

# On the Singularization of the Two-Dimensional Jacobi–Perron Algorithm

B. Schratzberger

## CONTENTS

- 1. Introduction
- 2. Ingredients
- 3. The Process of Insertion and Singularization
- 4. Examples
- Acknowledgments
- References

---

We present a constructive method to convert the (two-dimensional) Jacobi–Perron evolution of  $(x_1, x_2) \in [0, 1]^2$  into the corresponding evolution of Podsypanin, and vice versa. A similar approach allows us to extend the result to Brun’s algorithm. The method, based on the techniques of singularization and insertion, is built up in steps. From experiments, we assume that each step terminates after finitely many states almost everywhere.

---

## 1. INTRODUCTION

In [Iosifescu and Kraaikamp 02], it is demonstrated in detail how a multitude of semiregular one-dimensional continued-fraction algorithms (the class of *S-expansions*) are related to one another (cf. also [Kraaikamp 91]). In particular, it is shown how such algorithms might be transformed into other algorithms of the same class by the techniques of *singularization* and *insertion*. Apart from the classification, this allows us to transfer certain statistical and approximation properties from one algorithm to another in an intuitive way.

Although this classification is now quite well understood in dimension one, little is known about similar relations in higher dimensions (cf. [Schratzberger 04]). In this paper, we will show that in two dimensions, the Podsypanin algorithm and the Jacobi–Perron algorithm can be linked by very similar techniques. We present a constructive process for converting one algorithm into the other by means of singularization and insertion. The method is built up in steps. From experiments, we assume that each step of this process terminates after a finite number of stages. Apart from [Schratzberger 04], this is to our knowledge the first time that this method has been applied in dimension two.

We now describe the fundamental definitions and ideas. Let  $X$  be some suitable subset of  $\mathbb{R}^2$ . Most generally, we consider algorithms  $T : \{1, 2\} \times \mathbb{N}_0 \times \mathbb{Z} \times X \rightarrow X$ ,

2000 AMS Subject Classification: Primary 11K50  
Keywords: Jacobi–Perron algorithm, Podsypanin algorithm, singularization, S-expansions

which can be described as follows:

$$T(j, A, C, x_1, x_2) := \begin{cases} \left( \frac{1}{x_2} - A, \frac{x_1}{x_2} - C \right) & \text{if } j = 1, \\ \left( \frac{x_2}{x_1} - C, \frac{1}{x_1} - A \right) & \text{if } j = 2. \end{cases}$$

The local inverse branches  $V : \{1, 2\} \times \mathbb{N}_0 \times \mathbb{Z} \times X \rightarrow X$ , again dependent on the *digits*  $j, A$ , and  $C$ , are given as

$$V(j, A, C, x_1, x_2) := \begin{cases} \left( \frac{x_2 + C}{x_1 + A}, \frac{1}{x_1 + A} \right) & \text{if } j = 1, \\ \left( \frac{1}{x_2 + A}, \frac{x_1 + C}{x_2 + A} \right) & \text{if } j = 2. \end{cases}$$

We refer to  $j \in \{1, 2\}$  as the *type*, and to  $A$  and  $C$  as the *partial quotients* of  $T$  and  $V$ , respectively. If the digits are uniquely specified, we will often write  $T(x_1, x_2)$  and  $V(x_1, x_2)$  instead of  $T(j, A, C, x_1, x_2)$  and  $V(j, A, C, x_1, x_2)$ .

An extensive account on piecewise fractional linear maps is given in [Schweiger 00]. We adopt the following definitions:

**Definition 1.1.** Let  $X$  be a set,  $T : X \rightarrow X$ . If there exists a partition  $\{X(i) : i \in I\}$  of  $X$ , where  $I$  is finite or countable, such that the restriction of  $T$  to  $X(i)$  is injective, then  $(X, T)$  is called a *fibred system*.

The partition  $\{X(i) : i \in I\}$  is called the *time-1 partition*.

**Definition 1.2.** A cylinder of rank  $t$  is the set

$$X(i_1, \dots, i_t) := \{x : i(x) = i_1, \dots, i(T^{t-1}(x)) = i_t\}.$$

A cylinder  $X(i_1, \dots, i_t)$  is called *full* if

$$T^t X(i_1, \dots, i_t) = X.$$

In particular, we are interested in two well-known special cases of  $T$ : The first example of a generalization of the regular one-dimensional continued-fraction algorithm was given by [Jacobi 68] in an (unsuccessful) attempt to generalize Lagrange’s theorem on quadratic irrational numbers and the eventual periodicity of their expansions. His algorithm was revived in [Perron 07]. For a general description of the Jacobi–Perron algorithm and its properties, see [Schweiger 00].

Other types of multidimensional continued-fraction algorithm were later proposed by several authors to generalize some of the classical properties of the regular one-dimensional algorithm. We consider an algorithm introduced by Podsypanin [Podsypanin 77] as a generalization of Brun’s algorithm (see also [Pustyl’nikov 03]).

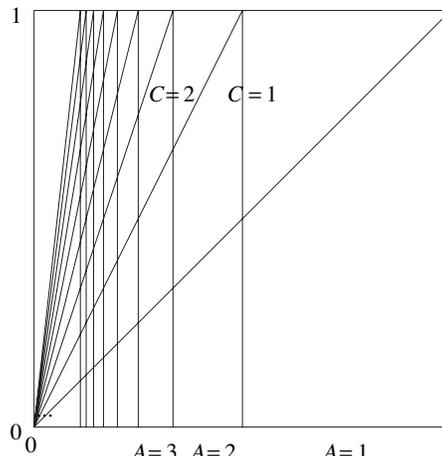


FIGURE 1. The time-1 partition of the JPA.

We give the definitions of the two algorithms in question. Here and in the following,  $[x]$  will denote the integer part of some  $x \in \mathbb{R}$ .

**Definition 1.3.** [Perron 07] The (two-dimensional) Jacobi–Perron algorithm (JPA) is generated by a map  $T_J : [0, 1]^2 \rightarrow [0, 1]^2$ ,

$$T_J(x_1, x_2) = \left( \frac{x_2}{x_1} - C, \frac{1}{x_1} - A \right),$$

where  $A := \left[ \frac{1}{x_1} \right]$  and  $C := \left[ \frac{x_2}{x_1} \right]$ .

Let  $t \geq 1$ , and define

$$A^{(t)} = A^{(t)}(x_1^{(0)}, x_2^{(0)}) := A(T_J^{t-1}(x_1^{(0)}, x_2^{(0)})),$$

$$C^{(t)} = C^{(t)}(x_1^{(0)}, x_2^{(0)}) := C(T_J^{t-1}(x_1^{(0)}, x_2^{(0)})),$$

while  $j^{(t)} := 2$ . The algorithm is ergodic and conservative, and it is known to admit an invariant measure. Not all cylinders  $X_J((A^{(1)}, C^{(1)}), \dots, (A^{(t)}, C^{(t)}))$  are full. In particular, the *conditions of Perron* state that whenever  $A^{(t)} = C^{(t)}$  for some  $t$ , then  $C^{(t+1)} > 0$ . See Figure 1.

The two-dimensional JPA is a projective version of a linear three-dimensional algorithm: Let  $M := \{(b_0, b_1, b_2) : b_0 \geq b_1 \geq 0, b_0 \geq b_2 \geq 0\}$ , and define  $\tau_J : M \rightarrow M$  by

$$\tau_J(b_0, b_1, b_2) = (b_1, b_2 - Cb_1, b_0 - Ab_1),$$

where  $A := \left[ \frac{b_0}{b_1} \right]$  and  $C := \left[ \frac{b_2}{b_1} \right]$ . Now using the projection map  $p : M \rightarrow [0, 1]^2$  defined by

$$p(b_0, b_1, b_2) = \left( \frac{b_1}{b_0}, \frac{b_2}{b_0} \right)$$

yields the two-dimensional JPA. The matrices  $\alpha_J^{(t)} := \alpha_J(A^{(t)}, C^{(t)})$  of  $\tau_J$  are given as

$$\alpha_J(A, C) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & -C & 1 \\ 1 & -A & 0 \end{pmatrix}.$$

The inverses  $\beta_J^{(t)} := \beta_J(A^{(t)}, C^{(t)})$  of the matrices of the algorithm,

$$\beta_J(A, C) = \begin{pmatrix} A & 0 & 1 \\ 1 & 0 & 0 \\ C & 1 & 0 \end{pmatrix},$$

produce a series of convergence matrices  $\{\Omega_J^{(t)}\}_{t=0}^\infty$  as follows:

$$\Omega_J^{(0)} = \begin{pmatrix} q^{(0)} & q^{(-2)} & q^{(-1)} \\ p_1^{(0)} & p_1^{(-2)} & p_1^{(-1)} \\ p_2^{(0)} & p_2^{(-2)} & p_2^{(-1)} \end{pmatrix} := E,$$

and

$$\Omega_J^{(t)} = \begin{pmatrix} q^{(t)} & q^{(t-2)} & q^{(t-1)} \\ p_1^{(t)} & p_1^{(t-2)} & p_1^{(t-1)} \\ p_2^{(t)} & p_2^{(t-2)} & p_2^{(t-1)} \end{pmatrix} := \Omega_J^{(t-1)} \beta_J^{(t)}.$$

The columns of the convergence matrices produce Diophantine approximations

$$\left( \frac{p_1^{(t)}}{q^{(t)}}, \frac{p_2^{(t)}}{q^{(t)}} \right)$$

to  $(x_1^{(0)}, x_2^{(0)})$ .

**Definition 1.4.** [Podsypanin 77] The (two-dimensional) Podsypanin algorithm (PA) is generated by a map  $T_P : [0, 1]^2 \rightarrow [0, 1]^2$ , where

$$T_P(x_1, x_2) = \begin{cases} \left( \frac{1}{x_2} - A, \frac{x_1}{x_2} \right) & \text{if } x_2 \geq x_1 \ (j = 1), \\ A := \left\lfloor \frac{1}{x_2} \right\rfloor, \\ \left( \frac{x_2}{x_1}, \frac{1}{x_1} - A \right) & \text{if } x_1 \geq x_2 \ (j = 2), \\ A := \left\lfloor \frac{1}{x_1} \right\rfloor. \end{cases}$$

As above,

$$j^{(t)} = j^{(t)}(x_1^{(0)}, x_2^{(0)}) := j(T_P^{t-1}(x_1^{(0)}, x_2^{(0)})),$$

$$A^{(t)} = A^{(t)}(x_1^{(0)}, x_2^{(0)}) := A(T_P^{t-1}(x_1^{(0)}, x_2^{(0)})),$$

and  $C^{(t)} := 0$ . All cylinders

$$X_P((j^{(1)}, A^{(1)}), \dots, (j^{(t)}, A^{(t)}))$$

are full, and the algorithm again is ergodic and conservative. The density of the invariant measure is explicitly known [Schweiger 78]. See Figure 2.

The inverses are given as  $\beta_P^{(t)} := \beta_P(j^{(t)}, A^{(t)})$ , where

$$\beta_P(1, A) = \begin{pmatrix} A & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

and

$$\beta_P(2, A) = \begin{pmatrix} A & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}.$$

The convergence matrices  $\Omega_P^{(t)}$  and the corresponding Diophantine approximations are defined similarly to the above. We may generalize the definitions of the matrices  $\beta_J(A, C)$  and  $\beta_P(j, A)$  to  $\beta(j, A, C)$ , where

$$\beta(1, A, C) = \begin{pmatrix} A & 1 & 0 \\ C & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

and

$$\beta(2, A, C) = \begin{pmatrix} A & 0 & 1 \\ 1 & 0 & 0 \\ C & 1 & 0 \end{pmatrix}.$$

Thus  $\beta_J(A, C) = \beta(2, A, C)$  and  $\beta_P(j, A) = \beta(j, A, 0)$ . Similar to the above,  $j$  is referred to as the type of the matrix  $\beta(j, A, C)$ .

Assume that we want to transform the JPA evolution of a given  $(x_1^{(0)}, x_2^{(0)}) \in [0, 1]^2$  into the corresponding PA evolution. Note that as long as  $x_2^{(t)} \leq x_1^{(t)}$ , the JPA equals the PA. Thus the method starts at  $(x_1^{(t)}, x_2^{(t)})$ , where  $t \geq 0$  is the smallest integer such that  $x_2^{(t)} > x_1^{(t)}$ . In particular, we need to determine the size of  $\left\lceil 1/x_2^{(t)} \right\rceil$ .

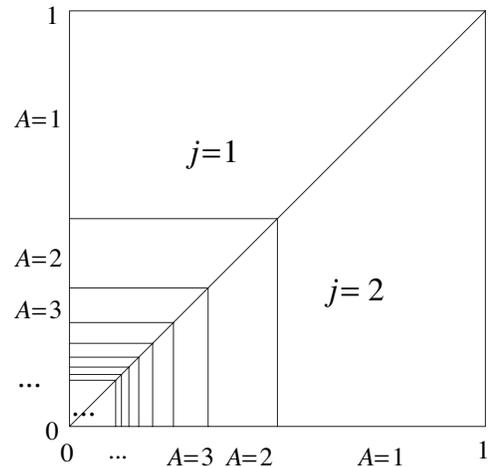


FIGURE 2. The time-1 partition of the PA.

Note that for  $i \geq 1$ , there are uniquely defined digits  $(A^{(t+1)}, C^{(t+1)}), \dots, (A^{(t+i)}, C^{(t+i)})$  such that

$$(x_1^{(t)}, x_2^{(t)}) \in X_J((A^{(t+1)}, C^{(t+1)}), \dots, (A^{(t+i)}, C^{(t+i)})).$$

Hence  $\lceil 1/x_2^{(t)} \rceil$  is dependent on the digits of the JPA evolution of  $(x_1^{(t)}, x_2^{(t)})$  and the size of

$$X_J\left(\left(A^{(t+1)}, C^{(t+1)}\right), \dots, \left(A^{(t+i)}, C^{(t+i)}\right)\right) \rightarrow 0$$

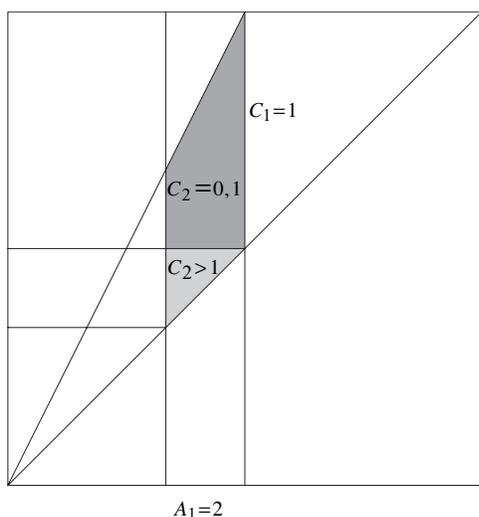
as  $i \rightarrow \infty$ ; in fact,  $(x_1^{(t)}, x_2^{(t)})$  can be defined as

$$\bigcap_{i=1}^{\infty} X_J\left(\left(A^{(t+1)}, C^{(t+1)}\right), \dots, \left(A^{(t+i)}, C^{(t+i)}\right)\right).$$

We try to derive the digits of the PA evolution from the digits of the JPA evolution.

**Example 1.5.** Consider some suitable  $(x_1, x_2) \in X_J((2, 1), (A_2, C_2))$ . If  $C_2 \leq 1$ , then  $\lceil \frac{1}{x_2} \rceil = 1$ , while if  $C_2 > 1$ , then  $\lceil \frac{1}{x_2} \rceil = 2$ . See Figure 3.

The basic idea of the method proposed in the following sections is that not only the first digits, but the entire PA evolution of a.e. such  $(x_1^{(t)}, x_2^{(t)})$  can be derived by manipulating the digits of the JPA evolution of  $(x_1^{(t)}, x_2^{(t)})$ . This will be done by operations on the matrices  $\beta(j^{(t+i)}, A^{(t+i)}, C^{(t+i)})$ ,  $i \geq 1$ . In particular, we generalize the techniques of singularization and insertion. A detailed discussion of the concept of S-expansions



**FIGURE 3.**  $(x_1, x_2) \in X_J((2, 1), (A_2, C_2))$ . In this case, the size of  $\lceil 1/x_2 \rceil$  only depends on the size of  $C_2$ .

in the one-dimensional case is given in [Iosifescu and Kraaikamp 02]. A first attempt at a generalization to two dimensions, applied to Brun’s algorithm, has been given in [Schratzberger 04].

The fundamental idea of singularization, as introduced in [Kraaikamp 91], was to improve approximation properties of the (one-dimensional) regular continued-fraction algorithm. In particular, Kraaikamp was interested in semiregular continued-fraction algorithms whose sequences of convergents  $\{p^{(t)}/q^{(t)}\}_{t=1}^{\infty}$  are subsequences of the sequence of regular convergents to  $x$ .

This was done by removing certain matrices  $\beta^{(t)}$  from the sequence of inverse matrices  $\{\beta^{(t)}\}_{t=0}^{\infty}$  of the regular continued-fraction algorithm (where the matrices  $\beta^{(t)}$  are defined in a spirit similar to that of the above). The process was defined by a matrix identity and a *law of singularization*, which in an unambiguous way determined the matrices to be singularized. We generalize this definition of singularization. For more details on the two-dimensional case, we refer to [Schratzberger 04].

**Definition 1.6.** A transformation  $\sigma_t$  defined by a matrix identity that removes the matrix  $\beta^{(t)}$  from the sequence of inverse matrices that changes an algorithm into a new form such that the sequence of Diophantine approximations  $\left\{\left(\frac{p_1^{*(t)}}{q^{*(t)}}, \frac{p_2^{*(t)}}{q^{*(t)}}\right)\right\}_{t=0}^{\infty}$  obtained from the new algorithm is a subsequence of the original one is called a *singularization*. We say that we have *singularized* the matrix  $\beta^{(t)}$ .

The process works in both directions: under certain criteria defined by a *law of insertion*, in using the inverse matrix identities from above we will be able to insert a matrix  $\beta^{*(t)}$ .

**Definition 1.7.** A transformation  $\sigma_t$  defined by a matrix identity that inserts the matrix  $\beta^{*(t)}$  into the sequence of inverse matrices that changes an algorithm into a new form such that the sequence of Diophantine approximations  $\left\{\left(\frac{p_1^{(t)}}{q^{(t)}}, \frac{p_2^{(t)}}{q^{(t)}}\right)\right\}_{t=0}^{\infty}$  obtained from the original algorithm is a subsequence of the sequence of approximations  $\left\{\left(\frac{p_1^{*(t)}}{q^{*(t)}}, \frac{p_2^{*(t)}}{q^{*(t)}}\right)\right\}_{t=0}^{\infty}$  of the new algorithm is called an *insertion*. We say we have *inserted* the matrix  $\beta^{*(t)}$ .

Recall that in the case of the PA, we have  $\beta_P(j, A) = \beta(j, A, 0)$  (i.e.,  $C = 0$ ), while all matrices of the JPA are of type 2. Hence when transforming the JPA evolution of some initial  $(x_1^{(0)}, x_2^{(0)})$ , we first need to switch the type  $j^{(t+1)}$ , where  $C^{(s)} = 0$  for  $1 \leq s \leq t$  and  $C^{(t+1)} \geq 1$ . In particular, we will try to insert a matrix  $\beta\left(1, \left[\frac{1}{x_2^{(t)}}\right], 0\right)$ .

This insertion will be defined by the matrix identities described in Section 2 and a law of insertion described in Section 3.

As opposed to the conversion processes described in [Iosifescu and Kraaikamp 02] and [Schratzberger 04], we use different matrix identities, and more of them. The process of insertion and singularization to be defined is much more complex, and depends on the choice of the initial  $(x_1^{(0)}, x_2^{(0)})$ . Hence it cannot be described by a simple law of insertion and singularization comparable with the laws of singularization LM, L1, and Lq from [Schratzberger 04], for example.

Instead, we will give certain criteria related to the digits that allow the insertion of the matrix  $\beta(1, [\frac{1}{x_2}], 0)$ . In case these criteria are not met, we will further describe how we may modify some matrices  $\beta(j^{(t+i)}, A^{(t+i)}, C^{(t+i)})$ , where  $i > 1$ , with similar operations, in order to fulfill the criteria afterward. Experiments indicate that this can be done in a finite number of operations a.e.

In the opposite case, if we want to transform the PA evolution of some corresponding  $(x_1^{(t)}, x_2^{(t)})$  into its JPA evolution, under certain criteria we may singularize the matrix  $\beta(1, A^{(t+1)}, 0)$ . Again, if the criteria are not met, we may perform modifications on some subsequent matrices  $\beta(j^{(t+i)}, A^{(t+i)}, C^{(t+i)})$  in order to guarantee the criteria afterward.

Since the matrices  $\beta(j^{(t)}, A^{(t)}, C^{(t)})$  are defined as the inverses of the matrices of the algorithm, their modification obviously yields the modification of the original algorithm. In particular, in the course of the transformation of one algorithm into the other, we will temporarily encounter the following algorithm:

**Definition 1.8.**  $T_S : \{1, 2\} \times [0, 1]^2 \rightarrow [0, 1]^2$  is defined as

$$T_S(j, x_1, x_2) = \begin{cases} \left( \begin{array}{l} \left( \frac{1}{x_2} - A, \frac{x_1}{x_2} - C \right) & \text{if } j = 1, A := \left[ \frac{1}{x_2} \right], \\ C := \left[ \frac{x_1}{x_2} \right], \end{array} \right. & \\ \left( \begin{array}{l} \left( \frac{x_2}{x_1} - C, \frac{1}{x_1} - A \right) & \text{if } j = 2, A := \left[ \frac{1}{x_1} \right], \\ C := \left[ \frac{x_2}{x_1} \right]. \end{array} \right) & \end{cases}$$

Let  $j_1, \dots, j_{t-1}$  be uniquely specified, and

$$\begin{aligned} (x_1^{(1)}, x_2^{(1)}) &= T_S(j_1, x_1^{(0)}, x_2^{(0)}), \dots, (x_1^{(t-1)}, x_2^{(t-1)}) \\ &= T_S(j_{t-1}, x_1^{(t-2)}, x_2^{(t-2)}). \end{aligned}$$

Analogously to the above, we define

$$A^{(t)} := A(j_t, x_1^{(t-1)}, x_2^{(t-1)})$$

and

$$C^{(t)} := C(j_t, x_1^{(t-1)}, x_2^{(t-1)}).$$

Not all cylinders

$$X\left(\left(j_1, A^{(1)}, C^{(1)}\right), \dots, \left(j_t, A^{(t)}, C^{(t)}\right)\right)$$

of this algorithm are full, and we have an analogue of the conditions of Perron (the *extended conditions of Perron*): if for some  $t$  we have  $A^{(t)} = C^{(t)}$  and  $j_t = j_{t+1}$ , then  $C^{(t+1)} > 0$ ; if  $A^{(t)} = C^{(t)}$  and  $j_t \neq j_{t+1}$ , then  $C^{(t+1)} = 0$ .

The inverses of the matrices of this algorithm are given in the obvious way:  $\beta^{(t)} := \beta(j_t, A^{(t)}, C^{(t)})$ . We may now specify the qualities of the generalized process of singularization and insertion to be defined: as indicated above, in transforming the JPA into the PA, our method starts with the first matrix  $\beta(j_t, A^{(t)}, C^{(t)})$ , where  $C^{(t)} > 0$  (note that in this case,  $j_1 = \dots = j_t = 2$ ). In the opposite case, in transforming the PA into the JPA, our method starts with the first matrix  $\beta(1, A^{(t)}, 0)$ .

For a given  $j \in \{1, 2\}$ , define  $j^*$  such that  $jj^* = 2$ . The matrices  $\beta(j_{t+i}, A^{(t+i)}, C^{(t+i)})$ , where  $i \geq 0$ , and thus the evolution of  $(x_1^{(t-1)}, x_2^{(t-1)})$ , will be modified in a way such that the following conditions hold:

1. For almost every such  $(x_1^{(t-1)}, x_2^{(t-1)})$ , after a finite number of operations, a matrix  $\beta(j_t^*, [1/x_2^{(t-1)}], 0)$  can be inserted or the matrix  $\beta(j_t, A_t, 0)$  can be singularized.
2. The evolution thus modified might contain an episode in which it can be described neither by the algorithm  $T_J$  nor by  $T_P$ . This episode can be described by the algorithm  $T_S$ , where the types are uniquely defined by the method.

After having successfully inserted

$$\beta(j_{t-1}^*, [1/x_2^{(t-1)}], 0)$$

or singularized  $\beta(j_t, A_t, 0)$ , we continue considering the next matrix  $\beta(j_{t+k}, A^{(t+k)}, C^{(t+k)})$  (of the modified evolution), where for  $k \geq 1$ , we either have  $C^{(t+k)} > 0$  (singularization) or  $C^{(t+k)} = 0$  (insertion).

We claim that:

3. The method also applies to  $T_S$  and its related matrices.

Note that although it is assumed from experimental evidence, at this point there is no proof that condition 1 (in particular, the word “finite”) holds for the process described in the following sections. Together with condition 3, this would result in a fourth condition:

4. The length of the episode described by  $T_S$  will be 0 infinitely often a.e.

The latter proposition would imply that for almost every  $(x_1, x_2)$ , its JPA evolution and the PA evolution “meet” infinitely often. Further, since they are defined by the matrices  $\beta^{(t)}$ , both algorithms would produce infinitely many common convergents

$$\left( \frac{p_1^{(t)}}{q^{(t)}}, \frac{p_2^{(t)}}{q^{(t)}} \right)$$

(cf. [Schratzberger 04]).

## 2. INGREDIENTS

Before we actually develop the method to transform the PA into the JPA, and vice versa, we establish the method’s ingredients, based on the techniques of singularization and insertion as described in [Iosifescu and Kraaikamp 02] and [Schratzberger 04]. As indicated, these techniques are based on some matrix identities, to be applied to the inverse matrices of an algorithm. Hence in the current section, we present the matrix identities used in the process described in Section 3. The following matrix identity of type  $I_1$ , and its inverse of type  $I_2$ , are fundamental:

Type  $I_1$ :

$$\begin{aligned} &\beta(j, A_1, 0)\beta(j, A_2, C_2)\beta(j^*, A_3, C_3) \\ &= \beta(j^*, A_1A_2 + C_2, A_2)\beta(j^*, A_3, A_1 + C_3). \end{aligned}$$

Type  $I_2$ :

$$\begin{aligned} &\beta(j, A_1, C_1)\beta(j, A_2, C_2) \\ &= \beta(j^*, X, 0)\beta(j^*, C_1, A_1 - XC_1)\beta(j, A_2, C_2 - X). \end{aligned}$$

Thus whenever in the PA evolution of an initial  $(x_1^{(0)}, x_2^{(0)})$  we encounter a  $j^{(t)} = 1$ , we will try to use matrix identity type  $I_1$  to switch the type of the matrix  $\beta^{(t)}$  from type 1 to type 2. Since  $j^{(t)} = 1$  means that  $x_2^{(t-1)} > x_1^{(t-1)}$ , this is exactly what the JPA evolution would look like (at time  $t$ ) (cf. Figures 1 and 2).

If  $j^{(t)} = j^{(t+1)} \neq j^{(t+2)}$  and  $A^{(t+2)} \geq A^{(t)} + C^{(t+2)}$  (where the extended conditions of Perron are fulfilled), we may directly apply matrix identity type

$I_1$ . In all other cases we first have to modify the digits  $(j^{(t)}, A^{(t)}, C^{(t)})$ ,  $(j^{(t+1)}, A^{(t+1)}, C^{(t+1)})$ , and so on to guarantee the criteria afterward. We state the following lemma.

**Lemma 2.1.** *Consider some initial  $(x_1, x_2) \in \mathbb{R}^2$  and a corresponding  $T$  evolution specified by some digits  $(j_1, A_1, C_1), \dots, (j_t, A_t, C_t)$ . If for all  $i, t \geq i \geq 1$ ,*

- (i)  $T^t(x_1, x_2) \in [0, 1]^2$ ,
- (ii)  $A_i \in \mathbb{N}, C_i \in \mathbb{N}_0$ ,
- (iii) *we have exactly one of*
  - (a)  $A_i > C_i$ ,
  - (b)  $A_i = C_i, j_i = j_{i+1}$ , and  $C_{i+1} > 0$ ,
  - (c)  $A_i = C_i, j_i \neq j_{i+1}$ , and  $C_{i+1} = 0$ ,

then for every  $k$  such that  $t - 1 \geq k \geq 0$ , we have

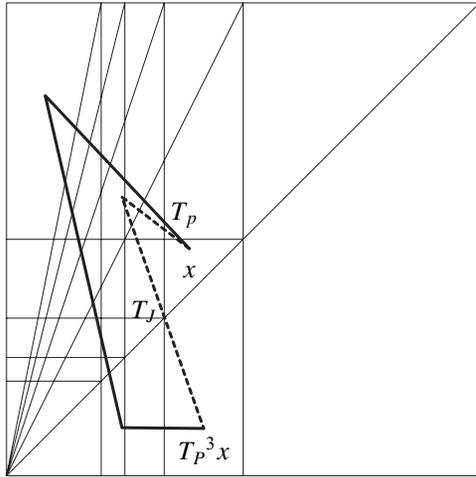
$$\begin{aligned} &T^k(x_1, x_2) \\ &\in X\left( (j_{k+1}, A_{k+1}, C_{k+1}), \dots, \right. \\ &\quad \left. (j_{t+1}, A(j_{t+1}, T^t(x_1, x_2)), C(j_{t+1}, T^t(x_1, x_2))) \right). \end{aligned}$$

**Remark 2.2.** The assumptions in item (iii) of Lemma 2.1 concerning  $C_{i+1}$  subject to  $j_i$  and  $j_{i+1}$  if  $A_i = C_i$  are implied by the extended conditions of Perron.

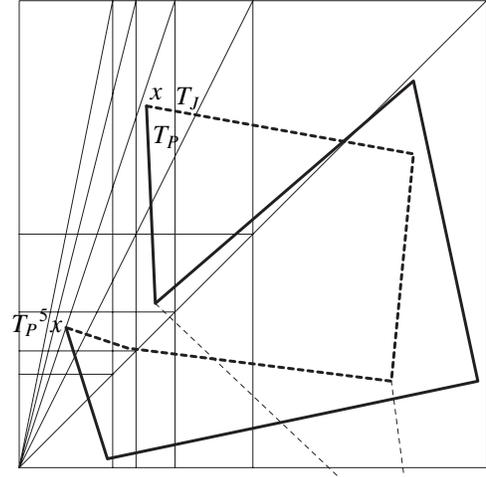
*Proof of Lemma 2.1:* Define  $\Delta(x, x', x'')$  to be the interior of the triangle defined by the vertices  $x, x'$ , and  $x''$ , and let  $\square(x, x', x'', x''')$  denote the interior of the quadrangle defined by the vertices  $x, x', x''$ , and  $x'''$ . The result follows inductively ( $k \rightarrow k - 1$ ) from the application of the local inverse  $V$ , with the appropriate digits, to  $\square((0, 0), (1, 0), (1, 1), (0, 1))$  (if  $A_k > C_k$ ) or  $\Delta((0, 0), (1, 0), (1, 1))$  and  $\Delta((0, 0), (1, 1), (0, 1))$  (if  $A_k = C_k$  and  $j_k = 1$  or  $j_k = 2$ , respectively).  $\square$

**Example 2.3.** Let  $(x_1, x_2) \in X_P((1, 2), (1, 1), (2, 4))$ . After the application of matrix identity type  $I_1$ , the modified evolution is specified by the digits  $(2, 2, 1), (2, 4, 2)$ . Hence in this case, the singularization leads directly to the JPA evolution of  $(x_1, x_2)$ . See Figure 4.

Recall that to apply the above matrix identities successfully, there are strict assumptions concerning the types. In particular, if  $j_2 = j_3$ , we may alternatively use the following matrix identities:



**FIGURE 4.**  $x \in X_P((1, 2), (1, 1), (2, 4))$ . The dashed line shows the evolution of  $x$  after singularization.



**FIGURE 5.**  $x \in X_P((1, 1), (1, 2), (2, 1), (2, 1), (2, 5))$ . The dashed lines show the evolution of  $x$  after application of matrix identities  $II_3$  (thin line) and  $I_1$ .

Type  $II_1$ :

$$\begin{aligned} &\beta(j, A_1, 0)\beta(j, A_2, C_2)\beta(j, A_3, 0)\beta(j, A_4, C_4) \\ &= \beta(j^*, A_1 A_2 + C_2, A_2)\beta(j, A_3, 0) \\ &\quad \times \beta(j, A_1 + A_4, C_4 - A_1 A_3). \end{aligned}$$

Type  $II_2$ :

$$\begin{aligned} &\beta(j, A_1, C_1)\beta(j^*, A_2, 0)\beta(j^*, A_3, C_3) \\ &= \beta(j^*, X, 0)\beta(j^*, C_1, A_1 - X C_1)\beta(j^*, A_2, 0) \\ &\quad \times \beta(j^*, A_4 - X, C_4 + X A_2). \end{aligned}$$

Note that matrix identity type  $I_1$  can be directly applied only if  $A_3 \geq A_1 + C_3$  (with respect to the extended conditions of Perron). Similar restrictions subject to the partial quotients  $A_i$  and  $C_i$  ( $i \geq 1$ ) are true for matrix identities types  $I_2$ ,  $II_1$ , and  $II_2$ . In these cases, we first need to modify some digits  $(j_i, A_i, C_i)$  ( $i \geq 2$ ) to meet the criteria. For example, we need to reduce  $C_3$ . We propose the following matrix identities:

Type  $I_3$ :

$$\begin{aligned} &\beta(j, A_1, C_1)\beta(j, A_2, C_2)\beta(j^*, A_3, C_3) \\ &= \beta(j, A_1 - K, C_1)\beta(j, A_2, C_2 + K A_2) \\ &\quad \times \beta(j^*, A_3, C_3 + K), \\ &\beta(j, A_1, C_1)\beta(j^*, A_2, C_2)\beta(j, A_3, C_3) \\ &= \beta(j, A_1, C_1 - K)\beta(j^*, A_2, C_2 + K A_2) \\ &\quad \times \beta(j, A_3, C_3 + K). \end{aligned}$$

Type  $II_3$ :

$$\begin{aligned} &\beta(j, A_1, C_1)\beta(j, A_2, C_2)\beta(j, A_3, 0)\beta(j, A_4, C_4) \\ &= \beta(j, A_1 - K, C_1)\beta(j, A_2, C_2 + K A_2)\beta(j, A_3, 0) \\ &\quad \times \beta(j, A_4 + K, C_4 - K A_3), \\ &\beta(j, A_1, C_1)\beta(j^*, A_2, C_2)\beta(j^*, A_3, 0)\beta(j^*, A_4, C_4) \\ &= \beta(j, A_1, C_1 - K)\beta(j^*, A_2, C_2 + K A_2)\beta(j^*, A_3, 0) \\ &\quad \times \beta(j^*, A_4 + K, C_4 - K A_3). \end{aligned}$$

**Example 2.4.** Let

$$(x_1, x_2) \in X_P((1, 1), (1, 2), (2, 1), (2, 1), (2, 5)).$$

We cannot apply matrix identity type  $II_1$ , since  $j_2 \neq j_3$ ; nor can we apply  $I_1$ , since  $A_3 = A_1 + C_3$ ,  $j_3 = j_4$ , and  $C_4 = 0$  (thus the extended conditions of Perron with respect to the digits  $j_4$  and  $C_4$  would not be fulfilled). Hence we first have to modify  $\beta_3$  in applying matrix identity type  $II_3$  (where  $K = -1$ ) from the second matrix, which results in an evolution specified by the digits  $(1, 1, 0)$ ,  $(1, 2, 1)$ ,  $(2, 1, -1)$ ,  $(2, 1, 0)$ ,  $(2, 4, 1)$ . Note that this forces the thus modified orbit of  $(x_1, x_2)$  to leave the unit square temporarily (cf. Figure 5). Now the conditions for matrix identity  $I_1$  are fulfilled, and we may finish the singularization process. The modified evolution of  $(x_1, x_2)$  is specified by the digits  $(2, 3, 2)$ ,  $(2, 1, 0)$ ,  $(2, 1, 0)$ ,  $(2, 4, 1)$ .

Example 2.4 illustrates how, in some cases, the manipulation of some digits  $(j_i, A_i, C_i)$  ( $i \geq 2$ ) is necessary

to meet the criteria to successfully singularize or insert a matrix  $\beta(j_1, A_1, C_1)$ . In particular, these manipulations may result in partial quotients  $C_i < 0$ . The following matrix identities will be used in related cases:

Type  $III_1$ :

$$\begin{aligned} &\beta(j, A_1, C_1)\beta(j, 1, 0)\beta(j, A_3, -C_3)\beta(j, 1, 0)\beta(j^*, A_5, C_5) \\ &= \beta(j, A_1, C_1 + 1)\beta(j^*, A_3 - C_3 + 1, C_3 - 1) \\ &\quad \times \beta(j^*, A_5 + 1, C_5 - 1), \\ &\beta(j, A_1, C_1)\beta(j^*, 1, 0)\beta(j^*, A_3, -C_3)\beta(j^*, 1, 0)\beta(j, A_5, C_5) \\ &= \beta(j, A_1 + 1, C_1)\beta(j, A_3 - C_3 + 1, C_3 - 1) \\ &\quad \times \beta(j, A_5 + 1, C_5 - 1). \end{aligned}$$

Type  $III_2$ :

$$\begin{aligned} &\beta(j, A_1, C_1)\beta(j, A_2, -C_2)\beta(j, A_3, C_3) \\ &= \beta(j, A_1 - 1, C_1)\beta(j^*, 1, 0)\beta(j^*, A_2 - C_2, C_2 - 1) \\ &\quad \times \beta(j^*, 1, 0)\beta(j, A_3 - 1, C_3 + 1), \\ &\beta(j, A_1, C_1)\beta(j^*, A_2, -C_2)\beta(j^*, A_3, C_3) \\ &= \beta(j, A_1, C_1 - 1)\beta(j, 1, 0)\beta(j, A_2 - C_2, C_2 - 1) \\ &\quad \times \beta(j, 1, 0)\beta(j^*, A_3 - 1, C_3 + 1). \end{aligned}$$

Type  $IV_1$ :

$$\begin{aligned} &\beta(j, A_1, C_1)\beta(j^*, 1, -1)\beta(j^*, A_3, C_3) \\ &= \beta(j, A_1 + 1, C_1 - 1)\beta(j, 1, -1)\beta(j, C_3 + 1, A_3 - 1) \end{aligned}$$

Type  $IV_2$ :

$$\begin{aligned} &\beta(j, A_1, C_1)\beta(j, 1, -1)\beta(j, A_3, C_3) \\ &= \beta(j, A_1 - 1, C_1 + 1)\beta(j^*, 1, -1)\beta(j^*, C_3 + 1, A_3 - 1) \end{aligned}$$

Finally, to overcome potential situations in which for some  $i \geq 1$ , we have  $A_i = C_i = 0$ , we propose the following matrix identities:

Type  $V_1$ :

$$\begin{aligned} &\beta(j, A_1, C_1)\beta(j, 0, 0)\beta(j^*, A_3, C_3) \\ &= \beta(j, A_1 + C_3, A_3 + C_1), \\ &\beta(j, A_1, C_1)\beta(j^*, 0, 0)\beta(j, A_3, C_3) \\ &= \beta(j, A_1 + A_3, C_1 + C_3). \end{aligned}$$

Type  $V_2$ :

$$\beta(j, A_1, C_1) = \beta(j, Y, X)\beta(j^*, 0, 0)\beta(j, A_1 - Y, C_1 - X).$$

Type  $V_3$ :

$$\beta(j, A_1, C_1) = \beta(j, Y, X)\beta(j, 0, 0)\beta(j^*, C_1 - X, A_1 - Y).$$

Based on these matrix identities, we will now define a process of insertion and singularization.

### 3. THE PROCESS OF INSERTION AND SINGULARIZATION

Based on the previous section, we will describe a constructive method to transform the PA into the JPA (and vice versa). The process is constructed as a set of states.

Depending on the digits, in each state the process either terminates (i.e., a matrix has successfully been singularized, or inserted, or modified) or leads to another state, with similar options. In the latter case, after successful completion of the second state, the process returns to the initial state.

The technical ingredients have been established in the previous section. Hence in the following, we mainly specify the order of application of the above matrix identities. For  $i \geq 1$ , we most generally assume  $A_i \in \mathbb{N}_0$  and  $C_i \in \mathbb{Z}$ . The initial situations are characterized in states  $P$  (from PA to JPA) and  $J$  (from JPA to PA).

State  $S$  describes singularization criteria for specific situations that do not occur in any regular PA evolution (where  $C_i = 0$ ) or in a regular JPA evolution (where  $j_i = 2$ ). However, these situations might temporarily occur in the course of the transformation of one algorithm into the other.

Note that state  $J$  also includes the case  $C_1 > 0$  and  $j_1 = j_2 = 1$ , which again is typical neither of a regular PA evolution nor of a regular JPA evolution. However, this case is treated exactly the same way as the case  $C_1 > 0$  and  $j_1 = j_2 = 2$ , the initial situation of transforming a JPA evolution into a PA evolution.

At any state, we first check the types, to decide which suitable matrix identity can be applied. If none of the matrix identities described in Section 2 applies, we need to switch some particular  $j_t$ . This is again done by continuing at state  $P$ ,  $J$ , or  $S$  (depending on  $C_t$ ,  $j_t$ , and  $j_{t+1}$ ), starting with matrix  $\beta_t$ . After successful completion of the second state, we return to the initial state.

Once the types are fixed, and hence the particular matrix identity to be applied, we compare the modified partial quotients with the criteria for a successful application of this matrix identity.

Again, if the criteria are not met, we need to modify some partial quotients  $A_{t+1}$  or  $C_{t+1}$ . This is done by continuing at states  $H1_1$ – $H1_3$  and  $H2_1$ – $H1_4$ , starting with matrix  $\beta_t$ . In particular, at states  $H2_1$  and  $H2_2$ ,  $C_{t+1}$  is enlarged, while at states  $H2_3$  and  $H2_4$ ,  $C_{t+1}$  is reduced (by at least 1). Thus  $H2_1$  and  $H2_2$  describe exactly the inverse modifications, as opposed to  $H2_3$  and  $H2_4$ . Similarly, state  $H1_1$  is inverse to states  $H1_2$  and  $H1_3$ , and state  $P$  is inverse to states  $J$  and  $S$ .

### 3.1 State $P$

Consider the  $T$  evolution of some  $(x_1, x_2)$  specified by the digits  $(j_1, A_1, C_1), (j_2, A_2, C_2), \dots$  and its matrices  $\beta_1, \beta_2, \dots$ , where  $A_1, A_2, A_3 \in \mathbb{N}$ ,  $C_1 = 0$ ,  $C_2 \in \mathbb{N}_0$ , and  $C_3 \in \mathbb{Z}$ .

If  $j_1 \neq j_2$ : Switch the type  $j_2$  (state  $P, J$ , or  $S$ , accordingly, starting with matrix  $\beta_2$ ).

If  $j_1 = j_2$ :

- (1) If  $j_2 = j_3$  and  $C_3 = 0$ : If  $j_3 \neq j_4$ , then switch the type  $j_4$  (state  $P, J$ , or  $S$ , accordingly, starting with matrix  $\beta_4$ ).

If  $j_3 = j_4$ :

If  $C_4 \geq A_1 A_3$ : Apply matrix identity type  $II_1$ , starting with matrix  $\beta_1$ . The process terminates.

If  $C_4 < A_1$  and  $A_3 = 1$ : Continue with state  $H1_1$ , starting with matrix  $\beta_2$ .

If  $C_4 < A_1 A_3$  and  $A_3 > 1$ : Switch the type  $j_3$  (state  $P$ , starting with matrix  $\beta_3$ ).

- (2) If  $j_2 = j_3$  and  $C_3 > 0$ : Switch the type  $j_3$  (state  $P, J$ , or  $S$ , accordingly, starting with matrix  $\beta_3$ ).
- (3) If  $j_2 \neq j_3$ : If  $A_3 \geq A_1 + C_3$  (where the extended conditions of Perron with respect to the digits  $j_4$  and  $C_4$  are fulfilled), then apply matrix identity type  $I_1$  from  $\beta_1$ . The process terminates.

In all other cases, reduce  $C_3$  (states  $H2_3$  and  $H2_4$ , starting with matrix  $\beta_2$ ). See Figure 6.

### 3.2 State $J$

Consider the  $T$  evolution of some  $(x_1, x_2)$  specified by the digits  $(j_1, A_1, C_1), (j_2, A_2, C_2), \dots$  and its matrices  $\beta_1, \beta_2, \dots$ , where  $A_1, A_2, C_1 \in \mathbb{N}$ ,  $A_1 \geq C_1$ , and  $j_1 = j_2$ . Define, for  $i \geq 1$ ,

$$\left[ \frac{A_i}{C_i} \right] =: \frac{A_i - r_i}{C_i}.$$

Thus  $r_i \leq \min\{A_i - C_i, C_i - 1\}$ , and  $r_i \leq \frac{A_i - 1}{2}$ .

- (1) If  $C_2 \geq \frac{A_1 - r_1}{C_1}$ , then apply matrix identity type  $I_2$ , starting with matrix  $\beta_1$ . Here and in the following, the  $X$  in matrix identity type  $I_2$  is replaced by  $\frac{A_1 - r_1}{C_1}$ . The process terminates.
- (2) If  $C_2 = \frac{A_1}{C_1} - 1$  (thus  $\frac{A_1}{C_1} \in \mathbb{N}$ ) and  $A_1 > C_1$ , first apply matrix identity type  $I_2$ , starting with matrix  $\beta_1$ . Define  $\tilde{\beta}_i = \beta(\tilde{j}_i, \tilde{A}_i, \tilde{C}_i)$  to be the  $i$ th matrix

of the modified evolution after the insertion. Then apply matrix identity type  $I_3$ , starting with matrix  $\tilde{\beta}_1$ , where  $K = 1$ . The process terminates.

- (3) If  $C_2 < \frac{A_1 - r_1}{C_1} - 1$ ,  $r_1 > 0$ , and  $A_2 > \frac{A_1 - r_1}{C_1} - C_2$ : If  $j_2 \neq j_3$ , then switch the type  $j_3$  (state  $P, J$ , or  $S$ , accordingly, starting with matrix  $\beta_3$ ).

If  $j_2 = j_3$ , first apply matrix identity type  $I_2$ , starting with matrix  $\beta_1$ . Define  $\tilde{\beta}_i = \beta(\tilde{j}_i, \tilde{A}_i, \tilde{C}_i)$  in analogy to the above; that is, here and in the following,  $\tilde{\beta}_i$  will always be the  $i$ th matrix of the modified evolution after the most recent modification. Continue with state  $H1_2$ , starting with matrix  $\tilde{\beta}_2$ . The process terminates.

- (4) If  $C_2 < \frac{A_1}{C_1} - 1$  (thus  $\frac{A_1}{C_1} \in \mathbb{N}$ ) and  $A_2 + 1 > \frac{A_1}{C_1} - C_2$ : If  $j_2 \neq j_3$ , then switch the type  $j_3$  (state  $P, J$ , or  $S$ , accordingly, starting with matrix  $\beta_3$ ).

If  $j_2 = j_3$ , first apply matrix identity type  $I_2$ , starting with matrix  $\beta_1$ . Continue with matrix identity type  $I_3$ , starting with matrix  $\tilde{\beta}_1$ , where  $K = 1$ . Then continue with state  $H1_2$ , starting with matrix  $\tilde{\beta}_2$ . The process terminates.

- (5) If  $C_2 = \frac{A_1 - r_1}{C_1} - 1$ ,  $r_1 > 0$  and  $A_2 = 1$ , then if  $j_2 \neq j_3$ : Switch the type  $j_3$  (state  $P, J$ , or  $S$ , accordingly, starting with matrix  $\beta_3$ ).

If  $j_2 = j_3$ , first apply matrix identity type  $I_2$ , starting with matrix  $\beta_1$ , and continue with state  $H1_3$ , starting with matrix  $\tilde{\beta}_2$ . The process terminates.

- (6) If  $C_2 = \frac{A_1}{C_1} - 2$  (thus  $\frac{A_1}{C_1} \in \mathbb{N}$ ) and  $A_2 = 1$ : If  $j_2 \neq j_3$ , then switch the type  $j_3$  (state  $P, J$ , or  $S$ , accordingly, starting with matrix  $\beta_3$ ).

If  $j_2 = j_3$ , first apply matrix identity type  $I_2$ , starting with matrix  $\beta_1$ . Continue with matrix identity type  $I_3$ , starting with matrix  $\tilde{\beta}_1$ , where  $K = 1$ . Then continue with state  $H1_3$ , starting with matrix  $\tilde{\beta}_2$ . The process terminates.

In all other cases, enlarge  $C_2$  (state  $H2_1$  or  $H2_2$ , starting with  $\beta_1$ ).

### 3.3 State $S$

Consider the  $T$  evolution of some  $(x_1, x_2)$  specified by the digits  $(j_1, A_1, C_1), (j_2, A_2, C_2), \dots$  and its matrices  $\beta_1, \beta_2, \dots$ , where  $A_1, A_2, C_1 \in \mathbb{N}$ ,  $A_1 \geq C_1$ , and  $j_1 \neq j_2$ .

If  $C_2 = 0$  and  $j_2 \neq j_3$ , then switch the type  $j_3$  (state  $P, J$ , or  $S$ , accordingly, starting with matrix  $\beta_3$ ).

If  $C_2 = 0$ ,  $j_2 = j_3$ , and  $A_3 \geq (A_2 + 1)\frac{A_1 - r_1}{C_1} + C_3$ , apply matrix identity type  $I_2$ , starting with matrix  $\beta_1$ .



If  $\tilde{A}_2 \geq \tilde{C}_2$ , and the extended conditions of Perron with respect to the digits  $\tilde{j}_3$  and  $\tilde{C}_3$  are fulfilled, the process terminates.

### 3.5 State $H1_2$

Consider the  $T$  evolution of some  $(x_1, x_2)$  specified by the digits  $(j_1, A_1, C_1), (j_2, A_2, C_2), \dots$  and its matrices  $\beta_1, \beta_2, \dots$ , where either  $A_2 \neq 1$  or both  $C_2 \neq 0$  and  $C_2 \neq -1$ , and  $j_2 = j_3$ .

If  $A_3 \leq C_3 + 2$ , where the extended conditions of Perron with respect to the digits  $j_4$  and  $C_4$  would not be fulfilled, reduce  $C_3$  (states  $H2_3$  and  $H2_4$ , accordingly, starting with matrix  $\beta_2$ ).

If  $A_3 \geq C_3 + 2$ , and the extended conditions of Perron with respect to the digits  $j_4$  and  $C_4$  would be fulfilled:

- (1) If  $A_3 \geq 2$  and  $C_3 \geq -1$ : Apply matrix identity type  $III_2$ , starting with  $\beta_1$ .

If  $\tilde{A}_3 = \tilde{C}_3$ , where the extended conditions of Perron with respect to the digits  $\tilde{j}_4$  and  $\tilde{C}_4$  are not fulfilled, or  $\tilde{A}_3 < \tilde{C}_3$ , reduce  $\tilde{C}_3$  (states  $H2_3$  and  $H2_4$ , accordingly, starting with matrix  $\tilde{\beta}_2$ ).

If  $\tilde{A}_3 \geq \tilde{C}_3$ , where the extended conditions of Perron with respect to the digits  $\tilde{j}_4$  and  $\tilde{C}_4$  are fulfilled, the process terminates.

- (2) If  $A_3 \geq 2$  and  $C_3 = -2$ : Apply matrix identity type  $III_2$ , starting with matrix  $\beta_1$ , and continue with matrix identity type  $I_3$ , where  $K = 1$ , starting with matrix  $\tilde{\beta}_3$ .

If  $\tilde{A}_3 = \tilde{C}_3$ , where the extended conditions of Perron with respect to the digits  $\tilde{j}_4$  and  $\tilde{C}_4$  are not fulfilled, or  $\tilde{A}_3 < \tilde{C}_3$ , reduce  $\tilde{C}_3$  (states  $H2_3$  and  $H2_4$ , accordingly, starting with matrix  $\tilde{\beta}_2$ ).

If  $\tilde{A}_3 \geq \tilde{C}_3$ , where the extended conditions of Perron with respect to the digits  $\tilde{j}_4$  and  $\tilde{C}_4$  are fulfilled, the process terminates.

- (3) If  $A_3 = 1$  and  $C_3 = -1$ : If  $j_3 = j_4$ , then switch the type  $j_4$  (state  $P, J$ , or  $S$ , accordingly, starting with matrix  $\beta_4$ ).

If  $j_3 \neq j_4$ , apply matrix identity type  $III_2$ , starting with matrix  $\beta_1$ , and continue with matrix identity type  $V_1$ , starting with matrix  $\beta_4$ .

If  $\tilde{A}_3 = \tilde{C}_3$ , where the extended conditions of Perron with respect to the digits  $\tilde{j}_4$  and  $\tilde{C}_4$  are not fulfilled, or  $\tilde{A}_3 < \tilde{C}_3$ , reduce  $\tilde{C}_3$  (states  $H2_3$  and  $H2_4$ , accordingly, starting with matrix  $\tilde{\beta}_2$ ).

If  $\tilde{A}_3 \geq \tilde{C}_3$ , where the extended conditions of Perron with respect to the digits  $\tilde{j}_4$  and  $\tilde{C}_4$  are fulfilled, the process terminates.

### 3.6 State $H1_3$

Consider the  $T$  evolution of some  $(x_1, x_2)$  specified by the digits  $(j_1, A_1, C_1), (j_2, A_2, C_2), \dots$  and its matrices  $\beta_1, \beta_2, \dots$ , where  $A_2 = 1$  and  $C_2 = -1$ .

If  $j_2 \neq j_3$ , then switch the type  $j_3$  (state  $P, J$ , or  $S$ , starting with matrix  $\beta_3$ , accordingly).

If  $j_2 = j_3$ :

- (1) If  $A_3 \geq C_3 + 2$ : Apply matrix identity type  $IV_1$  or  $IV_2$ , accordingly, starting with matrix  $\beta_1$ . If  $\tilde{A}_3 = \tilde{C}_3$ , where the extended conditions of Perron with respect to the digits  $\tilde{j}_4$  and  $\tilde{C}_4$  are not fulfilled, or  $\tilde{A}_3 < \tilde{C}_3$ , reduce  $\tilde{C}_3$  (states  $H2_3$  and  $H2_4$ , accordingly, starting with matrix  $\tilde{\beta}_2$ ).

If  $\tilde{A}_3 \geq \tilde{C}_3$ , where the extended conditions of Perron with respect to the digits  $\tilde{j}_4$  and  $\tilde{C}_4$  are fulfilled, continue with state  $H1_2$ , starting with matrix  $\tilde{\beta}_1$ . The process terminates.

- (2) If  $A_3 \leq C_3 + 1$ : If  $j_3 = j_4$  and  $C_4 > 0$ , or  $j_3 \neq j_4$  and  $C_4 = 0$ , apply matrix identities type  $IV_1$  or  $IV_2$ , accordingly, starting with matrix  $\beta_1$ .

If  $j_3 = j_4$  and  $C_4 = 0$ , or  $j_3 \neq j_4$  and  $C_4 > 0$ , reduce  $C_3$  (states  $H2_3$  and  $H2_4$ , accordingly, starting with matrix  $\beta_2$ ). Continue with state  $H1_2$ , starting with matrix  $\tilde{\beta}_1$ . The process terminates.

### 3.7 State $H2_1$

Consider the  $T$  evolution of some  $(x_1, x_2)$  specified by the digits  $(j_1, A_1, C_1), (j_2, A_2, C_2), \dots$  and its matrices  $\beta_1, \beta_2, \dots$ , where  $j_2 \neq j_3$ .

- (1) If  $A_3 = C_3 + 1$ , and the extended conditions of Perron with respect to the digits  $j_4$  and  $C_4$  would be fulfilled, or  $A_3 > C_3 + 1$ , apply matrix identity type  $I_3$ , where  $K = 1$ , starting with matrix  $\beta_1$ . The process terminates.

- (2) If  $A_3 = C_3 + 1$ , and the extended conditions of Perron with respect to the digits  $j_4$  and  $C_4$  would not be fulfilled, apply matrix identity type  $I_3$ , where  $K = 1$ , starting with matrix  $\beta_1$ .

- (3) If  $A_3 = C_3$ , then if either  $j_3 = j_4$  and  $C_4 = 0$ , or  $j_3 \neq j_4$  and  $C_4 > 0$ , reduce  $C_3$  (states  $H2_3$  and  $H2_4$ , starting with matrix  $\beta_2$ , respectively). The process terminates.

If  $j_3 \neq j_4$  and  $C_4 = 0$ , switch the type  $j_4$  (states  $P$ ,  $J$ , and  $S$ , starting with matrix  $\beta_4$ , respectively).

If  $j_3 = j_4$  and  $C_4 > 0$ , apply matrix identity type  $I_2$ , starting with matrix  $\beta_3$ , and continue with state  $H2_2$ , starting with matrix  $\beta_1$ .

### 3.8 State $H2_2$

Consider the  $T$  evolution of some  $(x_1, x_2)$  specified by the digits  $(j_1, A_1, C_1)$ ,  $(j_2, A_2, C_2), \dots$ , and its matrices  $\beta_1, \beta_2, \dots$ , where  $j_2 = j_3$ .

(1) If  $C_3 > 0$ , then switch the type  $j_3$  (state  $P$ ,  $J$ , or  $S$ , starting with matrix  $\beta_3$ , accordingly) and continue with state  $H2_1$ , starting with matrix  $\tilde{\beta}_1$ .

(2) If  $C_3 = 0$ : If  $j_3 \neq j_4$ , then switch the type  $j_4$  (states  $P$ ,  $J$  and  $S$ , starting with matrix  $\beta_4$ , respectively).

If  $j_3 = j_4$ :

(2.1) If  $C_4 \geq A_3$ , apply matrix identity type  $II_3$ , where  $K = 1$ , starting with matrix  $\beta_1$ . The process terminates.

(2.2) If  $C_4 < A_3$  and  $A_3 \geq 2$ , enlarge  $C_4$  (states  $H2_1$  and  $H2_2$ , starting with matrix  $\beta_3$ , respectively).

(2.3) If  $A_3 = 1$  and  $C_4 = 0$ , enlarge  $C_4$  (states  $H2_1$  and  $H2_2$ , starting with matrix  $\beta_3$ , respectively). Then,

If  $\tilde{A}_3 = 1$  and  $\tilde{C}_3 = 0$ : Apply matrix identity type  $II_3$ , where  $K = 1$ , starting with matrix  $\beta_1$ . The process terminates.

If  $\tilde{A}_3 = \tilde{C}_3 = 0$ : If  $\tilde{j}_4 \neq \tilde{j}_5$ , switch the type  $\tilde{j}_5$  (states  $J$  and  $S$ , starting with matrix  $\tilde{\beta}_5$ , respectively).

If  $\tilde{j}_4 = \tilde{j}_5$ , apply matrix identity type  $I_2$ , starting with matrix  $\tilde{\beta}_4$ . Continue with matrix identity type  $V_1$ , starting with matrix  $\tilde{\beta}_2$ . If  $\tilde{C}_4 < 0$ , continue with states  $H1_1$  and  $H1_3$ , accordingly, starting with matrix  $\tilde{\beta}_3$ , followed by matrix identity type  $I_3$ , starting with matrix  $\tilde{\beta}_1$ . The process terminates.

### 3.9 State $H2_3$

Consider the  $T$  evolution of some  $(x_1, x_2)$  specified by the digits  $(j_1, A_1, C_1)$ ,  $(j_2, A_2, C_2), \dots$  and its matrices  $\beta_1, \beta_2, \dots$ , where  $j_2 \neq j_3$ .

(1) If  $C_3 > 0$ , apply matrix identity type  $I_1$ , where  $K = -1$ , starting with matrix  $\beta_1$ . The process terminates.

(2) If  $C_3 = 0$ , then if  $j_3 \neq j_4$  and  $A_4 = C_4$ , switch the type  $j_4$  (states  $P$ ,  $J$  and  $S$ , starting with matrix  $\beta_4$ , respectively).

If  $j_3 = j_4$  and  $A_4 = C_4 = 1$ : Reduce  $C_4$  (states  $H2_3$  and  $H2_4$ , accordingly, starting with matrix  $\beta_3$ )

If  $j_3 = j_4$ ,  $j_4 \neq j_5$ ,  $A_4 = 1$  and  $C_4 = 0$ : Switch the type  $j_5$  (state  $P$ ,  $J$ , or  $S$ , accordingly, starting with matrix  $\beta_5$ ).

If  $j_3 = j_4 = j_5$ ,  $A_4 = 1$  and  $C_4 = 0$ , apply matrix identity type  $I_3$ , where  $K = -1$ , starting with matrix  $\beta_1$ . Continue with state  $H1_1$ , starting with matrix  $\tilde{\beta}_3$ , followed by matrix identity type  $V_3$ , where  $Y = \tilde{A}_2$  and  $X = \tilde{C}_2 - 1$ , starting with matrix  $\tilde{\beta}_2$ . Then switch the type  $\tilde{j}_4$  (state  $P$ , starting with matrix  $\tilde{\beta}_4$ ), to continue with state  $H2_4$ , starting with matrix  $\tilde{\beta}_3$ , until  $\tilde{A}_3 > 0$ . The process terminates.

In any other case, enlarge  $C_3$  (states  $H2_1$  and  $H2_2$ , starting with matrix  $\beta_2$ , respectively).

### 3.10 State $H2_4$

Consider the  $T$  evolution of some  $(x_1, x_2)$  specified by the digits  $(j_1, A_1, C_1)$ ,  $(j_2, A_2, C_2), \dots$  and its matrices  $\beta_1, \beta_2, \dots$ , where  $j_2 = j_3$ .

(1) If  $C_3 > 0$ , then switch the type  $j_3$  (state  $P$ ,  $J$ , or  $S$ , accordingly, starting with matrix  $\beta_3$ ) and continue with state  $H2_3$ , starting with matrix  $\beta_1$ .

(2) If  $C_3 = 0$ : If  $j_3 \neq j_4$ , then switch the type  $j_4$  (state  $P$ ,  $J$ , or  $S$ , accordingly, starting with matrix  $\beta_4$ ).

If  $j_3 = j_4$ :

(2.1) If either  $A_4 - 1 = A_3 + C_4$ , and the extended conditions of Perron with respect to  $j_5$  and  $C_5$  would be fulfilled, or  $A_4 - 1 > A_3 + C_4$ , then apply matrix identity type  $II_3$ , where  $K = -1$ , starting with matrix  $\beta_1$ . The method terminates.

(2.2) If  $A_4 > 1$ , and either  $A_4 - 1 = A_3 + C_4$ , where the extended conditions of Perron with respect to  $j_5$  and  $C_5$  would not be fulfilled, or  $A_4 - 1 < A_3 + C_4$ , then reduce  $C_4$  (states  $H2_3$  and  $H2_4$ , starting with matrix  $\beta_3$ , respectively).

(2.3) If  $A_4 = 1$ , then switch the type  $j_3$  (state  $P$ , starting with matrix  $\beta_3$ ) and continue with state  $H2_3$ , starting with matrix  $\beta_1$ .

#### 4. EXAMPLES

**Example 4.1.** Let

$$(x_1, x_2) \in X_P((1, 3), (1, 4), (1, 1), (1, 4), (1, 2)).$$

We start at state  $P$ . Since  $j_1 = j_2 = j_3 = j_4$ ,  $A_3 = 1$ , and  $C_4 < A_1$ , we continue at state  $H1_1$ : we apply matrix identities  $V_2$ ,  $III_2$ , and  $I_3$  in turn, and return to state  $P$ . Now  $\tilde{j}_2 \neq \tilde{j}_3$ , and  $\tilde{A}_3 \geq \tilde{A}_1 + \tilde{C}_3$ . We may therefore apply matrix identity  $I_1$ , and the process terminates after three states:

$$\begin{aligned} & \beta(1, 3, 0)\beta(1, 4, 0)\beta(1, 1, 0)\beta(1, 4, 0)\beta(1, 2, 0) \\ &= \beta(1, 3, 0)\beta(1, 4, 0)\beta(1, 1, 0)\beta(1, 4, 0)\beta(1, 1, 0) \\ & \quad \beta(2, 0, 0)\beta(1, 1, 0) \\ &= \beta(1, 3, 0)\beta(1, 4, 1)\beta(2, 5, -1)\beta(2, 1, -1)\beta(1, 1, 0) \\ &= \beta(1, 3, 0)\beta(1, 4, 1)\beta(2, 4, -1)\beta(2, 1, 0)\beta(1, 1, 1) \\ &= \beta(2, 13, 4)\beta(2, 4, 2)\beta(2, 1, 0)\beta(1, 1, 1). \end{aligned}$$

The resulting evolution is specified by the digits  $(2, 13, 4)$ ,  $(2, 4, 2)$ ,  $(2, 1, 0)$ , and  $(1, 1, 1)$ .

**Example 4.2.** Consider the  $T$  evolution of some  $(x_1, x_2)$  specified by the digits  $(1, 1, 1)$ ,  $(2, 1, 0)$ , and  $(2, 2, 0)$ . We start at state  $S$ . We may directly apply matrix identity  $II_1$ . The method terminates after the first state:

$$\begin{aligned} & \beta(1, 1, 1)\beta(2, 1, 0)\beta(2, 2, 0) \\ &= \beta(2, 1, 0)\beta(2, 1, 0)\beta(2, 1, 0)\beta(2, 1, 1) \end{aligned}$$

If we combine Examples 4.1 and 4.2, we find that

$$\begin{aligned} & X_P((1, 3), (1, 4), (1, 1), (1, 4), (1, 2), (2, 1), (2, 2)) \\ &= X_J((13, 4), (4, 2), (1, 0), (1, 0), (1, 0), (1, 0), (1, 1)). \end{aligned}$$

In particular, for

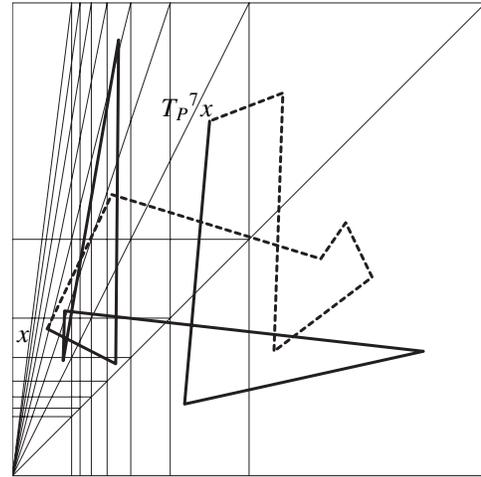
$$x \in X_P((1, 3), (1, 4), (1, 1), (1, 4), (1, 2), (2, 1), (2, 2)),$$

we have

$$T_P^7 x = T_J^7 x,$$

that is, the orbits of the PA evolution and the JPA evolution meet again after seven iterations each. See Figure 7.

The basic principle of the above method is that if we cannot directly apply one of the matrix identities to insert or singularize a suitable matrix, we may always modify some subsequent digits in order to guarantee the appropriate assumptions. In particular, assume that we are



**FIGURE 7.**  $x \in X_P((1, 3), (1, 4), (1, 1), (1, 4), (1, 2), (2, 1), (2, 2))$ . The full line shows the orbit of the PA evolution, the dashed line that of the JPA evolution of  $x$ . The two orbits meet again after seven iterations.

at some state  $\varsigma_1 \in \{P, J, S\}$ , with matrix  $\beta_t$ . Then either the process terminates (i.e., we may successfully singularize or insert a matrix), or we continue at some prescribed state

$$\varsigma_2 \in \{P, J, S, H1_1, H1_2, H1_3, H2_1, H2_2, H2_3, H2_4\}$$

with some prescribed matrix  $\beta_{t+k}$ ,  $k \geq 1$ .

Again, at state  $\varsigma_2$ , the process either terminates (and we return to state  $\varsigma_1$ , to try again) or leads to another state  $\varsigma_3$ , with similar options. The question now is whether success at  $\varsigma_1$  (almost) always follows after a finite number of additional states. Assume that we want to transform the PA into the JPA. In particular, we are interested in the asymptotic behavior of the following sets:

**Definition 4.3.** Let  $s \geq 0$ . We define

$$\mathcal{G}_s := \{(x_1, x_2) \in [0, 1]^2 : \text{the method terminates after } s \text{ or fewer states}\}$$

and

$$\mathcal{B}_s := [0, 1]^2 \setminus \mathcal{G}_s.$$

Thus  $\mathcal{B}_s$  is the “bad” set with respect to the number  $s$  of states visited. We assume that  $\lambda(\mathcal{B}_s) \rightarrow 0$  as  $s \rightarrow \infty$ , where  $\lambda$  is Lebesgue measure on  $[0, 1]^2$ .

By the construction of the process described in Section 3, it is clear that at each state, there is a set of positive measure where the method does not terminate.

Therefore, the method cannot work for all  $(x_1, x_2) \in [0, 1]^2$ , and their PA evolutions, in finitely many states, but only for almost all  $(x_1, x_2) \in [0, 1]^2$ . The following example is due to F. Schweiger [Schweiger 03].

**Example 4.4.** Consider the point  $(x_1, x_2) = (-2\rho^2 + 11\rho - 4, \rho^2 - 5\rho)$ , where  $\rho$  is the largest root of  $x^3 - 6x^2 + 5x - 1$ . Thus  $\rho \approx 5,05$ ,  $x_1 \approx 0,56$ ,  $x_2 \approx 0,24$ , and  $(x_1, x_2) \in X_P(\overline{(2, 1), (1, 1), (1, 1), (2, 1)})$ . Note that  $x^3 - 6x^2 + 5x - 1$  is the characteristic polynomial of the matrix product  $\beta_P(2, 1)\beta_P(1, 1)\beta_P(1, 1)\beta_P(2, 1)$ .

Since the first matrix is of type  $j_1 = 2$ , to transform the PA evolution of  $(x_1, x_2)$  into its JPA evolution, we start at state  $P$  with matrix  $\beta_2$ . We must not directly apply matrix identity type  $I_1$  due to the extended conditions of Perron ( $A_4 = 1$ ,  $A_2 + C_4 = 1$ ,  $j_4 = j_5 (= j_1) = 2$ , but  $C_5 = 0$ ), but continue with state  $H_{24}$  from matrix  $\beta_3$ , to reduce  $C_4$ . Since  $C_5 (= C_1) = 0$  and  $j_5 (= j_1) \neq j_6 (= j_2)$ , we need to switch the type  $j_6 (= j_2)$ , which is exactly where we started from. Hence the transformation of this particular PA evolution into a JPA evolution cannot be accomplished in finitely many states.

The JPA evolution of  $(x_1, x_2)$  turns out to be periodic again, namely  $(x_1, x_2) \in X_J(\overline{(1, 0), (2, 1), (1, 0), (3, 0)})$  (cf. [Schweiger 03]). We may estimate the approximation exponents  $d_P$  and  $d_J$  using the eigenvalues of the convergence matrices defined by the digits, to obtain  $d_P \approx 1.27$ , while  $d_J \approx 1.26$ . The two algorithms are essentially different in the sense that the corresponding evolutions of  $(x_1, x_2)$  never meet, and they have at most a finite number of convergents in common.

## ACKNOWLEDGMENTS

This paper is part of research done in connection with the project P16964 (Multidimensional Continued Fractions) of the Austrian Science Foundation FWF.

Bernhard Schratzberger, FB Mathematik, Universität Salzburg, Hellbrunnerstrae 34, 5020 Salzburg, Austria  
(bernhard.schratzberger@sbg.ac.at)

Received November 8, 2005; accepted in revised form January 24, 2007.

I would especially like to thank project leader Professor F. Schweiger for stimulating discussions and valuable suggestions and comments.

## REFERENCES

- [Brun 57] V. Brun. "Algorithmes euclidiens pour trois et quatre nombres." *Skand. Mat.-Kongr. Helsinki* 13 (1957), 45–64.
- [Iosifescu and Kraaikamp 02] M. Iosifescu and C. Kraaikamp. *Metrical Theory of Continued Fractions*. Dordrecht: Kluwer Academic Publishers, 2002.
- [Jacobi 68] C. G. J. Jacobi. "Allgemeine Theorie der kettenbruchähnlichen Algorithmen, in welchen jede Zahl aus drei vorhergehenden gebildet wird." *J. Reine Angew. Mathematik* 69 (1868), 29–64.
- [Kraaikamp 91] C. Kraaikamp. "A New Class of Continued Fraction Expansions." *Acta Arith.* 57 (1991), 1–39.
- [Perron 07] O. Perron. "Grundlagen für eine Theorie des Jacobischen Kettenbruchalgorithmus." *Math. Ann.* 64 (1907), 1–76.
- [Podsypanin 77] E. V. Podsypanin. "On a Generalization of the Continued Fraction Algorithm Related with Viggo Brun's Algorithm." *Zap. Nauchn. Semin. Leningr. Otd. Mat. Inst. Steklova* 67 (1977), 184–194.
- [Pustyl'nikov 03] L. D. Pustyl'nikov. "Generalized Continued Fractions and Ergodic Theory." *Russ. Math. Surv.* 58:1 (2003), 109–159 (translation from *Usp. Mat. Nauk* 58:1 (2003), 113–164).
- [Schratzberger 04] B. Schratzberger. "S-Expansions in Dimension Two." *JTNB* 16 (2004), 705–732.
- [Schweiger 78] F. Schweiger. *A modified Jacobi–Perron Algorithm with Explicitly Given Invariant Measure*. New York: Springer-Verlag, 1978.
- [Schweiger 00] F. Schweiger. *Multidimensional Continued Fractions*. Oxford: Oxford University Press, 2000.
- [Schweiger 03] F. Schweiger. "The Brun Algorithm and the Jacobi Algorithm Are Essentially Different." Preprint, Universität Salzburg, 2003.