Equilibrium Spline Interface (ESI) for magnetic confinement codes

Leonid E. Zakharov, Stephen C. Jardin, Morrell S. Chance, Roscoe B. White
Princeton University, Princeton Plasma Physics Laboratory, Princeton NJ
Grigorii V. Pereverzev
Max-Plank Institute of Plasma Physics, Garching, Germany

Abstract

A compact and comprehensive interface between magneto-hydrodynamic (MHD) equilibrium codes and gyro-kinetic, particle orbit, MHD stability, and transport codes is presented. Its irreducible set of equilibrium data consists of four 2- or 3-D functions of coordinates and four 1-D radial profiles together with their first derivatives. The C reconstruction routines, accessible also from Fortran, allow the calculation of basis functions and their first derivatives at any position inside the plasma. After this all vector fields and geometric coefficients required for the above mentioned types of codes can be calculated using only algebraic operations with no further interpolation or differentiation.

For easy navigation the enumeration in the Table of Contents, and the "(to ToC)" right after the section names are the forward and backward hyperlinks between Table of Contents and the beginning of sections.

Contents

1 Introduction *(to ToC)*

In plasma simulations, which deal with highly anisotropic equations, the equilibrium codes play a fundamental role in supplying the information about the magnetic configuration. For more sophisticated plasma models: stability, transport, particle orbit or gyro-kinetic, this information is needed in flux coordinates related to the geometry of the magnetic field.

While the equilibrium configurations depend essentially only on two radial profiles and the shape of the plasma boundary, the request of other codes may contain numerous 2- or 3-dimensional functions representing components of the magnetic fields, current density, curvature of the field lines, or the magnitude of the magnetic field and its derivatives; all of these functions are inter-related. Because of finite accuracy of the equilibrium codes it is always a matter of preference as to what should be taken as input data and what should be calculated inside the "client" codes.

There are two extreme approaches to making such a decision, and both rely on the exchange of data files. One of them requires all functions necessary for the client code at all necessary positions to be supplied by the equilibrium codes. Working only for closely related pairs of codes and making them highly dependent, this approach requires vast storage capacities because the required data (e.g., some combinations of metric coefficients) are not very smooth. Being widespread in the plasma physics community, it is totally irrelevant to the situation when the client codes (like particle motion) need physics variables at unpredictable positions inside the plasma.

Another extreme approach uses only primitive information from the equilibrium codes, like coordinates of grid points and a few radial profiles, while all necessary functions are reconstructed internally inside the client codes. The problem here is in calculation of space derivatives, entering into many physics variables. With no universal mathematical rules of doing numerical differentiation, this approach unavoidably introduces its own inaccuracies and causes potential convergence problems in interaction with equilibrium codes.

The proposed Equilibrium Spline Interface (`ESI`) consists of data (stored in files or shared memory) and a C-source file (e.g., `escZ.c`) with C- or FORTRAN callable `ESI` initiation and reconstruction routines. This paper and other documentation on the current state of `ESI` can be found in `esiZ.c.d` file.

The interface is compact and comprehensive. `ESI` data represent a Hermit polynomial representation of only well behaved functions in terms of their values and first derivatives on a mesh. A limited set of basis functions which consists of four radial profiles and four 2-D (for axisymmetric configurations like tokamaks) or 3-D (for stellarators) functions of space coordinates.

The reconstruction routines can calculate basis functions and their first derivatives at any point inside the plasma. Then, the client code can calculate the necessary vector or scalar field variables by algebraic operations without use of higher derivatives.

Sect. 2 introduces notations and basis of `ESI` functions and their relations with the physics variables and plasma profiles. Sec. 3 describes initiation of `ESI` and a call main reconstruction routine at the use side. Sec. 4 contains basic geometrical relationships, while Sec. 5 provides the self consistent recipes for calculating parameters of magnetic configurations.

Sect. 6 outlines the use of `ESI` for tracing field lines and particle orbits. The Hamiltonian canonical coordinates are introduced for guiding center motion in 3-D nested configurations. Boozer coordinates are not canonical, as is often mistakenly believed. Sect. 7 shows sufficiency of `ESI` for MHD stability codes.

Finally, Sect. 8 and Sect. 9 describes the data storage structure, file formats and the set of `ESI` routines.

2 The basis functions of `ESI` *(to ToC)*

The curvilinear coordinates a, θ, ζ for describing magnetic configurations for plasma confinement can be represented parametrically by equations

$$r = r(a, \theta, \zeta), \quad z = z(a, \theta, \zeta), \quad \varphi = \varphi(a, \theta, \zeta), \tag{2.1}$$

where r, φ, z are laboratory cylindrical coordinates. Coordinates are assumed to be nested and their Jacobian J

$$J \equiv \frac{1}{(\nabla a \times \nabla \theta) \cdot \nabla \zeta} \tag{2.2}$$

is not vanishing.

In the ideal situation, the magnetic surfaces are also nested and it is possible to consider so-called "flux" coordinates, when

$$\mathbf{B} \cdot \nabla a = 0. \tag{2.3}$$

The vector potential of the magnetic field $\mathbf{B}$ has the following covariant representation

$$\mathbf{A} = -\bar{\Phi}' \eta \nabla a + \bar{\Phi} \nabla \theta + \bar{\Psi} \nabla \zeta. \tag{2.4}$$

In nested coordinates the dependencies on angles θ, ζ in function $\bar{\Phi}$ can be eliminated by adding ∇u to the vector potential and then by massaging the radial coordinate $a = a + \xi(a, \zeta)$

$$\bar{\Phi} = \bar{\Phi}(a). \tag{2.5}$$

After this, $\Phi(a) \equiv 2\pi\bar{\Phi}(a)$ becomes the flux of the magnetic field through the contour $a = const, \zeta = const$. Only the reverse field pinch (RFP) configuration would require $\bar{\Phi} = \bar{\Phi}(a, \zeta)$.

In the function $\bar{\Psi} = \bar{\Psi}(a, \theta, \zeta)$, *it is not possible* to eliminate the dependence on angle coordinates in all situations. The best representation

$$\bar{\Psi} = \bar{\Psi}_{00}(a) + \sum_{m,n} \psi_{mn}(a, \theta, \zeta), \quad \bar{\Psi}_{00}(a) \equiv \frac{1}{4\pi^2} \oint \oint \bar{\Psi} d\theta \; d\zeta \tag{2.6}$$

can be achieved in a special coordinate system, called the Reference Magnetic Coordinates (RMC), where only resonant Fourier harmonics ψ_{mn} enter into $\bar{\Psi}$. In equilibrium configurations these oscillating terms determine the magnetic islands, which make the topology of the magnetic field different from the coordinate system.

The averaged $2\pi\bar{\Psi}_{00}$ is equal to the poloidal Ψ flux of the magnetic field through the contours $a = const, \theta = const$. For the purpose of `ESI` we consider only simple nested magnetic configurations, where

$$\bar{\Psi} = \bar{\Psi}_{00}(a) = \bar{\Psi}(a). \tag{2.7}$$

`ESI` can be easily extended in order to cover the general case (2.6) as well.

The periodic function $\eta = \eta(a, \theta, \zeta)$ is oscillating in both θ, ζ

$$\int \eta d\theta = \int \eta d\zeta = 0. \tag{2.8}$$

For special purposes, it can be eliminated by massaging the angles

$$\theta \to \bar{\theta} = \theta + \alpha, \quad \zeta \to \bar{\zeta} + \beta, \quad \bar{\Phi}'\alpha + \bar{\Psi}'\beta = \bar{\Phi}'\eta \tag{2.9}$$

in order to produce the so-called "straight field line" coordinates

$$\mathbf{A} = \bar{\Phi} \nabla \bar{\theta} + \bar{\Psi} \nabla \varphi. \tag{2.10}$$

"Barred" notations $a, \bar{\theta}, \bar{\zeta}$ are used for the straight field line coordinates.

In RMC, the magnetic field has the following contravariant representation

$$\begin{aligned} &\mathbf{B} = \psi'_\theta (\nabla\theta \times \nabla\zeta) - (\bar{\Psi}' + \psi'_a + \bar{\Phi}'\eta'_\zeta)(\nabla\zeta \times \nabla a) + \bar{\Phi}'(1 + \eta'_\theta)(\nabla a \times \nabla\theta), \\ &B^a \equiv \mathbf{B} \cdot \nabla a = \frac{\psi'_\theta}{J}, \quad B^\theta \equiv \mathbf{B} \cdot \nabla\theta = -\frac{\bar{\Psi}' + \psi'_a + \bar{\Phi}'\eta'_\zeta}{J}, \quad B^\zeta \equiv \mathbf{B} \cdot \nabla\zeta = \frac{\bar{\Phi}'(1 + \eta'_\theta)}{J}, \end{aligned} \tag{2.11}$$

where ψ with resonant harmonics describes magnetic islands. In flux coordinates ($\mathbf{B}\cdot\nabla a = 0$, $\psi = 0$), the magnetic field $\mathbf{B}$ has a simpler form

$$\mathbf{B} = -(\bar{\Psi}' + \bar{\Phi}'\eta'_\zeta)(\nabla\zeta \times \nabla a) + \bar{\Phi}'(1+\eta'_\theta)(\nabla a \times \nabla\theta). \tag{2.12}$$

In 2-D case $\mathbf{B}$ is reduced to

$$\mathbf{B} = \bar{\Psi}'(\nabla a \times \nabla\varphi) + \bar{F}\nabla\varphi. \tag{2.13}$$

The definitions (2.1, 2.12) introduce two (out of a total of four in `ESI`) 1-D profiles

$$\bar{\Phi}'(a), \quad \bar{\Psi}'(a) \tag{2.14}$$

and, together with $|\mathbf{B}|$, four basis functions of `ESI` interface

$$\begin{array}{l} r = r(a,\theta,\zeta), \quad z = z(a,\theta,\zeta), \quad |\mathbf{B}| = B(a,\theta,\zeta), \quad \eta'_\theta \equiv \eta'_\theta(a,\theta,\zeta), \\ \eta'_\zeta \equiv \eta'_\zeta(a,\theta,\zeta), \quad \varphi = \varphi(a,\theta,\zeta). \end{array} \tag{2.15}$$

In `ESI`, the units of r, z are in [m], B is in [T], η, φ are dimensionless, while fluxes Φ, Ψ are in [V·sec]. Units of a, θ, ζ are not essential.

Note, that the basic functions (2.14,2.15) represent any nested magnetic configurations, even unrelated to a static equilibrium.

In magneto-static equilibrium

$$\begin{array}{l} \nabla\bar{p} = \bar{\mathbf{j}} \times \mathbf{B}, \quad \bar{\mathbf{j}} = \nabla \times \mathbf{B}, \quad \bar{p} \equiv \mu_0 p, \quad \bar{\mathbf{j}} \equiv \mu_0 \mathbf{j}, \\ \mathbf{B}\cdot\nabla\bar{p} = 0, \quad \bar{\mathbf{j}}\cdot\nabla a = 0, \end{array} \tag{2.16}$$

where p is the plasma pressure in [MPa], $\mathbf{j}$ is the current density in [MA/m^2], and $\mu_0 = 0.4\pi$, one can introduce two additional 1-D profiles, i.e., $P(a), T(a)$

$$P(a) \equiv \frac{d\bar{p}}{d\bar{\Psi}}, \quad T(a) \equiv \bar{F}\frac{d\bar{F}}{d\bar{\Psi}}, \quad \bar{F} \equiv 0.2F. \tag{2.17}$$

Here $F = 5\bar{F}$ is the total poloidal current in [MA] through the contour $a = const, \theta = const$ (analog of magnetic flux Ψ). In axisymmetric configurations

$$\bar{F} = rB_\varphi. \tag{2.18}$$

`ESI` relies on the fact that four space coordinated functions (2.15) and four 1-D profiles (2.14), (2.17) together with their first derivatives contain all necessary information for transport, stability, particle motion and gyro-kinetic codes. They are called hereafter the "basis" functions.

As a special case, `ESI` accepts 2-D data on a rectangular grid r, z from free boundary codes in the form of $\bar{\Psi}(r,z), \bar{\Psi}'_r, \bar{\Psi}'_z, \bar{\Psi}''_{rz}$. This is not sufficient to generate all possible information for other magnetic confinement codes. The special `ESI` routine can convert $r-z$ data into basis functions of flux coordinates.

`ESI` *reconstruction routines do not provide the second derivatives of basis functions.* The following sections describe the self-consistent rules of using basis function for different kind of numerical codes without use of second derivatives.

3 Main reconstruction routine of `ESI` *(to ToC)*

Technically, `ESI` consists of a data set (file, or a segment of the shared memory), which should be provided by an equilibrium code, and a universal C-source file (`esiZ.c`), which contains `ESI` initiation routines, the main reconstruction routine, and an unspecified set of service routines (see Fig.1)

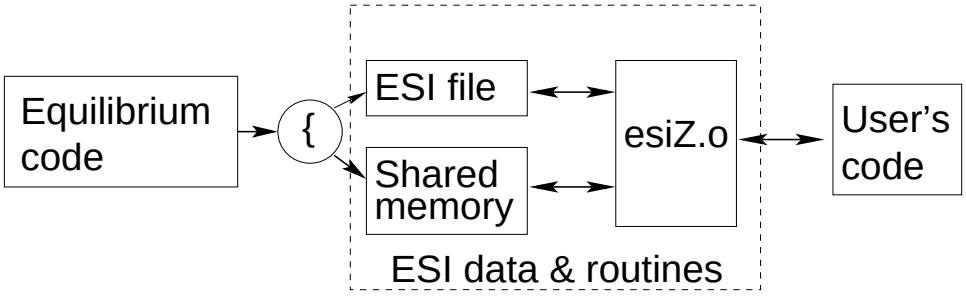

Fig.1. Data generating equilibrium code, `ESI`*, and the user's sides.*

The source code `esiZ.c` is self-contained and can be compiled by a simple command

```
cc -c -o esiZ.o esiZ.c
```

All its routines are `int` functions, which return 0 in the case of success, and non-zero otherwise. In order to initiate `ESI` from a data file (which should be created by an equilibrium code), the user should use the C-code, like

```
if(File2ESI(FileName)){
  printf("Failure: %s is not an ESI-file\n'',FileName);
}
```

or FORTRAN-code, like

```
    external integer file2esi
    if(file2esi(FileName).ne.0) then
      write(*,'(aaa)')'Failure: ',FileName,' is not an ESI-file'
    endif
```

The FORTRAN file name `FileName` *in* `file2esi` ***should contain*** *a space character* `' '` *at the end !*

In the case of using shared memory the corresponding function is `Shmem2ESI(int key)`, where `'key'` is a unique identification number for the memory segment, known to mutually communicating equilibrium and user codes.

The main reconstruction routine of `ESI` takes coordinates from a number of points inside the plasma and puts the values of the basis profiles and space functions into the arrays, allocated at the user side. *The user code should provide the storage for these arrays.* The example is the following C-code

```
int n;
double a[N],gq[N];
double F[N],Fa[N],gFa[N][N],gFaa[N],gYa[N],gYaa[N]
  ,T[N],Ta[N],P[N],Pa[N]
  ,r[N],ra[N],rq[N],z[N],za[N],zq[N]
  ,B[N],Ba[N],Bq[N],gh[N],gha[N],ghq[N];
int isw[N];
double gz[N],rz[N],zz[N],Bz[N],ghz[N];
```

The number N should not be less than the number of points requested in reconstruction. The last line is necessary only for the 3-D case. The FORTRAN analog of the same would be

```
      integer n;
      double precision a(N),gq(N)
      double precision F(N),Fa(N),gFa(N)(N),gFaa(N),gYa(N),gYaa(N)
     &      ,T(N),Ta(N),P(N),Pa(N)
     &      ,r(N),ra(N),rq(N),z(N),za(N),zq(N)
     &      ,B(N),Ba(N),Bq(N),gh(N),gha(N),ghq(N)
      integer isw(N);
      double precision gz(N),rz(N),zz(N),Bz(N),ghz(N);
      external integer link2esi,link2esi3d,esi2all,esi2all3d
```

The meaning of this arrays are specified in the Table 1.

Set of output radial profiles			*Table 1*
C-code	Math	C-code	Math
F[],Fa[]	F, F'	isw[]	index of the particle
gFa[],gFaa[]	Φ', Φ''	gYa[],gYaa[]	Ψ', Ψ''
T[],Ta[]	FF'/Ψ'	P[],Pa[]	$\bar{p}'/\Psi'$
Set of 2-D and 3-D output space functions			
C-code	Math	C-code	Math
r[],ra[],rq[],rz[]	$r, r'_a, r'_\theta, r'_\zeta$	z[],za[],zq[],zz[]	$z, z'_a, z'_\theta, z'_\zeta$
B[],Ba[],Bq[],Bz[]	$B, B'_a, B'_\theta, B'_\zeta$	gh[],gha[],ghq[],ghz[]	$\eta, \eta'_a, \eta'_\theta, \eta'_\zeta$

The addresses of these arrays should be given to ESI, at least once, using

```
i=Link2ESI(F,Fa,gFa,gFaa,gYa,gYaa,T,Ta,P,Pa
          ,r,ra,rq,z,za,zq,B,Ba,Bq,gh,gha,ghq,isw);
i=Link2ESI3d(rz,zz,Bz,ghz);
```

from C-code or from FORTRAN as

```
      i=link2esi(F,Fa,gFa,gFaa,gYa,gYaa,T,Ta,P,Pa
     &   ,r,ra,rq,z,za,zq,B,Ba,Bq,gh,gha,ghq,isw)
      i=link2esi3d_(rz,zz,Bz,ghz)
```

Again, the last line is necessary only for the 3-D case. At present, the main reconstruction routine does not use the index array isw[i], which can be used as a marker of points to control some service routines.

After all of this, the call of the main reconstruction routine is simple, as soon as the coordinates of $\mathtt{n} \le \mathtt{N}$ points a, θ (and ζ in 3-D case) are put into array a[],gq[] (and gz[])

```
i=ESI2all(a,gq,n);
```

or for 3-D case

```
i=ESI2all3d(a,gq,gz,n);
```

The FORTRAN analog of the same is

```
      i=esi2all(a,gq,n);
      i=esi2all3d(a,gq,gz,n);
```

After these calls the information is is the mentioned arrays. In addition to the main reconstruction routine, ESI contains a set of service routine for calculating specialized information. Some of them are listed at the end of this document, and their set can be extended.

4 Metrics of toroidal configurations *(to ToC)*

The metric tensor g_{ik} ($i = 1, 2, 3$, $k = 1, 2, 3$) of the coordinate system $\{a, \theta, \zeta\} = \{x^1, x^2, x^3\}$ is defined by the element of length dl

$$
\begin{aligned}
&dl^2 \equiv dr^2 + dz^2 + r^2 d\varphi^2 = g_{aa} da^2 + 2g_{a\theta} da d\theta + g_{\theta\theta} d\theta^2 + 2g_{a\zeta} da d\zeta + 2g_{\theta\zeta} d\theta d\zeta + g_{\zeta\zeta} d\zeta^2,\\
&g_{aa} = r'_a r'_a + z'_a z'_a + r^2 \zeta'_a \zeta'_a, \quad g_{a\theta} = r'_a r'_\theta + z'_a z'_\theta + r^2 \zeta'_a \zeta'_\theta,\\
&g_{\theta\theta} = r'_\theta r'_\theta + z'_\theta z'_\theta + r^2 \zeta'_\theta \zeta'_\theta, \quad g_{a\zeta} = r'_a r'_\zeta + z'_a z'_\zeta + r^2 \zeta'_a \zeta'_\zeta,\\
&g_{\theta\zeta} = r'_\theta r'_\zeta + z'_\theta z'_\zeta + r^2 \zeta'_\theta \zeta'_\zeta, \quad g_{\zeta\zeta} = r'_\zeta r'_\zeta + z'_\zeta z'_\zeta + r^2 \zeta'_\zeta \zeta'_\zeta,\\
&J = \sqrt{g}, \quad g \equiv \mathrm{Det}(g_{ik}).
\end{aligned}
\tag{4.1}
$$

Gradient vectors of cylindrical and curvilinear coordinates are related with each other by a matrix of first derivatives D

$$
\mathsf{D} \equiv \begin{vmatrix} r'_a & r'_\theta & r'_\zeta \\ \varphi'_a & \varphi'_\theta & \varphi'_\zeta \\ z'_a & z'_\theta & z'_\zeta \end{vmatrix}, \quad \begin{vmatrix} \nabla r \\ \nabla \varphi \\ \nabla z \end{vmatrix} = \mathsf{D} \cdot \begin{vmatrix} \nabla a \\ \nabla \theta \\ \nabla \zeta \end{vmatrix}, \quad \begin{vmatrix} \nabla a \\ \nabla \theta \\ \nabla \zeta \end{vmatrix} = \mathsf{D}^{-1} \cdot \begin{vmatrix} \nabla r \\ \nabla \varphi \\ \nabla z \end{vmatrix}. \tag{4.2}
$$

The Jacobian J of the coordinate system can be calculated as the determinant D of the matrix of first derivatives of cylindrical coordinates

$$D = \mathrm{Det}(\mathsf{D}), \quad J = rD = \sqrt{g}, \tag{4.3}$$

where g is the determinant of the matrix g_{ik}.

In axisymmetric configurations, only $g_{aa}, g_{a\theta}, g_{\theta\theta}, g_{\varphi\varphi}$ are non-vanishing

$$\begin{aligned} &dl^2 \equiv dr^2 + dz^2 + r^2 d\varphi^2 = g_{aa} da^2 + 2 g_{a\theta} da d\theta + g_{\theta\theta} d\theta^2 + r^2 d\varphi^2, \\ &g_{aa} = r'_a r'_a + z'_a z'_a, \quad g_{\theta\theta} = r'_\theta r'_\theta + z'_\theta z'_\theta, \quad g_{\varphi\varphi} = r^2, \quad D = r'_\theta z'_a - r'_a z'_\theta. \end{aligned} \tag{4.4}$$

`ESI` reconstruction routines provide all necessary derivatives for calculation of g_{ik}, elements of matrix D, and Jacobian J at the user side. *The derivatives of* g_{ik}, D, J *are not defined by* `ESI`.

4.1 Behavior near the origin $a = 0$ *(to ToC)*

In configurations with a smooth current density near the axis $a = 0$, flux coordinates have a special behavior near the axis. Thus, in some coordinate systems (n.n. 0,1,2,3,7 in Table 2), Fourier coefficients of $r(a, \theta, \zeta), z(a, \theta, \zeta)$

$$r = r_0 + \sum_{m=1}^{m \le M_\theta} \left[r_m e^{im\theta} + r_m^* e^{-im\theta} \right], \quad z = z_0 + \sum_{m=1}^{m \le M_\theta} \left[z_m e^{im\theta} + z_m^* e^{-im\theta} \right] \tag{4.5}$$

vanish at $a = 0$ at a certain rate. If $b = b(a)$ is a function proportinal to the distance from the axis, then asymptotically

$$r_0(a, \zeta) \simeq R_0(\zeta) + \bar{r}_0(\zeta) b^2, \quad r_1(a, \zeta) \simeq \bar{r}_1(\zeta) b, \quad r_2(a, \zeta) \simeq \bar{r}_m(\zeta) b^2, \quad r_{m>2}(a, \zeta) \simeq \bar{r}_m(\zeta) b^m. \tag{4.6}$$

The `ESC` *data generated by the equilibrium codes in these coordinates should reflect this behaviour for* $r_0, r_1, r_2, z_0, z_1, z_2$. All other coefficients should vanish near the axis, at least, like b^2. This would allow to resolve the details, essential for some instabilities.

In other coordinates like polar ($\mathtt{ID}_\theta$=8 in Table 2), otherwise very good, such a bahavior is hidden more deeply. This may slightly limit the application of `ESI` in such coorinates.

4.2 Conversion of laboratory r, φ, z into a, θ, ζ *(to ToC)*

For the purpose of diagnostics on the experimental machines, it is necessary to have an ablity of conversion of laboratory coordinates r, φ, z into curvilinear coordinates a, θ, ζ. This is done in `ESI` by the Newton iterative procedure

$$\begin{vmatrix} da^{(k)} \\ d\theta^{(k)} \\ d\zeta^{(k)} \end{vmatrix} = \left(\mathsf{D}^{-1} \right)^{(k)} \cdot \begin{vmatrix} r - r^{(k)} \\ \varphi - \varphi^{(k)} \\ z - z^{(k)} \end{vmatrix}, \quad \begin{aligned} a^{(k+1)} &= a^{(k)} + da^{(k)} \\ d\theta^{(k+1)} &= \theta^{(k)} + d\theta^{(k)}, \\ d\zeta^{(k+1)} &= \zeta^{(k)} + d\zeta^{(k)} \end{aligned} \tag{4.7}$$

where k is the iteration counter.

5 Nested magnetic configurations *(to ToC)*

The vector potential $\mathbf{A}$ in Eq.(2.4) contains integrals from the basis profiles of `ESI`

$$\Phi = 2\pi \int_0^a \bar{\Phi}'(a) da, \quad \Psi = 2\pi \int_0^a \bar{\Psi}'(a) da, \quad \eta = \int_0^\theta \eta'_\theta d\theta + \int_0^\zeta \left(\frac{1}{2\pi} \oint \eta'_\zeta d\theta \right) d\zeta. \tag{5.1}$$

There is a special service routine in `ESI` which calculates Ψ, Φ for the transport codes, while η and its derivatives are calculated by the main reconstruction routine.

5.1 Magnetic field *(to ToC)*

The contravariant representation of the magnetic field is given by Eq.(2.12), or in the 2-D case by Eq.(2.13). It can be reconstructed from `ESI` basis functions in a straightforward manner. On the other hand the derivatives of individual components of the magnetic field are not defined.

The covariant components of magnetic field B_i (integer subscript i is used to distinguish them from the physical components),

$$\mathbf{B} = B_1\nabla a + B_2\nabla\theta + B_3\nabla\zeta, \quad B_i = g_{ik}B^k, \quad i = 1,2,3, \quad k = 1,2,3 \tag{5.2}$$

are useful for particle motion. Herethe summation convention is assumed.

In flux coordinates (no islands) $B^1 = B^a = 0$ and only the following metric tensor combinations are present in Eq.(5.2)

$$K \equiv \frac{g_{\theta\theta}}{J}, \quad M \equiv \frac{g_{aa}}{J}, \quad N \equiv \frac{g_{a\theta}}{J}, \quad N_{a\zeta} \equiv \frac{g_{a\zeta}}{J}, \quad N_{\theta\zeta} \equiv \frac{g_{\theta\zeta}}{J}, \quad L \equiv \frac{J}{g_{\zeta\zeta}}. \tag{5.3}$$

All of them can be calculated at the user side using the relationship between co- and contra-variant components of the vector $\mathbf{B}$.

In the case of an "ideal" (isotropic) equilibrium, the covariant components have a special form

$$\mathbf{B} = (\nu + \sigma_a')\nabla a + (\bar{I} + \sigma_\theta')\nabla\theta + (\bar{F} + \sigma_\zeta')\nabla\zeta \tag{5.4}$$

and can be calculated using

$$\begin{aligned}
\nu + \sigma_a' &= -N(\bar{\Psi}' + \bar{\Phi}'\eta_\zeta') + N_{a\zeta}\bar{\Phi}'(1+\eta_\theta'),\\
\bar{I} + \sigma_\theta' &\equiv -K(\bar{\Psi}' + \bar{\Phi}'\eta_\zeta') + N_{\theta\zeta}\bar{\Phi}'(1+\eta_\theta'),\\
\bar{F} + \sigma_\zeta' &= -N_{\theta\zeta}(\bar{\Psi}' + \bar{\Phi}'\eta_\zeta') + \frac{1+\eta_\theta'}{L}\bar{\Phi}'.
\end{aligned} \tag{5.5}$$

For the case of axisymmetry

$$\nu + \sigma_a' = -N\bar{\Psi}', \quad \bar{I} + \sigma_\theta' \equiv -K\bar{\Psi}', \quad \bar{F} = \frac{1+\eta_\theta'}{L}\bar{\Phi}', \quad L = \frac{D}{r}. \tag{5.6}$$

5.2 Current density *(to ToC)*

In equilibrium configurations the current density $\bar{\mathbf{j}}$ has the following contravariant form

$$\bar{\mathbf{j}} = -(\bar{F}' - \nu_\zeta')(\nabla\zeta\times\nabla a) + (\bar{I}' - \nu_\theta')(\nabla a\times\nabla\theta), \tag{5.7}$$

where $I(a)$ is the toroidal current through the poloidal cross-section of magnetic surfaces. Both $\bar{F}, \bar{F}'$, together with $\bar{p}, \bar{p}'$ are calculated inside `ESI` using

$$\bar{F}' = \frac{T\bar{\Psi}'}{\bar{F}}, \quad \bar{F}(a) = \sqrt{\bar{F}_{ref}^2 + 2\int_{a_{ref}}^{a} T(a)\bar{\Psi}'da}, \quad \bar{p}' = P\bar{\Psi}', \quad \bar{p}(a) = \int_{a_v}^{a} P\bar{\Psi}'da, \quad . \tag{5.8}$$

Here, a_{ref} is a reference radius, where the function $\bar{F}$ has a prescribed value, $\bar{F}(a_{ref}) = \bar{F}_{ref}$. Typically a_{ref} corresponds to the plasma boundary. The exceptional case is the RFP configurations, where a_{ref} should specify the surface with $\bar{F}(a_{ref}) = 0$. The main `ESI` reconstruction routine delivers $\bar{F}, \bar{F}'$, while $\bar{p}, \bar{p}'$ are provided by a special service routine. *For RFP case the sign of $\bar{F}$ cannot be determined from the* `ESI` *data. It is chosen using convention $\bar{F}(0) > 0$.*

The radial component of equilibrium equation (2.16) (the pressure balance)

$$J\bar{p}' = -(\bar{\Phi}' + \bar{\Phi}'\eta_\theta')(\bar{F}' - \nu_\zeta') + (\bar{\Psi}' + \bar{\Phi}'\eta_\zeta')(\bar{I}' - \nu_\theta') \tag{5.9}$$

determines $\bar{I}'(a)$ inside `ESI` from the averaged Eq.(5.9)

$$PJ_{00} = -\bar{\Phi}'\frac{T}{\bar{F}} + \bar{I}', \quad J_{00} \equiv \frac{1}{4\pi^2}\oint\oint J d\theta d\zeta, \quad J \equiv J_{00} + J_{\sim\sim}. \tag{5.10}$$

The oscillatory part of this equation, having the form of the so-called magnetic differential equation (MDE)

$$(\bar{\Phi}' + \bar{\Phi}'\eta'_\theta)\nu'_\zeta - (\bar{\Psi}' + \bar{\Phi}'\eta'_\zeta)\nu'_\theta = \bar{p}' J_{\sim\sim} + \bar{\Phi}'(\bar{F}'\eta'_\theta - \bar{I}'\eta'_\zeta) \tag{5.11}$$

allows one to determine the function ν. In the case of axisymmetry this is simply given by an integral

$$\bar{I}' - \nu'_\theta = TL + PJ, \quad \nu'_\theta = -TL_{\sim\sim} - PJ_{\sim\sim}, \quad \nu = -\int_0^\theta (TL_{\sim\sim} + PJ_{\sim\sim})d\theta. \tag{5.12}$$

In the case of 3-D equilibrium configurations, it necessary to solve MDE (5.11). In configurations with simple nested surfaces (with no islands), the Jacobian does not contain resonance harmonics, and MDE for ν has a nonsingular periodic solution. This is described in the next section. A special service routine of `ESI` generates ν.

Thus, all physical quantities in the equilibrium configurations can be calculated from `ESI` basis functions without using their second derivatives.

Note, that the output information from `ESI` can appear to be inconsistent. For example, in 2-D, the third equation in (5.6) gives the relationship

$$\bar{\Phi}' = L_0\bar{F}, \quad L_0 \equiv \frac{1}{2\pi}\oint\frac{D}{r}d\theta, \tag{5.13}$$

which gives an alternative to the Eq.(5.8) way of calculating $\bar{F}$. The level of discrepancy between the two calculations may be used for testing the accuracy of `ESI` data.

In fact, the user has no information from `ESI` on how to perform integration without interpolation of the integrand, which makes such comparison "illegal" within `ESI` rules. Instead, the main reconstruction routine of `ESI` provides the values of $\bar{F}, \bar{F}', \bar{\Phi}, \bar{\Phi}'$ explicitly, thus, giving L_0, L'_0. Alternatively, a specially written service routine of `ESI` can provide L_0, L'_0, consistent with relation (5.13).

5.3 Magnetic differential equation (MDE) *(to ToC)*

Assume straight field line coordinates a, θ, ζ with $\eta = 0$. Then, the magnetic differential equation (5.11) has a solution

$$\begin{aligned} &q \equiv -\frac{\bar{\Phi}'}{\bar{\Psi}'}, \quad \nu'_\theta + q\nu'_\zeta = P(a)\tilde{J}(a,\theta,\zeta) \equiv S(a,\theta,\zeta), \\ &\nu(a,\theta,\zeta) = A(a,\zeta - q\theta) - \int_0^\theta S(a,\alpha,\zeta + q\alpha - q\theta)d\alpha, \\ &A(a,\zeta) - A(a,\zeta - 2\pi q) = \int_0^{2\pi} S(a,\alpha,\zeta + q\alpha - 2\pi q)d\alpha, \end{aligned} \tag{5.14}$$

where a 2-D function $A(a,\zeta)$ is added in order to provide the periodicity of ν as function of θ. The equation for A can be solved directly numerically in Fourier space

$$S(a,\theta,\zeta) \equiv \sum_n S_n(a,\theta)e^{in\zeta}, \quad A(a,\zeta) \equiv \sum_n A_n(a)e^{in\zeta}, \quad A_n(a) = \frac{e^{-in\pi q}}{2\sin n\pi q}\int_0^{2\pi} S_n(a,\alpha)e^{inq\alpha}d\alpha. \tag{5.15}$$

In the general case when $\eta \neq 0$, the equation for ν (5.11) can be written as

$$q(1+\eta'_\theta)\nu'_\zeta + (1 - q\eta'_\zeta)\nu'_\theta = \tilde{J}P - q(\bar{F}'\eta'_\theta - \bar{I}'\eta'_\zeta) \equiv (1+\eta'_\theta)S(a,\theta,\zeta). \tag{5.16}$$

A transition to a new variable $\bar{\theta} = \theta + \eta$ in $\nu(a, \theta + \eta, \zeta)$ leads to

$$q\nu'_\zeta + \nu'_{\bar{\theta}} = S(a, \bar{\theta} - \eta, \zeta), \tag{5.17}$$

whose solution is given by Eqs.(5.14,5.15).

Because of resonant denominators $nq = m$ (m, n are integers), the function ν may have a rather complicated behavior even when the MDE can be resolved. It cannot be well represented by smooth polynomials on a finite grid, and its calculations cannot be delegated to the user. Specially designed `ESI` routines should reconstruct ν upon the user's request.

5.4 Curvature of the magnetic field lines *(to ToC)*

The curvature of the magnetic field lines, essential for some instabilities,

$$\mathbf{k} \equiv \left(\frac{\mathbf{B}}{|B|}\cdot\nabla\right)\frac{\mathbf{B}}{|B|} = -\left(\frac{\mathbf{B}}{|B|}\times\left(\nabla\times\frac{\mathbf{B}}{|B|}\right)\right) = \frac{\bar{p}'}{|B|^2}\nabla a - \frac{((\mathbf{B}\cdot\nabla)|B|)}{|B|^3}\mathbf{B} + \frac{|B|'_a}{|B|}\nabla a + \frac{|B|'_\theta}{|B|}\nabla\theta \tag{5.18}$$

can be calculated at the user's side using the data from `ESI` reconstruction routines.

5.5 Evolution equation for magnetic fluxes *(to ToC)*

In evolving magnetic configurations, the electric field $\mathbf{E}$ can be written as

$$\mathbf{E} = -\frac{\partial \mathbf{A}}{\partial t} - \nabla\Phi_E. \tag{5.19}$$

Elimination of the scalar potential $\nabla\Phi_E$ from the classical parallel Ohm's law

$$\sigma_\parallel(\mathbf{B}\cdot\mathbf{E}) = (\mathbf{B}\cdot\mathbf{j}) \tag{5.20}$$

using averaging gives the evolution equation for magnetic fluxes

$$\bar{\Phi}'_a\Psi'_t - \bar{\Psi}'_a\Phi'_t = \frac{1}{\sigma_\parallel}(\bar{F}\bar{I}' - \bar{I}\bar{F}'). \tag{5.21}$$

It can be rewritten as a magnetic diffusion equation

$$\begin{aligned}
&\Psi'_t - \frac{\bar{\Psi}'_a}{\bar{\Phi}'_a}\Phi'_t = \frac{1}{\sigma_\parallel}\left[\frac{\bar{\Phi}'\bar{F} - \bar{\Psi}'\bar{I}}{\bar{\Phi}'^2}\left(\frac{g_{\theta\theta}}{\sqrt{g}}(\Psi' + \eta'_\zeta) - \frac{g_{\theta\zeta}}{\sqrt{g}}(\Phi' + \eta'_\theta)\right)'_{00} - \frac{\bar{I}\bar{p}'}{\bar{\Phi}'^2}(\sqrt{g})_{00}\right], \\
&\bar{I} = -\left(\frac{g_{\theta\theta}}{\sqrt{g}}(\Psi' + \eta'_\zeta) - \frac{g_{\theta\zeta}}{\sqrt{g}}(\Phi' + \eta'_\theta)\right)_{00}.
\end{aligned} \tag{5.22}$$

The averaging can be performed on the user's side, while `ESI` reconstruction provides data for all coefficients in this equation.

When the radial coordinate a represents the toroidal magnetic flux (e.g., in the ASTRA code $a = \sqrt{\frac{2\Phi}{B_0}}$, $B_0 = \text{const}$), the convective term in the left hand side vanishes, $\bar{\Phi}'_t = 0$.

5.6 Transport equations *(to ToC)*

The structure of the transport equations is given by

$$\frac{\partial}{\partial t}(Jn)_{00} + \frac{\partial}{\partial a}\left(J\vec{\Gamma}\cdot\nabla a\right)_{00} = (JS)_{00}, \quad J \equiv \sqrt{g}, \tag{5.23}$$

where n is a physics variable, Γ is its flux through the magnetic surface (both diffusive and convective), and S are the sources and sinks.

`ESI` reconstruction routines provide the necessary information for calculating coefficients of the transport equations on the user's side. In contrast, the approach based on "cooking" out all the inforamtion inside the equilibrium codes is always deficient due to the unpredictable nature of plasma transport models.

5.7 Transition to the straight field line (SFL) coordinates *(to ToC)*

Straight field line coordinates are important for stability theory and for stellarator equilibrium calculations. `ESI` reconstruction routines provide the necessary information for making transition to the SFL coordinates $a, \theta, \bar{\zeta}$

$$\begin{aligned} &\theta = \theta - \alpha(a,\theta,\bar{\zeta}), \quad \zeta = \bar{\zeta} - \beta(a,\theta,\bar{\zeta}), \\ &d\theta = (1-\alpha'_\theta)d\theta - \alpha'_a da - \alpha'_\zeta d\bar{\zeta}, \quad d\zeta = (1-\beta'_{\bar{\theta}})d\bar{\theta} - \beta'_a da - \beta'_\theta d\theta, \end{aligned} \tag{5.24}$$

where either α or β is an arbitrary function, related by Eq. (5.24)

$$\bar{\Phi}'\alpha + \bar{\Psi}'\beta = \bar{\Phi}'\eta. \tag{5.25}$$

Substitution of $d\theta, d\zeta$ (5.24) info element of the length (4.1) allows calculation of the metric tensor in new coordinates. Thus, with the function η in the set of basis function the `ESI` main routine provides information for transition into any SFL coordinates.

6 Field lines and particle motion *(to ToC)*

6.1 Equation for magnetic field lines *(to ToC)*

In curvilinear coordinates, the magnetic field lines are determined by three ordinary differential equations

$$\frac{d\theta}{dl} = \frac{B^\theta}{|B|}, \quad \frac{d\zeta}{dl} = \frac{B^\zeta}{|B|}, \quad \frac{da}{dl} = \frac{B^a}{|B|}, \tag{6.1}$$

where B^a, B^θ, B^ζ are given by Eq.(2.11). In the case of of magnetic configurations with the nested surfaces, a =const, and only the two first equations are essential. `ESI` provides a special step advancing routine `ESI2mfl()` for calculating derivatives for further use in ODE solvers.

6.2 Guiding center Lagrangian equations *(to ToC)*

The guiding center Lagrangian obtained by Littlejohn has the following normalized form

$$\begin{aligned} &L = \left(\mathbf{A} + \rho_\parallel \mathbf{B}\right)\cdot \mathbf{v} - H, \quad \mathbf{v} \equiv \frac{d\mathbf{r}}{d\tau}, \quad \tau \equiv \Omega_c t, \quad \Omega_c \equiv \frac{eB_0}{mc}, \\ &H \equiv \rho_\parallel^2 \frac{B^2}{2} + \mu B + \phi_E, \quad \mu \equiv \frac{\rho_\perp^2}{2B}, \quad \phi_E \equiv \frac{mc^2\Phi_E}{e} \equiv \rho_E R_0 B_0^2, \quad \rho_E \equiv \frac{c\Phi_E}{R_0 B_0 \Omega_c}. \end{aligned} \tag{6.2}$$

Here, $\mathbf{v}$ is considered as the normalized time derivative of the particle position, Ω_c is the ion cyclotron frequency in the reference magnetic field B_0 whatever is used for normalization (e.g., for `ESI` Units, $B_0 = 1$ T and $\Omega_c = 95.79 \cdot 10^{+6}$ sec^{-1} for protons), $\rho_\parallel, \rho_\perp$ are the ion Larmor radii calculated based on particle velocities parallel and perpendicular to the field. In the Hamiltonian H the last term ϕ_E is associated with the electric field potential Φ_E and can be expressed in terms of the Larmor radius ρ_E calculated based on drift velocity in the effective electric field Φ_E/R_0 where R_0 is some characteristic unit length (e.g., 1 m).

Written for generalized coordinates of a particle a, θ, ζ, the Lagrangian has the form

$$\begin{aligned} &L = P_a\dot{a} + P_\theta\dot{\theta} + P_\zeta\dot{\zeta} - H, \\ &P_a \equiv \rho_\parallel B_a, \quad P_\theta \equiv \bar{\Phi} + \int_0^a \bar{\Phi}'\eta'_\theta da + \rho_\parallel B_\theta, \quad P_\zeta \equiv \bar{\Psi} + \int_0^a \bar{\Phi}'\eta'_\zeta da + \rho_\parallel B_\zeta. \end{aligned} \tag{6.3}$$

The straightforward variation of L with respect to $\rho_\parallel, a, \theta, \zeta$ leads to the following system of coupled equations

of motion

$$\begin{vmatrix} 0 & B_a & B_\theta & B_\zeta \\ -B_a & 0 & J(B^\zeta+\rho_\parallel\bar{j}^\zeta) & -J(B^\theta+\rho_\parallel\bar{j}^\theta) \\ B_\theta & J(B^\zeta+\rho_\parallel\bar{j}^\zeta) & 0 & -J(B^a+\rho_\parallel\bar{j}^a) \\ B_\zeta & -J(B^\theta+\rho_\parallel\bar{j}^\theta) & J(B^a+\rho_\parallel\bar{j}^a) & 0 \end{vmatrix} \cdot \begin{vmatrix} \dot{\rho}_\parallel \\ \dot{a} \\ \dot{\theta} \\ \dot{\zeta} \end{vmatrix} = \begin{vmatrix} \dfrac{\partial H}{\partial \rho_\parallel} \\ \dfrac{\partial H}{\partial a} \\ -\dfrac{\partial H}{\partial \theta} \\ -\dfrac{\partial H}{\partial \zeta} \end{vmatrix}. \tag{6.4}$$

or to wrong ones

$$\begin{vmatrix} 0 & B_a & B_\theta & B_\zeta \\ -B_a & 0 & \sqrt{g}(B^\zeta+\rho_\parallel j^\zeta) & -\sqrt{g}(B^\theta+\rho_\parallel j^\theta) \\ B_\theta & \sqrt{g}(B^\zeta+\rho_\parallel j^\zeta) & 0 & 0 \\ B_\zeta & -\sqrt{g}(B^\theta+\rho_\parallel j^\theta) & 0 & 0 \end{vmatrix} \cdot \begin{vmatrix} \dot{\rho}_\parallel \\ \dot{a} \\ \dot{\theta} \\ \dot{\zeta} \end{vmatrix} = \begin{vmatrix} \dfrac{\partial H}{\partial \rho_\parallel} \\ \dfrac{\partial H}{\partial a} \\ -\dfrac{\partial H}{\partial \theta} \\ -\dfrac{\partial H}{\partial \zeta} \end{vmatrix}. \tag{6.5}$$

Using the matrix

$$\mathsf{A} \equiv \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}, \quad |\mathsf{A}| \equiv a_{11}a_{22} - a_{12}a_{21} = \sqrt{g}[|B|^2 + \rho_\parallel(\mathbf{B}\cdot(\nabla\times\mathbf{B}))], \tag{6.6}$$

where

$$a_{11} \equiv B_\theta, \quad a_{12} \equiv \sqrt{g}(B^\zeta + \rho_\parallel j^\zeta), \quad a_{21} \equiv B_\zeta, \quad a_{22} \equiv -\sqrt{g}(B^\theta + \rho_\parallel j^\theta), \tag{6.7}$$

the equation of drift motion can be written in the explicit form

$$\begin{vmatrix} \dot{\rho}_\parallel \\ \dot{a} \\ \dot{\omega} \\ \dot{\zeta} \end{vmatrix} = \begin{vmatrix} 0 & 0 & \dfrac{a_{22}}{|\mathsf{A}|} & -\dfrac{a_{12}}{|\mathsf{A}|} \\ 0 & 0 & -\dfrac{a_{21}}{|\mathsf{A}|} & \dfrac{a_{11}}{|\mathsf{A}|} \\ \dfrac{a_{22}}{|\mathsf{A}|} & -\dfrac{a_{21}}{|\mathsf{A}|} & 0 & -\dfrac{B_a}{|\mathsf{A}|} \\ -\dfrac{a_{12}}{|\mathsf{A}|} & \dfrac{a_{11}}{|\mathsf{A}|} & \dfrac{B_a}{|\mathsf{A}|} & 0 \end{vmatrix} \cdot \begin{vmatrix} \dfrac{\partial H}{\partial \rho_\parallel} \\ \dfrac{\partial H}{\partial a} \\ -\dfrac{\partial H}{\partial \theta} \\ -\dfrac{\partial H}{\partial \zeta} \end{vmatrix}. \tag{6.8}$$

`ESI` allows calculation of the right hand side of the guiding center equations at any point inside the plasma without interpolations. In fact, `ESI` contains a guiding center motion routine which calculates the derivatives $\dot{\rho}_\parallel, \dot{a}, \dot{\omega}, \dot{\zeta}$ given $\rho_\parallel, a, \theta, \zeta$.

6.3 Hamiltonian equations for guiding center motion *(to ToC)*

By transformation of angle coordinates it is possible to eliminate the $B_a = \nu + \sigma_a$ term in the covariant representation of $\mathbf{B}$ (5.4) together with the P_a term in Lagrangian. For this purpose, new, canonical coordinates $\hat{\theta}, \hat{\zeta}$ are determined by

$$\hat{\theta} = \theta + \alpha, \quad \hat{\zeta} = \zeta + \beta, \tag{6.9}$$

where α, β satisfy equations

$$\begin{aligned}
&\nu\nabla a + \nabla\sigma + \nabla(\bar{I}\alpha) + \nabla(\bar{F}\beta) - \bar{I}'\alpha\nabla a - \bar{F}'\beta\nabla a = 0,\\
&\bar{I}'\alpha + \bar{F}'\beta = \nu, \quad \bar{I}\alpha + \bar{F}\beta = -\sigma, \quad \alpha = -\frac{\bar{F}\nu + \bar{F}'\sigma}{\bar{F}\bar{I}' - \bar{I}\bar{F}'}, \quad \beta = \frac{\bar{I}\nu + \bar{I}'\sigma}{\bar{F}\bar{I}' - \bar{I}\bar{F}'}.
\end{aligned} \tag{6.10}$$

In canonical coordinates $a, \hat{\theta}, \hat{\zeta}$ the covariant representation of the magnetic field is very simple

$$\mathbf{B} = \bar{I}\nabla\hat{\theta} + \bar{F}\nabla\hat{\zeta}. \tag{6.11}$$

The vector potential becomes

$$\mathbf{A} = -\bar{\Phi}'\hat{\eta}\nabla a + \bar{\Phi}\nabla\hat{\theta} + \bar{\Psi}\nabla\hat{\zeta}, \quad \bar{\Phi}'\hat{\eta} \equiv \bar{\Phi}'\eta - \bar{\Phi}'\alpha - \bar{\Psi}'\beta \tag{6.12}$$

Accordingly, the Lagrangian is reduced to the Hamiltonian form

$$L = P_{\hat{\theta}}\dot{\hat{\theta}} + P_{\hat{\zeta}}\dot{\hat{\zeta}} - H, \quad P_{\hat{\theta}} \equiv \bar{\Phi} + \int_0^a \bar{\Phi}'\hat{\eta}'_\theta da + \rho_\parallel \bar{I}, \quad P_{\hat{\zeta}} \equiv \bar{\Psi} + \int_0^a \bar{\Phi}'\hat{\eta}'_\zeta da + \rho_\parallel \bar{F}. \tag{6.13}$$

with no explicit time derivative $\dot{a}$. The equation of motion for a, θ, ζ can be now written as

$$\begin{vmatrix} 0 & 0 & \dfrac{\partial P_{\hat{\theta}}}{\partial \rho_\parallel} & \dfrac{\partial P_{\hat{\zeta}}}{\partial \rho_\parallel} \\ 0 & 0 & \dfrac{\partial P_{\hat{\theta}}}{\partial a} & \dfrac{\partial P_{\hat{\zeta}}}{\partial a} \\ \dfrac{\partial P_{\hat{\theta}}}{\partial \rho_\parallel} & \dfrac{\partial P_{\hat{\theta}}}{\partial a} & 0 & 0 \\ \dfrac{\partial P_{\hat{\zeta}}}{\partial \rho_\parallel} & \dfrac{\partial P_{\hat{\zeta}}}{\partial a} & 0 & 0 \end{vmatrix} \cdot \begin{vmatrix} \dot{\rho}_\parallel \\ \dot{a} \\ \dot{\hat{\theta}} \\ \dot{\hat{\zeta}} \end{vmatrix} = \begin{vmatrix} \dfrac{\partial H}{\partial \rho_\parallel} \\ \dfrac{\partial H}{\partial a} \\ -\dfrac{\partial H}{\partial \hat{\theta}} \\ -\dfrac{\partial H}{\partial \hat{\zeta}} \end{vmatrix}. \tag{6.14}$$

These equations are equivalent to Hamiltonian equations, if momenta $P_{\hat{\theta}}, P_{\hat{\zeta}}$ are used instead of $\rho_\parallel, a$

$$\dot{\theta} = \frac{\partial H}{\partial P_{\hat{\theta}}}, \quad \dot{\zeta} = \frac{\partial H}{\partial P_{\hat{\zeta}}}, \quad \dot{P}_{\hat{\theta}} = -\frac{\partial H}{\partial \hat{\theta}}, \quad \dot{P}_{\hat{\zeta}} = -\frac{\partial H}{\partial \hat{\zeta}}. \tag{6.15}$$

The Hamiltonian form of the equations of motion guarantees that the guiding center motion not only preserves the value of Hamiltonian H (and, in the case of symmetry, the momentum $P_{\hat{\zeta}}$), but also preserves the phase volume of an ensemble of particles in the phase space $\hat{\theta}, \hat{\zeta}, P_{\hat{\omega}}, P_{\hat{\zeta}}$.

For calculations of guiding center motion in unperturbed equilibrium configurations the Hamiltonian form does not suggest any advantage with respect to the Lagrangian form. Canonical variables are essential only for studies of perturbed particle motion, where adiabatic invariants play the key role.

Note, that there are other choices of canonic variables than Eq.(6.10). Indeed, in order to eliminate B_a it is sufficient to satisfy only the ∇a component of the first equation in Eq.(6.10)

$$\nu + \sigma'_a + \bar{I}\alpha'_a + \bar{F}\beta'_a = 0, \quad \beta = -\int_0^a \frac{\nu + \sigma'_a + \bar{I}\alpha'_a}{\bar{F}} da. \tag{6.16}$$

The case with $\alpha = 0$ was pointed out earlier. The attempt to use the freedom for elimination of the η term in the vector potential (making SFL canonical coordinates) leads to substantial, unjustified complications in calculations. The canonic variables $a, \hat{\theta}, \hat{\zeta}$, suggested by Eq.(6.10) represent the best choice.

`ESI` interface is capable of providing a transition to canonical coordinates $a, \hat{\theta}, \hat{\zeta}$ and to a Hamiltonian representation of guiding center motion.

7 The MHD energy principle *(to ToC)*

The well-known functional of potential energy in ideal magneto-hydrodynamics

$$W = \frac{1}{8\pi}\int \left\{\widetilde{\mathbf{B}}^2 + (\vec{\xi}\cdot\nabla\bar{p})(\nabla\cdot\vec{\xi}) - \vec{\xi}\times(\nabla\times\mathbf{B})\cdot\widetilde{\mathbf{B}} + \gamma_0\bar{p}(\nabla\cdot\vec{\xi})^2\right\} dV, \quad \widetilde{\mathbf{B}} \equiv (\mathbf{B}\times(\nabla\times\vec{\xi})), \tag{7.1}$$

where $\vec{\xi}$ is the test function, does not contain anything that cannot be calculated using basis functions of `ESI`. Also, the kinetic energy term

$$K = \gamma^2 \int \rho\vec{\xi}^2 dV \tag{7.2}$$

needs only the metric tensor for its calculation.

At a deeper level, instead of three contravariant components of $\vec{\xi}$

$$\xi^{a,\theta,\zeta} \equiv \left\{\xi, \xi^\theta, \xi^\zeta\right\}, \quad \xi \equiv \xi^a \tag{7.3}$$

another set of test functions is used, i.e, ξ, Λ, X, where

$$X \equiv (\nabla\cdot\vec{\xi}), \quad \Lambda \equiv \bar{\Phi}'\xi^\theta + \bar{\Psi}'\xi^\zeta. \tag{7.4}$$

In terms of ξ and Λ the perturbation of the magnetic field can be written as

$$J\widetilde{\mathbf{B}} = \Lambda(\nabla\theta\times\nabla\zeta) - \bar{\Phi}'\xi(\nabla\zeta\times\nabla a) - \bar{\Psi}'\xi(\nabla a\times\nabla\theta). \tag{7.5}$$

In new variables the potential energy has the form

$$\begin{aligned} W = \frac{1}{8\pi}\int \Big\{ &\frac{\gamma_0\bar{p}}{\sqrt{g}}X^2 + \sqrt{g}\left(\tilde{B}_a\tilde{B}^a + \tilde{B}_\theta\tilde{B}^\theta + \tilde{B}_\zeta\tilde{B}^\zeta\right) - P'\bar{\Psi}'\sqrt{g}\xi^2 + [P\bar{\Psi}'\sqrt{g}\xi^2]'_a \\ &+2(\bar{I}' - \nu'_\theta)\Lambda'_\zeta\xi - 2(\bar{F}' - \nu'_\zeta)\Lambda'_\theta\xi + q'(\bar{F}' - \nu'_\zeta)\bar{\Psi}'\xi^2\Big\}\, da d\theta d\zeta. \end{aligned} \tag{7.6}$$

`ESI` reconstruction routines provide sufficient data for calculating the coefficients in this functional on the user side.

8 Storage and `ESI` format of the equilibrium output file *(to ToC)*

Full description on the data structure and `ESI` routines can be found in the `esiZ.c.d` documentation file. This section specifies what information should be provided by the equilibrium codes and in what format in order to be interfaced by `ESI`.

Both binary and ASCII files are suitable for storing the `ESI` data. The suggested names of the data files are constructed as

`esiE.00` for ASCII data-file, or a pair of

`esiE.00b` binary data-file and `esiE.00m` (ASCII map-file),

The extension `.00` specifies the sequence number. The pair of binary data- and map files has `b` and `m` at the end of the extension. The capital letters in the name stand for different coordinate systems. The suggested choices are listed in Table 2.

$\mathtt{ID}_\theta$ **of angle** θ			*Table 2*
$\mathtt{ID}_\theta$	Symbol	Name	Specific property
0	`E`	ESC	$a, \theta, \varphi,\ r = r(a,\theta),\ z = z_0(a) + b(a)\sin\theta$
1	`K`	KINX, polar	$r = r_0 + \rho(a,\theta,\zeta)\cos\theta,\ z = z_0 + \rho(a,\theta,\zeta)\sin\theta$
2	`H`	Hamada	$J = J(a)$
3	`P`	PEST	$a, \theta, \varphi,\ J = f(a)r^2$
4	`B`	Boozer	SFL $a, \bar{\theta}, \bar{\zeta}$ with $\eta = 0$ and $J = \frac{f(a)}{\lvert B\rvert^2}$
5	`L`	Laboratory rz	$\bar{\Psi}(r,z), \bar{\Psi}'_r(r,z), \bar{\Psi}'_z(r,z)$
6	`C`	Canonical	$a, \hat{\theta}, \hat{\zeta}$, Eqs.(6.9,6.10)
9	`U`	General	unspecified general coordinates, Eq.(2.1)
10+*	`1*`	VMEC-like	coordinates with separation of odd and even parts, Eq.(2.1)

Table 2 introduces also a numerical identification number $\texttt{ID}_\theta$ depending on the choice of the polidal angle. It should be ennhanced by 10, if there is a separation of the odd and even parts in presentation of r, z coordinates. Correspondingly the digit '1' is recommended in the names of files before the symbol of coordiantes.

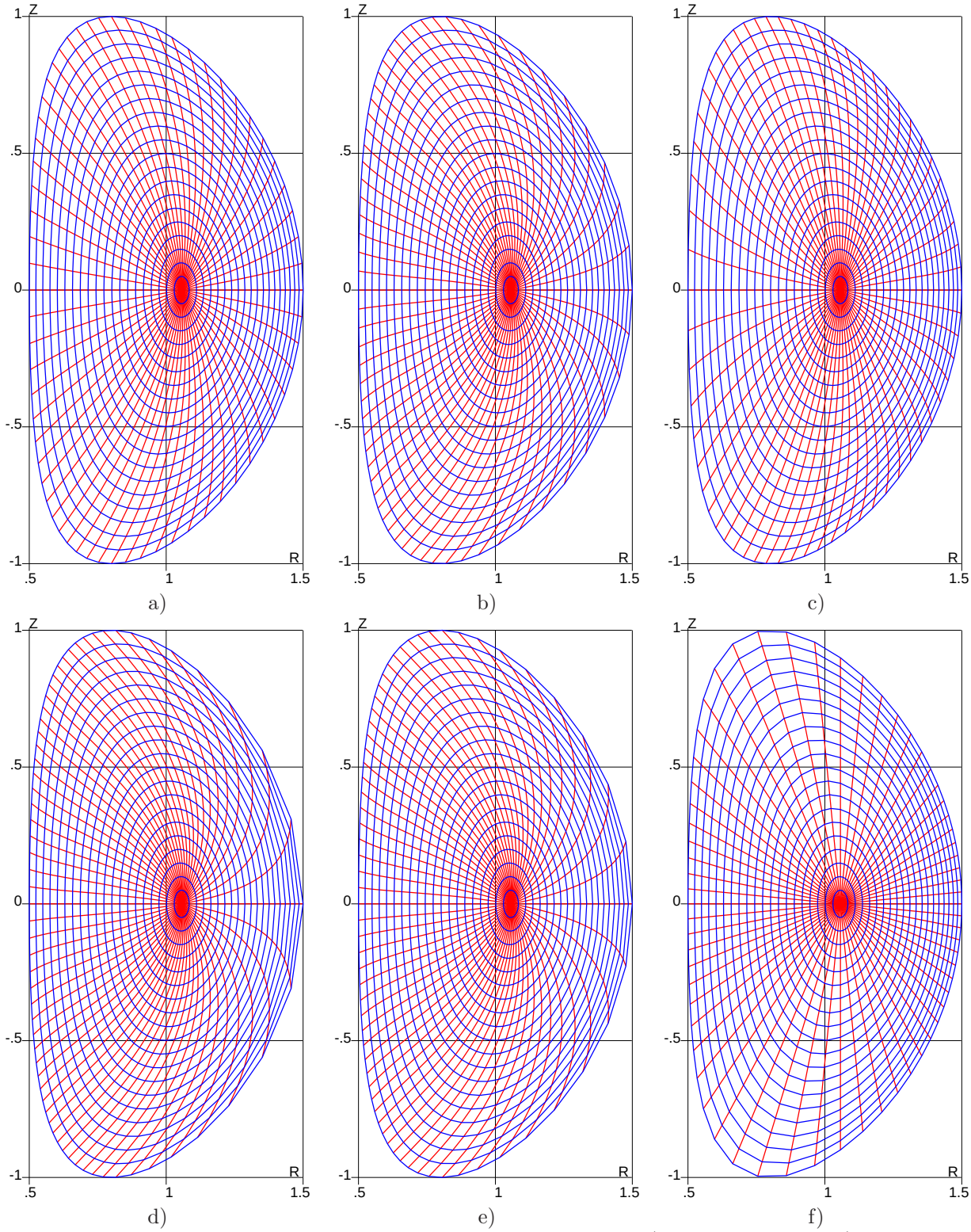

Fig.2. Typical coordinate systems for equilibrium configurations: a) ECS coordinates, b) canonical "hat"-coordinates, c) Hamada coordinates, d) PEST coordinates, e) Boozer coordinates, f) KINX (polar) coordinates.

Before specifying the format of `ESI` data files, the distribution of storage in the case of theuse of shared memory is presented in the next section.

8.1 Storage in the shared memory *(to ToC)*

Table 3 specifies all variables of `ESI` routines. Parameters L_θ, and L_ζ represent periodicity of the system,

$$f(a,\theta,\zeta) = f(a,\theta, L_\theta\theta + L_\zeta\zeta),. \tag{8.1}$$

N_a, N_θ are the number of mesh intervals along a,θ. In the toroidal direction only one period is represented by the number N_ζ. In the case of axisymmetry $N^\zeta = 0$ or $L_\zeta = 0$, while a special value of $L_\theta = -1$ is for up-down symmetry.

ESI data structure				*Table 3*
C name	Type	Math	#	Group name
Np1	int	$N_\theta + 1$	0	<ESI dimensions>
Na1	int	$N_a + 1$		
ID	int	Type of a		
Nt1	int	$N_\zeta + 1$		
Lp	int	L_θ		
Lt	int	L_ζ		
rBtor	double	$\bar{F}_{ref} \equiv \bar{F}(a_{ref})$		
Rext	double	R_{ext}		
aref	double	a_{ref}		
bsp1	double	$\bar{p}_{boundary}$		
ToC[0x40][2]	int		1	*Holds group addresses and sizes*
gq[Na1]	double	θ_j	2	<ESI [gq]>
sa[Na1]	double	a_i	3	<ESI [sa]>
aF [Na1]	double	$F(a_i)$	4	<ESI [baF bsp gF gY]>
bsp[Na1]	double	$\bar{p}(a_i)$		*The group is informational only.*
gF [Na1]	double	$\Phi(a_i)$		*It is generated during*
gY [Na1]	double	$\Psi(a_i)$		*initialization of* ESI
gFa [Na1]	double	$\left(\frac{\bar{\Phi}'}{a}\right)(a_i)$	5	<ESI bgFa bgFaa bgYa bgYaa>
gFaa[Na1]	double	$\left(\frac{\bar{\Phi}'}{a}\right)'_i$		
gYa [Na1]	double	$\left(\frac{\bar{\Psi}'}{a}\right)(a_i)$		
gYaa[Na1]	double	$\left(\frac{\bar{\Psi}'}{a}\right)'_i$		
T [Na1]	double	$T(a_i)$	6	<ESI aT aTa aP aPa>
Ta[Na1]	double	$T'(a_i)$		
P [Na1]	double	$P(a_i)$		
Pa[Na1]	double	$P'(a_i)$		
ESC geometry data			7	Group name
C name	Type	Math	8	<ESI r ra rq raq>
r [Nt1][Na1][Np1]	double	$r(a_i, \theta_j, \zeta_k)$	9	
ra [Nt1][Na1][Np1]	double	$r'_a(a_i, \theta_j, \zeta_k)$		
rq [Nt1][Na1][Np1]	double	$r'_\theta(a_i, \theta_j, \zeta_k)$		
raq[Nt1][Na1][Np1]	double	$r''_{a\theta}(a_i, \theta_j, \zeta_k)$		
same for $z, z'_a, z'_\theta, z''_{a\theta}$			10	<ESI z za zq zaq>
same for $\|B\|, \|B\|'_a, \|B\|'_\theta, \|B\|''_{a\theta}$			11	<ESI B Ba Bq Baq>
ghq [Nt1][Na1][Np1]	double	$\eta'_\theta(a_i, \theta_j, \zeta_k)$	12	<ESI ghq ghaq ghqq gaqq>
ghaq [Nt1][Na1][Np1]	double	$\eta''_{a\theta}(a_i, \theta_j, \zeta_k)$		
ghqq [Nt1][Na1][Np1]	double	$\eta''_{\theta\theta}(a_i, \theta_j, \zeta_k)$		
ghaqq[Nt1][Na1][Np1]	double	$\eta'''_{a\theta\theta}(a_i, \theta_j, \zeta_k)$		
gh [Nt1][Na1][Np1]	double	$\eta(a_i, \theta_j, \zeta_k)$		*generated during initialization*
gha [Nt1][Na1][Np1]	double	$\eta'_a(a_i, \theta_j, \zeta_k)$		*generated during initialization*

Functions $\Phi, \Psi, \bar{F}, \bar{p}, \eta, \eta'_a$ are calculated during the initialization of ESI. Indexes i, j, k are used for the mesh points along a, θ, ζ correspondingly

$$0 \le i \le N_a, \quad 0 \le j \le N_\theta, \quad 0 \le k \le N_\zeta. \tag{8.2}$$

The multidimensional arrays here are equivalent to a 1-D array referenced as [(k*Na1+i)*Np1+j]. In the 3-D case the following storage is added for derivatives with respect to ζ.

ESI profile data			Table 3 (cont)	
C name	Type	Math	#	Group name
gz [Nt1]	double	ζ_k	13	<ESI [gz]>
rz[Nt1][Na1][Np1]	double	$\mathrm{liner}'_\zeta(a_i,\theta_j,\zeta_k)$	14	<ESI rz raz rqz>
raz[Nt1][Na1][Np1]	double	$r''_{a\zeta}(a_i,\theta_j,\zeta_k)$		
rqz[Nt1][Na1][Np1]	double	$r''_{\theta\zeta}(a_i,\theta_j,\zeta_k)$		
raqz[Nt1][Na1][Np1]	double	$r'''_{a\theta\zeta}(a_i,\theta_j,\zeta_k)$		
same for $z'_\zeta, z''_{a\zeta}, z''_{\theta\zeta}$			15	<ESI rz raz rqz>
same for $\lvert B\rvert'_\zeta, \lvert B\rvert''_{a\zeta}, \lvert B\rvert''_{\theta\zeta}$			16	<ESI Bz Baz Bqz>
ghz [Nt1][Na1][Np1]	double	$\eta''_{a\theta}(a_i,\theta_j,\zeta_k)$	17	<ESI ghqz ghaqz ghqqz>
ghaz [Nt1][Na1][Np1]	double	$\eta''_{\theta\theta}(a_i,\theta_j,\zeta_k)$		
ghqz[Nt1][Na1][Np1]	double	$\eta''_{\theta\zeta}(a_i,\theta_j,\zeta_k)$		
ghaqz[Nt1][Na1][Np1]	double	$\eta'''_{a\theta\zeta}(a_i,\theta_j,\zeta_k)$		

The resulting total size of ESI data (without counting generated by ESC) is

`(6+128)*sizeof(int)+3+Np1+9*Na1+Nt1+16*Na1*Np1*Nt1*sizeof(double)`

and, e.g., for `Na1=21,Np1=65,Nt1=33` this is equal to 5,768,352 bytes only.

The parameter ID specifies the type of the coordinate system a, θ, ζ. It is composed from ID_a (Table 4) ID_θ (Table 2), and ID_φ (Table 4) in the following manner

$$\mathrm{ID} = \mathrm{ID}_\zeta * 1000 + \mathrm{ID}_\theta * 100 + \mathrm{ID}_a \tag{8.3}$$

The ID_a is a two digit decimal number 'Dd', where 'd' determine the physics meaning of a (e.g., vertical size, volume, magnetic fluxes, etc), while tens 'D' is a modifier. D=0 and D=2 stands for normalized $0 \le a \le 1$, D=0 and D=1 specify "ρ"-like a (like "minor" radius) when, e.g., $\bar\Phi'(0) = 0$, while D=2 and D=3 are for "V"-like a (similar to volume of magnetic surfaces). Table 4 shows the possible values of ID_a and ID_ζ.

ID_a of radial coordinate a in ESI				Table 4
ID_a	a	ID	a	Comment
0	$a = \frac{b}{b_{boundary}}$	20	$a = \frac{b^2}{b^2_{boundary}}$	b is a vertical semiaxis of the cross-section of the magnetic surfaces
10	$a = b$	30	$a = b^2$	
1	$a = \sqrt{\frac{V}{V_{boundary}}}$	21	$a = \frac{V}{V_{boundary}}$	V is a volume of magnetic surface, a_{ref} [m], V_{ref} [m^3] are some dimensional reference numbers.
11	$a = a_{ref}\sqrt{\frac{V}{V_{ref}}}$	31	$a = V$	
2	$a = \sqrt{\frac{\Phi}{\Phi_{boundary}}}$	22	$a = \frac{\Phi}{\Phi_{boundary}}$	Φ is a volume of magnetic surface, B_{ref} [T] is some dimensional reference numbers (ASTRA code).
12	$a = \sqrt{\frac{\Phi}{\pi B_{ref}}}$	32	$a = \Phi$	
3	$a = \sqrt{\frac{\Psi}{\Psi_{boundary}}}$	23	$a = \frac{\Psi}{\Psi_{boundary}}$	Ψ is a volume of magnetic surface, B_{ref} [T] is some dimensional reference numbers (ASTRA code).
13	$a = \sqrt{\frac{\Psi}{\pi B_{ref}}}$	33	$a = \Psi$	
9	$a, \Phi'(0) = 0$	29	$a, \Phi'(0) \ne 0$	unspecified normalized a
19	$a, \bar\Phi'(0) = 0$	39	$a, \bar\Phi'(0) \ne 0$	unspecified dimensional a

ID_φ of angle ζ				
ID_ζ	ζ	ID_ζ	ζ	Comment
0	$\zeta = \varphi$	1	$\zeta = \bar\zeta$	ζ is a cylindrical azimuth or the straight field line angle
2	$\zeta = \hat\zeta$	9	ζ	ζ is a canonical or unspecified

Use of "V"-like radial coordinates is discouraged because of possible singularities in first derivatives of space functions at the magnetic axis.

In specific cases some groups of `ESI` data could be dropped. Thus, transport codes do not need the group η. For PEST or SFL coordinates this group is automatically absent because $\eta = 0$. *The array* `ToC`*, which keeps addresses and sizes of the groups, marks the absent groups by zero size.*

8.2 Format of ASCII `ESI` data file *(to ToC)*

Every equilibrium code is capable of creating the `ESI` data file. This section specifies the format of the ASCII file, which should be created by individual equilibrium codes.

The ASCII `ESI` file consists of a number of groups of records. Each group starts with an identification line which contain the predefined name of the group, number of records and a string with format of the records. The beginning of an `ESI` ASCII file can serve as a representative example

```
<ESI dimensions>[1](%d x %d %d %e %s) ESC Date: 12/17/06 at 23:05
!Np1 x Na1   ID RBtor Name
 65 x  21   102 1.50  PEST
<ESI [gq]>[1](65%e)
```

Here '`<ESI dimensions>`' is the predefined name of the very first group of `ESI` file, '`[1]`' means 1 record in the group, and the string '`(%d x %d %e)`' specifies the format for reading (3 integers with `'x'` between the first two, one double, and one string). What is after the record head information (the name of the code and the date) is a comment, ignored by the reading routine.

The following line, starting with '`!`' is also a comment. The next line contains the data itself. (The string `"PEST"` is, in fact, only informational, rather than a part of `ESI`).

After this there is the head line of the next group. All names of the groups are given in the Table 3. Groups with names of variables inside '`[]`', like '`<ESI [baF bsp gF gY]>`' are optional. These data are initiated by `ESI`. Also '`<ESI [gq]>`', '`<ESI [sa]>`', '`<ESI [gq]>`' are optional. Their data are replaced by equidistant $0 \le a_i \le 1$, $0 \le \theta_j \le 2\pi$, $0 \le \zeta_k \le 2\pi/L_\zeta$.

The comments can be freely distributed outside the group of data. The short comments can be placed into the data lines: (a) after the format string in the ID line, and (b) after the end of each group of data. With leading '`!`' comment lines can be placed between groups of data.

Note, that only 3 parameters, out of 8 from Table 3, are present in the example. Parameters `Nt1,Lp,Lt,bspb` are dropped from the record. Their default values are `1,0,0,0`. Because the order of parameters is predefined by Table 3 the presence of a format string makes reading unambiguous.

The groups of arrays are recorded in the file by columns with one column for each variable, listed in the name of the group, i.e.,

```
<ESI r ra rgq raq>[1365](%e %e %e %e)
!                   r                   r'_a                   r'_q                  r"_aq
1.1189417832363331e+00  6.1752680497674650e-01 -0.0000000000000000e+00 -9.1619079100425833e-03
1.1189417832363331e+00  6.1365522061023337e-01 -0.0000000000000000e+00 -6.9646002319922323e-02
```

The use of named groups and format lines makes `ESI` *format very flexible.* Thus, reduced versions of `ESI` data set are possible. For example, in communication with the transport simulation codes, the function η is not necessary and its group of data may not be present. The format string identifies which data are present in the records inside the group. *If specified in the format string, enumeration can be in front of the records for human readability. The reading routine would ignore the enumeration.*

8.3 `ESI` binary data- and map- files *(to ToC)*

In this case, the short ASCII file represents the data for reading the binary file.

It starts with the group `<ESI dimensions>`, described in the previous section. It may contain groups with plasma profiles.

After this it lists groups, present in the binary file (the group '<ESI [baF bsp gF gY]>' is useless to keep in the binary files). Instead of real data, the data in the map file represent the address (in bytes) of the beginning of the group and its size (in bytes as well), e.g.

```
  <ESI r ra rq raq}>[2](%d %d)
672 43680
```

9 Basic set of ESI routines *(to ToC)*

The prototypes of ESI C-routines for the user's side (some not yet available) are listed in Table 5. The non-descriptive names of variables are constructed using the following conventions: 'g' prefixes the name of Greek math variables, 'b' prefixes "barred" variables, 'd' prefixes time derivatives, names with 'a','q''z' after the symbol of are used for derivatives. FORTRAN-callable analogs of C-routines have the same names with capital latters replaced by low case ones.

Minimum set of ESI routines. Marked by * are not yet available *Table 5*		
example of C call	*	Comment
`i=File2ESC(FNm);`		`'FNm'` is the name of the `ESI` data-file.
`i=FreeESI();`		frees the memory allocated for `ESI` by `File2ESC()`.
`i=FShM2ESC(key);`		`'key'` is the ID of the shared memory segment with `ESI` data
`i=Link2ESC(F,Fa ,gFa,gFaa ,gYa,gYaa ,T,Ta ,P,Pa ,r,ra,rq,z,za,zq ,B,Ba,Bq ,gh,gha,ghq,isw);`		Called once, it sends to `ESI` the addresses of arrays, which will contain values of the basic functions. `isw` is the integer array which could be necessary for marking the particles.
`i=Link2ESI3d(rz,zz ,Bz,ghz);`	*	Specifies additional links for 3-D case.
`i=ESI2all(a,gq,n);`		The main reconstruction routine, which calculates the basic functions for n particles in positions given by `a[i],q[i]`.
`i=ESI2all3d(a,gq ,gz,n);`	*	3-D analog of `ESI2all()`.
`i=ESI2PlP(F,Fa ,gFa,gFaa ,gYa,gYaa ,T,Ta ,P,Pa,a,n);`		Puts the values of plasma profiles $F, F', \Phi', \Phi'', \Psi', \Psi'', T, T', P, P'$ into arrays `F[], Fa[], gFa[], gFaa[], gYa[], gYaa[], T[], Ta[], P[], Pa[]` for n points specified in `a[]`.
`i=ESI2bsp(bsp,bspa ,a,n);`		Puts the values of $\bar{p}, \bar{p}'$ into arrays `bsp[], bspa[]` for n points specified in `a[]`.
`i=ESI2gFgY(gF,gY ,a,n);`		Puts the values of magnetic fluxes Φ, Ψ (un-barred) into arrays `gF[], gY[]` for n points specified in `a[]`.
`i=ESIrz2agq(a,gq ,r,z,n);`		Converts n cylindrical coordinates r, z into a, θ.
`i=ESIrzf2agq3d(a,gq ,gz ,r,z,gf,n);`	*	3D version of `ESIrz2agq()`.
`i=rzESI2agq(ID ,na,nq);`	*	Converts original `ESI` data on $r-z$ grid into `ESI` data in nested flux coordinates specified by the identification number `ID` with `na,nq` radial and poloidal intervals.
`i=ESI2gcm(dgr,da ,dgq,dgf ,gr,a,gq,gm,n);`	2D	Time advancing routine for guiding center motion. Calculates the time derivatives $\dot{\rho}_{\|}, \dot{a}, \dot{\theta}, \dot{\varphi}$ in Eq.(6.8) for n particles. `gm[]` contains magnetic moments of the particles. Integer marker `isw[]` is used behind the scene. *Only 2-D version is available.*
`i=ESI2mfl(dgqL,dgfL ,a,gq,n);`	2D	Calculates derivatives of θ_l', φ_l' in Eq.(6.1) with respect to length l along n magnetic field lines. *Only 2-D version is available.*

More complete information is given in the `escZ.c.d` document file (which contains as a part also this text). At present `ESI` is functional with Equilibrium and Stability Code (ESC) for axisymmetric tokamak equilibrium configurations. ESC is interfaced with the transport simulation ASTRA code using `ESI` through the shared memory, and using the data files with a number of author's codes.

9.1 Service routines of `ESI`. *(to ToC)*

Some available and possible "convenience" routines are listed in Table 6.

ESI service routines. Marked by * are not yet available		*Table 6*
example of C call	*	Comment
i=ESI2geom(r,ra,rq, z,za,zq, a,gq,n);		Calculates $r, r'_a, r'_\theta, z, z'_a, z'_\theta$ for n points.
i=ESI2rzB(r,z,B ,a,gq,n);		Calculates $r, z, \|B\|$ for n points.
i=rzESI2gcm(dgr,dr ,dz,dgf ,r,z,gm,n);		Analog of ESI2gcm() with use of $r-z$ ESI data. Calculates the time derivatives $\dot{\rho}_\parallel, \dot{r}, \dot{z}, \dot{\varphi}$ for guiding center motion for n particle specified by their r, z and magnetic moment μ.

The set of service routines can be expanded with more practice.

9.2 ESI as a structure. *(to ToC)*

In fact, the Table 3 shows the necessary elements of a single C data-structure which is typedefined as ESIstruct. The user can work simultaneously with several of them (from 0 to 7), e.g., for comparison of different codes. But only one structure is functional at the ESI side. By default the ESI[0] is activated. Using the call

```
i=ESI2up(1);
```

one can switch between active structures.

9.3 ESI as a virtual machine. *(to ToC)*

In a primitive manner, ESI can mimic the "virtual machine" approach of OpenGL. A particular service, like calculation of function $|B|$, gh, or graphics output can be activated any time by using

```
i=esiEnable(esiGL,NULL);
i=esiEnable(esiBASIS,ESIB);
i=esiEnable(esiBASIS,ESIgh);
```

and disabled by

```
i=esiDisable(esiGL ,NULL);
i=esi2Disable(esiBASIS,ESIB);
```

when it is not needed anymore. By default all calculations corresponding to a predefined parameter esiBASIS are activated.

The full set of possibilities is specified ESI documentation file esiZ.c.d.

For use of possibilities of a virtual machine the definition file esiZ.h *(or* esiZ.inc *for FORTRAN) should be included into the user source code. Their content can be found in the* esiZ.c *file.*

10 Summary *(to ToC)*

The described ESI interface has unique properties in being organized, comprehensive, compact and independent of the computer architecture. For many plasma physics codes it can provide a uniform access to the background information related to magnetic configurations, independent of how this information was created and by what equilibrium code.

Without change in the format of communications, ESI allows one to drop some groups of data if they are not necessary for a particular communication. On the other hand, within the same approach it is extendible to more general cases, e.g., anisotropic equilibria or equilibria with perturbed magnetic surfaces. At this moment, one of the crucial extension should be introduction of the separatrix boundary layer, where the flux coordinates have a special behavior near the X-points.