

FAST NUMERICAL ALGORITHMS FOR THE COMPUTATION OF INVARIANT TORI IN HAMILTONIAN SYSTEMS

GEMMA HUGUET, RAFAEL DE LA LLAVE AND YANNICK SIRE

ABSTRACT. In this paper, we develop numerical algorithms that use small requirements of storage and operations for the computation of invariant tori in Hamiltonian systems (exact symplectic maps and Hamiltonian vector fields). The algorithms are based on the parameterization method and follow closely the proof of the KAM theorem given in [LGJV05] and [FLS07]. They essentially consist in solving a functional equation satisfied by the invariant tori by using a Newton method. Using some geometric identities, it is possible to perform a Newton step using little storage and few operations.

In this paper we focus on the numerical issues of the algorithms (speed, storage and stability) and we refer to the mentioned papers for the rigorous results. We show how to compute efficiently both maximal invariant tori and whiskered tori, together with the associated invariant stable and unstable manifolds of whiskered tori.

Moreover, we present fast algorithms for the iteration of the quasi-periodic cocycles and the computation of the invariant bundles, which is a preliminary step for the computation of invariant whiskered tori. Since quasi-periodic cocycles appear in other contexts, this section may be of independent interest.

The numerical methods presented here allow to compute in a unified way primary and secondary invariant KAM tori. Secondary tori are invariant tori which can be contracted to a periodic orbit.

We present some preliminary results that ensure that the methods are indeed implementable and fast. We postpone to a future paper optimized implementations and results on the breakdown of invariant tori.

CONTENTS

1. Introduction	3
2. Setup and conventions	7
3. Equations for invariance	8
3.1. Functional equations for invariant tori	8
3.2. Equations for the invariant whiskers	12
3.3. Fourier-Taylor discretization	13

2000 *Mathematics Subject Classification.* Primary: 70K43, Secondary: 37J40 .

Key words and phrases. quasi-periodic solutions, KAM tori, whiskers, quasi-periodic cocycles, numerical computation.

4.	Numerical algorithms to solve the invariance equation for invariant tori	16
4.1.	The large matrix method	17
4.2.	The Newton method and uniqueness	19
5.	Fast Newton methods for Lagrangian tori	20
5.1.	The Newton method for Lagrangian tori in exact symplectic maps	20
5.2.	The Newton method for Lagrangian tori in Hamiltonian flows	24
6.	Fast iteration of cocycles over rotations. Computation of hyperbolic bundles	27
6.1.	Some standard definitions on cocycles	27
6.2.	Hyperbolicity of cocycles	28
6.3.	Equivalence of cocycles, reducibility	29
6.4.	Algorithms for fast iteration of cocycles over rotations	30
6.5.	The “straddle the saddle” phenomenon and preconditioning	32
6.6.	Computation of rank-1 stable and unstable bundles using iteration of cocycles	34
7.	Fast algorithms to solve difference equations with non constant coefficients	35
8.	Fast Newton methods for whiskered isotropic tori	39
8.1.	General strategy of the Newton method	40
8.2.	A Newton method to compute the projections	42
9.	Computation of rank-1 whiskers of an invariant torus	46
9.1.	The order by order method	47
9.2.	A Newton method to compute simultaneously the invariant torus and the whiskers	48
9.3.	A Newton method to compute the whiskers	55
10.	Algorithms to compute rank-1 invariant whiskers for flows	58
10.1.	The order by order method	58
10.2.	A Newton method to compute simultaneously the invariant torus and the whiskers	59
10.3.	A Newton method to compute the whiskers	65
11.	Initial guesses of the iterative methods	68
12.	Numerical Examples	69
12.1.	Computation of primary and secondary KAM tori for the standard map	69
12.2.	4D-symplectic maps: The Froeschlé map	73
	Acknowledgements	77
	References	77

1. INTRODUCTION

The goal of this paper is to describe efficient algorithms to compute invariant manifolds in Hamiltonian systems. The invariant manifolds we are considering are invariant tori such that the motion on them is conjugate to a rigid rotation and their whiskers.

The tori we consider here can have stable and unstable directions. The standard theory [Fen72, HPS77] ensures the existence of invariant manifolds tangent to these spaces. We also consider the computation of these stable and unstable manifolds.

By invariant torus, we mean an invariant manifold topologically equivalent to a power of \mathbb{T} with quasi-periodic dynamics on it and a dimension equal to the number of independent frequencies, which we will be assumed to be *Diophantine*. Invariant tori have been an important object of study since they provide landmarks that organize the long term behavior of the dynamical system. There are several variants of these tori; in this paper we will consider both *maximal tori* and *whiskered tori*.

Tori of maximal dimension are quasi-periodic solutions of n frequencies in Hamiltonian systems with n -degrees of freedom. It is well known that for $n \leq 2$, they provide long term stability. In contrast, whiskered tori are tori with ℓ independent frequencies in systems with n -degrees of freedom. Symplectic geometry asserts that, in the normal direction there is at least an ℓ dimensional family of neutral directions (the variational equations on them grow only polynomially). Whiskered tori are such that there are $n - \ell$ directions which, under the linearized equation contract exponentially in the future (stable directions) or in the past (unstable directions). It is well known that these infinitesimal stable (resp. unstable) directions lead to stable (resp. unstable) manifolds consisting on the points that converge exponentially fast in the future (resp. in the past) to the torus. The persistence of these manifolds under a perturbation has been widely studied (see [Fen72, Fen74, Fen77, HPS77, Pes04]). Note that the whiskered tori are not normally hyperbolic manifolds since there are, at least, ℓ neutral directions. The persistence of whiskered tori with a Diophantine rotation has been studied in [Gra74, Zeh75, LY05, FLS07].

The whiskered tori and their invariant manifolds organize the long term behavior and provide routes which lead to large scale instability. Indeed, the well known paper [Arn64] showed that, in some particular example, one can use the heteroclinic intersections among these manifolds to produce large scale motions. In [Arn63] this is conjectured as a generic mechanism. The transitions between different kinds of whiskered invariant tori have been the basis of many of the theoretical models of Arnol'd diffusion [DLS06, DH08].

Invariant objects including tori play also an important role in several areas of applied sciences, such as astrodynamics and theoretical chemistry. In the monographs [Sim99, GJSM01b, GJSM01a], it is shown that computing these invariant objects in realistic models of the Solar System provides orbits of practical importance for the design of space missions.

The numerical method we use is based on the parameterization methods introduced in [CFL03b, CFL03c] and the algorithms we present are very similar to the proofs in [FLS07].

The main idea of the method consists in deriving a functional equation satisfied by the parameterization of the invariant manifold and then implement a Newton method to solve it. The parameterization method is well suited for the numerical implementation because it uses functions of the same number of variables as the dimension of the objects that we want to compute, independently of the number of dimensions of the phase space. The main goal of the present paper is to design very efficient numerical algorithms to perform the required Newton step. What we mean by efficiency is that, if the functional equation is discretized using N Fourier coefficients, one Newton step requires only storage of $O(N)$ and takes only $O(N \log N)$ operations. Note that a straightforward implementation of the Newton method (usually referred to as the *large matrix method*) requires to store an $N \times N$ matrix and solve the linear equation, which requires $O(N^3)$ operations. We include a comparison with the standard Newton method in Section 4.1.

For the case of quasi-periodic systems, algorithms with the same features were discussed in [HL06c, HL06b, HL07] and, for some Lagrangian systems (some of which do not admit a Hamiltonian interpretation) in [CL08]. There are other algorithms in the literature.

The papers [JO05, JO08] present and implement calculations of *reducible* tori. This includes tori with normally elliptic directions. The use of reducibility indeed leads to very fast Newton steps, but it still requires the storage of a large matrix. As seen in the examples in [HL07, HL06a], reducibility may fail in a codimension 1 set (in a Cantor set of codimension surfaces). There are other methods which yield fast computations, notably, the *“fractional iteration method”* [Sim00]. We think that it would be very interesting to formalize and justify the fractional iteration method.

One key ingredient in our calculations – and in all subsequent calculations – is the phenomenon of “automatic reducibility.” This phenomenon, which also lies at the basis of the rigorous proofs [LGJV05, Lla01b, FLS07], uses the observation that the preservation of the symplectic structure implies that the Newton equations can be reduced—by explicit changes of variables—to upper triangular difference equations with diagonal constant coefficients. These equations can be solved very

efficiently in Fourier coefficients. The changes of variables are algebraic expressions involving derivatives of the parameterization. We note that derivatives can be fastly computed in Fourier representation whereas the algebraic expressions can be fastly computed in real space representation. Therefore, the algorithm to produce a Newton step consists of a small number of steps, each of which is diagonal either in real space or in Fourier space. Of course, the FFT algorithm allows us to switch from real space to Fourier space in $O(N \log N)$ computations.

We also note that the algorithms mirror very closely the proofs of the KAM theorem. In [LGJV05] and [FLS07] we can find proofs that the algorithms considered here converge to a true solution of the problem provided that the initial approximation solves the invariance equation with good enough accuracy and satisfies some appropriate non-degeneracy conditions. Furthermore, the true solution is close to the approximate one, the distance from the true solution to the unperturbed one being bounded by the error. In numerical analysis this is typically known as *a posteriori estimates* [dLLR91].

It is important to remark that the algorithms that we will present can compute in a unified way both primary and secondary tori. We recall here that *secondary tori* are invariant tori which are contractible to a torus of lower dimension, whereas this is not the case for primary tori. The tori which appear in integrable systems in action-angle variables are always primary. In quasi-integrable systems, the tori which appear through Lindstedt series or other perturbative expansions starting from those of the integrable system are always primary. Secondary tori, however, are generated by resonances. In numerical explorations, secondary tori are very prominent features that have been called “islands”. In [HL00], one can find arguments showing that these solutions are very abundant in systems of coupled oscillators. As an example of the importance of secondary tori we will mention that in the recent paper [DLS06] they constituted the essential object to overcome the “large gap problem” and prove the existence of diffusion. In [DH08], one can find a detailed and deep analysis of these objects.

In this paper, we will mainly discuss algorithms for systems with dynamics described by diffeomorphisms. For systems described through vector fields, we note that, taking time-1 maps, we can reduce the problem with vector fields to a problem with diffeomorphisms. However, in some practical applications, it is convenient to have a direct treatment of the system described by vector fields. For this reason, we include algorithms that are specially designed for flows, in parallel with the algorithms designed for maps.

The paper is organized as follows. In Section 2 we summarize the notions of mechanics and symplectic geometry we will use. In Section 3 we formulate the invariance equations for the objects of interest (invariant tori, invariant bundles and invariant manifolds) and we will present

some generalities about the numerical algorithms. In Section 5 we specify the fast algorithm to compute maximal tori –both primary and secondary– and we compare it with a straightforward Newton method (Section 4).

In Section 6 we present fast algorithms for the iteration of cocycles over rotations and for the calculation of their invariant bundles. The main idea is to use a renormalization algorithm which allows to pass from a cocycle to a longer cocycle. Since quasi-periodic cocycles appear in many other applications, we think that this algorithm may be of independent interest.

The calculation of invariant bundles for cocycles is a preliminary step for the calculation of whiskered invariant tori. Indeed, these algorithms require the computation of the projections over the linear subspaces of the linear cocycle. In Section 8.2 we present an alternative procedure to compute the projections based on a Newton method. Algorithms for whiskered tori are discussed in Section 8.

In Section 9 we discuss fast algorithms to compute rank-1 (un)stable manifolds of whiskered tori. Again, the key point is that taking advantage of the geometry of the problem we can devise algorithms which implement a Newton step without having to store—and much less invert—a large matrix. We first discuss the so-called order by order method, which serves as a comparison with more efficient methods based on the reducibility. We present algorithms that compute at the same time the torus and the whiskers and algorithms that given a torus and the linear space compute the invariant manifold tangent to it. It is clearly possible to extend the method to compute stable and unstable manifolds in general dimensions (or even non-resonant bundles) by a modification of the method. To avoid increasing even more the length of this paper and since interesting examples happen in high dimension, which is hard to do numerically, we postpone this to a future paper.

One remarkable feature of the method discussed here is that it does not require the system to be close to integrable. We only need a good initial guess for the Newton method. Typically, one uses a continuation method starting from an integrable case, where the solutions are trivial and can be computed analytically. However, in the case of secondary KAM tori, which do not exist in the integrable case, one requires other types of methods. In Section 11 we include a discussion of the different possibilities.

Finally, in Section 12 we include examples of the numerical implementation we have carried out. In Section 12.1 we computed maximal invariant tori, both primary and secondary, for the standard maps and in Section 12.2 we computed maximal and hyperbolic invariant tori for the Froeschlé map. We also provide details of storage and running times.

2. SETUP AND CONVENTIONS

We will be working with systems defined on an Euclidean phase space endowed with a symplectic structure. The phase space under consideration will be

$$\mathcal{M} \subset \mathbb{R}^{2d-\ell} \times \mathbb{T}^\ell.$$

We do not assume that the coordinates in the phase space are action-angle variables. Indeed, there are several systems (even quasi-integrable ones) which are very smooth in Cartesian coordinates but less smooth in action-angle variables (e.g., neighborhoods of elliptic fixed points [FGB98, GFB98], hydrogen atoms in crossed electric and magnetic fields [RC95, RC97] several problems in celestial mechanics [CC07])

We will assume that the Euclidean manifold \mathcal{M} is endowed with an exact symplectic structure $\Omega = d\alpha$ (for some one-form α) and we have

$$\Omega_z(u, v) = \langle u, J(z)v \rangle,$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product on the tangent space of \mathcal{M} and $J(z)$ is a skew-symmetric matrix.

An important particular case is when J induces an almost-complex structure, i.e.

$$(2.1) \quad J^2 = -\text{Id}.$$

Most of our calculations do not need this assumption. One important case, where the identity (2.1) is not satisfied, is when J is a symplectic structure on surfaces of section chosen arbitrarily in the energy surface.

As previously mentioned, we will be considering systems described either by diffeomorphisms or by vector-fields. In the first case, we will consider maps $F: \mathcal{U} \subset \mathcal{M} \mapsto \mathcal{M}$ which are not only symplectic (i.e. $F^*\Omega = \Omega$) but exact symplectic, that is

$$F^*\alpha = \alpha + dP,$$

for some smooth function P , called the *primitive function*.

In the case of vector fields, we will assume that the system is described by a globally Hamiltonian vector-field X , that is

$$X = J\nabla H$$

where H is a globally defined function on \mathcal{M} .

As far as quasi-periodic motions are concerned, we will always assume that the frequencies $\omega \in \mathbb{R}^\ell$ are Diophantine (as it is standard in the KAM theory). We recall here that the notion of Diophantine is different for flows and for diffeomorphisms. Therefore, we define

$$(2.2) \quad \begin{aligned} \mathcal{D}^{\text{aff}}(\nu, \tau) &= \{\omega \in \mathbb{R}^\ell \mid |\omega \cdot k|^{-1} \leq \nu |k|^\tau \forall k \in \mathbb{Z}^\ell - \{0\}\}, \quad \nu \geq \ell - 1 \\ \mathcal{D}(\nu, \tau) &= \{\omega \in \mathbb{R}^\ell \mid |\omega \cdot k - n|^{-1} \leq \nu |k|^\tau \forall k \in \mathbb{Z}^\ell - \{0\}, n \in \mathbb{Z}\}, \quad \nu > \ell \end{aligned}$$

which correspond to the sets of Diophantine frequencies for flows and maps, respectively.

It is well known that for non-Diophantine frequencies substantially complicated behavior can appear [Her92, FKW01]. Observing convincingly these Liouvilian behaviors seems a very ambitious challenge for numerical exploration.

3. EQUATIONS FOR INVARIANCE

In this section, we discuss the functional equations for the objects of interest, that is, invariant tori and the associated whiskers. These functional equations, which describe the invariance of the objects under consideration, are the cornerstone of the algorithms. We will consider at the same time the equations for maps and the equations for vector-fields.

3.1. Functional equations for invariant tori. At least at the formal level, it is natural to search quasi-periodic solutions with frequency ω (independent over the integers) under the form of Fourier series

$$(3.1) \quad \begin{aligned} x(t) &= \sum_{k \in \mathbb{Z}^\ell} \hat{x}_k e^{2\pi i k \cdot \omega t} \\ x^{(n)} &= \sum_{k \in \mathbb{Z}^\ell} \hat{x}_k e^{2\pi i k \cdot \omega n} , \end{aligned}$$

where $\omega \in \mathbb{R}^\ell$, $t \in \mathbb{R}$ and $n \in \mathbb{Z}$.

Note that we allow some components of x to be angles. In that case, it suffices to take some of the components of (3.1) modulo 1.

It is then natural to describe a quasi-periodic function using the so-called ‘‘hull’’ function $K : \mathbb{T}^\ell \rightarrow \mathcal{M}$ defined by

$$K(\theta) = \sum_{k \in \mathbb{Z}^\ell} \hat{x}_k e^{2\pi i k \cdot \theta} ,$$

so that we can write

$$\begin{aligned} x(t) &= K(\omega t), \\ x^{(n)} &= K(n\omega). \end{aligned}$$

The geometric interpretation of the hull function is that it gives an embedding from \mathbb{T}^ℓ into the phase space. In our applications, the embedding will actually be an immersion.

It is clear that quasi-periodic functions will be orbits for a vector field X or a map F if and only if the hull function K satisfies:

$$(3.2) \quad \begin{aligned} \partial_\omega K - X \circ K &= 0, \\ F \circ K - K \circ T_\omega &= 0, \end{aligned}$$

where

- ∂_ω stands for the derivative along direction ω , i.e.

$$(3.3) \quad \partial_\omega = \sum_{k=1}^{\ell} \omega_k \partial_{\theta_k}.$$

- T_ω denotes a rigid rotation

$$(3.4) \quad T_\omega(\theta) = \theta + \omega.$$

A modification of the invariance equations (3.2) which we will be important for our purpose consists in considering

$$(3.5) \quad \begin{aligned} \partial_\omega K - X \circ K - J(K_0)^{-1}(DX \circ K_0)\lambda &= 0, \\ F \circ K - K \circ T_\omega - (J(K_0)^{-1}DK_0) \circ T_\omega \lambda &= 0, \end{aligned}$$

where the unknowns are now $K : \mathbb{T}^\ell \rightarrow \mathcal{M}$ (as before) and $\lambda \in \mathbb{R}^\ell$. Here, K_0 denotes a given approximate (in a suitable sense which will be given below) solution of the equations (3.2).

It has been shown in [FLS07] that, for exact symplectic maps, if (K, λ) satisfy the equation (3.5) with K_0 close to K , then at the end of the iteration of the Newton method, we have $\lambda = 0$ and K is a solution of the invariance equation (3.2). Of course, for approximate solutions of the invariance equation (3.2), there is no reason why λ should vanish. The vanishing of λ depends on global considerations that are discussed in Section 3.1.1.

The advantage of equation (3.5) is that it makes easier to implement a Newton method in the cases that, for the approximate solutions, certain cancelations do not apply. This is particularly important for the case of secondary tori that we will discuss in Section 3.1.2.

The equations (3.2) and (3.5) will be the centerpiece of our treatment. We will discretize them using Fourier series and study numerical methods to solve the discretized equations.

It is important to remark that there are *a posteriori* rigorous results for equations (3.2). That is, there are theorems that ensure that given a function which satisfies (3.2) rather approximately and which, at the same time, satisfies some non-degeneracy conditions, then there is a true solution nearby. These results, stated in [LGJV05, FLS07] and whose proof is the basis for the algorithms we discuss, give us some extra quantities to monitor so that we can be confident that the numerical solutions computed are not spurious effects induced by the truncation.

Remark 1. Notice that for whiskered tori the dimension of the torus ℓ is smaller than half the dimension of the phase space d . In the case of maximal tori, we have $\ell = d$. Hence, the algorithm suggested here has the advantage that that it looks for a function K which is always a function of ℓ variables (and allows to compute invariant objects of dimension ℓ). This is important because the cost of handling functions

grows exponentially fast with the number of variables. Indeed, to discretize a function of ℓ variables into \mathbb{R}^n in a grid of side h , one needs to store $(1/h)^\ell \cdot n$ values.

Remark 2. Recall that, taking time-1 maps, one can reduce the problem of vector fields to the problem of diffeomorphisms. Furthermore, since autonomous Hamiltonian systems preserve energy, we can take a surface of section and deal with the return map. This reduces by 1 the number of variables needed to compute invariant tori.

Remark 3. Equations (3.2) do not have unique solutions. Observe that if K is a solution, for any $\sigma \in \mathbb{R}^\ell$, $K \circ T_\sigma$ is also a solution. In [LGJV05] and [FLS07], it is shown that, in many circumstances, this is the only non uniqueness phenomenon in a sufficiently small neighborhood of K . Hence, it is easy to get rid of it by imposing some normalization. See Section 4.2.

3.1.1. *Some global topological considerations.* In our context, both the domain \mathbb{T}^ℓ and the range of K have topology. As a consequence, there will be some topological considerations in the way that the torus \mathbb{T}^ℓ gets embedded in the phase space. Particularly, the angle variables of \mathbb{T}^ℓ can get wrapped around in different ways in the phase space.

A concise way of characterizing the topology of the embedding is to consider the lift of K to the universal cover, i.e.

$$\widehat{K} : \mathbb{R}^\ell \rightarrow \mathbb{R}^{2d-\ell} \times \mathbb{R}^\ell,$$

in such a way that K is obtained from \widehat{K} by identifying variables in the domain and in the range that differ by an integer.

It is therefore clear that $\forall e \in \mathbb{Z}^\ell$

$$(3.6) \quad \begin{aligned} \widehat{K}_p(\theta + e) &= \widehat{K}_p(\theta), \\ \widehat{K}_q(\theta + e) &= \widehat{K}_q(\theta) + I(e), \end{aligned}$$

where $\widehat{K}_p, \widehat{K}_q$ denote the projections of the lift on each of the components of \mathcal{M} and $I(e)$ is an integer. It is easy to see that $I(e)$ is a linear function of e , namely

$$I(e)_{i=1,\dots,\ell} = \left(\sum_{j=1}^{\ell} I_{ij} e_j \right)_{i=1,\dots,\ell}$$

with $I_{ij} \in \mathbb{Z}$.

We note that if a function \widehat{K}_q satisfies

$$\widehat{K}_q(\theta + e) = \widehat{K}_q(\theta) + I(e),$$

the function

$$(3.7) \quad \widetilde{K}_q(\theta) \equiv \widehat{K}_q(\theta) - I(\theta)$$

is e -periodic. The numerical methods will always be based on studying the periodic functions \widetilde{K}_q , but we will not emphasize this unless it can lead to confusion.

Of course, the integer valued matrix $I = \{I_{ij}\}_{ij}$ remains constant if we modify the embedding slightly. Hence, it remains constant under continuous deformation. For example, in the integrable case with $\ell = d$, invariant tori satisfy $\widehat{K}_q(\theta) = \theta$, so that we have $I = \text{Id}$ and, therefore, all the invariant tori which can be continued from tori of the integrable system will also have $I = \text{Id}$.

3.1.2. Secondary tori. One can produce other ℓ -dimensional tori for which the range of I is of dimension less than ℓ . It is easy to see that if $\text{rank}(I) < \ell$ we can contract $K(\mathbb{T}^\ell)$ to a diffeomorphic copy of $\mathbb{T}^{\text{rank}(I)}$. Even in the case of maximal tori $\ell = d$, one can have contractible directions. The most famous example are the “islands” generated in twist maps around resonances. These tori are known as *secondary tori* and they do not exist in the integrable system. They are generated by the perturbation and therefore they cannot be obtained by continuation, as standard KAM theory.

Perturbative proofs of existence of secondary tori are done in [LW04] and in [DLS06]. The properties of these tori are studied in great detail in [DH08]. They were shown to have an important role in Arnol’d diffusion [DLS03, DLS06, GL06, DH08] to overcome the so-called large gap problem. In [Dua94] one can find rigorous results showing that these islands have to be rather abundant (in different precise meanings). In particular, for standard-like maps they can appear at arbitrarily large values of the parameter.

In [HL00], there are heuristic arguments and numerical simulations arguing that in systems of coupled oscillators, the tori with contractible directions are much more abundant than the invariant tori which can be continued from the integrable limit.

In view of these reasons, we will pay special attention to the computation of these secondary tori in the numerical examples presented in Section 12.

One of the novelties of the method described here is that we can deal in a unified way both primary and secondary KAM tori. We want to emphasize on some features of the method presented here, which are crucial for the computation of secondary tori:

- The method does not require neither the system to be close to integrable nor to be written in action-angle variables.
- The modification of the invariance equations (3.2) allows to adjust some global averages required to solve the Newton equations (see Section 5 and also [FLS07]).

- The periodicity of the function \tilde{K} can be adjusted by the matrix I introduced in (3.6). Hence, the rank of the matrix I has to be chosen according to the number of contractible directions.

3.2. Equations for the invariant whiskers. Invariant tori with $\ell < d$ may have associated invariant bundles and whiskers. We are interested in computing the invariant manifolds which contain the torus and are tangent to the invariant bundles of the linearization around the torus. This includes the stable and unstable manifolds but also invariant manifolds associated to other invariant bundles of the linearization, such as the slow manifolds, associated to the less contracting directions.

Using the parameterization method, it is natural to develop algorithms for invariant manifolds tangent to invariant sub-bundles that satisfy a non-resonance condition. See [CFL03b]. This includes as particular cases, the stable/unstable manifolds, the strong stable and strong unstable ones as well as some other slow manifolds satisfying some non-resonance conditions.

Nevertheless, to avoid lengthening the paper and since these examples happen only in higher dimensional problems that are harder to implement, we restrict in this paper just to the one-dimensional manifolds (see Section 9). We think that, considering this particular case, we can state in a more clear and simpler way the main idea behind the algorithms. We hope to come back to the study of higher dimensional manifolds in future work.

We once again use a parameterization. This amounts to find a solution u of the equations of motion under the form

$$u(t) = W(\omega t, se^{\lambda t})$$

in the continuous time case and

$$u^{(n)} = W(\omega n, \lambda^n s)$$

in the discrete time case, where $W : \mathbb{T}^\ell \times (V \subset \mathbb{R}^{d-\ell}) \rightarrow \mathcal{M}$ and $\lambda \in \mathbb{R}$. The function W has then to satisfy the following invariance equations

$$(3.8) \quad \begin{aligned} F(W(\theta, s)) &= W(\theta + \omega, \lambda s), \\ \partial_\omega W(\theta, s) + \lambda s \frac{\partial}{\partial s} W(\theta, s) &= (X \circ W)(\theta, s), \end{aligned}$$

for the case of maps and flows, respectively. See (3.3) for the definition of the operator ∂_ω .

Note that equations (3.8) imply that in variables (θ, s) the motion on the torus consists of a rigid rotation of frequency ω whereas the motion on the whiskers consists of a contraction (or an expansion) by a constant λ (e^λ in the case of flows). In case of contraction, this amounts to assume that $|\lambda| < 1$ for maps and $\lambda < 0$ for flows. The expanding case is assumed to have $|\lambda| > 1$ for maps and $\lambda > 0$ for flows. Note that if $W(\theta, s)$ satisfies (3.8) then $W(\theta, 0)$ is a solution of

(3.2). We also note that the solutions of equations (3.8) are not unique. Indeed, if $W(\theta, s)$ is a solution, for any $\sigma \in \mathbb{T}^\ell$, $b \in \mathbb{R}$, we have that $\tilde{W}(\theta, s) = W(\theta + \sigma, sb)$ is also a solution. This phenomenon turns out to be the only non-uniqueness of the problem and it can be removed by supplementing the invariance equation with a normalization condition.

Some suitable normalization conditions that make the solutions unique are

$$(3.9) \quad \begin{aligned} \int_{\mathbb{T}^\ell} K_0(\theta) - \theta &= 0, \\ DF(K_0(\theta))DW(\theta, 0) &= \lambda DW(\theta, 0), \\ \|DW(\cdot, 0)\| &= \rho \end{aligned}$$

where $\rho > 0$ is any arbitrarily chosen number and $\|\cdot\|$ stands for a suitable norm.

The fact that the solutions of (3.2) supplemented by (3.9) are locally unique is proved rigorously in [FLS07]. We will see that these normalizations allow to uniquely determine the Taylor expansions (in s) of the function W whenever the first term W_1 is fixed.

The first equation in (3.9) amounts to choosing the origin of coordinates in the parameterization of the torus and, therefore eliminates the ambiguity corresponding to σ . (Check how does (3.9) change when we choose σ).

The other equations fix the scale in the variables s . See that, setting a b amounts to multiplying W_1 by b . Hence, setting the norm of DW sets the b .

From the mathematical point of view, all choices of ρ are equivalent. Nevertheless, from the numerical point of view, it is highly advantageous to choose $\|DW_1\|$ so that the numerical coefficients of the expansion (in s) of W have norms that neither grow nor decrease fast. This makes the computation more immune to round off error since round-off error becomes more important when we add numbers of very different sizes.

3.3. Fourier-Taylor discretization.

3.3.1. *Fourier series discretization.* Since the invariant tori are parameterized by a function K which is periodic in the angle variable θ , it is natural to discretize K using Fourier modes and retaining a finite number of them,

$$(3.10) \quad K(\theta) = \sum_{k \in \mathbb{Z}^\ell, k \in \mathcal{O}_N} c_k e^{2i\pi k \cdot \theta},$$

where

$$\mathcal{O}_N = \{k \in \mathbb{Z}^\ell \mid |k| \leq N\}.$$

Since we will deal with real-valued functions, we have $c_k = \bar{c}_{-k}$ and one can just consider the following cosine and sine Fourier series,

$$(3.11) \quad K(\theta) = a_0 + \sum_{k \in \mathbb{Z}^\ell, k \in \mathcal{O}_N} a_k \cos(2\pi k \cdot \theta) + b_k \sin(2\pi k \cdot \theta).$$

From a practical point of view, in order to store K , we can either keep the values of the function in a grid of $2N$ points or keep the $N + 1$ Fourier modes of the Fourier series.

The main practical shortcoming of Fourier series discretization is that they are not adaptative and that for discontinuous functions, they converge very slowly and not uniformly. These shortcomings are however not very serious for our applications.

Since the tori are invariant under rigid rotations, they tend to be very homogeneous, so that adaptativity is not a great advantage. The fact that the Fourier series converge slowly for functions with discontinuities is a slight problem. It is known that, when KAM tori have enough C^r regularity, they are actually analytic [LGJV05, FLS07]. Nevertheless, when the system is written as a perturbation (of size ε) of an integrable one, for certain values of the parameter ε , the equation (3.2) admits solutions—corresponding to Aubry-Mather sets—which are discontinuous (the theory is much more developed for twist maps). As we increase ε , the problem switches from having analytic solutions to having discontinuous solutions (this is the so-called breakdown of analyticity [Aub83, ALD83, Gre79, McK82, CFL04, OP08]). For values of parameters which are close to the analyticity breakdown, the Fourier discretization tends to behave in a rather surprising way and leads to spurious solutions (solutions of the truncated equations which are not close to truncations of true solutions of the equations. They can be identified using the *a posteriori* KAM theorems, but one has to design algorithms so that they are avoided).

We also note that the evaluation of $F \circ K$ is also very fast if we discretize on a grid (we just need to evaluate the function F for each of the points on the grid). Hence, our iterative step will consist in the application of several operations, all of which being fast either in Fourier mode representation or in a grid representation.

Of course, using the Fast Fourier Transform, we can pass from a grid representation to Fourier coefficients in $O(N \log N)$ operations. There are extremely efficient implementations of the FFT algorithm that take into account not only operation counts but also several other characteristics (memory access, cache, etc.) of modern computers.

3.3.2. Cohomology equations and Fourier discretization. An important advantage of the Fourier discretization is that the cohomology equations, which play a very important role in KAM theory and in our treatment, are straightforward to solve. This section provides a sketch of the resolution of the cohomology equations. Since in this paper we

are only dealing with algorithms and not with estimates, we will not identify what are the regularity requirements in the hypothesis nor the regularity conclusions. Since this is a rather standard part of the KAM argument, there are very detailed estimates in the literature (notably [Rüs75]).

In iterating the Newton algorithm to construct KAM tori, one faces with the so-called *small divisor* problem: let η be a periodic (on \mathbb{T}^ℓ) function. We want to find a function φ , which is also periodic, solving (the first equation is a small divisor equation for flows and the second one for maps)

$$(3.12) \quad \begin{aligned} \partial_\omega \varphi &= \eta, \\ \varphi - \varphi \circ T_\omega &= \eta. \end{aligned}$$

As it is well known, equations (3.12) have a solution provided that $\hat{\eta}_0 \equiv \int_{\mathbb{T}^\ell} \eta = 0$. The Fourier coefficients $\hat{\varphi}_k$ of the solution φ are then given by

$$(3.13) \quad \begin{aligned} \hat{\varphi}_k &= \frac{\hat{\eta}_k}{2\pi i \omega \cdot k}, \\ \hat{\varphi}_k &= \frac{\hat{\eta}_k}{1 - e^{2\pi i k \cdot \omega}}, \end{aligned}$$

where $\hat{\eta}_k$ are the Fourier coefficients of the function η . Notice that the solution φ is unique up to the addition of a constant (the average $\hat{\varphi}_0$ of φ is arbitrary). Equations (3.12) and their solutions (3.13) are very standard in KAM theory (see the exposition in [Lla01b]). Very detailed estimates can be found in [Rüs75], when ω is Diophantine (which is our case).

3.3.3. Algebraic operations and elementary transcendental functions with Fourier series. Algebraic operations (sum, product, division) and elementary transcendental functions (sin, cos, exp, log, power, ...) of Fourier series can be computed either by manipulation of the Fourier coefficients or by using FFT.

For example, the product $h = f \cdot g$ of two Fourier series can be computed either by the Cauchy formula

$$(3.14) \quad h_k = \sum_{i=0}^k f_{k-i} g_i,$$

or by applying the inverse FFT to the coefficients of f and g , computing the product function on each point of the grid in real space and then applying the FFT. The first method clearly takes $O(N^2)$ operations while the second only $O(N \ln N)$.

A modification of the FFT algorithm which leads to some improvement consists in considering Fourier series of length $2N$, compute the inverse FFT on $2N$ points, perform the product and then take the FFT

back. Note that, at this point, except for round-off errors, this algorithm is exact for trigonometric polynomials of degree $2N$. The final step is to truncate again to a polynomial of degree N .

The analysis of algorithms of multiplication from the point of view of theoretical computer science have been undertaken in [Knu97], but to our knowledge, there are few studies of the effects of truncation. An empirical study of roundoff and related numerical stability for the case of functions of one variable was undertaken in [CL08].

In the case of functions of several variables, the issues of numerical stability remain, but we also note that, from the point of view of efficiency, the way that the multiple loops involved in the evaluation of (3.14) are organized becomes crucial. These considerations depend on details of the computer architecture and are poorly understood. Some empirical studies can be found in [Har08].

3.3.4. Fourier-Taylor series. For the computation of whiskers of invariant tori, we will use Fourier-Taylor expansions of the form

$$(3.15) \quad W(\theta, s) = \sum_{n=0}^{\infty} W_n(\theta) s^n,$$

where W_n are 1-periodic functions in θ which we will approximate using Fourier series (3.10).

In order to manipulate this type of series we will use the so called *automatic differentiation algorithms* (see [Knu97]). For the basic algebraic operations and the elementary transcendental functions (exp, sin, cos, log, power, etc.), they provide an expression for the Taylor coefficients of the result in terms of the coefficients of each of the terms.

4. NUMERICAL ALGORITHMS TO SOLVE THE INVARIANCE EQUATION FOR INVARIANT TORI

In this section, we will design a Newton method to solve equations (3.2) and discuss several algorithms to deal with the linearized equations.

We define the following concept of approximate solution.

Definition 1. We say that K is an approximate solution of equations (3.2) if

$$(4.1) \quad \begin{aligned} \partial_{\omega} K - X \circ K &= E, \\ F \circ K - K \circ T_{\omega} &= E \end{aligned}$$

where E is small.

For equations (3.5), the modified equations are

$$(4.2) \quad \begin{aligned} \partial_{\omega} K - X \circ K - (J \circ K_0)^{-1} (DX \circ K_0) \lambda &= E, \\ F \circ K - K \circ T_{\omega} - ((J \circ K_0)^{-1} DK_0) \circ T_{\omega} \lambda &= E \end{aligned}$$

where K_0 is a given embedding satisfying some non-degeneracy conditions.

The Newton method consists in computing Δ in such a way that setting $K \leftarrow K + \Delta$ and expanding the LHS of (4.1) in Δ up to order $\|\Delta\|^2$, it cancels the error term E .

Remark 4. Throughout the paper, we are going to denote $\|\cdot\|$ some norms in functional spaces without specifying however what they are exactly. We refer the reader to the papers [FLS07, CFL03a] where the whole theory is developed and the convergence of the algorithms is proved.

Performing a straightforward calculation, we obtain that the Newton procedure consists in finding Δ satisfying

$$(4.3) \quad \begin{aligned} \partial_\omega \Delta - (DX \circ K)\Delta &= -E, \\ (DF \circ K)\Delta - \Delta \circ T_\omega &= -E. \end{aligned}$$

For the modified invariance equations (3.5), given an approximate solution K , the Newton method consists in looking for (Δ, δ) in such a way that $K + \Delta$ and $\lambda + \delta$ eliminate the first order error. The linearized equations in this case are

$$(4.4) \quad \begin{aligned} \partial_\omega \Delta - (DX \circ K)\Delta - (J \circ K_0)^{-1}(DX \circ K_0)\delta &= -E, \\ DF \circ K\Delta - \Delta \circ T_\omega - ((J \circ K_0)^{-1}DK_0) \circ T_\omega \delta &= -E, \end{aligned}$$

where one can take $K_0 = K$.

The role of the parameter δ is now clear. It allows us to adjust some global averages that we need to be able to solve equations (4.4) (see Section 3.3.2).

As it is well known, the Newton method converges quadratically in $\|E\|$ and the error \tilde{E} at step $K + \Delta$ is such that

$$\|\tilde{E}\| \leq C\|E\|^2$$

where E is the error at the previous step.

The main problem of the Newton method is that it needs a good initial guess to start the iteration. We will discuss several possibilities in Section 11. Of course, any reasonable algorithm can be used as an input to the Newton method. Indeed, our problems have enough structure so that one can use Lindstedt series, variational methods, approximation by periodic orbits, frequency methods, besides the customary continuation methods.

4.1. The large matrix method. The most straightforward method to implement the Newton method is

Algorithm 1 (Large Matrix Algorithm). *Discretize equations (3.2) using truncated Fourier series up to order N and apply the Newton method to the discretization.*

A slight variation is

Algorithm 2 (Variant of the Large Matrix Algorithm). *Discretize equations (3.2) on a grid of $2N$ points and compute E . Discretize (4.3) using truncated Fourier series up to order N , solve the equation using a linear solver and apply the solution.*

The difference between algorithms 1 and 2 is that the first one requires that the approximate derivative we are inverting is the derivative of the discretization.

We note that this extra symmetry is implementable using symbolic manipulation methods. Either of these algorithms requires a storage of a full $N \times N$ matrix. The solution of N linear equations requires $O(N^3)$ operations. There are several variations which are worth noting.

- (1) It is sometimes convenient to use

$$K \leftarrow K + h\Delta$$

with $0 < h < 1$. This, of course, converges more slowly for very small h .

- (2) As we mentioned before in Remark 3, the solutions of the equations are not unique. One can cope with this by imposing some normalizations. A general solution is to use the singular value decomposition (SVD) (see [GVL96]). The pseudo-inverse method then gives increment Δ 's which reduce the residual as much as possible, which is all that is needed by the Newton method. We also note that, in contrast to Gaussian elimination which is numerically unstable (the numerical instability can be partially mitigated by pivoting), the SVD computation is numerically stable. In terms of speed, the SVD method is only a factor ≈ 4 slower than Gaussian elimination. For the cases that we will consider in this paper, we think that the SVD is vastly superior to Gaussian elimination.
- (3) Since the most expensive part of the above scheme is the generation of the derivative matrix and its inversion, it is interesting to mention an improved scheme [Hal75] (see also the exposition in [Mos73, p. 151 ff.] and the geometric analysis in [McG90]). This gives the following algorithm

Algorithm 3 (Hald algorithm). (a) *Let K_0 be a given approximate solution with frequency ω . Compute Γ_0 defined by*

$$\Gamma_0 = D\mathcal{F}(K_0)^{-1}$$

where

$$\mathcal{F}(K) = F \circ K - K \circ T_\omega.$$

(b) *Recursively set*

$$(4.5) \quad \begin{aligned} K_{k+1} &= K_i - \Gamma_k \mathcal{F}(K_k) \\ \Gamma_{k+1} &= \Gamma_k + \Gamma_k(1 - D\mathcal{F}(K_{k+1}))\Gamma_k. \end{aligned}$$

In practical implementations Γ_k is not computed explicitly. We just realize that Γ_0 is obtained by applying an LU or an SVD decomposition to the full matrix $D\mathcal{F}(K_0)$ and then applying back substitution or the pseudo-inverse algorithm.

Note that these calculations are only $O(N^2)$. Similarly, note that the application of $D\mathcal{F}(K_{k+1})$ to a vector can also be done using the explicit formulas and it does not require to generate a full matrix.

Applying the recursive relation (4.5), it is not difficult to reduce Γ_k to several applications of Γ_0 and multiplications by $D\mathcal{F}(K_k)$.

For example, applying the iteration twice we obtain

$$(4.6) \quad \begin{aligned} K_1 &= K_0 - \Gamma_0 \mathcal{F}(K_0), \\ K_2 &= K_1 - \Gamma_0 \mathcal{F}(K_1) - \Gamma_0(1 - D\mathcal{F}(K_1))\Gamma_0 \mathcal{F}(K_1). \end{aligned}$$

Hence two steps of the Newton method can be computed with a number of operations similar to one of one step.

Even if it is not difficult to apply this to higher order expressions, we have found it difficult to obtain improvements. Note that adding quantities of similar sizes to obtain cancelations is very dependent to round-off error.

Remark 5. Another method that has quadratic convergence is the Broyden method [PTVF92, Bro65]. We do not know if the method remains quadratically convergent when we consider it in infinite dimensions and we do not know whether it leads to practical algorithms.

4.2. The Newton method and uniqueness. As we have mentioned in Remark 3, the solutions of (3.2) are not unique. Therefore, the implementations of the Newton method have to be implemented with great care to avoid non-invertible matrices (or to use SVD).

As we mentioned in Section 3.1.1, we will be looking for $\tilde{K}(\theta) = \hat{K}(\theta) - I(\theta)$ introduced in (3.7). Note that for $K_\sigma = K \circ T_\sigma$ we have

$$\tilde{K}_\sigma = \tilde{K} \circ T_\sigma - I\sigma.$$

Hence, if $\{\nu_i\}_{i=1}^L$ is a basis for $\text{Range}(I)$ (L being the dimension), one can impose the conditions

$$(4.7) \quad \int_{\mathbb{T}^\ell} \tilde{K} \cdot \nu_i = 0 \quad \text{for } i = 1, \dots, L$$

and we only need to deal with periodic functions which satisfy (4.7).

In the case that the dimension of the range of I is ℓ —the dimension of the torus—this leads to a unique solution (in the non-degenerate cases, according to the results of [LGJV05]) and we can expect that the matrices appearing in the discretization of the Newton method are invertible.

Two important examples of this situation are primary Lagrangian tori and some whiskered tori. In the case of secondary tori, as we will see, it is advantageous to use the extra variable λ to make progress in the Newton method.

5. FAST NEWTON METHODS FOR LAGRANGIAN TORI

In this section we follow the proof in [LGJV05] to design a Newton method for maximal tori ($\ell = d$). We present an algorithm so that the Newton step does not need to store any $N \times N$ matrix and only requires $O(N \log N)$ operations. We first discuss it for maps.

5.1. The Newton method for Lagrangian tori in exact symplectic maps. The key observation is that the Newton equations (4.3) and (4.4) are closely related to the dynamics and that, therefore, we can use geometric identities to find a linear change of variables that reduces the Newton equations to upper diagonal difference equations with constant coefficients. This phenomenon is often called “automatic reducibility”. The idea is stated in the following proposition:

Proposition 4 (Automatic reducibility). *Given an approximation K of the invariance equation as in (4.1), denote*

$$(5.1) \quad \begin{aligned} \alpha(\theta) &= DK(\theta) \\ N(\theta) &= ([\alpha(\theta)]^T \alpha(\theta))^{-1} \\ \beta(\theta) &= \alpha(\theta)N(\theta) \\ \gamma(\theta) &= (J \circ K(\theta))^{-1} \beta(\theta) \end{aligned}$$

and form the following matrix

$$(5.2) \quad M(\theta) = [\alpha(\theta) \mid \gamma(\theta)],$$

where by $[\cdot \mid \cdot]$ we denote the $2d \times 2d$ matrix obtained by juxtaposing the two $2d \times d$ matrices that are in the arguments.

Then, we have

$$(5.3) \quad (DF \circ K(\theta))M(\theta) = M(\theta + \omega) \begin{pmatrix} \text{Id} & A(\theta) \\ 0 & \text{Id} \end{pmatrix} + \widehat{E}(\theta)$$

where

$$(5.4) \quad A(\theta) = \beta(\theta + \omega)^T [(DF \circ K(\theta))\gamma(\theta) - \gamma(\theta + \omega)],$$

and $\|\widehat{E}\| \leq \|DE\|$ in the case of (4.3) or $\|\widehat{E}\| \leq \|DE\| + |\lambda|$ in the case of (4.4).

Remark 6. If the symplectic structure induces an almost-complex one (i.e. $J^2 = -\text{Id}$), we have that

$$\beta(\theta + \omega)^T \gamma(\theta + \omega) = 0,$$

since the torus is isotropic (in this case Lagrangian). Then $A(\theta)$ has a simpler expression given by

$$A(\theta) = \beta(\theta + \omega)^T (DF \circ K)(\theta) \gamma(\theta).$$

Once again, we omit the definition of the norms used in the bounds for \widehat{E} . For these precisions, we refer to the paper [LGJV05], where the convergence of the algorithm is established.

It is interesting to pay attention to the geometric interpretation of the identity (5.3). Note that, taking derivatives with respect to θ in (4.1), we obtain that

$$(DF \circ K)DK - DK \circ T_\omega = DE,$$

which means that the vectors DK are invariant under $DF \circ K$ (up to a certain error). Moreover, $(J \circ K)^{-1}DKN$ are the symplectic conjugate vectors of DK , so that the preservation of the symplectic form clearly implies (5.3). The geometric interpretation of the matrix $A(\theta)$ is a shear flow near the approximately invariant torus. See Figure 1.

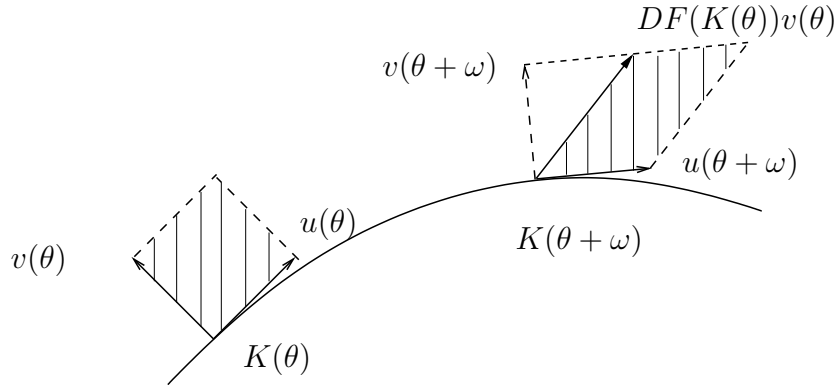


FIGURE 1. Geometric representation of the automatic reducibility where $v = (J \circ K)^{-1}DK$.

In the following, we will see that the result stated in Proposition 4 allows to design a very efficient algorithm for the Newton step.

Notice first that if we change the unknowns $\Delta = MW$ in (4.3) and (4.4) and we use (5.3) we obtain

$$(5.5) \quad M(\theta + \omega) \begin{pmatrix} \text{Id} & A(\theta) \\ 0 & \text{Id} \end{pmatrix} W(\theta) - M(\theta + \omega)W(\theta + \omega) \\ - (J(K_0(\theta + \omega)))^{-1}DK_0(\theta + \omega)\delta = -E(\theta).$$

Of course, the term involving δ has to be omitted when considering (4.3).

Note that, multiplying (5.5) by $M(\theta + \omega)^{-1}$ we are left with the system of equations

$$(5.6) \quad \begin{aligned} W_1(\theta) + A(\theta)W_2(\theta) - B_1(\theta)\delta - W_1(\theta + \omega) &= -\tilde{E}_1(\theta) \\ W_2(\theta) - W_2(\theta + \omega) - B_2(\theta)\delta &= -\tilde{E}_2(\theta) \end{aligned}$$

where

$$\begin{aligned} \tilde{E}(\theta) &= M(\theta + \omega)^{-1}E(\theta) \\ B(\theta) &= M(\theta + \omega)^{-1}(J(K_0(\theta + \omega)))^{-1}DK_0(\theta + \omega) \end{aligned}$$

and the subindexes $i = 1, 2$ indicate symplectic coordinates.

Notice that when K is close to K_0 , we expect that B_2 is close to the d -dimensional identity matrix and B_1 is small.

The next step is to solve equations (5.6) for W (and δ). Notice that equations (5.6) are equations of the form considered in (3.12) and they can be solved very efficiently in Fourier space.

More precisely, the second equation of (5.6) is uncoupled from the first one and allows us to determine W_2 (up to a constant) and δ . Indeed, one can choose δ so that the term $B_2(\theta)\delta - \tilde{E}_2$ has zero average (which is a necessary condition to solve small divisor equations as described in Section 3.3.2). This allows to solve the equation for W_2 according to (3.13) with one degree of freedom which is the average of W_2 . We then denote

$$W_2(\theta) = \tilde{W}_2(\theta) + \overline{W}_2$$

where $\tilde{W}_2(\theta)$ has average zero and $\overline{W}_2 \in \mathbb{R}$.

Once we have \tilde{W}_2 , we can substitute W_2 in the first equation. We get \overline{W}_2 imposing that the average of

$$B_1(\theta)\delta - A(\theta)\tilde{W}_2(\theta) - A(\theta)\overline{W}_2(\theta) - \tilde{E}_1(\theta)$$

is zero and then we can find W_1 up to a constant according to (3.13).

We have the following algorithm:

Algorithm 5 (Newton step for maps). *Consider given F , ω , K_0 and an approximate solution K (resp. K, λ). Perform the following calculations*

1. (1.1) Compute $F \circ K$.
- (1.2) Compute $K \circ T_\omega$.
2. Set $E = F \circ K - K \circ T_\omega$. (resp. set $E = F \circ K - K \circ T_\omega - (J \circ K_0)^{-1}DK_0\lambda$).
3. Following (5.1)
 - (3.1) Compute $\alpha(\theta) = DK(\theta)$.
 - (3.2) Compute $N(\theta) = ([\alpha(\theta)]^T \alpha(\theta))^{-1}$.

- (3.3) Compute $\beta(\theta) = \alpha(\theta)N(\theta)$.
(3.4) Compute $\gamma(\theta) = (J(K(\theta)))^{-1}\beta(\theta)$.
(3.5) Compute $M(\theta) = [\alpha(\theta) \mid \gamma(\theta)]$.
(3.6) Compute $M(\theta + \omega)$.
(3.7) Compute $M(\theta + \omega)^{-1}$.
(3.8) Compute $\tilde{E}(\theta) = M(\theta + \omega)^{-1}E(\theta)$.
(3.9) Compute

$$A(\theta) = \beta(\theta + \omega)^T [(DF \circ K(\theta))\gamma(\theta) - \gamma(\theta + \omega)]$$

as indicated in (5.4).

4. (4.1) Solve for W_2 satisfying

$$W_2 - W_2 \circ T_\omega = -\tilde{E}_2 - \int_{\mathbb{T}^\ell} \tilde{E}_2$$

(resp.

- (4.1') Solve for δ such that

$$\int_{\mathbb{T}^\ell} \tilde{E}_2 - \left[\int_{\mathbb{T}^\ell} B_2 \right] \delta = 0.$$

- (4.2') Solve for W_2 satisfying

$$W_2 - W_2 \circ T_\omega = -\tilde{E}_2 + B_2\delta.$$

Set W_2 such that the average is 0.)

5. (5.1) Compute $A(\theta)W_2(\theta)$.
(5.2) Solve for \overline{W}_2 satisfying

$$0 = \int_{\mathbb{T}^\ell} \tilde{E}_1(\theta) + \int_{\mathbb{T}^\ell} A(\theta)W_2(\theta) + \left[\int_{\mathbb{T}^\ell} A(\theta) \right] \overline{W}_2.$$

- (5.3) Find W_1 solving

$$W_1 - W_1 \circ T_\omega = -\tilde{E}_1 - A(W_2 + \overline{W}_2).$$

Normalize it so that $\int_{\mathbb{T}^\ell} W_1 = 0$

(resp.

- (5.1') Compute $A(\theta)W_2(\theta)$.
(5.2') Solve for \overline{W}_2 satisfying

$$0 = \int_{\mathbb{T}^\ell} \tilde{E}_1(\theta) - \int_{\mathbb{T}^\ell} B_1(\theta)\delta + \int_{\mathbb{T}^\ell} A(\theta)W_2(\theta) + \left[\int_{\mathbb{T}^\ell} A(\theta) \right] \overline{W}_2.$$

- (5.3') Find W_1 solving

$$W_1 - W_1 \circ T_\omega = -\tilde{E}_1 - A(W_2 + \overline{W}_2) + B_1\delta.$$

Normalize it so that $\int_{\mathbb{T}^\ell} W_1 = 0$.)

6. The improved K is $K(\theta) + M(\theta)W(\theta)$
(resp. the improved λ is $\lambda + \delta$).

Notice that steps (1.2), (3.1), (3.6), (4.1) (resp. (4.2')), (5.3) (resp. (5.3')) in Algorithm 5 are diagonal in Fourier series, whereas the other steps are diagonal in the real space representation. Note also that the algorithm only stores vectors which are of order N .

Remark 7. One can perform step (3.7) – the computation of M^{-1} – by just computing the inverse of $M(\theta)$ for all the points in a grid. This requires less than $O(N)$ operations with the constant being the inversion of finite dimensional matrices.

An alternative procedure is to observe that

$$(5.7) \quad M^T(\theta)J(K(\theta))M(\theta) = \begin{pmatrix} 0 & \text{Id} \\ -\text{Id} & 0 \end{pmatrix} + O(\|DE\|).$$

Hence, denoting $J_0 = \begin{pmatrix} 0 & \text{Id} \\ -\text{Id} & 0 \end{pmatrix}$, we see that

$$(5.8) \quad J_0^{-1}M^T(\theta)J(K(\theta))$$

is an approximate inverse of $M(\theta)$, which may be easier to compute.

Using the approximate inverse in place of the inverse leads to a numerical method that also converges quadratically.

5.2. The Newton method for Lagrangian tori in Hamiltonian flows. As we mentioned in Remark 2 it is possible to reduce the treatment of differential equations to that of maps in numerically efficient ways. Nevertheless, it is interesting to present a direct treatment of the differential equation case of (3.2) or (3.5).

The main idea of the algorithm for flows is contained in the following Proposition:

Proposition 6. *Using the same notations (5.1) as in Proposition 4 and considering the matrix M as in (5.2), we have*

$$(5.9) \quad \partial_\omega M(\theta) - DX \circ K(\theta)M(\theta) = M(\theta) \begin{pmatrix} 0 & S(\theta) \\ 0 & 0 \end{pmatrix} + \widehat{E}(\theta)$$

where

$$(5.10) \quad S(\theta) = \beta^T(\theta)[\partial_\omega \gamma(\theta) - DX(K(\theta))\gamma(\theta)]$$

or, equivalently,

$$S(\theta) = \beta^T(\theta)(\text{Id}_{2d} - \beta(\theta)\alpha(\theta)^T)(DX(K(\theta)) + DX(K(\theta))^T)\beta(\theta)$$

and, as before, $\|\widehat{E}\| \leq \|DE\|$ in the case of (4.3) or $\|\widehat{E}\| \leq \|DE\| + |\lambda|$ in the case of (4.4).

As before we refer the reader to the paper [LGJV05] for the definition of the norms and the proof of the convergence of the algorithm.

Again it is not difficult to see how to obtain the result stated in Proposition 6. Considering the approximate invariance equation (4.1)

in the case of flows and taking derivatives with respect to θ we obtain

$$(5.11) \quad \partial_\omega DK - (DX \circ K)DK = DE.$$

Then, if we consider the change of variables M defined in Proposition 4, it is clear that the first n -columns of

$$\widetilde{M}(\theta) = \partial_\omega M(\theta) - DX \circ K(\theta)M(\theta)$$

are equal to zero (up to an error which is bounded by the error in the invariance equation). Finally by equation (5.11) and the Hamiltonian character of the vector field, (5.9) follows.

As in the case of symplectic maps we use equation (5.9) to transform the linearized Newton equation so that it can be solved in a very efficient way. Hence, if we change the unknowns as $\Delta = MW$ and we use (5.9), equations (4.3) and (4.4) for flows reduce to

$$(5.12) \quad M(\theta) \begin{pmatrix} 0 & S(\theta) \\ 0 & 0 \end{pmatrix} W(\theta) + M(\theta) \partial_\omega W(\theta) - J(K_0(\theta))^{-1} DX(K_0(\theta)) \delta = -E(\theta)$$

and by multiplying by $M(\theta)^{-1}$ on both sides we are left with the system of equations

$$(5.13) \quad \begin{aligned} \partial_\omega W_1(\theta) + S(\theta)W_2(\theta) - B_1(\theta)\delta &= -\widetilde{E}_1(\theta) \\ \partial_\omega W_2(\theta) - B_2(\theta)\delta &= -\widetilde{E}_2(\theta) \end{aligned}$$

where

$$\begin{aligned} \widetilde{E}(\theta) &= M(\theta)^{-1}E(\theta) \\ B(\theta) &= M(\theta)^{-1}J(K_0(\theta))^{-1}DX(K_0(\theta)). \end{aligned}$$

Notice that in the case of equation (4.3) we just omit the δ .

Equations (5.13) reduce to solving an equation of the form (3.12). Hence, we determine first δ by imposing that the RHS of the second equation has average zero. Then, the second equation determines W_2 , up to a constant which is fixed by the first equation by imposing that the average its RHS is zero. Finally, we obtain $W_1(\theta)$ up to constant.

The algorithm for flows is the following:

Algorithm 7 (Newton step for flows). *Consider given $X = J(K_0)\nabla H$, ω , K_0 and an approximate solution K (resp. K, λ). Perform the following calculations*

1. (1.1) Compute $\partial_\omega K$. (See (3.3) for the definition).
- (1.2) Compute $X \circ K$.
2. Set $E = \partial_\omega K - X \circ K$ (resp. set $E = \partial_\omega K - X \circ K - (J \circ K_0)^{-1}(DX \circ K_0)\lambda$).
3. Following (5.1)
 - (3.1) Compute $\alpha(\theta) = DK(\theta)$.

(3.2) Compute $\beta(\theta) = J(K_0(\theta))^{-1}\alpha(\theta)$.

(3.3) Compute $\beta(\theta) = \alpha(\theta)N(\theta)$.

(3.4) Compute $\gamma(\theta) = (J(K(\theta)))^{-1}\beta(\theta)$.

(3.5) Compute $M(\theta) = [\alpha(\theta) \mid \gamma(\theta)]$.

(3.6) Compute $M(\theta + \omega)$.

(3.7) Compute $M(\theta + \omega)^{-1}$.

(3.8) Compute $\tilde{E}(\theta) = M(\theta + \omega)^{-1}E(\theta)$.

(3.9) Compute

$$S(\theta) = \beta^T(\theta)(\text{Id}_{2d} - \beta(\theta)\alpha(\theta)^T)(DX(K(\theta)) + DX(K(\theta))^T)\beta(\theta)$$

as indicated in (5.10).

4. (4.1) Solve for W_2 satisfying

$$\partial_\omega W_2 = -\tilde{E}_2 - \int_{\mathbb{T}^\ell} \tilde{E}_2$$

(resp.

(4.1') Solve for δ satisfying

$$\int_{\mathbb{T}^\ell} \tilde{E}_2 - \left[\int_{\mathbb{T}^\ell} B_2 \right] \delta = 0.$$

(4.2') Solve for W_2 satisfying

$$\partial_\omega W_2 = -\tilde{E}_2 + B_2\delta.$$

Set W_2 such that its average is 0.)

5. (5.1) Compute $S(\theta)W_2(\theta)$.

(5.2) Solve for \overline{W}_2 satisfying

$$\int_{\mathbb{T}^\ell} \tilde{E}_1(\theta) + \int_{\mathbb{T}^\ell} S(\theta)W_2(\theta) + \left[\int_{\mathbb{T}^\ell} S(\theta) \right] \overline{W}_2 = 0.$$

(5.3) Find W_1 solving

$$\partial_\omega W_1 = -\tilde{E}_1 - S(W_2 + \overline{W}_2).$$

Normalize it so that $\int_{\mathbb{T}^\ell} W_1 = 0$

(resp.

(5.1') Compute $S(\theta)W_2(\theta)$.

(5.2') Solve for \overline{W}_2 satisfying

$$\int_{\mathbb{T}^\ell} \tilde{E}_1(\theta) + \int_{\mathbb{T}^\ell} B_1(\theta)\delta - \int_{\mathbb{T}^\ell} S(\theta)W_2(\theta) - \left[\int_{\mathbb{T}^\ell} S(\theta) \right] \overline{W}_2 = 0.$$

(5.3') Find W_1 solving

$$\partial_\omega W_1 = -\tilde{E}_1 - S(W_2 + \overline{W}_2) + B_1\delta.$$

Normalize it so that $\int_{\mathbb{T}^\ell} W_1 = 0$).

6. The improved K is $K(\theta) + M(\theta)W(\theta)$

(resp. the improved λ is $\lambda + \delta$).

Notice again that steps (1.1), (3.1), (4.1) (resp. (4.2')), (5.3) (resp. (5.3')) in Algorithm 7 are diagonal in Fourier series, whereas the other steps are diagonal in the real space representation. As before, the algorithm only stores vectors which are of length of order N .

Remark 8. As in the case of maps (see Remark 7) the matrix M here satisfies (5.7). Hence, it is possible to use the approximate inverse given by (5.8).

6. FAST ITERATION OF COCYCLES OVER ROTATIONS. COMPUTATION OF HYPERBOLIC BUNDLES

It is clear from the previous sections that the linearized Newton equations of the invariance equations are very closely tied to the long term behavior of the equations of variation describing the propagation of infinitesimal disturbances around an invariant object. This connection will become more apparent in our discussion on the computation of whiskered tori (see Section 8). Indeed, the relation between structural stability and exponential rates of growth has been one of the basic ingredients of the theory of Anosov systems [Ano69].

In the present section, we study some algorithms related to the iteration of cocycles over rotations. These algorithms will be ingredients of further implementations for the computations of whiskered tori.

Since quasi-periodic cocycles appear in several other situations, the algorithms presented here may have some independent interest and we have striven to make this section independent of the rest of the paper.

6.1. Some standard definitions on cocycles. Given a matrix-valued function $M : \mathbb{T}^\ell \rightarrow GL(2d, \mathbb{R})$ and a vector $\omega \in \mathbb{R}^\ell$, we define the *cocycle* over the rotation T_ω associated to the matrix M by a function $\mathcal{M} : \mathbb{Z} \times \mathbb{T}^\ell \rightarrow GL(2d, \mathbb{R})$ given by

$$(6.1) \quad \mathcal{M}(n, \theta) = \begin{cases} M(\theta + (n-1)\omega) \cdots M(\theta) & n \geq 1, \\ \text{Id} & n = 0, \\ M^{-1}(\theta + (n+1)\omega) \cdots M^{-1}(\theta) & n \leq -1. \end{cases}$$

Equivalently, a cocycle is defined by the relation

$$(6.2) \quad \begin{aligned} \mathcal{M}(0, \theta) &= \text{Id}, \\ \mathcal{M}(1, \theta) &= M(\theta), \\ \mathcal{M}(n+m, \theta) &= \mathcal{M}(n, T_\omega^m(\theta)) \mathcal{M}(m, \theta). \end{aligned}$$

We will say that M is the generator of \mathcal{M} . Note that if $M(\mathbb{T}^\ell) \subset G$ where $G \subset GL(2d, \mathbb{R})$ is a group, then $\mathcal{M}(\mathbb{Z}, \mathbb{T}^\ell) \subset G$.

The main example of a cocycle in this paper is

$$M(\theta) = (DF \circ K)(\theta),$$

for K a parameterization of an invariant torus satisfying (3.2). Other examples appear in discrete Schrödinger equations [Pui02]. In the

above mentioned examples, the cocycles lie in the symplectic group and in the unitary group, respectively.

Similarly, given a matrix valued function $M(\theta)$, a *continuous in time cocycle* $\mathcal{M}(t, \theta)$ is defined to be the unique solution of

$$(6.3) \quad \begin{aligned} \frac{d}{dt} \mathcal{M}(t, \theta) &= M(\theta + \omega t) \mathcal{M}(t, \theta), \\ \mathcal{M}(0, \theta) &= \text{Id}. \end{aligned}$$

From the uniqueness part of Cauchy-Lipschitz theorem, we have the following property

$$(6.4) \quad \begin{aligned} \mathcal{M}(\theta, t + s) &= \mathcal{M}(\theta + \omega t, s) \mathcal{M}(\theta, t), \\ \mathcal{M}(\theta, 0) &= \text{Id}. \end{aligned}$$

Note that (6.3) and (6.4) are the exact analogues of (6.1) and (6.2) in a continuous context. Moreover, if $M(\mathbb{T}^\ell) \subset \mathfrak{G}$, where \mathfrak{G} is a subalgebra of the Lie algebra of the Lie group G , then $\mathcal{M}(\mathbb{R}, \mathbb{T}^\ell) \subset G$.

The main example for us of a continuous in time cocycle will be

$$M(\theta) = (DX \circ K)(\theta),$$

where K is a solution of the invariance equation (3.2) and X is a Hamiltonian vector field. In this case, the cocycle $\mathcal{M}(\theta, t)$ is symplectic.

6.2. Hyperbolicity of cocycles. One of the most crucial property of cocycles is hyperbolicity (or spectral dichotomies) as described in [MS89, SS74, SS76a, SS76b, Sac78].

Definition 2. Given $0 < \lambda < \mu$ we say that a cocycle $\mathcal{M}(n, \theta)$ (resp. $\mathcal{M}(t, \theta)$) has a λ, μ -dichotomy if for every $\theta \in \mathbb{T}^\ell$ there exist a constant $c > 0$ and a splitting depending on θ ,

$$T\mathbb{R}^{2d} = \mathcal{E}^s \oplus \mathcal{E}^u$$

which is characterized by:

$$(6.5) \quad \begin{aligned} (x_\theta, v) \in \mathcal{E}^s &\Leftrightarrow |\mathcal{M}(n, \theta)v| \leq c\lambda^n |v|, & \forall n \geq 0 \\ (x_\theta, v) \in \mathcal{E}^u &\Leftrightarrow |\mathcal{M}(n, \theta)v| \leq c\mu^n |v|, & \forall n \leq 0 \end{aligned}$$

or, in the continuous time case

$$(6.6) \quad \begin{aligned} (x_\theta, v) \in \mathcal{E}^s &\Leftrightarrow |\mathcal{M}(t, \theta)v| \leq c\lambda^t |v|, & \forall t \geq 0 \\ (x_\theta, v) \in \mathcal{E}^u &\Leftrightarrow |\mathcal{M}(t, \theta)v| \leq c\mu^t |v|, & \forall t \leq 0. \end{aligned}$$

The notation \mathcal{E}^s and \mathcal{E}^u is meant to suggest that an important case is the splitting between stable and unstable bundles. This is the case when $\lambda < 1 < \mu$ and the cocycle is said to be hyperbolic. Nevertheless, the theory developed in this section assumes only the existence of a spectral gap.

In the application to the computation of tori, $M(\theta) = (DF \circ K)(\theta)$ and $x_\theta = K(\theta)$. The existence of the spectral gap means that at every point of the invariant torus $K(\theta)$ one has a splitting so that the vectors

grow with appropriate rates λ, μ under iteration of the cocycle. In the case of the invariant torus, it can be seen that the cocycle is just the fundamental matrix of the variational equations so that the cocycle describes the growth of infinitesimal perturbations.

It is well known that the mappings $\theta \rightarrow \mathcal{E}_{x_\theta}^{s,u}$ are C^r if $M(\cdot, \cdot) \in C^r$ for $r \in \mathbb{N} \cup \{\infty, \omega\}$ [HL06c]. This result uses heavily that the cocycles are over a rotation.

A system can have several dichotomies, but for the purposes of this paper, the definition 2 will be enough, since we can perform the analysis presented here for each gap.

One fundamental problem for subsequent applications is the computation of the invariant splittings (and, of course, to ensure their existence). The computation of the invariant bundles is closely related to the computation of iterations of the cocycle.

The first algorithm that comes to mind, is an analogue of the power method to compute leading eigenvalues of a matrix. Given a typical vector $(x_\theta, v) \in \mathcal{E}^u$, we expect that, for $n \gg 1$, $\mathcal{M}(n, \theta)v$ will be a vector in $\mathcal{E}_{x_{T_\omega^n(\theta)}}^u$. Even if there are issues related to the θ dependence, this may be practical if \mathcal{E}^u is a 1-dimensional bundle.

6.3. Equivalence of cocycles, reducibility. Reducibility is a very important issue in the theory of cocycles. We have the following definition.

Definition 3. We say that a cocycle $\widetilde{M}(\theta)$ is equivalent to another cocycle $M(\theta)$ if there exists a matrix valued function $Q : \mathbb{T}^\ell \rightarrow GL(2d, \mathbb{R})$ such that

$$(6.7) \quad \widetilde{M}(\theta) = Q(\theta + \omega)^{-1}M(\theta)Q(\theta).$$

It is easy to check that \widetilde{M} being equivalent to M is an equivalence relation.

If \widetilde{M} is equivalent to a constant cocycle (i.e. independent of θ), we say that \widetilde{M} is “reducible.”

The important point is that, when (6.7) holds, we have

$$(6.8) \quad \widetilde{\mathcal{M}}(n, \theta) = Q(\theta + n\omega)^{-1}\mathcal{M}(n, \theta)Q(\theta).$$

In particular, if M is a constant matrix, we have

$$\widetilde{\mathcal{M}}(n, \theta) = Q^{-1}(\theta + n\omega)M^nQ(\theta),$$

so that the iterations of reducible cocycles are very easy to compute.

We will also see that one can alter the numerical stability properties of the iterations of cocycles by choosing appropriately Q . In that respect, it is also important to mention the concept of “quasi-reducibility” introduced by Eliasson [Eli01].

6.4. Algorithms for fast iteration of cocycles over rotations. In its simplest form, the algorithm for fast iteration of cocycles is:

Algorithm 8 (Iteration of cocycles 1). *Given $M(\theta)$, compute*

$$(6.9) \quad \widehat{M}(\theta) = M(\theta + \omega)M(\theta).$$

Set $\widehat{M} \rightarrow M$, $2\omega \rightarrow \omega$ and iterate the procedure.

We refer to \widehat{M} as the renormalized cocycle and the procedure as a renormalization procedure.

The important property is that applying k times the renormalization procedure described in Algorithm 8 amounts to compute the cocycle $\mathcal{M}(2^k, \theta)$.

Then, if we discretize the matrix $M(\theta)$ taking N points (or N Fourier modes) and denote by $C(N)$ the number of operations required to perform a step of Algorithm 8, we can compute 2^k iterates at a cost of $kC(N)$ operations.

Notice that the important point is that the cost of computing 2^k iterations is proportional to k . Of course, the proportionality constant depends on N . The form of this dependence on N depends on the details on how we compute the shift and the product of matrix valued functions.

There are several alternatives to perform the transformation (6.9). The main difficulty arises from the fact that, if we have points on an equally spaced grid, then $\theta + \omega$ will not be in the same grid. We have at least three alternatives:

- (1) Keep the discretization by points in a grid and compute $M(\theta + \omega)$ by interpolating with nearby points.
- (2) Keep the discretization by points in a grid but note that the shift is diagonal in Fourier space. Of course, the multiplication of the matrix is diagonal in real space.
- (3) Keep the discretization in Fourier space but use the Cauchy formula for the product.

The cost factor of each of these alternatives is, respectively,

$$(6.10) \quad \begin{aligned} C_1(N) &= O(N), \\ C_2(N) &= O(N \log N), \\ C_3(N) &= O(N^2). \end{aligned}$$

Besides their cost, the above algorithms may have different stability and roundoff properties. We are not aware of any study of these stability or round-off properties. The properties of interpolation depend on the dimension.

In each of the cases, the main idea of the method is to precompute some blocks of the iteration, store them and use them in future iteration. One can clearly choose different strategies to group

the blocks. Possibly, different methods can lead to different numerical (in)stabilities. At this moment, we lack a definitive theory of stability which would allow us to choose the blocks.

Next, we will present an alternative consisting of using the QR decomposition for the iterates. As described, for instance in [Ose68, ER85, DVV02], the QR algorithm seems to be rather stable to compute iterates. One advantage is that, in the case of several gaps, it can compute all the eigenvalues in a stable way.

Algorithm 9 (Computation of cocycles with QR). *Given $M(\theta)$ and a QR decomposition of $M(\theta)$,*

$$M(\theta) = Q(\theta)R(\theta),$$

perform the following operations:

- Compute $S(\theta) = R(\theta + \omega)Q(\theta)$.
- Compute pointwise a QR decomposition of S , $S(\theta) = \bar{Q}(\theta)\bar{R}(\theta)$.
- Compute $\tilde{Q}(\theta) = Q(\theta + \omega)\bar{Q}(\theta)$,
 $\tilde{R}(\theta) = \bar{R}(\theta)R(\theta + \omega)$,
 $\tilde{M}(\theta) = \tilde{Q}(\theta)\tilde{R}(\theta)$.
- Set $M \leftarrow \tilde{M}$
 $R \leftarrow \tilde{R}$
 $Q \leftarrow \tilde{Q}$
 $2\omega \leftarrow \omega$

and iterate the procedure.

Since the QR decomposition is a fast algorithm, the total cost of the implementation depends on the issues previously discussed (see costs in (6.10)). Instead of using QR decomposition, one can also perform a SVD decomposition (which is somewhat slower).

In the case of one-dimensional maps, one can be more precise in the description of the method. Indeed, if the frequency ω has a continued fraction expansion

$$\omega = [a_1, a_2, \dots, a_n, \dots],$$

it is well known that the denominators q_n of the convergents of ω (i.e. $p_n/q_n = [a_1, \dots, a_n]$) satisfy

$$\begin{aligned} q_n &= a_n q_{n-1} + q_{n-2}, \\ q_1 &= a_1, \\ q_0 &= 1. \end{aligned}$$

As a consequence, we can consider the following algorithm for this particular case:

Algorithm 10 (Iteration of cocycles 1D). *Given $\omega = [a_1, \dots, a_n, \dots]$ and the cocycle over T_ω generated by $M(\theta)$, define*

$$\omega^0 = 0, \quad \omega^1 = \omega, \quad M^0(\theta) = \text{Id}, \quad M^1(\theta) = M(\theta + (a_1 - 1)\omega) \cdots M(\theta).$$

Then, for $n \geq 2$

$$M^{(n)}(\theta) = M^{(n-1)}(\theta + (a_n - 1)\omega^{n-1}) \cdots M^{(n-1)}(\theta)M^{(n-2)}(\theta)$$

is a cocycle over

$$\omega^{n-1} = a_n\omega^{n-1} + \omega^{n-2}$$

and we have

$$\mathcal{M}(q_n, \theta) = \mathcal{M}^{(n)}(1, \theta).$$

The advantage of this method is that the effective rotation is decreasing to zero so that the cocycle is becoming in some ways similar to the iteration of a constant matrix. This method is somehow reminiscent of some algorithms that have appeared in the mathematical literature [Ryc92, Kri99b, Kri99a].

6.5. The “straddle the saddle” phenomenon and preconditioning. The iteration of cocycles has several pitfalls compared with the iteration of matrices. The main complication from the numerical point of view is that the (un)stable bundle does depend on the base point.

In this section we describe a geometric phenomenon that causes some instability in the iteration of cocycles. This instability –which is genuine– becomes catastrophic when we apply some of the fast iteration methods described in Section 6.4. The phenomenon we will discuss was already observed in [HL06a].

Since we have the inductive relation,

$$\mathcal{M}(n, \theta) = \mathcal{M}(n - 1, \theta + \omega)M(\theta),$$

we see that we can think of computing $\mathcal{M}(n, \theta)$ by applying $\mathcal{M}(n - 1, \theta + \omega)$ to the column vectors of $M(\theta)$.

The j^{th} -column of M , which we will denote by $m_j(\theta)$, can be thought geometrically as an embedding from \mathbb{T}^ℓ to \mathbb{R}^{2d} and is given by $M(\theta)e_j$ where e_j is the j^{th} vector of the canonical basis of \mathbb{R}^{2d} . If the stable space of $\mathcal{M}(n - 1, \theta + \omega)$ has codimension ℓ or less, there could be points $\theta^* \in \mathbb{T}^\ell$ such that $m_j(\theta^*) \in \mathcal{E}_{x_{\theta^*}}^s$ and such that for every $\theta \neq \theta^*$ we have $m_j(\theta) \notin \mathcal{E}_{x_\theta}^s$.

Clearly, the quantity

$$\mathcal{M}(n - 1, \theta^* + \omega)m_j(\theta^*)$$

decreases exponentially. Nevertheless, for all θ in a neighborhood of θ^* such that $\theta \neq \theta^*$

$$\mathcal{M}(n - 1, \theta + \omega)m_j(\theta)$$

will grow exponentially. The direction along which the growth takes place depends on the projection of $m_j(\theta)$ on $\mathcal{E}_{x_{\theta+\omega}}^u$.

For example, in the case $d = 2$, $\ell = 1$ and the stable and unstable directions are one dimensional, the unstable components will have different signs and the vectors $\mathcal{M}(n - 1, \theta + \omega)m_j(\theta)$ will align in opposite directions. An illustration of this phenomenon happens in Figure 2.

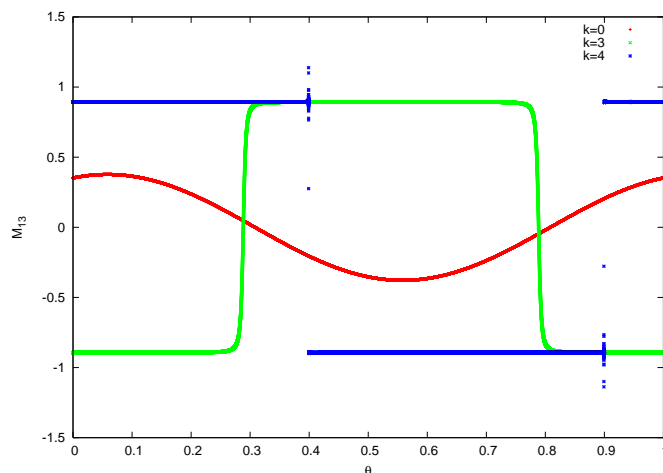


FIGURE 2. The straddle the saddle phenomenon. We plot one of the components of the cocycle $\mathcal{M}(2^k, \theta)$ for the values $k = 0, 3, 4$. The case $k = 0$ was scaled by a factor 200.

The transversal intersection of the range of $m_j(\theta)$ with \mathcal{E}^s is indeed a true phenomenon, and it is a true instability.

Unfortunately, if $m_j(\theta)$ is very discontinuous as a function of θ , the discretization in Fourier series or the interpolation by splines will be extremely inaccurate so that the Algorithm 8 fails to be relevant.

This phenomenon is easy to detect when it happens because the derivatives grow exponentially fast in some localized spots.

One important case where the straddle the saddle is unavoidable is when the invariant bundles are non-orientable. This happens near resonances (see [HL07]). In [HL07], it is shown that, by doubling the angle the case of resonances can be studied comfortably because then, non-orientability is the only obstruction to the triviality of the bundle.

6.5.1. *Eliminating the “straddle the saddle” in the one-dimensional case.* Fortunately, once the phenomenon is detected, it can be eliminated. The main idea is that one can find an equivalent cocycle which does not have the problem (or presents it in a smaller extent).

In more geometric terms we observe that, even if the stable and unstable bundles are geometrically natural objects, the decomposition of a matrix into columns is coordinate dependent. Hence, if we choose some coordinate system which is reasonably close to the stable and unstable manifolds and we denote by Q the change of coordinates, then the cocycle

$$\widetilde{\mathcal{M}}(\theta) = Q(\theta + \omega)^{-1}M(\theta)Q(\theta),$$

is close to a constant. Remark that this is true only in the one-dimensional case. The picture is by far more involved when the bundles have higher rank.

This may seem somewhat circular, but the circularity can be broken using continuation methods. Given a cocycle which is close to constant, fast iteration methods work and they allow us to compute the splitting. Then if we have computed Q for some M , we can use it to precondition the computation of neighboring M .

The algorithms for the computation of bundles will be discussed next.

6.6. Computation of rank-1 stable and unstable bundles using iteration of cocycles. The algorithms described in the previous section provide a fast way to iterate the cocycle. We will see that this iteration method, which is similar to the power method, gives the dominant eigenvalue $\lambda_{max}(\theta)$ and the corresponding eigenvector $m(\theta)$.

The methods based on iteration rely strongly on the fact that the cocycle has one dominating eigenvalue which is simple. This is the case in the numerical examples we considered in Section 12.

Consider that we have performed k iterations of the cocycle (of course we perform scalings at each step) and we have computed $\mathcal{M}(n, \theta)$, with $n = 2^k$. Then, one can easily read the dominant rank-1 bundle from the QR decomposition of the cocycle $\mathcal{M}(n, \theta)$, just taking the column of Q associated to the largest value in the diagonal of the upper triangular matrix R . One obtains a vector $m(\theta + 2^k\omega)$ (and therefore $m(\theta)$ by performing a shift of angle $-2^k\omega$) of modulus 1 spanning the unstable manifold. Since,

$$M(\theta)m(\theta) = \lambda_{max}(\theta)m(\theta + \omega),$$

we have then

$$\lambda_{max}(\theta) = ([M(\theta)m(\theta + \omega)]^T [M(\theta)m(\theta + \omega)])^{1/2}.$$

As it is standard in the power method, we perform scalings at each step dividing all the entries in the matrix $M(\theta)$ by the maximum value among the entries of the matrix.

Hence, for the simplest case that there is one dominant eigenvalue, the method produces a section m (spanning the unstable subbundle) and a real function λ_{max} , which represents the dynamics on the rank 1 unstable subbundle, such that

$$M(\theta)m(\theta) = \lambda_{max}(\theta)m(\theta + \omega).$$

Following [HL06b], under certain non-resonant conditions which are satisfied in the case of the stable and unstable subspaces, one can reduce the 1-dimensional cocycle associated to M and ω to a constant. Hence, we look for a positive function $p(\theta)$ and a constant μ such that

$$(6.11) \quad \lambda_{max}(\theta)p(\theta) = \mu p(\theta + \omega).$$

If $\lambda_{max}(\theta) > 0$, we take logarithms on both sides of the equation (6.11). This leads to

$$\log \lambda_{max}(\theta) + \log p(\theta) = \log \mu + \log p(\theta + \omega),$$

and taking $\log \mu$ to be the average of $\log \lambda_{max}(\theta)$ the problem reduces to solve for $\log p(\theta)$ an equation of the form (3.12). The case $\lambda_{max}(\theta) < 0$ is analogous. Of course, $p(\theta)$ and μ can be obtained just exponentiating.

7. FAST ALGORITHMS TO SOLVE DIFFERENCE EQUATIONS WITH NON CONSTANT COEFFICIENTS

In this section we present fast algorithms to solve for $\Delta(\theta)$ the cohomology equation with non constant coefficients

$$(7.1) \quad A(\theta)\Delta(\theta) - \Delta(\theta + \omega)B(\theta) = \eta(\theta)$$

for given $A(\theta)$, $B(\theta)$ and $\eta(\theta)$ satisfying either $\|A\| < 1$, $\|B^{-1}\| < 1$ or $\|A^{-1}\| < 1$, $\|B\| < 1$.

This type of equation appears in the computation of the projections using a Newton method (see equations (8.30)-(8.31)) as well as in the computation of whiskered tori (this is the resulting equation of the projection of the linearized equation of the Newton method onto the hyperbolic subspaces, see (8.4) and (8.6)).

The algorithms we present here use the contraction properties and they are of iterative nature. Interestingly here, for the 1-dimensional case, we present an amazingly fast algorithm which does not use the contraction properties but Fourier transforms and solves the equation exactly. The main shortcoming is that it involves small divisors, whereas it is not the case for the iterative methods. From the point of view of analysis, the present method leads to estimates which are uniform as the contraction rate goes to 1. See [Her83].

7.0.1. Fast algorithm for the 1-D cohomology equation. In this section we present an efficient algorithm for the one-dimensional version of equation (7.1). It is an adaptation of Herman's "tricks" in [Her83].

Consider the following equation,

$$(7.2) \quad \frac{A(\theta)}{B(\theta)}\Delta(\theta) - \Delta(\theta + \omega) = \frac{\eta(\theta)}{B(\theta)}$$

which is obtained from (7.1) multiplying by $B^{-1}(\theta)$ (recall that in this case $B(\theta)$ is just a number).

We will solve (7.2) in two steps:

1. Find $C(\theta)$ and $\lambda \in \mathbb{R}$ such that

$$(7.3) \quad \frac{A(\theta)}{B(\theta)} = \lambda \frac{C(\theta)}{C(\theta + \omega)}.$$

2. Applying (7.3) in (7.2) and multiplying by $C(\theta + \omega)$ we obtain

$$(7.4) \quad \lambda C(\theta)\Delta(\theta) - C(\theta + \omega)\Delta(\theta + \omega) = \tilde{\eta}(\theta)$$

where $\tilde{\eta}(\theta) = C(\theta + \omega)B^{-1}(\theta)\eta(\theta)$.

If we change the unknowns in (7.4) by $W(\theta) = C(\theta)\Delta(\theta)$, we are left with the equation

$$(7.5) \quad \lambda W(\theta) - W(\theta + \omega) = \tilde{\eta}(\theta).$$

Of course, if $|\lambda| \neq 1$, equation (7.5) can be solved in Fourier space. That is, we can obtain the Fourier coefficients of W as:

$$\widehat{W}_k = \frac{\widehat{\tilde{\eta}}_k}{\lambda - e^{2\pi i k \omega}},$$

and the solution is unique. Notice that whenever $|\lambda| = 1$, equation (7.5) involves small divisors, which is not the case for the iterative methods that will be discussed in the following section.

Finally, once we have $W(\theta)$ we get

$$\Delta(\theta) = C^{-1}(\theta)W(\theta).$$

Step 1 can be achieved by taking logarithms of (7.3). Assume that $A(\theta)/B(\theta) > 0$, otherwise we change the sign. Then, we get

$$\log A(\theta) - \log B(\theta) = \log \lambda + \log C(\theta) - \log C(\theta + \omega).$$

Taking $\log \lambda$ to be the average of $\log A(\theta) - \log B(\theta)$, the problem reduces to solve for $\log C(\theta)$ an equation of the form (3.12). Then $C(\theta)$ and λ can be obtained by exponentiation. Notice that $\log C(\theta)$ is determined up to a constant. We will fix it by assuming that it has average 0.

Hence, we have the following algorithm:

Algorithm 11 (Solution of difference equations with non constant coefficient (1D)). *Given $A(\theta)$, $B(\theta)$ and $\eta(\theta)$. Perform the following instructions:*

1. (1.1) Compute $L(\theta) = \log(A(\theta)) - \log(B(\theta))$.
- (1.2) Compute $\overline{L} = \int_{\mathbb{T}^d} L$.
2. Solve for L_C satisfying

$$L_C(\theta) - L_C \circ T_\omega(\theta) = L(\theta) - \overline{L}$$

as well as having zero average.

3. (3.1) Compute $C(\theta) = \exp(L_C(\theta))$.
- (3.2) Compute $\lambda = \exp(\overline{L})$.
4. Compute $\tilde{\eta}(\theta) = C(\theta + \omega)B^{-1}(\theta)\eta(\theta)$.
5. Solve for W satisfying

$$\lambda W(\theta) - W(\theta + \omega) = \tilde{\eta}(\theta).$$

6. The solution of (7.1) is $\Delta(\theta) = C^{-1}(\theta)W(\theta)$.

7.0.2. *Fast iterative algorithms for the cohomology equation.* In this section we will present a fast algorithm to solve equation (7.1) using iterative methods. The main idea is the same as the one described in Section 6 for the fast iteration of cocycles.

We will consider the case $\|A^{-1}\| < 1$ and $\|B\| < 1$. Then, multiplying (7.1) by $A^{-1}(\theta)$ on the LHS, we obtain

$$(7.6) \quad \Delta(\theta) = A^{-1}(\theta)\Delta(\theta + \omega)B(\theta) + A^{-1}(\theta)\eta(\theta).$$

Next, we compute $\Delta(\theta + \omega)$ by shifting (7.6) and substituting again in (7.6), so that we get

$$\begin{aligned} \Delta(\theta) &= A^{-1}(\theta)\eta(\theta) \\ &\quad + A^{-1}(\theta)A^{-1}(\theta + \omega)\eta(\theta + \omega)B(\theta) \\ &\quad + A^{-1}(\theta)A^{-1}(\theta + \omega)\Delta(\theta + 2\omega)B(\theta + \omega)B(\theta). \end{aligned}$$

Notice that if we define

$$\bar{\eta}(\theta) = A^{-1}(\theta)\eta(\theta)$$

and

$$\begin{aligned} A_1^{-1}(\theta) &= A^{-1}(\theta)A^{-1}(\theta + \omega), \\ B_1(\theta) &= B(\theta + \omega)B(\theta), \\ \eta_1(\theta) &= \bar{\eta}(\theta) + A^{-1}(\theta)\bar{\eta}(\theta + \omega)B(\theta), \end{aligned}$$

we have that

$$\Delta(\theta) = \eta_1(\theta) + A_1^{-1}(\theta)\Delta(\theta + 2\omega)B_1(\theta)$$

which has the same structure as (7.6) and we can repeat the same scheme. This leads to an iterative procedure to compute $A(\theta)$, converging superexponentially in the number of iterations. Thus, define

$$\begin{aligned} A_{n+1}^{-1}(\theta) &= A_n^{-1}(\theta)A_n^{-1}(\theta + 2^n\omega), \\ B_{n+1}(\theta) &= B_n(\theta + 2^n\omega)B_n(\theta), \\ \eta_{n+1}(\theta) &= \eta_n(\theta) + A_n^{-1}(\theta)\eta_n(\theta + 2^n\omega)B_n(\theta), \end{aligned}$$

for $n \geq 0$, with

$$\begin{aligned} A_0^{-1}(\theta) &= A^{-1}(\theta), \\ B_0(\theta) &= B(\theta), \\ \eta_0(\theta) &= \bar{\eta}(\theta). \end{aligned}$$

Then,

$$\Delta(\theta) = \eta_{n+1}(\theta) + A_{n+1}^{-1}(\theta)\Delta(\theta + 2^{n+1}\omega)B_{n+1}(\theta), \quad \forall n \geq 0$$

and

$$\Delta(\theta) = \lim_{n \rightarrow +\infty} \eta_{n+1}(\theta).$$

The convergence of the algorithm is guaranteed by the contraction of A^{-1} and B . The cost of computing 2^n terms in the sum is proportional to n since it involves only n steps of the algorithm.

The iterative algorithm is the following:

Algorithm 12 (Solution of difference equations with non constant coefficient). *Given $A(\theta)$, $B(\theta)$ and $\eta(\theta)$ perform the following operations:*

1. Compute $\Delta(\theta) = A^{-1}(\theta)\eta(\theta)$.
2. Compute
 - (2.1) $\tilde{\Delta}(\theta) = A^{-1}(\theta)\Delta(\theta + \omega)B(\theta) + \Delta(\theta)$.
 - (2.2) $\tilde{A}^{-1}(\theta) = A^{-1}(\theta)A^{-1}(\theta + \omega)$.
 - (2.3) $\tilde{B}(\theta) = B(\theta + \omega)B(\theta)$.
3. Set

$$\begin{aligned} \tilde{\Delta} &\rightarrow \Delta \\ \tilde{A} &\rightarrow A \\ \tilde{B} &\rightarrow B \\ 2\omega &\rightarrow \omega \end{aligned}$$
4. Iterate points 2 – 3.

The case when $\|A\| < 1$ and $\|B^{-1}\| < 1$ can be done similarly. In this case, we multiply (7.1) by $B^{-1}(\theta)$ on the LHS so that we obtain

$$\Delta(\theta + \omega) = A(\theta)\Delta(\theta)B^{-1}(\theta) - \eta(\theta)B^{-1}(\theta).$$

Before applying the iterative scheme we shift by $-\omega$. In this way, we have

$$\Delta(\theta) = A(\theta')\Delta(\theta')B^{-1}(\theta') - \eta(\theta')B^{-1}(\theta')$$

where $\theta' = T_{-\omega}\theta$.

Define

$$\bar{\eta}(\theta') = \eta(\theta')B^{-1}(\theta')$$

and

$$\begin{aligned} A_{n+1}(\theta') &= A_n(\theta')A_n(\theta' - 2^n\omega), \\ B_{n+1}(\theta') &= B_n^{-1}(\theta' - 2^n\omega)B_n^{-1}(\theta'), \\ \eta_{n+1}(\theta') &= \eta_n(\theta') + A_n(\theta')\eta_n(\theta' - 2^n\omega)B_n^{-1}(\theta'), \end{aligned}$$

for $n \geq 0$ with

$$\begin{aligned} A_0(\theta') &= A(\theta'), \\ B_0(\theta') &= B^{-1}(\theta'), \\ \eta_0(\theta') &= \bar{\eta}(\theta'), \end{aligned}$$

then

$$\Delta(\theta) = A_{n+1}(\theta')\Delta(\theta' - 2^{n+1}\omega)B_{n+1}^{-1}(\theta') - \eta_{n+1}(\theta')$$

and

$$\Delta(\theta) = - \lim_{n \rightarrow +\infty} \eta_n(\theta').$$

Again the convergence is superexponential in n .

The iterative algorithm in this case is

Algorithm 13. Given $A(\theta)$, $B(\theta)$ and $\eta(\theta)$, perform the following operations:

1. Compute $\Delta(\theta) = -\eta(\theta)B^{-1}(\theta)$.
2. Compute
 - (2.1) $\tilde{\Delta}(\theta) = A(\theta)\Delta(\theta - \omega)B^{-1}(\theta) + \Delta(\theta)$.
 - (2.2) $\tilde{A}(\theta) = A(\theta)A(\theta - \omega)$.
 - (2.3) $\tilde{B}^{-1}(\theta) = B^{-1}(\theta - \omega)B^{-1}(\theta)$.
3. Set

$$\begin{aligned}\tilde{\Delta} &\rightarrow \Delta \\ \tilde{A} &\rightarrow A \\ \tilde{B} &\rightarrow B \\ 2\omega &\rightarrow \omega\end{aligned}$$
4. Iterate parts 2–3.

This algorithm gives $\Delta(\theta + \omega)$. Shifting it by $-\omega$ we get $\Delta(\theta)$.

8. FAST NEWTON METHODS FOR WHISKERED ISOTROPIC TORI

In this section we follow [FLS07] and develop an efficient Newton method to solve the invariance equations (3.2) and (3.5) for the case of whiskered tori, that is invariant tori with associated stable and unstable manifolds. We focus on the case of maps (the case for vector fields is similar).

As in the case of maximal KAM tori, we will assume that the motion on the torus is a rigid rotation with a Diophantine frequency $\omega \in \mathbb{R}^\ell$. As we have already shown, the invariance implies that the vectors in the range of DK are invariant under DF . The preservation of the symplectic structure, implies that the vectors in the range of $(J \circ K)^{-1}DK$ grow at most polynomially under iteration. We note also that tori with an irrational rotation are co-isotropic, $(DK)^T(J \circ K)^{-1}DK = 0$, i.e. $\text{Range } DK \cap \text{Range } (J \circ K)^{-1}DK = \{0\}$ and therefore $\dim \text{Range } DK \oplus \text{Range } (J \circ K)^{-1}DK = 2\ell$. Therefore, at any point of the invariant torus of dimension ℓ with motion conjugate to a rotation, we can find a 2ℓ -dimensional space of vectors that grow at most polynomially under iteration.

The tori that we will consider are as hyperbolic as possible, given the previous argument. We will consider tori that have a hyperbolic splitting

$$(8.1) \quad T_{K(\theta)}\mathcal{M} = \mathcal{E}_{K(\theta)}^c \oplus \mathcal{E}_{K(\theta)}^s \oplus \mathcal{E}_{K(\theta)}^u$$

such that there exist $0 < \mu_1, \mu_2 < 1$, $\mu_3 > 1$ satisfying $\mu_1\mu_3 < 1$, $\mu_2\mu_3 < 1$ and $C > 0$ such that for all $n \geq 1$ and $\theta \in \mathbb{T}^\ell$

$$\begin{aligned}
(8.2) \quad & v \in \mathcal{E}_{K(\theta)}^s \iff |\mathcal{M}(n, \theta)v| \leq C\mu_1^n |v| \quad \forall n \geq 1 \\
& v \in \mathcal{E}_{K(\theta)}^u \iff |\mathcal{M}(n, \theta)v| \leq C\mu_2^n |v| \quad \forall n \leq -1 \\
& v \in \mathcal{E}_{K(\theta)}^c \iff |\mathcal{M}(n, \theta)v| \leq C\mu_3^n |v| \quad \forall n \in \mathbb{Z}
\end{aligned}$$

where $\mathcal{M}(n, \theta)$ is the cocycle with generator $M(\theta) = DF(K(\theta))$ and frequency ω (see Definition 6.1) and we will assume that $\dim \mathcal{E}_{K(\theta)}^c = 2\ell$, $\dim \mathcal{E}_{K(\theta)}^s = \dim \mathcal{E}_{K(\theta)}^u = d - \ell$.

We associate to the splitting (8.1) the projections $\Pi_{K(\theta)}^c$, $\Pi_{K(\theta)}^s$ and $\Pi_{K(\theta)}^u$ over the invariant spaces $\mathcal{E}_{K(\theta)}^c$, $\mathcal{E}_{K(\theta)}^s$ and $\mathcal{E}_{K(\theta)}^u$.

It is important to note that since F is symplectic (i.e. $F^*\Omega = \Omega$), for all $n \geq 1$ and $n \leq -1$

$$\Omega(u, v) = \Omega(DF^n u, DF^n v)$$

so that, if u, v have rates of growth, by taking limits in the appropriate direction we obtain that Ω is zero. That is, we get

$$\begin{aligned}
\Omega(\mathcal{E}^s, \mathcal{E}^s) &= 0, & \Omega(\mathcal{E}^u, \mathcal{E}^u) &= 0, \\
\Omega(\mathcal{E}^c, \mathcal{E}^s) &= 0, & \Omega(\mathcal{E}^c, \mathcal{E}^u) &= 0.
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
(J(K(\theta)))^{-1} \mathcal{E}_{K(\theta)}^c &= \mathcal{E}_{K(\theta)}^c, \\
(J(K(\theta)))^{-1} \mathcal{E}_{K(\theta)}^s &= \mathcal{E}_{K(\theta)}^u, \\
(J(K(\theta)))^{-1} \mathcal{E}_{K(\theta)}^u &= \mathcal{E}_{K(\theta)}^s.
\end{aligned}$$

Remark 9. As we will see, the only property we will essentially use is that there is a spectral gap. Similar arguments will apply in other frameworks.

In Section 6 we have given a method to compute the rank-1 bundles by iterating the cocycle. Of course, once we have computed the vector spanning the rank-1 (un)stable bundle it is very easy to obtain the projections. In Section 8.2 we discuss an alternative to compute the projections by means of a Newton method. In that case we do not need to assume that the bundle is 1-dimensional.

8.1. General strategy of the Newton method. Recall that we want to design a Newton method to solve the invariance equation (3.2) and (3.5). We are left with solving the linearized equations (4.3) and (4.4). The main difference with respect to maximal tori is that we first will project them on the invariant subspaces \mathcal{E}^c , \mathcal{E}^u and \mathcal{E}^s , and then solve an equation for each subspace.

Thus, let us denote

$$(8.3) \quad \begin{aligned} \Delta^{s,c,u}(\theta) &= \Pi_{K(\theta)}^{s,c,u} \Delta(\theta), \\ E^{s,c,u}(\theta) &= \Pi_{K(\theta+\omega)}^{s,c,u} E(\theta), \end{aligned}$$

such that $\Delta(\theta) = \Delta^s(\theta) + \Delta^c(\theta) + \Delta^u(\theta)$. Then, by the invariant properties of the splitting, the linearized equations for the Newton method (4.3) and (4.4) split in

$$(8.4) \quad \begin{aligned} DF(K(\theta))\Delta^c(\theta) - \Delta^c \circ T_\omega(\theta) &= -E^c(\theta), \\ DF(K(\theta))\Delta^s(\theta) - \Delta^s \circ T_\omega(\theta) &= -E^s(\theta), \\ DF(K(\theta))\Delta^u(\theta) - \Delta^u \circ T_\omega(\theta) &= -E^u(\theta) \end{aligned}$$

and

$$(8.5) \quad \begin{aligned} DF(K(\theta))\Delta^c(\theta) - \Delta^c \circ T_\omega(\theta) \\ + \Pi_{K(\theta+\omega)}^c (J \circ K_0(\theta + \omega))^{-1} DK_0(\theta + \omega)\delta &= -E^c(\theta), \\ DF(K(\theta))\Delta^s(\theta) - \Delta^s \circ T_\omega(\theta) \\ + \Pi_{K(\theta+\omega)}^s (J \circ K_0(\theta + \omega))^{-1} DK_0(\theta + \omega)\delta &= -E^s(\theta), \\ DF(K(\theta))\Delta^u(\theta) - \Delta^u \circ T_\omega(\theta) \\ + \Pi_{K(\theta+\omega)}^u (J \circ K_0(\theta + \omega))^{-1} DK_0(\theta + \omega)\delta &= -E^u(\theta). \end{aligned}$$

We solve for Δ^c and δ the equation on the center subspace using the algorithm described in Section 5. Notice that once δ is obtained the equations (8.5) for the hyperbolic spaces reduce to the equations (8.4). More precisely,

$$(8.6) \quad DF(K(\theta))\Delta^{s,u}(\theta) - \Delta^{s,u} \circ T_\omega(\theta) = -\tilde{E}^{s,u}(\theta)$$

where

$$\tilde{E}^{s,u} = E^{s,u}(\theta) + \Pi_{K(\theta+\omega)}^{s,u} (J \circ K_0(\theta + \omega))^{-1} DK_0(\theta + \omega)\delta.$$

Equations (8.4) and (8.5) for the stable and unstable spaces can be solved iteratively using the contraction properties of the cocycles on the hyperbolic spaces given in (8.2). Indeed, a solution for equations (8.6) is given explicitly by

$$(8.7) \quad \begin{aligned} \Delta^s(\theta) &= \tilde{E}^s \circ T_{-\omega}(\theta) \\ &+ \sum_{k=1}^{\infty} (DF \circ K \circ T_{-\omega}(\theta) \times \cdots \times DF \circ K \circ T_{-k\omega}(\theta)) (\tilde{E}^s \circ T_{-(k+1)\omega}(\theta)) \end{aligned}$$

for the stable equation, and
(8.8)

$$\Delta^u(\theta) = - \sum_{k=0}^{\infty} (DF^{-1} \circ K(\theta) \times \cdots \times DF^{-1} \circ K \circ T_{k\omega}(\theta)) (\tilde{E}^u \circ T_{k\omega}(\theta))$$

for the unstable direction. Of course, the contraction of the cocycles guarantees the uniform convergence of these series.

Instead of using formulae (8.7)-(8.8) to compute the projections on the stable and unstable bundles, we prefer to use the algorithms designed in Section 6.6 or in the following Section 8.2.

Hence, the algorithm for whiskered tori that we summarize here will be a combination of several algorithms:

Algorithm 14. *Consider given F , ω , K_0 and an approximate solution K (resp. K, λ), perform the following operations:*

- (1) *Compute the projections associated to the cocycle $M(\theta) = DF \circ K(\theta)$ and ω using the algorithms described either in Section 6.6 or 8.2.*
- (2) *Project the linearized equation on the center subspace and use the algorithm 5 to obtain Δ^s and δ .*
- (3) *Project the linearized equation to the hyperbolic space and use the algorithms described in Section 7 to obtain $\Delta^{s,u}$.*
- (4) *Set $K + \Delta^s + \Delta^u + \Delta^c \rightarrow K$ and $\lambda + \delta \rightarrow \lambda$ and iterate.*

Next, we will explain in detail each of the previous steps.

8.2. A Newton method to compute the projections. In this section we will discuss a Newton method to compute the projections $\Pi_{K(\theta)}^c$, $\Pi_{K(\theta)}^s$ and $\Pi_{K(\theta)}^u$ associated to the linear spaces $\mathcal{E}_{K(\theta)}^c$, $\mathcal{E}_{K(\theta)}^s$ and $\mathcal{E}_{K(\theta)}^u$ where K is an (approximate) invariant torus. More precisely, we will design a Newton method to compute $\Pi_{K(\theta)}^s$ and $\Pi_{K(\theta)}^{cu} = \Pi_{K(\theta)}^c + \Pi_{K(\theta)}^u$. Similar arguments allow to design a Newton method to compute $\Pi_{K(\theta)}^u$ and $\Pi_{K(\theta)}^{cs} = \Pi_{K(\theta)}^c + \Pi_{K(\theta)}^s$. Then, of course, $\Pi_{K(\theta)}^c$ is given by

$$\Pi_{K(\theta)}^c = \Pi_{K(\theta)}^{cs} \Pi_{K(\theta)}^{cu} = \Pi_{K(\theta)}^{cu} \Pi_{K(\theta)}^{cs} .$$

Let us discuss first a Newton method to compute $\Pi_{K(\theta)}^s$ and $\Pi_{K(\theta)}^{cu}$. To simplify notation, from now on, we will omit the dependence in $K(\theta)$.

We will look for maps $\Pi^s: \mathbb{T}^\ell \rightarrow GL(2d, \mathbb{R})$ and $\Pi^{cu}: \mathbb{T}^\ell \rightarrow GL(2d, \mathbb{R})$ satisfying the following equations:

$$(8.9) \quad \Pi^{cu}(\theta + \omega)M(\theta)\Pi^s(\theta) = 0,$$

$$(8.10) \quad \Pi^s(\theta + \omega)M(\theta)\Pi^{cu}(\theta) = 0,$$

$$(8.11) \quad \Pi^s(\theta) + \Pi^{cu}(\theta) = \text{Id},$$

$$(8.12) \quad [\Pi^s(\theta)]^2 = \Pi^s(\theta),$$

$$(8.13) \quad [\Pi^{cu}(\theta)]^2 = \Pi^{cu}(\theta),$$

$$(8.14) \quad \Pi^s(\theta)\Pi^{cu}(\theta) = 0,$$

$$(8.15) \quad \Pi^{cu}(\theta)\Pi^s(\theta) = 0$$

where $M(\theta) = DF(K(\theta))$.

Notice that the system of equations (8.9)–(8.15) is redundant. It is easy to see that equations (8.13), (8.14) and (8.15) follow from equations (8.11) and (8.12). Therefore, the system of equations that needs to be solved is reduced to equations (8.9)–(8.12).

We are going to design a Newton method to solve equations (8.9)–(8.10) and use equations (8.11)–(8.12) as constraints. In this context, by approximate solution of equations (8.9)–(8.10), we mean a solution (Π^s, Π^{cu}) of the following system

$$(8.16) \quad \Pi^{cu}(\theta + \omega)M(\theta)\Pi^s(\theta) = E^{cu}(\theta),$$

$$(8.17) \quad \Pi^s(\theta + \omega)M(\theta)\Pi^{cu}(\theta) = E^s(\theta),$$

$$(8.18) \quad \Pi^s(\theta) + \Pi^{cu}(\theta) = \text{Id},$$

$$(8.19) \quad [\Pi^s(\theta)]^2 = \Pi^s(\theta).$$

Notice that the error in equation (8.16) has components only on the center and unstable “approximated” subspaces and we denote it by E^{cu} . The same happens with the equation (8.17) but on the “approximated” stable subspace. We assume that E^{cu} and E^s are both small.

As standard in the Newton method, we will look for increments Δ^s and Δ^{cu} in such a way that setting $\Pi^s \leftarrow \Pi^s + \Delta^s$ and $\Pi^{cu} \leftarrow \Pi^{cu} + \Delta^{cu}$, the new projections solve equations (8.9) and (8.10) up to order $\|E\|^2$ where $\|E\| = \|E^s\| + \|E^{cu}\|$ for some norm $|\cdot|$.

The functions Δ^s and Δ^{cu} solve the following equations

$$(8.20) \quad \begin{aligned} \Delta^{cu}(\theta + \omega)M(\theta)\Pi^s(\theta) + \Pi^{cu}(\theta + \omega)M(\theta)\Delta^s(\theta) &= -E^{cu}(\theta) \\ \Delta^s(\theta + \omega)M(\theta)\Pi^{cu}(\theta) + \Pi^s(\theta + \omega)M(\theta)\Delta^{cu}(\theta) &= -E^s(\theta) \end{aligned}$$

with the constraints

$$(8.21) \quad \Delta^s(\theta) + \Delta^{cu}(\theta) = 0$$

$$(8.22) \quad \Pi^s(\theta)\Delta^s(\theta) + \Delta^s(\theta)\Pi^s(\theta) = \Delta^s(\theta) .$$

Notice that by (8.21) we only need to compute Δ^s since $\Delta^{cu} = -\Delta^s$. We now work out equations (8.20), (8.21) and (8.22) so that we can find Δ^s .

Denote

$$(8.23) \quad \begin{aligned} \Delta_s^s &= \Pi^s \Delta^s, \\ \Delta_{cu}^s &= \Pi^{cu} \Delta^s, \end{aligned}$$

so that

$$(8.24) \quad \Delta^s = \Delta_s^s + \Delta_{cu}^s.$$

Then equation (8.22) reads

$$(8.25) \quad \Delta_s^s(\theta) + \Delta^s(\theta)\Pi^s(\theta) = \Delta_s^s(\theta) + \Delta_{cu}^s(\theta),$$

or equivalently,

$$(8.26) \quad \Delta^s(\theta)\Pi^s(\theta) = \Delta_{cu}^s(\theta).$$

Notice that by (8.18), (8.26) and (8.24) we have that

$$(8.27) \quad \Delta^s(\theta)\Pi^{cu}(\theta) = \Delta^s(\theta) - \Delta^s(\theta)\Pi^s(\theta) = \Delta^s(\theta) - \Delta_{cu}^s(\theta) = \Delta_s^s(\theta).$$

Now, using (8.21), equations (8.20) transform to

$$(8.28) \quad \begin{aligned} -\Delta^s(\theta + \omega)M(\theta)\Pi^s(\theta) + \Pi^{cu}(\theta + \omega)M(\theta)\Delta^s(\theta) &= -E^{cu}(\theta), \\ \Delta^s(\theta + \omega)M(\theta)\Pi^{cu}(\theta) - \Pi^s(\theta + \omega)M(\theta)\Delta^s(\theta) &= -E^s(\theta). \end{aligned}$$

Denoting

$$\begin{aligned} N_s(\theta) &= \Pi^s(\theta + \omega)M(\theta)\Pi^s(\theta), \\ N_{cu}(\theta) &= \Pi^{cu}(\theta + \omega)M(\theta)\Pi^{cu}(\theta), \end{aligned}$$

and using that $\Pi^{cu}(\theta + \omega)M(\theta)\Pi^s(\theta)$ and $\Pi^s(\theta + \omega)M(\theta)\Pi^{cu}(\theta)$ are small by (8.16)–(8.17) and $\Pi^s(\theta) + \Pi^{cu}(\theta) = \text{Id}$ by (8.18), it is enough for the Newton method to solve for Δ^s via the following equations

$$(8.29) \quad \begin{aligned} -\Delta^s(\theta + \omega)\Pi^s(\theta + \omega)N_s(\theta) + N_{cu}(\theta)\Pi^{cu}(\theta)\Delta^s(\theta) &= -E^{cu}(\theta), \\ \Delta^s(\theta + \omega)\Pi^{cu}(\theta + \omega)N_{cu}(\theta) - N_s(\theta)\Pi^s(\theta)\Delta^s(\theta) &= -E^s(\theta). \end{aligned}$$

Finally, by expressions (8.26) and (8.27) and taking into account the notations introduced in (8.23), equations (8.29) read out

$$(8.30) \quad -\Delta_{cu}^s(\theta + \omega)N_s(\theta) + N_{cu}(\theta)\Delta_{cu}^s(\theta) = -E^{cu}(\theta),$$

$$(8.31) \quad \Delta_s^s(\theta + \omega)N_{cu}(\theta) - N_s(\theta)\Delta_s^s(\theta) = -E^s(\theta).$$

Equations (8.30)–(8.31) are of the form (7.1) for $A(\theta) = N_{cu}(\theta)$, $B(\theta) = N_s(\theta)$ and $\eta(\theta) = -E^{cu}(\theta)$ in the case of equation (8.30) and $A(\theta) = N_s(\theta)$, $B(\theta) = N_{cu}(\theta)$ and $\eta(\theta) = +E^s(\theta)$ in the case of equation (8.31). Notice that $\|N_s\| < 1$ and $\|N_{cu}^{-1}\| < 1$. Hence, they can be solved iteratively using the fast iterative algorithms described in Section 7.

The explicit expressions for Δ_{cu}^s and Δ_s^s are

$$(8.32) \quad \Delta_{cu}^s(\theta) = - \left[N_{cu}^{-1}(\theta) E^{cu}(\theta) + \sum_{n=1}^{\infty} N_{cu}^{-1}(\theta) \times \cdots \times N_{cu}^{-1}(\theta + n\omega) E^{cu}(\theta + n\omega) N_s(\theta + (n-1)\omega) \times \cdots \times N_s(\theta) \right]$$

and

$$(8.33) \quad \Delta_s^s(\theta) = E^s(\theta - \omega) N_{cu}^{-1}(\theta - \omega) + \sum_{n=1}^{\infty} N_s(\theta - \omega) \times \cdots \times N_s(\theta - (n+1)\omega) E^s(\theta - (n+1)\omega) N_{cu}^{-1}(\theta - (n+1)\omega) \times \cdots \times N_{cu}^{-1}(\theta - \omega).$$

Remark 10. Notice that by the way $N_{cu}(\theta)$ is defined, it is a matrix which does not have full rank. Therefore, we denote $N_{cu}^{-1}(\theta)$ to refer to the ‘‘pseudoinverse’’ matrix.

Finally, let us check that $\Delta^s = \Delta_{cu}^s + \Delta_s^s$ also satisfies the constraints. In order to check that constraint (8.22), which is equivalent to (8.26) is satisfied we will use the expressions (8.32) and (8.33). Notice first that

$$(8.34) \quad N_s(\theta) \Pi^s(\theta) = N_s(\theta)$$

and

$$(8.35) \quad N_{cu}^{-1}(\theta - \omega) \Pi^s(\theta) = 0.$$

Moreover, from (8.16) and using (8.19) one can see that

$$(8.36) \quad E^{cu}(\theta) \Pi^s(\theta) = \Pi^{cu}(\theta + \omega) M(\theta) [\Pi^s(\theta)]^2 = E^{cu}(\theta).$$

Then, from expressions (8.32) and (8.33) and the above expression (8.34), (8.35) and (8.36), it is clear that

$$\Delta^s(\theta) \Pi^s(\theta) = \Delta_s^s(\theta) \Pi^s(\theta) + \Delta_{cu}^s(\theta) \Pi^s(\theta) = 0 + \Delta_{cu}^s,$$

hence, constraint (8.26) is satisfied.

Now, using equation (8.21) we get $\Delta^{cu}(\theta) = -(\Delta_s^s(\theta) + \Delta_{cu}^s(\theta))$ and the improved projections are

$$\begin{aligned} \tilde{\Pi}^s(\theta) &= \Pi^s(\theta) + \Delta_s^s(\theta) + \Delta_{cu}^s(\theta) \\ \tilde{\Pi}^{cu}(\theta) &= \Pi^{cu}(\theta) + \Delta^{cu}(\theta). \end{aligned}$$

The new error for equations (8.9) and (8.10) is now $\|\tilde{E}\| \leq C\|E\|^2$ where $\|E\| = \|E^{cu}\| + \|E^s\|$. Of course equation (8.11) is clearly satisfied but (8.12) is satisfied up to an error which is quadratic in $\|E\|$. However it is easy to get an exact solution for (8.12) and the correction is quadratic in Δ^s (and therefore in Δ^{cu}). To do so, we just take the SVD decomposition of $\tilde{\Pi}^s$ and we set the values in the singular value decomposition to be either 1 or 0.

In this way we obtain new projections Π_{new}^s and $\Pi_{\text{new}}^{cu} = \text{Id} - \Pi_{\text{new}}^s$ satisfying

$$\begin{aligned}\|\Pi_{\text{new}}^s - \tilde{\Pi}^s\| &< \|\Delta^s\|^2 \\ \|\Pi_{\text{new}}^{cu} - \tilde{\Pi}^{cu}\| &< \|\Delta^{cu}\|^2,\end{aligned}$$

so that the error for equations (8.9) and (8.10) is still quadratic in $\|E\|$. Moreover, they satisfy equations (8.12) and, of course, (8.11) exactly.

Hence, setting $\Pi^s \leftarrow \Pi_{\text{new}}^s$ and $\Pi^{cu} \leftarrow \Pi_{\text{new}}^{cu}$ we can repeat the procedure described in this section and perform another Newton step.

Consequently, the algorithm of the Newton method to compute the projections is:

Algorithm 15 (Computation of the projections by a Newton method). *Consider given F, K, ω and an approximate solution (Π^s, Π^{cu}) of equations (8.9)-(8.10). Perform the following calculations:*

1. Compute $M(\theta) = DF \circ K(\theta)$.
2. (2.1) Compute $E^{cu}(\theta) = \Pi^{cu}(\theta + \omega)M(\theta)\Pi^s(\theta)$.
 (2.2) Compute $E^s(\theta) = \Pi^s(\theta + \omega)M(\theta)\Pi^{cu}(\theta)$.
3. (3.1) Compute $N_s(\theta) = \Pi^s(\theta + \omega)M(\theta)\Pi^s(\theta)$.
 (3.2) Compute $N_{cu}(\theta) = \Pi^{cu}(\theta + \omega)M(\theta)\Pi^{cu}(\theta)$.
4. (4.1) Solve for Δ_s^s satisfying

$$N_s(\theta)\Delta_s^s(\theta) - \Delta_s^s(\theta + \omega)N_{cu}(\theta) = E^s(\theta).$$

4. (4.2) Solve for Δ_{cu}^s satisfying

$$N_{cu}(\theta)\Delta_{cu}^s(\theta) - \Delta_{cu}^s(\theta + \omega)N_s(\theta) = -E^{cu}(\theta).$$

5. (5.1) Compute $\tilde{\Pi}^s(\theta) = \Pi^s(\theta) + \Delta_s^s(\theta) + \Delta_{cu}^s(\theta)$.
 (5.2) Compute the SVD decomposition of $\tilde{\Pi}^s(\theta)$: $\tilde{\Pi}^s(\theta) = U(\theta)\Sigma(\theta)V^T(\theta)$.
 (5.3) Set the values in $\Sigma(\theta)$ equal to the closer integer (which will be either 0 or 1).
 (5.4) Recompute $\bar{\Pi}^s(\theta) = U(\theta)\Sigma(\theta)V^T(\theta)$.
6. Set $\bar{\Pi}^s \rightarrow \Pi^s$
 $\text{Id} - \bar{\Pi}^s \rightarrow \Pi^{cu}$
 and iterate the procedure.

Notice that the algorithm requires to store a full matrix and that the matrix multiplication is diagonal in real space representation, whereas the phase shift is diagonal in Fourier space. A discussion on how to perform step 4 efficiently was given in Section 7.

9. COMPUTATION OF RANK-1 WHISKERS OF AN INVARIANT TORUS

In this section, we present algorithms to compute the whiskers associated to an invariant torus, that is the invariant manifolds that contain the torus and are tangent to the invariant bundles.

For the sake of simplicity and in order to state in a clear way the main idea behind the methods we will discuss the case when the invariant whiskers are one-dimensional (i.e. $d - \ell = 1$). However, this idea can be extended to compute invariant manifolds of any rank. We plan to come back to this issue in the future.

As already mentioned, we will look for the whiskers by finding a parameterization for them, so we will search for a function $W: \mathbb{T}^\ell \times (U \subset \mathbb{R}^{d-\ell}) \rightarrow \mathcal{M}$ and a scalar λ satisfying equation (3.8).

We will consider three different methods to solve equation (3.8). We will first discuss the order by order method, which has the main disadvantage that one needs to store and invert a full matrix. The other two methods are based on the philosophy of quasi-Newton methods. Using the phenomenon of “automatic reducibility”, we are able to design a very efficient Newton method. The first method allows to compute simultaneously the invariant tori and the whiskers, whereas the second one assumes that the invariant tori and the tangent bundles are already known.

We considered only the case of maps because the same ideas work for the case of vector fields. Similar algorithms were developed and implemented in [HL06b, HL07] for the slightly simpler case of quasi-periodic maps.

9.1. The order by order method. In this section we follow [CFL05] and refer to it for the proof of the convergence of the series.

We will find a solution (W, λ) of the invariance equation (3.8) discretizing it in Fourier-Taylor series. Hence, we will look for W as a power series

$$(9.1) \quad W(\theta, s) = \sum_{n=0}^{\infty} W_n(\theta) s^n,$$

and match similar coefficients in s^n on both sides of equation (3.8).

For $n = 0$, one obtains

$$(9.2) \quad F(W_0(\theta)) = W_0(\theta + \omega),$$

which is equation (3.2) for the invariant torus. Therefore, we have $W_0(\theta) = K(\theta)$, where K is a parametrization of the invariant torus.

For $n = 1$, we obtain

$$(9.3) \quad DF \circ K(\theta) W_1(\theta) = W_1(\theta + \omega) \lambda,$$

which tells us that $W_1(\theta)$ is an eigenfunction with eigenvalue λ of the operator $\mathcal{M}(1, \theta)$ defined in equation (6.1).

Equation (9.3) states that the bundle spanned by W_1 is invariant for the linearization of F , and the dynamics on it is reduced to a contraction/expansion by a constant λ . Therefore, one can compute W_1 and λ using the algorithms given in Section 6.6.

Remark 11. Notice that if $W_1(\theta)$ is a solution of equation (9.3), then $bW_1(\theta)$, for any $b \in \mathbb{R}$, is also a solution. Even though all the choices of $W_1(\theta)$ are mathematically equivalent, the choice affects the numerical properties of the algorithm. As we mentioned in Section 3.2, $W(\theta, bs)$ is also a solution. The choice of the factor W_1 is the same as the choice of the scale in s because, as we will see, all the subsequent orders are determined. It is convenient to choose b so that all the coefficients remain of order 1 to avoid round-off errors.

For $n \geq 2$, we obtain

$$(9.4) \quad DF \circ K(\theta)W_n(\theta) = W_n(\theta + \omega)\lambda^n + R_n(W_0, \dots, W_{n-1}),$$

where R_n is an explicit polynomial in W_0, \dots, W_{n-1} whose coefficients are derivatives of F evaluated at W_0 .

Equation (9.4) can be solved provided that λ is such that λ^n is not in the spectrum of the operator $\mathcal{M}(1, \theta)$. This condition is clearly satisfied in the case of (un)stable bundles which are one-dimensional but it can also be satisfied by other bundles.

Equation (9.4) can be solved using the large matrix method explained in Section 4.1. Hence, we will discretize equation (9.4) using Fourier series and reduce the problem to solving a linear system in Fourier space, where the unknowns are the Fourier coefficients of the matrix W_n .

Notice that once $W_0(\theta)$ and $W_1(\theta)$ are fixed, the solution $W_n(\theta)$ for $n \geq 2$ of equation (9.4) is uniquely determined.

Finally, if the non resonance condition is satisfied, we know from [CFL05] that the series constructed here converges to a true analytic solution of the problem.

Notice that the inductive equations for W_n do not involve any small divisors.

9.2. A Newton method to compute simultaneously the invariant torus and the whiskers. Instead of solving equation (3.8) step by step as we discussed in the previous section, we can use a Newton method. We start with an initial approximation (W, λ) (resp. (W, λ, μ)) satisfying

$$(9.5) \quad \begin{aligned} F(W(\theta, s)) - W(\theta + \omega, \lambda s) &= E(\theta, s) \\ F(W(\theta, s)) - W(\theta + \omega, \lambda s) - J(K_0(\theta + \omega))^{-1}DK_0(\theta + \omega)\mu &= E(\theta, s) \end{aligned}$$

and we look for an improved solution

$$\begin{aligned} W &\leftarrow W + \Delta \\ \lambda &\leftarrow \lambda + \delta \\ \mu &\leftarrow \mu + \delta_\mu \end{aligned}$$

by solving the linearized equation

$$(9.6) \quad [DF \circ W(\theta, s)]\Delta(\theta, s) - \Delta(\theta + \omega, \lambda s) - s\partial_s W(\theta + \omega, \lambda s)\delta \\ - ((J \circ K_0)^{-1}DK_0) \circ T_\omega(\theta)\delta_\mu = -E(\theta, s).$$

Remark 12. As in the previous cases, the role of the parameter μ is to adjust some averages to solve the equations for the case $n = 0$.

We will try to solve equation (9.6) by discretizing it in Fourier-Taylor series. Notice that equation (9.6) is not diagonal when discretized in Fourier-Taylor series because of the term $DF \circ W$. However, there is a way to make it diagonal using the geometric identities which are a direct generalization of those used for the automatic reducibility of Lagrangian tori in Section 5.1.

We first give the idea of the automatic reducibility when W is such that

$$(9.7) \quad (F \circ W)(\theta, s) = W(\theta + \omega, \lambda s).$$

Taking derivatives with respect to θ and s , we see that

$$DF \circ W(\theta, s)D_\theta W(\theta, s) = D_\theta W(\theta + \omega, \lambda s), \\ DF \circ W(\theta, s)\partial_s W(\theta, s) = \lambda\partial_s W(\theta + \omega, \lambda s).$$

From there we read that the quantity $D_\theta W(\theta, s)$ remains invariant under

$DF \circ W(\theta, s)$, whereas the vector $\partial_s W(\theta, s)$ is multiplied by a factor λ .

The vectors $(J \circ W)^{-1}D_\theta W N$ and $(J \circ W)^{-1}\partial_s W \tilde{N}$, where N and \tilde{N} are normalization matrices (see (9.8)) are the symplectic conjugate vectors of $D_\theta W$ and $\partial_s W$, respectively. The preservation of the symplectic structure implies that

$$(DF \circ W(\theta, s))(J(W(\theta, s)))^{-1}D_\theta W(\theta, s)N(\theta, s) = \\ (J(W(\theta + \omega, \lambda s)))^{-1}D_\theta W(\theta + \omega, \lambda s)N(\theta + \omega, \lambda s) \\ + A(\theta, s)D_\theta W(\theta + \omega, \lambda s), \\ (DF \circ W(\theta, s))(J(W(\theta, s)))^{-1}\partial_s W(\theta, s)\tilde{N}(\theta, s) = \\ \frac{1}{\lambda}(J(W(\theta + \omega, \lambda s)))^{-1}\partial_s W(\theta + \omega, \lambda s)\tilde{N}(\theta + \omega, \lambda s) \\ + B(\theta, s)\partial_s W(\theta + \omega, \lambda s).$$

where $A(\theta, s)$ and $B(\theta, s)$ are some matrices, which will be computed as before.

We can, therefore, see that in the basis $D_\theta W, (J \circ W)^{-1}D_\theta W N, \partial_s W, (J \circ W)^{-1}\partial_s W \tilde{N}$, the matrix $DF \circ W$ is upper triangular with constant coefficients on the diagonal.

Following the proofs in [LGJV05] for instance, one can prove the following proposition which generalizes Proposition 4.

Proposition 16. *Denote*

$$\begin{aligned}
(9.8) \quad & \alpha(\theta, s) = D_\theta W(\theta, s), \\
& \beta(\theta, s) = \partial_s W(\theta, s), \\
& P(\theta, s) = \alpha(\theta, s)N(\theta, s), \\
& Q(\theta, s) = \beta(\theta, s)\tilde{N}(\theta, s), \\
& N(\theta, s) = (\alpha(\theta, s)^T \alpha(\theta, s))^{-1}, \\
& \tilde{N}(\theta, s) = (\beta(\theta, s)^T \beta(\theta, s))^{-1}, \\
& \gamma(\theta, s) = (J \circ W(\theta, s))^{-1}P(\theta, s), \\
& \eta(\theta, s) = (J \circ W(\theta, s))^{-1}Q(\theta, s)
\end{aligned}$$

and form the following matrix

$$(9.9) \quad M(\theta, s) = [\alpha(\theta, s) \mid \gamma(\theta, s) \mid \beta(\theta, s) \mid \eta(\theta, s)]$$

where we denote by $[\cdot \mid \cdot \mid \cdot \mid \cdot]$ the $2d \times 2d$ matrix obtained by juxtaposing the two $2d \times \ell$ matrices and the two $2d \times (d - \ell)$ matrices that are in the arguments.

Then

$$(DF \circ W)(\theta, s)M(\theta, s) = M(\theta + \omega, \lambda s)R(\theta, s) + O(E),$$

where

$$(9.10) \quad R(\theta, s) = \left(\begin{array}{cc|cc} \text{Id} & A(\theta, s) & & \circ \\ 0 & \text{Id} & & \circ \\ \hline & & \lambda & B(\theta, s) \\ \circ & & 0 & 1/\lambda \end{array} \right)$$

with

$$\begin{aligned}
A(\theta, s) &= P(\theta, s)^T [(DF \circ W)(\theta, s)\gamma(\theta, s) - \gamma(\theta + \omega, \lambda s)], \\
B(\theta, s) &= Q(\theta, s)^T [(DF \circ W)(\theta, s)\eta(\theta, s) - \frac{1}{\lambda}\eta(\theta + \omega, \lambda s)],
\end{aligned}$$

and E is the error in (9.5).

Remark 13. If the symplectic structure induces an almost-complex one (i.e. $J^2 = -\text{Id}$), we have that

$$\begin{aligned}
A(\theta, s) &= P(\theta, s)^T (DF \circ W)(\theta, s)\gamma(\theta, s), \\
B(\theta, s) &= Q(\theta, s)^T (DF \circ W)(\theta, s)\eta(\theta, s).
\end{aligned}$$

Now if we change the unknowns $\Delta = MV$ in (9.6) and multiply by $M^{-1}(\theta + \omega, \lambda s)$ the LHS, by Proposition 16, we are left with the following system of equations

$$(9.11) \quad R(\theta, s)V(\theta, s) - V(\theta + \omega, \lambda s) - C(\theta, s)\delta_u = -\tilde{E}(\theta, s) + sH(\theta, s)\delta,$$

where $R(\theta, s)$ is given in (9.10) and

$$C(\theta, s) = M^{-1}(\theta + \omega, \lambda s)(J(K_0(\theta + \omega)))^{-1}DK_0(\theta + \omega),$$

$$\tilde{E}(\theta, s) = M^{-1}(\theta + \omega, \lambda s)E(\theta, s),$$

$$H(\theta, s) = M^{-1}(\theta + \omega, \lambda s)\partial_s W(\theta + \omega, \lambda s).$$

We expand the terms in (9.11) in power series up to some order L and match coefficients of the same order on both sides of the equation. We use subindexes to denote coordinates and superindexes to denote the order. Hence, for order s^0 we have

$$(9.12) \quad V_1^0(\theta) - V_1^0(\theta + \omega) + A^0(\theta)V_2^0(\theta) - C_1^0(\theta)\delta_\mu = -\tilde{E}_1^0(\theta),$$

$$(9.13) \quad V_2^0(\theta) - V_2^0(\theta + \omega) - C_2^0(\theta)\delta_\mu = -\tilde{E}_2^0(\theta),$$

$$(9.14) \quad \lambda V_3^0(\theta) - V_3^0(\theta + \omega) + B^0(\theta)V_4^0(\theta) - C_3^0(\theta)\delta_\mu = -\tilde{E}_3^0(\theta),$$

$$(9.15) \quad \frac{1}{\lambda}V_4^0(\theta) - V_4^0(\theta + \omega) - C_4^0(\theta)\delta_\mu = -\tilde{E}_4^0(\theta).$$

Notice that (9.12) and (9.13) can be solved using Algorithm 5 described in Section 5. Hence, we determine V_1^0 , V_2^0 and δ_μ . Once we know δ_μ , we can solve uniquely for V_3^0 and V_4^0 in equations (9.14) and (9.15). These equations do not have any small divisors nor obstructions since $|\lambda| \neq 1$.

For order s^1 we have

$$(9.16)$$

$$\begin{aligned} V_1^1(\theta) - \lambda V_1^1(\theta + \omega) + A^0(\theta)V_2^1(\theta) + A^1(\theta)V_2^0(\theta) \\ = -\tilde{E}_1^1(\theta) + \delta H_1^0(\theta) + \delta_\mu C_1^1(\theta) \end{aligned}$$

$$(9.17)$$

$$V_2^1(\theta) - \lambda V_2^1(\theta + \omega) = -\tilde{E}_2^1(\theta) + \delta H_2^0(\theta) + \delta_\mu C_2^1(\theta),$$

$$(9.18)$$

$$\begin{aligned} \lambda V_3^1(\theta) - \lambda V_3^1(\theta + \omega) + B^0(\theta)V_4^1(\theta) + B^1(\theta)V_4^0(\theta) \\ = -\tilde{E}_3^1(\theta) + \delta H_3^0(\theta) + \delta_\mu C_3^1(\theta), \end{aligned}$$

$$(9.19)$$

$$\frac{1}{\lambda}V_4^1(\theta) - \lambda V_4^1(\theta + \omega) = -\tilde{E}_4^1(\theta) + \delta H_4^0(\theta) + \delta_\mu C_4^1(\theta).$$

Notice that once we choose δ , equations (9.16) and (9.17) are uniquely solvable for V_1^1 and V_2^1 . Recall that δ_μ is known because it has been computed in the case of order 0 equations.

Similarly, equation (9.19) do not involve small divisors nor obstructions. However, equation (9.18) does have obstructions and small divisors. In order to overcome this problem, we denote by F and G the solutions of

$$\begin{aligned}\frac{1}{\lambda}F(\theta) - \lambda F(\theta + \omega) &= H_4^0(\theta), \\ \frac{1}{\lambda}G(\theta) - \lambda G(\theta + \omega) &= D_4^1(\theta)\end{aligned}$$

where

$$D_4^1(\theta) = -\tilde{E}_4^1(\theta) + \delta_\mu C_4^1(\theta).$$

This gives

$$V_4^1(\theta) = \delta F(\theta) + G(\theta).$$

Taking the average of equation (9.18), we have that

$$\overline{D_3^1} + \delta \overline{H_3^0} - \overline{B^0 F} \delta - \overline{B^0 G} - \overline{B^1 V_4^0} = 0,$$

so we can solve for δ provided that $\overline{H_3^0} - \overline{B^0 F} \neq 0$.

The other orders do not have any problem. For s^n , with $n \geq 2$, we have

$$\begin{aligned}V_1^n(\theta) - \lambda^n V_1^n(\theta + \omega) + \sum_{k=0}^n A^{n-k}(\theta) V_2^k(\theta) &= -\tilde{E}_1^n(\theta) + \delta H_1^{n-1}(\theta) + \delta_\mu C_1^n(\theta), \\ V_2^n(\theta) - \lambda^n V_2^n(\theta + \omega) &= -\tilde{E}_2^n(\theta) + \delta H_2^{n-1}(\theta) + \delta_\mu C_2^n(\theta), \\ \lambda V_3^n(\theta) - \lambda^n V_3^n(\theta + \omega) + \sum_{k=0}^n B^{n-k}(\theta) V_4^k(\theta) &= -\tilde{E}_3^n(\theta) + \delta H_3^{n-1}(\theta) + \delta_\mu C_3^n(\theta), \\ \frac{1}{\lambda} V_4^n(\theta) - \lambda^n V_4^n(\theta + \omega) &= -\tilde{E}_4^n(\theta) + \delta H_4^{n-1}(\theta) + \delta_\mu C_4^n(\theta),\end{aligned}\tag{9.20}$$

and equations (9.20) can be solved uniquely for V_1^n , V_2^n , V_3^n and V_4^n , for $n = 2, \dots, L$, where L is the degree for the Taylor expansion. Hence, we have obtained $\delta, \delta_\mu \in \mathbb{R}$ and

$$V(\theta, s) = \sum_{n=0}^L V^n(\theta) s^n,$$

so that the improved solution is

$$\begin{aligned}W &\leftarrow W + MV, \\ \lambda &\leftarrow \lambda + \delta, \\ \mu &\leftarrow \mu + \delta_\mu.\end{aligned}$$

Remark 14. For $L = 1$, the algorithm allows us to compute simultaneously the invariant torus and the associated linear subspaces.

The algorithm for the simultaneous computation of the whiskers and the invariant torus is

Algorithm 17 (Computation of tori and whiskers simultaneously).
Consider given F , ω , K_0 and a fixed order L . Given an approximate solution (W, λ, μ) , perform the following calculations

1. Compute $E(\theta, s) = F \circ W(\theta, s) - W(\theta + \omega, \lambda s) - ((J \circ K_0)^{-1} DK_0)(\theta + \omega)\mu$.
2. Compute
 - (2.1) $\alpha(\theta, s) = D_\theta W(\theta, s)$.
 - (2.2) $\beta(\theta, s) = \partial_s W(\theta, s)$.
 - (2.3) $N(\theta, s) = [\alpha(\theta, s)^T \alpha(\theta, s)]^{-1}$.
 - (2.4) $\tilde{N}(\theta, s) = [\beta(\theta, s)^T \beta(\theta, s)]^{-1}$.
 - (2.5) $P(\theta, s) = \alpha(\theta, s)N(\theta, s)$.
 - (2.6) $Q(\theta, s) = \beta(\theta, s)\tilde{N}(\theta, s)$.
 - (2.7) $\gamma(\theta, s) = (J \circ W(\theta, s))^{-1}P(\theta, s)$.
 - (2.8) $\eta(\theta, s) = (J \circ W(\theta, s))^{-1}Q(\theta, s)$.
 - (2.9) $M(\theta, s) = [\alpha(\theta, s) \mid \gamma(\theta, s) \mid \beta(\theta, s) \mid \eta(\theta, s)]$.
 - (2.10) $[M(\theta, s)]^{-1}$.
3. Compute
 - (3.1) $C(\theta, s) = M^{-1}(\theta + \omega, \lambda s)(J(K_0(\theta + \omega)))^{-1}DK_0(\theta + \omega)$.
 - (3.2) $\tilde{E}(\theta, s) = M^{-1}(\theta + \omega, \lambda s)E(\theta, s)$.
 - (3.3) $H(\theta, s) = M^{-1}(\theta + \omega, \lambda s)\alpha(\theta + \omega, \lambda s)$.
4. Compute
 - (4.1) $A(\theta, s) = P(\theta, s)^T[(DF \circ W)(\theta, s)\gamma(\theta, s) - \gamma(\theta + \omega, \lambda s)]$.
 - (4.2) $B(\theta, s) = Q(\theta, s)^T[(DF \circ W)(\theta, s)\eta(\theta, s) - \frac{1}{\lambda}\eta(\theta + \omega, \lambda s)]$.
- 5.(5.1) Solve for δ_μ satisfying

$$\int_{\mathbb{T}^\ell} \tilde{E}_2^0 - \left[\int_{\mathbb{T}^\ell} C_2^0 \right] \delta_\mu = 0.$$

- (5.2) Solve for V_2^0 satisfying

$$V_2^0 - V_2^0 \circ T_\omega = -\tilde{E}_2^0 + C_2^0 \delta_\mu.$$

Set V_2^0 such that the average is 0.

- 6.(6.1) Compute $A^0(\theta)V_2^0(\theta)$.

- (6.2) Solve for \bar{V}_2^0 satisfying

$$\int_{\mathbb{T}^\ell} \tilde{E}_1^0 - \int_{\mathbb{T}^\ell} C_1^0(\theta)\delta_\mu + \int_{\mathbb{T}^\ell} A^0 V_2^0 + \left[\int_{\mathbb{T}^\ell} A^0 \right] \bar{V}_2^0 = 0.$$

- (6.3) Set $V_2^0 = V_2^0 + \bar{V}_2^0$.

- (6.4) Solve for V_1^0 satisfying

$$V_1^0 - V_1^0 \circ T_\omega = -\tilde{E}_1^0 - A^0 V_2^0 + C_1^0 \delta_\mu.$$

(6.5) *Normalize so that $\int_{\mathbb{T}^\ell} V_1^0 = 0$.*

7. *Solve for V_4^0 satisfying*

$$\frac{1}{\lambda}V_4^0 - V_4^0 \circ T_\omega = -\tilde{E}_4^0 + C_4^0\delta_\mu.$$

8. *Solve for V_3^0 satisfying*

$$\lambda V_3^0 - V_3^0 \circ T_\omega = -\tilde{E}_3^0 + C_3^0\delta_\mu - B^0V_4^0.$$

9.(9.1) *Solve for F satisfying*

$$\frac{1}{\lambda}F - \lambda F \circ T_\omega = H_4^0.$$

(9.2) *Solve for G satisfying*

$$\frac{1}{\lambda}G - \lambda G \circ T_\omega = -\tilde{E}_4^1 + \delta_\mu C_4^1.$$

(9.3) *Solve for δ satisfying*

$$\left(-\overline{\tilde{E}_3^1} + \delta_\mu \overline{C_3^1} - \overline{B^0G} - \overline{B^1V_4^0}\right) + \delta(\overline{H_3^0} - \overline{B^0F}) = 0.$$

(9.4) *Set $V_4^1 = \delta F + G$*

10. (10.1) *Solve for V_3^1 satisfying*

$$\lambda V_3^1 - \lambda V_3^1 \circ T_\omega = -\tilde{E}_3^1 + \delta H_3^0 + \delta_\mu C_3^1 - B^0V_4^1 - B^1V_4^0.$$

(10.2) *Normalize so that $\int_{\mathbb{T}^\ell} V_3^1 = 0$.*

(10.3) *Solve for V_2^1 satisfying*

$$V_2^1 - \lambda V_2^1 \circ T_\omega = -\tilde{E}_2^1 + \delta H_2^0 + \delta_\mu C_2^1.$$

(10.4) *Solve for V_1^1 satisfying*

$$V_1^1 - \lambda V_1^1 \circ T_\omega = -\tilde{E}_1^1 + \delta H_1^0 + \delta_\mu C_1^1 - A^0V_2^1 - A^1V_2^0.$$

11. *For $n = 2 \dots L$ do*

(11.1) *Solve for V_2^n satisfying*

$$V_2^n - \lambda^n V_2^n \circ T_\omega = -\tilde{E}_2^n(\theta) + \delta H_2^{n-1} + \delta_\mu C_2^n.$$

(11.2) *Compute*

$$\tilde{A}^n = \sum_{k=0}^n A^{n-k} V_2^k.$$

(11.3) *Solve for V_1^n satisfying*

$$V_1^n - \lambda^n V_1^n \circ T_\omega = -\tilde{E}_1^n + \delta H_1^{n-1} + \delta_\mu C_1^n - \tilde{A}^n.$$

(11.4) *Solve for V_4^n satisfying*

$$\frac{1}{\lambda}V_4^n - \lambda^n V_4^n \circ T_\omega = -\tilde{E}_4^n + \delta H_4^{n-1} + \delta_\mu C_4^n.$$

(11.5) *Compute*

$$\tilde{B}^n = \sum_{k=0}^n B^{n-k} V_4^k.$$

(11.6) *Solve for V_3^n satisfying*

$$\lambda V_3^n - \lambda^n V_3^n \circ T_\omega = -\tilde{E}_3^n + \delta H_3^{n-1} + \delta_\mu C_3^n - \tilde{B}^n.$$

12. *Compute*

$$V(\theta, s) = \sum_{n=0}^L V^n(\theta) s^n.$$

13. *Set* $W \leftarrow W + MV$
 $\lambda \leftarrow \lambda + \delta$
 $\mu \leftarrow \mu + \delta_\mu.$

9.3. A Newton method to compute the whiskers. Assuming that we have computed exactly an invariant torus $K(\theta)$ with the associated stable direction $V^s(\theta)$ (resp. unstable direction $V^u(\theta)$) and the rate of contraction (resp. expansion) λ , we can use a Newton method to compute the whiskers.

We consider the invariance equation (3.8), and we assume that we have an initial approximation W for the whiskers, expressed as a power series

$$W(\theta, s) = \sum_{n=0}^{\infty} W^n(\theta) s^n$$

and such that

$$W^0(\theta) = K(\theta) \quad \text{and} \quad W^1(\theta) = V^s(\theta)$$

(the unstable case is analogous).

Then, it is clear that the error E for the initial approximation W is such that

$$E(\theta, s) = \sum_{n \geq 2} E^n(\theta) s^n,$$

since this is exact for the terms of order 0 and 1.

For a given function $G: \mathbb{T}^\ell \times \mathbb{R} \rightarrow \mathcal{M}$, we denote

$$(9.21) \quad G(\theta, s) = G^{[<L]}(\theta, s) + G^{[\geq L]}(\theta, s)$$

where

$$G^{[<L]}(\theta, s) = \sum_{n=0}^{L-1} G^n(\theta) s^n$$

and

$$G^{[\geq L]}(\theta, s) = \sum_{n=L}^{\infty} G^n(\theta) s^n.$$

Using this notation, the linearized equation for the Newton method is

$$[DF \circ W(\theta, s)]\Delta^{[\geq 2]}(\theta, s) - \Delta^{[\geq 2]}(\theta + \omega, \lambda s) = -E^{[\geq 2]}(\theta, s).$$

Similarly as we did in the previous section, we can perform the change of coordinates given by the matrix $M(\theta, s)$ in (9.9) and reduce the problem to solving for $V(\theta, s)$ the following equation, which is diagonal in Fourier-Taylor series,

$$R(\theta, s)V^{[\geq 2]}(\theta, s) - V^{[\geq 2]}(\theta + \omega, \lambda s) = -\tilde{E}^{[\geq 2]}(\theta, s),$$

with $R(\theta, s)$ given in (9.10) and $\tilde{E}(\theta, s) = M(\theta + \omega, \lambda s)^{-1}E(\theta, s)$.

Notice that in this case, we do not have to solve the system of equations for order 0 and 1 and we can go straight to order $n \geq 2$. We use subindexes to denote coordinates and superindexes to denote the order. Hence, for order $n \geq 2$ we need to solve the system of equations

$$(9.22) \quad \begin{aligned} V_1^n(\theta) - \lambda^n V_1^n(\theta + \omega) + \sum_{k=2}^n A^{n-k}(\theta)V_2^k(\theta) &= -\tilde{E}_1^n(\theta), \\ V_2^n(\theta) - \lambda^n V_2^n(\theta + \omega) &= -\tilde{E}_2^n(\theta), \\ \lambda V_3^n(\theta) - \lambda^n V_3^n(\theta + \omega) + \sum_{k=2}^n B^{n-k}(\theta)V_4^k(\theta) &= -\tilde{E}_3^n, \\ \frac{1}{\lambda}V_4^n(\theta) - \lambda^n V_4^n(\theta + \omega) &= -\tilde{E}_4^n. \end{aligned}$$

Notice that the solution of (9.22) for $n = 2, 3$ provides an exact solution of the invariance equation up to order 4. That is, if we set

$$V^{[<4]}(\theta, s) = V^2(\theta, s) + V^3(\theta, s)$$

where V^2 and V^3 are obtained by solving equations (9.22), then the improved solution \bar{W} is given by

$$\bar{W}(\theta, s) = W(\theta, s) + M(\theta, s)V^{[<4]}(\theta, s),$$

where $M(\theta, s)$ was introduced in (9.9). The function W approximates the solution of the invariance equations with an error \bar{E} such that

$$\bar{E}(\theta, s) = \bar{E}^{[\geq 4]}(\theta, s).$$

This process can be iterated and at each step we solve the invariance equation exactly up to an order which is the double of the one we had for the initial approximation. Thus, if we assume that the initial guess W is such that the error in (9.5) satisfies

$$E = E^{[\geq L]},$$

then the modified linearized equation for the Newton method is such that

$$R(\theta, s)V^{[\geq L]}(\theta, s) - V^{[\geq L]}(\theta + \omega, \lambda s) = -\tilde{E}^{[\geq L]}(\theta, s),$$

with $R(\theta, s)$ given in (9.10). If we solve the system of equations (9.22) for $n = L \dots (2L - 1)$ then the improved \bar{W} is

$$\bar{W}(\theta, s) = W(\theta, s) + M(\theta, s)V^{[<2L]}(\theta, s),$$

with $M(\theta, s)$ as in (9.9), and the new error \bar{E} satisfies $\bar{E}(\theta, s) = \bar{E}^{[\geq 2L]}(\theta, s)$.

The algorithm in this case is

Algorithm 18 (Computation of whiskers). *Given F , ω as well as K, V^s, λ and an approximate solution W such that*

$$F(W(\theta, s)) - W(\theta + \omega, \lambda s) = E^{[\geq L]}(\theta, s)$$

with $L \geq 2$ and $W(\theta, 0) = K(\theta)$ and $\partial_s W(\theta, 0) = V^s(\theta)$. Perform the following calculations:

1. Compute $E^{[\geq L]}(\theta, s) = F \circ W(\theta, s) - W(\theta + \omega, \lambda s)$.

2. Compute

$$(2.1) \quad \alpha(\theta, s) = D_\theta W(\theta, s).$$

$$(2.2) \quad \beta(\theta, s) = \partial_s W(\theta, s).$$

$$(2.3) \quad N(\theta, s) = [\alpha(\theta, s)^T \alpha(\theta, s)]^{-1}.$$

$$(2.4) \quad \tilde{N}(\theta, s) = [\beta(\theta, s)^T \beta(\theta, s)]^{-1}.$$

$$(2.5) \quad P(\theta, s) = \alpha(\theta, s)N(\theta, s).$$

$$(2.6) \quad Q(\theta, s) = \beta(\theta, s)\tilde{N}(\theta, s).$$

$$(2.7) \quad \gamma(\theta, s) = (J \circ W(\theta, s))^{-1}P(\theta, s).$$

$$(2.8) \quad \eta(\theta, s) = (J \circ W(\theta, s))^{-1}Q(\theta, s).$$

$$(2.9) \quad M(\theta, s) = [\alpha(\theta, s) \mid \gamma(\theta, s) \mid \beta(\theta, s) \mid \eta(\theta, s)].$$

$$(2.10) \quad [M(\theta, s)]^{-1}.$$

3. Compute

$$\tilde{E}^{[\geq L]}(\theta, s) = M^{-1}(\theta + \omega, \lambda s)E^{[\geq L]}(\theta, s).$$

4. Compute

$$(4.1) \quad A(\theta, s) = P(\theta, s)^T[(DF \circ W)(\theta, s)\gamma(\theta, s) - \gamma(\theta + \omega, \lambda s)].$$

$$(4.2) \quad B(\theta, s) = Q(\theta, s)^T[(DF \circ W)(\theta, s)\eta(\theta, s) - \frac{1}{\lambda}\eta(\theta + \omega, \lambda s)].$$

5. For $n = L \dots 2L - 1$ do

(5.1) Solve for V_2^n satisfying

$$V_2^n - \lambda^n V_2^n \circ T_\omega = -\tilde{E}_2^n(\theta).$$

(5.2) Compute

$$\tilde{A}^n = \sum_{k=L}^n A^{n-k} V_2^k.$$

(5.3) Solve for V_1^n satisfying

$$V_1^n - \lambda^n V_1^n \circ T_\omega = -\tilde{E}_1^n - \tilde{A}^n.$$

(5.4) Solve for V_4^n satisfying

$$\frac{1}{\lambda} V_4^n - \lambda^n V_4^n \circ T_\omega = -\tilde{E}_4^n.$$

(5.5) *Compute*

$$\tilde{B}^n = \sum_{k=L}^n B^{n-k} V_4^k.$$

(5.6) *Solve for V_3^n satisfying*

$$\lambda V_3^n - \lambda^n V_3^n \circ T_\omega = -\tilde{E}_3^n - \tilde{B}^n.$$

6. *Compute*

$$V(\theta, s) = \sum_{n=L}^{2L-1} V^n(\theta) s^n.$$

7. *Set $W \leftarrow W + MV$.*

10. ALGORITHMS TO COMPUTE RANK-1 INVARIANT WHISKERS FOR FLOWS

This section contains a formulation of the algorithms described in the previous section in the context of Hamiltonian vector fields. The reader interested only in the main idea behind the algorithms or happy to use surfaces of section can skip this section. Recall that in the case of vector fields the invariance equation reads

$$(10.1) \quad \left(\partial_\omega + \lambda s \frac{\partial}{\partial s} \right) W(\theta, s) = X(W(\theta, s)),$$

where the unknowns are W and λ . The operator ∂_ω was defined in (3.3).

As in the case of maps, we consider three different methods to solve equation (3.8).

10.1. The order by order method. As done in Section 9.1, we look for W as a power series

$$W(\theta, s) = \sum_{n=0}^{\infty} W_n(\theta) s^n,$$

and match similar coefficients in s^n on both sides of equation (10.1).

For $n = 0$, one obtains

$$(10.2) \quad \partial_\omega W_0(\theta) = (X \circ W_0)(\theta)$$

which admits the solution $W_0(\theta) = K(\theta)$, where K is a parametrization of the invariant torus.

For $n = 1$, we obtain

$$(10.3) \quad \partial_\omega W_1(\theta) + W_1(\theta) \lambda = (DX \circ K(\theta)) W_1(\theta),$$

from where we read that $W_1(\theta)$ is an eigenfunction with eigenvalue $-\lambda$ of the operator \mathcal{L}_ω

$$\mathcal{L}_\omega := \partial_\omega - DX \circ K(\theta).$$

Again, we note that, multiplying a solution of (10.3) by a scalar $b \in \mathbb{R}$, we also obtain a solution. See Remark 11.

For $n \geq 2$, we obtain

$$(10.4) \quad \partial_\omega W_n(\theta) + W_n(\theta)n\lambda = (DX \circ K(\theta))W_n(\theta) + R_n(W_0, \dots, W_{n-1}),$$

where R_n is an explicit polynomial in W_0, \dots, W_{n-1} whose coefficients are derivatives of X evaluated at $W_0 = K$.

Notice that, in this case, equation (10.4) can be solved provided that $n\lambda$ is not in the spectrum of the operator \mathcal{L}_ω (this is a non-resonance condition which is clearly satisfied since the stable spaces are 1-dimensional). As in the case of maps, the previous equation can be solved using the large matrix method.

10.2. A Newton method to compute simultaneously the invariant torus and the whiskers. As in the case of maps, we can solve equation (10.1) by using a Newton method. Hence, we start with an initial approximation (W, λ) (resp. (W, λ, μ) for the invariance equation (10.1)), that is

(10.5)

$$X(W(\theta, s)) - \left(\partial_\omega + \lambda s \frac{\partial}{\partial s} \right) W(\theta, s) = E(\theta, s),$$

$$X(W(\theta, s)) - \left(\partial_\omega + \lambda s \frac{\partial}{\partial s} \right) W(\theta, s) - (J^{-1}DX) \circ K_0 \mu = E(\theta, s)$$

and we will look for an improved solution

$$W \rightarrow W + \Delta$$

$$\lambda \rightarrow \lambda + \delta$$

$$\mu \rightarrow \mu + \delta_\mu$$

by solving the linearized equation

(10.6)

$$\begin{aligned} (DX \circ (W(\theta, s)))\Delta(\theta, s) - \left(\partial_\omega + \lambda s \frac{\partial}{\partial s} \right) \Delta(\theta, s) - s \frac{\partial}{\partial s} W(\theta, s) \delta \\ - (J^{-1}DX) \circ K_0 \delta_\mu = -E(\theta, s). \end{aligned}$$

Once again, we will use a reducibility argument similar to the automatic reducibility of Lagrangian tori. This will lead to a diagonal equation. Applying the operators D_θ and ∂_s to equations (10.5), we obtain

$$DX(W(\theta, s))D_\theta W(\theta, s) - \left(\partial_\omega + \lambda s \frac{\partial}{\partial s} \right) D_\theta W(\theta, s) = O(E),$$

$$\begin{aligned} DX(W(\theta, s))\partial_s W(\theta, s) - \left(\partial_\omega + \lambda s \frac{\partial}{\partial s} \right) \partial_s W(\theta, s) \\ = \lambda \partial_s W(\theta, s) + O(E). \end{aligned}$$

The vectors $(J \circ W)^{-1}D_\theta W N$ and $(J \circ W)^{-1}\partial_s W \tilde{N}$, where N and \tilde{N} are normalization matrices (see (9.8)), are the symplectic conjugate vectors of $\partial_\theta W$ and $\partial_s W$, respectively.

By the Hamiltonian character of the vector field, we have that

$$\begin{aligned} & (DX \circ W(\theta, s))((J \circ W)^{-1}D_\theta W N)(\theta, s) \\ & \quad - \left(\partial_\omega + \lambda s \frac{\partial}{\partial s} \right) ((J \circ W)^{-1}D_\theta W N)(\theta, s) \\ & \quad \quad \quad = S(\theta, s)D_\theta W(\theta, s) + O(E) \\ & (DX \circ W(\theta, s))((J \circ W)^{-1}\partial_s W \tilde{N})(\theta, s) \\ & \quad - \left(\partial_\omega + \lambda s \frac{\partial}{\partial s} \right) ((J \circ W)^{-1}\partial_s W \tilde{N})(\theta, s) \\ & \quad \quad \quad = -\lambda((J \circ W)^{-1}\partial_s W \tilde{N})(\theta, s) + B(\theta, s)\partial_s W(\theta, s) + O(E) \end{aligned}$$

where $S(\theta, s)$ and $B(\theta, s)$ are matrices which can be computed.

We summarize these properties in the following proposition:

Proposition 19. *Using the same notations (9.8) as in Proposition 16 and considering the matrix $M(\theta, s)$ introduced in (9.9), we have*

$$DX \circ W(\theta, s)M(\theta, s) - \left(\partial_\omega + \lambda s \frac{\partial}{\partial s} \right) M(\theta, s) = M(\theta, s)\mathcal{R}(\theta, s) + O(E)$$

where

$$(10.7) \quad \mathcal{R}(\theta, s) = \left(\begin{array}{cc|cc} 0 & S(\theta, s) & & \circ \\ 0 & 0 & & \circ \\ \hline & & \lambda & B(\theta, s) \\ \circ & & 0 & -\lambda \end{array} \right)$$

with

$$\begin{aligned} S(\theta, s) &= P^T(\theta, s)[\partial_\omega \gamma(\theta, s) - DX(W(\theta, s))\gamma(\theta, s)] \\ B(\theta, s) &= Q^T(\theta, s)[\partial_\omega \eta(\theta, s) - DX(W(\theta, s))\eta(\theta, s)] \end{aligned}$$

or, equivalently,

$$\begin{aligned} S(\theta) &= P^T(\theta, s)(\text{Id} - P(\theta, s)\alpha(\theta, s)^T)(DX(W(\theta, s))) \\ & \quad \quad \quad + DX(W(\theta, s))^T P(\theta, s) \\ B(\theta) &= Q^T(\theta, s)(\text{Id} - Q(\theta, s)\beta(\theta, s)^T)(DX(W(\theta, s))) \\ & \quad \quad \quad + DX(W(\theta, s))^T Q(\theta, s) \end{aligned}$$

and E is the error in (9.5).

Now if we change the unknowns $\Delta = MV$ in (10.6) and multiply by $M^{-1}(\theta, s)$ the LHS, by Proposition 19, we are left with the following system of equations

$$(10.8) \quad \mathcal{R}(\theta, s)V(\theta, s) - \left(\partial_\omega + \lambda s \frac{\partial}{\partial s} \right) V(\theta, s) - C(\theta, s)\delta_u = -\tilde{E}(\theta, s) + s\delta H(\theta, s)$$

where $\mathcal{R}(\theta, s)$ is given in (10.7) and

$$\begin{aligned} C(\theta, s) &= M^{-1}(\theta, s)(J^{-1}DX) \circ K_0(\theta), \\ \tilde{E}(\theta, s) &= M^{-1}(\theta, s)E(\theta, s), \\ H(\theta, s) &= M^{-1}(\theta, s)\partial_s W(\theta, s). \end{aligned}$$

We expand the terms in (10.8) as a power series up to some order L and match coefficients of the same order on both sides of the equation. We use subindexes to denote coordinates and superindexes to denote the order. Hence, for order s^0 we have

$$(10.9) \quad -\partial_\omega V_1^0(\theta) + S^0(\theta)V_2^0(\theta) - C_1^0(\theta)\delta_\mu = -\tilde{E}_1^0(\theta),$$

$$(10.10) \quad -\partial_\omega V_2^0(\theta) - C_2^0(\theta)\delta_\mu = -\tilde{E}_2^0(\theta),$$

$$(10.11) \quad \lambda V_3^0(\theta) - \partial_\omega V_3^0(\theta) + B^0(\theta)V_4^0(\theta) - C_3^0(\theta)\delta_\mu = -\tilde{E}_3^0(\theta),$$

$$(10.12) \quad -\lambda V_4^0(\theta) - \partial_\omega V_4^0(\theta) - C_4^0(\theta)\delta_\mu = -\tilde{E}_4^0(\theta).$$

Notice that (10.9) and (10.10) can be solved using the Algorithm 7. Hence, we determine V_1^0 , V_2^0 and δ_μ . Once we know δ_μ , we can solve uniquely for V_3^0 and V_4^0 in equations (10.11) and (10.12). These equations do not have any small divisors nor obstructions.

For order s^1 we have

$$(10.13) \quad \begin{aligned} -\partial_\omega V_1^1(\theta) - \lambda V_1^1(\theta) + S^0(\theta)V_2^1(\theta) + S^1(\theta)V_2^0(\theta) \\ = -\tilde{E}_1^1(\theta) + \delta H_1^0(\theta) + \delta_\mu C_1^1(\theta), \end{aligned}$$

$$(10.14) \quad -\partial_\omega V_2^1(\theta) - \lambda V_2^1(\theta) = -\tilde{E}_2^1(\theta) + \delta H_2^0(\theta) + \delta_\mu C_2^1(\theta),$$

$$(10.15) \quad \begin{aligned} -\partial_\omega V_3^1(\theta) + B^0(\theta)V_4^1(\theta) + B^1(\theta)V_4^0(\theta) \\ = -\tilde{E}_3^1(\theta) + \delta H_3^0(\theta) + \delta_\mu C_3^1(\theta), \end{aligned}$$

$$(10.16) \quad -\partial_\omega V_4^1(\theta) - 2\lambda V_4^1(\theta) = -\tilde{E}_4^1(\theta) + \delta H_4^0(\theta) + \delta_\mu C_4^1(\theta).$$

Notice that once we choose δ , equations (10.13) and (10.14) are uniquely solvable for V_1^1 and V_2^1 . Recall that δ_μ is known, since it has been computed in the case of order 0 equations.

Similarly, equation (10.16) can be solved without small divisors nor obstructions. However, equation (10.15) does have obstructions and small divisors. In order to overcome this problem, we denote by F and G the solutions of

$$\begin{aligned} -\partial_\omega F(\theta) - 2\lambda F(\theta) &= H_4^0(\theta), \\ -\partial_\omega G(\theta) - 2\lambda G(\theta) &= D_4^1(\theta) \end{aligned}$$

where

$$D_4^1(\theta) = -\tilde{E}_4^1(\theta) + \delta_\mu C_4^1(\theta),$$

then

$$V_4^1(\theta) = \delta F(\theta) + G(\theta).$$

Taking averages of the equation for V_3^1 we get

$$\overline{D_3^1} + \delta \overline{H_3^0} - \overline{B^0 F} \delta - \overline{B^0 G} - \overline{B^1 V_4^0} = 0.$$

So we can solve for δ provided that $\overline{H_3^0} - \overline{B^0 F} \neq 0$.

Now the other orders do not have any obstructions,

(10.17)

$$\begin{aligned} -\partial_\omega V_1^n(\theta) - n\lambda V_1^n(\theta) + \sum_{k=0}^n S^{n-k}(\theta) V_2^k(\theta) &= \\ &= -\tilde{E}_1^n(\theta) + \delta H_1^{n-1}(\theta) + \delta_\mu C_1^n(\theta), \\ -\partial_\omega V_2^n(\theta) - n\lambda V_2^n(\theta) &= -\tilde{E}_2^n(\theta) + \delta H_2^{n-1}(\theta) + \delta_\mu C_2^n(\theta), \\ -\partial_\omega V_3^n(\theta) - (n-1)\lambda V_3^n(\theta) + \sum_{k=0}^n B^{n-k}(\theta) V_4^k(\theta) &= \\ &= -\tilde{E}_3^n(\theta) + \delta H_3^{n-1}(\theta) + \delta_\mu C_3^n(\theta), \\ -\partial_\omega V_4^n(\theta) - (n+1)\lambda V_4^n(\theta) &= -\tilde{E}_4^n(\theta) + \delta H_4^{n-1}(\theta) + \delta_\mu C_4^n(\theta). \end{aligned}$$

for $n \geq 2$ and they can be solved uniquely for V_1^n, V_2^n, V_3^n and V_4^n , for $n = 2, \dots, L$, where L is the degree for the Taylor expansion. Hence, we have obtained $\delta, \delta_\mu \in \mathbb{R}$ and

$$V(\theta, s) = \sum_{n=0}^L V^n(\theta) s^n$$

and the improved solution is

$$\begin{aligned} W &\leftarrow W + MV \\ \lambda &\leftarrow \lambda + \delta \\ \mu &\leftarrow \mu + \delta_\mu \end{aligned}$$

The algorithm for the whiskers and the invariant torus, analogous to Algorithm 17, is

Algorithm 20 (Computation of whiskers and tori for flows). *Consider given X , ω , K_0 and a fixed order L . Given an approximate solution (W, λ, μ) , perform the following calculations*

1. Compute $E(\theta, s) = X(W(\theta, s)) - (\partial_\omega + \lambda s \partial_s)W(\theta, s) - (J^{-1}DX) \circ K_0(\theta)\mu$.
2. Compute
 - (2.1) $\alpha(\theta, s) = D_\theta W(\theta, s)$.
 - (2.2) $\beta(\theta, s) = \partial_s W(\theta, s)$.
 - (2.3) $N(\theta, s) = [\alpha(\theta, s)^T \alpha(\theta, s)]^{-1}$.
 - (2.4) $\tilde{N}(\theta, s) = [\beta(\theta, s)^T \beta(\theta, s)]^{-1}$.
 - (2.5) $P(\theta, s) = \alpha(\theta, s)N(\theta, s)$.
 - (2.6) $Q(\theta, s) = \beta(\theta, s)\tilde{N}(\theta, s)$.
 - (2.7) $\gamma(\theta, s) = (J \circ W(\theta, s))^{-1}P(\theta, s)$.
 - (2.8) $\eta(\theta, s) = (J \circ W(\theta, s))^{-1}Q(\theta, s)$.
 - (2.9) $M(\theta, s) = [\alpha(\theta, s) \mid \gamma(\theta, s) \mid \beta(\theta, s) \mid \eta(\theta, s)]$.
 - (2.10) $[M(\theta, s)]^{-1}$.
3. Compute
 - (3.1) $C(\theta, s) = M^{-1}(\theta, s)(J^{-1}DX) \circ K_0(\theta)$.
 - (3.2) $\tilde{E}(\theta, s) = M^{-1}(\theta, s)E(\theta, s)$.
 - (3.3) $H(\theta, s) = M^{-1}(\theta, s)\beta(\theta, s)$.
4. Compute
 - (4.1) $S(\theta, s) = P^T(\theta, s)[\partial_\omega \gamma(\theta, s) - DX(W(\theta, s))\gamma(\theta, s)]$.
 - (4.2) $B(\theta, s) = Q^T(\theta, s)[\partial_\omega \eta(\theta, s) - DX(W(\theta, s))\eta(\theta, s)]$.
- 5.(5.1) Solve for δ_μ satisfying

$$\int_{\mathbb{T}^\ell} \tilde{E}_2^0 - \left[\int_{\mathbb{T}^\ell} C_2^0 \right] \delta_\mu = 0.$$

- (5.2) Solve for V_2^0 satisfying

$$-\partial_\omega V_2^0 = -\tilde{E}_2^0 + C_2^0 \delta_\mu.$$

Set V_2^0 such that the average is 0.

- 6.(6.1) Compute $S^0(\theta)V_2^0(\theta)$.

- (6.2) Solve for \bar{V}_2^0 satisfying

$$\int_{\mathbb{T}^\ell} \tilde{E}_1^0 - \int_{\mathbb{T}^\ell} C_1^0(\theta)\delta_\mu + \int_{\mathbb{T}^\ell} S^0 V_2^0 + \left[\int_{\mathbb{T}^\ell} S^0 \right] \bar{V}_2^0 = 0.$$

- (6.3) Set $V_2^0 = V_2^0 + \bar{V}_2^0$.

- (6.4) Solve for V_1^0 satisfying

$$-\partial_\omega V_1^0 = -\tilde{E}_1^0 - S^0 V_2^0 + C_1^0 \delta_\mu.$$

- (6.5) Normalize so that $\int_{\mathbb{T}^\ell} V_1^0 = 0$.

7. Solve for V_4^0 satisfying

$$-\lambda V_4^0 - \partial_\omega V_4^0 = -\tilde{E}_4^0 + C_4^0 \delta_\mu.$$

8. Solve for V_3^0 satisfying

$$\lambda V_3^0 - \partial_\omega V_3^0 = -\tilde{E}_3^0 + C_3^0 \delta_\mu - B^0 V_4^0.$$

9.(9.1) Solve for F satisfying

$$-\partial_\omega F - 2\lambda F = H_4^0.$$

(9.2) Solve for G satisfying

$$-\partial_\omega G - 2\lambda G = -\tilde{E}_4^1 + \delta_\mu C_4^1.$$

(9.3) Solve for δ satisfying

$$\left(-\overline{\tilde{E}_3^1} + \delta_\mu \overline{C_3^1} - \overline{B^0 G} - \overline{B^1 V_4^0}\right) + \delta(\overline{H_3^0} - \overline{B^0 F}) = 0.$$

(9.4) Set $V_4^1 = \delta F + G$.

10. (10.1) Solve for V_3^1 satisfying

$$-\partial_\omega V_3^1 = -\tilde{E}_3^1 + \delta H_3^0 + \delta_\mu C_3^1 - B^0 V_4^1 - B^1 V_4^0.$$

(10.2) Normalize so that $\int_{\mathbb{T}e_{II}} V_3^1 = 0$.

(10.3) Solve for V_2^1 satisfying

$$-\partial_\omega V_2^1 - \lambda V_2^1 = -\tilde{E}_2^1 + \delta H_2^0 + \delta_\mu C_2^1.$$

(10.4) Solve for V_1^1 satisfying

$$-\partial_\omega V_1^1 - \lambda V_1^1 = -\tilde{E}_1^1 + \delta H_1^0 + \delta_\mu C_1^1 - S^0 V_2^1 - S^1 V_2^0.$$

11. For $n = 2 \dots L$ do

(11.1) Solve for V_2^n satisfying

$$-\partial_\omega V_2^n - n\lambda V_2^n = -\tilde{E}_2^n(\theta) + \delta H_2^{n-1} + \delta_\mu C_2^n.$$

(11.2) Compute

$$\tilde{S}^n = \sum_{k=0}^n S^{n-k} V_2^k.$$

(11.3) Solve for V_1^n satisfying

$$-\partial_\omega V_1^n - n\lambda V_1^n = -\tilde{E}_1^n + \delta H_1^{n-1} + \delta_\mu C_1^n - \tilde{S}^n.$$

(11.4) Solve for V_4^n satisfying

$$-\partial_\omega V_4^n - (n+1)\lambda V_4^n = -\tilde{E}_4^n + \delta H_4^{n-1} + \delta_\mu C_4^n.$$

(11.5) Compute

$$\tilde{B}^n = \sum_{k=0}^n B^{n-k} V_4^k.$$

(11.6) Solve for V_3^n satisfying

$$-\partial_\omega V_3^n - (n-1)\lambda V_3^n = -\tilde{E}_3^n + \delta H_3^{n-1} + \delta_\mu C_3^n - \tilde{B}^n.$$

12. *Compute*

$$V(\theta) = \sum_{n=0}^L V^n(\theta) s^n.$$

13. *Set* $W \leftarrow W + MV$
 $\lambda \leftarrow \lambda + \delta$
 $\mu \leftarrow \mu + \delta_\mu.$

10.3. A Newton method to compute the whiskers. As in the case of maps, assuming that we have computed exactly an invariant torus $K(\theta)$ with the associated stable bundle V^s (resp. unstable bundle V^u) and the rate of contraction (resp. expansion) λ , we can use a Newton method to compute the whiskers.

We consider the invariance equation (10.1) and we assume that we have an initial approximation W for the whiskers, expressed as a power series

$$W(\theta, s) = \sum_{n=0}^{\infty} W^n(\theta) s^n$$

and such that

$$W^0(\theta) = K(\theta) \quad \text{and} \quad W^1(\theta) = V^s(\theta)$$

(the case unstable is analogous).

Then, it is clear that the error E for the initial approximation W is such that

$$E(\theta, s) = \sum_{n \geq 2} E^n(\theta) s^n$$

because the approximation is exact for the terms of order 0 and 1.

Using the notation introduced in (9.21), the linearized equation for the Newton method is

$$[DX \circ W(\theta, s)] \Delta^{[\geq 2]}(\theta, s) - (\partial_\omega + \lambda s \partial_s) \Delta^{[\geq 2]}(\theta, s) = -E^{[\geq 2]}(\theta, s).$$

Proceeding as in the previous section we can perform the change of coordinates given by the matrix $M(\theta, s)$ in (9.9) and reduce the problem to solving for $V(\theta, s)$ the following equation, which is diagonal in Fourier-Taylor series,

$$\mathcal{R}(\theta, s) V^{[\geq 2]}(\theta, s) - (\partial_\omega + \lambda s \partial_s) V^{[\geq 2]}(\theta, s) = -\tilde{E}^{[\geq 2]}(\theta, s),$$

with $\mathcal{R}(\theta, s)$ given in (10.7) and $\tilde{E}(\theta, s) = M(\theta, s)^{-1} E(\theta, s)$.

Notice that in this case, we do not have to solve the system of equations for order 0 and 1 and we can go straight to order $n \geq 2$. We use subindexes to denote coordinates and superindexes to denote the

order. Hence, for order $n \geq 2$, we need to solve the system of equations

$$\begin{aligned}
& -\partial_\omega V_1^n(\theta) - n\lambda V_1^n(\theta) + \sum_{k=2}^n S^{n-k}(\theta)V_2^k(\theta) = -\tilde{E}_1^n(\theta), \\
& -\partial_\omega V_2^n(\theta) - n\lambda V_2^n(\theta) = -\tilde{E}_2^n(\theta), \\
& -\partial_\omega V_3^n(\theta) - (n-1)\lambda V_3^n(\theta) + \sum_{k=2}^n B^{n-k}(\theta)V_4^k(\theta) = -\tilde{E}_3^n, \\
& -\partial_\omega V_4^n(\theta) - (n+1)\lambda V_4^n(\theta) = -\tilde{E}_4^n.
\end{aligned}
\tag{10.18}$$

Notice that now the solution of (10.18) for $n = 2, 3$ provides an exact solution of the invariance equation up to order 4. That is, if we set

$$V^{[<4]}(\theta, s) = V^2(\theta, s) + V^3(\theta, s)$$

where V^2 and V^3 are obtained by solving equations (10.18), then the improved solution \bar{W} given by

$$\bar{W}(\theta, s) = W(\theta, s) + M(\theta, s)V^{[<4]}(\theta, s),$$

where $M(\theta, s)$ was introduced in (9.9), satisfies that it approximates the solution of the invariance equations with an error \bar{E} such that

$$\bar{E}(\theta, s) = \bar{E}^{[\geq 4]}(\theta, s).$$

This process can be iterated and at each step we solve the invariance equation exactly up to an order which is the double of the one we had for the initial approximation. Thus, if we assume that the initial guess W is such that the error in (10.5) satisfies that

$$E = E^{[\geq L]},$$

then the modified linearized equation for the Newton method is such that

$$\mathcal{R}(\theta, s)V^{[\geq L]}(\theta, s) - (\partial_\omega + \lambda s \partial_s)V^{[\geq L]}(\theta, s) = -\tilde{E}^{[\geq L]}(\theta, s),$$

with $\mathcal{R}(\theta, s)$ given in (10.7). If we solve the system of equations (10.18) for $n = L \dots (2L - 1)$ then the improved \bar{W} is

$$\bar{W}(\theta, s) = W(\theta, s) + M(\theta, s)V^{[< 2L]}(\theta, s),$$

with $M(\theta, s)$ as in (9.9), and the new error \bar{E} satisfies $\bar{E}(\theta, s) = \bar{E}^{[\geq 2L]}(\theta, s)$.

The algorithm in this case is

Algorithm 21 (Computation of whiskers for vector-fields). *Given X , ω as well as K, V^s, λ and an approximate solution W such that*

$$X \circ W(\theta, s) - (\partial_\omega + \lambda s \partial_s)W(\theta, s) = E^{[\geq L]}(\theta, s)$$

with $L \geq 2$ and $W(\theta, 0) = K(\theta)$ and $\partial_s W(\theta, 0) = V^s(\theta)$, perform the following calculations:

1. Compute $E^{[\geq L]}(\theta, s) = X \circ W(\theta, s) - (\partial_\omega + \lambda s \partial_s)W(\theta, s)$.
2. Compute
 - (2.1) $\alpha(\theta, s) = D_\theta W(\theta, s)$.
 - (2.2) $\beta(\theta, s) = \partial_s W(\theta, s)$.
 - (2.3) $N(\theta, s) = [\alpha(\theta, s)^T \alpha(\theta, s)]^{-1}$.
 - (2.4) $\tilde{N}(\theta, s) = [\beta(\theta, s)^T \beta(\theta, s)]^{-1}$.
 - (2.5) $P(\theta, s) = \alpha(\theta, s)N(\theta, s)$.
 - (2.6) $Q(\theta, s) = \beta(\theta, s)\tilde{N}(\theta, s)$.
 - (2.7) $\gamma(\theta, s) = (J \circ W(\theta, s))^{-1}P(\theta, s)$.
 - (2.8) $\eta(\theta, s) = (J \circ W(\theta, s))^{-1}Q(\theta, s)$.
 - (2.9) $M(\theta, s) = [\alpha(\theta, s) \mid \gamma(\theta, s) \mid \beta(\theta, s) \mid \eta(\theta, s)]$.
 - (2.10) $[M(\theta, s)]^{-1}$.
3. Compute

$$\tilde{E}^{[\geq L]}(\theta, s) = M^{-1}(\theta, s)E^{[\geq L]}(\theta, s).$$

4. Compute
 - (4.1) $S(\theta, s) = P^T(\theta, s)[\partial_\omega \gamma(\theta, s) - DX(W(\theta, s))\gamma(\theta, s)]$.
 - (4.2) $B(\theta, s) = Q^T(\theta, s)[\partial_\omega \eta(\theta, s) - DX(W(\theta, s))\eta(\theta, s)]$.
5. For $n = L \dots 2L - 1$ do

- (5.1) Solve for V_2^n satisfying

$$-\partial_\omega V_2^n(\theta) - n\lambda V_2^n(\theta) = -\tilde{E}_2^n(\theta).$$

- (5.2) Compute

$$\tilde{S}^n = \sum_{k=L}^n S^{n-k} V_2^k.$$

- (5.3) Solve for V_1^n satisfying

$$-\partial_\omega V_1^n(\theta) - n\lambda V_1^n(\theta) = -\tilde{E}_1^n - \tilde{S}^n.$$

- (5.4) Solve for V_4^n satisfying

$$-\partial_\omega V_4^n(\theta) - (n+1)\lambda V_4^n(\theta) = -\tilde{E}_4^n.$$

- (5.5) Compute

$$\tilde{B}^n = \sum_{k=L}^n B^{n-k} V_4^k.$$

- (5.6) Solve for V_3^n satisfying

$$-\partial_\omega V_3^n(\theta) - (n-1)\lambda V_3^n(\theta) = -\tilde{E}_3^n - \tilde{B}^n.$$

6. Compute

$$V(\theta, s) = \sum_{n=L}^{2L-1} V^n(\theta) s^n.$$

7. Set $W \leftarrow W + MV$.

11. INITIAL GUESSES OF THE ITERATIVE METHODS

All the methods we have discussed so far are based on considering an initial approximation for the invariant object we are looking for and then using a Newton method to improve this approximation. It remains to discuss how to obtain a good initial guess. We will only discuss the case of maps, the case of vector fields being analogous.

A very standard method, specially when one studies a family of maps $\{F_\varepsilon\}_\varepsilon$ indexed by a parameter ε , is to use a continuation method. Typically, one starts with a value of the parameter (say ε_0) for which the map F_{ε_0} can be studied analytically or a solution is known and then use this solution as an initial guess for the Newton method to compute the solution for the map F_ε for $|\varepsilon - \varepsilon_0|$ small enough. An improvement of this method could be to extrapolate an initial guess for F_ε from the previous computed solutions for different values of ε .

A very classical example is the case of maps which are a perturbation of an integrable one, for which the dynamics is very simple and well known. Nevertheless, we recall that the methods presented in this paper do not require the map to be close to the integrable case. Therefore, one needs to develop other methods to compute an initial approximation.

An alternative to get an initial approximation for the computation of primary invariant tori is to use the Percival variational principle ([Per79]). It is easy to see that primary invariant tori correspond to critical points of a certain functional. Under some convexity assumptions, one can prove that primary tori are minimizers. In this case, one can find minimizers of this functional using minimization algorithms such as conjugate gradient methods. One problem of this method is that the precision for the solution will be $CE^{1/2}$, where E is the error for the computation of the functional (roundoff and truncation). Nevertheless, one can use the minimizer obtained this way as an initial guess for the Newton method.

None of the methods above mentioned work for secondary tori. To obtain an initial guess to compute secondary tori can be done via averaging methods (see [DH08, DLS06]). Another possibility consists of computing the rotation number for several points and look for a point p that rotates with a frequency which is close to the Diophantine frequency of the invariant torus we are looking for. Of course, taking some iterates $\{F^{(n)}(p)\}_{n=0}^N$ of this point we can obtain an initial guess for the Newton method.

However, we already mentioned in the previous sections that in one step of the Newton method we need to perform some operations in Fourier space. Hence, we need to apply the FFT algorithm and this requires to have the values on an equidistant grid which is not the case

when one considers iterates of a point p by a map F . In order to get rid of this problem, one can use cubic interpolation and calculate the values of the interpolating function at the points $\{\theta_i\}_{i=0}^N$, taking into account the periodicity of θ . This is the method we used in the numerical implementation we carried out. See Section 12.1.3 for a more detailed description. An algorithm that can perform Fourier Transforms on grids of non equally spaced points is the USFFT.

12. NUMERICAL EXAMPLES

In this section we will discuss some implementation details as well as some efficiency properties and we will show some preliminary examples of the results obtained.

The algorithms have been implemented in C language and have been run under the Linux environment. For the computation of the FFT we used the `fftw3` library (see <http://www.fftw.org/>) and we also used some of the functions available in the LAPACK and BLAS routines (see <http://www.netlib.org/lapack/>).

12.1. Computation of primary and secondary KAM tori for the standard map. The first example that we consider is the very well known standard map introduced by Chirikov [Chi79]. It is a 2D-symplectic map from the cylinder $\mathbb{R} \times \mathbb{T}$ to itself and it is given by

$$(12.1) \quad \begin{aligned} \bar{p} &= p - \varepsilon V'(q), \\ \bar{q} &= q + \bar{p} \pmod{1}, \end{aligned}$$

where ε is a positive parameter and V is a 1-periodic smooth function called the potential, which is given by

$$(12.2) \quad V(q) = \frac{1}{(2\pi)^2} \cos(2\pi q).$$

We refer to (p, q) as the action-angle variables.

12.1.1. Computation of primary invariant tori. We start from the integrable case $\varepsilon = 0$ where we have a 1-parameter family of 1-dimensional invariant tori indexed by the frequency of rotation. The celebrated KAM Theorem (see [Lla01a] for a survey) ensures that those invariant tori present in the unperturbed system with a Diophantine frequency will persist under the perturbation for $\varepsilon < \varepsilon_0$ where ε_0 is a certain critical value at which they break down (phenomenon usual referred as breakdown of analyticity).

In the example of the standard map, we first considered the invariant curve with a frequency of rotation equal to the golden mean, that is $\omega_g = (\sqrt{5} - 1)/2$, which is conjectured to be the most robust curve. The Greene's method ([Gre79]) estimates that for this curve the critical value ε_0 is close to 0.971635406.

We follow a continuation method, starting from the integrable case $\varepsilon = 0$ and then increasing the parameter ε by steps of size 0.01 up to where our Newton method fails to converge, which turned out to be $\varepsilon = 0.96$. For the computations we have used $N = 2^{11}$ Fourier coefficients and each step of the continuation method takes 0.0305 seconds in average in a Intel(R) Core(TM)2, 2.15 GHz. The errors in the functional equation (3.2) are smaller than 10^{-10} . We show some of the curves obtained in Figure 3.

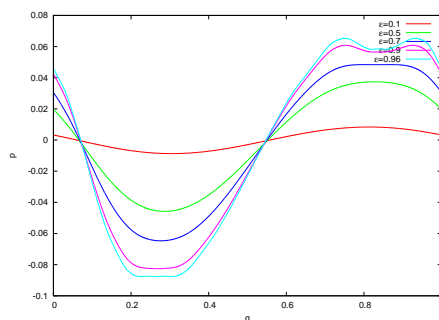


FIGURE 3. The invariant curve associated to the golden mean frequency ω_g for the standard map for different values of ε . Notice that they are shifted to have 0 offset.

We computed also invariant tori corresponding to frequencies of the form

$$(12.3) \quad \omega_\alpha = \frac{1}{\alpha + \omega_g}, \quad \alpha \in \mathbb{N}.$$

More precisely, we applied our method to the cases $\alpha = 5$ and $\alpha = 50$ and we managed to continue the invariant curve up to $\varepsilon = 0.73$ and $\varepsilon = 0.068$, respectively, with similar time running estimates as in the previous case. In Figure 4 we display some of these curves for different values of ε .

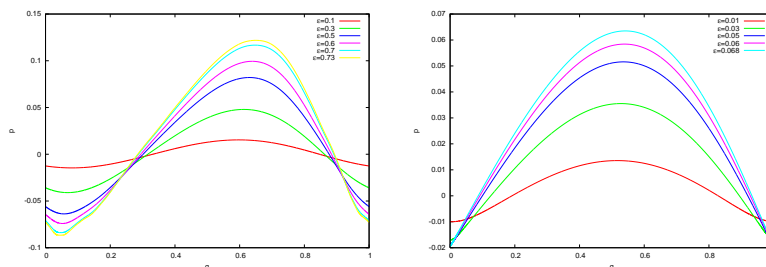


FIGURE 4. The invariant curves associated to the frequencies ω_5 (left) and ω_{50} (right) for the standard map for different values of ε . Notice that they are shifted to have 0 offset.

12.1.2. *Standard-like maps.* We also considered the case of a system of the form (12.1) but with a potential V which has an infinite number of harmonics. We chose V such that

$$(12.4) \quad V'(q) = 1 - \frac{1}{1 - 0.9 \sin(2\pi q)},$$

and we studied the invariant torus for this system associated to the golden mean frequency.

We computed the invariant curves corresponding to the golden mean frequency using a continuation method with smaller steps (in this case the step size was 0.001) for different values of ε (the program stopped at $\varepsilon = 23 \cdot 10^{-3}$). We used $N = 2^{11}$ Fourier modes and it took 0.0406 seconds to perform one step of the continuation method. In Figure 5 we show some of the curves computed for different values of ε .

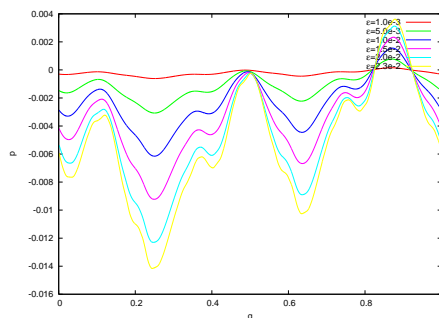


FIGURE 5. The invariant curves associated to the frequency ω_g for the standard-like map with a potential satisfying (12.4) for different values of ε . Notice that they are shifted to have 0 offset.

12.1.3. *Computation of secondary invariant tori.* As we already mentioned, our method can compute also secondary KAM tori. Recall that, in this case, we cannot use a continuation method starting from the integrable case $\varepsilon = 0$ since these tori do not exist in the unperturbed case (they are created by the perturbation). In order to get an initial approximation we will use some of the techniques described in Section 11.

Another point that we need to take into account are the periodicities of the functions that appear in the Newton procedure, the matrix I introduced in (3.6) having rank 0.

Let us recall, first, that for $\varepsilon \neq 0$ the standard map has two fixed points corresponding to $(0, 0)$ and $(0, 1/2)$ and, by a simple linear stability analysis, it is easy to see that if ε is small they are hyperbolic and elliptic fixed points, respectively. Moreover, in a neighborhood of the elliptic fixed point delimited by the stable and unstable manifolds of the hyperbolic fixed point, there are born a family of invariant curves

of different topology of the ones that existed in the unperturbed case. These curves are contractible to a point and form secondary KAM tori.

In order to get an initial approximation for these tori, we will first compute the rotation number for several points on the axis $p = 0$ between the hyperbolic and the elliptic fixed points. Recall that the rotation number provides information about how much turns in average every iterate of the standard map. We use a method by C. Simó that computes an upper and a lower bound for the rotation number. For the sake of completeness we include it here.

- Compute n iterates of a point x^0 on the axis $p = 0$. For each iterate x^i , $i = 0, \dots, n$, compute the number of turns n_i and the angle θ_i (modulus 2π) that form the points x^0 , x^* and x^i , where x^* is the elliptic fixed point.
- Sort the angles θ_i in increasing order and keep the information by arranging the indexes i of the angles (we used the “quicksort” algorithm).
- Take two consecutive indexes i and j in the arrangement and perform the following computation:
 - If $i < j \Rightarrow \rho > \frac{n_j - n_i}{j - i}$.
 - If $i > j \Rightarrow \rho < \frac{n_i - n_j}{i - j}$.

Hence, we can obtain ρ_{\min} and ρ_{\max} such that $\rho_{\min} \leq \rho \leq \rho_{\max}$. Then we can approximate the rotation number ρ , either by ρ_{\min} or ρ_{\max} with an error of order $1/n^2$, where n is the number of iterates considered.

In Figure 6 we have plotted the computed rotation numbers with 2000 iterates for $\varepsilon = 0.1$ and $\varepsilon = 0.5$. In the cases where ρ_{\min} is bigger than ρ_{\max} , the invariant curve has been destroyed.

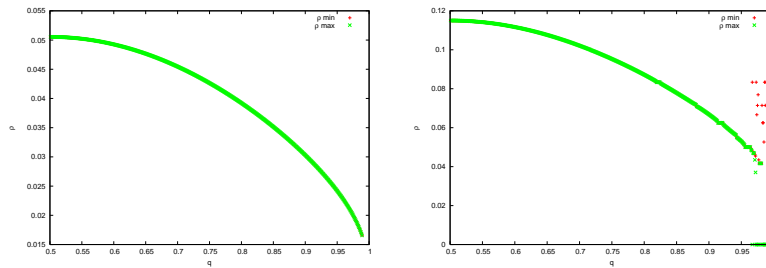


FIGURE 6. The rotation number for different values on the $p = 0$ axis under the standard map (12.1) for $\varepsilon = 0.1$ (left) and $\varepsilon = 0.5$ (right). Notice that the hyperbolic and elliptic fixed points correspond to $(0, 1)$ and $(1/2, 0)$, respectively.

We fixed a Diophantine frequency in the interval of allowed frequencies and using a bisection method we found a point x^0 on the axis $p = 0$

which rotates with a frequency which is close to the Diophantine frequency. Then, we computed several iterates of this point and interpolating using splines, we have obtained an approximation of the parameterization of the secondary invariant torus evaluated on an equidistant grid. We used the cubic interpolation routines `spline` and `seval` from [FMM77], taking into account the periodicity of θ .

In particular, we computed the invariant tori corresponding to the frequencies $3/40\omega_g \approx 0.04635026$ and $0.18\omega_g \approx 0.111246$ starting at $\varepsilon = 0.1$ and $\varepsilon = 0.5$, respectively. Using a continuation method (with a step size of 0.001) we computed the invariant tori associated to the corresponding frequencies up to $\varepsilon = 0.401$ and $\varepsilon = 0.853$, respectively. We used $N = 2^9$ Fourier modes and each step of the continuation method takes 0.01469 seconds in average in an Intel(R) Core(TM)2, 2.15 GHz. In Figure 7 we show the computed secondary tori.

Again, as in the case of primary tori, the invariant tori are computed with errors in the functional equations (3.2) smaller than 10^{-10} .

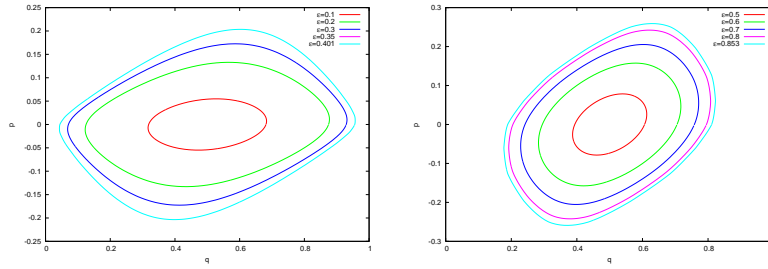


FIGURE 7. The secondary tori associated to the frequencies $3/40\omega_g$ (left) and $0.18\omega_g$ (right) of the standard map for some values of the parameter ε .

12.2. 4D-symplectic maps: The Froeschlé map. In this section we will discuss the computation of maximal and hyperbolic invariant tori for the Froeschlé map. The Froeschlé map is a 4D symplectic map defined on $\mathbb{T}^2 \times \mathbb{R}^2$, consisting of two coupled standard maps. It was introduced by Froeschlé in [Fro72] and it is given by

$$\begin{aligned}
 \bar{p}_1 &= p_1 - \varepsilon \left(\frac{\lambda_1}{2\pi} \sin(2\pi q_1) + \frac{\lambda_{12}}{2\pi} \sin(2\pi(q_1 + q_2)) \right), \\
 \bar{p}_2 &= p_2 - \varepsilon \left(\frac{\lambda_2}{2\pi} \sin(2\pi q_2) + \frac{\lambda_{12}}{2\pi} \sin(2\pi(q_1 + q_2)) \right), \\
 \bar{q}_1 &= q_1 + \bar{p}_1 \pmod{1}, \\
 \bar{q}_2 &= q_2 + \bar{p}_2 \pmod{1},
 \end{aligned}
 \tag{12.5}$$

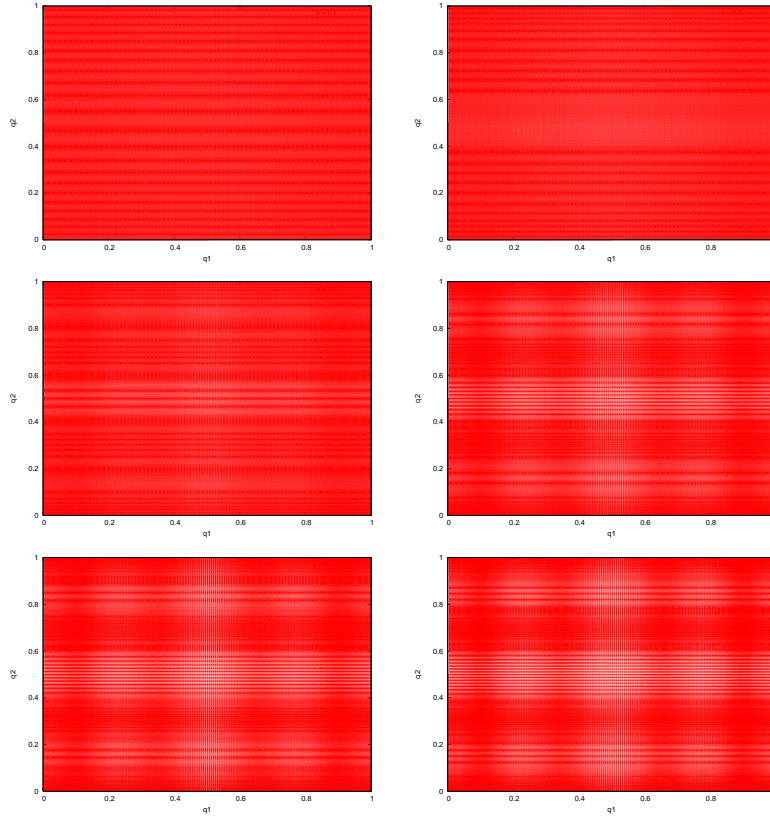


FIGURE 8. The angular variables (q_1, q_2) for the primary maximal tori of the Froeschlé map associated to the frequency ω_u for different values of the parameter ε

where λ_{12} is a coupling parameter. The potential for this case is given by

$$V(q_1, q_2) = - \left(\frac{\lambda_1}{(2\pi)^2} \cos(2\pi q_1) + \frac{\lambda_2}{(2\pi)^2} \cos(2\pi q_2) + \frac{\lambda_{12}}{(2\pi)^2} \cos(2\pi(q_1 + q_2)) \right).$$

Notice that when $\lambda_{12} = 0$, the problem reduces to two uncoupled 2-dimensional standard maps.

For the Froeschlé map, we can consider maximal invariant tori, which are 2-dimensional invariant tori or hyperbolic invariant tori, that is 1-dimensional invariant tori with associated stable and unstable manifolds.

12.2.1. *Computation of maximal tori.* We will follow the behaviour of invariant tori associated to a certain frequency $\omega = (\omega_1, \omega_2)$ as ε increases. Of course, the choice of the frequency vector strongly influences the dynamics. In this study, we will just restrict to one particular case in order to show the way the method works.

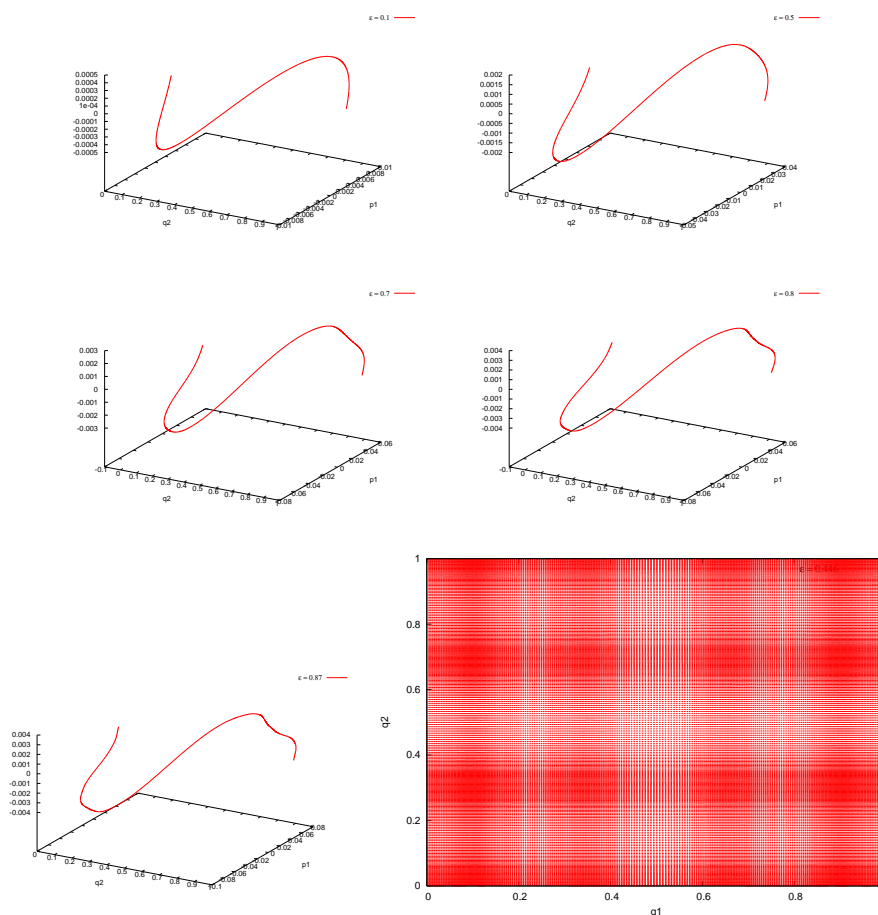


FIGURE 9. The $q_1q_2p_1$ projection of the primary hyperbolic invariant tori of the Froeschlé map associated to the golden mean frequency for different values of the parameter ε .

In order to get a 2-dimensional frequency vector which is Diophantine, one needs to use some results in number theory. In this example, we considered one of the rotation vectors studied in [CFL04], which is given by

$$(12.6) \quad \omega_u = \left(\frac{1}{s}, s - 1 \right) = (0.754877\dots, 0.324717\dots)$$

where $s = 1.8392\dots$ is the real root of the polynomial of degree 3

$$t^3 - t^2 - t - 1 = 0.$$

In [CFL04] the authors studied also tori with other frequencies. We also refer to [HS95] for a study of the breakdown of 2-dimensional invariant tori of the Froeschlé map.

We studied the case when $\lambda_1 = \lambda_2 = 1$ and $\lambda_{12} = 0.001$, starting from $\varepsilon = 0$. Recall that for $\varepsilon = 0$, the system consists of two uncoupled

integrable standard maps. Hence, the invariant tori are given by the cross product of the invariant tori of each of the subsystems, which are trivial ($p_i = \text{cst}$, $i = 1, 2$). Using a continuation method we computed the invariant tori associated to the frequency ω_u in (12.6), for different values of ε up to $\varepsilon = 0.446$. In [CFL04], the breakdown for this torus was estimated to be close to 0.55.

Recall that we are computing and storing a function K of two variables. For the computations we used $N = 2^8 \times 2^8$ Fourier modes and one step of the continuation method takes in average 4.5 seconds in an Intel(R) Core(TM)2, 2.15 GHz.

We only computed primary KAM tori, hence the matrix I in (3.6) is the identity. For the case of secondary tori, there are different possibilities: we can take $I = \text{diag}(1, 0)$ or $I = \text{diag}(0, 1)$, which correspond to invariant tori which in the case $\lambda_{12} = 0$ (the standard maps are uncoupled) and $\lambda_1 \neq 0$ or $\lambda_2 \neq 0$ consist of the cartesian product of a primary torus of one of the standard maps and a secondary torus of the other standard map. Another possibility is to consider $I = \text{diag}(0, 0)$, which in the case $\lambda_{12} = 0$ (the standard maps are uncoupled) and $\lambda_1 \neq 0$ and $\lambda_2 \neq 0$, the invariant tori consist of the cartesian product of two secondary tori, each one corresponding to a secondary invariant torus of each of the standard maps. It is clear how to obtain an initial guess to compute these tori: we start with $\lambda_{12} = 0$ and we obtain the secondary invariant torus for each of the subsystems using the method described in Sections 12.1.3. Then, we use a continuation method increasing λ_{12} .

In Figure 8, the angular components (q_1, q_2) of the invariant tori obtained are drawn and one can see the metamorphoses with respect to the parameter ε .

While the breakdown of invariant curves is rather well understood for the case of 2-D maps, there are very few results concerning higher dimensions. In the future, we plan to pursue the investigations to study the breakdown of 2-D invariant tori of the Froeschlé map for a wide range of frequencies.

12.2.2. *Computation of whiskered tori.* Recall that whiskered invariant tori for the Froeschlé map are 1-D invariant tori with associated rank-1 stable and unstable manifolds. Starting with $\lambda_2 = 0$, $\lambda_{12} = 0$ and $\lambda_1 = 1$ and ε small (in the computations we started with $\varepsilon = 0.01$, so that the two standard maps are uncoupled), the whiskered tori are given by the cross product of the hyperbolic fixed point of one of the coupled standard maps and a primary invariant torus of the second one. Moreover, the stable and unstable manifolds of the invariant torus are inherited from the ones of the hyperbolic fixed point. Therefore, the invariant torus has constant tangent bundles, independent of θ .

In the next step, we set $\lambda_2 = 1$ and $\lambda_{12} = 0.1$ and performed a continuation method increasing ε by a step size 0.01. For the continuation method, the (un)stable bundle computed previously is used as an approximation for the (un)stable bundle of the increased parameter, to perform a change of coordinates to the cocycle in order to avoid the *straddle the saddle phenomenon* discussed in Section 6.5. The computations used $N = 2^{12}$ Fourier modes and reached up to $\varepsilon = 0.87$. Since these tori have rank-1 bundles, we used the algorithms described in Section 6.6 and 7.0.1. In Figure 9 we display the figures obtained. They correspond to the Froeschlé map obtained by performing the change $(p, q) = (-p, -q)$ in the map (12.5).

ACKNOWLEDGEMENTS

The work of R.L. and G.H. has been partially supported by NSF grants.

G.H. has also been supported by the Spanish Grant MTM2006-00478 and the Spanish Fellowship AP2003-3411.

We thank Á Haro, C. Simó for several discussions and for comments on the paper. The final version was written while we were visiting CRM during the Research Programme *Stability and Instability in Mechanical Systems (SIMS08)*, for whose hospitality we are very grateful. YS would like to thank the hospitality of the department of Mathematics of University of Texas at Austin, where part of this work were carried out.

REFERENCES

- [ALD83] S. Aubry and P. Y. Le Daeron. The discrete Frenkel-Kontorova model and its extensions. I. Exact results for the ground-states. *Phys. D*, 8(3):381–422, 1983.
- [Ano69] D. V. Anosov. *Geodesic flows on closed Riemann manifolds with negative curvature*. American Mathematical Society, Providence, R.I., 1969.
- [Arn63] V. I. Arnol'd. Proof of a theorem of A. N. Kolmogorov on the invariance of quasi-periodic motions under small perturbations. *Russian Math. Surveys*, 18(5):9–36, 1963.
- [Arn64] V.I. Arnold. Instability of dynamical systems with several degrees of freedom. *Sov. Math. Doklady*, 5:581–585, 1964.
- [Aub83] S. Aubry. The twist map, the extended Frenkel-Kontorova model and the devil's staircase. *Phys. D*, 7(3):240–258, 1983.
- [Bro65] C. G. Broyden. A class of methods for solving nonlinear simultaneous equations. *Math. Comp.*, 19:577–593, 1965.
- [CC07] Alessandra Celletti and Luigi Chierchia. KAM stability and celestial mechanics. *Mem. Amer. Math. Soc.*, 187(878):viii+134, 2007.
- [CFL03a] Xavier Cabré, Ernest Fontich, and Rafael de la Llave. The parameterization method for invariant manifolds. I, II. *Indiana Univ. Math. J.*, 52(2):283–328, 329–360, 2003.
- [CFL03b] Xavier Cabré, Ernest Fontich, and Rafael de la Llave. The parameterization method for invariant manifolds. I. Manifolds associated to non-resonant subspaces. *Indiana Univ. Math. J.*, 52(2):283–328, 2003.

- [CFL03c] Xavier Cabré, Ernest Fontich, and Rafael de la Llave. The parameterization method for invariant manifolds. II. Regularity with respect to parameters. *Indiana Univ. Math. J.*, 52(2):329–360, 2003.
- [CFL04] A. Celletti, C. Falcolini, and U. Locatelli. On the break-down threshold of invariant tori in four dimensional maps. *Regul. Chaotic Dyn.*, 9(3):227–253, 2004.
- [CFL05] Xavier Cabré, Ernest Fontich, and Rafael de la Llave. The parameterization method for invariant manifolds. III. Overview and applications. *J. Differential Equations*, 218(2):444–515, 2005.
- [Chi79] B.V. Chirikov. A universal instability of many-dimensional oscillator systems. *Phys. Rep.*, 52(5):264–379, 1979.
- [CL08] R. Calleja and R. de la Llave. Fast numerical computation of quasi-periodic equilibrium states in 1-d statistical mechanics, including twist maps. 2008. Manuscript.
- [DH08] Amadeu Delshams and Gemma Huguet. Geography of resonances and Arnold diffusion in a priori unstable Hamiltonian systems. *Preprint*, http://www.ma.utexas.edu/mp_arc, 08–227, 2008.
- [dlLR91] R. de la Llave and D. Rana. Accurate strategies for K.A.M. bounds and their implementation. In *Computer Aided Proofs in Analysis (Cincinnati, OH, 1989)*, pages 127–146. Springer, New York, 1991.
- [DLS03] Amadeu Delshams, Rafael de la Llave, and Tere M. Seara. A geometric mechanism for diffusion in Hamiltonian systems overcoming the large gap problem: announcement of results. *Electron. Res. Announc. Amer. Math. Soc.*, 9:125–134 (electronic), 2003.
- [DLS06] A. Delshams, R. de la Llave, and T. M. Seara. A geometric mechanism for diffusion in Hamiltonian systems overcoming the large gap problem: heuristics and rigorous verification on a model. *Mem. Amer. Math. Soc.*, 179(844):viii+141, 2006.
- [Dua94] Pedro Duarte. Plenty of elliptic islands for the standard family of area preserving maps. *Ann. Inst. H. Poincaré Anal. Non Linéaire*, 11(4):359–409, 1994.
- [DVV02] Luca Dieci and Erik S. Van Vleck. Lyapunov spectral intervals: theory and computation. *SIAM J. Numer. Anal.*, 40(2):516–542 (electronic), 2002.
- [Eli01] L. H. Eliasson. Almost reducibility of linear quasi-periodic systems. In *Smooth ergodic theory and its applications (Seattle, WA, 1999)*, volume 69 of *Proc. Sympos. Pure Math.*, pages 679–705. Amer. Math. Soc., Providence, RI, 2001.
- [ER85] J.-P. Eckmann and D. Ruelle. Ergodic theory of chaos and strange attractors. *Rev. Modern Phys.*, 57(3, part 1):617–656, 1985.
- [Fen72] Neil Fenichel. Persistence and smoothness of invariant manifolds for flows. *Indiana Univ. Math. J.*, 21:193–226, 1971/1972.
- [Fen77] N. Fenichel. Asymptotic stability with rate conditions. II. *Indiana Univ. Math. J.*, 26(1):81–93, 1977.
- [Fen74] N. Fenichel. Asymptotic stability with rate conditions. *Indiana Univ. Math. J.*, 23:1109–1137, 1973/74.
- [FGB98] F. Fassò, M. Guzzo, and G. Benettin. Nekhoroshev-stability of elliptic equilibria of hamiltonian systems. *Comm. Math. Phys.*, 197(2):347–360, 1998.

- [FKW01] Bassam Fayad, Anatole Katok, and Alistar Windsor. Mixed spectrum reparameterizations of linear flows on \mathbb{T}^2 . *Mosc. Math. J.*, 1(4):521–537, 644, 2001. Dedicated to the memory of I. G. Petrovskii on the occasion of his 100th anniversary.
- [FLS07] E. Fontich, R. de la Llave, and Y. Sire. Construction of invariant whiskered tori by a parametrization method. part I: Maps and flows in finite dimensions. Preprint, 2007.
- [FMM77] George E. Forsythe, Michael A. Malcolm, and Cleve B. Moler. *Computer methods for mathematical computations*. Prentice-Hall Inc., Englewood Cliffs, N.J., 1977. Prentice-Hall Series in Automatic Computation.
- [Fro72] C. Froeschlé. Numerical study of a four-dimensional mapping. *Astronom. and Astrophys.*, 16:172–189, 1972.
- [GFB98] M. Guzzo, F. Fassò, and G. Benettin. On the stability of elliptic equilibria. *Math. Phys. Electron. J.*, 4:Paper 1, 16 pp. (electronic), 1998.
- [GJSM01a] G. Gómez, À. Jorba, C. Simó, and J. Masdemont. *Dynamics and mission design near libration points. Vol. III*, volume 4 of *World Scientific Monograph Series in Mathematics*. World Scientific Publishing Co. Inc., River Edge, NJ, 2001. Advanced methods for collinear points.
- [GJSM01b] G. Gómez, À. Jorba, C. Simó, and J. Masdemont. *Dynamics and mission design near libration points. Vol. IV*, volume 5 of *World Scientific Monograph Series in Mathematics*. World Scientific Publishing Co. Inc., River Edge, NJ, 2001. Advanced methods for triangular points.
- [GL06] Marian Gidea and Rafael de la Llave. Topological methods in the instability problem of Hamiltonian systems. *Discrete Contin. Dyn. Syst.*, 14(2):295–328, 2006.
- [Gra74] Samuel M. Graff. On the conservation of hyperbolic invariant tori for Hamiltonian systems. *J. Differential Equations*, 15:1–69, 1974.
- [Gre79] J. M. Greene. A method for determining a stochastic transition. *Jour. Math. Phys.*, 20:1183–1201, 1979.
- [GVL96] Gene H. Golub and Charles F. Van Loan. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, third edition, 1996.
- [Hal75] O. H. Hald. On a Newton-Moser type method. *Numer. Math.*, 23:411–426, 1975.
- [Har08] Alex Haro. Automatic differentiation tools in computational dynamical systems. Manuscript, 2008.
- [Her83] Michael-R. Herman. *Sur les courbes invariantes par les difféomorphismes de l’anneau. Vol. 1*, volume 103 of *Astérisque*. Société Mathématique de France, Paris, 1983. With an appendix by Albert Fathi, With an English summary.
- [Her92] M.-R. Herman. On the dynamics of Lagrangian tori invariant by symplectic diffeomorphisms. In *Progress in Variational Methods in Hamiltonian Systems and Elliptic Equations (L’Aquila, 1990)*, pages 92–112. Longman Sci. Tech., Harlow, 1992.
- [HL00] A. Haro and R. de la Llave. New mechanisms for lack of equipartition of energy. *Phys. Rev. Lett.*, 89(7):1859–1862, 2000.
- [HL06a] À. Haro and R. de la Llave. Manifolds on the verge of a hyperbolicity breakdown. *Chaos*, 16(1):013120, 8, 2006.
- [HL06b] À. Haro and R. de la Llave. A parameterization method for the computation of invariant tori and their whiskers in quasi-periodic maps:

- numerical algorithms. *Discrete Contin. Dyn. Syst. Ser. B*, 6(6):1261–1300 (electronic), 2006.
- [HL06c] A. Haro and R. de la Llave. A parameterization method for the computation of invariant tori and their whiskers in quasi-periodic maps: rigorous results. *J. Differential Equations*, 228(2):530–579, 2006.
- [HL07] A. Haro and R. de la Llave. A parameterization method for the computation of whiskers in quasi periodic maps: numerical implementation and examples. *SIAM Jour. Appl. Dyn. Syst.*, 6(1):142–207, 2007.
- [HPS77] M.W. Hirsch, C.C. Pugh, and M. Shub. *Invariant manifolds*, volume 583 of *Lecture Notes in Math*. Springer-Verlag, Berlin, 1977.
- [HS95] A. Haro and C. Simó. A numerical study of the breakdown of invariant tori in 4D symplectic maps. In *XIV CEDYA/IV Congress of Applied Mathematics (Spanish)(Vic, 1995)*, page 9 pp. (electronic). Univ. Barcelona, Barcelona, 1995.
- [JO05] Àngel Jorba and Estrella Olmedo. A parallel method to compute quasi-periodic solutions. In *EQUADIFF 2003*, pages 181–183. World Sci. Publ., Hackensack, NJ, 2005.
- [JO08] Àngel Jorba and Estrella Olmedo. On the computation of reducible invariant tori in a parallel computer. *Preprint*, (08-2), 2008.
- [Knu97] Donald E. Knuth. *The art of computer programming. Vol. 2: Seminumerical algorithms*. Addison-Wesley Publishing Co., Reading, Mass.-London-Don Mills, Ont, third revised edition, 1997.
- [Kri99a] Raphaël Krikorian. C^0 -densité globale des systèmes produits-croisés sur le cercle réductibles. *Ergodic Theory Dynam. Systems*, 19(1):61–100, 1999.
- [Kri99b] Raphaël Krikorian. Réductibilité des systèmes produits-croisés à valeurs dans des groupes compacts. *Astérisque*, (259):vi+216, 1999.
- [LGJV05] R. de la Llave, A. González, À. Jorba, and J. Villanueva. KAM theory without action-angle variables. *Nonlinearity*, 18(2):855–895, 2005.
- [Lla01a] Rafael de la Llave. Remarks on Sobolev regularity in Anosov systems. *Ergodic Theory Dynam. Systems*, 21(4):1139–1180, 2001.
- [Lla01b] Rafael de la Llave. A tutorial on KAM theory. In *Smooth ergodic theory and its applications (Seattle, WA, 1999)*, pages 175–292. Amer. Math. Soc., Providence, RI, 2001.
- [LW04] R. de la Llave and C. E. Wayne. Whiskered and low dimensional tori in nearly integrable Hamiltonian systems. *Math. Phys. Electron. J.*, 10:Paper 5, 45 pp. (electronic), 2004.
- [LY05] Yong Li and Yingfei Yi. Persistence of hyperbolic tori in Hamiltonian systems. *J. Differential Equations*, 208(2):344–387, 2005.
- [McG90] R. McGehee. A note on the Moser-Hald variation of Newton’s method. In *Analysis, et cetera*, pages 495–499. Academic Press, Boston, MA, 1990.
- [McK82] R. S. McKay. *Renormalisation in Area Preserving Maps*. PhD thesis, Princeton University, 1982.
- [Mos73] J. Moser. *Stable and random motions in dynamical systems*. Princeton University Press, Princeton, N. J., 1973. With special emphasis on celestial mechanics, Hermann Weyl Lectures, the Institute for Advanced Study, Princeton, N. J, Annals of Mathematics Studies, No. 77.
- [MS89] Kenneth R. Meyer and George R. Sell. Mel’nikov transforms, Bernoulli bundles, and almost periodic perturbations. *Trans. Amer. Math. Soc.*, 314(1):63–105, 1989.

- [OP08] A. Olvera and N. P. Petrov. Regularity properties of critical invariant circles of twist maps and their universality. *SIAM J. Appl. Dyn. Syst.*, 2008.
- [Ose68] V. I. Oseledec. A multiplicative ergodic theorem. Characteristic Ljapunov, exponents of dynamical systems. *Trudy Moskov. Mat. Obsč.*, 19:179–210, 1968.
- [Per79] I. C. Percival. A variational principle for invariant tori of fixed frequency. *J. Phys. A*, 12(3):L57–L60, 1979.
- [Pes04] Yakov B. Pesin. *Lectures on partial hyperbolicity and stable ergodicity*. Zurich Lectures in Advanced Mathematics. European Mathematical Society (EMS), Zürich, 2004.
- [PTVF92] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical recipes in C*. Cambridge University Press, Cambridge, second edition, 1992. The art of scientific computing.
- [Pui02] J. Puig. Reducibility of linear differential equations with quasi-periodic coefficients: a survey. http://www.ma.utexas.edu/mp_arc, 02–246, 2002.
- [RC95] M. J. Raković and Shih-I Chu. New integrable systems: hydrogen atom in external fields. *Phys. D*, 81(3):271–279, 1995.
- [RC97] M. J. Raković and Shih-I Chu. Phase-space structure of a new integrable system related to hydrogen atoms in external fields. *J. Phys. A*, 30(2):733–753, 1997.
- [Rüs75] H. Rüssmann. On optimal estimates for the solutions of linear partial differential equations of first order with constant coefficients on the torus. In *Dynamical Systems, Theory and Applications (Battelle Rencontres, Seattle, Wash., 1974)*, pages 598–624. Lecture Notes in Phys., Vol. 38, Berlin, 1975. Springer.
- [Ryc92] Marek Rychlik. Renormalization of cocycles and linear ODE with almost-periodic coefficients. *Invent. Math.*, 110(1):173–206, 1992.
- [Sac78] R.J. Sacker. Existence of dichotomies and invariant splittings for linear differential systems. IV. *J. Differential Equations*, 27(1):106–137, 1978.
- [Sim99] Carles Simó, editor. *Hamiltonian systems with three or more degrees of freedom*, Dordrecht, 1999. Kluwer Academic Publishers Group.
- [Sim00] Carles Simó. private communication. 2000.
- [SS74] R.J. Sacker and G.R. Sell. Existence of dichotomies and invariant splittings for linear differential systems. I. *J. Differential Equations*, 15:429–458, 1974.
- [SS76a] R.J. Sacker and G.R. Sell. Existence of dichotomies and invariant splittings for linear differential systems. II. *J. Differential Equations*, 22(2):478–496, 1976.
- [SS76b] R.J. Sacker and G.R. Sell. Existence of dichotomies and invariant splittings for linear differential systems. III. *J. Differential Equations*, 22(2):497–522, 1976.
- [Zeh75] E. Zehnder. Generalized implicit function theorems with applications to some small divisor problems. I. *Comm. Pure Appl. Math.*, 28:91–140, 1975.

RAFAEL DE LA LLAVE
 DEPARTMENT OF MATHEMATICS
 THE UNIVERSITY OF TEXAS AT AUSTIN
 AUSTIN, TX 78712

E-mail address: llave@math.utexas.edu

GEMMA HUGUET
CENTRE DE RECERCA MATEMÀTICA
APARTAT 50, 08193 BELLATERRA (BARCELONA), SPAIN
E-mail address: `gemma.huguet@upc.edu`

YANNICK SIRE
UNIVERSITÉ PAUL CÉZANNE
LABORATOIRE LATP UMR 6632
MARSEILLE, FRANCE
AND
LABORATOIRE PONCELET UMI 2615
MOSCOW, RUSSIA
E-mail address: `sire@cmi.univ-mrs.fr`