An Efficient Representation of the Integers for the Distribution of Partial Quotients over the Continued Fractions

HIDETOSHI YOKOO*

A binary representation of the positive integers is proposed which is close to optimal for the distribution of partial quotients over the continued fraction expansions of rational numbers. The average length of the proposed representation is about 1% greater than the entropy of the distribution. This representation is order-preserving in the sense that the lexicographic ordering of the bit strings corresponds to numeric ordering of the values. As a result, it can be applied to lexicographic continued fraction representation for an internal representation of real numbers in computers.

1. Introduction

The problem, called a representation or an encoding of the integers, is one of the most fundamental and important problems of information sciences. In Turing machine theory, it is required to represent integers separably as sequences of symbols on the tape.

In information theory, the objective of source coding is to find a particular representation for a certain source so as to minimize the expected length of the representation. In general, however, there is no constructive way for infinite prefix-free encoding of the integers, so that the notions of universality and asymptotic optimality are introduced instead and the performance of the representations is discussed in terms of such criteria [1]. In particular, asymptotically optimal representations have been presented in [2], [7], which reveal that the quantity $\log_2^* i = \log_2 i + \log_2 \log_2 i + \ldots$ (only the positive terms are included in the sum.) plays an important role for the distribution of integer i.

While these discussions on the asymptotic optimality are mainly conducted by the theoretical interests, specific representations for some particular distributions on the integers become important in practice. In run-length encodings [4] for the compression of binary images, geometric distributions are assumed for the integers. Humblet [5] proposes an optimal representation of the integers whose distribution is Poisson. The particular case that we deal with in this paper is the distribution of partial quotients over the continued fraction expansions of rational numbers [8]. This paper proposes an efficient representation of the integers for that distribution. Since the proposed representation has another advantage that lexicographic order of the codewords corresponds to the numeric order of the integers, it can be applied to lexicographic continued fraction representation of rational numbers which is introduced by Matula & Kornerup [9] as an internal representation of real numbers in computers.

In the next section, we give a formal definition of a representation of the integers and describe the distribution of partial quotients over the continued fraction expansions of rational numbers. Section 3 presents the new representation of the integers, whose efficiency is explained in comparison with an alternative simpler representation of the integers. Section 4 gives remarks on the application of the proposed representation to the lexicographic continued fraction representation as a computer internal representation of real numbers.

In the sequel, $\log x$ and $\ln x$ mean $\log_2 x$ and $\log_e x$, respectively.

2. Preliminaries

2.1 Representation of the Integers

In the following, a source (Z^+, P) is a set Z^+ of the positive integers and a probability distribution function $P: Z^+ \to (0, 1]$ that assigns a positive probability $P(j) > 0$ to each integer $j \in Z^+$.

Let $B = \{0, 1\}$ be the code alphabet and B^* denote the set of all finite sequences of symbols, each symbol selected from the set B. A representation of the integers is a codeword set $C \subset B^*$ and a one-to-one encoding function f: $Z^+ \to C$, which assigns a distinct codeword $f(j) \in C$ to each integer j. The set C and the encoding function f are said to be uniquely decipherable if for each sequence of integers, the sequence of codewords corresponding to that integer sequence is different from the sequence of codewords corresponding to any other integer sequence.

*Department of Electrical Engineering, Faculty of Engineering, Yamagata University, Yonezawa 992, Japan.

A uniquely decipherable set $C \subset B^*$ is said to be complete iff adding any new sequence $c' \in B^*$, $c' \notin C$, to C gives a set $C' = C \cup \{c'\}$ that is not uniquely decipherable.

Let c be a codeword in C of length l and represented by $c = c_1 c_2 \ldots c_l$, where $c_1, c_2, \ldots, c_l$ denote the individual symbols in B. Any initial part of c, that is, $c_1 c_2 \ldots c_m$ for some $m \leqq l$ is called a prefix of c. A prefix condition set is defined as a set in which no codeword is the prefix of any other codeword. A prefix condition set is uniquely decipherable, as is easily shown.

A codeword $c = c_1 c_2 \ldots c_l$ is said to be lexicographically smaller than a codeword $d = d_1 d_2 \ldots d_m$ if $c_1 < d_1$; or if $c_i = d_i$ for $i < n$ and $c_n < d_n$ for some $n \leqq l, m$; or if $c_i = d_i$ for $1 \leqq i \leqq l$ and $l < m$. The encoding function f is said to be order-preserving iff $f(i)$ is lexicographically smaller than $f(j)$ for $i < j$.

The length function L of the set C is defined by

$$L(j) = |f(j)|,$$

where $|f(j)|$ denotes the length of the codeword $f(j)$.

Our principal objective in this paper is to find a set C and an encoding function f, which satisfy the following conditions:

(a) C is a prefix condition set;
(b) f is order-preserving;

and minimize the average value of the function L with respect to the given P,

$$E_P(L) = \sum_{j=1}^{\infty} P(j) L(j). \quad (1)$$

If a code has the minimum average codeword length, the minimum being taken over all uniquely decipherable codes, then the code is said to be optimal. It is known from information theory [3] that if the source (Z^+, P) has a finite value of entropy

$$H(P) = -\sum_{j=1}^{\infty} P(j) \log P(j), \quad (2)$$

$E_P(L)$ is lower bounded by $H(P)$.

2.2 Distribution of Partial Quotients over the Continued Fractions

Utilizing the notation $/a_1, a_2, a_3, \ldots/$ for the continued fraction

$$\cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \ddots}}},$$

where the partial quotients a_i's are assumed to be integers, every rational number v/u in the range $0 < v/u < 1$ has a finite expansion

$$\frac{v}{u} = /a_1, a_2, \ldots, a_m/, \quad (3)$$

which is unique with the added requirement $a_m \geqq 2$. For the distribution of partial quotients when (3) is uniformly distributed in the range, it can be demonstrated [8] that, for sufficiently large i, the probability that any particular partial quotient a_i takes a specific value j depends only on j and is given by

$$P(j) = \log \frac{(j+1)^2}{j(j+2)}. \quad (4)$$

This satisfies the requirement

$$\sum_{j=1}^{\infty} P(j) = 1,$$

and has the entropy

$$H(P) = -\sum_{j=1}^{\infty} \log \frac{(j+1)^2}{j(j+2)} \log \log \frac{(j+1)^2}{j(j+2)} \doteqdot 3.45, \quad (5)$$

which is numerically evaluated.

3. Proposed Representation

3.1 Primitive Method

It seems so difficult to find an efficient representation of the integers, in the sense that it minimizes the average length (1) for the distribution (4), which satisfies both the requirements (a) and (b) in section 2.1. However, we see that the following primitive method not only satisfies the requirements but also is fairly efficient.

A full k-bit integer is a positive integer j whose ordinary binary representation contains k bits with leading bit 1, so then $2^{k-1} \leqq j \leqq 2^k - 1$. The codeword of the primitive method for full k-bit integer j is formed from the ordinary binary expansion simply by replacing the leading 1 by a sequence of $(k-1)$ 1's and a zero [7], [9]. That is, if the ordinary binary expansion of j is 1α where α is any sequence of 0's and 1's, the representation of j by the primitive method is

$$1^{|\alpha|} 0 \alpha,$$

where $|\alpha|$ denotes the length of α and the part α may be called a postfix of the representation.

This method has the length function

$$L(j) = 2 \lfloor \log j \rfloor + 1,$$

($\lfloor x \rfloor$ means the greatest integer not greater than x.) and the average representation length

$$E_P(L) = \sum_{j=1}^{\infty} (2 \lfloor \log j \rfloor + 1) \log P(j) \doteqdot 3.51,$$

which is less than 2% greater than the entropy (5).

3.2 Proposed Method

Our new method may be considered as an improvement of the primitive method, because the new representation has the same leading part $1^{|\alpha|}0$ for an integer j whose ordinary binary expansion is 1α. The encoding function of the proposed method is represented as

$$f(j)=1^{|\alpha|}0\beta(j),$$

where $\beta(j)\in B^*$ is called the postfix of a codeword $f(j)$. In the proposed method, $\beta(1)$ is an empty string, and for $j\geqq 2$, $\beta(j)$ consists of a leading bit $\beta_0(j)$ and the rest $\beta'(j)$, that is,

$$\beta(j)=\beta_0(j)\beta'(j).$$

For a full k-bit integer j, let

$$\mu(k)=\frac{1}{3}(2^{k-1}-(-1)^{k-1}), \tag{6}$$

which is always an integer for $k\geqq 1$. Then the leading bit $\beta_0(j)$ is set to 0 for $j<2^{k-1}+\mu(k)$ and to 1 for $j\geqq 2^{k-1}+\mu(k)$. Thus, among all full k-bit integers, the first $\mu(k)$ codewords have 0 for $\beta_0(j)$ and the rest $2^{k-1}-\mu(k)$ codewords have 1 for $\beta_0(j)$. Note that

$$2^{k-1}-\mu(k)=\mu(k+1). \tag{7}$$

In $j\geqq 2^{k-1}+\mu(k)$, the part $\beta'(j)$ is defined as an ordinary binary expansion of $j-2^{k-1}-\mu(k)$ of length $(k-2)$ bits for $j<2^{k-1}+2\mu(k)$, and as an ordinary binary expansion of $j-2^{k-1}$ of length $(k-1)$ bits for $j\geqq 2^{k-1}+2\mu(k)$. For $j<2^{k-1}+\mu(k)$, $\beta'(j)$ is recursively defined as

$$\beta'(j)=\beta'(j-\mu(k)). \tag{8}$$

Table 1 gives the representations of the primitive method and the proposed method. In the table, copying processes correspond to the facts (7) and (8).

It is easy to see that the proposed representation satisfies the requirements (a) and (b), and also that the codeword set is complete. Therefore, it is possible not only to represent any sequence of positive integers by the finite bit sequence, but also to interpret any infinite binary sequence as the infinite sequence of positive integers. An interpretation algorithm is given as follows, which can decode an input binary representation into an integer j by a left-to-right scan.

Interpretation procedure

I1. Set $k\leftarrow 1$, $s\leftarrow$input symbol $\in\{0, 1\}$.
I2. While $s=1$, repeatedly set $k\leftarrow k+1$, $s\leftarrow$input symbol.
I3. If $k=1$, set $j\leftarrow 1$ and end.
I4. Set $s\leftarrow$input symbol.
I5. If $s=0$, set $j\leftarrow\mu(k)+R(k-1)$.
Otherwise, set $j\leftarrow R(k)$.
I6. End.

Function $R(k)$
R1. Set $R\leftarrow 1$.
R2. If $k=1$, return.
R3. Repeatedly $(k-2)$ times set
$s\leftarrow$input symbol,
$R\leftarrow 2R+s$.
R4. Set $R\leftarrow R+\mu(k)+2^{k-2}$.
R5. If $R\geqq 2^{k-1}+2\mu(k)$, set $s\leftarrow$input symbol,
$R\leftarrow 2R-2\mu(k)-2^{k-1}+s$.
R6. Return.

Table 1 The primitive representation and the proposed representation of the integers 1, 2, . . . , 21. A space between the leading part and the postfix is inserted for clarity.

k	integer j	Primitive	Proposed
1	1	0	0
2	2	10 0	10 0
	3 $(=2^1+\mu(2))$	10 1	10 1
3	4 $(=2^2)$	110 00	110 0
	5 $(=2^2+\mu(3))$	110 01	110 10
	6 $(=2^2+2\mu(3))$	110 10	110 110
	7	110 11	110 111
4	8 $(=2^3)$	1110 000	1110 00
	9	1110 001	1110 010
	10	1110 010	1110 011
	11 $(=2^3+\mu(4))$	1110 011	1110 100
	12	1110 100	1110 101
	13	1110 101	1110 110
	14 $(=2^3+2\mu(4))$	1110 110	1110 1110
	15	1110 111	1110 1111
5	16 $(=2^4)$	11110 0000	11110 000
	17	11110 0001	11110 001
	18	11110 0010	11110 010
	19	11110 0011	11110 0110
	20	11110 0100	11110 0111
	21 $(=2^4+\mu(5))$	11110 0101	11110 1000

3.3 Near Optimality

Unfortunately, we have not yet proved or disproved the optimality of the proposed representation, but it is easy to show that the proposed method is more efficient than the primitive method for the distribution (4). This will be shown below after giving an explanation why the primitive method is so efficient.

First, since $P(1)=0.415\ldots$, the integer 1 should be coded by a single bit, i.e., $|L(1)|=1$ (e.g., see [6]).

Next, noting that full k-bit integers have all the same representation length $2k-1$ in the primitive method, let us consider the source (Z^+, Q) with

$$Q(j)=2^{-(2k-1)} \quad \text{for} \quad 2^{k-1}\leqq j\leqq 2^k-1,\ k=1, 2, \ldots.$$

Note that

$$\sum_{j=1}^{\infty} Q(j)=1$$

and that the primitive method is optimal for this source. Then, we can show that the distribution Q is close to (4) in the sense that the inequalities

$$P(2^{k-1})>Q(j)>P(2^k-1) \tag{9}$$

hold for $k\geqq 2$. In order to prove the first inequality of (9), i.e.,

$$\log\frac{(2^{k-1}+1)^2}{2^{k-1}(2^{k-1}+2)}>2^{-(2k-1)}, \qquad (10)$$

we set $y=2^{k-1}$ for notational convenience, then $y\geqq 2$ for $k\geqq 2$. According to a useful formula

$$\ln(1+x)\leqq x \quad \text{for} \quad x>-1 \qquad (11)$$

and its variation

$$-\ln(1-x)\geqq x \quad \text{for} \quad x<1, \qquad (12)$$

we have

$$\begin{aligned}\ln\frac{(2^{k-1}+1)^2}{2^{k-1}(2^{k-1}+2)}&=\ln\frac{(y+1)^2}{y(y+2)}\\ &=-\ln\left(1-\frac{1}{(y+1)^2}\right)\\ &\geqq\frac{1}{(y+1)^2} \qquad (13)\end{aligned}$$

since $0<1/(y+1)^2\leqq 1/9$ for $y\geqq 2$. Further, we have

$$\begin{aligned}\frac{1}{(y+1)^2}-\frac{\ln 2}{2y^2}&=\frac{1}{2y^2(y+1)^2}\{(2-\ln 2)y^2-2\ln 2\cdot y-\ln 2\}\\ &>0 \quad \text{for} \quad y\geqq 2.\end{aligned}$$

This and (13) prove (10).

We proceed to the proof of the last inequality of (9), i.e.,

$$2^{-(2k-1)}>\log\frac{2^{2k}}{(2^k-1)(2^k+1)}. \qquad (14)$$

If we set $z=2^k$, then $z\geqq 4$ for $k\geqq 2$. According to (11),

$$\begin{aligned}\ln\frac{2^{2k}}{(2^k-1)(2^k+1)}&=\ln\frac{z^2}{(z-1)(z+1)}\\ &=\ln\left(1+\frac{1}{z^2-1}\right)\\ &\leqq\frac{1}{z^2-1} \qquad (15)\end{aligned}$$

since $0<1/(z^2-1)\leqq 1/15$ for $z\geqq 4$. Combining (15) with

$$\begin{aligned}\frac{2\ln 2}{z^2}-\frac{1}{z^2-1}&=\frac{(2\ln 2-1)z^2-2\ln 2}{z^2(z^2-1)}\\ &>0 \quad \text{for} \quad z\geqq 4,\end{aligned}$$

we have

$$\ln\frac{2^{2k}}{(2^k-1)(2^k+1)}<\frac{2\ln 2}{z^2},$$

which proves (14). Thus, we have evaluated the efficiency of the primitive method by the inequalities (9).

In order to see that the proposed method is an improvement of the primitive method, it is sufficient to compare the average postfix lengths of both representations, because the leading part $1^{|\alpha|}0$ is common in both representations. In the primitive method, full k-bit integers have all the same postfix length: k-1. The average postfix length of the full k-bit integers is

$$\begin{aligned}L_{\text{prim}}(k)&=\sum_{j=2^{k-1}}^{2^k-1}P(j)(k-1)\\ &=(k-1)\log\frac{2^k+2}{2^k+1}.\end{aligned}$$

In the proposed method, on the other hand, when we denote the postfix length for an integer j as $l(j)$, then

$$l(j)=\begin{cases}0 & \text{for } j=0,\\ k-2 & \text{for } 3\leqq k \text{ and } 2^{k-1}\leqq j\leqq 2^{k-1}+\mu(k-1)-1,\\ k-1 & \text{for } 2\leqq k \text{ and } 2^{k-1}+\mu(k-1)\leqq j\leqq 2^{k-1}+2\mu(k)-1,\\ k & \text{for } 3\leqq k \text{ and } 2^{k-1}+2\mu(k)\leqq j\leqq 2^k-1.\end{cases}$$

Let $L_{\text{prop}}(k)$ be the average postfix length of full k-bit integers in the proposed representation and let

$$\begin{aligned}\zeta(k)&=\frac{(2^{k-1}+1)(2^{k-1}+\mu(k-1))}{2^{k-1}(2^{k-1}+\mu(k-1)+1)},\\ \eta(k)&=\frac{(2^{k-1}+\mu(k-1)+1)(2^{k-1}+2\mu(k))}{(2^{k-1}+\mu(k-1))(2^{k-1}+2\mu(k)+1)},\\ \xi(k)&=\frac{(2^{k-1}+2\mu(k)+1)2^k}{(2^{k-1}+2\mu(k))(2^k+1)}.\end{aligned}$$

Then

$$L_{\text{prop}}(k)=\begin{cases}0 & \text{for} \quad k=1,\\ (k-2)\log\zeta(k)+(k-1)\log\eta(k)+k\log\xi(k) & \text{for} \quad k\geqq 2.\end{cases}$$

Since $L_{\text{prim}}(k)$ can be expanded into

$$L_{\text{prim}}(k)=(k-1)\{\log\zeta(k)+\log\eta(k)+\log\xi(k)\},$$

we have

$$L_{\text{prim}}(k)-L_{\text{prop}}(k)=\begin{cases}0 & \text{for } k=1,\\ \log\zeta(k)-\log\xi(k) & \text{for } k\geqq 2.\end{cases}$$

If we set $z=2^k$ and $d=(-1)^k$ for $k\geqq 2$, then

$$\zeta(k)=\frac{(z+2)(7z-4d)}{z(7z-4d+12)},$$

$$\xi(k)=\frac{(5z+4d+6)z}{(5z+4d)(z+1)}$$

and

$$\zeta(k)-\xi(k)=\frac{3z^3-2z^2+(16d-48)z-32}{(7z-4d+12)(5z+4d)(z^2+z)}$$

$$=\begin{cases}\dfrac{3z^3-2z^2-32z-32}{(7z+8)(5z+4)(z^2+z)} & \text{if } k \text{ is even},\\[2ex] \dfrac{3z^3-2z^2-64z-32}{(7z+16)(5z-4)(z^2+z)} & \text{if } k \text{ is odd}.\end{cases} \quad (16)$$

Since the denominators of (16) are positive for $z\geqq 1$ and the numerators $3z^3-2z^2-32z-32$ and $3z^3-2z^2-64z-32$ are positive for $z\geqq 4$ and $z\geqq 5.18\ldots$, respectively, then we have

$$\zeta(k)\geqq\xi(k) \quad \text{for} \quad k\geqq 2.$$

Thus,

$$L_{\text{prim}}(k)\geqq L_{\text{prop}}(k)$$

holds for all $k\geqq 1$.

By numerical computation, the average representation length of the proposed method is shown to be 3.49, which is about 1% greater than the entropy (5).

4. An Application to the Lexicographic Continued Fraction Representation of Rational Numbers

An efficient order-preserving representation of the real numbers is considered, where every rational number has a finite representation length. Such a representation is important in that, for many applications, exact arithmetic operations between rational numbers are feasible, and that theoretically the behavior of the representation length of rational numbers could be investigated. Lexicographic continued fraction (LCF) representation introduced in [9] is one of the representations which provide the above properties. In this section, brief comparisons are made between the original LCF−representation and the LCF−representation which utilizes the proposed method for the representation of the integers.

Let C and f be a codeword set and an encoding function which realize the requirements (a) and (b) respectively, and $f^c(j)$ be the complement of $f(j)$ which satisfies

$$|f^c(j)|=|f(j)|,$$

$$f^c(j)\oplus f(j)=1^{|f(j)|},$$

where $\oplus$ denotes the bitwise XOR.

Noting that a continued fraction (3) is equal to $/a_1, a_2, \ldots, a_m-1, 1/$ and that the continued fraction increases or decreases when the partial quotient a_i increases, according as i is even or odd, the LCF-representation corresponding to the continued fraction which is unique in terminal index even form $v/u=/a_1, a_2, \ldots, a_{2n}/$ is defined by

$$\text{LCF}\left(\frac{v}{u}\right)=\begin{cases}0.0 & \text{for } v=0,\\ 0.f^c(a_1)f(a_2)\ldots f^c(a_{2i-1})f(a_{2i})\ldots f^c(a_{2n-1})f(a_{2n}) & \text{for } 0<v<u,\end{cases}$$

which can be extended to the right with an arbitrary number of zeros*. In this definition, the odd indexed quotients are represented in complement form so that an order-preserving representation is obtained. The LCF-representation can be naturally extended to the positive and negative real numbers. A fuller description can be found in the original paper [9].

Consider, for example, the LCF-representation of 26/161, which has the ordinary binary expansion $0.\dot{0}010100101\ 0101110111\ 0100000100\ 11\dot{0}$ (repeating indefinitely) and the continued fraction form:

$$\frac{26}{161}=/6, 5, 5/ \qquad \text{(terminal index is odd)}$$

$$=/6, 5, 4, 1/ \quad \text{(terminal index is even)}.$$

If we use the primitive method for C and f, which is the case in [9], then

$$\text{LCF}\left(\frac{26}{161}\right)=0.f^c(6)f(5)f^c(4)f(1)$$
$$=0.00101\cdot 11001\cdot 00111\cdot 0$$
$$=0.001011100100111.$$

If we use the proposed representation instead, we have

$$\text{LCF}\left(\frac{26}{161}\right)=0.f^c(6)f(5)f^c(4)f(1)$$
$$=0.001001\cdot 11010\cdot 0011\cdot 0$$
$$=0.001001110100011.$$

These encoding processes are cleary reversible. Specifically, by extending any finite bit sequence with an sequence of zeros, a left-to-right scan uniquely determines a terminal index even continued fraction.

*The decimal point "." for the fixed-point representation, which is not used in the original LCF-representation, is here used for an analogy of the ordinary binary expansion.

While these representations have fine and interesting properties, they obviously have the disadvantage that the approximation error obtained by rounding to k-bit LCF-representations is not uniform. In other words, the gap sizes between neighboring k-bit representable numbers are not uniform. Of course, this problem never arises in the ordinary binary representation, which doesn't provide for exact representation of rational numbers, however.

If an irreducible fraction v/u has an LCF-representation

$$\mathrm{LCF}\left(\frac{v}{u}\right)=0.b_1b_2b_3\ldots$$

and $b_i=0$ for all $i>k$, then the fraction v/u is said to be a k-bit continued fraction. There are 2^k distinct k-bit continued fractions in the interval [0, 1). Let $q_k(i)$ be the ith k-bit continued fraction for $0\leqq i\leqq 2^k-1$, that is, for $i=2^{k-1}b_1+2^{k-2}b_2+\ldots+b_k$,

$$\mathrm{LCF}(q_k(i))=0.b_1b_2\ldots b_k.$$

In order to compare the dispersions of the gap sizes, we set

$$\delta_k(i)=\begin{cases} q_k(i)-q_k(i-1) & \text{for} \quad 1\leqq i\leqq 2^k-1,\\ 1-q_k(2^k-1) & \text{for} \quad i=2^k,\end{cases}$$

and

$$S_k^2=\sum_{i=1}^{2^k}\{\delta_k(i)-2^{-k}\}^2,$$

which is a measure of the variance of the gap sizes. The fact that equivalent uniform spacing as in ordinary fixed-point binary expansion has a uniform gap size of 2^{-k} corresponds to $S_k^2=0$, which is desirable.

Plots of this S_k^2 as a function of k can be made and are shown in Figure 1. A comparison of two curves in this figure shows that substantial improvement in the distribution of the gap sizes is gained by introducing the proposed representation of the integers into LCF-representation.

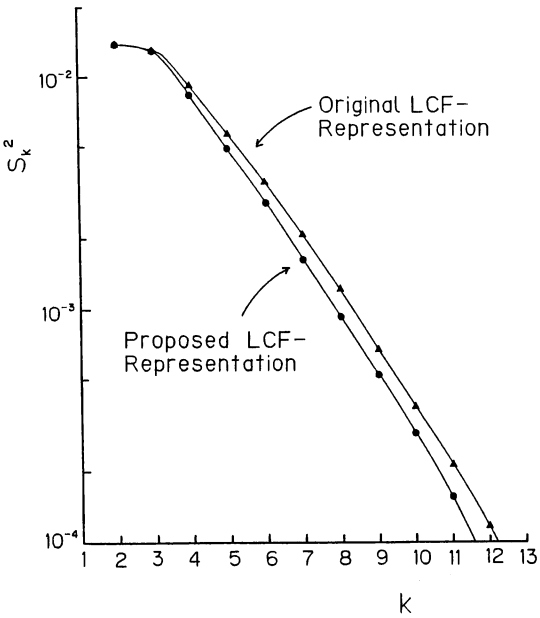

Fig. 1 Comparison of the variances of gap sizes for k-bit continued fractions.

5. Conclusions

This paper has proposed a binary representation of the integers which is efficient for the distribution of partial quotients over the continued fraction expansions of rational numbers. The proposed method gives an improved version of an internal representation of real numbers in computers, which is called LCF-representation.

When we consider a variation of LCF-representation to develop the merit of it, it is important to avoid heuristic techniques and to define it as axiomatically as possible. From this point of view, we must incorporate a representation of the integers which is optimal in some sense. Therefore, the significance of our proposed method should be noted in this respect.

While an ordinary b-ary representation is well defined for any base b, it is not obvious whether or not a representation of the integers and so the corresponding LCF-representation can be easily extended to the case with any base. This is one of open problems which require further investigations.

Acknowledgement

The author would like to express his appreciation to M. Aoki of Yamagata University for her help in preparing the manuscript. Thanks are also due to the referees for improving the clarity of the paper.

References

1. Elias, P. Universal Codeword Sets and Representation of the Integers. *IEEE Trans. Inform. Theory*, IT-**21**, 2 (March 1975), 194–203.
2. Even, S. and Rodeh, M. Economical Encoding of Commas Between Strings. *Comm. ACM*, **21**, 4 (April 1978), 315–317.
3. Gallager, R. G. Information Theory and Reliable Communication. John Wiley and Sons, New York, 1968.
4. Golomb, S. W. Run-Length Encodings. *IEEE Trans. Inform. Theory*, IT-**12**, 3 (July 1966), 399–401.
5. Humblet, P. A. Optimal Source Coding for a Class of Integer Alphabets. *IEEE Trans. Inform. Theory*, IT-**24**, 1 (Jan. 1978), 110–112.
6. Johnsen, O. On the Redundancy of Binary Huffman Codes. *IEEE Trans. Inform. Theory*, IT-**26**, 2 (March 1980), 220–222.
7. Knuth, D. E. Super Natural Numbers, in Mathematical Gardener, Klarner, D. A. ed., Prindle, Weber & Schmidt, Boston, 1980.
8. Knuth, D. E. The Art of *Computer Programming*, Vol. 2: Seminumerical Algorithms, 2nd ed., Addison-Wesley, Reading, Mass., 1981.
9. Matula, D. W. and Kornerup, P. An Order Preserving Finite Binary Encoding of the Rationals. *Proc. 6th IEEE Symp.* on Computer Arithmetic, 1983, 201–209.

(Received November 9, 1987; revised May 16, 1988)