ELSEVIER

Computer Physics Communications 97 (1996) 219–260

Computer Physics Communications

The CHEASE code for toroidal MHD equilibria

H. Lütjens [1], A. Bondeson [2], O. Sauter [3]

Centre de Recherches en Physique des Plasmas, Association Euratom, Confédération Suisse, Ecole Polytechnique Fédérale de Lausanne, Bâtiment PPB, CH-1015 Lausanne, Switzerland

Received 6 October 1995; revised 6 March 1996

Abstract

The CHEASE code (Cubic Hermite Element Axisymmetric Static Equilibrium) solves the Grad-Shafranov equation for toroidal MHD equilibria using a Hermite bicubic finite element discretization with pressure, current profiles and plasma boundaries specified by analytical forms or sets of experimental data points. Moreover, CHEASE allows the automatic generation of pressure profiles marginally stable to ballooning modes or with a prescribed fraction of bootstrap current. The code provides equilibrium quantities for several stability and global wave propagation codes.

Keywords: Plasma physics; Magnetohydrodynamics (MHD); Equilibrium; Grad-Shafranov equation; Cubic Hermite finite elements; Mapping to magnetic flux coordinates; Ballooning modes; Local interchange modes; Bootstrap current

PROGRAM SUMMARY

Title of program: CHEASE

Catalogue identifier: ADDH

Program obtainable from: CPC Program Library, Queen's University of Belfast, N. Ireland

Licensing provisions: none

Computer for which the program is designed and others on which it has been tested:
Computers: CRAY, NEC-SX3, SUN, IBM, SG and Hewlett Packard workstations;
Installations: Centre de Recherches en Physique des Plasmas, Ecole Polytechnique Fédérale de Lausanne, Switzerland;
Department of Technology, Uppsala University, Sweden;
Centro Ricerche Energia Frascati, ENEA, Italy;
Istituto Gas Ionizzati Padova, ENEA, Italy;
Ecole Polytechnique, Palaiseau, France;
General Atomics, San Diego, California, USA;
University of Wisconsin, USA;
UKAEA Fusion, Culham, Abingdon, United Kingdom;
Institute for Electromagnetic Field Theory, Chalmers University of Technology, Göteborg, Sweden;
ITER-San Diego, California, USA;
CEA sur la fusion, Cadarache, France;
Alfvén Laboratory, Royal Institute of Technology, Stockholm, Sweden;

Operating systems under which the program has been tested: Uni-

[1] Present address: Centre de Physique Théorique, CNRS UPR-14, Ecole Polytechnique, F-91128 Palaiseau Cedex, France. E-mail: lutjens@orphee.polytechnique.fr

[2] Present address: Institute for Electromagnetic Field Theory, Euratom-NFR Association, Chalmers University of Technology, S-412 96 Göteborg, Sweden. E-mail: elfab@elf.chalmers.se

[3] Present address: ITER-San Diego, 11025 N.Torrey Pines Rd, La Jolla, CA 92037, USA. E-mail: sauter@crppsun.epfl.ch

PII S0010-4655(96)00046-X

cos, SunOS, HP-UX.10, etc.

Programming language used: FORTRAN 77

Compiler options
SUN-SPARC Station 10: f77 -r8 -i4 -Nl30 -Nx300
HP-K200: f77 +autodblpad +Onolimit -O
IBM-RS6000: xlf -O -qautodbl=dblpad -qaux_size=16384 -qtkq_size=20000 -qst_size=3072
Silicon Graphics Indigo-2: f77 -O3 -r8 -i4
Cray C-90: cf77 -Wf"-o aggress" -Zp

Memory required to execute with typical data: About 8 MWords, runs with virtual memory on workstations without much time loss

No. of bits in a word: 64

Peripherals used: Disk files

No. of bytes in distributed program, including test data, etc.: 346968

Distribution format: uuencoded compressed tar file

Keywords: plasma physics, magnetohydrodynamics (MHD), equilibrium, Grad–Shafranov equation, cubic Hermite finite elements, mapping to magnetic flux coordinates, ballooning modes, local interchange modes, bootstrap current

Nature of the physical problem
CHEASE [1] solves the Grad–Shafranov equation [2–4] for the MHD equilibrium of a Tokamak-like plasma with pressure and current profiles specified by analytic forms or sets of data points. Equilibria marginally stable to ballooning modes [5] or with a prescribed fraction of bootstrap current [6–8] can be computed. The code provides a mapping to magnetic flux coordinates, suitable for MHD stability calculations or global wave propagation studies. The code computes equilibrium quantities for the stability codes ERATO [9], MARS [10], PEST [11,12], NOVA-W [13] and XTOR [14] and for the global wave propagation codes LION [15] and PENN [16].

Method of solution
The two-dimensional MHD equilibrium (Grad–Shafranov) equation is solved in variational form. The discretization uses a bicubic Hermite finite elements with continuous first order derivatives for the poloidal flux function Ψ. The nonlinearity of the problem is handled by a Picard iteration. The mapping to flux coordinates is carried out with a method which conserves the accuracy of the cubic finite elements.

Typical running time
Typical running times are given after the test run output.

Unusual features of the program
The code uses routines from the CRAY libsci.a program library. However, all these routines are included in the CHEASE package itself. If CHEASE computes equilibrium quantities for MARS with fast Fourier transforms, the NAG library is required. CHEASE is written in standard FORTRAN-77, except for the use of the input facility NAMELIST. CHEASE uses variable names with up to 8 characters, and therefore violates the ANSI standard. CHEASE transfers plot quantities through an external disk file to a plot program named PCHEASE using the UNIRAS or the NCAR plot package.

References
[1] H. Lütjens, A. Bondeson and A. Roy, Comput. Phys. Commun. 69 (1992) 287.
[2] V.D. Shafranov, ZhETF 33 (1957) 710; Sov. Phys. JETP 8 (1958) 494.
[3] R. Lüst and A. Schlüter, Z. Naturforsch. 129 (1957) 850.
[4] H. Grad and H. Rubin, Proc. 2nd Int. Conf. on the Peaceful Uses of Atomic Energy, Vol. 31 (United Nations, Geneva, 1958) p. 190.
[5] J.W. Connor, R.J. Hastie and J.B. Taylor, Phys. Rev. Lett. 40 (1978) 396.
[6] M.N. Rosenbluth, R.D. Hazeltine and F.L. Hinton, Phys. Fluids 15 (1972) 116.
[7] R.D. Hazeltine, F.L. Hinton and M.N. Rosenbluth, Phys. Fluids 16 (1973) 1645.
[8] S.P. Hirshman, Phys. Fluids 31 (1988) 3150.
[9] R. Gruber, F. Troyon, D. Berger, L.C. Bernard, S. Rousset, R. Schreiber, W. Kerner, W. Schneider and K.V. Roberts, Comput. Phys. Commun. 21 (1981) 323.
[10] A. Bondeson, G. Vlad and H. Lütjens, Phys. Fluids B 4 (1992) 1889.
[11] R.C. Grimm, J.M. Greene and J.L. Johnson, Methods Comput. Phys. 9 (1976) 253.
[12] R.C. Grimm, R.L. Dewar and J. Manickam, J. Comput. Phys. 49 (1983) 94.
[13] D.J. Ward and S.C. Jardin, Nucl. Fusion 32 (1992) 973.
[14] K. Lerbinger and J.F. Luciani, J. Comput. Phys. 97 (1991) 444.
[15] L. Villard, K. Appert, R. Gruber and J. Vaclavik, Comput. Phys. Reports 4 (1986) 95.
[16] A. Jaun, K. Appert, H. Lütjens, S. Brunner, J. Vaclavik, L. Villard, Theory of Fusion Plasmas, Proc. Int. Workshop, Varenna, 1994 (Editrice Compositori, Bologna, 1994) p. 369.

LONG WRITE-UP

1. Introduction

An accurate reconstruction of toroidal magnetohydrodynamic (MHD) equilibria is essential for the study of tokamak plasmas, in particular for understanding their stability or the propagation of electromagnetic waves. For this purpose, a fast, accurate and versatile equilibrium code is required, and this was the motivation for constructing the toroidal MHD equilibrium solver CHEASE [1]. This code is now extensively used in many laboratories. In Ref. [1] the good convergence properties of CHEASE due to the bicubic Hermite finite element discretization were reported. The object of this paper is a more complete documentation of the current version of CHEASE, including several extensions from the early version [1].

CHEASE now includes several ways of specifying the equilibrium profiles for current and pressure. It also allows automatic generation of pressure profiles marginally stable to ballooning modes [5] (localized pressure driven instabilities with high toroidal mode number n). The pressure profile can also be adjusted to generate equilibria with a prescribed profile of bootstrap current [6–8] (current along the magnetic field lines caused by the equilibrium pressure gradient). The present version of CHEASE supplies equilibrium quantities for the MHD stability codes ERATO [9] and MARS [10] (as documented in Ref. [1]), and also for the stability codes PEST [11,12], NOVA-W [13] and XTOR [14], and the wave propagation codes LION [15] and PENN [16]. The PENN code requires equilibrium quantities with continuous second derivatives, one order higher than what is provided by the bicubic Hermite finite elements used in CHEASE. The desired smoothness of the equilibrium is obtained by a bicubic spline interpolation of the bicubic Hermite element solution.

CHEASE has been modified to deal with equilibria having up–down asymmetric cross sections. This option is essential because many tokamaks, including JET (Joint European Torus) [18] and the planned ITER (International Thermonuclear Experimental Reactor) [19] operate in single-null divertor mode, thus generating equilibria with a magnetic separatrix and an X-point at the bottom of the cross section. The input to CHEASE, the plasma boundary and the equilibrium profiles, can be prescribed by functional forms or by sets of points. The latter option is used for the reconstruction of experimental equilibria.

2. The toroidal MHD equilibrium problem

2.1. General equations

The MHD equilibrium equations read

$$\begin{aligned} &\boldsymbol{J} \times \boldsymbol{B} = \nabla p\,, \\ &\nabla \times \boldsymbol{B} = \boldsymbol{J}\,, \\ &\nabla \cdot \boldsymbol{B} = 0\,, \end{aligned} \tag{1}$$

where $\boldsymbol{B}$ denotes the magnetic field, $\boldsymbol{J}$ the current density and p the plasma pressure, which is assumed to be isotropic. In axisymmetric geometry, the magnetic field can be represented as

$$\boldsymbol{B} = T\nabla\phi + \nabla\phi \times \nabla\Psi\,, \tag{2}$$

where ϕ is the ignorable toroidal angle (see Fig. 1) and Ψ is the poloidal magnetic flux function. In the following only static MHD equilibria are considered, and for these, the pressure p and the poloidal current flux function T are functions of Ψ only. Substituting Eq. (2) into Eq. (1) leads to the elliptic second-order nonlinear partial differential Grad–Shafranov equation [2–4],

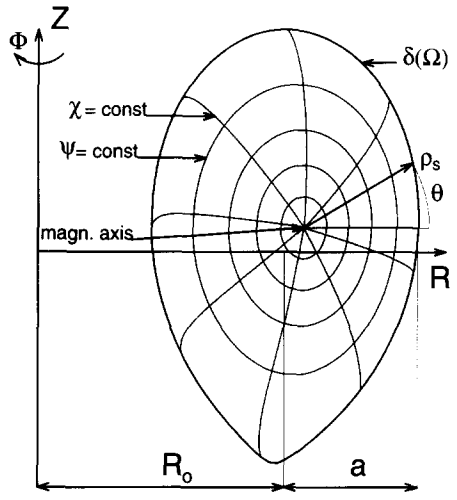

Fig. 1. The cylindrical coordinates $(R; Z; \phi)$ in toroidal geometry.

$$\nabla \cdot \frac{1}{R^2}\nabla\Psi = \frac{j_\phi}{R} = -p'(\Psi) - \frac{1}{R^2}TT'(\Psi)\,, \tag{3}$$

where j_ϕ denotes the toroidal plasma current density, R the major radius of the torus and prime the derivative with respect to Ψ. The nature of the equilibria (i.e. tokamak, reversed field pinch, etc.) is determined by the two free functions $p'(\Psi)$ and $TT'(\Psi)$.

In the following the plasma cross section Ω is assumed to be known. Thus, we restrict consideration of Eq. (3) to the fixed boundary case with $\Psi \equiv 0$ at the plasma edge $\delta\Omega$. In CHEASE, the shape of $\delta\Omega$ is rather arbitrary, and its prescription is described in Section 6.4.1. Furthermore, only cases with a single magnetic axis (where $\nabla\Psi = 0$) are considered. It is also assumed that $\Psi < 0$ everywhere inside the plasma and that the total plasma current

$$I = \int_\Omega j_\phi \, dS \tag{4}$$

is positive.

For the solution of the equilibrium equation (3), CHEASE transforms the plasma cross section Ω in Fig. 1 into a rectangular region $0 \le \sigma \le 1$, $0 \le \theta \le 2\pi$. The nonorthogonal coordinate system (σ, θ) is related to cylindrical coordinates (R, Z) by

$$\begin{aligned} R &= \sigma\rho_s(\theta)\cos\theta + R_c\,, \\ Z &= \sigma\rho_s(\theta)\sin\theta + Z_c\,. \end{aligned} \tag{5}$$

The variational form of Eq. (3) is discretized using bicubic Hermite finite elements. The nonlinear discretized system is solved by a Picard iteration (see Section 5.2). Here, we only point out that the equilibrium is computed in two steps. First a solution is generated on a coarse grid with a polar mesh centred at the geometrical midpoint $(R_0, 0)$ of the cross section. (The coordinate system is shifted so that $Z_0 = 0$.) Then the grid is refined and the origin of the polar coordinates (R_c, Z_c) is moved to the magnetic axis $(R_{\mathrm{mag}}, Z_{\mathrm{mag}})$ of the coarse solution. This is done to facilitate the subsequent mapping to flux coordinates.

2.2. Specification of the two free functions in the Grad–Shafranov equation

There are many different ways of defining the two free functions $p'(\Psi)$ and $TT'(\Psi)$ in Eq. (3). For example, early stability optimizations were made by specifying $p'(\Psi)$ and $TT'(\Psi)$ independently [20]. However, this method makes it difficult to control equilibrium quantities such as the safety factor

$$q(\Psi) = \frac{T(\Psi)}{2\pi} \oint_{\Psi=\text{const.}} \frac{dl}{R|\nabla\Psi|} \tag{6}$$

or the current density profile. The specification of a suitable averaged current density profile instead of the $TT'(\Psi)$ profile solves this problem [21,22].

In CHEASE, equilibrium profiles are given as functions of s, which denotes a function of the normalized poloidal flux $(\Psi - \Psi_{\text{edge}})/(\Psi_0 - \Psi_{\text{edge}})$, where Ψ_0 and Ψ_{edge} are the flux at the magnetic axis and at the plasma boundary, respectively. Two profiles need to be specified to define an equilibrium, roughly speaking one for the pressure $p' = dp/d\Psi$ and one for the current. The present version of CHEASE can treat three different options for specifying the current profile. Profiles can be prescribed for either of

- $TT'(s)$,
- the surface averaged current density,

$$I^*(s) = \frac{\oint_{s=\text{const.}} j_\phi (J/R)\, d\chi}{\oint_{s=\text{const.}} (J/R)\, d\chi} = -\frac{C_1}{C_0} p'(s) - \frac{C_2}{C_0} TT'(s)\,, \tag{7}$$

- the averaged parallel current density,

$$I_{||}(s) = \frac{\oint_{s=\text{const.}} \boldsymbol{J}.\boldsymbol{B} J\, d\chi}{\oint_{s=\text{const.}} \boldsymbol{B}.\nabla\phi J\, d\chi} = -\frac{C_1 p'(s)}{C_2} - TT'(s)\left(1 + \frac{1}{T^2(s)}\frac{C_3}{C_2}\right). \tag{8}$$

Here,

$$\left\{C_0(s), C_1(s), C_2(s), C_3(s)\right\} = \oint_{s=\text{const.}} \left\{\frac{1}{R}, 1, \frac{1}{R^2}, \frac{|\nabla\Psi|^2}{R^2}\right\} J\, d\chi \tag{9}$$

are surface integrals. The toroidal current density in Eq. (3) can be expressed as

$$j_\phi = \frac{1}{R}\frac{C_0}{C_2} I^*(s) + \left(\frac{1}{R}\frac{C_1}{C_2} - R\right) p'(s) \tag{10}$$

if $I^*(s)$ is specified and

$$j_\phi = \frac{1}{yR} I_{||}(s) + \frac{1}{y}\left(\frac{1}{R}\frac{C_1}{C_2} - Ry\right) p'(s)\,, \tag{11}$$

where

$$y = 1 + \frac{1}{T^2(s)}\frac{C_3}{C_2}$$

if $I_{||}(s)$ is prescribed. If the I^* or the $I_{||}$ profile is specified, the integrals (9) are evaluated for a given set of s values, and interpolated with cubic spline functions for the calculation of j_ϕ. This requires a mapping of the equilibrium solution into flux coordinates $(s(\Psi), \chi, \phi)$, where χ is a generalized poloidal angle (see Section 3). In that case the Grad–Shafranov equation is solved by two nested Picard iterations: one inner loop to solve for Ψ where j_ϕ is computed from (10) or (11) with fixed surface integrals (9) and one outer loop iterating on the integrals. These are the two innermost loops in Fig. 2.

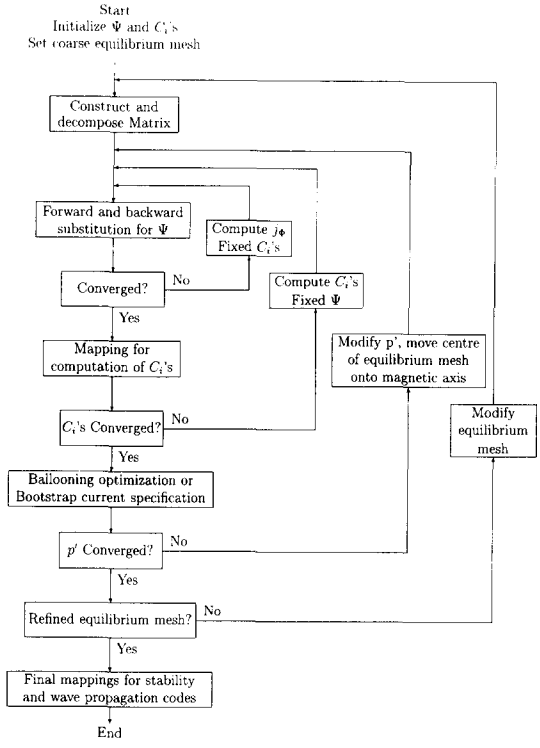

Fig. 2. Flow diagram of CHEASE.

2.3. *Transformation of the equilibrium*

A single solution of the Grad–Shafranov equation (3) can be rescaled to generate a whole sequence of equilibria with fixed poloidal beta and internal inductance, but with different plasma current, rotational transform and toroidal beta (see Section 5.3). In CHEASE these transformations allow the generation of equilibria with prescribed values of either the total current (4) or of the safety factor (6) at some arbitrary flux surface Ψ_q, and simultaneously with a specified value of T at a given flux surface Ψ_T.

After these transformations, all relevant physical quantities characterizing the equilibrium are computed. Table 1 shows the definitions of the most important global quantities evaluated in CHEASE. (Some of the quantities in Ref. [1], Table 1, have been rescaled by a factor R_{mag}/R_0 to agree with the commonly used definitions.)

3. Mapping of the equilibrium into flux coordinates

CHEASE provides the equilibrium quantities for the MHD stability codes ERATO [9], MARS [10], NOVA-W [13], PEST [11,12] and XTOR [14], and for the global wave propagation codes LION [15] and PENN [16]. All these codes use a flux coordinate system (s, χ, ϕ), where the radial coordinate s is defined as

$$s \equiv s(\Psi) = \sqrt{\frac{|\Psi_{\mathrm{edge}} - \Psi|}{|\Psi_{\mathrm{edge}} - \Psi_0|}}\,. \tag{12}$$

Ψ is the solution of Eq. (3), χ is a generalized poloidal angle and ϕ is the geometrical toroidal angle. The nonzero terms of the contravariant metric tensor of these coordinates are

Table 1
Physical quantities computed by CHEASE

Global quantities	
Total plasma volume / 2π	$V_{\rm tot} = \int_\Omega J\, d\Psi\, d\chi$
Volume average	$\overline{f} = \left(\int_\Omega fJ\, d\Psi\, d\chi\right) / V_{\rm tot}$
Total toroidal current	$I = \int_\Omega j_\phi (J/R)\, d\Psi\, d\chi$
Normalized total toroidal current	$I_N = \frac{I}{aT_{\rm edge}}$
Internal inductance	$\ell_i = \frac{4\pi}{I^2 R_0} \int_\Omega \frac{\lvert\nabla\Psi\rvert^2}{R^2} J\, d\Psi\, d\chi$
Pressure peaking factor	ppf. = $\frac{p_0}{\overline{p}}$
Total beta	$\beta = \frac{2\overline{p}}{B^2}$
Fusion beta	$\beta^* = \frac{2(\overline{p^2})^{1/2}}{B^2}$
Total experimental beta	$\beta_x = \frac{2R_0^2\overline{p}}{T_{\rm edge}^2}$
Total poloidal beta	$\beta_{p,\rm tot} = \frac{8\pi}{I^2 R_0} \overline{p} V_{\rm tot}$
Flux surface quantities	
Volume of Ψ = const. / 2π	$V(\Psi) = \int_{\Psi'=\Psi_{\rm min}}^{\Psi} \oint J\, d\chi\, d\Psi'$
Generalized radius of Ψ = const.	$\rho(\Psi) = \left(V(\Psi)/V_{\rm tot}\right)^{1/2}$
Toroidal current within Ψ = const.	$I_\phi(\Psi) = \int_{\Psi'=\Psi_{\rm min}}^{\Psi} \oint j_\phi (J/R)\, d\chi\, d\Psi'$
Poloidal beta on Ψ = const.	$\beta_p(\Psi) = -\frac{8\pi}{I_\phi^2(\Psi) R_0} \int_{\Psi'=\Psi_{\rm min}}^{\Psi} p'(\Psi') V(\Psi')\, d\Psi'$
Global shear on Ψ = const.	$\hat{s}(\Psi) = \frac{\rho}{q(\Psi)} \frac{dq}{d\rho}(\Psi)$

$$g^{11} = \left(\frac{\partial s}{\partial \Psi}\right)^2 |\nabla\Psi|^2, \quad g^{12} = g^{21} = \frac{\partial s}{\partial \Psi} \nabla\Psi \cdot \nabla\chi \equiv \frac{\partial s}{\partial \Psi} |\nabla\Psi|^2 \beta_{\Psi\chi},$$
$$g^{22} = |\nabla\chi|^2, \quad g^{33} = |\nabla\phi|^2 = \frac{1}{R^2}. \tag{13}$$

The covariant metric tensor $g_{ij} = (g^{ij})^{-1}$ is obtained by inverting Eq. (13).

In ERATO, MARS, LION, NOVA-W and PEST, the angular variable χ is specified by the choice of the Jacobian $J = [(\nabla\Psi \times \nabla\chi)\cdot\nabla\phi]^{-1}$ of the mapping from (Ψ, χ, ϕ) space to Cartesian coordinates. In CHEASE, J can have the functional form

$$J = C(\Psi) R^\alpha |\nabla\Psi|^\mu, \tag{14}$$

where α and μ are integers. $C(\Psi)$ is determined by demanding that χ increases by 2π per poloidal turn. The generalized poloidal angle χ and the nonorthogonality $\beta_{\Psi\chi}$ can be expressed in the equilibrium coordinates (σ, θ),

$$\chi(\theta) = \int_0^\theta \frac{R\sigma\rho_s^2(\tilde{\theta})}{J\, \partial\Psi/\partial\sigma} d\tilde{\theta},$$

$$\beta_{\Psi\chi}(\theta) = \int_0^{\theta} \left\{ \frac{Rj_\phi}{|\nabla\Psi|^2} + (2-\alpha)\left[\frac{\partial(\ln R)}{\partial\Psi}\right]_n - (\mu+2)\left[\frac{\partial(\ln|\nabla\Psi|)}{\partial\Psi}\right]_n - \frac{C'(\Psi)}{C(\Psi)} \right\} \frac{R\sigma\rho_s^2(\tilde{\theta})}{J\,\partial\Psi/\partial\sigma} d\tilde{\theta}\,. \quad (15)$$

For completeness, a derivation of (15) is given in Appendix A. The subscript n in (15) stands for the normal derivative with respect to Ψ, j_ϕ is defined by Eq. (3) and $C'(\Psi)$ is computed from the periodicity condition $\beta_{\Psi\chi}(0) = \beta_{\Psi\chi}(2\pi)$. In CHEASE the generalized poloidal angle χ and the nonorthogonality $\beta_{\Psi\chi}$ in Eq. (15), the safety factor q (6) and the four integrals (9) are computed by Gauss integrations (see Ref. [1], Section 4.1) that preserve the convergence rate of the cubic Hermite elements. Section 5.4 gives a list of the Equilibrium Quantities (EQ's) required by the different stability and wave propagation codes, together with the method of computation.

4. Automatic generation of pressure profiles

CHEASE contains options for automatic generation of pressure profiles to either of the following two criteria:
- a profile marginally stable to ballooning modes (ballooning optimization, or BO);
- a pressure profile giving a certain profile of bootstrap current (specification of bootstrap current, or SBC).

To arrive at reasonable equilibria, it is usually preferable to specify the current profile by the I^* or $I_{||}$ options.

4.1. Ballooning, ideal and resistive interchange criteria

Ballooning modes are internal toroidal pressure-driven modes [5]. In the limit of an infinite toroidal mode number n, their potential energy reads ([23], Chapter 10.5.3)

$$\delta\mathcal{W}_p(n\to\infty) = \tfrac{1}{2}\int_{-\infty}^{+\infty} \left\{ c_1 \left|\frac{\partial\xi_r}{\partial\chi}\right|^2 + c_2|\xi_r|^2 \right\} J\,d\chi\,, \quad (16)$$

where ξ_r is the radial component of the displacement vector, and χ is a generalized poloidal angle extending from $-\infty$ to $+\infty$. For ballooning stability, $\delta\mathcal{W}_p$ must be positive definite on every flux surface. The quantities appearing in (16) are [5]

$$\begin{aligned}
c_1 &= \frac{1}{J^2|\nabla\Psi|^2}\left(1 + \frac{|\nabla\Psi|^4}{B^2}g^2\right),\\
c_2 &= -\frac{2p'}{B^2}\left[\left(\frac{\partial\tilde{P}}{\partial\Psi}\right)_n - \frac{gT}{B^2}\frac{1}{J}\left(\frac{\partial\tilde{P}}{\partial\chi}\right)_\Psi\right],\\
g &= \nu\beta_{\Psi\chi} + \int_{\chi_0}^{\chi}\left(\frac{\partial\nu}{\partial\Psi}\right)_\chi d\chi',\\
\nu &= JT/R^2,\\
B^2 &= (T^2 + |\nabla\Psi|^2)/R^2,\\
\tilde{P} &= p + B^2/2\,.
\end{aligned} \quad (17)$$

g in Eq. (17) is evaluated in CHEASE with straight fieldline coordinates characterized by $J = qR^2/T$ [11,12]. Thus, g reads

$$g = \nu(\chi_0)\beta_{\Psi\chi}(\chi_0) + q(\beta^{\text{s.f.}}_{\Psi\chi}(\chi) - \beta^{\text{s.f.}}_{\Psi\chi}(\chi_0)) + q'(\chi^{\text{s.f.}}(\chi) - \chi^{\text{s.f.}}(\chi_0))\,, \quad (18)$$

where the superscript s.f. stands for straight fieldline. More details about the resolution of Eq. (16) can be found in Ref. [1], Appendix C.3.

The Mercier [24] and the resistive interchange [26] criteria are checked on every constant poloidal flux surface. A given flux surface is stable to ideal interchanges if the Mercier criterion $-D_I > 0$ is satisfied, where

$$-D_I = \left(\frac{p'TJ_2}{q'} - \tfrac{1}{2}\right)^2 + \frac{p'}{q'^2}(J_5' - p'J_3)(T^2J_1 + J_4)\,. \tag{19}$$

Resistive interchanges are stable if $-D_R > 0$, with

$$-D_R = -D_I - (H - 1/2)^2 \tag{20}$$

and

$$H = \frac{Tp'}{q'}\left(J_2 - \frac{J_5(J_4 + T^2J_1)}{J_6 + T^2J_4}\right)\,. \tag{21}$$

The primes in Eqs. (18), (19), (21) denote the derivative with respect to Ψ. Appendix B.1 shows the derivation of these criteria in terms of the integrals

$$\left\{J_1, J_2, J_3, J_4, J_5, J_6\right\} = \frac{1}{2\pi}\oint_{\Psi=\text{const.}} \left\{\frac{1}{R^2|\nabla\Psi|^2}, \frac{1}{|\nabla\Psi|^2}, \frac{R^2}{|\nabla\Psi|^2}, \frac{1}{R^2}, 1, \frac{|\nabla\Psi|^2}{R^2}\right\} Jd\chi \tag{22}$$

and Appendix B.2 details about the numerical evaluation of the Mercier criterion.

It is well known that ballooning stability is a more restrictive condition than Mercier stability. However, for practical reasons, ballooning stability is computed by truncating the integration in (16) to a finite number of turns in θ. As a consequence the test for ballooning stability may fail to detect unstable, so-called weakly ballooning modes, which occur for low shear. However, this type of instability is detected by the Mercier criterion. Therefore, to ensure local ideal stability, it is standard practice to use a rather moderate integration interval in θ for ballooning, say $10 \times 2\pi$ and also to check for Mercier stability.

4.2. *Formula for the bootstrap current*

In tokamaks, the toroidal current may be generated by applying a toroidal electric field induced by a time-varying magnetic flux down the center column (Ohmic current), or by means of radiofrequency waves (RF current) or of neutral particle beam injection (NBI), which both affect the particle distributions. In addition to these externally generated currents, there is the "bootstrap current" [6–8] that is generated by the pressure gradient of the plasma (if the collisionality is sufficiently low). In "advanced tokamak" scenarios, a large fraction of the current comes from the bootstrap effect. For such applications it is useful to be able to specify the bootstrap current profile. In CHEASE the bootstrap current is computed using the formulas given by Hirshman [8] for low collisionality plasmas (in the banana regime), specialized to the case of a two component plasma with equal temperatures $T_e = T_i$,

$$\langle \boldsymbol{J}\cdot\boldsymbol{B}\rangle_{\text{bs}} = -L_{31}\left[A_1 + \frac{\alpha_i}{Z_i}A_2^i\right] - L_{32}A_2^e\,, \tag{23}$$

where

$$L_{31} = j_0 x\left[0.754 + 2.21Z_i + Z_i^2 + x(0.348 + 1.243Z_i + Z_i^2)\right]/D(x)\,,$$
$$L_{32} = -j_0 x(0.884 + 2.074Z_i)/D(x)\,,$$

$$\alpha_i = -1.172/(1+0.462x)\,,$$
$$D(x) = 1.414Z_i + Z_i^2 + x(0.754 + 2.657Z_i + 2Z_i^2) + x^2(0.348 + 1.243Z_i + Z_i^2)\,, \tag{24}$$

and

$$x = \frac{1-f_c}{f_c}\,, \qquad f_c = \tfrac{3}{4}\langle B^2\rangle \int_0^{B_{\max}^{-1}} \frac{y\,dy}{\langle (1-yB)^{1/2}\rangle}\,,$$
$$A_1 = p'(\Psi)\,, \qquad A_2^e = A_2^i = \frac{\eta_i}{1+\eta_i}\frac{Z_i}{Z_i+1}p'(\Psi)\,,$$
$$j_0 = T(\Psi)\,, \qquad \eta_i = \frac{d(\log T_e)}{d(\log n)}\,. \tag{25}$$

The brackets $\langle\cdots\rangle$ denote the flux surface average $\oint_{\Psi=\text{const.}}\cdots J\,d\chi / \oint_{\Psi=\text{const.}} J\,d\chi$ and f_c is the fraction of circulating particles on a flux surface.

4.3. Method of solution

Except in regions of weak or negative shear, the normal inward pressure gradient destabilizes ballooning modes. Furthermore, the bootstrap current (23) is proportional to p'. Therefore, for a given current profile (whether it is specified by TT', I^* or $I_{||}$), the pressure profile can be adjusted in such a manner that the equilibrium is (a) marginally stable to ballooning modes or (b) the fraction of the parallel current driven by bootstrap is a prescribed function of the radial coordinate s. As shown in Fig. 2, CHEASE accomplishes this by adding a loop of iteration for the pressure profile, external to the two loops for solving the Grad–Shafranov equation with given pressure and current profiles. The pressure profile at iteration step $k+1$ is generated from the pressure profile of iteration step k by different algorithms for the ballooning optimization and the specification of the bootstrap current profile. Further information about these algorithms is given in Section 5.5.

5. Organization of CHEASE

The computations in CHEASE are directed by the subroutine STEPON and can be subdivided into the solution of the Grad–Shafranov equation, described in Section 5.2, and the mapping to flux coordinates for different codes, described in Section 5.4. Before the mapping, the equilibrium is scaled according to the scaling laws presented in Section 5.3.

5.1. Input files and initializations

The different steps in solving the Grad–Shafranov equation (3) are shown in the flow diagram of Fig. 2. The code starts by setting the default cases in subroutine PRESET and by reading the Namelist variables from input channel 5. The default case is the first of the test cases presented in [1]. For an equilibrium reconstructed with experimental data, equilibrium profiles and a set of boundary coordinates are read in subroutine AUXVAL from file EXPEQ or EQDSK. This operation is described in Sections 6.4.1 and 6.4.2.

The equilibrium is first computed on a coarse grid centred at $(R_0, 0)$ and then on a refined grid centred at the magnetic axis of the previous equilibrium. The shift of the grid centre makes it easy to trace the constant-Ψ surfaces closest to the magnetic axis (which are required to surround the grid centre after the shift). The size of the coarse equilibrium mesh is prescribed in subroutine EQDIM. Good results are obtained with $N_\sigma = N_\theta = 24$ for that grid. (However, if a very dense radial stability mesh is required close to the magnetic

axis, or an equilibrium with a complex plasma boundary shape is computed, it may be necessary to increase the dimensions of the "coarse" mesh.)

The Picard iteration is initialized in subroutine GUESS by using a paraboloid centred at $(R_0, 0)$ for the poloidal flux function. If the current density j_ϕ is computed using Eq. (11), the T profile is initialized to 1 for the first iteration over the integrals (9).

5.2. Solution of the Grad–Shafranov equation

5.2.1. Variational formulation

CHEASE uses a variational finite element method for solving the Grad–Shafranov equation (3) [29],

$$\int_\Omega \frac{1}{R}\nabla\omega\cdot\nabla\Psi\, dS + \int_\Omega \omega j_\phi\, dS = 0\,, \tag{26}$$

where ω is an arbitrary weighting function from the same function space as Ψ. Eq. (26) is solved numerically in the standard manner of the finite element method by expanding Ψ in Hermite bicubics on the rectangular grid (σ, θ) [28,29]. The unknowns of the discretized equilibrium problem are the values of the function Ψ, its first derivatives $\partial\Psi/\partial\sigma$ and $\partial\Psi/\partial\theta$ and the mixed second derivative $\partial^2\Psi/\partial\sigma\partial\theta$, all at the nodes of the mesh. The integrals in Eq. (26) are carried out numerically using Gaussian quadrature.

The nonlinear equation (26) is solved by Picard iteration, i.e. the source term for the $(k+1)$th iteration is computed from the solution of the kth iteration,

$$\int_\Omega \frac{1}{R}\nabla\omega\cdot\nabla\Psi_{k+1}\, dS = -\int_\Omega \omega j_\phi(\Psi_k)\, dS\,. \tag{27}$$

The Picard iteration is interrupted when

$$\|\Psi_{k+1} - \Psi_k\| < \epsilon\,, \tag{28}$$

where ϵ is a predefined number. The norm used in Eq. (28) is $\|u\| = \left[\int_\Omega u^2\, dS\right]^{1/2}$.

5.2.2. Boundary conditions

As mentioned in Section 2.1, we consider the fixed boundary case $\Psi \equiv 0$ at $\delta\Omega$. This implies $\Psi = 0$ and $\partial\Psi/\partial\theta = 0$ for all the boundary points $\sigma = 1$ and $\theta = \theta_j, j = 1, \ldots, N_\theta$, where N_θ is the number of intervals in the θ-direction.

The origin of the polar coordinate system requires extra care as the coordinate transformation (5) becomes singular there and one single geometrical point is represented by N_θ mesh points. Conditions have to be imposed to ensure that Ψ is a regular function of R and Z at the origin. Taylor expansion of Ψ around (R_c, Z_c), when expressed in terms of (σ, θ), gives

$$\Psi = \Psi_c + \sigma\rho_s(\theta)[\Psi_R\cos\theta + \Psi_Z\sin\theta] + \mathcal{O}(\sigma^2)\,. \tag{29}$$

It follows that the regularity condition forces the $4N_\theta$ unknowns Ψ, $\partial\Psi/\partial\sigma$, $\partial\Psi/\partial\theta$ and $\partial^2\Psi/\partial\sigma\partial\theta$ for $\sigma = 0$ and $\theta = \theta_j, j = 1, \ldots, N_\theta$ to be replaced by the three unknowns Ψ_c, Ψ_R and Ψ_Z. The following conditions are imposed by collocation at the N_θ grid points for $\sigma = 0$:

$$\Psi = \Psi_c\,,$$
$$\frac{\partial\Psi}{\partial\theta} = 0\,,$$

$$\frac{\partial \Psi}{\partial \sigma} = \rho_s(\theta)\,[\Psi_R \cos\theta + \Psi_Z \sin\theta]\,,$$
$$\frac{\partial^2 \Psi}{\partial \sigma \partial \theta} = \rho_s(\theta)\,[-\Psi_R \sin\theta + \Psi_Z \cos\theta] + \frac{d\rho_s}{d\theta}\,[\Psi_R \cos\theta + \Psi_Z \sin\theta]\,. \tag{30}$$

The conditions (30) ensure the continuity of Ψ and $\nabla\Psi$ at the mesh center.

5.2.3. Implementation into CHEASE

The construction and the decomposition of the left-hand side of the variational form (26) is required only once per discretization mesh. The corresponding matrix is built in subroutine SETUPA, and the boundary conditions are imposed in subroutine LIMITA. The resulting band matrix A is symmetric positive definite, and therefore only the upper half-band needs to be stored. The numbering of the $\theta_j, j = 1, \ldots, N_\theta$ grid is alternative up–down in CHEASE, which reduces the memory requirement for the storage of A by about a factor 2 as compared to a clockwise numbering. This numbering is related to an inverse clockwise numbering i by

$$j = \begin{cases} i, & i = 1, \\ 2(i-1), & 2 \le i \le N_\theta/2 + 1, \\ 2(N_\theta - i) + 3, & N_\theta/2 + 2 \le i \le N_\theta. \end{cases} \tag{31}$$

The matrix elements are localized vertically by an index array which is computed once per discretization mesh in subroutine INITIA, and horizontally by a statement function defined in the include file BNDIND.inc. The matrix is decomposed into LDL^t, where D is diagonal and L^t fills the upper half-band of A in subroutine ALDLT by a standard Gauss–Seidel algorithm.

The vector corresponding to the right-hand side of the variational form (26) is computed in subroutine SETUPB, and the boundary conditions are imposed in subroutine LIMITB. The calculation of the source term j_ϕ is performed in subroutine CURENT, whatever option is chosen for its specification (see Section 2.2). If the current density is given in terms of $I^*(s)$ or $I_{||}(s)$, the integrals (9) required at the Gaussian quadrature points for the integration of Eq. (26) are obtained by cubic spline interpolations on a prescribed set of s-values. The computation of the 4 integrals at these s-values is performed in subroutine PROFIL. In this subroutine, first the constant-Ψ surfaces are determined in terms of the equilibrium coordinates (σ, θ) in subroutine ISOFIND, and next, the integrals (9) are computed by Gaussian quadrature along these surfaces in subroutine CINT. The integration is done with a method described in Ref. [1], Section 4.1 which preserves the accuracy of the bicubic finite element solution.

As discussed in Section 2.2, the nonlinearity of the Grad–Shafranov equation (3) is solved by two nested Picard iterations. For the inner loop lead by subroutine NONLIN, the integrals (9) are held fixed until convergence. Then, the integrals (9) are reevaluated from the new equilibrium solution. This requires a mapping of the equilibrium into flux coordinates. The iterations for the integrals (9) is directed by subroutine ITIPR, and the process stops when

(i) Eq. (28) is satisfied.

(ii) $\left[\sum_{s-grid} \{C_{i,l+1}(s) - C_{i,l}(s)\}^2\right]^{1/2} < \epsilon$ where $i = 1, \ldots, 4$ in eq. (9) and l labels the iteration loop over the integrals.

The iterations over Ψ in Eq. (27) only require forward and backward substitutions on the right-hand side of the variational form (26) as long as the discretization grid remains unchanged. These operations are executed in subroutine SOLVIT.

5.3. Scaling of the equilibrium

There are two transformation rules that can be used to generate families of solutions to the Grad–Shafranov equation starting from a single solution. The first is a rescaling,

$$\Psi_{\text{new}} = a_1 \Psi_{\text{old}}\,, \quad T_{\text{new}} = a_1 T_{\text{old}}\,, \quad p_{\text{new}} = a_1^2 p_{\text{old}}\,. \tag{32}$$

and the second is a shift of T^2,

$$T^2_{\text{new}} = T^2_{\text{old}} + a_2\,, \tag{33}$$

with Ψ and p unchanged. These transformations leave the pressure and toroidal current profiles (as well as the poloidal beta and internal inductance) unchanged and are applied to generate equilibria satisfying normalization conditions with regard to the plasma current. These operations are carried out in the subroutine NOREPT by two different methods:

- The total plasma current is specified. This is accomplished by first applying the scaling (32) with $a_1 = I_{\text{spec}}/I_{\text{old}}$ and then shifting T^2 using (33) with $a_2 = T^2_{\text{spec}}(\Psi_T) - T^2(\Psi_T)$, where T denotes the value after rescaling.
- The safety factor q is prescribed at a chosen value for the normalized flux function Ψ_q. In this case, T^2 is first shifted by (33) with $a_2 = [q^2_{\text{spec}}/q^2_{\text{old}}(\Psi_q) - 1]T^2(\Psi_q)$, and then the solution is rescaled by $a_1 = T_{\text{spec}}/T(\Psi_T)$, where T denotes the value after the shift.

For equilibria with prescribed TT' or I^* profiles for the definition of the source term j_ϕ in Section 2.2, these operations are applied to the converged equilibrium solution of Eq. (27). However, if the current density is given in terms of $I_{||}$ by Eq. (11), a shift of T modifies the profile for $j_\phi(\Psi)$, contrary to the other current profile definitions. Unlike the equilibria with given TT' or I^* profiles, the coefficients defining $I_{||}$ in Section 6.4.2 must be adjusted iteratively so that the converged equilibrium satisfies $q(\Psi_q) = q_{\text{spec}}$ and $T(\Psi_T) = T_{\text{spec}}$. Numerical tests have shown that optimal convergence properties are obtained when the scaling is executed once at every iteration over the p' profile if a ballooning optimized equilibrium or an equilibrium with specified bootstrap current is computed, and once after every iteration over the flux surface integrals (9) otherwise. In order to provide a good initial guess, $I_{||}$ is first scaled in subroutine GUESS by a factor $2/q_{\text{spec}} I_{||}(0)$.

5.4. Mappings for global mode codes

After scaling (Section 5.3), the equilibrium is mapped into flux coordinates, as used by the stability and wave codes. The code for which CHEASE produces EQ's is selected by means of the Namelist parameter NIDEAL as follows:

- NIDEAL = 0 : MARS
- NIDEAL = 1 : ERATO
- NIDEAL = 2 : LION
- NIDEAL = 3 : NOVA-W and PEST
- NIDEAL = 4 : PENN
- NIDEAL = 5 : XTOR
- NIDEAL = 6 : EQDSK file (US standard input/output, e.g. of EFIT [17])

The different mappings have a common core, which consists of first tracing the constant-Ψ surfaces, and second computing flux surface integrals such as Eq. (6) or Eq. (15). The flux surfaces are traced in the subroutine ISOFIND, which computes the (σ, θ) coordinates for the intersections of the constant-Ψ surfaces with the equilibrium discretization mesh and for the Gaussian quadrature points along the constant-Ψ surfaces, used for the flux surface integrations. For every intersection, a cubic equation has to be solved, and performance tests have shown that numerical evaluation of the roots by a bissection method is much cheaper in terms of cpu consumption than using the analytical Cardan formulas. Moreover, χ and $\beta_{\Psi\chi}$ are computed at the intersections of the constant-Ψ surfaces with the equilibrium discretization mesh (σ, θ) in that subroutine. As shown by Eq. (15) the generalized poloidal angle χ is determined by specifying the Jacobian J (14).

Table 2
Vacuum quantities for MARS

j	EQLV(j)
1	$\frac{g_{ss}}{J_v} = \frac{1}{J_v}\left((R_v - R_{vc})^2 + (Z_v - Z_{vc})^2\right)$
2	$\frac{g_{\chi\chi}}{J_v} = \frac{s^2}{J_v}\left(\left(\frac{\partial R_v}{\partial \chi}\right)^2 + \left(\frac{\partial Z_v}{\partial \chi}\right)^2\right)$
3	$\frac{g_{\phi\phi}}{J_v} = \frac{1}{J_v}(R_{vc} + s(R_v - R_{vc}))^2$
4	$\frac{g_{s\chi}}{J_v} = \frac{s}{J_v}\left((R_v - R_{vc})\frac{\partial R_v}{\partial \chi} + (Z_v - Z_{vc})\frac{\partial Z_v}{\partial \chi}\right)$

5.4.1. ERATO and LION

The EQ's for the linear ideal MHD code ERATO [9] and the wave propagation code LION [15] are identical, and are given in Ref. [1], Appendix C1, Table 2. For both codes, all distances and profiles must be rescaled so that $R_{mag} = 1$ and $T_{mag} = 1$ (see Section 6.4.4). Therefore, the equilibrium is first rescaled according to the rules in Section 5.3. Subsequently, the magnetic fields and spatial scales are redefined as follows:

$$
\begin{aligned}
&\Psi_{\text{ERATO, LION}} = \Psi_{\text{CHEASE}}/R_{mag}\,,\\
&R_{\text{ERATO, LION}} = R_{\text{CHEASE}}/R_{mag}\,,\\
&Z_{\text{ERATO, LION}} = Z_{\text{CHEASE}}/R_{mag}\,,\\
&TT'_{\text{ERATO, LION}} = TT'_{\text{CHEASE}}R_{mag}\,,\\
&p_{\text{ERATO, LION}} = p_{\text{CHEASE}}R^2_{mag}\,,\\
&p'_{\text{ERATO, LION}} = p'_{\text{CHEASE}}R^3_{mag}\,,\\
&I^*_{\text{ERATO, LION}} = I^*_{\text{CHEASE}}R^2_{mag}\,,\\
&I_{||\text{ERATO, LION}} = I_{||\text{CHEASE}}R_{mag}\,,
\end{aligned}
\tag{34}
$$

before computing the EQ's.

ERATO and LION require EQ's on a (Ψ, χ) mesh. This χ-mesh is not the same as the $\chi(\sigma, \theta)$-values computed in subroutine SURFACE. The (σ, θ) coordinates of the (Ψ, χ) nodes used for computing the EQ's are calculated in subroutine CHIPSI by cubic spline interpolations of the $\chi(\sigma, \theta)$ and the $\beta_{\Psi\chi}(\sigma, \theta)$ values obtained from subroutine SURFACE along a constant-Ψ surface. Then, every EQ required by ERATO or LION is computed in terms of the equilibrium coordinates (σ, θ) in subroutine ERDATA. The (χ, Ψ) mesh is also used for the evaluation of the ballooning stability criterion (16) and quantities for plot files, such as the local shear and the magnetic field-line curvature.

5.4.2. MARS

The resistive MHD linear stability code MARS [10] uses Fourier expansion in the poloidal direction, and a radial discretization with piecewise linear and constant functions [30]. No specific scaling of the equilibrium is necessary for MARS (although the standard prescription is to set $R_0 = 1$ and $T = 1$ in the vacuum). The EQ's used by this code are Fourier transforms in χ of quantities defined at constant-Ψ surfaces, at both integer and half-integer meshes. They are presented in Ref. [1], Table 3. A more recent version of MARS (including resistive walls) also requires geometrical quantities in the vacuum region surrounding the plasma. For completeness, these quantities are documented in Table 2. The vacuum mesh $(s; \chi; \phi)$ for MARS is defined so that

$$R = R_{vc} + s(R_v - R_{vc})\,,$$

Table 3
Equilibrium quantities for NOVA-W and PEST

1	p	$s_{l+1/2}$
2	$p'(\Psi)$	s_l
3	q	s_l
4	$q'(\Psi)$	s_l
5	T	s_l
6	$T'(\Psi)$	s_l
7	T/q	s_l
8	$\frac{d}{d\Psi}\left[\frac{T}{q}\right]$	s_l
9	Ψ	s_l
10	Ψ_m	$s_{l+1/2}$
11	R	(s_l, χ_k)
12	Z	(s_l, χ_k)
13	J	$(s_l, \chi_{k+1/2})$
14	J	(s_l, χ_k)

$$Z = Z_{vc} + s(Z_v - Z_{vc}) , \tag{35}$$

where $(R_v; Z_v)$ are the Cartesian coordinates of the $(s; \chi)$ nodes at the plasma surface and $(R_{vc}; Z_{vc})$ is the centre of the vacuum mesh. Therefore, the Jacobian of the transformation from the $(s; \chi; \phi)$ space to Cartesian coordinates in the vacuum is given by

$$J_v = sR\left((R_v - R_{vc})\frac{\partial Z_v}{\partial \chi} - (Z_v - Z_{vc})\frac{\partial R_v}{\partial \chi}\right) . \tag{36}$$

All the quantities required by MARS in the plasma and the vacuum are Fourier transformed in CHEASE. Two possibilities exist: (a) The Fourier transform described in Ref. [1], Appendix C.2 by setting the Namelist parameter NFFTOPT = 0 and (b) a fast Fourier transform (NFFTOPT = 1). If the FFT's are selected, the code needs the NAG library. For every Fourier transform, χ and sometimes $\beta_{\Psi\chi}$ are required at a set of points along constant-Ψ surfaces. These quantities are computed in a similar way as the (Ψ, χ) nodes for ERATO in Section 5.4.1 with cubic spline interpolations in subroutine GCHI. The EQ's in Table 2 and Ref. [1], Table 3 are first calculated at a set of points in subroutine GIJLIN, and eventually Fourier transformed in subroutine FOURIER if NFFTOPT = 0 or FOURFFT if NFFTOPT = 1. Similarly, the vacuum EQ's in Table 2 are computed and Fourier transformed in subroutine VACUUM or VACUFFT depending on the choice for NFFTOPT.

5.4.3. NOVA-W and PEST

The stability codes NOVA-W and PEST require EQ's at both the integer and the half-integer s and χ meshes. The integer grids are indexed s_l and χ_k in Table 3, whereas the half-integer grids are labeled $s_{l+1/2}$ and $\chi_{k+1/2}$. NOVA-W and PEST use a different length scaling than CHEASE, ERATO or MARS: all distances are given in meters, while $RB_\phi = T$ is normalized to unity in the vacuum. Therefore, the EQ's for these codes are rescaled (in a separate program processing the CHEASE output before use in PEST or NOVA-W) according to the following rules:

$$p'_{\text{NOVA-W, PEST}} = p'_{\text{CHEASE}}/R^2_{\text{maj}} ,$$
$$T'_{\text{NOVA-W, PEST}} = T'_{\text{CHEASE}}/R_{\text{maj}} ,$$
$$q'_{\text{NOVA-W, PEST}} = q'_{\text{CHEASE}}/R^2_{\text{maj}} ,$$
$$(T/q)_{\text{NOVA-W, PEST}} = R_{\text{maj}}(T/q)_{\text{CHEASE}} ,$$
$$\Psi_{\text{NOVA-W, PEST}} = R^2_{\text{maj}}\Psi_{\text{CHEASE}} ,$$

Table 4
Equilibrium quantities for XTOR

1	s	$s_l, s_{l+1/2}$
2	Ψ	$s_l, s_{l+1/2}$
3	$\frac{\partial \Psi}{\partial s}$	$s_l, s_{l+1/2}$
3	p	$s_l, s_{l+1/2}$
3	$p'(\Psi)$	$s_l, s_{l+1/2}$
4	T	$s_l, s_{l+1/2}$
5	$TT'(\Psi)$	$s_l, s_{l+1/2}$
6	R	$(s_l, \chi_k), (s_{l+1/2}, \chi_k)$
6	Z	$(s_l, \chi_k), (s_{l+1/2}, \chi_k)$
7	$J = \frac{1}{R}\left(\frac{\partial s}{\partial R}\frac{\partial \chi}{\partial Z} - \frac{\partial s}{\partial Z}\frac{\partial \chi}{\partial R}\right)$	$(s_l, \chi_k), (s_{l+1/2}, \chi_k)$
8	$g^{ss} = \left(\frac{\partial s}{\partial R}\right)^2 + \left(\frac{\partial s}{\partial Z}\right)^2$	$(s_l, \chi_k), (s_{l+1/2}, \chi_k)$
9	$g^{s\chi} = \frac{\partial s}{\partial R}\frac{\partial \chi}{\partial R} + \frac{\partial s}{\partial Z}\frac{\partial \chi}{\partial Z}$	$(s_l, \chi_k), (s_{l+1/2}, \chi_k)$
10	$g^{\chi\chi} = \left(\frac{\partial \chi}{\partial R}\right)^2 + \left(\frac{\partial \chi}{\partial Z}\right)^2$	$(s_l, \chi_k), (s_{l+1/2}, \chi_k)$
11	$g^{\Phi\Phi} = \frac{1}{R^2}$	$(s_l, \chi_k), (s_{l+1/2}, \chi_k)$

$$
\begin{aligned}
R_{\text{NOVA-W, PEST}} &= R_{\text{maj}} R_{\text{CHEASE}}\,, \\
Z_{\text{NOVA-W, PEST}} &= R_{\text{maj}} Z_{\text{CHEASE}}\,, \\
J_{\text{NOVA-W, PEST}} &= R_{\text{maj}} J_{\text{CHEASE}}\,,
\end{aligned} \tag{37}
$$

where R_{maj} is the major axis of the torus in meters. The EQ's in Table 3 are computed in subroutine OUTNVW by a method similar to the one used for the EQ's of ERATO in Section 5.4.1.

5.4.4. XTOR

The resistive MHD nonlinear time evolution code XTOR [14] uses the radial flux coordinate (12) and an equidistant poloidal angle χ equal to the geometrical polar angle for the discretization. The EQ's for XTOR are computed in subroutine OUTXT. A list of these quantities is shown in Table 4. Identical symbolic rules as in Section 5.4.3 are used for the indexation of the integer and half-integer grids. As for ERATO, PEST and NOVA-W, the scalings used in XTOR differ from the ones in CHEASE. Here, the equilibrium is first rescaled such that RB_Φ is unity at the magnetic axis (see Section 5.3) and second that the minor axis of the torus is unity, i.e.

$$
\begin{aligned}
\Psi_{\text{XTOR}} &= A^3 \Psi_{\text{CHEASE}} \\
\left(\frac{\partial \Psi}{\partial s}\right)_{\text{XTOR}} &= A^3 \left(\frac{\partial \Psi}{\partial s}\right)_{\text{CHEASE}} \\
p_{\text{XTOR}} &= A^2 p_{\text{CHEASE}}\,, \\
p'_{\text{XTOR}} &= p'_{\text{CHEASE}}/A\,, \\
T_{\text{XTOR}} &= A^2 T_{\text{CHEASE}}\,, \\
TT'_{\text{XTOR}} &= A T T'_{\text{CHEASE}}\,, \\
R_{\text{XTOR}} &= A R_{\text{CHEASE}}\,, \\
Z_{\text{XTOR}} &= A Z_{\text{CHEASE}}\,, \\
J_{\text{XTOR}} &= J_{\text{CHEASE}}/A^3\,,
\end{aligned}
$$

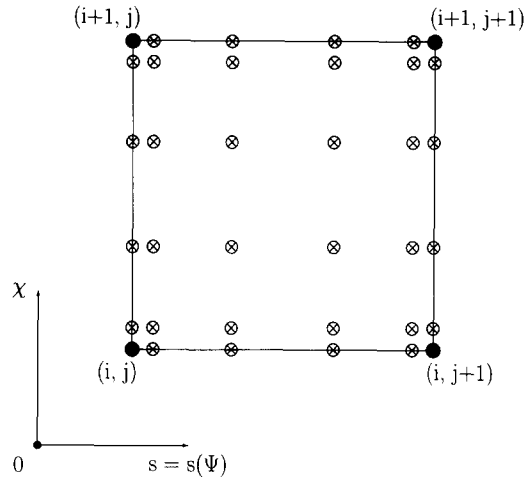

Fig. 3. Locations of PENN EQ's in a (s, χ) cell.

$$
\begin{aligned}
g^{ss}_{\mathrm{XTOR}} &= g^{ss}_{\mathrm{CHEASE}}/A^2 , \\
g^{s\chi}_{\mathrm{XTOR}} &= g^{s\chi}_{\mathrm{CHEASE}}/A^2 , \\
g^{\chi\chi}_{\mathrm{XTOR}} &= g^{\chi\chi}_{\mathrm{CHEASE}}/A^2 , \\
g^{\Phi\Phi}_{\mathrm{XTOR}} &= g^{\Phi\Phi}_{\mathrm{CHEASE}}/A^2 ,
\end{aligned} \tag{38}
$$

where $A = R_{\mathrm{maj}}/a$ is the aspect ratio of the torus. As for PEST and NOVA-W these transformations are not executed in CHEASE.

5.4.5. PENN

As XTOR, the global wave propagation code PENN [16] works in flux coordinates defined by Eq. (12) in the radial direction and the poloidal angle χ equal to the geometrical polar angle. PENN uses a variational formulation in terms of the EQ's,

$$
Q , \quad \frac{\partial Q}{\partial R} , \quad \frac{\partial Q}{\partial Z} , \quad \frac{\partial^2 Q}{\partial R^2} , \quad \frac{\partial^2 Q}{\partial Z^2} , \quad \frac{\partial^2 Q}{\partial R \partial Z} , \tag{39}
$$

where $Q = s(\Psi)$ or χ. This variational form is integrated with a 4-point Gaussian quadrature in both s and χ. The EQ's (39) are required at the Gauss quadrature points and the discretization cell nodes shown in Fig. 3. The derivatives of χ are expressed as

$$
\begin{aligned}
\ell \frac{\partial \chi}{\partial R} &= -(Z - Z_{\mathrm{mag}}) , \\
\ell \frac{\partial \chi}{\partial Z} &= R - R_{\mathrm{mag}} , \\
\ell^2 \frac{\partial^2 \chi}{\partial R^2} &= 2(Z - Z_{\mathrm{mag}})(R - R_{\mathrm{mag}}) , \\
\ell^2 \frac{\partial^2 \chi}{\partial Z^2} &= -2(Z - Z_{\mathrm{mag}})(R - R_{\mathrm{mag}}) , \\
\ell^2 \frac{\partial^2 \chi}{\partial R \partial Z} &= (Z - Z_{\mathrm{mag}})^2 - (R - R_{\mathrm{mag}})^2 ,
\end{aligned} \tag{40}
$$

where $\ell = (Z - Z_{\mathrm{mag}})^2 + (R - R_{\mathrm{mag}})^2$ and $(R_{\mathrm{mag}}, Z_{\mathrm{mag}})$ are the coordinates of the magnetic axis. No particular scaling is required for the EQ's. PENN needs an equilibrium solution with continuous second derivatives, which is one order higher than what is provided by the cubic Hermite elements. Therefore, the equilibrium solution is smoothed using bicubic spline functions in subroutine SMOOTH. The bicubic spline interpolation of the bicubic Hermite solution is documented in Appendix C. All the equilibrium quantities for PENN are computed in subroutine OUTPEN.

5.4.6. EQDSK input/output file

The file EQDSK, which is the standard input/output file for e.g. the equilibrium code EFIT [17], has the following format (all the variables in upper-case letters below are Namelist variables in CHEASE, see Section 6.3):

```
write(format=(a48,3i5)) ' comments and date', i3, NRBOX, NZBOX
write(format=(5e16.9 )) RBOXLEN, ZBOXLEN, R0EXP, RBOXLFT, zero
write(format=(5e16.9 )) RAXIS, ZAXIS, PSIAXIS, zero, B0EXP
write(format=(5e16.9 )) CURRENT, PSIAXIS, zero, RAXIS, zero
write(format=(5e16.9 )) ZAXIS, zero, zero, zero, zero
write(format=(5e16.9 )) (T_MKSA(i), i=1,NRBOX)
write(format=(5e16.9 )) (p_MKSA(i), i=1,NRBOX)
write(format=(5e16.9 )) (TT'_MKSA(i), i=1,NRBOX)
write(format=(5e16.9 )) (p'_MKSA(i), i=1,NRBOX)
write(format=(5e16.9 )) ((Ψ(R_i, Z_j), i=1,NRBOX),j=1,NZBOX)
write(format=(5e16.9 )) (q(i), i=1,NRBOX)
write(format=(2i5 )) npbound, nlimiter
write(format=(5e16.9 )) (rbound(i), zbound(i), i=1,npbound)
write(format=(5e16.9 )) (rlimiter(i), zlimiter(i), i=1, nlimiter)
```

where i3=3, zero=0., (RAXIS, ZAXIS) is the position of the magnetic axis, CURRENT is the total plasma current, PSIAXIS=$(\Psi_{\max} - \Psi_{\min})$. The poloidal flux is given for an equidistant (R, Z) mesh, with (NRBOX - 1) and (NZBOX - 1) intervals and so that

```
rmin = RBOXLFT
rmax = RBOXLFT + RBOXLEN
zmin = - ZBOXLEN / 2
zmax = ZBOXLEN / 2
```

The limiter boundary is set to this (R, Z) rectangle. The profiles T, p, TT' and p' are given for an equidistant mesh in Ψ with (NRBOX - 1) intervals. All the quantities are given in MKSA. Thus the CHEASE variables are transformed using the Namelist variables R0EXP and B0EXP specified in meters and Tesla, respectively, as follows:

$$R = R_{\mathrm{CHEASE}} * R0EXP\,,$$

$$Z = Z_{\mathrm{CHEASE}} * R0EXP\,,$$

$$\Psi = \Psi_{\mathrm{CHEASE}} * R0EXP^2 * B0EXP\,,$$

$$I = I_{\mathrm{CHEASE}} * R0EXP * B0EXP/\mu_0\,,$$

$$T = T_{\mathrm{CHEASE}} * R0EXP * B0EXP\,,$$

$$p = p_{\mathrm{CHEASE}} * B0EXP^2/\mu_0\,,$$

$$TT' = TT'_{\mathrm{CHEASE}} * B0EXP\,,$$

$$p' = p'_{\mathrm{CHEASE}} * B0EXP/(\mu_0 * R0EXP^2)\,,$$

$$\mu_0 = 4\pi \times 10^{-7}\,. \tag{41}$$

Before computing an equilibrium with CHEASE by reading experimental data in an EQDSK file (see Section 6.4.1), it is of most importance to check if the experimental equilibrium was obtained assuming $T_0 = 1$ or $T_{\text{edge}} = 1$ by comparing the value of R0EXP*B0EXP and T_0, and to prescribe the CHEASE Namelist parameter NTMF0 accordingly as described in Section 6.4.4.

5.5. *Iteration over p' for the ballooning optimization and the specification of the bootstrap current*

The ballooning optimization (BO) and the specification of the bootstrap current (SBC) are directed by subroutine BALLIT. These options require the mapping of the equilibrium into flux coordinates because the the stability criteria (16), (19) and (20) or the flux surface integrals (23) and (44) must be evaluated at every iteration over the p' profile (see Fig. 2).

For the ballooning optimization, the pressure profile is modified iteratively in subroutine PPRM according to the following algorithm:

- **Do** for $\Psi_j, j = 1, \ldots, N_\Psi$:
  $p'_{0,j}, j = 1, .., N_\Psi$, given as in Section 6.4.2.
  $\delta p'_j = p'_{0,j}$.
  $\lambda_{0,j} = 1$.
  $\delta\lambda_{0,j} = 0.1$.
- **Do** for $k = 0$ until convergence
  Do for a chosen set of poloidal flux surfaces $\Psi_j, j = 1, \ldots, N_\Psi$:
  (i) Solve the Grad–Shafranov equation (3) with a coarse discretization grid.
  (ii) Scale equilibrium solution according to Section 5.3.
  (iii) Compute the Ballooning and Mercier stability criteria (16), (19) for every Ψ_j.
  (iv) If flux surface j stable at step k and unstable at step $k+1$ or vice-versa: $\delta\lambda_{k+1,j} = \frac{1}{2}\delta\lambda_{k,j}$.
  If flux surface j stable at steps k and $k+1$: $\delta\lambda_{k+1,j} = \frac{6}{5}\delta\lambda_{k,j}$.
  (v) If flux surface j unstable at step $k+1$: $\lambda_{k+1,j} = \lambda_{k,j} - \delta\lambda_{k+1,j}$.
  If flux surface j stable at step $k+1$: $\lambda_{k+1,j} = \lambda_{k,j} + \delta\lambda_{k+1,j}$.
  (vi) $p'_{k+1,j} = \lambda_{k+1,j}\delta p'_j$.
  $r_{k+1,j} = |p'_{k+1,j} - p'_{k,j}|$.
  (vii) If $p'_{k+1,j} > 0$ then $p'_{k+1,j} = 0$.
  If $p'_{k,j} = 0$ and $p'_{k+1,j} > -5.10^{-3}$ then $p'_{k+1,j} = 0$.
  End Do
  Error = $\max(r_{k+1,j}$, for $j = 1, .., N_\Psi)$.
  if Error $< \epsilon$ exit do loop.
  $k = k + 1$.
  End Do
- Compute equilibrium with refined discretization mesh and optimized p' profile.

In order to prevent an uncontrolled excursion of the optimization when the equilibrium enters the second region of stability for ballooning modes (this usually occurs when the magnetic shear is small or negative, i.e. typically in the central region of the plasma), p' is restricted so that

$$|p'(\Psi)| < \lambda_1 |q'(\Psi)|\,, \tag{42}$$

where λ_1 is a user-defined constant.

Frequently, a ballooning optimized equilibrium is unstable against MHD modes of low toroidal mode number n [27]. Therefore CHEASE contains the option of rescaling the ballooning optimized p' profile by a user-defined factor λ_2. This can be used to find equilibria stable against local as well as global MHD modes.

For the SBC, the p' profile is adjusted iteratively in subroutine PPBSTR so that

$$p'_{k+1}(s) = C(s)\frac{\langle \boldsymbol{J}\cdot\boldsymbol{B}\rangle_{\rm bs}(s)}{\langle \boldsymbol{J}\cdot\boldsymbol{B}\rangle(s)}p'_k(s)\,, \tag{43}$$

where $\langle \boldsymbol{J}\cdot\boldsymbol{B}\rangle_{\rm bs}$ is given by Eq. (23) and

$$\langle \boldsymbol{J}\cdot\boldsymbol{B}\rangle = -Tp'(\Psi) - T'(\Psi)\langle B^2\rangle\,. \tag{44}$$

The total toroidal current from bootstrap is then computed as

$$I_{\rm bs} = \int_{\Omega}\frac{\langle \boldsymbol{J}\cdot\boldsymbol{B}\rangle_{\rm bs}}{\langle \boldsymbol{J}\cdot\boldsymbol{B}\rangle}j_\phi\,dS\,. \tag{45}$$

The BO and the SBC are performed with the coarse equilibrium discretization mesh defined in subroutine EQDIM and a reduced set of radial surfaces $s_{\rm opt}$ (the number of flux surfaces for the ballooning optimization NPPR is usually taken as 30, but this number can be modified in the Namelist). The Gaussian quadrature for the integration of (26) or the computation of the EQ's for the stability codes uses p' at different radial locations than the $s_{\rm opt}$ grid. In a similar manner as for the I^* or the $I_{||}$ profiles defined in Section 2.2, these quantities are interpolated with cubic spline functions on the $s_{\rm opt}$ mesh. For both types of automatic generation of the pressure profile, the iteration over p' is stopped when $\|p'_{k+1} - p'_k\| < \epsilon$.

5.6. I/O files

An overview of all I/O files of CHEASE is shown in Fig. 4, and their characteristics and contents are given in Table 5. These files can be subdivided into three categories. First, the files which intervene in the equilibrium calculation itself (Namelist EQDATA, NIN, NOUT, EXPEQ, EQDSK), second the equilibrium diagnostic files (Output, NUPLO) and last the files used as input by the codes linked to CHEASE. The Namelist EQDATA is described in details in Section 6.3 and the EXPEQ files in Sections 6.4.1 and 6.4.2.

The I/O files NIN and NOUT allow the user of CHEASE to take advantage of the equilibrium scaling laws described in Section 5.3. Every quantity required for the reconstruction of the converged solution of Eq. (26) is stored into the file NOUT before the rescaling of the equilibrium. This file can be reutilized for a subsequent equilibrium calculation which only differs from the first by another scaling in Section 5.3. If NOPT $\neq$ 0 in the Namelist, CHEASE reads an equilibrium from the file NIN, which is organized in an identical way to NOUT.

CHEASE produces the disk file EXPEQ.OUT which stores the equilibrium boundary and profile data in the same format as in EXPEQ (boundary and equilibrium profiles specified by a set of data points). However, different ways of current profile specification (TT', I^* or $I_{||}$) can be used on the two files. This makes it possible to switch between different current specifications, which is useful, e.g., when experimental equilibria are analyzed.

6. Directions for the users of CHEASE

6.1. Sizing the arrays

The dimensions of the arrays in CHEASE are controlled in every case by two different quantities. First, parameters are used to dimension arrays of the executable. These give upper limits for the dimensions of a run. The parameters have the syntax NPxx, where xx is a suffix which may contain one or more characters or numbers. Second, dimensioning variables are determined for every run (without recompilation) from the

Table 5
I/O disk files: characteristics and contents

File	Unit number	Def	Format	Stat	Content
Namelist	5	5	Namelist	def.	CHEASE Namelist EQDATA
Output	6	6	formatted	def.	
INP1	INP1	46	unformatted	def.	EQ's for NOVA-W or PEST
NIN	NIN	10	unformatted	old	read equilibrium; used only if NOPT $\neq 1$
NOUT	NOUT	11	unformatted	new	store equilibrium
MEQ	MEQ	4	unformatted	new	EQ's for ERATO
NDES	NDES	16	u.(ERATO)	new	plot EQ's for
			f.(MARS)	new	ERATO or MARS
NVAC	NVAC	17	unformatted	new	vacuum EQ's for ERATO
NSAVE	NSAVE	8	formatted	new	ERATO Namelist NEWRUN
EQU01	NO	21	formatted	new	EQ's for MARS
ETAVAC	NETVAC	23	formatted	new	vacuum EQ's for MARS
NUPLO	NUPLO	33	formatted	new	plot quantities
EXPEQ	NXIN	48	formatted	old	experimental equilibrium
EXPEQ.OUT	NXOUT	50	formatted	new	restore experimental data
NPENN	NPENN	49	unformatted	new	EQ's for PENN
OUTXTOR	NXTOR	37	unformatted	new	EQ's for XTOR
EQDSK	IEQDSK	38	formatted	new	EQDSK I/O file

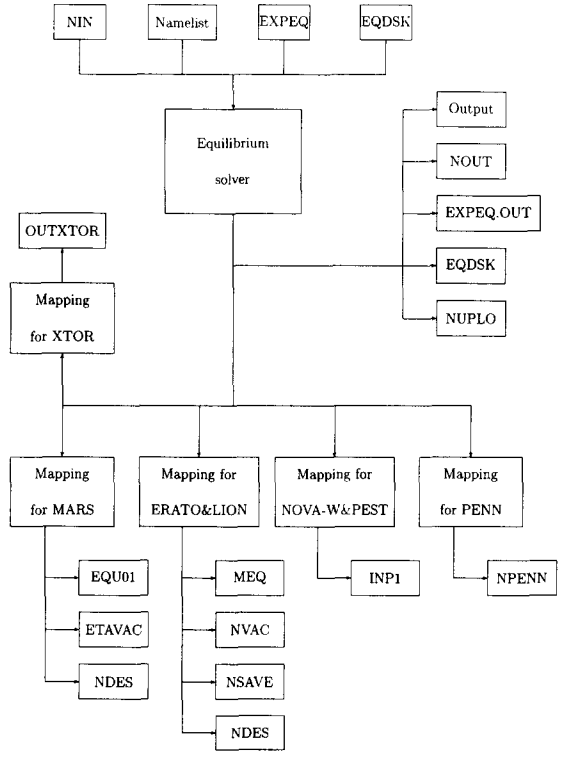

Fig. 4. I/O files of CHEASE.

Table 6
Dimensioning parameters

Parameter	Corresp. namelist variable	Definitions
MPSMAX	MSMAX	Number of poloidal Fourier modes for MARS.
NPBLC0	NBLC0	Number of integration constants χ_0 in the ballooning integral.
NPBPS	NBPS	Number of (R, Z) nodes for the definition of experimental plasma boundary.
NPCHI	NCHI	Number of χ-intervals for ERATO and ballooning stability calculation.
NPISO	NISO	Number of s-intervals to define $I^*(s)$ or $I_{\|\|}(s)$.
NPPSI	NPSI	Number of s-intervals for stability calculation.
NPS	NS	Number of σ-intervals for equilibrium calculation.
NPSMAX	NSMAX	Number of toroidal Fourier modes for MARS.
NPT	NT	Number of θ-intervals for equilibrium calculation.
NPTURN	NTURN	Number of 2π-intervals for ballooning integration.
NPV	NV	Number of radial vacuum s-intervals for MARS.
NPSGS	NSGAUS	Number of Gaussian quadrature points in σ-direction.
NPTGS	NTGAUS	Number of Gaussian quadrature points in θ-direction.
NPMGS	NMGAUS	Number of Gaussian quadrature points along constant-Ψ surface.
NPPSBAL	none	Number of Ψ surfaces where ballooning criterion is computed in parallel.
NPPSCUB	none	Number of Ψ surfaces traced in parallel.
MFLGERL	none	Controls extra memory space required by mappings for ERATO and LION.
MFLGMAR	none	Controls extra memory space required by mapping for MARS.
MFLGNVW	none	Controls extra memory space required by mappings for NOVA-W and PEST.
MFLGPEN	none	Controls extra memory space required by mapping for PENN.

Namelist input. The values of these variables are checked in subroutine COTROL, and the program automatically stops if one of them is larger than its corresponding parameter. A list and the description of all the dimensioning parameters and variables is given in Table 6. The quantities NPMGS, NPSGS and NPTGS and their corresponding variables are usually set to 4, which implies that 4 Gaussian quadrature points are used per integration interval for the computation of the variational form (26) and the flux surface integrals (6), (9), (15), etc. Extensive use of the code has shown that this is the best choice in terms of cpu time and memory requirements.

With the 6 last parameters in Table 6, the memory requirements of the code can be reduced. The worst (resp. best) vector performance is obtained when NPPSBAL = NPPSCUB = 1 (resp. NPPSI+1). The MFLGxxx parameters are flags which control the memory requirements for the different mappings produced by the code. If the user sets one of them to 0, every array required by the corresponding mapping is sized to 1. Default values for these parameters are NPPSBAL = NPPSCUB = 10 and MFLGxxx = 1.

Table 7
Include files: statement functions

Include file	Content
BNDIND.inc	horizontal band matrix indexes
CUCCCC.inc	cubic Lagrange interpolation
CUCDCD.inc	cubic Hermite interpolation
HERMIT.inc	1-D Hermite basis functions
QUAQDQ.inc	quadratic interpolation with 1 derivative
QUAQQQ.inc	quadratic Lagrange interpolation
SOLOV.inc	analytic Solovev equilibrium

Table 8
Include files: common blocks

Include file	Content	Memory requirement (64 bit Words)
COMDIM.inc	Dimensioning parameters	67
COMBAL.inc	Quantities for ballooning calculation	4*NPPSBAL*(NPTURN*NPCHI+1)+NPBLC0*(43*NPPSI+1)
COMBLA.inc	Equilibrium matrix and RHS	16*NPT*(NPT+5)*(NPS+1)
COMBND.inc	Plasma surface parameters	11*NPBPS+40
COMCON.inc	Control parameters	33
COMDAT.inc	Equilibrium Namelist	
COMEQD.inc	EQDSK output quantities	(NPPSI+1)**2+5
COMERA.inc	ERATO equilibrium quantities	34*NPCHI*(NPPSI+1)
COMESH.inc	Mesh quantities	2*NPCHI*(NPMGS+2)+5*NPPSI+2*(NPS+NPV+2*NPT)+163
COMETA.inc	MARS equilibrium quantities	19*NPT*NPMGS+(NPPSI+1)*(MPSMAX*(52+24*NPSMAX))
COMINT.inc	Quantities for integration of equilibrium variational form	NPSGS*NPTGS*{4*NPT*(4*NPS+1)+3}+NPS*(NPT+20)
COMIOD.inc	I/O control parameters	17
COMISO.inc	Quantities required for evaluation of flux surface integrals	2*(NPT+1)*NPMGS*(16*NPPSI+17)
COMLAB.inc	Labels for equilibrium	5
COMMAP.inc	Values of flux surface integrals at equilibrium cell boundaries	2*NPCHI*(NPPSI+NPMGS+18*NPT+10)
COMNUM.inc	Dimensioning variables	31
COMPHY.inc	Physical quantities	188
COMPLO.inc	Plot quantities	(NPPSI+1)*(NPCHI+8)+1
COMSOL.inc	Control quantities for equilibrium solver	13
COMSUR.inc	Flux surface quantities	119*(NPPSI+1)
COMVAC.inc	Vacuum quantities for MARS	16*(NPV+1)*MPSMAX
COMVEV.inc	Quantities computed for Solovev equilibria only	11*NPT*(NPS+1)+4
NEWRUN.inc	ERATO Namelist	
PROCESS.inc	Free deck for IBM extended memory commands	

6.2. *Include files*

Commons and statement functions are used extensively throughout CHEASE. All commons and statement functions are coded in separate files and inserted at the compilation with include statements. Therefore, a new variable can easily be added to a common by modifying the appropriate include file and the include file COMDIM.inc (where the common length are specified for the initializations in subroutine CLEAR at the beginning of every run). The include files containing statement functions are listed in Table 7, and those containing common blocks are shown in Table 8. Table 8 also gives the dimensions of all the commons in terms of the dimensioning parameters in Table 6.

6.3. *Namelist variables*

The quantities characterizing an equilibrium can be modified in the Namelist. The Namelist statement is not standard FORTRAN, but is implemented in most computers. CHEASE reads the Namelist from input channel 5, which must have the following form:

```
****
**** 4 character lines of maximum length 80
```

Table 9
Namelist EQDATA

Variable	Type	Def.	Definitions
AFBS,AFBS2	RA's	10*0.	Coefficients used for definition of bootstrap current profile; only used if NBSOPT = 1 or 2
AP,AP2	RA's	10*0.	Coefficients used for definition of p'(s) profile
APLACE,BPLACE, CPLACE,DPLACE, EPLACE	RA's	10*0.	Localizations where respectively stability-s, profile-s, σ, θ and χ mesh is densified
ASPCT	R	1/3	Inverse aspect ratio of plasma
AT,AT2,AT3,AT4	RA's	10*0.	Coefficients used for definition of TT'(s), $I^*(s)$ and $I_{\|\|}(s)$ profiles
AWIDTH,BWIDTH, CWIDTH,DWIDTH EWIDTH	RA's	10*0.	Width of mesh densification at APLACE, BPLACE, CPLACE, DPLACE, EPLACE, respectively
BEANS	R	0.	Indentation of plasma surface
BSFRAC	R	0.5	Fraction of bootstrap current; only used if NBSOPT = 1 and NBSTRP = 2
B0EXP	R	1.	Magnetic field at magnetic axis in Tesla for EQDSK
CETA	R	0.	Boundary parameter (see Eqs. (49), (50))
CFBAL	R	1.	Coefficient which limits pressure profile when NBLOPT = 1 or NBSOPT = 1
COMPTYP	A	cray	Selects time and date routines specific to Cray, Sun or Silicon Graphics
CPRESS	R	1.	Coefficient which rescales ballooning optimized or experimental pressure profile
CQ0	R	0.75	Solovev equilibria: safety factor at magnetic axis; only used with NSURF = 1
CSSPEC	R	0.	s or ρ value where q is specified when equilibrium is scaled; only used if NCSCAL = 1 or 3
CURRT	R	0.5	Total plasma current, CHEASE normalization
DELTA	R	0.	only used if NSURF = 4 (see Eq. (50))
ELONG	R	1.	Elongation of plasma cross section
EPSLON	R	10^{-9}	Precision required for equilibrium solution
ETAEI	R	3/2	$\eta_i = d\log T_e / d\log n$; only used if NBSRTP = 1
GAMMA	R	5/3	Ratio of specific heat
PREDGE	R	0.	Pressure at plasma boundary of experimental equilibrium
PSISCL	R	1.	Fraction of equilibrium poloidal flux kept for stability calculation
QSPEC	R	1.	Specified q-value for scaling of equilibrium; only used if NCSCAL = 1 or 3
QPLACE	RA	10*0.	q-values where stability s-mesh is densified

****
****
&EQDATA
List of Namelist variables for CHEASE.
&END
&NEWRUN
List of Namelist variables for ERATO.
&END

The 4 character lines are used to label the run and must imperatively exist at the beginning of input channel 5, even if they remain blank. They will appear at the beginning of the output file on channel 6 and in the plots. An example of a run output is given in Section 7. A list of all Namelist variables, including their description, default value and type is given in Tables 9–12. Directions for the specifications of these variables are given in the next section.

Table 10
Namelist EQDATA

Variable	Typ.	Def.	Definitions
QWIDTH	RA	10*0.	As AWIDTH, but relative to QPLACE
RBOXLEN	R	1.6	R length of (R,Z) box used to save equilibrium in EQDSK
RBOXLFT	R	1.5	$R_{\min}$ of (R,Z) box used to save equilibrium in EQDSK
RC	R	1.	Centre of plasma boundary (if NSURF=6)
REXT	R	1.	Vacuum radius for ERATO and MARS
RELAX	R	0.	Under-relaxation parameter used if magnetic axis converges slowly
RNU	R	0.	Only used if NSURF=4 (see Eq. (50))
RZION	R	1.	Ion charge in bootstrap current formula; only used if NBSTRP=1
R0,RZ0	R's	1.,0.	(R, Z) position of equilibrium mesh centre
R0W,RZ0W	R's	1.,0.	(R, Z) position of MARS vacuum mesh centre
R0EXP	R	1.	Major radius of magnetic axis in meters for EQDSK
SGMA	R	0.	Boundary parameter (see Eqs. (49), (50))
SOLPDA,SOLPDB, SOLPDC,SOLPDD, SOLPDE	R's	5*0.	Fraction of mesh, corresponding, respectively, to APLACE, BPLACE, CPLACE, DPLACE, and EPLACE, which is kept equidistant
THETA0	R	0.	Angular position of X-point if plasma surface is defined with NSURF=4 (see Eq. (50))
TRIANG	R	0.3	Triangularity of plasma cross section
TRIPLT	R	0.	Only used if NSURF=4 (see Eq. (50))
XI	R	0.	Boundary parameter (see Eq. (49))
ZBOXLEN	R	1.5	Z length of (R,Z) box used to save equilibrium in EQDSK
MSMAX	I	10	Number of poloidal modes for MARS
NANAL	I	0	Flag for analytic Solovev equilibrium
NBAL	I	1	Flag for solution of ballooning stability
NBLC0	I	16	Number of values used for integration constant χ_0 in ballooning stability criterion
NBLOPT	I	0	Flag for ballooning optimization
NBPSOUT	I	300	Number of plasma boundary points saved in EXPEQ.OUT
NBSFUN	I	1	Selects type of functional to define bootstrap current
NBSOPT	I	0	Flag for bootstrap specification
NBSTRP	I	1	Selects relation between parallel and bootstrap current

6.4. *Setting up an equilibrium*

6.4.1. *Specification of the plasma boundary*

Several methods are implemented in CHEASE for the specification of the plasma boundary $\delta\Omega$ (see Fig. 1). The boundary calculation is executed in subroutine BOUND, and the different boundary definitions are selected by means of the Namelist parameter NSURF.

The first family of solutions is obtained by setting NSURF=1. This choice leads to a class of the analytic Solovev equilibria [31]. These equilibria are characterized by

$$P(\Psi) = -\frac{1+\kappa^2}{\kappa R_0^3 q_0}\Psi,$$
$$T(\Psi) = T_0 = 1 \tag{46}$$

(so that the Grad–Shafranov equation becomes linear in Ψ). Ψ is given by

$$\Psi = \frac{\kappa}{2R_0^3 q_0}\left(\frac{R^2Z^2}{\kappa^2} + \tfrac{1}{4}(R^2 - R_0^2)^2 - a^2R_0^2\right). \tag{47}$$

Table 11
Namelist EQDATA

Variable	Typ.	Def.	Definitions
NCHI	I	100	Number of poloidal nodes for ballooning, ERATO stability meshes and MARS mapping
NCSCAL	I	2	Selects equilibrium scaling law
NDIFPS	I	1	Flag for automatic packing of stability s-mesh
NDIFT	I	1	Selects automatic equilibrium θ-mesh packing
NEGP	I	−1	$\|\nabla\Psi\|$ exponent in flux coordinate Jacobian J
NEQDSK	I	0	Flag for reading EQDSK file. Set NSURF = 6 & NEQDSK = 1 when EXPEQ = EQDSK
NER	I	1	R exponent in flux coordinate Jacobian J
NFFTOPT	I	0	Flag for computing MARS EQ's with FFT's
NFUNC	I	1	Selects functional form used to define $TT'(s)$, $I^*(s)$ or $I_{\|\|}(s)$ profile
NIDEAL	I	0	Selects mapping
NINMAP	I	20	Max. number of iterations over current profiles
NINSCA	I	20	Max. number of iterations over nonlinearity
NIPR	I	1	Selects functional form of $s(\Psi)$ for $I^*(s)$ or $I_{\|\|}(s)$ profiles if NSTTP = 2
NISO	I	100	Number of s-intervals to define $I^*(s)$ or $I_{\|\|}(s)$
NMESHA,NMESHB, NMESHC,NMESHD, NMESHE	I's	0.	Flags for densification of mesh corresponding, respectively, to APLACE, BPLACE, CPLACE, DPLACE, EPLACE
NMGAUS	I	4	Number of Gaussian quadrature points used for integrations along constant flux surfaces
NOPT	I	0	Flag for reading stored equilibrium
NPLOT	I	1	Flag for production of plot quantities
NPOIDA,NPOIDB, NPOIDC,NPOIDD, NPOIDE,NPOIDQ	I's	0.	Number of locations where mesh corresponding to APLACE, BPLACE, CPLACE, DPLACE, EPLACE, QPLACE is densified
NPP	I	0	Selects functional form of $s(\Psi)$ for $p'(s)$ profile
NPPFUN	I	1	Selects functional form of $p'(s)$ profile
NPPR	I	30	Number of s-values where $p'(s)$ is modified during ballooning optimization or specification of bootstrap current density
NPROPT	I	1	Selector for reading experimental TT', I^* or $I_{\|\|}$
NPRPSI	I	0	Flag for printing equilibrium solution
NPSI	I	100	Number of radial stability-s intervals
NRBOX	I	33	Number of R points used to save equilibrium in EQDSK
NRSCAL	I	0	Flag for rescaling distances so that $R_{\text{mag}} = 1$

In Eq. (46) $\boldsymbol{K}$ denotes the elongation, R_0 the major plasma radius (which is equal to 1 per default in CHEASE), a the minor radius of the torus and q_0 is the safety factor at the magnetic axis. The plasma boundary is given by

$$R = R_0(1 + 2\epsilon_a\cos\theta)^{1/2},$$
$$Z = \frac{R_0\epsilon_a\boldsymbol{K}\sin\theta}{(1 + 2\epsilon_a\cos\theta)^{1/2}}. \tag{48}$$

These equilibria are determined by $\epsilon_a = a/R_0$ = ASPCT, R_0 = RC, $\boldsymbol{K}$ = ELONG and q_0 = CQ0 which are all prescribed in the Namelist. The first test case in [1] is obtained with ASPCT = 1/3, ELONG = 1 and CQ0 = 3/4.

Second, the boundary can be specified by a generalized form of the INTOR formula,

$$R = R_0 + R_0\tilde{a}f(\theta)\cos(\theta + \delta\sin\theta - \xi\sin2\theta)(1 + b\cos\theta),$$
$$Z = R_0\tilde{a}\boldsymbol{K}\sin(\theta + \zeta\sin2\theta). \tag{49}$$

Here

Table 12
Namelist EQDATA

Variable	Typ.	Def.	Definitions
NS	I	40	Number of radial equilibrium-σ intervals
NSGAUS	I	4	Number of Gaussian quadrature points used for equilibrium integration in radial direction
NSMOOTH	I	1	Flag for bicubic spline smoothing of equilibrium solution
NSOUR	I	0	If NFUNC = 1 or NPPFUN = 1, degree of polynomial used to define corresponding profile
NSTTP	I	1	Selects $TT'(s)$, $I^*(s)$ or $I_{\|\|}(s)$ profile
NSURF	I	1	Selects shape of plasma surface
NSYM	I	0	Flag for symmetric/asymmetric versionof ERATO
NT	I	40	Number of poloidal equilibrium-θ intervals
NTCASE	I	0	Selects pre-defined test cases
NTGAUS	I	4	Number of Gaussian quadrature points used for equilibrium integration in poloidal direction
NTNOVA	I	64	Number of poloidal-χ intervals used by NOVA-W, PEST and XTOR
NTEST	I	0	Check accuracy of numerical Solovev equilibrium
NTMF0	I	0	Selects equilibrium scaling law for T
NTURN	I	10	Number of poloidal turns left and right of χ_0 for integration of ballooning integral
NV	I	20	Number of radial s-intervals in MARS vacuum
NVEXP	I	0	Flag for exponential packing of MARS vacuum s-mesh close to plasma surface
NZBOX	I	65	Number of Z points used to save equilibrium in EQDSK

$$f(\theta) = 1 + \frac{\tilde{\sigma}}{1 + \left[\sin^2\left(\frac{1}{2}(\theta - \theta_0)\right) + \tilde{\xi}\right]^{\nu}/\Delta} \tag{50}$$

can be used to prescribe a "bump" on the plasma surface [32]. The standard INTOR formula is obtained when $\tilde{\sigma} = \xi = \zeta = 0$. The minor radius $\tilde{a}$ in Eq. (49) is obtained by demanding that the inverse aspect ratio

$$\epsilon_a = \frac{a}{R_0} = \frac{R_{\max} - R_{\min}}{R_{\max} + R_{\min}} \tag{51}$$

is equal to the Namelist parameter ASPCT. Here, $R_{\min}$ and $R_{\max}$ denote the minimum and the maximum R values of the plasma boundary, respectively. The definition (49) groups together two different options in CHEASE.

- If NSURF=2, then $f(\theta) \equiv 1$, and therefore plasma boundaries defined by Eq. (49) lead to equilibria symmetric about the $Z = 0$ plane. In this case, the plasma boundary is entirely determined by the Namelist parameters RC = R_0, TRIANG = δ, ELONG = κ, ASPCT = a/R_0, BEANS = b, XI = ξ and CETA = ζ.
- If NSURF=4, $\xi \equiv 0$ and $\zeta \equiv 0$, and the boundary shape is in general asymmetric about the $Z = 0$ plane. The following Namelist variables determine the plasma boundary: RC = R_0, TRIANG = δ, ELONG = κ, ASPCT = a/R_0, BEANS = b, THETA0 = θ_0, RNU = ν, XI = $\tilde{\xi}$, SGMA = $\tilde{\sigma}$ and DELTA = Δ.

Last, the plasma boundary is specified by a set of experimental (R, Z) coordinates if NSURF=6. If NEQDSK=0, these data are read in file EXPEQ in the following format:

```
read(48,998) np
read(48,999) (rbound(i), zbound(i), i=1,np)
998 format(i5)
999 format(2e18.8)
```

The boundary information required by CHEASE is computed with cubic spline interpolations of these data in subroutines BNDSPL and BOUND. If NSURF=6 and NEQDSK=1, the data are read in an EQDSK file as described in Section 5.4.6.

Table 13
Definition of p' with NPPFUN = 2

Definition	NPP = 1	NPP = 2
t	$1 - \frac{\Psi}{\Psi_{\min}}$	$\left(1 - \frac{\Psi}{\Psi_{\min}}\right)^{1/2}$
t_1	AP(1)	AP2(1)
t_2	AP(2)	AP2(2)
$-p'(t_2)$	AP(3)	AP2(3)
$-p''(t_1)$	AP(4)	AP2(4)
$-p'(0)$	AP(5)	AP2(5)
$-p'(1)$	AP(6)	AP2(6)
$-p''(t_2)$	AP(7)	AP2(7)
free	AP(8)	AP2(8)
free	AP(9)	AP2(9)
free	AP(10)	AP2(10)

6.4.2. Specification of the pressure and current profiles

The analytic forms used to specify the current profile are functions of the normalized flux $1 - \Psi/\Psi_{\min}$. The profiles are computed in subroutine PREMAP. The p' profile is calculated in subroutine PPRIME, and the TT', I^* and $I_{||}$ profiles in subroutine PRFUNC. If I^* or $I_{||}$ are specified, or the profiles are given in terms of sets of experimental data, the values of p' and/or TT' required for the integration of the source term in Eq. (26) are obtained by cubic spline interpolations. The functional forms used for the specification of these profiles can easily be modified by the user.

Pressure profile

Three different functional forms are currently available for the definition of the p' profile.

- First, p' can be specified as a polynomial of degree NSOUR - 1 by setting NPPFUN = 1,

$$p'(\Psi) = \sum_{i=1}^{NSOUR} AP(i) \left(1 - \frac{\Psi}{\Psi_{\min}}\right)^{(i-1)} . \tag{52}$$

NPPFUN, NSOUR and AP are Namelist variables.

- Second, if NPPFUN = 2, p' is given by a combination of polynomials on three different subintervals of [0, 1], such that p' and its first derivative are continuous. Let t stand for $1 - \Psi/\Psi_{\min}$ if NPP = 1, and for $(1 - \Psi/\Psi_{\min})^{1/2}$ if NPP = 2 in the following. The functions defined by polynomials in three sections are defined as follows. For $t \in [0, t_1]$, $p'(t)$ is linear, for $t \in [t_1, t_2]$, $p'(t)$ is cubic and for $t \in [t_2, 1]$, $p'(t)$ is quadratic. In the present version of CHEASE, $p'(t)$ can be specified as a superposition of two such functions composed of polynomials in three sections. Table 13 shows an overview of the Namelist parameters used for the specification of the two functions.
- Third, if NPPFUN = 3, p' is given as

Table 14
Definition of p' with NPPFUN = 3

Definition	Namelist variable
p_0	AP(1)
α	AP(2)
β	AP(3)

$$p'(\Psi) = p_0 \left[1 - \left(1 - \frac{\Psi}{\Psi_{\min}}\right)^{\beta}\right]^{\alpha} . \tag{53}$$

The Namelist parameters used for the definition (53) are given in Table 14.

- If NPPFUN = 4, p' is given by a set of data points. If NEQDSK = 0, these data are read in the file EXPEQ in the following format:

```
read(48,998) NPPF1, NSTTP
read(48,999) (FCSM(i), i=1,NPPF1)
if (NPPFUN.Eq. 4) read(48,999) (RPPF(i), i=1,NPPF1)
if (NFUNC.Eq. 4) read(48,999) (RFUN(i), i=1,NPPF1)
998 format(i5)
999 format(1e18.8)
```

If NSURF = 6 (i.e. the plasma boundary is specified by a set of points), the data for the profiles must be stored **after** the boundary coordinates in EXPEQ. NPPF1 is the number of radial $s = (1 - \Psi/\Psi_{\min})^{1/2}$ grid points for which the profiles are specified. These s-values are stored in FCSM and the corresponding p' values in RPPF. The contents of RFUN will be described later in this Section. If NEQDSK = 1, the data for p' are read in file EQDSK (see Section 5.4.6).

Current profiles

The choice of the current density specification (see Section 2.2) is controlled by means of the Namelist parameter NSTTP in subroutine ISOFUN. If NSTTP = 1, 2 or 3, j_ϕ is specified in terms of TT', I^* or $I_{||}$, respectively. In the following, let F stand for TT', I^* or $I_{||}$. Several functional forms are implemented in CHEASE for F. The choice is controlled by the Namelist parameters NFUNC and NIPR and the profiles are computed in subroutine PRFUNC.

- If NFUNC = 1, F is given as a polynomial of degree NSOUR - 1 and with coefficients AT(i), i = 1,...,NSOUR, in a similar way as p' in Eq. (52).
- If NFUNC = 2, F is a superposition of three functions which all are composed of polynomials in three sections, with a continuous first derivative. The functions for the current profiles are defined differently than those for p'. Let t stand for $1 - \Psi/\Psi_{\min}$ if NIPR = 1, $(1 - \Psi/\Psi_{\min})^{1/2}$ if NIPR = 2, and $(1 - \Psi/\Psi_{\min})^{1/4}$ if NIPR = 3. For $t \in [0, t_1]$, $F(t)$ is quadratic, for $t \in [t_1, t_2]$, $F(t)$ is cubic and for $t \in [t_2, 1]$, $F(t)$ is linear. Moreover, a Gaussian defined by

$$F(t) = h_g \exp\left[-\left(\frac{t - c_g}{w_g}\right)^2\right] , \tag{54}$$

where $t = 1 - \Psi/\Psi_{\min}$ can be added to the previous 3 functions by setting NIPR = 4. Table 15 shows an overview of the Namelist parameters used for the specification of these functionals.

- If NFUNC = 3, F is given in an analogous way as Eq. (53). In that case, the F profile is specified as p' in Table 14 with the Namelist array AT instead of AP.
- Last, if NFUNC = 4, F is given by a set of experimental data, which are read from the file EXPEQ into the array RFUN with the same format as discussed for p' if NEQDSK = 0, and from the file EQDSK if NEQDSK = 1 (see Section 5.4.6).

6.4.3. Control parameters for the ballooning optimization and the specification of the bootstrap current

The ballooning optimization (BO) is activated by setting the Namelist variable NBLOPT to 1. The initial p' profile is specified as in Section 6.4.2. CFBAL is the coefficient λ_1 which limits p' according to Eq. (42). The converged optimized equilibrium is stored into the disk file NOUT. It is emphasized here that the ballooning

Table 15
Definition of F with NFUNC = 2

Definition	NIPR = 1	NIPR = 2	NIPR = 3	NIPR = 4
t	$1-\frac{\psi}{\psi_{\min}}$	$\left(1-\frac{\psi}{\psi_{\min}}\right)^{1/2}$	$\left(1-\frac{\psi}{\psi_{\min}}\right)^{1/4}$	$1-\frac{\psi}{\psi_{\min}}$
t_1	AT(1)	AT2(1)	AT3(1)	
t_2	AT(2)	AT2(2)	AT3(2)	
$F(0)$	AT(3)	AT2(3)	AT3(3)	
$F'(0)$	AT(4)	AT2(4)	AT3(4)	
$F'(t_1)$	AT(5)	AT2(5)	AT3(5)	
$F(1)$	AT(6)	AT2(6)	AT3(6)	
$F'(t_2)$	AT(7)	AT2(7)	AT3(7)	
free	AT(8)	AT2(8)	AT3(8)	
free	AT(9)	AT2(9)	AT3(9)	
free	AT(10)	AT2(10)	AT3(10)	
t_g				AT4(1)
w_g				AT4(2)
h_g				AT4(3)

stability is affected by the equilibrium transformations in Section 5.3. However, in order to obtain an equilibrium globally stable to every MHD modes if the ballooning optimized equilibrium is not, it is possible to recompute an equilibrium with the optimized p' profile rescaled by a factor λ_2 = CPRESS.

For the computation of an equilibrium with a specified bootstrap current (SBC), the Namelist parameter NBSOPT must be 1. At present, for that purpose two different methods are implemented into CHEASE. First, if NBSTRP = 1, the converged equilibrium satisfies Eq. (43) so that the bootstrap fraction $\langle \mathbf{J}\cdot\mathbf{B}\rangle_{\rm bs}/\langle \mathbf{J}\cdot\mathbf{B}\rangle$, specified by Namelist parameter BSFRAC, is independent of s. Second, if NBSTRP = 2, the fraction of bootstrap current is a function $C(s)$ which is specified in the same way as p' in Section 6.4.2, except that the Namelist arrays AFBS and AFBS2 are used for that purpose instead of AP and AP2.

6.4.4. Control parameters for the equilibrium transformation rules

The two equilibrium scaling rules in Section 5.3 are directed by the Namelist parameters NTMF0, NCSCAL, CURRT, CSSPEC and QSPEC.

- The T profile can be modified so that T is 1 either at the magnetic axis or at the plasma boundary. This is controlled by NTMF0, which must be set to 1 for T_0 = 1 and to 0 for $T_{\rm edge}$ = 1. The modification of T is applied with one of the two next transformation rules.
- The total plasma current can be specified. This is activated if NCSCAL = 2. The equilibrium is rescaled in such a manner that the total current is equal to CURRT.
- The safety factor q given by Eq. (6) is specified at a certain plasma radius, i.e. q(CSSPEC) = QSPEC. Two options are implemented at present. First, if NCSCAL = 1, CSSPEC is a radial coordinate defined by Eq. (12), and second if NCSCAL = 3, CSSPEC is a generalized radius ρ defined in Table 1.

ERATO and LION require T_0 = 1 and $R_{\rm mag}$ = 1. The first condition is satisfied if NTMF0 = 1 and the second if the Namelist parameter NRSCAL is set to 1. For XTOR, NRSCAL must always be 0, i.e. R_0 = 1, and NTMF0 = 1.

6.4.5. Control parameters for the mesh densification

The meshes used for the resolution of the equilibrium equation (3) or by the different codes linked to CHEASE can be densified locally (usually referred to as "packing"). These mesh-densifications are classified into two categories. First, densifications which allow the packing of the different meshes at user-defined locations. The method used for this purpose is presented in [9], Section 6.2. Table 16 gives an overview of the

Table 16
Namelist parameters for "manual" mesh densifications

Ref. [9], Section 6.2	Stability-s	s for I^* or $I_{\|\|}$	σ	θ	χ
NMESH	NMESHA	NMESHB	NMESHC	NMESHD	NMESHE
$\frac{3+\text{NSING}}{3(1+\text{NSING})}$	SOLPDA	SOLPDB	SOLPDC	SOLPDD	SOLPDE
NSING	NPOIDA	NPOIDB	NPOIDC	NPOIDD	NPOIDE
x_j	APLACE	BPLACE	CPLACE	DPLACE	EPLACE
Δ_j	AWIDTH	BWIDTH	CWIDTH	DWIDTH	EWIDTH
	[0; 1]	[0; 1]	[0; 1]	$[-\pi;\pi]$	$[-\pi;\pi]$

different Namelist parameters involved in these mesh densifications. The first column relates our nomenclature for the control parameters to the one used in [9]. The last row shows the interval of definition used for the specification of A-EPLACE. The parameters SOLPDA-E control the fraction of the mesh points which remain undensified. The θ and χ-meshes are densified with "periodic Lorentzians" defined by

$$f(x) = \frac{1}{\Delta_j^2 + \sin^2\left((x - x_j)/2\right)}. \tag{55}$$

Moreover, different automatic mesh densifications are implemented in CHEASE. First, the stability-s mesh can be densified so that the generalized radius ρ defined in Table 1 is equidistant. This operation is executed if the Namelist parameter NDIFPS is set to 1. Second, the equilibrium θ-mesh can be densified so that the poloidal flux area is constant in every θ interval by setting the Namelist parameter NDIFT to 1. If NDIFT = 2, the equilibrium θ-mesh is densified in such a manner that the arc-length is constant in every θ-interval at the plasma surface.

Last, an option exists for densifying the stability s-grid at user-selected values of the safety factor. This is particularly useful in MHD stability studies where the solution often varies rapidly at certain rational q-values. This densification is performed if NMESHA = 2. In that case, the stability s-mesh is packed at NPOIDQ locations specified in the Namelist array QPLACE, and the s-width of every of these packings is specified in the Namelist array QWIDTH. The fraction of undensified mesh is still given by SOLPDA in that case.

7. Run output, convergence test and performances

```
 C  CCCCCCCC      H        HH      E  EEEEEEEE      A  AAAAAAAA      S  SSSSSSSS      E  EEEEEEEE
 CC  CCCCCCCC     HH       HHH     EE  EEEEEEEE     AA  AAAAAAAA     SS  SSSSSSSS     EE  EEEEEEEE
 CCC  CCCCCCC     HHH      HHH     EEE  EEEEEEE     AAA  AAAAAAA     SSS  SSSSSSS     EEE  EEEEEEE
 CCC      CCC     HHH      HHH     EEE              AAA      AAA     SSS              EEE
 CCC              HHH      HHH     EEE              AAA      AAA     SSS              EEE
 CCC              HHHHHHH  HHH     EEEEEEE          AAAAAAA  AAA     SSSSSSSSSSS      EEEEEEE
 CCC              HHHHHHHH HHH     EEEEEEEE         AAAAAAAA AAA     SSSSSSSSSSSS     EEEEEEEE
 CCC              HHHHHHHHHHH     EEEEEEEEE        AAAAAAAAAAA     SSSSSSSSSSSS     EEEEEEEEE
 CCC              HHH      HHH     EEE              AAA      AAA             SSS      EEE
 CCC      CCC     HHH      HHH     EEE              AAA      AAA             SSS      EEE
 CCCCCCCCCCCC     HHH      HHH     EEEEEEEEEEE      AAA      AAA     SSSSSSSSSSS      EEEEEEEEEEE
 CCCCCCCCCCCC     HHH      HHH     EEEEEEEEEEE      AAA      AAA     SSSSSSSSSSS      EEEEEEEEEEE
  CCCCCCCCCC       HH       HH      EEEEEEEEEE       AA       AA      SSSSSSSSS       EEEEEEEEEE

                CUBIC HERMIT ELEMENT AXISYMETRIC STATIC EQUILIBRIUM
***
***
*** Up-down asymmetric test case
***

NAMELIST VARIABLES
```

```
******************

&EQDATA
  NSURF=4,NSYM=0,EPSLON=1.E-8,
  ASPCT=.27400000,DELTA=0.5,ELONG=1.35,RNU=0.45,XI=8.E-4,THETA0=-1.57079633,SGMA=1.2,
  NS= 30,NT= 30,NISO=100,NDIFT=1,
  NMESHD=1,NPOIDD=1,SOLPDD=.6,
  DPLACE(1)=-1.57079633,DWIDTH(1)=0.157079633,
  NCHI=100,NPSI=15,NEGP=-1,NER=1,NDIFPS=0,
  NMESHA=0,
  NINMAP=50,NINSCA=50,
  NFUNC=2,NPPFUN=2,NSTTP=2,NPP=1,NIPR=1,
  AT(1) = 0.16, 1.0,  1.0, -1.1,  -1.1,
  AP(1) = 0.3, 0.5,  0.4, 0.0, 0.4, 0.0, 0.0,
  NOPT=0,NIDEAL=0,
  NCSCAL=1,NRSCAL=0,NTMFO=1,CURRT=0.22,
  QSPEC=1.00000,CSSPEC=0.33,
  NBAL=1,NBLCO=1,
  NPLOT=1,&END

 NS =   24   NT =   24   NISO =  100   NPSI =   15   NCHI =  100   NINSCA =   50   NINMAP =   50

SIGMA - MESH
   0.0000E+00   4.1667E-02   8.3333E-02   1.2500E-01   1.6667E-01   2.0833E-01   2.5000E-01   2.9167E-01
   3.3333E-01   3.7500E-01   4.1667E-01   4.5833E-01   5.0000E-01   5.4167E-01   5.8333E-01   6.2500E-01
   6.6667E-01   7.0833E-01   7.5000E-01   7.9167E-01   8.3333E-01   8.7500E-01   9.1667E-01   9.5833E-01
   1.0000E+00
THETA - MESH
  -2.5941E-01   2.9494E-01   8.3981E-01   1.2934E+00   1.6929E+00   2.1173E+00   2.6295E+00   3.1969E+00
   3.6881E+00   4.0274E+00   4.2440E+00   4.3865E+00   4.4867E+00   4.5626E+00   4.6243E+00   4.6778E+00
   4.7277E+00   4.7793E+00   4.8372E+00   4.9068E+00   4.9960E+00   5.1194E+00   5.3028E+00   5.5890E+00
   6.0238E+00
RHOS - MESH
   2.8223E-01   2.7343E-01   3.0185E-01   3.3758E-01   3.4493E-01   3.1699E-01   2.8086E-01   2.7404E-01
   3.0488E-01   3.5141E-01   3.9407E-01   4.2829E-01   4.5582E-01   4.7909E-01   4.9973E-01   5.1726E-01
   5.2159E-01   5.0700E-01   4.8725E-01   4.6513E-01   4.3950E-01   4.0815E-01   3.6871E-01   3.2228E-01
   2.8223E-01

POSITION OF THE CALCULATION MESH CENTER : RO =   1.00000000E+00   ZO =   0.00000000E+00

     PSIMIN =  -1.836534545503E-02   RMAG =     1.021597404506E+00   ZMAG =    -3.719708298367E-02
                                                RESIDU = 0.288355E-01   EPSLON = 0.100000E-05
     PSIMIN =  -1.661582459457E-02   RMAG =     1.030745990227E+00   ZMAG =    -4.385874559009E-02
                                                RESIDU = 0.794323E-03   EPSLON = 0.100000E-05
     PSIMIN =  -1.596711173598E-02   RMAG =     1.035426319390E+00   ZMAG =    -4.640274059845E-02
                                                RESIDU = 0.286070E-03   EPSLON = 0.100000E-05
     PSIMIN =  -1.571235962674E-02   RMAG =     1.037772140897E+00   ZMAG =    -4.766188361930E-02
                                                RESIDU = 0.112875E-03   EPSLON = 0.100000E-05
     PSIMIN =  -1.560810137573E-02   RMAG =     1.038919956968E+00   ZMAG =    -4.835864925118E-02
                                                RESIDU = 0.469769E-04   EPSLON = 0.100000E-05
     PSIMIN =  -1.556458112799E-02   RMAG =     1.039469569079E+00   ZMAG =    -4.876666308186E-02
                                                RESIDU = 0.201878E-04   EPSLON = 0.100000E-05
     PSIMIN =  -1.554620520859E-02   RMAG =     1.039728588415E+00   ZMAG =    -4.901273135310E-02
                                                RESIDU = 0.892763E-05   EPSLON = 0.100000E-05
     PSIMIN =  -1.553838828270E-02   RMAG =     1.039849217392E+00   ZMAG =    -4.916349192749E-02
                                                RESIDU = 0.408882E-05   EPSLON = 0.100000E-05
     PSIMIN =  -1.553504877484E-02   RMAG =     1.039904859678E+00   ZMAG =    -4.925667265705E-02
                                                RESIDU = 0.196375E-05   EPSLON = 0.100000E-05
     PSIMIN =  -1.553362089938E-02   RMAG =     1.039930306134E+00   ZMAG =    -4.931455687664E-02
                                                RESIDU = 0.100159E-05   EPSLON = 0.100000E-05
     PSIMIN =  -1.553301251006E-02   RMAG =     1.039941843507E+00   ZMAG =    -4.935062320680E-02
                                                RESIDU = 0.544595E-06   EPSLON = 0.100000E-05
  RESIDU OF ITERATION OVER MAPPING =      0.9497831927E-02

     PSIMIN =  -1.552904151419E-02   RMAG =     1.039991293601E+00   ZMAG =    -4.937510237078E-02
                                                RESIDU = 0.971877E-05   EPSLON = 0.100000E-05
     PSIMIN =  -1.550072540141E-02   RMAG =     1.040050867132E+00   ZMAG =    -4.939091390732E-02
                                                RESIDU = 0.103072E-04   EPSLON = 0.100000E-05
     PSIMIN =  -1.549272513547E-02   RMAG =     1.040088727257E+00   ZMAG =    -4.940077705803E-02
                                                RESIDU = 0.307884E-05   EPSLON = 0.100000E-05
     PSIMIN =  -1.549009240143E-02   RMAG =     1.040109035493E+00   ZMAG =    -4.940672639684E-02
```

```
                                                    RESIDU = 0.107164E-05   EPSLON = 0.100000E-05
     PSIMIN =  -1.548912222623E-02   RMAG =     1.040119185832E+00   ZMAG =    -4.941027420930E-02
                                                    RESIDU = 0.414671E-06   EPSLON = 0.100000E-05
  RESIDU OF ITERATION OVER MAPPING =     0.6986186106E-04

 NS =   30   NT =   30   NISO =  100   NPSI =   15   NCHI =  100   NINSCA =   50   NINMAP =   50

SIGMA - MESH
   0.0000E+00   3.3333E-02   6.6667E-02   1.0000E-01   1.3333E-01   1.6667E-01   2.0000E-01   2.3333E-01
   2.6667E-01   3.0000E-01   3.3333E-01   3.6667E-01   4.0000E-01   4.3333E-01   4.6667E-01   5.0000E-01
   5.3333E-01   5.6667E-01   6.0000E-01   6.3333E-01   6.6667E-01   7.0000E-01   7.3333E-01   7.6667E-01
   8.0000E-01   8.3333E-01   8.6667E-01   9.0000E-01   9.3333E-01   9.6667E-01   1.0000E+00
THETA - MESH
  -2.8214E-01   2.9600E-01   8.1069E-01   1.1855E+00   1.4673E+00   1.7079E+00   1.9398E+00   2.1851E+00
   2.4605E+00   2.7732E+00   3.1118E+00   3.4443E+00   3.7358E+00   3.9691E+00   4.1464E+00   4.2793E+00
   4.3798E+00   4.4579E+00   4.5209E+00   4.5738E+00   4.6204E+00   4.6644E+00   4.7091E+00   4.7571E+00
   4.8110E+00   4.8750E+00   4.9566E+00   5.0701E+00   5.2447E+00   5.5355E+00   6.0010E+00
RHOS - MESH
   2.3756E-01   2.4146E-01   2.8581E-01   3.4138E-01   3.8200E-01   3.9946E-01   3.9542E-01   3.7593E-01
   3.4951E-01   3.2602E-01   3.1424E-01   3.1802E-01   3.3459E-01   3.5748E-01   3.8116E-01   4.0297E-01
   4.2227E-01   4.3937E-01   4.5465E-01   4.6780E-01   4.7517E-01   4.6821E-01   4.5272E-01   4.3546E-01
   4.1730E-01   3.9754E-01   3.7487E-01   3.4723E-01   3.1214E-01   2.7042E-01   2.3756E-01

POSITION OF THE CALCULATION MESH CENTER : RO =   1.04011919E+00   ZO =  -4.94102742E-02

     PSIMIN =  -1.548855605378E-02   RMAG =     1.040089928184E+00   ZMAG =    -4.946354814463E-02
                                                    RESIDU = 0.919745E-05   EPSLON = 0.100000E-07
     PSIMIN =  -1.548806515524E-02   RMAG =     1.040091548477E+00   ZMAG =    -4.946269950300E-02
                                                    RESIDU = 0.188023E-06   EPSLON = 0.100000E-07
     PSIMIN =  -1.548793646731E-02   RMAG =     1.040092611634E+00   ZMAG =    -4.946233782789E-02
                                                    RESIDU = 0.532150E-07   EPSLON = 0.100000E-07
     PSIMIN =  -1.548789390676E-02   RMAG =     1.040093210281E+00   ZMAG =    -4.946213409907E-02
                                                    RESIDU = 0.194271E-07   EPSLON = 0.100000E-07
     PSIMIN =  -1.548787702450E-02   RMAG =     1.040093508471E+00   ZMAG =    -4.946203231838E-02
                                                    RESIDU = 0.817001E-08   EPSLON = 0.100000E-07
  RESIDU OF ITERATION OVER MAPPING =     0.4915395039E-05

     PSIMIN =  -1.548783841872E-02   RMAG =     1.040093670249E+00   ZMAG =    -4.946197655891E-02
                                                    RESIDU = 0.116762E-07   EPSLON = 0.100000E-07
     PSIMIN =  -1.548783650404E-02   RMAG =     1.040093746675E+00   ZMAG =    -4.946194414907E-02
                                                    RESIDU = 0.156279E-08   EPSLON = 0.100000E-07
  RESIDU OF ITERATION OVER MAPPING =     0.5127213531E-07

VOLUME AVERAGED QUANTITIES
**************************

AVERAGED PRESSURE        =   6.77931829E-03
TOTAL CURRENT            =   2.19518061E-01
NORMALIZED CURRENT       =   8.06185293E-01
IN     (MA,T,M)          =   6.41867272E-01
PRESSURE PEAKING FACTOR  =   3.21108781E+00
POLOIDAL BETA            =   1.23974779E+00
POLOIDAL BETA (GA)       =   1.37449768E+00
LI                       =   1.21381786E+00
LI (GA)                  =   1.34574939E+00
BETA  [%]                =   1.33495655E+00
BETA* [%]                =   1.85708258E+00
BETAX [%]                =   1.37292064E+00
G      (MA,T,M)          =   2.07980155E+00
G*     (MA,T,M)          =   2.89325015E+00
GEXP   (MA,T,M)          =   2.13894788E+00
FO=IB.S./ITOT (NUE*=0)   =   4.25089229E-01

QUANTITIES EXTRAPOLATED ON MAGNETIC AXIS
****************************************

T(PSI)ON AXIS            =   1.00000002E+00
TT-PRIME ON AXIS         =  -1.24804921E+00
Q(PSI) ON AXIS           =   9.32510105E-01
DQ/DPSI ON AXIS          =   1.57453021E+01
PRESSURE ON AXIS         =   2.17689863E-02
```

```
DP/DPSI ON AXIS        =  -8.21931046E-01
I - PRIME ON AXIS      =   2.05482761E+00
<J.B>/<T/R**2> ON AXIS =   2.13721714E+00

FUNCTIONS OF S = SQRT(PSI/PSIMIN)
*********************************

CS  - MESH
   0.0000E+00   6.6667E-02   1.3333E-01   2.0000E-01   2.6667E-01   3.3333E-01   4.0000E-01   4.6667E-01
   5.3333E-01   6.0000E-01   6.6667E-01   7.3333E-01   8.0000E-01   8.6667E-01   9.3333E-01   1.0000E+00
BETA-POLOIDAL(CS)
   4.1651E-01   4.1829E-01   4.2234E-01   4.2905E-01   4.3880E-01   4.5205E-01   4.6944E-01   4.9214E-01
   5.2291E-01   5.6544E-01   6.2549E-01   7.1049E-01   8.2530E-01   9.6821E-01   1.1232E+00   1.2117E+00
CSM - MESH
   3.3333E-02   1.0000E-01   1.6667E-01   2.3333E-01   3.0000E-01   3.6667E-01   4.3333E-01   5.0000E-01
   5.6667E-01   6.3333E-01   7.0000E-01   7.6667E-01   8.3333E-01   9.0000E-01   9.6667E-01   1.0000E+00
PSI(CSM)
   3.5361E-05   3.1825E-04   8.8402E-04   1.7327E-03   2.8642E-03   4.2787E-03   5.9760E-03   7.9562E-03
   1.0219E-02   1.2765E-02   1.5594E-02   1.8706E-02   2.2101E-02   2.5778E-02   2.9739E-02   3.1825E-02
T (CSM)
   9.9996E-01   9.9961E-01   9.9893E-01   9.9795E-01   9.9673E-01   9.9533E-01   9.9386E-01   9.9241E-01
   9.9116E-01   9.9026E-01   9.8992E-01   9.9027E-01   9.9125E-01   9.9254E-01   9.9359E-01   9.9377E-01
T*DT/DPSI(CSM)
  -1.2454E+00  -1.2242E+00  -1.1818E+00  -1.1183E+00  -1.0340E+00  -9.2895E-01  -8.0218E-01  -6.4504E-01
  -4.5624E-01  -2.3921E-01  -2.3169E-03   2.1410E-01   3.4178E-01   3.3937E-01   1.6496E-01   2.4802E-15
P (CSM)
   2.1740E-02   2.1507E-02   2.1042E-02   2.0345E-02   1.9415E-02   1.8252E-02   1.6857E-02   1.5230E-02
   1.3369E-02   1.1277E-02   8.9517E-03   6.4215E-03   3.8694E-03   1.6361E-03   2.1009E-04   0.0000E+00
DP/DPSI(CSM)
  -8.2193E-01  -8.2193E-01  -8.2193E-01  -8.2193E-01  -8.2193E-01  -8.2193E-01  -8.2193E-01  -8.2193E-01
  -8.2193E-01  -8.2193E-01  -8.2193E-01  -7.9660E-01  -6.9763E-01  -5.0598E-01  -2.0140E-01  -2.5551E-14
Q (CSM)
   9.3302E-01   9.3807E-01   9.4848E-01   9.6466E-01   9.8736E-01   1.0177E+00   1.0575E+00   1.1102E+00
   1.1817E+00   1.2810E+00   1.4225E+00   1.6317E+00   1.9580E+00   2.5093E+00   3.5935E+00   5.1073E+00
DQ/DPSI(CSM)
   1.8644E+01   1.7899E+01   1.8721E+01   1.9458E+01   2.0682E+01   2.2304E+01   2.4702E+01   2.8718E+01
   3.4763E+01   4.3642E+01   5.7136E+01   7.8843E+01   1.1666E+02   1.9174E+02   3.9925E+02   2.6471E+03
SHEAR(CS)
   0.0000E+00   5.3931E-03   2.1938E-02   5.0127E-02   9.1045E-02   1.4641E-01   2.1864E-01   3.1693E-01
   4.5284E-01   6.3493E-01   8.7390E-01   1.1832E+00   1.5767E+00   2.0705E+00   2.7100E+00   4.6056E+00
I-STAR(CSM)
   2.0523E+00   2.0322E+00   1.9920E+00   1.9318E+00   1.8514E+00   1.7509E+00   1.6290E+00   1.4769E+00
   1.2923E+00   1.0777E+00   8.3959E-01   5.9022E-01   3.4928E-01   1.4552E-01   1.8666E-02   2.1901E-14
I//=<J . B> / <T / R**2>
   2.1346E+00   2.1135E+00   2.0712E+00   2.0076E+00   1.9226E+00   1.8158E+00   1.6857E+00   1.5231E+00
   1.3259E+00   1.0971E+00   8.4461E-01   5.8350E-01   3.3693E-01   1.3531E-01   1.5902E-02   2.0446E-14
RHO(CS)
   0.0000E+00   5.0579E-02   1.0135E-01   1.5252E-01   2.0429E-01   2.5691E-01   3.1065E-01   3.6582E-01
   4.2286E-01   4.8241E-01   5.4537E-01   6.1303E-01   6.8735E-01   7.7152E-01   8.7129E-01   1.0000E+00
MERCIER
  -2.2177E+01  -2.5317E+00  -6.3785E-01  -1.6150E-01   4.1237E-02   1.4016E-01   1.9731E-01   2.3202E-01
   2.5258E-01   2.6441E-01   2.7159E-01   2.7592E-01   2.7716E-01   2.7442E-01   2.6241E-01   2.5000E-01
RESISTIVE INTERCHANGE
  -2.2401E+01  -2.7549E+00  -8.6119E-01  -3.8431E-01  -1.8109E-01  -8.1420E-02  -2.3514E-02   1.1535E-02
   3.2540E-02   4.5438E-02   5.4674E-02   6.1382E-02   6.2537E-02   5.3202E-02   1.9817E-02  -1.7764E-15
H OF GLASSER, GREENE & JOHNSON
   2.5992E-02   2.7570E-02   2.7411E-02   2.7965E-02   2.8482E-02   2.9278E-02   3.0078E-02   3.0437E-02
   3.0921E-02   3.2057E-02   3.4257E-02   3.6815E-02   3.6726E-02   2.9656E-02   7.4643E-03  -3.2953E-15
NTURN =          10
NCBAL
           0            0            0            0            0            0            0            0
           0            0            0            0            0            0            0            0
CHIO VALUES
   0.0000E+00
```

Interpretation of the run output

The test run output shows the result for an ITER like equilibrium asymmetric about the $Z = 0$ plane (see Fig. 1). The current density j_ϕ is specified with Eq. (10), and both p and I^* are parabolic functions of $s = 1 - \Psi/\Psi_{\min}$ up to a certain radius, and vanish smoothly at the plasma surface. The run output first gives

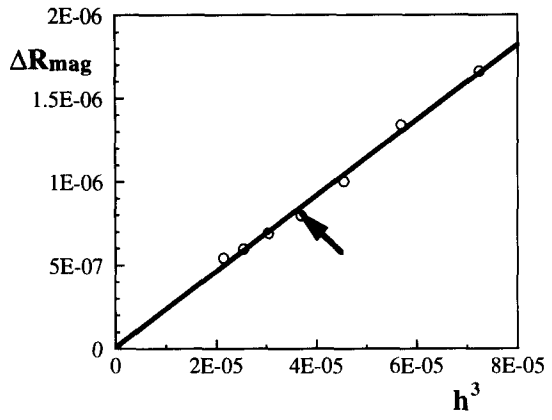

Fig. 5. Convergence of the major radius of the magnetic axis. The regression curve is $|R_{mag}/R_0 - 1.040093| = 2.25.10^{-2}h^3$. The arrow indicates the error for $N_\sigma = N_\theta = 30$.

the equilibrium Namelist (EQDATA) variables used for the computation of that equilibrium. The equilibrium (σ, θ) discretization grid is printed next together with the RHOS = $\rho_s(\theta)$ plasma radii and the position of the centre of the equilibrium discretization mesh (see Eq. (5)). For every Picard iteration in Eq. (27), the output shows the value of Ψ_{min}, the position (R_{mag}, Z_{mag}) of the magnetic axis and the residue (28). Moreover, the residue given in Section 5.2.3 is printed at every iteration over the integrals (9) required for the calculation of j_ϕ. As mentioned earlier, the equilibrium is first computed with a coarse discretization grid (here, NS = $N_\sigma = 24$ and NT = $N_\theta = 24$) centred at (R_c, Z_c) = (R0, RZ0) = (1,0) and second with a refined grid (30×30 in this example) centred at the position of the magnetic axis obtained previously.

This equilibrium is scaled so that $T_0 = 1$ (NTMF0 = 1) and $q(s = 0.33) = 1$. (NCSCAL = 1, QSPEC = 1. and CSSPEC = 0.33) by the equilibrium transformation rules in Section 5.3. After the scaling, results are given for global equilibrium quantities (see Table 1) and poloidal flux surface quantities (certain quantities are given at the integer s_l = CS stability mesh, and others at the half-integer $s_{l+1/2}$ = CSM mesh). The arrays MERCIER and RESISTIVE INTERCHANGE show the values of the $-D_I$ and $-D_R$ parameters in Eqs. (19) and (20). A negative value in these arrays means that the corresponding poloidal flux surface is unstable with respect to ideal or resistive interchange modes. This equilibrium is stable with respect to $n \to \infty$ ballooning modes because all the values in the array NCBAL are 0. A nonzero value implies that the corresponding flux surface is ballooning unstable. NTURN is the number of 2π turns in χ used to the left and to the right of the ballooning angle χ_0 = CHI0 (see Eqs. (17) and (18)) for the integration of the ballooning integral (16). For this equilibrium calculation, NBLC0 = 1. Therefore, the ballooning stability criterion is checked only for $\chi_0 = 0$.

In Ref. [1], convergence tests are presented for a Solovev and a JET-shaped equilibrium with prescribed p' and TT'. Fig. 5 shows the convergence of the magnetic axis for the asymmetric equilibrium with prescribed p' and I^* used here as running test. Despite the additional loop over the intergrals (9), the $\mathcal{O}(h^3)$ convergence rate of the magnetic axis predicted by theory and observed for the test equilibria in Ref. [1] is preserved, and good accuracy is already obtained with a NS = NT = 30 equilibrium discretization mesh (arrow in Fig. 5). The run test equilibrium was executed with different computers and Table 17 shows the cpu time consumption. The scalar/vector ratio on Cray C-90 is of about 8.7 (scalar run with compiler option -Wf"-o novector").

Acknowledgements

The work on CHEASE was made over several years. The coding and most of the testing of CHEASE took place while the authors were with the Centre de Recherches en Physique des Plasmas, EPF Lausanne. This

work was supported in part by the Swiss National Science Foundation. The present paper was written mainly while H. Lütjens was at Ecole Polytechnique, Palaiseau, supported by the EURATOM fellowship no. ERB5000 CT 93-5004.

Several persons have provided invaluable help in preparing the output for various codes and reporting problems with CHEASE. We would like to thank, in chronological order, R. Gruber, F. Troyon, A. Roy, G. Vlad, L. Villard, S. Bernel, D. J. Ward, A. Jaun, J. F. Luciani, A. Pletzer and M. S. Chu.

Appendix A. Derivation of expressions involved in the flux coordinate transformation

The purpose of this appendix is to document the relations between the equilibrium coordinates and the generalized poloidal angle χ. These relations are used in CHEASE for the integration of χ and the nonorthogonality $\beta_{\Psi\chi}$.

A.1. Expression of χ in terms of θ on a constant poloidal flux surface

The line element dl along a constant poloidal flux surface is related to the variation $d\chi$ of χ by

$$d\chi = \nabla\chi \cdot \boldsymbol{dl} = \frac{R}{J|\nabla\Psi|}\, dl\,, \tag{A.1}$$

where $J = [(\nabla\Psi \times \nabla\chi).\nabla\phi]^{-1}$. Furthermore, using the definition of dl and Eqs. (1), (2)

$$\begin{aligned} dl^2 &\equiv g_{\sigma\sigma} d\sigma^2 + g_{\theta\theta}\, d\theta^2 + 2g_{\sigma\theta}\, d\sigma d\theta \\ &= \rho_s^2(\theta)\, d\sigma^2 + \sigma^2 \left[\rho_s^2(\theta) + \left(\frac{d\rho_s}{d\theta}\right)^2\right] d\theta^2 + 2\sigma\rho_s(\theta)\frac{d\rho_s}{d\theta}\, d\sigma d\theta\,. \end{aligned} \tag{A.2}$$

Along a constant poloidal flux surface, $d\sigma = -(\partial\Psi/\partial\theta)/(\partial\Psi/\partial\sigma)d\theta$. Therefore, Eq. (A.1) becomes

$$d\chi = \frac{R\sigma\rho_s^2(\theta)}{J\frac{\partial\Psi}{\partial\sigma}}\, d\theta\,, \tag{A.3}$$

which leads to Eq. (15).

A.2. Relation between the nonorthogonality and the current density

With Eqs. (1), (2), the toroidal current density can be expressed as

$$j_\phi = \boldsymbol{e}_\phi.\nabla \times (\nabla\phi \times \nabla\Psi)$$

Table 17
Cpu time required on different computers

Computer	Processors	Cpu time (seconds)
Cray C-90	Cray 250 MHz	11.8
Sun Sparcstation 10	Supersparc 50 MHz	504.9
Hewlett Packard K200	PA7200 100 MHz	165.6
IBM-RS6000 Model 390	POWER2 71,5 MHz	159.0
Silicon Graphics Indigo-2	R4400 200 MHz	183.9

$$= \frac{R}{J_\perp}\frac{\partial}{\partial\Psi}\left(\frac{|\nabla\Psi|^2 J_\perp}{R^2}\right)_n , \tag{A.4}$$

where the subscript n denotes the normal derivative, $(\Psi, \chi_\perp, \phi)$ is an orthogonal flux coordinate system, and $J_\perp = [(\nabla\Psi \times \nabla\chi_\perp).\nabla\phi]^{-1}$. Therefore,

$$\frac{R^2}{J_\perp}\left(\frac{\partial}{\partial\Psi}\frac{J_\perp}{R^2}\right)_n = \frac{Rj_\phi}{|\nabla\Psi|^2} - 2\left(\frac{\partial \ln|\nabla\Psi|}{\partial\Psi}\right)_n . \tag{A.5}$$

Moreover,

$$\beta_{\Psi\chi} = \frac{\nabla\Psi\cdot\nabla\chi}{|\nabla\Psi|^2} = \left(\frac{\partial\chi}{\partial\Psi}\right)_n \quad \text{and} \quad \left(\frac{\partial\chi}{\partial\chi_\perp}\right)_\Psi = \frac{J_\perp}{J} . \tag{A.6}$$

Eventually, using Eq. (A.5),

$$\begin{aligned}\left(\frac{\partial\beta_{\Psi\chi}}{\partial\chi}\right)_\Psi &= \frac{J}{J_\perp}\left(\frac{\partial}{\partial\Psi}\frac{J_\perp}{J}\right)_n \\ &= -\frac{1}{J}\left(\frac{\partial J}{\partial\Psi}\right)_n + \frac{Rj_\phi}{|\nabla\Psi|^2} + 2\left(\frac{\partial \ln R}{\partial\Psi}\right)_n - 2\left(\frac{\partial \ln|\nabla\Psi|}{\partial\Psi}\right)_n .\end{aligned} \tag{A.7}$$

Substituting J by Eq. (14) and integrating with respect to θ using Eq. (A.3) leads to the expression of $\beta_{\Psi\chi}$ in Eq. (15).

Appendix B. Evaluation of the Mercier criterion in CHEASE

In this appendix, first informations about the formulation of the Mercier criterion (19) are quoted, and second the numerical method used for the evaluation of $-D_I$ is described.

B.1. Relation between different formulations of the Mercier criterion

According to Freidberg [23], Chapter 10.5.3, Eq. (10.160), the Mercier criterion for local interchange stability reads $D_m < 1/4$, where

$$D_m = \frac{p'(T^2J_1 + J_4)}{2\pi(q')^2}\oint\left[\frac{2\kappa_n}{|\nabla\Psi|} + \frac{T}{B^2}\left(\frac{Tp'}{|\nabla\Psi|^2} - \frac{1}{J}\frac{\partial g}{\partial\chi}\right)\right] J\,d\chi + \frac{Tp'J_2}{(q')^2}\left(q' - Tp'J_2\right) \tag{B.1}$$

and the J_i's are given by Eq. (22). However, certain terms cancel in this form of D_m. The magnetic field line curvature,

$$\boldsymbol{\kappa} = \frac{\boldsymbol{B}\cdot\nabla\boldsymbol{B}}{B^2} = \frac{1}{B^2}\left(\nabla p + \tfrac{1}{2}\nabla_\perp B^2\right) , \tag{B.2}$$

is perpendicular to $\boldsymbol{B}$. Thus, $\boldsymbol{\kappa} = (0, \kappa_n, \kappa_t)$ in the orthonormal magnetic coordinate system $(\boldsymbol{B}/B; \boldsymbol{n} = \nabla\Psi|\nabla\Psi|; \boldsymbol{t})$ and with Eq. (2),

$$\kappa_n = \frac{\nabla\Psi}{|\nabla\Psi|}\cdot\frac{\partial}{\partial\Psi}\left(p + \tfrac{1}{2}B^2\right)\frac{\nabla\Psi}{B^2} \equiv \left[\frac{\partial\tilde{P}}{\partial\Psi}\right]_n \frac{|\nabla\Psi|}{B^2} , \tag{B.3}$$

where $\tilde{P}$ is defined in Eq. (17). Using Eq. (B.3) for κ_n, and Eq. (17) for g and $\tilde{P}$, the integrand in Eq. (B.1) is transformed as follows:

$$\frac{2\kappa_n}{|\nabla\Psi|}+\frac{T}{B^2}\left(\frac{Tp'}{|\nabla\Psi|^2}-\frac{1}{J}\frac{\partial g}{\partial\chi}\right)=\frac{R^2p'}{|\nabla\Psi|^2}+\frac{1}{B^2}\left(p'+\frac{TT'}{R^2}+\frac{\partial}{\partial\Psi}\left[\frac{|\nabla\Psi|^2}{R^2}\right]_n-\frac{T^2}{JR^2}\left[\frac{\partial J}{\partial\Psi}\right]_n\right.$$
$$\left.-\frac{T^2}{R^2}\left[\frac{\partial\beta_{\Psi\chi}}{\partial\chi}\right]_\Psi\right). \tag{B.4}$$

The right-hand side of the Grad–Shafranov equation (3) can be recognized in Eq. (B.4). Using the relation between j_ϕ and $\beta_{\Psi\chi}$ in Appendix A, Eq. (A.7), it reads

$$p'+\frac{TT'}{R^2}=-\frac{j_\phi}{R}=-\frac{\partial}{\partial\Psi}\left[\frac{|\nabla\Psi|^2}{R^2}\right]_n-\frac{|\nabla\Psi|^2}{R^2}\left(\left[\frac{\partial\beta_{\Psi\chi}}{\partial\chi}\right]_\Psi+\frac{1}{J}\left[\frac{\partial J}{\partial\Psi}\right]_n\right). \tag{B.5}$$

Substituting that into Eq. (B.1) leads to

$$\frac{2\kappa_n}{\nabla\Psi}+\frac{T}{B^2}\left(\frac{Tp'}{|\nabla\Psi|^2}-\frac{1}{J}\frac{\partial g}{\partial\chi}\right)=\frac{R^2p'}{|\nabla\Psi|^2}-\left[\frac{\partial\beta_{\Psi\chi}}{\partial\chi}\right]_\Psi-\frac{1}{J}\left[\frac{\partial J}{\partial\Psi}\right]_n, \tag{B.6}$$

and therefore,

$$D_m=\frac{p'}{(q')^2}\left(T^2J_1+J_4\right)\left(p'J_3-J_5'\right)+\frac{Tp'J_2}{(q')^2}\left(q'-Tp'J_2\right), \tag{B.7}$$

where the J_i's are defined in Eq. (22). Therefore, the Mercier criterion for interchange stability is equivalent to Eq. (19).

B.2. Numerical evaluation of $-D_I$

An inspection of the integrals (22) involved in the Mercier criterion (19) shows that J_1, J_2 and J_3 are quantities of the same order, as well as q' and J_5'. For certain tokamak equilibria, $J_1\sim J_2\sim J_3\sim 10^5$, $J_5'\sim q'\sim 10^2$ and therefore $-D_I$, which results from a substraction of terms of order J_2^2 and which can be of order 1 or less, is very sensitive to numerical cancellation errors. $-D_I$ is most accurate if terms of the same order are calculated by the same numerical method. In (σ,θ) coordinates,

$$q'(\Psi)=T'J_4+\frac{T}{2\pi}\oint_{\Psi=\text{const.}}\left(1-\frac{\sigma}{R}\frac{\partial R}{\partial\sigma}-\sigma\frac{\partial^2\Psi/\partial\sigma^2}{\partial\Psi/\partial\sigma}\right)\frac{\rho_s^2(\theta)}{R(\partial\Psi/\partial\sigma)^2}d\theta,$$
$$J_5'(\Psi)=\oint_{\Psi=\text{const.}}\left(2-\frac{R_0}{R}-\sigma\frac{\partial^2\Psi/\partial\sigma^2}{\partial\Psi/\partial\sigma}\right)\frac{R\rho_s^2(\theta)}{(\partial\Psi/\partial\sigma)^2}d\theta, \tag{B.8}$$

where all derivatives against σ are evaluated at constant θ. In CHEASE, $J_1,\ldots,J_5$, J_5' and q' are computed in subroutine SURFACE with the integration method described in [1], Section 4.1, and the Mercier and resistive interchange criteria are evaluated in subroutine GLOQUA. Acceptable results are also obtained if both q' and J_5' are differentiated numerically.

Appendix C. Bicubic spline interpolation of the cubic Hermite equilibrium solution

The purpose of this appendix is to show how the solution for Ψ given in Hermite bicubic basis functions can be smoothed by interpolation using bicubic splines. The smoothing algorithm applied in CHEASE uses a bicubic spline interpolation of Ψ from the values at the (σ,θ) nodes, ignoring the derivatives Ψ_σ, Ψ_θ and $\Psi_{\sigma\theta}$ of the bicubic Hermite solution. The new values of Ψ_σ, Ψ_θ and $\Psi_{\sigma\theta}$ are computed at the nodes such that the

Hermite bicubics have continuous second derivatives. Here we discuss how Ψ_σ, Ψ_θ and $\Psi_{\sigma\theta}$ are computed for the smoothed solution.

The cubic spline interpolation for a tabulated function $f_i = f(x_i), i = 1,\ldots,N$ in the interval $[x_j, x_{j+1}]$ is given by [33,34]

$$f(x) = A_1 f_j + A_2 f_{j+1} + A_3 f_j'' + A_4 f_{j+1}'', \tag{C.1}$$

where

$$\begin{aligned} A_1 &= \frac{x_{j+1} - x}{x_{j+1} - x_j}, \\ A_2 &= \frac{x - x_j}{x_{j+1} - x_j}, \\ A_3 &= \tfrac{1}{6} A_1 (A_1^2 - 1)(x_{j+1} - x_j)^2, \\ A_4 &= \tfrac{1}{6} A_2 (A_2^2 - 1)(x_{j+1} - x_j)^2. \end{aligned} \tag{C.2}$$

The second derivatives $f_i'' = f''(x_i), i = 1,\ldots,N$ required for the evaluation of (C.1) are computed by imposing that the first derivative,

$$\frac{df}{dx} = \frac{f_{j+1} - f_j}{x_{j+1} - x_j} - \frac{3A_1^2 - 1}{6} f_j'' + \frac{3A_2^2 - 1}{6} f_{j+1}'', \tag{C.3}$$

is continuous at $x = x_i, i = 1,\ldots,N$. This condition is satisfied for $i = 2,\ldots,N-1$ if

$$\frac{x_i - x_{i-1}}{6} f_{i-1}'' + \frac{x_{i+1} - x_{i-1}}{3} f_i'' + \frac{x_{i+1} - x_i}{6} f_{i+1}'' = \frac{f_{i+1} - f_i}{x_{i+1} - x_i} - \frac{f_i - f_{i-1}}{x_i - x_{i-1}}, \tag{C.4}$$

and the values for f_1' and f_N' are given as boundary conditions.

For a tabulated function defined on a rectangular grid (x, y), the bicubic spline interpolation is a product of one-dimensional splines taken in the x and in the y directions. Therefore, the bicubic Hermite finite element solution Ψ on a rectangular (σ, θ) mesh, with periodic boundary conditions in θ (i.e. $\Psi(\sigma, \theta+2\pi) = \Psi(\sigma, \theta)$) will have smooth first and second derivatives if

(i) Eq. (C.4) is solved for all $\theta_k, k = 1,\ldots,N_\theta$, with
- $x_i = \sigma_i, i = 1,\ldots,N_\sigma$;
- $f_i = \Psi(\sigma_i, \theta_k), i = 1,\ldots,N_\sigma$;
- Boundary conditions: $f_1' = (\partial\Psi/\partial\sigma)(\sigma_1, \theta_k)$ and $f_{N_\sigma}' = (\partial\Psi/\partial\sigma)(\sigma_{N_\sigma}, \theta_k)$, given by the bicubic Hermite solution.

(ii) $\partial\Psi/\partial\sigma$ is reevaluated at every (σ, θ) node with Eq. (C.3).

(iii) Eq. (C.4) is solved for all $\sigma_k, k = 1,\ldots,N_\sigma$, with
- $x_i = \theta_i, i = 1,\ldots,N_\theta$;
- $f_i = \Psi(\sigma_k, \theta_i), i = 1,\ldots,N_\theta$;
- Periodic boundary conditions, $f_1' = (\partial\Psi/\partial\theta)(\sigma_k, \theta_1) = (\partial\Psi/\partial\theta)(\sigma_k, \theta_{N_\theta+1}) = f_{N_\theta+1}'$;

and with
- $x_i = \theta_i, i = 1,\ldots,N_\theta$;
- $f_i = (\partial\Psi/\partial\sigma)(\sigma_k, \theta_i), i = 1,\ldots,N_\theta$;
- Periodic boundary conditions, $f_1' = (\partial^2\Psi/\partial\sigma\partial\theta)(\sigma_k, \theta_1) = (\partial^2\Psi/\partial\sigma\partial\theta)(\sigma_k, \theta_{N_\theta+1}) = f_{N_\theta+1}'$.

For periodic boundary conditions, the system (C.4) becomes cyclic.

(iv) $(\partial\Psi/\partial\theta)$ and $(\partial^2\Psi/\partial\sigma\partial\theta)$ are recomputed at every (σ, θ) node with Eq. (C.3).

Appendix D. Symbolic names of some important variables in CHEASE

Symbol	Definition	Symbolic name	Subroutine
a/R_0	(47,49)	ASPCT	Namelist
a_1	(32)	SCALE	NOREPT
a_2	(33)	ZCSHFT	NOREPT
$A(\Psi)$		ARATIO	SURFACE
c_1, c_2	(17)	ZF, ZG	BALOON
$C(\Psi)$	(14)	CP	SURFACE
$C'(\Psi)$	(15)	CPDP	SURFACE
$C_0, ..., C_3$	(9)		CINT
$-D_I(\Psi)$	(19)	SMERCI	GLOQUA
$-D_R(\Psi)$	(20)	SMERCR	GLOQUA
f_c	(25)	RFCIRC	SURFACE
g	(17)	ZGBAR	BALOON
H	(21)	HMERCR	GLOQUA
I	(4),Table 1	RITOT	GLOQUA
$I^*(\Psi)$	(7)	CIPR	ISOFUN, PRFUNC
I_N	Table 1	RINOR	GLOQUA
$I_\phi(\Psi)$	Table 1	ZJPSI	GLOQUA
$I_{\|\|}(\Psi)$	(8)	CIPR	ISOFUN, PRFUNC
j_ϕ	(3,10,11)	PJIPHI	CURENT
J	(14)	ZJAC	SURFACE, ERDATA, JNOVAW, BALOON, VACUUM
J_v	(36)	ZJAC1	VACUUM
$\langle \boldsymbol{J} \cdot \boldsymbol{B} \rangle$	(44)	RJPAR	SURFACE
$\langle \boldsymbol{J} \cdot \boldsymbol{B} \rangle_{\text{bs}}$	(23)	RJBSH	SURFACE
$J_1, ..., J_6, J_5'$	(22)	RJ1,...,RJ6,RJ5P	SURFACE
ℓ_i	Table 1	RINDUC	GLOQUA
N_m		MSMAX	Namelist
N_s		NPSI	Namelist
N_θ		NT	Namelist
N_σ		NS	Namelist
N_χ		NCHI	Namelist
$p(\Psi)$	(3)	CPR	PPRIME
$p'(\Psi)$	(3)	CPPR	PPRIME
ppf.	Table 1	CPPF	GLOQUA
$q(\Psi)$	(6)	QPSI	SURFACE
$q'(\Psi)$		CDQ	SURFACE
q_0		Q0	MAPPIN
R_0		1	
R_c	(5)	R0	Namelist
R_{mag}		RMAG	MAGAXE
R_{vc}	(35)	R0W	Namelist
s_k	(12)	CS	MESH
$s_{k+1/2}$	(12)	CSM	MESH
$\hat{s}(\Psi)$	Table 1	SHEAR	GLOQUA
$T(\Psi)$	(3)	TMF	ISOFUN, PRFUNC
$TT'(\Psi)$	(3)	TTP	ISOFUN, PRFUNC
$V(\Psi)$	Table 1	VSURF	GLOQUA
$2\pi V_{\text{tot}}$	Table 1	VOLUME	GLOQUA

Symbol	Definition	Symbolic name	Subroutine
Z_c	(5)	RZ0	Namelist
Z_{mag}		RZMAG	MAGAXE
Z_{vc}	(35)	RZ0W	Namelist
α	(14)	NER	Namelist
β	Table 1	BETA	GLOQUA
β^*	Table 1	BETAS	GLOQUA
$\beta_p(\Psi)$	Table 1	BETAB	GLOQUA
$\beta_{p,tot}$	Table 1	BETAP	GLOQUA
β_x	Table 1	BETAX	GLOQUA
$\beta_{\Psi,\chi}$	(15)	BCHIN, BCHIO	SURFACE
Γ		GAMMA	Namelist
δ	(49)	TRIANG	Namelist
ϵ	(28)	EPSLON	Namelist
ζ	(49)	CETA	Namelist
θ	(5)	CT	MESH
θ_0	(50)	THETA0	Namelist
κ	(49)	ELONG	Namelist
λ_1	(42)	CFBAL	Namelist
λ_2		CPRESS	Namelist
μ	(14)	NEGP	Namelist
ν	(50)	RNU	Namelist
$\tilde{\xi}, \xi$	(49,50)	XI	Namelist
ξ_r	(16)		BALOON
$\rho(\Psi)$	Table 1	RSURF	GLOQUA
$\rho_s(\theta)$	(5)		BOUND
σ	(5)	CSIG	MESH
$\tilde{\sigma}$	(50)	SGMA	Namelist
χ_k	(14)	CHI	MESH
$\chi_{k+1/2}$	(14)	CHIM	MESH
Ψ	(3)	CPSI	SOLVIT
$\Psi_{Solovev}$	(47)		TEST

References

[1] H. Lütjens, A. Bondeson and A. Roy, Comput. Phys. Commun. 69 (1992) 287.
[2] V.D. Shafranov, ZhETF 33 (1957) 710; Sov. Phys. JETP 8 (1958) 494.
[3] R. Lüst and A. Schlüter, Z. Naturforsch. 129 (1957) 850.
[4] H. Grad and H. Rubin, Proc. 2nd Int. Conf. on the Peaceful Uses of Atomic Energy, Vol. 31 (United Nations, Geneva, 1958) p. 190.
[5] J.W. Connor, R.J. Hastie and J.B. Taylor, Phys. Rev. Lett. 40 (1978) 396.
[6] M.N. Rosenbluth, R.D. Hazeltine and F.L. Hinton, Phys. Fluids 15 (1972) 116.
[7] R.D. Hazeltine, F.L. Hinton and M.N. Rosenbluth, Phys. Fluids 16 (1973) 1645.
[8] S.P. Hirshman, Phys. Fluids 31 (1988) 3150.
[9] R. Gruber, F. Troyon, D. Berger, L.C. Bernard, S. Rousset, R. Schreiber, W. Kerner, W. Schneider and K.V. Roberts, Comput. Phys. Commun. 21 (1981) 323.
[10] A. Bondeson, G. Vlad and H. Lütjens, Phys. Fluids B 4 (1992) 1889.
[11] R.C. Grimm, J.M. Greene and J.L. Johnson, Methods Comput. Phys. 9 (1976) 253.
[12] R.C. Grimm, R.L. Dewar and J. Manickam, J. Comput. Phys. 49 (1983) 94.
[13] D.J. Ward and S.C. Jardin, Nucl. Fusion 32 (1992) 973.
[14] K. Lerbinger and J.F. Luciani, J. Comput. Phys. 97 (1991) 444.
[15] L. Villard, K. Appert, R. Gruber and J. Vaclavik, Comput. Phys. Reports 4 (1986) 95.
[16] A. Jaun, K. Appert, H. Lütjens, S. Brunner, J. Vaclavik and L. Villard, Theory of Fusion Plasmas, Proc. Int. Workshop, Varenna, 1994 (Editrice Compositori, Bologna, 1994) p. 369.
[17] L.L. Lao et al., Nucl. Fusion 30 (1990) 1035.

[18] P.H. Rebut, JET team, Plasma Physics and Controlled Nuclear Fusion Research 1986, Proc. 11th Int. Conf., Kyoto, 1986, Vol. 1 (IAEA, Vienna, 1987) p. 31.
[19] P.H. Rebut, V. Chuyanov, M. Huguet et al., paper CN-60-E-1-I-1, Plasma Physics and Controlled Nuclear Fusion Research 1994, Proc. 15th Int. Conf., Seville, 1994 (IAEA, Vienna, 1995);
F. Perkins, P. Barabaschi, G. Bosia et al., paper CN-60-E-1-I-3, Plasma Physics and Controlled Nuclear Fusion Research 1994, Proc. 15th Int. Conf., Seville, 1994 (IAEA, Vienna, 1995).
[20] F. Troyon, R. Gruber, H. Sauremann, S. Semenzato and S. Succi, Plasma Physics and Controlled Fusion 26 (1984) 209.
[21] A.D. Turnbull, M.A. Secrétan, F. Troyon, S. Semenzato and R. Gruber, Comput. Phys. Commun. 66 (1986) 391.
[22] J. Delucia, S.C. Jardin and A.M.M. Todd, J. Comput. Phys. 37 (1980) 183.
[23] J.P. Freidberg, Ideal Magnetohydrodynamics (Plenum Press, New York, 1987).
[24] C. Mercier, Nucl. Fusion Suppl. 1 (1960) 47.
[25] W. Graeub, Die Grundlehren der mathematischen Wissenschaften, Band XCVII, Lineare Algebra (Springer, Berlin, Goettingen, Heidelberg, 1958) p. 157.
[26] A.H. Glasser, J.M. Greene and J.L. Johnson, Phys. Fluids 18 (1975) 875.
[27] S.C. Jardin, A. Bhattacharjee, A. Bondeson, M.S. Chance, S.C. Cowley, G. Eriksson, J.M. Greene, F. Hofmann, M. Hughes, R. Iacono, D.W. Ignat, C. Kessel, H. Lütjens, J. Manickam, D.A. Monticello, F.W. Perkins, M. Phillips, N. Pomphrey, J. Ramos, A.H. Reiman, P.H. Rutherford, E.J. Valeo, L. Villard, C. Wang and D.J. Ward, Proc. 14th Int. Conf. on Plasma Physics and Controlled Fusion, Vol. 2 (IAEA, Würzburg, Germany, 1992) p. 285.
[28] G. Strang and G. Fix, An Analysis of the Finite Element Method (Prentice-Hall, Englewood Cliffs, NJ, 1973).
[29] K.W. Morton, Comput. Phys. Report 6 (1987) 1.
[30] A. Bondeson and G.Y. Fu, Comput. Phys. Commun. 66 (1991) 167.
[31] L.S. Solovev, Zh. Tekh. Fiz. 53 (1967) 626; JETP 26 (1968) 400.
[32] A. Roy and F. Troyon, Theory of Fusion Plasmas, Proc. Joint Varenna-Lausanne Int. Workshop, 1988 (Editrice Compositori, Bologna, 1988) p. 175.
[33] G.E. Forsythe, M.A. Malcolm and C.B. Moler, Computer Methods for Mathematical Computations (Prentice-Hall, Englewood Cliffs, NJ, 1977) p. 70.
[34] W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Vetterling, Numerical Recipes (Cambridge Univ. Press, Cambridge, 1989) p. 86.