WHISTEL: A Stellarator Transport Code

G.A. Emmert, K.J. Lee, and A. Bailey

September 1982

UWFDM-477

FUSION TECHNOLOGY INSTITUTE

UNIVERSITY OF WISCONSIN

MADISON WISCONSIN

WHISTEL: A Stellarator Transport Code

G.A. Emmert, K.J. Lee, and A. Bailey

Fusion Technology Institute
University of Wisconsin
1500 Engineering Drive
Madison, WI 53706

http://fti.neep.wisc.edu

September 1982

UWFDM-477

WHISTEL: A STELLARATOR TRANSPORT CODE

A.W. Bailey

G.A. Emmert

K.J. Lee

Fusion Engineering Program
Nuclear Engineering Department
University of Wisconsin-Madison
Madison, Wisconsin 53706

October 1982

UWFDM-477

Abstract

WHISTEL is a version of the ORNL WHIST tokamak transport code adapted to model transport in stellarators. This document is intended as a manual for those wishing to utilize the code or interpret its output. Only the alterations to the original code are described in detail; for information on the unaltered portions of the code, see WHIST code documentation. The WHISTEL code is available on the National Magnetic Fusion Energy Computing Center CDC-7600. The code is written in FORTRAN IV and should be readily transportable.

I. Introduction

The WHISTEL code is a stellarator transport code, an adaptation of the ORNL WHIST version 34 tokamak transport code (hence the name, WHIST for STELlarators). The code was adapted to model stellarators, torsatrons, heliotrons, etc., by allowing for an externally applied rotational transform and by alteration of the diffusion and conductivity transport coefficients to properly reflect transport in stellarators. In addition, a variety of other more minor changes have been made. A feedback routine to set a desired beta limit has been added to allow determination of steady-state plasma parameters in thermally unstable regimes. The contribution of hot fusion born ions to the pressure and losses of these ions due to trapping in helical ripple wells are now included. The code has been adapted to allow for continuous alteration of the mesh spacing from uniform to more closely spaced near the separatrix. The divertor routine has been adapted to model the intrinsic divertor in both helical and modular coil machines. Provision has been made for modeling ICRH as well as ECRH heating.

An attempt has been made to keep the stellarator version of the code as much like the original tokamak code as is feasible in order to simplify its use by those familiar with the WHIST code, and to allow for ease of transfer of the new stellarator routines to updated versions of the WHIST code. For more information on the WHIST code, see Refs. (1-6).

The WHISTEL code was developed as part of the UWTOR-M modular stellarator reactor design project.[7-9] The code is presently configured to run under ORDERLIB on the National Magnetic Fusion Energy Computing Center CDC-7600 but is written in FORTRAN IV and should be readily transportable elsewhere. For more information on access to the code, contact the authors.

II. Transport Model

Given the initial temperature and density profiles, the temperatures and densities at future times are computed by solving the following transport equations.

$$\frac{\partial n_j}{\partial t} = -\frac{1}{r}\frac{d}{dr}(r\Gamma_{\perp j}) + S_{FUS_j} + S_{BEAM_j} + S_{GAS_j} + S_{PEL_j} - \vec{\nabla}_{\parallel}\cdot\vec{\Gamma}_{\parallel j}$$

$$\frac{\partial}{\partial t}\left(\frac{3}{2} n_e T_e\right) = -\frac{1}{r}\frac{d}{dr}\left[r\left(Q_{\perp e} + \frac{3}{2} T_e \Gamma_{\perp e}\right)\right] - Q_{ei} + P_{FUS_e} - P_{RAD} + P_{OHM}$$

$$+ P_{RF_e} + P_{BEAM_e} - P_{NEUT_e} - \vec{\nabla}_{\parallel}\cdot\vec{Q}_{\parallel e}$$

$$\frac{\partial}{\partial t}\left(\frac{3}{2} n_i T_i\right) = -\frac{1}{r}\frac{d}{dr}\left[r\left(Q_{\perp i} + \sum_j \frac{3}{2} T_i \Gamma_{\perp j}\right)\right] + Q_{ei} + P_{FUS_i} + P_{RF_i}$$

$$+ P_{BEAM_i} - P_{NEUT_i} - \vec{\nabla}_{\parallel}\cdot\vec{Q}_{\parallel i}$$

where:

n_j = density of j^{th} ion species

n_i = total ion density

n_e = electron density

T_e, T_i = electron and ion temperatures

S_{FUS_j} = source or sink of ions of species j due to fusion

S_{BEAM_j} = source of ions of species j due to beam injection

S_{GAS_j} = source of ions of species j due to gas puffing

S_{PEL_j} = source of ions of species j from pellet fueling

Q_e, Q_i = electron and ion energy fluxes perpendicular to the magnetic flux surfaces

Q_{ei} = electron-ion re-thermalization

P_{FUS_e}, P_{FUS_i} = power to the electrons and ions due to fusion

P_{RAD} = power lost from the electrons due to radiation

P_{OHM} = power to the electrons from ohmic heating

P_{RF_e}, P_{RF_i} = power to the electrons and ions due to rf heating

P_{BEAM_e}, P_{BEAM_i} = power to the electrons and ions due to neutral beam heating

P_{NEUT_e}, P_{NEUT_i} = power lost from the electrons and ions due to ionization and charge exchange of neutrals

$\Gamma_{\perp j}$ = net flux of ion species j perpendicular to the magnetic flux surfaces

$\Gamma_{\parallel j}$ = net flux of ion species j along field lines in the divertor zone

$Q_{\perp e}, Q_{\perp i}$ = net electron and ion conduction energy fluxes perpendicular to the magnetic flux surfaces

$Q_{\parallel e}, Q_{\parallel i}$ = net electron and ion conduction energy fluxes along field lines in the divertor zone.

Up to five ion species may be included in the simulation, allowing modeling of catalytic D-D reactors as well as D-T reactors. The electron density is computed as the ion density times the charge state of each ion, summed over all ion species. Electrons due to impurity species are included in the computation. The perpendicular loss terms are calculated for the transport coefficients such that

$$\Gamma_{\perp j} = - \sum_j D_{jj'} \frac{\partial n_{j'}}{\partial r} - D_{ji} \frac{\partial T_i}{\partial r} - D_{je} \frac{\partial T_e}{\partial r}$$

$$Q_{\perp i} = - \sum_j K_{ij} \frac{\partial n_j}{\partial r} - K_{ii} \frac{\partial T_i}{\partial r} - K_{ie} \frac{\partial T_e}{\partial r}$$

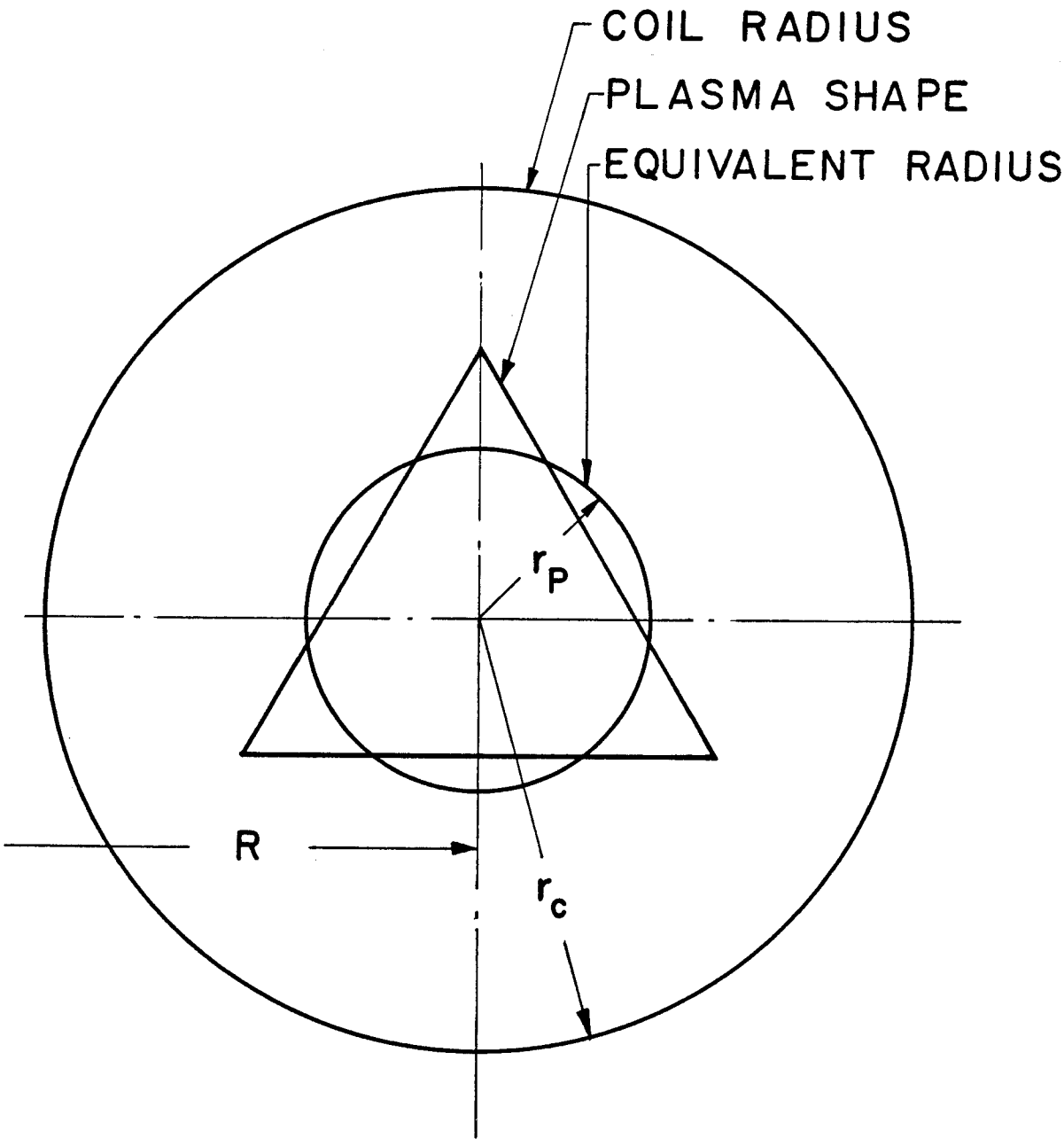
$$Q_{\perp e} = - \sum_j K_{ej} \frac{\partial n_j}{\partial r} - K_{ei} \frac{\partial T_i}{\partial r} - K_{ee} \frac{\partial T_e}{\partial r}$$

where $D_{jj} = D$, $K_{ii} = \chi_i n_i$, and $K_{ee} = \chi_{ee} n_e$. There is thus provision not only for a diagonal model in which particle diffusion is on their own density gradient, electron heat conduction is on the electron temperature gradient; and ion heat conduction is on the ion temperature gradient, but there is also allowance for particle diffusion and heat conduction on other gradients. Even the diffusion of species j on the gradient in the density of species j' is allowed. It is necessary, therefore, to solve a set of simultaneous partial differential equations that are first order in time and second order in space. These equations are solved with finite differences utilizing a modified Crank-Nicholson scheme, made more implicit (future-weighted) than the usual time-centered formulation. Nonlinear effects can cause the time-centered Crank-Nicholson scheme to become numerically unstable with reasonably-sized time steps; slightly increasing the implicitness introduces some dissipation and makes the scheme stable, only slightly decreasing the accuracy.

The noncircular flux surfaces of stellarators are modeled as equivalent circles. The mapping is such as to preserve volume: the areas of the actual and equivalent tubes of flux are the same (Fig. 1). This mapping produces some underestimate of the areas of the flux surface and thus of the transport.

The spatial finite differences are established by setting up a radial grid across the minor radius of the plasma. The mesh can be chosen to be increasingly fine near the separatrix so as to more accurately model the stronger temperature and density gradients near the boundary of the plasma. A fine uniform mesh is used in the scrape-off zone. The densities, temperatures, and currents are computed at each grid point, "R". Additional inter-

Fig. 1. Actual plasma flux tubes and equivalent-volume circular flux tubes used by the WHISTEL code.



R = MAJOR RADIUS

A_c = COIL ASPECT RATIO = R / r_c

A_P = PLASMA ASPECT RATIO = R/r_P

mediate, "RMID", mesh points halfway between the main gridpoints are utilized as well. The fluxes, diffusion coefficients, magnetic fields, and rotational transforms are computed at these intermediate grid points (Fig. 2). This procedure preserves the accuracy of the central-differencing scheme.

The time step is chosen so as not to exceed 1/10 of the characteristic time rate of change of the temperature, densities, and magnetic field strengths at any grid point. A very small time step is used for the initial step and after any major changes in the plasma profile due to disruptions or pellet injection.

The energy and particle sources and sinks from fusion are computed in such a way as to explicitly account for all reaction paths (e.g., D-D and T-T as well as D-T fusion are considered). The energy from fusion is apportioned by formula to both the ions and electrons.[10] A provision has been made for the loss of hot fusion ions in unconfined orbits to the divertors, a loss fraction of $\sqrt{2r/R_0}$ being assumed, where r is the minor radius of the flux surface and R_0 is the major radius. Ions that do not escape are assumed to thermalize on the flux surface on which they are born.

The sources due to beams are computed using an analytic pencil-beam routine that compares well with tokamak Monte-Carlo beam routines but is far faster.[11] The routine does not account for the loss of primary beam ions lost by trapping in helical ripple wells. This can be a serious loss in off-tangential beam injection.[12] Beam-plasma fusion is included in the simulation.

Sources from pellets and puff gas are also included.[13,14] Pellet fueling may be modeled using discrete pellets or a "continuous pellet" model in which the source is continuous rather than occuring at a single time step

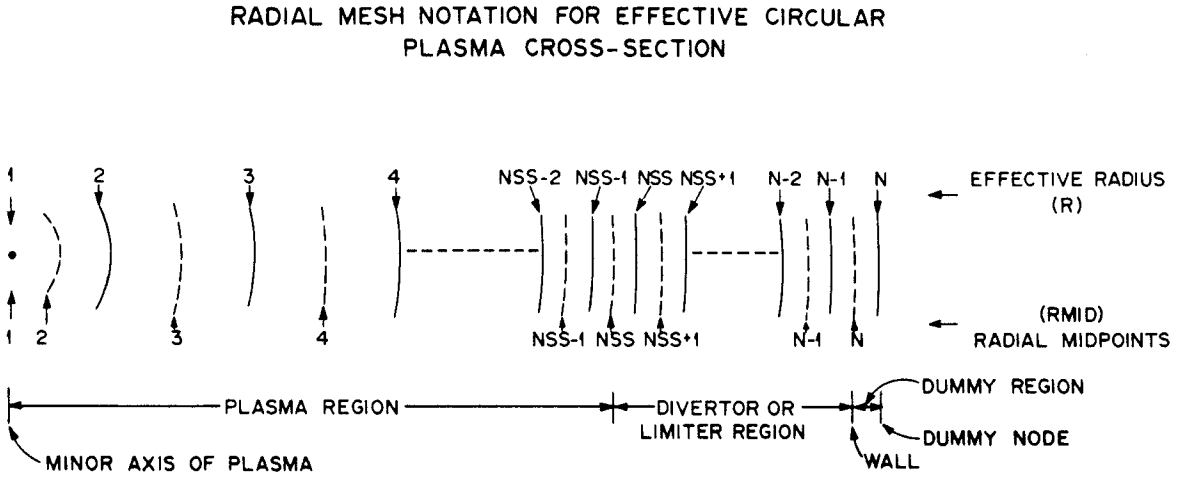


Fig. 2. Nonuniform mesh used in the WHISTEL code. (From Ref. 4. Used with permission of A.T. Mense.)

for each pellet. This model allows the code to run faster by ignoring short-term fluctuations and is accurate if the interval between pellets is short compared to the confinement times. Gas puffing is modeled using the one-dimensional slab neutral transport routine SPUDNUT.[13] A feedback routine is included to maintain a desired density by either pellets or gas puffing. An incomplete provision has been made for the recycling of neutral gas from divertors or limiters. This recycling routine conflicts with the continuous pellet feedback routine; they cannot be used together. The energy to the plasma from the recycling of helium is ignored.

RF heating powers are computed using a simple model which distributes the energy in a gaussian profile in a vertical slab imposed on the plasma cross section. Either ECRH or ICRH heating is allowed. For ICRH, a simple linear model is utilized in which a fraction T_e (keV)/13.5 of the energy is assumed

to go to the electrons, the rest to go to the ions. All the ECRH energy is assumed to go to the electrons.

The ohmic heating is computed for an imposed current from the Spitzer resistivity, in which neoclassical corrections may be included. The radiation loss terms include bremsstrahlung, synchrotron, and impurity radiation. The impurity radiation output is taken from data by D.E. Post et al. and includes most, but not all elements of interest.[15] Radiative energy transfer within the plasma is not considered.

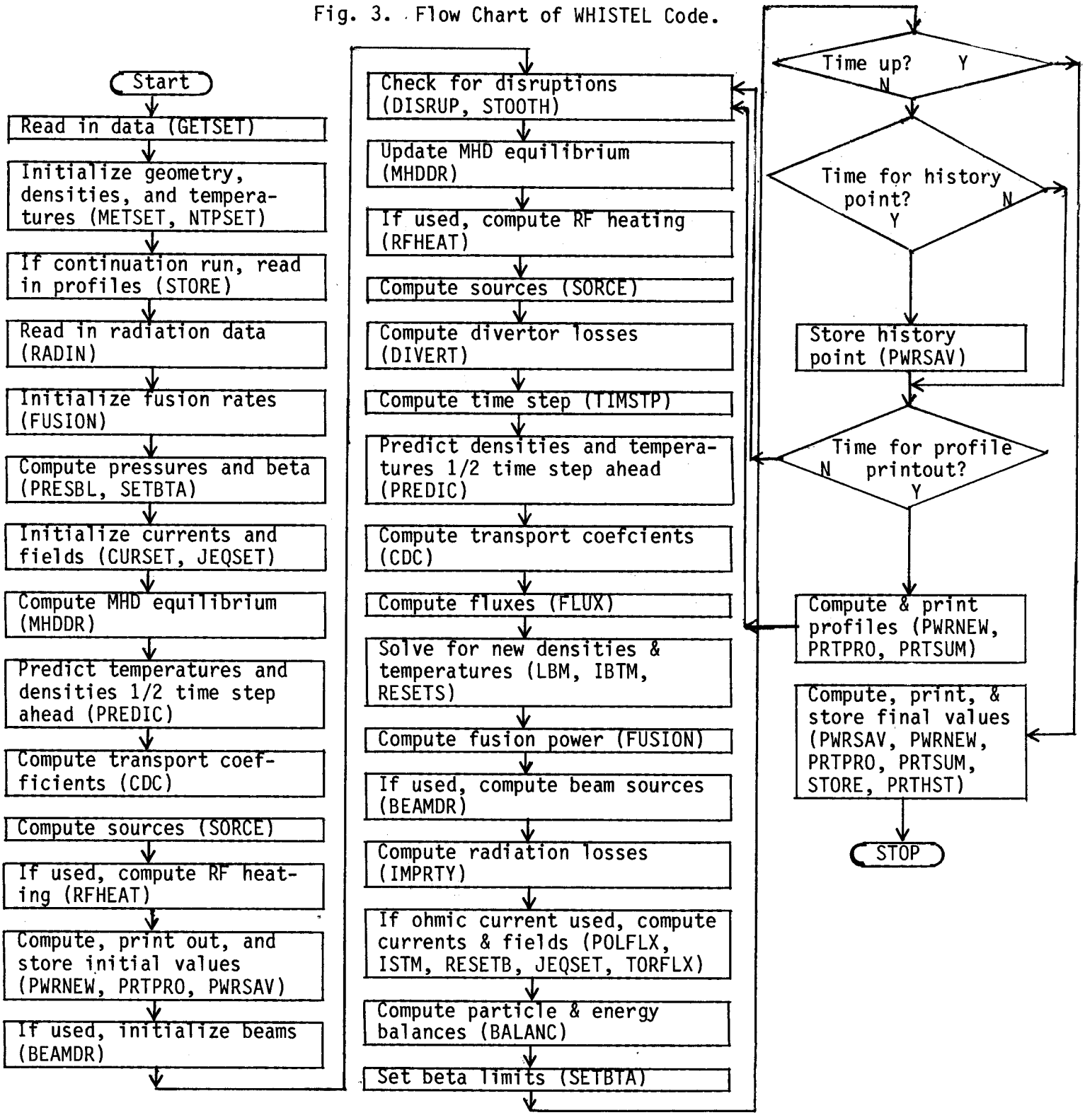
III. Architecture of the Code

A. Main Iteration Loop

The code is organized around a loop in the subroutine AAMAIN, which is traversed once each time step. As shown in Fig. 3, the program first reads in the input data in subroutine GETSET, which calls METSET, NTPSET, STORE, RADIN, IMPRTY, FUSION, PRESBL, SETBTA, CURSET, JEQSET, MHDDR, PREDIC, CDC, and SORCE to set the initial values for the geometrical parameters, densities and temperatures, radiation, fusion rates, pressures, currents, fields, transport coefficients and particle sources. If RF heating is to be used initially, then RFHEAT is then called by AAMAIN. The global values are computed by PWRNEW, the initial profiles printed out by PRTPRO, and the values saved by PWRSAV. If beams are to be used, BEAMDR is called.

The main simulation loop then begins. DISRUP and STOOTH are called to check for disruptions. MHDDR is called to update the Shafranov shift and re-compute the fields. If RF heating is desired then the RFHEAT routine is called to compute the power. The sources and sinks of particles and their energies are computed in SORCE and DIVERT. From the rates of change of the densities, temperatures, and flux changes, the length of the next time step is

Fig. 3. Flow Chart of WHISTEL Code.



computed in TIMSTP. PREDIC then extrapolates the densities and temperature profiles 1/2 time step, the new transport coefficients being computed using these values in CDC. These transport coefficients are then used to calculate the particle and energy fluxes in FLUX. The new values for the temperatures and densities at the next time step are those computed by the Crank-Nicholson method. This involves solving a matrix. The matrix values are loaded in LBM, the matrix inverted by IBTM, and the values reset in RESETS. FUSION, BEAMDR, IMPRTY, and PRESBL are called to compute the fusion power, beam deposition, radiation, and pressure. If there is a current imposed on the plasma POLFLX is called to load the matrix for recomputation of the poloidal magnetic field, ISTM inverts the matrix, and RESETB and JEQSET are then called to reset the fields, currents, and transforms. Finally TORFLX is called to recompute the toroidal flux and poloidal current, BALANC is called to compute the particle and energy balances, and SETBTA is called to compute the plasma beta and apply feedback as needed to maintain it.

This ends the main simulation loop. PWRSAV, PWRNEW, PRTPRO, PRTSUM, and STORE are called to store and print data at desired times. The loop is continued until the desired amount of simulation time has elapsed, the maximum number of iterations has been performed, or the maximum allotted amount of computer time has been used. A listing of plasma parameters is then printed out by PRTHST and execution is terminated, leaving a dropfile which may be restarted if desired. A restarted file is given 9 more minutes of computer time.

B. Helix Common Block

In order to keep the stellarator-dependent portions of the WHISTEL code transportable to updated versions of the WHIST code, the WHIST common blocks

have not been altered. One new common block, dubbed HELIX, has been added. The variables therein are described in this section.

BRIOTA -- The magnetic rotational transform, iota-bar, at each zone boundary.

BVAC -- The vacuum poloidal magnetic field, i.e. the field with no toroidal current, at each zone boundary.

LNUM -- The poloidal harmonic number of the stellarator or torsatron.

NFP -- The number of field periods for the machine.

ROTIN -- Used to input the vacuum rotational transform. The transform is taken to be

$$\text{Vacuum Transform} = \text{ROTIN}(1) + \text{ROTIN}(2)*(r/a) + \text{ROTIN}(3)*(r/a)^2 + \text{ROTIN}(4)*(r/a)^3 + \text{ROTIN}(5)*(r/a)^4$$

where "a" is the minor radius of the separatrix.

FALPHA -- The fraction of the hot fusion ions in each radial zone which do not escape to the divertors before slowing down.

PRESF -- The pressure in each zone due to hot fusion ions which are slowing down.

BETAT -- The toroidal beta in each zone.

KALPHA -- A switch to suppress the direct losses of the hot fusion ions. If KALPHA=0, FALPHA=1.0 in all zones.

BETA -- The volume averaged toroidal beta in the plasma.

BTAMAX -- The maximum desired average toroidal beta in the machine. This value is maintained by the routine SETBTA when KBETA > 0.

KBETA -- A switch telling the routine SETBTA how to fix the beta limit. If KBETA=0, no beta limit is set. If KBETA=1, the subroutine HOMOGE is called which homogenizes the inner zones of the plasma until transport

losses stabilize the beta. This is intended to simulate the natural beta limits of a device. If KBETA=2, gas puffing is used to control the beta. Increased gas puffing increases the charge exchange and ionization energy losses. If KBETA=3, the magnetic ripple strength is fed back to control the beta through ripple transport of energy. If KBETA=4, the density of impurity #1 is used to control the beta.

NBETA -- The number of radial zones to be homogenized to control the beta.

TBETA -- The feedback response time of the beta setting routine SETBTA.

RADLN -- The energy lost by radiation due to impurity species #1.

TAUGLB -- The global energy confinement time, calculated in subroutine GLOBAL.

XRIP -- Multiplier to the ripple strength for beta setting feedback on the magnetic field ripple.

KCUR -- Switch to allow for the ohmic heating current in the plasma to be turned on or off.

C. Subroutines

The following subroutines were written or modified in the conversion of the WHIST code to model stellarators.

AAMAIN -- The main routine containing the main iteration loop which is traversed once each time step. The routine has been modified, moving the initial calls to FUSION and SORCE to within GETSET in order to properly account for the hot alpha pressure.

DIVERT -- Models the divertors or limiters. The subroutine has been modified to model the intrinsic divertor action of stellarators and torsatrons. In a tokamak with a single poloidal divertor, a field line makes a number of toroidal transits around the torus equal to the safety factor between entrances to the divertor. In a stellarator or torsatron the field lines

enter a divertor each toroidal transit with a frequency equal to the number of field periods minus the edge rotational transform, i.e. the effect is like having that number of aperture limiters. This implies that the parallel flow distance is typically an order of magnitude less than in a tokamak, giving a much more efficient divertor action. The routine has been modified to reflect this shorter flow length. The input option KSO=1 models the helical divertors of continuous-coil machines. KSO=2 models the helical bundle divertors of Rehker-Wobig-type modular coils,[16] using the bundle divertor model of Emmert.[17] If there were a constant electric potential along a field line, only (1/divertor upstream mirror ratio) of the ions encountering the divertor coil would escape to the divertors: the rest would mirror. The drop in the electric potential in a well-pumped divertor allows for greater losses, thus a fraction (1.2/divertor upstream mirror ratio) of the ions are assumed to escape carrying (1.6/divertor upstream mirror ratio) of the energy. The divertor upstream and downstream mirror ratios are taken to be equal to the poloidal harmonic number, LNUM. Provision is also made for modeling a single rail limiter (KSO=3) or a single full toroidal limiter (KSO=4), where the parallel flow lengths are the same as they would be in a tokamak. In all cases, ions are assumed to flow to the divertors at the ion thermal velocity. Electron energy losses are calculated by requiring a target sheath potential drop sufficient to assure ambipolar flow to the divertor target plate and calculating the electron energy loss as that of the electrons with energy sufficient to overcome the potential drop, where a Maxwellian electron distribution is assumed.

FUSION -- Computes the energy deposition and the particle source and loss rates due to fusion reactions. The routine can handle all the reaction involving D, T, and He^3. The routine has been modified to include the pressure due to fusion-born ions slowing down (copying the formulation in the more recent versions of the WHIST code) and to allow for the loss of hot ions lost due to trapping in helical ripple wells.

PRTHST -- Prints out a history of the plasma parameters at the end of a run. A printout of the total D-T power output replaces that of the toroidal voltage. A printout of the total plasma volume has also been added. The routine also writes out the data for the plotting routine. A three-dimensional plot of the safety factor has been replaced by a three-dimensional plot of the average toroidal beta.

PRTPRO -- Print out the plasma profiles. Printouts of the toroidal voltage and safety factor have been replaced by printouts of the toroidal beta and the rotational transform, iota-bar.

PWRSAV -- Saves the values to be printed out in the history by PRTHST. The routine calls GLOBAL to compute and save the global confinement times.

RFHEAT -- Computes the RF energy deposition. RF power is deposited in a gaussian-shaped profile in a vertical slab superimposed on the plasma cross-section. The code has been modified to model ICRH as well as ECRH. When ECRH is used (KRF=1), all the energy is presumed to go to the electrons. When ICRH is used (KRF=2), a fraction $T_e/13.5$ keV of the energy is presumed to go to the electrons, the rest to the ions.

SORCE -- Models particle and energy sources and sinks due to gas puffing and pellet fueling. Gas puffing is computed by the slab neutral transport routine SPUDNUT.[13] The pellet deposition is computed by the routine

PELLET, normally using the ablation model by Milora and Foster.[14,18] The time over which a pellet deposits particles is much less than the perpendicular diffusive times and thus is modeled as an abrupt change in the plasma density. Provision is also made for a "continuous pellet" model in which the particle source due to pellet fueling is averaged out. This allows the code to ignore the oscillations in the plasma profiles due to pellet fueling and allows large time steps to be taken. This approximation is valid if the time between pellets is short compared to the confinement times. Provision has also been made of the recycle of neutral gas from the divertors. The recycling routine interferes with feedback using the continuous pellet model, so both cannot be utilized simultaneously. The deposition of recycled gas is computed using SPUDNUT for the hydrogenic species and treating the helium as a beam with a injection energy EHE. The energy to the plasma for recycled helium was previously treated as RF power to the plasma, but with the new RF routine it is now simply ignored.

GLOBAL -- Computes the global confinement times. The HELIX common block has been added so that the global energy confinement time TAUGLB may be available to SETBTA.

AACDC -- Prints out the date of last revision of the CDC routines.

CDC -- Computes the transport coefficients. The routine calls individual subroutines PCLASS, NCLASS, FULNEO, BOHM, ALCAT, W7A, RTMODE, BDMODE to compute transport due to pseudoclassical transport, diagonal neoclassical transport ($\Gamma_{\perp j} = -D \frac{\partial n_j}{\partial r}$, $Q_{\perp i} = -n_i \chi_i \frac{\partial T_i}{\partial r}$, and $Q_{\perp e} = -n_e \chi_e \frac{\partial T_e}{\partial r}$),[19] Hinton-Moore full neoclassical transport,[20] Bohm diffusion (in the scrape-off layer), Alcator scaling, reduced Alcator scaling (this routine replaces

the PLT scaling routine), ripple trapping (revised to model stellarator ripple), and banana drift diffusion (now unused and inappropriate for stellarators, but still physically present in the code). The individual subroutines generally call MAXTDC to load the transport coefficients into memory to record which mechanism (IRTDC) is predominantly responsible for transport.

RTMODE -- Models transport due to ripple trapping. This routine has been totally rewritten. The transport due to helical ripple is that of the model by Connor and Hastie[21] as amended by Shaing and Callen.[22] The latter have found that as very low levels of collisionality are approached the transport ceases to rise and finally slowly falls. To reflect this the ion-ion collision frequency, ν^*_{ii}, is taken to have an effective minimum value of

$$88\ \varepsilon_h \left(\frac{T_i}{eBR_o}\right) \frac{1}{r}$$

and the electron-ion collision frequency, ν^*_{ei}, an effective minimum value of

$$32\ \varepsilon_h \left(\frac{T_i}{eBR_o}\right) \frac{1}{r}$$

where ε_h is the peak to average helical field ripple. The ion heat diffusivity is taken to be

$$\chi_i = \frac{46.5\ \varepsilon_h^{3/2}}{\nu^*_{ii\ eff}} \left(\frac{T_i}{eBR_o}\right)^2$$

and the diffusion coefficients are taken to be

$$D_{jj} = 4.34 \frac{\varepsilon_h^{3/2}}{\nu^*_{ei\ eff}} \left(\frac{T_e}{eBR_o}\right)^2 \left(1 + \frac{T_i}{T_e}\right)$$

$$D_{je} = 4.34 \frac{\varepsilon_h^{3/2}}{\nu^*_{ei\ eff}} \left(\frac{T_e}{eBR_o}\right)^2 3.37 \frac{n_e}{T_e}$$

$$D_{ji} = 4.34 \frac{\varepsilon_h^{3/2}}{\nu^*_{ei\ eff}} \left(\frac{T_e}{eBR_o}\right)^2 3.45 \frac{n_i}{T_i} .$$

W7A -- Models reduced Alcator-type scaling. This routine (replacing subroutine PLT) reflects the decrease in anomalous transport observed in stellarators when current-free operation is approached. When no ohmic heating current is used, 1/5 the Alcator scaling transport is computed. With an ohmic heating current, the added anomalous transport is assumed proportional to the current such that full Alcator scaling would result if all the rotational transform were due to the current. That is

$$\chi_e = \left(\frac{0.8\ ZJ}{ZJTOK} + 0.2\right) \frac{1.0 \times 10^{17}}{n_e(cm^{-3})} \frac{cm^2}{sec}$$

$$D = \left(\frac{0.8\ ZJ}{ZJTOK} + 0.2\right) \frac{2.0 \times 10^{16}}{n_e(cm^{-3})} \frac{cm^2}{sec}$$

where ZJ is the toroidal current density and ZJTOK is the current that would be needed to produce the same rotational transform in a tokamak, i.e. if there were no externally applied transform.

CURSET -- CURSET initializes the current profiles, the poloidal magnetic field, and the rotational transform. The routine has been altered to reflect the externally imposed rotational transform of stellarators.

DISRUP -- Models the double-tearing-mode disruption. When the input option KDIS is set equal to 1, a hollow current profile is eliminated by redistributing the current, density, and energy uniformly over the hollow profile region. The routine has been altered so that the new routine HOMOGE does the redistribution.

GETSET -- Reads in the input data, checks it for consistency, and writes it into the output file. GETSET then initializes the program for a fresh start, setting up the grid mesh and calling METSET, NTPSET, STORE, RADIN, IMPRTY, FUSION, PRESBL, SETBTA, CURSET, JEQSET, MHDDR, PREDIC, CDC, and SORCE to compute initial values for the plasma variables. Previously, a choice was available between a uniform and 1/r grid spacing. Now a continuum of choice is offered with the mesh spacing proportional to $(1/r)^{SDR}$. For large values of field ripple the gradients near the separatrix are very large and a value of SDR between 1.0 and 1.5 is useful, giving a very fine mesh in this region.

HOMOGE -- Redistributes the current, density, and energy in the central region of a plasma. This subroutine is called by DISRUP, SETBTA, and STOOTH to simulate the double-tearing-mode disruption, the breakdown in confinement due to approaching the beta limits, or to sawtooth disruptions. This routine was previously a part of the DISRUP and STOOTH subroutines, but has now been made into a separate subroutine.

IMPRTY -- Calculates the energy losses due to radiation. Bremsstrahlung, synchrotron, and impurity radiation losses are included. The impurity losses are calculated in the routine RADI. The routine IMPTAR to calculate the diffusion of impurities has not been implemented; the impurity density is taken to be flat. The routine now separately computes the

global radiation losses, RADLN, due to impurity species #1, this quantity being used by SETBTA when the beta limits are set by feedback on the impurity concentration.

JEQSET -- Computes the rotational transform, safety factor, total current, effective currents, and electric fields in the plasma. This routine has been altered to account for the imposed rotational transform in stellarators.

METSET -- Initializes the geometric parameters of the plasma. The routine has not been altered to properly model the shapes of stellarator flux surfaces: circular flux tubes are assumed. The routine now also calculates FALPHA, the fraction of hot fusion ions not lost due to trapping in helical ripple wells.

MHDDR -- The driving routine for updating the MHD equilibrium. The routine calls MHDEQ to recompute the equilibrium if the pressure or transform profiles have altered appreciably. At present only the Shafranov shift of circular flux surfaces is modeled.

MHDEQ -- Computes the MHD equilibrium. At present no true MHD-solving routine is in place and only a crude eyeballed fit to ℓ=2 Chodura code results is used for the Shafranov shift and the rotational transform.[23]

POLFLX -- Loads a matrix for calculation of the poloidal field from Maxwell's equations. The matrix is inverted by ISTM, and the values reset in RESETB. The routine has been altered to accept an externally applied rotational transform as well as that due to currents in the plasma.

PRESBL -- Calculates the plasma pressure and beta profiles. The routine now also considers the pressure due to hot fusion-born ions slowing down, computed in FUSION.

RADI -- Computes the radiated power per unit impurity density using approximate fits to data compiled by D.E. Post et al.[15] stored in the file CORONA. The routine has been altered to smoothly trail off the radiation at low temperature rather than simply set it to zero below a certain point. This prevents temperature oscillation about the cutoff temperature at low scrape-off zone densities.

RESETB -- Resets the magnetic fields after ISTM has inverted the matrix set in POLFLX. The routine has been altered to accept an externally imposed rotational transform.

SETBTA -- Calculates the average toroidal beta and supplies feedback to contain the beta within desired limits. The routine models the feedback as critically damped harmonic motion. That is

$$\frac{d^2(\beta - \beta_{MAX})}{dt^2} + \frac{2}{t_{BETA}} \frac{d(\beta - \beta_{MAX})}{dt} + \frac{(\beta - \beta_{MAX})}{t_{BETA}^2} = 0$$

where t_{BETA} is the beta limit feedback response time and β_{MAX} is the desired beta limit. The rate of change of the energy in the plasma $d(\beta - \beta_{MAX})/dt$ needed is computed to solve the equation. Feedback is produced by homogenizing the inner zones of the plasma to model breakdown of the inner flux surfaces by calling HOMOGE (option KBETA=1), gas puffing to cool the plasma (KBETA=2), altering the ripple strength to cool the plasma (KBETA=3), or controlling the impurity density (KBETA=4). Setting KBETA=0 leaves the beta setting routine off.

STOOTH -- Uses Kadomtsev's model for sawtooth disruption,[24,25] homogenizing the inner zones of the plasma when the safety factor on axis falls below

the critical value QSO. The homogenization was previously calculated in the routine itself but is now implemented by calling the routine HOMOGE.

TORFLX -- Computes the toroidal flux (poloidal current) from the pressure-balance equaton. The routine has been adapted to include the magnetic pressure from an externally imposed rotational transform.

IV. Plotting Routine

The WHIST plotting routine WHPLOT has been adapted for the WHISTEL code and dubbed STELPLOT. WHPLOT uses the DISSPLA[26] graphics library to produce a large number of two-dimensional plots, 4 to a page. It also produces six three-dimensional plots of electron temperature, ion temperature, electron density, log of the source rate, safety factor, and current density, all as functions of time and plasma minor radius. In STELPLOT, the safety factor plot has been replaced by a plot of the toroidal beta, and the current density plot is not produced when the current is zero to avoid numerical overflow errors.

V. Acknowledgments

The authors would like to thank W.A. Houlberg and S.E. Attenberger for making the WHIST code available to us and for their helpful advice in converting it to a stellarator code. We would also like to thank D.T. Anderson, J.A. Derr, and J.L. Shohet for their advice on stellarator physics, J.E. Scharer for his advice on ICRH heating, and J.D. Callen and K.C. Shaing for their advice on ripple transport and other matters.

Support for this work was provided by the U.S. Department of Energy.

References

1. W.A. Houlberg and R.W. Conn, Nucl. Sci. Engr. 64 (1977) 141. Also available as UWFDM-201 (1977).

2. W.A. Houlberg, S.E. Attenberger, and A.T. Mense, Nucl. Fusion 20 (1980) 811. Also available as ORNL/TM-6730 (1979).

3. W.A. Houlberg, A.T. Mense, S.E. Attenberger, ORNL/TM-6730 (1979).

4. A.T. Mense, W.A. Houlberg, S.E. Attenberger, ORNL/TM-6841 (1980).

5. W.A. Houlberg, H.C. Howe, S.E. Attenberger, ORNL/TM-7124 (1980).

6. W.A. Houlberg, S.E. Attenberger, L.L. Lau, ORNL/TM-8193 (1982).

7. I.N. Sviatoslavsky et al., Proc. of the Ninth International Conference on Plasma Physics and Controlled Nuclear Fusion Research, IAEA-CN-41/E-3, Vienna (1982). Also UWFDM-480 (1982).

8. I.N. Sviatoslavsky et al., IEEE Trans. Plasma Sci. PS-9 (1981) 163.

9. B. Badger et al., UWFDM-550 (1982).

10. W.A. Houlberg, UWFDM-103 (1974).

11. R.M. Wieland, W.A. Houlberg, and A.T. Mense, ORNL/TM-6550 (1979).

12. D.T. Anderson, J.L. Shohet, S. Rehker, and J.A. Tataronis, Nucl. Fusion 20 (1980) 197.

13. K. Audenaerde, G.A. Emmert, and M. Gordinier, J. Comp. Phys. 343 (1980) 238. Also available as UWFDM-259 (1978).

14. W.A. Houlberg, M.A. Iskra, H.C. Howe, and S.E. Attenberger, ORNL/TM-6549 (1979).

15. D.E. Post, R.V. Jensen, C.B. Tarter, W.H. Grasberger, and W.A. Lokke, Atomic Data and Nuclear Tables 20 (1977) 397.

16. J.A. Derr and J.L. Shohet, IEEE Trans. Plasma Sci. PS-9 (1981) 234.

17. G.A. Emmert, UWFDM-343 (1980).

18. S.L. Milora and C.A. Foster, IEEE Trans. Plasma Sci. PS-6 (1978) 578.

19. F.L. Hinton and R.D. Hazeltine, Rev. of Mod. Phys. 48 (1976) 239.

20. F.L. Hinton and T.B. Moore, Nucl. Fusion 14 (1974) 639.

21. J.W. Conner and R.J. Hastie, Nucl. Fusion 13 (1973) 221.

22. K.C. Shaing and J.D. Callen, UWFDM-416 (1981).

23. R. Chodura, A. Schluter, IPP 1/180 (1980).

24. B.B. Kadomstev, Sov. J. Plasma Phys. 1 (1975) 389.

25. G.L. Jahns, M. Soler, B.V. Waddell, J.D. Callen, and H.R. Hicks, Nucl. Fusion 18 (1978) 609.

26. DISSPLA User's Manual, ISSCO, San Diego (1981).