UNIVERSITAT DE BARCELONA

Treball final de grau

GRAU DE MATEMÀTIQUES

Facultat de Matemàtiques i Informàtica
Universitat de Barcelona

QUASI-PERIODIC SOLUTIONS IN QUASI-PERIODIC SYSTEMS VIA FOURIER TRANSFORMS

Autor: Eric Sandín Vidal

Director: Dr. Alex Haro Provinciale
Realitzat a: Departament de Matemàtica Aplicada i Anàlisi

Barcelona, 27 de juny de 2018

Contents

Abstract

Within the field of dynamical systems, one shall find a special sort of systems, systems that revolve around external perturbations, either periodic or quasi-periodic, the so called skew-product dynamical systems. Even though the study of these systems can be a very helpful source of tools for an engineering or more practical work that may present systems alike, the main focus of this project is of a more theoretical nature. What is about to be presented is an approach to response solutions of such perturbated systems and the proof of existence and uniqueness of invariant fiberwise hyperbolic tori given an approximate torus of such characteristics, and the further expression of the proof's conditions in computable terms via Fourier transforms. Besides the theoretical part, it is also provided an algorithm and its respective computational implementation that allows to simplify the expression of the linear dynamics of a system under a quasi-periodic perturbation into a diagonal matrix at the cost of an error that will significantly decrease at each step of the algorithm.

2010 Mathematics Subject Classification. 37C55, 37D10, 65P99

Acknowledgements

It has been a great journey writing this project, as well as my pass through the whole degree. Throughout the years I have met tons of great people who, in one way or another, have helped me succesfully understand so many concepts and develop so many tasks. Between all of them, I would specially like to thank my beloved friend Sergio Gor for always offering his help for programming or harmonic analysis issues and for our long talks about general science topics, my fellow Pol Casellas for being with me during the whole degree and supporting me in tough academic times, my colleague Alexis Arraz for helping me countless times in several subjects and without whom I might not have completed my degree, and Andrea Ruiz for, as always, helping me designing and drawing the graphics of this project with her great artistic skills.

And al last but not least, Doctor Alex Haro, for his never-ending patience and endless dedication to me despite his incredibly tight agenda, as well as for all the talks about movies, sports and mostly, progressive music.

Introduction

It is beyond argument that dynamical systems have resulted one of the most prolific areas in mathematics, allowing us to predict events that might go from natural phenomena such as meteorological behavior or the development and population growth of a given species, to more artificial and human made situations, such as economic operations or particle movements in physics or engineering experiments. As a result, the research and development of this field have become more and more important, leading to a considerably large range of topics within this area itself.

One type of dynamical system is the skew-product dynamical system, and is the one we are treating in this project. These kinds of system usually appear in models that operate under external perturbations, that is why the proper understanding of them can be the key for solving all sort of problems that may arise in other areas like Physics or Engineering. One aspect that is crucial when studying any kind of dynamical system, is the search for invariant objects, and for our case we will not proceed otherwise. In skew-product systems, the simplest invariant object is called invariant section, and is considered to be the response to external perturbations. These external perturbations can be of all kind, but this time we will deal with the ones where the external forcing is a rigid rotation in a torus. Systems under such forcing are called quasi-periodic dynamical systems, and hence, an invariant tori will carry a quasi-periodic dynamics.
In order to compute invariant objects or merely prove its existence, one shall start off with an initial data which, if the proper conditions are fulfilled, may induce that an invariant object must exist. Such conditions are robustness and hyperbolicity, among others, which, in some sense, guarantee the presence of a true invariant object near an approximate one. Though it is not a trivial task, there are validation theorems that allow the user to prove the existence and local uniqueness of a true solution, given that the initial data, as we have said, satisfies non-degeneracy conditions and its error of invariance is properly bounded. This theorem also provides a rigorous upper bound of the distance of our true solution to the approximate one and a rigorous lower bound of the distance of other possible solutions. The non-degeneracy condition implies the existence of a hyperbolicity bound, and, in the same way as the invariance error bound, it can be explicitly calculated.

In order to perform this task in a practical situation, we will consider the case in which the external perturbation is quasi-periodic. In such case, Fourier methods are specially tailored to

the problem at hand. Then, we will introduce a way to express any periodic function in terms of values that can be calculated by a computer, and this procedure is the Fourier Transform. Fourier Transforms are operations that are able to take values of a function and turn them into the so called Fourier coefficients, that will make the expression of the function in terms of a trigonometric polynomial possible. These resulting series are called Fourier series.
Historically, Fourier series were discovered by Joseph Fourier when trying to solve problems related to the heat equation (a very important subject in differential equations), and nowadays are commonly used in harmonic analysis (in mathematics) and in sound engineering, given that a Fourier Transform is the perfect tool to transform a signal represented in time domain, in frequency domain.
In the current context, we will use the discrete version of the Fourier Transform, the Discrete Fourier Transform for a further implementation into a computer. This version, of course, lacks of information that the continuous version has, and therefore, there is an error committed when using the discrete transform instead of the regular transform. The bound for this error will become the key for computing the invariance and hyperbolicity errors, since it will provide a very suitable and computable value for said bounds.

Understanding now the bigger picture, coming up next we will provide a rigorous and explicit calculation of the error comitted when discretizing the Fourier Transform, which will directly lead us to find a computable expression for both invariance error and the hyperbolicity bound, giving first the preparatory definitions and the proof of a validation theorem, and explaining right afterwards a method to simplify the expression of the linearized dynamics of a system, providing as well an effective algorithm and an explanation of the computer implementation in C language that can be found in the Annex.

Chapter 1

Skew-product Dynamical Systems

In this very first chapter, we are going to present the basic notions of what a skew-product dynamical system is, as well as some other useful properties and definitions that will be used further ahead in the project. But before diving directly in, we will need some general notions about bundles, fiber bundles and other concepts in order to fully understand the particular case that a skew-product system is.

1.1 Introductory Definitions

Definition 1.1. *A bundle is a triple (E, π, B) where E is a set called the total space, B is a set called the base space of the bundle and $\pi : E \to B$ is the projection map.*[1]

Definition 1.2. *Let (E_1, π_1, B) and (E_2, π_2, B) be bundles and $f : B_1 \to B_2$ a map. Then a bundle map $F : E_1 \to E_2$ covering f is a map such that $\pi_2 \circ F = f \circ \pi_1$, that is*

$$\begin{array}{ccc} E_1 & \xrightarrow{F} & E_2 \\ {\scriptstyle \pi_1}\big\downarrow & & \big\downarrow{\scriptstyle \pi_2} \\ B_1 & \xrightarrow{f} & B_2 \end{array}$$

Definition 1.3. *Let (E_1, π_1, B_1) and (E_2, π_2, B_2) be bundles and $F : E_1 \to E_2$ be a bundle map covering $f : B_1 \to B_2$. If $B_1 = B_2$ and $f = id$, then F is a bundle map over $B = B_1 = B_2$ such that $\pi_2 \circ F = \pi_1$. That is, the following diagram should commute*

$$\begin{array}{ccc} E_1 & \xrightarrow{F} & E_2 \\ {\scriptstyle \pi_1}\big\downarrow & \swarrow{\scriptstyle \pi_2} & \\ B & & \end{array}$$

Equivalently, for any point $x \in B$, F maps the fiber $E_{1_x} = \pi_1^{-1}(\{x\})$ of E_1 over x to the fiber $E_{2_x} = \pi_2^{-1}(\{x\})$ of E_2 over x.[2]

Definition 1.4. *Let* (E, π, B) *be a bundle, then a section of that bundle is a continuous map* $\sigma : B \to E$ *such that* $\pi(\sigma(x)) = x$ *for all* $x \in B$. *That is,* $\pi \circ \sigma = id$ *which means that the following diagram commutes* [8]

$$\begin{array}{ccc} E & \xrightarrow{\pi} & B \\ \sigma\uparrow & \nearrow_{id} & \\ B & & \end{array}$$

Definition 1.5. *Let* (E, π, B) *be a bundle, given a bundle map* $F : E \to E$ *covering* $f : B \to B$, *an F-invariant section is a section that satisfies that* $F \circ \sigma = \sigma \circ f$, *which means the following diagram commutes*

$$\begin{array}{ccc} E & \xrightarrow{F} & E \\ \sigma\uparrow & & \uparrow\sigma \\ B & \xrightarrow[f]{} & B \end{array}$$

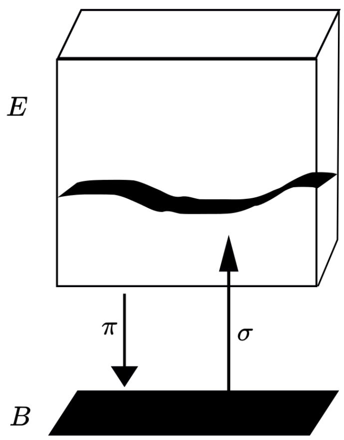

Figure 1.1: *A section* σ *of a bundle* $\pi : E \to B$. *A section* σ *allows the base space* B *to be identified with a subspace* $\sigma(B)$ *of* E.[14]

Definition 1.6. *A fiber bundle is a structure* (E, π, B, P), *where* E, B *and* P *are topological spaces and* $\pi : E \to B$ *is a continuous surjection satisfying a local triviality outlined below. The space* B *is called the base space of the bundle,* E *the total space, and* P *the fiber. The map* π *is called the projection map (or bundle projection). From now on, we will assume the base space* B *is connected.*
We require that for every $x \in E$, *there is an open neighborhood* $U \subset B$ *of* $\pi(x)$ *(which will be called a trivializing neighborhood) such that there is a homeomorphism* $\varphi : \pi^{-1}(U) \to U \times P$ *(where* $U \times P$ *is the product space) in such a way that* π *agrees with the projection onto the first factor. That is, the following diagram should commute*

$$\begin{array}{ccc} \pi^{-1}(U) & \xrightarrow{\varphi} & U \times P \\ \pi\downarrow & \nearrow_{proj_1} & \\ U & & \end{array}$$

where $proj_1 : U \times P \to U$ *is the natural projection and* $\varphi : \pi^{-1}(U) \to U \times P$ *is a homeomorphism. The set of all* $\{(U_i, \varphi_i)\}$ *is called a local trivialization of the bundle.*
Thus for any $y \in B$*, the preimage* $\pi^{-1}(\{y\})$ *is homeomorphic to* P *(since* $proj_1^{-1}(\{y\})$ *clearly is) and is called the fiber over* y*. Every fiber bundle* $\pi : E \to B$ *is an open map, since projections of products are open maps. Therefore* B *carries the quotient topology determined by the map* π*.*

For a better understanding of the fiber bundle concept, one shall see E locally like the product $B \times P$, except that the fibers $\pi(x)^{-1}$ for $x \in B$ may be a bit "twisted".[5]
Notice that a bundle is a generalization of a fiber bundle but with the sets lacking of a topology, which makes the condition of a local product structure drop.[1]

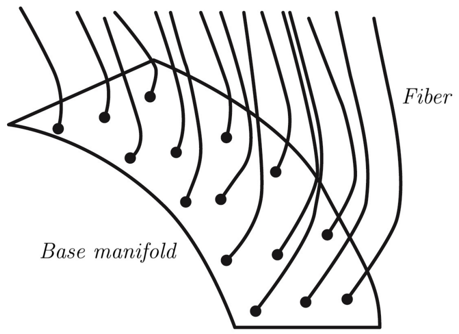

Figure 1.2: *A fiber bundle.*[12]

Remark 1.7. Let $E = B \times P$ and let $\pi : E \to B$ be the projection onto the first factor. Then E is a fiber bundle (of P) over B. Here E is not just locally a product but globally one. Any such fiber bundle is called a trivial bundle.[5]

Definition 1.8. *A real vector bundle consists of a fiber bundle* (E, π, B, P) *with* $P = \mathbb{R}^k$*, where the compatibility condition is satisfied, that is,* $\forall p \in B$*, there is an open neighborhood* $U \subseteq B$*, and a homeomorphism* $\varphi : U \times \mathbb{R}^k \to \pi^{-1}(U)$*, such that* $\forall x \in U$*;*

1. $(\pi \circ \varphi)(x, v) = x \quad \forall v \in \mathbb{R}^k$.
2. *The map* $v \mapsto \varphi(x, v)$ *is a linear isomorphism between the vector spaces* $\mathbb{R}^k$ *and* $\pi^{-1}(\{x\})$*.*

Remark 1.9. The open neighborhood U together with the homeomorphism φ is called a local trivialization of the vector bundle. The local trivialization shows that, locally, the map π looks like the projection of $U \times \mathbb{R}^k$ on U.[13]

We can extend some standard operations between vector spaces such as the direct sum to the context of vector bundles.

Definition 1.10. *A Whitney sum is an operation that takes two vector bundles over a fixed space and produces a new vector bundle over the same space. If* E_1 *and* E_2 *are vector bundles over* B*, then the Whitney sum* $E_1 \oplus E_2$ *is the vector bundle over* B *such that each fiber over* B *is naturally the direct sum of the* E_1 *and* E_2 *fibers over* B*.*
The Whitney sum is therefore the fiber for fiber direct sum of the two bundles E_1 *and* E_2*.*[15]

1.2 Skew-product Dynamical Systems

There are lots of things in mathematics that work perfectly in the context introduced by the previous definitions, but in order to simplify and focus on our main topic, we will consider in the whole project the "trivial" case, which takes $\mathbb{T}^d \times \mathbb{R}^n$ as a trivial vector bundle over $\mathbb{T}^d$ and $\pi : \mathbb{T}^d \times \mathbb{R}^n \to \mathbb{T}^d$ the corresponding bundle projection.
We consider in $\mathbb{T}^d \times \mathbb{R}^n$ the product topology, so that the bundle projection is continuous. Coming up next, we introduce the concept of Finsler norm.

Definition 1.11. *Let $\mathbb{T}^d \times \mathbb{R}^n$ be a trivial fiber bundle with projection $\pi : \mathbb{T}^d \times \mathbb{R}^n \to \mathbb{T}^d$. A Finsler norm in the bundle is a continuous map*

$$\begin{aligned} |\cdot| : \mathbb{T}^d \times \mathbb{R}^n &\longrightarrow \mathbb{R}_+ \\ (\theta, x) &\longrightarrow |(\theta, x)| = |x|_\theta \end{aligned}$$

such that, for each $\theta \in \mathbb{T}^d$, $|\cdot|_\theta : \mathbb{R}^n \to \mathbb{R}_+$ is a norm.

In simpler terms, a Finsler norm in $\mathbb{T}^d \times \mathbb{R}^n$ is a norm $|\cdot|_\theta$ on each fiber $\{\theta\} \times \mathbb{R}^n$ that depends continuously on θ. Examples of Finsler norms are the constant Finsler norm $|\cdot|$, independent of θ, or given a norm $|\cdot|$ on $\mathbb{R}^n$, and a continuous matrix map $P : \mathbb{T}^d \to GL(\mathbb{R}^n)$, the Finsler norm $|x|_\theta = |P(\theta)x|$. We will usually eliminate the dependence on θ of $|\cdot|_\theta$ when it is clear from the context.[10]

Definition 1.12. *An annulus $\mathscr{A}$ is an open set $\mathscr{A} \subset \mathbb{T}^d \times \mathbb{R}^n$ that is homotopic to $\mathbb{T}^d \times U$, where $U \subset \mathbb{R}^n$ is an open set.*

Definition 1.13. (Skew-product Dynamical System) *Let $f : \mathbb{T}^d \to \mathbb{T}^d$ be a homeomorphism. A skew-product dynamical system in $\mathbb{R}^n$ over f is a bundle map*

$$\begin{aligned} (f, F) : \mathscr{A} \subset \mathbb{T}^d \times \mathbb{R}^n &\longrightarrow \mathbb{T}^d \times \mathbb{R}^n \\ (\theta, y) &\longrightarrow (f(\theta), F(\theta, y)) \end{aligned}$$

where $F : \mathscr{A} \to \mathbb{R}^n$ is C^1 with respect to y.

From now on, we will refer to a continuous torus as a continuous section on the bundle $\mathbb{T}^d \times \mathbb{R}^n$, that is, a continuous map of the form $(id, K) : \mathbb{T}^d \to \mathbb{T}^d \times \mathbb{R}^n$, where $K : \mathbb{T}^d \to \mathbb{R}^n$ is continuous. The torus $\mathscr{K} = graph(K) := \{(\theta, K(\theta)) \mid \theta \in \mathbb{T}^d\}$ is said to be graphed by K. Hence, from the triviality of the bundle $\mathbb{T}^d \times \mathbb{R}^n$, we identify the space of continuous sections of the bundle, $\Gamma(\mathbb{T}^d \times \mathbb{R}^n)$, with the space of continuous functions, $C^0(\mathbb{T}^d, \mathbb{R}^n)$. These are endowed with the supremum norm: $\|(id, K)\| = \|K\| = \sup_{\theta \in \mathbb{T}^d} |K(\theta)|_\theta$.
The set of continuous sections with image in the annulus $\mathscr{A}$ is denoted by $\Gamma(\mathscr{A})$, and we note $C^0_{\mathscr{A}}(\mathbb{T}^d, \mathbb{R}^n) = \{K \in C^0(\mathbb{T}^d, \mathbb{R}^n) \mid (id, K) \in \Gamma(\mathscr{A})\}$.
A section $\sigma = (id, K)$ is invariant under the skew-product (f, F) if

$$F(\theta, K(\theta)) = K(f(\theta)) \tag{1.1}$$

for all $\theta \in \mathbb{T}^d$. The Equation (1.1) is the so called invariance equation.

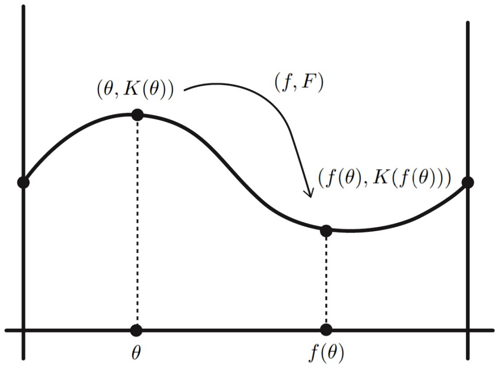

Figure 1.3: *A continuous torus.*

Another way to see this is by looking at K as a fixed point of the graph transform functional $\mathscr{G} : C^0_{\mathscr{A}}(\mathbb{T}^d, \mathbb{R}^n) \to C^0(\mathbb{T}^d, \mathbb{R}^n)$ defined by

$$\mathscr{G}(K)(\theta) = F(f^{-1}(\theta), K(f^{-1}(\theta))),$$

which satisfies that $\mathscr{G}(\sigma) = F \circ \sigma \circ f^{-1}$.

The corresponding graph, $\mathscr{K} = graph(K)$, is invariant under (f, F), which means that $\mathscr{K}$ is an invariant manifold of (f, F), modeled by $\mathbb{T}^d$, for which the internal dynamics is f (as shown in [10]).

It is interesting to introduce some concepts concerning the linear dynamics of our system that will help us to understand the following validation theorem and the forthcoming Chapter 3. The linearized dynamics around the torus $\mathscr{K}$ is given by the vector bundle map

$$\begin{aligned} (f, M) : \mathbb{T}^d \times \mathbb{R}^n &\longrightarrow \mathbb{T}^d \times \mathbb{R}^n \\ (\theta, x) &\longrightarrow (f(\theta), M(\theta)x) \end{aligned}$$

where $M(\theta) = D_y F(\theta, K(\theta))$ is called the transfer matrix.[10]
The linear skew-product (f, M) induces a transfer operator $\mathscr{M} : C^0(\mathbb{T}^d, \mathbb{R}^n) \to C^0(\mathbb{T}^d, \mathbb{R}^n)$, defined as

$$\mathscr{M}(\xi)(\theta) = M(f^{-1}(\theta))\xi(f^{-1}(\theta))$$

for $\xi \in C^0(\mathbb{T}^d, \mathbb{R}^n)$. The operator norm of the transfer operator $\mathscr{M}$ is

$$\|\mathscr{M}\| = \sup\{\|\mathscr{M}\xi\|_\infty \; : \; \xi \in C^0(\mathbb{T}^d, \mathbb{R}^n), \|\xi\|_\infty = 1\},$$

where $\|\xi\|_\infty = \max_{\theta \in \mathbb{T}^d} |\xi(\theta)|_\theta$. Remarkably, the transfer operator's norm coincides with the norm of the matrix M, which is

$$\|M\| = \sup_{\theta \in \mathbb{T}^d} \|M(\theta)\| = \sup_{\theta \in \mathbb{T}^d} \sup_{|v|_\theta \leq 1} |M(\theta)v|_{f(\theta)}.$$

Following up, we are giving a general definition of what a (functionally) fiberwise hyperbolic torus is in order to understand one of the validation theorem's hypotheses. A deeper approach into fiberwise hyperbolicity will be carried out in its corresponding Section 5.2. This definition extends naturally from the definition of hyperbolicity of a fixed point of an autonomous discrete dynamical system to hyperbolicity of an invariant section of a non-autonomous discrete dynamical system.

Definition 1.14. *An invariant torus $\mathscr{K}$ of the skew-product (f, F) in (1.13), graphed by $K \in C^0_{\mathscr{A}}(\mathbb{T}^d, \mathbb{R}^n)$, is said to be (functionally) fiberwise hyperbolic if the corresponding transfer operator $\mathscr{M}$ is hyperbolic. That is, if the spectrum of $\mathscr{M}$ has empty intersection with the unit circle $\{z \in \mathbb{C} \,:\, |z| = 1\}$.*

In other words, hyperbolicity means that for all z such that $|z| = 1$, and for all $w \in C^0_{\mathscr{A}}(\mathbb{T}^d, \mathbb{R}^n)$ there exists a unique $v \in C^0_{\mathscr{A}}(\mathbb{T}^d, \mathbb{R}^n)$ such that

$$M(f^{-1}(\theta))v(f^{-1}(\theta)) - zv(\theta) = w(\theta),$$

for all $\theta \in \mathbb{T}^d$. That is, $v = (\mathscr{M} - zId)^{-1}w$. In virtue of the Open Mapping Theorem (see below), there exists a positive constant c_H, the so called hyperbolicity bound, such that $\|(\mathscr{M} - zId)^{-1}\| \leq c_H$. That is, $\|v\| \leq c_H\|w\|$.[10]

Theorem 1.15. (Banach Open Mapping Theorem) *If X and Y are Banach spaces and $\mathscr{T} : X \to Y$ is a surjective continuous linear operator, then $\mathscr{T}$ is an open map. If moreover, $\mathscr{T} : X \to Y$ is bijective, then $\mathscr{T}^{-1} : Y \to X$.*[7]

Chapter 2

The Kantorovich-type Validation Theorem

As the main result in this project, in this chapter we are going to deal with the validation theorem that proves the existence and uniqueness of a fiberwise hyperbolic invariant torus provided that there exists an approximately invariant one. Said theorem also provides lower and upper bounds for how close the effectively invariant torus is to the approximate one and for the radius of an annalus wherein there is only one fiberwise hyperbolic invariant torus.

2.1 Fixed Point Theorems

Theorem 2.1. (Banach Fixed Point Theorem) *Let (X, d) be a complete metric space and $f : X \to X$ a contractive map with contraction factor $K \in [0, 1)$, then exists a unique $x_* \in X$ such that $f(x_*) = x_*$.*

Proof. Start by taking a $x_0 \in X$, and then defining the sequence $(x_n)_n$ as $x_n = f^n(x_0)$. Since our metric space is complete, it suffices to prove that our sequence is a Cauchy one. $\forall n$ and $\forall p \geq 0$

$$\begin{aligned} d(x_{n+p}, x_n) &\leq d(x_{n+p}, x_{n+p-1}) + \ldots + d(x_{n+1}, x_n) \\ &\leq (K^{n+p-1} + K^{n+p-2} + \ldots + K^n) d(x_1, x_0) \\ &\leq K^n(1 + K + \ldots + K^{p-1}) d(x_1, x_0) \leq \frac{K^n}{1-K} d(x_1, x_0). \end{aligned}$$

Where, in the third step, we have applied that

$$d(x_{m+1}, x_m) \leq K d(x_m, x_{m-1}) \leq \ldots \leq K^m d(x_1, x_0)$$

using the contractive property and a geometric sum in the last step. From the inequality we obtain $\lim\limits_{n\to\infty} \sup\limits_{p\geq 0} d(x_{n+p}, x_n) = 0$, since $\sup\limits_{p\geq 0} d(x_{n+p}, x_n) \leq \frac{K^n}{1-K} d(x_1, x_0)$, hence it is a Cauchy sequence and therefore $(x_n)_n$ converges to a certain x_*. Thus $x_{n+1} = f(x_n) \xrightarrow[n\to\infty]{} x_* = f(x_*)$

and x_* is a fixed point of f.

The uniqueness is easily proved by assuming there are two different fixed points, x_*, y_*, and therefore

$$0 < d(x_*, y_*) = d(f(x_*), f(y_*)) \leq Kd(x_*, y_*) \to d(x_*, y_*) \leq Kd(x_*, y_*)$$

which is a contradiction since $K \in [0, 1)$. □

Theorem 2.2. (Rigorous Fixed Point Theorem) *Let X be a Banach space and and let $x_0 \in X$. Let now $\mathscr{T} : B_R(x_0) \subset X \to X$ be a map in the open set $B_R(x_0)$ such that $\forall\, r \in (0, R)$, $\mathscr{T}_{\big|\overline{B}_r(x_0)}$ is Lipschitz, where $\overline{B}_r(x_0) = \{x \in X : \|x - x_0\| \leq r\}$ and*

$$\begin{aligned} L : (0, R) &\longrightarrow \mathbb{R}_+ \\ r &\longmapsto L(r) = \sup_{\substack{x_1, x_2 \in \overline{B}_r(x_0) \\ x_1 \neq x_2}} \frac{\|\mathscr{T}(x_2) - \mathscr{T}(x_1)\|}{\|x_2 - x_1\|}. \end{aligned}$$

Notice that L is an increasing function.
Assume that $\|\mathscr{T}(x_0) - x_0\| \leq \varepsilon$, where $\varepsilon > 0$ is the error bound of the fixed point condition, and pick $r \in (\varepsilon, R)$. Then if $\frac{\varepsilon}{r} + L(r) - 1 \leq 0$, there exists a unique $x_* \in \overline{B}_r(x_0)$ such that $\mathscr{T}(x_*) = x_*$.

Proof. Since X is a Banach space, and therefore a complete space, Theorem 2.1 allows us to reduce the proof to the following two steps:

1. $\mathscr{T}(\overline{B}_r(x_0)) \subseteq \overline{B}_r(x_0)$, so the image of the ball won't escape the ball itself and $\overline{B}_r(x_0)$ will become a complete subspace.

2. $\mathscr{T}_{\big|\overline{B}_r(x_0)}$ is contractive.

For the first step we pick $x \in \overline{B}_r(x_0)$ and we see

$$\begin{aligned} \|\mathscr{T}(x) - x_0\| &\leq \|\mathscr{T}(x) - \mathscr{T}(x_0)\| + \|\mathscr{T}(x_0) - x_0\| \leq L(r)\|x - x_0\| + \varepsilon \\ &\leq L(r)r + \varepsilon = r\left(L(r) + \frac{\varepsilon}{r}\right) \leq r \end{aligned}$$

which means that $\mathscr{T}(x)$ is in $\overline{B}_r(x_0)$.

Since our function $\mathscr{T}$ is already Lipschitz, we only need to see if the Lipschitz constant $L(r)$ dwells in the $(0, 1)$ interval. By hypothesis, $\frac{\varepsilon}{r} + L(r) - 1 \leq 0$ which leads to $L(r) \leq 1 - \frac{\varepsilon}{r} < 1$. □

2.2 Validation Theorem

The last two results will be very useful to help us prove the validation theorem, which claims as follows.

Theorem 2.3. *Let $(f, F) : \mathscr{A} \subset \mathbb{T}^d \times \mathbb{R}^n \to \mathbb{T}^d \times \mathbb{R}^n$ be a skew-product on the annulus $\mathscr{A}$, with F being of class C^{1+Lip} with respect to the fiber variable y. This means that $F(\theta, y)$ is of class C^1 with respect to y and $DF(\theta, y)$ is Lipschitz with respect to y. Assume we are given:*

1.1) an approximately invariant torus $\mathscr{K}_0 = graph(K_0)$ with $(id, K_0) \in \Gamma(\mathscr{A})$;

1.2) a Finsler norm $|\cdot| : \mathbb{T}^d \times \mathbb{R}^n \longrightarrow \mathbb{R}_+$;

1.3) a closed annulus around $\mathscr{K}_0$ of radius R inside $\mathscr{A}$:

$$\bar{\mathscr{A}}(K_0, R) := \{(\theta, y) \in \mathbb{T}^d \times \mathbb{R}^n \mid \forall \theta \in \mathbb{T}^d, |K(\theta) - y|_\theta \leq R\} \subset \mathscr{A}.$$

Let $\hat{\varepsilon}$ be an error bound of the invariance equation for K_0, c_H be a hyperbolicity bound of the transfer operator $\mathscr{M}_0$ associated to the linear skew-product (f, M_0) given by the transfer matrix $M_0(\theta) = D_y F(\theta, K_0(\theta))$, and b be the Lipschitz constant of the differential of the skew-product with respect to y in $\bar{\mathscr{A}}(K_0, R)$. That is,

2.1) for each $\theta \in \mathbb{T}^d$, $|F(\theta, K_0(\theta)) - K_0(f(\theta))|_{f(\theta)} \leq \hat{\varepsilon}$;

2.2) for each $z \in \mathbb{C}$ with $|z| = 1$, $\|(\mathscr{M}_0 - zId)^{-1}\| \leq c_H$;

2.3) for each $(\theta, y_1), (\theta, y_2) \in \bar{\mathscr{A}}(K_0, R)$, $x \in \mathbb{R}^n$,

$$|(D_y F(\theta, y_1) - D_y F(\theta, y_2))x|_{f(\theta)} \leq b\, |y_1 - y_2|_\theta\, |x|_\theta.$$

Assume that

3.1) $c_H^2 b\, \hat{\varepsilon} \leq h < \frac{1}{4}$;

3.2) $(1 - \sqrt{1 - 4h})(2c_H b)^{-1} \leq r_0 \leq r_1 < \min((1 + \sqrt{1 - 4h})(2c_H b)^{-1}, R)$.

Then there exists a unique torus $\mathscr{K}_* = graph(K_*)$ with $(id, K_*) \in \Gamma(\mathscr{A})$ such that:

a.1) for each $\theta \in \mathbb{T}^d$, $F(\theta, K_*(\theta)) - K_*(f(\theta)) = 0$, that is, $\mathscr{K}_*$ is invariant;

a.2) for each $\theta \in \mathbb{T}^d$, $|K_*(\theta) - K_0(\theta)|_\theta \leq r_1$, which means it is locally unique.

Moreover:

a.3) for each $\theta \in \mathbb{T}^d$, $|K_*(\theta) - K_0(\theta)|_\theta \leq r_0$, which means $\mathscr{K}_*$ is close to $\mathscr{K}_0$;

a.4) $\mathscr{K}_*$ is a fiberwise hyperbolic invariant torus.

Remark 2.4. Keep in mind that the $\hat{\varepsilon}$ from the theorem is different than the ε from Theorem 2.2.

Remark 2.5. In hypothesis (3.1), we could take $c_H^2 b\,\varepsilon = h$ and the result would hold true as well, but we take the inequality given that works better when bounding in validations.

Remark 2.6. In $\bar{\mathscr{A}}(K_0, r_1)$ there is a unique invariant graph which, in fact, is contained in $\bar{\mathscr{A}}(K_0, r_0)$.

Proof. We look for a solution $(id, K) \in \Gamma(\bar{\mathscr{A}}(K_0, R))$ of the invariance equation

$$F(\theta, K(\theta)) - K(f(\theta)) = 0 \tag{2.1}$$

of the form

$$K(\theta) = K_0(\theta) + \xi(\theta). \tag{2.2}$$

Wrapping up a bit, what we know so far is that we are looking for the exact solution $K(\theta)$ of the Equation (2.1), knowing $K_0(\theta)$, our approximate solution. Expressing $K(\theta)$ such as in (2.2), permits us to turn the current problem into proving the existence of a correction function $\xi(\theta)$ for the $K_0(\theta)$ function. The proof of existence of such object can be performed taking profit of the previous Theorem 2.2, noticing that our current objects satisfy the theorem's hypotheses.
Let us define the closed set for $r \leq R$

$$X_r = \{\xi \in C^0(\mathbb{T}^d, \mathbb{R}^n) \mid \|\xi\| \leq r\}.$$

Notice X_r is shaped as a closed ball $\overline{B}_r(0)$ in $C^0(\mathbb{T}^d, \mathbb{R}^n)$.
Substituting the expression (2.2) into the invariance equation (2.1), we obtain

$$F(\theta, K_0(\theta) + \xi(\theta)) - K_0(f(\theta)) - \xi(f(\theta)) = 0. \tag{2.3}$$

Using Taylor's Theorem, we have

$$\begin{aligned} F(\theta, K_0(\theta) + \xi(\theta)) &= F(\theta, K_0(\theta)) + D_y F(\theta, K_0(\theta))\,\xi(\theta) \\ &\quad + \int_0^1 (D_y F(\theta, K_0(\theta) + s\xi(\theta)) - D_y F(\theta, K_0(\theta)))\,\xi(\theta) ds. \end{aligned}$$

Then equation (2.3) is equivalent to

$$M_0(\theta)\,\xi(\theta) - \xi(f(\theta)) = \mathscr{N}(\xi)(f(\theta))$$

where $M_0(\theta) = D_y F(\theta, K_0(\theta))$, and

$$\begin{aligned} \mathscr{N}(\xi)(f(\theta)) &= -\left[F(\theta, K_0(\theta)) - K_0(f(\theta))\right] \\ &\quad - \int_0^1 (D_y F(\theta, K_0(\theta) + s\xi(\theta)) - D_y F(\theta, K_0(\theta)))\,\xi(\theta) ds. \end{aligned}$$

We now push forward, obtaining

$$M_0(f^{-1}(\theta))\,\xi(f^{-1}(\theta)) - \xi(\theta) = \mathscr{N}(\xi)(\theta).$$

Furthermore, we can express the equation in terms of the transfer operator

$$\mathscr{M}_0(\xi)(\theta) = M_0(f^{-1}(\theta))\xi(f^{-1}(\theta)).$$

Now, by the hypothesis of fiberwise hyperbolicity, $\mathscr{M}_0(\xi)(\theta)$ is an invertible operator, so we can rewrite the invariance equation as a fixed point equation for ξ:

$$\xi = (\mathscr{M}_0 - Id)^{-1} \circ \mathscr{N}(\xi).$$

Now that we have turned our problem into an explicit fixed point finding problem, we shall follow the notation of the previous theorem and consider the operator

$$\mathscr{T} = (\mathscr{M}_0 - Id)^{-1} \circ \mathscr{N} : X_R = B_R(0) \to C^0(\mathbb{T}^d, \mathbb{R}^n).$$

For $\xi_0 = 0$, we evaluate $\|\mathscr{T}(\xi_0) - \xi_0\| = \|(\mathscr{M}_0 - Id)^{-1} \circ \mathscr{N}(0)\| \leq c_H \hat{\varepsilon} = \varepsilon$, where ε is the actual ε from Theorem 2.2 and $\mathscr{N}(0) = -\left[F(\theta, K_0(\theta)) - K_0(f(\theta))\right]$.
Recalling that R is the radius of the closed annulus $\bar{\mathscr{A}}$ wherein the approximate torus $\mathscr{K}_0$ is contained, we shall now define the following increasing function

$$\begin{aligned} L : (0, R) &\longrightarrow \mathbb{R}_+ \\ r &\longmapsto L(r) = \sup_{\substack{\xi_1, \xi_2 \in X_r \\ \xi_1 \neq \xi_2}} \frac{\|\mathscr{T}(\xi_2) - \mathscr{T}(\xi_1)\|}{\|\xi_2 - \xi_1\|}. \end{aligned}$$

Let's then calculate the Lipschitz constant for the operator $\mathscr{T}$

$$\begin{aligned} &\|(\mathscr{M}_0 - Id)^{-1} \circ \mathscr{N}(\xi_2) - (\mathscr{M}_0 - Id)^{-1} \circ \mathscr{N}(\xi_1)\| \leq \\ &\|(\mathscr{M}_0 - Id)^{-1} \circ (\mathscr{N}(\xi_2) - \mathscr{N}(\xi_1))\| \leq \\ &\|(\mathscr{M}_0 - Id)^{-1}\| \cdot \|\mathscr{N}(\xi_2) - \mathscr{N}(\xi_1)\|. \end{aligned}$$

Since we know by hypothesis that $\|(\mathscr{M}_0 - Id)^{-1}\| \leq c_H$, we calculate the other term of the expression separately. In the following calculation we will be using the Finsler norm evaluated over $f(\theta)$, that is $|\cdot|_{f(\theta)}$, but for the sake of the reader and the writer's convenience, we shall omit the $f(\theta)$ subindex.

$$
\begin{aligned}
&|\mathscr{N}(\xi_2)(f(\theta)) - \mathscr{N}(\xi_1)(f(\theta))| \\
&= \Big| \int_0^1 \Big(D_yF(\theta, K_0(\theta) + s\xi_2(\theta)) - D_yF(\theta, K_0(\theta)) \Big)\, \xi_2(\theta) \\
&- \Big(D_yF(\theta, K_0(\theta) + s\xi_1(\theta)) - D_yF(\theta, K_0(\theta)) \Big)\, \xi_1(\theta)\, ds \Big| \\
&= \Big| \int_0^1 \Big(D_yF(\theta, K_0(\theta) + s\xi_2(\theta)) - D_yF(\theta, K_0(\theta)) \Big)\, \xi_2(\theta) \\
&+ \Big(D_yF(\theta, K_0(\theta) + s\xi_2(\theta)) - D_yF(\theta, K_0(\theta)) \Big)\, \xi_1(\theta) \\
&- \Big(D_yF(\theta, K_0(\theta) + s\xi_2(\theta)) - D_yF(\theta, K_0(\theta)) \Big)\, \xi_1(\theta) \\
&- \Big(D_yF(\theta, K_0(\theta) + s\xi_1(\theta)) - D_yF(\theta, K_0(\theta)) \Big)\, \xi_1(\theta)\, ds \Big| \\
&\leq \Big| \int_0^1 \Big(D_yF(\theta, K_0(\theta) + s\xi_2(\theta)) - D_yF(\theta, K_0(\theta)) \Big)(\xi_2(\theta) - \xi_1(\theta))\, ds \Big| \\
&+ \Big| \int_0^1 \Big(D_yF(\theta, K_0(\theta) + s\xi_2(\theta)) - D_yF(\theta, K_0(\theta) + s\xi_1(\theta)) \Big)\, \xi_1(\theta)\, ds \Big| \\
&\leq \int_0^1 |D_yF(\theta, K_0(\theta) + s\xi_2(\theta)) - D_yF(\theta, K_0(\theta))|\, ds\, |\xi_2(\theta) - \xi_1(\theta)| \\
&+ \int_0^1 |D_yF(\theta, K_0(\theta) + s\xi_2(\theta)) - D_yF(\theta, K_0(\theta) + s\xi_1(\theta))|\, ds\, |\xi_1(\theta)| \\
&\leq \int_0^1 bs\, |\xi_2(\theta)|\, ds\, |\xi_2(\theta) - \xi_1(\theta)| + \int_0^1 bs\, |\xi_2(\theta) - \xi_1(\theta)|\, ds\, |\xi_1(\theta)| \\
&\leq \frac{1}{2} br\, |\xi_2(\theta) - \xi_1(\theta)| + \frac{1}{2} b\, |\xi_2(\theta) - \xi_1(\theta)|\, r \leq br\, |\xi_2(\theta) - \xi_1(\theta)|.
\end{aligned}
$$

For some of the final steps we have used the fact that $\xi_1, \xi_2 \in X_r$ and therefore $|\xi_i(\theta)| \leq r$ for $i = 1, 2$, and the hypothesis (2.3) of the theorem. Joining results we finally obtain

$$
\|(\mathscr{M}_0 - Id)^{-1} \circ \mathscr{N}(\xi_2) - (\mathscr{M}_0 - Id)^{-1} \circ \mathscr{N}(\xi_1)\| = \|T(\xi_2) - T(\xi_1)\| \leq c_H br\, \|\xi_2 - \xi_1\|.
$$

And so, the Lipschitz factor of the operator is $L(r) = c_H br$. Our goal is to prove there exists at least one r that satisfies $\frac{\varepsilon}{r} + c_H br - 1 \leq 0$ in order to apply Theorem 2.2.
The r's that satisfy $\frac{\varepsilon}{r} + c_H br - 1 = 0$ are the x-axis cutpoints of the r function described by the equation, which are

$$
r_+ = \frac{1 + \sqrt{1 - 4c_H^2 b\, \hat{\varepsilon}}}{2c_H b} \quad , \quad r_- = \frac{1 - \sqrt{1 - 4c_H^2 b\, \hat{\varepsilon}}}{2c_H b}.
$$

Picking an arbitrary $r_* \in (r_-, r_+)$ $(r_* \leq R)$ such as $r_* = \frac{1}{2c_H b}$ is easy to see that indeed $\frac{\varepsilon}{r_*} + c_H br_* - 1 \leq 0$ and hence, the $\mathscr{T}$ operator is well defined in X_{r_*} and, by Theorem 2.2, there exists a unique fixed point of $\mathscr{T}$, and that is $\xi(\theta)$. Notice that $\mathscr{T}$ is also well defined in X_{r_*} for $r_* \in [r_0, r_1]$ under the hypotheses (3.1) and (3.2).

What we have left to prove is that $\mathscr{K}_*$ is fiberwise hyperbolic. Let $\mathscr{M}_*$ be the transfer operator associated to the linear skew-product (f, M_*) given by the transfer matrix $M_*(\theta) = D_yF(\theta, K_*(\theta))$. Notice that $\mathscr{M}_0$ is a hyperbolic operator, and that $\mathscr{M}_*$ is close to $\mathscr{M}_0$. Then, for any $z \in \mathbb{C}$ with $|z| = 1$,

$$\mathscr{M}_* - zId = (\mathscr{M}_0 - zId)(Id + (\mathscr{M}_0 - zId)^{-1}(\mathscr{M}_* - \mathscr{M}_0)) \tag{2.4}$$

Since

$$\|(\mathscr{M}_0 - zId)^{-1}(\mathscr{M}_* - \mathscr{M}_0)\| \leq c_H b \|\xi_* - 0\| \leq c_H b\, r_- < 1,$$

given that $r_- < 1$. And following standard Neumann series arguments, the right-hand side of (2.4) is invertible, and so then $\mathscr{M}_* - zId$ is invertible, which proves the theorem (adapted from [10]).

□

Even though this is a very useful theoretical result, one might consider the idea of implementing this on a numerical level. If this is the case, other questions may arise, for instance, which is the best way to express computer-wise our skew-product system or its linearized dynamics, how can we calculate a continuous torus, or most importantly, how do we calculate or find explicit and computable expressions for the conditions of the previous theorem, such as the invariance equation error bound or the hyperbolicity bound.

A great approach for tackling this matter is by using the language of Fourier series, a language that a computer understands and is comfortable working with. Nonetheless, one must proceed with caution, because taking the regular Fourier series of a function will not answer our questions since it will still be a continuous object that a computer cannot deliberately use without committing any truncation error. That is why, our main interest will be focused on the ways of expressing our objects via finite Fourier series with Fourier coefficients calculated using discrete methods. The process we will explain is called the Discrete Fourier Transform. This process turns function evaluated points over a grid into Fourier coefficients, and, unlike the regular continuous Fourier Transform, it does it through a sum of elements instead of an integral.

In Chapter 4, we are going to give the main definitions and results regarding the Fourier Transform, the Discrete Fourier Transform and the error produced when applying a Discrete Fourier Transform in comparison with a regular Fourier Transform. We will also give a very efficient algorithm to calculate a Discrete Fourier Transform. This process is called the Fast Fourier Transform and provides a computation time of the order of $N \log N$ in contrast with the N^2 of a direct Discrete Fourier Transform.

But before that, we are going to dive in the study of the linearized dynamics before using Fourier series to calculate the hyperbolicity bound of the theorem.

Chapter 3

On the Notion of Fiberwise Hyperbolicity

In Section 1.2 we introduced the concept of fiberwise hyperbolicity of an invariant torus in functional terms. We will introduce here the dynamical sense of the concept.

3.1 Dynamical Definition of Fiberwise Hyperbolicity

In this section we will work with the linearized dynamics around an invariant torus $\mathscr{K}$ of a skew-product dynamical system, which is given by a vector bundle map

$$\begin{aligned}(f, M) : \mathbb{T}^d \times \mathbb{R}^n &\longrightarrow \mathbb{T}^d \times \mathbb{R}^n \\ (\theta, x) &\longrightarrow (f(\theta), M(\theta)x).\end{aligned}$$

The condition of (uniform) hyperbolicity of the vector bundle map (f, M) assumes that there exists a continuous decomposition of $\mathbb{T}^d \times \mathbb{R}^n$ in Whitney sum of two vector bundles, a stable one, $\mathscr{B}^S$, and an unstable one, $\mathscr{B}^U$. Given $v \in \mathscr{B}^S_\theta$ and since F moves a point $(\theta, K(\theta))$ in the invariant torus to another one $F(\theta, K(\theta))$ in the same torus $\mathscr{K}$ (condition of invariance), the differential matrix of F, $DF(\theta, K(\theta))$ will send $v \in \mathscr{B}^S_\theta$ to a $\bar{v} \in \mathscr{B}^S_{f(\theta)}$, that is $M(\theta)v^S_\theta \in \mathscr{B}^S_{f(\theta)}$. Iterating the process for a $v^S_\theta \in \mathscr{B}^S_\theta$ we obtain $M(\theta)v^S_\theta \in \mathscr{B}^S_{f(\theta)}$, $M(f(\theta))M(\theta)v^S_\theta \in \mathscr{B}^S_{f^2(\theta)}$ and for any $l \geq 0$ $M(f^{l-1}(\theta)) \cdots M(\theta)v^S_\theta \in \mathscr{B}^S_{f^l(\theta)}$. Since we are working with the stable bundle, the differential operation we do as $M(\theta)v^S_\theta$ is a contractive operation because of the eigenvalues of M being in the $(0, 1)$ interval. At this point we finally can introduce the cocycle notation defined as $M(\theta, l) = M(f^{l-1}(\theta)) \cdots M(\theta)$.
The procedure for the unstable bundle is slightly different given that applying the differential matrix is an expansive operation (the eigenvalue is greater than 1). For a clearer view of the argument we shall express it in terms of matrices

$$\begin{pmatrix} v \\ \theta \end{pmatrix} \to \begin{pmatrix} M(\theta)v \\ f(\theta) \end{pmatrix} = \begin{pmatrix} \bar{v} \\ \bar{\theta} \end{pmatrix}$$

and by isolating v we obtain

$$\begin{pmatrix} v \\ \theta \end{pmatrix} = \begin{pmatrix} M(f^{-1}(\theta))^{-1}\,\bar{v} \\ f^{-1}(\bar{\theta}) \end{pmatrix}.$$

Hence, we shall dub $M(\theta,-l) = M(f^{-l}(\theta))^{-1}\cdots M(f^{-1}(\theta))^{-1}$.
In the $l=0$ case, $M(\theta,l) = Id$. [10]

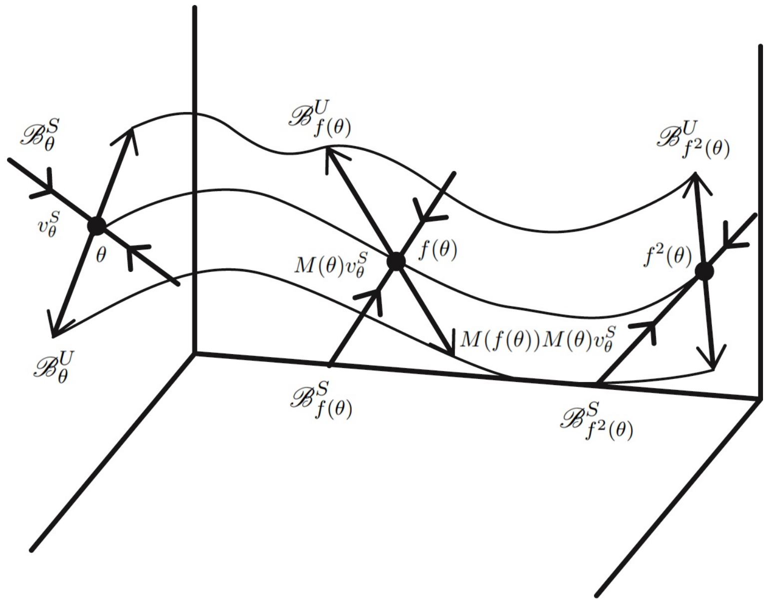

Figure 3.1: *Schematic representation of the stable and unstable bundles and the action of the differential matrix over vectors from the bundles.*

Wrapping up this section, we introduce the formal definition of a (dynamically) fiberwise hyperbolic invariant torus.

Definition 3.1. *An invariant torus $\mathscr{K}$ of a skew-product dynamical system (f,F), graphed by $K \in C^0_{\mathscr{A}}(\mathbb{T}^d,\mathbb{R}^n)$, is said to be (dynamically) fiberwise hyperbolic if the corresponding linear skew-product (f,M) is uniformly hyperbolic. Specifically, if there exist a continuous decomposition $\mathbb{T}^d\times\mathbb{R}^n = \mathscr{B}^S\oplus\mathscr{B}^U$ in Whitney sum of two vector bundles, and constants $C>0$ and $0<\lambda_S<1<\lambda_U$ such that:*

1. *$v\in\mathscr{B}^S_\theta$ if and only if $|M(\theta,l)\,v_\theta| \le C\lambda_S^l|v_\theta|\,,\ \forall l\ge 0.$*
2. *$v\in\mathscr{B}^U_\theta$ if and only if $|M(\theta,-l)\,v_\theta| \le C\lambda_U^{-l}|v_\theta|\,,\ \forall l\ge 0.$*

3.1.1 Relation Between Functional and Dynamical Definition of Fiberwise Hyperbolicity

So far we have introduced two concepts of fiberwise hyperbolicity, a functional one and a dynamical one. The relation between the two concepts has been object of study by many authors since J. Mather. It turns out that both concepts end up being equivalents, although we will focus in just one implication.

Theorem 3.2. *If an invariant torus $\mathscr{K}$ of a skew-product dynamical system (f, F) is dynamically fiberwise hyperbolic, then it is functionally fiberwise hyperbolic.*

Proof. We have to prove that for any $z \in \mathbb{C}$ such that $|z| = 1$, the operator $\mathscr{M} - zId$ is bijective, that is, for any $w \in C^0(\mathbb{T}^d, \mathbb{R}^n)$ there exists a unique $v \in C^0(\mathbb{T}^d, \mathbb{R}^n)$ such that

$$M(f^{-1}(\theta))v(f^{-1}(\theta)) - zv(\theta) = w(\theta).$$

Let's then prove the existence of such v. First we write the previous cohomological equation in terms of the stable and unstable components via projection to their respective bundles.

$$\begin{cases} M(f^{-1}(\theta))\, v^S(f^{-1}(\theta)) - zv^S(\theta) = w^S(\theta) \\ M(f^{-1}(\theta))\, v^U(f^{-1}(\theta)) - zv^U(\theta) = w^U(\theta). \end{cases}$$

Let's pick first the stable one. One shall write the equation as a fixed point type equation as follows

$$v^S(\theta) = \frac{1}{z}\Big(- w^S(\theta) + M(f^{-1}(\theta))\, v^S(f^{-1}(\theta))\Big).$$

And substituting iteratively we obtain

$$\begin{aligned} v^S(\theta) &= \frac{1}{z}\Big(- w^S(\theta) + M(f^{-1}(\theta))\, \frac{1}{z}\left[-w^S(f^{-1}(\theta)) + M(f^{-2}(\theta))\, v^S(f^{-2}(\theta))\right] \Big) \\ &= -\frac{1}{z} w^S(\theta) - \frac{1}{z^2}(f^{-1}(\theta))\, w^S(f^{-1}(\theta)) - \frac{1}{z^2}(f^{-1}(\theta)) M(f^{-2}(\theta))\, v^S(f^{-2}(\theta)) \\ &= -\sum_{k=0}^{\infty} \Big[M(f^{-1}(\theta)) \cdots M(f^{-k+1}(\theta))\, M(f^{-k}(\theta))\Big]\, \frac{1}{z^{k+1}} w^S(f^{-k}(\theta)) \\ &= -\sum_{k=0}^{\infty} \frac{1}{z^{k+1}} M(f^{-k}(\theta), k)\, w^S(f^{-k}(\theta)). \end{aligned}$$

We know $w^S(f^{-k}(\theta)) \in \mathscr{B}^S_{f^{-k}(\theta)}$, hence

$$\begin{aligned} |v^S(\theta)| &\leq \sum_{k=0}^{\infty} \frac{1}{|z|^{k+1}} |M(f^{-k}(\theta), k)\, w^S(f^{-k}(\theta))| \leq C \sum_{k=0}^{\infty} \lambda_S^k\, |w^S(f^{-k}(\theta))| \\ &\leq C \sum_{k=0}^{\infty} \lambda_S^k\, \|w^S\|_\infty = \frac{C}{1-\lambda_S} \|w^S\|_\infty < \infty \end{aligned}$$

since $\lambda_S \in (0, 1)$, and by the Weierstrass M-test, the original series also converges and therefore exists the solution $v^S(\theta)$ for the equation.

Let's prove it now for the unstable one. This time we will isolate $v^U(f^{-1}(\theta))$ from the unstable projection of the cohomological equation since it models the backwards dynamics. Thus, from

$$M(f^{-1}(\theta))\, v^U(f^{-1}(\theta)) - zv^U(\theta) = w^U(\theta)$$

will be more useful to compose with $f(\theta)$ in order to operate with $v^U(\theta)$. This way, isolating, we finally obtain the fixed point type equation

$$v^U(\theta) = M(\theta)^{-1}\, w^U(f(\theta)) + zM(\theta)^{-1}\, v^U(f(\theta)).$$

Now we shall proceed as we did with the stable bundle, iterating.

$$\begin{aligned} v^U(\theta) &= M(\theta)^{-1}\, w^U(f(\theta)) + zM(\theta)^{-1}[M(f(\theta))^{-1}\, w^U(f^2(\theta)) + zM(f(\theta))^{-1}\, v^U(f^2(\theta))] \\ &= M(\theta)^{-1} w^U(f(\theta)) + zM(\theta)^{-1} M(f(\theta))^{-1} w^U(f^2(\theta)) + zM(\theta)^{-1} M(f(\theta))^{-1} v^U(f^2(\theta)) \\ &= \sum_{k=0}^{\infty} \left[M(\theta)^{-1}\, M(f(\theta))^{-1} \cdots M(f^k(\theta))^{-1}\right] z^k w^U(f^{k+1}(\theta)) = \\ &= \sum_{k=0}^{\infty} z^k M(f^k(\theta), -(k+1))\, w^U(f^{k+1}(\theta)). \end{aligned}$$

We know $w^U(f^{k+1}(\theta)) \in \mathscr{B}^U_{f^{k+1}(\theta)}$, hence

$$\begin{aligned} |v^U(\theta)| &\le \sum_{k=0}^{\infty} |z|^k |M(f^k(\theta), -(k+1))\, w^U(f^{k+1}(\theta))| \le C \sum_{k=0}^{\infty} \lambda_U^{-(k+1)}\, |w^U(f^{k+1}(\theta))| \\ &\le C \sum_{k=0}^{\infty} \lambda_U^{-(k+1)}\, \|w^U\|_\infty = \frac{C}{\lambda_U - 1} \|w^U\|_\infty < \infty \end{aligned}$$

since $\lambda_U > 1$ and then $\lambda_U^{-(k+1)} < 1$ for any $k > 0$. Using the Weierstrass M-test as before, we prove the convergence of the original series as well as the existence of such a solution $v^U(\theta)$. Having at last the two of them, $v^S(\theta)$ and $v^U(\theta)$, our solution for the whole cohomological equation is $v(\theta) = v^S(\theta) + v^U(\theta)$. Thus, the operator is hyperbolic. □

Remark 3.3. If $\mathscr{M}$ is a hyperbolic transfer operator with spectrum inside the unit circle, then we say that the torus K is an attractor. If the spectrum of $\mathscr{M}$ is outside the unit circle, then the torus K is a repeller. Otherwise we say that the torus K is a saddle.

3.2 Hyperbolicity Bound

One of the hypothesis required in Theorem 2.3 is the hyperbolicity property of the linear skew-product (f, M_0) associated to the approximate invariant torus (id, K_0). As it is normal, we have to verify such property and even more, calculate the hyperbolicity bound used in the aforementioned theorem. To do so, we approximate the stable and unstable bundles and construct adapted frames in which it is easier to measure the hyperbolicity property. The simplest case is the one we are treating in this section, which is when the invariant bundles are trivial (or easily trivializable), and we can construct global frames.

Assume we are capable of defining a matrix-valued map $P : \mathbb{T}^d \to GL(\mathbb{R}^n)$ (the adapted frame), whose first n_S columns parametrize an approximation of the stable bundle $\mathscr{B}^S$ (of rank n_S) and the last n_U columns parametrize an approximation of the unstable bundle $\mathscr{B}^U$ (of rank n_U). In particular, assume that

$$P(f(\theta))^{-1}M_0(\theta)P(\theta) - \Lambda(\theta) = E_{red}(\theta),$$

where (f, Λ) is a block-diagonal linear skew-product

$$\Lambda(\theta) = \begin{pmatrix} \Lambda_S(\theta) & 0 \\ 0 & \Lambda_U(\theta) \end{pmatrix}$$

and (f, E_{red}) is the error in the reducibility. Also assume that the dynamics of (f, Λ_S) (in $\mathbb{T}^d \times \mathbb{R}^{n_S}$) is uniformly contracting and the dynamics of (f, Λ_U) (in $\mathbb{T}^d \times \mathbb{R}^{n_U}$) is uniformly expanding.

Summing up, the main assumption of this section is that the linear skew-product (f, M_0) is approximately reducible to a block-diagonal linear skew-product (f, Λ) that is uniformly hyperbolic.

From now on, in order to simplify the manipulation of transfer operators, we will use the following notation; $\mathscr{P} := (id, P)$, $\mathscr{L} := (f, \Lambda)$, $\mathscr{L}_S := (f, \Lambda_S)$, $\mathscr{L}_U := (f, \Lambda_U)$, $\mathscr{E}_{red} := (f, E_{red})$. The fact that the dynamics of (f, Λ_S) and (f, Λ_U) are uniformly contracting and expanding, respectively, are rephrased by saying that the spectra of $\mathscr{L}^S$ and $(\mathscr{L}_U)^{-1}$ are inside the unit circle. Therefore there exist $\lambda_S < 1 < \lambda_U$, respectively, such that $\|\Lambda_S\|_\rho \leq \lambda_S$ and $\|(\Lambda_U)^{-1}\|_\rho \leq \lambda_U^{-1}$ (as shown in [10]).

Theorem 3.4. *Let $(f, M_0) : \mathbb{T}^d \times \mathbb{R}^n \to \mathbb{T}^d \times \mathbb{R}^n$ be a linear skew-product. Assume we are given:*
1.1) a continuous matrix-valued map $P : \mathbb{T}^d \to GL(\mathbb{R}^n)$, defining a linear skew-product (id, P) in $\mathbb{T}^d \times \mathbb{R}^n$;
1.2) a continuous matrix-valued map $\Lambda = \Lambda_S \times \Lambda_U : \mathbb{T}^d \to L(\mathbb{R}^{n_S}) \times GL(\mathbb{R}^{n_U})$, defining a block-diagonal linear skew-product (f, Λ) in $\mathbb{T}^d \times \mathbb{R}^{n_S} \times \mathbb{R}^{n_U}$;
1.3) a norm $\|\cdot\|_\rho$ over $\mathbb{T}^d_\rho$, with $\rho \geq 0$.

Let (f, E_{red}) be the error in the reducibility equation, where

$$E_{red}(\theta) = P(f(\theta))^{-1}M_0(\theta)P(\theta) - \Lambda(\theta).$$

Let λ be the hyperbolicity constant and σ be the error bound of the reducibility equation for Λ and P. This means that:
2.1) $\|\mathscr{L}_S\|_\rho \leq \lambda$, $\|(\mathscr{L}_U)^{-1}\|_\rho \leq \lambda$;
2.2) $\|\mathscr{E}_{red}\|_\rho \leq \sigma$;
2.3) $\lambda + \sigma < 1$.

Then, the linear skew-product (f, M_0) is uniformly hyperbolic and for all $z \in \mathbb{C}$ with $|z| = 1$,

$$\|(\mathscr{M}_0 - zId)^{-1}\|_\rho \leq \frac{1}{1 - \lambda - \sigma}.$$

This means that we can take $c_H \geq \frac{1}{1-\lambda-\sigma}$ *as the hyperbolicity bound.*

Proof. Using Theorem's 3.4 proof, we can see that for each $z \in \mathbb{C}$ with $|z| = 1$ and for each $\eta \in C^0(\mathbb{T}^d, \mathbb{R}^n)$, there exists a unique $\xi \in C^0(\mathbb{T}^d, \mathbb{R}^n)$ solving the cohomological equation

$$\Lambda(\theta)\xi(\theta) - z\xi(f(\theta)) = \eta(f(\theta))$$

such that $(\mathscr{L} - zId)$ is invertible and $\xi = (\mathscr{L} - zId)^{-1}\,\eta$.
Recalling the last part of the theorem's proof, we can now write

$$\|(\mathscr{L} - zId)^{-1}\|_\rho \leq \frac{1}{1-\lambda}.$$

Consider

$$\mathscr{M}_0 - zId = \mathscr{P}(\mathscr{L} - zId)(Id + (\mathscr{L} - zId)^{-1}\mathscr{E}_{red})\mathscr{P}^{-1}.$$

Since $\|(\mathscr{L} - zId)^{-1}\mathscr{E}_{red}\|_\rho \leq (1-\lambda)^{-1}\sigma < 1$ thanks to hypothesis 2.3), $\mathscr{M}_0 - zId$ is invertible and

$$(\mathscr{M}_0 - zId)^{-1} = \mathscr{P}(I + (\mathscr{L} - z)^{-1}\mathscr{E}_{red})^{-1}(\mathscr{L} - zId)^{-1}\mathscr{P}^{-1}.$$

Then, for any $\eta \in C^0(\mathbb{T}^d, \mathbb{R}^n)$:

$$\begin{aligned}\|(\mathscr{M}_0 - zId)^{-1}\,\eta\|_\rho &= \|\mathscr{P}^{-1}(\mathscr{M}_0 - zId)^{-1}\,\eta\|_\rho \leq \|\mathscr{P}^{-1}(\mathscr{M}_0 - zId)^{-1}\,\mathscr{P}\|_\rho\,\|\mathscr{P}^{-1}\eta\|_\rho \\ &\leq \|(I + (\mathscr{L} - z)^{-1}\mathscr{E}_{red})^{-1}\|_\rho\,\|(\mathscr{L} - zId)^{-1}\|_\rho\,\|\mathscr{P}^{-1}\eta\|_\rho \\ &\leq \frac{1}{1 - \frac{1}{1-\lambda}\sigma}\frac{1}{1-\lambda}\|\eta\|_\rho = \frac{1}{1-\lambda-\sigma}\|\eta\|_\rho.\end{aligned}$$

Finally $\|(\mathscr{M}_0 - zId)^{-1}\|_\rho \leq \frac{1}{1-\lambda-\sigma}$ (as shown in [10]).

□

Chapter 4

Fourier Transforms and Approximation Results on the One Dimensional Case

As we have seen in Theorem 2.3, there are some hypotheses that need to be satisfied in order to ensure the existence and uniqueness of a fiberwise hyperbolic invariant torus, and some of them require an explicit bound to express its true meaning, such as the invariance error for approximate invariant torus or the hyperbolicity bound. In order to compute those values, we must first think of a way to express all the information we have of our system in computable terms, and that way is Fourier series.
In this chapter we introduce a very basic notion of analytic functions followed by the concepts of Fourier Transform and Discrete Fourier Transform as well as the error committed when using one instead of the other. This leads to the main result of the chapter, which consists in a computable expression of the error produced when using approximate Fourier series. This method will be the leading force to calculate the validation theorem's bounds, which will be presented in the forthcoming chapters.

4.1 Analytic Functions, Norms and Strips

Definition 4.1. *A function f is real analytic on an open set $D \subset \mathbb{R}$ if for any $x_0 \in D$ one can write*

$$f(x) = \sum_{n=0}^{\infty} a_n (x - x_0)^n$$

where $a_i \in \mathbb{R}$ and the series is convergent to $f(x)$ for x in a neighborhood of x_0.

Now we will work with Banach spaces of real analytic functions in complex neighborhoods of real domains, which is perfectly tailored for our context.

A complex strip of $\mathbb{T}$ of width $\rho > 0$ is defined as

$$\mathbb{T}_\rho = \{\theta \in {}^{\mathbb{C}}\!/\!_{\mathbb{Z}} \, : \, |Im\, \theta| < \rho\}.$$

A function defined on $\mathbb{T}$ is real analytic if it can be analytically extended to a complex strip $\mathbb{T}_\rho$, whose boundary is $\partial\,\mathbb{T}_\rho = \{\theta \in {}^{\mathbb{C}}\!/_{\mathbb{Z}} \,:\, |\theta| = \rho\}$.
We consider analytic functions $u : \mathbb{T}_\rho \to \mathbb{C}$ such that they can be continuously extended up to the boundary of $\mathbb{T}_\rho$. We endow these functions with the norm $\|u\|_\rho = \sup\limits_{\theta\in\mathbb{T}_\rho} |u(\theta)| = \max\limits_{\theta\in\overline{\mathbb{T}}_\rho} |u(\theta)| = \max\limits_{\theta\in\partial\mathbb{T}_\rho} |u(\theta)|$, where the last equality holds by the maximum modulus principle.
Moreover, we write the Fourier expansion

$$u(\theta) = \sum_{k\in\mathbb{Z}} \hat{u}_k\, e^{2\pi i k\theta}, \quad \hat{u}_k = \int_0^1 u(\theta)\, e^{-2\pi i k\theta} d\theta$$

and we note the average of u as $< u >= \hat{u}_0 = \int_0^1 u(\theta) d\theta$. Notice that $\hat{u}_k^* = \hat{u}_{-k}$, where $\hat{u}_k^*$ denotes the complex conjugate of $\hat{u}_k$.
Then we consider the Fourier norm

$$\|u\|_{F,\,\rho} = \sum_{k\in\mathbb{Z}} |\hat{u}_k|\, e^{2\pi|k|\rho}.$$

We observe that $\|u\|_\rho \le \|u\|_{F,\,\rho}\,,\ \forall \rho > 0$.

Given an annulus $\mathscr{A} \subset \mathbb{T}\times\mathbb{R}$, a complex strip of $\mathscr{A}$ is a complex connected open neighborhood $B \subset \left({}^{\mathbb{C}}\!/_{\mathbb{Z}}\right) \times \mathbb{C}$ of $\mathscr{A}$ that projects surjectively on $\mathbb{T}$. A function on $\mathscr{A}$ is real analytic if it can be analytically extended to a complex strip B. Given a bounded analytic function $u : B \to \mathbb{C}$, we introduce the norm $\|u\|_B = \sup\limits_{z\in B} |u(z)|$.[9]

4.2 The Fourier Transform and the Discrete Fourier Transform

Now that we are ready to introduce the Discrete Fourier Transform and its properties, we provide (once again) the definition of Fourier series given a function $f : \mathbb{T} \longrightarrow \mathbb{C}$;

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

where the Fourier coefficients are given by the Fourier Transform (FT)

$$\hat{f}_k = \int_0^1 f(\theta)\, e^{-2\pi i k\theta} d\theta. \tag{4.1}$$

We consider a sample of points on the regular grid of size $N \in \mathbb{N}$, $\theta_j := \frac{j}{N}$, where $0 \le j < N$. This defines a sampling $\{f_j\}$, with $f_j = f(\theta_j)$ and a total number of points N.
The integrals in (4.1) are approximated using the trapezoidal rule on the regular grid, obtaining the Discrete Fourier Transform (DFT)

$$\tilde{f}_k = \frac{1}{N} \sum_{j=0}^{N-1} f_j\, e^{-2\pi i k\theta_j}.$$

Remark 4.2. $\tilde{f}_k$ can be defined for all $k \in \mathbb{Z}$. Moreover, they are periodic with period N, $\tilde{f}_{k+N} = \tilde{f}_k$.

The function f is approximated by the discrete Fourier approximation

$$\tilde{f}(\theta) = \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} \tilde{f}_k \, e^{2\pi i k\theta}.$$

Along this section, we will use the standard notation $[x] = \max\{j \in \mathbb{Z} : j \leq x\}$ for the integer part of x.[9]

Remark 4.3. The DFT approximation $\tilde{f}(\theta)$ interpolates the data on the grid. That is $\forall j = 0, \ldots, N-1$, $\tilde{f}(\theta_j) = f(\theta_j)$.

Notice that the stated process turns the sampling of points on the grid onto the Fourier coefficients for the DFT. The inverse process will get the Fourier coefficients for the DFT and turn them onto the values on the grid. This process is called the Inverse Discrete Fourier Transform (IDFT) and uses the following formula

$$f_j = \sum_{k=0}^{N-1} \tilde{f}_k \, e^{\frac{2\pi i}{N} jk}.$$

Remark 4.4. As we have previously stated, the Fourier coefficients are symmetrical, that is, $\hat{f}_k^* = \hat{f}_{-k}$, which holds for the DFT coefficients as well, $\tilde{f}_k^* = \tilde{f}_{-k}$. This presents a problem regarding the way we have defined the DFT. See that since we are treating the real analytic case, our function f evaluated over the points of the grid will acquire real values, but depending on the parity of the size of the grid, N, the discrete approximation will not. The reason behind this phenomenon lies on the fact that if N is odd, due to the coefficients' symmetry, the resulting function will remain real, but if N is even, then $N-1$ is odd, which means that the term $-\left[\frac{N}{2}\right]$ of the sum, called the Niquish term, will be unpaired. The lack of its symmetrical pair results on a complex function whose derivative will have the imaginary term i. This does not present a major issue since the Niquish term will naturally be very small. Nonetheless, if it is desired to look for a way to express the function f in terms of its DFT without this little problem, one shall eliminate the Niquish term, thus obtaining

$$p(\theta) = \sum_{k=-\left[\frac{N-1}{2}\right]}^{\left[\frac{N-1}{2}\right]} \tilde{f}_k \, e^{2\pi i k\theta}.$$

Although this solves the previous issue, it presents another one, the main reason why we are not taking p in our process. Since we have set the Niquish term to 0, this approximation will not interpolate the data on the grid, which is a property of great use to us. Thus we will keep using $\tilde{f}$.

4.3 Error Estimates on Approximations

4.3.1 Analytic Periodic Functions

As we have seen, there are discrete ways of expressing a function in terms of a trigonometric polynomial. The DFT supposes a great advantage for computing Fourier series with a machine. But of course, the loss of exact information when interpolating between grid points produces an approximation error. Coming up next we present the error between DFT coefficients and FT coefficients and the error when approximating a function with the DFT approximation.

Lemma 4.5. *The coefficients of the DFT are obtained from the coefficients of the FT by*

$$\tilde{f}_k = \sum_{m\in\mathbb{Z}} \hat{f}_{k+Nm}.$$

Proof. The proof for the Lemma starts by substituting f_j by its aforementioned Fourier series expression

$$\tilde{f}_k = \frac{1}{N}\sum_{j=0}^{N-1} f_j\, e^{-2\pi i k\theta_j} = \frac{1}{N}\sum_{j=0}^{N-1}\sum_{l\in\mathbb{Z}} \hat{f}_l\, e^{2\pi i l\theta_j}\, e^{-2\pi i k\theta_j} = \sum_{l\in\mathbb{Z}} \hat{f}_l \left(\frac{1}{N}\sum_{j=0}^{N-1} e^{2\pi i(l-k)\frac{j}{N}}\right).$$

Notice that $\frac{1}{N}\sum\limits_{j=0}^{N-1} e^{2\pi i(l-k)\frac{j}{N}} = 1$ if $l-k$ is a multiple of N since $\frac{l-k}{N}, j \in \mathbb{Z}$ and then $e^{2\pi i(l-k)\frac{j}{N}} = 1$ and $\frac{1}{N}\sum\limits_{j=0}^{N-1} e^{2\pi i(l-k)\frac{j}{N}} = \frac{1}{N}\sum\limits_{j=0}^{N-1} 1 = 1$.
Let's see now the case where $l-k$ is not a multiple of N.

$$\frac{1}{N}\sum_{j=0}^{N-1} e^{2\pi i(l-k)\frac{j}{N}} = \frac{1}{N}\sum_{j=0}^{N-1}(e^{2\pi i\frac{(l-k)}{N}})^j = \frac{1}{N}\frac{1-(e^{2\pi i\frac{(l-k)}{N}})^N}{1-e^{2\pi i\frac{(l-k)}{N}}} = \frac{1}{N}\frac{1-e^{2\pi i(l-k)}}{1-e^{2\pi i\frac{(l-k)}{N}}}.$$

Since $l-k\in\mathbb{Z}$, $e^{2\pi i(l-k)} = 1$ and $1-e^{2\pi i(l-k)} = 0$. By hypothesis, $l-k$ is not a multiple of N, which means that $1-e^{2\pi i\frac{(l-k)}{N}} \neq 0$.
Wrapping up, we have

$$\frac{1}{N}\sum_{j=0}^{N-1} e^{2\pi i(l-k)\frac{j}{N}} = \begin{cases} 1 & \text{if } l-k \text{ is a multiple of } N \\ 0 & \text{otherwise} \end{cases}.$$

This means that the first sum will only have terms if $l-k = Nm$ for $m\in\mathbb{Z}$, that is for the terms $l = k+Nm$ and hence

$$\tilde{f}_k = \sum_{l\in\mathbb{Z}} \hat{f}_l \left(\frac{1}{N}\sum_{j=0}^{N-1} e^{2\pi i(l-k)\frac{j}{N}}\right) = \sum_{m\in\mathbb{Z}} \hat{f}_{k+Nm}.$$

□

Proposition 4.6. *Let $f : \mathbb{T}_{\hat\rho} \longrightarrow \mathbb{C}$ be a real analytic and bounded function in the complex strip $\mathbb{T}_{\hat\rho}$ of size $\hat\rho > 0$. Let $\tilde f$ be the discrete Fourier approximation of f in the regular grid of size $N \in \mathbb{N}$ with Fourier coefficients $\tilde f_k$. Then for $k = -\left[\frac{N}{2}\right], \cdots, \left[\frac{N-1}{2}\right]$,*

$$|\tilde f_k - \hat f_k| \le S_N^*(k,\hat\rho)\cdot \|f\|_{\hat\rho}$$

where

$$S_N^*(k,\hat\rho) = \frac{e^{-2\pi\hat\rho N}}{1-e^{-2\pi\hat\rho N}}\left(e^{-2\pi\hat\rho k} + e^{2\pi\hat\rho k}\right).$$

Proof. Let $k \in \mathbb{Z}$. From Lemma 4.5 and standard bounds of the Fourier coefficients of analytic functions, we obtain

$$\begin{aligned}|\tilde f_k - \hat f_k| &= \Big|\sum_{m\in\mathbb{Z}} \hat f_{k+Nm} - \hat f_k\Big| = \Big|\sum_{m\in\mathbb{Z}\backslash\{0\}} \hat f_{k+Nm}\Big| \le \\ &\le \sum_{m\in\mathbb{Z}\backslash\{0\}} |\hat f_{k+Nm}| \le \sum_{m\in\mathbb{Z}\backslash\{0\}} e^{-2\pi\hat\rho|k+Nm|}\cdot\|f\|_{\hat\rho}.\end{aligned}$$

Then, we define

$$S_N^*(k,\hat\rho) = \sum_{m\in\mathbb{Z}\backslash\{0\}} e^{-2\pi\hat\rho|k+Nm|}$$

so we have $|\tilde f_k - \hat f_k| \le S_N^*(k,\hat\rho)\cdot\|f\|_{\hat\rho}$. Notice that if $m > 0$, $k + Nm > 0$, and if $m < 0$, $k + Nm < 0$. We must find then a suitable expression for $S_N^*(k,\hat\rho)$, so

$$\begin{aligned}S_N^*(k,\hat\rho) &= \sum_{m>0} e^{-2\pi\hat\rho(k+Nm)} + \sum_{m<0} e^{-2\pi\hat\rho(-k-Nm)} = e^{-2\pi\hat\rho k}\sum_{m>0} e^{-2\pi\hat\rho Nm} + e^{2\pi\hat\rho k}\sum_{m<0} e^{2\pi\hat\rho Nm} \\ &\le e^{-2\pi\hat\rho k}\sum_{m>0} e^{-2\pi\hat\rho Nm} + e^{2\pi\hat\rho k}\sum_{m>0} e^{-2\pi\hat\rho Nm} = \frac{e^{-2\pi\hat\rho N}}{1-e^{-2\pi\hat\rho N}}\left(e^{-2\pi\hat\rho k} + e^{2\pi\hat\rho k}\right).\end{aligned}$$

□

Theorem 4.7. *Let $f : \mathbb{T}_{\hat\rho} \longrightarrow \mathbb{C}$ be a real analytic and bounded function in the complex strip $\mathbb{T}_{\hat\rho}$ of size $\hat\rho > 0$. Let $\tilde f$ be the discrete Fourier approximation of f in the regular grid of size N. Then, for $0 \le \rho < \hat\rho$, we have*

$$\|\tilde f - f\|_\rho \le C_N(\rho,\hat\rho)\cdot\|f\|_{\hat\rho}$$

where

$$C_N(\rho,\hat\rho) = S_N^{*1}(\rho,\hat\rho) + S_N^{*2}(\rho,\hat\rho) + T_N(\rho,\hat\rho)$$

with

$$\begin{aligned}S_N^{*1}(\rho,\hat\rho) &= \frac{e^{-2\pi\hat\rho N}}{1-e^{-2\pi\hat\rho N}}\frac{e^{-2\pi(\hat\rho+\rho)} - 2e^{2\pi(\hat\rho+\rho)\left[\frac{N-1}{2}\right]} + e^{2\pi(\hat\rho+\rho)(\left[\frac{N}{2}\right]-1)} - e^{2\pi(\hat\rho+\rho)\left[\frac{N}{2}\right]} + 1}{e^{-2\pi(\hat\rho+\rho)} - 1} \\ S_N^{*2}(\rho,\hat\rho) &= \frac{e^{-2\pi\hat\rho N}}{1-e^{-2\pi\hat\rho N}}\frac{e^{2\pi(\hat\rho-\rho)} - 2e^{-2\pi(\hat\rho-\rho)\left[\frac{N-1}{2}\right]} + e^{-2\pi(\hat\rho-\rho)(\left[\frac{N}{2}\right]-1)} - e^{-2\pi(\hat\rho-\rho)\left[\frac{N}{2}\right]} + 1}{e^{2\pi(\hat\rho-\rho)} - 1} \\ T_N(\rho,\hat\rho) &= \frac{e^{-2\pi(\hat\rho-\rho)\left[\frac{N}{2}\right]} + e^{-2\pi(\hat\rho-\rho)(\left[\frac{N}{2}\right]+1)}}{1-e^{-2\pi(\hat\rho-\rho)}}.\end{aligned}$$

Proof. From the definition of the discrete Fourier approximation $\tilde{f}$ of f, we have

$$\|\tilde{f}-f\|_\rho \leq \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} |\tilde{f}_k - \hat{f}_k|\, e^{2\pi\rho|k|} + \sum_{k=-\infty}^{-\left[\frac{N+1}{2}\right]} |\hat{f}_k|\, e^{2\pi\rho|k|} + \sum_{k=\left[\frac{N}{2}\right]}^{\infty} |\hat{f}_k|\, e^{2\pi\rho|k|}.$$

From Proposition 4.6 and the growth rate properties of the Fourier coefficients of an analytic function, we get

$$\|\tilde{f}-f\|_\rho \leq (S_N^*(\rho,\hat{\rho}) + T_N(\rho,\hat{\rho}))\cdot\|f\|_{\hat{\rho}}$$

where

$$S_N^*(\rho,\hat{\rho}) = \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} S_N^*(k,\hat{\rho})\, e^{2\pi\rho|k|}$$

and

$$T_N(\rho,\hat{\rho}) = \sum_{k=-\infty}^{-\left[\frac{N+1}{2}\right]} e^{2\pi(\rho-\hat{\rho})|k|} + \sum_{k=\left[\frac{N}{2}\right]}^{\infty} e^{2\pi(\rho-\hat{\rho})|k|}.$$

Let's express $T_N(\rho,\hat{\rho})$ in computable terms. Notice that

$$\begin{aligned} T_N(\rho,\hat{\rho}) &= \sum_{k=\left[\frac{N+1}{2}\right]}^{\infty} e^{2\pi(\rho-\hat{\rho})k} + \sum_{k=\left[\frac{N}{2}\right]}^{\infty} e^{2\pi(\rho-\hat{\rho})k} = 2\sum_{k=\left[\frac{N}{2}\right]}^{\infty} e^{2\pi(\rho-\hat{\rho})k} - e^{2\pi(\rho-\hat{\rho})\left[\frac{N}{2}\right]} \\ &= \frac{2e^{-2\pi(\hat{\rho}-\rho)\left[\frac{N}{2}\right]}}{1-e^{-2\pi(\hat{\rho}-\rho)}} - e^{-2\pi(\hat{\rho}-\rho)\left[\frac{N}{2}\right]} = \frac{e^{-2\pi(\hat{\rho}-\rho)\left[\frac{N}{2}\right]} + e^{-2\pi(\hat{\rho}-\rho)(\left[\frac{N}{2}\right]+1)}}{1-e^{-2\pi(\hat{\rho}-\rho)}}. \end{aligned}$$

To obtain a suitable expression for $S_N^*(\rho,\hat{\rho})$, it will be useful to define

$$S_N(k,\hat{\rho}) = \sum_{m\in\mathbb{Z}} e^{-2\pi\hat{\rho}|k+Nm|},$$

such that, in the same way we did in the previous proposition,

$$S_N(k,\hat{\rho}) = \sum_{m\geq 0} e^{-2\pi\hat{\rho}(k+Nm)} + \sum_{m<0} e^{-2\pi\hat{\rho}(-k-Nm)} = \frac{e^{2\pi\hat{\rho}(k-N)} + e^{-2\pi\hat{\rho}k}}{1-e^{-2\pi\hat{\rho}N}}.$$

And so, we compute

$$\begin{aligned} S_N(\rho,\hat{\rho}) &= \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} S_N(k,\hat{\rho})\, e^{2\pi\rho|k|} = \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} e^{-\pi\hat{\rho}N}\, \frac{e^{2\pi\hat{\rho}(|k|-N/2)} + e^{-2\pi\hat{\rho}(|k|-N/2)}}{1-e^{-2\pi\hat{\rho}N}}\, e^{2\pi\rho|k|} \\ &= \frac{e^{-\pi\hat{\rho}N}}{1-e^{-2\pi\hat{\rho}N}} \sum_{\sigma\in\{-1,1\}} \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} e^{-2\pi(\sigma\hat{\rho}-\rho)|k|}\, e^{\pi\sigma\hat{\rho}N} \\ &= \frac{e^{-\pi\hat{\rho}N}}{1-e^{-2\pi\hat{\rho}N}} \sum_{\sigma\in\{-1,1\}} e^{\pi\sigma\hat{\rho}N}\cdot\nu(\sigma\hat{\rho}-\rho) \end{aligned}$$

where

$$\nu(\delta) = \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} e^{-2\pi\delta|k|} = \sum_{k=-\left[\frac{N-1}{2}\right]}^{\left[\frac{N-1}{2}\right]} e^{-2\pi\delta|k|} + e^{-2\pi\delta\left|-\left[\frac{N}{2}\right]\right|} = 2\sum_{k=1}^{\left[\frac{N-1}{2}\right]} e^{-2\pi\delta k} + 1 + e^{-2\pi\delta\left[\frac{N}{2}\right]}$$
$$= 2\left(\frac{e^{-2\pi\delta} - (e^{-2\pi\delta})^{\left[\frac{N-1}{2}\right]+1}}{1-e^{-2\pi\delta}}\right) + 1 + e^{-2\pi\delta\left[\frac{N}{2}\right]} = \frac{2 - 2e^{-2\pi\delta\left[\frac{N-1}{2}\right]}}{e^{2\pi\delta}-1} + 1 + e^{-2\pi\delta\left[\frac{N}{2}\right]}$$
$$= \frac{e^{2\pi\delta} - 2e^{-2\pi\delta\left[\frac{N-1}{2}\right]} + 1}{e^{2\pi\delta}-1} + e^{-2\pi\delta\left[\frac{N}{2}\right]} =$$
$$= \frac{e^{2\pi\delta} - 2e^{-2\pi\delta\left[\frac{N-1}{2}\right]} + e^{-2\pi\delta(\left[\frac{N}{2}\right]-1)} - e^{-2\pi\delta\left[\frac{N}{2}\right]} + 1}{e^{2\pi\delta}-1}.$$

Finally, we have that $S_N^*(\rho,\hat\rho)$ will be

$$S_N(\rho,\hat\rho) - \nu(\hat\rho-\rho) = \frac{e^{-\pi\hat\rho N}}{1-e^{-2\pi\hat\rho N}} \sum_{\sigma\in\{-1,1\}} e^{\pi\sigma\hat\rho N}\cdot\nu(\sigma\hat\rho-\rho) - \nu(\hat\rho-\rho)$$
$$= \frac{e^{-\pi\hat\rho N}}{1-e^{-2\pi\hat\rho N}}(e^{-\pi\hat\rho N}\nu(-\hat\rho-\rho) + e^{\pi\hat\rho N}\nu(\hat\rho-\rho)) - \nu(\hat\rho-\rho)$$
$$= \frac{e^{-2\pi\hat\rho N}}{1-e^{-2\pi\hat\rho N}}\nu(-\hat\rho-\rho) + \nu(\hat\rho-\rho)\left(\frac{1}{1-e^{-2\pi\hat\rho N}} - 1\right)$$
$$= \frac{e^{-2\pi\hat\rho N}}{1-e^{-2\pi\hat\rho N}} \frac{e^{-2\pi(\hat\rho+\rho)} - 2e^{2\pi(\hat\rho+\rho)\left[\frac{N-1}{2}\right]} + e^{2\pi(\hat\rho+\rho)(\left[\frac{N}{2}\right]-1)} - e^{2\pi(\hat\rho+\rho)\left[\frac{N}{2}\right]} + 1}{e^{-2\pi(\hat\rho+\rho)}-1} +$$
$$+ \frac{e^{-2\pi\hat\rho N}}{1-e^{-2\pi\hat\rho N}} \frac{e^{2\pi(\hat\rho-\rho)} - 2e^{-2\pi(\hat\rho-\rho)\left[\frac{N-1}{2}\right]} + e^{-2\pi(\hat\rho-\rho)(\left[\frac{N}{2}\right]-1)} - e^{-2\pi(\hat\rho-\rho)\left[\frac{N}{2}\right]} + 1}{e^{2\pi(\hat\rho-\rho)}-1}$$
$$= S_N^{*1}(\rho,\hat\rho) + S_N^{*2}(\rho,\hat\rho).$$

□

Proofs for the previous theorem and proposition have been adapted from [9].

4.3.2 Matrices of Periodic Functions

Our goal in this section is to control the propagation of the error when we perform matrix operations, mainly products and inverses, though the procedures for other operations are analogous. The results hereby presented are no more than consequences of Theorem 4.7 from the previous section.

Corollary 4.8. *Let us consider two matrix functions $A : \mathbb{T} \to \mathbb{C}^{m_1\times m_2}$, and $B : \mathbb{T} \to \mathbb{C}^{m_2\times m_3}$, such that their entries are real analytic and bounded functions in the complex strip $\mathbb{T}_{\hat\rho}$ of size $\hat\rho > 0$. We denote by AB the product matrix and $\widetilde{AB}$ the corresponding approximation given by DFT. Given a grid of size $N \in \mathbb{N}$, we evaluate A and B in the grid, and we interpolate the points $AB(\theta_j) = A(\theta_j)B(\theta_j)$. Then, we have*

$$\|AB - \widetilde{AB}\|_\rho \le C_N(\rho,\hat\rho)\,\|A\|_{\hat\rho}\,\|B\|_{\hat\rho}$$

for every $0 \le \rho < \hat\rho$.

Corollary 4.9. *Let us consider a matrix function $A : \mathbb{T} \to \mathbb{C}^{m\times m}$ whose entries are real analytic and bounded functions in the complex strip $\mathbb{T}_{\hat{\rho}}$ of size $\hat{\rho} > 0$. Given a grid of size $N \in \mathbb{N}$, we evaluate A in the grid and compute the inverses $X(\theta_j) = A(\theta_j)^{-1}$. Then, if $\widetilde{X}$ is the corresponding discrete Fourier approximation associated with the sample $X(\theta_j)$, the error $E(\theta) = Id_m - A(\theta)\widetilde{X}(\theta)$ satisfies*

$$\|E\|_\rho \leq C_N(\rho, \hat{\rho})\, \|A\|_{\hat{\rho}}\, \|\widetilde{X}\|_{\hat{\rho}}$$

for $0 \leq \rho < \hat{\rho}$. Moreover, if $\|E\|_\rho < 1$, there exists an analytic inverse $A^{-1} : \mathbb{T} \to \mathbb{C}^{m\times m}$ satisfying

$$\|A^{-1} - \widetilde{X}\|_\rho \leq \frac{\|\widetilde{X}\|_{\hat{\rho}}\, \|E\|_\rho}{1 - \|E\|_\rho}.$$

Proof. To obtain the first inequality of the Corollary, we observe that if $\widetilde{A\widetilde{X}}$ is the discrete Fourier approximation of $A\widetilde{X}$, then it turns out that

$$(A\widetilde{X})(\theta_j) = A(\theta_j)\widetilde{X}(\theta_j) = Id_m$$

for all points in the grid. This implies that $\widetilde{A\widetilde{X}} = Id_m$, and we end up with

$$\|E\|_\rho = \|Id_m - A\widetilde{X}\|_\rho = \|\widetilde{A\widetilde{X}} - A\widetilde{X}\|_\rho$$

and the inequality follows applying Corollary 4.8. The second inequality follows from the expression $E = Id_m - A\widetilde{X}$, simply writing $A^{-1} = \widetilde{X}(Id_m - E)^{-1}$ and using a Neumann series argument.[9] □

4.4 The Fast Fourier Transform

A Fast Fourier Transform (FFT) is an implementation algorithm for the Discrete Fourier Transform (DFT) but with a significant decrease of computational cost. Even though the number of operations of a regular DFT has a $O(N^2)$ order, the number of operations for the FFT has a $O(N \log N)$ order. There are several algorithms that are able to achieve such low computational cost, but the most common and used is the Cooley-Tukey FFT algorithm, which is the one we are going to explain in this section.[4][11]

The main idea of the Cooley-Tukey algorithm is to break down a DFT of any composite size $N = N_1N_2$ into many smaller DFTs of sizes N_1 and N_2. This allows us to combine this algorithm with any other algorithm for the DFT, for instance algorithms that are able to handle large prime factors that cannot be decomposed by Cooley-Tukey.
The decomposition we are going to explain is the one used in the best known use of the Cooley-Tukey algorithm. It divides the transform into two pieces of size $N/2$ at each step, which limites itself to values of $N = 2^p$ for $p \in \mathbb{N}$. This is not a problem in general since the number of sample points N can usually be chosen freely. This decomposition is called the radix-2 case, and for other factorizations of N we call them the mixed-radix cases or split-radix.

The radix-2 decimation-in-time (DIT) FFT divides a DFT of size N into two interleaved DFTs of size $N/2$ with each recursive stage.
The DFT is defined, as we have previously seen, by the formula

$$\tilde{f}_k = \frac{1}{N}\sum_{j=0}^{N-1} f_j\, e^{-\frac{2\pi i}{N} j\, k}.$$

The radix-2 DIT first computes the DFTs of the even-indexed inputs ($f_{2m} = f_0, f_2, \ldots \ldots, f_{N-2}$) and of the odd-indexed inputs ($f_{2m+1} = f_1, f_3, \ldots, f_{N-1}$), and then combines those two results to produce the DFT of the whole sequence. The algorithm rearranges the DFT of the function f_j into a sum over the even-numbered indices $j = 2m$ and a sum over the odd-numbered indices $j = 2m + 1$.

$$\begin{aligned}
\tilde{f}_k &= \frac{1}{2}\left(\frac{1}{N/2}\sum_{m=0}^{N/2-1} f_{2m}\, e^{-\frac{2\pi i}{N}(2m)\,k}\right) + \frac{1}{2}\left(\frac{1}{N/2}\sum_{m=0}^{N/2-1} f_{2m+1}\, e^{-\frac{2\pi i}{N}(2m+1)\,k}\right) \\
&= \frac{1}{2}\left(\frac{1}{N/2}\sum_{m=0}^{N/2-1} f_{2m}\, e^{-\frac{2\pi i}{N/2}m\,k}\right) + \frac{1}{2}\, e^{-\frac{2\pi i}{N}k}\frac{1}{N/2}\left(\sum_{m=0}^{N/2-1} f_{2m+1}\, e^{-\frac{2\pi i}{N/2}m\,k}\right) \\
&= \frac{1}{2}E_k + \frac{1}{2}\, e^{-\frac{2\pi i}{N}k} O_k.
\end{aligned}$$

It is clear that the sums within the last two parentheses are the DFT of the even-indexed part f_{2m} and the DFT of odd-indexed part f_{2m+1} of the function f_j. We can denote the DFT of the even-indexed part f_{2m} by E_k and the DFT of the odd-indexed part by O_k and simplify the resulting expression.

Taking advantage of the periodicity of the DFT, we know that $E_{k+\frac{N}{2}} = E_k$ and $O_{k+\frac{N}{2}} = O_k$ if $k < N/2$. Thus, we can rewrite the previous equation as

$$\tilde{f}_k = \begin{cases} \frac{1}{2}E_k + \frac{1}{2}\, e^{-\frac{2\pi i}{N}k}\, O_k\,, & for \quad 0 \le k < N/2 \\ \frac{1}{2}E_{k-N/2} + \frac{1}{2}\, e^{-\frac{2\pi i}{N}k}\, O_{k-N/2}\,, & for \quad N/2 \le k < N\,. \end{cases}$$

Noticing that

$$e^{-\frac{2\pi i}{N}(k+N/2)} = e^{-\frac{2\pi i}{N} - \pi i} = e^{-\pi i} e^{-\frac{2\pi i}{N}k} = -e^{-\frac{2\pi i}{N}k}$$

we can express $\tilde{f}_k$ as

$$\begin{aligned}
\tilde{f}_k &= \frac{1}{2}E_k + \frac{1}{2}\, e^{-\frac{2\pi i}{N}k}\, O_k \qquad for \quad 0 \le k < N/2\,, \\
\tilde{f}_{k+N/2} &= \frac{1}{2}E_k - \frac{1}{2}\, e^{-\frac{2\pi i}{N}k}\, O_k \quad for \quad 0 \le k < N/2\,.
\end{aligned}$$

Applying this method recursively, splitting into two half-size DFTs, gives a final output of a combination of E_k and $e^{-\frac{2\pi i}{N}k}\, O_k$, which is a very simple size-2 DFT. This procedure can reduce the overall runtime of the DFT, which is $O(N^2)$, to $O(N \log N)$, and moreover, increase the precision of the final results.[3]

Notice that, even though we have explained the Cooley-Tukey algorithm to transform grid points into Fourier coefficients, the algorithm works as well for the inverse process. The only difference in the procedure is the disappearance of the $1/N$ factor and the change of sign of the exponent of the complex exponential, given that the formula for the IDFT, as we stated previously, is

$$f_j = \sum_{k=0}^{N-1} \tilde{f}_k \, e^{\frac{2\pi i}{N} jk}.$$

Thus, the factor 1/2 preceeding the sums also disappears, leaving us the formula

$$f_k = \sum_{m=0}^{N/2-1} \tilde{f}_{2m} \, e^{\frac{2\pi i}{N}(2m)\,k} + \sum_{m=0}^{N/2-1} \tilde{f}_{2m+1} \, e^{\frac{2\pi i}{N}(2m+1)\,k}.$$

Manipulating these terms in the same way we previously did, we obtain

$$f_k = \sum_{m=0}^{N/2-1} \tilde{f}_{2m} \, e^{\frac{2\pi i}{N/2} m\,k} + e^{\frac{2\pi i}{N} k} \sum_{m=0}^{N/2-1} \tilde{f}_{2m+1} \, e^{\frac{2\pi i}{N/2} m\,k} = \widetilde{E}_k + e^{\frac{2\pi i}{N} k} \, \widetilde{O}_k.$$

Again, $\widetilde{E}_{k+\frac{N}{2}} = \widetilde{E}_k$ and $\widetilde{E}_{k+\frac{N}{2}} = \widetilde{O}_k$ for $k < N/2$. We can now express f_k as

$$f_k = \begin{cases} \widetilde{E}_k + e^{\frac{2\pi i}{N} k} \, \widetilde{O}_k \,, & for \quad 0 \leq k < N/2 \\ \widetilde{E}_{k-N/2} + e^{\frac{2\pi i}{N} k} \, \widetilde{O}_{k-N/2} \,, & for \quad N/2 \leq k < N \,. \end{cases}$$

This time we have

$$e^{\frac{2\pi i}{N}(k+N/2)} = e^{\frac{2\pi i}{N} + \pi i} = e^{\pi i} e^{\frac{2\pi i}{N} k} = -e^{\frac{2\pi i}{N} k}$$

Which finally gives us

$$\begin{aligned} f_k &= \widetilde{E}_k + e^{\frac{2\pi i}{N} k} \, \widetilde{O}_k & for \quad 0 \leq k < N/2 \,, \\ f_{k+N/2} &= \widetilde{E}_k - e^{\frac{2\pi i}{N} k} \, \widetilde{O}_k & for \quad N/2 \leq k < N \,. \end{aligned}$$

Chapter 5

Effective Calculation of the Error Bounds

As the title says, in this chapter we are going to explicitly calculate the invariance error bound for an approximately invariant torus and the hyperbolicity bound using some previously obtained results.

5.1 The Invariance Error Bound

From this chapter and on, we consider the class of skew-products over rotations, $(R_\omega, F) : \mathscr{A} \subset \mathbb{T} \times \mathbb{R}^n \to \mathbb{T} \times \mathbb{R}^n$ that is, we assume that the dynamics on the base torus is a rotation $f(\theta) = R_\omega(\theta) = \theta + \omega$, where $\omega \in \mathbb{T}$. Such skew-product is referred to as a *quasi-periodically forced system.* We will also assume that F is real-analytic and hence can be extended holomorphically to a complex neighborhood $\mathscr{B} \subset \mathbb{T}_r \times \mathbb{C}^n$ of $\mathscr{A}$, and moreover, $\mu_0 = \|F\|_{\mathscr{B}} := \sup_{(\theta,x)\in\mathscr{B}} |F(\theta, x)| < \infty$ and $\mu_1 = \|DF\|_{\mathscr{B}} := \sup_{(\theta,x)\in\mathscr{B}} |D_x F(\theta, x)| < \infty$. The invariance equation now is

$$F(\theta, K(\theta)) - K(\theta + \omega) = 0 \tag{5.1}$$

with $K : \mathbb{T} \to \mathbb{R}^n$.

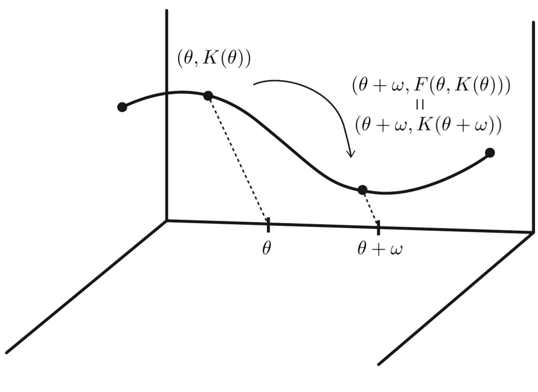

Figure 5.1: *One dimensional invariant graph under a rotation.*

The graph $\mathscr{K}$ corresponding to a solution of (5.1) is invariant under the quasi-periodically forced system, and it is also said that it is a response torus (to the quasi-periodic forcing), or that it is a *quasi-periodic invariant torus.*
In this section, we will note K_0 as the continuous map for our approximate invariant torus, allowing us to write the invariance equation in terms of K_0 and the error resulting from the current approximation, $E(\theta)$. Since our torus is approximately invariant under this perturbation, it is clear that the following expression will be satisfied

$$F(\theta, K_0(\theta)) - K_0(\theta + \omega) = E(\theta). \tag{5.2}$$

Since our theorem input object is the approximately invariant torus, we will take it as a finite sum, and in case we pick N even, the Niquish term will already be set to 0. Such object will have the form

$$K_0(\theta) = \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} \widetilde{K}_{0,k}\, e^{2\pi i k\theta} = \sum_{k=-\left[\frac{N-1}{2}\right]}^{\left[\frac{N-1}{2}\right]} \widetilde{K}_{0,k}\, e^{2\pi i k\theta}.$$

This expression is very useful since we can now easily obtain an analogous expression for $K_0(\theta+\omega)$,

$$K_0(\theta + \omega) = \sum_{k=-\left[\frac{N-1}{2}\right]}^{\left[\frac{N-1}{2}\right]} (\widetilde{K}_{0,k}\, e^{2\pi i k\omega})\, e^{2\pi i k\theta}.$$

We should keep in mind that our main goal in this section is to find a computable value for the error bound of the invariance equation, which will lead us at some point to manipulate the function $F(\theta, K_0(\theta))$ and its norm. Since the Fourier series of $\varphi(\theta) = F(\theta, K_0(\theta))$ is an infinite sum, we would like to approximate it by a finite sum

$$\widetilde{\varphi}(\theta) = \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} \widetilde{\varphi}_k\, e^{2\pi i k\theta}.$$

Then, we would like to obtain a rigorous bound of $\|\varphi(\theta) - \tilde{\varphi}(\theta)\|_{\rho=0}$.
Recalling now Equation (5.2) and taking norms and adding and substracting $\widetilde{\varphi}(\theta)$ we see that, for $\rho = 0$,

$$\|E(\theta)\|_\rho \le \|F(\theta, K_0(\theta)) - \widetilde{\varphi}(\theta)\|_\rho + \|\widetilde{\varphi}(\theta) - K_0(\theta+\omega)\|_\rho.$$

Since F is real-analytic, φ can be analytically extended to a complex strip of width $\hat{\rho} > \rho$, $\mathbb{T}_{\hat{\rho}}$, and assuming that $\forall \theta \in \mathbb{T}_{\hat{\rho}}$, $(\theta, \varphi(\theta)) \in \mathscr{B}$ and using Theorem 4.7 we see

$$\|F(\theta, K_0(\theta)) - \widetilde{\varphi}(\theta)\|_\rho = \|\varphi(\theta) - \widetilde{\varphi}(\theta)\|_\rho \le C_N(0, \hat{\rho})\, \|\varphi\|_{\hat{\rho}} \le C_N(0, \hat{\rho})\, \|F\|_{\mathscr{B}} = C_N(0, \hat{\rho})\, \mu_0.$$

We have then left to calculate the second term of the sum, which follows

$$\begin{aligned}\|\widetilde{\varphi}(\theta) - K_0(\theta+\omega)\|_\rho &= \left\| \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} (\widetilde{\varphi}_k - \widetilde{K}_{0,k}\, e^{2\pi i k\omega})\, e^{2\pi i k\theta} \right\|_\rho \\ &\le \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} |\widetilde{\varphi}_k - \widetilde{K}_{0,k}\, e^{2\pi i k\omega}| \le \tilde{\varepsilon},\end{aligned}$$

where $\tilde{\varepsilon}$ is the error at the grid. Then we find

$$\|F(\theta, K_0(\theta)) - K_0(\theta+\omega)\|_\rho \leq C_N(0,\hat{\rho})\, \|\varphi\|_{\hat{\rho}} + \tilde{\varepsilon} = C_N(0,\hat{\rho})\, \mu_0 + \tilde{\varepsilon} \leq \hat{\varepsilon},$$

where $\hat{\varepsilon}$ is the invariance bound in Theorem 2.3, and $C_N(0,\hat{\rho})$, even though it depends on the system, is very small.

5.2 The Hyperbolicity Bound

Once we have the proof from Chapter 3 of how the hyperbolicity bound is obtained, we shall follow the path drawn by the previous section and find a computable way to calculate the hyperbolicity bound of a system in $\mathbb{T} \times \mathbb{R}^n$ given a series of numerical inputs.

As we have previously done, we will focus on skew-products over rotations, for which Fourier methods are very effective. For that, following Theorem 3.4, we assume we have as inputs a continuous matrix-valued map $P : \mathbb{T} \to GL(\mathbb{R}^n)$ defining a linear skew-product (id, P) in $\mathbb{T} \times \mathbb{R}^n$ such that

$$P(\theta) = \begin{pmatrix} \sum_{k=-\left[\frac{N-1}{2}\right]}^{\left[\frac{N-1}{2}\right]} p_{1,1,k}\, e^{2\pi i k\theta} & \cdots & \sum_{k=-\left[\frac{N-1}{2}\right]}^{\left[\frac{N-1}{2}\right]} p_{1,n,k}\, e^{2\pi i k\theta} \\ \vdots & \ddots & \vdots \\ \sum_{k=-\left[\frac{N-1}{2}\right]}^{\left[\frac{N-1}{2}\right]} p_{n,1,k}\, e^{2\pi i k\theta} & \cdots & \sum_{k=-\left[\frac{N-1}{2}\right]}^{\left[\frac{N-1}{2}\right]} p_{n,n,k}\, e^{2\pi i k\theta} \end{pmatrix} =$$

$$= \sum_{k=-\left[\frac{N-1}{2}\right]}^{\left[\frac{N-1}{2}\right]} \begin{pmatrix} p_{1,1,k} & \cdots & p_{1,n,k} \\ \vdots & \ddots & \vdots \\ p_{n,1,k} & \cdots & p_{n,n,k} \end{pmatrix} e^{2\pi i k\theta}.$$

And a continuous matrix-valued map $\Lambda = \Lambda_S \times \Lambda_U : \mathbb{T} \to L(\mathbb{R}^{n_S}) \times GL(\mathbb{R}^{n_U})$ defining a block-diagonal linear skew-product (f, Λ) in $\mathbb{T} \times \mathbb{R}^{n_S} \times \mathbb{R}^{n_U}$ into which we want to reduce our system.

$$\Lambda(\theta) = \sum_{k=-\left[\frac{N-1}{2}\right]}^{\left[\frac{N-1}{2}\right]} \begin{pmatrix} \lambda_{1,1,k} & \cdots & \lambda_{1,n_S,k} & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \lambda_{n_S,1,k} & \cdots & \lambda_{n_S,n_S,k} & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \lambda_{n_S+1,n_S+1,k} & \cdots & \lambda_{n_S+1,n,k} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \lambda_{n,n_S+1,k} & \cdots & \lambda_{n,n,k} \end{pmatrix} e^{2\pi i k\theta}$$

$$\Lambda(\theta) = \begin{pmatrix} \Lambda_S(\theta) & 0 \\ 0 & \Lambda_U(\theta) \end{pmatrix}.$$

We need now to approximate $M_0(\theta)$ with a DFT approximation with a sampling of N points over the regular grid, where we have our N fixed to a even number. Thus we will have

$$\widetilde{M_0}(\theta) = \begin{pmatrix} \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} \widetilde{m}_{1,1,k}\, e^{2\pi i k\theta} & \cdots & \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} \widetilde{m}_{1,n,k}\, e^{2\pi i k\theta} \\ \vdots & \ddots & \vdots \\ \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} \widetilde{m}_{n,1,k}\, e^{2\pi i k\theta} & \cdots & \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} \widetilde{m}_{n,n,k}\, e^{2\pi i k\theta} \end{pmatrix} =$$

$$= \sum_{k=-\left[\frac{N-1}{2}\right]}^{\left[\frac{N-1}{2}\right]} \begin{pmatrix} \widetilde{m}_{1,1,k} & \cdots & \widetilde{m}_{1,n,k} \\ \vdots & \ddots & \vdots \\ \widetilde{m}_{n,1,k} & \cdots & \widetilde{m}_{n,n,k} \end{pmatrix} e^{2\pi i k\theta}.$$

Recalling that we are working with a $\rho = 0$, we will continue using the $\|\cdot\|_\rho$ norm as previously stated. The theorem also talks about the hyperbolicity constant λ. To verify this condition, we have to check that there exists a λ such that $\|\Lambda_S\|_\rho \leq \lambda < 1$. For this we will need to define the Fourier norm of a matrix. There are a couple ways we can do that. Being A an $n \times n$ matrix depending on θ, these are

1. $\|A\|_{F,\rho} = \max_{\theta \in \mathbb{T}_\rho} \|A(\theta)\|_\infty$.

2. $\|A\|_{F,\rho} = \max_{1 \leq i \leq n} \sum_{j=1}^{n} \|a_{ij}\|_{F,\rho}$.

The second option is the most convenient between those two, given that is the supremum norm of a numerical matrix made of the Fourier norm of each term of the original Fourier series matrix.[6] That is the one we are using. This norm still satisfies that $\|A(\theta)\|_\rho \leq \|A(\theta)\|_{F,\rho}$.

Now that we already have the necessary tools for bounding, we have to check if we can find a value $\lambda < 1$ such that $\|\Lambda_S\|_\rho \leq \|\Lambda_S\|_{F,\rho} \leq \lambda$. If this value exists, we have to check the second hypothesis, which is $\|\Lambda_U^{-1}\|_\rho \leq \lambda$. The calculation of this norm is not as direct as the previous one. The fact that Λ_U^{-1} is the inverse of a matrix of Fourier series breaks the correspondance between grid points and Fourier coefficients, hence, we will have to proceed differently.
Notice that

$$\|\Lambda_U^{-1}\|_\rho \leq \|\Lambda_U^{-1} - \widetilde{\Lambda_U^{-1}}\|_\rho + \|\widetilde{\Lambda_U^{-1}}\|_\rho \leq \|\Lambda_U^{-1} - \widetilde{\Lambda_U^{-1}}\|_\rho + \|\widetilde{\Lambda_U^{-1}}\|_{F,\rho}.$$

We shall take the first term of the sum apart in order to apply Corollary (4.9), which handles the error while applying a DFT upon inverted matrices as long as the function entries of our matrix can be analytically extended to a complex strip of width $\hat{\rho}$, $\mathbb{T}_{\hat{\rho}}$, which holds true for our case since we are working with real analytic functions.

$$\|\Lambda_U^{-1} - \widetilde{\Lambda_U^{-1}}\|_\rho \leq \frac{\|\widetilde{\Lambda_U^{-1}}\|_{\hat{\rho}}\, \|E(\theta)\|_\rho}{1 - \|E(\theta)\|_\rho}$$

where $E(\theta) = Id_{n_U} - \Lambda_U \, \widetilde{\Lambda_U^{-1}}$ as used in Corollary (4.9), which also gave us a very useful result, that claimed

$$\|E(\theta)\|_\rho \leq C_N(\rho, \hat{\rho}) \, \|\Lambda_U\|_{\hat{\rho}} \, \|\widetilde{\Lambda_U^{-1}}\|_{\hat{\rho}}.$$

Now we can finally write

$$\|\Lambda_U^{-1} - \widetilde{\Lambda_U^{-1}}\|_\rho \leq \frac{\|\widetilde{\Lambda_U^{-1}}\|_{\hat{\rho}} \, C_N(\rho, \hat{\rho}) \, \|\Lambda_U\|_{\hat{\rho}} \, \|\widetilde{\Lambda_U^{-1}}\|_{\hat{\rho}}}{1 - C_N(\rho, \hat{\rho}) \, \|\Lambda_U\|_{\hat{\rho}} \, \|\widetilde{\Lambda_U^{-1}}\|_{\hat{\rho}}} \leq \frac{C_N(\rho, \hat{\rho}) \, \|\Lambda_U\|_{F,\hat{\rho}} \, \|\widetilde{\Lambda_U^{-1}}\|_{F,\hat{\rho}}^2}{1 - C_N(\rho, \hat{\rho}) \, \|\Lambda_U\|_{F,\hat{\rho}} \, \|\widetilde{\Lambda_U^{-1}}\|_{F,\hat{\rho}}}.$$

The last inequality holds thanks to the fact that Λ_U is a matrix of Fourier series, which means that there is no error produced while turning back to the points of the grid and forth again to the Fourier series. However, as we have said before, this is not true for Λ_U^{-1} given that the invertion of the matrix breaks the direct and errorless correspondance between grid points and DFT coefficients.

Once we have expressed the desired norm in computable terms, is time now to check if $\|\Lambda_U^{-1}\|_\rho \leq \lambda$, that is, if

$$\|\Lambda_U^{-1}\|_\rho \leq \frac{C_N(\rho, \hat{\rho}) \, \|\Lambda_U\|_{F,\hat{\rho}} \, \|\widetilde{\Lambda_U^{-1}}\|_{F,\hat{\rho}}^2}{1 - C_N(\rho, \hat{\rho}) \, \|\Lambda_U\|_{F,\hat{\rho}} \, \|\widetilde{\Lambda_U^{-1}}\|_{F,\hat{\rho}}} + \|\widetilde{\Lambda_U^{-1}}\|_{F,\hat{\rho}} \leq \lambda.$$

In case this condition is not satisfied witht the first λ we have picked, it may be interesting to play around with the λ value and try to find another $\lambda' < 1$ such that satisfies both conditions.

Once the issue is settled and we have a suitable λ, the next step is to find the σ that bounds the error. Let E_{red} be the error in the reducibility equation, where

$$E_{red} = P(\theta + \omega)^{-1} M_0(\theta) P(\theta) - \Lambda(\theta).$$

In order to calculate a suitable bound for $\|E_{red}\|$, we can extract the norm of $P(\theta + \omega)^{-1}$ to simplify the operations.

$$\|E_{red}\|_\rho = \|P(\theta + \omega)^{-1} M_0(\theta) P(\theta) - \Lambda(\theta)\|_\rho \leq \|P(\theta + \omega)^{-1}\|_\rho \, \|M_0(\theta) P(\theta) - P(\theta + \omega)\Lambda(\theta)\|_\rho.$$

For this, we will first compute $\|P(\theta + \omega)^{-1}\|_\rho$. Notice that

$$\|P(\theta + \omega)^{-1}\|_\rho \leq \|P(\theta + \omega)^{-1} - \widetilde{P(\theta + \omega)^{-1}}\|_\rho + \|\widetilde{P(\theta + \omega)^{-1}}\|_{F,\rho}.$$

The procedure is exactly the same as the one we previously did for $\|\Lambda_U^{-1}\|_\rho$. So

$$\|P(\theta + \omega)^{-1} - \widetilde{P(\theta + \omega)^{-1}}\|_\rho \leq \frac{\|\widetilde{P(\theta + \omega)^{-1}}\|_{\hat{\rho}} \, \|E(\theta)\|_\rho}{1 - \|E(\theta)\|_\rho}.$$

Where $E(\theta) = Id_n - P(\theta + \omega) \, \widetilde{P(\theta + \omega)^{-1}}$ as we already know, plus

$$\|E(\theta)\|_\rho \leq C_N(\rho, \hat{\rho}) \, \|P(\theta + \omega)\|_{\hat{\rho}} \, \|\widetilde{P(\theta + \omega)^{-1}}\|_{\hat{\rho}}.$$

And finally

$$\|P(\theta+\omega)^{-1} - \widetilde{P(\theta+\omega)^{-1}}\|_\rho \leq \frac{C_N(\rho,\hat{\rho})\,\|P(\theta+\omega)\|_{F,\hat{\rho}}\,\|\widetilde{P(\theta+\omega)^{-1}}\|_{F,\hat{\rho}}^2}{1 - C_N(\rho,\hat{\rho})\,\|P(\theta+\omega)\|_{F,\hat{\rho}}\,\|\widetilde{P(\theta+\omega)^{-1}}\|_{F,\hat{\rho}}}.$$

The reasoning behind these inequalities and procedures is exactly the same we used for $\|\Lambda_U^{-1}\|_\rho$, since $P(\theta+\omega)^{-1}$ is as well the inverse of a trigonometric polynomials matrix.

This finishes the calculation of a computable formula for $\|P(\theta+\omega)^{-1}\|_\rho$. What we have left to discover is a way to express $\|M_0(\theta)P(\theta) - P(\theta+\omega)\Lambda(\theta)\|_\rho$ in computable terms. For that, we may proceed as we did in Chapter 3, but this time separating onto three norms

$$\begin{aligned}&\|M_0(\theta)P(\theta) - P(\theta+\omega)\Lambda(\theta)\|_\rho \leq \|M_0(\theta)P(\theta) - \widetilde{M_0(\theta)P(\theta)}\,\|_\rho + \\ &+ \|\,\widetilde{M_0(\theta)P(\theta)} - \widetilde{P(\theta+\omega)\Lambda(\theta)}\|_\rho + \|\widetilde{P(\theta+\omega)\Lambda(\theta)} - P(\theta+\omega)\Lambda(\theta)\|_\rho.\end{aligned}$$

Using Corollary (4.8) on the first and third term of the sum and the inequality of the Fourier norm, we obtain

$$\begin{aligned}&\|M_0(\theta)P(\theta) - P(\theta+\omega)\Lambda(\theta)\|_\rho \leq C_N(\rho,\hat{\rho})\,\|M_0(\theta)\|_{\hat{\rho}}\,\|P(\theta)\|_{F,\hat{\rho}} + \\ &+ \|\,\widetilde{M_0(\theta)P(\theta)} - \widetilde{P(\theta+\omega)\Lambda(\theta)}\|_{F,\hat{\rho}} + C_N(\rho,\hat{\rho})\,\|P(\theta+\omega)\|_{F,\hat{\rho}}\,\|\Lambda(\theta)\|_{F,\hat{\rho}}\end{aligned}$$

since the second term of the sum is the norm of the difference of two matrices of trigonometric polynomials, which is a matrix of trigonometric polynomials.
Notice that $\|M_0(\theta)\|_{\hat{\rho}} \leq \|M_0(\theta)\|_{\mathscr{B}} = \mu_1$, this means that $\|M_0(\theta)\|_{\hat{\rho}} \leq \mu_1$.

Now that everything is calculated, we shall write it all together for a better and more compact visualization of the expression

$$\begin{aligned}\|E_{red}\|_\rho \leq &\left(\frac{C_N(\rho,\hat{\rho})\,\|P(\theta+\omega)\|_{F,\hat{\rho}}\,\|\widetilde{P(\theta+\omega)^{-1}}\|_{F,\hat{\rho}}^2}{1 - C_N(\rho,\hat{\rho})\,\|P(\theta+\omega)\|_{F,\hat{\rho}}\,\|\widetilde{P(\theta+\omega)^{-1}}\|_{F,\hat{\rho}}} + \|\widetilde{P(\theta+\omega)^{-1}}\|_{F,\rho}\right)\cdot \\ &\cdot\Big(C_N(\rho,\hat{\rho})\,\mu_1\,\|P(\theta)\|_{F,\hat{\rho}} + \|\,\widetilde{M_0(\theta)P(\theta)} - \widetilde{P(\theta+\omega)\Lambda(\theta)}\|_{F,\hat{\rho}} + \\ &+ C_N(\rho,\hat{\rho})\,\|P(\theta)\|_{F,\hat{\rho}}\,\|\Lambda(\theta)\|_{F,\hat{\rho}}\Big) = \sigma.\end{aligned}$$

At last, we can pick our hyperbolicity bound as $c_H \geq \frac{1}{1-\lambda-\sigma}$.

Remark 5.1. Bounding $\|D_x^k F(\theta,x)\|_{\mathscr{B}} \leq \mu_k$, for $k = 0, 1, 2$, suffices to satisfy Theorem's 2.3 hypotheses.

Chapter 6

The Reducibility Method

There exist several algorithms to solve the invariance equation for finding invariant tori by using Newton-like methods. These refining algorithms lead immediately to continuation methods to obtain good initial approximations. This is the frame where we are going to work in, specifically, we are going to treat one of those methods, the so called reducibility method.

6.1 Reducibility Method

In Section 3.2 we have assumed that there is a frame adapted to the geometrical and dynamical properties of the torus (the matrix-valued map P), in such a way that the linearization of the dynamics around the torus has a simpler form (the form of a block-diagonal matrix Λ). Often one can also optimize the choice of the frame in such a way that the linearization is constant (and possibly diagonal). This reduction is not always possible but, when it holds and some extra non-resonance conditions are fulfilled, the Newton step is extremely fast and accurate when using Fourier series. Moreover, reducibility is a geometrically important property, since it gives full information about the linearization. We introduce now the formal definition of reducibility.

Definition 6.1. *An invariant torus $\mathscr{K}$ = graph(K) is reducible if the linear skew-product, with rotation ω and transfer matrix $M(\theta) = D_yF(\theta, K(\theta))$, is reducible to a constant matrix, that is, there exists a constant matrix Λ and a change of variables $P(\theta)$, known as Floquet transformation, such that the reducibility equation*

$$P(\theta+\omega)^{-1}M(\theta)P(\theta) - \Lambda = 0 \tag{6.1}$$

is satisfied.

The Floquet transformation is assumed to be 2-periodic, instead of 1-periodic, in order to include non-orientable bundles.
The key idea of the reducibility method is to consider both the invariance equation and the reducibility equation (6.1), in such a way that at each step of Newton's method the linear equation to be solved is, somehow, diagonalized.

Assume that we have an approximate invariant torus graphed by K, and we have also produced an approximate invariant frame P with reduced constant diagonal dynamics $\Lambda = diag(\lambda_1, \dots, \lambda_n)$, in such a way that

$$F(\theta, K(\theta)) - K(\theta + \omega) = E(\theta) \tag{6.2}$$

and

$$P(\theta + \omega)^{-1} D_y F(\theta, K(\theta)) P(\theta) - \Lambda = E_{red}(\theta) \tag{6.3}$$

are small.
See that if $\overline{K}(\theta) = K(\theta) + \Delta K(\theta)$ is a new approximation of a guess K, then, from equation (6.2), holds

$$D_y F(\theta, K(\theta)) \Delta K(\theta) - \Delta K(\theta + \omega) = -E(\theta). \tag{6.4}$$

In order to improve the estimates K, P, Λ, we look for $\Delta K = P\xi$, $\Delta P = PQ$, $\Delta = diag(\delta_1, \dots, \delta_n)$, so that the new approximations are $\overline{K} = K + P\xi$, $\overline{P} = P + PQ$, $\Lambda = \Lambda + \Delta$. By multiplying equation (6.4) with $P(\theta + \omega)^{-1}$, we obtain

$$P(\theta + \omega)^{-1} D_y F(\theta, K(\theta)) P(\theta) \xi(\theta) - \xi(\theta + \omega) = \eta(\theta)$$

where $\eta(\theta) = -P(\theta+\omega)^{-1} E(\theta)$. Using (6.3), and skipping the quadratically small error terms $E_{red}(\theta)\xi(\theta)$, we are lead to the cohomological equation

$$\Lambda\, \xi(\theta) - \xi(\theta + \omega) = \eta(\theta).$$

Writing the n components of the previous equation, we obtain, for $i = 1, \dots, n$:

$$\lambda_i \xi^i(\theta) - \xi^i(\theta + \omega) = \eta^i(\theta).$$

Each of these equations is diagonal in the Fourier space, obtaining for any $k \in \mathbb{Z}$:

$$(\lambda_i - e^{2\pi i k \omega}) \hat{\xi}^i_k = \hat{\eta}^i_k.$$

From here, the new approximation of the torus is given by $\overline{K} = K + P\xi$.
The corrections Q and Δ are obtained from the cohomological equation

$$\Lambda Q(\theta) - Q(\theta + \omega)\Lambda - \Delta(\theta) = -E_{red}(\theta)$$

where $E_{red}(\theta) = P(\theta + \omega)^{-1} D_y F(\theta, \overline{K}(\theta)) P(\theta) - \Lambda$. Writing the last cohomological equation component-wise we obtain, for any $i, j = 1, \dots, n$:

$$\begin{aligned} \lambda_i Q^{ij}(\theta) - Q^{ij}(\theta + \omega)\lambda_j &= -E^{ij}_{red}(\theta)\,, \quad if\ i \neq j \\ \lambda_i Q^{ii}(\theta) - Q^{ii}(\theta + \omega)\lambda_i - \delta_i &= -E^{ii}_{red}(\theta)\,, \quad if\ i = j \end{aligned}$$

These equations are, again, diagonal in Fourier space. Notice also that the adjustment δ_i in the second equation is adequate to match the average of the right-hand side (as seen in [10]). In the following section, we will specify an algorithm of the reducibility method for computing reducible invariant tori.

6.2 Reducibility Method Algorithm

Let $\mathscr{K}$ = graph(K) be an approximate fiberwise hyperbolic invariant torus graphed by $K(\theta)$, $P(\theta)$ be the matrix of an approximate adapted frame, $P^{-}(\theta)$ be the inverse of $P(\theta)$, and $\Lambda = diag(\lambda_1, \ldots, \lambda_n)$ be the reduced constant matrix on the approximate (complex) 1D invariant bundles generated by the columns of $P(\theta)$, so

$$F(\theta, K(\theta)) - K(\theta + \omega) = E(\theta),$$
$$P^{-}(\theta + \omega) D_y F(\theta, K(\theta)) P(\theta) - \Lambda = E_{red}(\theta),$$

where E and E_{red} are small.
One step on the reducibility method consists in computing the new approximations $\overline{K}, \overline{P}, \overline{\Lambda}$ through the following substeps:

1. Compute the error in the adapted frame from

$$\eta(\theta) = -P^{-}(\theta + \omega) E(\theta),$$

   and then compute the Fourier coefficients of the correction ξ from cohomological equation $\Lambda\, \xi(\theta) - \xi(\theta + \omega) = \eta(\theta)$. That is, for $i = 1, \ldots, n$, and for each $k \in \mathbb{Z}$, set

$$\hat{\xi}_k^i = \frac{1}{\lambda_i - e^{2\pi i k \omega}}\, \hat{\eta}_k^i.$$

2. Compute

$$\overline{K} = K + P\xi,$$

   and the new error in the reducibility,

$$E_{red}(\theta) = P^{-}(\theta + \omega) D_y F(\theta, \overline{K}(\theta)) P(\theta) - \Lambda.$$

3. Compute $\Delta = diag(\delta_1, \ldots, \delta_n)$ and the Fourier coefficients of the matrix Q from the cohomological equation $\Lambda Q(\theta) - Q(\theta + \omega)\Lambda - \Delta = -E_{red}(\theta)$. That is:
   For $i, j = 1, \ldots, n$, $i \neq j$, and for each $k \in \mathbb{Z}$ set

$$\hat{Q}_k^{ij} = \frac{-1}{(\lambda_i - e^{2\pi i k \omega} \lambda_j)}\, \hat{E}_{red,k}^{ij}.$$

   For $i = 1, \ldots, n$, set

$$\delta_i = \hat{E}_{red,0}^{ii}\ ,\ \hat{Q}_0^{ii} = 0,$$

   and for each $k \in \mathbb{Z} \backslash \{0\}$

$$\hat{Q}_k^{ii} = \frac{-1}{\lambda_i (1 - e^{2\pi i k \omega})}\, \hat{E}_{red,k}^{ii}\ .$$

4. Compute the new approximations

$$\overline{P} = P + PQ$$

   and

$$\overline{\Lambda} = \Lambda + \Delta.$$

(Algorithm adapted from [10]).

Chapter 7

A Case of Study: The Quasi Periodically Forced Standard Map

In this chapter we will apply the previous results, both theoretical and practical, and present the computer implementation of the reducibility method for the so called quasi-periodically forced standard map.

7.1 Quasi-periodically Forced Standard Map

The quasi-periodically forced standard map $(R_\omega, F_\varepsilon) : \mathbb{T} \times \mathbb{R}^2 \to \mathbb{T} \times \mathbb{R}^2$ is defined as

$$\begin{cases} \bar{\theta} = \theta + \omega \\ \bar{x} = x + y - \frac{b}{2\pi} sin(2\pi x) - \varepsilon sin(2\pi\theta) \\ \bar{y} = y - \frac{b}{2\pi} sin(2\pi x) - \varepsilon sin(2\pi\theta) \end{cases}$$

where we fix our rotation to the irrational coefficient $\omega = \frac{\sqrt{5}-1}{2}$.
Let's begin by finding the fixed points of the non-perturbated standard map, that is, all the points (x, y) such that

$$\begin{cases} x = x + y - \frac{b}{2\pi} sin(2\pi x) \\ y = y - \frac{b}{2\pi} sin(2\pi x), \end{cases}$$

that is, the points that satisfy

$$\begin{cases} y - \frac{b}{2\pi} sin(2\pi x) = 0 \\ -\frac{b}{2\pi} sin(2\pi x) = 0 \end{cases} \iff \begin{cases} y = 0 \\ 2\pi x = k\pi, \end{cases}$$

with $k \in \mathbb{Z}$. Hence, the fixed points are of the form $(k/2, 0)$ with $k \in \mathbb{Z}$. In particular, for $K = (1/2, 0)$ (and for all fixed points with even k) the differential matrix is

$$\begin{pmatrix} 1+b & 1 \\ b & 1 \end{pmatrix},$$

and its eigenvalues are

$$\lambda_{1,2} = \frac{2 + b \pm \sqrt{b(b+4)}}{2}.$$

Notice that for $b > 0$, then $\lambda_1 > 1$ and $\lambda_2 = 1/\lambda_1 < 1$, hence our fixed point $K = (1/2, 0)$ is hyperbolic. Therefore, from now on, we will fix $b > 0$.
Moreover, we can define

$$\Lambda = \begin{pmatrix} \frac{2+b+\sqrt{b(b+4)}}{2} & 0 \\ 0 & \frac{2+b-\sqrt{b(b+4)}}{2} \end{pmatrix}.$$

For the entries of the P matrix we will need the two eigenvectors corresponding to each eigenvalue. Taking them with unitary module, we can build our P matrix as

$$P = \begin{pmatrix} \frac{\sqrt{2}}{\sqrt{b^2+2b-b\sqrt{b(b+4)}+2}} & \frac{b-\sqrt{b(b+4)}}{\sqrt{2b^2+4b-2b\sqrt{b(b+4)}+4}} \\ \frac{-b+\sqrt{b(b+4)}}{\sqrt{2b^2+4b-2b\sqrt{b(b+4)}+4}} & \frac{\sqrt{2}}{\sqrt{b^2+2b-b\sqrt{b(b+4)}+2}} \end{pmatrix}.$$

In summary, the torus graphed by $K : \mathbb{T} \to \mathbb{R}^2$ given by $K(\theta) = (1/2, 0)$ is a fiberwise hyperbolic invariant torus of the map (R_ω, F_0). Notice that K is a fiberwise hyperbolic approximately invariant torus for $(R_\omega, F_\varepsilon)$ for ε small enough. Hence, as a consequence of Theorem 2.3, there is a fiberwise hyperbolic invariant torus for $(R_\omega, F_\varepsilon)$ for ε small enough.

7.2 Programming Procedure

Since the reducibility method provides a cleansing algorithm for the initial approximations in order to reduce the invariance and reducibility errors, we shall pick our first approximations in such a way that a continuation method regarding ε works better. The continuation method starts off from the values obtained once the algorithm is finished for the first value of ε, that is $\varepsilon = 0$. Since $\varepsilon = 0$ does not perturbate our original standard map, the initial approximations K, P and Λ for the method will be the same as the ones picked firstly for $\varepsilon = 0$. The approximations that work best for reducing the invariance error and the reducibility error, are picking Λ as the diagonal matrix with the eigenvalues of the standard map associated to a fixed hyperbolic point, which forms the K approximation, and taking P as the matrix whose columns are generated by the eigenvectors with eigenvalues corresponding to the elements in Λ.

In order to ease the possible implementation of a Fast Fourier Transform algorithm such as the Cooley-Tukey explained in Section 4.4, we will use a sampling of size $N = 2^p$, for some $p \in \mathbb{N}$.

There is a very important thing to take into account when operating with arrays of Fourier coefficients, and is related with the way we have truncated our Fourier series, which is, for a certain function f,

$$\tilde{f}(\theta) = \sum_{k=-\left[\frac{N}{2}\right]}^{\left[\frac{N-1}{2}\right]} \tilde{f}_k \, e^{2\pi i k\theta},$$

with Niquish term set to 0.
See that some terms of the sum are evaluated over negative values of k, but, of course, the indexing of an array of size N goes from 0 to $N-1$, which forces us to use $k-N$ instead of k in every point of the array between $N/2+1$ and $N-1$ when operating with the series, to displace one half of the terms to its proper place. The Niquish term, corresponding to the $N/2$ position in the array, is set to 0.

Once the initial data is properly introduced, the algorithm will provide new K, P and Λ approximations in such a way that at each step the error in both invariance and reducibility is significantly decreased. This process will run in loop until a stopping condition is satisfied, and that is, until both errors get smaller than a given tolerance. Notice that, as the ε value approaches a certain point, more steps will require the process and more slowly the error in reducibility will begin to decrease.

Even though the main objective of the algorithm is to find suitable approximations for K, P and Λ reducing the errors, we will also implement a checking of the Fourier coefficients in vector K. As we know, Fourier coefficients tend to 0 when approaching high values of N, which means that the tails (to the left and to the right) of the Fourier coefficients of K must tend to 0 (see Figure 7.1). As we have previously stated, the distribution of Fourier coefficients in an array differs from the theoretical distribution, therefore, the ends of an array will contain the major weight of Fourier coefficients while the tails will lie around the $N/2$ position (see Figure 7.2).

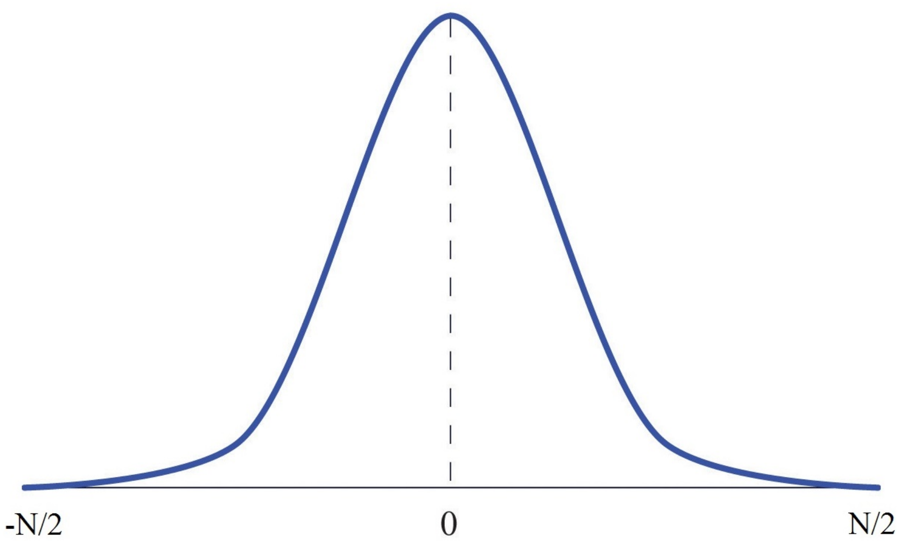

Figure 7.1: *Theoretical representation of the distribution of Fourier coefficients.*

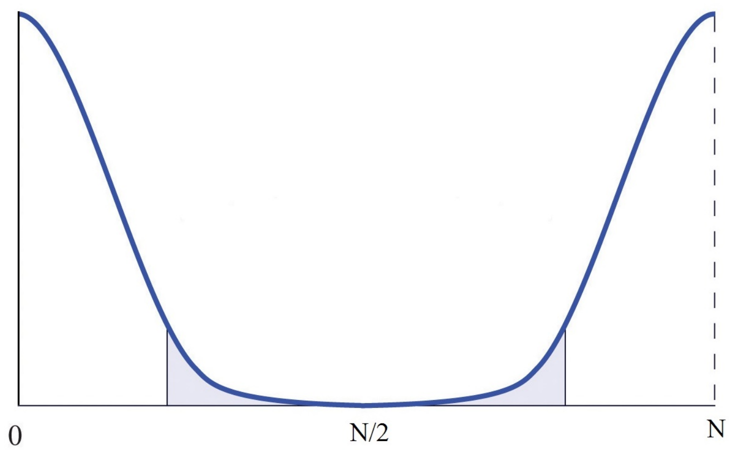

Figure 7.2: *Distribution of Fourier coefficients on an N size array (keep in mind that these graphs have been drawn with a line for a better understanding of the concept, although a representation true to the computer reality would use dots to represent each node instead of a continuous line).*

It is possible, though, that, probably due to the precision of the computer or a poor amount of grid nodes, those tails acquire higher values than expected. This can be solved by doubling the grid size, although the process cannot be taken lightly. Keep in mind that, in the array, the Fourier tails (the colored area in Figure 7.2) will dwell in the $(N/4,\ 3N/4)$ range (given the symmetry with respect to $N/2$) and hence, when extending the grid size to $N' = 2N$, we will have that $N/2 = N'/4$, but we will still want that symmetry but now against $N'/2$, so we can "move the second half to the negative positions".

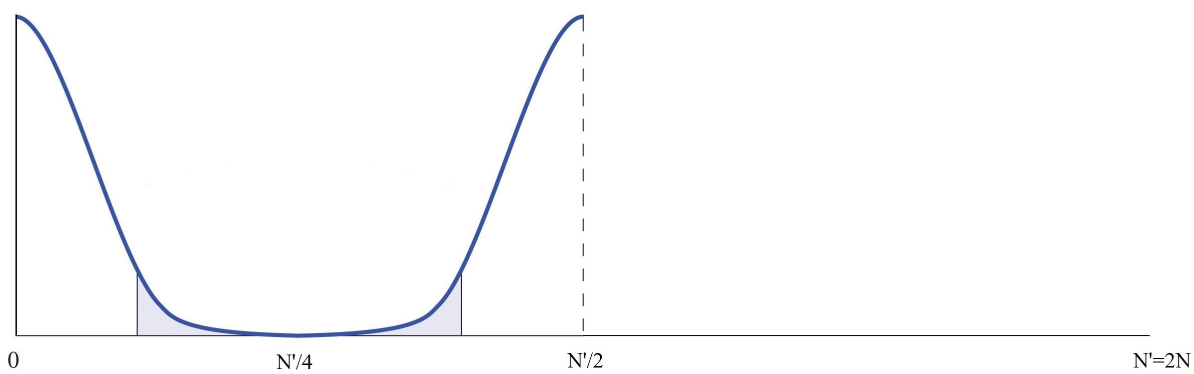

Figure 7.3: *Original distribution of Fourier coefficients once duplicated the grid size.*

To do so, we only have to move the second half of the original array to the end of the new one, by transporting the coefficients contained in the $(N'/4,\ N'/2)$ range to the $(3N'/4,\ N')$ range, that is, moving each point $N'/2$ positions to the right (assuming at every moment that the original grid size is a multiple of 4).

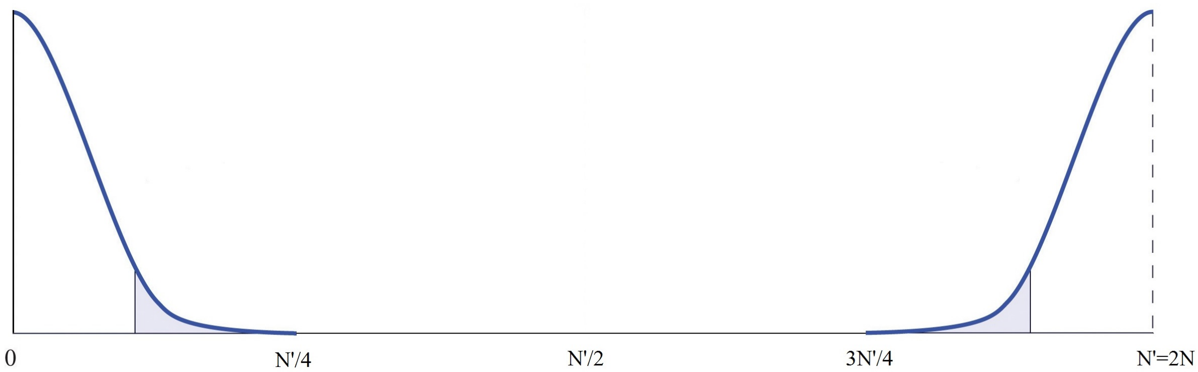

Figure 7.4: *New distribution of Fourier coefficients in a double sized grid and in between zeros.*

The checking of the size of the tails will be done once the algorithm has proved convergence for the given ε, and if the test results true, the objects that mainly rule the method (K and P) will be extended in the aforementioned way and put back to the loop to run the algorithm again until both, errors and Fourier tails, are reduced beyond a given tolerance.

Remark 7.1. When speaking about the size of the error or the size of the tails, we are always speaking about norms, such as the supremum norm when evaluating over a grid or the $\|\cdot\|_\infty$ norm when operating with Fourier coefficients.

7.3 Computation Results

The implementation of the previous methods on a program allows us to see the behavior, speed and precision of the algorithm. In order to illustrate the methodology, we have selected $b = 1.3$. Picking the initial data as explained and doubling the grid size when necessary, leads the computer to push the algorithm to its natural limit. Such limit is an ε value for which the hyperbolicity breaks and begins a chaotic behavior. This approaching to the critical value ε_c is noticeable in the disposition of our K points, given that as we are getting closer to that ε_c, K begins to fractalize. This is the reason why even though starting the algorithm with $N = 512$, it ended up suddenly increasing to $N = 16384$. Regardless the huge amount of nodes in the grid at the end of the algorithm, a FFT implmentation accelerates significantly the process. A speedtest run in the program shows that the algorithm implemented with a DFT lasted 9581 seconds to run, which are a bit more than two hours and a half, as opposed to a FFT implementation, which lasted 3.418 seconds.

To visualize the fractalization effect, we provide the plot of the x-projection against θ for several values of $\varepsilon > 0$ (since we know for $\varepsilon = 0$ our x-projected torus will be a constant line in $x = 1/2$) until we reach the closest ε to ε_c that the computer has been able to calculate.

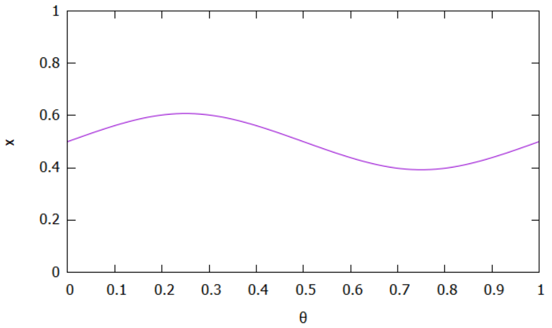

Figure 7.5: *x-projection of the curve for* $\varepsilon = 0.5$.

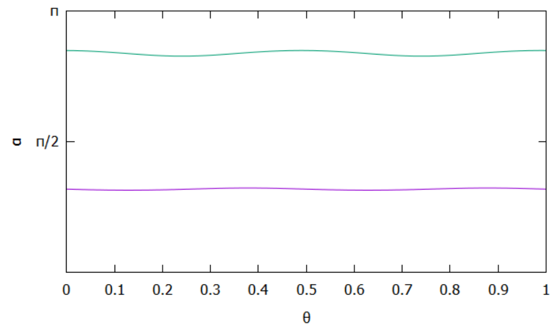

Figure 7.6: *Angles for* $\varepsilon = 0.5$.

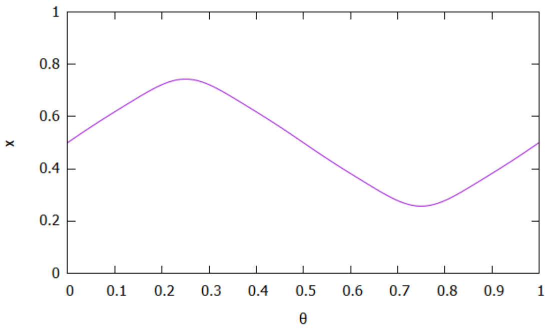

Figure 7.7: *x-projection of the curve for* $\varepsilon = 1.0$.

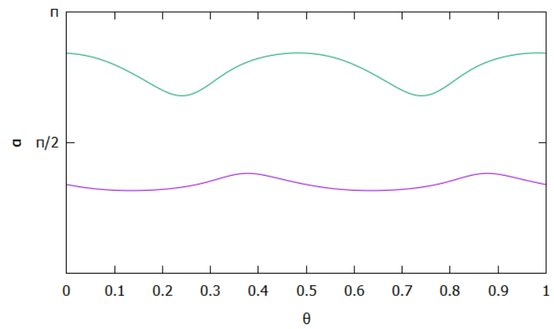

Figure 7.8: *Angles for* $\varepsilon = 1.0$.

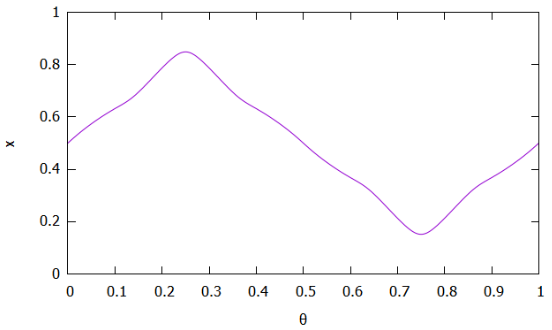

Figure 7.9: *x-projection of the curve for* $\varepsilon = 1.2$.

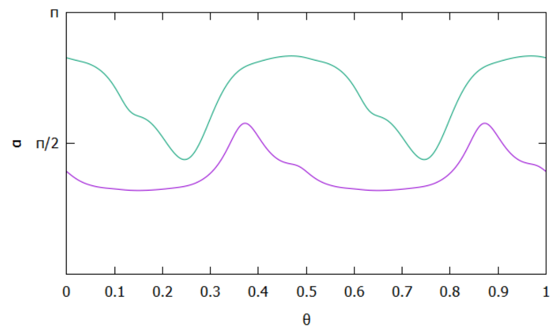

Figure 7.10: *Angles for* $\varepsilon = 1.2$.

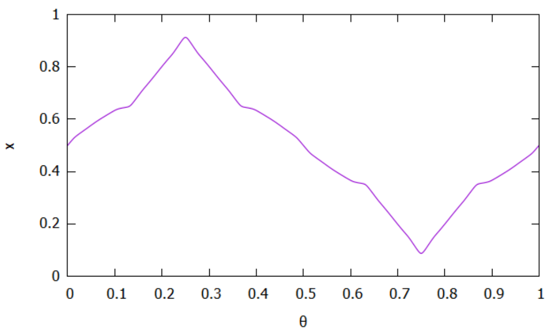

Figure 7.11: *x-projection of the curve for* $\varepsilon = 1.2342$.

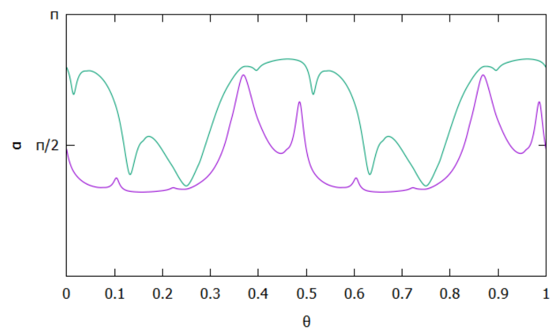

Figure 7.12: *Angles for* $\varepsilon = 1.2342$.

Figure 7.13: *Breakdown of an invariant torus. The right-side graphics are the stable and unstable subbundles represented by the angle between them and the horizontal line.*

In order to describe the properties of fiberwise hyperbolicity, we consider as observables the eigenvalues associated to the invariant bundles and the minimum and maximum distance between the invariant subbundles, as displayed in the following figures

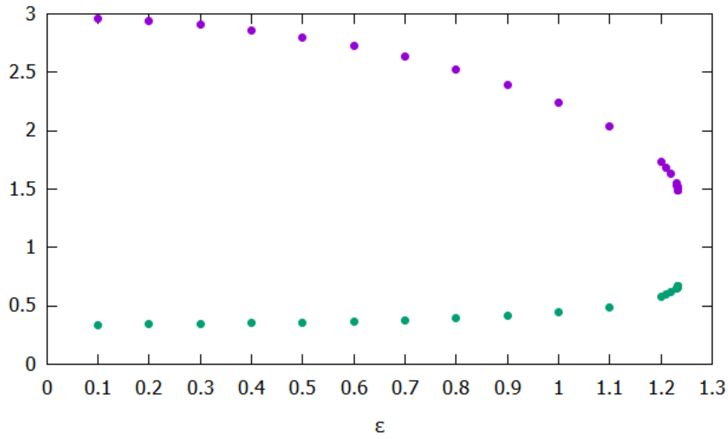

Figure 7.14: *Eigenvalues associated to the invariant bundles as a function of parameter ε.*

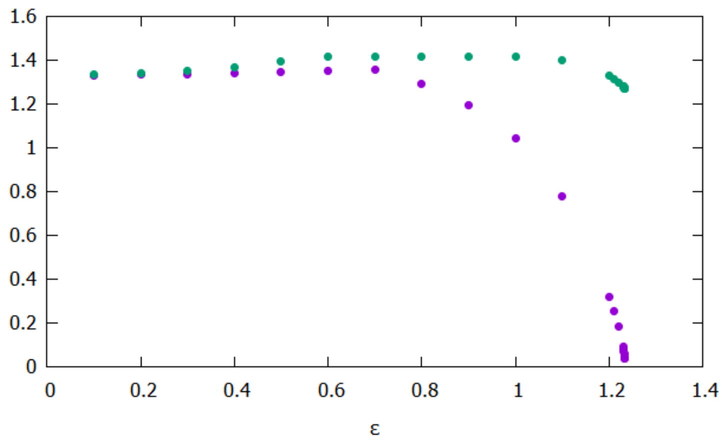

Figure 7.15: *Minimum and maximum distance between invariant subbundles as a function of parameter ε.*

In these two pictures we somehow quantify the deterioration of fiberwise hyperbolicity properties, consisting in the collapse of the invariant bundles, and then the destruction of the invariant splitting, even though the rates of contraction and expansion remain apart. This phenomenon leads to the destruction of the torus.

Conclusions

It is by now clear that skew-product systems' properties are great tools for tackling problems involving quasi-periodic forcings. In the same way, we have proved once again the efficiency of Fourier series, specially Fourier Transforms, regarding numerical computation. I personally find very interesting the differentiation between the DFT and FFT and how one of them can make your algorithm run for hours while the other runs it in seconds.

As an extension of this work, one shall wonder if, for instance, it is possible to prove the existence of real-analytic invariant tori K given an approximately invariant real-analytic torus. The answer is yes, and the proof is at hand with the methodology introduced in this work. It is interesting as well study with detail the dynamics in a torus at the verge of breakdown or explicitly calculate the computational cost of an FFT algorithm.
As for now, we can be completely satisfied with the computation of an invariant torus with a computer or the proof of such an important result as the validation theorem is.

Annex

This code calculates fiberwise hyperbolic invariant tori using the reducibility method in a continuation process for the parameter ε as well as some properties of its invariant subbundles.

```
#include <stdio.h>
#include <math.h>
#include <complex.h>
#include <stdlib.h>

#define PI 3.14159265358979323846

void dft(complex *, complex *, int);
void idft(complex *, complex *, int);
void fft(complex *, complex *, int);
void ifft(complex *, complex *, int);

double supnorm(complex *x, int N){
  /* Calculates the supremum norm */
  int k;
  double sup= 0;
  for (k= 0; k<N; k++)
    if(cabs(x[k])>sup) sup=cabs(x[k]);
  return sup;
}

double infnorm(complex *m[2][2], int N){
  /* Calculates the sub-infinity norm */
  double err00, err01, err10, err11, err= 0;
  err00=supnorm(m[0][0], N);
  err01=supnorm(m[0][1], N);
  err10=supnorm(m[1][0], N);
  err11=supnorm(m[1][1], N);

  err= err00;
  if (err01>err) err= err01;
  if (err10>err) err= err10;
  if (err11>err) err= err11;
  return err;
}

double l1norm(complex *x, int N){
  /* Calculates L1 norm */
```

```
  int k;
  double l1= 0.;
  for(k=N/2; k<N; k++){
    l1+= cabs(x[k]);
    l1+= cabs(x[N-1-k]);
  }
  return l1;
}

double l1tail(complex *x, int N){
  /* Calculates the norm of the tails of a Fourier array */
  int k;
  double l1= 0.;
  for (k= N/2; k<3*N/4; k++){
    l1+= cabs(x[k]);
    l1+= cabs(x[N-1-k]);
  }
  return l1;
}

double linftail(complex *x, int N){
  /* Calculates the sub-infinity norms of the tails */
  int k;
  double li= 0., c;
  for (k= N/2; k<3*N/4; k++){
    c= cabs(x[k]);
    li= li<c ? c : li;
  }
  return li;
}

void sum(complex *x, complex *y, complex *s, int N){
  int k;
  for(k=0; k<N; k++)
    s[k]= x[k]+y[k];
}

void mult(complex *x, complex *y, complex *m, int N){
  int k;
  for(k=0; k<N; k++)
    m[k]=x[k]*y[k];
}

void rest(complex *x, complex *y, complex *r, int N){
  int k;
  for(k=0; k<N; k++)
    r[k]=x[k]-y[k];
}

void esc(complex *x, complex *e, complex a, int N){
  int k;
  for(k=0; k<N; k++)
```

```
    e[k]=a*x[k];
}

void F(complex *K0, complex *K1, complex *FK0, complex *FK1, double b,
    double e,  int N){
  /* Calculates the image through the Standard Map */
  int j;
  for(j=0; j<N; j++){
    FK1[j]=K1[j]-(b/(2*PI))*csin(2*PI*K0[j])- e*csin((2*PI*j)/N);
    FK0[j]=K0[j]+FK1[j];
  }
}

void fourierrot(complex *x, complex *xrot, double om, int N){
  /* Rotates the Fourier coefficients */
  int k;
  for(k=0; k<N/2; k++)
    xrot[k]=x[k]*cexp(2*PI*I*k*om);
  for(k=N/2+1; k<N; k++)
    xrot[k]=x[k]*cexp(2*PI*I*(k-N)*om);
}

void cohom1(complex *y, complex *x, double lam, double mu, double om, int N
    ){
  /* Solves the cohomological equation for lambda != mu */
  int k;
  for(k=0; k<N/2; k++)
    x[k]=y[k]/(lam-mu*cexp(2*PI*I*k*om));
  for(k=N/2+1; k<N; k++)
    x[k]=y[k]/(lam-mu*cexp(2*PI*I*(k-N)*om));
  x[N/2]=0;
}

void cohom2(complex *y, complex *x, double lam, double om, int N){
  /* Solves the cohomological equation when lambda = mu */
  int k;
  x[0]=0;
  for(k=1; k<N/2; k++)
    x[k]=y[k]/(lam*(1-cexp(2*PI*I*k*om)));
  x[N/2]=0;
  for(k=N/2+1; k<N; k++)
    x[k]=y[k]/(lam*(1-cexp(2*PI*I*(k-N)*om)));
}

void checkcoh1(complex *x, complex *y, double lam, double mu, double om,
    int N){
  /* Checks the solution of the cohomological equation for lambda != mu */
  int k;
  complex *xrot=(complex*)malloc(N*sizeof(complex));
  esc(x, x, lam, N);
  fourierrot(xrot, xrot, om, N);
  esc(xrot, xrot, mu, N);
```

```
  rest(x, x, xrot, N);
  printf("\nCohom eq \t\t Eta");
  for(k=0; k<N; k++){
    printf("\n%lf+(%lf)i \t\t %lf+(%lf)i",
    creal(x[k]), cimag(x[k]), creal(y[k]), cimag(y[k]));
  }
  free(xrot);
}

void allocm(complex *m[2][2], unsigned N){
  int i, j;
  for(i=0; i<2; i++)
    for(j=0; j<2; j++)
      m[i][j]= (complex *) malloc(N*sizeof(complex));
}

void allocv(complex *v[2], unsigned N){
  int i;
  for(i=0; i<2; i++)
    v[i]= (complex *) malloc(N*sizeof(complex));
}

void reallocm(complex *m[2][2], unsigned N){
  int i, j;
  for(i=0; i<2; i++)
    for(j=0; j<2; j++)
      m[i][j]= (complex *) realloc(m[i][j], N*sizeof(complex));
}

void reallocv(complex *v[2], unsigned N){
  int i;
  for(i=0; i<2; i++)
    v[i]= (complex *) realloc(v[i], N*sizeof(complex));
}

void freev(complex *v[2]){
  int i;
  for(i=0; i<2; i++)
    free(v[i]);
}

void freem(complex *m[2][2]){
  int i, j;
  for(i=0; i<2; i++)
    for(j=0; j<2; j++)
      free(m[i][j]);
}

void matrixmult(complex *x[2][2], complex *y[2][2], complex *z[2][2], int N
    ){
  int i, j, k, l;
  complex p[2][2];
```

```
  for(k=0; k<N; k++){
    for(i=0; i<2; i++){
      for(j=0; j<2; j++){
        p[i][j]=0;
        for(l=0; l<2; l++){
          p[i][j]+=x[i][l][k]*y[l][j][k];
        }
      }
    }
    for(i= 0; i<2; i++)
      for(j= 0; j<2; j++)
        z[i][j][k]= p[i][j];
  }
}

void matrixvectmult(complex *x[2][2], complex *v[2], complex *r[2], int N){
  int i, j, k;
  complex p[2];
  for(k=0; k<N; k++){
    for(i=0; i<2; i++){
      p[i]=0;
      for(j=0; j<2; j++){
        p[i]+=x[i][j][k]*v[j][k];
      }
    }
    for(i=0; i<2; i++){
      r[i][k]=p[i];
    }
  }
}

void matrixesc(complex *x[2][2], complex *y[2][2], double a, int N){
  int i, j, k;
  for(k=0; k<N; k++){
    for(i=0; i<2; i++){
      for(j=0; j<2; j++){
        y[i][j][k]=x[i][j][k]*a;
      }
    }
  }
}

void inverse(complex *a[2][2], complex *inv[2][2], int N){
  /* Inverts a matrix */
  int k;
  complex det, adj[2][2];
  for(k=0; k<N; k++){
    det=a[0][0][k]*a[1][1][k]-a[0][1][k]*a[1][0][k];
    adj[0][0]= a[1][1][k];
    adj[1][1]= a[0][0][k];
    adj[0][1]=-a[1][0][k];
    adj[1][0]=-a[0][1][k];
```

```
      inv[0][0][k]= adj[0][0]/det;
      inv[1][1][k]= adj[1][1]/det;
      inv[1][0][k]= adj[0][1]/det;
      inv[0][1][k]= adj[1][0]/det;
    }
}

void reducerr(complex *prinv[2][2], complex *dif[2][2], complex *p[2][2],
    complex l0, complex l1, complex *err[2][2], int N){
  /* Calculates the reducibility error */
  int k;
  matrixmult(prinv, dif, err, N);
  matrixmult(err, p, err, N);
  for(k=0; k<N; k++){
    err[0][0][k]-=l0;
    err[1][1][k]-=l1;
  }
}

void matrixgf(complex *grid[2][2], complex *coef[2][2], int N){
  /* Transforms an array of matrices evaluated over a grid into matrices of
      Fourier coefficients */
  int k;
  fft(grid[0][0], coef[0][0], N);
  fft(grid[0][1], coef[0][1], N);
  fft(grid[1][0], coef[1][0], N);
  fft(grid[1][1], coef[1][1], N);
}

void matrixfg(complex *coef[2][2], complex *grid[2][2], int N){
  /* Transforms an array of matrices of Fourier coefficients into matrices
     evaluated over a grid */
  int k;
  ifft(coef[0][0], grid[0][0], N);
  ifft(coef[0][1], grid[0][1], N);
  ifft(coef[1][0], grid[1][0], N);
  ifft(coef[1][1], grid[1][1], N);
}

void matrixsum(complex *x[2][2], complex *y[2][2], int N){
  int k;
  for(k=0; k<N; k++){
    x[0][0][k]+=y[0][0][k];
    x[0][1][k]+=y[0][1][k];
    x[1][0][k]+=y[1][0][k];
    x[1][1][k]+=y[1][1][k];
  }
}

void matrixres(complex *x[2][2], complex *y[2][2], int N){
  int k;
  for(k=0; k<N; k++){
```

```
    x[0][0][k]-=y[0][0][k];
    x[0][1][k]-=y[0][1][k];
    x[1][0][k]-=y[1][0][k];
    x[1][1][k]-=y[1][1][k];
  }
}

double inverr(complex *K[2], complex *err[2], double b, double e, double om, int N){
  /* Calculates the invariance error */
  int i, k;
  complex *KrotG[2], *KrotF[2], *FK[2];
  double error;
  allocv(KrotG, N);
  allocv(KrotF, N);
  allocv(FK, N);
  F(K[0], K[1], FK[0], FK[1], b, e, N);
  for(k=0; k<N; k++){
    KrotG[0][k]=K[0][k];
    KrotG[1][k]=K[1][k];
  }
  fft(KrotG[0], KrotF[0], N);
  fft(KrotG[1], KrotF[1], N);
  fourierrot(KrotF[0], KrotF[0], om, N);
  fourierrot(KrotF[1], KrotF[1], om, N);
  ifft(KrotF[0], KrotG[0], N);
  ifft(KrotF[1], KrotG[1], N);
  printf("\nTK1 %.2le \t %.2le\n", l1tail(KrotF[0], N), l1tail(KrotF[1], N));
  printf("TKI %.2le \t %.2le\n",  linftail(KrotF[0],N),  linftail(KrotF[1], N));
  rest(FK[0], KrotG[0], err[0], N);
  rest(FK[1], KrotG[1], err[1], N);
  freev(KrotG);
  freev(KrotF);
  freev(FK);
  {
    double err0, err1;
    err0= supnorm(err[0], N);
    err1= supnorm(err[1], N);
    if(err0>err1)
      return err0;
    else
      return err1;
  }
}

void difmatrix(complex *K[2], complex *dif[2][2], double b, int N){
  /* Evaluates the differential matrix of the Standard Map over K */
  int k;
  for(k=0; k<N; k++){
    dif[0][0][k]=1-b*ccos(2*PI*K[0][k]);
```

```
      dif[0][1][k]=1;
      dif[1][1][k]=1;
      dif[1][0][k]=-b*ccos(2*PI*K[0][k]);
    }
}

void dft(complex *grid, complex *coef, int N){
  int j, k;
  long double complex sum;
  for(k=0; k<N; k++){
    sum= 0.;
    for(j=0; j<N; j++){
      sum+= ((long double complex) grid[j])*cexpl(-((long double complex) 2.0l*PI*I*k*j)/N);
    }
    coef[k]= (complex) (sum/N);
  }
}

void idft(complex *coef, complex *grid, int N){
  int j, k;
  long double complex sum;
  for(k=0; k<N; k++){
    sum= 0;
    for(j=N/2; j<N; j++){
      sum+= ((long double complex) coef[j])*cexpl(((long complex)2.0l*PI*I*k*j)/N);
      sum+= ((long double complex) coef[N-1-j])*cexpl(((long complex) 2.0l*PI*I*k*(N-1-j))/N);
    }
    grid[k]= (complex) sum;
  }
}

void separate (complex *a, int n){
  /* Copies all even elements to lower-half of a[]
  and all odd elements to upper-half of a[] */
  complex b[n/2];
  int i;
  for(i=0; i<n/2; i++)
    b[i]=a[i*2+1];
  for(i=0; i<n/2; i++)
    a[i]=a[i*2];
  for(i=0; i<n/2; i++)
    a[i+n/2]=b[i];
}

void _fft(complex *X, int N){
  int i, k;
  complex e, o, w;
  if(N<2){
  }else{
```

```
    separate(X,N);
    _fft(X, N/2);
    _fft(X+N/2, N/2);
    for(k=0; k<N/2; k++){
      e=X[k];
      o=X[k+N/2];
      w=cexp(-2.*PI*I*k/N);
      X[k]=e/2.+w*o/2.;
      X[k+N/2] =e/2.-w*o/2.;
    }
  }
}

void fft(complex *grid, complex *coef, int N){
  int k;
  for (k=0; k<N; k++){
    coef[k]=grid[k];
  }
  _fft(coef, N);
}

void _ifft(complex *X, int N){
  int i, k;
  complex e, o, w;
  if(N<2){
  }else{
    separate(X,N);
    _ifft(X, N/2);
    _ifft(X+N/2, N/2);
    for(k=0; k<N/2; k++){
      e=X[k];
      o=X[k+N/2];
      w=cexp(2.*PI*I*k/N);
      X[k]=e+w*o;
      X[k+N/2]=e-w*o;
    }
  }
}

void ifft(complex *coef, complex *grid, int N){
  int k;
  for(k=0; k<N; k++){
    grid[k]=coef[k];
  }
  _ifft(grid, N);
}

int main(){

  int N= 1024, NMAX= 16384;
  int i, maxiter= 6, j, k, r=0;
  double om=(-1+sqrt(5))/2, b= 1.3, e, e0, eok, step;
```

```c
complex *K[2], *K0[2], *P0[2][2], *KF[2], *deltaK[2], *etaG[2], *etaF[2], *P[2][2], *Prot[2][2], *PinvG[2][2], *PinvF[2][2], *ProtinvG[2][2], *ProtinvF[2][2], *dif[2][2], *invaerr[2], *reduerrG[2][2], *reduerrF[2][2], *inverserr[2][2],
*auxG[2][2], *auxF[2][2], *QF[2][2], *QG[2][2], *xiF[2], *xiG[2], v0[2], v1[2];
double lam0, lam1, delta0, delta1, error, errorredu, totalerror, tolerror = 1.e-9, tolerrorredu= 1.e-7, tail, ang0, ang1, mod, dmin, dmax, d;
char duplicate;
FILE *globalfile;
char generalfile[100];

globalfile= fopen("info.txt", "w");
if (!globalfile) {
  puts("File Error");
}

allocv(etaG, N);
allocv(etaF, N);
allocv(invaerr, N);
allocv(xiG, N);
allocv(xiF, N);
allocv(K, N);
allocv(K0, N);
allocv(KF, N);
allocv(deltaK, N);
allocm(P, N);
allocm(ProtinvG, N);
allocm(Prot, N);
allocm(ProtinvF, N);
allocm(PinvG, N);
allocm(PinvF, N);
allocm(dif, N);
allocm(reduerrG, N);
allocm(reduerrF, N);
allocm(QG, N);
allocm(QF, N);
allocm(inverserr, N);
allocm(auxG, N);
allocm(P0, N);
allocm(auxF, N);

/* Start off with e=0, and hyperbolic fixed point (1/2, 0), filling P with eigenvectors and Lambda with its eigenvalues */
lam0=(2+b+sqrt(b*(b+4)))/2;
lam1=(2+b-sqrt(b*(b+4)))/2;

for(k=0; k<N; k++){
  K0[0][k]=1/2.;
  K0[1][k]=0;
  P0[0][0][k]=sqrt(2)/sqrt(b*b+2*b-b*sqrt(b*(b+4))+2);
  P0[1][0][k]=(-b+sqrt(b*(b+4)))/sqrt(2*b*b+4*b-2*b*sqrt(b*(b+4))+4);
```

```c
      P0[0][1][k]=(b-sqrt(b*(b+4)))/sqrt(2*b*b+4*b-2*b*sqrt(b*(b+4))+4);
      P0[1][1][k]=sqrt(2)/sqrt(b*b+2*b-b*sqrt(b*(b+4))+2);
    }
    printf("\nK (%021e \t %021e)", creal(K[0][0]), creal(K[1][0])); fflush(stdout);

    e0=0;
    eok= 0;

    step=1e-1;
    duplicate= 0;
    do{
      e=e0+step;

      for(k=0; k<N; k++){
        K[0][k]=K0[0][k];
        K[1][k]=K0[1][k];
        P[0][0][k]=P0[0][0][k];
        P[0][1][k]=P0[0][1][k];
        P[1][0][k]=P0[1][0][k];
        P[1][1][k]=P0[1][1][k];
      }

      printf("\n\nEPS= %lf\n", e);
      i= 0;
      do{
        /* This loop applies the Newton method */
        printf("\n\nSTEP %i\n", i+1);
        i++;

        /* Step 1: Calculate Invariance Error */
        error=inverr(K, invaerr, b, e, om, N);
        printf("\nInvariance error (%021e)", error);
        if(error>100.){
          i=maxiter;
          break;
        }

        inverse(P, PinvG, N);
        matrixgf(PinvG, PinvF, N);
        fourierrot(PinvF[0][0], ProtinvF[0][0], om, N);
        fourierrot(PinvF[0][1], ProtinvF[0][1], om, N);
        fourierrot(PinvF[1][0], ProtinvF[1][0], om, N);
        fourierrot(PinvF[1][1], ProtinvF[1][1], om, N);
        matrixfg(ProtinvF, ProtinvG, N);
        matrixvectmult(ProtinvG, invaerr, etaG, N);

        if(error<tolerror) goto step3;

        /* Step 2: K correction and reducibility error */
        esc(etaG[0], etaG[0],-1, N);
        esc(etaG[1], etaG[1],-1, N);
```

```
      fft(etaG[0], etaF[0], N);
      fft(etaG[1], etaF[1], N);
      cohom1(etaF[0], xiF[0], lam0, 1, om, N);
      cohom1(etaF[1], xiF[1], lam1, 1, om, N);
      ifft(xiF[0], xiG[0], N);
      ifft(xiF[1], xiG[1], N);

      matrixvectmult(P, xiG, deltaK, N);
      sum(K[0], deltaK[0], K[0], N);
      sum(K[1], deltaK[1], K[1], N);
      printf("\nNew K(0) = (%021e \t %021e)", creal(K[0][0]), creal(K[1][0])); fflush(stdout);

      /* Step 3: Calculate Q and deltas */
      step3:
      difmatrix(K, dif, b, N);
      reducerr(ProtinvG, dif, P, lam0, lam1, reduerrG, N);
      errorredu=infnorm(reduerrG, N);
      printf("\nReducibility Error Norm (%021e)", errorredu);
      printf("\nReducibility Error (%021e %021e)", supnorm(reduerrG[0][0], N), supnorm(reduerrG[0][1], N));
      printf("\n (%021e %021e)\n", supnorm(reduerrG[1][0], N), supnorm(reduerrG[1][1], N));
      if(errorredu>100.){
        i= maxiter;
        break;
      }
      if(errorredu<tolerrorredu) goto step5;

      /* Step 4: Correction of P and lambda */
      matrixgf(reduerrG, reduerrF, N);
      cohom1(reduerrF[0][1], QF[0][1], -lam0, -lam1, om, N);
      cohom1(reduerrF[1][0], QF[1][0], -lam1, -lam0, om, N);
      delta0=reduerrF[0][0][0];
      delta1=reduerrF[1][1][0];
      cohom2(reduerrF[0][0], QF[0][0], -lam0, om, N);
      cohom2(reduerrF[1][1], QF[1][1], -lam1, om, N);

      matrixfg(QF, QG, N);
      matrixmult(P, QG, auxG, N);
      matrixsum(P, auxG, N);
      lam0+=delta0;
      lam1+=delta1;

      step5:
      /* Checks Fourier tails */
      fft(K[0], KF[0], N);
      fft(K[1], KF[1], N);
      matrixgf(P, PinvF, N);

      {
        double t0, t1;
```

```
        t0=linftail(KF[0], N); t1=linftail(KF[1], N);
        tail= t0>t1 ? t0 : t1;
      }

      if(isnan(tail) || tail>tolerror) i=maxiter ;

    } while((error>tolerror || errorredu>tolerrorredu) && i<maxiter);

    if( i==maxiter && ((error>tolerror || errorredu>tolerrorredu))){
      if(duplicate){
        step*=0.1;
        duplicate=0;
        continue;
      }
      duplicate=1;
      N*=2;
      if(N>NMAX) break;
      printf("Duplication to %i\n", N);

      reallocv(etaG, N);
      reallocv(etaF, N);
      reallocv(invaerr, N);
      reallocv(xiG, N);
      reallocv(xiF, N);
      reallocv(K, N);
      reallocv(K0, N);
      reallocv(KF, N);
      reallocv(deltaK, N);
      reallocm(P, N);
      reallocm(P0, N);
      reallocm(ProtinvG, N);
      reallocm(Prot, N);
      reallocm(ProtinvF, N);
      reallocm(PinvG, N);
      reallocm(PinvF, N);
      reallocm(dif, N);
      reallocm(reduerrG, N);
      reallocm(reduerrF, N);
      reallocm(QG, N);
      reallocm(QF, N);
      reallocm(inverserr, N);
      reallocm(auxG, N);
      reallocm(auxF, N);

      if(i==maxiter && ((error>tolerror || errorredu>tolerrorredu))){
        printf("Reduction of step\n");
        step*= 1.0e-1;
        fft(K0[0], KF[0], N/2);
        fft(K0[1], KF[1], N/2);
        matrixgf(P0, PinvF, N/2);
      }

```

```
    for(k=N/4; k<N/2; k++){
      KF[0][k+N/2]=KF[0][k];
      KF[1][k+N/2]=KF[1][k];
      PinvF[0][0][k+N/2]=PinvF[0][0][k];
      PinvF[0][1][k+N/2]=PinvF[0][1][k];
      PinvF[1][0][k+N/2]=PinvF[1][0][k];
      PinvF[1][1][k+N/2]=PinvF[1][1][k];
    }

    for(k=N/4; k<3*N/4; k++){
      KF[0][k]=0;
      KF[1][k]=0;
      PinvF[0][0][k]=0;
      PinvF[0][1][k]=0;
      PinvF[1][0][k]=0;
      PinvF[1][1][k]=0;
    }

    {
      printf("TK1 %.2le \t %.2le\n", l1tail(KF[0],N), l1tail(KF[1],N));
      printf("TKI %.2le \t %.2le\n", linftail(KF[0],N), linftail(KF[1],
          N));

      ifft(KF[0], deltaK[0], N);
      ifft(KF[1], deltaK[1], N);

      fft(deltaK[0], KF[0], N);
      fft(deltaK[1], KF[1], N);

      printf("TK1 %.2le \t %.2le\n", l1tail(KF[0],N), l1tail(KF[1],N));
      printf("TKI %.2le \t %.2le\n", linftail(KF[0],N), linftail(KF[1],
          N));
    }

    ifft(KF[0], K0[0], N);
    ifft(KF[1], K0[1], N);
    matrixfg(PinvF, P0, N);
  }else{
    duplicate= 0;
    if(i<=3 && e0!=0) step*=10.;
    e0=e;
    for(k=0; k<N; k++){
      K0[0][k]=K[0][k];
      K0[1][k]=K[1][k];
      P0[0][0][k]=P[0][0][k];
      P0[0][1][k]=P[0][1][k];
      P0[1][0][k]=P[1][0][k];
      P0[1][1][k]=P[1][1][k];
    }

    {
      FILE *file;
```

```
        char myfile[100];
        sprintf(myfile, "K%.6lf.txt", e0);
        printf("%s\n", myfile);
        file= fopen(myfile, "w");
        if(!file){
          puts("File Error");
        }else{
          fprintf(globalfile,"%lf %le %le ", e0, lam0, lam1);
          dmin= sqrt(2.);
          dmax= 0;

          for(k=0; k<N; k++){
            ang0=atan2(creal(P0[0][0][k]), creal(P0[1][0][k]));
            if(ang0<0) ang0+=PI;
            ang1=atan2(creal(P0[0][1][k]), creal(P0[1][1][k]));
            if(ang1<0) ang1+=PI;

            fprintf(file, "%.8lf % .8lf % .8lf % .8lf % .8lf\n", (double)k/
                N, creal(K0[0][k]), creal(K0[1][k]), ang0, ang1);

            v0[0]=creal(P[0][0][k]); v0[1]=creal(P[1][0][k]);
            v1[0]=creal(P[0][1][k]); v1[1]=creal(P[1][1][k]);
            mod=sqrt(v0[0]*v0[0]+v0[1]*v0[1]);
            v0[0]/=mod, v0[1]/=mod;
            mod=sqrt(v1[0]*v1[0]+v1[1]*v1[1]);
            v1[0]/=mod, v1[1]/=mod;

            double d1, d2;
            d1=hypot(v0[0]-v1[0],v0[1]-v1[1]);
            d2=hypot(v0[0]+v1[0],v0[1]+v1[1]);
            d=d1 < d2 ? d1 : d2;

            if(d<dmin) dmin=d;
            if(d>dmax) dmax=d;
          }
          fprintf(globalfile, " %le %le\n", dmin, dmax);
          fclose(file);
        }
      }
    }

  } while(step>1e-4);

  fclose(globalfile);
  printf("\n\nFINAL EPS OK= %lf\n", e0);
  printf("\n\nFINAL EPS KK= %lf\n", e);
  printf("FINAL N=   %i\n", N);

  freev(etaG);
  freev(etaF);
  freev(invaerr);
  freev(xiG);
```

```
  freev(xiF);
  freev(K);
  freev(KF);
  freev(deltaK);
  freem(P);
  freem(ProtinvG);
  freem(Prot);
  freem(ProtinvF);
  freem(PinvG);
  freem(PinvF);
  freem(dif);
  freem(reduerrG);
  freem(reduerrF);
  freem(QG);
  freem(QF);
  freem(inverserr);
  freem(auxG);
  freem(auxF);

  return 0;
}
```

Bibliography

[1] Bundle. https://en.wikipedia.org/wiki/Bundle_(mathematics). [Accessed June 23, 2018].

[2] Bundle map. https://en.wikipedia.org/wiki/Bundle_map. [Accessed June 23, 2018].

[3] Cooley-Tukey Algorithm. https://en.wikipedia.org/wiki/Cooley%E2%80%93Tukey_FFT_algorithm. [Accessed April 26, 2018].

[4] Fast Fourier Transform. https://en.wikipedia.org/wiki/Fast_Fourier_transform. [Accessed April 26, 2018].

[5] Fiber Bundle. https://en.wikipedia.org/wiki/Fiber_bundle#Formal_definition. [Accessed June 23, 2018].

[6] Matrix norm. https://en.wikipedia.org/wiki/Matrix_norm. [Accessed May 17, 2018].

[7] Open Mapping Theorem (functional analysis). https://en.wikipedia.org/wiki/Open_mapping_theorem_(functional_analysis). [Accessed May 6, 2018].

[8] Section (fiber bundle). https://en.wikipedia.org/wiki/Section_(fiber_bundle). [Accessed June 23, 2018].

[9] J-Ll Figueras, Alex Haro, and Alejandro Luque. Rigorous computer-assisted application of kam theory: a modern approach. *Foundations of Computational Mathematics*, 17(5):1123–1193, 2017.

[10] Alex Haro, Marta Canadell, Jordi-Lluis Figueras, Alejandro Luque, and Josep-Maria Mondelo. The parameterization method for invariant manifolds. *Appl. Math. Sci*, 195, 2016.

[11] Jake VanderPlas. Understanding the FFT Algorithm. https://jakevdp.github.io/blog/2013/08/28/understanding-the-fft/. [Accessed April 26, 2018].

[12] Rowland, Todd. "Fiber Bundle." From MathWorld–A Wolfram Web Resource. http://mathworld.wolfram.com/WhitneySum.html. [Accessed May 4, 2018].

[13] Rowland, Todd. "Vector Bundle." From MathWorld–A Wolfram Web Resource, created by Eric W. Weisstein. http://mathworld.wolfram.com/FiberBundle.html. [Accessed April 13, 2018].

[14] Silly rabbit at the English Wikipedia. Bundle section. https://en.wikipedia.org/wiki/File:Bundle_section.svg. [Accessed June 23, 2018].

[15] Weisstein, Eric W. "Whitney Sum." From MathWorld–A Wolfram Web Resource, created by Eric W. Weisstein. http://mathworld.wolfram.com/WhitneySum.html. [Accessed May 4, 2018].