PI AND ARCHIMEDES POLYGON METHOD

KYUTAE PAUL HAN

1. Introduction

Archimedes was not the first to use a method involving polygons. Other Greek mathematicians before him had attempted to use the area of polygons inside a circle to approximate π. However he was the first to use the perimeter of the polygon in an attempt to find the magic ratio between the diameter and the circumference. And his was the primary method that mathematicians from Europe to Middle-East to China used to find the digits of π for the next 1900 years. I sought to convert his geometric formulas into simpler iterations, so that the method as a whole is easier to analyze. Then I seek to figure out the complexity of his method, and make a stab at how many calculations van Ceulen must have computed to get 35 digits of π using the method.

2. Upper Bound for Pi: Perimeter of a circumscribed Polygon

Before we start on Archimedes' proof, we first need to prove one of Euclid's theorems, though for our purposes we only have to prove the sufficient statement.

Theorem 1. *If a straight line bisects an angle of a triangle and cuts the base then the resulting segments of the base have the same ratio as the remaining sides of the triangle.*

Proof. If we have $\triangle AOC$ and OD bisects $\angle AOC$, then $\frac{CO}{OA} = \frac{CD}{DA}$.

First thing we do is draw a line starting from C that is parallel to OD, and also extend AO so that the two lines meet at E.

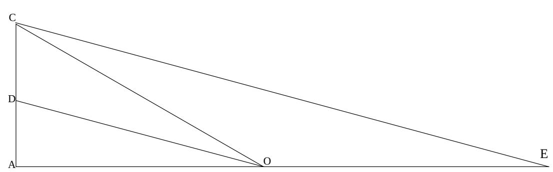


Figure 2.1. $\triangle AOC$ with the extra lines drawn in

Then:

$$\begin{aligned} m\angle DOA &= m\angle CEA && \text{Corresponding angles} \\ m\angle COD + m\angle DOA + m\angle COE &= 180 && \text{Supplementary angles} \\ m\angle OCE + m\angle OEC + m\angle COE &= 180 && \text{Triangle} \\ m\angle COD + m\angle DOA &= m\angle OCE + m\angle OEC && \text{Substitution} \\ \triangle COE &\text{ is isoceles} && \text{Sub. and def. of isos. tri.} \end{aligned}$$

With AA similarity Theorem, $\triangle CEA$ is similar to $\triangle DOA$. Therefore $\frac{CA}{AE} = \frac{DA}{AO} \Rightarrow \frac{CA}{DA} = \frac{AE}{AO}$. $CA = CD + DA$ and $AE = AO + OE$.

$$\begin{aligned} \frac{CD+DA}{DA} &= \frac{AO+OE}{AO} && \text{Sub.} \\ \frac{CD}{DA} &= \frac{OE}{AO} && \text{Distrib. then subt.} \\ \frac{CD}{DA} &= \frac{CO}{OA} && \text{Isosceles tri.} \end{aligned}$$

□

2.1. **Original Geometric Method of Archimedes.** We can now move on to Archimedes's geometric method/proof to measure pi. We shall start with the circumscribed polygons.

Date: May 28th, 2013.

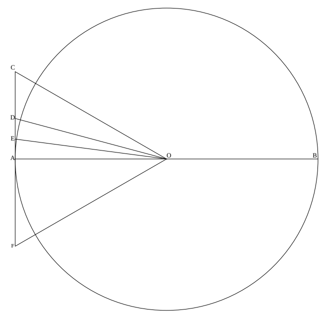


Figure 2.2. Circumscribed Case

Look to Figure 2.2. Let AB be the diameter, let O be the center, let AC be tangent at A; and let the $\angle AOC$ be one-third of a right angle or $\frac{\pi}{6}$. First it is important to note that assuming $CF = 2 * CA$, then CF is the side of a circumscribed hexagon (Figure 2.3). The $\frac{OA}{AC}$ ratio is very important, because that is the ratio of $\frac{\text{Half of the diameter (radius)}}{\text{Half of the side of a circumscribed polygon}}$ which is equivalent to $\frac{\text{Diameter}}{\text{Length of a side of a circumscribed polygon}}$. We take the reciprocal and multiply the result by the number of sides and we get $\frac{\text{Perimeter of circumscribed polygon}}{\text{Diameter}}$ - an upper bound for a value of π, since π is the ratio$=\frac{Circumference}{Diameter}$.

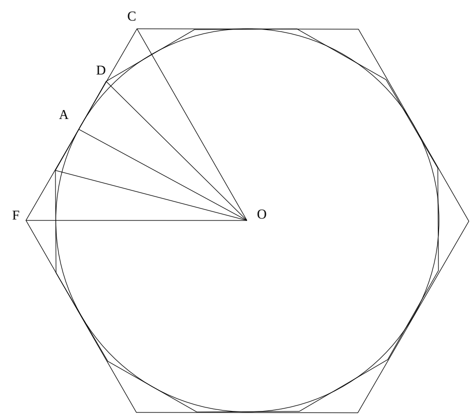


Figure 2.3. Circumscribed Hexagon and 12-gon

Before I give you Archimedes's geometric method, it is helpful to keep in mind what he is attempting to do. He starts with a known ratio for $\frac{OA}{AC}$ and $\frac{OC}{CA}$. Then he bisects $\angle AOC$ with OD, and using the initial ratios, he gets $\frac{OA}{AD}$. Since he doesn't just want to stop at $\frac{OA}{AD}$, he needs to find $\frac{OD}{DA}$, which serves the same role as $\frac{OC}{CA}$. Then he could iteratively use his method over and over, bisecting the triangle again and again, which means the polygon that the ratio pertains to is of greater n sides (double the number of sides to be exact, because you bisect the angle each time).

Suppose we bisect $\angle AOC$ using OD and make D stop at AC (Figure 2.2). Using Euclid's theorem $\frac{CD}{DA} = \frac{CO}{OA}$ Archimedes makes a brilliant observation. From this (or from our proof of Euclid's), he figures out $\frac{CD+DA}{DA} = \frac{CO+OA}{OA} = \frac{CA}{DA} \Rightarrow \frac{CO+OA}{CA} = \frac{OA}{DA}$. Therefore:

$$\frac{CO}{CA} + \frac{OA}{AC} = \frac{OA}{AD} \tag{2.1}$$

Additionally Archimedes found a way to get $\frac{OD}{DA}$ by using the Pythagorean Theorem.

$$OA^2 + AD^2 = OD^2$$

$$\Rightarrow \frac{OA^2}{AD^2} + 1 = \frac{OD^2}{DA^2}$$

$$\sqrt{\frac{OA^2}{AD^2} + 1} = \frac{OD}{DA} \tag{2.2}$$

Archimedes' proof does not assume to know what $m\angle AOC$ was.

Archimedes starts with an approximation. $\frac{OA}{AC} > \frac{265}{153}$. Notice how $\frac{OA}{AC} = cot(\frac{\pi}{6})$. We, as mathematicians endowed with the knowledge of trigonometry, know that $\frac{OA}{AC} = \sqrt{3}$. Archimedes knew that $\sqrt{3} > \frac{265}{153}$. And this is okay because then our approximation for the reciprocal $\frac{AC}{OA}$ ($\frac{153}{265}$) is bigger than the actual value ($\frac{1}{\sqrt{3}}$). Since the reciprocal times the number of sides (in this case 6) is an upper bound, Archimedes' approximation cannot be incorrect. You end up with an upper bound that is slightly bigger than the actual upper bound, had you been exact. And he knows that $\frac{OC}{CA} = \frac{2}{1} = \frac{306}{153}$, for us this is $csc(\frac{\pi}{6}) = \frac{1}{sin(\frac{\pi}{6})}$. Use (2.1):

$$\frac{306}{153} + \frac{265}{153} < \frac{OA}{AD}$$
$$\frac{571}{153} < \frac{OA}{AD}$$

Notice how if we only wanted to find the ratio $\frac{\text{Perimeter of a circumscribing regular 12-gon}}{\text{Diameter}}$, then we could stop here and get that ratio by taking $12 * \frac{AD}{OA}$ (Figure 2.3). This is because we bisected $\angle AOC$, which means $m\angle DOA = \frac{\pi}{12}$. Also notice how $\frac{OA}{AD} = cot(\frac{\pi}{12})$. Now to get $\frac{OD}{DA}$, use (2.2)

Archimedes once again uses bounded approximations. $\frac{OA^2}{AD^2} + 1 > \frac{349450}{23409} \Rightarrow \sqrt{\frac{OA^2}{AD^2} + 1} > \frac{\sqrt{349450}}{153}$. Archimedes knew that $\sqrt{349450} > 591\frac{1}{8}$. So $\frac{OD}{DA} > \frac{591\frac{1}{8}}{153}$. Also notice how $\frac{OD}{DA} = csc(\frac{\pi}{12})$.

And bounding $\frac{OD}{DA}$ and $\frac{OA}{AD}$ with a lower bound is good because at the next step, when he bisects the $\triangle AOD$ again by OE, by the same reasoning as before, $\frac{OA}{AE} = \frac{OD}{DA} + \frac{OA}{AD} \Rightarrow \frac{OA}{AE} > \frac{591\frac{1}{8}}{153} + \frac{571}{153} = \frac{1162\frac{1}{8}}{153}$. So then the approximation of the reciprocal is bigger than the real value.

All of this works inductively, because notice how we never had to depend on the specific values for AC, CO, OA, or $m\angle AOC$ for the proof of the big statements (2.1 and 2.2)

Bisect again to get a smaller triangle $\triangle AOE$, part of a 24-gon circumscribed the circle. Then use $\frac{OD}{DA}$ and $\frac{OA}{AD}$ and (2.1) to get $\frac{OA}{AE}$. Then use $\frac{OA}{AE}$ and (2.2) again to find $\frac{OE}{EA}$. From our reasoning before it is easy to see that $\frac{\text{Perimeter of a circumscribed regular 24-gon}}{\text{Diameter}} = 24 * \frac{AE}{OA}$.

Archimedes does this bisection two more times (each time with an appropriate approximation) to get a triangle that is a part of a regular 96-gon and its respective ratios. This is how he got his upper bound $3\frac{1}{7}$ for the value of π.

2.2. **The Trigonometric View on Archimedes Method for Upper Bound for Pi.** At each step we noted that the ratios we are using have trigonometric equivalents. For example, $\frac{OA}{AC}$ is equal to $cot(\frac{\pi}{6})$. We could use it to find $\frac{\text{Perimeter of circumscribed 6-gon}}{\text{Diameter}}$, by taking the reciprocal and multiplying by 6. A similar thing happened for $\frac{OA}{AD}$; it equaled $cot(\frac{\pi}{12})$, and its reciprocal multipled by 12 was equal to $\frac{\text{Perimeter of circumscribed 12-gon}}{\text{Diameter}}$. This is due to each bisection (or each step in the method) creating a triangle that is a part of double the number of sides polygon, while cutting the angle in half. This eases us into the claim that $\frac{\text{Perimeter of a circumscribed regular n-gon}}{\text{Diameter}}$ is equivalent to $ntan(\frac{\pi}{n})$.

Proof. Let's say the circumscribed regular n-gon has vertices $v_1, v_2, v_3 ... v_n$. Since it's regular, I can arbitrarily label consecutive vertices v_i and v_{i+1} traveling from edge to edge in a cycle, without losing generality (the last edge will be from v_n to v_1). The circle has center O. Some properties of regular circumscribed polygons include that every edge is tangent to the circle at its midpoint[1]. Additionally, the radius and the tangent line are perpendicular to each other. Therefore all $\triangle v_i O v_{i+1}$ where $0 \leq i \leq n-1$, are congruent triangles, including $\triangle v_n O v_1$. Then $m\angle v_i O v_{i+1} = \frac{2\pi}{n}$ where $0 \leq i \leq n-1$. $m\angle v_n O v_1 = \frac{2\pi}{n}$.

That means $tan(\frac{\pi}{n})$ is equivalent to $\frac{\text{Half of the side of a circumscribed polygon}}{\text{Half of the diameter (radius)}} = \frac{\text{Side of circumscribed polygon}}{\text{Diameter}}$. Multiply by n and you get:

$$ntan(\frac{\pi}{n}) = \frac{\text{Perimeter of a circumscribed regular n-gon}}{\text{Diameter}} \tag{2.3}$$

□

For now let's call $ntan(\frac{\pi}{n}) = UB_n$.

If you relook over the whole proof with a trigonometric view, you can see that Archimedes is proving some trig identities. The $\frac{OC}{CA} + \frac{OA}{AC} = \frac{OA}{AD}$ becomes:

$$csc(\theta) + cot(\theta) = cot(\frac{\theta}{2}) \tag{2.4}$$

[1]The apothem is the incircle's radius.

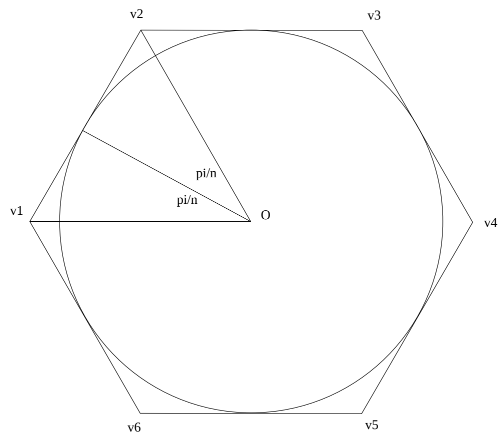


FIGURE 2.4. Circumscribed Hexagon

$\sqrt{\frac{OA^2}{AD^2}+1} = \frac{OD}{DA}$ becomes:

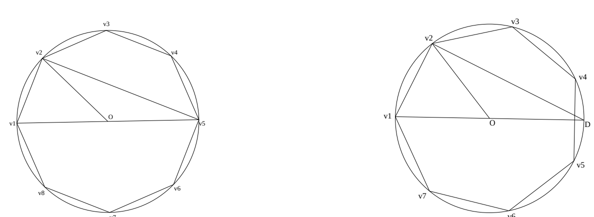
$$\sqrt{cot^2(\theta)+1} = csc(\theta) \tag{2.5}$$

3. LOWER BOUND FOR PI: PERIMETER OF A INSCRIBED POLYGON

I could present the next section in terms of Archimedes' original geometric proofs, but now that it is clear that Archimedes was actually dealing with trigonometry (very ahead of his time), we can walk through how Archimedes arrived at the $\frac{\text{Perimeter of a inscribed regular n-gon}}{\text{Diameter}}$, using our known trig identities.

First I claim that $\frac{\text{Perimeter of an inscribed regular n-gon}}{\text{Diameter}} = nsin(\frac{\pi}{n})$.

Proof. Let's say the inscribed regular n-gon has vertices $v_1, v_2, v_3 ... v_n$. Since it's regular, I can arbitrarily label consecutive vertices v_i and v_{i+1} traveling from edge to edge in a cycle, without losing generality (the last edge will be from v_n to v_1). The circle has center O. Each edge of the n-gon is a chord and they are all the same length. Therefore each $m\angle v_i O v_{i+1} = \frac{2\pi}{n}$ where $0 \leq i \leq n-1$, including $m\angle v_n O v_1 = \frac{2\pi}{n}$. Extend the segment $v_i O$ to become the diameter. Call the other endpoint D (there could already be a vertex there). From the inscribed angle



postulate we know that $\frac{m\angle v_i O v_{i+1}}{2} = m\angle v_i D v_{i+1} = \frac{\pi}{n}$. Due to $v_i D$ being a diameter, $\angle D v_{i+1} v_i$ is a right angle. Therefore the $sin(\frac{\pi}{n}) = \frac{v_i v_{i+1}}{v_i D} = \frac{\text{Length of a side of regular n-gon}}{\text{Diameter}}$.

$$nsin(\frac{\pi}{n}) = \frac{\text{Perimeter of a inscribed regular n-gon}}{\text{Diameter}} \tag{3.1}$$

□

For now let's call $nsin(\frac{\pi}{n}) = LB_n$.

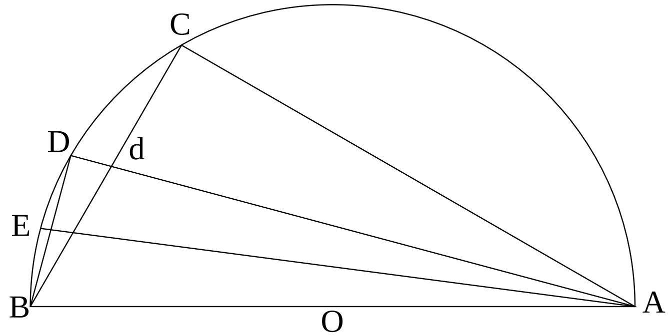


FIGURE 3.1. Inscribed Case

3.1. **What Archimedes Did**[2]**.** Suppose AB is the diameter, O is the center, C is on the circle, and $\angle CAB$ is a third of a right angle or $\frac{\pi}{6}$. This implies $\angle ACB$ is a right angle, and $\triangle ACD$ is a right triangle. Plus, $m\angle COB = 2m\angle CAB$.

Archimedes starts with an approximation for $\frac{AC}{CB}$ (the approximation is bigger than the actual value) and for $\frac{AB}{BC}$ (although this is not an approximation we will assume it is for the sake of induction). The first ratio is $cot(m\angle CAB)$. Second one is $csc(m\angle CAB)$. Archimedes proves that we can get $cot(\frac{m\angle CAB}{2})$ by adding together $cot(m\angle CAB)$ and $csc(m\angle CAB)$. But this is just our trig identity (2.4). Plus he proves we can get $csc(\frac{m\angle CAB}{2})$ by square rooting $cot^2(\frac{m\angle CAB}{2}) + 1$. Once again this is our trig identity (2.5).

Both $csc(m\angle CAB)$ and $csc(\frac{m\angle CAB}{2})$ can easily be made to look like $nsin(\frac{\pi}{n})$, since $m\angle CAB$ is already in the form $\frac{\pi}{n}$, and all we have to do is take the reciprocal of the cosecants, and multiply by n.

Again it makes sense to keep all approximations bigger than the actual value, so that when we take the reciprocal of cosecant, our approximate lower bound is slightly smaller than the actual lower bound.

4. CONVERGENCE!

4.1. **Nice Formulae.** To find the complexity of Archimedes's method, I need to figure out the rate of convergence. To do that, we are going to need a nice iterative formula. A lot of that has already been done during the trigonometric sections. It is safe enough to claim that at each step Archimedes does both equation (2.4) and (2.5):

$$csc(\frac{\pi}{n}) + cot(\frac{\pi}{n}) = cot(\frac{\pi}{2n}) \tag{4.1}$$

$$\sqrt{cot^2(\frac{\pi}{2n}) + 1} = csc(\frac{\pi}{2n}) \tag{4.2}$$

After the initial $csc(\frac{\pi}{n})$ and $cot(\frac{\pi}{n})$ are given, he always need to do (4.1) first, then (4.2). Doing both of these equations and acquiring the next set of trigonometric values can count as one iteration. But the problem with this set of nice formulae, is that we can't do much of error analysis since those ratios themselves are not the values bounding pi.

So I manipulate them, keeping in mind my $nsin(\frac{\pi}{n})$ and $ntan(\frac{\pi}{n})$, which I call LB_n and UB_n respectively. (4.1) first:

$$\begin{aligned} csc(\frac{\pi}{n}) + cot(\frac{\pi}{n}) &= cot(\frac{\pi}{2n}) \\ \frac{1}{sin(\frac{\pi}{n})} + \frac{1}{tan(\frac{\pi}{n})} &= \frac{1}{tan(\frac{\pi}{2n})} \\ \frac{1}{2nsin(\frac{\pi}{n})} + \frac{1}{2ntan(\frac{\pi}{n})} &= \frac{1}{2ntan(\frac{\pi}{2n})} \\ \frac{1}{2LB_n} + \frac{1}{2UB_n} &= \frac{1}{UB_{2n}} \\ \frac{UB_n + LB_n}{2LB_nUB_n} &= \frac{1}{UB_{2n}} \end{aligned}$$

$$UB_{2n} = \frac{2LB_nUB_n}{UB_n + LB_n} \tag{4.3}$$

[2]If you want to know Archimedes's geometric proof look at [5]

This is nice to have.

Next (4.2):

$$\sqrt{cot^2(\frac{\pi}{2n})+1} = csc(\frac{\pi}{2n})$$
$$\sqrt{\frac{1}{tan^2(\frac{\pi}{2n})}+1} = \frac{1}{sin(\frac{\pi}{2n})}$$
$$\sqrt{\frac{tan^2(\frac{\pi}{2n})+1}{tan^2(\frac{\pi}{2n})}} = \frac{1}{sin(\frac{\pi}{2n})}$$
$$\sqrt{\frac{tan^2(\frac{\pi}{2n})}{tan^2(\frac{\pi}{2n})+1}} = sin(\frac{\pi}{2n})$$
$$\sqrt{\frac{tan^2(\frac{\pi}{2n})}{sec^2(\frac{\pi}{2n})}} = sin(\frac{\pi}{2n})$$
$$\sqrt{tan^2(\frac{\pi}{2n})cos^2(\frac{\pi}{2n})} = sin(\frac{\pi}{2n})$$
$$\sqrt{tan(\frac{\pi}{2n})sin(\frac{\pi}{2n})cos(\frac{\pi}{2n})} = sin(\frac{\pi}{2n})$$
$$\sqrt{4n^2tan(\frac{\pi}{2n})\frac{sin(\frac{\pi}{n})}{2}} = 2nsin(\frac{\pi}{2n})$$
$$\sqrt{2ntan(\frac{\pi}{2n})nsin(\frac{\pi}{n})} = 2nsin(\frac{\pi}{2n})$$

$$\sqrt{UB_{2n}LB_n} = LB_{2n} \tag{4.4}$$

If I had to do this by hand, I would much prefer using the iterations of (4.1) and (4.2), because (4.1) is simple addition. (4.3) on the other hand requires multiplication, addition, and division. The advantage of (4.3) and (4.4) however is that I can use them to directly get me what $UB_{2n} - LB_{2n}$ is. And the number of digits UB_m and LB_m, where m is the number of sides in both of inscribed and the circumscribed regular polygons the compute-r (Archimedes, Al-Kashi, Liu-Hui, van Ceulen, to name a few) went up to, had in common is the accuracy of the respective compute-r's approximation of π. Or in other words the number of zeros in $UB_m - LB_m$ is the digits accuracy.

4.2. **Convergence Rate of Archimedes Method.** I started from guesses. For Archimedes had gotten 2 decimal digits of accuracy from a 96-gon, or $6*2^4$ sides. The Chinese mathematician, Liu-Hui, using very similar methods got 5 decimal digits of accuracy from a 3072-gon, or $6*2^9$ sides. Liu-Hui got 3 more digits by running Archimedes method 5 more times. In principle, Al-Kashi used Archimedes's method except with constant error-checking methods and got 14 decimal digits using $6*2^{27}$-gon. He ran the iteration 23 more steps and got 12 more decimal digits. Lastly the famous Ludolph van Ceulen who also used Archimedes method except with slightly different initial conditions got 35 decimal digits using a 2^{62}-gon[3]. The most reasonable guess seems to be that he started with a square, 4-gon, and did the iteration 60 times. So then with a roughly similar starting condition, van Ceulen calculated 56 more iterations than Archimedes and got 33 more decimal digits. The difference may be somewhat attributable to the accuracy of the square root approximations. In the end though for about every two iterations I get a digit of accuracy. I get the same pattern with my Pseudo-Archimedes code, which runs the iterations doing all the calculations at a 71 digits of arbitrary precision accuracy, without worrying about ceilings and floors (Table 1).

This suggests that at each step of the iteration $UB_{4n} - LB_{4n} \approx \frac{1}{10}(UB_n - LB_n)$ or $UB_{2n} - LB_{2n} \approx \frac{1}{\sqrt{10}}(UB_n - LB_n)$.

After a lot of fiddling and false starts (some of the equations looked so much like the arithmetic mean - look at the 4th step in the proof for 4.3 - and other ones looked so much like the geometric mean - equation 4.4 - but I knew it was definitely not quadratic convergence), I finally noted LB_{2n} to be always greater than LB_n. This is somewhat obvious geometrically as the inscribed regular n-gon gains more sides to look more like a circle. Plus, we know that any UB_n should always be bigger than any LB_m, considering the unstated but fundamental premise of Archimedes method, that the perimeter of the circumscribed regular n-gon should never be smaller than the circumference, and

[3] All this information was taken from [1, 2]

i	Digits accuracy of UB_{6*2^i}, LB_{6*2^i}
2	2
4	3
6	4
9	6
12	8
24	15
27	17
39	24
60	37

TABLE 1. Pseudo-Archimedes accuracy of $UB_{6*2^i} - LB_{6*2^i}$ versus i

the perimeter of the inscribed regular n-gon should never be bigger than the circumference (The second statement might be able to fail(?) if the polygon wasn't an inscribed regular n-gon, and was instead some concave squiggly thing). The first statement should be correct. The second one is right because the chords can never be longer than their respective arcs. So then the geometric mean $\sqrt{UB_{2n}LB_n}$ should always return a LB_{2n} which is bigger than LB_n.

Therefore:

$$\begin{aligned}
UB_{2n} - LB_{2n} \leq& UB_{2n} - LB_n \\
UB_{2n} - LB_n =& \frac{2LB_nUB_n}{UB_n + LB_n} - LB_n \\
=& \frac{2LB_nUB_n}{UB_n + LB_n} - \frac{LB_n(UB_n + LB_n)}{UB_n + LB_n} \\
=& \frac{LB_nUB_n - LB_n^2}{UB_n + LB_n} \\
=& \frac{LB_n(UB_n - LB_n)}{UB_n + LB_n} \\
\frac{UB_{2n} - LB_{2n}}{UB_n - LB_n} \leq& \frac{LB_n}{UB_n + LB_n}
\end{aligned}$$

Using the same reasoning that UB_n is greater than LB_n, $\frac{LB_n}{UB_n+LB_n} \leq \frac{1}{2}$.

$$\frac{UB_{2n} - LB_{2n}}{UB_n - LB_n} \leq \frac{1}{2} \tag{4.5}$$

So at every iteration our error is divided by 2. This is somewhat close to, and includes our preliminary guess as to what the rate of convergence might be ($\frac{1}{\sqrt{10}} < \frac{1}{2}$). We can express the convergence of Archimedes' method as:

$$\begin{aligned}
error_i = O(2^{-i}) \ & \text{where } i \text{ is the number of iterations} \\
& \text{and } error_i = UB_{a*2^i} - LB_{a*2^i} \\
& \text{where } a \text{ is some integer}
\end{aligned}$$

It is exponential convergence with rate $\frac{1}{2}$.

Note: looking at the experimental data however, it is suggestive that we potentially could make an even sharper convergence rate.

5. Guessing At Actual Complexities Of Archimedes's Method

Ludolph van Ceulen found π to 35 digits of accuracy. The Germans were so impressed that they frequently called π the Ludolphine number. So then approximately how many arbitrary precision calculations must have van Ceulen computed?

However there must be a disclaimer. Quite often, by doing things a little differently complexity can frequently be improved. Also most mathematicians (except Ludolph) did not take Archimedes's method to really high iterations, therefore their complexity time was very much reduced by the apriori knowledge of lower and upper bounds for certain square roots. And frequently they do not reveal how they found the numbers. And as all practiced compute-rs, they likely had little short cuts. Moreover, issues arise in actual time differences in computing certain arithmetic operations. For example, in general I would say multiplication is faster than division of the same N-digit numbers, but in arbitrary precision, both of their complexities are denoted by the same big-O notation. Same goes for addition and subtraction. So to project the computer's complexity to the human complexity that must have been required from these ancient mathematicians to calculate the digits of pi, is a speculation at best.

5.1. **Brute Force.** You can use equations (4.3) and (4.4) to just plug and chug your way, and do every calculation to the N-digit precision, where N is the number of decimal digits of accuracy you want. Since the $error_i = O(2^{-i})$ we need roughly $log_2 10$ iterations for each decimal digit. So we would need approximately $Nlog_2 10$ iterations. But each iteration takes multiplication of equally N-precise numbers, $O(N^2)$, and then addition ,$O(N)$, plus a division, $O(N^2)$, (4.3), AND another multiplication $O(N^2)$ and a square root (4.4). I shall assume that most mathematicians used some shape or form of the Babylonian method to compute square roots (computers today still use this method). Babylonian method is a special case of Newton's method ($f(x) = x^2 - y$), and quadratically convergent [3], so it would require $O(log_2 N)$ iterations. At each iteration of the Babylonian method, we require a division of two numbers with approximately $2N$ digits and N digits, $O(2N^2)$, addition, $O(N)$, and a multiplication by $\frac{1}{2}$ which is just $O(N)$ complexity [4]. Let's put that all together and we get:

$$Nlog_2 10\left(O(N^2) + O(N) + O(N^2) + O(N^2) + O(log_2 N)\big(O(2N^2) + O(N) + O(N)\big)\right)$$

For really large $N \to \infty$, complexity is around $O(N^3 log_2 N)$.

5.2. **Archimedes's Way Decimal Version.** Archimedes chose his fractions and his approximations to square roots without much explanation or reason, other than that they were pretty good lower or upper bounds. There was no established consistency that I could observe in the mathematicians' methods. A consistent way to do this would take the same route Archimedes did (equations 4.1 and 4.2) except stick to decimals and a certain amount of decimal digits precision at each step. The complexity of the decimal version of Archimedes's method ought to be somewhat similar to the complexity of the fractional version, or at least equal O(Complexity of fractional version). For example van Ceulen eventually turned to a decimal-esque system. His tombstone had $\frac{314159265358979323846264338327950288}{100000000000000000000000000000000000}$ and $\frac{314159265358979323846264338327950289}{100000000000000000000000000000000000}$ as a lower bound and upper bound, respectively, for π.

So suppose we want π to N-digit precision. Assume that at each step when we find the square root, we find it to N-digit accuracy. Use equations (4.1) and (4.2), we would need to do an addition, $O(N)$ (4.1), AND a multiplication, $O(N^2)$, an addition by 1 which we shall assume takes $O(1)$ complexity, and a square root (4.2). The square root, we again assume Babylonian Method, and just like before requires division, $O(2N^2)$, addition, $O(N)$, and a multiplication by $\frac{1}{2}$, $O(N)$. In all we would have to run (4.1) amd (4.2) approximately $Nlog_2 10$ times. Then at the very last iteration, we have to take the reciprocal and multiply by a constant, $O(N^2)$ So this comes out to be

$$Nlog_2 10\left(O(N) + O(1) + O(N^2) + O(log_2 N)\big(O(2N^2) + O(N) + O(N)\big)\right) + O(N^2)$$

For really large $N \to \infty$, complexity is around $O(N^3 log_2 N)$, same as the brute force way. You would have to do this twice, once for the perimeter of the circumscribed polygon and another for the perimeter of the inscribed polygon. If you plug in $N = 35$, the number of digits van Ceulen computed to, he must have done ≈ 3544200 calculations. And since he should have had to double check or triple check each one, it jumps to around 7 to 10 million calculations he must have done. Wow. I feel like I could potentially have done 7 to 10 million calculations in my entire lifetime. Moral of the story is anything is possible.

6. What More Could be Done?

Although I am happy with my current bound I could attempt to find a sharper bound for the convergence rate, because evidence suggests this is so. Also looking back at Table 1, we can see that the convergence does not get too much better even calculating everything in 71-digit precision. So the next attempt would be to find a reasonable optimization of complexity and convergence through experimentation (I have been rather unsuccessfully trying to theorize an optimization of the square roots approximations).

Codes

Pseudo-Archimedes:

```
sage: #Pseudo-Archimedes: I ignore all the ceilings and the floors that Archimedes had to deal with.
sage: #I wrote this to check my convergence rate.
sage: d=70  #How many digits you want
sage: import mpmath as mp
sage: mp.mp.dps=d+1 #I take this precaution since I am not thinking about ceilings and floors.
```

```
sage: def BoundPi(n,o): #n is number of Archimedes' iterations, o is the starting regular o-gon; Archimedes
...       a=mp.mpf(cot(pi/o))
...       b=mp.mpf(csc(pi/o))
...       for i in range (0,n):
...           a=a+b         #cot(k)+csc(k)=cot(k/2)
...           b=sqrt(a^2+1) #csc(k/2)=sqrt(cot^2(k/2)+1)
...       return [1/a*o*2**n,1/b*o*2**n]
...
...
sage: def DigitsAcc(n,o):
...       #Finds the digits in common between UB_o*2^n and LB_o*2^n
...       a=BoundPi(n,o)[0]-BoundPi(n,o)[1]
...       b=0
...       while a<=1:
...           a=a*10
...           b=b+1
...       return b
```

A Fractional Archimedes:

```
sage: #More Historical Archimedes: I try my best to keep to what he did, but hard to do
sage: #since I do not know the exact method through which he approximated square roots.
sage: #But it makes this one is more realistic, uses "fractions"
sage: d=80
sage: import mpmath as mp
sage: mp.mp.dps=d+1 #I do this so I can find correct ceilings and floors.
sage: def UBoundPi(n,acc):        # acc=the initial decimal digit accuracy desired of sqrt(3)
...       a=floor(sqrt(mp.mpf('3'))*10**(acc-1))  # For the Upper bound, we needed lower
...                                               #bound estimates for the square roots.
...       b=10^(acc-1)                            # The denominator of cot(30 deg)
...       c=2*10^(acc-1)                          # numerator of csc(30)
...       d=10^(acc-1)                            # Denominator of csc(30)
...       for i in range (0,n):
...           a=a+c                               #Just add numerators
...           c=floor(sqrt(a^2+b^2))              #Just take the integer part of square root
...       return b/a*6*2**n
...
...
sage: def LBoundPi(n,acc):        # acc=the initial decimal digit accuracy desired of sqrt(3)
...       a=ceil(sqrt(mp.mpf('3'))*10**(acc-1))  # For the Lower bound, we needed Upper
...                                               #bound estimates for the square roots.
...       b=10^(acc-1)                            # The denominator of cot(30 deg)
...       c=2*10^(acc-1)                          # numerator of csc(30)
...       d=10^(acc-1)                            # Denominator of csc(30)
...       for i in range (0,n):
...           a=a+c                               #Just add numerators
...           c=ceil(sqrt(a^2+b^2))            #Just raise it by 1
...       return d/c*6*2**n
...
...
sage: def HistDigitsAcc(n,acc):
...       #Finds the digits in common between UB_6*2^n and LB_6*2^n
...       #When they started with an initial accuracy of sqrt(3)
...       a=UBoundPi(n,acc)-LBoundPi(n,acc)
...       b=0
...       while a<=1:
...           a=a*10
...           b=b+1
```

```
...         return b
```

References

[1] Arndt, Jörg, and Christoph Haenel. Pi-unleashed. Berlin: Springer, 2001. Google Books. Google. Web. 6 May 2013.

[2] Gourdon, Xavier, and Pascal Sebah. "Pi and Its Computation through the Ages." Numbers, Constants and Computations. N.p., n.d. Web. 6 May 2013.

[3] Johnson, Gersham. "The Babylonian Method and Its Properties." QED Chicago's Youth Math Symposium. N.p., n.d. Web. 25 May 2013.

[4] Kosheleva, O., "Babylonian method of computing the square root: Justifications based on fuzzy techniques and on computational complexity," Fuzzy Information Processing Society, 2009. NAFIPS 2009. Annual Meeting of the North American , vol., no., pp.1,6, 14-17 June 2009

[5] Lindsey, Chuck. "Archimedes' Approximation of Pi." Florida Gulf Coast University. Florida Gulf Coast University, n.d. Web. 6 May 2013.