Appendix A. Codes for Physics Assessments and Design

Physics and engineering design of the QPS experiment required development of new tools or extension of existing tools for magnetic configuration optimization, MHD equilibrium and stability, neoclassical transport and energetic orbit confinement, and design of complex coils. The magnetic configuration optimization procedure is described in Sect. 3.1 and is implemented in the ORNL code called STELLOPT.

A.1. MHD Equilibrium and Stability Codes

A.1.1. VMEC

The ORNL VMEC equilibrium code [1] has been used for the routine generation of three-dimensional (3-D) equilibria for stability and transport studies, and it has been incorporated in the configuration optimizer for generating candidate QPS configurations and assessing coil set flexibility. The VMEC code solves the 3-D equilibrium equations using a representation for the magnetic field that assumes nested flux surfaces. It uses an inverse moments method, in which the geometric coordinates R and Z are expanded in Fourier series in both a poloidal angle variable and the toroidal angle(for three dimensions). The coefficients R_{mn} and Z_{mn} in this series expansion are functions of the normalized toroidal flux s, where s = 0 is the magnetic axis (which can be a helical curve in three dimensions) and s = 1 is the plasma boundary. Here, m is the poloidal and n is the toroidal Fourier mode number. The boundary Fourier coefficients $R_{mn}(1)$ and $Z_{mn}(1)$ can either be constant (corresponding to a "fixed-boundary" equilibrium calculation), or they may be self-consistently computed from the MHD force balance equation at the plasma-vacuum boundary (for a "free-boundary" calculation [2]). Internally, VMEC computes an additional "renormalization" stream function (λ) which is used to optimize, dynamically and at every radial surface, the convergence rate in Fourier space for the spectral sum $\Sigma\,(R_{mn}^2 + Z_{mn}^2)$. In the original VMEC, radial mesh gridding is staggered, with the $R_{mn}(s)$ and $Z_{mn}(s)$ coefficients defined on integral radial mesh points $s_j = (j-1)/(N_s-1)$, where N_s is the number of radial nodes, and the lambda coefficients on half-integer mesh points interleaving the R_{mn}, Z_{mn} mesh. This scheme has been proven to lead to excellent radial resolution as well as minimal mesh separation (at least for large aspect ratio plasmas and with limited angular resolution meshes).

VMEC has been used as the primary design tool for all of today's existing stellarator experiments. As such, it has been extensively benchmarked against other 3D equilibrium codes including the Chodura-Schlüter code, BETA, PIES, and HINT. In addition, experimental "benchmarking" of VMEC calculations has occurred via electron fluorescence mapping of vacuum magnetic surfaces in ATF and HSX (see the HSX web site). Recently, X-Ray emissivity measurements at finite β on W7-AS [3] have confirmed the accuracy of the VMEC flux surface calculations.

Significant improvements have been made to the VMEC code in the context of the NCSX and QPS design efforts. It has been redifferenced to improve the convergence both on finer angular and radial meshes as well as for equilibria with a wide range of rotational transform profiles. In VMEC,

the inverse equations are cast as second order equations (in radius) for the Fourier components of R, Z, and λ. As noted above, λ has been previously differenced radially on a mesh centered between R, Z nodes, which greatly improved the radial resolution. This could be done to second order accuracy (in $h_s = 1/[N_s-1]$) since no radial derivatives of λ appear in its defining equation, $J^s = 0$ (here, J^s is the contravariant radial component of the current). Near the magnetic axis, however, a type of numerical interchange instability (mesh separation) has been observed as the angular resolution is refined. This behavior has prevented the temporal convergence of 3-D solutions with large numbers of poloidal (*m*) and toroidal (*n*) modes (typically, *m ~ 6-8* was the practical limitation). It has also produced convergence problems for equilibria with low ι (<<1) where field lines must encircle the magnetic axis many times to define magnetic surfaces. The new differencing scheme computes the stream function on the same mesh as R and Z (although the output values of λ continue to be on the centered-grid for backwards compatibility), which leads to numerical stabilization of the origin interchange. To avoid first order errors (in h_s) near the plasma boundary resulting from the new representation of λ, the radial current J^s continues to be internally represented (in terms of λ) on the interlaced-grid. This maintains the good radial spatial resolution associated with the original half-grid representation for λ. As a result, computation of accurate, convergent solutions with substantially higher mode numbers is now possible using VMEC (m < 20). This corresponds to much finer spatial resolution and significantly improved force balance in the final equilibrium state. It also allows the calculation of equilibria with lower values of ι, which were difficult to obtain with the previous differencing scheme.

An additional improvement in the output from VMEC includes a recalculation (once the VMEC equilibrium has been obtained) of the magnetic force balance $\mathbf{F} \equiv \mathbf{J} \times \mathbf{B} - \nabla p = 0$. The radial ($\nabla s$) component of **F** is solved in terms of the non-vanishing contravariant components of B (B^u and B^v) and the metric elements determined by VMEC, as a magnetic differential equation for B_s. An angular collocation procedure (with grid points matched to the Nyquist spatial frequency of the modes) is used to avoid aliasing arising from nonlinear mode coupling of the Fourier harmonics of R and Z in the inverse representation of the equilibrium equation. The accurate determination of B_s, together with the higher angular resolution afforded by the larger limits on the allowable *m,n* spectra in VMEC, permits an accurate assessment for the parallel current (which contains angular derivatives of B_s) as a function of poloidal mode number, to be performed.

A.1.2. PIES

Three-dimensional magnetic fields can have magnetic islands and regions of stochastic field lines. The VMEC code uses a representation of the magnetic field that assumes nested flux surfaces. The PPPL PIES code [4] is a 3-D equilibrium code that uses a general representation for the magnetic field which is capable of calculating equilibria with islands and stochastic field line trajectories. There is an extensive set of publications on the algorithm, implementation, validation, convergence properties and applications of the PIES code (see references 9-13, 16-40 in Chapter 4 of the NCSX PVR documentation).

The PIES code solves the MHD equilibrium equations using a Picard iteration scheme,

$$\nabla \times \mathbf{B}^{n+1} = \mathbf{J}(\mathbf{B}^{n}),$$

$$\nabla \cdot \mathbf{B}^{n+1} = 0,$$

where $\mathbf{B}^{n}$ is the magnetic field at the start of the nth iteration, and $\mathbf{J}(\mathbf{B}^{n})$ is the current found from the force balance equation, $\mathbf{J} \times \mathbf{B} = \nabla p$, and the constraint $\nabla \cdot \mathbf{J} = 0$. This scheme is closely related to the Picard algorithm widely used to solve the axisymmetric Grad-Shafranov equation in the form $\Delta^* \psi_{n+1} = j_\phi(\psi_n)$. As with the Picard iteration scheme for the Grad-Shafranov equation, under-relaxation is used to extend the domain of convergence of the Picard iteration.

An advantage of the Picard scheme is that it provides an accurate calculation of resonant pressure driven currents, which are believed to play an important role in determining island widths At each iteration, the code solves for the current from the force balance equation. Writing $\mathbf{J} = \mu\mathbf{B} + \mathbf{J}_\perp$, $\mathbf{J}_\perp = \mathbf{B} \times \nabla p / B^2$ gives, $\mathbf{B} \cdot \nabla\mu = -\nabla \cdot \mathbf{J}_\perp$. Integration of this magnetic differential equation gives an accurate method for determining the currents. In implementing a numerical scheme for solving the magnetic differential equation, explicit upper bounds on the associated numerical errors were derived and are used to allow the specification of required tolerances in the code.

As the PIES code iterates, the pressure and current are flattened in islands and stochastic regions. Several numerical diagnostics in the code allow the determination of the location of these regions.

The PIES code has been validated by testing of the individual components, by internal checks in the code that monitor the accuracy with which the equilibrium equations are satisfied, and by comparison with analytic solutions and with other codes. Analytic solutions against which the code has been compared have included Solove'ev equilibria, large-aspect-ratio stellarator expansions, helical force-free Bessel function equilibria with islands, and analytic solutions for saturated tearing modes with narrow islands. Comparison of PIES with other codes has included: comparison with axisymmetric j-solver equilibria for TFTR and DIII-D; comparison with Biot-Savart vacuum field solvers; comparison with marginal stability for tearing modes calculated by a linearized resistive time-dependent code; and comparison with VMEC [5]. Ref. [5] contains a careful comparison between the VMEC code and the PIES code solutions. The devices modeled were the ATF and TJ-II stellarators, for transforms where low order rationals were absent. The flux surface shape, location of the magnetic axis and the value of ι as a function of flux surface were monitored as a function of β and radial resolution. An extrapolation in radial resolution was used to verify the quantitative agreement of the codes. The comparison with VMEC was continued in reference [6]. There, the rotational transform as a function of radius was in excellent agreement between the two codes for the W7-X stellarator, at $\langle\beta\rangle = 5\%$.

In the context of the NCSX design effort, several modifications have been made to the PIES code that have increased its speed by about an order of magnitude, allowing routine application of the code to evaluate flux surfaces in candidate NCSX configurations. The speed of the code was im-

proved by modifications to use an improved method for PIES initialization with a VMEC solution, to convert to a spline representation for field line following, and to store matrix inverses.

Compared with VMEC, the PIES code has a more time-consuming algorithm, which is needed for a general representation for the magnetic field. Time is saved by initializing PIES using a converged VMEC solution. For this purpose, the under-relaxation scheme in PIES has been modified to provide an improved coupling to the VMEC solution. This involves blending with the VMEC field in the first PIES iteration. The previous under-relaxation scheme blended the current rather than the fields. The under-relaxation was skipped in the first PIES iteration, allowing a large step from the VMEC field, and slowing the ultimate convergence. The PIES code follows magnetic field lines as a preliminary step to solving the magnetic differential equation determining the Pfirsch-Schlüter current. Conversion from a Fourier representation to a spline representation of the field has sped up the code by about a factor of two.

In each iteration of the PIES code, a discretized Ampere's law is solved by the inversion of a block-tridiagonal matrix. The elements of the blocks are determined by metric elements of a "background coordinate system" that does not change from one iteration to the next, allowing time to be saved by storing the inverses of the blocks. For high resolution calculations, this changes the scaling of the code's execution time from m^3n^3k to a much more favorable m^2n^2k where m and n are the number of the poloidal and toroidal modes retained, and k is the number of radial grid surfaces.

A.1.3. COBRA

The standard ballooning mode equation in magnetic coordinates is

$$\rho\gamma^2 \left(\mathbf{k}_\perp{}^2/B^2\right) \Phi - \mathbf{B} \bullet \nabla\left(\mathbf{k}_\perp{}^2/B^2\right)\mathbf{B} \bullet \nabla\Phi - p'/B^2 \left(\mathbf{k}_\perp \times \mathbf{B}\right) \bullet \kappa\, \Phi = 0$$

where $\mathbf{k}_\perp = \nabla\phi - q(\psi)\nabla\theta - q'(\theta - \theta_k)\nabla\psi$ with θ_k being the radial wave number. In this equation, the first term accounts for inertia, the second term corresponds to the field line bending energy, and the last term corresponds to the destabilizing drive due to bad curvature (κ) and pressure gradient (p'). The ORNL ballooning code COBRA [7] solves the ideal ballooning equation for the growth rate γ using a finite difference scheme. The ballooning mode equation then becomes a matrix equation. The computation can be done in an extremely efficient and accurate way by taking advantage of the Stürm-Lioville character of the ballooning equation. This property allows one to estimate the growth rate to 4th order on the mesh step size along the magnetic field line by variationally refining a 2nd order estimate obtained from a standard matrix method. Fast evaluation is made possible by coupling this process to a Richardson extrapolation scheme, that will extrapolate to zero mesh step size from a few previous evaluations of the growth rates computed on very coarse (and therefore very efficient to evaluate) meshes. Important speed enhancements (of hundreds of times) relative to standard codes have been achieved in this way.

A.1.4. TERPSICHORE

Stability of QPS to global finite-n external kink modes and vertical modes was analyzed using the 3D MHD stability code TERPSICHORE [8]. The code determines the eigenvalues of the ideal MHD equations by minimizing the perturbed plasma potential energy. It uses a pseudo-plasma mesh method for evaluation of the magnetic perturbation in the vacuum, treating it as a pressureless and currentless plasma that can be solved consistently with the interior plasma perturbations.

A.1.5. AVAC

The ORNL code AVAC [9] integrates magnetic field line equations in real space Cartesian coordinates. AVAC computes the magnetic field, vector potential, and their partial derivatives due to a known current distribution in vacuum and without permeable materials. It evolved from the field line orbit code FLOC [10]. Filaments of each coil center or multi-filaments for the cross-section of the coils are input into the code. From these closed polygons, the magnetic field is calculated using the Biot-Savart Law at arbitrary locations in the plasma region. The current distribution is represented by filamentary circular and/or polygonal current loops. The Cartesian components of the magnetic field are used in the field line equations to determine the location of the field line in cylindrical R,ϕ,Z space. The field lines are integrated to fixed values of the toroidal angle and these locations are saved to generate puncture plots. The code first locates the magnetic axis and then expands into the plasma region searching for magnetic field lines that stay inside the coils for a specified number of field periods. The rotational transform is calculated as the limit (for a large number of transits) of the ratio of the poloidal transits to the toroidal transits of the field line.

The current distribution is represented by filamentary circular and/or polygonal current loops. The basic magnetic quantities can be expressed in a relatively simple closed forms for these two special shapes of current loops. Therefore, calculation of field lines, guiding center orbits, etc. in such field configurations is carried out by AVAC in a rather straightforward way with high accuracy.

A.2. Neoclassical Transport Codes

A.2.1. DKES

The drift kinetic equation solver code (DKES) was developed [11,12] to calculate the full stellarator neoclassical transport coefficient matrix for realistic magnetic field spectra having arbitrary helicities. This code calculates monoenergetic transport coefficients on a single flux surface as a function of collisionality and radial electric field by using Fourier expansions in toroidal and poloidal angle variables and Legendre expansions in pitch angle. By splitting the distribution function into separate pieces that are symmetric/antisymmetric with respect to time reversal, a variational form was developed based on the entropy production rate. The utility of this formulation is that upper and lower bounds are automatically obtained for all the transport coefficients, resulting in a convenient measure of the discretization error. The convergence of the upper and lower bounds can then generally be improved by adding more terms to the Fourier/Legendre expansions. This becomes important at low collisionalities where the distribution function develops increasingly

narrow boundary layer structures in velocity space near the transitions between the various classes of trapped and passing orbits.

For a given configuration, the DKES transport matrix can be developed for a range of flux surfaces after a database of monoenergtic coefficients with dependencies on collisionality, electric field, and flux surface label is accumulated. Energy integrations must then be performed which involve multiplying by a Maxwellian times appropriate powers of velocity. For this purpose, a separate post-processing code (LIJS) is used. This code makes two-dimensional spline fits to the collisionality (ν/v) and electric field (E_r/v) dependence of the coefficients and then uses an adaptive routine to carry out the energy integration.

There are several numerical issues that currently limit the ranges of collisionality and electric field over which DKES can be applied. As mentioned above, increasing numbers of Fourier and Legendre modes must be included as the collisionality is lowered in order to converge the upper and lower bounds on the transport coefficients. At some point in this process, one encounters either memory or CPU time limits which prevent further decreases of collisionality. These limits are exacerbated at low aspect ratio due to the larger mixture of helicities in the equilibrium magnetic field which lead to requirements for larger number of modes in the representation of the distribution function. The constraint due to the CPU time limit has recently been extended by using OpenMP parallelism for certain parts of DKES; on the 4 processor per node IBM-SP Eagle system at ORNL this has resulted in a speed-up of about a factor of 3. For low aspect ratio cases without electric field, we currently find that it is generally not possible to go below the range of $\nu/v = 10^{-5}$ to 10^{-6} m^{-1}. When the electric field is finite another numerical limitation enters in; this is caused by ill-conditioning of the linear system of equations that must be solved to obtain transport coefficients. This limit is also apparently more severe for low aspect ratio devices. Evidence of this limit shows up through high values of the residuals (calculated by substituting the solution back into the original equations). This can be partially solved by solving the linear system in quad precision. However, this can slow the code down by a factor of 20-30 and generally makes it impossible to afford to be able to add sufficient modes to converge the upper and lower bounds on the transport coefficients below some level of collisionality. The above issues as well as the extension of DKES to higher dimensionality will be addressed by an ORNL Seed Money project which will reformulate the representation of the distribution function to better take into account the high gradient boundary layer regions.

A.2.2. GTC

GTC is a stellarator Monte Carlo transport code [13] developed at PPPL; this is an outgrowth of an earlier tokamak gyrokinetic turbulence code. It uses domain decomposition in the toroidal angle and MPI language calls for interprocessor communication to achieve parallelization. The transport properties of thermal ions and electrons can be studied by tracking the local particle and energy fluxes through a set of annular flux surfaces. Both δf and full-f options are available. Hamiltonian guiding center equations for each particle are advanced using a Runge-Kutta integration scheme.

As particles leave the outer flux surface they are not replaced, allowing the total number of particles to decay in time. We have used this code to benchmark confinement times derived from the DELTA5D code for isolated cases; however, the different particle replacement strategies has prevented exact comparisons.

A.2.3. DELTA5D

DELTA5D is a Monte Carlo transport simulation code that follows groups of particles on different processors in parallel using the MPI language for inter-processor communication. It has been adapted to both the Cray T3E and IBM-SP computers. DELTA5D is an outgrowth of an earlier serial Monte Carlo code [14] developed at ORNL. This solves four coupled Hamiltonian guiding center equations for each particle, advancing the particles in the two angular coordinates (poloidal and toroidal angles in Boozer coordinates) and the conjugate momenta. Equilibrium magnetic fields are obtained from the VMEC stellarator equilibrium code which are then transformed to Boozer coordinates. Collisions with a static background plasma consisting of electrons and one background ion species are simulated using a Monte Carlo collision operator based on pitch angle and energy scattering terms, taking into account the full velocity-dependent potentials. Collisions are allocated on a fixed time step Δt_{coll} which is chosen so as to maintain $\nu\Delta t_{coll} << 1$ and to allow a smooth granularity in modeling the collisional processes. The time integration step for the orbit integration is controlled by the ordinary differential equation solver LSODE which internally chooses an integration time step so as to maintain a prescribed accuracy level. As particles leave the outer flux surface, they may either be replaced back at their initial location (for thermal plasma global transport options), or, not replaced (in the case of beam and alpha slowing-down options).

DELTA5D currently includes options for following populations of thermal plasma ion and electrons, alpha particles, injected beam ions, and ICRF tails. For the case of ICRF tails, a Monte Carlo version of a quasilinear diffusion operator has been included. Several options are available for evaluating thermal plasma confinement; these include calculations of the local diffusivity of particles started at a single magnetic surface, the energy and particle loss rates through the outer magnetic surface from particles started out at a single surface, and the energy loss rates and confinement time for a distribution of particles distributed throughout the volume. Also, data on particle and energy fluxes passing through a set of annular flux surfaces is retained. In addition, there is an option to evaluate the bootstrap current. Particle loss locations can be plotted superimposed on the 3D rendered outer flux surface as a post-processing option. For the evaluation and comparison of different QPS configurations, the option of energy loss rates and confinement time for a distribution of particles distributed throughout the volume has been used most frequently.

A2.4. BOOTSJ

The bootstrap code used in the configuration optimization is based on the physics analysis and computer implementation described in Ref 15. The underlying analytical model is valid in the long-mean-free path limit. Limited benchmarking has been performed against the NIFS bootstrap code, which relies on similar physics, and Monte Carlo and DKES codes. Accurate agreement with the Monte Carlo code is difficult in the low collision frequency regime of interest due to relatively poor statistics achieved so far. Similarly, accurate DKES results in this regime are difficult to obtain because of the large number of pitch-angle modes required to resolve the sharp velocity-space boundary layers.

A.3. Coil Optimization Codes

A.3.1. NESCOIL

The Neumann Equation Solver code NESCOIL [16] is used to find the continuous current distribution that minimizes B-normal. NESCOIL uses a Green's function method to solve for a current potential on a given coil winding surface (CWS) enclosing the plasma, reproducing the prescribed plasma magnetic geometry to a high level of accuracy. To improve the targeting of both physics objectives (represented by B-normal) and engineering objectives (*i.e.*, low current density, gentle bend radii of the coils) for QPS, a singular value decomposition (SVD) technique was developed. The new code, NESVD, runs as fast as the original NESCOIL code, which enables it to be placed within an optimizer loop so that the shape of the coil winding surface can be adjusted to yield further improvements in the target criteria. A newly developed method [17] based on a genetic algorithm (GA) is available, if desired, to convert surface current potentials computed in NESCOIL or NESVD to an optimized set of discrete coils.

A.3.2. COILOPT

The COILOPT [18] code solves the stellarator magnetic coil optimization problem by determining the coefficients in an explicit representation of modular coils on a toroidal CWS, together with the coefficients describing the spatial position of the optimal winding surface. Target functions in the optimization problem include the error in the normal component of the magnetic field at the plasma edge (B-normal), the lengths of individual coils, the minimum coil radius of curvature, and the minimum separation between adjacent coils and between the coils and the plasma. The engineering (geometric) constraints are introduced in the form of weighted penalty functions. The plasma boundary shape arises from a fixed-boundary VMEC equilibrium (which results from the physics optimization process), and the part of the normal magnetic field due to currents in the plasma is computed with the BNORM code.

Modular coils are represented by a current-carrying filament (corresponding to the center of a coil cross-section) with a winding law given by a Fourier series for the toroidal and poloidal angles, ϕ_i and θ_i, of the ith coil in terms of the curve parametric t:

$$\phi_i(t) = \Sigma a_{ik}\cos(kt) + b_{ik}\sin(kt),$$
$$\theta_i(t) = t + \Sigma c_{ik}\cos(kt) + d_{ik}\sin(kt).$$

The parameter t, ($0 \le t < 2\pi$) must be specified (for example, in terms of arc-length along the coil) for this representation to be unique. The secular term in the poloidal angle corresponds to the net poloidal current carried by the modular coils (there is no net toroidal current for modular coils). All the calculations reported here were done with the unique choice $t = \theta_i$ for each coil, which precludes wind-backs in the coils where $\phi(\theta)$ is not single-valued. (Inclusion of wind-backs may be helpful to generate coils which are better able to match the desired physics and engineering properties.) It should be noted that $\theta = 2\pi u$, $N\phi = 2\pi v$, are related to the NESCOIL angles (u, v) defined on the unit square in a field period.

The coils are constrained to a CWS which is represented in cylindrical coordinates (R, ϕ,Z), with stellarator symmetry, by

$$R = \Sigma\, r_{mn} \cos(m\theta + nN\phi), \quad Z = \Sigma\, z_{mn} \sin(m\theta + nN\phi)$$

COILOPT uses a modified Levenberg-Marquardt method to solve the nonlinear least-squares problem arising from the approximation of the magnetic field at the plasma edge due to a discrete set of coil currents.

The primary goal of the coil optimization is to find a solution satisfying engineering feasibility constraints that will reproduce, or reconstruct, the targeted fixed-boundary equilibrium flux surfaces and plasma properties when the coils and currents are used to create the external magnetic field in a free-boundary VMEC equilibrium.

References

[1] S. P. Hirshman and J. C. Whitson, Phys. Fluids **26** 3553 (1983).

[2] S.P. Hirshman, D. K. Lee, Comput. Phys. Comm. **39** (1986) 143.

[3] A. Weller, C. Görner, and D. Gonda, Rev. Scientific Instruments **70**, 1 (1999).[4] A. Reiman and H. S. Greenside, Comput. Phys. Commun. **43** (1986) 157.

[5] J. L. Johnson, D. A. Monticello, A. H. Reiman, et. al., Comp. Phys. Comm. **77** (1993) 1.

[6] P. Merkel, J. L. Johnson, D. A. Monticello, et. al., Proceedings of the 14th International Conference on Plasma Physics and Controlled Nuclear Fusion Research paper IAEA-CN-60 D-P-II-10 (1994).

[7] R. Sanchez, S. P. Hirshman, J. C. Whitson, and A. S. Ware, J. Comput. Physics **161**, 576 (2000).

[8] D. V. Anderson, W. A. Cooper, R. Gruber, S. Merazzi, and U. Schwenn, Scient. Comp. Supercomputer II, 159 (1990).

[9] D. K. Lee, Comput. Phys. Comm. 25, 181 (1982).

[10] R.H. Fowler, D.K. Lee, P.W. Gaffney, and J.A. Rome, ORNL/TM-6293 (1978).

[11] W. I. van Rij, S. P. Hirshman, Phys. Fluids B, **1** (March, 1989).
[12] S. P. Hirshman, K. C. Shaing, et al., Phys. Fluids, **29** (September, 1986).
[13] Z. Lin, et al., Phys. Plasmas **2**, 2975 (1995).
[14] R.H. Fowler, J.A. Rome, J.F. Lyon, Phys. Fluids **28**, 338 (1985).
[15] K. C. Shaing,, E.C. Crume, Jr., J.S. Tolliver, S.P. Hirshman, W.I. van Rij, Phys. Fluids B1, 148 (1989).
[16] Merkel, P., Nuclear Fusion **27** (1987).
[17] Miner, W.H.,Jr., et.al., "Use of a genetic algorithm for compact Stellarator coil design," to appear in Nuclear Fusion.
[18] Strickler, D.J., Berry, L.A., Hirshman, S.P., "Optimization of coils for compact Stellarators," (in preparation).