A COMPARATIVE STUDY OF ALGORITHMS FOR COMPUTING CONTINUED FRACTIONS OF ALGEBRAIC NUMBERS (EXTENDED ABSTRACT)

Richard P. Brent, Alfred J. van der Poorten and Herman J.J. te Riele

1. Introduction

The obvious way to compute the continued fraction of a real number $\alpha > 1$ is to compute a very accurate numerical approximation of α, and then to iterate the well-known truncate-and-invert step which computes the next partial quotient $a = \lfloor \alpha \rfloor$ and the next complete quotient $\alpha' = 1/(\alpha - a)$. This method is called the *basic method*. In the course of this process precision is lost, and one has to take precautions to stop before the partial quotients become incorrect. Lehmer [6] gives a safe stopping-criterion, and a trick to reduce the amount of multi-length arithmetic. Schönhage [12] describes an algorithm for computing the greatest common divisor of u and v and the related continued fraction expansion of u/v in $\mathcal{O}(n \log^2 n \log\log n)$ steps if both u and v do not exceed 2^n.

A disadvantage of this approach is that if we wish to *extend* the list of partial quotients computed from an initial approximation of α, we have to compute a *more accurate* initial approximation of α, compute the new complete quotient using this new approximation *and* the partial quotients already computed from the old approximation, and then extend the list of partial quotients using that new complete quotient (we notice that Shiu in [13, p.1312] incorrectly states that *all* the previous calculations have to be repeated, but the partial quotients computed so far don't have to be recomputed).

Bombieri and Van der Poorten [1], and Shiu [13] have recently proposed a remedy for this problem. They give a formula for computing a rational approximation of the next complete convergent from the first n partial quotients. From that complete convergent about n new partial quotients can be computed. So each step gives an approximate doubling of the number of partial quotients. To start the method, a few partial quotients have to be computed with the basic or indirect method. In [1] this method is proposed for algebraic numbers (which are zeros of polynomials) of degree ≥ 3, whereas Shiu also applies it to more general numbers, namely to transcendental numbers that can be defined as the zero of a function for which the logarithmic derivative at a rational point can be computed with arbitrary precision. This includes numbers like π, $\log \pi$, and $\log 2$. For each of thirteen different numbers, Shiu computes 10000 partial quotients. Their frequency distributions are compared with the one which almost all numbers should obey, according to the Khintchine–Lévy theory [3, 7]. No significant deviations from this theory are reported. Shiu calls his method the *direct method*.

Curiously, Shiu does not refer to what we would call the *polynomial method* for algebraic numbers [2, 5, 11] of degree ≥ 3, which computes the partial quotients of α using *only* the coefficients of its defining polynomial. Moreover, Shiu gives neither implementational

---

1991 *Mathematics Subject Classification.* 11A55, 11K50.

Key words and phrases. Continued fractions, algebraic numbers.

Presented at the *Algorithmic Number Theory Symposium II* (ANTS II), Université Bordeaux I, Talence, France, 18-23 May 1996. Appeared in *Algorithmic Number Theory* (Henri Cohen, editor), *Lecture Notes in Computer Science*, Vol. 1122, Springer-Verlag, Berlin, 1996, 35-47.

details of his direct method, nor of the indirect method mentioned above (which he applies to four numbers which can not be handled with the direct method). He concludes that his direct method is "superior in the sense that the computing times for a modest number of partial quotients using the indirect and the direct method are similar, whereas it becomes prohibitively long for the basic algorithm".

Since this is not quite a reproducible conclusion, and since the polynomial method is not included in Shiu's study, we felt stimulated to produce a more explicit comparison of the various methods. In addition, we have taken the occasion to compute about 200000 partial denominators of six different algebraic numbers.

A second motivation for this study is the use of the continued fraction expansion of certain algebraic numbers in solution methods for certain Diophantine inequalities. For example, in [10] the system of inequalities

$$|x^3 + x^2y - 2xy^2 - y^3| \leq 200 \text{ and } |y| \leq 10^{500}$$

(known to have finitely many integral solution pairs (x, y)) was solved with the help of the computation of a (modest) number of partial quotients of the continued fraction expansion of one of the real roots of the third degree polynomial $x^3 + x^2 - 2x - 1$.

2. Notation and error control

2.1 Notation.

Let α be a real number > 1. The continued fraction expansion of α is defined by

$$\alpha = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cdots}},$$

where $a_i = \lfloor \alpha_i \rfloor$, $\alpha_{i+1} = 1/(\alpha_i - a_i)$, $i = 0, 1, \ldots$, with $\alpha_0 = \alpha$. The positive integers a_0, $a_1, \ldots$ are called the *partial quotients* of α and the real numbers α_i are called the *complete quotients* of α. For simplicity, one writes

$$\alpha = [a_0, a_1, a_2, \ldots] = [a_0, a_1, \ldots, a_n, \alpha_{n+1}],$$

where $\alpha_{n+1} = [a_{n+1}, a_{n+2}, \ldots]$.

If α is rational, say $\alpha = u/v$, then its continued fraction expansion terminates (with some $\alpha_i = 0$) and the basic method is nothing but the Euclidean algorithm for computing the greatest common divisor of u and v.

The rational approximation

$$[a_0, a_1, \ldots, a_n] = \frac{p_n}{q_n}$$

of α is called the n-th *convergent* of α. The numerators and denominators of these approximations can be computed with the following formulas:

$$\left.\begin{aligned} p_{i+1} &= a_{i+1}p_i + p_{i-1} \\ q_{i+1} &= a_{i+1}q_i + q_{i-1} \end{aligned}\right\} \quad i = 0, 1, \ldots,$$

where $p_0 = a_0$, $q_0 = 1$, $p_{-1} = 1$, and $q_{-1} = 0$. In matrix notation, we have

$$\begin{pmatrix} p_{i+1} & p_i \\ q_{i+1} & q_i \end{pmatrix} = \begin{pmatrix} a_0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} a_1 & 1 \\ 1 & 0 \end{pmatrix} \cdots \begin{pmatrix} a_{i+1} & 1 \\ 1 & 0 \end{pmatrix},$$

which implies, by taking determinants, that

$$p_{i+1}q_i - p_iq_{i+1} = (-1)^i, \quad i = 0, 1, \ldots. \tag{1}$$

2.2 Error control.

When we compute the partial quotients a_0, a_1, ... from a numerical approximation $\overline{\alpha}$ of α, we loose precision. The error can be controlled with the help of the following two lemmas. Lemma 1 gives a sufficient condition for $\lfloor\overline{\alpha}\rfloor = \lfloor\alpha\rfloor$ to hold. Lemma 2 gives an upper bound for the relative error in $\overline{\alpha}' = 1/(\overline{\alpha} - \lfloor\overline{\alpha}\rfloor)$ as a function of $\overline{\alpha}$, the relative error in $\overline{\alpha}$, and $\overline{\alpha}'$.

Lemma 1. *Let $\overline{\alpha}$ be a numerical (rational) approximation of α with relative error bounded by δ, i.e., $\overline{\alpha} = \alpha(1+\epsilon)$ with $|\epsilon| < \delta$. If $\delta < 1/2$ and*

$$(\lfloor\overline{\alpha}\rfloor + 1)\,\delta < \overline{\alpha} - \lfloor\overline{\alpha}\rfloor < 1 - (\lfloor\overline{\alpha}\rfloor + 1)\,\delta, \tag{2}$$

then $\lfloor\overline{\alpha}\rfloor = \lfloor\alpha\rfloor$.

Proof. We show that $\lfloor\overline{\alpha}\rfloor < \alpha < \lfloor\overline{\alpha}\rfloor + 1$.

Firstly, since $1 - \delta < 1/(1+\epsilon)$ for $\delta < 1$, we have

$$\overline{\alpha}(1-\delta) < \frac{\overline{\alpha}}{1+\epsilon} = \alpha.$$

Furthermore, we have $\overline{\alpha}\,\delta < (\lfloor\overline{\alpha}\rfloor + 1)\,\delta$ so that, by the left inequality in (2), $\overline{\alpha}\,\delta < \overline{\alpha} - \lfloor\overline{\alpha}\rfloor$, which with the above inequality implies that $\lfloor\overline{\alpha}\rfloor < \overline{\alpha}(1-\delta) < \alpha$.

Secondly, since $1/(1+\epsilon) < 1/(1-\delta)$ for $\delta < \frac{1}{2}$, we have

$$\alpha = \frac{\overline{\alpha}}{1+\epsilon} < \frac{\overline{\alpha}}{1-\delta}.$$

From the right inequality in (2) we have $\overline{\alpha} < (\lfloor\overline{\alpha}\rfloor+1)(1-\delta)$, so that $\overline{\alpha}/(1-\delta) < \lfloor\overline{\alpha}\rfloor+1$. □

Lemma 2. *Suppose the conditions of Lemma 1 hold, and let*

$$\alpha' = \frac{1}{\alpha - \lfloor\alpha\rfloor}, \quad \overline{\alpha}' = \frac{1}{\overline{\alpha} - \lfloor\overline{\alpha}\rfloor}.$$

Then an upper bound for the relative error in $\overline{\alpha}'$ with respect to α' is given by $\overline{\alpha}\,\overline{\alpha}'\delta/(1-\delta)$.

Proof. We have

$$\left|\frac{\overline{\alpha}' - \alpha'}{\alpha'}\right| = \left|\frac{\dfrac{1}{\overline{\alpha} - \lfloor\overline{\alpha}\rfloor} - \dfrac{1}{\alpha - \lfloor\alpha\rfloor}}{\dfrac{1}{\alpha - \lfloor\alpha\rfloor}}\right| = \left|\frac{\alpha - \overline{\alpha}}{\overline{\alpha} - \lfloor\overline{\alpha}\rfloor}\right| =$$
$$= \overline{\alpha}\,\overline{\alpha}'\left|1 - \frac{\alpha}{\overline{\alpha}}\right| = \overline{\alpha}\,\overline{\alpha}'\left|\frac{\epsilon}{1+\epsilon}\right| < \overline{\alpha}\,\overline{\alpha}'\frac{\delta}{1-\delta}.$$

□

An additional way to control the computation is based on the following well-known property of continued fractions which we leave to the reader to prove.

Lemma 3. *If β_0/β_1 and γ_0/γ_1 are rational numbers such that*

$$\frac{\beta_0}{\beta_1} < \alpha < \frac{\gamma_0}{\gamma_1},$$

then as long as the partial quotients of β_0/β_1 and γ_0/γ_1 coincide, they are the partial quotients of α. The first time the partial quotients do not coincide, they form a lower and an upper bound for the correct value.

This suggests Lehmer's method [6] to reduce the amount of multi-precision work. Assuming that we have a very accurate rational approximation u/v of the real number $\alpha > 1$ with very large numbers u and v, we can form a suitable lower and upper bound for u/v by just taking the first ten (say) decimal digits of u and v: if $\lceil \log_{10} u \rceil = k$, take $u_0 = \lfloor u/10^{k-10} \rfloor$ and $v_0 = \lfloor v/10^{k-10} \rfloor$ and choose[1]

$$\beta_0 = u_0, \quad \beta_1 = v_0 + 1, \quad \gamma_0 = u_0 + 1, \text{ and } \gamma_1 = v_0.$$

Now we compute partial quotients $a_0, a_1, \ldots, a_{i_0}$ of γ_0/γ_1 and hence of α as follows:

$$\tag{3} \left.\begin{array}{l} a_i = \lfloor \gamma_i/\gamma_{i+1} \rfloor \\ \gamma_{i+2} = \gamma_i - a_i\gamma_{i+1}, \quad \beta_{i+2} = \beta_i - a_i\beta_{i+1} \\ \textbf{if } \beta_{i+2} < 0 \vee \beta_{i+2} \geq \beta_{i+1} \textbf{then} \\ \quad i_0 = i - 1, \text{ stop} \\ \textbf{endif} \end{array}\right\} \quad i = 0, 1, \ldots$$

Notice that we do *not* have to compute the partial quotients of β_0/β_1 (contrary to what is suggested in [4, p.328]) since as long as $0 \leq \beta_{i+2} < \beta_{i+1}$, we are sure that a_i is also the correct partial quotient of β_0/β_1. After (3) has stopped, we have to update the fraction u/v with the computed partial quotients $a_0, a_1, \ldots, a_{i_0}$. So with a_0 we replace u/v by $v/(u - a_0 v)$. In matrix notation,

$$\begin{pmatrix} u \\ v \end{pmatrix} := \begin{pmatrix} 0 & 1 \\ 1 & -a_0 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix},$$

and in general, for $a_0, \ldots, a_{i_0}$ we have

$$\begin{pmatrix} u \\ v \end{pmatrix} := \begin{pmatrix} 0 & 1 \\ 1 & -a_{i_0} \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & -a_{i_0-1} \end{pmatrix} \cdots \begin{pmatrix} 0 & 1 \\ 1 & -a_0 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}.$$

The product of the 2×2 matrices in the right hand side is built up first, and next it is multiplied by the vector $(u \ \ v)^T$, which is the only high-precision computation.

3. The basic, polynomial and direct methods

In this section we will describe the three methods which we have considered in this study, namely the basic method, the polynomial method, and the direct method, which is derived from Shiu's direct method.

3.1 The basic method. With the notation of Section 2.1, let $\overline{\alpha}_i$ be a rational approximation of α_i with relative error bounded by δ_i. The basic method for computing the continued fraction expansion of $\alpha = \alpha_0$ with safe error control (based on Lemmas 1 and 2) reads as follows.

$$\tag{4} \left.\begin{array}{l} a_i = \lfloor \overline{\alpha}_i \rfloor \\ \textbf{if } (a_i + 1)\,\delta_i < \overline{\alpha}_i - a_i < 1 - (a_i + 1)\,\delta_i \textbf{ then} \\ \quad \overline{\alpha}_{i+1} = 1/(\overline{\alpha}_i - a_i) \\ \quad \delta_{i+1} = \overline{\alpha}_i \overline{\alpha}_{i+1} \delta_i/(1 - \delta_i) \\ \textbf{else} \\ \quad \text{stop} \\ \textbf{endif} \end{array}\right\} \quad i = 0, 1, \ldots$$

[1] If $v_0 = 0$, the first partial quotient of u/v is extremely large, and we have to increase the number of decimal digits in u_0 and v_0 accordingly.

Since the $\overline{\alpha}_i$ are rational numbers, we can use Lemma 3 and (3) to reduce the amount of multi-precision computations. The numbers $(a_i + 1)\,\delta_i$ and δ_{i+1} are computed in (floating-point) single precision. Since (3) works with low-precision approximations γ_i/γ_{i+1} and β_i/β_{i+1} of $\overline{\alpha}_i$, some care has to be taken in the check of the inequalities in (4) and in the computation of δ_{i+1} from δ_i in (4). Here we can use that

$$\frac{\beta_{2i}}{\beta_{2i+1}} < \overline{\alpha}_{2i} < \frac{\gamma_{2i}}{\gamma_{2i+1}},$$

and

$$\frac{\gamma_{2i+1}}{\gamma_{2i+2}} < \overline{\alpha}_{2i+1} < \frac{\beta_{2i+1}}{\beta_{2i+2}}.$$

as long as a_{2i} and a_{2i+1} are the correct partial quotients of $\overline{\alpha}_{2i}$ and $\overline{\alpha}_{2i+1}$, respectively. In the full version of this paper, we will give a complete listing of the algorithmic steps in this method.

From the metric theory of continued fractions it is known [9] that if from d decimal digits of α we can compute p partial quotients of its continued fraction, then for almost all α

$$\lim_{d\to\infty} \frac{p}{d} = \frac{6 \log 2 \log 10}{\pi^2} = 0.970....$$

To illustrate this: Lochs [8] has computed 968 partial quotients of π from its first 1000 decimals.

A disadvantage of the basic method is that if we have computed as many as possible partial quotients from a given initial approximation of α, and if we would like to compute more partial quotients, we have to compute a more accurate initial approximation, use the known partial quotients to compute the last possible complete convergent, and from that extend the list of partial quotients.

3.2 The polynomial method. Let $\alpha > 1$ be an an algebraic number of degree $d > 2$ with defining polynomial $f(x)$ (with integral coefficients), so that $f(\alpha) = 0$. Let $f(x)$ have the following three properties:

(i) its leading coefficient is positive;
(ii) it has a unique root $\alpha > 1$ which is simple, i.e., $f'(\alpha) \neq 0$;
(iii) α is irrational.

The polynomial method [5] for computing the continued fraction expansion of α reads as follows. Let $f_0(x) = f(x)$.

$$\left.\begin{array}{l} a_i = \max\{n \in \mathbb{N},\ f_i(n) < 0\} \\ g_i(x) = f_i(x + a_i) \\ f_{i+1}(x) = -x^d g_i(1/x) \end{array}\right\} \quad i = 0, 1, \ldots \tag{5}$$

It is easy to see that $f_1(x)$ has the same three properties as $f_0(x)$ and that the unique root > 1 of $f_1(x)$ is given by $1/(\alpha - a_0)$. It follows that the unique root > 1 of the polynomial $f_i(x)$ is the i-th complete quotient of the continued fraction expansion of α, and that this algorithm finds the corresponding partial quotients. The time-consuming work lies in the computation of the coefficients of $f_{i+1}(x)$ from those of $f_i(x)$ (which grow with i). The number a_i can be computed quickly as follows. If we write $f_i(x) = c_{id}x^d + c_{i,d-1}x^{d-1} + \ldots$, then the sum of the roots of $f_i(x)$ is given by $s_i = -c_{i,d-1}/c_{id}$. Since, for $i \geq 1$, the $d-1$ roots of $f_i(x)$ which have their real part < 1, are all located in the interval $(-1, 0)$, the number s_i approximates a_i with an error not greater than $d - 1$; the precise value of a_i is found from s_i by trial and error (with an average of $(d-1)/2$ trials).

Similarly to the basic method, the polynomial method may be accelerated [5] by computing several successive partial quotients (with the basic method) from a low-precision approximation of the real root > 1 of $f_n(x)$. Then it is possible to compute $f_{n+m}(x)$ from

$f_n(x)$ and a_n, a_{n+1}, ..., a_{n+m-1} with less computation than is needed to compute all the intermediate polynomials $f_{n+1}(x)$, ..., $f_{n+m-1}(x)$. We have not (yet) pursued this.

An advantage of this method is that the computation can always be continued, without any recomputation, provided that we save the exact integral values of the coefficients of the last used polynomial $f_i(x)$. To illustrate the growth of these, for $f(x) = f_0(x) = x^3-8x-10$, the four coefficients of $f_{100}(x)$ are integers of 68 decimal digits each, and the four coefficients of $f_{1000}(x)$ are integers of 570, 571, 570, and 568 decimal digits, respectively.

3.3 The direct method. The direct method which we formulate here is based on ideas expressed in [1] and [13], combined with error control facilities described in Section 2. The aim is to compute a very good rational approximation of the complete quotient α_{n+1} when the partial quotients $a_0, a_1, \ldots, a_n$ are known, and from that approximation compute about n partial quotients of α_{n+1}. This is done as follows. We have

$$\alpha = [a_0, a_1, \ldots, a_n, \alpha_{n+1}] = \frac{\alpha_{n+1}p_n + p_{n-1}}{\alpha_{n+1}q_n + q_{n-1}},$$

from which we find, using (1), that

$$\alpha_{n+1} = \frac{(-1)^{n-1}}{q_n(p_n - \alpha q_n)} - \frac{q_{n-1}}{q_n}.$$

Now using the mean value theorem and $f(\alpha) = 0$, we replace the difference $\frac{p_n}{q_n} - \alpha$ by $f(\frac{p_n}{q_n})/f'(\frac{p_n}{q_n})$, and obtain the approximation

$$\alpha_{n+1} \approx \frac{(-1)^{n-1}}{q_n^2}\frac{f'(\frac{p_n}{q_n})}{f(\frac{p_n}{q_n})} - \frac{q_{n-1}}{q_n}. \tag{6}$$

The error in this approximation is approximately

$$\frac{|f''(\alpha)|}{q_n^2|f'(\alpha)|}.$$

From this rational approximation of α_{n+1} partial quotients $a_{n+1}, a_{n+2}, \ldots, a_{n+m}, \ldots$ can be computed as long as $q_{n+m} < bq_n^2$, for some small $b = b(\alpha) > 0$. The direct method for computing N partial quotients of the continued fraction expansion of α now reads as follows. *Step 1* Use the basic method (4) to compute a *small* number of partial quotients and the corresponding partial convergents of α, say up to a_n, p_n, q_n.
Step 2 (Check) If $p_nq_{n-1} - p_{n-1}q_n \neq (-1)^{n-1}$ then stop.
Compute the next rational approximation α' of α_{n+1} by

$$\alpha' = \frac{(-1)^{n-1}}{q_n^2}\frac{f'(p_n/q_n)}{f(p_n/q_n)} - \frac{q_{n-1}}{q_n}. \tag{7}$$

Let $B = bq_n^2$ for some suitable constant $b = b(\alpha)$.
Compute the next partial quotients a_{n+1}, $a_{n+2}, \ldots, a_{n+m}, \ldots$ with the basic method (4) (using Lemma 3 and (3)) as long as $n + m \leq N$ and $q_{n+m} < B$.
Step 3 Put $n = n + m$; if $n < N$ go back to Step 2.

The number of partial quotients which can be computed in Step 2 is roughly equal to n so that *after* the completion of Step 2, the number of partial quotients computed has roughly been doubled compared with *before* Step 2. Since (7) is very time-consuming, it is worthwhile to choose n in Step 1 such that the last time Step 2 is carried out, it starts with a value of n which is *slightly larger* than $N/2$. Since in the beginning of the method the behaviour of Step 2 may be rather erratic, one should compute so many partial quotients of α in Step 1 that the "stable" behaviour phase of Step 2 (an approximate doubling of the number of partial quotients) is reached. In practice, this works for $n \approx 100$, but this also depends on the sizes of the first partial quotients of the continued fraction of α.

4. Experiments

We have implemented the three methods described in Section 3 on a SUN workstation, partially in GP/PARI and partially in Magma. The first package is developed by Henri Cohen and his co-workers at Université Bordeaux I, the second comes from John Cannon and his group at the University of Sydney. Initially, we only worked with GP, but at a certain point in the direct method we ran into problems with the stack size, due to the enormous size of the integers involved in this method. Later we learned that these problems can be solved, for example, by programming PARI in Library Mode, but in the meantime we learned about the Magma-package at the University of Sydney and decided to experiment with that. With Magma we did not encounter any stack problems.

In Table 1 we give some timings with Magma and GP for the basic, the polynomial, and the direct methods. Based on these results, we decided to run bigger experiments with our Magma implementation of the direct method.

In Table 2 we give the frequency distributions of the first 200001 partial quotients of the continued fraction of six algebraic numbers, computed with the direct method. For comparison, the last column gives the frequencies of occurrence of partial quotients j:

$$\log_2\left(1+\frac{1}{j}\right)-\log_2\left(1+\frac{1}{j+1}\right)$$

from the well-known Gauss-Kusmin Theorem. Let

$$K(\alpha,n)=(a_0a_1\ldots a_n)^{1/(n+1)}$$

and

$$L(\alpha,n)=q_n^{1/(n+1)}.$$

Then for almost all α

$$\lim_{n\to\infty}K(\alpha,n)=\prod_{k=1}^{\infty}\left(1+\frac{1}{k(k+2)}\right)^{\log k/\log 2}=2.68545...,$$

and

$$\lim_{n\to\infty}L(\alpha,n)=\exp\left(\frac{\pi^2}{12\log 2}\right)=3.27582....$$

The latter result implies that for almost all α the number of decimal digits in q_n is about $n\log_{10}L\approx 0.515n$. Table 2 gives the values of $K(\alpha,200000)$ for the six algebraic numbers which we considered. Table 2 also lists the largest partial quotient a_n found, and the corresponding index n. Only in case (A) there is an early occurrence of a large partial quotient ($a_{121}=16467250$), but soon after that, no extremely large partial quotients occur anymore. To illustrate this, Table 3 lists a_n for $n=0,\ldots,200$ and for $n=199901,\ldots,200000$. The "abnormal" initial behaviour is explained in [14]. Table 4 presents, for some values of n, the number of decimal digits in q_n and that number divided by n. The values of n in Table 4 are those for which the direct method computed a new rational approximation of α_n: it illustrates the approximate doubling of these n-values, especially for larger values of n. The last column shows good convergence to the value $\pi^2/(12\log 2\log 10)=0.51532....$

5. Conclusion

We have compared three different methods (the basic, the polynomial, and the direct method) for computing the continued fraction expansion of algebraic numbers, and observed that the direct method is the most efficient one in terms of CPU-time and memory, at least for our implementations (in GP/PARI and Magma). We have applied the direct method to the computation of 200,001 partial quotients of six different algebraic numbers, and found no apparent deviation from the theory of Khintchine and Lévy, which holds for almost all real numbers.

Acknowledgment

Part of this research was carried out while the third author was visiting the ce*NTR*e for *N*umber *T*heory *R*esearch at Macquarie University, Sydney, and the Computer Science Laboratory of the Australian National University, Canberra, in September–November 1995. He thanks the first and the second author for their hospitality. We also like to acknowledge the help of Wieb Bosma of the University of Sydney with the use of the Magma package.

References

1. Enrico Bombieri & Alfred J. van der Poorten, *Continued fractions of algebraic numbers*, Computational Algebra and Number Theory, Sydney 1992 (Wieb Bosma & Alf van der Poorten, ed.), Kluwer, 1995, pp. 137–152.
2. David G. Cantor, Paul G. Galyean & Horst G. Zimmer, *A continued fraction algorithm for real algebraic numbers*, Mat. Comp. **26** (1972), 785–791.
3. A. Khintchine, *Metrische Kettenbruchprobleme*, Compositio Math. **1** (1935), 361–382.
4. Donald E. Knuth, *The Art of Computer Programming, Volume 2, Seminumerical Algorithms*, Addison-Wesley, Reading, Mass., Second Edition, 1981.
5. Serge Lang & Hale Trotter, *Continued fractions for some algebraic numbers*, J. reine angew. Math. **255** (1972), 112–134.
6. D.H. Lehmer, *Euclid's algorithm for large numbers*, The Amer. Math. Monthly **45** (1938), 227–233.
7. P. Lévy, *Sur le développement en fraction continue d'un nombre choisi au hasard*, Compositio Math. **3** (1936), 286–303.
8. Gustav Lochs, *Die ersten 968 Kettenbruchnenner von* π, Monatsh. Math. **67** (1963), 311–316.
9. Gustav Lochs, *Vergleich der Genauichkeit von Dezimalbruch und Kettenbruch*, Abh. Math. Seminar Hamburg **27** (1964), 142–144.
10. Attila Pethö, *On the resolution of Thue inequalities*, J. Symb. Comp. **4** (1987), 103–109.
11. R.D. Richtmyer, Marjorie Devaney & N. Metropolis, *Continued fraction expansions of algebraic numbers*, Numer. Math. **4** (1962), 68–84.
12. A. Schönhage, *Schnelle Berechnung von Kettenbruchentwicklungen*, Acta Informatica **1** (1971), 139–144.
13. P. Shiu, *Computation of continued fractions without input values*, Math. Comp. **64** (1995), 1307–1317.
14. H.M. Stark, *An explanation of some exotic continued fractions found by Brillhart*, Computers in Number Theory (A.O.l. Atkin & B.J. Birch, ed.), Academic Press, 1971, pp. 21–35.

TABLE 1 *CPU-time in seconds to compute 10000 partial quotients of the root* $\alpha = 3.31862...$ *of* $f(x) = x^3 - 8x - 10$.

	basic	polynomial	direct
with Magma	98		55
with GP		192	172

TABLE 2 *Frequency distribution of the first 200001 partial quotients of the continued fraction of various algebraic numbers.*

(A)	$f(x) = x^3 - 8x - 10,$	$\alpha = 3.31862...$
(B)	$f(x) = x^3 - 2,$	$\alpha = 1.25992... = 2^{1/3}$
(C)	$f(x) = x^3 - 5,$	$\alpha = 1.70997... = 5^{1/3}$
(D)	$f(x) = x^4 + 6x^3 + 7x^2 - 6x - 9$	$\alpha = 1.03224...$ $= (5^{1/2} - 1)/2 + 2^{1/2} - 1$ $= [0, \overline{1}\,] + [0, \overline{2}\,]$
(E)	$f(x) = x^3 + x^2 - 2x - 1$	$\alpha = 1.24697... = 2\cos(2\pi/7)$
(F)	$f(x) = x^6 - 9x^4 - 4x^3 + 27x^2 - 36x - 23$	$\alpha = 2.99197... = 2^{1/3} + 3^{1/2}$

digit	(A)	(B)	(C)	(D)	(E)	(F)	"expected"
1	82705	82862	83186	82865	83159	82566	83008
2	34277	34180	33883	34538	33900	34382	33985
3	18641	18680	18570	18588	18560	18616	18622
4	11693	11795	11785	11503	11835	11931	11779
5	8192	8114	8165	8114	8070	8083	8128
6	6082	5900	5864	5880	5826	5916	5949
7	4470	4443	4535	4512	4519	4532	4544
8	3470	3636	3557	3540	3671	3594	3584
9	2862	2841	2975	2896	2866	2911	2900
10	2474	2424	2329	2400	2428	2347	2395
>10 and ≤ 100	22156	22240	22298	22309	22302	22230	22264
>100	2979	2886	2854	2856	2865	2893	2843
K	2.6944	2.6871	2.6832	2.6848	2.6844	2.6919	2.68545
$\max(a_n)$	16467250	320408	489859	7295890	179545	1075748	
index n	121	190270	21125	142839	44595	52062	
CPU-time in minutes	109	99	107	196	113	424	

Table 3 *Some partial quotients a_n of the real root $\alpha = 3.31861...$ of $x^3 - 8x - 10$ (those $>$ 10000 are underlined).*

n	a_n
0	3
1 − 20	3 7 4 2 30 1 8 3 1 1 1 9 2 2 1 3 <u>22986</u> 2 1 32
21 − 40	8 2 1 8 55 1 5 2 28 1 5 1 <u>1501790</u> 1 2 1 7 6 1 1
41 − 60	5 2 1 6 2 2 1 2 1 1 3 1 3 1 2 4 3 1 <u>35657</u> 1
61 − 80	17 2 15 1 1 2 1 1 5 3 2 1 1 7 2 1 7 1 3 25
81 − 100	<u>49405</u> 1 1 3 1 1 4 1 2 15 1 2 83 1 162 2 1 1 1 2
101–120	2 1 <u>53460</u> 1 6 4 3 4 13 5 15 6 1 4 1 4 1 1 2 1
121–140	<u>16467250</u> 1 3 1 7 2 6 1 95 20 1 2 1 6 1 1 8 1 <u>48120</u> 1
141–160	2 17 2 1 2 1 4 2 3 1 2 23 3 2 1 1 1 2 1 27
161–180	<u>325927</u> 1 60 1 87 1 2 1 5 1 1 1 2 2 2 2 2 17 4 9
181–200	9 1 7 11 1 2 9 1 14 4 6 1 22 11 1 1 1 1 4 1
199901–199920	1 1 2 1 2 1 3 3 3 4 2 2 1 1 1 1 1 8 11 1
199921–199940	10 1 4 1 1 2 4 2 1 1 3 5 1 6 1 2 7 7 50 1
199941–199960	2 12 1 1 1 3 1 7 1 6 2 2 1 1 2 3 2 6 1 3
199961–199980	1 3 10 1 1 1 3 1 6 2 1 16 1 1 1 1 8 1 1 1
199981–200000	2 5 1 1 2 1 2 2 3 2 1 1 5 2 5 1 1 1 268 1

Table 4 *Sizes of q_n for the continued fraction expansion of the real root $\alpha = 3.31862...$ of $x^3 - 8x - 10$.*

n	# dec. digits of q_n	# dec. digits/n
301	192	0.638
654	380	0.581
1347	752	0.558
2830	1499	0.530
5667	2992	0.528
11502	5977	0.520
22982	11945	0.520
46208	23880	0.517
92514	47754	0.516
		$0.51532...=\frac{\pi^2}{12\log 2\log 10}$

Computer Sciences Laboratory, Research School of Information Sciences and Engineering, Australian National University, Canberra, ACT 0200 Australia
E-mail address: Richard.Brent@anu.edu.au

ceNTRe for Number Theory Research, School of Mathematics, Physics, Computing and Electronics, Macquarie University, NSW 2109 Australia
E-mail address: alf@macadam.mpce.mq.edu.au

CWI, Department of Numerical Mathematics, Kruislaan 413, 1098 SJ Amsterdam, The Netherlands
E-mail address: herman@cwi.nl