LYAPUNOV EXPONENTS FOR NON-CLASSICAL MULTIDIMENSIONAL CONTINUED FRACTION ALGORITHMS

V. Baladi and A. Nogueira

November 1995

Abstract. We introduce a simple geometrical two-dimensional continued fraction algorithm inspired from dynamical renormalization. We prove that the algorithm is weakly convergent, and that the associated transformation admits an ergodic absolutely continuous invariant probability measure. Following Lagarias, its Lyapunov exponents are related to the approximation exponents which measure the diophantine quality of the continued fraction. The Lyapunov exponents for our algorithm and related ones, also introduced in this article, are studied numerically.

1. Introduction

Most one-dimensional or multidimensional continued fraction algorithms, for example the Jacobi-Perron algorithm, can be given a simple geometrical definition as follows. Fixing the dimension $d \geq 1$, our aim is to obtain successive rational approximations of vectors θ in $[0,1]^d$. We view such a vector as the direction $\Theta = (\theta_1, \dots, \theta_d, 1)$ of a half-line in the positive quadrant of $\mathbb{R}^{d+1}$ and the algorithm produces a sequence of unimodular bases $(\ell_{1j}, \ell_{2j}, \dots, \ell_{(d+1)j})$ of the lattice $\mathbb{Z}^{d+1}$ such that the direction Θ lies in the positive quadrant of all bases and the directions of the basis vectors converge to Θ (an algorithm with this last property for all θ — sometimes only almost all θ — is called *weakly convergent*). The approximations for θ are given by the vectors $(\ell_{1j}/\ell_{(d+1)j}, \dots, \ell_{dj}/\ell_{(d+1)j})$. See e.g. [Br2] for references about this geometrical description, and also [B] or [L] for other definitions of convergence and a list of classical continued fraction algorithms.

Inspired by the renormalization techniques described in [BRTT] for torus rotations, we introduce in this paper a simple two-dimensional continued fraction algorithm, called algorithm $\mathcal{A}$, which has a rather natural geometric definition. This algorithm can be viewed as a path in a tree of possible algorithms, and its weak convergence can be proved by a very economical application of finite-dimensional projective metrics. Other ergodic properties are obtained in a similar way, allowing us to apply a theorem of Lagarias [L] on the existence of Lyapunov exponents for the algorithm and their relationship with the approximation exponents (see Section 2 for definitions). A numerical study of this

1991 *Mathematics Subject Classification.* 11J70, 11K50, 28D05.

Typeset by $\mathcal{AMS}$-TEX

algorithm and of the ones corresponding to some other paths (including randomly chosen ones) in our tree indicates that they can be very favorably compared with commonly used methods such as the Jacobi-Perron algorithm. We also define (and obtain partial results on) adaptations of the algorithm to higher-dimensions.

The paper is organized as follows: In Section 2 we give the geometrical and analytical definitions of algorithm $\mathcal{A}$, show how it can be viewed as a path in a tree of algorithms related to the renormalizations introduced in [BRTT], and state our main result (Theorem 1). In Section 3, we show that our algorithm is (a little bit better than) weakly convergent (Lemma 2). In Section 4, we prove that the transformation associated with the algorithm admits an ergodic absolutely continuous invariant probability measure with good properties (Lemma 3) which yields Theorem 1 by using [L]. Finally, we present our numerical results in Section 5.

We are very grateful to E. Ghys and K. Khanin for useful conversations. V.B. is also specially thankful to S. Oliffson Kamphorst and C. Tresser. This article was initiated during a visit of A.N. to E.N.S. Lyon and completed during visits of V.B. to U.F. Rio de Janeiro and I.M.P.A., and of A.N. to E.T.H. Zürich. The hospitality and financial support of these institutions is gratefully acknowledged. V.B. is also thankful to the C.C.C.I. (France) and the Société Académique de Genève (Switzerland) for financial support.

2. Definition of the algorithms – Statement of the main result

We shall describe below in each dimension $d \geq 1$ a tree of algorithms related to the renormalizations studied in [BRTT]. We concentrate now on one specific two-dimensional algorithm (that we call algorithm $\mathcal{A}$) which will occur as a particular path in the tree, and which turns out to be the easiest to study rigorously. As we shall see in the course of this article, some of our proofs apply to other paths, whereas we only have conjectures and/or numerical evidence in the general case.

2.1 Geometric and analytical description of the algorithm.

Let $\theta = (\theta_1, \theta_2)$ be the vector to be approximated (we may assume $0 < \theta_1 \leq \theta_2 \leq 1 + \theta_1$) and $\Theta = (\theta_1, \theta_2, 1)$ be the direction we wish to approach. Representing Θ by the first two coordinates $\alpha(\theta) = (\alpha_1, \alpha_2) = (\theta_1/(\theta_1 + \theta_2 + 1), \theta_2/(\theta_1 + \theta_2 + 1))$ of the intersection of the ray $\lambda\Theta$, $\lambda > 0$, with the two-dimensional simplex $\bar{\Delta} = \{(x_1, x_2, x_3) \in \mathbb{R}^3_+ \mid x_1 + x_2 + x_3 = 1\}$, we observe that $0 < \alpha_1 \leq \alpha_2 \leq 1/2$. (See Figure 1.) We denote by

$$\Delta = \Delta_2 = \{(\alpha_1, \alpha_2, \alpha_3) \in \mathbb{R}^3_+ \mid \alpha_1 + \alpha_2 + \alpha_3 = 1\,, 0 \leq \alpha_1 \leq \alpha_2 \leq 1/2\}$$

the subset of the simplex $\bar{\Delta}$ thus obtained.

Figure 1 Intersection of the ray $\lambda\Theta$ with $\Delta = \Delta_2$

Writing points in the coordinates given by the basis e_1, e_2, e_3 constructed at the $n-1$-th step, the ray $\lambda\Theta$ intersects each of the three two-dimensional planes $P_i = \{x \in$

$\mathbb{R}^3 \mid x_i = 1\}$ $(i = 1, 2, 3)$ at a point $\lambda_{j(i)}\Theta$ for some $0 < \lambda_1 \le \lambda_2 \le \lambda_3$. Observe that if the three intersection points coincide, then we may (but will not) stop iterating the algorithm because our ray has positive integer coordinates in the current basis.

Let us first assume that the three intersection points are distinct. Then the *last intersection point* $p = \lambda_3\Theta$ lies in the interior of a square $\{(u, v) \mid 0 \le u, v \le 1\}$ of the lattice $\mathbb{Z}^2$ of the corresponding hyperplane $P_{i(3)}$. Dividing this square in four subsquares according to whether $u \ge 1/2$ or $u \le 1/2$ and $v \ge 1/2$ or $v \le 1/2$, and assuming first that $u(p)$ and $v(p)$ are both different from $1/2$, the point p lies in a well-defined subsquare. (See Figure 2.a.)

We shall now define the new simplex $\bar{\Delta}'$ and the new basis e_1', e_2', e_3'. We will write (α_1', α_2') for the (e_j') coordinates of the intersection of the ray $\lambda\Theta$ and the new simplex $\bar{\Delta}'$. We choose the three new basis elements by setting e_3' to be the common vertex of the square and the selected subsquare, and e_1', e_2' two vertices joined to e_3' by an edge of the square, ordered in such a way as to guarantee the property $\alpha_1' \le \alpha_2'$. Then $\bar{\Delta}'$ is the simplex generated by the e_i' and we may define a subsimplex $\Delta' \subset \bar{\Delta}'$ containing p as above. (See Figure 2.b.) Observe that we could have used exactly the same procedure by choosing the second or first intersected plane P_i instead of the last one.

Figure 2 a) Intersection of the ray Θ with the planes $\mathcal{P}_i$ b) Determination of the new basis

If there is any ambiguity in the above choices ($\alpha_1' = \alpha_2'$, or $u(p)$ or $v(p)$ equal to $1/2$, or coincidence of intersection points of the ray and the hyperplanes) we may either use a predefined rule or pick arbitrarily among the several possibilities: this will not influence the discussion in the present work.

Clearly, by construction, the ray to be approximated lies in the positive quadrant of each basis and the property $0 \le \alpha_1 \le \alpha_2 \le 1/2$ is preserved. However, the property that $\alpha_1 > 0$ is not necessarily preserved. If at some iterate $\alpha_1 = 0$ but $\alpha_2 > 0$, the ray $\lambda\Theta$ will only intersect the hyperplanes P_2 and P_3. In this case we could (but will not) decide to switch to a one-dimensional continued fraction algorithm, instead, we will just continue applying algorithm $\mathcal{A}$ taking at each step the last (i.e. second) intersected hyperplane (note that the property $\alpha_1 = 0$ will remain true for further iterations). If at some step $\alpha_1 = \alpha_2 = 0$ then the ray only intersects the hyperplane P_3. We could then decide to stop iterating our algorithm (the continued fraction having been completely obtained) but will instead continue iterating using the only available hyperplane. We shall call the cases where $\alpha_1 = 0$ *degenerate.*

We now turn to the analytic description of the algorithm $\mathcal{A}$. We describe the algorithm with an auxiliary map $T : \Delta \to \Delta$ and partial quotient matrix $A : \Delta \to GL(3, \mathbb{Z})$ (both defined Lebesgue almost everywhere), associating to $\alpha \in \Delta$ a sequence of matrices in $GL(3, \mathbb{Z})$ by setting $L^{(0)} \equiv Id$ and

$$L^{(n)}(\alpha) = A(\alpha)A(T(\alpha))A(T^2(\alpha)) \cdots A(T^{n-1}(\alpha)) \,. \tag{2.1}$$

The three columns $(\ell_{1j}, \ell_{2j}, \ell_{3j}) \in \mathbb{Z}^3$, $j = 1, 2, 3$ of the matrix $L^{(n)}(\alpha)$ are the elements of the n-th basis and the three rational vectors $(\ell_{1j}/\ell_{3j}, \ell_{2j}/\ell_{3j})$, $j = 1, 2, 3$ are the successive approximations of (θ_1, θ_2) if $\alpha = \alpha(\theta)$. An algorithm defined by two maps T, A as above (up to trivial changes of coordinates such as viewing T as a function of θ instead of α and transposing matrices) is called a *Markovian multidimensional continued fraction algorithm* (see e.g. [L]).

We now give the definitions for T and L, using $\{x\}$ and $[x]$ do denote the fractional and integer parts of a real number x. First consider the case where the last intersected hyperplane is P_1: this is equivalent to the condition

$$\theta_2/\theta_1 = \alpha_2/\alpha_1 \geq 1 \text{ (always satisfied)} \qquad \text{and } 1/\theta_1 = (1-\alpha_1-\alpha_2)/\alpha_1 \geq 1\,. \tag{2.2}$$

Then we set

$$\begin{cases} \beta_1(\alpha_1,\alpha_2) &= \min(\{\frac{\alpha_2}{\alpha_1}\}, 1-\{\frac{\alpha_2}{\alpha_1}\})\,, \\ \beta_2(\alpha_1,\alpha_2) &= \min(\{\frac{1-\alpha_1-\alpha_2}{\alpha_1}\}, 1-\{\frac{1-\alpha_1-\alpha_2}{\alpha_1}\})\,, \end{cases} \tag{2.3}$$

and

$$T(\alpha_1,\alpha_2) = (\min(\beta_1,\beta_2), \max(\beta_1,\beta_2))\,. \tag{2.4}$$

Setting

$$\begin{cases} k &= [\theta_2/\theta_1] = [\alpha_2/\alpha_1] \\ j &= [1/\theta_1] = [(1-\alpha_1-\alpha_2)/\alpha_1]\,, \end{cases} \tag{2.5}$$

we have $k \geq 1$ and $j \geq 1$ except if we are in a degenerate case. We also have, for example,

$$A(\alpha_1,\alpha_2) = \begin{cases} \begin{pmatrix} 1 & 1 & 1 \\ k+1 & k & k \\ j & j+1 & j \end{pmatrix}, & \text{if } \{\frac{\alpha_2}{\alpha_1}\} < \{\frac{1-\alpha_1-\alpha_2}{\alpha_1}\} < 1/2, \\ \begin{pmatrix} 1 & 1 & 1 \\ k & k+1 & k \\ j+1 & j & j \end{pmatrix}, & \text{if } \{\frac{1-\alpha_1-\alpha_2}{\alpha_1}\} < \{\frac{\alpha_2}{\alpha_1}\} < 1/2. \end{cases} \tag{2.6}$$

In the cases left undefined in (2.6), the matrix A takes one of the following forms (up to exchanging the first two columns):

$$\begin{pmatrix} 1 & 1 & 1 \\ k & k+1 & k+1 \\ j+1 & j & j+1 \end{pmatrix}, \begin{pmatrix} 1 & 1 & 1 \\ k & k+1 & k \\ j & j+1 & j+1 \end{pmatrix}, \begin{pmatrix} 1 & 1 & 1 \\ k & k+1 & k+1 \\ j & j+1 & j \end{pmatrix}. \tag{2.7}$$

The maps T and A corresponding to the last intersection with P_2 or P_3 are similar. More precisely, one just needs to exchange α_2 and α_1 in (2.3–2.6) to get the formula for the intersection with P_2 (this case only occurs on the zero-measure set defined by $\alpha_1 = \alpha_2$), and one should take

$$\begin{cases} \beta_1(\alpha_1,\alpha_2) &= \min(\{\frac{\alpha_1}{1-\alpha_1-\alpha_2}\}, 1-\{\frac{\alpha_1}{1-\alpha_1-\alpha_2}\})\,, \\ \beta_2(\alpha_1,\alpha_2) &= \min(\{\frac{\alpha_2}{1-\alpha_1-\alpha_2}\}, 1-\{\frac{\alpha_2}{1-\alpha_1-\alpha_2}\})\,, \end{cases} \tag{2.8}$$

and

$$\begin{cases} k & = [\alpha_1/(1-\alpha_1-\alpha_2)] \\ j & = [\alpha_2/(1-\alpha_1-\alpha_2)]\,, \end{cases} \tag{2.9}$$

when considering the intersection with P_3 (again, $k \geq 1$ and $j \geq 1$ except in the degenerate cases).

Note that the formulas for T and A that we would obtain by considering the first (respectively the second) intersected hyperplane instead of the last one would be exactly the same: the only difference is that k and/or j in the expressions for $A(\alpha_1, \alpha_2)$ would vanish even in non degenerate cases.

Observe also that the geometrical description of algorithm $\mathcal{A}$ implies that it is of the *linear simplex-splitting* type (see e.g. [L]), i.e., the matrix $\widetilde{A} := A^{-1}$ defines a piecewise linear map $\widetilde{T}$ on the homogeneous cone $\{y \in \mathbb{R}^3 \mid y_i \geq 0\,, y_3 = \max_i y_i\}$, for which there is a countable partition into homogeneous cones on each of which $\widetilde{T}$ is homogeneous linear with $\tilde{T}(y) = \tilde{A}(y)$. Finally, the map $\widetilde{T}$ is constant on rays, and a suitable choice of representative produces the map T. (See the proof of Lemma 3 in Section 4 for more about this.)

2.2 Approximation exponents.

In order to state our main result, we need some definitions valid for general $d \geq 1$ (we follow the terminology in [L]). The *Diophantine approximation exponent* $\eta(\theta)$ of $\theta \in [0,1]^d$ is defined by

$$\eta(\theta) = \limsup_{q\to\infty} \sup_{\substack{w=(p_1,\ldots,p_d,q)\in\mathbb{Z}^{d+1} \\ 0\leq p_i\leq q}} \eta(w,\theta)\,, \tag{2.10}$$

where the *Roth exponent* $\eta(w,\theta)$ for $w = (p_1, \ldots, p_d, q)$ is set to be

$$\eta(w,\theta) = \frac{-\log\|\theta - \tilde{w}\|}{\log q}\,, \tag{2.11}$$

where $\tilde{w} = (p_1/q, \ldots, p_d/q) \in \mathbb{Q}^d$ and $\|\cdot\|$ denotes euclidean norm. If an algorithm $\mathcal{D}$ produces positive matrices $L^{(n)} \in GL(d+1,\mathbb{Z})$ whose columns $w_i^{(n)}$ describe the successive approximations of $\Theta(\theta)$, we define the *best approximation exponent for θ using the algorithm $\mathcal{D}$* by

$$\eta_{\mathcal{D}}(\theta) := \limsup_{n\to\infty}\Big\{\max_{1\leq i\leq d+1} \eta(w_i^{(n)}(\theta),\theta)\Big\}\,, \tag{2.12}$$

and the *uniform approximation exponent for θ using $\mathcal{D}$* by

$$\eta^*_{\mathcal{D}}(\theta) := \liminf_{n\to\infty}\Big\{\min_{1\leq i\leq d+1} \eta(w_i^{(n)}(\theta),\theta)\Big\} \leq \eta_{\mathcal{D}}(\theta)\,. \tag{2.13}$$

It follows from Dirichlet's theorem (see e.g. [L]) that $\eta^*_{\mathcal{D}}(\theta) \leq 1+(1/d)$ for any algorithm $\mathcal{D}$ and almost all θ. An algorithm $\mathcal{D}$ which realizes the Dirichlet bound $\eta^*_{\mathcal{D}}(\theta) = 1+$

$(1/d)$ for almost every θ is called an *optimal* algorithm. The ordinary one-dimensional continued fraction algorithm is optimal but the existence of optimal algorithms for any $d \geq 2$ is a basic open question.

The relevance of Oseledec's theorem in the study of these approximation algorithms was first pointed out by Kosygin ([K]) and then applied by several authors (in particular [Ba1, Ba2], [L], and [Br2] who gives a clear overview).

Referring the reader e.g. to Section 3 in [L] for a statement of Oseledec's theorem and a definition of the Lyapunov exponents λ_i, we now state our main theoretical result:

Theorem 1 (Approximation exponents and Lyapunov exponents). *Let $\mathcal{A}$ be the two-dimensional continued fraction algorithm defined by the maps T and A in Section 2.1. There is a unique absolutely continuous T-invariant probability measure μ on Δ_2. This measure is ergodic and Oseledec's theorem holds for (T, A, μ), yielding Lyapunov exponents $\lambda_1 > \lambda_2 \geq \lambda_3$. The maximal exponent λ_1 is strictly positive and the corresponding space has dimension one. There are constants $c(\mathcal{A}), c^*(\mathcal{A})$ such that for Lebesgue-almost every $\theta \in [0,1]^2$*

$$\eta_{\mathcal{A}}(\theta) = c(\mathcal{A}) \geq 1 - \frac{\lambda_2}{\lambda_1}, \quad \eta^*_{\mathcal{A}}(\theta) = c^*(\mathcal{A}) = 1 - \frac{\lambda_2}{\lambda_1}. \tag{2.14}$$

The numerical experiments described in Section 5 seem to indicate that $\lambda_1(\mathcal{A}) \sim 2.329$, $\lambda_2(\mathcal{A}) \sim -0.722$ ($\lambda_3 = -\lambda_1 - \lambda_2$ since all matrices A have determinant 1), so that $c^*_{\mathcal{A}} \sim 1.31$. It would be interesting to prove rigorously that $\lambda_2(\mathcal{A}) < 0$.

To prove Theorem 1, we shall use a slight modification of results of Lagarias [L, Theorem 4.1, Theorem 4.2] which gives conditions under which Oseledec's theorem can be applied to control convergence exponents. Since some of these conditions are interesting in themselves, we state them now:

Lemma 2 (Semi-weak convergence of $\mathcal{A}$). *Let $w_j^{(n)}(\theta)$ be the columns of the matrices $L^{(n)}(\alpha(\theta))$ produced by $\mathcal{A}$. There are $c_0 > 1$ and $n_0 \geq 1$ so that for all $\theta \in [0,1]^2$ we have*

$$\max_{1 \leq j \leq 3} \|\theta - \tilde{w}_j^{(n)}(\theta)\| \leq c_0^{-n}, \quad \forall n \geq n_0. \tag{2.15}$$

The semi-weak convergence property (2.15) is defined in [L] and is stronger than the weak convergence property in our introduction. If it holds for an algorithm $\mathcal{D}$ and a set of values of θ of Lebesgue measure one, we say that algorithm $\mathcal{D}$ is *semi-weakly convergent.*

Recall also that an algorithm described by matrices $L^{(n)}$ is called *strongly convergent* if all vectors given by the columns of $L^{(n)}(\alpha(\theta))$ become arbitrarily close to the ray $\lambda\Theta$ as $n \to \infty$. The strong convergence of algorithm $\mathcal{A}$ (maybe only for a set of θ values of Lebesgue measure one) is an open problem.

Lemma 3 (Absolutely continuous ergodic probability for T). *The auxiliary map $T : \Delta \to \Delta$ associated with algorithm $\mathcal{A}$ admits an ergodic absolutely continuous* *invariant probability measure μ with a continuous density which is bounded away from zero.*

The proof of Lemma 3 uses results of Mayer [M] (which yield in fact stronger properties for μ, in particular mixing) and is an application of transfer operator techniques for locally expanding maps, using an infinite-dimensional version of the projective methods applied in the proof of Lemma 2.

2.3 Generalizations and variants: a tree of algorithms.

Before we mention higher-dimensional generalizations, we would like to point out that instead of assuming that $\alpha_i \leq 1/2$ and considering partitions of the squares into four subsquares, it would also be possible to consider the full simplex $\bar{\Delta}$ of points $0 \leq \alpha_1 \leq \alpha_2 \leq 1$ and divide the squares into two simplices along the anti-diagonal. The numerical experiments described in Section 5 (cf. Table 2) seem to indicate that the subsquare procedure is preferable.

At first sight the generalization of algorithm $\mathcal{A}$ to arbitrary dimension d seems quite straightforward: consider the $\lambda\Theta$ ray in $\mathbb{R}^{d+1}$ and the d-dimensional hyperplanes P_i, $i = 1, \ldots, d+1$, and divide the d-dimensional hypercube into 2^d sub-hypercubes. However, there is a small complication as pointed out in [BRTT]. Indeed, if $d \geq 3$, the union of all the d-dimensional simplices generated by d edges of a d-dimensional cube having a common vertex does not cover the cube. (Points in the cube whose coordinates $(x_1, \ldots x_d)$ in any basis formed by d such edges satisfy $\sum_i x_i > 1$ — such as $(1/2, 1/2, \ldots, 1/2)$ — will never be covered.) One way out is to perform a basis change as described in Remark 2.1 of [BRTT]: this corresponds to selecting a subdivision of the d-dimensional cube into simplices whose vertices are vertices of the cube, and then choosing the new basis by taking vectors whose endpoints are given by the vertices of the simplex containing the intersection point of our ray with the hyperplane. We call this the modification *of the first type* and note that it yields again a Markovian linear simplex-splitting algorithm (see Figure 3). Another solution, described in Remark 2.5 of [BRTT], is to consider, for each hyperplane P_i, the first intersection of the positive ray $\lambda\Theta$ which *does* belong to a suitable simplex (such an intersection always exist by Kronecker's theorem). This algorithm (which we will say is obtained by a *modification of the second type*) does not seem to be very tractable analytically, and has the annoying property that the corresponding matrices in general do not have determinant equal to one.

Observe finally that for both types of modification, one may (or may not) decide to use the subcube selection algorithm.

Figure 3 Decomposition of the 3-dimensional hypercube into 6 simplices (two of which are not visible)

We now explain how algorithm $\mathcal{A}$ can be viewed as a path in a $d+1$-ary tree of algorithms in any dimension d. Recall that a generic ray intersects the $d+1$ hyperplanes successively in $d+1$ distinct points (where we use perhaps the modification of the second type of the notion of first intersection with a hyperplane) and that we choose to construct the n-th approximation basis by considering the last such intersection. At each step of the algorithm, we could have used some other selection rule to choose among the $d+1$ hyperplanes. (This idea is similar, but not identical, to the branching algorithm studied by Pipping, see [B] for references.) In this paper, we restrict ourselves to Markovian choices, namely we use a selection process (which may depend on the current value of the point in Δ_d) which is *the same at each iteration* (i.e., does not depend on the "past"). We will also consider (Markovian) *random* algorithms, where we establish a list of N selection rules and choose at each iteration the rule to apply using a probability vector $p_i > 0$, with $\sum_{i=1}^{N} p_i = 1$.

Note that the one-dimensional reduction of our algorithm $\mathcal{A}$ is essentially the classical Gauss (division) continued fraction algorithm, whereas the one-dimensional path corresponding to choosing the first intersection at each step would be related to the Farey (subtraction) algorithm.

Observe also that in dimension $d = 2$, the algorithm corresponding to selecting the first intersected hyperplane is similar, but not at all identical with the Poincaré algorithm described e.g. in [N]. Indeed, the Poincaré algorithm always uses the same partition of the two-dimensional cube into six simplices. This leads to convergence problems and undesirable ergodic properties thoroughly analysed in [N]. The numerical results in Section 5 indicate that our simplex selection procedure (see Figure 2) suppresses this difficulty.

2.4 Renormalization.

It remains to discuss the relationship between the tree of algorithms described above and the higher-dimensional renormalizations introduced in [BRTT]. The renormalizations there simply correspond to the map $T(\alpha_1, \alpha_2)$ described in (2.3-2.4, 2.8). More precisely, [BRTT] associated to a torus rotation of angle (α_1, α_2) a partition of the torus in subdomains on which the first return map was again a torus rotation. The precise claim is that the choice of a subdomain there corresponds to the choice of a hyperplane P_i here, and the new rotation angle is given by the associated map $T(\alpha_1, \alpha_2)$.

This analogy also holds in higher dimensions. Since [BRTT] were not concerned by approximations, and in particular did not need to preserve the property that the original vector lied in the positive quadrant of each basis, the fact that the d-dimensional simplices described above do not cover the d-dimensional cube when $d \geq 3$ was not as problematic there as it is here.

We refer to [KO] for a discussion of a related continued fractions algorithm also connected with dynamical renormalization.

3. Proof of the semi-weak convergence of algorithm $\mathcal{A}$

In this section, we use projective metric techniques in finite dimension to show Lemma 2 for Algorithm $\mathcal{A}$. We also explain how to adapt the argument to make it work for some variants of algorithm $\mathcal{A}$ described in Section 2.

The version of the projective results for linear transformations of positive cones that we shall use may be found in [F]. We first recall that the *projective distance* $\Gamma(x, y) =$

$\Gamma_V(x, y)$ between two vectors x, y in the positive cone $V = V_{d+1} = \{(v_1, \ldots, v_d) \mid v_i \geq 0\}$ of $\mathbb{R}^{d+1}$ $(d \geq 1)$ can be defined by the formula

$$\Gamma(x, y) = \begin{cases} 0 & \text{if } x_i y_i = 0 \text{ and } x_i + y_i > 0 \text{ for some } i, \\ \inf_{x_i, y_j \neq 0}(x_j y_i)/(x_i y_j) & \text{otherwise.} \end{cases} \tag{3.1}$$

Recall also that the *V-projective bound* $\delta(A) = \delta_V(A)$ of a $(d+1) \times (d+1)$ matrix A with nonnegative real coefficients a_{ij} (which is a measure of the projective diameter of the image of the positive cone, as we shall see) is given by

$$\delta(A) = \begin{cases} 0 & \text{if } a_{ir} a_{js} = 0 \text{ and } a_{ir} + a_{js} > 0, \\ & \qquad \text{for some } i = j \text{ and } r, s, \\ & \qquad \text{or } r = s \text{ and } i, j, \\ \min_{i,j,r,s}(a_{ir} a_{js})/(a_{is} a_{jr}) & \text{otherwise.} \end{cases} \tag{3.2}$$

(In other words, $\delta(A) = \inf_{x, y \in V_{d+1}} \Gamma(Ax, Ay)$.) We shall consider the projective space $\mathcal{P} = \mathcal{P}_V$ obtained by identifying two points $x, y \in V$ such that $x = \lambda y$ for some $\lambda > 0$ (we then write $x \sim y$), denoting $\bar{x} \in \mathcal{P}$ for the equivalence class of $x \in V$.

Lemma 4 (Projective metrics [F, Lemmas 15.1, 15.2]).

(1) *The function $D_V(\bar{x}, \bar{y}) = -\log \Gamma(x, y)$ where x and y are arbitrary representatives of $\bar{x}, \bar{y} \in \mathcal{P}$ is a metric on $\mathcal{P}_V$.*

(2) *The action induced on $\mathcal{P}_V$ by a $(d+1) \times (d+1)$ matrix A with nonnegative coefficients is a contraction for the metric D_V. More precisely, we have*

$$\Gamma_V(Ax, Ay) \geq \frac{\Gamma_V(x, y) + \delta_V(A)}{\delta_V(A)\Gamma_V(x, y) + 1} \geq \Gamma_V(x, y)\,. \tag{3.3}$$

(The more general formulation given in [F] allows replacing V by another cone W – see the proof of Lemma 3.)

Corollary 5. *The transformation of $\mathcal{P}$ induced by a nonnegative matrix A with projective bound $\delta(A) > 0$ is a strict contraction for the metric D by a factor*

$$\epsilon \leq (1 - \delta(A))/(1 + \delta(A)) < 1\,.$$

Proof of Corollary 5. It suffices to check that for any $\epsilon < 1$ with $\epsilon > (1 - \delta)/(1 + \delta)$ we have $\Gamma(Ax, Ay) \geq \Gamma(x, y)^\epsilon$. This follows from from (3.3) by a concavity argument using the auxiliary functions $f(\Gamma) = (\Gamma + \delta)/(\delta\Gamma + 1)$ and $g(\Gamma) = \Gamma^\epsilon$ on $[0, 1]$. $\square$

In order to prove Lemma 3, we need a comparison lemma between the projective distance D and the distance $D_1(u, v) = \sum_{i=1}^{d} |u_i - v_i|$ on the simplex $\bar{\Delta}_d = \{u \in \mathbb{R}^{d+1} \mid u_i \geq 0\,, \sum_{i=1}^{d+1} u_i = 1\}$:

Lemma 6 (Comparison between distances). *For any $d \geq 1$ and $u, v \in \bar{\Delta}_d$:*

$$D_1(u,v) \leq d \cdot D(\bar{u}, \bar{v})\,. \tag{3.4}$$

The proof is probably standard, we include it for completeness:

Proof of Lemma 6. If $D(\bar{u}, \bar{v}) = \infty$, there is nothing to prove. Otherwise there are indices $k(u,v)$ and $\ell(u,v)$ so that

$$\begin{aligned} D(\bar{u}, \bar{v}) &= \log v_\ell - \log u_\ell + \log u_k - \log v_k \\ &= |\log v_\ell - \log u_\ell| + |\log u_k - \log v_k| \\ &\geq \max_{1\leq i\leq d+1} |\log u_i - \log v_i| \\ &\geq \max_{1\leq i\leq d} |\log u_i - \log v_i|\,. \end{aligned} \tag{3.5}$$

Therefore, since $t - s \leq \log t - \log s$ for any $0 < s \leq t \leq 1$, we have

$$D_1(u,v) \leq \sum_{i=1}^{d} |\log u_i - \log v_i| \leq d \cdot D(\bar{u}, \bar{v})\,. \quad \square \tag{3.6}$$

We now prove Lemma 2 from Section 2:

Proof of Lemma 2. The basic remark is that, except in the degenerate cases, all matrices $A(\alpha_1, \alpha_2)$ appearing in algorithm $\mathcal{A}$ have a projective bound satisfying

$$\delta(A(\alpha_1, \alpha_2)) \geq 1/4\,. \tag{3.7}$$

(The degenerate cases, which only concern a set of Lebesgue measure zero of values of θ, may be dealt with by hand.) Indeed, it is obvious from the definitions that

$$\delta(A(\alpha_1, \alpha_2)) = \frac{j}{j+1} \cdot \frac{k}{k+1} \geq \frac{1}{4}\,. \tag{3.8}$$

(We used that $k \geq 1$ and $j \geq 1$.) Corollary 5 thus implies that the transformation induced by each $A(\alpha_1, \alpha_2)$ is a contraction by a factor $\epsilon \leq 3/5$.

Therefore, writing e_1, e_2, e_3 for the standard basis in $\mathbb{R}^3$, and using $\bar{u}$ to denote also the representative of $u \in V$ in Δ_d, we get by applying Corollary 5 and Lemma 6 that for any $n \geq 1$, and θ

$$\begin{aligned} \frac{1}{\sqrt{2}} \max_{1\leq j\leq 3} \|\theta - \tilde{w}_j^{(n)}(\theta)\| &\leq \frac{1}{\sqrt{2}} \max_{1\leq j\leq 3} \|\theta - \tilde{w}_j^{(n)}(\theta)\|_1 \\ &\leq \max_{1\leq j\leq 3} D_1(\overline{\Theta(\theta)}, \overline{L^{(n)}(\alpha(\theta))e_j}) \\ &\leq \max_{1\leq i<j\leq 3} D_1(\overline{L^{(n)}(\alpha(\theta))e_i}, \overline{L^{(n)}(\alpha(\theta))e_j}) \\ &\leq 2 \max_{1\leq i<j\leq 3} D(\overline{L^{(n)}(\alpha(\theta))e_i}, \overline{L^{(n)}(\alpha(\theta))e_j}) \\ &\leq 2\,(3/5)^{n-1} \max_{1\leq i<j\leq 3\,,\alpha\in\Delta_2} D(\overline{A(\alpha)e_i}, \overline{A(\alpha)e_j}) \\ &\leq \frac{1}{2}\,(3/5)^{n-1}\,. \end{aligned} \tag{3.9}$$

(We have used that the ray $\lambda\Theta$ is always in the positive quadrant of the basis $L^{(n)}(\alpha)(e_i)$, $i = 1, 2, 3$, and also the fact that the maximum distance between any vertex of a simplex and an interior point of the simplex is bounded by the maximum distance between two vertices.) $\square$

Remark 3.1. Exactly the same arguments as those used in the proof of Lemma 2 show that this lemma holds for the higher dimensional version of algorithm $\mathcal{A}$ obtained by the modifications of the first and second type. In particular, these algorithms have the semi-weak convergence property. (Indeed, the corresponding matrices for nondegenerate points are all projectively bounded by $\delta \geq 1/4$ in any dimension.)

Remark 3.2. In dimension $d = 2$ or higher, one may consider other paths in the tree of algorithm described in Section 2. Since the cases $k = 0$ and/or $j = 0$ can then occur, it is necessary to work with a cone strictly larger than the positive quadrant V_{d+1} (and use the corresponding generalizations of Γ, δ, and Lemmas 3 and 4 which are to be found in Lemma 15.1, 15.2 of [F]) in order to get strictly positive projective bounds $\delta(A)$ for the matrices A which appear. We now explain what can be done.

In dimension $d = 2$, we have verified by constructing explicitly such suitable cones that for *any* Θ the transformations induced by the matrices A are strict contractions with a uniform coefficient $\delta(A) \geq \delta > 0$, for all paths (i.e. selection rules) which avoid the choice "intersection with P_3" whenever this intersection is the first one on our ray. Let us call this choice the *least efficient* one. Therefore all selection rules which can be proved to avoid the least efficient choice infinitely many often (for all or almost all θ) yield a weakly convergent algorithm, whereas all paths which avoid this choice for a given (uniform) proportion of iterations (for all or almost all θ) yield a semi-weakly convergent algorithm. In particular, any random path which chooses the first intersected plane with probability strictly below 1 or which avoids the intersection with P_3 with probability 1 will be semi-weakly convergent. (A random path is called semi-weakly convergent if (2.15) for the random approximation $(w_j^{(n)})_\omega$ associated to a given sequence of selection rules $(\omega_0, \omega_1, \dots)$ holds with probability 1.)

We *conjecture* (see the experimental evidence presented in Section 5) that *any* deterministic or random Markov path in the tree of two-dimensional algorithms is semi-weakly convergent (for almost each vector θ).

We also expect the (almost everywhere) semi-weak convergence to hold in any dimension $d \geq 3$ for both modifications of our tree of algorithms.

4. Ergodic properties and Lyapunov exponents

The main goal of this section is to prove Lemma 3 for the map T arising in the two-dimensional algorithm $\mathcal{A}$. Using also Lemma 2 which was proved in the previous section, we shall then apply the result from [L] on Lyapunov exponents. As in Section 3, we also mention several more or less straightforward generalizations to other paths in dimensions two and higher.

We shall use the following result of Mayer:

Theorem 7 (Main Theorem in [M]). *Let I^d be the d-dimensional unit cube $\{x \in \mathbb{R}^d \mid 0 \leq x_j \leq 1, 1 \leq j \leq d\}$ endowed with Lebesgue measure, and let $T : I^d \to I^d$ be an almost everywhere defined map such that there is a finite or countable (measure-theoretical) partition $I^d = \cup_{i\in\mathcal{I}} O_i$ into disjoint open sets with the measure of $\cup_i \partial O_i$ equal to zero and such that:*

(1) *T maps O_i bijectively to the interior of I^d for all i; each inverse branch $\psi_i : I^d \to O_i$ is real analytic and may be extended to a holomorphic open domain $I^d \subset \Omega$ of $\mathbb{C}^d$ such that every $\psi_i(\overline{\Omega})$ is a strict subset of Ω. Moreover, on some open domain Ω' with $\overline{\Omega} \subset \Omega'$ we have $det\psi_i'(z) \neq 0$ and $\sum_{i\in\mathcal{I}} |det\psi_i'(z)| < \infty$.*
(2) *The set $Per(T) = \{x \in I^d \mid \exists\, n, T^n x = x\}$ is dense in I^d.*

Then there is a T-invariant absolutely continuous probability measure μ with an analytic density f. The measure μ is ergodic (in fact, mixing) for T and has full support.

Mayer in fact proves much more in [M] and requires weaker assumptions, in particular, he applies his result to the Jacobi-Perron algorithm (see also [Br1]).

Proof of Lemma 3. One may obviously replace I^d by a simplex of the form $\Delta_d = \{x \in \mathbb{R}^d \mid 0 \leq x_1 \leq x_2 \ldots \leq x_d \leq 1/2\}$ in the statement of Theorem 7. Therefore, we only need to verify that assumptions (1-2) in that theorem are satisfied by our map $T : \Delta_2 \to \Delta_2$.

Clearly, there is a countable partition of Δ_2 into open two-dimensional simplices O_i, for $i \in \mathcal{I} \subset \{(j,k,\ell,m) \mid j \geq 1, k \geq 1, 1 \leq \ell \leq 3, 1 \leq m \leq 4\}$ with the property that T maps O_i bijectively to the interior of Δ_2. (The index ℓ corresponds to the number of the last intersected hyperplane, m tells us which subsquare is selected, and k and j are as in (2.5, 2.9). Note that the degenerate cases correspond to an edge of Δ_2 and may thus be neglected.) Also, it follows from the formulas in Section 2 that T and its local inverses ψ_i are projective transformations. In particular, the ψ_i are analytic maps on Δ_2 which admit a holomorphic extension to the entire complex plane.

The key remark now is that the map ψ_i associated to a quadruple $i = (j,k,l,m)$ is nothing but the projective map induced by the corresponding matrix A. To check that these two maps coincide, observe that it suffices to verify that they send the vertices of the simplex Δ_2 to the same three points by projectivity. (Note that this is the linear simplex-splitting property.) Since these matrices are uniformly projectively bounded by (3.7), it will not be difficult to see that there is a domain Ω as in assumption (1) by applying Corollary 5.

Indeed, we only need to show that the euclidean or D_1 metric on Δ_2 is equivalent to a suitable projective metric. We may use Lemma 6 to bound the metric on the simplex by the projective metric D_V. However, since the projective metric D_V diverges on the boundary of the positive cone V it cannot be controlled by a bounded metric such as D_1. To obtain the reverse bound, we thus need to find a strictly larger cone $V \subset V'$ with the property that the associated projective bound $\delta_{V'}(A)$ is bounded away from zero for all matrices A appearing in our algorithm. (We are using here the projective distance $\Gamma_W(x,y)$ between two vectors x, y in a cone W, given by $\Gamma_W(x,y) = K_W(x,y)K_W(y,x)$ where $K_W(x,y) = \sup\{\lambda \in \mathbb{R} \mid x - \lambda y \in W\}$, to define the projective bound $\delta_W(A)$

of a matrix A such that that $AW \subset W$ by setting $\delta_W(A) = \inf_{x,y \in W} \Gamma_W(Ax, Ay)$.) We then wish to apply the general version of Lemma 4 (Lemmas 15.1, 15.2 from [F]). To prove that a cone V' with the above property exists, we first use the fact that $\Gamma_W(r,s) \geq \Gamma_V(r,s)$ for any $r, s \in V \subset W$ (in particular $r = Ax$ and $s = Ay$ with $x, y \in V$), and then apply continuity in r, s of $\Gamma_W(r,s)$. Since the restriction to V of the projective distance on V' (or any cone W whose interior contains the closure of V: just note that the continuous function $\Gamma_W(x,y)$ does not vanish on the boundary of V) is bounded above by a constant multiple of the distance D_1 on Δ_2, we get the desired reverse bound. (Note that the bound from Lemma 6 still holds for V'.)

To check the summability condition on the determinants we need to quantify the contraction rate for areas. One way to do this is to note that the inverse branch $\psi_{j,k,l,m}(\alpha_1, \alpha_2) = (\alpha_1', \alpha_2')$ has the following form when $\ell = 1$ (i.e., when the last intersection is with P_1) and $m = 1$ (i.e., we are in the subsquare which does not require any change of coordinates)

$$\begin{cases} \alpha_1' &= \frac{1}{\alpha_1+\alpha_2+j+k+1} \\ \alpha_2' &= \frac{\alpha_1+k}{\alpha_1+\alpha_2+j+k+1}, \end{cases} \tag{4.1}$$

(the formula for $\ell = 2$ and $m = 1$ is symmetric) and the following form when $\ell = 3$ and $m = 1$

$$\begin{cases} \alpha_1' &= \frac{\alpha_1+k}{\alpha_1+\alpha_2+j+k+1} \\ \alpha_2' &= \frac{\alpha_2+j}{\alpha_1+\alpha_2+j+k+1}. \end{cases} \tag{4.2}$$

Explicit computation of the partial derivatives yields a constant C such that

$$|\det \psi_{j,k,\ell,1}'(\alpha_1, \alpha_2)| \leq C \frac{1}{(k+j)^3} \tag{4.3}$$

for $\ell = 1$ or $\ell = 3$, all j, k, α_1, and α_2. By symmetry, (4.3) also holds for $\ell = 2$, and since the additional changes of coordinates associated with the cases $m = 2, 3, 4$ are isometries (they are suitable compositions of the maps $(x,y) \mapsto (y,x)$ and $(x,y) \mapsto (1-x, y)$), the same bound holds also there. The condition $\sup_{\alpha_1,\alpha_2} \sum_{j,k,l,m} |\det \psi_{j,k,l,m}'(\alpha_1, \alpha_2)| < \infty$ then follows from (4.3).

It remains to prove that periodic points of T are dense in Δ_2. But this is an obvious consequence of the fact that each injective branch of T uniformly expands a simplex to the full domain of definition Δ_2 so that each branch of T^n expands a simplex of exponentially shrinking diameter to Δ_2 (then apply the Brouwer fixed point theorem to T^n). $\square$

Proof of Theorem 1. In order to apply Theorem 4.1 and Theorem 4.2 in [L] we only need to check that properties (H1)-(H5) from [L, Section 4] are satisfied. (Note that the formulation there uses maps on the unit square but that one may again translate this into our simplicial setting in a straightforward manner.)

Property (H1) says that T must have an ergodic absolutely continuous invariant measure $d\mu$ with support Ω a T-invariant Borel subset of positive Lebesgue measure.

This is clearly a consequence of our Lemma 3. Since Ω is in our case the full simplex Δ_2, property (H2) in [L] reduces to the requirement that T is piecewise continuous with almost everywhere nonvanishing Jacobian, and this is true too. Property (H3) is the semi-weak convergence of the algorithm which we proved in Lemma 2, and (H5) is a "partial quotient mixing" assumption which trivially holds in our case because all of our matrices are strictly positive.

It remains to check the boundedness condition (H4) which reads

$$E\big[\log(\max\|A\|,1)\big]=\int_\Delta \log(\max(1,\|A(\alpha_1,\alpha_2)\|)\,d\mu(\alpha_1,\alpha_2)<\infty\,. \tag{4.4}$$

Since μ has a continuous density we may replace it by Lebesgue measure in (4.3). To obtain the bound, we may use any matrix norm for example the l^1 norm for which we have

$$\|A(\alpha_1,\alpha_2)\|_1\le\max(j(\alpha_1,\alpha_2),k(\alpha_1,\alpha_2))+1\,. \tag{4.5}$$

It is easy to check that the sets (corresponding to "last intersection with P_1")

$$\left\{\begin{aligned} R^1_{k_0} &= \{\alpha_1\mid(\alpha_1,\alpha_2)\in\Delta_2\,,k(\alpha_1,\alpha_2)=[\alpha_2/\alpha_1]=k_0\}\,,\\ S^1_{j_0} &= \{\alpha_1\mid(\alpha_1,\alpha_2)\in\Delta_2\,,j(\alpha_1,\alpha_2)=[(1-\alpha_2)/\alpha_1]=j_0\}\end{aligned}\right. \tag{4.6}$$

for $j_0,k_0\ge1$ have Lebesgue measure bounded by $m(R^1_{k_0})\le C/k_0^2$, $m(S^1_{j_0})\le C/j_0^2$ where C is some uniform constant. We obtain the same bounds for the sets $R^2_{k_0}$, $S^2_{j_0}$, $R^3_{k_0}$, $S^3_{j_0}$ associated with the intersection with P_2 or P_3 (recall (2.8)).

Therefore, using (4.5), we find

$$E\big[\log(\max\|A\|,1)\big]\le\sum_{j=1}^{\infty}C\frac{\log(j+1)}{j^2}+\sum_{k=1}^{\infty}C\frac{\log(k+1)}{k^2}<\infty \tag{4.7}$$

as desired. $\square$

Remark 4.1. The same arguments may be used to show that Lemma 3 and Theorem 1 hold for the higher-dimensional version of algorithm $\mathcal{A}$ obtained by the modification of the first type.

Remark 4.2. Consider the other paths in the tree of algorithms from Section 2.

In dimension $d=2$, we may use Remark 3.2 to adapt the proofs of Lemma 3 and Theorem 1 for all paths in the tree which avoid the least efficient choice infinitely often (at least for almost all θ) and which have the partial quotient mixing property (H5).

It is also possible, although more cumbersome, to consider random paths. We sketch now how this can be done. The version of Lemma 3 relevant in the random case involves the invariant measure for the Markov chain associated to the random map. (See [K1, K2] for general information on random dynamical systems and their ergodic properties.) The definitions (2.12)-(2.13) can be modified to define random approximation exponents by taking the essential supremum (respectively infimum) over all random sequences of choices $(\omega_0, \omega_1, \dots)$. Using the standard notion of Lyapunov exponents for random dynamical systems (see e.g. [ACE]), the results and arguments of Lagarias may be adapted to the random case in a straightforward manner. In particular (2.14) would have the same form where the Lyapunov exponents are replaced by the almost everywhere constant value of the random Lyapunov exponents. The final result is that the random analogue of Theorem 1 holds for any random path which chooses the first intersected plane with probability strictly below 1, or which avoids the intersection with P_3 with strictly positive probability whenever the partial mixing condition (H5) from [L] holds.

We *conjecture* (see also the numerical data in the next section) that *any* deterministic or random Markov path in the tree of two-dimensional algorithms satisfies the (if necessary, randomized) conclusions of Lemma 3 and Theorem 1.

We also expect similar results to hold in any higher dimension $d \geq 3$ for both modifications of our tree of algorithms.

5. Numerical results

In this section, we present our numerical study of the Lyapunov exponents of algorithm $\mathcal{A}$ and three other two-dimensional algorithms constructed from our tree (Table 1). We include also the numerical results for the Jacobi-Perron and ordered Jacobi-Perron algorithms (they are consistent with Baldwin's results from [Ba2]), and we reproduce Baldwin's results on his GFCP algorithm for comparison. (See Broise [Br2] for rigorous results on the Lyapunov exponents of the Jacobi-Perron algorithm.) In Table 2 we present the data obtained by using the variant of algorithm $\mathcal{A}$ and the three other algorithms when the subsquare selection is replaced by a triangular selection process as described in the first paragraph of Section 2.3.

The three other two-dimensional algorithms correspond respectively to the choice of the second intersected hyperplane (algorithm $\mathcal{B}$), the first intersected hyperplane (algorithm $\mathcal{C}$) and a random algorithm where we choose the j^{th} intersected hyperplane with probability 1/3 for $1 \leq j \leq 3$ at each step (algorithm $\mathcal{R}$). Although we have written the proofs in detail only for algorithm $\mathcal{A}$, Remarks 3.1–3.2, and 4.1–4.2 can be used to show that Theorem 1 also applies to algorithms $\mathcal{B}$ and $\mathcal{R}$. The case of algorithm $\mathcal{C}$ is not at all clear from a mathematical point of view, but the numerical behaviour is quite satisfactory.

Before discussing the values of the exponents in Table 1, we briefly explain how our programs work. (They have been written in the C language, using double precision, any request for the source code should be sent to the first named author.)

We start from some "arbitrary" vector (α_1, α_2), initialize Q to be the 3×3 identity matrix, and ℓ to be the vector $(0, 0, 0)$, and repeat a great number of times (one million iterates) the following procedure (inspired from the proof in [JPS] and suggested to us by S. Oliffson Kamphorst):

- `let` A `be the matrix associated to` (α_1, α_2) `by the algorithm;`
- `let` $(\alpha_1, \alpha_2) := T(\alpha_1, \alpha_2)$`;`
- `let` $B := AQ$`;`
- `decompose` B `as` $B = QR$ `where` Q `is unitary and` R `upper-triangular;`

- `let` $\ell(i) = \ell(i) + \log R(i,i)$ `for` $1 \leq i \leq 3$.

At the end of the process, $\ell(i)$ divided by the number of iterations contains an approximation to the i^{th} Lyapunov exponent.

Recall that $\lambda_1 > 0$ controls the exponential growth of denominators in the approximations (see [L, (4.22)]) and that $-(\lambda_1 - \lambda_2) < 0$ controls the exponential decay of $\|\theta - \tilde{w}_i^{(n)}(\theta)\|$ (see [L, (4.18), last formula of p. 317]). A high value for $\lambda_1 - \lambda_2$ seems therefore to be a desirable feature. Another goal is that the uniform approximation exponent $c^*(\mathcal{A}) = 1 - \lambda_2/\lambda_1 \leq 1.5$ be as close as possible to 1.5. Keeping this in mind, we see that algorithm $\mathcal{A}$ compares very favorably to both versions of the Jacobi-Perron algorithm and to the GCFP algorithm of Baldwin in terms of control of the rate of convergence of $\|\theta - \tilde{w}_i^{(n)}(\theta)\|$. It is slightly better than the ordered Jacobi-Perron algorithm and slightly worse than the classical Jacobi-Perron algorithm as far as the uniform approximation exponent c^* is considered. The GCFP algorithm produces a better convergence exponent than algorithm $\mathcal{A}$.

Considering now the other variants, we see that the best uniform approximation exponent of all algorithms (including the GCFP one) is obtained by algorithm $\mathcal{B}$ (we find $c^*(\mathcal{B}) \sim 1.396$), and the best convergence rate by far is given by algorithm $\mathcal{A}$ ($\lambda_1 - \lambda_2 \sim 3.05$). When both the approximation exponent and the convergence rate are taken into account, algorithm $\mathcal{B}$ does quite well.

Algorithm	λ_1	λ_2	$c^* = 1 - \lambda_2/\lambda_1$	$\lambda_1 - \lambda_2$
$\mathcal{A}$	**2.3293**	-0.7218	1.310	**3.0511**
$\mathcal{B}$	1.3091	-0.5190	**1.396**	1.8281
$\mathcal{C}$	0.5091	-0.1953	1.384	0.7044
$\mathcal{R}$	1.4334	-0.5080	1.354	1.9414
J-P	1.2006	-0.4486	1.374	1.6492
ordered J-P	1.6421	-0.4594	1.280	2.1015
GCFP (from [Ba2])	0.6249	-0.2418	1.387	0.8667

Table 1 (Recall $\lambda_3 = -\lambda_1 - \lambda_2$)

Table 2 shows that, although the process which cuts the squares along the antidiagonal (see Section 2.3) gives a very high value of λ_1 (and thus $\lambda_1 - \lambda_2$) for the modification $\mathcal{A}'$ of algorithm $\mathcal{A}$, the approximation exponents are on the whole not as good as those of the algorithms in Table 1.

Algorithm	λ_1	λ_2	$c^* = 1 - \lambda_2/\lambda_1$	$\lambda_1 - \lambda_2$
$\mathcal{A}'$	**2.5966**	-0.5548	1.214	**3.1514**
$\mathcal{B}'$	1.3931	-0.4318	1.310	1.8249
$\mathcal{C}'$	0.4217	-0.1627	**1.386**	0.5844
$\mathcal{R}'$	1.5191	-0.4080	1.269	1.9271

Table 2 (Division along antidiagonal instead of subsquare selection)

References

ACE. L. Arnold, H. Crauel, and J.-P. Eckmann (eds.), *Lyapunov exponents (Proceedings Oberwolfach 1990)*, Springer (Lecture Notes in Math. **1486**), Berlin, 1991.

BRTT. V. Baladi, D. Rockmore, N. Tongring, and C. Tresser, *Renormalization on the n-dimensional torus*, Nonlinearity **5** (1992), 1111-1136.

Ba1. P.R. Baldwin, *A multidimensional continued fraction and some of its statistical properties*, J. Statist. Phys. **66** (1992), 1463–1505.

Ba2. P.R. Baldwin, *A convergence exponent for multidimensional continued fractions*, J. Statist. Phys. **66** (1992), 1507–1526.

B. A. J. Brentjes, *Multi-dimensional continued fraction algorithms (Math. Centre Tract. No 145)*, Math. Centre, Amsterdam, 1981.

Br1. A. Broise, *Aspects stochastiques de certains systèmes dynamiques: transformations dilatantes de l'intervalle, fractions continues multidimensionnelles. Partie II: Fractions continues multidimensionnelles et lois stables*, PH.D. thesis, Université de Rennes, to appear Bull. Soc. Math. France (1994).

Br2. A. Broise, *Aspects stochastiques de certains systèmes dynamiques: transformations dilatantes de l'intervalle, fractions continues multidimensionnelles. Partie III et Annexe: Fractions continues multidimensionnelles et approximations diophantiennes*, PH.D. thesis, Université de Rennes (1994).

F. H. Furstenberg, *Stationary processes and prediction theory*, Princeton U. Press, Princeton, NJ, 1960.

JPS. R.A. Johnson, K.J. Palmer, and G.R. Sell, *Ergodic properties of linear dynamical systems*, SIAM J. Math. Anal. **18** (1987), 1–33.

K1. Y. Kifer, *Ergodic Theory of Random Perturbations*, Birkhäuser, Boston Basel, 1986.

K2. Y. Kifer, *Random Perturbations of Dynamical Systems*, Birkhäuser, Boston Basel, 1988.

KO. S. Kim and S. Ostlund, *Simultaneous rational approximations in the study of dynamical systems*, Phys. Rev. A **34** (1986), 3426–3434.

K. D. Kosygin, *Multidimensional KAM theory from the renormalization group viewpoint*, Advances Sov. Math. **3** (1991), 99–129.

L. J.C. Lagarias, *The quality of the Diophantine approximations found by the Jacobi-Perron algorithm and related algorithms*, Monatsh. Math. **115** (1993), 299-328.

M. D. Mayer, *Approach to equilibrium for locally expanding maps in $\mathbb{R}^k$*, Comm. Math. Phys. **95** (1984), 1–15.

N. A. Nogueira, *The 3-dimensional Poincaré continued fraction algorithm*, Israel J. Math. **90** (1995), 373–401.

V. Baladi: Mathématiques, Université de Genève, CH-1211 Geneva 24, SWITZERLAND (on leave from CNRS, UMR 128, ENS Lyon, France)
E-mail address: baladi@sc2a.unige.ch

A. Nogueira: Instituto de Matemática, Universidade Federal do Rio de Janeiro Caixa Postal 68.530, 21.945 Rio de Janeiro RJ, BRAZIL
E-mail address: nogueira@vms1.nce.ufrj.br