Algorithms for Square Root Extraction

Patrick Johnson

April 28, 2010

Introduction

This paper will explain several methods for calculating the square roots of positive real numbers and prove that the results of the algorithms converge to the square root.

Contents

1 Greek Method

$\forall\, a \in \mathbb{Q}$ such that $a > 1$, the sequences $\{a_n\}$ and $\{b_n\}$ both converge to $\sqrt{a}$ when defined as:
$a_i = a$
$b_i = 1$
$a_{n+1} = \frac{a_n + b_n}{2}$
$b_{n+1} = \frac{2a_n b_n}{a_n + b_n}$

1.1 The Product is Always a

$\forall\, n \in \mathbb{N},\ a_n b_n = a$

Base Case for Proof by Induction

$n = 1$:

$$a_1 b_1 = a \times 1 = a$$

Induction Step

Assume $a_k b_k = a$, then

$$a_{k+1} b_{k+1} = \left(\frac{a_k + b_k}{2}\right)\left(\frac{2a_k b_k}{a_k + b_k}\right) = a_k b_k = a$$

1.2 a_n is Always Greater Than b_n

$\forall n \in \mathbb{N}, a_n \geq b_n$

Base Case for Induction

$n = 1$:

$$a_1 = a > b_1 = 1$$

Induction Step

Assume $a_k > b_k$, then

$$(a_k - b_k)^2 > 0$$

$$a_k^2 - 2a_k b_k + b_k^2 > 0$$

$$a_k^2 + 2a_k b_k + b_k^2 > 4a_k b_k$$

$$(a_k + b_k)^2 > 4ab_k$$

$$\frac{a_k + b_k}{2} > \frac{2a_k b_k}{a_k + b_k}$$

$$a_{k+1} > b_{k+1}$$

1.3 $\{a_n\}$ are $\{b_n\}$ Monotone

$\{a_n\}$ is Strictly Decreasing

Since $a_n > b_n$ $\forall n \in \mathbb{N}$:

$$a_n > b_n$$

$$2a_n > a_n + b_n$$

$$a_n > \frac{a_n + b_n}{2}$$

$$a_n > a_{n+1}$$

$\{b_n\}$ is Strictly Increasing

Since $a_n > b_n$ $\forall n \in \mathbb{N}$:

$$b_n < a_n$$

$$b_n^2 < a_n b_n$$

$$a_n b_n + b_n^2 < 2a_n b_n$$

$$b_n \left(a_n + b_n\right) < 2a_n b_n$$

$$b_n < \frac{2a_n b_n}{a_n + b_n}$$

$$b_n < b_{n+1}$$

1.4 $\{a_n\}$ and $\{b_n\}$ are Cauchy

We know $a_n > b_n$ and $a_n b_n = a$, so:

$$a_n > b_n$$

$$a_n > \frac{a}{a_n}$$

$$a_n^2 > a$$

$$a_n > \sqrt{a}$$

Similarly, $b_n < \sqrt{a}$ for all n. Thus $\{a_n\}$ is monotone decreasing and bounded below and$\{b_n\}$ is monotone increasing and bounded above. Both sequences must be Cauchy, and since $\mathbb{R}$ is complete, both sequences must converge.

1.5 $\{a_n\}$ and $\{b_n\}$ Converge to $\sqrt{a}$

$\{a_n^2\}$ Converges to a

Since $\{a_n\}$ is a Cauchy sequence, $\forall\,\varepsilon > 0,\ \exists\,N_1 \in \mathbb{N}$ such that $\forall\,n, m > N_1,\ |a_n - a_m| < \frac{1}{2a}\varepsilon$. Also, $\{a_n\}$ is strictly decreasing, so if $m > n$, $a_m < a_n$ and therefore $|a_n - a_m| = a_n - a_m$. From this:

$$\begin{aligned}
\frac{1}{2a}\varepsilon &> a_n - a_{n+1} \\
\frac{1}{2a}\varepsilon &> a_n - \frac{a_n + b_n}{2} \\
\frac{1}{2a}\varepsilon &> \frac{2a_n}{2} - \frac{a_n + \frac{a}{a_n}}{2} \\
\frac{1}{2a}\varepsilon &> \frac{\frac{a}{a_n} - a_n}{2} \\
\varepsilon &> a\left(\frac{a}{a_n} - a_n\right) \\
\varepsilon &> \frac{a}{a_n}\left(a - a_n^2\right) \\
\lim_{n\to\infty} a_n^2 &= a
\end{aligned}$$

$\{b_n^2\}$ Converges to a

Since $\{b_n\}$ is a Cauchy sequence, $\forall\,\varepsilon > 0,\ \exists\ N_2 \in \mathbb{N}$ such that $\forall\,n, m > N_2,\ |b_n - b_m| < \varepsilon$. Also, $\{b_n\}$ is strictly increasing, so if $m > n$, $b_m > b_n$ and therefore $|b_n - b_m| = b_m - b_n$. From this:

$$\begin{aligned}
\varepsilon &> b_{n+1} - b_n \\
\varepsilon &> \frac{2a_n b_n}{a_n + b_n} - b_n \\
\varepsilon &> \frac{2\frac{a}{b_n} b_n}{\frac{a}{b_n} + b_n} - b_n \\
\varepsilon &> \frac{2ab_n}{a + b_n^2} - \frac{\left(a + b_n^2\right) b_n}{a + b_n^2} \\
\varepsilon &> \frac{\left(a - b_n^2\right) b_n}{a + b_n^2}
\end{aligned}$$

Since b_n is increasing and bounded above, it does not converge to zero nor does b_n^2 diverge to infinity. Also, a is constant so $\left(a - b_n^2\right)$ must converge to zero, which means:

$$\lim_{n\to\infty} b_n^2 = a$$

$\{a_n\}$ and $\{b_n\}$ Converge to $\sqrt{a}$

Since $\lim_{n\to\infty} a_n^2 = \lim_{n\to\infty} b_n^2 = a$, we have shown that the squares of both sequences converge to a, and thus each sequence must converge to $\sqrt{a}$.

1.6 Alternate Proof that $\lim_{n\to\infty} a_n = \lim_{n\to\infty} b_n = \sqrt{a}$

Since $\{a_n\}$ is a Cauchy sequence, $\forall\,\varepsilon > 0,\ \exists\,N \in \mathbb{N}$ such that $\forall\,n, m > N,\ |a_n - a_m| < \frac{1}{2}\varepsilon$. The sequence $\{a_n\}$ is monotone decreasing so if $m > n$, $a_m < a_n$ and therefore $|a_n - a_m| = a_n - a_m$. Thus:

$$\begin{aligned}
\frac{1}{2}\varepsilon &> a_n - a_{n+1} \\
\frac{1}{2}\varepsilon &> a_n - \frac{a_n + b_n}{2} \\
\frac{1}{4}\varepsilon^2 &> \left(\frac{a_n - b_n}{2}\right)^2 \\
\frac{1}{4}\varepsilon^2 &> \frac{a_n^2 - 2a_n b_n + b_n^2}{4} \\
\varepsilon^2 &> a_n^2 - 2a + b_n^2 \\
\varepsilon^2 &> (a_n - b_n)^2 \\
\varepsilon &> a_n - b_n \\
\lim_{n\to\infty} a_n &= \lim_{n\to\infty} b_n
\end{aligned}$$

Since $\lim_{n\to\infty} a_n = \lim_{n\to\infty} b_n$, and $a_n b_n = a$,

$$\lim_{n\to\infty} a_n^2 = \lim_{n\to\infty} b_n^2 = \lim_{n\to\infty} a_n b_n = a$$

$$\lim_{n\to\infty} a_n = \lim_{n\to\infty} b_n = \sqrt{a}$$

2 Newton's Method[1]

Newton's method for finding the square root of a is to begin by making an initial guess b_1 such that $1 \leq b_1 \leq a$. Ideally, b_1 will be the integer nearest a, but initial proximity only reduces the number of iterations necessary to achieve the desired accuracy. Dividing a by b_1 and averaging the quotient with b_n will result in a better approximation for $\sqrt{a}$. This is repeated until the desired accuracy is achieved. To state the algorithm as a recursive sequence,

$$b_{n+1} = \frac{b_n + \frac{a}{b_n}}{2}$$

To prove that the algorithm works, it shall first be shown that it works for the trivial case where the initial guess, b_1, is equal to $\sqrt{a}$. Then a full proof for any value of b_1 such that $1 \leq b_1 \leq \sqrt{a}$ shall be provided.

2.1 $b_1 = \sqrt{a}$

Since the algorithm allows any initial guess between one and a, it is possible that $b_1$is equal to the square root of a. In this trivial case, each term of the sequence is identically $\sqrt{a}$ as shown by starting with $b_k = \sqrt{a}$:

$$b_{k+1} = \frac{b_k + \frac{a}{b_k}}{2} = \frac{\sqrt{a} + \frac{a}{\sqrt{a}}}{2} = \frac{\sqrt{a} + \sqrt{a}}{2} = \sqrt{a}$$

Since $b_1 = \sqrt{a}$, $b_n = \sqrt{a}\ \forall\, n \in \mathbb{N}$ by induction. Incidentally, this is the only case where *any* terms of the sequence will equal $\sqrt{a}$.

2.2 The Tail of $\{b_n\}$ is Bounded Below

For most initial guesses, b_1 will not be equal to $\sqrt{a}$ and it must be shown that $\{b_n\}$ converges to $\sqrt{a}$. In this case we can let $\delta_n = \sqrt{a} - b_n$. For any b_n, we have:

$$\begin{aligned}
\delta_n &\geq 0 \\
\delta_n^2 &\geq 0 \\
\delta_n^2 + 2a - 2\sqrt{a} &\geq 2a - 2\sqrt{a}\delta_n \\
a - 2\sqrt{a} + \delta_n^2 + a &\geq 2\sqrt{a}\left(\sqrt{a} - \delta_n\right) \\
\left(\sqrt{a} - \delta_n\right)^2 + a &\geq 2\sqrt{a}\left(\sqrt{a} - \delta_n\right) \\
\frac{\left(\sqrt{a} - \delta_n\right)^2 + a}{\sqrt{a} - \delta_n} &\geq 2\sqrt{a} \\
\sqrt{a} - \delta_n + \frac{a}{\sqrt{a} - \delta_n} &\geq 2\sqrt{a} \\
\frac{\sqrt{a} - \delta_n + \frac{a}{\sqrt{a}-\delta_n}}{2} &\geq \sqrt{a} \\
\frac{b_n + \frac{a}{b_n}}{2} &\geq \sqrt{a} \\
b_{n+1} &\geq \sqrt{a}
\end{aligned}$$

[1]Also known as the Babylonian method or Heron's method

Thus if we ignore the first term, the tail of the sequence is bounded below by $\sqrt{a}$. From this $\frac{a}{b_n}$ must be bounded above by $\sqrt{a}$ for all $n > 1$.

$$\begin{aligned} b_n &\geq \sqrt{a} \\ b_n\sqrt{a} &\geq a \\ \sqrt{a} &\geq \frac{a}{b_n} \end{aligned}$$

2.3 The Tail of $\{b_n\}$ is Monotone Decreasing

Next, it shall be shown that this tail is decreasing. Since $b_n \geq \sqrt{a} \geq \frac{a}{b_n}\ \forall\, n > 1$,

$$\begin{aligned} b_n &\geq \frac{a}{b_n} \\ 2b_n &\geq b_n + \frac{a}{b_n} \\ b_n &\geq \frac{b_n + \frac{a}{b_n}}{2} \\ b_n &\geq b_{n+1} \end{aligned}$$

2.4 $\{b_n\}$ Converges to $\sqrt{a}$

Since the tail of $\{b_n\}$ is bounded below and monotone decreasing, it is a Cauchy sequence and $\forall\, \varepsilon > 0,\ \exists\ N \in \mathbb{N}$ such that $\forall\, n, m > N,\ |b_n - b_m| < \frac{1}{2a}\varepsilon$. Also, if $m > n$, then $b_m < b_n$ and therefore $|b_n - b_m| = b_n - b_m$. Finally, $a_n/b_n \geq 1\ \forall\, n$, so:

$$\begin{aligned} \frac{1}{2a}\varepsilon &\geq b_n - b_{n+1} \\ \frac{1}{2a}\varepsilon &\geq \frac{2b_n}{2} - \frac{b_n + \frac{a}{b_n}}{2} \\ \frac{1}{2a}\varepsilon &\geq \frac{b_n - \frac{a}{b_n}}{2} \\ \varepsilon &\geq a\left(b_n - \frac{a}{b_n}\right) \\ \varepsilon &\geq \frac{a}{b_n}\left(b_n^2 - a\right) \geq b_n^2 - a \\ \varepsilon &\geq b_n^2 - a \end{aligned}$$

$$\lim_{n\to\infty} b_n^2 = a$$

$$\lim_{n\to\infty} b_n = \sqrt{a}$$

$$\{b_n\} \to \sqrt{a}$$

3 Bahkshali Method

The Bahkshali formula was originally written to give the approximate square root as a single expression, although iteration is possible. As originally presented,

$$\sqrt{a} = \sqrt{b^2 + d} \approx b + \frac{d}{2b} - \frac{\left({}^{d}\!/_{2b}\right)^2}{2\left(b + \frac{d}{2b}\right)}$$

To find $\sqrt{a}$, one must first guess at b and then determine d by the relation $d = a - b^2$. The approximate square root is then given by the formula. The result from the approximation can be used as b for a subsequent iteration.

3.1 Sequential Form

To iterate for a more accurate solution, a sequence can be defined as:

$$\begin{aligned} d_n &= a - b_n^2 \\ b_{n+1} &= b_n + \frac{d_n}{2b_n} - \frac{\left(\frac{d_n}{2b_n}\right)^2}{2\left(b_n + \frac{d_n}{2b_n}\right)} \end{aligned}$$

Rewriting the sequence in terms of only a and b_n, we have:

$$\begin{aligned} b_{n+1} &= b_n + \frac{d_n}{2b_n} - \frac{\left(\frac{d_n}{2b_n}\right)^2}{2\left(b_n + \frac{d_n}{2b_n}\right)} \\ &= b_n + \frac{d_n}{2b_n} - \frac{d_n^2}{8b_n^3 + 4b_n d_n} \\ &= b_n + \frac{a - b_n^2}{2b_n} - \frac{\left(a - b_n^2\right)^2}{8b_n^3 + 4b_n\left(a - b_n^2\right)} \\ &= b_n + \frac{a - b_n^2}{2b_n} - \frac{a^2 - 2ab_n^2 + b_n^4}{4b_n^3 + 4ab_n} \\ &= b_n + \frac{a - b_n^2}{2b_n} - \frac{a^2 - 2ab_n^2 + b_n^4}{4b_n\left(b_n^2 + a\right)} \\ &= \frac{4b_n^2\left(b_n^2 + a\right) + 2\left(a - b_n^2\right)\left(b_n^2 + a\right) - \left(a - b_n^2\right)^2}{4b_n^3 + 4ab_n} \\ &= \frac{a^2 + 6ab_n^2 + b_n^4}{4b_n^3 + 4ab_n} \end{aligned}$$

This form of the equation might not be as elegant, but being in standard form makes it simpler to find an equivalent expression.

3.2 Newton's Method Revisited

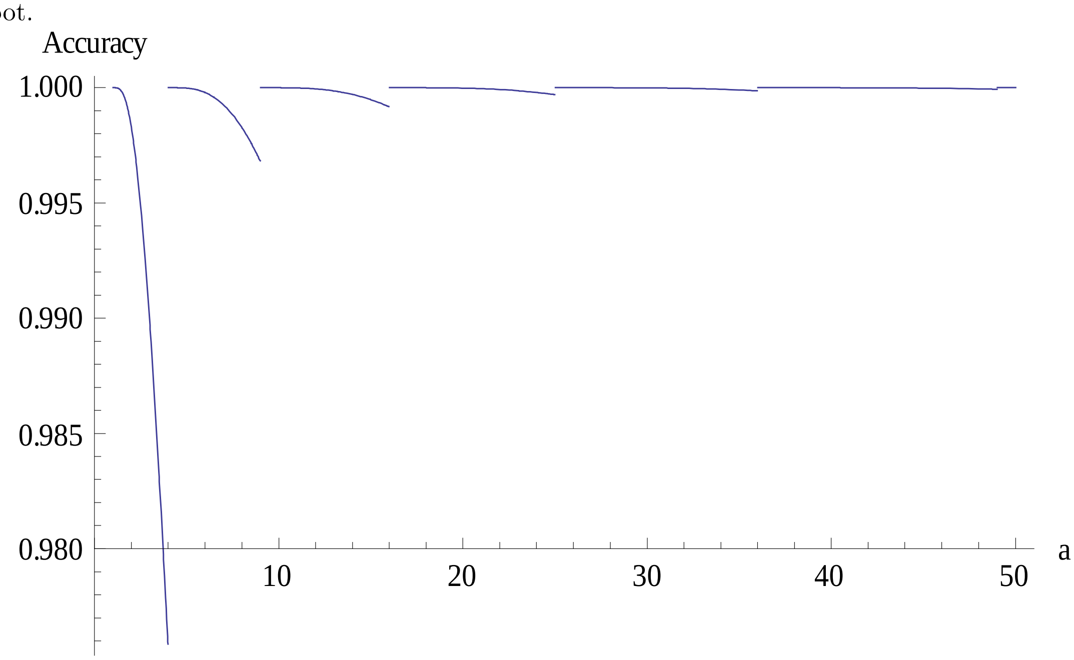
$$
\begin{aligned}
b_{n+2} &= \frac{b_{n+1} + {}^{a}\!/_{b_{n+1}}}{2} = \frac{\frac{b_n + {}^{a}\!/_{b_n}}{2} + \frac{a}{\frac{b_n + {}^{a}\!/_{b_n}}{2}}}{2} \\
&= \frac{b_n + {}^{a}\!/_{b_n} + \frac{4a}{b_n + {}^{a}\!/_{b_n}}}{4} \\
&= \frac{b_n}{4} + \frac{a}{4b_n} + \frac{ab_n}{b_n^2 + a} \\
&= \frac{b_n^2\left(b_n^2 + a\right) + a\left(b_n^2 + a\right) + 4ab_n^2}{4b_n\left(b_n^2 + a\right)} \\
b_{n+2} &= \frac{a^2 + 6ab_n^2 + b_n^4}{4b_n^3 + 4ab_n}
\end{aligned}
$$

The unexpected result shown here is that the Bahkshali formula is equivalent to two iterations of Newton's method. As it has been shown that Newton's method converges to the square root, it is clear that the Bahkshali method will also converge.

3.3 Efficiency

The similarity to Newton's method makes this an ineffective algorigthm for computers, which can perform simple operations very quickly. A single iteration of the Bahkshali method requires a larger stack and more operations than two iterations of Newton's method, which is algebraically identical.

The functionality of the formula is in the repetition of the terms $\frac{d}{2b}$ and $b + \frac{d}{2b}$. This makes it convenient for hand calculations, where this repetition can save a substantial amount of time. Provided a reasonable first guess, the Bahkshali formula provides a very accurate approximation, as shown in the following table. For creating the chart, b was taken as the floor function of the square root.



4 High School Method[2]

The High School method is a procedural algorithm for computing square roots, a digit-by-digit computation that ensures each digit is accurate as it is calculated. It also terminates when the exact square root has been found. (Of course this only occurs when the root is a rational number with a finite decimal expansion.)

4.1 Procedure

1. Divide the number, a, into pairs of digits, positioned to keep the decimal point between pairs.
2. The first digit, d_1, of the root, b, is the largest integer whose square is less than the first pair of digits. (The first "*pair*" may only be one digit.)
3. Square the first digit and write it below the first pair of digits, then subtract.
4. Bring down the next pair of digits to complete the difference, similar to long division.
5. The next digit, d_2, of the root is the largest integer such that $d_2\,(20d_1 + d_2)$ is less than or equal to the completed difference.
6. Subtract the product from step 5 from the difference, then bring down the next pair. Zeroes can be appended to the end to form pairs, exactly as for long division.
7. The next digit is the largest integer such that $d_3\,(200d_1 + 20d_2 + d_3)$ is less than or equal to the difference from the previous step with the appended digits. Note that $200d_1 + 20d_2 + d_3$ is twice $(d_1 d_2)$ with d_3 appended.
8. This continues until the desired accuracy is reached or the difference is 0.

When the residue is zero and there are no more pairs of digits to bring down, the exact square root has been found. This makes this procedure different from the other methods discussed, which approach the square root but never equal it.

Example

	3 2. 4
	1 0 4 9. 7 6
3	9
	1 4 9
62	1 2 4
	2 5 7 6
644	2 5 7 6
	0

Note that in the example the values $20d_1 + d_2$ and $200d_1 + 20d_2 + d_3$ are written down the left margin to aid in keeping track.

It is clear from the method of construction that the sequence is monotone increasing, but showing that this method of choosing digits results in a sequence whose supremum is the square root of a isn't so clear. The algorithm is simply a tabular form for managing the terms of a nested binomial expansion. When each digit is chosen, it is chosen so that the value calculated thus far is never greater than the square root.

[2]Perhaps this is a misnomer, as it seem most high schools now teach the Babylonian divide and average method

4.2 Binomial Expansion

The two digit decimal represented by $d_1d_2$is equal to $10d_1 + d_2$, so its square equals:

$$\begin{aligned} 'd_1d_2'^2 &= (10d_1 + d_2)^2 \\ &= 100d_1^2 + 20d_1d_2 + d_2^2 \\ &= 100d_1^2 + (20d_1 + d_2)\, d_2 \end{aligned}$$

Furthermore, the square of a three digit number, $d_1d_2d_3$is equal to $100d_1 + 10d_2 + d_3$, so the square equals

$$\begin{aligned} 'd_1d_2d_3'^2 &= (100d_1 + 10d_2 + d_3)^2 \\ &= (10\,(10d_1 + d_2) + d_3)^2 \\ &= 100\,(10d_1 + d_2)^2 + 20\,(10d_1 + d_2)\, d_3 + d_3^2 \\ &= 100\,(10d_1 + d_2)^2 + (200d_1 + 20d_2 + d_3)\, d_3 \\ &= 100'd_1d_2'^2 + (20'd_1d_2' + d_3) + d_3 \end{aligned}$$

The relationship between these expansions and the High School method is made more obvious by the following example, where the digits have been replaced by variables.

	$d_1 \quad d_2. \quad d_3$
	$a_1a_2\ a_3a_4.\ a_5a_6$
d_1	$-d_1^2$
	$a_1a_2 - d_1^2\ \ a_3a_4$
$20d_1 + d_2$	$-\,(20d_1 + d_2)\, d_2$
	$\dots$
$200d_1 + 20d_2 + d_3$	$-\,(20'd_1d_2' + d_3)\, d_3$

Looking at the extraction performed with variables, it is easy to see that for the second iteration the two values subtracted sum to the binomial expansion of $'d_1d_2'^2$. Note that instead of multiplying the previous sum by 100, the new values are shifted over two decimal places. On the third iteration, the sum of all terms subtracted is equal to $'d_1d_2d_3'^2$. The tabular arrangement aligns the values with respect to the decimal place, and the procedure subtracts the additional terms of the expansion in succession. Having demonstrated how the procedure works, it is now time for a more rigorous proof.

4.3 Proof of the Method

First we need to account for the magnitude of a. Let $k = \lfloor \log_{100} a \rfloor + 1$, i.e. k is such that $100^{k-1} \leq a < 100^k$. Furthermore, k will be the number of digits in the integer portion of the square root as k is also the number of pairs of digits to the left of the decimal point

Accounting for the position of the decimal place, d_1 actually represents $'d_1' \times 10^{k-1}$, $d_2$represents $'d_2' \times 10^{k-2}$, and so on. For clarity, the digit $'d_i'$ in single quotes will represent the single digit and the variable d_i will represent $'d_i' \times 10^{k-i}$

Let $\{b_n\}$be defined as $b_n = \sum_{i=1}^{n} d_i$. Obviously $\{b_n\}$ is an increasing function, and it shall be shown that it converges to $\sqrt{a}$.

The procedure for finding the first digit of the root is equivalent to choosing $'d_1'$such that $d_1^2 \leq a < \left(d_1 + 10^{k-1}\right)^2$. The second inequality ensures that if a higher digit would work, that should be d_1. The next step, subtracting the square of $'d_1'$ from the first pair of digits is identical to subtracting d_1^2 from a. The first digits of $a - d_1^2$ will be the same as the remainder term with the next pair of digits brought down in the algorithm. In the tabular form, the zeroes are excluded and digits are only brought down as necessary.

The next $'d_n'$ is chosen as high as possible so that the product of the new digit and the sum of 20 times the digits already found plus the new digit is less than what is left from the subtraction. This can be expressed as:

$$d_n \times (2b_{n-1} + d_n) \le a - b_{n-1}^2 < \left(d_n + 10^{k-n}\right) \times \left(2b_{n-1} + \left(d_n + 10^{k-n}\right)\right)$$

The difference between b_{n-1} and b_n is d_n, so we can take:

$$\begin{aligned} b_n - b_{n-1} &= d_n \\ b_n &= d_n + b_{n-1} \\ b_n^2 &= d_n^2 + 2d_n b_{n-1} + b_{n-1}^2 \\ b_n^2 - 2b_{n-1}d_n - d_n^2 &= b_{n-1}^2 \end{aligned}$$

Substituting this into our inequality,

$$\begin{aligned} d_n \times (2b_{n-1} + d_n) &\le a - b_{n-1}^2 \\ 2b_{n-1}d_n + d_n^2 &\le a - b_n^2 + 2b_{n-1}d_n + d_n^2 \\ \\ b_n^2 &\le a \\ b_n &\le \sqrt{a} \end{aligned}$$

So $'d_n'$ is chosen so that $b_n \le \sqrt{a}$ and therefore bounded above. If we focus on the other side of the inequality (the strict inequality), we get:

$$\begin{aligned} a - b_{n-1}^2 &< \left(d_n + 10^{k-n}\right) \times \left(2b_{n-1} + \left(d_n + 10^{k-n}\right)\right) \\ a - b_{n-1}^2 &< 2b_{n-1}\left(d_n + 10^{k-n}\right) + \left(d_n + 10^{k-n}\right)^2 \\ a - b_{n-1}^2 &< 2b_{n-1}d_n + 2 \times 10^{k-n}b_{n-1} + d_n^2 + 2 \times 10^{k-n}d_n + 100^{k-n} \\ a - b_n^2 + 2b_{n-1}d_n + d_n^2 &< 2b_{n-1}d_n + 2 \times 10^{k-n}b_{n-1} + d_n^2 + 2 \times 10^{k-n}d_n + 100^{k-n} \\ a - b_n^2 &< 2 \times 10^{k-n}b_{n-1} + 2 \times 10^{k-n}d_n + 100^{k-n} \\ a - b_n^2 &< 10^{k-n}(2b_{n-1} + 2d_n + 10^{k-n}) \\ a - b_n^2 &< 10^{k-n}(2b_n + 10^{k-n}) \end{aligned}$$

$\{b_n\}$ is bounded above by $\sqrt{a}$, so for any $\varepsilon > 0$, $\exists\, N \in \mathbb{N}$ such that $10^{k-N} < \frac{\varepsilon}{2\sqrt{a}}$, thus $\forall\, n > N$:

$$10^{k-N} < \frac{\varepsilon}{2\sqrt{a}} \le \frac{\varepsilon}{2b_n} < \frac{\varepsilon}{2b_n + 10^{k-N}}$$

$$10^{k-N}\left(2b_n + 10^{k-N}\right) < \varepsilon$$

$$a - b_n^2 < \varepsilon$$

Since the limit of $\left\{b_n^2\right\}$ is a, the limit of $\{b_n\}$ is $\sqrt{a}$ and the procedure has been proven.