Computer Architecture and Systems
D.P. Siewiorek Editor

An Overrelaxation for a Numerical Inverse of a Constant

Hitohisa Asai
University of Wisconsin-Stout

When division is performed by a power series implementation with additions, subtractions, digit shifts, and multiplications, the convergence rate of the power series is important in practical application. Particularly if the rate of the power series is close to one, the convergence is slow and therefore a special method to accelerate the convergence is needed. Without such an acceleration, the power series implementation is less attractive. An acceleration method is proposed for the slow convergence rate. First, the worst case convergence rate of the power series is determined for a given appropriate acceleration factor. Next, a simple way to choose the appropriate acceleration factor is presented.

Key Words and Phrases: division, numerical inverse, radix, pseudoradix, power series implementation, overrelaxation, worst case convergence, acceleration constant, subdivision, contracting mapping, acceleration factor, approximate quotient, numerical error

CR Categories: 5.11, 6.32

1. Introduction

A division process is not only the most complex arithmetic process but also the most time-consuming arithmetic operation in a digital computer. Any simplifications and timesaving processes for arithmetic division are useful.

The classical division processes for positive radix numbers are essentially repeated operations of divisor subtractions from dividend and divisor digit shifting [6, 11]. Division processes for negative radix numbers are also a sequence of subtractions and digit shifts of divisor quantity [1, 13]. An operational sequence which performs a division in a positive or negative radix is described in the form of a power series [4].

Gilman proposed a division process by multiplication [8]. In his process, divisor and dividend are initially stored in a divisor register and a dividend register, respectively. Each step in his iterative process is a serial operation of (1) a multiplication of a constant to the current contents of these registers and (2) an addition (or a subtraction) of the results of the multiplication to the current contents in order to produce new register contents. The multiplying constant is chosen by sensing D-1 or zero digit in the second most significant digit position of the divisor register where D denotes a radix. The determination of the first multiplying constant is specially treated as a preliminary operation. A power series algorithm of a generalized Newton's approximation method for division process using a parallel multiplier was reported by Ferrari [7]. A product form of the power series has been discussed by many other authors [2, 3, 11, 14].

All the division processes by multiplication are iterative and converge quadratically to obtain a quotient so that a number of the iterative steps to get a desirable accuracy of a quotient depends upon a starting value used in the iteration whose value is predetermined from a given divisor. Anderson et al. [3] and Ferrari [7] discussed selection of the starting value of the iteration from a suitably predetermined value in order to attain fewer steps. In this presentation, a new process to determine the smallest possible starting value for an iterative evaluation of terms in the power series is proposed.

Although the product form of a power series has been implemented in the CRAY-1 [12] and the IBM System/360 Model 91 [3], it is very important not only for the power series to converge but also for it, in a practical application, to converge quickly. When convergence of the power series for division is slow, an acceleration factor may be introduced, forcing the division by the power series to converge more quickly. Thus, this process becomes practical and useful as long as the appropriate acceleration factor is chosen for the division. Furthermore, the acceleration factor eventually becomes an approximate quotient for the reciprocal of a divisor.

When the convergence ratio of a power series for a numerical inverse of a constant is very close to one, a lengthy and time-consuming iteration to obtain a reasonably good approximation of the reciprocal results, even if the convergence is a second order. An accelerating process for slowly converging ratios of the power series is presented.

Svoboda proposed a division algorithm that is based on the fact that when the divisor B is of the form $B = 1 + h$ where h is suitably small, the information about determining the digit of the quotient can be estimated by considering the highest digit of the dividend [15]. An improvement of Svoboda's transformation of the divisor

Author's present address: H. Asai, Computer Science Department, Christopher Newport College of the College of William and Mary, Newport News, VA 23606.

B into the form $1 + h$ was presented by Klir [10]. Their transformation processes provide for a value of h as large as $1/D$ where D denotes a radix used by representing divisor and dividend, and each digit of the quotient is determined by a multiplication and an addition (or a subtraction).

This proposed method guarantees that a value of h is as large as $1/(2D\text{-}3)$. Furthermore, this method can be used repeatedly by expanding a radix D to the radix D^2 [5]. Moreover, an advantage of this method is summarized as follows: It is well-known that the m most significant bits of the reciprocal of an m-bit constant can be obtained in essentially the same time it takes to multiply two m-bit numbers. This statement can be described in a form such as $R(m) \leq CM(m)$ where $R(m)$, $M(m)$, and C denote a time to compute the reciprocal of an m-bit constant, a time to multiply two m-bit numbers, and a constant, respectively [2]. This proposed acceleration process is an attempt to lower the constant C. It is demonstrated in [5] that by defining a complexity measure of multiplication steps under certain conditions, the number of multiplication steps of the division process using the proposed acceleration is less than half the number of steps of the best known process [14], when an approximate reciprocal of a 32-bit constant is computed. The complexity measure is obtained from Karatsuba's complexity of multiplication [2] and [14] by generalizing it for a multiplication of two arbitrary digit lengths.

However, this reduction of the multiplication steps achievable through the proposed process must be paid for in terms of a more complex control requirement such as a table lookup mechanism and additional logic control of bit level multiplication, in which a table lookup mechanism determines acceleration constants used in this process. Although costs are incurred when implementing logic circuits in a table lookup and a bit level control in hardware, the expenses of such implementations are worthwhile to ensure the reduction to half the number of multiplication steps.

2. An Acceleration Factor

A division A/B where A and B are represented in a radix D (assuming $|D| < 1$) can be accomplished by evaluating a power series

$$A/B = A\{1 - P/D + (P/D)^2 - (P/D)^3 + \ldots\}/D \quad (2.1)$$

if $|P/D| < 1$ where $B = D + P$. This evaluation does not involve any division since $(1/D)^i$ is a digit shift operation to the left by i-digits. The series evaluates the numerical inverse of a constant B. When a pseudoradix D' is introduced such that $D' = D^n$ with $|D^n| > |B|$ for the smallest integer $n = 1, 2, 3, 4, \ldots$, the convergence condition $|P/D'| < 1$ in eq. (2.1) is always satisfied. However, when the ratio $|P/D'|$ is close to one, the convergence rate of (2.1) is slow and the division process of (2.1) is practically useless due to the required computation of many terms. Since the truncation errors $|e|$ at the i^{th} term of (2.1) are given by $|A/B| \cdot |P/D|^i$, the number i could be a large quantity to obtain a small e. Therefore, an acceleration factor α for a slowly convergent power series is needed. The factor α is a constant which maps P into a narrow vicinity of D^m where $m \geq n$, in order to obtain the smallest possible $|P/D^m|$.

2.1 The Largest Ratio P/D Without Applying an Acceleration Factor

Consider the ratio P/D' for a division A/B in a radix D, where $D' = D^n$ is a pseudoradix, and $B = D^n + P$. We shall call n the power of a pseudoradix of B. When the power is increased by one, the value of P, for a given B, becomes a negative number. When P is a positive number, we shall denote it as q. Thus we have the following eqs. when $D^n \leq B = D^n + q \leq D^{n+1}$

$$P = q \quad \text{if } P > 0 \quad (2.2a)$$
$$P = (D^n + q) - D^{n+1} \quad \text{if } P < 0 \quad (2.2b)$$

Note: When $P = 0$, namely $B = D' = D^n$ or D^{n+1}, the division process A/B becomes A/D'; that is, just a digit shift operation. The ratio P/D' in (2.1) is represented by functions of q.

$$P/D^n = f_1(q) = q/D^n \quad \text{for } P > 0 \quad (2.3a)$$
$$P/D^{n+1} = -f_2(q) = -\{D^{n+1} - (D^n + q)\}/D^{n+1} \quad \text{for } P < 0 \quad (2.3b)$$

From eqs. (2.2) the domain of q is $[0, D^n(D-1)]$ where D^n is a pseudoradix. Since q depends on the pseudoradix, the domain may be considered a fixed interval $[0, D'(D-1)]$ relative to a pseudoradix D'. From inspection of eqs. (2.3), we obtain the following theorem:

THEOREM 1. *If a constant B exists in a range between D^n and D^{n+1}, then the maximum ratio $|P/D'|$ in* (2.1) *is equal to $(D-1)/(D+1)$ and to $(D+1)/(D-1)$ when $D > +1$ and $D < -1$, respectively.*

PROOF. We shall discuss two cases: (1) when the radix is positive and (2) when the radix is negative.

CASE 1: $D > +1$. The ratio is represented by $f_1(q)$ or $f_2(q)$ depending upon whether B is measured from D^n or from D^{n+1}. Since $f_1(q)$ is monotonically increasing and $f_2(q)$ is monotonically decreasing with respect to q, there is an intersection of both the functions at $q = q^*$ so that the minimum of the maximum (hereafter called the maximum) of the ratio is $f_1(q^*) = f_2(q^*)$. We obtain the following eq.:

$$q^*/D^n = (D-1)/(D+1) \quad (2.4a)$$

CASE 2: $D < -1$. When considering the absolute values of the functions $|f_1(q)|$ and $|f_2(q)|$, the correct sign must be chosen when removing the absolute value operators (i.e., whether n is odd or even). A similar computation of that in Case 1 gives the following eq.:

$$|q^*/D^n| = (D+1)/(D-1) \quad (2.4b)$$

based on the fact that $|f_1(q)| = q/(\pm D^n)$ and $|f_2(q)| =$

$\{\mp D^{n+1} - (\pm D^n + q)\}/(\mp D^{n+1})$ in which a correct sign from the compound sign (plus/minus $\pm$ and minus/plus $\mp$) is chosen as described below.

Since the quantity of an absolute value is positive, the upper sign of each compound sign must be chosen when n is an even number, and the lower sign must be chosen when n is an odd number. We have thus completed a proof of Theorem 1. □

When $|D| = 2$, that is a positive or negative binary radix, the power series (2.1) for the division process is convergent with the ratio 1/3 at most. When $|D|$ is greater than two, the best case ratio increases since it is defined as $(D - 1)/(D + 1)$ for $D > +1$ and $(D + 1)/(D - 1)$ for $D < -1$. The convergence of (2.1) thus becomes slower as $|D|$ increases. This disadvantage caused by a larger radix can be overcome by using an acceleration factor.

2.2 The Largest Ratio P/D with an Acceleration Factor

For simplicity without loss of generality, we shall restrict ourselves to the fact that A and B in a division (A/B) are integers. Consequently, P and q are also integers by assuming a radix-point at the least significant digit position. Other registers hold the exponents of the numbers A and B.

Let us consider an overrelaxation of $f_2(q)$ with a parameter a.

$$g(q, a) = \{D^{n+1} - a(D^n + q)\}/D^{n+1} = -aq/D^{n+1} + (D - a)/D \tag{2.5}$$

where a is an integer in the interval $[1, D]$. Note: As special cases, $g(q, 1) = -f_2(q)$ and $g(q, D) = -f_1(q)$. Using reasons similar to those employed in proving Theorem 1, we obtain the following corollary by considering the fact $f_1(q^*) = g(q^*, a)$ for $a = 1, 2, 3, \ldots, D - 1$. Hereafter, we shall restrict ourselves to discussing only the case $D > +1$ since a discussion of the case $D < -1$ is very similar.

COROLLARY 1. *If a constant B is in a range between D^n and D^{n+1}, then a local maximum of the ratio in* (2.1) *is*

$$q^*(a)/D^n = (D - a)/(D + a) \tag{2.6}$$

at $a = 1, 2, \ldots, D - 1$ where a is regarded as an argument of q^*.

The proof is almost the same as that for Theorem 1; therefore, we have omitted it here. Corollary 1 tells us that the ratio is decreased as the parameter a increases, and that a may offset a slower convergence caused by a larger radix.

Figure 1 illustrates the intersections M_a for $a = 1, 2, \ldots, (D - 1)$ between $f_1(q)$ and a family of $g(q, a)$ for $a = 1, 2, \ldots, D$. Each line of $g(q, a)$ crosses the abscissa when the zero of $g(q, a)$ occurs at $q = D^n(D - a)/a$. Since the domain q is $[0, D^n(D - 1)]$, there are D zeros in the domain including $B = D^n$ and $B = D^{n+1}$. Consider the saw-shaped zig-zag lines crossing the abscissa which are connected to another line of the $g(q, a)$ family. The maximum of the ratio, where $M_a = (D - a)/(D + a)$, could be reduced significantly to those points indicated with M'_a for $a = 1, 2, \ldots, (D - 1)$. This is stated in the next theorem.

THEOREM 2. *If a constant B is in a range between D^n and D^{n+1}, then the maximum of the ratio in* (2.1) *is 1/3.*

PROOF. Consider the situation $-g(q^+, a) = g(q^+, a - 1)$ where the quantity of $q^+(a)$ is determined as follows:

$$q^+(a) = (2D - 2a + 1)D^n/(2a - 1) \tag{2.7}$$

for $a = 2, 3, \ldots, D$ where a is regarded as an argument of q^+. The value of $q^+(a)$ is a boundary of a different parameter applicable to the regions of q. We shall define these regions as follows:

$$\begin{aligned}
&S_1\colon q^+(2) = (2D - 3)D^n/3 < q \le D^n(D - 1)\\
&S_2\colon q^+(3) = (2D - 5)D^n/5 < q \le q^+(2) = (2D - 3)D^n/3\\
&S_3\colon q^+(4) = (2D - 7)D^n/7 < q \le q^+(3) = (2D - 5)D^n/5\\
&\vdots\\
&S_i\colon q^+(i + 1) = (2D - 2i - 1)D^n/(2i + 1) < q \le q^+(i)\\
&\qquad\qquad = (2D - 2i + 1)D^n/(2i - 1)\\
&\vdots\\
&S_{D-3}\colon q^+(D - 2) = 5D^n/(2D - 5) < q \le q^+(D - 3)\\
&\qquad\qquad = 7D^n/(2D - 7)\\
&S_{D-2}\colon q^+(D - 1) = 3D^n/(2D - 3) < q \le q^+(D - 2)\\
&\qquad\qquad = 5D^n/(2D - 5)\\
&S_{D-1}\colon q^+(D) = D^n/(2D - 1) < q \le q^+(D - 1) = 3D^n/(2D - 3)\\
&S_D\colon 0 < q \le q^+(D) = D^n/(2D - 1)
\end{aligned} \tag{2.8}$$

We shall call the regions subdivisions of q. By substituting the eq. (2.7) in $-g(q, a)$, we obtain a ratio as follows:

$$-g(q^+(a), a) = 1/(2a - 1) \tag{2.9}$$

for $a = 2, 3, \ldots, D$. From the definition of the subdivisions S_a for $a = 1, 2, 3, \ldots, D$ and the linearity of $g(q, a)$ with respect to q, we can conclude the maximum of the ratio exists at $q^+(2)$ where $a = 2$.

$$\max_{1<a\le D} |P/D'| = 1/(2a - 1) = 1/3 \tag{2.10}$$

□

As shown by Theorem 1, the convergence of (2.1) becomes slower as $|D|$ increases. This indicates that the maximum of the ratio is 1/3 when $|D| = 2$. The disadvantage caused by using a larger radix is offset by adopting a proper parameter a, where q exists in a subdivision S_a. From the above discussion, we obtain the following corollary.

COROLLARY 2. *If a constant $B(=D^n + q)$ is in a subdivision S_a for $a = 2, 3, \ldots, D$, then the local maximum of the ratio is $1/(2a - 1)$. If a constant B is in S_1, then the local maximum is* 1/3.

We have omitted the proof. The following observation can be drawn from Theorem 2 and Corollary 2. As the parameter a increases, the local maximum in subdivision S_a decreases and moves closer to $1/(2D - 3)$, the limit attained in S_{D-1}. This observation provides insight

Fig. 1. Family of $g(q, a), f_1(q)$, and Subdivisions S_a.

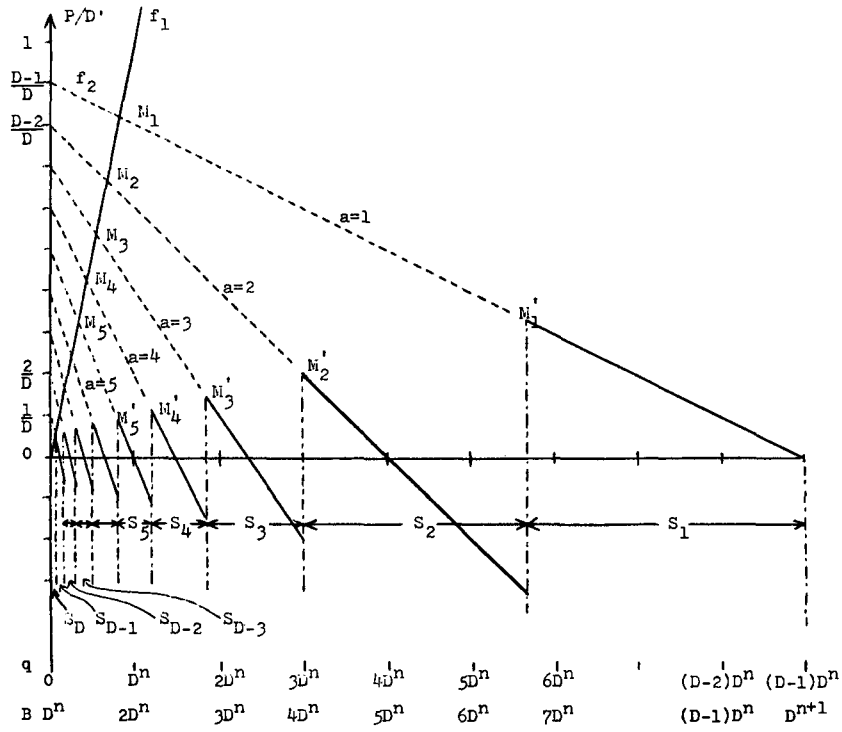
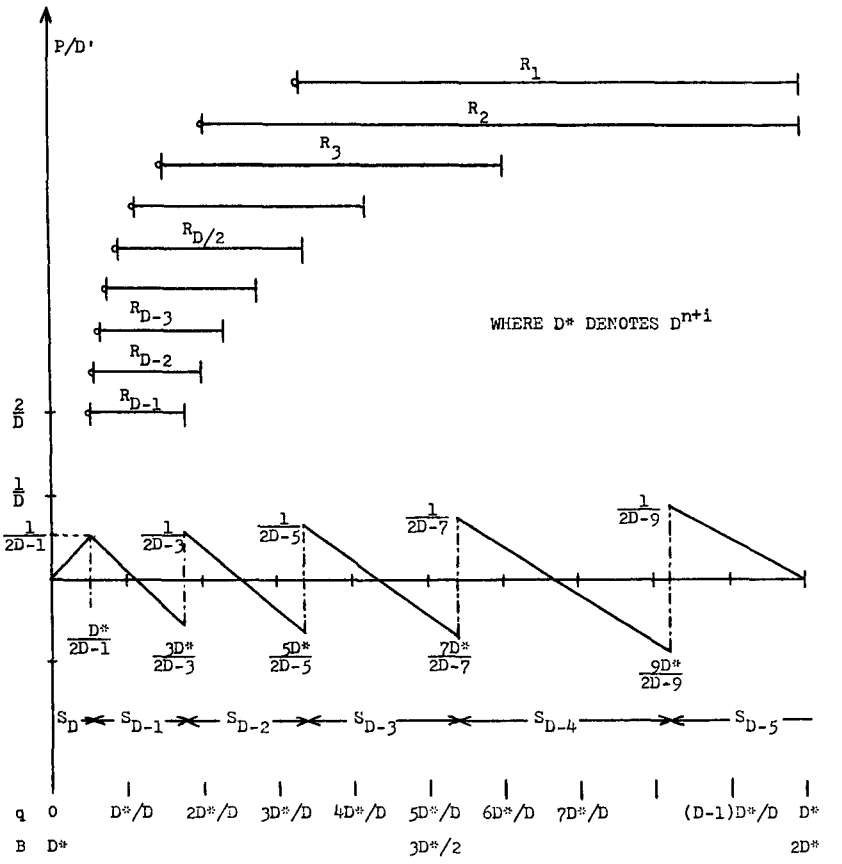


Fig. 2. The Ranges of R_a and Subdivisions S_a Between 0 and D^{n+i}.



into constructing a contraction mapping of q; this is discussed in the next section.

2.3 The Largest Ratio P/D with a Successive Acceleration Factor

We shall further improve the maximum of the ratio by using a greater pseudoradix and another parameter a. First, we note the following relations from Figure 1 where q exists in the S_a subdivision.

$$|g(q, a \pm 1)| > |g(q, a)| \tag{2.11}$$

and

$$g(q, a-1) > g(q, a) > g(q, a+1) \tag{2.12}$$

Next, we observe the following situation: Although $|g(q, a+1)|$ is greater than $|g(q, a)|$, q_2/D^n is less than q_1/D^n where the value of q_2 is computed from $q_2 = -g(q_1, a+1)D^n$. Thus, the value of q_2 may belong to another subdivision S_i where $i \geq a$ and q_1 is assumed in S_a. The function $-g(q, a+1)D^n$ shall be denoted as $q_2 = h(q_1, a)$. The entire computed mapping range of $h(q, a)$ from the q_1 domain $[0, D^n(D-1)]$ is the interval $[0, D^{n+1}]$. The individual subranges R_a of $h(q_1, a)$ for each corresponding domain S_a for $a = 1, 2, 3, \ldots, (D-1)$ are computed as follows:

$$\begin{aligned}
&R_1\colon D^{n+1}/3 < h(q^{\pm}(2), 2) \leq q_2 \leq h(D^n(D-1), 2) = D^{n+1}\\
&R_2\colon D^{n+1}/5 < h(q^{\pm}(3), 3) \leq q_2 \leq h(q^{+}(2), 3) = D^{n+1}\\
&R_3\colon D^{n+1}/7 < h(q^{\pm}(4), 4) \leq q_2 \leq h(q^{+}(3), 4) = 3D^{n+1}/5\\
&\vdots\\
&R_i\colon D^{n+1}/(2i+1) < h(q^{\pm}(i+1), i+1)\\
&\quad \leq q_2 \leq h(q^{+}(i), i+1) = 3D^{n+1}/(2i-1)\\
&\vdots\\
&R_{D-3}\colon D^{n+1}/(2D-5) < h(q^{\pm}(D-2), D-2)\\
&\quad \leq q_2 \leq h(q^{+}(D-3), D-2) = 3D^{n+1}/(2D-7)\\
&R_{D-2}\colon D^{n+1}/(2D-3) < h(q^{\pm}(D-1), D-1)\\
&\quad \leq q_2 \leq h(q^{+}(D-2), D-1) = 3D^{n+1}/(2D-5)\\
&R_{D-1}\colon D^{n+1}/(2D-1) < h(q^{\pm}(D), D) \leq q_2\\
&\quad \leq h(q^{+}(D-1), D) = 3D^{n+1}/(2D-3)
\end{aligned} \tag{2.13}$$

where $q^{\pm}(a)$ for $a = 2, 3, \ldots, D$ is the inferior limit of $q^{+}(a)$ in S_a. Figure 2 illustrates the subranges R_a and the detailed $f_1(q)$ and $g(q, a)$. Note: After an application of this mapping, the power of the present pseudoradix is incremented by one as shown in (2.13).

In order to apply the function $h(q, a)$ and $g(q, a)$ successively we will use the suffix n. Thus, $h(q, a)$ and $g(q, a)$ become $h_n(q, a)$ and $g_n(q, a)$, respectively. The q domain for n is the interval $[0, D^n(D-1)]$ and for $n+1$ the q domain becomes $[0, D^{n+1}(D-1)]$. Similarly, for applying S_a recursively we use S_a^n with the q domain as above.

The mapping function $h_{n+i}(q_i, a_i)$ is a piecewise linear function that maps the interval of q, $[0, D^{n+i}(D-1)]$, into the interval $[0, D^{n+i+1}]$. When the pseudoradices D^{n+i} and D^{n+i+1} are neglected due to the relative definition of the subdivisions, these intervals (the individual subranges described in (2.13) and each corresponding subdivision defined in (2.8)) may be considered fixed upper/lower limits relative to a pseudoradix and are called relative ranges and relative domains of q, respectively. An interval consisting of the upper/lower limits of each subdivision is greater than the one of the corresponding range limits except for $a = D-1$. In other words, a mapped relative range, which becomes the next relative domain, is narrower than its relative domain. When a successive application of $h(q, a)$ is taking place, the distance measured from the origin in the relative q coordinate contracts until the distance reaches the subdivision S_{D-1} which is an invariant interval. The gradually shrinking distances are successive images of the q mapping. Thus there is a sequence of subdivisions that are a visiting order of successive images. This is described in the following lemma.

LEMMA 1. *Let $B = D^n + q_0$ be a constant. If a successive q_i is determined by $q_{i+1} = h_{n+i}(q_i, a_i)$ where q_i in $S_{a_i}^{n+i}$ for $i = 0, 1, 2, \ldots$, then parameters $a_0, a_1, a_2, \ldots$, and a_j in generating a sequence by $h(q, a)$ have a relation $a_0 \leq a_1 \leq a_2 \leq \ldots \leq a_j = D-1$ where q_j is in S_{D-1}^{n+j}.*

A repeated application of the mapping to value q will result in the movement of successive images into the subdivision S_{D-1}. As soon as the last image q_j that belongs to S_{D-1} is obtained, the function $g(q_j, D-1)$ is evaluated, in order to find the final ratio q_f. The conclusion of our discussion is that the worst case ratio of the maximum $|P/D'|$ is equal to $1/(2D-3)$. This is stated in the following theorem.

THEOREM 3. *Let $B = D^n + q_0$ be a constant in $[D^n, D^{n+1}]$.*

(1) *If q_0 is in a subdivision S_{a_0} when an integer a_0 is in $[1, D-1]$ and the last q_j which is the first visit in the subdivision S_{D-1} is obtained through a sequence of q_i by successive mapping $q_{i+1} = h_{n+i}(q_i, a_i)$, then an acceleration factor α' is*

$$\alpha' = (D-1)\prod_{i=0}^{j-1}(a_i + 1) \tag{2.14}$$

The final ratio P/D' is obtained from $q_f = -g_{n+j}(q_j, D-1)$.

(2) *If q_0 is in the subdivision S_D, then an acceleration factor α' is D. The ratio P/D' is determined by $q_f = f_1(q_0)$.*

Thus, a numerical inverse of a constant B is evaluated as follows:

$$1/B = \alpha'\{1 - q_f + q_f^2 - q_f^3 + q_f^4 - \ldots\}/D^{n+j+1} \tag{2.15}$$

where $|q_f| \leq 1/(2D-3)$ is guaranteed.

PROOF.

(1) From Lemma 1, there is a sequence of a_i for $i = 0, 1, 2, \ldots, (D-1)$ and the value of $|g_{n+j}(q_j, D-1)|$ for q_j in S_{D-1} is less than or equal to $1/(2D-3)$ from Corollary 2.

(2) From Corollary 2, the value of $|f_1(q_0)|$ for q_0 in S_D is less than or equal to $1/(2D-1)$.

Hence,

$$|q_f| \le 1/(2D-3) \tag{2.16}$$

□

After we obtain a ratio q_f from the above method, the quantity of (2.15) must be evaluated. However, a careful inspection of the inside terms in the brace of (2.15) leads us to the following factorial terms:

$$\begin{aligned}
&1 - q + q^2 - q^3 + q^4 - q^5 + q^6 - q^7 + q^8 - q^9 + \ldots \\
&= 1 - q + q^2(1-q) + q^4(1 - q + q^2 - q^3) \\
&\quad + q^8(1 - q + q^2 - q^3 + q^4 - \ldots - q^7) \\
&\quad + q^{16}(1 - q + q^2 - \ldots)\ldots \\
&= (1-q)(1+q^2) + q^4\{(1-q)(1+q^2)\} + q^8\{(1-q)(1+q^2) \\
&\quad + q^4(1-q)(1+q^2)\} + q^{16}\ldots \\
&= (1-q)(1+q^2)(1+q^4)(1+q^8)(1+q^{16})\ldots.
\end{aligned}$$

We consider these terms as other acceleration factors and denote them as follows:

$$b_0 = D(1 - q_f) \tag{2.17a}$$

and

$$b_k = D(1 + q_f^{2^k}) \quad \text{for} \quad k = 1, 2, 3, \ldots \tag{2.17b}$$

Thus,

$$1/B = (D-1)\prod_{i=0}^{j-1}(a_i + 1)\prod_{k=0}^{l} b_k \{1 + q_f^{2^{l+1}} - (q_f^{2^{l+1}})^2 + \ldots\}/D^{n+j+l+2} \tag{2.18}$$

Ignoring the terms that are smaller than $q_f^{2^l}$ in (2.18), an approximate reciprocal of B becomes as follows:

$$1/B = \alpha' \prod_{k=0}^{l} b_k/D^{n+j+l+2} = \alpha/D^{n+j+l+2} \tag{2.19}$$

Example

Consider the evaluation 1/189 where $B = 189$, $D = 10$, and the power of a pseudoradix is 2 so $D' = D^2$. Before we compute the numerical inverse, a table of subdivision limits is needed for the radix $D = 10$. These limits are listed in Table I. Now we can compute the inverse.

Since $q_0 = 89$, it exists in the fifth subdivision S_5 and a_0 equals five. (We believe that a binary search technique of table lookup is the best way to determine a subdivision of a value q.) Thus, since q_0 is not in S_{D-1}, we apply recursively the mapping function $h(q, a)$ on q_0 in order to obtain q_1.

$$\begin{aligned}
q_1 &= h_2(q_0, a_0) = -g_2(q_0, a_0 + 1)D^2 \\
&= -\{1000 - 6(100 + 89)\}/10 = 13.4
\end{aligned}$$

Now by inspecting Table I we see that the value q_1 is in the subdivision S_9 and a_1 equals nine. Note: The power of the present pseudoradix must be incremented by one; in other words, a new pseudoradix D^{n+1} is used in the denominator of (2.15), the unit of q_1 is shifted by one digit to the right. Thus, q_1 becomes 134 based on the new radix. This is the first visit to S_9 so we do not need another application of $h(q, a)$.

The final q_f is evaluated by

$$\begin{aligned}
q_f &= -g_3(q_1, 9) = \\
&\quad -\{10000 - 9(1000 + 134)\}/10000 = 206/10000.
\end{aligned}$$

Table I. Subdivision upper/lower limits for $D = 10$ and $D' = 100$.

Subdivision number	Lower limit		q		Upper limit
1	(17) × 100 = 566.66...	<	q	≦	900.
2	(15) × 100 = 300.00	<	q	≦	566.66...
3	(13) × 100 = 185.71...	<	q	≦	300.00
4	(11) × 100 = 122.22...	<	q	≦	185.71...
5	(9) × 100 = 81.81...	<	q	≦	122.22...
6	(7) × 100 = 53.84...	<	q	≦	81.81...
7	(5) × 100 = 33.33...	<	q	≦	53.84...
8	(3) × 100 = 17.64...	<	q	≦	33.33...
9	(1) × 100 = 5.263...	<	q	≦	17.64...
10	0.00	≦	q	≦	5.263...

The inverse is determined from the following computation of (2.18) by using up to q_f^4, or $b_0 = 9.794$, $b_1 = 10.0042436$, and $b_2 = 10.00000180081$ derived from (2.17),

$$\begin{aligned}
1/189 &= 6/1134 = 6\times 9/10206 \\
&= 54\{1 - 206/10000 + (206/10000)^2 - \ldots\}/10^4 \\
&= 54\times 9.794\times 10.0042436 \\
&\quad \times 10.00000180081\{1 + (206/10000)^8 - \ldots\}/10^7 \\
&\approx 5.29100529100295\ldots\times 10^{-3}.
\end{aligned}$$

From Theorem 3, as the radix D increases, the worst case ratio of $|P/D'|$ is decreasing. Suppose we use $D = 100$ instead of $D = 10$. Then the worst case ratio decreases to $1/(200 - 3) = 0.005076$ from $1/(20 - 3) = 0.0588\ldots$ (refer to (2.16)) so that an evaluation of (2.15) becomes simpler due to fewer term accumulations. A simple computation process is described here. We start with $D = 100$, $q_0 = 89$, and $n = 1$. Using a table of subdivision limits for $D = 100$, we determine that q_0 is in S_{53} (the interval (86.91..., 90.47...]) and $a_0 = 53$, so q_1 is $q_1 = h_1(89, 53) = -g(89, 54)100 = 2.06$. Now the value of q_1 is in S_{98} (the interval (1.522..., 2.564...]), and $a_1 = 98$ so one more application of the mapping is needed.

$$q_2 = h_2(206, 98) = -g_2(206, 99)100 = 1.0394.$$

The value q_2 is finally in S_{99} (the interval (0.502..., 1.522...]) and $a_2 = 99$; then the final value of q_f is computed from $g_3(q_2, a_2)$ as shown below.

$$q_f = -g_3(10394, 99) = 0.00029006.$$

Thus, by using $b_0 = 99.970994$ and $b_1 = 100.0000084138036$, we obtain

$$\begin{aligned}
1/189 &= 54\times 99\times 99\{1 - 0.00029006 + (0.00029006)^2 - \ldots\}/100^4 \\
&= 529254\times 99.970994\times 100.0000084138036 \\
&\quad \{1 + (0.00029006)^4 - \ldots\}/100^6 \\
&\approx 5.291005291005243\ldots\times 10^{-3}.
\end{aligned}$$

Comparing these results with the real quotient ($5.2910052910052910\times 10^{-3}$), we find that these numerical accuracies are correct to 12 digits (when $D = 10$) and 14 digits (when $D = 100$). In the example, we needed to evaluate acceleration constants of two a's (one integer multiplication; note that $(D - 1)$ multiplication is performed with a shift to the left and a subtraction) and three b's (twice of squaring operations) for the case D

= 10, and three a's (twice of integer multiplications) and two b's (once of squaring operation) for the case $D = 100$. Also, it was necessary to multiply these constants by each other (four multiplications) in order to obtain the numerical quotient.

Consider a straightforward application of the eq. (2.17), that is quadratic convergence, to this example. We would probably need to determine nine b acceleration constants starting with $b_0 = 10\times(1 - 89/100) = 1.1$ in order to attain at least the same accuracy or a better one. This implies that eight squaring operations, eight multiplications, and nine subtractions must be computed. The calculation of a numerical inverse using the acceleration factor compares very well with this.

When a divisor consists of many digits, the divisor can be split into two or more separate digit strings. A digit splitting scheme for a division was reported by Jacobsohn [9]. Consider $B = B_1D^m + B_2$ where B_1 and B_2 are the upper-half significant digits and the lower-half significant digits, respectively, and D^m is a pseudo-radix that indicates the digit boundary between B_1 and B_2. Thus,

$$\begin{aligned} 1/B &= (1/B_1D^m)(1/(1+q_b)) \\ &= \left\{(D-1)\prod_{i=0}^{j-1}(a_i+1)\prod_{k=0}^{\infty}(b_k/D)/D^{n+j+2}\right\} \\ &\times \left\{\prod_{r=0}^{\infty} c_r/D^m\right\} \end{aligned} \tag{2.20}$$

where $q_b = B_2/B_1D^m$, and $c_0 = 1 - q_b$ and $c_r = 1 + q_b^{2^r}$ for $r = 1, 2, 3, \ldots$. The inside of the first brace is a computation of $1/B_1$ and the second one is the power series of $1/(1 + q_b)$.

An approximate value of $1/B_1$ can be computed by the division process discussed previously. Since the term q_b of the power series that shall be denoted as P_{cr} contains the factor $1/B_1$, a truncation error of P_{cr} is effected with the error magnitude of B_1 in (2.19). This effect is discussed here. By substituting the quantity $1/B_1$ of (2.18) into the terms q_b of P_{cr}, we obtain the following eq.:

$$\begin{aligned} P_{cr} &= \{1-\beta(B_1^*+\epsilon)\}\{1+\beta^2(B_1^*+\epsilon)^2\} \\ &\quad \{1+\beta^4(B_1^*+\epsilon)^4\}\{1+\beta^8(B_1^*+\epsilon)^8\}\{1+\ldots \\ &= 1-\beta B_1^*\{1-\beta B_1^*+(\beta B_1^*)^2-(\beta B_1^*)^3+(\beta B_1^*)^4 \\ &\quad -(\beta B_1^*)^5+(\beta B_1^*)^6-\ldots\} \\ &\quad -\beta\epsilon\{1-2\beta B_1^*+3(\beta B_1^*)^2-4(\beta B_1^*)^3 \\ &\quad +5(\beta B_1^*)^4-6(\beta B_1^*)^5+7(\beta B_1^*)^6-\ldots\} \\ &\quad +\beta^2\epsilon^2\{1-\ldots\} \\ &\quad -\ldots. \end{aligned} \tag{2.21}$$

where B_1^* is an approximate value of $1/B_1$, ϵ is the truncation error determined from (2.18) for B_1^*, and $\beta = B_2/D^m$.

The first-order term of ϵ in (2.21) becomes $-\beta\epsilon/(1 + \beta B_1^*)^2$. The higher order terms of ϵ may be neglected so that an estimated error bound ϵ_c due to the truncation error ϵ and to a finite term accumulation in P_{cr} is $|\epsilon| + |\beta\epsilon| \geq \epsilon_c \geq |\epsilon| + |\beta\epsilon/(1+\beta B_1^*)^2|$ from (2.20) and (2.21). When a divisor B is split into three separate (B_i for $i = 1, 2, 3$) digits, namely $B = B_1D^{m_1} + B_2D^{m_2} + B_3$, we have an eq. similar to (2.20), but in continued fraction form, where m_1 and m_2 are boundary digits of these subdivisors.

3. Summary and Conclusions

We have considered a numerical evaluation of $1/B$ (or A/B) as

$$1/B = 1/D^n \sum_{i=0}^{\infty} (P/D^n)^i$$

where $B = D^n + P$ and D is a radix, and we have found that $|P/D^n|$ can be as large as $|(D-1)/(D+1)|$ which is greater than $1/(2D-3)$. In order to accelerate the convergence, we proposed an acceleration factor for the inverse evaluation of a constant by implementing a power series expansion as follows:

$$1/B = \alpha/\alpha B = \alpha A/D^m \sum_{i=0}^{\infty} (-q/D^m)^i \approx \alpha/D^m$$

with $\alpha B = D^m + q$ and $m \geq n$. The quantity α is an acceleration factor and is an approximation to several significant digits.

The fact is that the worst case convergence of the power series is the ratio $1/(2D-3)$ and an approach to determining an acceleration factor has been discussed in order to accomplish such a power series evaluation with the smallest ratio. We believe, however, that the worst case ratio is small enough to apply this implementation in practical use.

Utilizing a larger radix D and an acceleration constant, the worst convergence ratio is reduced considerably, as shown in Theorem 3 and the example. Also, an approximation to more significant digits can be realized. But when a chosen radix is too large, table searching to determine a subdivision (or an acceleration constant) becomes lengthy and time-consuming. Therefore, a trade-off between both aspects must be made; i.e., the advantage of using a smaller convergent rate in a power series and the disadvantage of a larger table searching time to obtain an appropriate acceleration constant. Also, we have shown that the repeated use of the mapping function $h(q, a)$ leads to the invariant interval S_{D-1}. Thus, there is a limit for repeated application. A continuous application does not guarantee a successive convergence unless we adapt a larger radix that will provide more digits of an acceleration constant (i.e., for $D = 10$, $D - 1 = 9$ we have one digit, and for $D = 100$, $D - 1 = 99$ we have two digits).

Actually, we may not need to expand a radix; rather, it is more important to obtain one more digit of an acceleration constant at each successive application $h(q, a)$. One more digit of an acceleration constant can be obtained by using linear interpolation of the contraction mapping function except at a boundary of subdivisions.

Additionally, the proposed process can be intuitively improved by considering the two separate domains of each subdivision based on whether the range of $g(q, a)$ is positive or negative as shown in Figure 1. Since the quantity q_{i+1} obtained in this process must be positive, the value of q_{i+1} is calculated from $h(q_i, a_i)$ $(= -g(q_i, a_i + 1)D^n)$, only when q^*_{i+1} $(= -g(q_i, a_i)D^n)$ is negative. When q^*_{i+1} is, however, positive, the determined q^*_{i+1} is used in the process, instead of q_{i+1}, due to its positive value. Consequently, using value q^*_{i+1} could result in fewer iterations. As soon as a ratio (a value of q_i) is attained below the worst case ratio $d/(2D - 3)$ during a successive determination of acceleration constants, the successive determination may be terminated where $d < 1$ is a discount factor. Thus, the last q value could be used in a power series evaluation. The discount factor d has the following meaning.

When $1/(2D - 3)$ is the threshold value of a termination, the last q value can be as large as $1/(2D - 3)$. If the successive determination could be continued, we may have a smaller q at the end than the terminated value q, which is close to the threshold value. A continuation of the successive determination may be worthwhile in some cases. The discount factor d which provides a lower threshold value is a constant, forcing the continuation when the current value q does not belong in S_{D-1}.

Finally, we have discussed a divisor splitting into two or more subdivisors when the divisor consists of many digits. An error estimation for digit splitting into two subdivisors indicates that a truncation error of the first subdivisor B_1 is dominated in a result obtained from the reciprocal evaluation process. But the desired accuracy of a reciprocal is obtainable by splitting divisor digits as long as a truncation error of $1/B_1$ is appropriately smaller.

Acknowledgments. The author wishes to thank the referees and the department editor for their suggestions on improving the presentation of this manuscript.

Received 7/77; revised 11/78; accepted 5/80

References

1. Agrawal, D.P. Arithmetic algorithms in a negative base. *IEEE Trans. Comptrs. C-24*, 10 (1975), 998–1000.
2. Aho, A.V., Hopcropt, J.E., and Ullman, J.D. *The Design and Analysis of Computer Algorithms.* Addison-Wesley, Reading, Mass., 1974.
3. Anderson, S.F., et al. The IBM system/360 model 91: Floating point execution unit. *IBM J. Res. Develop. 11*, 1 (Jan. 1967), 34–53.
4. Asai, H. A recursive radix conversion formula and its application to multiplication and division. *J. Comp. and Math. with Appls. 2*, 3/4 (1976), 255–265.
5. Asai, H. A complexity measure of a division process—a comparison analysis of AMONIC (submitted to a technical journal).
6. Chu, Y. *Digital Computer Design Fundamentals.* McGraw-Hill, N.Y., 1962.
7. Ferrari, D. A division method using a parallel multiplier. *IEEE Trans. EC-16*, 2 (1967), 224–226.
8. Gilman, R.E. A mathematical procedure for machine division. *Comm. ACM 2*, 4 (April 1959), 10–12.
9. Jacobsohn, D.H. A combinatoric division algorithm for fixed-integer division. *IEEE Trans. Comptrs. C-22*, 6 (1973), 608–610.
10. Klir, J. A note on Svoboda's algorithm for division. Inform. Proc. Machine No. 9, Prague, Czechoslovakia, 1963, pp. 35–39.
11. Knuth, D.E. *The Art of Computer Programming, Vol. 2.* Addison-Wesley, Reading, Mass., 1969.
12. Russell, R.M. The CRAY-1 computer system. *Comm. ACM 21*, 1 (Jan. 1978), 63–72.
13. Sanker, P.V., et al. Arithmetic algorithm in negative base. *IEEE Trans. Comptrs. C-22*, 2 (1973), 125–128.
14. Savage, J.E. *The Complexity of Computing.* John Wiley & Sons, N.Y., 1976.
15. Svoboda, A. An algorithm for division. Inform. Proc. Machine No. 9, Prague, Czechoslovakia, 1963, pp. 25–34.