Corey Foote

Dr. Thomas Kent

Honors Number Theory

Date: 11-30-11

The Pollard's Rho Method for Factoring Numbers

We are all familiar with the concepts of prime and composite numbers. We also know that a number is either prime or a product of primes. The Fundamental Theorem of Arithmetic states that every integer $n \geq 2$ is either a prime or a product of primes, and the product is unique apart from the order in which the factors appear (Long, 55). The number 7, for example, is a prime number. It has only two factors, itself and 1. On the other hand 24 has a prime factorization of $2^3 \times 3$. Because its factors are not just 24 and 1, 24 is considered a composite number. The numbers 7 and 24 are easier to factor than larger numbers. We will look at the Sieve of Eratosthenes, an efficient factoring method for dealing with smaller numbers, followed by Pollard's rho, a method that allows us how to factor large numbers into their primes.

The Sieve of Eratosthenes allows us to find the prime numbers up to and including a particular number, n. First, we find the prime numbers that are less than or equal to $\sqrt{n}$. Then we use these primes to see which of the numbers $\sqrt{n} \leq n - k, ..., n - 2, n - 1 \leq n$ these primes properly divide. The remaining numbers are the prime numbers that are greater than$\sqrt{n}$ and less than or equal to n. This method works because these prime numbers clearly cannot have any prime factor less than or equal to$\sqrt{n}$, as the number would then be composite. Also, it cannot be the product of two numbers greater than or equal to$\sqrt{n}$, as the number would still be composite and greater than n (Nagel, 51-52).

We will now use the Sieve of Eratosthenes to find the prime numbers between 1 and 31. Since $5 \leq \sqrt{31} \leq 6$, we will cross out all of the numbers between 2 and 31 that are properly divisible by 2, 3, or 5. We do this because these numbers are composite, as each has factors other than itself and 1. For example, we cross the number 14 out because 2 divides it. Since 2 is a factor of 14, 14 is clearly composite. The remaining numbers are the prime numbers from 2 to 31 inclusive. They are 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, and 31. This method is effective for finding the prime numbers up to and including a number, *n*, that is relatively small. If we needed to find the primes from 2 to 10^6, we would have to do approximately 10^3 trial divisions using the Sieve of Eratosthenes. Since this method is not efficient for large numbers, we can use Pollard's rho to factor large numbers into their primes.

Before we study how Pollard's rho works on paper, we will first look at the pseudo code that can be implemented into a computer program, using a computer language such as C++. Once the pseudo code is correctly translated into the specific programming language, the program will output the prime factorization for a number (small or large) that we input. Look at Figure 1, which portrays the pseudo code for Pollard's rho. The numbers on the left represent the steps of the algorithm.

```
POLLARD-RHO($n$)
(1)  $i \leftarrow 1$
(2)  $x_1 \leftarrow$ RANDOM($0, n - 1$)
(3)  $y \leftarrow x_1$
(4)  $k \leftarrow 2$
(5)  while TRUE
(6)    do $i \leftarrow i + 1$
(7)         $x_i \leftarrow (x^2_{i-1} - 1) \bmod n$
(8)         $d \leftarrow \gcd(y - x_i, n)$
(9)         if $d \neq 1$ and $d \neq n$
(10)        then print $d$
(11)       if $i = k$
(12)      then $y \leftarrow x_i$
(13)               $k \leftarrow 2k$
```

Figure 1

We will demonstrate this algorithm step-by step by factoring 1909 into its primes. In step 1, we initialize *i* to be 1. Thus, we will start at $x_{1.}$ In step 2, we set x_1 to a random number between 0 and *n* -1. For the purposes of this paper, we will set $x_1 = 2$. For step 3, we store our x_1 as our y. Thus, we save 2 as our *y*-value. Line 5 starts our process of finding factors of a number *n*; in our case, 1909. Line 6 increments our value of *i* so that we produce x_1, x_2, $x_{3,}$ and so on infinitely.

We will now proceed to compute Pollard's rho by hand. We start with the formula $x_{i+1} = (x_i^2 - 1) \bmod n$, where *i* denotes the i^{th} *x*-term and *n* denotes the number to be factored. We already set $x_1 = 2$, so using the formula, we obtain $x_2 = (2^2 - 1) \bmod 1909 = 3 \bmod 1909 = 3$. Next, $x_3 = (3^2 - 1) \bmod 1909 = 8 \bmod 1909 = 8$. Similarly, $x_4 = 63$.

Following this pattern, $x_5 = 150$, $x_6 = 1500$, $x_7 = 1197$, $x_8 = 1058$, $x_9 = 689$, $x_{10} = 1288$, $x_{11} = 22$, $x_{12} = 483$, $x_{13} = 390$, and $x_{14} = 1288$. Notice that 1288 repeats itself. We shift our attention to Figure 2 below. Please note that the values denoted by *i* represent our x_i values. Notice the values of x_{10}, x_{11}, $x_{12,}$ and x_{13} repeat infinitely in a loop, thus forming a rho.

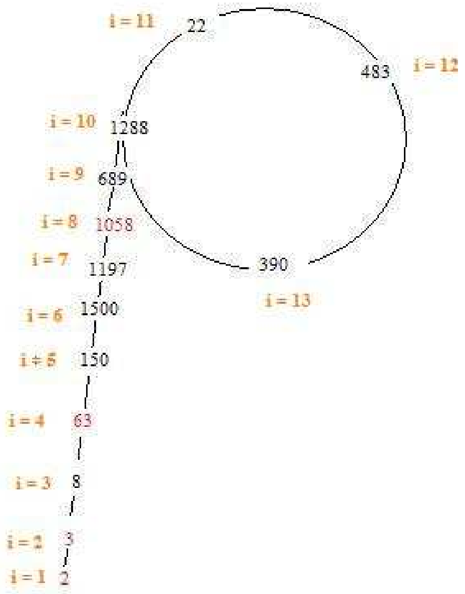

Figure 2

Since our x_i values represent the remainders, these values will have to repeat since there can only be 1909 possible remainders for the number 1909. This is because when dividing a number by 1909, the remainder is always between 0 and 1908. So, if we run the algorithm for 1910 steps, at least one of these x_i-values (our remainders) will have to appear more than once. We should find at most 1909 remainders. However, in practice, we generally repeat well before our remainders are used up. In this case, we repeated after having seen 14 different remainders.

The next step to finding the primes of 1909 using this factoring method is to find $d = (y - x_i, n)$, the greatest common divisor. Note that this is line 8 of the pseudo code. We need to find our y-values in order to find the greatest common divisor. To do this, we must first understand the pseudo code. Note that line 4 of the pseudo code initializes some k to 2. We jump to line 11, where there is an if-statement that is utilized when $i = k$. When this happens, the current x_i value is stored as a y-value, and the current k-value is multiplied by 2. In the case of our example, we store x_1 as a y-value initially. Then we look at x_2. Since $i = k = 2$, we utilize our if-statement. Our x_2-vlaue is stored as a y-value. Also, our k is doubled to 4. This y-value remains the same until $i = k$ again. So we store x_4 as our y-value. Similarly, we store x_8 and x_{16} as y-values. We can continue this infinitely, as long as our i-values are powers of 2. Our stored y-values are written in purple in Figure 2.

Now we can find the greatest common divisor. The first calculation that we perform is $(2 - 2, 1909) = (0, 1909) = 1909$. We use the Euclidian Algorithm to find the greatest common divisors; however, since showing the steps is a trivial matter, we will proceed without explanation. We change our y-values accordingly and proceed until we find the greatest common divisor. When we subtract our x_{10} from our current y-value, we find the greatest common divisor. Thus, $(1058 - 1288, 1909) = (-230, 1909) = 23$. Look at line 9 of the pseudo code. Since $d \neq 1$

and $d \neq n$, the program will print d (Cormen, 845-847). In our case, the program would print 23 as one of the factors of 1909. This factor also happens to be prime, which is what we are hoping for. We use this method of finding the greatest common divisor because it is known to work; it is a successful heuristic technique. Look at Figure 3 below, which shows the x_i-values, the y-values, the calculations for the greatest common divisor, and the greatest common divisor for the first twenty i-values.

Figure 3:

i	x_i	y	(y - x_i, 1909)	gcd
1	2	2	(2-2, 1909)	1909
2	3 = (2^2-1) mod 1909	3	(3-3, 1909)	1909
3	8 = (3^2-1) mod 1909	3	(3-8, 1909)	1
4	63 = (8^2-1) mod 1909	63	(63-63, 1909)	1909
5	150 = (63^2-1) mod 1909	63	(63-150, 1909)	1
6	1500= (150^2-1) mod 1909	63	(63-1500, 1909)	1
7	1197= (1500^2-1) mod 1909	63	(63-1197, 1909)	1
8	1058= (1197^2-1) mod 1909	1058	(1058-1058, 1909)	1909
9	689= (1058^2-1) mod 1909	1058	(1058-689, 1909)	1
10	1288= (689^2-1) mod 1909	1058	(1058-1288, 1909)	23
11	22= (1288^2-1) mod 1909	1058	(1058-1288, 1909)	1
12	483= (22^2-1) mod 1909	1058	(1058-483, 1909)	1
13	390= (483^2-1) mod 1909	1058	(1058-390, 1909)	1
14	1288= (390^2-1) mod 1909	1058	(1058-1288, 1909)	23
15	22= (1288^2-1) mod 1909	1058	(1058-22, 1909)	1

16	483= (22^{2}-1) mod 1909	483	(483-483, 1909)	1909
17	390= (483^{2}-1) mod 1909	483	(483-390, 1909)	1
18	1288= (390^{2}-1) mod 1909	483	(483-1288, 1909)	1
19	22= (1288^{2}-1) mod 1909	483	(483-22, 1909)	1
20	483= (22^{2}-1) mod 1909	483	(483-483, 1909)	1909

Thus, 1909 is composite. Pollard’s rho found 23 to be a factor of 1909. In order to factor the number further, we could run the algorithm on 23 and 1909/23 = 83.

Now we look at a larger number, 11347, to factor using Pollard’s rho. As in the previous example, we set $x_1 = 2$. We find $x_2 = 3$, $x_4 = 63$, $x_5 = 3968$, and $x_6 = 6734$. Pollard’s rho will find a prime factor after we reach x_6. We perform $d = (y - x_{i,} n)$ = (63 - 6734, 11347) = (-6671, 11347) = 7. A second implementation of Pollard’s rho on 7 and 11347/7 shows that our factors, which are prime, of 11347 are 7 and 1621.

Now we will study how long it takes to repeat remainders; the repetition could occur during early iterations or much later on. When looking at 1909, our numbers started repeating at x_{14}. Note that it could take as many repetitions as there are remainders for a number to repeat. When looking at 11347, it could potentially take until x_{11347} to start repeating the cycle and forming the rho. We will look at Figure 4 to see the first fifty remainders of 11347 using Pollard’s rho. Notice how not one of these numbers repeats itself. If we were to continue trying to find where the remainders repeat, we could find the repetition as soon as x_{51} or as late as x_{11347}, so we will just stop here.

Figure 4:

$x_1 = 2$	$x_2 = 3$	$x_3 = 8$	$x_4 = 63$	$x_5 = 3968$
$x_6 = 6734$	$x_7 = 4143$	$x_8 = 7784$	$x_9 = 9022$	$x_{10} = 4452$
$x_{11} = 8441$	$x_{12} = 2667$	$x_{13} = 9666$	$x_{14} = 357$	$x_{15} = 2631$
$x_{16} = 490$	$x_{17} = 1812$	$x_{18} = 4060$	$x_{19} = 7755$	$x_{20} = 924$
$x_{21} = 2750$	$x_{22} = 5397$	$x_{23} = 11206$	$x_{24} = 8533$	$x_{25} = 9736$
$x_{26} = 8204$	$x_{27} = 6558$	$x_{28} = 2233$	$x_{29} = 4955$	$x_{30} = 8463$
$x_{31} = 104$	$x_{32} = 10815$	$x_{33} = 10695$	$x_{34} = 5264$	$x_{35} = 321$
$x_{36} = 917$	$x_{37} = 1210$	$x_{38} = 336$	$x_{39} = 10772$	$x_{40} = 1561$
$x_{41} = 8462$	$x_{42} = 5873$	$x_{43} = 8595$	$x_{44} = 5054$	$x_{45} = 818$
$x_{46} = 10997$	$x_{47} = 9029$	$x_{48} = 5992$	$x_{49} = 2155$	$x_{50} = 3101$

We will factor one more number using Pollard's rho: 1695. This is slightly different than the other numbers, but we start the same way. Figure 5 below demonstrates our process.

Figure 5:

i	x_i	y	$(y - x_i, 1695)$	gcd
1	2	2	(2-2, 1695)	1695
2	$3= (2^2 – 1) \bmod 1695$	3	(3-3, 1695)	1695
3	$8= (3^2 – 1) \bmod 1695$	3	(3-8, 1695)	1
4	$63= (8^2 – 1) \bmod 1695$	63	(63-63, 1695)	1695
5	$578= (63^2 – 1) \bmod 1695$	63	(63-578, 1695)	5

Pollard’s rho finds a factor of 5. So, we run Pollard’s rho on 5 and 339, the quotient of 1695/5. Figure 6 shows us this process on the number 339.

Figure 6:

i	x_i	y	(y - x_i, 339)	gcd
1	2	2	(2-2, 339)	339
2	3=(2^2-1) mod 339	3	(3-3, 339)	339
3	8=(3^2-1) mod 339	3	(3-8, 339)	1
4	63=(8^2-1) mod 339	63	(63-63, 339)	339
5	239=(63^2-1) mod 339	63	(63-239, 339)	1
6	168=(239^2-1) mod 339	63	(63-168, 339)	3

We proceed to run Pollard’s rho on 3 and 339/3 = 113, and Pollard’s rho finds 3 and 13 as prime factors of 1695. Thus, the prime factors of 1695 are 3, 5, and 113.

Pollard’s rho will not always work. For example, it will not factor 12 into its primes; subsequently, the program will say that 12 is a prime number. Look at the following table that represents the process of Pollard’s rho on 12.

Figure 7:

i	x_i	y	(y - x_i, 12)	gcd
1	2	2	(2-2, 12)	12
2	3 = (2^2 – 1) mod 12	3	(3-3, 12)	12
3	8 = (3^2 – 1) mod 12	3	(3-8, 12)	1
4	3 = (8^2 – 1) mod 12	3	(3-3, 12)	12

5	8 = (3^2 – 1) mod 12	3	(3-8, 12)	1
6	3 = (8^2 – 1) mod 12	3	(3-3, 12)	12
7	8 = (3^2 – 1) mod 12	3	(3-8, 12)	1
8	3 = (8^2 – 1) mod 12	3	(3-3, 12)	12

Our program enters an infinite loop at x_4, and it repeats 3 and 8 infinitely. We normally can be sure it is prime because for our y - x_i, we are only going to subtract 3-3 and 3-8 infinitely, which are relatively prime to 12. Since our greatest common divisors are 12 and 1 respectively, Pollard's rho will assume 12 is prime, even though it is not. For example, if we wanted to factor 24, Pollard's rho will say that the prime factors of 24 are 12 and 2. Thus, Pollard's rho is not completely reliable. Similarly, Pollard's rho will find 4 and 6 to be prime, and possibly other numbers.

Now we will perform analysis of Pollard's rho. First, we will look to see just how many iterations it takes to actually find a prime factor. If we are "lucky," we should find our prime factor by around $\sqrt[4]{n}$ iterations (Cormen 845). We will refer to two of our previous examples. Note that $\sqrt[4]{1909} \approx 6.6$, so we could have been as lucky as to find our prime divisor around six or seven iterations. Unfortunately, it took us until our tenth iteration to find our prime divisor. For our example 11347, we expected to find our prime divisor around ten or eleven iterations, but we were extremely lucky and found it at the sixth iteration.

A probabilistic analysis explains why we can expect to find the factor so quickly. A famous example of probabilistic analysis is the "birthday paradox," which states that if there are 23 people in one room, there is a 50% chance that two of those people have the same birthday. We start this calculation with just one person in the room. The probability that this person will not have the same birthday as anyone else in the room is 365/365 = 100%. (Clearly this has to be the case if there is only one person present.) We will call this event P(A). Next, we find the probability that two people in the room do not have the same birthday. Note that these events are independent of one another. We find the probability that the second person in the room does not have the same birthday as anyone else in the room. This probability is 364/365, and we will denote this by P(B). We make this probability 364/365 because there is a 1/365 chance that this second person will have the same birthday as the first. When we multiply P(A) and P(B) to get (365/365)(364/365), we find there is approximately a 99.73% chance that two people in the room will not have the same birthday.

We continue this process in similar fashion. For three people in the room, when we multiply (365/365)(364/365)(363/365), we find there is approximately a 99.18% chance that three people in the room will not have the same birthday. We continue the calculation up until we have 23 people. We multiply (365/365)(364/365)(363/365)(362/365)…(343/365)(342/365), which approximately equals a 50.0% chance that two people in the room will not have the same birthday. Of course, we could rephrase it to say there is approximately a 50% chance that two people in the room will have the same birthday when there are 23 people present. Note that 23 ≈ $\sqrt{365}$ ≈ 19. Of course, this probability will increase as more people are added to the room.

A similar probabilistic analysis is used for determining how likely it is that our remainders will repeat. Note that this procedure is the same as the birthday paradox. We will look at our example of 1909 to illustrate this. We look for the probability that choosing one number out of 1909 possible remainders will be different from the rest of the remainders. This probability is (1909/1909) = 1. We proceed to find the probability that choosing two numbers out of 1909 possible remainders will be different from each other. When we multiply

(1909/1909)(1908/1909), we find there is approximately a 99.95% chance that choosing two numbers out of 1909 possible remainders will be different from one another. We continue this pattern. When we multiply (1909/1909)(1908/1909)(1907/1909)(1906/1909)…(1895/1909), we find there is approximately a 94.64 % chance that choosing two remainders out of 1909 possible remainders will be different from each other after choosing fifteen numbers. Clearly, this probability will decrease as we choose more numbers. Note that for our example of 1909, we found our repeating remainders at our fourteenth iteration. We definitely "beat the odds" of having our remainders repeat so early in the process

It is here that we note the significance of$\sqrt{n}$. It can be shown that a repeat in remainders should happen with a 50% probability around $\sqrt{n}$ iterations. To find a 50% chance that there are repeating remainders of the number 1909, one must do fifty-two iterations. This is approximately equal to $\sqrt{1909} \approx 43$.

All of this shows that we expect to find a divisor of *n* around $\sqrt[4]{n}$ iterations. Since we expect our remainders to start repeating around$\sqrt{n}$, we can expect to find our divisor around$\sqrt[4]{n}$. The reason we expect this is because when we try to find $d = (y - x_i, n)$, we choose our *y*-values in such a way (x_i-values where the *i*-values are powers of 2) so that the x_i-values become exponentially larger than the *y*-values. This increases our chances of efficiently finding $d = (y - x_i, n)$, and thus leading to a divisor, hopefully one that is prime, of a number *n*.

Pollard's rho is a method we use for factoring large numbers into their primes. We can do this method by hand or via a computer program. It can be very efficient if we are "lucky" or very inefficient if we are "unlucky." It does not always work, as numbers such as 4, 6, and 12 cannot be effectively factored into their primes. We can test our efficiency of Pollard's Rho by using probabilistic analysis. Although not the most efficient of factoring methods, it is more efficient than the very tedious Sieve of Eratosthenes.

Works Cited

Cormen, Thomas H., Charles E. Leiserson, and Ronald L. Rivest. *Introduction to Algorithms*. 23rd ed. The Massachusetts Institute of Technology: McGraw-Hill Book Company, 1999. Print.

Long, Calvin T. *Elementary Introduction to Number Theory*. 3rd ed. Long Grove: Waveland Press, Inc., 1995. Print.

Nagell, Trygve. *Introduction to Number Theory*. New York: Chelsea Publishing Company, 1964. Print.