

Representation and modeling of charged particle distributions in tokamaks

Andreas Bierwage^{a,*}, Michael Fitzgerald^b, Philipp Lauber^c, Mirko Salewski^d, Yevgen Kazakov^e, Žiga Štancar^{f,b}

^a*QST, Rokkasho and Naka Fusion Institutes, Japan*

^b*Culham Centre for Fusion Energy of UKAEA, Culham Science Centre, Abingdon, United Kingdom*

^c*Max-Planck-Institut für Plasmaphysik, Garching, Germany*

^d*Department of Physics, Technical University of Denmark, Kgs. Lyngby, Denmark*

^e*Laboratory for Plasma Physics, LPP-ERM/KMS, Partner in the Trilateral Euregio Cluster (TEC), Brussels, Belgium*

^f*Jožef Stefan Institute, Ljubljana, Slovenia*

Abstract

Experimental diagnostics, analysis tools and simulations represent particle distributions in various forms and coordinates. Algorithms to manage these data are needed on platforms like the ITER Integrated Modelling & Analysis Suite (IMAS), performing tasks such as archiving, modeling, conversion and visualization. A method that accomplishes some of the required tasks for distributions of charged particles with arbitrarily large magnetic drifts in axisymmetric tokamak geometry is described here. Given a magnetic configuration, we first construct a database of guiding center orbits, which serves as a basis for representing particle distributions. The orbit database contains the geometric information needed to perform conversions between arbitrary coordinates, modeling tasks, and resonance analyses. Using that database, an imported or newly modeled distribution is mapped to an exact equilibrium, where the dimensionality is reduced to three constants of motion (CoM). The orbit weight is uniquely given when the input is a true distribution: one that measures the true number of physical particles per unit of phase space volume. Less ideal inputs, such as distributions estimated without drifts, or models of particle sources, can also be processed. As an application example, we reconstruct the drift-induced features of a distribution of fusion-born alpha particles in a large tokamak, given only a birth profile, which is not a function of the alpha's CoM. Repeated back-and-forth transformations between CoM space and energy-pitch-cylinder coordinates are performed for verification and as a proof-of-principle for IMAS.

Keywords: Tokamak plasma, Equilibrium, Particle distribution, Guiding centers, Fusion products

Contents

1	Introduction and workflow	1
2	Classes of distributions and transformations	5
3	Geometry, coordinates and normalization	6
4	CoM mesh and drift orbit types	9
5	Equations of motion	11
6	Volume elements and weighting	12
7	Modeling CoM distributions with large drifts	15
8	Application and verification	16
9	Summary and conclusion	23
	Appendices with supplementary information	24

1. Introduction and workflow

1.1. Motivation, purpose and scope

The ITER Integrated Modelling & Analysis Suite (IMAS) [1] offers various ways to store particle distributions for the study of magnetically confined fusion plasmas. Different representations include different choices of coordinates and different discretization methods (mesh grids or marker particles). The Energetic Particle Topical Group of the International Tokamak Physics Activity (ITPA) is currently driving an action to equip IMAS with tools to model and convert distributions of fast particles between different representations that arise in experimental and computational work.

In experimental work, the observed position space of an energetic particle diagnostic is often given by the line-of-sight of the diagnostic in cylinder coordinates (R, z, ζ) , where R is the major radius, z the height and ζ the toroidal angle. In tokamaks, ζ is often an ignorable symmetry coordinate. For many diagnostics, the observed velocity-space is given by weight functions that

*Email address: bierwage.andreas@qst.go.jp

are derived from energy and momentum conservation [2, 3, 4, 5]. The diagnostic weights are often expressed as functions of kinetic energy $E = Mv^2/2$ and velocity pitch $\lambda = v_{\parallel}/v$, where v is the magnitude of the velocity vector \mathbf{v} , v_{\parallel} its component parallel to the magnetic field vector \mathbf{B} , and M the particle mass. Diagnosticians hence have a good understanding of the observation regions of diagnostics in phase space (E, λ, R, z) . Synthetic diagnostics calculating expected measurements based on numerical simulations are usually based on such representations of local velocity distributions.

If one assumes that the plasma is not only toroidally symmetric but also quiescent, the distribution functions of confined charged particles can be taken to depend only on three constants of motion (CoM). Analyses of resonant instabilities are usually performed with respect to such an equilibrated reference state, where the orbit time τ , gyrophase ξ and toroidal angle ζ are ignorable symmetry coordinates. The respective conserved actions — namely, the kinetic energy E , magnetic moment μ , and canonical toroidal angular momentum P_{ζ} — combined with an index σ specifying the sign of v_{\parallel} on passing orbits, constitute a natural set of CoM coordinates $(E, \mu, P_{\zeta}; \sigma)$ that is common in theoretical analyses, although other useful (and equivalent) sets of CoM exist. Recently, an experimental measurement of a fast ion distribution function in three-dimensional (3-D) CoM space was demonstrated for fast-ion D- α spectroscopy [6]. Efforts are underway to apply this orbit tomography method [7] to other fast-ion diagnostics [8].

In order to make predictions of a measurement based on a stability calculation or carry out stability analyses based on measured data, we must be able to transform distribution functions between 3-D CoM space and 4-D representations given in various coordinates as illustrated in Fig. 1. Related problems have been tackled to various degrees by different research groups with different codes and methods. For instance, TRANSP/NUBEAM [9] and LIGKA/HAGIS [10] employ binning and smoothing procedures to map distributions of marker particles onto a mesh and compute their gradients in CoM space. While TRANSP/NUBEAM uses Monte-Carlo sampling, LIGKA/HAGIS uses a database of guiding center (GC) orbits for all particle species. Related activities were also reported by the ASCOT [11] and MEGA [12] groups. Although these methods have been used for many years, detailed documentations in the literature are rare. The above references [9, 10, 11, 12] all point to presentations given at technical meetings. Useful elements can also be found in many papers, but it is necessary to assemble that information into a complete goal-oriented workflow.

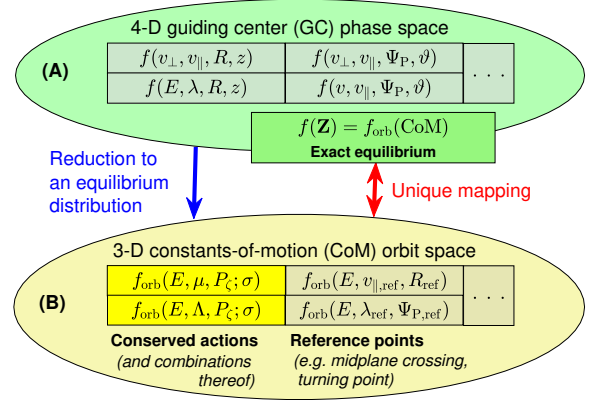


Figure 1: Examples of coordinate representations and transformations (arrows) of a toroidally symmetric distribution function. We require that every 4-D distribution $f(\mathbf{Z})$ be reduced to an exact equilibrium $f_{\text{orb}}(\text{CoM})$, so that the mapping between 4-D GC phase space and 3-D CoM orbit space becomes unique and readily reversible. — Position coordinates: major radius R ; height z ; poloidal magnetic flux $\Psi_P(R, z)$; poloidal angle ϑ . — Velocity coordinates: GC velocity components v_{\perp} and v_{\parallel} relative to the magnetic field \mathbf{B} ; kinetic energy $E = Mv^2/2$ with $v^2 = v_{\perp}^2 + v_{\parallel}^2$; pitch $\lambda = v_{\parallel}/v$; sign $\sigma = v_{\parallel}J_{\parallel}/|v_{\parallel}J_{\parallel}|$ relative to the parallel current density J_{\parallel} . — Mixed coordinates: magnetic moment $\mu = Mv_{\perp}^2/(2B)$; conserved pitch $\Lambda = \mu B_0/E$; canonical toroidal angular momentum P_{ζ} ; coordinates $\Psi_{P, \text{ref}}, \lambda_{\text{ref}}$, etc., evaluated at a certain reference point, such as a midplane crossing or a turning point.

Thus motivated, this paper documents the workflow used in our computer code VisualStart [12, 13]. Since 2009, we have used this code to initialize hybrid simulations with modeled or numerically computed beam ion distributions [14, 15, 16, 17] and to analyze their resonant interactions with Alfvén waves [18, 19, 20]. VisualStart represents GC distributions using marker particles, which can be directly used in full- f simulations¹ and yield a quiet start [13]. The representation in CoM space with an underlying GC orbit database will also allow us to directly embed the distribution functions in general phase-space transport theories [21, 22, 23, 24], both formally and technically.

Here we propose the orbit-based representation and modeling technique as a solution for ITER IMAS and other platforms that need to store and process distributions of charged particles in magnetically confined plasmas. The subject of smoothing is discussed briefly in the following section, but the documentation and verification of smoothing algorithms is left for future work, as is the associated problem of initializing delta- f simulations that require the evaluation of gradients. We note that a similar and in many ways complementary effort has been undertaken by S. Benjamin *et al.* [25].

¹For instance, see Fig. 3 in Ref. [14], and Fig. 4 in Ref. [15] for fast ion tails, and Fig. 4 in Ref. [16] for a birth distribution.

1.2. Basic rules for practical situations

The theoretical foundations for transforming distribution functions are relatively straightforward, but it is also clear that compromises and creativity are needed when tackling real-world data with numerical techniques, where accuracy has to be traded for computational performance and where data may have varying degrees of quality and completeness. For instance, the singularities and topological boundaries of CoM space [26] require attention when cutting a mesh to define phase space volume elements. Experimentally measured distributions and numerically computed distributions are often incomplete and noisy, so some amount of modeling is needed, including smoothing, interpolation and extrapolation.

In view of these challenges, some basic rules should be followed in order to maintain the physical and numerical integrity of the data, independently of the particular workflow used to process distribution functions:

- (i) Smoothing, interpolation and extrapolation are best applied at the preparation stage, where a new distribution function is first constructed or imported. Such potentially irreversible manipulations should be minimized in all subsequent operations.
- (ii) Pay attention to the fact that, in toroidal geometry, resolution and noise levels inevitably vary in space and in time, because the mirror force and magnetic drifts cause particles to be distributed nonuniformly along their orbits. In particular:
 - Inherent topological discontinuities and singularities, as found in CoM space, should be preserved; e.g., through mesh accumulation and by smoothing only around, not across trapped-passing boundaries and loss cones.
 - The optimal representation method (maximizing accuracy) may differ between archiving a distribution function and using it in a simulation. While archiving merely requires adequate resolution, simulations should also minimize signal-noise correlations [13].

These issues are most pronounced in but not limited to cases with large magnetic drifts.

- (iii) Ensure that the distribution to be transformed is an exact equilibrium in the given magnetic configuration; i.e., its shape must not vary in time when evolved using unperturbed equations of motion.

The equilibrium constraint (iii) is crucial, so it deserves further explanation: Any distribution function f can be mapped to CoM space, where the result f_{orb}

measures the time-averaged particle densities on unperturbed GC orbits. This operation is indicated by a blue arrow in Fig. 1. However, when the original f is not an equilibrium, this mapping operation is time-dependent and it is impractical to store the information needed to invert it. The equilibrium constraint reduces the effective dimensionality to three CoM by determining the longitudinal distribution of particles along their GC orbit, which is now time-independent and with it all coordinate transformations as well. The coordinate transformations $f \leftrightarrow f_{\text{orb}}$ can then be readily performed in any direction, as indicated by a red double arrow in Fig. 1.

Nonequilibrium distributions are very likely to be encountered in IMAS, where one can expect to receive particle data from a measurement for which the magnetohydrodynamic (MHD) equilibrium is not exactly known or from a simulation where waves or magnetic islands were present. Enforcing the equilibrium constraint means that the input data are transformed into a distribution that is consistent with a modeled magnetic configuration, which generally differs somewhat from the conditions where the input data originated from.

1.3. Orbit-based representation and modeling

The above considerations suggest that it is useful and advantageous to represent distribution functions in CoM space with an underlying database of unperturbed GC orbits. With that geometric information, it is straightforward to reconstruct a 4-D distribution in arbitrary coordinates by loading a suitable number of markers uniformly in time along each GC orbit [13]. Projections to various coordinates and views of user-defined subspaces with user-defined resolution can then be readily obtained by binning the markers on demand. This minimizes the amount of transformations and associated data corruption, as demanded in items (i) and (ii) above. Marker particles are a natural choice for representing fast particle distributions since [the latter](#) are often computed using particle codes. Moreover, by organizing the markers on orbit contours, one can identify and deal with inherent singularities and discontinuities as required in item (ii). Last but not least, by representing a distribution function in CoM orbit space, the equilibrium constraint (iii) is automatically enforced.

We note that the implementation and enforcement of the equilibrium constraint only requires a conversion to CoM coordinates: $f \rightarrow f_{\text{orb}}$. The Jacobians required for such transformations can be determined using the Monte-Carlo method as implemented in TRANSP/NUBEAM by Breslau & Liu [9]. In this way, it is possible to avoid integration along GC orbits.

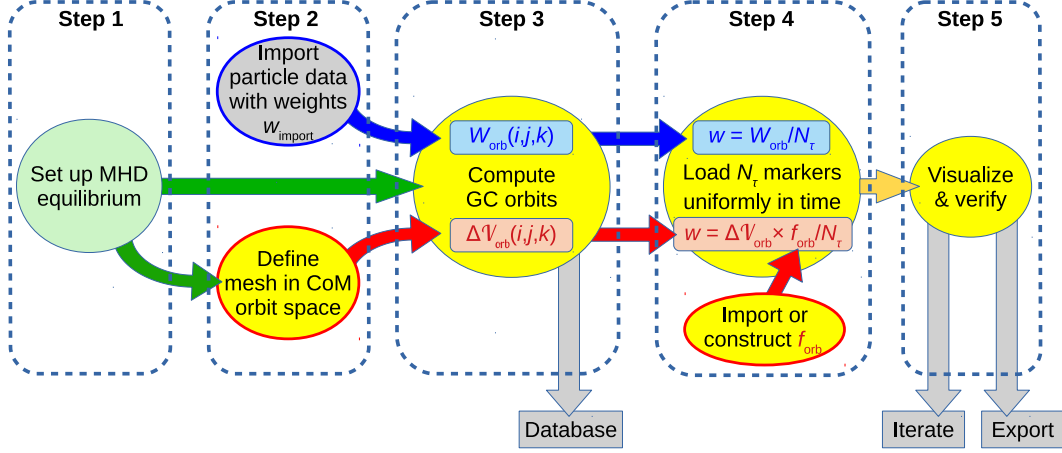


Figure 2: Relevant part of the VisualStart workflow in five steps as described in the text. This paper focuses on the tasks highlighted yellow.

However, the computational cost associated with building an orbit database is more than repaid by the information one gains about the shape of the orbits, the longitudinal particle distribution, and the transit times. The transit times are needed for resonance analyses and their variation near trapped-passing boundaries can be used for mesh accumulation. The information about the orbit geometry is also useful for reduced models, for instance to capture *finite-orbit-width* effects [27], and it allows to define and make use of a variety of specialized GC coordinates, like the radii R_m where an orbit crosses the midplane. Last but not least, the information contained in the orbit database is also valuable for carrying out modeling tasks, such as the following example:

Given a model function G_{mdl} that may not be an exact solution of a kinetic equation for physical particles or GCs, we wish to construct an equilibrium distribution f_{orb} . A common example are models of the form $G_{\text{mdl}}(E, \lambda, \Psi_p)$ that contain measured radial profiles as functions of flux Ψ_p and a velocity distribution computed by a bounce-averaged Fokker-Planck code. In such a case, the orbit database allows to incorporate the effect of magnetic drifts *a posteriori* [28]. There are, of course, various possibilities to perform this operation; in other words, there is some freedom to interpret the values of G_{mdl} as weights in an orbit distribution f_{orb} .² An intuitive and physically meaningful choice is to perform an orbit-time average $\langle \dots \rangle_{\text{orb}}$ followed by a suitable normalization: $f_{\text{orb}} = \text{const.} \times \langle G_{\text{mdl}} \rangle_{\text{orb}}$.

²Functions such as $G_{\text{mdl}}(\mathbf{Z})$ described in this example will here be called “quasi-distributions”, because they do not measure the true number of objects of interest (here physical particles or GCs) in a volume element $d\mathbf{Z}$ but require more elaborate transformations than a mere conversion of coordinates. See Section 2 for details.

1.4. Workflow

The workflow described in this paper is shown schematically in Fig. 2 and consists of five steps:

- **Step 1:** Define an axisymmetric magnetic field \mathbf{B} ; e.g., using a dedicated MHD equilibrium code.
- **Step 2:** Define samples in 3-D CoM orbit space, with cell indices (i, j, k) . The blue and red workflow branches in Fig. 2 begin with different inputs:
 - (blue) Import marker particles from another code as in earlier works [15, 16, 17]. The weights of the imported samples are reinterpreted as orbit weights: $w_{\text{import}} \rightarrow W_{\text{orb}}(i, j, k)$.
 - (red) Use a meshing algorithm to divide the CoM space into cells of size $\Delta V_{\text{orb}}(i, j, k)$. This path is described in detail in this paper.
- **Step 3:** Integrate along GC orbits starting from the samples defined in Step 2. Store in a database.
- **Step 4:** Represent an equilibrium distribution function f_{orb} using weighted markers loaded on the orbits in the database. When loading N_τ markers uniformly in time, their weight is N_τ ’s fraction of the orbit weight: $w = W_{\text{orb}}/N_\tau$. In the case of the red workflow in Fig. 2, $W_{\text{orb}} = \Delta V_{\text{orb}} \times f_{\text{orb}}$ is the product of the volume element ΔV_{orb} and the desired phase space density function f_{orb} . The latter may be imported or constructed from a model.
- **Step 5:** Visualize and verify. Iterate if necessary; e.g., changing the resolution in CoM space (Step 2), the number of markers on the orbits, or the weight assignment (Step 4). Export when ready.

Note that MHD wave spectra can be obtained from the data available in Step 1, and resonance maps can be computed from the orbit database built in Step 3. Procedures for smoothing the distribution, computing phase space gradients and analyzing the stability of resonant modes remain to be added in or around Step 5.

1.5. Outline

Clear definitions of symbols and terminology are a prerequisite for successful conversion operations. Thus, we begin our treatise in Section 2 with a discussion of different classes of distributions functions and how they are transformed. **Step 1** of our workflow is part of Section 3, where we describe the tokamak plasma that serves as a working example, define coordinates and normalizations. **Step 2**, the construction of a phase space mesh, is covered in Section 4, where we also discuss orbit classes and the phase space topology in our scenario. The GC equations of motion we solve to construct the orbit database in **Step 3** are given in Section 5, and the formula for the orbit volume elements is derived in Section 6. **Steps 4 & 5** are covered in Sections 7 and 8, where we construct a distribution function f from a model, visualize and analyze the result, and verify the integrity and accuracy of our scheme by transforming f back and forth between two different representations. Modeling is a rich subject on its own, and our example is only meant to highlight the utility of an orbit database and to serve as a proof-of-principle for IMAS applications. The Appendices contain supplementary information on binning procedures and numerical accuracy.

2. Classes of distributions and transformations

There are not only different coordinates and different numerical representations of distributions; the latter also come in different types whose identity must be clearly defined since it determines how they are transformed. Here, we distinguish between three types of distributions: (i) density functions, (ii) histograms, and (iii) quasi-distributions.

The first two can be characterized as follows:

- (i) The symbols $f(\mathbf{Z})$ represent *phase space densities*, which have units of $[\text{m}^{-6}\text{s}^{-3}]$ and whose values are *independent* of the coordinates \mathbf{Z} appearing in the argument, so $f(\mathbf{Z}(\mathbf{v}, \mathbf{x})) = f(\mathbf{v}, \mathbf{x})$. The argument merely specifies the representation space.

The same is true for velocity and volume integrals:

$$n(\mathbf{x}) = \int d^3v f : \text{density field}, \quad (1a)$$

$$\nu(\mathbf{v}) = \int_V d^3x f : \begin{array}{l} \text{velocity distribution} \\ \text{in a volume } V. \end{array} \quad (1b)$$

- (ii) The symbols $h(\mathbf{Z})$ represent *histograms*, which have the units of the inverse volume element $|d\mathbf{Z}|^{-1}$ of the coordinates used for “binning” (counting the objects of interest). Histograms are convenient to construct and integrate since no Jacobian is required (except for coordinate conversions).

The phase space density function f is related to histogram functions h, h' , etc., as

$$f(\mathbf{Z}) = \mathcal{J}_Z^{-1} h(\mathbf{Z}) = \mathcal{J}_{Z'}^{-1} h'(\mathbf{Z}') = \dots \quad (2)$$

where \mathcal{J}_Z is the Jacobian for the transformation from Cartesian coordinates $(\mathbf{v}, \mathbf{x}) = (v_x, v_y, v_z, x, y, z)$ to another arbitrary set of phase space coordinates \mathbf{Z} .

A phase space density function f is an exact solution of a kinetic equation for the objects of interest; in our case, physical particles and guiding centers (GC). In this work, we assume that all GC coordinates are defined such that a single distribution function f represents both the phase densities of physical particles and GCs; i.e., f is a solution of both kinetic equations:

$$0 = (\partial_t + \dot{\mathbf{Z}} \cdot \partial_{\mathbf{Z}}) f \quad (\text{physical particles}) \quad (3)$$

$$\begin{aligned} &= \partial_t f + \underbrace{(\dot{\mathcal{J}}_{\text{gc}} \mathbf{Z}_{\text{gc}} + \mathcal{J}_{\text{gc}} \dot{\mathbf{Z}}_{\text{gc}})}_0 \cdot \frac{\partial_{\mathbf{Z}_{\text{gc}}} f}{\mathcal{J}_{\text{gc}}} \\ &= (\partial_t + \dot{\mathbf{Z}}_{\text{gc}} \cdot \partial_{\mathbf{Z}_{\text{gc}}}) f \quad (\text{guiding centers}). \end{aligned} \quad (4)$$

This is the case when the Jacobian \mathcal{J}_{gc} for the transformation $\mathbf{Z} \rightarrow \mathbf{Z}_{\text{gc}}$ from physical particle to GC coordinates satisfies the GC phase space conservation law

$$\dot{\mathcal{J}}_{\text{gc}} \equiv \partial_t \mathcal{J}_{\text{gc}} + \partial_{\mathbf{Z}_{\text{gc}}} \cdot (\mathcal{J}_{\text{gc}} \dot{\mathbf{Z}}_{\text{gc}}) = 0. \quad (5)$$

Equation (5) is trivially satisfied for canonical coordinates, since their volume elements differ only by a constant factor, so $\mathcal{J}_{\text{gc}} = \text{const.}$ All our CoM coordinates — namely, $(E, \mu, P_\zeta; \sigma)$ and any equivalent set of GC orbit coordinates, such as $(v, v_{\parallel, \text{ref}}, R_{\text{ref}})$ — are, by definition, conserved by the GC equations of motion that we solve. This means that, even when the equations of motion are not in Hamiltonian form, our orbit coordinates combined with the triplet of ignorable angle variables (τ, ξ, ζ) — orbit time, gyrophase, toroidal angle — can be considered to be canonical coordinates, whose Jacobians are constants satisfying Eq. (5) down to the numerical accuracy of our particle-pushing algorithm.

A well-known set of noncanonical GC coordinates is $(\mu, v_{\parallel}, \mathbf{x}_{\text{gc}})$, whose volume elements transform as

$$d^3v d^3\mathbf{x} = \frac{2\pi B_{\parallel}^*}{M} d\mu dv_{\parallel} d^3\mathbf{x}_{\text{gc}}. \quad (6)$$

The Jacobian factor B_{\parallel}^* satisfies Eq. (5) as shown by Littlejohn [29]. Equation (6) is very useful, because it can be easily converted to other velocity coordinates that are defined locally in position space — such as the pitches $\lambda = v_{\parallel}/v$ and $\Lambda = \mu B_0/E$ — since their Jacobian factors can be derived analytically (Appendix A) [30].

Both (i) densities and (ii) histograms are *true* distributions in the sense that, when integrated over a phase space volume element $\Delta\mathcal{V}$,

$$\int_{\Delta\mathcal{V}} d\mathbf{Z} h(\mathbf{Z}) = \int_{\Delta\mathcal{V}} d^3v d^3\mathbf{x} f = \mathcal{N}, \quad (7)$$

they yield the true number \mathcal{N} of objects contained inside that volume element. Functions or data that do not satisfy this condition *for the specified kind of objects* constitute the third type in our classification scheme:

- (iii) The symbol $G(\mathbf{Z})$ represents *quasi-distributions*, which do not yield the true number of objects by mere volume integration in the configuration at hand, but require more complex transformations.

Quasi-distributions often arise in experimental or modeling work. A typical example in experimental measurements is the line-of-sight density, where the transformation $G \rightarrow f$ takes the form of an inverse problem. In fact, since the magnetic configuration in an experiment is rarely known accurately, any experimentally measured particle statistics should be treated as quasi-distributions when used as input for modeling distributions in a numerically constructed MHD equilibrium. In general, the process of “modeling” essentially consists of choosing a physically meaningful transformation $G \rightarrow f$. We will return to this topic in Sections 7 and 8, where we apply our methods to model a distribution of fusion-born alpha particles, based on a quasi-distribution $G(\Psi_p)$ that represents their birth profile.

Finally, we emphasize that when a GC orbit distribution function is given in the form $f_{\text{orb}}(E, \mu, P_{\zeta}; \sigma)$, or an equivalent set of CoM coordinates, the Jacobian \mathcal{J}_{CoM} must be accompanied by instructions concerning how to deal with the index σ that specifies the sign of the parallel velocity. First, it must be clarified whether σ is the sign relative to the magnetic field or the plasma current if their directions differ. Second, it must be clarified how particles trapped in a magnetic mirror are to be counted, because σ is redundant for such trapped particle orbits.

Depending on the choice made, the integral in Eq. (7) can be written in at least three ways:

$$\mathcal{N} = \sum_{\sigma=\pm 1} \int dE d\mu dP_{\zeta} \underbrace{\mathcal{J}_{\text{CoM}}^{t/p}(\text{trap./pass.})}_{h_{\text{orb}}^{t/p}(E, \mu, P_{\zeta}; \sigma)} f_{\text{orb}}, \quad (8a)$$

$$= \sum_{\sigma_{\text{pass}}=0, \pm 1} \int dE d\mu dP_{\zeta} \underbrace{\mathcal{J}_{\text{CoM}}}_{h_{\text{orb}}(E, \mu, P_{\zeta}; \sigma_{\text{pass}})} f_{\text{orb}}, \quad (8b)$$

$$= \sum_{\sigma_{\text{HFS}}=\pm 1} \sum_{\sigma_{\text{LFS}}=\pm 1} \int dE d\mu dP_{\zeta} \underbrace{\frac{\mathcal{J}_{\text{CoM}}}{2}}_{h_{\text{orb}}(E, \mu, P_{\zeta}; \sigma)} f_{\text{orb}}. \quad (8c)$$

In the first case (8a), the summation over σ counts each trapped orbit twice, which means that the histogram $h_{\text{orb}}^{t/p}$ has two sets of identical entries each representing *half* of the particles in the domain of trapped orbits. In the second case (8b), the summation index $\sigma_{\text{pass}} = \{0, \pm 1\}$ is three-valued, identifying trapped (0), co- and counter-passing orbits (± 1), so that h_{orb} represents the *full* number of particles in all regions of GC phase space without duplicate entries. In mathematical form:

$$h_{\text{orb}}(\sigma_{\text{pass}} = 0) = 2 \times h_{\text{orb}}^{t/p}(\text{trap.}), \quad (9)$$

$$\mathcal{J}_{\text{CoM}} = \mathcal{J}_{\text{CoM}}^{t/p}(\text{pass.}) = 2 \times \mathcal{J}_{\text{CoM}}^{t/p}(\text{trap.}). \quad (10)$$

The values of the density function f_{orb} itself are, of course, independent of the convention used. The point we want to make is that it is crucial to ensure consistency between the summation over σ and the Jacobian when counting particles by integrating f_{orb} in $(E, \mu, P_{\zeta}; \sigma)$ space as in Eq. (8).

In this paper, we adopt the third option (8c), with the single-valued constant Jacobian $\mathcal{J}_{\text{CoM}}/2$. Our CoM phase space mesh defined in Section 4 will double-count all orbits — passing and trapped — when they cross the plasma midplane: once on the high-field side (HFS) and once on the low-field side (LFS). The factor 2 dividing the Jacobian in Eq. (8c) compensates this.

3. Geometry, coordinates and normalization

System geometry. The magnetic field vector is written

$$\mathbf{B} = \nabla \times \mathbf{A} = \nabla \zeta \times \nabla \Psi_p + B_{\zeta} \nabla \zeta \quad (11)$$

$$= \nabla \zeta \times \nabla \Psi_p + \nabla \Psi \times \nabla \vartheta, \quad (12)$$

where $2\pi\Psi = \frac{1}{2\pi} \int d\Psi d\vartheta d\zeta \mathcal{J}_f^x \mathbf{B} \cdot \nabla \zeta$ is the toroidal flux and $2\pi\Psi_p = -2\pi A_{\zeta} = \frac{1}{2\pi} \int d\Psi d\vartheta d\zeta \mathcal{J}_f^x \mathbf{B} \cdot \nabla \vartheta$ the poloidal flux. These fluxes are related via the tokamak safety factor $q = d\Psi/d\Psi_p$ measuring the mean magnetic

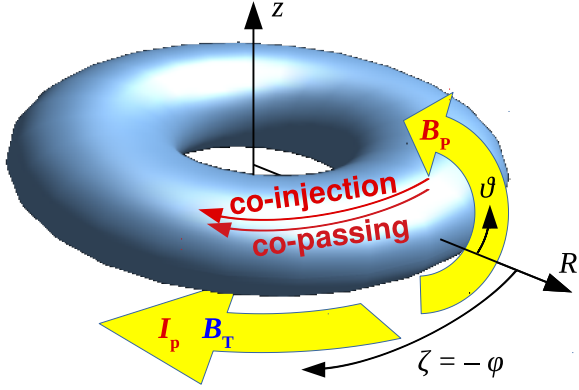


Figure 3: Toroidal geometry and coordinates. Right-handed cylinder coordinates (R, z, ζ) and the poloidal angle θ are indicated. The yellow arrows show the orientation of the toroidal magnetic field \mathbf{B}_T , the plasma current I_p and the associated poloidal magnetic field \mathbf{B}_p in our working example, which corresponds to the situation in typical JET and JT-60U tokamak plasmas. The co-/counter injection of beams is defined relative to the plasma current, as is the direction of co-/counter passing particles. Definitions in the literature vary, but in the present case there is no risk of confusion since I_p and \mathbf{B}_T are aligned.

field line pitch. $\mathcal{J}_f^x = [\nabla\Psi \cdot (\nabla\theta \times \nabla\zeta)]^{-1} = 1/B_\zeta$ is the Jacobian for the transformation from Cartesian position coordinates \mathbf{x} to the toroidal flux coordinates (Ψ, θ, ζ) , where θ is a poloidal angle and $\zeta = -\varphi$ is the geometric toroidal angle, whose orientations are indicated in Fig. 3. Also shown in Fig. 3 are the axes of the right-handed cylinder coordinate system (R, z, ζ) , with major radius $R = \sqrt{x^2 + y^2}$ and vertical coordinate z .

The magnetic flux surface geometry and profiles for the MHD equilibrium that we use as a working example in this paper is shown in Fig. 4. This plasma is partly based on recent experiments with central radio-frequency (RF) heating using a 3-ion scheme [31, 32] performed at the Joint European Torus (JET). The magnetic X-point has been removed and the profile of the safety factor q is chosen to increase monotonically from values near unity in the core ($q_0 = 0.98$ on axis) to $q_a = 5.44$ at the boundary. This simulation scenario is also used in ongoing studies of MHD instabilities and fast ion physics [33]. The q profile in the above-mentioned experiments varied dynamically and is likely to differ from ours at most times.

Besides R and z , we will also use the auxiliary coordinates $X = R - R_0$ and $Y = z - z_0$ measuring horizontal and vertical positions relative to the magnetic axis, which is located here at $(R_0, z_0) = (2.98 \text{ m}, 0.27 \text{ m})$.

Magnetic flux labels. The normalized toroidal and poloidal fluxes (ψ, ψ_p) or their square roots (ρ_T, ρ_p) can

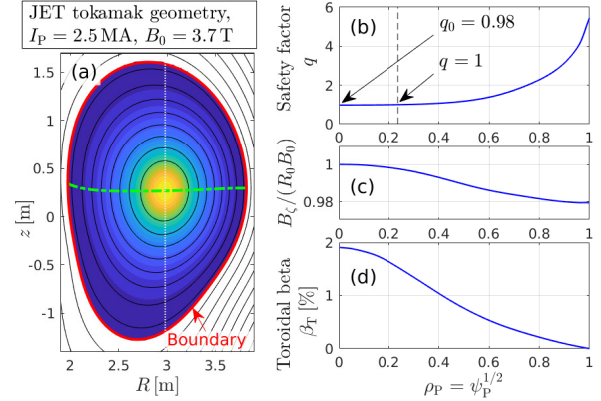


Figure 4: Our working example takes the basic plasma geometry from JET [31, 32], with plasma current $I_p = 2.5 \text{ MA}$, on-axis magnetic field $B_0 = 3.7 \text{ T}$, and magnetic axis location $(R_0, z_0) = (2.98 \text{ m}, 0.27 \text{ m})$. Panel (a) shows contours of the poloidal flux $\Psi_p(R, Z)$ (black) and scalar pressure $P(R, Z)$ (color), as well as the midplane (green, Eq. (18)). The X-point has been removed and the plasma boundary (red) is located slightly inside of would have been the last closed flux surface. Panels (b)–(d) show the radial profiles of the safety factor $q = d\Psi/d\Psi_p$, the covariant toroidal field component B_ζ and the toroidal beta $\beta_T = 2\mu_0 P/B_0^2$ (with $\mu_0 = 4\pi \times 10^{-7} \text{ H/m}$) as functions of ρ_p defined in Eq. (13).

serve as convenient minor radial coordinates:

$$\rho_T^2 = \psi = \frac{\int_0^{V(\Psi)} dV/\mathcal{J}_f^x}{\int_0^{V(\Psi_a)} dV/\mathcal{J}_f^x}, \quad \rho_p^2 = \psi_p = \frac{\Psi_p - \Psi_{p,0}}{\Psi_{p,a} - \Psi_{p,0}}, \quad (13)$$

where $V(\Psi)$ is the volume within the flux surface labeled Ψ . $\Psi_{p,0}$ and $\Psi_{p,a}$ are the values of the poloidal flux at the magnetic axis ($r = 0$) and boundary ($r = a$). For the toroidal flux we have $\Psi_a - \Psi_0 = \int_{\Psi_{p,0}}^{\Psi_{p,a}} d\Psi_p q$. In some occasions, we use a volume-averaged minor radius $0 \leq r(\Psi) \leq a$, which is defined here as

$$r^2 = \frac{R_0}{\pi} \int_0^{V(r)} \frac{dV}{\mathcal{J}_f^x B_\zeta} = \frac{R_0}{\pi} \int_0^{V(r)} \frac{d\Psi}{B_\zeta} \approx a^2 \psi. \quad (14)$$

Our r reduces the geometric minor radius in the limit of a cylindrical plasma with circular cross-section. For $B_\zeta \approx R_0 B_0$, we have $r/a \approx \rho_T$, and $a \approx [2(\Psi_a - \Psi_0)/B_0]^{1/2}$ is the minor radius of the plasma boundary.

Particle and orbit coordinates. In the guiding center (GC) description, the velocity space consists of the ignorable gyrophase ξ , the perpendicular particle velocity v_\perp , and the parallel GC velocity $v_\parallel = \mathbf{v}_{gc} \cdot \hat{\mathbf{b}}$ relative to the magnetic field vector \mathbf{B} with unit vector $\hat{\mathbf{b}} \equiv \mathbf{B}/B$ and magnitude $B = |\mathbf{B}|$. The canonical momentum can be written in the form $\mathbf{P} = \mathbf{A}/B_0 + \Omega_0^{-1} \mathbf{v}$ for a particle with

gyrofrequency $\Omega_0 = QeB_0/M$, charge Qe and mass M . In GC coordinates, its covariant toroidal component,

$$P_\zeta = \mathbf{P}_{\text{gc}} \cdot \partial_\zeta \mathbf{x} = -\frac{\Psi_p}{B_0} + \frac{v_\parallel}{\Omega_0} \frac{B_\zeta}{B}, \quad (15)$$

is called the canonical toroidal angular momentum and is chosen here to have units of area per radian, $(2\pi)^{-1}[\text{m}^2]$. Together with the lowest-order magnetic moment μ and the signed kinetic energy $\mathcal{E} = \sigma_{\text{ref}} E$,

$$\mu = \frac{Mv_\perp^2}{2B}, \quad \mathcal{E} = \sigma_{\text{ref}} \frac{M}{2} v^2, \quad \sigma = \frac{v_\parallel J_\parallel}{|v_\parallel J_\parallel|}, \quad (16)$$

this yields a set of coordinates $(\mathcal{E}, \mu, P_\zeta)$ that belongs to the class of constants of motion (CoM), which are conserved by our equations of motion (presented in Section 5) when they are solved in a time-independent field \mathbf{B} that is symmetric along ζ (or “axisymmetric” around \hat{e}_z), satisfying $\partial_\zeta B_\zeta = \partial_\zeta \Psi_p = 0$.

The sign σ of the parallel velocity is measured relative to the parallel current density $J_\parallel = \mathbf{J} \cdot \mathbf{B}/B$. While redundant or unnecessary for orbits trapped in a magnetic mirror (both signs are present), σ identifies whether a passing orbit with coordinates (E, μ, P_ζ) is co- or counter-going. In mathematical treatments, σ usually appears as a separate index, $(E, \mu, P_\zeta; \sigma)$, but this is inconvenient for numerical representations. In our orbit database, we define σ_{ref} at a certain reference point; namely, the starting point used for the orbit calculation. This or any equivalent convention allows us to attach σ_{ref} to E as in Eq. (16) and obtain a compact representation where all CoM arrays in the computer code have the same dimensionality: three.

It is also useful to define several different coordinates measuring the velocity pitch:

$$\Lambda \equiv \frac{\mu B_0}{E} = \frac{\cos^2 \alpha}{\hat{B}}, \quad \sin \alpha = \lambda \equiv \frac{v_\parallel}{v} = \sigma_B \sqrt{1 - \Lambda \hat{B}}, \quad (17)$$

with $\hat{B} = B/B_0 \approx R_0/R$ and $\sigma_B \equiv v_\parallel/|v_\parallel|$. Each pitch coordinate has its advantages in practical applications:

- The coordinate $\lambda = v_\parallel/v$ is widely used together with E or velocity $v = \sqrt{2E/M}$. The local velocity space Jacobian of this set is a CoM ($J_{E\lambda} \propto \sqrt{E}$, Eq. (A.12)), and the Fokker-Planck equation can be integrated analytically in these coordinates [34].
- The coordinate α is a true angle, so its value measures the velocity pitch in a very intuitive form. Moreover, the domains of co- and counter-passing particles are enlarged along the α -axis, making this coordinate a good choice for sampling the velocity

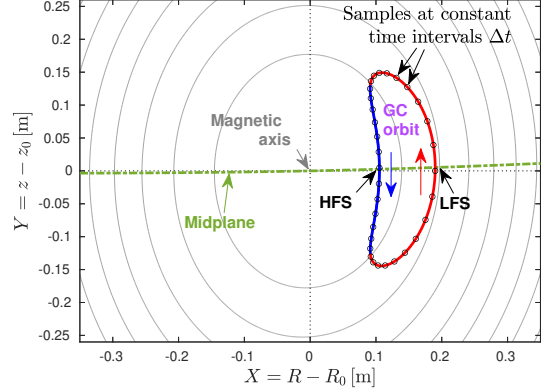


Figure 5: GC orbit of a core-localized trapped alpha particle with kinetic energy $E = 1.5$ MeV and pitch angle $\alpha = 0.056\pi$. This figure shows the inner region of the poloidal plasma cross-section of Fig. 4(a). The grey contours are uniformly spaced in poloidal magnetic flux Ψ_p . The orbit is sampled uniformly in time (black circles). The GC is co-moving ($v_\parallel \geq 0$) in the red and counter-moving ($v_\parallel < 0$) in the blue portion of the orbit contour. The high- and low-field-side intersections of the orbit with the midplane are labeled HFS, LFS.

distribution of tangentially injected beam ions [17] and distributions produced by central RF heating using a 3-ion scheme [31, 32, 35].

- The coordinate $\Lambda = \hat{B}^{-1} \cos^2 \alpha$ has the advantage of being a CoM and that (in contrast to μ) its upper limit is independent of energy: $0 \leq \Lambda \leq \hat{B}_{\text{min}}^{-1}$, where $B_{\text{min}} = \min\{\hat{B}\}$ is the minimal field strength in the considered domain. This makes Λ a convenient choice for defining the lines along which to divide the phase space as will be done in Section 4.

Unlike the conserved quantities E , μ and Λ , the pitch coordinates α and λ vary along a GC orbit. The latter can be turned into orbit coordinates (i.e., CoM), α_{ref} and λ_{ref} , by evaluating them at a well-defined reference point, as we did earlier for the sign σ_{ref} .

Reference points for orbit coordinates. In cases where we import computed particle distributions (blue workflow in Fig. 2), the initial reference point is taken to be the position of an imported marker particle, which may be located anywhere inside the plasma or in the surrounding vacuum.

In cases where we need to construct a new CoM mesh (red workflow in Fig. 2) the reference points lie in the midplane, which (at least in a usual tokamak plasma) contains all O-type stagnation points and is defined as the curve $z_m(X)$ along which the magnetic field is perpendicular to its gradient,

$$z_m(R) : \mathbf{B} \cdot \nabla B = 0, \quad (18)$$

The midplane is shown as a dash-dotted green line in Fig. 4. It contains the magnetic axis, $(R, z) = (R_0, z_0)$, but deviates from the horizontal plane $z = z_0$ in up-down asymmetric plasmas like ours. Every GC orbit crosses the midplane twice, once in a region with higher field strength and once in a region with lower field strength. Figure 5 shows an example, where the orbit's high- and low-field side crossings of the midplane are labeled “HFS” and “LFS”, respectively.

Normalization. Spatial positions and lengths are given in meters unless stated otherwise. Particle velocities are normalized by a reference value v_0 . Energy is normalized by Mv_0^2 (for each particle species) and the magnetic field by its on-axis value B_0 (here 3.7 Tesla):

$$\hat{v} = \frac{v}{v_0}, \quad \hat{E} = \frac{E}{Mv_0^2} = \frac{\hat{v}^2}{2}, \quad \hat{\mu} = \frac{\mu B_0}{Mv_0^2} = \frac{\hat{v}_\perp^2}{2\hat{B}}. \quad (19)$$

4. CoM mesh and drift orbit types

In this section, we describe how we sample the GC orbit space. The example in Fig. 6 is used for illustration. A low resolution is chosen in order to make all grid points clearly visible. All coordinates shown in Fig. 6 are evaluated at the height $z_m(X)$ of the midplane as defined in Eq. (18). This means that all coordinates appearing in this section are constants of motion (CoM). The set (\hat{E}, α, X) used in panels (a) and (b) yields particularly compact and, in our opinion, intuitive plots. Another view of pitch-position space in (Λ, X) coordinates is shown in panel (c). The plasma boundary is also taken to be the loss boundary for the GC orbits, so we do not include the vacuum here and our midplane mesh covers only the width of the plasma itself: $-1 \text{ m} \lesssim X \lesssim 0.84 \text{ m}$.

The reference velocity used for normalization is chosen to be $v_0 = 1.3 \times 10^7 \text{ m/s}$, which corresponds to $E_0 = Mv_0^2/2 \approx 3.5 \text{ MeV}$ for fusion-born alpha particles (${}^4_2\text{He}^{+2}$) or 1.75 MeV for deuterons (${}^2_1\text{H}^+$). Both species have the same charge-to-mass ratio Qe/M and their characteristic gyroradius in this example is $\rho_0 = v_0/\Omega_0 \approx 0.07 \text{ m} \approx 0.02 \times R_0$. We consider the full energy range $0 \leq \hat{E} \leq \hat{E}_0$ with $\hat{E}_0 = 0.5$ and define a mesh consisting of $N_E^{(g)}$ grid points that are the boundaries of $N_E = N_E^{(g)} - 1$ cells. The value of energy at the center of cell i is denoted \hat{E}_i . The example in Fig. 6(a) has $N_E = 15$ cells with equal sizes $\Delta\hat{E}_i = \hat{E}_0/N_E = \text{const.}$

The remaining phase space is divided along the coordinate lines of the conserved pitch Λ . These lines appear parabolic in the (α, X) -plane of panel (b) and horizontal

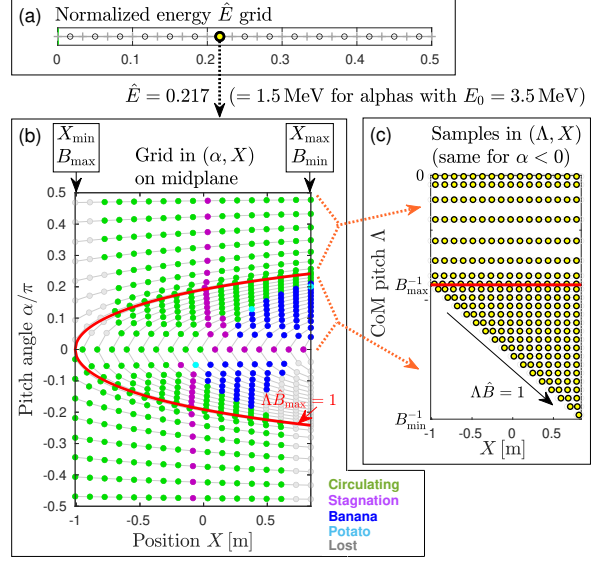


Figure 6: Method for sampling the GC orbit space in VisualStart. Panel (a) shows the grid in normalized kinetic energy $\hat{E} = \hat{v}^2/2$, here chosen to be uniformly spaced in \hat{v}^2 . Panel (b) shows the grid in pitch angle-position space (α, X) at the midplane of Fig. 4(a). The color of each grid point in (b) identifies the type of an orbit for a given energy, here $\hat{E} = 0.217$. These grid points lie along lines of $\Lambda = \text{const.}$, which appear parabolic in the (α, X) -plane of panel (b) and straight in panel (c). The black circles in (c) indicate the locations of the samples taken halfway between grid points. The bold red line indicates the contour where $\Lambda \hat{B}_{\max} = 1$. Each orbit is effectively sampled twice: trapped orbits appear above and below the $\alpha = 0$ line, and circulating orbits on the left and right of the stagnation points. Thus, except very close to a stagnation point, passing and trapped orbits can be treated equally, because both are effectively double-counted in this mesh.

in the (Λ, X) -plane of panel (c). Two different procedures are used in the domains above and below the red line in Fig. 6, which represents $\Lambda \hat{B}_{\max} = 1$:

- $0 < \Lambda < \hat{B}_{\max}^{-1}$: At $X = X_{\min}$, namely the left vertical axis of panel (b), we define a grid in the pitch angle coordinate $-\pi/2 \leq \alpha \leq \pi/2$. In the present example, it consists of $N_\alpha = 16$ equally sized cells with $\Delta\alpha_j = \pi/N_\alpha = \text{const.}$ The circles at $X = X_{\min}$ in panel (b) indicate the cell centers α_j with $j = 1 \dots N_\alpha$. These points (α_j, X_{\min}) are the starting points for lines $\Lambda = \cos^2 \alpha / \hat{B} = \text{const.}$ plotted gray in Fig. 6(b). Along each of these lines, we define a grid in the radial coordinate X . In the present example, it consists of $N_X = 16$ cells with $\Delta X_k = (X_{\max} - X_{\min})/N_X = \text{const.}$ The upper part of panel (c) shows the resulting samples (Λ_j, X_k) at the cell centers in the $\alpha > 0$ portion of panel (b).
- $B_{\max}^{-1} < \Lambda < \hat{B}_{\min}^{-1}$: At $\alpha = 0$, namely the horizontal axis at the center of panel (b), we define

a grid in the radial coordinate X . Our example has $N_X = 16$ equally sized cells. The circles at $\alpha = 0$ indicate the cell centers X_j with $j = 1 \dots N_X$, which form a diagonal ($\Lambda \hat{B} = 1$) in the lower part of panel (c). They are the origins of grid lines $\Lambda_j^{(g)} = \text{const.}$, which are plotted gray in panel (b). Here, we choose to have an approximately uniform sample density. For this purpose, we define an auxiliary coordinate $0 \leq d_j \leq L_j$ that measures the distance along a grid line $\Lambda_j^{(g)}$ in the $(\hat{\alpha}, \hat{X})$ -plane, with $\hat{\alpha} = \alpha/(2\pi)$ and $\hat{X} = X/(X_{\max} - X_{\min})$. Along each grid line, we create N_j cells with roughly equal sizes $\Delta d_{j,k} = L_j/N_j \approx \text{const.}$ The number of cells satisfies $2 \leq N_j \leq 2N_X$ and the cell index $1 \leq k \leq N_j$ covers positive and negative pitches, so there is at least 1 cell and at most N_X cells in each domain, $\alpha \gtrless 0$, respectively. The lower part of panel (c) shows the resulting samples (Λ_j, X_k) at the cell centers in the $\alpha > 0$ portion of panel (b).

The above procedure samples the phase space of confined GC orbits twice, once on the LFS and once on the HFS. This ensures full coverage of the phase space and eliminates the need to distinguish between orbit types inside the mesh-cutting algorithm. The double-counting will be corrected by a factor 1/2 in the volume elements derived in Section 6 below.

The color of each sample in Fig. 6(b) identifies the type of an orbit for a given energy. Here, we have chosen $\hat{E}_{i=7} = 0.217$. In the case of alpha particles with $E_0 = Mv_0^2/2 = 3.5$ MeV, this corresponds to an energy of 1.5 MeV. Orbits are divided into two groups: orbits that are trapped in a magnetic mirror, and passing orbits (co- and counter-current). Trapped orbits contain a point where v_{\parallel} changes sign, while passing orbits do not. Each group is further divided into two sub-categories as defined in Table 1. This orbit classification scheme is relatively simple; see Ref. [26] for more elaborate discussions. Note that for every tuple (E, Λ) , the midplane contains a pair of O-type stagnation points, where co- or counter-passing stagnation orbits are point-like in the poloidal plane. Since these O-type stagnation points alone are insignificant, we have chosen to expand the class of “stagnation orbits” to include all passing orbits that do not encircle the magnetic axis.

The classes of potato and stagnation orbits arise from magnetic drifts. The magnitude of the magnetic drift velocity v_{dB} (see Eq. (23) below) is proportional to the inverse aspect ratio a/R_0 and the kinetic energy. Furthermore, its (R, z) -component is inversely proportional to the poloidal field $B_p \propto I_p$, which tends to divert it into the toroidal direction. This means that magnetic drifts

	Encircle magnetic axis at $(X, Y) = (0, 0)$	
	Yes	No
Trapped:	Potato orbits	Banana orbits
Passing:	Circulating orbits	Stagnation orbits

Table 1: Orbit classification.

tend to be large for energetic particles in tokamak plasmas with relatively low plasma current and in compact torii. In such cases, the overall fraction of potato and stagnation orbits can be significant.

The mesh in Fig. 6(b) is rather sparse, and our automatic classification algorithm identified only 3 samples as potato orbits (see Fig. 8(a) for a denser mesh and higher energy). The $\alpha > 0$ legs of potato orbits are found around the upper rim of the trapped-passing boundary on the $X > 0$ side of Fig. 6(b). On the $X < 0$ side, they are located slightly below the $\alpha = 0$ line.

It is important to note that the trapped-passing (t-p) boundary on the $X < 0$ side also lies *below* the $\alpha = 0$ line. This is due to the magnetic drifts and has the implication that the turning points — where the magnetic mirror force causes a sign reversal in v_{\parallel} — are near but not identical to the V-type stagnation points of trapped particle orbits.³ The example in Fig. 7(a) illustrates this for the case of an orbit very close to a t-p boundary.

The t-p boundary refers to a separatrix in the orbit topology; i.e., it marks the topological transition of two V-type stagnation points via an X-type stagnation point as illustrated schematically in the box at the top left of Fig. 7. The marginally trapped orbit in panel (a) has two V-type stagnation points that nearly form an X. Across the t-p boundary, this orbit decomposes into a counter-passing orbit and a trapped orbit, each with a single V-type stagnation point that turns into a smooth part of the orbit contour as one departs from the t-p boundary.

We use the word “stagnation” to indicate that GCs spend a relatively large amount of time in a relatively small region of the poloidal plane. Around O-type stagnation points, this is the case for the entire orbit. In the case of X- and V-type stagnation points as in Fig. 7, GCs spend a large amount of time in a small portion of the orbit contour. Spatial stagnation also means that $v_{\parallel}/v = \sigma_B \sqrt{1 - \Lambda \hat{B}}$ stagnates as shown in panel (b).

The pitch angle scans of the poloidal and toroidal transit times in panels (c)–(f) of Fig. 7 show true singularities in τ_{pol} and τ_{tor} , and a topological discontinuity (see also Fig. 3 of Ref. [37]). The role of the transit time singularities will be discussed later in Section 6.

³The discussion below Eq. (19) in Ref. [36] is related to this.

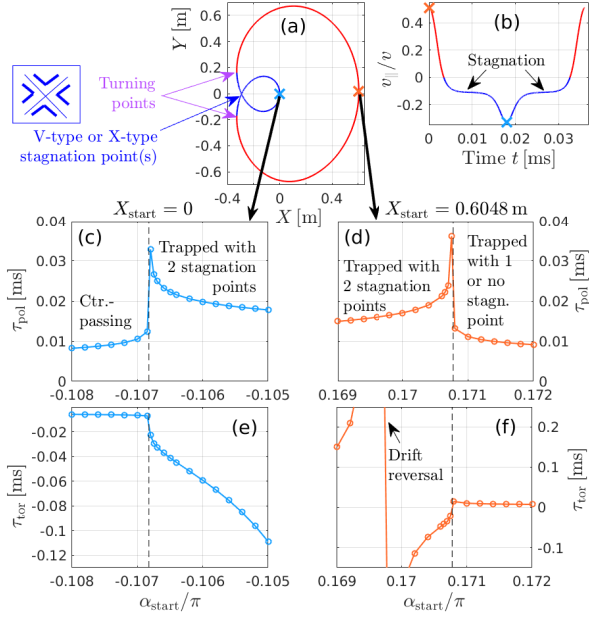


Figure 7: Properties of GC orbits near a trapped-passing (t-p) boundary. Panel (a) shows the poloidal orbit contour of a trapped alpha particle with $E = 3.5$ MeV located very close to the t-p boundary, which is a topological discontinuity occurring at the X-point of the orbit's counter-going leg ($v_{||} < 0$, blue). The box on the top left illustrates schematically the topological transition between two pairs of V-type stagnation points via an X-point at the t-p boundary. The turning points, where $v_{||}$ changes sign, are also indicated in (a). The time trace of the pitch $v_{||}/v = \sin \alpha$ is shown in panel (b) with time t in milliseconds. Panels (c)–(f) show the pitch angle dependence of the poloidal transit time τ_{pol} (c,d) and the toroidal transit time τ_{tor} (e,f) across the t-p boundary (vertical dashed line). The starting point ($E, \alpha_{\text{start}}, X_{\text{start}}$) lies on the midplane ($Y \approx 0$) as indicated by the crosses in panel (a): $X_{\text{start}} = 0$ is on the inner leg with $\alpha < 0$ for (c,e), and $X_{\text{start}} \approx 0.6$ m is on the outer leg with $\alpha > 0$ for (d,f). The poloidal transit time τ_{pol} exhibits singular behavior at the t-p boundary. Meanwhile, the apparent discontinuity merely reflects the partition of a phase space volume element between two orbits upon crossing the t-p boundary.

The discontinuity is a matter of perspective, as it merely reflects how a larger volume element of GC phase space is divided into two components represented by a pair of orbits that were unified on the other side of the t-p boundary. In some sense, the transition is smooth if one views the disconnected orbits as a pair.

Finally, we note that $v_{||} = 0$ constitutes a degeneracy in CoM space, since Λ and $P_{\zeta}(X)$ cannot be varied independently for fixed E at that location. This is why, geometrically, the parabolic $\Lambda = \text{const.}$ curves have their pole at that location in Fig. 6. Since we are defining our CoM mesh in the midplane, we are able to avoid these degenerate $v_{||} = 0$ points entirely by locating them along the X_k cell boundary on $(E, \Lambda) = \text{const.}$ lines. All cell centers are, thus, ensured to have $v_{||} \neq 0$.

5. Equations of motion

As indicated by the red and blue arrows between Steps 2 & 3 in our workflow Fig. 2, the GC phase space samples that are provided by

- (a) the mesh constructed in Section 4, or
- (b) particle codes like OFMC [38, 39] and MEGA [40, 41] (when operated in full- f mode [14, 42])

serve as initial conditions for the GC equations of motion, which we solve once around each orbit.

VisualStart solves the same equations with the same numerical scheme as its current target code MEGA. Following the formalism introduced by Littlejohn [29] and reviewed by Cary & Brizard [43], we use the effective electromagnetic fields $\mathbf{B}^* \equiv \nabla \times \mathbf{A}^*$ and $\mathbf{E}^* \equiv -\nabla \Phi^* - \partial_t \mathbf{A}^*$, with the potentials $\mathbf{A}^* \equiv \mathbf{A} + \frac{v_{||}}{\Omega} \mathbf{B}$ and $Qe\Phi^* \equiv Qe\Phi + \mu B$. Here, we consider the motion of GCs in an unperturbed magnetic field and in the absence of any electric field ($\partial_t \mathbf{B} = \Phi = 0$), so the effective fields reduce to

$$\mathbf{B}^* = \mathbf{B} + \frac{v_{||}}{\Omega} \mathbf{B} \nabla \times \hat{\mathbf{b}}, \quad \mathbf{E}^* = -\frac{\mu \nabla B}{Qe}, \quad (20)$$

and the equations for the GC velocity $\mathbf{v}_{\text{gc}} = \dot{\mathbf{x}}_{\text{gc}}$ and its parallel acceleration become

$$\dot{\mathbf{x}}_{\text{gc}} = v_{||} \frac{\mathbf{B}^*}{B_{||}^*} + \frac{\mathbf{E}^* \times \hat{\mathbf{b}}}{B_{||}^*} = \mathbf{v}_{||}^* + \mathbf{v}_{\text{dB}}^*, \quad (21)$$

$$\dot{v}_{||} = \frac{\Omega}{B} \frac{\mathbf{B}^* \cdot \mathbf{E}^*}{B_{||}^*}. \quad (22)$$

$\hat{\mathbf{b}} = \mathbf{B}/B$ is the unit vector along \mathbf{B} , and $B_{||}^* \equiv \mathbf{B}^* \cdot \hat{\mathbf{b}}$ is Littlejohn's GC Jacobian for the noncanonical set of GC coordinates $(\mu, u, \xi, \mathbf{x}_{\text{gc}})$ with $u \equiv \hat{\mathbf{b}}(\mathbf{x}_{\text{gc}}) \cdot \dot{\mathbf{x}}_{\text{gc}} \approx v_{||}$ (see also the discussion on p. 704 of Ref. [43]). The GC velocity is composed of a parallel component (modified by curvature effects) and magnetic drifts due to ∇B and curvature $\nabla \times \hat{\mathbf{b}}$:

$$\mathbf{v}_{||}^* = v_{||} \frac{\mathbf{B}}{B_{||}^*}, \quad \mathbf{v}_{\text{dB}}^* = \frac{\mu}{QeB_{||}^*} \hat{\mathbf{b}} \times \nabla B + \frac{v_{||}^2}{\Omega} \frac{B}{B_{||}^*} \nabla \times \hat{\mathbf{b}}. \quad (23)$$

In the normalization of VisualStart (Section 4):

$$\hat{\mathbf{v}}_{||}^* = \hat{v}_{||} \frac{\hat{\mathbf{B}}}{\hat{B}_{||}^*}, \quad \hat{\mathbf{v}}_{\text{dB}}^* = \rho_0 \frac{\hat{\mu}}{\hat{B}_{||}^*} \hat{\mathbf{b}} \times \nabla \hat{B} + \rho_0 \frac{\hat{v}_{||}^2}{\hat{B}_{||}^*} \nabla \times \hat{\mathbf{b}}, \quad (24)$$

with $\rho_0 = v_0/\Omega_0$. The computed GC orbits are recorded in a database, following accuracy checks (Appendix B).

The example in Fig. 6, gives a total of about 8000 orbit samples, 52% of which lie inside the region bounded

by the red curve ($\Delta\hat{B}_{\max} \leq 1$) for the present setup. About 1100 orbits (14%) are lost as they hit the (artificial) boundary of the simulation domain due to magnetic drifts. Using 3 processes on an 8th Generation Intel CORE i7-8565U processor (1.80 GHz \times 8), it took about 35 minutes to compute these 8000 orbits with our simple Matlab code in which there is much room left for optimization. A proper orbit database for the entire plasma volume may require at least 100 times more samples, which would take 150 core-hours with the present implementation. An optimized code in a compiled language such as Fortran or C is expected to need only a small fraction of this time. Making use of GPUs, the time required to compute such an orbit database should be reasonably short for routine work.

6. Volume elements and weighting

6.1. Derivation

Having computed the GC orbits using the unperturbed equations of motion in Section 5 starting from the midplane mesh defined in Section 4, the next step is to represent the distribution function f in discretized form using an ensemble of marker particles labeled $n = 1 \dots N_{\text{mrk}}$, which sample the orbits in our database. In other words, we seek the Klimontovich representation⁴

$$f(\mathbf{Z}) \approx \sum_{n=1}^{N_{\text{mrk}}} w_n \times \delta(\mathbf{Z}_{\text{gc},n} - \mathbf{Z}). \quad (25)$$

At this point, $\mathbf{Z}_{\text{gc},n}$ is the marker position in an arbitrary set of GC coordinates including ignorable angles, for instance $(v_{\perp}^2, v_{\parallel}, \xi, \mathbf{x}_{\text{gc}})$. The Dirac δ distribution has units of inverse phase space volume [$\text{m}^{-6}\text{s}^{-3}$] and may be viewed as a proxy for any sort of particle shape factor needed to map the weights w_n onto a discrete mesh. Although each marker follows the trajectory of a GC as indicated by the subscript in $\mathbf{Z}_{\text{gc},n}$, its weight factor w_n represents a certain number $[\Delta\mathcal{N}_{\text{phys}}]_n$ of physical particles (not GCs!):

$$\begin{aligned} w_n &= [\Delta\mathcal{N}_{\text{phys}}]_n = \Delta\mathcal{V}_n \times \frac{\mathcal{N}_{\text{phys}}(\Delta\mathcal{V}_n)}{\Delta\mathcal{V}_n}, \\ &= \Delta\mathcal{V}_n \times f. \end{aligned} \quad (26)$$

where $\Delta\mathcal{V}_n$ denotes a Cartesian volume element in physical coordinates that is attached to marker n .

Since we are considering only exact equilibrium distributions, the orbit time τ is an ignorable angle coordinate with period τ_{pol} and all factors in Eq. (26) are

independent of the three angle coordinates (τ, ξ, ζ) ; they depend only on three CoM coordinates, whose values we will represent by cell indices (i, j, k) . When we load N_{τ} markers uniformly in time along an unperturbed GC orbit, neighboring markers are indistinguishable. Each marker carries precisely an N_{τ} 's fraction of their orbit's volume $\Delta\mathcal{V}_{\text{orb}}$ and weight W :

$$\Delta\mathcal{V}_n \rightarrow \Delta\mathcal{V}_{ijkl} = \left[\frac{\Delta\mathcal{V}_{\text{orb}}}{N_{\tau}} \right]_{ijk} = \left[\frac{\Delta\mathcal{V}_{\text{orb}} \Delta\tau_l}{\tau_{\text{pol}}} \right]_{ijk}, \quad (27)$$

$$w_n \rightarrow w_{ijkl} = \left[\frac{W}{N_{\tau}} \right]_{ijk} \quad \text{for } l = 1, \dots, N_{\tau,ijk}. \quad (28)$$

As indicated in Eq. (27), the markers also represent equal fractions $\Delta\tau$ of an orbit's poloidal period τ_{pol} :

$$\Delta\tau_{ijkl} = \left[\frac{\tau_{\text{pol}}}{N_{\tau}} \right]_{ijk}. \quad (29)$$

Recalling from Eqs. (26) that the volume element is eventually applied to a distribution function f , it is clear that Eq. (29) actually implies an orbit time average:

$$\sum_{l=1}^{N_{\tau,ijk}} \Delta\tau_{ijkl} f(\mathbf{Z}_{\text{gc},ijkl}) \approx \oint_{\tau_{\text{pol}}} d\tau f \equiv \tau_{\text{pol}} \underbrace{\langle f \rangle_{\text{orb}}}_{f_{\text{orb}}}. \quad (30)$$

The average is trivial for true equilibrium distributions, which are independent of τ , so that $f = f_{\text{orb}}$. However, f in Eq. (30) may also be replaced by an arbitrary quasi-distribution G , so this equation also constitutes a recipe for transforming arbitrary models into true equilibria. In summary, Eq. (26) becomes

$$w_{ijkl} = \left[\frac{W}{N_{\tau}} \right]_{ijk} = \left[\frac{\Delta\mathcal{V}_{\text{orb}}}{N_{\tau}} \times f_{\text{orb}} \right]_{ijk}. \quad (31)$$

In applications where we merely import particle data computed by another code (blue workflow in Fig. 2), the weights w_l are provided with the data. Each sample is interpreted as the initial position of a new GC orbit, so the input weight is interpreted as the weight of an orbit: $[w_{\text{import}}]_l \rightarrow W_{ijk}$. This step enforces the equilibrium constraint and the first equality of Eq. (31) yields the marker weights. The volume element $\Delta\mathcal{V}$ and distribution function f do not appear explicitly in such a case.

In applications where we construct a new distribution function from a model or where we reexpress a mesh-based distribution function in the orbit-based representation (red workflow in Fig. 2), it is necessary to determine the GC orbit volume element $[\Delta\mathcal{V}_{\text{orb}}]_{ijk}$ for a given mesh in CoM space. In our case, the mesh has

⁴Appendix A describes in detail how we evaluate Eq. (25).

the form shown in Fig. 6 and the associated volume elements $[\Delta\mathcal{V}_{\text{orb}}]_{ijk}$ in units of $[\text{m}^6\text{s}^3]$ can be readily obtained when written in terms of the canonical action coordinates (E, μ, P_ζ) associated with the ignorable angles (τ, ξ, ζ) , taking advantage of the fact that the Jacobian for a canonical transformation is constant in both space and time (see, for instance, Ref. [36]).

With our definition $P_\zeta = -\Psi_P/B_0 + v_\parallel B_\zeta/(\Omega_0 B)$, which has units of $(2\pi)^{-1}[\text{m}^2]$, the transformation between Cartesian coordinates (\mathbf{v}, \mathbf{x}) for the phase space of physical particles and canonical coordinates $(E, \tau, \mu, \xi, P_\zeta, \zeta)$ for the phase space of GCs has the form

$$d^3\mathbf{v}d^3\mathbf{x} = \frac{1}{2} \times \left(\left[\frac{dE}{M} \frac{d\mu B_0}{M} dP_\zeta d\tau d\xi d\zeta \right]_{\sigma_{\text{HFS}}} + \left[\frac{dE}{M} \frac{d\mu B_0}{M} dP_\zeta d\tau d\xi d\zeta \right]_{\sigma_{\text{LFS}}} \right). \quad (32)$$

The index $\sigma = v_\parallel J_\parallel / |v_\parallel J_\parallel|$ determines in which domain (co- or counter-going) a volume element lies, irrespective of the orbit type (trapped or passing). Using the fact that canonical volume elements are identical anywhere on a GC orbit (E, μ, P_ζ) , Eq. (32) adds the volume elements at the high- and low-field side (HFS, LFS) midplane crossings and divides the result by 2, in accordance with our double-counting convention in Eq. (8c). Integrating over the angles ξ and ζ , using the pitch coordinate $\Lambda = \mu B_0/E$ instead of the magnetic moment, and applying our normalizations $\hat{v} = v/v_0$ and $\hat{E} = E/(Mv_0^2)$, we obtain

$$(2\pi)^2 \hat{E} d\hat{E} d\Lambda \times \left(\left[dP_\zeta d\tau \right]_{\sigma_{\text{HFS}}} + \left[dP_\zeta d\tau \right]_{\sigma_{\text{LFS}}} \right) \times \frac{v_0}{2} \\ \approx (2\pi)^2 \hat{E}_i \Delta \hat{E}_i \Delta \Lambda_j \Delta P_{\zeta,k} v_0 \Delta \tau_{ijkl} \times \frac{1}{2}, \quad (33)$$

for our CoM mesh with indices (i, j, k) as defined in Section 4, where the cell index k covers positive and negative signs of σ for both the HFS and LFS midplane crossings. It remains to specify the increment ΔP_ζ .

Since our CoM mesh in Fig. 6 is defined in the midplane, the canonical toroidal momentum P_ζ for fixed E and Λ is a function of major radius only, which is here expressed in terms of $X = R - R_0$:

$$[P_\zeta]_{E\Lambda}(X) = -\frac{\Psi_P(X)}{B_0} + \rho_0 \frac{\hat{v}_\parallel(X)}{B(X)} B_\zeta(X). \quad (34)$$

Its increment can then be evaluated as

$$\Delta P_{\zeta,ijk} \approx P_\zeta \left(X_{k+1}^{(g)} \middle| \hat{E}_i, \Lambda_j \right) - P_\zeta \left(X_k^{(g)} \middle| \hat{E}_i, \Lambda_j \right), \quad (35)$$

where $X_k^{(g)}$ and $X_{k+1}^{(g)}$ are the grid points adjacent to cell

X_k . Alternatively, using $v_\parallel = \sigma_B \sqrt{2E(1 - \Lambda \hat{B})}$, the increment ΔP_ζ can be expressed in terms of ΔX as

$$\Delta P_\zeta = \underbrace{\left[-\frac{\Psi'_P}{B_0} - \frac{\rho_0}{\hat{v}_\parallel} \left(\hat{E} + \frac{\hat{v}_\parallel^2}{2} \right) \frac{B_\zeta B'}{B^2} + \rho_0 \frac{\hat{v}_\parallel}{B} B'_\zeta \right]}_{|P'_\zeta|} \Delta X, \quad (36)$$

where radial derivatives like $P'_\zeta \equiv [\partial_X P_\zeta]_{E\Lambda}$ are taken along the midplane with fixed E and Λ , and they are evaluated at cell centers X_k .

In summary, substituting Eq. (29) into (33) and multiplying by the number N_τ of markers used to represent an orbit, we obtain the GC phase space volume element represented by the orbit sample in cell (E_i, Λ_j, X_k) in a form that can be readily evaluated numerically:⁵

$$[\Delta\mathcal{V}_{\text{orb}}]_{ijk} = 4\pi^2 v_0 \hat{E}_i \Delta \hat{E}_i \Delta \Lambda_j \underbrace{[\tau_{\text{pol}} \Delta P_\zeta]_{ijk}}_{\text{contains zeros \& singularities}} \times \frac{1}{2}. \quad (37)$$

The final factor 1/2 originates from Eq. (8c) via Eq. (32) and compensates our double-counting of all orbits in the domain $X_{\min} \leq X \leq X_{\max}$ in Fig. 6(b) via index k .

6.2. Discussion

The formula for the volume element in Eq. (37) may look harmless, but the factor $\tau_{\text{pol}} \Delta P_\zeta$ should be evaluated with care so as to minimize numerical inaccuracies. We have already seen in Fig. 7(c,d) that τ_{pol} possesses singularities at t-p boundaries, although not all of them are problematic as we will see shortly. The existence of zeros and singularities in the increment ΔP_ζ can be readily anticipated from the derivative $|P'_\zeta|_{E\Lambda}$ in Eq. (36), which can be approximated at leading order as

$$P'_\zeta(X) \approx -\frac{\Psi'_P(X)}{B_0} + \frac{\rho_0}{\hat{v}_\parallel(X)} \left(\hat{E} + \frac{1}{2} \hat{v}_\parallel^2(X) \right). \quad (38)$$

Clearly, $v_\parallel = 0$ constitutes a singularity and there are at least two points where the terms on the right-hand side cancel, giving $|P'_\zeta|_{E\Lambda} = 0$. Figure 8 shows an example for 3.5 MeV alpha particles. We consider orbits along the black parabolic line in Fig. 8(a), where $\Lambda = 0.865$.

Figure 8(b) shows the form of the increment $\Delta P_\zeta(X)$ evaluated using Eq. (35) (solid curves) and Eq. (36) (dotted). The values agree nicely nearly everywhere, except in the vicinity of the $v_\parallel = 0$ singularity labeled

⁵Eq. (37) differs from the formula for the volume element given in Eq. (40) of Ref. [13]. We consider the present derivation to be more rigorous. This affects mainly trapped particles, which were effectively absent in previous applications of the code *VisualStart*.

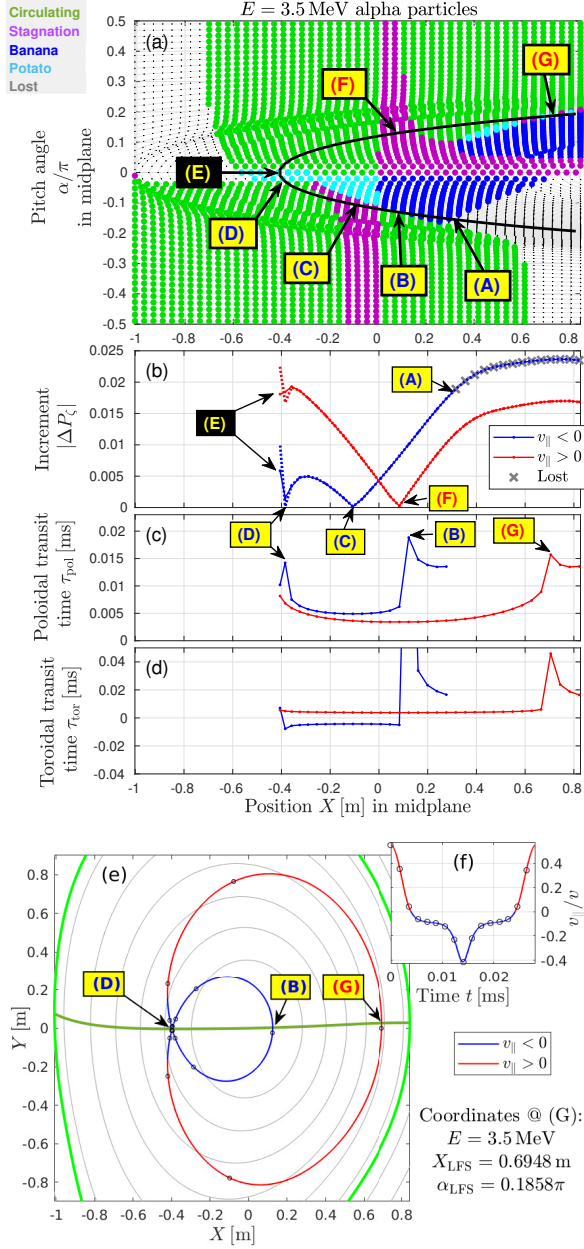


Figure 8: Radial dependence of the increment ΔP_ζ and the poloidal and toroidal transit times τ_{pol} and τ_{tor} on a contour $(E, \Lambda) = \text{const.}$ Panel (a) shows a CoM mesh constructed in the midplane using cell numbers $N_\Lambda = N_X = 48$ for alpha particles with $E = 3.5 \text{ MeV}$. The total number of $\Lambda = \text{const.}$ contours is $(N_\Lambda/2) + N_X = 72$ and we inspect contour $j = 40$ (counted from $\alpha = -\pi/2$), which is drawn as a black curve in (a). In panel (b), solid lines represent $|\Delta P_\zeta|$ given by Eq. (35) and the dotted lines are the alternative form $|P_\zeta \Delta X|$ in Eq. (36). Blue and red lines in (b)–(d) represent orbits starting with $v_{||} \leq 0$, respectively. Labeled arrows: (A) loss boundary, (B,D,G) t-p boundaries, (C,D,F) stagnation points, and (E) the $v_{||} = 0$ singularity. In fact, points (B,D,G) represent the same point in CoM space. This can be seen in panel (e), which shows the trajectory of a barely trapped orbit near the t-p separatrix. Panel (f) shows the time trace of $v_{||}/v$ on this orbit, and $v_{||} \leq 0$ legs are colored blue and red.

(E).⁶ A verification exercise similar to that performed later in Section 8.4 showed systematic errors when using Eq. (36). These errors seemed to be effectively absent when ΔP_ζ is evaluated using Eq. (35), which is therefore our default choice.

The zeros of $\Delta P_\zeta(X)$ correspond to stagnation points and imply that these and nearby orbits have a small weight factor W_{ijk} . In the example in Fig. 8 there are three such points: (C) and (F) are the O-type stagnation points in the domains of counter- and co-passing particles; whereas (D) is an X-type stagnation point situated on the separatrix of the t-p boundary.

The zero of $\Delta P_\zeta(X)$ at point (D) plays an important role: it cancels one of the three singularities of the poloidal transit time τ_{pol} in Fig. 8(c). In fact, a glance at the nearby barely trapped orbit in Fig. 8(e) shows that points (B,D,G) actually refer to the same transit time singularity; i.e., the same t-p separatrix. Strictly speaking, our algorithm counts such separatrices thrice, but one of these instances — namely the X-point at (D) — vanishes, owing to $\Delta P_\zeta(X)$ being zero at that location. This confirms that all orbits are effectively double-counted, so that the overall factor 1/2 appearing in Eq. (37) is justified even near the t-p boundary. Of course, on a discrete mesh the cancellation of the third sample at point (D) is unlikely to be perfect, so some amount of numerical inaccuracy must be expected.

The remaining transit time singularities — namely points (B) and (G) in our example — have the consequence that orbits representing those and nearby portions of GC phase space may accumulate a relatively large weight. One could compensate these singularities by demanding that the value of the distribution function f_{orb} vanishes at the t-p boundary; e.g., by including a factor τ_{pol}^{-1} in f_{orb} . However, this may not be physically meaningful as the following arguments show.

The particles populating the barely trapped orbit in Fig. 8(e) as well as nearby potato and counter-passing orbits spend a relatively large amount of time in a small portion of pitch-position space $(v_{||}/v, R, z)$ near V-type stagnation points. This fact is highlighted by the bunching of the small black circle symbols in Fig. 8(e,f), which represent 16 marker particles distributed uniformly in time. However, unlike the tiny O-type stagnation orbits around points (C) and (F), the large orbits near t-p boundaries like (B) and (G) can actually become densely populated in real plasmas. This becomes evident if one thinks in terms of particle sources: particles deposited on orbits located near a t-p boundary, like

⁶We expect that the red and blue curves in Fig. 8(b,c) connect smoothly near $X \approx -0.4 \text{ m}$ if one adds grid points closer to $v_{||} = 0$.

our example in Fig. 8(e), will be conveyed to the V-type stagnation points and spend a long time in that region, producing a local spike in the particle density.

Of course, the finite particle supply rate along with instabilities and collisions will prevent the formation of a true singularity in a real plasma. Nevertheless, the above arguments show that, **within certain limits**, the spikes in the transit times in Fig. 8(c) and resulting spikes in the orbit volume elements (37) and marker weights (31), are physical meaningful.

Such spikes can however cause numerical problems. In some cases, the resulting inaccuracies may be tolerable. In the worst case, the presence of singularities may prevent the attainment of numerical convergence, since a finer mesh will have grid points closer to the singularity. If necessary, one may adjust the weights as mentioned a few paragraphs earlier; e.g., by imposing an upper limit on their values (or their gradients) as in the flux limiter scheme of fluid dynamics. Although this trick may ensure convergence, it is likely to converge to a somewhat inaccurate value. On the other hand, one may argue that the correct value is unknown as it depends on physical mechanisms that have been ignored, such as scattering by collisions and field fluctuations. Perhaps the most elegant solution in the collisionless limit is to use an analytical estimate of a volume element near the singularity, provided that a well-behaved solution exists. Anyhow, we believe that the optimal approach to dealing with singularities depends on the application at hand, so we leave it for future research. It seems meaningful to deal with this problem in the context of smoothing techniques that will be needed to evaluate phase space gradients for instability analyses.

7. Modeling CoM distributions with large drifts

A computed or modeled equilibrium distribution f_{orb} is needed to determine the weight factor in Eq. (31). A model that is a function of CoM only, such as $G_{\text{mdl}}(\mathcal{E}, \mu, P_{\mathcal{L}})$, would directly yield an exact equilibrium distribution f_{orb} . While simple models may suffice for physics studies at a fundamental level, predictive simulations and simulations used to interpret observations require more realism. It would be convenient to have G_{mdl} given in an explicit, purely analytical form. However, for fast ions with significant magnetic drifts, a large number of control parameters is usually necessary to mimic realistic distribution functions, which tends to make the modeling job cumbersome, at least for non-

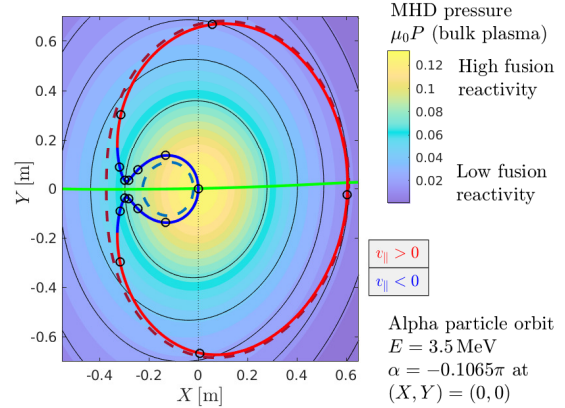


Figure 9: GC orbit of a barely trapped 3.5 MeV alpha particle born at the magnetic axis $(X, Y) = (0, 0)$ with pitch angle $\alpha = -0.1065\pi$. Small black circles indicate 16 marker particles separated by equal time intervals. At the nearby trapped-passing (t-p) boundary, this orbit decomposes into a small core-localized counter-passing orbit and a large potato orbit, which are indicated by dashed lines. Note that the outer orbit is disconnected from the hot central core, which appears yellow in the contours of the MHD pressure $P(\Psi_p)$ of the bulk plasma. This illustrates how the alpha particle distribution $f_{\text{orb}}(\mathcal{E}, \mu, P_{\mathcal{L}})$ can develop a strong nonuniformity around the t-p boundary since the fusion reactivity is not a function of the alpha's $P_{\mathcal{L}}$ but of the bulk pressure (color contours) and, thus, magnetic flux Ψ_p (black contours).

artificial intelligence.⁷

For instance, consider the problem of constructing a model for fusion-born alpha particles. Even if one assumes that their pitch angle distribution is uniform at birth time,⁸ strong pitch angle nonuniformities can promptly arise within one poloidal transit time. The loss cone, which contains orbits whose trajectories intersect plasma-facing components, is an obvious example. The magnetic drifts that cause those losses in toroidal geometry also affect the pitch angle distribution of confined orbits. For instance, for fusion-born alpha particles in ITER, the anisotropy associated with drift orbit topology was estimated to be around 10% (see Section 7 of Ref. [46]). Pitch anisotropies and bump-on-tail structures along the energy axis are also a well-known cause of velocity space instabilities that can be observed as ion cyclotron emissions (ICE) (e.g., see Refs. [47, 48]).

Let us consider a more concrete example. Figure 9

⁷With a human-manageable amount of control parameters, it can be difficult to design a CoM distribution that has both the desired radial profile and the desired pitch distribution, because both vary simultaneously when one varies parameters controlling the dependence of f_{mdl} on the canonical toroidal angular momentum $P_{\mathcal{L}}$.

⁸Precisely speaking, when an anisotropy exists in the distribution of fusion fuel particles, this anisotropy may also be imprinted on the fusion products due to angular correlations in nuclear reactions (e.g., see Ref. [44] and Fig. 10 in Ref. [45]). Here this effect is neglected.

shows the GC orbit of a barely trapped 3.5 MeV alpha particle, which traverses both the hot plasma core and the cooler plasma periphery. If one uses a slowly varying function $G_{\text{mdl}}(\mathcal{E}, \mu, P_\zeta)$ to model the overall distribution of alpha particles in this range of energies and pitch angles, the density will be similar on the barely trapped orbit shown in Fig. 9 and on the neighboring large potato and small counter-passing orbits indicated by dashed lines, because their coordinates have similar values.⁹ In other words, the particle density will be similar on both sides of the t-p boundary.

However, this may not be an accurate representation of a real distribution of fusion-born alphas, in particular when the fusion reactivity varies substantially in different portions of the orbit. To illustrate this, the 3.5 MeV alpha particle orbit in Fig. 9 has been underlaid with the contours of the sharply peaked MHD pressure field from our working example (cf., Fig. 4), which is characteristic for the bulk component of a JET plasma where 3-ion RF heating was applied near the magnetic axis [31, 32]. In such a case, fusion-born alphas from the plasma center will populate the entire orbit shown in Fig. 9, albeit in a nonuniform manner as indicated by the small black circles. Just across the t-p boundary, however, most of the newly born alphas will be concentrated on the small counter-passing orbit, while the large potato orbit would be populated only sparsely, since it is topologically disconnected from the particle source. In principle, this can produce a steep gradient in the distribution function $f_{\text{orb}}(\mathcal{E}, \mu, P_\zeta)$ across the t-p boundary. In reality, collisions or instabilities can be expected to have a smoothing effect, but depending on the relevant time scales, a strong nonuniformity may be sustained. Such information may be necessary for interpreting experimentally observed signals or for reproducing them in simulations, and it is advantageous to have the ability to construct suitable models.

A viable recipe for doing so can be found in Eq. (30). Such an orbit time average can be readily realized using our orbit database and constitutes a physically meaningful way to convert an arbitrary model function $G_{\text{mdl}}(\mathbf{Z})$ into a true equilibrium distribution f_{orb} :

$$f_{\text{orb}} = \langle G_{\text{mdl}}(\rho_P) \rangle_{\text{orb}} \equiv \frac{1}{\tau_{\text{orb}}} \oint d\tau G_{\text{mdl}}(\mathbf{Z}(\tau))$$

$$\approx f_{\text{orb}}(i, j, k) = \frac{1}{N_{\tau,ijk}} \sum_{l=1}^{N_{\tau,ijk}} G_{\text{mdl}}(\mathbf{Z}_{\text{gc},ijkl}); \quad (39)$$

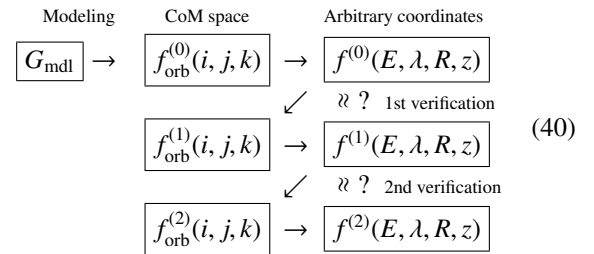
⁹In our example: $\mu B_0/E = 0.875...0.892$ and $\hat{P}_\zeta = 0.215...0.219$ with normalized $\hat{P}_\zeta \equiv \psi_P - \hat{v}_\parallel \frac{B_\zeta}{B} \frac{\rho_0 B_0}{\Psi_{Pa} - \Psi_{P,0}}$ and $0 \leq \psi_P \leq 1$.

where $\mathbf{Z}_{\text{gc},ijkl}$ is the position of marker l on orbit (i, j, k) . For instance, in the example discussed in the previous paragraphs, one may let $G_{\text{mdl}} \propto P(\Psi_P)$, so that the MHD pressure field in Fig. 9 acts as a particle source.

This approach can be viewed as taking the path of “self-organization”, where we leave it to the equations of motion to determine a distribution function that is consistent with the magnetic geometry and system boundaries at hand, and the constraints encoded in G_{mdl} determine the particular spatial profile and velocity distribution for a certain case. This idea also underlies Monte-Carlo codes that follow GC orbits many times around the torus in the presence of sources and collisions. We mimic a part of this process with reduced computational effort: the GC orbits in our database are followed for only one poloidal turn in the absence of any perturbations (field fluctuations and collisions).

8. Application and verification

In this section, we demonstrate how the methods that we presented in this paper can be applied to a practical example. Section 8.1 describes the setup and how we construct — from a simple model G_{mdl} — an equilibrium distribution f_{orb} of alpha particles that are assumed to be born near the center of a large tokamak plasma with an isotropic distribution of pitch angles at birth time. The results are discussed in Sections 8.2 and 8.3. In the final Section 8.4, we demonstrate how we use our orbit database to convert a binned 4-D distribution function $f^{(0)}(E, \lambda, R, z)$ (or any other coordinates) back to an orbit-based representation $f_{\text{orb}}^{(1)}$ in CoM space. The result is binned again to give $f^{(1)}$ for comparison with the original $f^{(0)}$. The procedure is then iterated one more time, yielding $f^{(2)}$, in order to reveal systematic errors. The contents of this section may be summarized in diagrammatic form like this:



Strictly speaking, we transform only between 2-D and 3-D spaces, $(A_j, P_\zeta(X_k)) \leftrightarrow (\lambda, R, z)$, since both sets share the energy coordinate E , which is a conserved quantity, so the mapping in that direction is identical. We will work here with an arbitrary normalization

Form of E -distrib.	Sampling parameters				$E_0 = 35 \text{ keV}$		$E_0 = 350 \text{ keV}$		$E_0 = 3.5 \text{ MeV}$	
	N_E	N_α	N_X	N_τ	N_{orb}	N_{mrk}	N_{orb}	N_{mrk}	N_{orb}	N_{mrk}
Mono- E_0	1	2×64	2×64	$128 \frac{L_{\text{orb}}}{L_{\text{bnd}}} \text{ (min.: 16)}$	33,335	1.88M	31,841	1.70M	27,684	1.29M
Flat- $E \leq E_0$	24	2×24	2×24	$48 \frac{L_{\text{orb}}}{L_{\text{bnd}}} \text{ (min.: 16)}$	112,483	2.74M	—	—	99,308	2.18M

Table 2: Phase space sampling parameters in our five test cases. N_E, N_α, N_X determine the number of cells in (E, α, X) -space defined in Section 4. The number N_τ of markers sampling an orbit is (with a lower bound of 16) chosen to be proportional to the orbit length L_{orb} , the reference value being the number of markers per plasma boundary length, here $L_{\text{bnd}} \approx 7.50 \text{ m}$. There are N_{mrk} markers sampling N_{orb} confined orbits.

$E [\text{keV}]$	Circ.	Stagn.	Banana	Potato	Lost \approx
35	25,780	359	7,158	38	780
350	24,430	1,066	6,191	154	2,280
3500	20,531	3,165	3,505	483	6,430
0...35	87,368	612	24,456	47	1,710
0...3500	75,178	7,208	15,861	1,061	14,880

Table 3: Partition of orbit types (circulating, stagnation, banana, potato, lost) as defined in Section 4. These numbers are *not* weighted by particle densities, so they only characterize the partition of orbit samples on our mesh. The actual prompt losses of real particles are negligible in our large plasma with centrally peaked density profile.

(henceforth indicated by hats), so the values of \hat{f}_{orb} of \hat{f} must match but bear no particular meaning.

Our distribution functions will be inspected here in the same way as an experimental diagnostician would: measure spatial distributions by integrating over certain portions of velocity space, and measure velocity distributions by integrating over certain portions of position space. The required binning operations and Jacobians are described in Appendix A.

8.1. Setup and parameters

In order to highlight how toroidal geometry affects the form of the constructed equilibrium distribution function via the mirror force and magnetic drifts, we start from a quasi-distribution G_{mdl} that is independent of pitch and has a relatively sharp peak in minor radius:

$$G_{\text{mdl}}(\rho_p) = C \times \hat{n}_{\text{mdl}}(\rho_p) \times H_{\text{mdl}}(E), \quad (41)$$

where C a normalization constant that is arbitrary here.

The radial density profile $\hat{n}_{\text{mdl}}(\rho_p)$ is shown as a solid black line in Fig. 10(a). Although this is primarily meant to be a toy model for testing our algorithms with a localized particle source and large magnetic drifts, we note that this profile is physically feasible if one considers scenarios with a sharply peaked ion temperature profile as produced in 3-ion-heated JET plasmas [31, 32].

The energy dependence of particle distributions in a real plasma is governed by processes such as collisional drag and instabilities (e.g., see Fig. 4 in Ref. [17]).

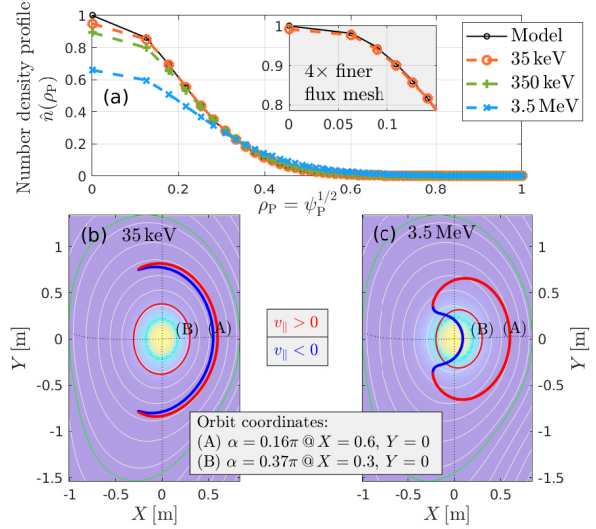


Figure 10: Panel (a) shows the radial profiles of our source model $\hat{n}_{\text{mdl}}(\rho_p)$ used in Eq. (41) (black) and constructed mono-energetic alpha particle densities $\hat{n}(\rho_p)$ (colored) as functions of the square root of normalized poloidal flux, $\rho_p = \psi_p^{1/2}$, on a grid that is uniform in ψ_p . For 35 keV and 3.5 MeV, panels (b) and (c) show the corresponding density fields $\hat{n}(X, Y)$ in the poloidal plane, overlaid with examples of (A) trapped and (B) circulating orbits as well as flux surfaces (gray).

These processes are not considered here. We use energy only to exemplify the effects of magnetic drifts, which increase with E ; or, rather, with its square root \sqrt{E} , because the effect of magnetic drifts is roughly proportional to the ratio v_{dB}/v of the drift velocity to the particle velocity. For this purpose, we use two types of distributions: one mono-energetic with $E = E_0$, and one that is flat in the range $0 \leq E \leq E_0$ and zero beyond:

$$H_{\text{mdl}}(E) = \begin{cases} \delta(E - E_0) & = \text{mono-energetic,} \\ \frac{1 - \int_0^E dE' \delta(E' - E_0)}{E_0} = \frac{1}{E_0} & \text{for } 0 \leq E \leq E_0, \end{cases} \quad (42)$$

where δ is the Dirac delta distribution.

Table 2 shows the parameters for five test cases that we will consider here: three mono-energetic alpha distributions with $E = E_0 = (35, 350, 3500) \text{ keV}$ and two distributions that are uniform in the energy range

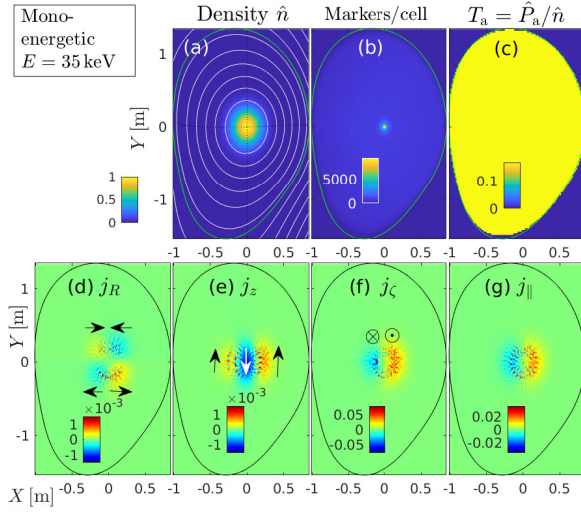


Figure 11: Moments of the mono-energetic alpha distribution with 35 keV. We integrated over velocity space using Eq. (A.5) with the following weight factors: (a) w for particle density $\hat{n}(X, Y)$; (b) 1 for the number of markers per cell; (c) $w \times E/3$ for alpha particle pressure \hat{P}_a , then divided by density to yield temperature $T_a = \hat{P}_a/\hat{n}$; (d)–(g) $w \times (v_R, v_z, v_\zeta, v_\parallel)$ for horizontal, vertical, toroidal and parallel current densities $(j_R, j_z, j_\zeta, j_\parallel)$. The amplitude is normalized to $\hat{n}_0 = 1$.

$0 \leq E \leq E_0$ with $E_0 = (35, 3500)$ keV. Table 3 summarizes the numbers of circulating, stagnation, banana, potato and lost orbit samples (CoM mesh points) in each case. Panels (b) and (c) of Fig. 10 show how the contours of a banana orbit (A) and a circulating orbit (B) change due to magnetic drifts when the kinetic energy is increased from 35 keV to 3.5 MeV.

While the model in Eq. (41) has a simple appearance, the actual form of the alpha particle distributions we construct here is determined by the combination of Eq. (41) and the equations of GC motion (21) and (22), whose solutions are folded into the simple model via the integral (39). In other words, the geometric effects are captured by the GC orbits in our database, whereas the model G_{mdl} effectively describes the structure of the particle sources that populate those orbits, and the integral (39) combines all that information.

The geometric effects that we expect to see in this setup may be summarized as follows. These properties of a toroidally confined and radially bounded plasma are well-known and are listed here only for completeness and to simplify the discussion of the results:

- The peripheral plasma will be populated by particles on orbits that also pass through the core as in Fig. 10(c). Such orbits are found in a certain range of pitch angles around the t-p boundary.
- Flux surface-averaged profiles tend to be smoothed

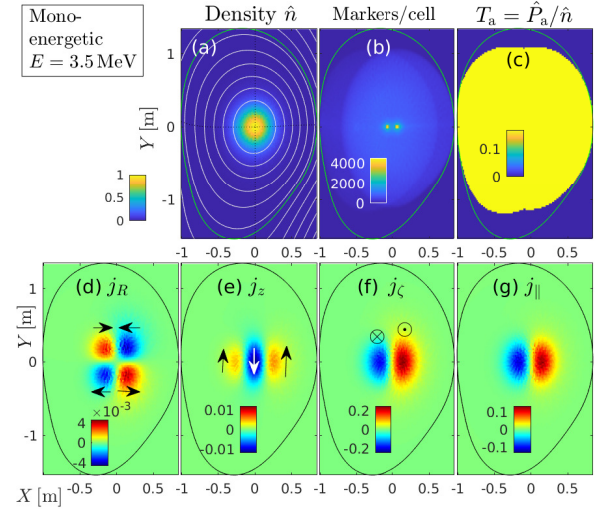


Figure 12: Moments of the mono-energetic alpha distribution with 3.5 MeV. Arranged as Fig. 11.

on the scale length of the magnetic drifts. The gyroradii have a similar effect and may be included via satellite particles [49] (not done here).

- Magnetic drifts shift co-passing orbits outward in R and counter-passing orbits inward.
- The mirror force confines some particles to the low-field side of the plasma and causes them to accumulate near V-type stagnation points.
- The energy- and pitch-dependence of the above-mentioned spatial nonuniformities gives rise to net flows both poloidally (due to radial nonuniformity) and toroidally (due to a local imbalance between co- and counter-going particles).

8.2. Properties of mono-energetic distributions

Figure 10(a) shows the computed density profiles $\hat{n}(\rho_p)$ on a mesh that is uniformly spaced in normalized poloidal flux $\psi_p = \rho_p^2$. The profile for 35 keV alphas is similar to the model $\hat{n}_{\text{mdl}}(\rho_p)$. The small discrepancy at $\rho_p = 0$ can be eliminated with a finer mesh as shown in the inset. The magnetic drifts increase with increasing energy, causing the profile $\hat{n}(\rho_p)$ to broaden and the central value $\hat{n}(0)$ to drop. The total number of particles in the plasma is nearly the same in all three cases, since prompt losses are relatively small: less than 1% for 350 keV and about 2.5% for 3.5 MeV.

Figures 11 and 12 show the form of various moments of the alpha distributions with kinetic energies 35 keV and 3.5 MeV in the poloidal plane (X, Y) :

- Particle density (a): Even for 3.5 MeV, our model yields an alpha density that is peaked at the axis, but the peak is wider than in the 35 keV case, especially in the horizontal ($X = R - R_0$) direction.
- Markers/cell (b): Since we have set a lower bound of 16 markers per orbit, the smallest stagnation orbits stand out in the plots showing the number of markers per cell, while the surroundings are fairly uniform. At 35 keV, the co- and counter-passing stagnation points effectively coincide with the magnetic axis. At 3.5 MeV, their mean separation along X is about 0.14 m (cf. Fig. 8(b)).
- Temperature (c): With the density nonuniformity divided out, the 3.5 MeV temperature field clearly shows that our alpha particle distribution effectively consists of two partially overlapping co- and counter-going mono-energetic beams, which are shifted radially out- and inward, respectively.
- Flows (d–g): The relative shift of the co- and counter-going beams causes net local currents. The poloidal current field (consisting of j_R and j_z) forms two vortices, one on the high-field side (HFS) of the magnetic axis rotating clockwise, and one on the low-field side (LFS) rotating counter-clockwise. Toroidal and parallel currents j_ζ and j_\parallel are very similar here, since the magnetic pitch is relatively uniform ($q \approx 1$) in the plasma center. The structure and spatial extent of the currents is similar at lower and higher energies, but the magnitude should be proportional to \sqrt{E} and, thus, differ by about a factor of 10. This seems to be the case here if one accounts for the fact that the 35 keV case has a smaller signal-to-noise ratio.

Figure 13 shows the pitch angle distribution of our mono-energetic alpha particles. The spatially integrated distributions $\hat{v}(\alpha|E_0)$ in panel (a) show that a pitch nonuniformity exists around $\alpha = 0$, up to intermediate pitch angles, $|\alpha| \lesssim 0.2\pi$. The deviation from the mean increases from about 10% for 35 keV to about 50% for 3.5 MeV. At larger pitch angles, $|\alpha| \gtrsim 0.2\pi$, the distributions are flat (except for [aliasing noise, to be discussed later](#)). More detailed views of the nonuniformities in the region $|\alpha| \lesssim 0.2\pi$ of the cases with $E_0 = 35$ keV and 3.5 MeV are provided in panels (b) and (c), where we plot the contours of $\hat{v}(\alpha, X|E_0)$ to show the radial dependence of the pitch angle distribution in the range $-0.48 \text{ m} < X < 0.48 \text{ m}$. Here, the summation over markers was performed over the Y coordinate for radial bins of size $\Delta X = 0.04 \text{ m}$.

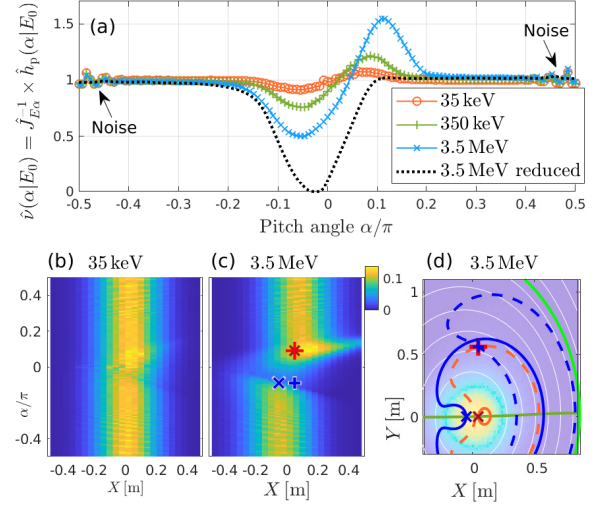


Figure 13: Pitch angle distributions for the mono-energetic cases. Panel (a) shows pitch profiles $\hat{v}(\alpha|E_0) = \hat{J}_{E\alpha}^{-1} \times \hat{h}_p(\alpha|E_0)$, where $\hat{J}_{E\alpha} = 2\hat{v}_\perp$ is the Jacobian from Eq. (A.12), and $\hat{h}_p(\alpha|E_0)$ is the pitch angle histogram integrated over the entire spatial domain and divided by B_\parallel^*/B as in Eq. (A.1b). The black dotted curve shows the 3.5 MeV result for a reduced domain size, where the artificial plasma boundary has been placed at 0.25% of the flux space [33], removing many large orbits (scaled to match the blue curve outside the loss cone). Panels (b) and (c) show contours of the local pitch distribution $\hat{v}(\alpha, X|E_0) = \hat{J}_{E\alpha}^{-1} \times \hat{h}_p(\alpha, X|E_0)$ obtained by integrating over the height of the spatial domain in radial bins of size $\Delta X = 0.04 \text{ m}$. The two crosses and two plus symbols (sometimes overlapping) in panels (c) and (d) mark the starting points of four alpha particle orbits with $E_0 = 3.5 \text{ MeV}$ whose poloidal contours are shown in panel (d). Blue and orange orbits start with $\alpha = -0.09\pi$ and $+0.09\pi$, respectively. Solid orbits start from $Y = 0$ (crosses) and dashed orbits from $Y = 0.56 \text{ m}$ (pluses). Panel (d) also shows the color contours of the 3.5 MeV alpha particle density field from Fig. 12(a), Ψ_P contours (white), the midplane (dark green), and the boundary (light green).

The structures seen in Fig. 13 can be readily explained on the basis of our model function G_{mdl} , the effects of toroidal geometry (mirror force and magnetic drifts), and the loss boundary. Rather than saying anything new, the purpose of the following discussion is to show that our procedures work reliably and produce the expected results. We focus on the 3.5 MeV case.

Figure 13(c) clearly shows the opposite radial shift of co- and counter-going particles due to their opposite magnetic drifts: the peak in the region $\alpha > 0$ is shifted towards the right ($X > 0$) and the peak in the region $\alpha < 0$ is shifted to the left ($X < 0$). The wedge-shaped structure of the pitch anisotropy around $\alpha = 0$ represents the widening of the domain of trapped particles towards the low-field side as expected from Fig. 8(a).

In order to explain the structure of the pitch anisotropy in the region $|\alpha| \lesssim 0.2\pi$, it is helpful to consider the trajectories of a few GC orbits in that re-

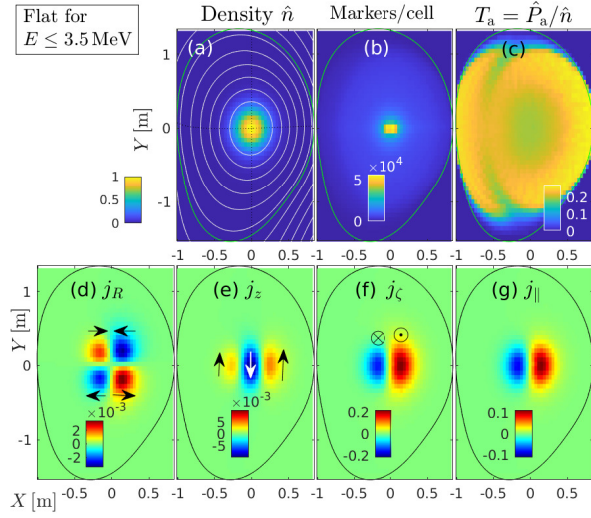


Figure 14: Moments of the alpha distribution that is uniform in the energy range $E = 0 \dots 3.5$ MeV. Arranged as Fig. 11.

gion. For this purpose, Fig. 13(d) shows the GC orbits of four simulation particles with energy $E = 3.5$ MeV. Two of them (solid lines) are launched near the mid-plane ($Y_{\text{start}} \approx 0$): the small orange orbit starts from $(X, \alpha)_{\text{start}} = (0.05 \text{ m}, 0.09\pi)$, and the blue orbit starts from $(X, \alpha)_{\text{start}} = (-0.05 \text{ m}, -0.09\pi)$, as indicated by the crosses in panels (c) and (d). The blue orbit is a large potato orbit that passes both near the center of the plasma and through the periphery. The small orange orbit is a stagnation orbit. Now recall Eq. (39), where we defined the orbit distribution f_{orb} to be the orbit time-average of the model $G_{\text{mdl}}(\rho_P)$ that represents our alpha particle birth profile. Since the latter is sharply peaked near the plasma center, the values of $G_{\text{mdl}}(\rho_P)$ can differ substantially on the above-mentioned pair of orbits.

Generally speaking, orbits that spend most of their time inside the peak of the source profile $G_{\text{mdl}}(\rho_P)$ are densely populated with physical particles, so our GC markers representing them have larger weights w_l . Orbits that spend most of the time outside are sparsely populated. The mirror force and magnetic drifts have the consequence that particles on counter-passing orbits spend less time inside the central peak of $G_{\text{mdl}}(\rho_P)$ than co-passing orbits. In addition, the losses of co- and counter-going particles are asymmetric, especially in the vicinity of the trapped-passing boundary. These effects are manifested in the minima and maxima of $\hat{v}(\alpha|E_0)$ and $\hat{v}(\alpha, X|E_0)$ in Fig. 13, which are most pronounced in the high-energy case due to its larger drifts.

Note that the distribution $\hat{v}(\alpha, X|E_0)$ in Fig. 13(c) was computed by integrating along the vertical coordi-

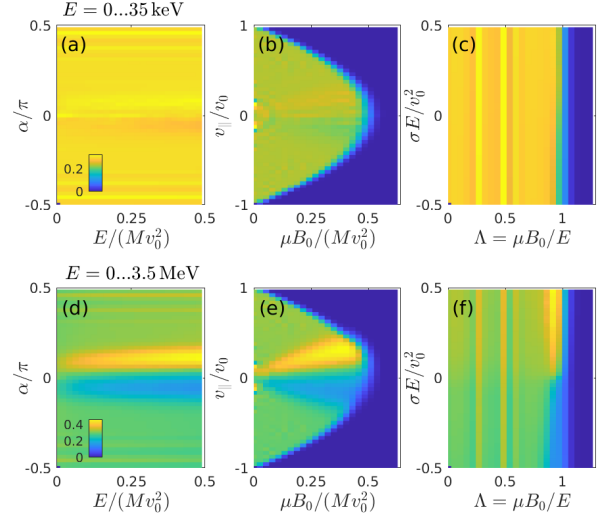


Figure 15: Spatially integrated velocity distributions $\hat{v}(a, b) = \mathcal{J}_{ab}^{-1} \times \hat{l}_p(a, b)$ in the cases that are uniform in the energy ranges $E = 0 \dots 35$ keV (top) and $0 \dots 3.5$ MeV (bottom), visualized in different coordinates: $\hat{v}(\alpha, \hat{E})$ on the left, $\hat{v}(\hat{v}_{\parallel}, \hat{\mu})$ in the central column, and $\hat{v}(\hat{E}, \Lambda)$ with $\hat{E} \equiv \sigma \hat{E}$ on the right. For plots of $\hat{v}(\alpha, \hat{E})$, see Fig. 16. Note that different color scales are used in the upper and lower row.

nate Y , so it also includes particles on orbits like those shown by dashed lines in panel (d). The dashed orbits were launched from $(X, \alpha)_{\text{start}} = (0.05 \text{ m}, \pm 0.09\pi)$, but more than half a meter above the midplane, from $Y_{\text{start}} = 0.56 \text{ m}$, as indicated by pluses in panels (c) and (d). The orbit starting with a positive pitch (orange) passes through the central plasma, where it acquires a relatively large weight due to the centrally peaked source profile $G_{\text{mdl}}(\rho_P)$. In contrast, the orbit starting with a negative pitch travels outward and nearly hits the boundary (light green curve).

In fact, the entire peak of $\hat{v}(\alpha|E_0)$ around $\alpha/\pi \approx +0.1$ can be explained in that way. We have verified this with a test case where the plasma boundary had been placed at $\psi_P = 0.25$ ($\rho_P = 0.5$) instead of $\psi_P = 1$, so that orbits are constrained to the inner 25% of the present flux space [33]. In that case, the plasma boundary is located around $X \approx 0.38 \text{ m}$ at the height of the outer midplane, so that the three large orbits in Fig. 13(d) are all lost. The resulting pitch angle profile is shown as a black dotted line in Fig. 13(a). It has a deeper and wider loss cone, and there is no peak at $\alpha/\pi \approx +0.1$.

8.3. Properties of flat-energy distributions

Realistic alpha particle distributions are nonuniform in energy. The reason for our use of a model that is flat in energy is that the E -dependence of geometrically induced nonuniformities can be easily seen, and it is also

easier to verify that our method works as expected.

The densities and current fields are very similar to those in the mono-energetic cases shown in Figs. 11 and 12 above. A few different features can be observed in the high energy case, $E = 0 \dots 3.5$ MeV, whose moments are shown in Fig. 14. The presence of particles with a wide range of energies can be inferred primarily from the blurring of the stagnation points in panel (b), and from the nonuniformity of the temperature field in (c). The latter resembles *somewhat* the letters “IO” as fast particles dominate outside the central density peak; especially the co-passing ones (forming the “O” shifted to the right) for the same reasons as discussed in the last paragraphs of Section 8.2 above.

Figure 15 shows the velocity distributions integrated over position space. One can see that, apart from noise, our result has the desired uniformity in both energy and pitch in the domains populated by passing particles. As discussed in Section 8.2 above, the pitch anisotropy around $\alpha = 0$ is induced by toroidal geometry (mirror force, magnetic drifts and resulting losses) in combination with the centrally peaked source profile. The anisotropy clearly increases with increasing energy.

Our orbit database is relatively sparse ($N_E \times N_\alpha \times N_X = 24 \times 48 \times 48$, see Table 2), and the bins used in Figs. 14 and 15 have effectively the same resolution: $N_E \times N_\alpha \times N_{Rz} = 24 \times 48 \times 48^2$. The binned results are smooth along energy, because the orbit samples coincide with the bins. Most of the noise in Fig. 15 is in the pitches α and Λ , where the distribution of bins is similar but not identical to the distribution of orbit samples (CoM space). In other words, the noise in the range $0.3 \lesssim |\alpha|/\pi \lesssim 0.5$ and $\Lambda \lesssim 0.7$ is effectively an *aliasing* effect. The noise is largely suppressed in the poloidal plane (X, Y) in Fig. 14, and in $(\hat{v}_\parallel, \hat{\mu})$ -space shown in the central column of Fig. 15, since the bins in these coordinates have a *shape very different from the mesh* we used to sample the CoM space (cf. Figs. 6 and 8).

Figure 15 also highlights potential difficulties in attempts to convert the binned distributions from one set of coordinates to another. In particular, the small region around the origin $(v_\parallel, \mu) = (0, 0)$ in the central column of Fig. 15 is expanded to the full length of an axis in (α, E) and (E, Λ) spaces on the left and right sides of Fig. 15, so some features of the distribution functions are likely to get lost in direct conversions unless a polar mesh is used in (v_\parallel, μ) coordinates, which is equivalent to using another set of coordinates, such as (E, α) . We emphasize that we have *not* performed direct conversions between binned distributions in Fig. 15. They were all obtained by binning, on the respective mesh, the marker particles of the orbit-based representation f_{orb} .

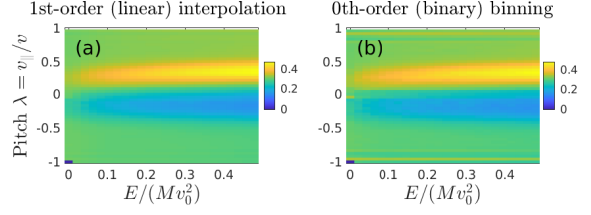


Figure 16: Comparison between 1st- and 0th-order binning results $\hat{v}(E, \lambda)$ in the flat-energy case with $E = 0 \dots 3.5$ MeV. Here, we loaded $N_\tau = 480 \times L_{\text{orb}}/L_{\text{bnd}}$ (min. 16) markers per orbit, giving a total of $N_{\text{mrk}} \approx 18\text{M}$ markers. Panel (b) is a projection of the input distribution $\hat{f}^{(0)}(E, \lambda, R, z)$ used in our verification exercise in Section 8.4.

The binning in Figs. 10–15 was performed using a 1st-order (linear) interpolation scheme as commonly used in *particle-in-cell* (PIC) codes, where each marker is assumed to have the shape of a top-hat function *whose width is equal to the bin size*. Naturally, the noise is larger with a 0th-order (binary) binning scheme, where markers resemble δ functions. The difference between 1st- and 0th-order binning can be seen by comparing panels (a) and (b) of Fig. 16, where we show the spatially integrated velocity distribution $\hat{v}(E, \lambda)$ with $\lambda = v_\parallel/v = \sin^{-1}(\alpha)$. Note also that the result in Fig. 16(a) appears smoother than that in 15(d) simply by using λ instead of α , because this change in coordinates reduces the above-mentioned aliasing effect. Another difference is that Fig. 16 was obtained with 10 times more markers per orbit, but this does not have a strong influence on the aliasing in the domain of deeply passing particles, since those markers rarely cross the bin boundaries.

8.4. Verification of reversibility and accuracy of coordinate conversions via an orbit-based representation

The results discussed in the previous sections were obtained by binning the marker weights of our orbit-based representation in CoM space to a mesh in arbitrary coordinates. The reverse transformation is also easily performed as we will demonstrate in this section. On platforms like ITER IMAS, we expect this to be an essential part of the transformation toolbox, especially if an orbit-based representation is used as a reference. Here, this operation serves us primarily as a verification exercise to test with what degree of accuracy we can perform and reverse coordinate conversions without applying any smoothing algorithms.

We consider the flat-energy case with $E_0 = 3.5$ MeV and follow the procedure that was outlined in Eq. (40). Let us now define the relevant steps in detail:

1. Using Eqs. (39) and (41), we construct a distribu-

tion function f_{orb} in the orbit-based representation:

$$G_{\text{mdl}} \rightarrow f_{\text{orb}}(i, j, k). \quad (43)$$

The CoM mesh that underlies our orbit database has the same number of cells as specified in Table 2: $N_E \times (N_\alpha \times N_X) = 24 \times 48^2$. The number of markers per boundary length is increased by a factor 10 to $N_\tau = 480 \times L_{\text{orb}}/L_{\text{bnd}}$ (min. 16), giving a total of $N_{\text{mrk}} = 18$ M markers.

2. The orbit-based CoM distribution f_{orb} is binned on a uniform 4-D mesh in the widely used coordinates (E, λ, R, z) . In compact form, this operation can be written as

$$f_{\text{orb}}^{(\eta)}(i, j, k) \rightarrow f^{(\eta)}(E, \lambda, R, z) \quad (44)$$

where $\eta = 0, 1, 2$ is the iteration number. The initial result $f^{(0)}$ is referred to as the “input” distribution for the subsequent back-and-forth transformations. More precisely, the binning operation we perform is given by Eq. (A.3) in the form

$$f(E_i, \lambda_j, R_k, z_m) = \sum_{n=1}^{N_{\text{mrk}}} \frac{w_n \times \Pi(\mathbf{Z}_{\text{gc},n} - \mathbf{Z}_{ijkm})}{(2\pi)^2 \left[\frac{B_{\parallel}^*}{B} \hat{\nu} R \right]_n \Delta E_i \Delta \lambda_j \Delta R_k \Delta z_m}. \quad (45)$$

where the shape factor $0 \leq \Pi \leq 1$ is a top-hat function, which means that the markers resemble δ functions. That is, we use the 0th-order binning scheme as in Fig. 16(b) and the number of bins (= cells) is $N_E \times N_\lambda \times (N_R \times N_z) = 24 \times 24 \times 48^2$. The pitch resolution was reduced from $N_\lambda = 48$ to 24 in order to reduce noise and aliasing (cf. Section 8.3).

3. For each orbit in the database, we compute new weight factors $W_{ijk}^{(\eta+1)}$ by integrating $f^{(\eta)}(E, \lambda, R, z)$ in time τ along the orbit contour as

$$W_{ijk}^{(\eta+1)} = \left[\Delta \mathcal{V}_{\text{orb}} \sum_{l=1}^{N_\tau} f^{(\eta)}(E, \lambda(\tau_l), R(\tau_l), z(\tau_l)) \right]_{ijk}, \quad (46)$$

which gives a new orbit-based distribution:

$$f^{(\eta)}(E, \lambda, R, z) \rightarrow f_{\text{orb}}^{(\eta+1)}(i, j, k). \quad (47)$$

4. Repeat step 2 and compare the result with the previous iteration:

$$f^{(\eta+1)} \stackrel{?}{\approx} f^{(\eta)}. \quad (48)$$

We have then iterated steps 3 and 4 one more time, looking for increasing deviations that may hint at systematic errors. The results are summarized in Figs. 17 and 18.

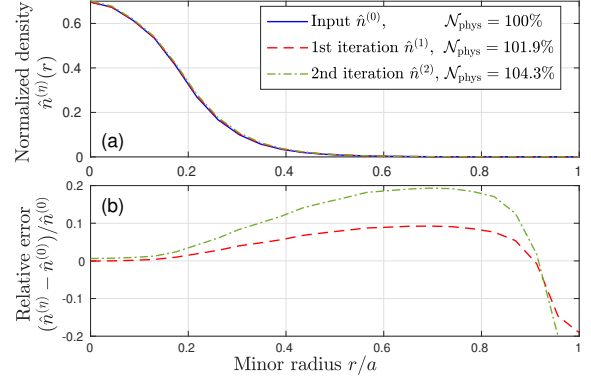


Figure 17: (a) Radial density profiles $\hat{n}^{(\eta)}(r)$ with iteration index $\eta = 0, 1, 2$, and (b) their relative errors in the flat-energy case $E = 0 \dots 3.5$ MeV. Unlike in Fig. 10, the radial profiles in panel (a) here are all normalized by the same value. The total number N_{phys} of physical particles increases slightly with each iteration as shown in the legend.

Figure 17(a) shows that the overall shape of the radial density profile is reproduced well. A closer inspection of the relative errors in Fig. 17(b) reveals that the first iteration changes the particle density by about +2% in the densely populated region $r/a \lesssim 0.3$. Around $r/a \approx 0.7$ the increase is about +10%, and near the boundary we have a reduction by about -20%. The relative error roughly doubles in the second iteration, indicating that it is not random but systematic. As indicated in the legend of Fig. 17(a), the total number of physical particles N_{phys} increases by 1.9% in the first and 2.4% in the second iteration, where the error accumulates to 4.3%.

Figure 17(b) shows that our procedure has the tendency to increase the particle density in the plasma interior and reduce it near the plasma boundary. We have not investigated why the systematic errors have this particular form and it may, in fact, depend on the particular magnetic configuration at hand. However, the larger errors in the outer region of the plasma are likely related to the fact that, in our test case, this region is populated almost exclusively by particles on large orbits near the trapped-passing boundary (cf. Fig. 10(c)), where the poloidal transit time τ_{pol} exhibits singular behavior (see Fig. 8) and inaccuracies can be expected. In the present case, the magnitude of these errors can be reduced by increasing the number of phase space samples as shown in Appendix B.4.

Figure 18 shows the results for the velocity space distributions $\hat{\nu}^{(\eta)}(E, \alpha)$ and $\hat{\nu}^{(\eta)}(E, \lambda)$ for iterations $\eta = 0, 1, 2$. No significant differences can be seen in the energy direction, which is to be expected since it is a conserved quantity and our energy bins are identical in both representations. But even along the pitch coordinates α

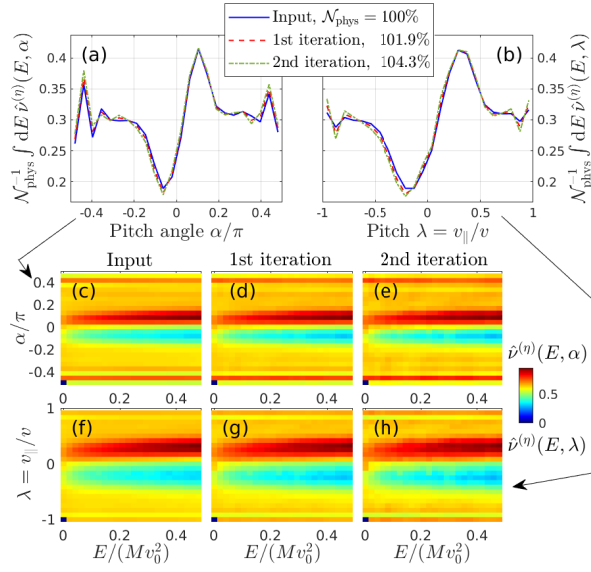


Figure 18: Spatially integrated pitch and energy distributions for the cases in Fig. 17. The curves in (a) were normalized by the number of physical particles N_{phys} at each iteration (shown in the legend) in order to highlight the degree of accuracy at which their shape is reproduced. The contours in (c)–(f) are colored differently from Figs. 15 and 16 to improve the visibility of small deviations. Usage of 0th-order (binary) binning is responsible for the aliasing noise being larger in panel (a) here than in Fig. 13(a), where a 1st-order (linear interpolation) scheme had been used.

and λ , excellent agreement is obtained after dividing out the systematic increase of N_{phys} . The noise at large pitch values retains its form as expected for an aliasing effect, and its magnitude increases systematically from one iteration to the next. The minimum associated with the loss cone tends to deepen with each iteration, which is also to be expected due to the irreversibility of losses. As was noted in connection with Fig. 16, we see again in Fig. 18 that binning on a uniform mesh in λ instead of α reduces aliasing and, thus, yields smoother results in the deeply passing domain ($|\alpha| \gtrsim 0.3\pi$ or $|\lambda| \gtrsim 0.8$).

Before proceeding to the conclusion of this paper, we shall add a few encouraging words for readers who may find the results in this section somewhat unsatisfactory. First, it should be noted that we have chosen a fairly challenging setup for our application example: Our density peak is localized deeply in the plasma core, where derivatives of the poloidal magnetic flux Ψ_p are small so that P_ζ is dominated by drift terms. This means the so-called “non-standard” orbits are actually prevalent here. Second, we have sampled the CoM space only relatively sparsely (Table 2) and used no smoothing procedure whatsoever. Even the binning of particles was done using a 0th-order scheme, which is why the noise

in Fig. 18 is larger than in Fig. 13.

In other words, one may say the we have been looking here at a worst-case scenario with the least degree of sophistication. This should be taken in a positive way, because it means that there is much room for improvement even with well-established techniques. Higher resolution in CoM space, higher-order binning schemes and other smoothing techniques are obvious solutions to reduce noise to a level where the distribution becomes sufficiently smooth to measure local phase space gradients of a particle distribution. The application of such techniques in combination with the orbit-based representation method described here is left for future work.

9. Summary and conclusion

This paper was motivated by the need to model and process distributions of charged particles in tokamak plasmas in a unified framework like the ITER Integrated Modelling & Analysis Suite (IMAS), which has to address the needs of a diverse community of users dealing with experimental and numerical data. One of the tasks IMAS will need to perform is the conversion of distribution functions between various coordinates. Our proposal is to construct a database of unperturbed GC drift orbits and use it as a basis for representing distribution functions in constants-of-motion (CoM) space. A concrete workflow based on our code VisualStart [12, 13] is proposed and described in detail in this paper.

It should be noted that dedicated IMAS data structures for storing GC orbit properties and quantities along GC trajectories of marker distributions have recently been added into the ‘distributions’ IDS (Interface Data Structures) [50], making IMAS readily compatible with the representation and methods proposed here.

An orbit-based representation maps any input distribution to an exact equilibrium that is consistent with the specified magnetic field configuration. While the result may generally differ from the original input, the equilibrium constraint ensures that all subsequent mappings from 3-D CoM orbit space back to arbitrary 4-D sets of coordinates become unique and straightforward. In fact, further operations like visualization or exporting are reduced to binning marker particles on user-defined meshes in arbitrary coordinates and arbitrary subspaces.

A new orbit database needs to be set up for each MHD equilibrium, and one may even have to set up one for different energy ranges (say, 1...tens keV, and 0.1...few MeV). The required computational effort should be manageable if the code is optimized and capable of exploiting state-of-the-art computing technology. Once the database has been constructed, it can be used

to represent and model various populations of charged particles that may be present in the same plasma, such as beam ions, RF-heated ions and fusion products. One orbit database may be shared between particle species with identical charge-to-mass ratio Qe/M by scaling their energies.

The time spent on preparing the orbit database is also justified by the fact that the acquired information can be used for other important tasks, such as resonance analyses and modeling, which must also be part of the IMAS toolbox. The degrees of freedom for modeling are vast, since each orbit can be assigned an independent weight. Rather than prescribing parametric functions of CoM — whose limitations were briefly discussed in Section 7 — it is worth exploring unconventional ways that make better use of the degrees of freedom available and take advantage of the fact that our orbit-based representation automatically enforces the equilibrium condition and physical constraints associated with toroidal geometry; namely, the mirror force, magnetic drifts and the resulting velocity space nonuniformities and boundary losses. As an example, we have demonstrated in Section 8 how the orbit database can be used to model a distribution of alpha particles with a core-localized birth profile that is not a function of the fast alpha’s CoM. This method to design distribution functions was inspired by orbit-following Monte-Carlo simulations, but is much cheaper computationally. We expect that similar strategies can be used to model distribution functions based on various types of experimentally measured data or from results of bounce-averaged Fokker-Planck codes that ignore magnetic drifts [28].

Finally, we demonstrated in Section 8.4 how a distribution of energetic alpha particles with large magnetic drifts can be converted [back and forth](#) between a 4-D mesh in arbitrary coordinates and the orbit-based representation in 3-D CoM space. Besides being an essential part of the coordinate transformation toolbox needed by platforms like IMAS, this procedure served us as a method to verify the numerical accuracy and logical integrity of the workflow. In fact, this procedure enabled us to identify and correct subtle problems that arose in early versions of our implementation of the algorithm. The lessons we learned are documented in this paper.

The methods presented here can serve as a starting point that already provides much of the required functionality required by IMAS. One of the next steps should be the adoption or development of smoothing techniques with the goal to mitigate the effect of discretization noise, while minimizing systematic errors associated with such irreversible data manipulations. This is a prerequisite for performing stability analyses

of resonant modes and for preparing initial conditions for delta- f simulations, which require explicit information about gradients in the phase space density.

We suggest the reader to also follow related and complementary developments in orbit tomography [6, 7, 8, 25]. For instance, those studies demonstrate an alternative way to sample the orbit space and employ methods for [adaptive meshing and reducing binning errors](#).

Acknowledgments

One of the authors (A.B.) acknowledges insightful discussions with Yasushi Todo concerning the handling of the Jacobian factor B_{\parallel}^* , and with Stuart Benjamin and David Pfefferle concerning methods for phase space discretization and mapping techniques. On a broader note, A.B. would like to thank Liu Chen, Zhihong Lin, Fulvio Zonca as well as Sergio Briguglio, Claudio DiTroia and Gregorio Vlad for their support during the early development stages of VisualStart in 2008–2010 for the hybrid code HMGC. Subsequent extensions enabling VisualStart to initialize long-time simulations performed for JT-60U with the code MEGA benefited from support by Kouji Shinohara, Yasushi Todo and Masatoshi Yagi, to whom A.B. would also like to express his gratitude. Finally, A.B. is grateful to Jeronimo Garcia and Shunsuke Ide for encouraging and supporting a fruitful collaboration between JET and QST. The working example used in the present paper was inspired by plasmas produced in recent JET experiments with central 3-ion RF heating, which posed a suitably challenging and practically relevant scenario for testing our numerical algorithms and workflow. We would like to thank all JET contributors for making these interesting experiments possible and allowing us to participate.

This work has been partially carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 and 2019-2020 under Grant Agreement No. 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

Appendix A. Binning operations and Jacobians

In this section, we describe how we evaluated the Klimontovich representation (25) to visualize our results in Section 8. The marker weights w_l in Eq. (31) represent physical particles. Meanwhile, the marker positions and orbits are those of GCs, since it is their equations of motion (21) and (22) that we solve. We must

thus be careful to determine where Littlejohn's GC Jacobian B_{\parallel}^* from Eq. (6) enters. The guiding principle is that the result (i.e., values of f) must be the same irrespective of whether we choose GC coordinates or physical particle coordinates for binning and visualization.

For clarity, we break the procedure up into (i) the computation of histograms h and (ii) their conversion to density functions f . As a concrete example, we consider the set of coordinates (E, λ, R, z) , with volume element $\Delta V_{ijkm} = \Delta E_i \Delta \lambda_j \Delta R_k \Delta z_m$.¹⁰ Depending on whether cell coordinates $\mathbf{Z}_{ijkm} = (E_i, \lambda_j, R_k, z_m)$ are interpreted as positions of GCs or physical particles ("phys"), two kinds of histograms can be computed:

$$h_{\text{gc}}(\mathbf{Z}_{ijkm}) = \sum_{l=1}^{N_{\text{mrk}}} w_l \frac{S(\mathbf{Z}_{\text{gc},l} - \mathbf{Z}_{ijkm})}{\Delta V_{ijkm}}, \quad (\text{A.1a})$$

$$h_{\text{phys}}(\mathbf{Z}_{ijkm}) = \sum_{l=1}^{N_{\text{mrk}}} w_l \frac{S(\mathbf{Z}_{\text{gc},l} - \mathbf{Z}_{ijkm})}{\left[\frac{B_{\parallel}^*}{B}\right]_l \Delta V_{ijkm}}. \quad (\text{A.1b})$$

The markers are located at GC positions $\mathbf{Z}_{\text{gc},l}$, but their weights w_l represent physical particles (cf. Eq. (26)). The shape function $0 \leq S \leq 1$ defines how we map these weights onto our mesh. Within the accuracy limits of our discrete mesh, we may assume that $[B_{\parallel}^*/B]_l \approx [B_{\parallel}^*/B]_{ijkm}$. The phase space density function f is then

$$f = \frac{h_{\text{gc}}}{\mathcal{J}_{\text{gc}}^{(x,v)}} = \frac{h_{\text{gc}}}{\mathcal{J}_{\text{phys}}^{(x,v)} \times B_{\parallel}^*/B} = \frac{h_{\text{phys}}}{\mathcal{J}_{\text{phys}}^{(x,v)}}. \quad (\text{A.2})$$

In the first equality, B_{\parallel}^*/B is contained in the Jacobian $\mathcal{J}_{\text{gc}}^{(x,v)}$ for the conversion from Cartesian to noncanonical GC coordinates (cf. Eq. (6) with $d\mu \rightarrow B^{-1} M dv_{\perp}^2/2$). In the last equality, B_{\parallel}^* is part of the histogram h_{phys} . Substituting Eq. (A.1) into (A.2), the formula for mapping the phase space density function f to a mesh becomes¹¹

$$f(E_i, \lambda_j, R_k, z_m) = \sum_{l=1}^{N_{\text{mrk}}} w_l \times S(\mathbf{Z}_{\text{gc},l} - \mathbf{Z}_{ijkm}) \left[\frac{B_{\parallel}^*}{B} \mathcal{J}_{R_z}^x \mathcal{J}_{E\lambda}^v \right]_l \Delta V_{ijkm}, \quad (\text{A.3})$$

¹⁰In contrast to the canonical 6-D phase space volume elements $\Delta \mathcal{V}_{\text{orb}}$ and $\Delta \mathcal{V}$ of GC orbits and their markers as defined in Section 6, the symbol ΔV appearing here denotes an element in an arbitrary mesh in an arbitrary number of dimensions.

¹¹Conventional PIC codes sample the phase space by loading marker particles in a uniformly randomized manner at the beginning of a simulation ($t = 0$). In that case, the initial marker weights (via their volume elements) carry a factor $B_{\parallel}^*(t=0)/B(t=0)$. When evaluating moments of the distribution function during a simulation, that factor is effectively multiplied by the inverse factor $B(t)/B_{\parallel}^*(t)$ that appears in Eq. (A.3). These two factors cancel approximately and may, thus, be omitted, if one ignores corrections of order $O(\rho_0^2/a^2)$.

which is independent of whether the cell coordinates $\mathbf{Z}_{ijkm} = (E_i, \lambda_j, R_k, z_m)$ are interpreted as those of GCs or physical particles.

Based on Eq. (A.3), we compute density fields in 1-D and 2-D by binning marker weights w_l as follows:

$$n(\psi_{\text{p},i}) = \sum_{l=1}^{N_{\text{mrk}}} \frac{w_l}{\left[\frac{B_{\parallel}^*}{B} \mathcal{J}_{\psi_{\text{p}}}^x(R, z)\right]_l} \frac{S(\mathbf{Z}_{\text{gc},l} - \mathbf{Z}_i)}{\Delta \psi_{\text{p},i}}, \quad (\text{A.4})$$

$$n(R_i, z_j) = \sum_{l=1}^{N_{\text{mrk}}} \frac{w_l}{\left[\frac{B_{\parallel}^*}{B} \mathcal{J}_{R_z}^x(R)\right]_l} \frac{S(\mathbf{Z}_{\text{gc},l} - \mathbf{Z}_{ij})}{\Delta R_i \Delta z_j}. \quad (\text{A.5})$$

The spatially integrated velocity distributions are

$$v(a_i, b_j) = \sum_{l=1}^{N_{\text{mrk}}} \frac{w_l}{\left[\frac{B_{\parallel}^*}{B} \hat{\mathcal{J}}_{ab}^v\right]_l} \frac{S(\mathbf{Z}_{\text{gc},l} - \mathbf{Z}_{ij})}{\Delta a_i \Delta b_j}. \quad (\text{A.6})$$

In Sections 8.2 and 8.3, we used 1st-order shape functions S for binning (linear interpolation), and in Section 8.4 we used 0th-order (binary) binning.

Finally, we present expressions for the Jacobians associated with the coordinates we use. Since the factor B_{\parallel}^*/B has already been singled out in (A.3), we are going to work in physical space. Converting to polar velocity coordinates and integrating over gyrophase, we obtain

$$d\hat{v}_x d\hat{v}_y d\hat{v}_z \times d^3 \mathbf{x} = 2\pi \hat{v}_{\perp} d\hat{v}_{\perp} d\hat{v}_{\parallel} \times d^3 \mathbf{x}. \quad (\text{A.7})$$

The Jacobian $\mathcal{J}_{\psi_{\text{p}}}^x$ for transforming between $d^3 \mathbf{x}$ and the normalized poloidal flux ψ_{p} is obtained numerically. The Jacobian for cylinder coordinates is $\mathcal{J}_{R_z}^x = 2\pi R$. Analytical expressions can also be derived for the Jacobians of all velocity coordinates that we use here. Namely, from the relations

$$\hat{v}_{\perp}^2 = 2\hat{E}\Lambda\hat{B} = 2\hat{E}\cos^2\alpha = 2\hat{E}(1 - \lambda^2), \quad (\text{A.8})$$

$$\hat{v}_{\parallel} = \sqrt{2\hat{E}}\sin\alpha = \sqrt{2\hat{E}}\lambda = \sigma_B \sqrt{2\hat{E}(1 - \Lambda\hat{B})}, \quad (\text{A.9})$$

with $\sigma_B \equiv v_{\parallel}/|v_{\parallel}|$, one readily obtains the Jacobians

$$\mathcal{J}_{ab}^v = 2\pi \left| \begin{array}{cc} \frac{1}{2} \partial_a v_{\perp}^2 & \partial_a v_{\parallel} \\ \frac{1}{2} \partial_b v_{\perp}^2 & \partial_b v_{\parallel} \end{array} \right| \quad (\text{A.10})$$

for locally (in \mathbf{x}) transforming velocity space elements $2\pi v_{\perp} dv_{\perp} dv_{\parallel}$ to other sets (a, b) , such as

$$\hat{\mathcal{J}}_{\mathcal{E}\mu}^v = \pi \hat{B}/|\hat{v}_{\parallel}|, \quad \hat{\mathcal{J}}_{\mathcal{E}\Lambda}^v = 2\pi \hat{E} \hat{B}/|\hat{v}_{\parallel}|, \quad (\text{A.11})$$

$$\hat{\mathcal{J}}_{E\alpha}^v = \hat{\mathcal{J}}_{v_{\perp}, v_{\parallel}}^v = 2\pi \hat{v}_{\perp}, \quad \hat{\mathcal{J}}_{E\lambda}^v = 2\pi \hat{v}. \quad (\text{A.12})$$

Appendix B. Numerical accuracy and data storage

Researchers tend to invest significant effort in the preparation of [their](#) simulation scenarios, balancing accuracy against speed. However, the data on platforms like IMAS are likely to be of variable quality, since some of it originates from automated or semi-automated workflows. The tools used to process those data should hence have a certain degree of tolerance with respect to inaccuracies and be able to detect and handle exceptions. The workflow in VisualStart also contains such sensibility checks and ways to mitigate the effects of inaccuracies. Some of the inaccuracies arising in VisualStart could be avoided with more sophisticated techniques, while others are practically inevitable due to numerical resolution being necessarily finite. This Appendix contains notes concerning some issues that are relevant for the subject of this paper or, at least, for the current implementation of VisualStart.

Appendix B.1. Particle pushing

At present, the particle pushing routine used to compute GC orbits in VisualStart uses the same finite-difference algorithm as the MHD-PIC hybrid code MEGA [40, 41]: a primitive 4th-order Runge-Kutta scheme advances the equations of motion expressed in simple cylinder coordinates. Derivatives are taken through simple finite differences. This choice has been made deliberately in order to have the same level of accuracy as the hybrid code for which VisualStart prepares initial conditions. This allows us to detect possible problems, if any, at an early stage, before launching expensive hybrid simulations. Of course, accuracy can be improved with various techniques, including a Hamiltonian formulation like that used in codes like ORBIT [51, 52, 53] and HAGIS [54, 55], albeit with the complication of having to work with a specialized set of coordinates. The benefits of such techniques become noticeable when following orbits for long periods of time (as in Poincaré analyses). For evaluating a single poloidal transit as we do here, simple techniques often suffice, but due attention must always be paid to regions near singularities and stagnation points like those shown in Figs. 7 and 8.

Appendix B.2. Orbit database

Although one may load GC phase space markers as soon as an orbit contour has been computed, it is advantageous to store orbits in a database and load the markers later as needed. One reason is that the orbits can be recycled when preparing distribution functions for different sets of particles: alpha particles and deuterons are

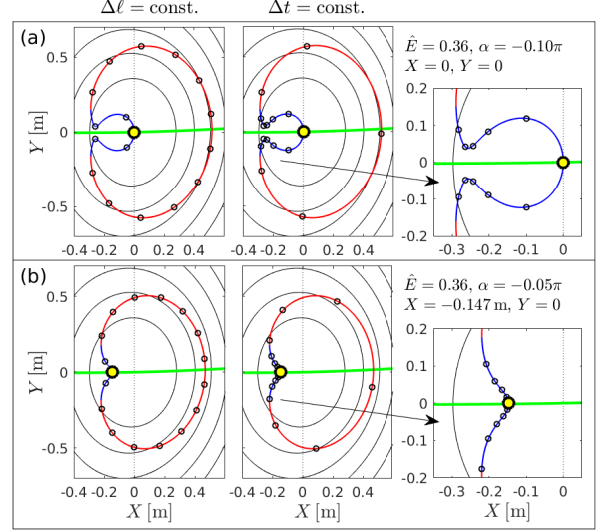


Figure B.19: Sampling of potato orbits. The two examples in (a) and (b) correspond to 2.52 MeV alpha particles with different pitch angles in the scenario of Fig. 4 that is partly based on JET. The coordinate values listed are those at the starting point of an orbit contour, which is indicated by a large yellow circle. The small black circles represent $N_r = 16$ samples spaced by equal length intervals ($\Delta\ell = L_{\text{orb}}/N_r$, left) or equal time intervals ($\Delta t = \tau_{\text{orb}}/N_r$, central and right column).

indistinguishable for identical velocities, and one may have multiple beams as well as RF-heated populations of a certain particle species in the same plasma. These sets of particles may all share the same orbit database.

In order to keep the size of the database manageable, the representation of the orbits must be efficient. VisualStart saves all orbits in discretized form, typically using 32–64 samples per orbit. The user can choose between three options for the distribution of samples: (i) uniform in space, (ii) uniform in time, or (iii) a 50:50 combination of (i) and (ii). We usually choose option (i) for beam-like distributions that consist only of passing particles. Option (iii) is preferred when trapped particles are present in order to ensure good resolution both near and far away from V-type stagnation points as the examples in Fig. B.19 show.

The recorded time arrays must include at least $R(t)$, $Z(t)$, $v_{\parallel}(t)$ and the time t itself. Our databases contain many other arrays that have been stored for various purposes, such as $\zeta(t)$, $v_R(t)$, $v_z(t)$, $v_{\zeta}(t)$ and $\psi_P(t)$. The database file from our low-resolution example in Fig. 6 has a size of 33 Mbyte in NetCDF format. The databases with the parameters in Table 2 occupy about 150 Mbyte each in the mono-energetic cases and 500 Mbyte in the flat-energy cases. These files also contain useful auxiliary information, such as the numerical parameters, poloidal and toroidal transit times, the or-

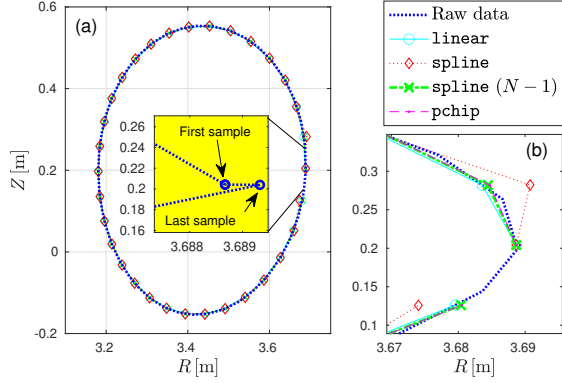


Figure B.20: Example of interpolation errors caused by a small mismatch between the start and end points of an orbit, which can be seen in the zoomed inset in panel (a). Panel (b) shows the results of various interpolation algorithms using the Matlab function `interp1`: linear interpolation, piecewise cubic spline (`spline`), `spline` without the last sample (“ $N-1$ ”), and shape-preserving piecewise cubic interpolation (`pchip`). The magnetic field in this example is based on a JT-60U plasma as used, for instance, in Ref. [17].

bit length, bounce angle, the orbit type, and some measures of the computational accuracy: energy conservation, and spatial mismatch between start- and end-point.

Orbits that have not been completed within a certain upper bound of time steps are labeled as “incomplete”. Before proceeding to Step 4 of our workflow in Fig. 2, we deal with such incomplete orbits and those that have insufficient accuracy. Bad orbits can be corrected if necessary, or simply discarded if sufficient accuracy is difficult to achieve and unnecessary (as may happen for tiny orbits very close to an O-type stagnation point).

Data related to marker weighting are also present; in particular, the volume element sizes ΔV_{orb} . There is an option to undo the factor 1/2 in the volume element given by Eq. (37) for stagnation orbits that are smaller than the cell size and, hence, cannot be double-counted. Their volume elements may then be rescaled in proportion to the orbit’s radial diameter, but the effect is usually negligible since such orbits carry little weight (and one should use a finer mesh if they are of interest).

Interpolation is necessary when converting the raw data from the particle pushing algorithm to the samples used to represent an orbit in the database. Further interpolation is necessary when loading markers on an orbit that has been retrieved from the database. As shown in Fig. B.20, we found that a tiny discontinuity in an orbit contour (mismatch between start and end points) can cause relatively large “ringing” in spline interpolations. Subsequent interpolations in that region — namely, when we load marker particles — can produce corrupt samples that may even lie outside the plasma.

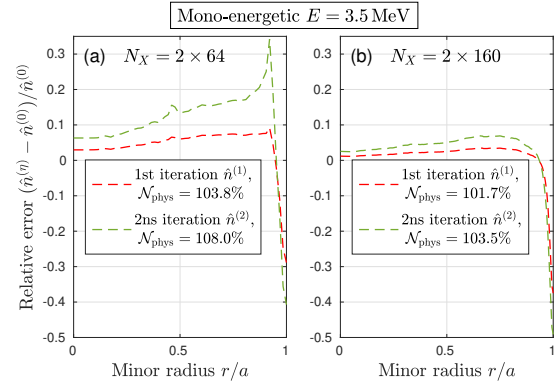


Figure B.21: Convergence test showing the reduction of the relative errors in the radial density profiles $\hat{n}^{(\eta)}(r)$ with iteration index $\eta = 1, 2$. The analysis is equivalent to that in Fig. 17, except that it is performed here for the monotonic-energetic case with $E = 3.5$ MeV. Panel (a) shows the result for the default number of $N_X = 2 \times 64$ radial grid points. In panel (b) it is $N_X = 2 \times 160$. The other parameters are the same as in Table 2, except that the number of markers per orbit N_τ was increased from $48 \frac{L_{\text{orb}}}{L_{\text{bnd}}}$ to the same value of $480 \frac{L_{\text{orb}}}{L_{\text{bnd}}}$ as used for Fig. 17. The relative change in the total number N_{phys} of physical particles is shown in the legends.

When applying the 1-D interpolation function `interp1` of Matlab for processing our orbits (and other potentially “noisy” data), we avoid such problems by using the `pchip` option instead of `spline`.

Appendix B.3. Marker loading

In order to avoid crowding at the midplane, the markers on each orbit are loaded with a random time offset $0 \leq \tau_0 < \tau_{\text{pol}}$ relative to the starting point at the midplane. When using the markers in a simulation, they are also spread (randomly) along the toroidal angle ζ .

Appendix B.4. Convergence test

Figure 17(b) showed indications of relatively large errors that increased systematically with each iteration of our back-and-forth transformation between a 4-D mesh in the coordinates (E, λ, R, z) and the orbit-based representation in 3-D CoM space. In our discussion in Section 8.4, we attributed this problem to inaccuracies in the estimation of the volume elements for orbits sampling the region near the trapped-passing boundary, where some factors exhibit singular behavior.

Further tests show that these errors can be reduced by simply increasing the number of phase space samples via the parameters N_α , N_X and N_τ in Table 2.¹² In

¹²Since kinetic energy is conserved and our 3-D and 4-D distributions both share the same energy mesh, the parameter N_E has no impact on the numerical accuracy in our verification exercise.

Fig. B.21 we demonstrate this for the case where we increased the number of radial grid points N_X . While the flat-energy case with $E = 0 \dots 3.5$ MeV was used in Fig. 17, we have chosen only mono-energetic alpha particles with $E = 3.5$ MeV for the convergence test in Fig. B.21. For an identical mesh, this results in larger values of the relative errors in Fig. 17 since only the particles with the largest magnetic drifts are present.

The comparison between panels (a) and (b) of Fig. B.21 shows a significant reduction of the numerical errors in both the local values of the alpha particle density profile $\hat{n}^{(\alpha)}(r)$ and the total number of particles N_{phys} . As before, the errors accumulate with each iteration, which confirms their systematic character.

Besides the brute-force approach of increasing the number of samples, there may exist other ways to correct or otherwise suppress the inaccuracies associated with singular volume elements. This is left for future work. A preliminary discussion was given in Section 6.2 of the main text.

References

- [1] S.D. Pinches *et al.* Progress in the ITER integrated modelling programme and the ITER scenario databases. *27th IAEA Fusion Energy Conference (FEC 2018)*, 2018. <https://nucleus.iaea.org/sites/fusionportal/Shared%20Documents/FEC%202018/fec2018-preprints/preprint0741.pdf>.
- [2] M. Salewski, M. Nocente, G. Gorini, A.S. Jacobsen, V.G. Kiptily, S.B. Korsholm, F. Leipold, J. Madsen, D. Moseev, S.K. Nielsen, J. Rasmussen, M. Stejner, M. Tardocchi, and JET Contributors. Velocity-space observation regions of high-resolution two-step reaction gamma-ray spectroscopy. *Nucl. Fusion*, 55(9):093029, 2015.
- [3] M. Salewski, M. Nocente, G. Gorini, A.S. Jacobsen, V.G. Kiptily, S.B. Korsholm, F. Leipold, J. Madsen, D. Moseev, S.K. Nielsen, J. Rasmussen, M. Stejner, M. Tardocchi, and JET Contributors. Fast-ion energy resolution by one-step reaction gamma-ray spectrometry. *Nucl. Fusion*, 56(4):040009, 2016.
- [4] M. Salewski, M. Nocente, A.S. Jacobsen, F. Binda, C. Cazaniga, J. Eriksson, B. Geiger, G. Gorini, C. Hellesen, V.G. Kiptily, T. Koskela, S.B. Korsholm, T. Kurki-Suonio, F. Leipold, D. Moseev, S.K. Nielsen, J. Rasmussen, P.A. Schneider, S.E. Sharapov, M. Stejner, M. Tardocchi, and JET Contributors, ASDEX Upgrade Team & EUROfusion MST1 Team. Bayesian integrated data analysis of fast-ion measurements by velocity-space tomography. *Fusion Sci. Technol.*, 74(1-2):23, 2018.
- [5] M. Salewski, M. Nocente, B. Madsen, I. Abramovic, G. Gorini, A.S. Jacobsen, V.G. Kiptily, S.B. Korsholm, D. Moseev, S.K. Nielsen, A.F.L. Poulsen, J. Rasmussen, M. Tardocchi, B. Geiger, J. Eriksson, the JET Contributors, the ASDEX Upgrade Team, and the EUROfusion MST1 Team. Diagnostic of fast-ion energy spectra and densities in magnetized plasmas. *JINST*, 14:C05019, 2019.
- [6] L. Stagner, W.W. Heidbrink, M. Salewski, A.S. Jacobsen, B. Geiger, and The DIII-D and ASDEX Upgrade Teams. Orbit tomography of energetic particle distribution functions. *Nucl. Fusion*, 62(2):026033, 2022.
- [7] L. Stagner. *Inference of the fast-ion distribution function*. PhD thesis, University of California, Irvine, 2018.
- [8] H. Järleblad, L. Stagner, M. Salewski, J. Eriksson, S. Benjamin, B. Madsen, M. Nocente, J. Rasmussen, and B.S. Schmidt. Fast-ion orbit sensitivity of neutron emission spectroscopy diagnostics. *Rev. Sci. Instr.*, 92(4):043526, 2021.
- [9] J. Breslau and D. Liu. Reconstructing fast ion distribution functions from NUBEAM. *TRANSP Users Group Meeting*, May 4, 2017.
- [10] Ph. Lauber *et al.* Energetic particle dynamics induced by off-axis neutral beam injection on ASDEX Upgrade, JT-60SA and ITER. *28th IAEA Fusion Energy Conference (FEC 2020)*, May 10–15, 2021.
- [11] E. Tholerus and M. Fitzgerald. Calculating the fast ion distribution function in constants-of-motion space for MHD instability analysis. *Joint T17-07 fast ion physics group & M15-24 meeting*, May 31, 2017.
- [12] A. Bierwage. VisualStart: GUI-Aided Unified Initialization Tool for Hybrid (MHD + Particle) Simulations. *17th NEXT Meeting*, Kashiwa, Japan, March 15-16, 2012.
- [13] A. Bierwage, C. Di Troia, S. Briguglio, and G. Vlad. Orbit-based representation of equilibrium distribution functions for low-noise initialization of kinetic simulations of toroidal plasmas. *Comp. Phys. Comm.*, 183:1107, 2012.
- [14] A. Bierwage, K. Shinohara, N. Aiba, and Y. Todo. Role of convective amplification of $n = 1$ energetic particle modes for N-NB ion dynamics in JT-60U. *Nucl. Fusion*, 53(7):073007, 2013.
- [15] A. Bierwage, Y. Todo, N. Aiba, and K. Shinohara. Dynamics of low- n shear Alfvén modes driven by energetic N-NB ions in JT-60U. *Nucl. Fusion*, 54(10):104001, 2014.
- [16] A. Bierwage, K. Shinohara, Y. Todo, N. Aiba, M. Ishikawa, G. Matsunaga, M. Takechi, and M. Yagi. Self-consistent long-time simulation of chirping and beating energetic particle modes in JT-60U plasmas. *Nucl. Fusion*, 57(1):016036, 2017.
- [17] A. Bierwage, K. Shinohara, Y. Todo, N. Aiba, M. Ishikawa, G. Matsunaga, M. Takechi, and M. Yagi. Simulations tackle abrupt massive migrations of energetic beam ions in a tokamak plasma. *Nature Comms.*, 9:3282, 2018.
- [18] A. Bierwage and K. Shinohara. Orbit-based analysis of resonant excitations of Alfvén waves in tokamaks. *Phys. Plasmas*, 21(11):112116, 2014.
- [19] A. Bierwage and K. Shinohara. Orbit-based analysis of non-linear energetic ion dynamics in tokamaks. I. Effective mode number profile and resonant frequency tracking. *Phys. Plasmas*, 23(4):042511, 2016.
- [20] A. Bierwage and K. Shinohara. Orbit-based analysis of non-linear energetic ion dynamics in tokamaks. II. Mechanisms for rapid chirping and convective amplification. *Phys. Plasmas*, 23(4):042512, 2016.
- [21] F. Zonca, L. Chen, S. Briguglio, G. Fogaccia, G. Vlad, and X. Wang. Nonlinear dynamics of phase space zonal structures and energetic particle physics in fusion plasmas. *New J. Phys.*, 17(1):013052, 2015.
- [22] L. Chen and F. Zonca. Physics of Alfvén waves and energetic particles in burning plasmas. *Rev. Mod. Phys.*, 88:15008, 2016.
- [23] M.V. Falessi and F. Zonca. Transport theory of phase space zonal structures. *Phys. Plasmas*, 26(2):022305, 2019.
- [24] M. V. Falessi F. Zonca, L. Chen and Z. Qiu. Nonlinear radial envelope evolution equations and energetic particle transport in tokamak plasmas. *J. Phys.: Conf. Ser.*, 1785:012005, 2021. Varenna-Lausanne 2020, 12-16 October, Lausanne, Switzerland.
- [25] S. Benjamin. *Distribution transforms for orbit tomography*. Bachelor Thesis, The Australian National University, 2021. Paper in preparation.

- [26] J.A. Rome and Y.-K.M. Peng. The topology of tokamak orbits. *Nucl. Fusion*, 19(9):1193, 1979.
- [27] Ph. Lauber and Z. Lu. Analytical finite-larmor-radius and finite-orbit-width model for the LIGKA code and its application to KGAM and shear Alfvén physics. *J. Phys.: Conf. Ser.*, 1125:012015, 2018. Varenna-Lausanne 2018, 27-31 August, Varenna, Italy.
- [28] A. Bierwage, M. Toma, and K. Shinohara. MHD and resonant instabilities in JT-60SA during current ramp-up with off-axis N-NB injection. *Plasma Phys. Control. Fusion*, 59(12):125008, 2017.
- [29] R.G. Littlejohn. Variational principles of guiding centre motion.
- [30] D. Moseev and M. Salewski. Bi-Maxwellian, slowing-down, and ring velocity distributions of fast ions in magnetized plasmas. *Phys. Plasmas*, 26(2):020901, 2019.
- [31] M. Nocente *et al.* Generation and observation of fast deuterium ions and fusion-born alpha particles in JET D-³He plasmas with the 3-ion radio-frequency heating scenario. *Nucl. Fusion*, 60(12):124006, 2020.
- [32] Y.O. Kazakov *et al.* Physics and applications of three-ion ICRF scenarios for fusion research. *Phys. Plasmas*, 28(2):020501, 2021.
- [33] A. Bierwage *et al.* Energy-selective confinement of fusion-born alpha particles during internal relaxations in a tokamak plasma. *Submitted*, 2022. Preprint: <http://arxiv.org/abs/2109.03427>.
- [34] M. Weiland, R. Bilato, R. Dux, B. Geiger, A. Lebschy, F. Felici, R. Fischer, D. Rittich, M. van Zeeland, the ASDEX Upgrade Team, and the Eurofusion MST1 Team. RABBIT: Real-time simulation of the NBI fast-ion distribution. *Nucl. Fusion*, 58(8):082032, 2018.
- [35] Ž. Štancar *et al.* Experimental validation of an integrated modelling approach to neutron emission studies at JET. *To appear in Nucl. Fusion*, 2021. <https://doi.org/10.1088/1741-4326/ac3021>.
- [36] F. Porcelli, R. Stankiewicz, W. Kerner, and H.L. Berk. Solution of the drift-kinetic equation for global plasma modes and finite particle orbit widths. *Phys. Plasmas*, 1(3):470, 1994.
- [37] Y. Zhao and R.B. White. Simulation of α -particle redistribution due to sawteeth on the Tokamak Fusion Test Reactor. *Phys. Plasmas*, 4(4):1103, 1997.
- [38] K. Tani, M. Azumi, H. Kishimoto, and S. Tamura. Effect of toroidal field ripple on fast ion behavior in a tokamak. *J. Phys. Soc. Japan*, 50(5):1726, 1981.
- [39] K. Tani and M. Azumi. Simulation studies on alpha-particle-driven current in tokamak reactors. *Nucl. Fusion*, 48(8):085001, 2008.
- [40] Y. Todo and T. Sato. Linear and nonlinear particle-magnetohydrodynamic simulations of the toroidal Alfvén eigenmode. *Phys. Plasmas*, 5(5):1321, 1998.
- [41] Y. Todo, K. Shinohara, M. Takechi, and M. Ishikawa. Nonlocal energetic particle mode in a JT-60U plasma. *Phys. Plasmas*, 12(1):012503, 2005.
- [42] Y. Todo, M.A. Van Zeeland, A. Bierwage, and W.W. Heidbrink. Multi-phase simulation of fast ion profile flattening due to Alfvén eigenmodes in a DIII-D experiment. *Nucl. Fusion*, 54:104012, 2014.
- [43] J. R. Cary and A. J. Brizard. Hamiltonian theory of guiding-center motion. *Rev. Mod. Phys.*, 81:693, 2009.
- [44] M. Deutsch. Angular correlations in nuclear reactions. *Rep. Progr. Phys.*, 14(1):196, 1951.
- [45] A.O. Hanson, R.F. Taschek, and J. H. Williams. Monoergic neutrons from charged particle reactions. *Rev. Mod. Phys.*, 21(4):635, 1949.
- [46] M. Salewski, M. Nocente, B. Madsen, I. Abramovic, M. Fitzgerald, G. Gorini, P.C. Hansen, W.W. Heidbrink, A.S. Jacobsen, T. Jensen, V.G. Kiptily, E.B. Klinkby, S.B. Korsholm, T. Kurki-Suonio, A.W. Larsen, F. Leipold, D. Moseev, S.K. Nielsen, S.D. Pinches, J. Rasmussen, M. Rebai, M. Schneider, A. Shevelev, S. Sipilä, M. Stejner, and M. Tardocchi. Alpha-particle velocity-space diagnostic in ITER. *Nucl. Fusion*, 58(9):096019, 2018.
- [47] S. Sumida, K. Shinohara, R. Ikezoe, M. Ichimura, M. Sakamoto, M. Hirata, and S. Ide. Characteristics of fast ³He ion velocity distribution exciting ion cyclotron emission on JT-60U. *Plasma Phys. Control. Fusion*, 61(2):025014, 2019.
- [48] S. Sumida, K. Shinohara, M. Ichimura, T. Bando, A. Bierwage, and S. Ide. Identification of slow-wave ion cyclotron emission on JT-60U. *Nucl. Fusion*, 61(11):116036, 2021.
- [49] A. Bierwage, Y. Todo, N. Aiba, and K. Shinohara. Sensitivity study for N-NB-driven modes in JT-60U: Boundary, diffusion, gyroaverage, compressibility. *Nucl. Fusion*, 56(10):106009, 2016.
- [50] ITER physics data model documentation (restricted access). https://sharepoint.iter.org/departments/POP/CM/IMDesign/Data%20Model/CI/imas-3.33.0/html_documentation.html. Status: Aug. 2021.
- [51] R.B. White and M.S. Chance. Hamiltonian guiding center drift orbit calculation for plasmas of arbitrary cross section. *Phys. Fluids*, 27(10):2455, 1984.
- [52] Y. Chen, R.B. White, G.-Y. Fu, and R. Nazikian. Numerical study of the nonlinear evolution of toroidicity-induced Alfvén eigenmodes. *Phys. Plasmas*, 6:226, 1999.
- [53] R.B. White, V.N. Duarte, N.N. Gorelenkov, E.D. Fredrickson, M. Podesta, and H.L. Berk. Modeling of chirping toroidal Alfvén eigenmodes in NSTX. *Phys. Plasmas*, 26:092103, 2019.
- [54] S.D. Pinches, L.C. Appel, J. Candy, S.E. Sharapov, H.L. Berk, D. Borba, B.N. Breizman, T.C. Hender, K.I. Hopcraft, G.T.A. Huysmans, and W. Kerner. The HAGIS self-consistent nonlinear wave-particle interaction model. *Comp. Phys. Comm.*, 111:133, 1998.
- [55] S.D. Pinches, H.L. Berk, M.P. Gryaznevich, S.E. Sharapov, and JET-EFDA Contributors. Spectroscopic determination of the internal amplitude of frequency sweeping TAE. *Plasma Phys. Control. Fusion*, 46(7):S47, 2004.