for
c the FMS markup language (fmsml), which provide instructions
c for various types of automated processing of the model routine.
c The commands for this language all begin with string "FMS:". The
c language fmsml is documented elsewhere.
c
c
c FMS:end{}
c

c FMS:tag              FMS-1D  MODEL ROUTINE
cFMS:begin{ variation}{}
      subroutine FMS2DModel( FMS_x, FMS_r, FMS_t, FMS_s, FMS_grad_s,
     & FMS_grad_s_r, FMS_lambda, FMS_boundary_index, FMS_N, FMS_C,
     & FMS_F_c, FMS_F_c_r, FMS_F_d, FMS_F_d_r, FMS_Src, FMS_B,
     & FMS_tau, FMS_s_0, FMS_bc_type,FMS_max_characteristic_speed,
     & FMS_max_characteristic_speed_r, FMS_s_scale, FMS_sample_s )

c FMS:begin{}{type declarations for all standard subroutine arguments}
      implicit none

      double precision FMS_x, FMS_t,FMS_s(*),FMS_grad_s(*),FMS_lambda,
     & FMS_C(*),FMS_F_c(*),FMS_F_d(*),FMS_Src(*),FMS_B(*),FMS_tau(*),
     & FMS_s_0(*), FMS_max_characteristic_speed(*), FMS_s_scale(*),
     & FMS_sample_s(*)
      double precision FMS_r, FMS_grad_s_r(*), FMS_F_c_r(*),
     & FMS_F_d_r(*), FMS_max_characteristic_speed_r(*)
      integer FMS_N, FMS_boundary_index
```

```
      character FMS_bc_type(*)*(*)
c FMS:end{}
c FMS:begin{}{ type declarations for all user defined names }
      double precision nH , nH2, np, v, Te, x, t, Cs, Vth_H, Vth_H2,
     & D_Spitzer_e, Dn0, D_nH, D_nH2, nu_H, nu_H2, SigmaVeifnc,
     & SigmaVeii, SigmaV_HonH, SigmaV_H2onH, SigmaV_H2onH2,
     & SigmaV_eonH2, SV2_2_10__e_H2__e_e_Hp_H, SV2_2_5__e_H2__e_H_H,
     & SV2_2_6__e_H2__e_H_H, SV2_2_7__e_H2__e_H_H,
     & SV2_2_8__e_H2__e_H_H, SV2_2_9__e_H2__e_e_H2p,qe_limit,
     & viscosity, NIonViscosity0, NElectronViscosity0, Rrecombine,
     & Nnue, nu_e, Eifnc, Ei, R_inout_asym, R_H, R_H2, RecombRate,
     & puff_profile, R_in_factor, H2toHpFrac, f_Hp, dEe_SV2_2_12,
     & dEe_SV2_2_10_tot, dEe_SV2_2_9_tot, dEe_SV2_2_8, dEe_SV2_2_7,
     & dEe_SV2_2_6, dEe_SV2_2_5  ,SV2_2_10, SV2_2_9, SV2_2_8, SV2_2_7,
     & SV2_2_6, SV2_2_5, mu_H2, mu_H, Nwci, wc_H2, wc_H, SV_coll_H2,
     & SV_coll_H, D_classical_H2, D_classical_H, D_perp_neutral,
     & D_perp_bohm, Bohm_coeff, NBohmDiffusionCoefficient, D_perp,
     & LimitedIdentity,junkmult,grad_nH ,grad_nH2, grad_np, grad_v,
     & grad_Te, qe_classical, qe, grad_p, grad_p_i, grad_p_e , r
      double precision scale_nH, scale_nH2, scale_np, scale_v,
     & scale_Te, max_speed, nH_0, nH2_0, np_0, v_0, Te_0
      double precision k, m_e, m_electron, m_proton, M, pi, BohrRadius,
     & BesselZero, DuctWidth, Lperp, Ldevice, Lslot, Chi_perp_e,
     & Chi_perp_i, B, ProjFactor, THmin, Nnun0, SigmaVin, SigmaVin_H2,
     & SigmaVcx, Sigma_H_H2, Sigma_H2_H2, Sigma_H_H, fluxlimit_e,
     & fluxlimit_i, nH_startcut, nH_cutoff, nH_target, nH2_target,
     & nH_inlet, nH2_inlet, grad_nH_inlet, grad_nH2_inlet, np_inlet,
     & grad_v_inlet, Te_target, Te_inlet, Ti, TH, TH2, Mach_inlet,
     & Mach_target, gas_source, TargetPuffFlux, Bohm_coeff_in,
     & Bohm_coeff_out, Dperpn0_coeff, R_out_factor, R_wallH2, R_wallH,
     & R_targetH2, R_targetH, H_targ_albedo, H2_targ_albedo,
     & Hp__H_wallfrac, Hp__H2_wallfrac, Hp__H_targfrac,
     & Hp__H2_targfrac, H_radius, H2_radius, x_pump_start, x_pump_peak,
     & x_pump_end, R_in_factor_tubeend, R_in_factor_target, qe_c, qe_l
      double precision v_target, qe_target, H_TargInFlux,
     & H2_TargInFlux , H_TargOutFlux, H2_TargOutFlux, H_TargRecycle,
     & H2_TargRecycle
      double precision junkcut, L_mfp_H, L_mfp_H2,
     & nH_limited, nH2_limited, np_limited, Te_limited
      logical heatfluxbc_Te
      integer Nin
c FMS:end{}
c FMS:begin{}{ FORTRAN language statement function definitions }
c FMS:end{}
c FMS:begin{}{ FORTRAN language parameter definitions }
c FMS:end{}
c FMS:begin{}{ local model term definitions }
       x  = FMS_x
       r  = FMS_r
       t  = FMS_t
       nH  = FMS_s(1)
       np  = FMS_s(2)
       v   = FMS_s(3)
       Te  = FMS_s(4)
       nH2 = FMS_s(5)
      grad_nH  = FMS_grad_s(1)
      grad_np  = FMS_grad_s(2)
      grad_v   = FMS_grad_s(3)
      grad_Te  = FMS_grad_s(4)
      grad_nH2 = FMS_grad_s(5)
c
c     derived quantities
c
c
c     define physical variables with limited values for
```

```
c     use in rate functions that have restricted range
c     of input values
c
      nH_limited  = LimitedIdentity(nH,1.0d10,2.0d10,1.0d15,2.0d15)
      np_limited  = LimitedIdentity(np,1.0d10,2.0d10,1.0d15,2.0d15)
      Te_limited  = LimitedIdentity(Te,0.3d0,0.5d0,1.0d3,2.0d3)
      nH2_limited = LimitedIdentity(nH2,1.0d10,2.0d10,1.0d16,2.0d16)
      k  = 1.602d-12
      m_e  = 9.109d-28
      m_electron  = m_e
      m_proton  = 1.6726d-24
      M  = 1.0d0 * m_proton
      pi  = 3.14159d0
      BohrRadius  = 5.2918d-9
      BesselZero  = 2.405d0
      DuctWidth  = 2.25d0
      Lperp  = DuctWidth/BesselZero
      Ldevice   = 100.0d0
      Lslot  = 90.0d0
      Chi_perp_e  = 0.0d0
      Chi_perp_i  = 0.0d0
      B  = 1900.0d0
      ProjFactor  = 1.0d0
      THmin  = 1.0d0
      Nnun0  = 1.5d-7
      SigmaVin  = 1.0d-8
      SigmaVin_H2  = 1.0d-8
      SigmaVcx  = 1.0d-8
      Sigma_H_H    = 3.0d-15
      Sigma_H2_H2  = 3.0d-15
      Sigma_H_H2   = 3.0d-15
      fluxlimit_e  = 0.5d0
      fluxlimit_i  = 0.5d0
      nH_startcut  = 5.0d13
      nH_cutoff  = 5.0d12
      nH_target  = 1.0d14
      nH2_target  = 5.0d14
      nH_inlet   = 1.00d12
      nH2_inlet   = 1.00d12
      grad_nH_inlet  = 0.0d0
      grad_nH2_inlet  = 0.0d0
      np_inlet   = 3.7d12
      grad_v_inlet  = 0.0d0
      Te_target  = 2.1d0
      Te_inlet   = 18.9d0
      Ti    = 2.0d0
      TH    = 0.5d0
      TH2   = 0.1d0
      Mach_inlet   = -0.1d0
      Mach_target  = -1.0d0
      TargetPuffFlux  = 0.52d19  * 0.8d0 * 1.00d0
      gas_source   = 0.52d19  * 0.2d0 * 1.00d0
      Bohm_coeff_in   = 1.0d0    * 1.5d0
      Bohm_coeff_out   = 1.0d0    * 1.5d0
      Dperpn0_coeff    = 1.0d0
      R_in_factor_tubeend = 0.98d0
      R_in_factor_target = 0.98d0
      R_out_factor   = 0.5d0
      R_wallH2   = 1.0d0
      R_wallH   = 1.0d0
      R_targetH2   = 0.98d0
      R_targetH   = 0.98d0
      H_targ_albedo   = 1.0d0
      H2_targ_albedo   = 1.0d0
      Hp__H_wallfrac   = 0.5d0
      Hp__H2_wallfrac  = 1.0d0 - Hp__H_wallfrac
      Hp__H_targfrac   = 0.5d0
      Hp__H2_targfrac  = 1.0d0 - Hp__H_targfrac
      H_radius   = BohrRadius
      H2_radius   = BohrRadius
      x_pump_start  = 0.0d0
      x_pump_peak   = 0.01d0
      x_pump_end   = 40.0d0
      heatfluxbc_Te   = .true.
      Cs     = SQRT(k*(Te_limited + Ti)/M)
      Vth_H  = SQRT(8.0d0*k*TH /(pi*M)      )
      Vth_H2 = SQRT(8.0d0*k*TH2/(pi*2.0d0*M))
```

```
      SigmaVeii     = SigmaVeifnc(np_limited,Te_limited)
      SigmaV_HonH   = Sigma_H_H   * Vth_H
      SigmaV_H2onH  = Sigma_H_H2  * Vth_H
      SigmaV_H2onH2 = Sigma_H2_H2 * Vth_H2
      SV2_2_5   = SV2_2_5__e_H2__e_H_H(Te_limited)
      SV2_2_6   = SV2_2_6__e_H2__e_H_H(Te_limited)
      SV2_2_7   = SV2_2_7__e_H2__e_H_H(Te_limited)
      SV2_2_8   = SV2_2_8__e_H2__e_H_H(Te_limited)
      SV2_2_9   = SV2_2_9__e_H2__e_e_H2p(Te_limited)
      SV2_2_10  = SV2_2_10__e_H2__e_e_Hp_H(Te_limited)
      SigmaV_eonH2  = SV2_2_5
     &              + SV2_2_6
     &              + SV2_2_7
     &              + SV2_2_8
     &              + SV2_2_9
     &              + SV2_2_10
      nu_e = Nnue(np_limited,Te_limited)
      nu_H  = 3.0*np_limited*(SigmaVeii + SigmaVcx + SigmaVin) +
     &                nH_limited*SigmaV_HonH +
     &                nH2_limited*SigmaV_H2onH
      nu_H2 = 3.0*np_limited*SigmaV_eonH2 +
     &                nH2_limited*SigmaV_H2onH2 +
     &                nH_limited*SigmaV_H2onH
      D_nH  = Dn0(TH,   M,nu_H )
      D_nH2 = Dn0(TH2,2*M,nu_H2)
      D_Spitzer_e =  (2.4d20)*(Te_limited**2.5d0)
      qe_limit = fluxlimit_e*SQRT(2.0d0*k*Te_limited/(m_e*pi))*np*k*Te
      viscosity = (4.0d0/3.0d0)*
     &( NElectronViscosity0(np_limited, Te_limited) +
     &  NIonViscosity0(np_limited,Ti,1.0d0,M) )
      f_Hp      = H2toHpFrac(Te_limited)
c
c     energy losses (in eV) due to dissociating
c     H_2 -> H + H+ and H2^+ -> H + H+ (From Janev's book);
c     these energy loss channels were not included
c     in Eifnc(n,Te), which is the energy loss only
c     of H -> ... -> H+ processes.
c     SV2_2_12 is the dominant H2^+ -> H + H+ reaction.
c     We add this loss to that of H2 -> H2^+ reactions,
c     to represent the energy loss in going from H2^+ to H, H+.
c
      dEe_SV2_2_12     = 10.5d0
      dEe_SV2_2_10_tot = 18.0d0  +  dEe_SV2_2_12
      dEe_SV2_2_9_tot  = 15.4d0  +  dEe_SV2_2_12
      dEe_SV2_2_8      = 21.5d0
      dEe_SV2_2_7      = 34.6d0
      dEe_SV2_2_6      = 15.3d0
      dEe_SV2_2_5      = 10.5d0
      if (x .lt. Lslot ) then
        R_in_factor =
     &         R_in_factor_target +
     &        (x/Lslot)* (R_in_factor_tubeend - R_in_factor_target)
        R_inout_asym    = R_in_factor
Bohm_coeff = Bohm_coeff_in
      else
        R_inout_asym    = R_out_factor
Bohm_coeff = Bohm_coeff_out
      endif
c     puff_profile normalized to have unit integral
      if     ( x .lt. x_pump_start ) then
        puff_profile =  0.0d0
      elseif (x .lt. x_pump_peak ) then
        puff_profile = (x - x_pump_start)/(x_pump_peak - x_pump_start)
      elseif ( x .lt. x_pump_end ) then
        puff_profile =
     &  1.0d0 - (x - x_pump_peak)/(x_pump_end - x_pump_peak)
```

```
      else
        puff_profile =  0.0d0
      endif
      puff_profile = puff_profile/
     & ( 0.5d0 * ( x_pump_peak - x_pump_start ) +
     &   0.5d0 * ( x_pump_end - x_pump_peak )
     & )
c
c     define D_perp---perp diffusion due to plasma-neutral
c     interactions + Bohm losses. The plasma neutral contribution
c     is a multiple of the classical formula (kTe/mu wc^2) * n0 <sv>,
c     and the bohm is a multiple of classical bohm.
c
      mu_H2 = M*(2*M)/(M + 2*M)
      mu_H  = M*M/(M + M)
      wc_H2 = Nwci(B,1.0d0,mu_H2)
      wc_H  = Nwci(B,1.0d0,mu_H)
      SV_coll_H2 = SigmaV_eonH2 + SigmaVin_H2
      SV_coll_H  = SigmaVeii + SigmaVcx + SigmaVin
      D_classical_H2 = (k*Te_limited/(mu_H2*wc_H2**2))*nH2*SV_coll_H2
      D_classical_H  = (k*Te_limited/(mu_H*wc_H**2))*nH*SV_coll_H
      D_perp_neutral = (D_classical_H2 + D_classical_H)
      D_perp_bohm    = NBohmDiffusionCoefficient(Te_limited,B)
      D_perp = Dperpn0_coeff*D_perp_neutral + Bohm_coeff*D_perp_bohm
c     turn off recombination
      RecombRate = 0.0d0
      R_H  = R_inout_asym * R_wallH * Hp__H_wallfrac
      R_H2 = R_inout_asym * R_wallH2 * Hp__H2_wallfrac * 0.5d0
c
c     energy loss per ionization
c
      Ei = Eifnc(np_limited,Te_limited)
c
c     derived gradient quantities
c
      qe = 1.5d0*np*k*Te*v - (ProjFactor**2)*D_Spitzer_e*k*grad_Te
c
c     note: grad_parallel == (ProjFactor * grad_poloidal)
c
      qe_classical = - D_Spitzer_e * k * (ProjFactor * grad_Te)
      grad_p  = np*k*grad_Te + grad_np*k*(Te + Ti)
      grad_p_i  = grad_np*k*Ti
      grad_p_e  = np*k*grad_Te + grad_np*k*Te
      qe_c = qe_classical
      qe_l = qe_limit
      if( nH .gt. nH_startcut ) then
          junkcut = 1.0d0
      elseif( nH .gt. nH_cutoff) then
          junkcut = (nH - nH_cutoff)/(nH_startcut - nH_cutoff)
      else
          junkcut = 0.0d0
      endif
c    define mean free paths for ionization
      L_mfp_H  = Vth_H /(nH  * SigmaVeii)
      L_mfp_H2 = Vth_H2/(nH2 * SigmaV_eonH2)
c
c     boundary condition terms
c
      v_target     = Mach_target * Cs * ProjFactor
      qe_target  =  4.5d0* np*k*Te*v_target
```

```
      H_TargInFlux   = 0.5d0 * Vth_H  * nH  - 0.5d0 * D_nH  * grad_nH
      H2_TargInFlux  = 0.5d0 * Vth_H2 * nH2 - 0.5d0 * D_nH2 * grad_nH2
      H_TargOutFlux  = 0.5d0 * Vth_H  * nH  + 0.5d0 * D_nH  * grad_nH
      H2_TargOutFlux = 0.5d0 * Vth_H2 * nH2 + 0.5d0 * D_nH2 * grad_nH2
      H_TargRecycle  = - R_targetH  * Hp__H_targfrac  * np * v_target
      H2_TargRecycle = - R_targetH2 * Hp__H2_targfrac * np * v_target
     &                * 0.5d0
c
c     initial data for states
c
      nH_0  = nH_target  * EXP((x/Ldevice)*LOG(nH_inlet /nH_target ))
      nH2_0 = nH2_target * EXP((x/Ldevice)*LOG(nH2_inlet/nH2_target))
      Te_0 = Te_target + (x/Ldevice)*(Te_inlet - Te_target)
      np_0 = np_inlet
      v_0  =  ProjFactor* SQRT(k*( Te_0 + Ti )/M)*(
     &      Mach_inlet+
     &     (Mach_target-Mach_inlet)*((Ldevice-x)/Ldevice)**20
     &                         )
c
c     maximum characteristic speed (= max eigenvalue of the linearized
c     convective flux (d[F_c]/d[C]) matrix), or |v| if using the SUPG
c     discretization
c
      max_speed = ABS(v)
c FMS:end{}
c FMS:begin{}{ FMS number of equations in the system }
      FMS_N = 5
c FMS:end{}
c FMS:begin{}{ FMS concentrations }
      FMS_C(1) = nH
      FMS_C(2) = np
      FMS_C(3) = M * np * v
      FMS_C(4) = (3.0/2.0) * np * k * Te
      FMS_C(5) = nH2
c FMS:end{}
c FMS:begin{}{ FMS convective fluxes }
      FMS_F_c(1) = 0.0d0
      FMS_F_c(2) = np * v
      FMS_F_c(3) = M * np * v * v  +  (ProjFactor**2)*np * k*(Te + Ti)
      FMS_F_c(4) = (5.0/2.0) * np * k * Te * v
      FMS_F_c(5) = 0.0d0
      FMS_F_c_r(1) = 0.0d0
      FMS_F_c_r(2) = 0.0d0
      FMS_F_c_r(3) = 0.0d0
      FMS_F_c_r(4) = 0.0d0
      FMS_F_c_r(5) = 0.0d0
c FMS:end{}
c FMS:begin{}{ FMS diffusive fluxes }
      FMS_F_d(1) = -Dn0(TH, M, nu_H) * grad_nH
      FMS_F_d(2) = 0.0d0
c     poloidal projection of parallel, with parallel v gradient:
      FMS_F_d(3) = ProjFactor*(- viscosity*(grad_v*ProjFactor) )
c     poloidal projection of parallel flux limited heat flux:
      FMS_F_d(4) = ProjFactor* qe_c/( 1.0d0 + ABS(qe_c/qe_l) )
      FMS_F_d(5) = -Dn0(TH2, 2.0d0 * M, nu_H2) * grad_nH2
      FMS_F_d_r(1) = 0.0d0
      FMS_F_d_r(2) = 0.0d0
```

```
      FMS_F_d_r(3) = 0.0d0
      FMS_F_d_r(4) = 0.0d0
      FMS_F_d_r(5) = 0.0d0
c FMS:end{}
c FMS:begin{}{ FMS sources }
c     neutral H density source
      FMS_Src(1) = - nH    * np * SigmaVeii
     &       + 2.0d0 * ( 1 - f_Hp ) * np * nH2 * SigmaV_eonH2
     &       + R_H   * np * D_perp/(Lperp**2)
     &       + np    * np * RecombRate
     &       - junkcut*0.3d0 * Vth_H * nH *(4.0d0/6.0d0)/DuctWidth
c     plasma density source
      FMS_Src(2) = + nH * np * SigmaVeii
     &           + 2.0d0   * f_Hp * nH2 * np * SigmaV_eonH2
     &           - np * D_perp/(Lperp**2)
     &           - np * np * RecombRate
c     plasma momentum source
      FMS_Src(3) = 0.d0
     &      + M * v * FMS_Src(2)
     &      - M * v * np * nH * SigmaVeii
     &      - M * v * 2.0d0   * f_Hp * nH2 * np * SigmaV_eonH2
     &   + 2.0*MIN(1.0d0, L_mfp_H/DuctWidth)*(
     &      - M * v * np * nH * SigmaVin
     &      - M * v * np * nH * SigmaVcx
     &                                   )
     &   + 2.0*MIN(1.0d0, L_mfp_H2/DuctWidth)*(
     &      - M * v * np * nH2 * SigmaVin_H2
     &                                     )
c     electron energy source
      FMS_Src(4) =
     &      + (3.0/2.0) * k * Te * FMS_Src(2)
c    can try not all energy is lost when a particle leaves radially;
c    0 = all lost, 1 = none lost:
     &      + 0.9 * (3.0/2.0) * k * Te * np * D_perp/(Lperp**2)
     &      - (3.0/2.0) * k * Te * nH * np * SigmaVeii
     &      - (3.0/2.0) * k * Te * 2.0d0*f_Hp*nH2*np*SigmaV_eonH2
     &      - k * Ei * np * nH * SigmaVeii
     &      - 3 * (m_e/M) * nu_e * np * k * (Te - Ti)
     &      - 3 * (m_e/M) * (Nnun0*nH)  * np * k * (Te - TH)
     &      - 3 * (m_e/(2.0d0*M)) * (Nnun0*nH2) * np * k * (Te - TH2)
     &      - k * dEe_SV2_2_5      * np * nH2 * SV2_2_5
     &      - k * dEe_SV2_2_6      * np * nH2 * SV2_2_6
     &      - k * dEe_SV2_2_7      * np * nH2 * SV2_2_7
     &      - k * dEe_SV2_2_8      * np * nH2 * SV2_2_8
     &      - k * dEe_SV2_2_9_tot  * np * nH2 * SV2_2_9
     &      - k * dEe_SV2_2_10_tot * np * nH2 * SV2_2_10
c     neutral H2 density source
      FMS_Src(5) = - nH2  * np *  SigmaV_eonH2
     &           + R_H2  * np * D_perp/(Lperp**2)
     &       + gas_source * puff_profile
     &      +junkcut*0.5d0*0.3d0 * Vth_H * nH *(4.0d0/6.0d0)/DuctWidth

c FMS:end{}
c FMS:begin{}{ FMS boundary condition terms }

c   note: boundary conditions are zero gradient by default
      if( FMS_boundary_index .eq. 1 ) then
           FMS_bc_type(1) = 'local'
           FMS_B(1)       = H_TargInFlux -
     &                (H_targ_albedo*H_TargOutFlux + H_TargRecycle)
           FMS_tau(1)     = 1.0d-6
           FMS_bc_type(2) = 'none'
```

```
           FMS_B(2)       = 0.0d0
           FMS_tau(2)     = 1.0d-8
           FMS_bc_type(3) = 'local'
           FMS_B(3)       = v  - v_target
           FMS_tau(3)     = 1.0d-7
           FMS_bc_type(4) = 'local'
           if( heatfluxbc_Te ) then
               FMS_B(4) = qe - qe_target
   else
       FMS_B(4) = Te - Te_target
   endif
           FMS_tau(4)     = 1.0d-6
           FMS_bc_type(5) = 'local'
           FMS_B(5)       = H2_TargInFlux -
     &       (H2_targ_albedo*H2_TargOutFlux + H2_TargRecycle
     &                                      + TargetPuffFlux)
           FMS_tau(5)     = 1.0d-6
      elseif( FMS_boundary_index .eq. 2 ) then
           FMS_bc_type(1) = 'local'
           FMS_B(1)       = grad_nH   - grad_nH_inlet
           FMS_tau(1)     = 1.0d-8
           FMS_bc_type(2) = 'local'
           FMS_B(2)       = np        - np_inlet
           FMS_tau(2)     = 1.0d-8
           FMS_bc_type(3) = 'local'
           FMS_B(3)       = grad_v    - grad_v_inlet
           FMS_tau(3)     = 1.0d-8
           FMS_bc_type(4) = 'local'
           FMS_B(4)       = Te        - Te_inlet
           FMS_tau(4)     = 1.0d-8
           FMS_bc_type(5) = 'local'
           FMS_B(5)       = grad_nH2  - grad_nH2_inlet
           FMS_tau(5)     = 1.0d-8
      endif
c FMS:end{}
c FMS:begin{}{ FMS initial data }
c     note: save effort by setting IC only at t=0,
c           this may be significant for interpolation ICs
      if (t.EQ.0.d0) then
         FMS_s_0(1) = nH_0
         FMS_s_0(2) = np_0
         FMS_s_0(3) =  v_0
         FMS_s_0(4) = Te_0
         FMS_s_0(5) = nH2_0
         call PISCESn_SUPG_064( r, x, FMS_s_0, Nin )
      endif
c FMS:end{}
c FMS:begin{}{ FMS max characteristic speeds }
c     note: the purely diffusive eqns have 0 characteristic speeds:
      FMS_max_characteristic_speed(1) = 0.0d0
      FMS_max_characteristic_speed(2) = max_speed
      FMS_max_characteristic_speed(3) = max_speed
      FMS_max_characteristic_speed(4) = max_speed
      FMS_max_characteristic_speed(5) = 0.0d0
      FMS_max_characteristic_speed_r(1) = 0.0d0
      FMS_max_characteristic_speed_r(2) = 0.0d0
      FMS_max_characteristic_speed_r(3) = 0.0d0
      FMS_max_characteristic_speed_r(4) = 0.0d0
```

```
      FMS_max_characteristic_speed_r(5) = 0.0d0
c FMS:end{}
c FMS:begin{}{ FMS state scales }
      FMS_s_scale(1) =  nH_limited
      FMS_s_scale(2) =  np_limited
      FMS_s_scale(3) =  MAX( ABS(v), ProjFactor * Cs * 0.1d0 )
      FMS_s_scale(4) =  Te_limited
      FMS_s_scale(5) =  nH2_limited
c FMS:end{}
c FMS:begin{}{ FMS sample state values }
      ENTRY sample_values(FMS_sample_s)
      FMS_sample_s(1) = 1.0d13
      FMS_sample_s(2) = 1.0d13
      FMS_sample_s(3) = 1.0d5
      FMS_sample_s(4) = 10.0d0
      FMS_sample_s(5) = 1.0d13
c FMS:end{}
c FMS:tag           END OF FMS MODEL ROUTINE
      return
      end
```

Bibliography

[1] T. Dolan. *Fusion Research*, ch. 2. Hemisphere Publishing Corporation (1982).

[2] R. W. Conn, *Magnetic Fusion Reactors.* in (E. Teller, Ed.) "Fusion". Academic Press, New York (1981).

[3] G. Bateman. *MHD Instabilities.* MIT Press, Cambridge, Mass. (1978).

[4] E. A. Lazarus, *Dependence of $\beta \cdot \tau$ on Plasma Shape in DIII–D.* Technical Report GA–A21306, General Atomics (1993).

[5] S. Hamada, *Hydromagnetic Equilibria and Their Proper Coordinates. Nuclear Fusion*, **2** p. 23 (1962).

[6] G. R. Tynan et. al., *Steady–State Convection and Fluctuation–Driven Particle Transport in the H-Mode Transition. Phys. Rev. Lett.*, **68**(20) p. 3032 (1992).

[7] P. H. Rebut, *The ITER Challenge.* in "Proc 17th Symposium on Fusion Technology" (Rome, 1992).

[8] W. J. Corbett et. al., *Boundary Asymmetries and Plasma Flow to the ALT-II Toroidal Belt Pump Limiter. J. Nucl. Mat.*, **162-164** p. 221 (1989).

[9] H. F. Dylla and the TFTR Team. *J. Nucl. Mat.*, **145-147** p. 48 (1987).

[10] G. W. Barnes et. al. in "Proc 13th IEEE Conf on Fusion Engineering", p. 937 (1989).

[11] ITER Director, *Detail of the ITER Outline Design Report.* presented to the 4th meeting of the ITER-TAC (10-12 January 1994).

[12] F. C. Chen. *Introduction to Plasma Physics and Controlled Fusion*, ch. 5. Plenum Press (1984).

[13] K. Borrass. *Nuclear Fusion*, **31** p. 1035 (1991).

[14] M. S. Day, *Modelling the Plasma Edge in Fusion Tokamaks.* Master's thesis, University of California, Los Angeles (1992).

[15] L. Spitzer Jr. *Physics of Fully Ionized Gases.* Interscience, New York (1956).

[16] C. S. Pitcher and P. C. Stangeby, *The Parallel Transport of Ions in a Magnetized Plasma. Physics Physics & Controlled Fusion*, **31**(8) p. 1305 (1989).

[17] F. C. Chen. *Introduction to Plasma Physics and Controlled Fusion.* Plenum Press (1984).

[18] E. L. Vold, *Transport in the Tokamak Plasma Edge.* PhD thesis, University of California, Los Angeles (1989).

[19] G. R. Tynan et. al., *Effects of Radial Electric Fields on the Turbulence and Transport in the TEXTOR edge and SOL Plasma. J. Nucl. Mat.*, **196-198** p. 770 (1992).

[20] E. Mazzucato at. al. *Phys. Rev. Lett.*, **36** p. 792 (1976).

[21] Y. Q. Tao, *Theoretical Analysis and Numerical Simulation of Plasma Behavior in a Strongly Sheared Electric Field.* PhD thesis, UCLA (1994 (forthcoming)).

[22] R. J. Groebner et. al. *Phys. Rev. Lett.*, **6**(25) p. 3015 (1990).

[23] W. L. Rowan et. al. *Nuclear Fusion*, **27**(7) p. 1105 (1987).

[24] K. W. Gentle et. al. *Plasma Phys. Contr. Fus.*, **29**(9) p. 1077 (1987).

[25] S. K. Erents. *Nuclear Fusion*, **28** p. 1209 (1988).

[26] P. C. Stangeby. *Nuclear Fusion*, **7** p. 543 (1990).

[27] M. Ulrickson and et. al., *A Review of Carbon Blooms on JET and TFTR.* in "Proc. 9th International Conference on Plasma-Surface Interations in Controlled Fusion Devices", Vol. 176-177, p. 44 (Bournemouth, UK, 21-25 May, 1990).

[28] G. R. Hopkins and J. M. Rawls. *Nucl. Tech.*, **43** p. 382 (1979).

[29] B. Badger et. al., *NUWMAK–A Tokamak Reactor Design Study, Ch. X.* Technical Report UWFDM-330, Univ. of WI, Madison (1979).

[30] G. M. McCracken and et. al., *Erosion and Redeposition Processes in Tokamaks.* in "15th Symposium on Fusion Technology", Vol. 11, p. 151 (Utrecht, Netherlands 19-23 Sept., 1988).

[31] J. M. Pedgley and G. M. McCracken, *Plasma Sheath Properties Calculated Using Measured Secondary Electron Emmision Coefficients. Plasma Phys. Control. Fusion*, **35**(3) p. 397 (1993).

[32] J. M. Pedgley and G. M. McCracken, *Measurement of Secondary Electron Emission for Fusion Related Materials. J. Nucl. Mat.*, **196-198** p. 1053 (1992).

[33] W. Eckstein, J. Bohdansky, and et. al., *Physical Sputtering. Atomic and Plasma-Material Interaction Data for Fusion*, **1** p. 51 (1991).

[34] W. Ehrenberg and et. al., *Hydrogen and Helium Recycling in Tokamaks with Carbon Walls.* in "Proc. 8th International Conference on Plasma-Surface Interations in Controlled Fusion Devices", Vol. 162-164, p. 264 (Julich, West Germany, 2-6 May, 1988).

[35] ASDEX team, *The H-mode of ASDEX. Nuclear Fusion*, **29**(11) p. 1959 (1989).

[36] K. H. Burrell et. al., *Energy Confinement in Auxiliary Heated Divertor and Limiter Discharges in the DIII-D Tokamak.* in "Proc. 12th Int'l Conf of the IAEA, Nice, France", Vol. 1, p. 193 (1988).

[37] D. N. Hill et. al., *The effect of ELMs on Edge Plasma Scaling in DIII-D. J. Nucl. Mat.*, **196-198** p. 204 (1992).

[38] P. C. Liewer et. al. *Nuclear Fusion*, **25**(5) p. 543 (1985).

[39] J. Boedo et. al., *Poloidal Asymmetries and Convection Cells in TEXTOR.* in "12th International Conference on Plasma Surface Interactions in Controlled Fusion Devices" (1991).

[40] K. McCormack, *Particle and Energy Transport Scalings in the ASDEX Scrape-Off Layer. J. Nucl. Mat.*, **196-198** p. 264 (1992).

[41] T. Petrie et. al., *Recent Gaseous Divertor Experiments in DIII-D.* in "Europhysics Abstracts, Proc. 18th European Conf. on Controlled Fusion and Plasma Physics, Berlin, Germany", Vol. 15C-III, p. 237 (1991).

[42] G. Janeschitz et. al. in "Europhysics Abstracts, Proc. 19th European Conf. on Controlled Fusion and Plasma Physics, Innsbruck, Austria", Vol. 16C-II, p. 727 (1992).

[43] The JET Team, *Latest Results From JET. J. Nucl. Mat.*, **196-198** p. 3 (1992).

[44] R. J. Hawryluk et. al. Technical Report PPPL-2417, Princeton Plasma Physics Laboratory (1987).

[45] W. J. Corbett, *Tokamak Experiments with a Toroidal Pump Limiter.* PhD thesis, UCLA (1989).

[46] D. S. Gray, *Investigation of Particle Exhaust During Auxillary Heating with the Pumped Limiter ALT-II in the TEXTOR Tokamak.* PhD thesis, UCLA (1994).

[47] C. P. Ritz et. al. *Nuclear Fusion*, **27** p. 1125 (1987).

[48] W. L. Rowan. *Nuclear Fusion*, **27** p. 1105 (1987).

[49] P. Coutiere. *Phys. Lett. A*, **163** p. 204 (1992).

[50] Hutchinson et. al., *The Physics and Engineering of Alcator C-MOD.* Technical Report PFC/RR-99-11, MIT PFC (1988).

[51] G. Campbell D. M. Goebel and R. W. Conn. *J. Nucl. Mat.*, **121** p. 27 (1984).

[52] D. Reiter, *Progress in Two-Dimensional Plasma Edge Modelling. J. Nucl. Mat.*, **196-198** p. 241 (1992).

[53] E. L. Vold, *Multidimensional and Multifluid Plasma Edge Modeling: Status and New Directions. Contrib. Plasma Physics*, **32** p. 404 (1992).

[54] David L. Book, *Plasma Formulary.* Technical Report 177-4405, Naval Research Laborartory (1990).

[55] I. Bernstein and P. Catto, *Generalized Gyrokinetics. Phys. Fl.*, **28** p. 1342 (1985).

[56] X. Lee, J. R. Myra, and P. Catto, *General Frequency Gyrokinetics. Phys. Fl.*, **26** p. 223 (1982).

[57] J. M. Dawson, C. G. Hsi, and R. Shanny, *Some Investigations of Nonlinear Plasma Behavior on One-Dimensional Plasma Models.* in "Third Conference of Plasma Physics and Controlled Nuclear Fusion Research, August 1968", Vol. 1, p. 735. International Atomic Energy Agency (1969).

[58] Marable and Oblow, *Applications of Neutron Transport Codes to Neutral Transport in Plasma. Nucl. Sci. and Eng*, **61** p. 90 (1976).

[59] M. Hasan. *J. Comp. Phys.*, **71** p. 371 (1987).

[60] D. Heifetz. *J. Comp. Phys.*, **46** p. 309 (1982).

[61] M. Petravic, *Modeling of Ultra-High Recycling Divertors with the PLANET Code.* Technical Report PPPL-2912, PPPL (1993).

[62] H. Grad, *On the Kinetic Theory of Rarefied Gases. Comm. Pure & App. Math*, **2** p. 331 (1949).

[63] S. Chapman and T. G. Cowling. *Mathematical Theory of Nonuniform Gases.* Cambridge University Press, Great Britain (1953).

[64] S. I. Braginskii, *Transport Processes in a Plasma. Reviews of Plasma Physics*, **1** p. 205 (1965).

[65] M. F. A. Harrison and P. J. Harbour, *Plasma Characteristics and Gas Transport in the Single-Null Poloidal Divertor of the International Tokamak Reactor. Nucl. Technol./Fusion*, **3** p. 432 (1983).

[66] P. J. Harbour, *Thickness of the Scrape-Off layer of a Large Tokamak with a Poloidal Divertor. Nuclear Fusion*, **24**(9) p. 1211 (1984).

[67] *International Tokamak Reactor: Phase One.* Technical Report STI/PUB/619, International Atomic Energy Agency (1982).

[68] R. B. Campbell et. al. *J. Nucl. Mat.*, **426** p. 196 (1992).

[69] D. A. Knoll, *Development and Application of a Direct Newton Solver for the Two-Dimensional Tokamak Edge Plasma Fluid Equations.* PhD thesis, University of New Mexico (1991).

[70] W. H. Press et. al. *Numerical Recipes: The Art of Scientific Computing, 2nd edition*, ch. 9. Cambridge University Press (1992).

[71] S. V. Patanker. *Numerical Heat Transfer and Fluid Flow.* Hemisphere Publishing Corporation (1981).

[72] M. L. Watkins and P.-H.Rebut. in "Europhysics Abstracts, Proc. 19th European Conf. on Controlled Fusion and Plasma Physics, Innsbruck, Austria", Vol. 16C-II, p. 731 (1992).

[73] K. Parbhaker, *The Coupling of Impurity Atoms with Scrape-Off Layer Plasmas. Phys. Plasmas*, **1**(8) p. 2798 (1994).

[74] B. L. Merriman, *Recent Results for 1-1/2D ITER Gas Target Divertor Modelling.* in "Proc. Scrape-Off Layer and Divertor Theory Meeting" (Dallas, TX 16-17 March 1994).

[75] L. Schmitz et. al., *Heat and Momentum Flow in a Simulated Tokamak Gas Target Divertor. submitted to Phys. Fluids.*

[76] B.J. Braams, *Computational Studies in Tokamak Equilibrium and Transport.* PhD thesis, University of Utrecht, Netherlands (1986).

[77] M. Petravic, *Orthogonal Grid Construction for Modeling of Transport in Tokamaks. J. Comp. Phys.*, **73**(1) p. 125 (1987).

[78] M. Day and B. Braams. *Private Communications, collaboration funded by NYU under DOE DE-FG02-86ER53223* (1993).

[79] D. K. Rayevski, *Radiation transport in a high density divertor plasma.* in "Bulletin of the APS", 2R16, p. 1920 (1993).

[80] P. Helander, S. I. Krasheninnikov, and P. J. Catto, *Fluid Equations for a Partially Ionized Plasma.* Technical Report CTH-IEFT/PP-1994-09:ISSN 0281-1308, Institute for Electromagnetic Field Theory and Plasma Physics (1994).

[81] T. D. Rognlien et. al., *A Fully Implicit, Time Dependent 2-D Fluid Code for Modelling Tokamak Edge Plasmas. J. Nucl. Mat.*, **196-198** p. 347 (1992).

[82] C. Hirsch. *Numerical Computation of Internal and External Flows.* J. W. Wiley (1988).

[83] D. A. Knoll, *Advanced Numerical Solution Algorithms Applied to the Multispecies Edge Plasma Equations.* in "Proc. Scrape-Off Layer and Divertor Theory Meeting" (Dallas, TX 16-17 March 1994).

[84] Dana A. Knoll, *Newton-Krylov-Schwarz Methods Applied to the Tokamak Edge Plasma Fluid Equations.* in (D. Keyes, Y. Saad, and D. Truhlar, Eds.) "Domain-Based Parallelism and Problem Decomposition Methods in Computational Science and Engineering", ch. 5. SIAM, Philidelphia, PA (1995).

[85] C. Koesoemodiprodjo, *Numerical Fluid Edge Plasma Simulation Using a Multigrid-Newton Scheme with Continuation.* in "Bulletin of the APS", 9V9, p. 2123 (1993).

[86] A. Brandt, *Multi-Level Adaptive Techniques.* Technical Report RC6026, IBM.

[87] T. D. Rognlien et. al., *2-D Edge Plasma Transport Calculations with UEDGE.* in "Bulletin of the APS", 2R20, p. 1920 (1993).

[88] R. Cohen and T. Rognlien, *Finite Mean Free Path Effects in Tokamak Scrape Off Layers.* in "Bulletin of the APS", 1Q9, p. 1892 (1993).

[89] S. I. Krasheninnikov, *Some Investigations of Nonlinear Plasma Behavior on One-Dimensional Plasma Models.* in "1993 U.S. Edge Physics: Theory and Applications Workshop" (Albuquerque, NM 26-28 April 1993).

[90] M. E. Rensink and et. al., *Simulation of Radiative Divertors.* in "1993 U.S. Edge Physics: Theory and Applications Workshop" (Albuquerque, NM 26-28 April 1993).

[91] D. Reiter. *The EIRENE Code, Version: Jan.92 Users Manual.* Berichte des Forschungszentrums Jülich, Jülich, Bundesrepublik Deutschland (1992).

[92] X. Bonnin et. al., *Neutral Transport Calculations in TdeV Using the EIRENE and B2.5 Codes.* in "Bulletin of the APS", 1Q9, p. 1892 (1993).

[93] M. Baelmans, *Code Improvements and Applications of a Two-Dimensional Edge Plasma Model for Toroidal Devices.* PhD thesis, Ecole Royale Militaire-Koninklijke Militaire School (1993).

[94] H. Gerhauser and H. Claasen, *Calculation of Poloidal Rotation in the Edge Plasma of Limiter Tokamaks.* Technical report, Institute fur Plasmaphysik (1987).

[95] K. Parbhaker, *The Effect of Diamagnetic Drift on the Boundary Conditions in Tokamak Scrape-Off-Layers and the Distribution of Plasma Fluxes Near the Target. Plasma Phys. Control. Fusion*, **36** p. 1485 (1994).

[96] M. Petravic. *J. Nucl. Mat.*, **196-198** p. 883 (1992).

[97] M. Petravic and M. Kuo-Petravic, *Effect of Ion Drifts on the Properties of the Tokamak Scrape-Off Plasma.* Technical Report PPPL-2554, PPPL (1988).

[98] M. Petravic, *Modeling of Gaseous Divertors with the PLANET code.* in "Proceedings of the 4th International Workshop on Plasma Edge Theory in Fusion Devices" (Varenna, Italy 4-6 Oct 1993).

[99] N. Ueda, M. Kasai, M. Tanaka, M. Sugihara, and S. Sengoku, *Development of a Two-Dimensional Fluid Code and its Application to the Doublet III Divertor Experiment. Nuclear Fusion*, **28**(7) p. 1183 (1988).

[100] R. A. Vesey, *A Finite Element Approach to Edge Plasma Modeling.* PhD thesis, Rensselaer Polytechnic Institute, Troy NY (1992).

[101] A. N. Brooks and T. J. R. Hughes, *Streamline Upwind/Petrov Galerkin Formulations for Convection Dominated Flows with Particular Emphasis on the Incompressible Navier-Stokes Equations. Comp. Meth. App. Mech. & Eng.*, **32** p. 199 (1982).

[102] D. A. Knoll and P. R. HcHugh, *Simulation of Radiative Divertors with Tilted Plates.* in "Bulletin of the APS", 2R24, p. 1921 (1993).

[103] A. P. Kuprat and A. H. Glasser, *Progress Toward a Moving Finite Element Model of the Tokamak Scrapeoff Region.* in "Bulletin of the APS", 1Q10, p. 1892 (1993).

[104] K. Miller and R. N. Miller, *Moving Finite Elements I. SIAM J. Num. Anal.*, **18**(6) p. 20 (1981).

[105] D. Post, *The Role of Atomic and Molecular Processes in Power and Particle Control in Fusion Experiments and ITER.* presentation to UC San Diego Fusion Group (1/12/95).

[106] D. J. Acheson. *Elementary Fluid Dynamics*, ch. 6. Clarendon Press, Oxford (1990).

[107] Richard E. Meyer. *Introduction to Mathematical Fluid Dynamics*, ch. 3. Dover, New York (1971).

[108] O. D. Kellogg. *Foundations of Potential Theory.* Springer, Berlin (1966).

[109] L. Boltzmann. *Lectures on Gas Theory (English translation from German).* University of California Press, Berkeley (1964).

[110] D. Ross, *On Standard Forms for Transport Equations and Fluxes. Comments Plasma Phys. Control. Fusion*, **12**(3) p. 155 (1989).

[111] D. Ross, *On Standard Forms for Transport Equations and Quasilinear Fluxes. Plasma Phys. Control. Fusion*, **34**(2) p. 137 (1992).

[112] F. Hinton, *The Radial Electric Field Near a Separatrix in a Diverted Tokamak.* in "Sherwood Fusion Theory Meeting". AIP (March 14-16, 1994).

[113] A. Jameson, *Numerical Solutions of the Euler Equations By a Finite Volume Method Using Runge-Kutta Time-Stepping Schemes. AIAA*, **81**(1259) (1981).

[114] Dale A. Anderson and et. al. *Computational Fluid Mechanics and Heat Transfer*, ch. 6. Hemisphere, New York (1984).

[115] T. J. R. Hughes and L. P. Franca, *A New Finite Element Method for Computational Fluid Dynamics: VII. The Stokes Problem with Various Well–Posed Boundary Conditions: Symmetric Formulations that Converge for All Velocity/Pressure Spaces. Comp. Meth. App. Mech. & Eng.*, **65** p. 85 (1987).

[116] J. W. Macarthur and S. V. Patanker, *Robust Semidirect Finite Difference Methods for Solving the Navier-Stokes and Energy Equations. Intl J. Num Meth in Fl*, **9** p. 325 (1989).

[117] K. A. Winkler et. al., *Implicit Adaptive-Grid Radiation Hydrodynamics.* in (J. U. Brackbill and B. I. Cohen, Eds.) "Multiple Time Scales". Academic Press Inc. (1985).

[118] J. J. Dogarra and E. Gosse, *Distribution of Mathematical Software via Electronic Mail. Comm. ACM*, **30** p. 403 (1987).

[119] M. Seager, *A SLAP for the Masses.* Technical Report UCRL–100267, Lawrence Livermore National Laboratory (1988).

[120] George and Liu. *Computer Solution of Large Sparse Positive Definite Systems.* Prentice Hall, New Jersey (1981).

[121] E. Anderson and et. al. *LAPACK User's Guide, second ed.* SIAM, Berlin (1994).

[122] D. Kincaid, R. Grimes, J. Respess, and D. Young, *ITPACK 2B: A Fortran Implementation of Adaptive Accelerated Iterative Methods for Solving Large Sparse Linear Systems.* Technical Report CNA-173, Center for Numerical Analysis, University of Texas, Austin (1981).

[123] W. H. Press et. al. *Numerical Recipes: The Art of Scientific Computing, 2nd edition.* Cambridge University Press (1992).

[124] J. A. George and J. W. Liu. *Computer Solution of Large Sparse Positive Definite Systems*, ch. 4. Prentice-Hall, Englewood-Cliffs, New Jersey (1981).

[125] J. M. Varah, *On the Solution of Block Tridiagonal Systems Arising from Certain Finite-Difference Equations. Math Comp.*, **26** p. 859 (1972).

[126] G. Golub and C. F. Van Loan. *Matrix Computations, 2nd edition*, ch. 2. The Johns Hopkins University Press (1990).

[127] J. R. Shewchuk, *An Introduction to the Conjugate Gradient Method Without the Agonizing Pain.* Technical Report CMU-CS-94-125, Carnegie Mellon University (1994). Avail. via anonymous ftp at 128.2.222.79:/1994/CMU-CS-94-125.ps.

[128] S. Cowin and R. Hart, *Errors in the Orientation of the Principal Stress Axes if Bone Tissue is Modeled as Isotropic. J. Biomech*, **23**(4) p. 349 (1990).

[129] C. Lamberti and F. Sgallari, *Edge Detection and Velocity Field for the Analysis of Heart Motion.* in (A.G. Amsterdam Cappellini, V.; Constantinides, Ed.) “Digital Signal Processing 91–Proceedings of the International Conference”, p. 603. Elsevier (1991).

[130] S. Paipetis et. al., *Constitutive Relations of Periodic Laminated Composites with Anisotropic Dissipation. Arch. of App. Mech.*, **64**(1) p. 32 (1993).

[131] U. Christensen, *Some Geodynamical Effects of Anisotropic Viscosity. Geophys. J. R. astr. Soc.*, **91** p. 711 (1987).

[132] P. Tyvand and L. Storesletten, *Onset of Convection in an Anisotropic Porous Medium with Oblique Principal Axes. J. Fl. Mech.*, **226** p. 371 (1991).

[133] Rong-Yeu Chang et. al., *Simulation of Director Orientation of Liquid Crystalline Polymers in 2-D Flows. J. Non-Newtonian Fl. Mech.*, **55**(1) p. 1 (1994).

[134] V. Canuto, *The Mixing Length Parameter Alpha. Astronomy and Astrophysics*, **227**(1) p. 282 (1990).

[135] R. Kunz and B. Lakshminarayana, *Explicit Navier-Stokes Computation of Cascade Flows Using the k- in Turbulence Model. AIAA*, **30**(1) p. 13 (1992).

[136] C. Rappaport and E. Smith, *Anisotropic FDFD Computed on Conformal Meshes. IEEE Transactions on Magnetics*, **27**(5) p. 3848 (1991). Proc. from Fourth IEEE Conf on Electromagnetic Field Comp.

[137] Yu Sheng-Tao et. al., *Direct Calculations of Waves in Fluid Flows using High-Order Compact Difference Scheme. AIAA*, **32**(9) p. 1766.

[138] M. Berger and J. Oliger, . *J. Comp. Phys.*, **53** p. 484 (1984).

[139] M. Berger and P. Colella, *Local Adaptive Mesh Refinement for Shock Hydrodynamics. J. Comp. Phys.*, **82** p. 64 (1989).

[140] J. H. Bramble and B. E. Hubbard, *A Theorem on Error Estimation for the Finite Difference Analogues of the Dirichlet Problem for Elliptic Equations. Contrib. to Diff. Eq.*, p. 319 (1962).

[141] W. Layton, *On the Principal Axes of Diffusion in Difference Schemes for 2D Transport Problems. J. Comp. Phys.*, **90** p. 336 (1990).

[142] L. Lyons and D. Pridmore-Browm, *Force Balance Near an X Line in a Collionless Plasma. J. Geophys. Research*, **95**(A12) p. 20903 (1990).

[143] CIMS-NYU B. Braams. *Private Communications* (1994).

[144] G. K. McCormack, S. A. Pietrzyk, and et. al., *Characterization of the ASDEX Scrape-Off-Layer Density Profile During Neutral Beam Heating.* in "Proc. 8th International Conference on Plasma-Surface Interations in Controlled Fusion Devices", Vol. 162-164, p. 264 (Julich, West Germany, 2-6 May, 1988).

[145] K. W. Gentle, *Determination of Particle Transport Coefficients in ASDEX By Gas Modulation. Nuclear Fusion*, **32**(2) p. 217 (1992).

[146] G. K. McCormack, *Particle and Energy Transport Scalings in the ASDEX Scrape-Off-Layer. J. Nucl. Mat.*, **196-198** p. 264 (1992).

[147] The ASDEX Group, *The H-mode of ASDEX. Nuclear Fusion*, **29**(11) p. 1959 (1989).

[148] W. Koppendorfer, *[ASDEX Upgrade: A Tokamak Experiment with a Reactor-Relavent Diverter]. Physickaliche Blaetter (in German)*, **46**(8) p. 324 (1990).

[149] T. Rognlien, *A Cartesian SOL Plasma Problem.* announced via electronic distribution (12/94).

[150] PPPL D. Post. *Reaction data distributed to the community via internet* (1993).

[151] R. K. Janev et. al. *Elementary Processes in Hydrogen and Helium Plasmas.* Number 4 in Springer Series on Atoms and Plasmas. Springer-Verlag (1987).

[152] J. G. Watkins and et. al., *Scrape-Off Layer Measurements in DIII-D. J. Nucl. Mat.*, **196-198** p. 829 (1992).

[153] A. Mirza and G. Murtaza, *An Improved Nonlocal Heat Transport Formula for Steep Temperature Gradients. Physica Scripta*, **41** p. 262 (1990).

[154] W. Fechner and F. Mayer, *Analysis of Flux-Relaxed Electron Heat Waves. Phys. Fluids*, **27**(6) p. 1538 (1984).

[155] J. Luciani et. al., *Quasistatic Heat Front and Delocalized Heat Flux. Phys. Fluids*, **28**(3) p. 835 (1985).

[156] A. Nishiguchi and et. al., *Kinetic Effects of Electron Thermal Conduction on Implosion Hydrodynamics. Phys. Fluids B*, **4**(2) p. 417 (1992).

[157] G. Pomraning, *A Comparison of Various Flux-Limiters and Eddington Factors.* Technical Report UCID-19220, Lawrence Livermore National Laboratory (1981).

[158] E. Zawaideh and et. al., *Generalized Parallel Heat Transport Equations in Collisional to Weakly Collisional Plasmas. Phys. Fluids*, **31**(11) p. 3280 (1988).

[159] W. Fechner and et. al., . *Phys. Fluids*, **27**(-) p. 1552 (1984).

[160] S. Krasheninnikov, *On Nonlocal Electron Heat Conduction. Phys. Fluids B*, **5**(1) p. 74 (1993).

[161] A. Winslow, . *Nucl. Sci. Eng.* , **32** p. 101 (1968).

[162] C. Lund and J. Wilson, . submitted (1980).

[163] C. Levermore and G. Pomraning, *A Flux-Limited Diffusion Theory. Astrophys. J.*, **248** p. 321 (1981).

[164] M. Day and B. Merriman, *Fluid Modeling of Divertors Using the FMS Code.* in "Bulletin of the APS", 1R17, p. 1531 (1994).

[165] L. Schmitz and R. Lehmer et. al. *J. Nucl. Mat.*, **176-177** p. 522 (1990).

[166] F. H. Tenney and G. Lewin. Technical Report MATT-1050, PPPL (1974).

[167] W. L. Barr and B. Grant Logan. *Fusion Technology*, **18** p. 251 (1990).

[168] Lipschulz. *Recent Gaseous Divertor Experiments in DIII-D* (to be published in J. Nucl. Mater.).

[169] G. Janeschultz, S. Clement, and et. al. in "Europhysics Abstracts, Proc. 19th European Conf. on Controlled Fusion and Plasma Physics, Innsbruck, Austria", Vol. 16C-II, p. 727 (1992).

[170] L. Schmitz, *Heat and Momentum Flow in a Simulated Tokamak Gas Target Divertor.* submitted to *Phys. Fl. B* (1994).

[171] D. A. Knoll and et. al., *2-D Fluid Modeling of ITER Divertor Scenarios.* in "Bulletin of the APS", 7R18, p. 1712 (1994).

[172] R. Janev et. al. *J. Nucl. Mat.*, **121** (1984).

[173] I. S. Gradshteyn and I. M. Ryzhik. *Tables of Integrals, Series and Products.* Academic Press, New York (1965).

[174] J. Duderstadt and W. Martin. *Transport Theory.* John Wiley & Sons (1978).

[175] B. J. Lee. *Private Communications* (1995).

[176] F. L. Hinton, *Collisional Transport in Plasma.* in (A. A. Galeev and R. N. Sudan, Eds.) “Handbook of Plasma Physics”, Vol. 1 “Basic Plasma Physics I”, p. 147. North Holland, Amsterdam (1983).

[177] Y. L. Igitkhanov. *Contrib. to Plasma Phys.*, **28** p. 477 (1988).

[178] R. Campbell and et. al., *One-Dimensional Modeling of Radiative Divertors.* in “1993 U.S. Edge Physics: Theory and Applications Workshop” (Albuquerque, NM 26-28 April 1993).

[179] D. A. Knoll et. al., . in “4th International Workshop on Plasma Edge Theory in Fusion Devices”. *to be published in* Contr. Plasma Phys. (Oct. 1993).