Johanna Ganglbauer, BSc

Application of Symplectic, Interpolated Maps to Fast Ion Motion in Fusion Devices

MASTER'S THESIS

to achieve the university degree of

Diplom-Ingenieurin

Master's degree programme:

Technical Physics

submitted to

Graz University of Technology

Supervisor

Ass.Prof. Dipl.-Ing. Dr.techn. Winfried Kernbichler, Ass.Prof. Dipl.-Ing. Dr.techn.

Co-Supervisor

Dipl.-Ing. Dr.rer.nat. Christopher Albert, Bsc.

Institute of Theoretical and Computational Physics

Graz, June 2020

Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis.

Date	Signature

Abstract

In designing fusion reactors, the alpha particle loss rate is of high importance. Up to now the computation is time consuming because it involves extensive magnetic field evaluations. An idea to speed up those computations is to replace conventional integration of particle trajectories by symplectic mappings, which are constructed by interpolation.

Symplectic mappings have been applied to problems in particle accelerators so far. In this work, they are tested for non-canonical systems including trapped and passing trajectories for the first time.

An analytical model field of a tokamak is used to test the mappings. The system is canonicalized and set up in a way that the mapping can be done in one dimension. Collocation methods are used to approximate the symplectic map by B-spline interpolation functions. Least-squares regression guarantees stability of the mapping despite the complexity of the constructed system.

Using parameters inspired by the ongoing fusion project ITER, the mapping was applied successfully during the alpha slowing down time of about 0.1 s. Particle trajectories slightly differ from those originated from 'conventional' integration, but still reflect the physical reality very well.

In the future, the method should be extended to higher dimensions and applied to non-axisymmetric systems like stellarators or tokamaks with resonant magnetic perturbation (RMP) coils to mitigate edge localized modes (ELM). Besides, different mesh-less interpolation methods should be tested.

Contents

1 Introduction

An important feature of fusion devices is, that the charged fusion products are kept in the chamber so long that they can transfer their energy to the plasma. This is essential to heat the plasma and enable ignition. Also, it avoids damages in the walls.

Current fusion research focuses on deuterium - tritium reactions, where the fusion products are a fast α-particle carrying 3.5 MeV and an extremely fast neutron with 14.1 MeV. As the neutron does not carry charge, it cannot be trapped magnetically and will pass its energy to the walls of the fusion device. In contrast, the α-particle is charged so it can be trapped magnetically and used to heat the plasma. The slowing down time of α-particles in fusion plasmas is of the order of 0.1 seconds.

In the axisymmetric, magnetic field of a tokamak, α-particles are well confined. However, even small perturbations of the magnetic field can cause a significant loss of α-particles (see S. Putvinski, 1999). In non-axisymmetric fusion devices like stellarators or tokamaks with resonant magnetic perturbation (RMP) coils to mitigate edge localized modes (ELM), α-particle losses are an important issue per se.

For that reason, when designing fusion reactors, the α-particle loss rate has to be taken into consideration. An effective way of theoretically anticipating this rate, is to calculate the trajectories of many α-particles with different initial conditions. Then one can find the ratio of α-particles , which stay in the device during the slowing down time, and those, which leave.

Heating the plasma means transferring energy by collisions. As the α-particle density is very low and the α-particle itself is fast and heavy compared to the particles of the main plasma, the collisions do not really affect its motion. Hence, the equations of motion of α-particles are solved neglecting

collisions. These calculations are time consuming, especially because the magnetic field has to be evaluated many times.

A way to circumvent magnetic field evaluations and speed up computation time is to find a generating function via interpolation and use it to replace integration of the equations of motion. Using a suitable generating function preserves also the symplectic structure and was introduced by Warnock, Cai, and Ellison, 2009, who applied the procedure to speed up calculations of trajectories in particle accelerators. The procedure acts on the assumption that the interpolated system is completely symplectic. In this work, also least-squares regression is used to approximate the generating function, which guarantees symplecticity of the system per definition. This further development favors long time stability and makes the idea more interesting for fusion research. Here, it is applied to a simple model field of a tokamak device. This work can be seen as a proof of principle and a motivation for doing further research and calculations.

To understand the application of symplectic mappings on the modeled tokamak system, one has to treat three main issues: Conventional orbit integration of particles in the model field, canonical transformations, and interpolation methods, like B-splines used in this work. These topics are treated in the first three chapters of this work. In the following, these issues are combined and the map construction, evaluation and application is explained in a general form. Then it is applied to a simple pendulum and the tokamak model system.

2 Guiding-Center Trajectories

The foundation of the mapping procedure will be an orbit integrator, which solves the equations of motion dependent on the initial conditions for single α-particles in given magnetic fields. In the first section 2.1 it is justified, why single particle motion can be considered. Then, in section 2.2, the averaged equations of motion, which neglect the gyromotion, are introduced. In the following, the model magnetic field of a tokamak is given (see section 2.3) and it is shown how non-canonical systems, which are characteristic for the averaged equations of motion, are canonicalized (see section 2.4). In section 2.4 some trajectories are calculated (see section 2.5). Finally, in the last section 2.6 trajectories in the subspace of constant toroidal momentum p_ϕ, which are of fundamental importance for the mapping procedure, are investigated.

2.1 Single particle motion

Single particle motion is considered, which means that all interactions between particles are neglected. This can be justified by the short slowing down time, the low α-particle density, and the fact that the alphas are heavier than particles of the main plasma. Also gravity and the fields, which are induced by the moving charge (the particle we follow), are not taken into consideration.

2.2 Equations of motion - guiding-center approach

To calculate the orbits of plasma particles in fusion devices one starts from the Lagrangian of a charged particle in an electromagnetic field. The trajectory of a charged particle in a field can be described by a superposition of the fast gyromotion and the motion of the point $\boldsymbol{R}$, which the particle is gyrating around. This point is called guiding-center. We start from the Lagrangian, which treats the guiding-center movement only:

$$\bar{L}(\boldsymbol{R}, \dot{\boldsymbol{R}}, t) = \frac{m_\alpha (\boldsymbol{b}(\boldsymbol{R}, t) \cdot \dot{\boldsymbol{R}})^2}{2} + \frac{Ze}{c} \boldsymbol{A}(\boldsymbol{R}, t) \cdot \dot{\boldsymbol{R}} - \mu B(\boldsymbol{R}, t) - \frac{Ze}{c} \Phi(\boldsymbol{R}, t), \tag{2.1}$$

given by Helander and Sigmar, 2002. This Lagrangian treats units in the cgs system. $\boldsymbol{b}$ is a unit vector pointing in the direction of the magnetic field, B the magnitude of the magnetic field, Φ the electrostatic potential, $\boldsymbol{A}$ the vector potential, μ the magnetic moment, e the elementary charge, m_α the mass of the alpha particle, Z the charge number, and c the speed of light.

The described superposition is valid as long as the motion of the guiding-center perpendicular to a surface of constant magnetic field is small compared to the gyroradius. To be more precise the ratio:

$$\epsilon = \frac{\rho}{L_B} << 1, \qquad \text{with} \qquad L_B = \frac{1}{|\boldsymbol{\nabla} \ln(B)|}, \tag{2.2}$$

of gyroradius ρ and characteristic length L_B of the magnetic field B has to be small.

If the condition given in Eq. 2.2 is fulfilled and the guiding-center approach (Eq. 2.1) is justified, the magnetic moment μ will be a constant of motion.

Finally, the equations of motion can be deduced by inserting the Lagrangian (Eq. 2.1) into the Euler Lagrange equation. However, the procedure is simplified by first introducing a new variable - the component of the particle velocity parallel to the magnetic field $v_{||} = \boldsymbol{b} \cdot \dot{\boldsymbol{R}}$. The right way to make the

Lagrangian (Eq. 2.1) dependent on the new variable $v_{||}$ is to build the phase space Lagrangian:

$$\bar{L}(\boldsymbol{R}, \dot{\boldsymbol{R}}, v_{||}, t) = \left(m_\alpha v_{||} \boldsymbol{b}(\boldsymbol{R}, t) + \frac{Ze}{c} \boldsymbol{A}(\boldsymbol{R}, t) \right) \cdot \dot{\boldsymbol{R}} - \frac{m_\alpha v_{||}^2}{2} - \mu B(\boldsymbol{R}, t) - \frac{Ze}{c} \Phi(\boldsymbol{R}, t), \tag{2.3}$$

following the procedure described by Littlejohn, 1983.

Now, the equations of motion (see Eq. 2.4) are derived by inserting the phase space Lagrangian (given in Eq. 2.3) into the Euler-Lagrange Equations and introducing the modified vector potential $\boldsymbol{A}^*$, the modified magnetic field $\boldsymbol{B}^*$ and the modified electric field $\boldsymbol{E}^*$:

$$\begin{aligned} \dot{v}_{||} &= - \frac{\boldsymbol{B}^*(\boldsymbol{R}, v_{||}, t) \cdot \left(\mu \boldsymbol{\nabla} B(\boldsymbol{R}, t) - \boldsymbol{E}^*(\boldsymbol{R}, v_{||}, t) \right)}{m_\alpha \boldsymbol{B}^*(\boldsymbol{R}, v_{||}, t) \cdot \boldsymbol{b}(\boldsymbol{R}, t)} \\ \dot{\boldsymbol{R}} &= \frac{v_{||} \boldsymbol{B}^*(\boldsymbol{R}, v_{||}, t) + \frac{c}{Ze} \boldsymbol{b}(\boldsymbol{R}, t) \times \left(\mu \boldsymbol{\nabla} B(\boldsymbol{R}, t) - \boldsymbol{E}^*(\boldsymbol{R}, v_{||}, t) \right)}{\boldsymbol{B}^*(\boldsymbol{R}, v_{||}, t) \cdot \boldsymbol{b}(\boldsymbol{R}, t)} \end{aligned} \tag{2.4}$$

$$\begin{aligned} \boldsymbol{A}^*(\boldsymbol{R}, v_{||}, t) &= \boldsymbol{A}(\boldsymbol{R}, t) + \frac{c m_\alpha v_{||}}{Ze} \boldsymbol{b}(\boldsymbol{R}, t) \\ \boldsymbol{B}^*(\boldsymbol{R}, v_{||}, t) &= \boldsymbol{\nabla} \times \boldsymbol{A}^*(\boldsymbol{R}, v_{||}, t) \\ \boldsymbol{E}^*(\boldsymbol{R}, v_{||}, t) &= \frac{Ze}{c} \left(\frac{\partial \boldsymbol{A}^*(\boldsymbol{R}, v_{||}, t)}{\partial t} - \boldsymbol{\nabla} \Phi(\boldsymbol{R}, t) \right). \end{aligned} \tag{2.5}$$

2.2.1 Normalization of the equations of motion

To make the representation of orbit variables clear and comparable the equations of motion 2.4 are normalized. In the following, normalized variables are indicated by a hat $\hat{}$. One wants to replace the parallel velocity $v_{||}$ by $\hat{v}_{||} = \frac{v_{||}}{v}$, the magnetic moment μ by $\hat{\mu} = \frac{\mu}{m_\alpha v^2}$ and the differential time element dt by $d\hat{t} = v \cdot dt$. To make this variable transformation possible, the

first equation of motion 2.4 has to be divided by v^2 and the second one by v resulting in:

$$\begin{aligned}\frac{d\hat{v}_{||}}{d\hat{t}} &= -\frac{\boldsymbol{B}^*(\boldsymbol{R},\hat{v}_{||},\hat{t})\cdot\left(\hat{\mu}\boldsymbol{\nabla}B(\boldsymbol{R},\hat{t})-\frac{\boldsymbol{E}^*(\boldsymbol{R},\hat{v}_{||},\hat{t})}{m_\alpha v^2}\right)}{\boldsymbol{B}^*(\boldsymbol{R},\hat{v}_{||},\hat{t})\cdot\boldsymbol{b}(\boldsymbol{R},\hat{t})} \\ \frac{d\boldsymbol{R}}{d\hat{t}} &= \frac{\hat{v}_{||}\boldsymbol{B}^*(\boldsymbol{R},\hat{v_{||}},\hat{t})+\frac{cm_\alpha v}{Ze}\boldsymbol{b}(\boldsymbol{R},\hat{t})\times\left(\hat{\mu}\boldsymbol{\nabla}B(\boldsymbol{R},\hat{t})-\frac{\boldsymbol{E}^*(\boldsymbol{R},\hat{v}_{||},\hat{t})}{m_\alpha v^2}\right)}{\boldsymbol{B}^*(\boldsymbol{R},\hat{v}_{||},\hat{t})\cdot\boldsymbol{b}(\boldsymbol{R},\hat{t})},\end{aligned} \tag{2.6}$$

with

$$\boldsymbol{B}^*(\boldsymbol{R},\hat{v}_{||},\hat{t}) = \boldsymbol{B}(\boldsymbol{R},\hat{t})+\frac{\hat{v}_{||}cm_\alpha v}{Ze}\boldsymbol{\nabla}\times\boldsymbol{b}(\boldsymbol{R},\hat{t}). \tag{2.7}$$

Due to the normalization, energy and momentum are dimensionless quantities ($\hat{p}_i = \frac{p}{m_\alpha v}$, and $\hat{H}_i = \frac{H}{m_\alpha v^2}$ respectively). As a result, a fast alpha fusion particle with 3.5 MeV carries an energy of $\hat{H} = 0.5$.

2.2.2 Equations of motion in curvilinear coordinates

The magnetic field of fusion devices (e. g. stellarator) will be given mostly in curvilinear coordinates. As the coordinate system does not have to be orthogonal in that case, covariant (lower index) and contravariant (upper index) vector components of the magnetic field are not the same and a distinction has to be made (D'haeseleer et al., 1991). In general, a covariant component A_v belongs to a contravarient unit vector $\boldsymbol{e}^v$, and a contravarient component A^v to a covarient unit vector $\boldsymbol{e}_v$. As the scalar product is defined by $\boldsymbol{AB} = A_j\boldsymbol{e}^j B^j\boldsymbol{e}_j = A_j B^j = A^j B_j$, the quantities in Eq. 2.6 and Eq. 2.7 are

expressed by the form:

$$\begin{aligned}
\boldsymbol{\nabla} B &= \frac{\partial B}{\partial x^{\nu}} \boldsymbol{e}^{\nu}, \\
\boldsymbol{b} &= \frac{B_{\nu}}{B} \boldsymbol{e}^{\nu}, \\
\boldsymbol{b} \times \boldsymbol{\nabla} B &= \frac{1}{\sqrt{g}} \epsilon^{ij\nu} b_j \frac{\partial B}{\partial u^i} \boldsymbol{e}_{\nu}, \\
\boldsymbol{B}^* &= B^{\nu} \boldsymbol{e}_{\nu} + \frac{\hat{v}_{||} c m_{\alpha}}{Z e \sqrt{g}} \epsilon^{ij\nu} \frac{\partial b_j}{\partial x^i} \boldsymbol{e}_{\nu},
\end{aligned}$$

where $\sqrt{g}$ is the root of the metric determinant being equal to the root of the Jacobian.

2.3 Model magnetic field: tokamak

The magnetic field of a tokamak can be approximated by an analytical model, which provides a test case for orbit integration. The field is axisymmetric (independent of the toroidal angle ϕ) and the flux surfaces, which are aligned parallel to the magnetic field, are nested tori.

The source of the analytic model field is given by Albert, Kasilov, and Kernbichler, 2020. It is defined by its covariant components of the vector potential $\boldsymbol{A}(r, \theta)$, its covariant components of the unit magnetic field vector $\boldsymbol{b}(r, \theta)$, its

magnitude $B(r,\theta)$ and the square root of the metric tensor $\sqrt{g}(r,\theta)$:

$$\begin{aligned} A_r &= 0 & b_r &= 0 \\ A_\theta &= B_0 \left(\frac{r^2}{2} - \frac{r^3}{3R_0}\right)\cos(\theta) & b_\theta &= \iota_0 \left(1 - \frac{r^2}{a^2}\right)\frac{r^2}{R_0} \\ A_\phi &= -\iota_0 B_0 \left(\frac{r^2}{2} - \frac{r^4}{4a^2}\right) & b_\phi &= R_0 + r\cos(\theta). \end{aligned} \tag{2.8}$$

$$\begin{aligned} B &= B_0 \left(1 - \frac{r}{R_0}\cos(\theta)\right), \\ \sqrt{g} &= r(R_0 + r\cos(\theta)) + \frac{{\iota_0}^2 (1 - \frac{r^2}{a^2})^2 r^3}{R_0 - r\cos(\theta).} \end{aligned}$$

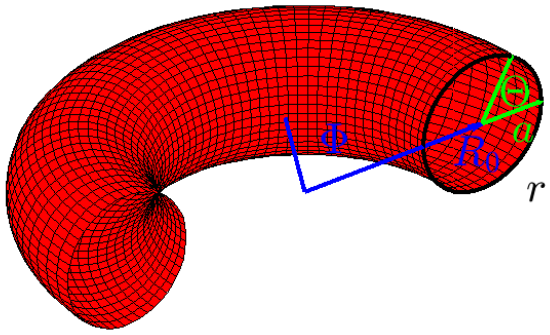


Figure 2.1: Outermost flux surface in red: Model field parameters R_0 and a as well as radial coordinate r, poloidal coordinate θ and toroidal coordinate ϕ

The meaning of coordinates r, ϕ and θ, as well as the major device radius R_0 and the plasma minor radius a are shown in Fig. 2.1. B_0 represents

the flux surface average magnetic field magnitude and ι_0 is the rotational transform at the center of the poloidal circle marked in black. In general, the rotational transform defines the ratio of poloidal turns of field lines per toroidal turn.

In this context the values of B_0, R_0, and a are inspired by the fusion device ITER, and set to $B_0 = 5 \cdot 10^4\ Gs$, $R_0 = 600\ cm$, $a = 200\ cm$ and $\iota_0 = 1$

In Fig. 2.2 the magnetic field magnitude evaluated by Eq. 2.8 is shown. Naturally, it is stronger at the inner part of the torus (where the radius corresponds to $R_0 - a$), and weaker at the outer part (where the radius corresponds to $R_0 + a$). As the field component B_r is zero, every torus within the torus given in Fig. 2.1 in red, represents a flux surface aligned parallel to the magnetic field. Black circles in Fig. 2.2 represent poloidal cuts of some of these flux surfaces.

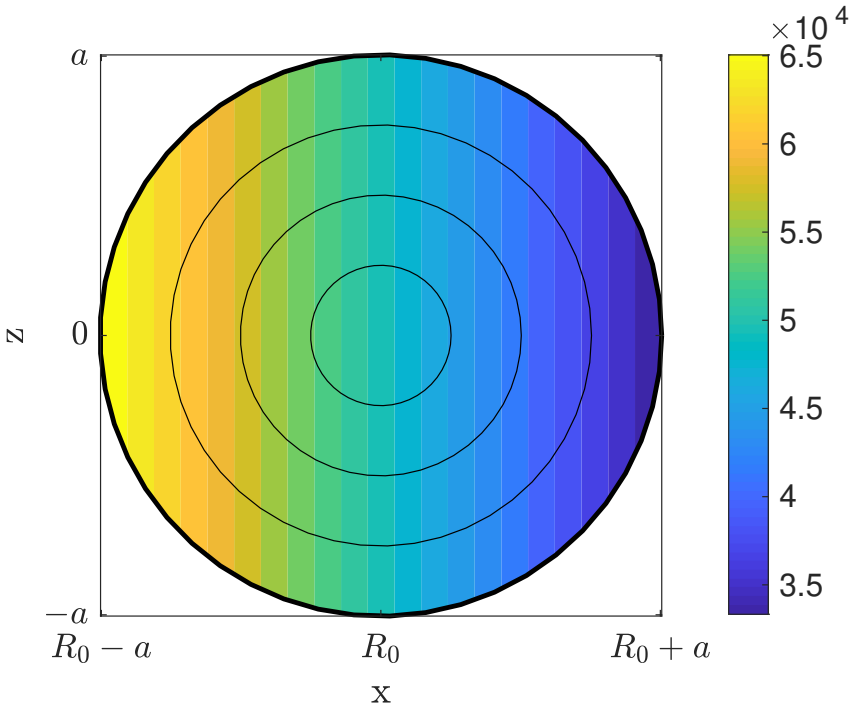


Figure 2.2: Magnetic field magnitude $B(r, \theta)$ given in units Gs in the poloidal plane: black circles represent poloidal cuts of flux surfaces

In order to evaluate the equations of motion (Eq. 2.6), one needs not only

the covariant components of the unit magnetic field vector $\boldsymbol{b}$, but also the contravariant components of the magnetic field vector given by:

$$\boldsymbol{B}(r,\theta) = \boldsymbol{\nabla} \times \boldsymbol{A}(r,\theta) = \frac{\epsilon^{ij\nu}}{\sqrt{g}(r,\theta)} \frac{\partial A_j(r,\theta)}{\partial x^i} \boldsymbol{e}_\nu, \tag{2.9}$$

leading to:

$$\begin{aligned} B^r &= 0 \\ B^\theta &= \frac{-\iota_0 B_0 (r - \frac{r^3}{a^2})}{\sqrt{g}(r,\theta)} \\ B^\phi &= \frac{-\iota_0 B_0 (r - \frac{r^2}{R_0}) \cos(\theta)}{\sqrt{g}(r,\theta)}. \end{aligned} \tag{2.10}$$

2.4 Canonicalization of non-canonical systems

Let us have a look back on the equations of motion of the guiding-center of a charged particle in an electromagnetic field (Eq. 2.6). By averaging over the gyro motion, six-dimensional phase space was reduced by two dimensions, resulting in a Lagrangian (see Eq. 2.3) dependent on 3 space-like variables $\boldsymbol{R}$ and one momentum like variable $\hat{v}_{||}$. Due to this inequality the variables are named non-canonical.

Basically, the coordinates can be canonicalized by building the conjugate (normalized) momentum:

$$\begin{aligned} \hat{\boldsymbol{P}} = \frac{\partial L(\boldsymbol{R}, \dot{\boldsymbol{R}}, \hat{v}_{||}, \hat{t}}{\partial \dot{\boldsymbol{R}}} &= \frac{Ze}{cm_\alpha v} \boldsymbol{A}(\boldsymbol{R}, \hat{t}) + \hat{v}_{||} \boldsymbol{b}(\boldsymbol{R}, \hat{t}) \\ &= \frac{Ze}{cm_\alpha v} \boldsymbol{A}^*(\boldsymbol{R}, \hat{v}_{||}, \hat{t}). \end{aligned} \tag{2.11}$$

Then, a component $\hat{P}_i$ of the conjugate momentum vector is used to express $\hat{v}_{||}$ and one turns back to Hamiltonian formulation:

$$\begin{aligned} \hat{H}(\boldsymbol{R}, \hat{P}_i, \hat{t}) &= \dot{\boldsymbol{R}} \hat{\boldsymbol{P}} - L(\boldsymbol{R}, \dot{\boldsymbol{R}}, \hat{v}_{||}(\hat{P}_i), \hat{t}) \\ &= \frac{m_\alpha \hat{v}_{||}^2(\hat{P}_i)}{2} + \hat{\mu} B(\boldsymbol{R}, \hat{t}) + \frac{Ze}{cm_\alpha v} \Phi(\boldsymbol{R}, \hat{t}). \end{aligned} \tag{2.12}$$

At this point we have three space-like variables $\boldsymbol{R}$ and three momentum-like variables $\hat{\boldsymbol{P}}$. The system is canonical, but the information of two components of the moment $\hat{\boldsymbol{P}}$ is redundant. A coordinate transformation leading to $\hat{P}_1 = 0$ everywhere because of $A_1(\boldsymbol{R}, \hat{t}) = 0$ and $b_1(\boldsymbol{R}, \hat{t}) = 0$ can be found, so that one component of the momentum is lost. Now, the system is described by two space like -, and two momentum like variables. One momentum like variable is used to implicitly determine the third space coordinate and the other one to evaluate $\hat{v}_{||}$. Canonicalization of general systems is described by Albert, Kasilov, and Kernbichler, 2020.

2.4.1 Canonicalization using the model magnetic field

The given model field of a tokamak (see section. 2.3) is already canonicalized, because $A_r = 0$ and $b_r = 0$. Turning back to the model field, the conjugate coordinate pairs $(\phi, \hat{p}_\phi)$ and $(\theta, \hat{p}_\theta)$ completely define the system. For example, the ϕ component of Eq. 2.11 can be used to determine $\hat{v}_{||}$:

$$\hat{v}_{||} = \frac{\hat{p}_\phi - \frac{Ze}{cm_\alpha v} A_\phi(r, \theta)}{b_\phi(r, \theta)}. \tag{2.13}$$

The θ component of Eq. 2.11 then can be used to implicitly define the coordinate r:

$$\frac{Ze}{cm_\alpha v} A_\theta(r, \theta) + \hat{v}_{||} b_\theta(r, \theta) - \hat{p}_\theta = 0. \tag{2.14}$$

2.5 α-guiding center trajectories in the model magnetic field

To calculate trajectories for the given model field, the equations of motion 2.6 have to be solved. The electrostatic potential Φ is set to zero and the time-independent magnetic field is described by the model introduced in the previous section 2.3. An adaptive step size Runge-Kutta scheme, more precisely the ordinary differential equation solver *ode45*, impelemented in Matlab, is used to integrate the equations of motion for given initial conditions.

2.5.1 Initial conditions

Primarily, constants are given by the speed of light $c \sim 3 \cdot 10^{10}\ \frac{cm}{s}$, the α-particle's mass $m_\alpha \sim 6.664 \cdot 10^{-24}\ g$, its charge $Ze \sim 2 \cdot 4.8032 \cdot 10^{-10}\ statC$, and the thermal velocity determined by its energy $E = 3.5\ MeV$, which corresponds to

$$v = \sqrt{\frac{2E}{m_\alpha}} = \sqrt{\frac{2 \cdot 3.5\ MeV \cdot 1.6 \cdot 10^{-6}\ \frac{ergs}{MeV}}{6.664 \cdot 10^{-24}\ g}} \sim 1.2964 \cdot 10^9\ \frac{cm}{s}. \tag{2.15}$$

Initial conditions r, θ, ϕ and $\hat{v}_{||}$ completely define the equations. Once initial conditions are chosen, the normalized magnetic moment can be calculated by:

$$\hat{\mu} = \frac{\hat{v}_\perp^2}{2B(r,\theta)} = \frac{(1 - \hat{v}_{||}^2)}{2B(r,\theta)}. \tag{2.16}$$

The orbit shape depends on both the initial position and the normalized parallel velocity $\hat{v}_{||}$ of the α-particle. In case of a high $\hat{v}_{||}$ we expect the guiding-center orbit to move along the magnetic field lines, resulting in a passing orbit. On the other hand a small $\hat{v}_{||}$ means high perpendicular velocity, so that the mirror effect plays an important role and the particle will be reflected, resulting in a trapped orbit. In order to investigate both cases, the alpha particle is placed somewhere inside the tokamak ($r = 80\ cm, \theta = \frac{2\pi}{3}, \phi = 0$) and the orbit is computed for both cases $\hat{v}_{||} = 0.9$ and $\hat{v}_{||} = 0.1$.

The integration time step is chosen by the adaptive step size Runge-Kutta algorithm using a relative tolerance of 10^{-12}. The trajectories are integrated up to $\hat{t} = 10^5\ cm$. Length units in time result from normalization: $\hat{t} = vt$.

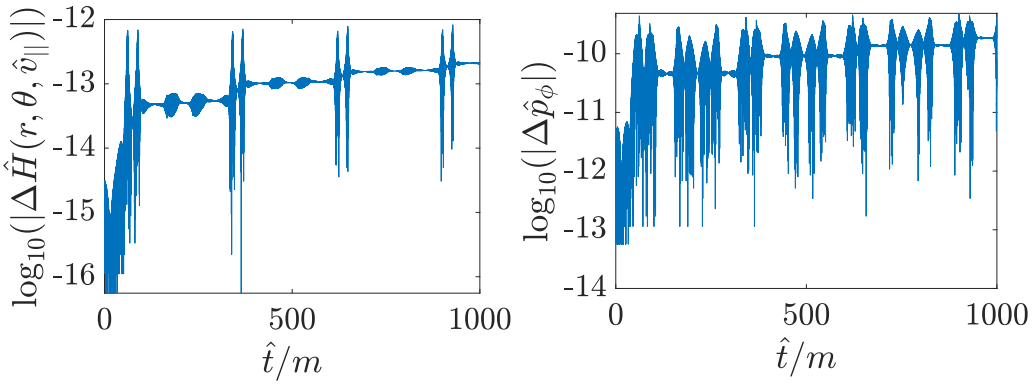
2.5.2 Results

A reliable way to check that orbit integration works correctly is to observe constants of motion. If those are conserved, the integration is correct. Normalized energy $\hat{H}(r, \theta, \hat{v}_{||})$ given by Eq. 2.12 is a constant of motion. Besides,

the toroidal momentum p_ϕ has to be conserved, because it is a cyclic variable. The field independence from ϕ leads to a vanishing derivative of the Lagrangian (given by Eq. 2.3) with respect to ϕ. As a result, the time derivative of the conjugate moment,

$$\frac{d}{dt}p_\phi = \frac{d}{dt}\frac{\partial \bar{L}(r,\dot{r},\theta,\dot{\theta},\dot{\phi})}{\partial \dot{\phi}} = 0,$$

is zero. This means that p_ϕ, and $\hat{p}_\phi$ are conserved. Both, conservation of energy and of the toroidal momentum $\hat{p}_\phi$ are shown in Fig. 2.3 and Fig. 2.4 for the trapped and the passing case respectively. Fluctuations result from inaccuracy of the ordinary differential equation solver *ode45*.

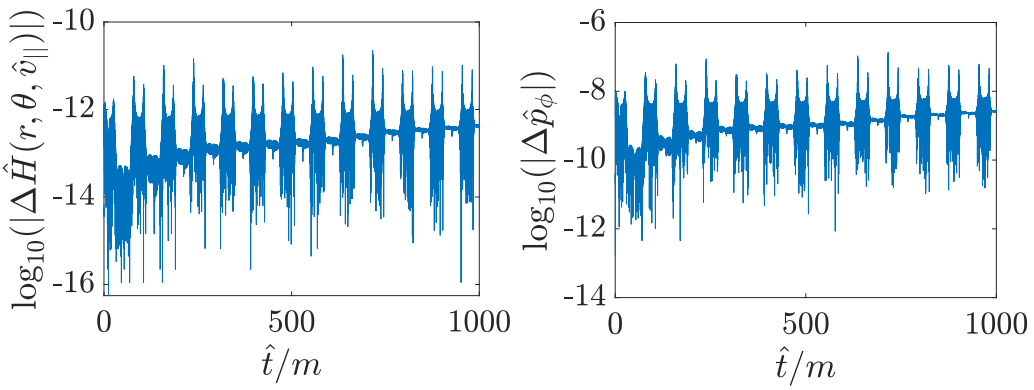


(a) $\log_{10}(|\hat{H}(r,\theta,\hat{v}_{||}) - \hat{H}_\alpha(t=0)|)$

(b) $\log_{10}(|\hat{p}_\phi(t) - p_\phi(t=0)|)$

Figure 2.3: Constants of motion: conservation of energy (a) and normalized toroidal momentum $\hat{p}_\phi$ (b) for initial conditions: $r = 80\ cm, \theta = \frac{2\pi}{3}, \phi = 0, \hat{v}_{||} = 0.9$ (passing orbit).

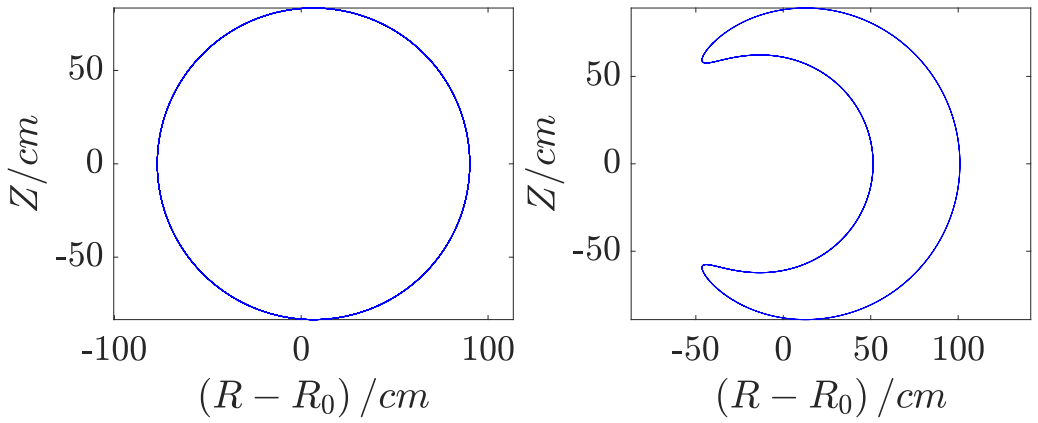
The coordinates r, θ and ϕ of the alpha particle are given at every integration step. For a better imagination of the particle movement, it is plotted for the trapped and passing case in three dimensional cartesian space. In Fig. 2.6a and Fig. 2.6b one can easily see that in the passing case the particle follows the field line without changing direction, while the trapped particle is reflected many times. In addition, the projection of the movement onto a poloidal cut of the torus (circle at $\phi = 0$) is plotted in Fig. 2.5.



(a) $\log_{10}(|\hat{H}(r,\theta,\hat{v}_{||}) - \hat{H}_{\alpha}(t=0)|)$

(b) $\log_{10}(|\hat{p}_{\phi}(t) - p_{\phi}(t=0)|)$

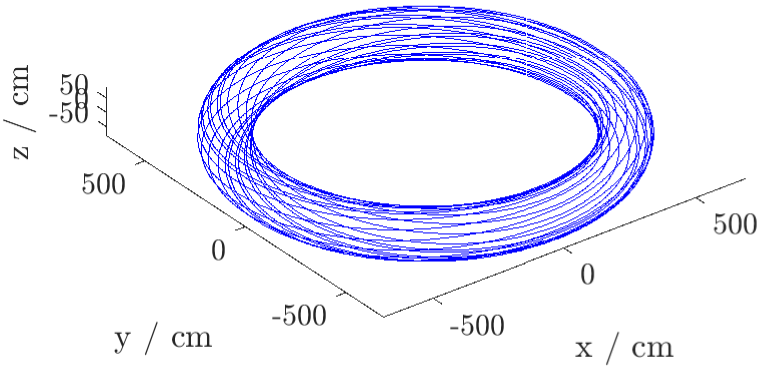
Figure 2.4: Constants of motion: conservation of energy (a) and normalized toroidal momentum $\hat{p}_{\phi}$ (b) for initial conditions: $r = 80\ cm$, $\theta = \frac{2\pi}{3}$,$\phi = 0$, $\hat{v}_{||} = 0.1$ (trapped orbit).



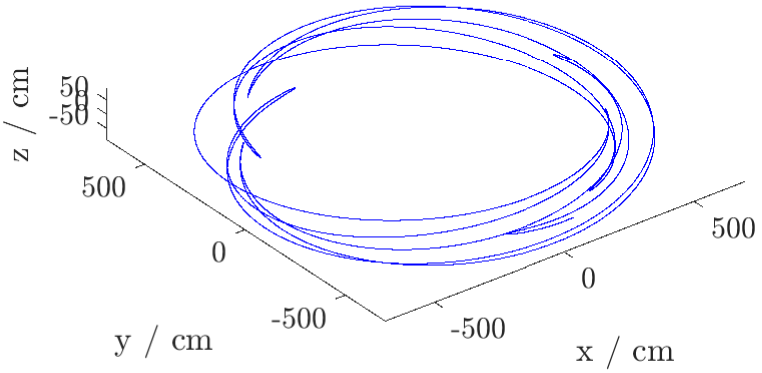
(a) Passing orbit

(b) Trapped orbit

Figure 2.5: Passing (a) and trapped (b) α-particle trajectories in poloidal plane for initial conditions: $r = 80\ cm$, $\theta = \frac{2\pi}{3}$, $\hat{v}_{||} = 0.9$ (a) and $\hat{v}_{||} = 0.1$ (b) respectively.



(a) Initial conditions ($r = 80\ cm, \theta = \frac{2\pi}{3}, \hat{v}_{||} = 0.9$) (passing orbit)



(b) Initial conditions ($r = 80\ cm, \theta = \frac{2\pi}{3}, \hat{v}_{||} = 0.1$) (trapped orbit)

Figure 2.6: α-particles trajectory in three dimensions

2.6 Subspace trajectories in the model field

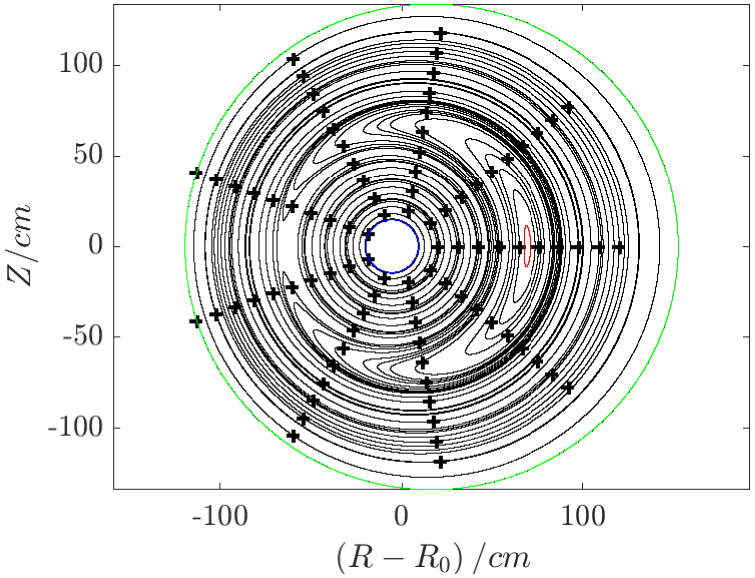
Here, particle trajectories in the subspace of constant normalized toroidal momentum $\hat{p}_\phi = -400$ and constant normalized magnetic moment $\hat{\mu} = 5 \cdot 10^{-6}\ Gs^{-1}$ are examined. Remaining parameters are chosen as above (see section 2.5).

Due to the toroidal invariance of the model field, poloidal coordinates r and θ are sufficient to describe the particle motion. Analyzing trajectories of the same constant toroidal momentum p_ϕ in the poloidal plane allows to reduce the canonicalized system to two dimensions $(\theta, \hat{p}_\theta)$. In this case, r is implicitly given by Eq. 2.14 and $\hat{v}_{||}$ is defined by Eq. 2.13.

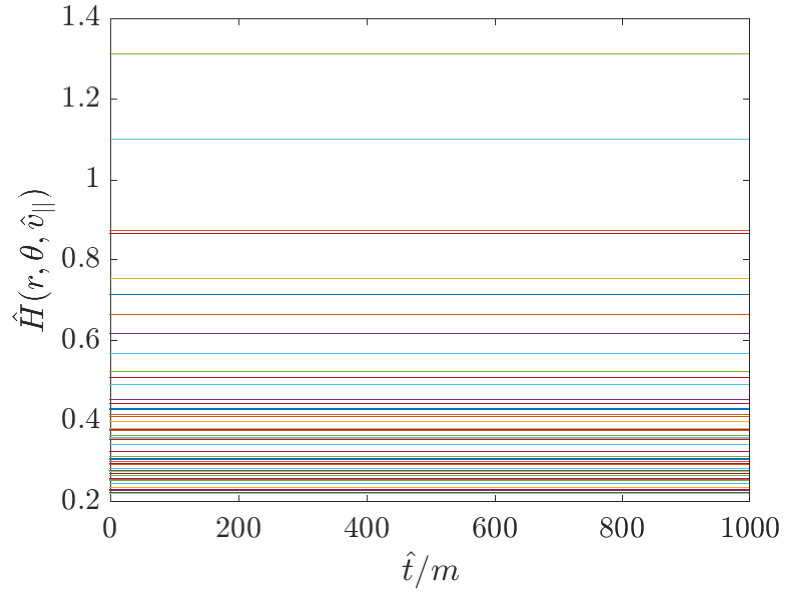
The magnetic moment $\hat{\mu}$ is set to a constant value for all considered particles, because at a later point it will be necessary that all particles used for a map construction correspond to the same Hamiltonian function (given by Eq. 2.12).

To get an overview of the orbits in the described subspace, the equations of motion Eq. 2.6 are solved for initial conditions covering parts of the poloidal plane of the magnetic field (see Fig 2.2) in a rough mesh with r going from $r = 0.1\ a$ to $r = 0.6\ a$ and θ from zero to 2π with 10 steps in each direction. Here, a corresponds to the plasma minor radius and $\hat{v}_{||}$ is given by Eq. 2.13 for each coordinate pair. The trajectories are integrated up to $\hat{t} = 10^3\ m$. Initial conditions and the resulting particle trajectories are illustrated in Fig. 2.7a. Analyzing these trajectories one notes that particles in the inner and outer region are passing. However, between these inner and outer circle regions, particles are trapped.

Particle energies vary, depending on their initial conditions, but they are conserved, which is shown in Fig. 2.7b. The highest particle energy of $\hat{H} \approx 1.3$ corresponds to the outermost passing trajectory (see Fig. 2.7a green). Expanding the initial conditions to greater r values would involve particles with energies exceeding $\hat{H} = 1.3$, which is already 2.6 times the energy of a fusion alpha. The expansion would additionally demand time refinement. In addition, at $r = 0$ the equations of motion are not solvable.



(a) α-particle trajectories: initial conditions are marked by +, the fastest trapped orbit is depicted in red, the fastest passing in blue and the green line corresponds to the trajectory with the highest particle energy.



(b) Energy conservation: the normalised energy of an α-particle carrying 3.5 MeV corresponds to $\hat{H} = 0.5$.

Figure 2.7: α-particle trajectories (a) and energy conservation (b) in the tokamak model field for constant $\hat{p}_\phi = -400$ and $\hat{\mu} = 5 \cdot 10^{-6}\ Gs^{-1}$

2.6.1 Bounce time

Bounce times of trapped and passing particles in the given model field can be calculated analytically in the large aspect ration limit ($R \gg r$) (see Albert, 2017). Here ($r = 200\ cm, R_0 = 600\ cm$)), they will be determined numerically.

In the passing case a bounce is defined by completing a poloidal circle. Assuming the particle started at a poloidal angle $\theta = \theta_0$, the bounce will be completed at $\theta = \theta_0 + 2\pi$. In the trapped case, the particle trajectory is called banana orbit, because of its shape. A bounce is defined by the way from one banana tip to the other and back. Whenever a particle is at the banana tip and changes its direction, the parallel velocity $\hat{v}_{||}$ changes sign.

The bounce time of the fastest trapped and passing orbit is estimated by integrating trajectories with initial conditions covering the area of interest, detecting two sign changes in $\hat{v}_{||}$ or angle extensions of $\theta - \theta_0 > 2\pi$, measuring the corresponding time, and finding the minimum.

In particular, the fastest trapped and passing orbits were determined for the case described above with initial conditions covering the area from $r = 0.1\ a$ to $r = 0.6\ a$ and $\theta = 0$ to $\theta = 2\pi$ with 100 steps in each direction. The location of the resulting fastest trapped and passing orbits is depicted in Fig 2.7a in red and blue respectively. The corresponding fastest normalized bounce times are $\hat{\tau}_{passing} = 54.65\ m$ and $\hat{\tau}_{trapped} = 254.7\ m$.

3 Canonical Transformations

The background of symplectic mappings are canonical transformations, which connect final coordinates $\boldsymbol{Q}$ and momenta $\boldsymbol{P}$ at time t with initial coordinates $\boldsymbol{q}$ and momenta $\boldsymbol{p}$ at $t = 0$. Integrating orbits for a long time can be replaced by carrying out this transformation many times.

Canonical in the context of transformations means that the Hamiltonian nature of the mapped system is not violated. Therefore, initial and final coordinates both have to fulfill Hamilton's canonical equations of motion. In the first section 3.1 one will see that such transformations can be described by generating functions.

In the following section 3.2 a short explanation of symplecticity is given, and it is shown that one dimensional canonical transformations, which are not explicitly time dependent are always symplectic and conserve volume in phase space.

3.1 Generating functions

The theory of canonical transformations is well described by Goldstein, 1963 and summarized here. One applies the principle of least action to both systems $(\boldsymbol{q}, \boldsymbol{p})$ and $(\boldsymbol{Q}, \boldsymbol{P})$. Lagrangians are constructed via Legendre transformations of the corresponding Hamiltonian functions $H(\boldsymbol{q}, \boldsymbol{p}, t)$, and $K(\boldsymbol{Q}, \boldsymbol{P}, t)$ respectively.

The principle of least action for both systems reads:

$$\begin{aligned} \delta \int_{t_1}^{t_2} (\dot{\boldsymbol{q}} \cdot \boldsymbol{p} - H(\boldsymbol{q}, \boldsymbol{p}, t))\, dt = 0, \\ \delta \int_{t_1}^{t_2} \left(\dot{\boldsymbol{Q}} \cdot \boldsymbol{P} - K(\boldsymbol{Q}, \boldsymbol{P}, t)\right) dt = 0, \end{aligned} \tag{3.1}$$

where the variation δ of an arbitrary function f, dependent on n-dimensional vectors $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$, and $\boldsymbol{p}$ is given by:

$$\delta f(\boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{p}, t) = \sum_{i=1}^{n} \frac{\partial f}{\partial q_i} \delta q_i + \frac{\partial f}{\partial \dot{q}_i} \delta \dot{q}_i + \frac{\partial f}{\partial p_i} \delta p_i.$$

This variation examines the changes of trajectories in the extended phase space $(\boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{p})$ between two fixed time points t_1 and t_2. As all trajectories have to go through the same start point $(\boldsymbol{q}(t_1), \dot{\boldsymbol{q}}(t_1), \boldsymbol{p}(t_1)) = (\boldsymbol{q_1}, \dot{\boldsymbol{q}}_1, \boldsymbol{p_1})$ and through the same end point $(\boldsymbol{q}(t_2), \dot{\boldsymbol{q}}(t_2), \boldsymbol{p}(t_2)) = (\boldsymbol{q_2}, \dot{\boldsymbol{q}}_2, \boldsymbol{p_2})$, variations of independent variables δq_i, $\delta \dot{q}_i$ and δp_i vanish at the limit points. Using this fact and partial integration shows that both integrals in Eq. 3.1 are zero, supposed that $H(\boldsymbol{q}, \boldsymbol{p}, t)$ and $K(\boldsymbol{Q}, \boldsymbol{P}, t)$ fulfill Hamilton's canonical equations of motion.

Adding the time derivative of a function G to the integrands is possible without violating the principle of least action (see Eq. 3.1), because naturally the variation of the function itself has to vanish at the limit points:

$$\delta \int_{t_1}^{t_2} \frac{dG(\ldots, t)}{dt} dt = \delta(G(\ldots, t_2)) - G(\ldots, t_1)) = 0.$$

This fact is used to connect the integrands given in Eq. 3.1:

$$\dot{\boldsymbol{Q}} \cdot \boldsymbol{P} - K(\boldsymbol{Q}, \boldsymbol{P}, t) + \frac{dG(\ldots, t)}{dt} = \dot{\boldsymbol{q}} \cdot \boldsymbol{p} - H(\boldsymbol{q}, \boldsymbol{p}, t). \tag{3.2}$$

This convenient connection already describes the desired transformation from initial coordinates $(\boldsymbol{q}, \boldsymbol{p})$ to final coordinates $(\boldsymbol{Q}, \boldsymbol{P})$. As the variational principle holds for both sets of coordinates and the canonical equations are

fulfilled respectively, the transformation is called canonical. In the following, G is called generating function.

In order to describe the canonical transformation from $2n$ initial to $2n$ final coordinates, $4n$ variables are needed. n of them can be eliminated because $\boldsymbol{q}$ and $\boldsymbol{p}$ are connected by Hamilton's equations. So are $\boldsymbol{Q}$ and $\boldsymbol{P}$. Thus the generating function has to depend on one coordinate vector of the initial phase space and one of the final phase space (in total $2n$ variables) only. This provides four possibilities of choosing a generating function:

$$G_1(\boldsymbol{q}, \boldsymbol{Q}, t), \qquad G_2(\boldsymbol{q}, \boldsymbol{P}, t), \qquad G_3(\boldsymbol{p}, \boldsymbol{Q}, t), \qquad G_4(\boldsymbol{p}, \boldsymbol{P}, t).$$

Inserting the time derivative of the first type generating function $G_1(\boldsymbol{q}, \boldsymbol{Q}, t)$ into Eq. 3.2 leads to:

$$\dot{\boldsymbol{Q}} \cdot \boldsymbol{P} - K(\boldsymbol{Q}, \boldsymbol{P}, t) + \frac{\partial G_1}{\partial \boldsymbol{q}} \dot{\boldsymbol{q}} + \frac{\partial G_1}{\partial \boldsymbol{Q}} \dot{\boldsymbol{Q}} + \frac{\partial G_1}{\partial t} = \dot{\boldsymbol{q}} \cdot \boldsymbol{p} - H(\boldsymbol{q}, \boldsymbol{p}, t).$$

Comparing coefficients, gives the following relations for the first type generating function:

$$\frac{\partial G_1}{\partial \boldsymbol{q}} = \boldsymbol{p}, \qquad \frac{\partial G_1}{\partial \boldsymbol{Q}} = -\boldsymbol{P}, \qquad H(\boldsymbol{q}, \boldsymbol{p}, t) = K(\boldsymbol{Q}, \boldsymbol{P}, t) - \frac{\partial G_1}{\partial t}. \tag{3.3}$$

Similar relations for the other types of generating functions (see Eq. 3.4) are obtained by Legendre transformations of this generating function and then again equating coefficients using Eq. 3.2. Time dependence of generating functions does not change due to the Legendre transformations. So, the relation of Hamiltonian functions H and K in Eq. 3.3 remains unchanged.

$$\begin{aligned}
G_1 &= G_2 - \boldsymbol{Q} \cdot \boldsymbol{P}, & \frac{\partial G_2}{\partial \boldsymbol{q}} &= \boldsymbol{p}, & \frac{\partial G_2}{\partial \boldsymbol{P}} &= \boldsymbol{Q}. \\
G_1 &= G_3 + \boldsymbol{q} \cdot \boldsymbol{p}, & \frac{\partial G_3}{\partial \boldsymbol{p}} &= -\boldsymbol{q}, & \frac{\partial G_3}{\partial \boldsymbol{Q}} &= -\boldsymbol{P}. \\
G_1 &= G_4 - \boldsymbol{Q} \cdot \boldsymbol{P} + \boldsymbol{q} \cdot \boldsymbol{p}, & \frac{\partial G_4}{\partial \boldsymbol{p}} &= -\boldsymbol{q}, & \frac{\partial G_4}{\partial \boldsymbol{P}} &= -\boldsymbol{Q}.
\end{aligned} \tag{3.4}$$

Next, to introduce a more general notation of Eq. 3.4, a second type generating function $G_2(\boldsymbol{q}, \boldsymbol{P})$ is used, in particular:

$$\begin{pmatrix} \partial/\partial \boldsymbol{q} \\ \partial/\partial \boldsymbol{P} \end{pmatrix} G_2(\boldsymbol{q}, \boldsymbol{P}) = \begin{pmatrix} \boldsymbol{p} \\ \boldsymbol{Q} \end{pmatrix}. \tag{3.5}$$

Remember, that $\boldsymbol{q}$, $\boldsymbol{Q}$, $\boldsymbol{p}$, and $\boldsymbol{P}$ are vectors. In order to keep the notation as simple as possible $\boldsymbol{q}$ and $\boldsymbol{P}$ are concatenated to one single vector $\boldsymbol{\omega}$. Likewise $\boldsymbol{p}$ and $\boldsymbol{Q}$ are merged to $\boldsymbol{\gamma}$:

$$\boldsymbol{\omega} = \begin{pmatrix} \boldsymbol{q} \\ \boldsymbol{P} \end{pmatrix}, \qquad \boldsymbol{\gamma}(\boldsymbol{\omega}) = \begin{pmatrix} \boldsymbol{p}(\boldsymbol{\omega}) \\ \boldsymbol{Q}(\boldsymbol{\omega}) \end{pmatrix}.$$

Then, Eq. 3.5 is simplified to:

$$\frac{\partial}{\partial \boldsymbol{\omega}} G_2(\boldsymbol{\omega}) = \boldsymbol{\gamma}. \tag{3.6}$$

From Warnock and Ellison, 1997 we know that the curl of the right hand side of Eq. 3.6 vanishes as canonical transformation are symplectic. As $\boldsymbol{\gamma}$ may have more than three dimensions, its curl is interpreted in a tensorial sense:

$$(\boldsymbol{\nabla} \times \boldsymbol{\gamma}(\boldsymbol{\omega}))_{lj} = \frac{\partial \gamma_j(\boldsymbol{\omega})}{\partial \omega_l} - \frac{\partial \gamma_l(\boldsymbol{\omega})}{\partial \omega_i} = 0. \tag{3.7}$$

In the following section we see that canonical transformations are symplectic and conserve volume in phase space, so that Eq. 3.7 is fulfilled.

3.2 Symplecticity of canonical transformations

Consider a two-dimensional configuration space (q, p): then symplecticity means that a certain area in phase space remains constant over time. Going to higher dimensions, symplectic means, that each projection of a certain volume on a two dimensional subspace (q_i, p_i) of configuration space remains constant over time. In particular, this implies phase volume conservation. Detailed definitions are given by Hairer, Lubich, and Wanner, 2006.

3.2.1 Proof in one dimension

Here it is shown that canonical transformations, which are not time dependent explicitly, are symplectic. As the mapping in the following chapter will be defined in one dimensional configuration space (q, p), $n = 1$ will be sufficient here.

From Hairer, Lubich, and Wanner, 2006 we know, that transformations are symplectic, whenever the condition:

$$\left(\frac{\partial \phi_t}{\partial y_0}\right)^T J \left(\frac{\partial \phi_t}{\partial y_0}\right) = J, \qquad J = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}, \tag{3.8}$$

is fulfilled. Final coordinates ϕ_t are denoted by (Q, P), and initial coordinates y_0 by (q, p). Thus, the expressions $\left(\frac{\partial \phi_t}{\partial y_0}\right)^T$ and $\left(\frac{\partial \phi_t}{\partial y_0}\right)$ may be replaced by:

$$\left(\frac{\partial \phi_t}{\partial y_0}\right)^T = \left(\frac{\partial (q,p)}{\partial (Q,P)}\right) = \left(\frac{\partial (q,p)}{\partial (q,P)}\right)\left(\frac{\partial (q,P)}{\partial (Q,P)}\right) = \begin{pmatrix} \frac{\partial q}{\partial Q} & 2\frac{\partial q}{\partial P} \\ 0 & \frac{\partial p}{\partial P} \end{pmatrix},$$
$$\left(\frac{\partial \phi_t}{\partial y_0}\right) = \left(\frac{\partial (Q,P)}{\partial (q,p)}\right) = \left(\frac{\partial (Q,P)}{\partial (q,P)}\right)\left(\frac{\partial (q,P)}{\partial (q,p)}\right) = \begin{pmatrix} \frac{\partial Q}{\partial q} & 0 \\ 2\frac{\partial P}{\partial q} & \frac{\partial P}{\partial p} \end{pmatrix},$$

where the fact, that (q, p) and (Q, P) are independent sets of variables is used. In addition, we use $\frac{\partial P}{\partial q}\frac{\partial q}{\partial Q} = 0$, $\frac{\partial Q}{\partial q}\frac{\partial q}{\partial P} = 0$ and carry out the matrix multiplication of the symplectic condition resulting in:

$$\left(\frac{\partial \phi_t}{\partial y_0}\right)^T J \left(\frac{\partial \phi_t}{\partial y_0}\right) = \begin{pmatrix} 0 & -\frac{\partial P}{\partial p}\frac{\partial q}{\partial Q} \\ \frac{\partial p}{\partial P}\frac{\partial Q}{\partial q} & 0 \end{pmatrix} = \begin{pmatrix} 0 & -\left(\frac{\partial^2 G_2(q,P)}{\partial q \partial P}\right)^{-2} \\ \left(\frac{\partial^2 G_2(q,P)}{\partial q \partial P}\right)^2 & 0 \end{pmatrix}. \tag{3.9}$$

On the right side, a second type generating function was used to express variables p and Q. Remember, that the generating function describes the

desired canonical transformation and note that it is not explicitly time dependent.

At the starting point ($t = 0$) final coordinates equal initial coordinates ($Q = q$, $P = p$), so that the central term of Eq. 3.9 equals J and the symplectic condition (Eq. 3.8) is true for $t = 0$.

The total time derivative of the right-hand side of Eq. 3.9 is zero. This can be deduced from:

$$\begin{aligned}\frac{d}{dt}\left(\frac{\partial^2 G_2(q,P)}{\partial q \partial P}\right) &= \frac{\partial^3 G_2(q,P)}{\partial q^2 \partial P}\frac{\partial q}{\partial t} + \frac{\partial^3 G_2(q,P)}{\partial q \partial P^2}\frac{\partial P}{\partial t} \\ &= \frac{\partial^2 p}{\partial q \partial P}\frac{\partial q}{\partial t} + \frac{\partial^2 Q}{\partial q \partial P}\frac{\partial P}{\partial t} = 0.\end{aligned}$$

Summing up, the symplectic condition holds true at the starting point $t = 0$ and its total time derivative is zero. This means that Eq. 3.9 remains constant in time. As a result, canonical transformations described by a generating function $G_2(q, P)$, are symplectic at any time.

3.2.2 Phase volume conservation: proof in two dimensions

The two-dimensional volume element of phase space,

$$\iiiint_{\partial V} dQ_1 dQ_2 dP_1 dP_2 = \iiiint_{\partial V} \left|\frac{\partial(Q_1,Q_2,P_1,P_2)}{\partial(q_1,q_2,p_1,p_2)}\right| dq_1 dq_2 dp_1 dp_2,$$

is invariant under canonical transformation, when the Jacobian equals one:

$$
\begin{aligned}
\left|\frac{\partial(Q_1,Q_2,P_1,P_2)}{\partial(q_1,q_2,p_1,p_2)}\right| &= \left|\frac{\partial(Q_1,Q_2,P_1,P_2)}{\partial(q_1,q_2,P_1,P_2)}\right| \cdot \left(\left|\frac{\partial(q_1,q_2,p_1,p_2)}{\partial(q_1,q_2,P_1,P_2)}\right|\right)^{-1} \\
&= \begin{vmatrix} \frac{\partial^2 G_2}{\partial P_1 \partial q_1} & \frac{\partial^2 G_2}{\partial P_1 \partial q_2} & 0 & 0 \\ \frac{\partial^2 G_2}{\partial P_2 \partial q_1} & \frac{\partial^2 G_2}{\partial P_2 \partial q_2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \cdot \left(\begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{\partial^2 G_2}{\partial q_1 \partial P_1} & \frac{\partial^2 G_2}{\partial q_1 \partial P_2} \\ 0 & 0 & \frac{\partial^2 G_2}{\partial q_2 \partial P_1} & \frac{\partial^2 G_2}{\partial q_2 \partial P_2} \end{vmatrix} \right)^{-1} \\
&= \frac{\partial^2 G_2}{\partial P_1 \partial q_1} \cdot \frac{\partial^2 G_2}{\partial P_2 \partial q_2} \cdot \left(\frac{\partial^2 G_2}{\partial P_1 \partial q_1} \cdot \frac{\partial^2 G_2}{\partial P_2 \partial q_2} \right)^{-1} = 1.
\end{aligned}
$$

In the first line the Jacobian was split. In the second line components $\boldsymbol{Q}$ and $\boldsymbol{p}$ were expressed via a generating function of second kind G_2 by the relations given in Eq. 3.6. Note that each coordinate is treated as independent from the other coordinates of the same set.

4 B-Spline Interpolation

In order to create the mapping, two-dimensional interpolation functions have to be constructed. In this approach B-spline interpolation is used, because it consists of integrable and differentiable basis functions. First, in section 4.1, B-spline interpolation is explained in one dimension. Then, in section 4.2, the approach is expanded to higher dimensions. Finally, in the last section 4.3, the collocation method, a way to construct a B-spline approximation function from known partial derivatives, is clarified.

4.1 B-spline interpolation in one dimension

4.1.1 Definition

The interpolation $f_i(x)$ of an arbitrary function $f(x)$ is given by:

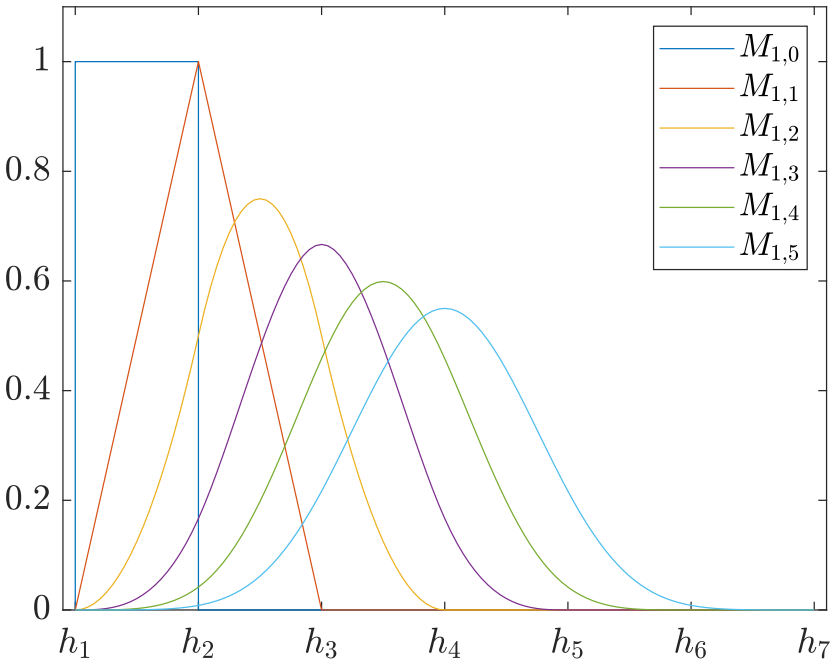
$$f_i(x) = \sum_{l=1}^{N_x} \lambda_l M_{l,k}(x). \tag{4.1}$$

It consists of the sum over N_x weights λ_l multiplied by the basis functions $M_{l,k}$ dependent on x. The second lower index k denotes the order of the basis functions, which are expressed by:

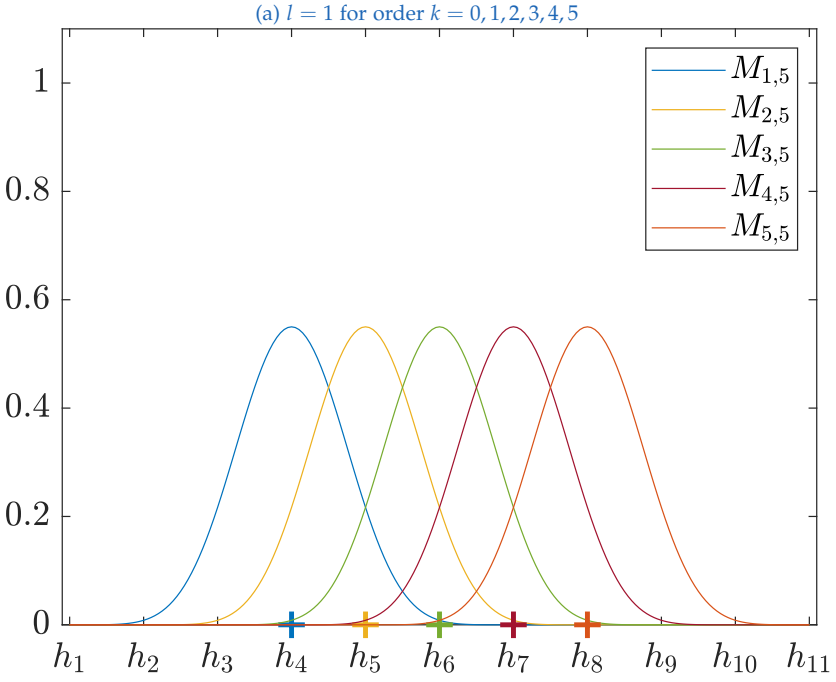
$$M_{l,0}(x) = \begin{cases} 1 & h_l \leq x < h_{l+1} \\ 0 & otherwise \end{cases},$$

$$M_{l,k}(x) = \frac{x - h_l}{h_{l+k} - h_l} M_{l,k-1} + \frac{h_{l+k+1} - x}{h_{l+k+1} - h_{l+1}} M_{l+1,k-1},$$

where h_l are $N_x + k + 1$ predefined knots with $h_l \leq h_{l+1}$. Fig. 4.1a shows the B-spline basis function $M_{1,k}(x)$ for order $k = 0, 1, 2, 3, 4, 5$.



(a) $l = 1$ for order $k = 0, 1, 2, 3, 4, 5$



(b) $l = 1, 2, 3, 4, 5$ for order $k = 5$; centers of the basis functions are marked by +'s.

Figure 4.1: B-spline basis functions, knots are labeled by h

The start point of the next basis function $M_{2,k}(x)$ is shifted from h_1 to h_2. Fig. 4.1b shows the B-spline basis function $M_{l,5}(x)$ for $l = 0, 1, 2, 3, 4, 5$. Once more, a given interpolation function with N_x weights demands $N_x + k + 1$ predefined knots. To illustrate this fact, centers of the B-spline basis functions are marked by +'s in Fig. 4.1b. Naturally, the number of centers matches the number of weights. In the given example $N_x + k + 1 = 11$ knots are needed to provide $N_x = 5$ B-spline basis functions of order $k = 5$.

4.1.2 Determining weights

Knowing function values $f(x_d) = z_d$ and the corresponding x_d for $d = 1, 2, \ldots, N_x$, weights λ_l are defined by a matrix equation. This interpolation problem can be solved by an external Matlab routine provided by Mineault, 2011. Depending on the chosen knots and the order of the interpolation function, there may be more weights than data points leading to an under-determined matrix equation. Using boundary conditions, equations add up to the matrix equation and the system then is well-defined. The Matlab routine does not support boundary conditions, so additional data has to be provided.

4.1.3 Derivatives and integrals

Differentiation and integration of the B-spline function, given in Eq. 4.1, were formulated by Heppler, Vermeulen, and Bartels, 1992:

$$\begin{aligned} \frac{d}{dx} \sum_{l=1}^{N_x} \lambda_l M_{l,k}(x) &= \sum_{l=1}^{N_x+1} \lambda_l{}^{(1)} M_{l,k-1}(x), \\ \int_{-\infty}^{x_{\max}} \sum_{l=1}^{N_x} \lambda_l M_{l,k}(x) &= \sum_{l=1}^{N_x} \lambda_l{}^{(-1)} M_{l,k+1}(x). \end{aligned} \tag{4.2}$$

The equations defining $\lambda_l^{(1)}$ and $\lambda_l^{(-1)}$ read:

$$\begin{aligned}\lambda_l{}^{(1)} =& \frac{k}{h_{l+k} - h_l}\lambda_l - \frac{k}{h_{l+k+1} - h_{l+1}}\lambda_{(l-1)},\\ \lambda_l{}^{(-1)} =& \frac{h_{l+k+1} - h_l}{k+1}\lambda_l + \frac{h_{l+k+1} - h_l}{h_{l+k} - h_{l-1}}\lambda_{(l-1)}{}^{(-1)},\end{aligned}$$

where the coefficients λ_0, $\lambda_{(N_x+1)}$ and $\lambda_0^{(-1)}$ are defined to be zero. Applying definite integration one should keep in mind to subtract the lower limit - from the upper limit integral.

Execution of the collocation method demands the derivative of the interpolation function to be expressed in terms of the basis function itself. For that reason the derivative in Eq. 4.2 is reformulated to:

$$\begin{aligned}\frac{d}{dx}\sum_{l=1}^{N_x}\lambda_l M_{l,k}(x) =& \sum_{l=1}^{N_x}\lambda_l M_{l,k-1}^{(1)}(x)\\ M_{l,k-1}^{(1)}(x) =& \frac{k}{h_{l+k} - h_l}M_{l,k-1}(x) - \frac{k}{h_{l+k+1} - h_{l+1}}M_{l+1,k-1}(x),\end{aligned} \tag{4.3}$$

where $M_{N_x+1,k-1}(x) = 0$.

4.1.4 B-spline interpolation of example function

To illustrate B-spline interpolation a simple example is shown here: the one-dimensional function $f(x) = x^2$.

Consider data $z_d = f(x_d)$ given on eleven points for $x = 0, 1, 2, \ldots, 10$. Third order interpolation ($k = 3$) is done. The range where basis functions are defined has to exceed the data range in a way that the first and the last data point are covered by rising and falling basis functions. Here, this problem is encountered by defining eleven basis functions and adding $k - 1$ at the edges. In total, $N_x = 11 + k - 1 = 13$ weights are defined. Therefore, $N_x + k + 1 = 17$ knots are needed, leading to an equidistant knot vector covering the range from -3 to 13 with a step of size one.

Supplying extra data points to enforce boundary conditions in particular means either evaluating the function on additional points on the left and right hand side at $x = -1$ and $x = 11$ or between the data points at the edges ($x = 0.5$ and $x = 9.5$).

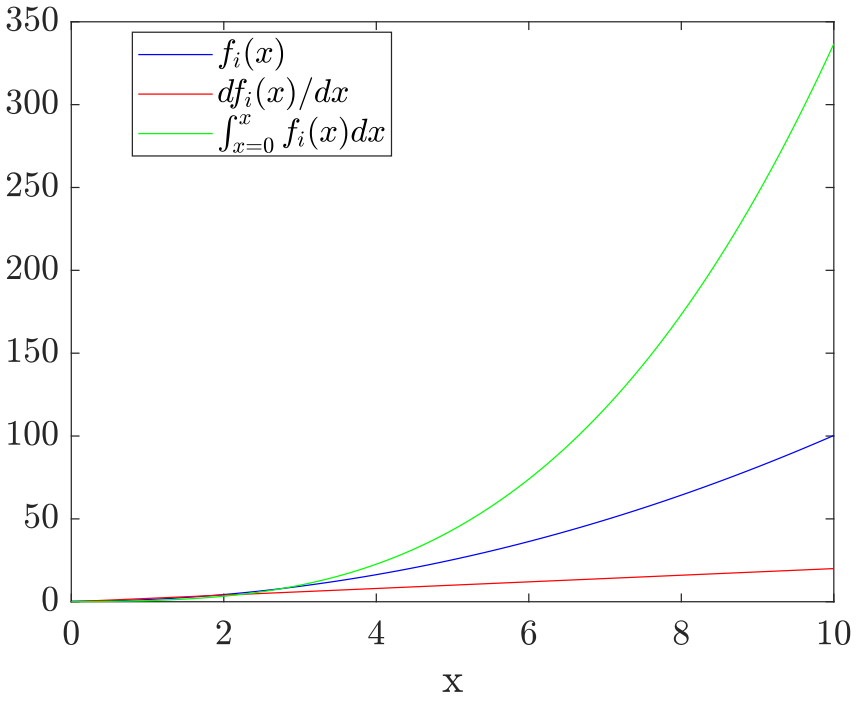


Figure 4.2: B-spline Interpolation of $f(x) = x^2$, its derivative and its integral providing additional data points to enforce boundary conditions

The resulting interpolation function $f_i(x)$, its derivatives and its integrals evaluated with Eq. 4.2 are shown in Fig. 4.2.

4.2 Tensor product B-splines

4.2.1 Definition

The surface interpolation $g_i(x,y)$ or approximation $g_a(x,y)$ of an arbitrary function $g(x,y)$ is given by the tensor product of basis functions:

$$g_{i,a}(x,y) = \sum_{l=1}^{N_x} \sum_{j=1}^{N_y} \lambda_{lj} M_{l,k_x}(x) M_{j,k_y}(y). \tag{4.4}$$

This is the sum over weights λ_{lj} multiplied by the basis functions $M_{l,k_x}(x)$ and $M_{j,k_y}(y)$ dependent on x and y respectively. The second lower indices k_x and k_y represent the order of the corresponding basis functions.

4.2.2 Determining weights

From known function values $z_{dm} = g(x_d, y_m)$ and the corresponding x_d and y_m for $d = 1,2,\ldots,N_d$ and $m = 1,2,\ldots,N_m$ the λ_{lj} are determined by $N_d \times N_m$ linear equations. If (x,y) data is defined on a regular grid and there are as many equations as unknowns λ_{lj}, the system can be solved by the method given in section 4.2.2. Else, least squares regression described in section 4.2.2 is used.

B-spline interpolation of data given on a regular grid

Generally, a grid is regular when the y_m values for $m = 1,2,\ldots,N_m$ are the same for varying x_d and vise versa. In the regular case, according to Piegl and Tiller, 1997, λ_{lj} can be obtained more simply and efficiently as a sequence of curve interpolations:

$$z_{dm} = \sum_{d=1}^{N_x} M_{l,k_x}(x_d) \left(\sum_{j=1}^{N_y} M_{j,k_y}(y_m) \lambda_{lj} \right) = \sum_{l=1}^{N_x} M_{l,k_x}(x_d) R_{ld}.$$

First, one dimensional interpolation over y on isoparametric curves is done, supplying additional data points at each boundary (as described for the one dimensional case). This means, that l is fixed, the interpolation is done N_x times for $l = 1, 2, \ldots, N_x$ and the coefficients R_{ld} are saved. Then the interpolation is done N_y times in the other direction x with fixed d.

B-spline approximation of scattered data

When data $z = g(x, y)$ is given for scattered x and y values, the system of equations 4.4 is solved as a whole. Although the data is scattered, the knots defining B-spline basis functions have to be defined on a regular grid.

Boundary conditions can be added as constraints to the system. In application, $k - 1$ constraints are formulated in each direction.

When the system is overdetermined, i. e. there are more data points than weights, it is solved to minimize least squares without violating constraints. In practice, this is implemented for example by the Matlab routine *lsqlin*.

4.2.3 Partial derivatives and integrals

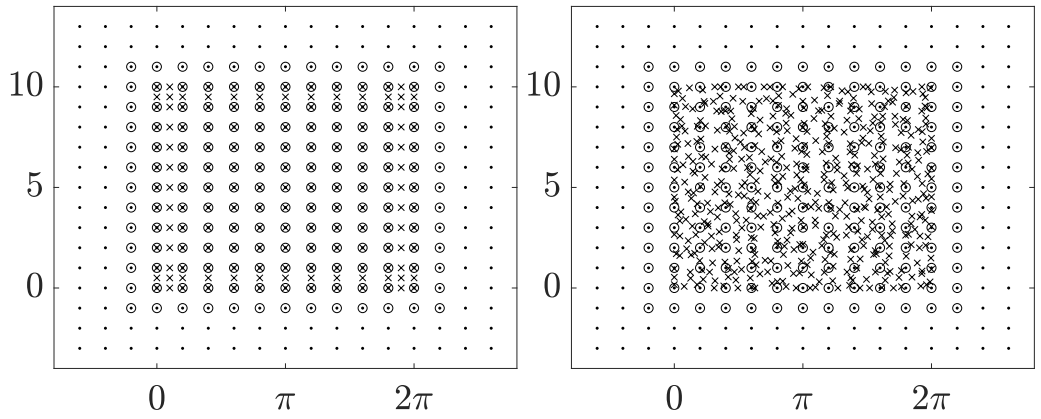
Extending the system to higher dimensions does not influence the relations given in Eq. 4.2. Partial derivatives and integrals in x implicate applying Eq. 4.2 N_y times on λ_{lj} for fixed $j = 1, 2, \ldots, N_y$ and changing the order of the basis functions $M_{l,k_x}(x)$ to $M_{l,k_x \pm 1}(x)$, while leaving $M_{j,k_y}(y)$ unchanged.

Likewise, partial derivatives and integrals in y implicate applying Eq. 4.2 N_x times on λ_{lj} for fixed $l = 1, 2, \ldots, N_x$ and changing the order of the basis functions $M_{j,k_y}(y)$ to $M_{j,k_y \pm 1}(y)$, while leaving $M_{l,k_x}(x)$ unchanged

4.2.4 B-spline interpolation of example function

To make the applications of this section clear, B-spline interpolation of data given on a regular grid and approximation of scattered data is done for

a two dimensional example function $g(x, y) = y^2 \cos(x)$. Both cases are illustrated in Fig. 4.3a and Fig. 4.3b, respectively.



(a) regular data (x)

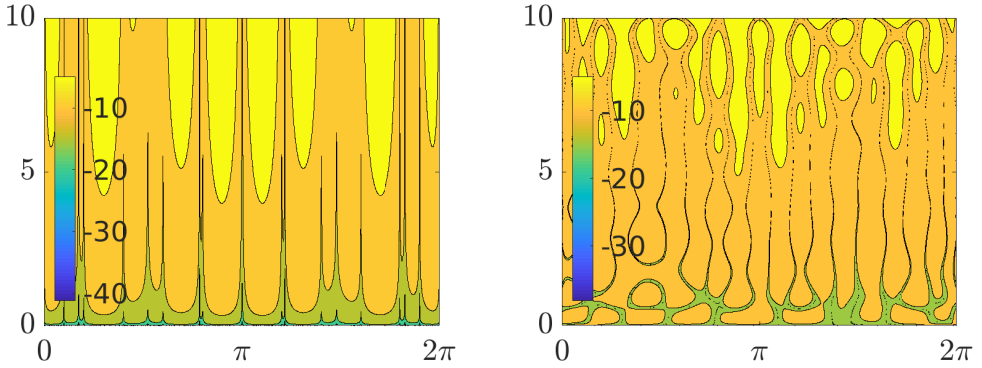
(b) overdetermined system: scattered data(x)

Figure 4.3: regular interpolation grid defined by knots (dots) and centers of the B-spline basis functions (o) for regular data (a) where boundary conditions are circumvented by supplying extra data on edges and scattered data (b) (twice as much data points as weights), where boundary conditions are formulated explicitly

In order to determine the weights λ_{lj} of the interpolation function $g_i(x, y)$ with given data illustrated in Fig. 4.3a the approach explained in section 4.2.2 is used.

In addition, the solution for λ_{lj} defining the approximation function $g_a(x, y)$ of the overdetermined, scattered system shown in Fig. 4.3b is found following the explanation given in section 4.2.2. In particular, boundary conditions in x are periodic ($g_a(0, y) = g_a(2\pi, y)$ and $\frac{\partial g_a(x,y)}{\partial x}|_{x=0} = \frac{\partial g_a(x,y)}{\partial x}|_{x=2\pi}$). Boundaries in y are set to constant values ($g_a(x, 0) = g(x, 0)$ and $g_a(x, 0) = g(x, 0)$). Both functions $g_i(x, y)$ and $g_a(x, y)$ are evaluated on a fine grid and compared to the analytic form given above. Results are shown in Fig. 4.4.

Comparing the two approaches, the quality is more or less the same. On the one hand, the interpolation approach is faster. On the other hand, the approximation procedure allows scattered data. However, it depends on the task which of the two approaches to choose.



(a) $\log_{10}(|g_i(x,y) - g(x,y)|)$ (b) $\log_{10}(|g_a(x,y) - g(x,y)|)$

Figure 4.4: Logarithmic deviation of the interpolation function $g_i(x,y)$ and the approximation function $g_a(x,y)$ from the analytic form $g(x,y)$. Corresponding interpolation (approximation) grids are given in Fig. 4.3a and Fig. 4.3b, respectively

4.3 Collocation method

The Collocation method offers a simple way to determine the approximation function of a primitive $G(x,y)$,

$$G_a(x,y) = \sum_{l=1}^{N_x} \sum_{j=1}^{N_y} \lambda_{lj} M_{l,k_x}(x) M_{j,k_y}(y),$$

although only the partial derivatives $\frac{\partial G(x,y)}{\partial x}|_{x_d,y_d} = z_x^{(d)}$ and $\frac{\partial G(x,y)}{\partial y}|_{x_d,y_d} = z_y^{(d)}$ are given.

Using Eq. 4.3 to express the partial derivatives gives the following relations:

$$\begin{aligned} z_x^{(d)} &= \sum_{l=1}^{N_x} \sum_{j=1}^{N_y} \lambda_{lj} M^{(1)}_{l,k_x-1}(x_d) M_{j,k_y}(y_d), \\ z_y^{(d)} &= \sum_{l=1}^{N_x} \sum_{j=1}^{N_y} \lambda_{lj} M_{l,k_x}(x_d) M^{(1)}_{j,k_y-1}(y_d). \end{aligned} \tag{4.5}$$

One can solve the system of equations for λ_{lj} by minimizing least squares following the description given in section 4.2.2. One main difference is that each data combination (x_d, y_d) supplies two equations (for $z_x^{(d)}$ and $z_y^{(d)}$ respectively). Additionally, twice as many data points as grid points are used.

As basis functions given in Eq. 4.5 are modified, the fast interpolation approach described in section 4.2.2 can not be used, which favors least squares approximation, even if the data was given on a regular grid.

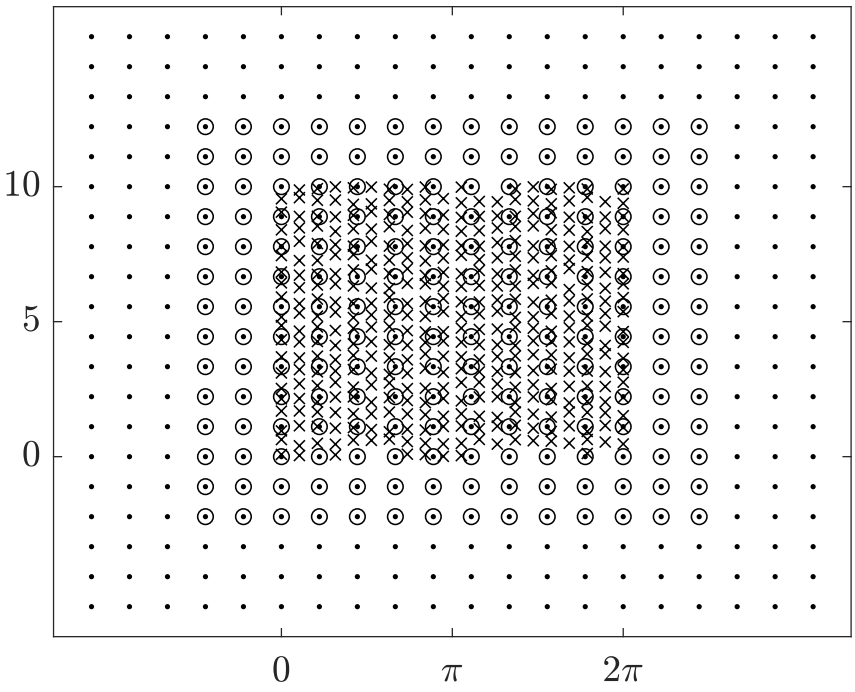
4.3.1 Application of the collocation method

Here, the collocation method is applied to a simple example function $g(x, y) = y^2 \cos(x)$. The partial derivatives $g_x(x, y)$ and $g_y(x, y)$ are given on a grid, which is regular in x and scattered in y, represented in Fig. 4.5a.

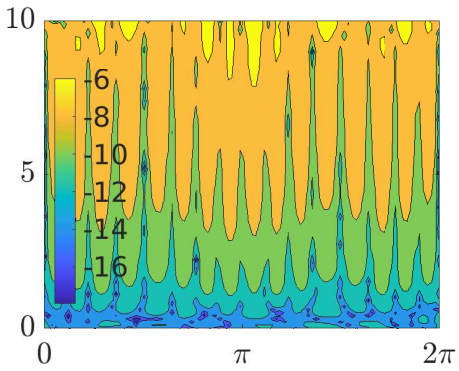
For each data point, Eq. 4.5 is used to set up the system of equations, for basis functions of fifth order ($k_x = 5, k_y = 5$). The system is then solved by minimizing least squares. Boundary conditions are periodic in x and read: $G_a(0, y) = G_a(2\pi, y)$, $\frac{\partial G_a(x,y)}{\partial x}|_{x=0} = \frac{\partial G_a(x,y)}{\partial x}|_{x=2\pi}$, and $\frac{\partial G_a(x,y)}{\partial y}|_{x=0} = \frac{\partial G_a(x,y)}{\partial y}|_{x=2\pi}$.

The resulting partial derivatives of the approximation function $\frac{\partial G_a(x,y)}{\partial x}$ and $\frac{\partial G_a(x,y)}{\partial y}$ are compared to its analytic forms $\frac{\partial g(x,y)}{\partial x}$, and $\frac{\partial g(x,y)}{\partial y}$ respectively. The logarithm of the absolute magnitude of the corresponding differences is illustrated in Fig. 4.5b and Fig. 4.5c.

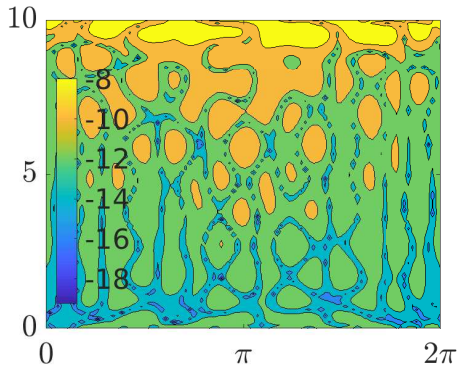
The values of the x dependent part of $f(x, y)$ ($\cos(x)$) vary between -1 and 1, whereas the values of the y dependent part (y^2) are limited to ± 100. As a result, inaccuracy of the partial derivative of the approximation of the x dependent part with respect to x is multiplied by values ranging from -100 to 100 (see Fig. 4.5b). In contrast, inaccuracy of the partial derivative of the approximation of the y dependent part with respect to y is multiplied by values ranging from -1 to 1 (see Fig. 4.5c). As a result the differences given in Fig. 4.5b and Fig. 4.5c vary by the order of 10^2.



(a) regular collocation grid: knots (dots), centers of basis functions (o) and scattered data (x)



(b) $\log_{10}(|\frac{\partial G_a(x,y)}{\partial x} - \frac{\partial g(x,y)}{\partial x}|)$



(c) $\log_{10}(|\frac{\partial G_a(x,y)}{\partial y} - \frac{\partial g(x,y)}{\partial y}|)$

Figure 4.5: Collocation method of fifth order applied to the function $g(x,y) = y^2\cos(x)$

5 Construction and Application of Symplectic Mappings

After the excursion to orbit integration, canonical transformations and B-Spline interpolation in the previous chapters, the theory of these chapters is combined to explain construction and application of symplectic mappings. These are canonical transformations described by interpolated generating functions.

The task is to solve Eq. 3.6 for an interpolation function $G_{i,a}(\boldsymbol{q}, \boldsymbol{P})$, with given initial conditions $\boldsymbol{q}$, $\boldsymbol{Q}$, $\boldsymbol{p}$ and $\boldsymbol{P}$. Here, two different methods are introduced: line integration of the interpolated vector valued function $\gamma(\boldsymbol{\omega})$ (see section 5.1) and approximation of the generating function by using the collocation method (see section 5.2).

Hereupon, in section 5.3 it is shown how the interpolated generating function $G_{i,a}(\boldsymbol{q}, \boldsymbol{P})$ can be used to replace orbit integration. Finally, in section 5.4, suggestions to measure the quality and stability of the mapping are given.

5.1 Interpolation of the generating function by line integration

Whenever a system $(\boldsymbol{q}, \boldsymbol{Q}, \boldsymbol{p}, \boldsymbol{P})$ is symplectic, the curl given in Eq. 3.7 vanishes. As a result, the generating function $G_2(\gamma(\boldsymbol{\omega}))$, which obeys Eq. 3.6, can be evaluated by a path-independent line integral:

$$G_2(\boldsymbol{\omega}) = \int_{\omega_0}^{\omega} \gamma(\boldsymbol{\omega}') d\boldsymbol{\omega}'. \tag{5.1}$$

The choice of the path is free. In the easiest case it is chosen to go in a straight line along the q_1 axis from $q_{0,1}$ to q_1 and then in the same manner along the $q_2, \ldots, q_N, P_1, \ldots, P_N$ axes.

Eq. 5.1 is solved for G_2 by finding the vector valued function $\gamma(\omega)$ by B-spline interpolation and than applying partial integration to these interpolated functions.

The idea behind this mapping procedure was introduced by Warnock, Cai, and Ellison, 2009 and details about B-spline interpolation of the generating function are given by Warnock and Ellison, 1999.

5.1.1 Detailed explanation for one-dimensional case

As a matter of simplicity and tractability, here, a detailed description about the evaluation of Eq. 5.1 is given for a one dimensional case: (q, Q, p, P).

B-spline interpolation functions of higher order poorly interpolate linear terms, so first of all, the generating function of second type $G_2(q, P)$ is transformed:

$$\tilde{G}_2(q, P) = G_2(q, P) - qP \tag{5.2}$$

For the given case Eq. 3.6 reads:

$$\begin{pmatrix} \partial/\partial q \\ \partial/\partial P \end{pmatrix} \tilde{G}_2(q, P) = \begin{pmatrix} p(q, P) - P \\ Q(q, P) - q \end{pmatrix}.$$

The transformation given in Eq. 5.2 is allowed because it leaves the curl (see Eq. 3.7) unchanged:

$$\frac{\partial(p(q, P) - P)}{\partial P} - \frac{\partial(Q(q, P) - q)}{\partial q} = \frac{\partial p(q, P)}{\partial P} - \frac{\partial Q(q, P)}{\partial q}.$$

Finally, mapping functions $\tilde{\gamma}_{1,i}(q, P) = p - P$ and $\tilde{\gamma}_{2,i}(q, P) = Q - q$ are constructed by discrete B-spline interpolation. Eq. 5.1 is then used to determine the interpolated generating function $\tilde{G}_i(q, P)$:

$$\tilde{G}_i(q, P) = \int_{P_0}^{P} \tilde{\gamma}_{2,i}(q_0, P')dP' + \int_{q_0}^{q} \tilde{\gamma}_{1,i}(q', P)dq. \tag{5.3}$$

It is not necessary to interpolate the generating function given in Eq. 5.3 itself, because the mapping described in section 5.3 only demands partial derivatives, which can be evaluated analytically via:

$$\begin{pmatrix} \partial/\partial q \\ \partial/\partial P \end{pmatrix} \tilde{G}_i(q,P) = \begin{pmatrix} \tilde{\gamma}_{1,i}(q,P) \\ \tilde{\gamma}_{2,i}(q_0,P) + \int_{q_0}^{q} \frac{\partial \tilde{\gamma}_{1,i}(q',P)}{\partial P} dq' \end{pmatrix}. \tag{5.4}$$

Discrete interpolation of the mapping functions

Mapping functions $\tilde{\gamma}_{1,i}(q,P)$ and $\tilde{\gamma}_{2,i}(q,P)$ can not be interpolated directly, because the basis for discrete B-spline interpolation in two dimensions has to be regular. Naturally, final coordinates P are scattered and not on a regular grid (see x's and o's in Fig. 5.1).

Therefore, a regular (q,P)-grid is constructed using original q values (initial coordinates q,p are defined on a regular grid) and artificial P values, which are chosen to be equally spaced from the minimal to the maximal values of the final coordinates P, with twice as many steps as the original p vector contained (regular, mixed-variable grid points are depicted as dots in Fig. 5.1).

Now, p and Q have to be related to the regular (q,P)-grid. p can be found by applying Newton's method to find the root of

$$P(q,p) - P = 0, \tag{5.5}$$

for every point on the regular (q,P)-grid. The function $P(q,p)$ corresponds to an integrator solving the equations of motion for a given time. To provide starting values and derivatives needed for Newton solvers, an interpolation function $P_i(q,p)$, which connects the original final momenta with the regular (q,p)-grid, is constructed. Besides, its partial derivative with respect to p is calculated. Summing up, a quasi-Newton method is used to solve Eq. 5.5, where $P(q,p)$ represents the orbit integrator and $\frac{\partial P_i(q,p)}{\partial p}$ is used to approximate the derivative.

Obviously, not all (q,P)-grid points have a valid p value. For example the maximum final momentum $P^{\max}$ corresponds to just one q. In contrast, on

the mixed-variable (q, P)-grid exist many $(q, P^{\max})$ combinations, where just one contains a valid p.

Discrete interpolation methods require valid values on every regular basis grid point. On that account, the mixed-variable (q, P)-grid has to be reduced by lines (or columns) containing invalid (q, P) combinations, resulting in a regular, surjective basis grid (see o's in Fig. 5.1). Furthermore, the surjective q-P data point density is increased on the edges, because additional data is needed to circumvent boundary conditions (see chapter 4).

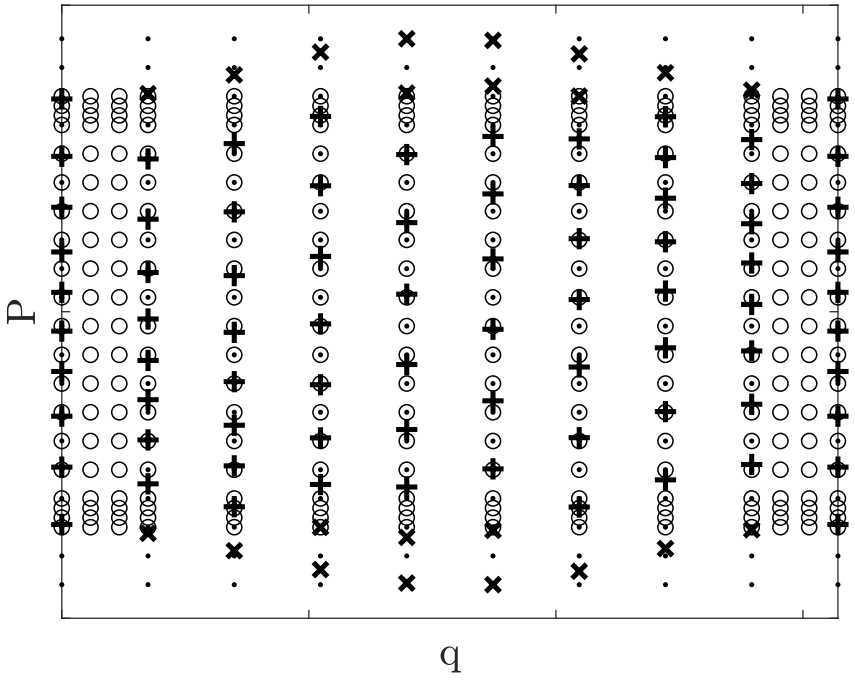


Figure 5.1: regular (q, P)-grid construction:
+ - exemplary, surjective (q, P) points originated from the pendulum
x - examplary, non surjective (q, P) points originated from the pendulum
. - regular, mixed-variable (q, P)-grid
o - surjective, regular, mixed-variable data points

By reducing the mixed-variable (q, P)-grid by non-surjective points, also the application area of the map is reduced.

Discrete interpolation of valid $p - P$ values on the surjective, regular (q, P)-grid results in the mapping function $\tilde{\gamma}_{1,i}(q, P)$, relating the initial momentum

subtracted by the final momentum to the mixed phase space variables q and P. The other mapping function $\tilde{\gamma}_{2,i} = Q_i(q,P) - q$ can be easily found by interpolating values of $Q(q, \tilde{\gamma}_{1,i}(q,P)) + P) - q$ on the surjective, mixed-variable (q,P)-grid. Here, the function $Q(q,p)$ corresponds to an integrator solving the equations of motion for a given time.

5.2 Approximation of the generating function using the collocation method

A different way to approximate the generating function is to define it directly by a B-spline approximation function:

$$G_a(\boldsymbol{\omega}) = \sum_{l}^{N_1}\sum_{m}^{N_2}\cdots\sum_{a}^{N_a}\lambda_{l,m,\ldots,a}M_{l,k_l}(\omega_1)M_{m,k_m}(\omega_2)\ldots M_{a,k_a}(\omega_{N_v}). \tag{5.6}$$

This is the tensor product of basis functions which element-wisely depend on $\boldsymbol{\omega}$, multiplied by a weight tensor λ.

N_v is the length of vector $\boldsymbol{\omega}$, which defines the number of used indices $l,m,\ldots,a$ and thereby the rank of the weight tensor $\lambda_{l,m,\ldots,a}$. In contrast, N_1, N_2,..., N_a determines the number of weights in the corresponding direction.

Now, Eq. 3.6 is used to set up a system of equations, which can be solved for the B-spline coefficients $\lambda_{l,m,\ldots,a}$ by minimizing least squares:

$$\begin{aligned}
\gamma_{1,a} &= \sum_{l}^{N_1}\sum_{m}^{N_2}\cdots\sum_{a}^{N_a}\lambda_{l,m,\ldots,a}M^{(1)}_{l,k_l-1}(\omega_1)M_{m,k_m}(\omega_2)\ldots M_{a,k_a}(\omega_{N_v})\\
\gamma_{2,a} &= \sum_{l}^{N_1}\sum_{m}^{N_2}\cdots\sum_{a}^{N_a}\lambda_{l,m,\ldots,a}M_{l,k_l}(\omega_1)M^{(1)}_{m,k_m-1}(\omega_2)\ldots M_{a,k_a}(\omega_{N_v})\\
&\vdots\\
\gamma_{N_v,a} &= \sum_{l}^{N_1}\sum_{m}^{N_2}\cdots\sum_{a}^{N_a}\lambda_{l,m,\ldots,a}M_{l,k_l}(\omega_1)M_{m,k_m}(\omega_2)\ldots M^{(1)}_{a,k_a-1}(\omega_{N_v})
\end{aligned} \tag{5.7}$$

Each known data point $\gamma_a(\omega_i) = z_i$ gives N_v equations, where partial derivatives $M^{(1)}_{i,k_i-1}$ of B-spline basis functions are defined by Eq. 4.3.

The big advantage of this approach is, that the curl defined in Eq. 3.7, is zero per definition:

$$\frac{\partial \gamma_{j,a}(\boldsymbol{\omega})}{\partial \omega_l} - \frac{\partial \gamma_{l,a}(\boldsymbol{\omega})}{\partial \omega_j} = \frac{\partial^2 G_a(\boldsymbol{\omega})}{\partial \omega_l \partial \omega_j} - \frac{\partial^2 G_a(\boldsymbol{\omega})}{\partial \omega_l \partial \omega_j} = 0. \tag{5.8}$$

As a result, the collocation method can be used to map slightly non-symplectic systems. The input in that case is 'symplectified' and the mapping procedure is still stable, at the cost of slightly shifted trajectories.

5.2.1 Detailed explanation for one-dimensional case

As before, a detailed description about the construction of the generating function is given for a one-dimensional case: (q, Q, p, P). With the same motivations, the transformation described in Eq. 5.2 is performed.

For the one dimensional case Eq. 5.6 reads:

$$\tilde{G}_a(q, P) = \sum_{l=1}^{N_q} \sum_{m=1}^{N_P} \lambda_{lm} M_{l,k_q}(q) M_{m,k_P}(P).$$

Its derivatives are expressed by Eq. 4.3 and Eq. 5.7 results in:

$$\begin{aligned} p - P &= \sum_{l=1}^{N_q} \sum_{m=1}^{N_P} \lambda_{lm} M^{(1)}_{l,k_q-1}(q) M_{m,k_P}(P) \\ Q - q &= \sum_{l=1}^{N_q} \sum_{m=1}^{N_P} \lambda_{lm} M_{l,k_q}(q) M^{(1)}_{m,k_P-1}(P) \end{aligned} \tag{5.9}$$

Now, all initial conditions are used to set up a matrix equation which can be solved for the B-spline coefficients λ_{ij} by minimizing least squares.

Least squares approximation

As B-spline approximation is a discrete method, the interpolation grid has to be regular. So the mixed-variable (q, P)-grid is created in a similar manner as before. Original q values are chosen and P values are equally spaced between P_l and P_u. Taking all q combinations for the biggest p value into consideration, P_u is the minimal value of the corresponding P's. P_l is the minimal value of the P's originated from all q combinations of the lowest p value. The grid is displayed in Fig. 5.2, where grid points are marked with o's. It is bigger than the area of definition, because B-spline approximation of higher order demands additional weights. In particular $\tilde{G}_a$ is approximated by basis functions of fifth order here, requiring $N = k - 1 = 4$ additional weights in each dimension.

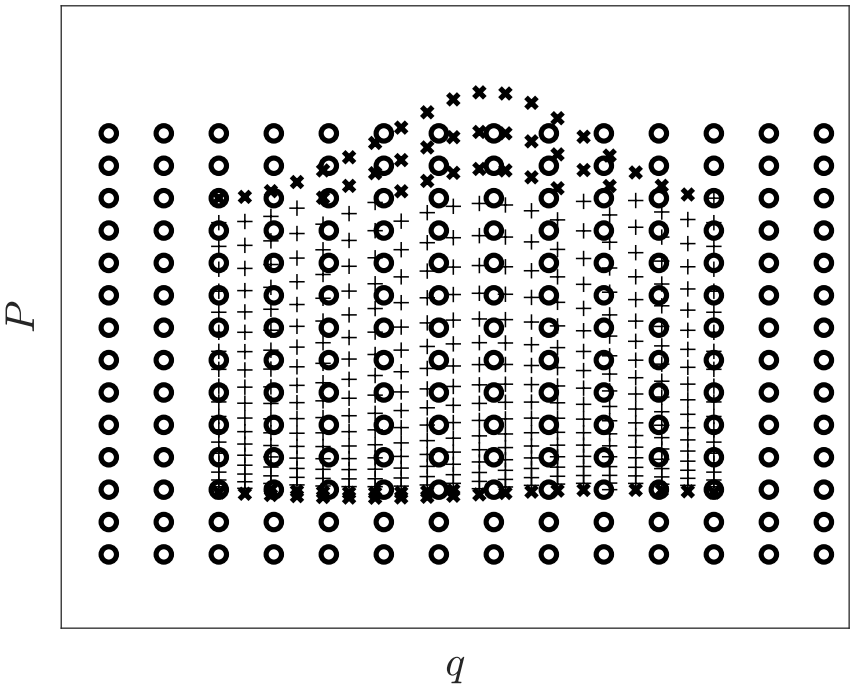


Figure 5.2: Regular (q, P)-grid construction:
+ - exemplary (q, P) data points
x - exemplary (q, P) data points out of mapping area
o - weights of regular, mixed-variable (q, P)-grid (fifth order)

Using approximation instead of interpolation allows the initial conditions (q, P) themselves to not be located exactly at the regular (q, P)-interpolation-grid (see +'s and x's in Fig. 5.2). For that reason, surjectivity in its strict sense is not demanded anymore. But still the system will be underdefined, if the grid includes areas where little or no data exist. Therefore, again the B-spline grid is reduced by the lines containing too little data, which reduces the mapping application area by the data points marked as x's in Fig. 5.2 accordingly. The lower and upper limits of the resulting mapping area correspond to P_l and P_u.

However, Eq. 5.9 is expressed for all valid initial condition pairs (+'s), and the collocation method, described in section 4.3, is applied to solve for the B-spline coefficients λ_{lm}.

5.3 Mapping in one dimension

As the partial derivatives of the generating function $\tilde{G}_{i,a}(q, P)$ are determined, final coordinates and momenta for any point between the grid points can be determined easily. Subscripts refer to generating functions originated from line integration (i) or collocation (a).

However, Newton's method is used to search the final momentum P for the phase space point (p, q) by solving:

$$\frac{\partial \tilde{G}_{i,a}(q, P)}{\partial q} - p + P = 0, \tag{5.10}$$

where $P_i(q, p)$ is used to define the initial guess. The final coordinate is then calculated by:

$$Q = \frac{\partial \tilde{G}_{i,a}(q, P)}{\partial P} + q \tag{5.11}$$

A big advantage of this approach is that the interpolated mapping function can be applied again and again. For any point in the mapping area,

Eq. 5.10 and Eq. 5.11 are applied. Then results (Q, P) are taken as new initial coordinates (q, p) and the procedure is started again.

The mapping procedure does not really change for canonicalized systems. One additional demand is an interpolation function $R_i(q, r)$ connecting the final coordinate R with the initial coordinates (r, q), which provides an initial condition to solve:

$$P(R, Q, V_{||}(R, Q, p_\phi = const)) - P = 0,$$

for R after each mapping step by Newton's method. Details are given in section 2.4.

5.4 Quality of mappings

5.4.1 Geometrical distance

A way to measure the mapping quality in an intuitive sense is to determine the geometrical distance. The idea is to compare the mapped orbit with the location of the 'exact' trajectory from the integrator used to create initial and final coordinates for the map construction. In order to calculate the geometrical distance one needs to evaluate C_N points using the integrator and take out T_N random points of the mapped orbit. For each of the mapped orbit points P_{Mi} the two closest 'exact' orbit points P_{1i} and P_{2i} are searched, a triangle is constructed and the normal distance g_{Ni} is calculated by:

$$g_{Ni} = \frac{|(P_{1i} - P_{2i}) \times (P_{Mi} - P_{2i})|}{|(P_{1i} - P_{2i})|}. \tag{5.12}$$

Then, the root mean square of the normal distances of all T_M test points is built and defined as the geometrical distance:

$$g = \sqrt{\frac{1}{T_N} \sum_{i=1}^{T_N} {g_{Ni}}^2} \tag{5.13}$$

5.4.2 Normalized energy oscillation

As energy is a constant of motion, its deviation can be additionally used to measure the mapping quality.

The normalized energy oscillation ΔH is given by the standard deviation σH devided by its mean $\bar{H}$:

$$\Delta H = \frac{\sigma H}{\bar{H}} \tag{5.14}$$

6 Application to the Pendulum

Symplectic mappings are tested for a canonical case- the simple pendulum in one dimensional phase space (q, p):

$$H = \frac{p^2}{2} - cos(q). \tag{6.1}$$

First, in section. 6.1 it is explained how initial conditions are constructed. Generally, one is free to choose parameters as for example t_M, the time one mapping involves, N_g, the number of grid points to create the initial conditions (q, p), or k, the order of B-spline basis functions.

A certain parameter triple (t_M, N_g, $k = 3$) is chosen and map construction, application and quality measure is done for the line integration method, as well as the collocation method in section 6.2, 6.3 and 6.4.

Finally, in section 6.5, different parameter combinations are tested for the line integration method.

6.1 Initial conditions for the mapping

To create sets of initial conditions (q, Q, p, P), first a regular (q, p)-grid with N_g points in each direction is created. Then, corresponding (Q, P) are found by solving Hamilton's canonical equations obeying Eq. 6.1 with a symplectic Euler scheme.

Trajectories of the pendulum may be either passing or trapped (see Fig. 6.1). Trapped and passing cases are separable by the parameter $\kappa = H$, where H represents the particle's energy given by Eq. 6.1 (see, e. g. Albert, 2017). So,

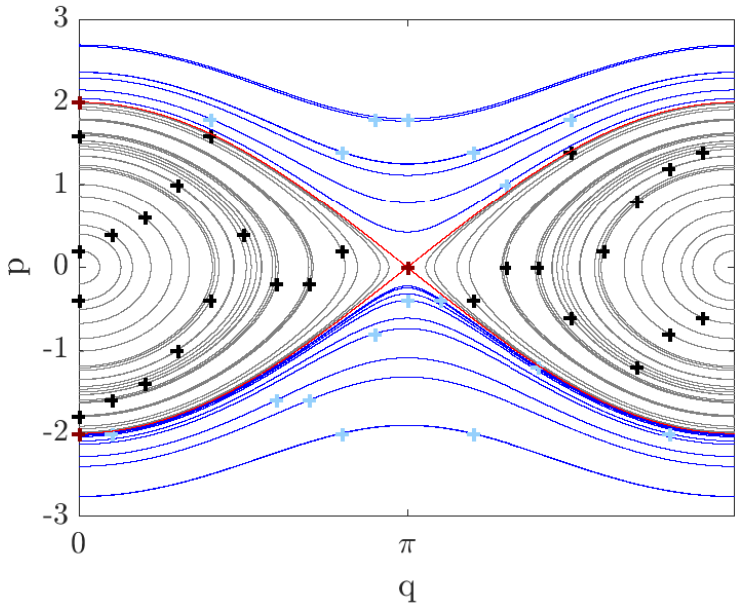


Figure 6.1: Symplectic Euler scheme applied to initial conditions (q, p) (marked by +): some initial conditions are randomly selected from the original set to keep the representation clear. Trapped Orbits are gray, Passing Orbits blue and the separatrix is depicted in red

generally κ depends on initial conditions (q, p). In case $\kappa > 1$ the particle is passing and in case $\kappa < 1$ it is trapped. The limiting case $\kappa = 1$ is called separatrix. In addition, the cycle duration τ for the trapped case can be calculated by an elliptical integral,

$$\tau = 4K(\kappa), \tag{6.2}$$

where K stands for:

$$K = \frac{1}{4} \int_{-2\arcsin(\sqrt{\kappa})}^{+2\arcsin(\sqrt{\kappa})} \frac{d\theta}{\sqrt{\kappa - \sin^2(\frac{\theta}{2})}}.$$

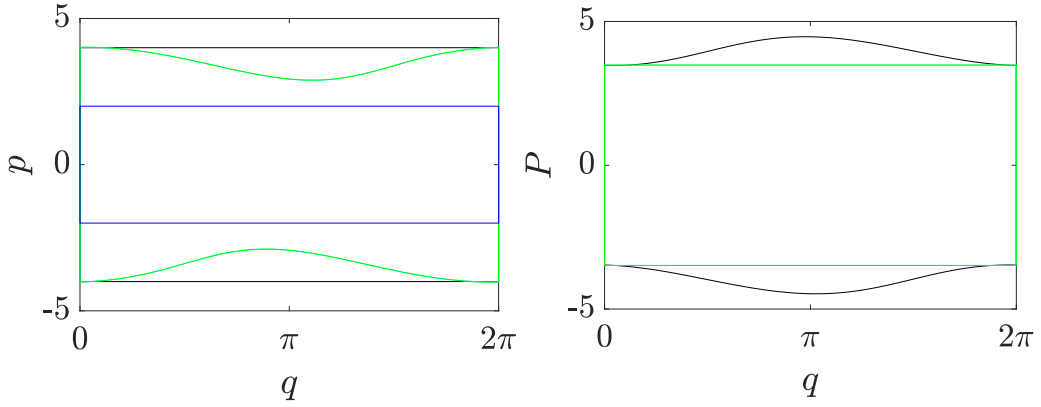
To find a convenient time-step for numerical integration, the linear bounce time τ_b is computed. τ_b is the period of the fastest trapped particle. It is calculated by applying Eq. 6.2 on every point of a test grid, where $H < 1$,

and then searching the minimal value. This results in $\tau_b = 6.283 \sim 2\pi$. Here, the integration time step t_i is set to $t_i = \frac{2\pi}{960}$. This ensures, that not only the fastest trapped particle, but also the fastest passing particle is treated on a sufficient fine time-scale. Besides, the mapping time t_M defining the time interval between (q, p) and (Q, P) has to be chosen.

6.2 Map creation

Here, it is shown how a map is created via line integration and collocation for the parameter triple: $N_g = 64$, $t_M = \frac{1}{8}\tau_b$ $k = 3$. k denotes the order of the original interpolation functions $\tilde{\gamma}_{1,i}(q, P)$ and $\tilde{\gamma}_{2,i}(q, P)$, or the order of the partial derivatives of the generating function $\frac{\partial \tilde{G}_a(q,P)}{\partial q}$ and $\frac{\partial \tilde{G}_a(q,P)}{\partial P}$, respectively.

First of all, the grid area of initial conditions, the resulting valid area of map application and the general test area are shown below.



(a) Initial phase space q-p

(b) miexed-variable phase space q-P

Figure 6.2: Provided data (black), surjective mapping area (green) and test area (blue) in phase space (a) and miexed-variable phase space (b)

Primarily, the grid of initial conditions used to create the mapping is defined (see Fig. 6.2 in black). This area is reduced by 'non-surjective points', resulting in the application area marked in Fig. 6.2 in green. Additionally,

a test grid is chosen (see 6.2a). As a matter of comparability it should be part of the application area of many t_M, N_g, k combinations. Therefore, it is smaller than the application area of the special case ($t_M = \frac{1}{8}\tau_b$, $N_g = 64$, $k = 3$). Still it involves both: trapped and passing particles.

To summarize, the area covered by initial conditions used to create the map is bigger than the area where the mapping can be applied to. This results from B-spline interpolation, which is bound on a regular, rectangular basis grid. So, the problem can be avoided by using mesh-less interpolation methods.

6.2.1 Map creation by line integration

The mapping functions $\tilde{\gamma}_{1,i}$ and $\tilde{\gamma}_{2,i}$ are constructed as described in section 5.1 for the one-dimensional case. The quality of the map depends on the symplecticity of the mapping functions, which is measured by the curl:

$$\nabla \times \tilde{\gamma}_i(q,P) = \frac{\partial \tilde{\gamma}_{2,i}(q,P)}{\partial q} - \frac{\partial \tilde{\gamma}_{1,i}(q,P)}{\partial P}. \tag{6.3}$$

It is evaluated at a fine $q - P$ mesh, and the logarithm of its absolute magnitude is displayed in Fig. 6.3. One can see that it is close to zero everywhere, which is a sign that the map is nearly symplectic. To perceive obvious non - symplecticity (important for testing new parameter combinations), an additional criterion is set up for the area of mapping: the absolute value of the curl (Eq. 3.7) has to be less than a limit value of 10^{-2}.

6.2.2 Map creation by collocation

As a second variant, the generating function is approximated by the collocation method described in section 5.2. As the curl, given in Eq. 5.8 vanishes per definition, the quality of the map is measured directly by analyzing the differences of values of the mapping function and the exact values at the points, where initial conditions are given: $\tilde{\gamma}_{1,a}(q,P) - p + P$ and $\tilde{\gamma}_{2,a}(q,P) - Q + q$. If the map was perfect, these values are zero everywhere. The actual values are given in Fig. 6.4.

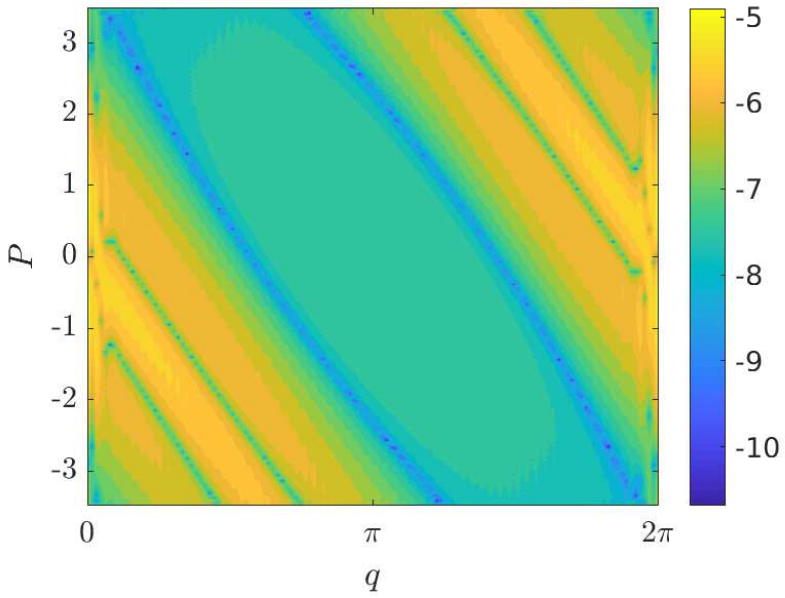
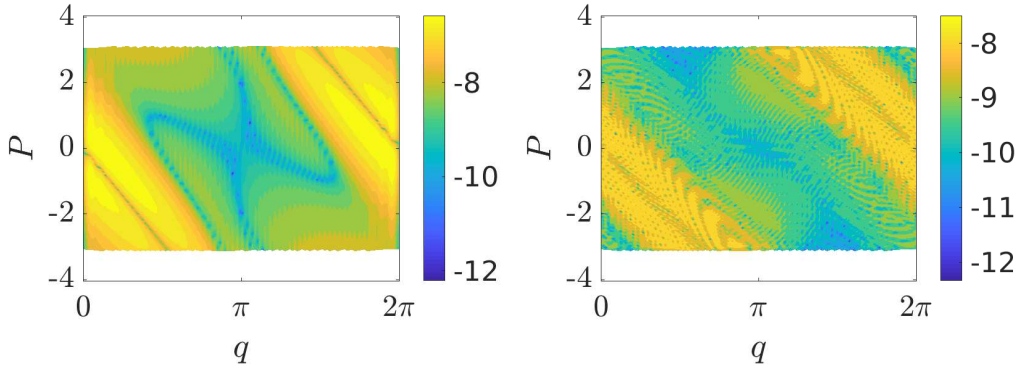


Figure 6.3: Line integration: approximation of map exactness by analyzing symplecticity of the interpolated mapping functions: $\log_{10}(|\nabla \times \tilde{\gamma}_i(q,P)|)$ (see Eq. 6.3) for $t_M = \frac{\tau_b}{8}$, $N_g = 64$ and $k = 3$



(a) $\log_{10}(|\tilde{\gamma}_{1,a}(q,P) - p + P|)$

(b) $\log_{10}(|\tilde{\gamma}_{2,a}(q,P) - Q + q|$

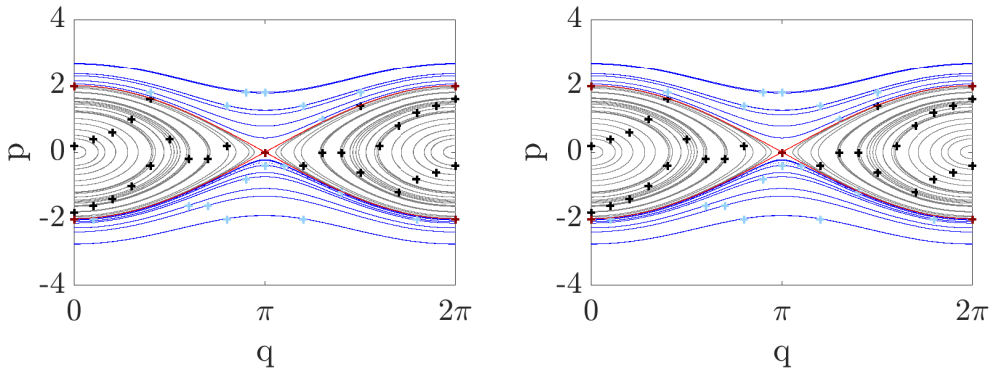
Figure 6.4: Collocation: measure of map exactness by analyzing the logarithm of the absolute value of the difference between mapping function and exact values for $\tilde{\gamma}_{1,a}(q,P)$ in (a) and $\tilde{\gamma}_{2,a}$ in (b)

6.3 Application of the map

The above constructed mapping functions are applied on the test grid ($q\ \epsilon[0, 2\pi], p\ \epsilon[-2, 2]$) marked in Fig. 6.2a in blue, following the description given in section 5.3. $N_t = 21$ test points are chosen in each direction. Note that the test grid points are shifted from the original ones.

Here, integration of 10^4 bounces of the fastest trapped orbit corresponding to $t_f = 10^4 \tau_b = 2 \cdot 10^4\ \pi$ is replaced by the mappings. Considering that the maps are constructed to replace integration from $t = 0$ to $t_M = \frac{1}{8}\tau_b$, the mapping has to be applied $N_M = \frac{t_f}{t_M} = 8 \cdot 10^4$ times.

The results of the mapping are illustrated in Fig. 6.5.



(a) Mapping originated from Line Integration

(b) Mapping originated from Collocation

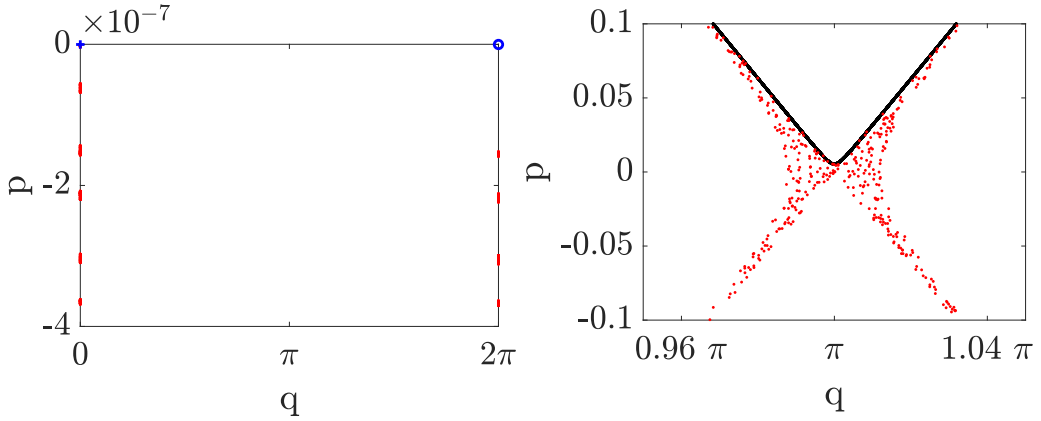
Figure 6.5: Mapped orbits for different initial conditions (q, p) (marked by +) applying the map originated from line integration (b) and from collocation (b).
trapped orbits are gray, passing orbits blue and the separatrix is depicted in red

6.4 Quality of the map

At first, the mappings in Fig. 6.5 look qualitatively similar to the integration depicted in Fig. 6.1. In addition, one wants to have comparable measures of the mapping quality and stability. This is given by the geometrical distance g defined in Eq. 5.13 and the normalized energy oscillation given by Eq. 5.14.

The geometrical distance is calculated for all $N_t^2 = 21 * 21 = 441$ points of the test grid, by evaluating the normal distance g_{N_i} in Eq. 5.12 for $T_N = 1000$ random mapped orbit points, taking $C_N = 50000$ integrated orbit points for comparison into consideration, and forming the root mean square. A few points show anomalies.

On the one hand, there are two unstable points at $(q = 0, p = 0)$ and $(q = 2\pi, p = 0)$. These are caused by the modulo function, which was used to project $q \notin [0, 2\pi[$ back to $q \in [0, 2\pi[$ (which is allowed because the Hamiltonian in Eq. 2.12 is periodic in 2π). Due to integration, these points will change neither momentum p nor position q. Due to the mapping, q and p vary a little bit. Small variations in q lead to a jump of $q = 0$ to $q = 2\pi$ (see Fig. 6.6a) or vice versa, which results in an extremely high geometrical distance g.

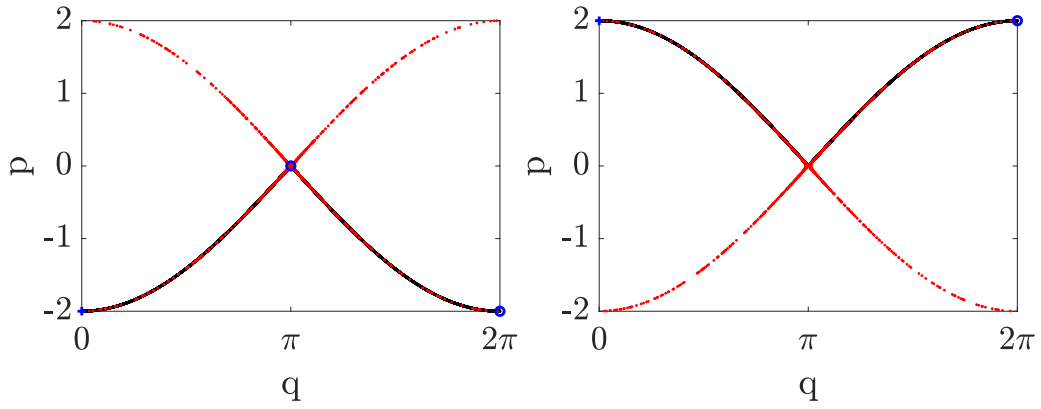


(a) unstable points caused by the modulo function

(b) zoom into separatrix

Figure 6.6: Integrated orbits (black) and mapped orbits (red) with initial Conditions (q, p) (marked by +); points leading to similar results are depicted by o

On the other hand, initial conditions lying on the seperatrix lead to deviations. Mathematically, orbits at the separatrix slow down, until they reach the instable cross point at $(q = \pi, p = 0)$, where they stay forever. Symplectic integration (like the Euler scheme used here) deforms this reality a little bit, leading to passing orbits slightly away from the separatrix. So depending on initial conditions a particle is restricted to either the upper or the lower part of the separatrix (see Fig. 6.7b and Fig. 6.7a respectively).



(a) lower seperatrix (b) upper seperatrix

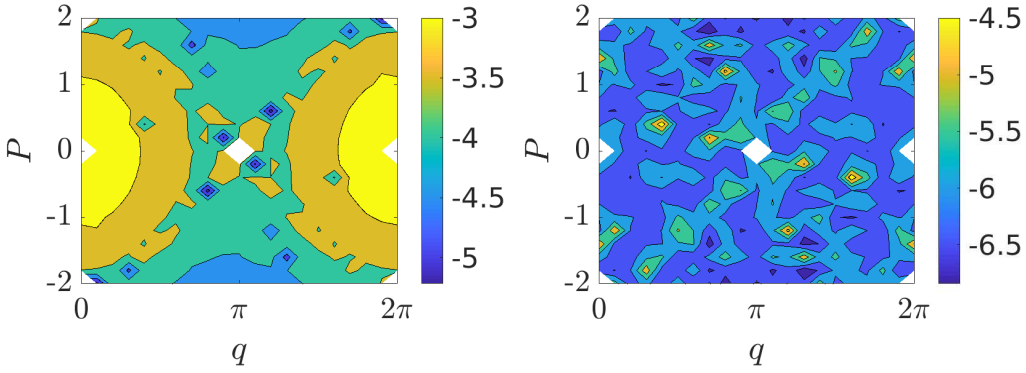
Figure 6.7: Integrated orbits (black) and mapped orbits (red) with initial conditions (q, p) (marked by +)
points leading to similar results are depicted by o

Mapping approximation leads to a scattering from the upper to the lower part and vice versa (again: see Fig. 6.7b and Fig. 6.7a), which results in high geometrical distances for these orbits. A closer look at the separatrix is given in Fig. 6.6b.

In summary, the geometrical distance of the whole map is determined by averaging over all geometrical distances of the test grid points and evaluating the standard deviation. All 7 anomalous initial conditions, depicted in Fig. 6.6a, Fig. 6.7a, and Fig. 6.7b in blue, are excluded from the calculation, because they lead to high values which do not correspond to the physical situations. The logarithm of the geometrical distances of the remaining test points are shown in Fig. 6.8.

In addition, the normalized energy oscillation is measured for each test point, by evaluating Eq. 6.1 at every time step, calculating mean value and standard deviation and building the ration given in Eq. 5.14. At $(p = 0, q = \frac{\pi}{2})$ and $(p = 0, q = \frac{3\pi}{2})$, the initial energy is zero, making normalization impossible. For that reason, those points are excluded from the calculation. Results are shown in Fig. 6.9 where two extra data points are excluded because their oscillation is so small, that they distort the scaling of the plots.

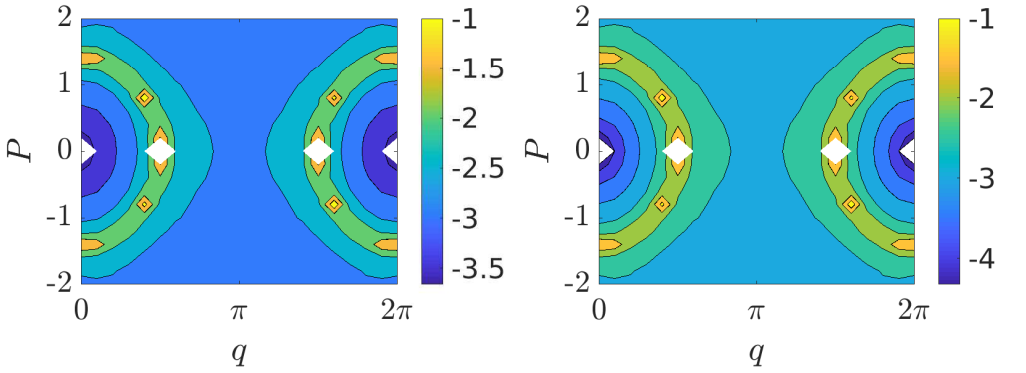
Comparing both methods, the collocation seems to be more precise. Whereas



(a) Mapping originated from Line Integration

(b) Mapping originated from Collocation

Figure 6.8: Logarithm of the geometrical distance $\log_{10}(g)$ (see Eq. 5.13) for mapping originated from line integration (a) and from collocation (b).



(a) Mapping originated from Line Integration

(b) Mapping originated from Collocation

Figure 6.9: Logarithm of the normalized energy oscillation $\log_{10}(\Delta H)$ (see Eq. 5.14) for mapping originated from line integration (a) and from collocation (b).

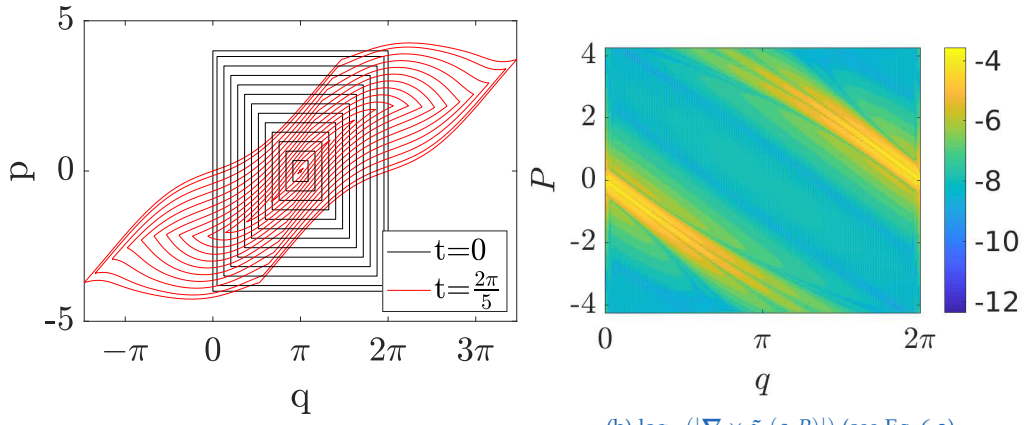
the mean value of the energy oscillation in both cases is nearly the same ($\Delta H \sim (0 \pm 0.02)$), the quality difference is reflected in the mean values of the geometrical distance: (0.0005 ± 0.0005) vs. $(3 \pm 8) \cdot 10^{-6}$. One may also note that the geometrical distance varies only locally in the case of collocation, whereas the line integration method creates regions with better mapping quality and regions with worse quality.

6.5 Line integration: parameter study

Finally one wants to know how the parameters t_M and N_g affect the mapping precision of the line integration method. Therefore, different parameter combinations are tested and the average of the normalized maximal energy oscillation as well as the geometrical distance are calculated.

Beforehand, the limits of both parameters are defined. Here, the lower limit for t_M is set to $t_M = \frac{\tau_b}{15}$ due to computation time and matrix sizes.

Concerning t_M, the upper limit refers to symplecticity and injectivity of the mapping functions. To make the statement clear area conservation and the curl given in Eq. 3.7 are observed for the cases $t_M = \frac{\tau_b}{3}$, $t_M = \frac{\tau_b}{4}$ and $t_M = \frac{\tau_b}{5}$ using a grid size of $N_g = 128$. In addition injectivity is examined qualitatively.



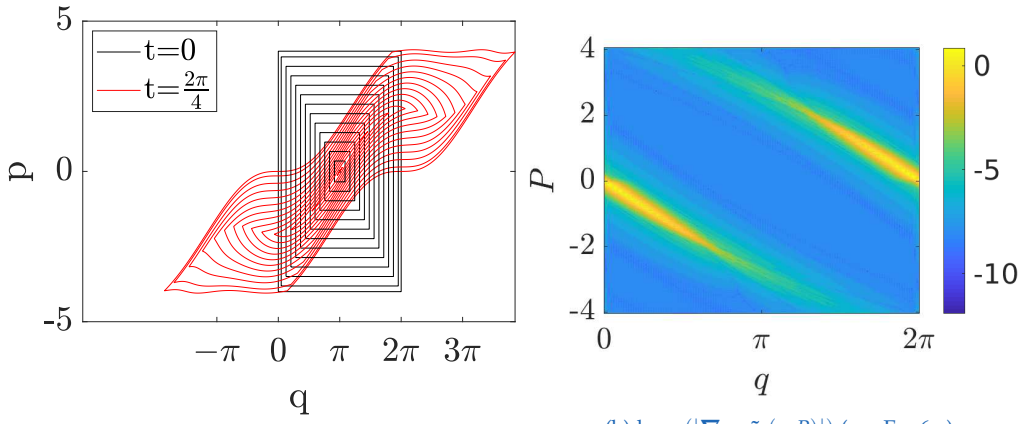
(a) Time evolution in phase space

(b) $\log_{10}(|\nabla \times \tilde{\gamma}_i(q,P)|)$ (see Eq. 6.3)

Figure 6.10: Mapping from (q,p) to (Q,P) for $t_M = \frac{\tau_b}{5}$ and $N_g = 128$

Area conservation in phase space is measured by the ratio of initial area (black rectangles in Fig. 6.10a, Fig. 6.11a and Fig. 6.12a) and time evoluted area (red shapes). It gives values close to one up to five digits in all three cases. Additionally, lines defining the shape in phase space do not cross each other, so the transformation is still symplectic.

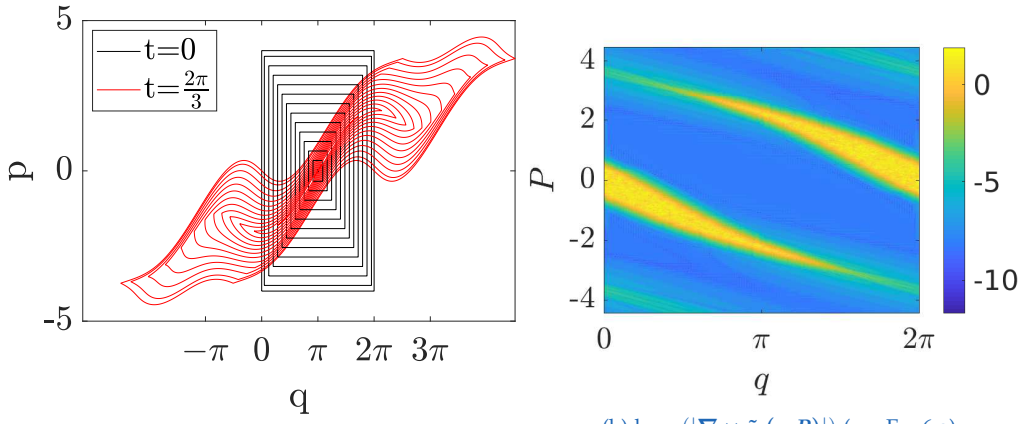
Turning to the mapping condition Eq. 3.7, which is close to zero if the mapping was symplectic and injective, one encounters the following situation:



(a) Time evolution in phase space

(b) $\log_{10}(|\nabla \times \tilde{\gamma}_i(q,P)|)$ (see Eq. 6.3)

Figure 6.11: Mapping from (q,p) to (Q,P) for $t_M = \frac{\tau_b}{4}$, $N_g = 132$



(a) time evolution in phase space

(b) $\log_{10}(|\nabla \times \tilde{\gamma}_i(\boldsymbol{q},\boldsymbol{P})|)$ (see Eq. 6.3)

Figure 6.12: mapping from (q,p) to (Q,P) for $t_M = \frac{\tau_b}{3}$, $N_g = 128$

In Fig. 6.11b one may observe that the curl grows to values close to one (for $t_M = \frac{\tau_b}{4}$) in the central mapping region, where the mapping area cannot be reduced. In case of $t_M = \frac{\tau_b}{3}$, the curl obviously exceeds one (see black framed yellow dots in the orange region of Fig. 6.12b).

As area conservation is fulfilled and lines do not cross each other, the only remaining problem is injectivity. A two-dimensional function has to be injective on one dimensional isolines in order to be injective as a whole.

Exemplary, the mapping function $\tilde{\gamma}_{1,i}(q, P)$ is observed at a fixed q value to explain the high curl values in the figures above. For $t_M = \frac{\tau_b}{5}$ the function is injective (see Fig. 6.13c). $t_M = \frac{\tau_b}{4}$ represents the limiting case, where $P = 0$ may have more than one solution (see Fig. 6.13b) and $t_M = \frac{\tau_b}{3}$ is definitely not injective (see Fig. 6.13a).

As it is impossible to reach global curl values below the limitation 10^{-2} for $t_M = \frac{\tau_b}{4}$ and $t_M = \frac{\tau_b}{3}$ with the maximal grid size of $N_g = 256$, the upper limit for the time one mapping involves is $t_M = \frac{\tau_b}{5}$.

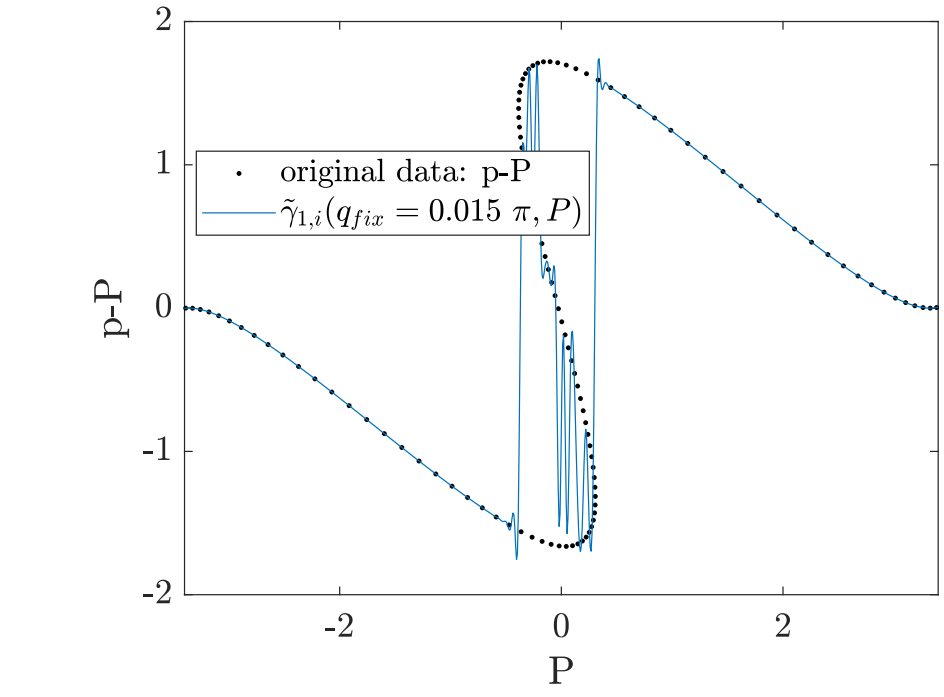
Considering the second parameter N_g, the upper limit is set to $N_g = 256$ because of memory and computation time. The lower limit for N_g is set to $N_g = 32$, where the curl defined in Eq. 6.3 does not fall below the limit for all considered cases of t_M.

Finally, knowing the limits, the logarithm of the geometrical distance and the normalized energy oscillation (described in the previous section) is determined for all valid parameter combinations $t_M = \frac{\tau_b}{5}, \frac{\tau_b}{6}, \frac{\tau_b}{8}, \frac{\tau_b}{10}, \frac{\tau_b}{12}, \frac{\tau_b}{15}$ and $N_g = 32, 64, 80, 96, 112, 128, 256$. The natural logarithm of the average geometrical distance and the normalized energy oscillations are depicted in Fig. 6.14.

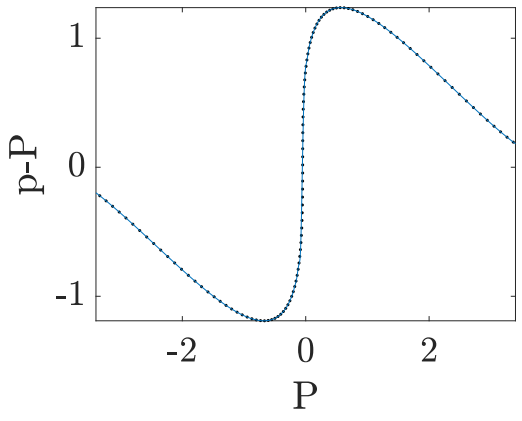
While the normalized energy oscillation is converging towards the same value for all map times t_M, the averaged geometrical distance improves by reducing the mapping time. This can be explained by the properties of the interpolated mapping functions γ_i, which are more exact and more 'symplectic' for smaller mapping times. Reducing t_M one should keep in mind the increasing computing time.

Besides, the mapping precision is quite sensitive to the grid size. Choosing insufficient points to interpolate the mapping functions naturally leads to less accuracy in the results. Choosing too many points, in contrast, implicates over fitting. The smaller the mapping time-step t_M, the smoother and less contorted is the data to interpolate $(Q - q)$ and $(p - P)$. Smoothness in the case of $k = 3$ favors overfitting, so that the optimal grid size is shifted to smaller values for smaller mapping time steps t_M.

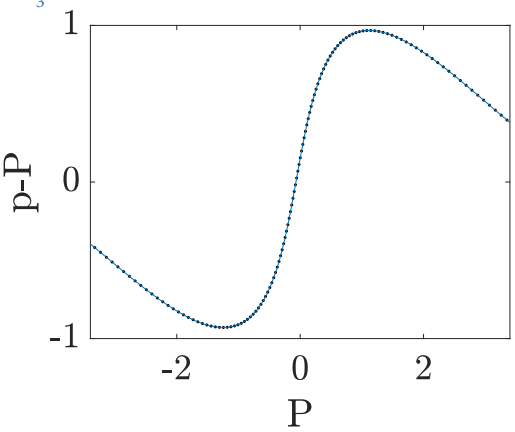
Here, it is important to note that N_g is the number of grid points in q direction of the mapping functions. In P, one started with $2N_g$ grid points,



(a) $t_M = \frac{\tau_b}{3}$



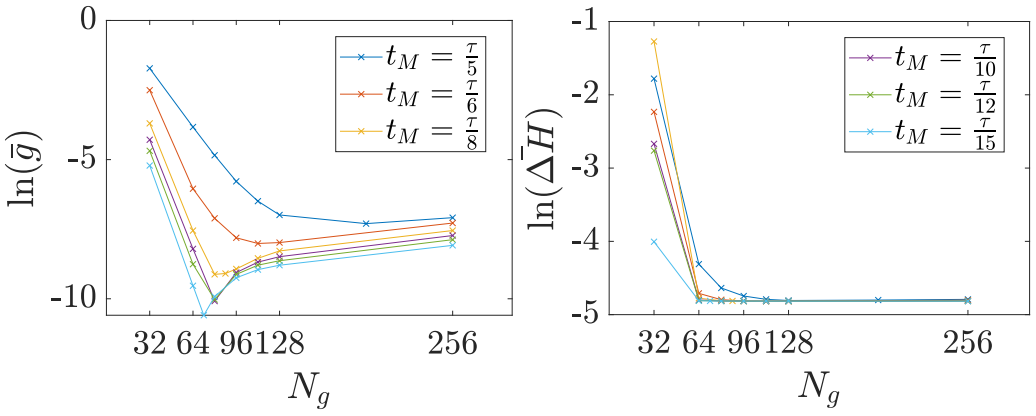
(b) $t_M = \frac{\tau_b}{4}$



(c) $t_M = \frac{\tau_b}{5}$

Figure 6.13: $P \rightarrow p$ relation at constant $q = \frac{\pi}{10}$

where some of them where cut off in order to represent valid mapping. This leads to approximately N_g grid points in P direction for the mapping area. Allowing the number of grid points to vary in both directions q and P could



(a) $\ln(\bar{g})$ (b) $\ln(|\bar{\Delta H}|)$

Figure 6.14: natural logarithm of the averaged (a) geometrical distance and (b) magnitude of the normalized energy oscillation for different parameter combinations t_M and N_g

further improve the mapping results here.

7 Application to the Tokamak Model Field

Symplectic mappings are tested for a non-canonical case with trajectories in the tokamak model field resulting from the guiding-center Lagrangian described in chapter 2.

First, in section. 7.1 further details about the construction of initial conditions are given.

Then, in section 7.2, 7.3 and 7.4, a certain parameter triple t_M, N_g and k is chosen to show how map construction, application and quality measure works.

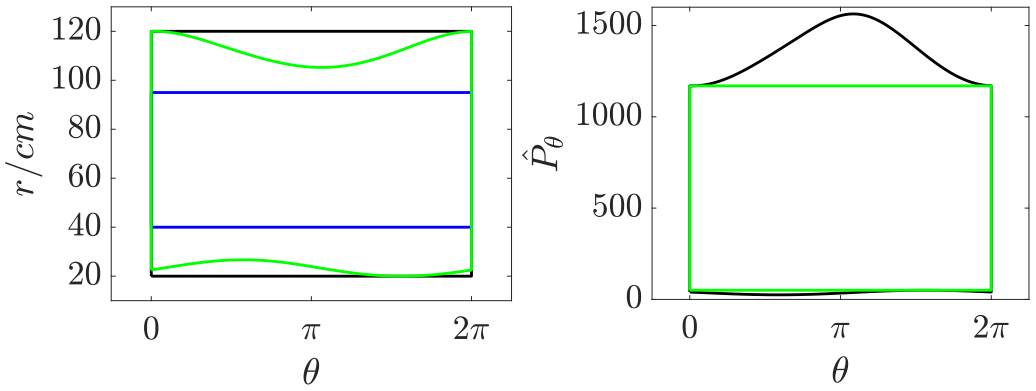
One will see, that in the model field case only collocation results in stable mappings. Finally, in section 7.5, different parameter combinations are tested and analyzed for the mentioned method.

7.1 Choice of initial conditions

The mapping is done for the one-dimensional, canonicalized tokamak system in the subspace $\hat{p}_\phi = -400$ and $\hat{\mu} = 5 \cdot 10^6\ Gs^{-1}$. In order to create initial conditions $(\theta, \Theta, \hat{p}_\theta, \hat{P}_\theta)$ for map interpolation, the equations of motion (see Eq. 2.6) are integrated following the description (and using the parameters) given in section 2.6.

Primarily, initial conditions for integration are defined on a (θ, r)-grid, depicted in Fig. 7.1a in black. N_g grid points are used in each direction. The corresponding $\hat{v}_{||}$ and $\hat{p}_\theta$ are given by Eq. 2.13 and Eq. 2.14. The triple

$(r, \theta, \hat{v}_{||})$ is integrated on the given grid up to the mapping time t_M, resulting in $R, \Theta, \hat{V}_{||}$. Again, $\hat{P}_\theta$ is evaluated by Eq. 2.14. Initial conditions for map interpolation in mixed variable phase space $(\theta, \hat{P}_\theta)$ are shown in Fig 7.1b in black.



(a) initial space θ-r

(b) mixed variable phase space θ-$\hat{P}_\theta$

Figure 7.1: Provided data (black), surjective mapping area (green) and test area (blue) in 'real' space (a) and mixed variable phase space (b)

In analogy to the harmonic oscillator, the area where the mapping can be applied to is smaller than the original area covered by initial conditions $(\theta, \Theta, \hat{p}_\theta, \hat{P}_\theta)$ because the interpolation has to be surjective. Here, this is illustrated for the special parameter triple: $N_g = 64$, $t_M = \frac{1}{10}\hat{\tau}_{passing}$ ($\hat{\tau}_{passing} = 54.65\ m$), $k = 4$ in Fig. 7.1 in green. To guarantee surjectivity for different parameter triples, the test area, where the map is finally applied to, is even smaller (see blue rectangle in Fig. 7.1a).

Trajectories of particles initiated in the test area are shown in Fig 7.2.

7.2 Map creation

To exemplify map construction the parameter triple: $N_g = 64$, $t_M = \frac{1}{10}\hat{\tau}_{passing}$, $k = 4$, is considered. In this context, k stands for the order of the partial derivative of a B-spline basis function describing $\tilde{G}_a(\theta, \hat{P}_\theta)$ or the

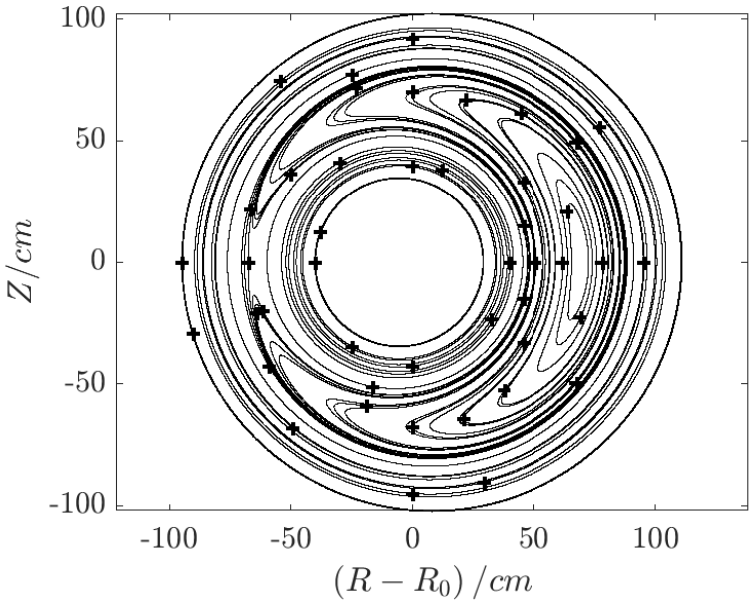


Figure 7.2: Particle trajectories for initial conditions marked by +: some initial conditions are randomly selected to keep the representation clear.

order of the interpolated mapping function $\tilde{\gamma}_i(\theta, \hat{P}_\theta)$. The canonicalized initial conditions $(\theta, \Theta, \hat{p}_\theta, \hat{P}_\theta)$ are taken to create a map via the line integration method on the one hand, and via collocation on the other hand.

7.2.1 Map creation by line integration

Essentially, the map is constructed following the description given in section 5.1. In analogy to the pendulum a first impression of the quality of the map is given by the curl of the mapping functions (see Eq. 6.3). The logarithm of its magnitude is depicted in Fig 7.3.

One should remember that the mapping is symplectic only if the curl is close to zero everywhere. In Fig. 7.3 one can see, that it exceeds 10^{-2}, which is a sign that symplecticity and stability will not be guaranteed in the long run.

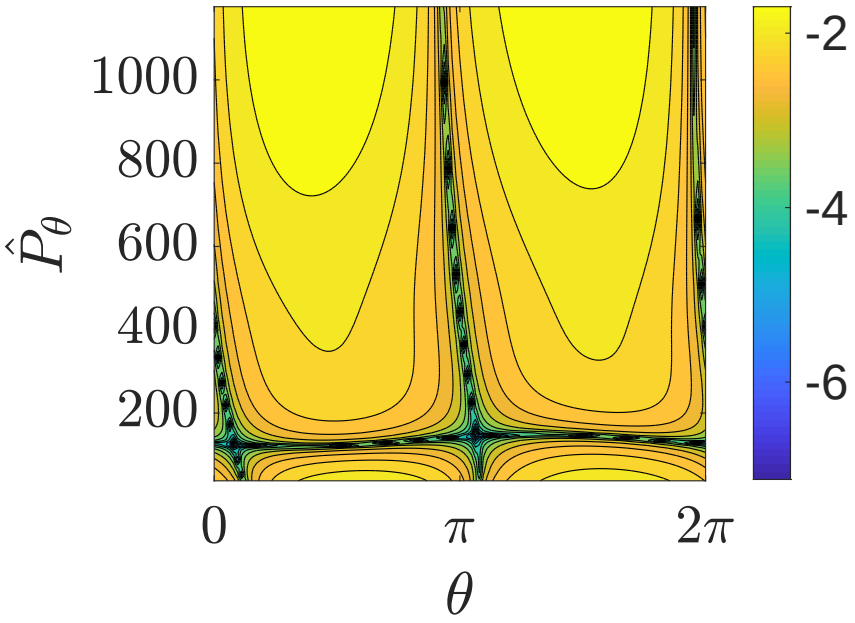


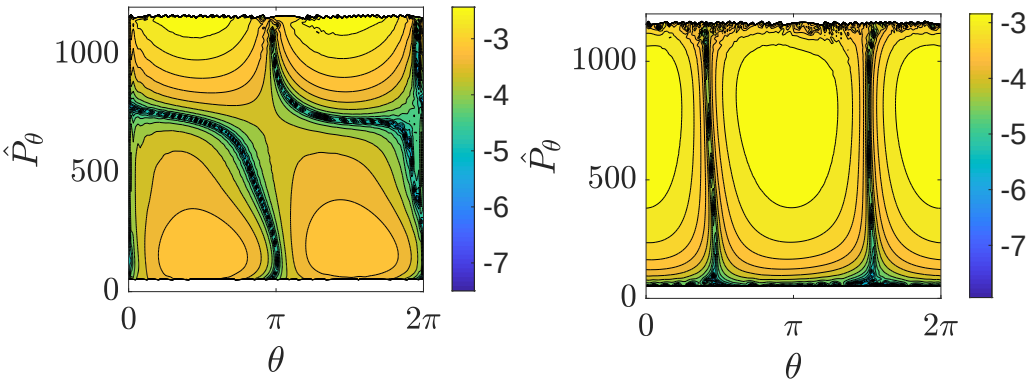
Figure 7.3: Line integration: approximation of map exactness by analyzing symplecticity of interpolated mapping functions: $\log_{10}(|\nabla \times \tilde{\gamma}_i(\theta, \hat{P}_\theta)|)$ (see Eq. 6.3) for $t_M = \frac{1}{10}\ \hat{\tau}_{passing}$, $N_g = 64$ and $k = 4$

The initial conditions involve particles carrying energies from about $\hat{H} = 0.2$ to $\hat{H} = 1.3$ (in terms of 7 MeV) (see Fig. 2.7b). The sensitivity to the integration time-step varies from orbit to orbit. Whereas original integration was done by an adaptive step-size Runge Kutta method, the mapping is done with the same step-size for all trajectories. This may lead to difficulties constructing such a map. In addition, the momenta $\hat{p}_\theta$ and $\hat{P}_\theta$ used as inital conditions for the map construction depend on integration variables $\hat{v}_{||}$,θ and r up to the forth power. Small deviations in the integration variables may lead to higher deviations in the mapping functions.

7.2.2 Map creation by collocation

Here, the map is constructed following the instructions given in section 5.2. The momenta are normalized, so that the least squares approximation is done on a unique square of $2\pi \times 2\pi$.

Considering collocation, the curl given in Eq. 5.8 vanishes anyway. So, the accuracy of the map is measured by analyzing the normalized differences of the evaluated mapping functions and the exact values, which is displaced in Fig. 7.4.The variation is in the same range for both coordinates. Regardless, one should consider that the orbits may not be reproduced exactly.



(a) $\log_{10}\left(\left|\frac{\tilde{\gamma}_{1,a}(\theta,\hat{P}_\theta)-\hat{p}_\theta+\hat{P}_\theta}{\hat{P}_\theta^{\max}}\right|\right)$ (b) $\log_{10}\left(\left|\frac{\tilde{\gamma}_{2,a}(\theta,\hat{P}_\theta)-\Theta+\theta}{2\pi}\right|\right)$

Figure 7.4: Collocation: measure of map accuracy by analyzing the logarithm of the normalized differences between the evaluated mapping function and the exact values for $\tilde{\gamma}_{1,a}(\theta,\hat{P}_\theta)$ in (a) and $\tilde{\gamma}_{2,a}(\theta,\hat{P}_\theta)$ in (b)

In principle, the deviations shown in Fig. 7.4 can be related to the curl illustrated in Fig. 7.3. γ is a gradient field (defined in Eq. 3.6), whose exact primitive will only exist if it was curl free. Applying the collocation method, the generating function is constructed in a way that the curl vanishes, whereas the gradient field γ_a slightly deviates from the given data.

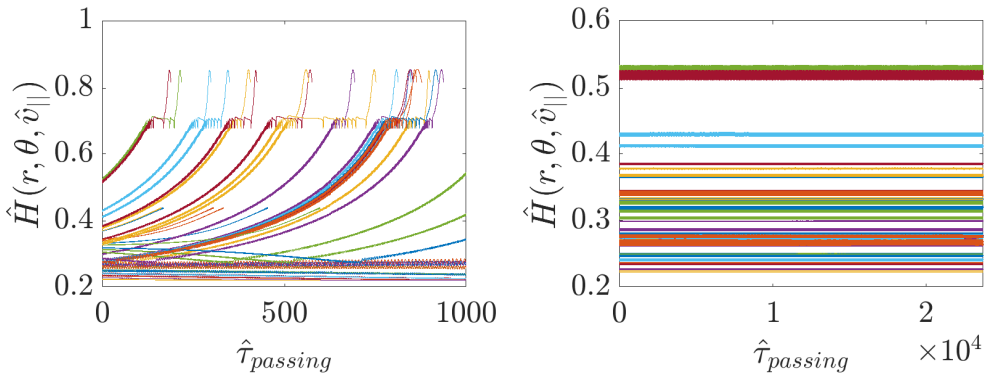
7.3 Application of the map

Here, the map is applied to the test particles. These originate from a grid covering the area marked in Fig. 7.1a in blue with 21×21 points.

The mapping is done for a time interval of 0.1 s, which approximately corresponds to the slowing down time. From another point of view, the mapping interval accounts for 23722 $\hat{\tau}_{passing}$, where $\tau_{passing} = \hat{\tau}_{passing} v =$

$4.2155 \cdot 10^{-6}$ s. Considering the mapping time $t_M = \frac{1}{10}\hat{\tau}_{passing}$, there are 237220 mapping steps in total.

Finally, mapping functions originating from both methods are applied to the test points following the guidelines given in section 5.3. The coordinate r is implicitly given by Eq. 2.14 and $\hat{v}_{||}$ is defined by Eq. 2.13. The energy of some particles is calculated at each time step and its conservation is observed for both methods (see Fig. 7.5). As expected, the mapping fails using the Line Integration method. However, collocation conserves energy very well. Regardless, energy may oscillate, which is the reason why the lines in Fig. 7.5b appear thicker.



(a) Line integration Method

(b) Collocation

Figure 7.5: Conservation of energy from mapping originated from (a) and (b), respectively. Note the difference in the time axes.

Particle trajectories resulting from the mapping originated from collocation are displayed in Fig. 7.6. At a first glance, the orbits are like the ones originated from conventional integration depicted in Fig. 7.2. Closely observing the separatrix, one may note differences.

7.4 Quality of the map

In this section the quality of the mapping originated from collocation is analysed. Both, geometrical distance and normalized energy oscillation are

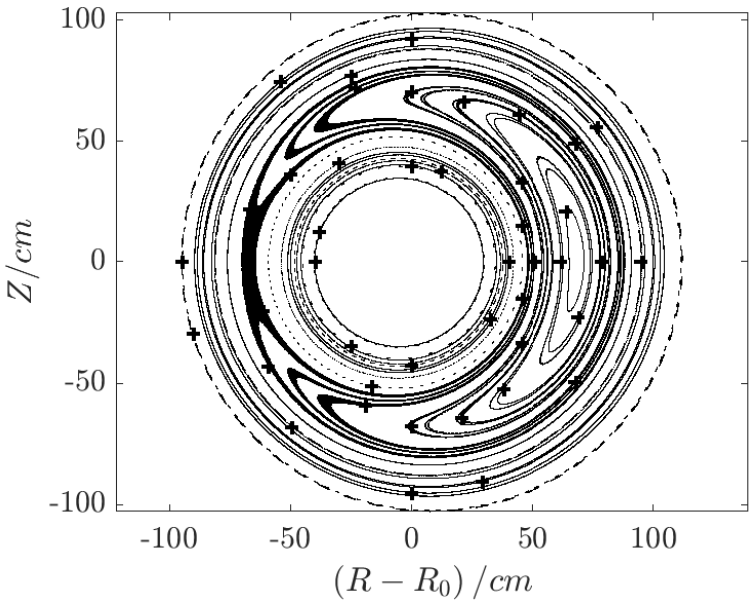
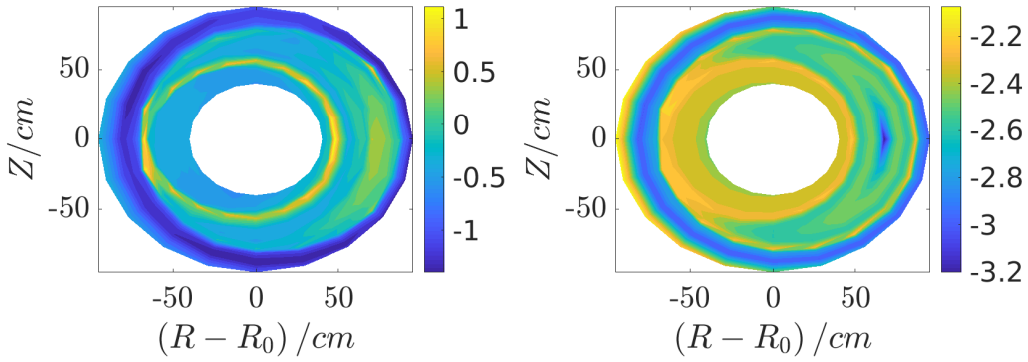


Figure 7.6: Collocation: Mapping over 0.1 s with parameters: $t_M = \frac{1}{10}\hat{\tau}_{passing}$, $N_g = 64$, $k = 4$. Initial conditions are marked by +. Note the difference to the integrated orbits in Fig. 7.2 at the separatrix.

calculated for each test point. The logarithm of its magnitude is illustrated in Fig. 7.7.

For the calculation of the geometrical distance, $T_N = 1000$ points of each mapped orbit are chosen randomly. For each point the normal distance g_{N_i} is computed evaluating Eq. 5.12 for the two closest out of $C_N = 50000$ reference points originated from conventional integration. Then the geometrical distance is given by the root mean square of the normal distances (see Eq. 5.13).

In Fig. 7.7, one notes that the accuracy of the mapping depends strongly on its initial conditions. Large geometrical distances and normalized energy oscillations are observed at the separatrix, where trapped orbits are separated from passing ones. Describing this distinct boundary by continuous functions will naturally fail, so at the separatrix the map is partly invalid, or rather the trajectories are shifted. Detailed and scientific explanations



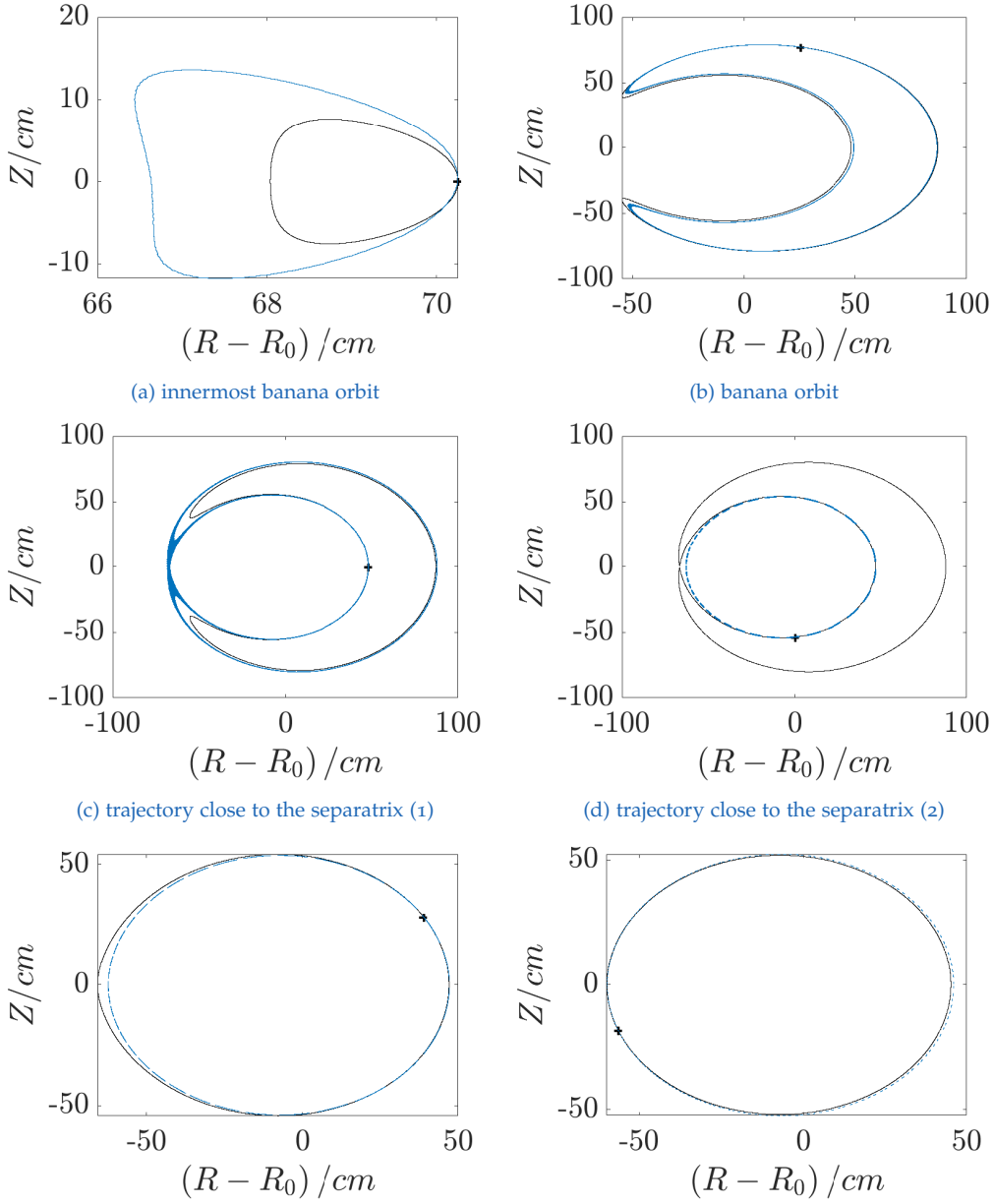
(a) $\log_{10}(g)$ (b) $\log_{10}(|\Delta H|)$

Figure 7.7: Collocation: Mapping quality for parameter triple $N_g = 64$, $t_M = \frac{1}{10}\hat{\tau}_{passing}$, $k = 4$ with a total mapping time of $t = 0.1\ s$. Logarithm of (a) geometrical distance (Eq. 5.13) and (b) magnitude of the normalized energy oscillation (Eq. 5.14)

are given by Lichtenberg and Lieberman, 1992. The behavior is similar to the behaviour of the trajectories of the pendulum described in the previous chapter.

Critical trajectories with large geometrical distances at the separatrix are illustrated in Fig. 7.8c, Fig. 7.8d, and Fig. 7.8e. In comparison, a 'common' banana and a 'common' passing trajectory are displayed in Fig. 7.8b and Fig. 7.8f respectively. As the geometrical distance of innermost bananas is increased as well (see Fig. 7.7a), the innermost banana is depicted in Fig. 7.8a.

In addition, the normalized energy oscillations are high for outermost orbits, where particle energies are large.



(a) innermost banana orbit

(b) banana orbit

(c) trajectory close to the separatrix (1)

(d) trajectory close to the separatrix (2)

(e) trajectory close to the separatrix (3)

(f) passing oribit

Figure 7.8: Particle trajectories (a),(b), (c),(d), (e) and (f) from the inside to the outside: black lines corresponds to 'conventionally' integrated orbits, whereas blue lines originates from the mapping ($t_M = \frac{1}{10}\hat{\tau}_{passing}$, $N_g = 64$, $k = 4$); initial conditions are marked by +

7.5 Collocation: parameter study

In this section, it is examined how the parameter pair N_g and t_M affects the quality of the mappings constructed by collocation for $k = 4$. As above, the mapping is applied to 441 test points covering the blue area given in Fig. 7.1a. To keep the computing time within acceptable limits it is done over 10^4 $\hat{\tau}_{passing}$.

Least squares approximation demands more memory than direct B-spline interpolation, therefore the upper limit for the grid size is set to $N_g = 96$ here. The lower limit is determined to be $N_g = 8$. The Matlab internal least squares solver *lsqlin* can be regulated either by tolerances or iterations. Here, it is adjusted to iterate a thousand times. The bigger the approximation grid, the slower the convergence. For that reason, the iteration number is increased to 3000 for $N_g = 96$.

Regarding the mapping time t_M, its upper limit is given by mapping stability. For $t_M = \frac{1}{3}\hat{\tau}_{passing}$, the inner- and outermost orbits start to leave the area where mapping is valid.

For the given point with $t_M = \frac{1}{3}\hat{\tau}_{passing}$ and $N_g = 64$, the line integration method is applied to measure the curl given by Eq. 6.3. It is illustrated in Fig. 7.9 and should be compared to Fig. 7.3, where one notes that the values are a little bit larger for $t_M = \frac{1}{3}\hat{\tau}_{passing}$. Generally, the higher the mapping time is chosen, the higher the curl values grow and the more the corresponding trajectories are distorted. In contrast to the previous chapter where the limit for the mapping time was determined by injectivity, which was accompanied by sudden rises in the curl value, here the curl values are high from the beginning and the mapping is still injective for slightly unstable mappings. So, the limit referring to the upper mapping time is not so obvious.

From that point of view, $t_M = \frac{1}{3}\ \hat{\tau}_{passing}$ is chosen as an upper limit because the mapping failed for one to four out of $N_t^2 = 441$ test particles. In addition, the lower limit is set to $t_M = \frac{1}{20}\hat{\tau}_{passing}$ due to computation time and memory.

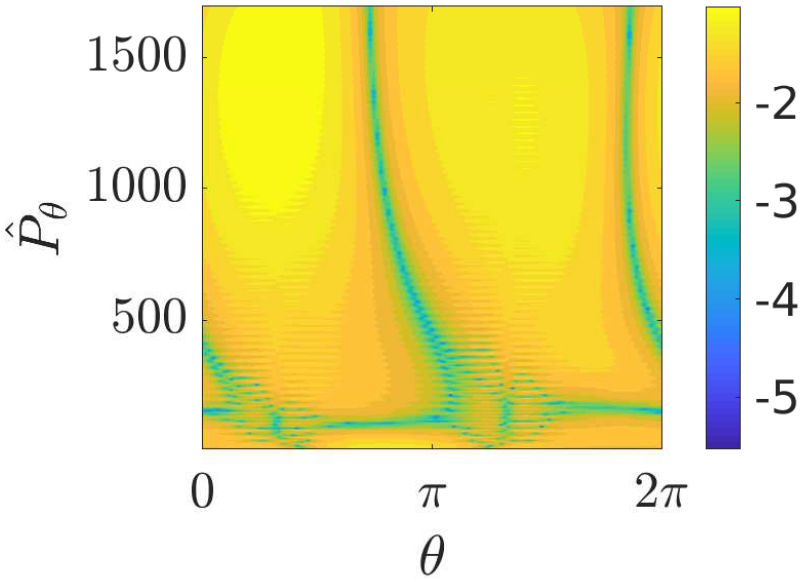


Figure 7.9: Line integration method: logarithm of the curl given by Eq. 6.3: $\log_{10}(|\nabla \times \gamma_i(\theta, \hat{P}_\theta)|)$; parameter combination $t_M = \frac{1}{3}\hat{\tau}_{passing}$ and $N_g = 64$.

In practice the mapping is done for the following parameter combinations: $N_g = 8, 16, 32, 64, 96$ and $t_M = \frac{1}{3}, \frac{1}{4}, \frac{1}{5}, \frac{1}{6}, \frac{1}{8}, \frac{1}{10}, \frac{1}{12}, \frac{1}{15}, \frac{1}{20}$ multiples of $\hat{\tau}_{passing}$.

For each parameter combination the averaged normalized energy oscillation (given by Eq. 5.14) is calculated and its natural logarithm plotted in Fig. 7.10. The lower the mapping time the exacter the solutions.

As least squares approximation is used, over fitting in the original sense is not a problem any more. For high mapping times t_M, the complexity of the system is increased and the convergence may be slowed down. This could explain the minima of the averaged energy oscillations at lower grid sizes for the red and blue line in Fig. 7.10.

Observing shapes of mapped trajectories, the scattering reduces by decreasing t_M and increasing N_g corresponding to the decrease in averaged normalized energy oscillations given in Fig. 7.10. To underline this state-

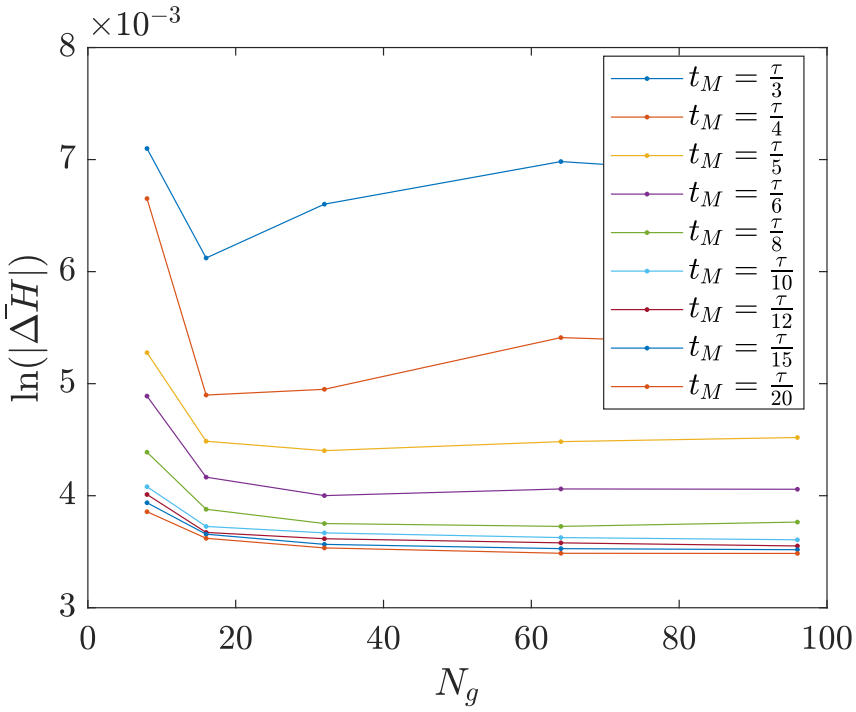
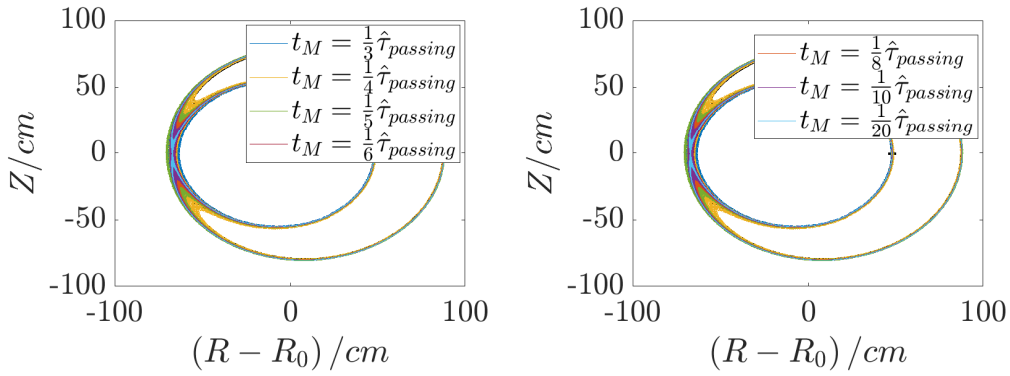


Figure 7.10: natural logarithm of normalized energy oscillations $\ln(|\Delta\bar{H}|)$ for different parameter combinations N_g and t_M. τ in the legend replaces $\hat{\tau}_{passing}$.

ment two particular trajectories are chosen and the mapped shapes are illustrated.

Trajectories close to the separatrix are depicted for a fixed grid size $N_g = 32$ and varying t_M in Fig. 7.11 and fixed t_M with varying grid sizes N_g in Fig. 7.12, respectively. One can see that orbits get less scattered and better resolved. Keep in mind that this does not necessarily mean that they are approaching to the shape originated from direct integration. These shifted orbits may be related to curl values (e. g. in Fig. 7.3 and Fig. 7.9), which do not vanish in any case. As the interpolated mapping function γ_i is not curl free it is impossible to find an exact primitive G_a. In that sense, collocation leads to slightly shifted orbits while preserving symplecticity and energy. These shifts especially occur close to the separatrix.

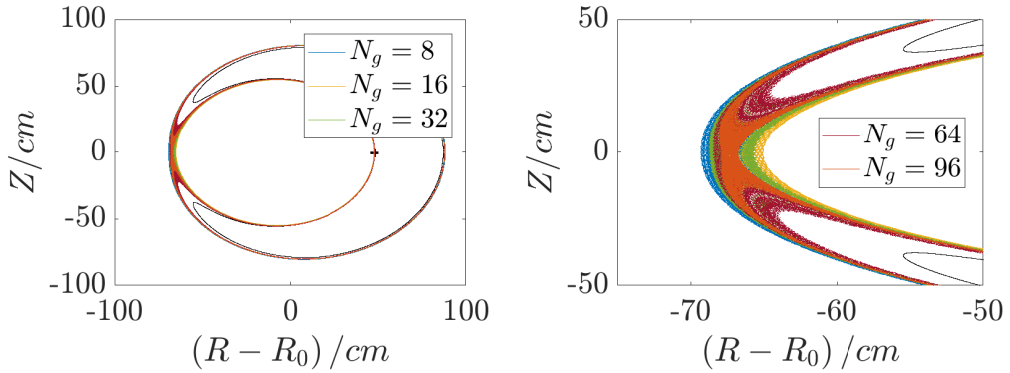
In addition, trajectories of a common banana orbit are depicted for a fixed mapping time t_M with varying grid sizes N_g in Fig. 7.13, and fixed grid size



(a) Separatrix

(b) Zoom into separatrix

Figure 7.11: Varying map time t_M for fixed grid size $N_g = 32$, reference shape originated from integration is marked in black

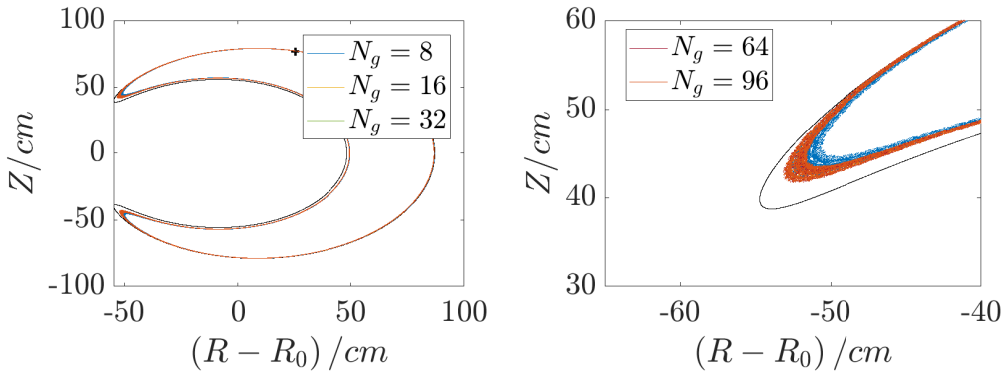


(a) Separatrix

(b) Zoom into separatrix

Figure 7.12: Varying gridsize N_g for fixed $t_M = \frac{1}{10}\ \hat{\tau}_{passing}$, reference shape originated from integration is marked in black

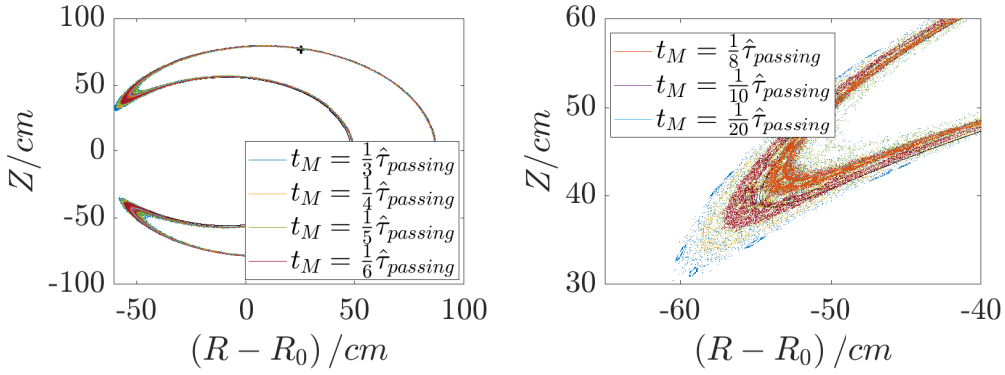
$N_g = 32$ with varying t_M in Fig. 7.14 respectively. Similarily, orbits get less scattered, but not necessarily closer to the integrated orbit line by increasing grid sizes and reducing mapping times.



(a) Banana

(b) Zoom into banana tip

Figure 7.13: Varying gridsize N_g for fixed $t_M = \frac{1}{10}\,\hat{\tau}_{passing}$, reference shape originated from integration is marked in black



(a) Banana

(b) Zoom into banana tip

Figure 7.14: Varying map time t_M for fixed grid size $N_g = 32$, reference shape originated from integration is marked in black

8 Summary and Outlook

In this work, symplectic mappings were applied to trace guiding-center orbits in fusion devices. It was shown that a stable mapping can be constructed for a tokamak model field using the collocation method. The used parameters were inspired by ITER.

Up to now, the mapping has been done for a fixed magnetic moment μ and a fixed toroidal moment p_ϕ leading to varying energies in the range of fast ion orbits originating from fusion. This was necessary to create a system where all particles obey to the same Hamiltonian function. Moreover, this system could be canonicalized and the mapping reduced to one dimension.

Generally, in the next step, the mapping will have to be extended to two dimensions in order to enable mappings for non-axisymmetric devices like stellarators or tokamaks with RMP coils to mitigate ELMs. It is already known how such systems can be canonicalized (Albert, Kasilov, and Kernbichler, 2020).

In addition, one has to keep in mind that the basic idea was to create mappings for predefined constant energies. Maybe one can transform the system or find a way to do the mapping on hypersurfaces of constant energy only.

High energies including trapped and passing orbits, complicated the mapping procedure and lead to small deviations of the mapped trajectories when compared to direct integration, especially at the separatrix. Another idea to improve the mapping quality would be to separate trapped and passing regions and construct the mappings independently for each region.

Finally, discrete B-spline interpolation (or approximation) used in this work could be replaced by mesh-less methods like radial basis functions. Due to

the regular interpolation grid, the area where the map can be applied to was restricted.

The big advantage of the symplectic mappings is, that once the map is created, evaluations of the magnetic field are avoided. Also, in calculating particle trajectories, there are less mapping steps needed than integration steps (Runge-Kutta, *ode45*). Nevertheless, one has to be careful with approximations concerning the improvement of computation time, because finally, when the mapping is applied to non-axisymmetric cases, the approximation of the map has to be done in four- or even five-dimensional space. This leads to an increase in memory demand and computation time. In contrast, the description of the magnetic field is given in three dimensions. In conclusion, the gain in computation time in real applications has to be still investigated.

Bibliography

Albert, C. G. (2017). "Hamilton Approach to Resonant Transport Regimes due to Non-Axisymmetric Magnetic Perturbations in Tokamak Plasmas." PhD thesis. TU Graz (cit. on pp. 18, 49).

Albert, C. G., S. V. Kasilov, and W. Kernbichler (Feb. 2020). "Symplectic integrators with non-canonical quadrature for guiding-center orbits in magnetic confinement devices." In: *Journal of computational physics* 403.4 (cit. on pp. 7, 11, 77).

D'haeseleer, W. D. et al. (1991). *Flux Coordinates and Magnetic Field Structure*. Springer Verlag (cit. on p. 6).

Goldstein, Herbert (1963). *Klassische Mechanik*. Frankfurt am Main: Akademische Verlagsgesellschaft (cit. on p. 19).

Hairer, E., C. Lubich, and G. Wanner (2006). *Geometric numerical integration*. Berlin: Springer Verlag (cit. on pp. 22, 23).

Helander, P. and D. J. Sigmar (2002). *Collisional Transport in Magnetized Plasmas*. Cambridge University Press (cit. on p. 4).

Heppler, G. R., A. H. Vermeulen, and R. H. Bartels (July 1992). "Integrating Products of B-Splines." In: *SIAM Journal on Scientific and Statistical Computing* 13.4, pp. 1025–1038 (cit. on p. 29).

Lichtenberg, A. J. and M. A. Lieberman (1992). Springer Verlag (cit. on p. 70).

Littlejohn, R. G. (June 1983). "Variational principles of guiding centre motion." In: *J. Plasma Physics* 29.1, pp. 111–125 (cit. on p. 5).

Mineault, Patrick (2011). *Fast B-spline class*. fileexchange (cit. on p. 29).

Piegl, L. and W. Tiller (1997). *The NURBS book*. 2. Berlin: Springer Verlag. Chap. 9 (cit. on p. 32).

S. Putvinski, et al. (1999). "Alpha-Particle Physics in Tokamaks [and Discussion]." In: *Philosophical Transactions: Mathematical, Physical and Engineering Sciences* 357.1752 (cit. on p. 1).

Warnock, R. L., Y. Cai, and J. A. Ellison (2009). "Construction of large-period symplectic Maps by interpolative Methods." In: *10th International Computational Accelerator Physics Conference (ICAP)* (cit. on pp. 2, 40).

Warnock, R. L. and J. A. Ellison (1997). "Convergence of a Fourier-Spline Representation for the Full-turn Map generator." In: *AIP Conference Proceedings* (cit. on p. 22).

Warnock, R. L. and J. A. Ellison (1999). "From symplectic integrator to Poincare map: spline expansion of a map generator in Cartesian coordinates." In: *Applied Numerical Mathematics* 29 (cit. on p. 40).