

# Choosing Starting Values for certain Newton-Raphson Iterations

Peter Kornerup<sup>a</sup> Jean-Michel Muller<sup>b</sup>

<sup>a</sup>*University of Southern Denmark  
Odense, Denmark*

<sup>b</sup>*CNRS-LIP-Arénaire  
Lyon, France*

---

## Abstract

We aim at finding the best possible seed values when computing  $a^{\frac{1}{p}}$  using the Newton-Raphson iteration in a given interval. A natural choice of the seed value would be the one that best approximates the expected result. It turns out that in most cases, the best seed value can be quite far from this natural choice. When we evaluate a monotone function  $f(a)$  in the interval  $[a_{\min}, a_{\max}]$ , by building the sequence  $x_n$  defined by the Newton-Raphson iteration, the natural choice consists in choosing  $x_0$  equal to the arithmetic mean of the endpoint values. This minimizes the maximum possible distance between  $x_0$  and  $f(a)$ . And yet, if we perform  $n$  iterations, what matters is to minimize the maximum possible distance between  $x_n$  and  $f(a)$ . In several examples, the value of the best starting point varies rather significantly with the number of iterations.

*Key words:* Computer arithmetic, Newton-Raphson iteration, Division, Square-Root, Square-Root Reciprocal, Root Extraction

---

## 1 Introduction

Newton-Raphson iteration is a well-known and useful technique for finding zeros of functions. It was first introduced by Newton around 1669 [12], to solve polynomial equations (without explicit use of the derivative), and generalized by Raphson a few years later [17]. NR-based division and/or square-root have been implemented on many recent processors [14,8,15,13,9].

As a matter of fact, the classical “Newton-Raphson” iteration for evaluating square-roots (deduced from the general iteration by looking for the zeros of function  $x^2 - a$ ) goes back to much earlier. Al-Khwarizmi mentions this

method in his arithmetic book [2]. Moreover, it was already used by Heron of Alexandria (this is why it is frequently quoted as “Heron iteration”), and seems to have been known by the Babylonians 2000 years before Heron [6].

Let us now turn to the modern Newton-Raphson (NR) iteration. Assume we want to compute a root  $\alpha$  of some function  $\phi$ . The NR iteration consists in building a sequence

$$x_{n+1} = x_n - \frac{\phi(x_n)}{\phi'(x_n)}. \quad (1)$$

If  $\phi$  has a continuous derivative and if  $\alpha$  is a single root (i.e.,  $\phi'(\alpha) \neq 0$ ), then the sequence converges quadratically to  $\alpha$ , provided that  $x_0$  is close enough to  $\alpha$ .

The choice of a good starting value for the square root iteration has been the subject of some research since the 1960'es. An early reference is [7] and later [1] also attempted to minimize the maximal error expressed as

$$\max_{x \in [a,b]} \left| \log \frac{G(x)}{\sqrt{x}} \right|,$$

using a polynomial or rational function  $G(x)$  of some prescribed degree. Similarly [4,11] minimized the relative error:

$$\max_{x \in [a,b]} \left| \frac{\sqrt{x} - G(x)}{\sqrt{x}} \right|,$$

where the latter reference showed, that for such functions the optimal starting value is independent of the number of iterations to be performed, except when the approximation is chosen to be a constant. [5] provided 9 different such approximating functions. [18] showed some simple relations between several of these optimization criteria. [19] investigated similarly the alternative iteration for the square root reciprocal

$$x_{n+1} = x_n(3 - ax_n^2)/2,$$

which avoids division, also minimizing the relative error.

More recently [10] discuss using absolute instead of relative error for the classical square root iteration, attempting to minimize the absolute error after a predetermined number of iterations. They concentrate on approximations in the form of linear functions, and a very small number of iterations ( $n = 1, 2$ ).

Due to the increased interest in speeding up division, algorithms based on obtaining good reciprocals has spurred a lot of activity in also obtaining good initial values for the Newton-Raphson reciprocal iteration

$$x_{n+1} = x_n(2 - ax_n).$$

In 1994 [16] developed explicit formulas for the optimal starting values for this iteration, as functions of the number  $n$  of iterations, and the interval  $(a, b)$

$$\beta_n = \frac{a^{2^{-n}} + b^{2^{-n}}}{a^{2^{-n}b} + b^{2^{-n}a}}, \quad (2)$$

and [3] discuss the construction of initial value tables for reciprocation.

Here we shall develop similar optimal starting values for obtaining roots of the function

$$\phi(x) = x^p - a,$$

i.e., for use in Newton-Raphson iterations to approximate  $f(a) = a^{\frac{1}{p}}$ .

In general we find the following iteration

$$x_{n+1} = \frac{x_n}{p} \left( p - 1 + \frac{a}{x_n^p} \right),$$

which specializes into

$$\underline{p = -1}$$

$$\phi(x) = \frac{1}{x} - a \quad \text{and iteration} \quad x_{n+1} = x_n(2 - ax_n)$$

This sequence goes to  $1/a$ : hence it can be used for computing reciprocals;

$$\underline{p = 2}$$

$$\phi(x) = x^2 - a \quad \text{and iteration} \quad x_{n+1} = \frac{1}{2} \left( x_n + \frac{a}{x_n} \right).$$

This sequence goes to  $\sqrt{a}$ . Note that this iteration requires a division, usually a fairly “expensive” operation, and thus often avoided.

$$\underline{p = -2}$$

$$\phi(x) = \frac{1}{x^2} - a \quad \text{and iteration} \quad x_{n+1} = \frac{x_n}{2} \left( 3 - ax_n^2 \right).$$

This sequence goes to  $1/\sqrt{a}$ . It is also frequently used to compute  $\sqrt{a}$ , obtained by multiplying the final result by  $a$ .

To make the iterations converge quickly, we have to make sure that  $x_0$  is close enough to the wanted result. It is also important to make sure that the number of required iterations is a small constant. This is frequently done by using the first, say  $k$ , bits of the input value  $a$  to address a table of suitable initial values. Hence, for all the input values with the same first  $k$  bits (they constitute some interval  $[a_{\min}, a_{\max}]$ ), the iterations will be started with the same  $x_0$ . A natural choice consists in choosing the value of  $x_0$  that minimizes

$$\max_{a \in [a_{\min}, a_{\max}]} |f(a) - x_0|.$$

If  $f$  is monotone, this is traditionally done (e.g., [3]) by taking  $x_0$  equal to the arithmetic mean

$$\frac{1}{2} (f(a_{\min}) + f(a_{\max})).$$

As said above, this minimizes the maximum possible distance between  $x_0$  and  $f(a)$ . And yet, if we perform  $n$  iterations, what really matters is to minimize the maximum possible distance between  $x_n$  and  $f(a)$ . In the following, we develop expressions for starting values for a specific number of iterations. These choices turns out to be much better than the natural choice. In the case of reciprocation, we actually find again the optimal choice of Eqn. (2) from [16].

## 2 Estimating the error

We wish to compute

$$\alpha = a^{1/p},$$

where  $p$  is a nonzero integer ( $p$  can be either positive or negative). This will be done by computing the zero of

$$\phi(x) = x^p - a,$$

using the Newton-Raphson iteration. The obtained iteration is

$$x_{n+1} = \frac{x_n}{p} (p - 1 + ax_n^{-p}). \quad (3)$$

We wish to find the best starting point for  $a \in [a_{\min}, a_{\max}]$ , assuming we will perform  $n$  iterations. To do that, we want to estimate  $|x_n - \alpha|$  from  $|x_0 - \alpha|$ .

Since the Newton-Raphson iteration has a quadratic convergence (that is, if  $x_0$  is close to  $\alpha$ , then  $|x_{n+1} - \alpha|$  is roughly proportional to the square of  $x_n - \alpha$ ), we shall try to estimate the coefficient of proportionality.

From (3), we get

$$\begin{aligned}
\frac{x_{n+1} - \alpha}{(x_n - \alpha)^2} &= \frac{1}{2} \frac{p-1}{\alpha} - \frac{1}{6} \frac{p^2-1}{\alpha^2} (x_n - \alpha) \\
&+ \frac{1}{24} \frac{(p+2)(p^2-1)}{\alpha^3} (x_n - \alpha)^2 \\
&- \frac{1}{120} \frac{(p+2)(p+3)(p^2-1)}{\alpha^4} (x_n - \alpha)^3 \\
&+ O((x_n - \alpha)^4)
\end{aligned} \tag{4}$$

The formula shows that if  $p = -1$  (i.e., in the case of the computation of a reciprocal), the coefficient of proportionality is a constant (it does not depend on  $x_n$ ). In that particular case, the solutions given later will be exact, not approximate.

For  $p \neq -1$  we haven't succeeded in getting from (4) a direct expression for  $x_n - \alpha$  in terms of  $x_0 - \alpha$ . And yet, since we assume that the interval  $[a_{\min}, a_{\max}]$  is small, it makes sense to assume that, as soon as  $n \geq 1$ , the terms

$$\begin{aligned}
&-\frac{1}{6} \frac{p^2-1}{\alpha^2} (x_n - \alpha) + \frac{1}{24} \frac{(p+2)(p^2-1)}{\alpha^3} (x_n - \alpha)^2 \\
&- \frac{1}{120} \frac{(p+2)(p+3)(p^2-1)}{\alpha^4} (x_n - \alpha)^3 \\
&+ O((x_n - \alpha)^4)
\end{aligned} \tag{5}$$

become negligible compared to  $(p-1)/(2\alpha)$ . Also, we may assume that for  $n = 0$ , the terms

$$\begin{aligned}
&\frac{1}{24} \frac{(p+2)(p^2-1)}{\alpha^3} (x_0 - \alpha)^2 \\
&- \frac{1}{120} \frac{(p+2)(p+3)(p^2-1)}{\alpha^4} (x_0 - \alpha)^3 \\
&+ O((x_0 - \alpha)^4)
\end{aligned} \tag{6}$$

can be neglected compared to

$$-\frac{1}{6} \frac{p^2-1}{\alpha^2} (x_0 - \alpha)$$

Thus we have

$$x_1 - \alpha \approx \left( \frac{p-1}{2\alpha} - \frac{p^2-1}{6\alpha^2} (x_0 - \alpha) \right) (x_0 - \alpha)^2 \quad (7)$$

and, for  $n \geq 1$ :

$$x_{n+1} - \alpha \approx \frac{p-1}{2\alpha} (x_n - \alpha)^2 \quad (8)$$

From (7) and (8), we find

$$\begin{aligned} x_n - \alpha &\approx \left( \frac{p-1}{2\alpha} \right)^{2^{n-1}-1} \\ &\times \left( \frac{p-1}{2\alpha} - \frac{p^2-1}{6\alpha^2} (x_0 - \alpha) \right)^{2^{n-1}} \\ &\times (x_0 - \alpha)^{2^n} \end{aligned} \quad (9)$$

Now, we have to find a starting point  $x_0$  that minimizes the maximum absolute value of  $|x_n - \alpha|$  (the maximum is taken for all  $a \in [a_{\min}, a_{\max}]$ , i.e., for all  $\alpha \in [a_{\min}^{1/p}, a_{\max}^{1/p}]$  — by convention, if  $y < x$ , then  $[x, y]$  is the interval  $[y, x]$ ).

It can be shown that the maximum value is attained for  $\alpha = a_{\min}^{1/p}$  or  $\alpha = a_{\max}^{1/p}$ , hence it will be minimized when the values for  $\alpha = a_{\min}^{1/p}$  and  $\alpha = a_{\max}^{1/p}$  are equal. Denoting  $\alpha_{\min} = a_{\min}^{1/p}$  and  $\alpha_{\max} = a_{\max}^{1/p}$  we get the following equation

$$\begin{aligned} &\left( \frac{p-1}{2\alpha_{\min}} \right)^{2^{n-1}-1} \left( \frac{p-1}{2\alpha_{\min}} - \frac{p^2-1}{6\alpha_{\min}^2} (x_0 - \alpha_{\min}) \right)^{2^{n-1}} \\ &= \\ &\left( \frac{p-1}{2\alpha_{\max}} \right)^{2^{n-1}-1} \left( \frac{p-1}{2\alpha_{\max}} - \frac{p^2-1}{6\alpha_{\max}^2} (x_0 - \alpha_{\max}) \right)^{2^{n-1}} \end{aligned} \quad (10)$$

After some simplifications, this equation becomes

$$\begin{aligned} & \alpha_{\max}^{1-1/2^{n-1}} \left( \frac{3}{\alpha_{\min}} - (x_0 - \alpha_{\min}) \frac{p+1}{\alpha_{\min}^2} \right) (x_0 - \alpha_{\min})^2 \\ & \qquad \qquad \qquad = \\ & \pm \alpha_{\min}^{1-1/2^{n-1}} \left( \frac{3}{\alpha_{\max}} - (x_0 - \alpha_{\max}) \frac{p+1}{\alpha_{\max}^2} \right) (x_0 - \alpha_{\max})^2 \end{aligned} \tag{11}$$

This new equation is a  $3^r$  degree polynomial equation in  $x_0$  (or more precisely, a set of two  $3^r$  degree equations, depending on the “ $\pm$ ”). It is therefore very easily solvable numerically, obtaining the root located in the interval  $[a_{\min}^{1/p}, a_{\max}^{1/p}]$ .

Now, let us as an example focus on the case of reciprocation. This is what we do in practice, and we call  $\beta_n$  the obtained starting point for  $n$  iterations.

### 3 Example, $p = -1$ , Newton-Raphson Reciprocation

As mentioned above, Newton-Raphson iteration for computing the reciprocal of a number  $a$  consists in performing the iteration

$$x_{n+1} = x_n(2 - ax_n) \tag{12}$$

In practice, when we wish to compute the reciprocal of a number  $a$  that will be assumed to be between 1 and 2, the first  $k$  bits of the binary representation of  $a - 1$  (the “implicit one” being omitted) are used as address bits to find in a table an adequate value of the *seed*  $x_0$ . This means that the same  $x_0$  will be used for all values of  $a$  in an interval

$$[a_{\min}, a_{\max}],$$

with  $a_{\max} - a_{\min}$  of the form  $2^{-k}$  in the most frequent cases. Fig. 1 shows that the choice of the starting point can have a huge influence on the final approximation error (for other values of  $p$ , we may get very similar figures).

As said in the introduction, it is frequently suggested to choose the arithmetic mean, e.g., as used in [3],

$$\beta_0 = \frac{1}{2} \left( \frac{1}{a_{\min}} + \frac{1}{a_{\max}} \right).$$

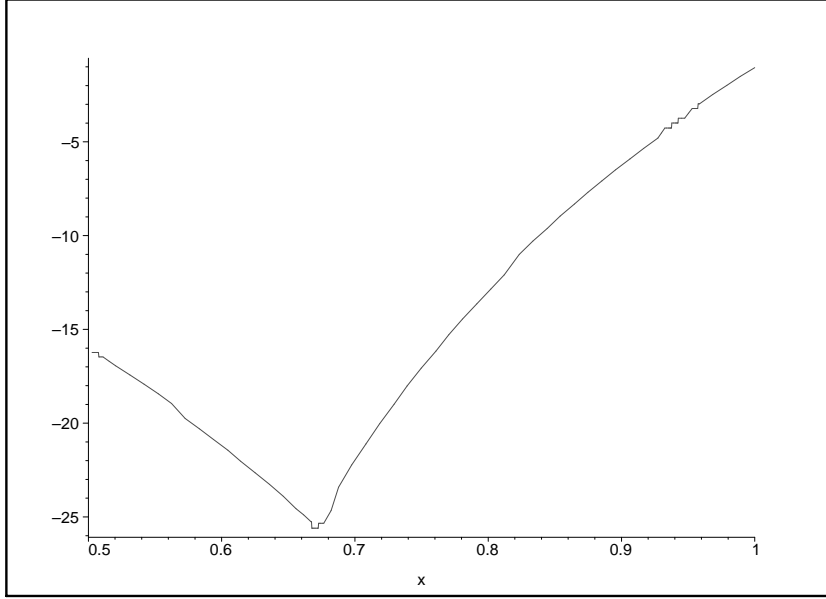


Fig. 1. Radix-2 logarithm of the maximum distance (for all  $a$  in  $[1, 2]$ ) between iterate  $x_4$  and  $1/a$ , depending on the choice of  $x_0$  in  $[1/2, 1]$ .

Let us try to minimize the distance between  $x_n$  and  $1/a$ . First, let us compute that distance. From (12), we get

$$x_{n+1} - \frac{1}{a} = 2x_n - ax_n^2 - \frac{1}{a} = -a \left( x_n - \frac{1}{a} \right)^2,$$

which is the very same equation as we would obtain with  $p = -1$  from (4).

Hence, by induction

$$x_n - \frac{1}{a} = -a^{2^n-1} \left( x_0 - \frac{1}{a} \right)^{2^n}. \quad (13)$$

What we now have to find is the value  $x_0$  (between  $1/a_{\min}$  and  $1/a_{\max}$ ) such that the maximum value (for  $a$  between  $a_{\min}$  and  $a_{\max}$ ) of  $|x_n - 1/a|$  is as small as possible. By examining the derivative of function:

$$g(a) = a^{2^n-1} \left( x_0 - \frac{1}{a} \right)^{2^n}$$

one immediately deduces that, for a given  $x_0$ , the maximum value of  $|x_n - 1/a|$  is obtained for  $a = a_{\min}$  or  $a = a_{\max}$ .

That is, the maximum error is either

$$E_1 = a_{\min}^{2^n-1} \left( x_0 - \frac{1}{a_{\min}} \right)^{2^n}$$



or

$$E_2 = a_{\max}^{2^n-1} \left( x_0 - \frac{1}{a_{\max}} \right)^{2^n}.$$

As before, this maximum value will be minimized when  $E_1 = E_2$ . This gives an equation that  $x_0$  must satisfy to be the best starting point for  $n$  iterations

$$a_{\min}^{2^n-1} \left( x_0 - \frac{1}{a_{\min}} \right)^{2^n} = a_{\max}^{2^n-1} \left( x_0 - \frac{1}{a_{\max}} \right)^{2^n}. \quad (14)$$

To solve this equation define

$$\lambda_n = a_{\min}^{1-2^{-n}} \quad \text{and} \quad \mu_n = a_{\max}^{1-2^{-n}}$$

From (14) we get

$$\left[ \lambda_n x_0 - \frac{\lambda_n}{a_{\min}} \right]^{2^n} = \left[ \mu_n x_0 - \frac{\mu_n}{a_{\max}} \right]^{2^n}.$$

And, since

$$\frac{1}{a_{\max}} \leq x_0 \leq \frac{1}{a_{\min}}$$

this gives

$$\lambda_n x_0 - \frac{\lambda_n}{a_{\min}} = \frac{\mu_n}{a_{\max}} - \mu_n x_0.$$

This is now very easily solved, and gives

$$x_0 = \frac{\frac{\mu_n}{a_{\max}} + \frac{\lambda_n}{a_{\min}}}{\lambda_n + \mu_n}.$$

From this we deduce the following result, which is identical to the result quoted above from [16].

**Theorem 1** *The maximum possible distance between  $x_n$  and  $1/a$  is smallest when  $x_0$  is equal to the number*

$$\beta_n = \frac{a_{\max}^{2^{-n}} + a_{\min}^{2^{-n}}}{a_{\max}^{2^{-n}} a_{\min} + a_{\min}^{2^{-n}} a_{\max}}. \quad (15)$$

Some values of  $\beta_n$  are of particular interest:

- $\beta_0$  is the arithmetic mean of  $1/a_{\min}$  and  $1/a_{\max}$ : we find again (which is not surprising) the value that minimizes the maximum distance between  $1/a$  and  $x_0$ ;

- $\beta_1$  is the geometric mean of  $1/a_{\min}$  and  $1/a_{\max}$ , that is,

$$\beta_1 = \frac{1}{\sqrt{a_{\min}a_{\max}}}.$$

- the limit value (when  $n \rightarrow \infty$ ) of  $\beta_n$  is

$$\beta_{\infty} = \frac{2}{a_{\min} + a_{\max}}$$

that is, the reciprocal of the midpoint of the interval  $[a_{\min}, a_{\max}]$ . This shows (and this will be confirmed below by the experiments) that this “naive” choice for  $x_0$  is far from being naive, and turns out to be a much better choice than the sophisticated value  $\beta_0$  that minimizes the maximum distance between  $1/a$  and  $x_0$ .

### 3.1 First example: $a_{\min} = 1$ and $a_{\max} = 2$ .

This example corresponds to the direct computations of reciprocals of mantissas of floating-point numbers without any tabulation. By (15) we find the following starting values

$$\begin{cases} \beta_0 = 3/4 \\ \beta_1 = 1/\sqrt{2} \\ \beta_2 = 0.68644\dots \\ \beta_3 = 0.67642\dots \\ \beta_{\infty} = 2/3 \end{cases}$$

We get, depending on the choice of  $x_0$ , the following approximation errors:

$x_0$	$\max  x_1 - 1/a $	$\max  x_2 - 1/a $	$\max  x_3 - 1/a $	$\max  x_4 - 1/a $	$\max  x_5 - 1/a $
$\beta_0$	$1.25 \times 10^{-1}$	$3.12 \times 10^{-2}$	$1.95 \times 10^{-3}$	$7.63 \times 10^{-6}$	$1.16 \times 10^{-10}$
$\beta_1$	<b><math>8.56 \times 10^{-2}</math></b>	$1.47 \times 10^{-2}$	$4.33 \times 10^{-4}$	$3.75 \times 10^{-7}$	$2.82 \times 10^{-13}$
$\beta_2$	$9.83 \times 10^{-2}$	<b><math>9.67 \times 10^{-3}</math></b>	$1.87 \times 10^{-4}$	$6.98 \times 10^{-8}$	$9.76 \times 10^{-15}$
$\beta_3$	$1.05 \times 10^{-1}$	$1.10 \times 10^{-2}$	<b><math>1.20 \times 10^{-4}</math></b>	$2.89 \times 10^{-8}$	$1.67 \times 10^{-15}$
$\beta_4$	$1.08 \times 10^{-1}$	$1.16 \times 10^{-2}$	$1.36 \times 10^{-4}$	<b><math>1.83 \times 10^{-8}</math></b>	$6.75 \times 10^{-16}$
$\beta_5$	$1.10 \times 10^{-1}$	$1.20 \times 10^{-2}$	$1.44 \times 10^{-4}$	$2.07 \times 10^{-8}$	<b><math>4.28 \times 10^{-16}</math></b>
$\beta_{\infty}$	$1.11 \times 10^{-1}$	$1.23 \times 10^{-2}$	$1.52 \times 10^{-4}$	$2.32 \times 10^{-8}$	$5.40 \times 10^{-16}$

Observe that the minimal values of the maximum errors occur after  $n$  iterations, when  $\beta_n$  is used as the starting value (emphasized in bold face).

For performing 5 iterations, choosing  $\beta_5$  is 272245 times more accurate than choosing  $\beta_0$ . This corresponds to more than 18 bits of difference in accuracy.

3.2 Second example:  $a_{\min} = 3/2$  and  $a_{\max} = 7/4$

Of course, when  $a_{\max} - a_{\min}$  decreases, the difference tends to be reduced (since the interval where  $x_0$  can lie shrinks). This is shown in the following table:

$x_0$	$\max  x_1 - 1/a $	$\max  x_2 - 1/a $	$\max  x_3 - 1/a $	$\max  x_4 - 1/a $	$\max  x_5 - 1/a $
$\beta_0$	$3.97 \times 10^{-3}$	$2.76 \times 10^{-5}$	$1.33 \times 10^{-9}$	$3.09 \times 10^{-18}$	$1.67 \times 10^{-35}$
$\beta_1$	<b><math>3.67 \times 10^{-3}</math></b>	$2.36 \times 10^{-5}$	$9.71 \times 10^{-10}$	$1.65 \times 10^{-18}$	$4.76 \times 10^{-36}$
$\beta_2$	$3.81 \times 10^{-3}$	<b><math>2.17 \times 10^{-5}</math></b>	$8.26 \times 10^{-10}$	$1.19 \times 10^{-18}$	$2.49 \times 10^{-36}$
$\beta_3$	$3.87 \times 10^{-3}$	$2.25 \times 10^{-5}$	<b><math>7.61 \times 10^{-10}</math></b>	$1.01 \times 10^{-18}$	$1.80 \times 10^{-36}$
$\beta_4$	$3.91 \times 10^{-3}$	$2.29 \times 10^{-5}$	$7.89 \times 10^{-10}$	<b><math>9.33 \times 10^{-19}</math></b>	$1.52 \times 10^{-36}$
$\beta_5$	$3.93 \times 10^{-3}$	$2.31 \times 10^{-5}$	$8.03 \times 10^{-10}$	$9.67 \times 10^{-19}$	<b><math>1.40 \times 10^{-36}</math></b>
$\beta_\infty$	$3.94 \times 10^{-3}$	$2.33 \times 10^{-5}$	$8.17 \times 10^{-10}$	$1.00 \times 10^{-18}$	$1.51 \times 10^{-36}$

#### 4 The general case of other roots

In the following we shall now look at other cases of finding roots of equations of the form

$$\phi(x) = x^p - a$$

for alternative values of  $p$ . For  $p \geq 2$  or  $p < -1$  recall that we can solve the  $3^{\text{rd}}$  degree polynomials (11) numerically, but that the starting values obtained this way are only approximations, as the error estimates of (9) are solutions to slightly perturbed problems.

The table below shows some starting values  $\beta_n$  for  $a_{\min} = 1$  and  $a_{\max} = 2$  for various values of  $p$  and  $0 \leq n \leq 5$ , together with the limiting values  $\beta_\infty$ .

	$p = -3$	$p = -2$	$p = -1$	$p = 2$	$p = 3$
$\beta_0$	0.89685026	0.85355339	3/4	1.20710678	1.12996052
$\beta_1$	0.88695734	0.83671927	0.70710678	1.20829381	1.13288765
$\beta_2$	0.88401897	0.83051406	0.68644244	1.19901822	1.12904943
$\beta_3$	0.88255736	0.82744145	0.67642857	1.19439264	1.12713081
$\beta_4$	0.88182871	0.82591381	0.67151443	1.19208497	1.12617201
$\beta_5$	0.88146495	0.82515229	0.66908205	1.19093267	1.12569277
$\beta_\infty$	0.88110158	0.82439236	2/3	1.18978149	1.12521367

#### 4.1 Case $p = -2$ , square root reciprocal

The conventional iteration  $x_{n+1} = \frac{1}{2}(x_n + \frac{a}{x_n})$  for square root is not frequently used, since it requires a division at each step, and division is significantly slower than multiplication on almost all systems. Hence one may prefer the following iteration:

$$x_{n+1} = \frac{x_n}{2} (3 - ax_n^2), \quad (16)$$

converging to  $1/\sqrt{a}$ . To get  $\sqrt{a}$  it suffices to multiply the final result by  $a$ .

We have performed the Newton-Raphson iteration with the starting values obtained above, and found the following maximum errors, with  $a_{\min} = 1$  and  $a_{\max} = 2$  we obtain:

$x_0$	$\max  x_1 - \frac{1}{\sqrt{a}} $	$\max  x_2 - \frac{1}{\sqrt{a}} $	$\max  x_3 - \frac{1}{\sqrt{a}} $	$\max  x_4 - \frac{1}{\sqrt{a}} $	$\max  x_5 - \frac{1}{\sqrt{a}} $
$\beta_0$	$4.86 \times 10^{-2}$	$4.90 \times 10^{-3}$	$5.09 \times 10^{-5}$	$5.49 \times 10^{-9}$	$6.39 \times 10^{-17}$
$\beta_1$	<b><math>3.78 \times 10^{-2}</math></b>	$2.98 \times 10^{-3}$	$1.88 \times 10^{-5}$	$7.50 \times 10^{-10}$	$1.19 \times 10^{-18}$
$\beta_2$	$4.07 \times 10^{-2}$	<b><math>2.45 \times 10^{-3}</math></b>	$1.26 \times 10^{-5}$	$3.37 \times 10^{-10}$	$2.41 \times 10^{-19}$
$\beta_3$	$4.21 \times 10^{-2}$	$2.62 \times 10^{-3}$	<b><math>1.03 \times 10^{-5}</math></b>	$2.24 \times 10^{-10}$	$1.06 \times 10^{-19}$
$\beta_4$	$4.28 \times 10^{-2}$	$2.71 \times 10^{-3}$	$1.10 \times 10^{-5}$	<b><math>1.82 \times 10^{-10}</math></b>	$6.99 \times 10^{-20}$
$\beta_5$	$4.32 \times 10^{-2}$	$2.75 \times 10^{-3}$	$1.14 \times 10^{-5}$	$1.95 \times 10^{-10}$	<b><math>5.68 \times 10^{-20}</math></b>
$\beta_\infty$	$4.35 \times 10^{-2}$	$2.80 \times 10^{-3}$	$1.18 \times 10^{-5}$	$2.08 \times 10^{-10}$	$6.50 \times 10^{-20}$

Repeating the computations, but now for a smaller interval,  $a_{\min} = 1$  and  $a_{\max} = 1 + 2^{-4}$  we find the following much smaller maximal errors.

$x_0$	$\max  x_1 - \frac{1}{\sqrt{a}} $	$\max  x_2 - \frac{1}{\sqrt{a}} $	$\max  x_3 - \frac{1}{\sqrt{a}} $	$\max  x_4 - \frac{1}{\sqrt{a}} $	$\max  x_5 - \frac{1}{\sqrt{a}} $
$\beta_0$	$3.46 \times 10^{-4}$	$1.85 \times 10^{-7}$	$8.96 \times 10^{-19}$	$4.37 \times 10^{-27}$	$2.96 \times 10^{-53}$
$\beta_1$	<b><math>3.39 \times 10^{-4}</math></b>	$1.78 \times 10^{-7}$	$8.77 \times 10^{-19}$	$3.72 \times 10^{-27}$	$2.13 \times 10^{-53}$
$\beta_2$	$3.42 \times 10^{-4}$	<b><math>1.75 \times 10^{-7}</math></b>	$8.70 \times 10^{-19}$	$3.49 \times 10^{-27}$	$1.89 \times 10^{-53}$
$\beta_3$	$3.43 \times 10^{-4}$	$1.77 \times 10^{-7}$	<b><math>8.67 \times 10^{-19}</math></b>	$3.39 \times 10^{-27}$	$1.77 \times 10^{-53}$
$\beta_4$	$3.44 \times 10^{-4}$	$1.77 \times 10^{-7}$	$8.69 \times 10^{-19}$	<b><math>3.34 \times 10^{-27}</math></b>	$1.72 \times 10^{-53}$
$\beta_5$	$3.44 \times 10^{-4}$	$1.78 \times 10^{-7}$	$8.70 \times 10^{-19}$	$3.36 \times 10^{-27}$	<b><math>1.69 \times 10^{-53}</math></b>
$\beta_\infty$	$3.44 \times 10^{-4}$	$1.78 \times 10^{-7}$	$8.70 \times 10^{-19}$	$3.39 \times 10^{-27}$	$1.72 \times 10^{-53}$

Although the effect of using the optimal starting value is much less significant here over a narrower interval, again we find the minimal values occurring after  $n$  iterations when using  $\beta_n$  as the starting point.

#### 4.2 Cube root reciprocal

With  $a_{\min} = 1$  and  $a_{\max} = 2$  for  $p = -3$  we obtain:

$x_0$	$\max  x_1 - \frac{1}{\sqrt[3]{a}} $	$\max  x_2 - \frac{1}{\sqrt[3]{a}} $	$\max  x_3 - \frac{1}{\sqrt[3]{a}} $	$\max  x_4 - \frac{1}{\sqrt[3]{a}} $	$\max  x_5 - \frac{1}{\sqrt[3]{a}} $
$\beta_0$	$2.92 \times 10^{-2}$	$2.10 \times 10^{-3}$	$1.11 \times 10^{-5}$	$3.09 \times 10^{-10}$	$2.41 \times 10^{-19}$
$\beta_1$	<b><math>2.37 \times 10^{-2}</math></b>	$1.39 \times 10^{-3}$	$4.83 \times 10^{-6}$	$5.88 \times 10^{-11}$	$8.71 \times 10^{-21}$
$\beta_2$	$2.49 \times 10^{-2}$	<b><math>1.22 \times 10^{-3}</math></b>	$3.71 \times 10^{-6}$	$3.47 \times 10^{-11}$	$3.04 \times 10^{-21}$
$\beta_3$	$2.55 \times 10^{-2}$	$1.28 \times 10^{-3}$	<b><math>3.26 \times 10^{-6}</math></b>	$2.65 \times 10^{-11}$	$1.78 \times 10^{-21}$
$\beta_4$	$2.58 \times 10^{-2}$	$1.31 \times 10^{-3}$	$3.42 \times 10^{-6}$	<b><math>2.34 \times 10^{-11}</math></b>	$1.36 \times 10^{-21}$
$\beta_5$	$2.59 \times 10^{-2}$	$1.32 \times 10^{-3}$	$3.50 \times 10^{-6}$	$2.45 \times 10^{-11}$	<b><math>1.20 \times 10^{-21}</math></b>
$\beta_\infty$	$2.61 \times 10^{-2}$	$1.34 \times 10^{-3}$	$3.58 \times 10^{-6}$	$2.57 \times 10^{-11}$	$1.32 \times 10^{-21}$

In this case, if we perform 5 iterations, starting the iterations from  $\beta_5$  leads to a result that is 201 times more accurate than starting with  $\beta_0$ .

#### 4.3 Square root

With  $a_{\min} = 1$  and  $a_{\max} = 2$  for  $p = 2$  we obtain:

$x_0$	$\max  x_1 - \sqrt{a} $	$\max  x_2 - \sqrt{a} $	$\max  x_3 - \sqrt{a} $	$\max  x_4 - \sqrt{a} $	$\max  x_5 - \sqrt{a} $
$\beta_0$	$1.78 \times 10^{-2}$	$1.55 \times 10^{-4}$	$1.20 \times 10^{-8}$	$7.23 \times 10^{-17}$	$2.61 \times 10^{-33}$
$\beta_1$	<b><math>1.80 \times 10^{-2}</math></b>	$1.58 \times 10^{-4}$	$1.25 \times 10^{-8}$	$7.85 \times 10^{-17}$	$3.08 \times 10^{-33}$
$\beta_2$	$1.93 \times 10^{-2}$	<b><math>1.34 \times 10^{-4}</math></b>	$9.00 \times 10^{-9}$	$4.05 \times 10^{-17}$	$8.21 \times 10^{-34}$
$\beta_3$	$2.02 \times 10^{-2}$	$1.43 \times 10^{-4}$	<b><math>7.58 \times 10^{-9}</math></b>	$2.88 \times 10^{-17}$	$4.14 \times 10^{-34}$
$\beta_4$	$2.07 \times 10^{-2}$	$1.49 \times 10^{-4}$	$7.87 \times 10^{-9}$	<b><math>2.42 \times 10^{-17}</math></b>	$2.92 \times 10^{-34}$
$\beta_5$	$2.09 \times 10^{-2}$	$1.53 \times 10^{-4}$	$8.24 \times 10^{-9}$	$2.40 \times 10^{-17}$	<b><math>2.45 \times 10^{-34}</math></b>
$\beta_\infty$	$2.12 \times 10^{-2}$	$1.56 \times 10^{-4}$	$8.61 \times 10^{-9}$	$2.62 \times 10^{-17}$	$2.43 \times 10^{-34}$

Notice that in this case  $\beta_5$  is slightly better than  $\beta_4$  for 4 iterations, and that  $\beta_\infty$  (and  $\beta_6$  but it is not shown in the table) is slightly better than

$\beta_5$  for 5 iterations. The same phenomenon occurs for  $\beta_1$  where  $\beta_0$  is a slightly better starting point. This is obviously an effect of solving a slightly perturbed problem.

#### 4.4 Fifth roots

With  $a_{\min} = 1$  and  $a_{\max} = 2$  we obtain:

$x_0$	$\max  x_1 - \sqrt[5]{a} $	$\max  x_2 - \sqrt[5]{a} $	$\max  x_3 - \sqrt[5]{a} $	$\max  x_4 - \sqrt[5]{a} $	$\max  x_5 - \sqrt[5]{a} $
$\beta_0$	$1.10 \times 10^{-2}$	$2.08 \times 10^{-4}$	$7.51 \times 10^{-8}$	$9.82 \times 10^{-15}$	$1.68 \times 10^{-28}$
$\beta_1$	<b><math>1.03 \times 10^{-2}</math></b>	$2.07 \times 10^{-4}$	$8.53 \times 10^{-8}$	$1.46 \times 10^{-14}$	$4.24 \times 10^{-28}$
$\beta_2$	$1.06 \times 10^{-2}$	<b><math>1.94 \times 10^{-4}</math></b>	$7.52 \times 10^{-8}$	$1.13 \times 10^{-14}$	$2.56 \times 10^{-28}$
$\beta_3$	$1.08 \times 10^{-2}$	$1.99 \times 10^{-4}$	<b><math>7.05 \times 10^{-8}</math></b>	$9.95 \times 10^{-15}$	$1.98 \times 10^{-28}$
$\beta_4$	$1.09 \times 10^{-2}$	$2.03 \times 10^{-4}$	$7.15 \times 10^{-8}$	<b><math>9.33 \times 10^{-15}</math></b>	$1.74 \times 10^{-28}$
$\beta_5$	$1.09 \times 10^{-2}$	$2.05 \times 10^{-4}$	$7.29 \times 10^{-8}$	$9.24 \times 10^{-15}$	<b><math>1.63 \times 10^{-28}</math></b>
$\beta_\infty$	$1.10 \times 10^{-2}$	$2.07 \times 10^{-4}$	$7.42 \times 10^{-8}$	$9.59 \times 10^{-15}$	$1.60 \times 10^{-28}$

In this case, although  $\beta_n$  is always a better starting point than  $\beta_0$  for  $n$  iterations, the difference is negligible.

## Conclusion

We have suggested a strategy for getting optimal starting points for Newton-Raphson-based iterations for approximating  $a^{1/p}$ . In many cases choosing these values, results in much smaller approximation errors, than using traditional seed values.

## References

- [1] W.J. Cody. Double Precision Square Root for the CDC-3600), *Communications of the ACM*, 7(12):715–718, 1964.
- [2] A. Dahan-Dalmedico and J. Peiffer *Histoire des Mathématiques*. (in French) Editions du Seuil, Paris, 1986.
- [3] D. DasSarma and D.W. Matula. Measuring the Accuracy of ROM Reciprocal Tables. *IEEE Transactions on Computers*, 43(8):932–940, August 1994.
- [4] J. Eve. Starting Approximations for the Iterative Calculation of Square Roots. *Computer J.*, 6:274–276, Oct. 1963.

- [5] C.T. Fike. Starting Approximations for Square Root Calculation on IBM System/360. *Communications of the ACM*, 9(4):297–299, April 1966.
- [6] D. Fowler and E. Robson. Square root approximations in old Babylonian mathematics: YBC 7289 in context. *Historia Mathematica*, 25:366–378, 1998.
- [7] H.J. Maehly. Approximations for the CDC 1604. Technical report, Control Data Corp., 1960.
- [8] P. W. Markstein. Computation of Elementary Functions on the IBM RISC System/6000 Processor. *IBM Journal of Research and Development*, 34(1):111–119, January 1990.
- [9] P. Markstein. *IA-64 and Elementary Functions : Speed and Precision*. Hewlett-Packard Professional Books. Prentice Hall, 2000. ISBN: 0130183482.
- [10] P. Montuschi and M. Mezzalama. Optimal Absolute Error Starting Values for Newton-Raphson Calculation of Square Root. *Computing*, 46:67–86, 1991.
- [11] D.G. Moursund. Optimal Starting Values for Newton-Raphson Calculation of  $\sqrt{x}$ . *Communications of the ACM*, 10(7):430–432, July 1967.
- [12] I. Newton. *Methodus Fluxionem et Serierum Infinitarum*. 1664-1671.
- [13] S. F. Oberman. Floating-point division and square root algorithms and implementation in the AMD-k7 microprocessor. In Koren and Kornerup, editors, *Proceedings of the 14th IEEE Symposium on Computer Arithmetic (Adelaide, Australia)*, pages 106–115, Los Alamitos, CA, April 1999. IEEE Computer Society Press.
- [14] C. V. Ramamoorthy, J. R. Goodman, and K. H. Kim. Some properties of iterative square-rooting methods using high-speed multiplication. *IEEE Transactions on Computers*, C-21:837–847, 1972. Reprinted in E. E. Swartzlander, *Computer Arithmetic*, Vol. 1, IEEE Computer Society Press Tutorial, Los Alamitos, CA, 1990.
- [15] D. Russinoff. A mechanically checked proof of IEEE compliance of a register-transfer-level specification of the AMD-k7 floating-point multiplication, division, and square root instructions. *LMS Journal of Computation and Mathematics*, 1:148–200, 1998.
- [16] M.J. Schulte, J. Omar, and E.E. Swartzlander. Optimal Initial Approximation for the Newton-Raphson Division Algorithm. *Computing*, 53:233–242, 1994.
- [17] P. Sebah and X. Gourdon. Newton’s method and high order iterations. Technical report, 2001.  
<http://numbers.computation.free.fr/Constants/Algorithms/newton.html>.
- [18] P.H. Sterbenz and C.T. Fike. Optimal Starting Approximations for Newtons Method. *Math. Comp.*, 23:313–318, 1969.
- [19] M. Wayne Wilson. Optimal Starting Approximations for Generating Square Root for Slow or No Divide. *Communications of the ACM*, 13(9):559–560, September 1970.