Rose-Hulman Undergraduate Mathematics Journal

Volume 10
Issue 2

Article 7

Statistical Investigation of Structure in the Discrete Logarithm

Andrew Hoffman
Wabash College, hoffmaan@wabash.edu

Statistical Investigation of Structure in the Discrete Logarithm

Andrew Hoffman
Wabash College
hoffmaan@wabash.edu

Abstract

The absence of an efficient algorithm to solve the Discrete Logarithm Problem is often exploited in cryptography. While exponentiation with a modulus, $b^x \equiv a \pmod{m}$, is extremely fast with a modern computer, the inverse is decidedly not. At the present time, the best algorithms assume that the inverse mapping is completely random. Yet there is at least some structure, such as the fact that $b^1 \equiv b \pmod{m}$. To uncover additional structure that may be useful in constructing or refining algorithms, statistical methods are employed to compare mappings, $x \mapsto b^x \pmod{m}$, to random mappings. More concretely, structure will be defined by representing the mappings as functional graphs and using parameters from graph theory such as cycle length. Since the literature for random permutations is more extensive than other types of functional graphs, only permutations produced from the experimental mappings are considered.

Introduction

The Discrete Logarithm Problem (DLP) is the analog of the canonical logarithm problem, finding $x = \log_b(y)$, in a finite cyclic group. For instance, when considering the integers under normal multiplication with a modulus the DLP becomes, "For which power(s) x is $b^x \equiv y \pmod{m}$?" While the problem may be posed in other groups, this paper will focus on the preceding example, a prevalent instance. More specifically, this paper will limit itself to prime moduli since this type of DLP with composite moduli reduces to solving the prime case after factoring. One may be tempted to consider the problem trivial since there are only finitely many possible answers. However the lack of any algorithm significantly more efficient than a brute force one makes the DLP a topic of much interest. Another reason the DLP is so studied is that the inverse operation is extremely efficient. Techniques such as successive squaring make modular exponentiation with large numbers feasible by hand calculation and trivial with computers.

The difficulty of the DLP coupled with its inverse's relative ease makes it particularly well-suited to cryptography. Cryptography is the art of transferring secure information. If a cryptographic system's method to encrypt and decrypt information takes too long, then the system will not be useful as information's usefulness may expire. Yet if there is a quick way to break the system and get the key, then the information is not secure. Cryptographic systems such as Elgamal [8, pages 476-478] rely on the ease of modular exponentiation for encryption and decryption and the difficulty of the DLP to secure the key.

The DLP's applications in cryptography create an interest in algorithms to solve it. There are algorithms, such as Pollard's Rho method given in [7], which moderately improve on brute forth methods. Yet all current algorithms work under the assumption that modular exponentiation behaves randomly and do not exploit any subtle structure in the mapping $x \mapsto b^x \pmod{p}$. There is of course some structure, such as the fact that $b^{p-1} \mapsto 1 \pmod{p}$ from Fermat's Little Theorem. To uncover additional, potentially exploitable structure, this paper will seek to quantify structure using ideas from graph theory, use combinatorial techniques to find the expected properties of random graphs, implement a computer program to collect experimental data, and finally employ statistical methods to check for significant differences in the observed versus the expected graph structure.

Quantifying Structure

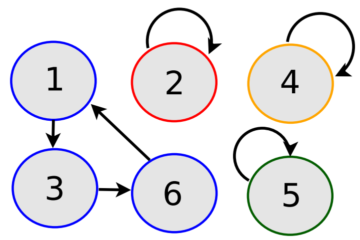
A first step to finding structure in the DLP is to view it as a function. As stated previously, the DLP asks for the inverse of $x \mapsto b^x \pmod{p}$. Therefore if we represent the forward mapping as a functional graph, finding structure in the graph may lead to exploitable structure for the DLP. The creation of functional graphs is clear-cut. One simply represents the x values as nodes and draws arrows for each of the mappings. For instance, suppose that one is considering $x \mapsto 3^x \pmod{7}$. First the various powers are calculated:

$3^1 = 3 \equiv 3 \pmod{7}$	$3^4 = 81 \equiv 4 \pmod{7}$
$3^2 = 9 \equiv 2 \pmod{7}$	$3^5 = 243 \equiv 5 \pmod{7}$
$3^3 = 27 \equiv 6 \pmod{7}$	$3^6 = 729 \equiv 1 \pmod{7}$

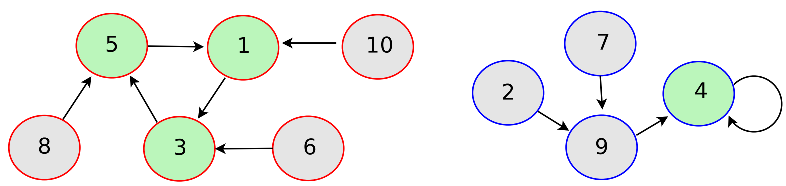
See that the essential information is the exponent and the resulting equivalence:

$1 \mapsto 3$	$4 \mapsto 4$
$2 \mapsto 2$	$5 \mapsto 5$
$3 \mapsto 6$	$6 \mapsto 1$

To obtain a functional graph one draws an arrow from 1 to 3, 3 to 6, 6 to 1, 2 to itself, etc., as follows:



If one uses 3 as the base and 11 as the modulus the following graph is produced:

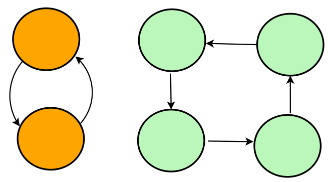


The above shows, among other things, that $3^1 = 3, 3^3 \equiv 5 \pmod{11}$, and $3^2 = 9$. Inversely, it shows the solution to a DLP such as, "3 to which power(s) equals $4 \pmod{11}$?" (The answers are 9 and 4.) The type of functional graph shown above will be called a binary functional graph since every node has either 0 or 2 nodes which map to it. Similarly there are ternary graphs where each node is mapped to by 0 or 3 others, quaternary, and more generally m-ary graphs. The first example of a functional graph will be referred to as a permutation graph since it represented a permutation, but equivalently it is a unary graph. The following theorem by Dan Cloutier in [4] describes the interaction between the base and the resulting graph:

Theorem 1. *If r is any primitive root modulo p and $g \equiv r^a \pmod{p}$, then the values of g that produce an m-ary graph are precisely those for which $\gcd(a, p-1) = m$.*

This paper will limit itself to permutations, which by the preceding theorem implies all bases are primitive roots. By limiting the investigation to only permutations, the extensive literature concerning random permutations may be exploited. Whereas the structure of random ternary graphs, for example, has not been studied extensively, random permutations have been of interest to mathematicians for decades. Therefore, since there is greater understanding of the expected structure, *viz.*, random permutations, one may more completely determine whether graphs produced from the DLP possess any dissimilar structure.

One byproduct of considering permutations is that every node is part of a cycle. If a node were not part of a cycle, then it would not be mapped to, which would violate the definition of a permutation. Since everything is in cycles, structure will solely be defined in terms of cycles. Specifically, there are three parameters that I will consider: number of cycles, maximum cycle length, and weighted average cycle length. The following generic graph will be used to illustrate their meanings.



The number of cycles equals 2 because there is a cycle on the left containing 2 nodes and another on the right containing 4. The maximum cycle length is 4 because the greatest number of nodes in a cycle is 4, present on the right. Weighted average cycle length requires a more thorough explanation because it is calculated from a node's perspective. From the graph's perspective, one cycle has length 4, the other length 2, so the average is $\frac{4+2}{2} = 3$. Yet from the node's perspective, 2 see a length of 2, and 4 see a length of 4. Therefore the weighted average would be $\frac{4\cdot4+2\cdot2}{6} = \frac{20}{6} \approx 3.3$. In contrast, six nodes could be arranged in two cycles of length three. In this case, the unweighted average would again be three, as would the weighted average since $\frac{3\cdot3+3\cdot3}{6} = \frac{18}{6} = 3$. This shows that the weighted cycle average reveals structure beyond the number of cycles. Knowing this structure would be useful in applications such as pseudorandom number generators because it determines the expected number of iterations which may be performed on a node before repetition occurs.

A brief elucidation on the mentioning of these parameters is useful at this time. Each prime modulus produces a permutation graph when the base is a primitive root. The parameter data is collected for each permutation, and then the averages and variances are computed across the graphs. These averages and variances then are associated with the prime. Note that the final parameter was an average, so in association with the prime there is the average of an average. This paper will attempt to make clear when the mean for the weighted average cycle length is being considered as opposed to the variance for the weighted average cycle length.

With structure now defined in terms of functional graphs and the three cycle-based parameters, comparisons are possible between random permutations and those constructed from the solution to the DLP. The comparison assumes there are known expected values for the random case and experimental values for the DLP case.

Expected Values

The process of finding theoretical values involves using marked generating functions and methods similar to those employed by Lindle in [6]. The generating function for putting objects into cycles is

$$c(z) = \ln \frac{1}{1-z}. \tag{1}$$

The process of turning these cycles into permutations is given by

$$f(z) = e^{c(z)} = \frac{1}{1-z}. \tag{2}$$

Therefore, to count the number of expected number of cycles in a permutation, we mark the function $c(z)$ with a u in $f(z)$, differentiate with respect to u, and then evaluate with $u = 1$ as follows:

$$\frac{d}{du}\left(e^{u \cdot \ln \frac{1}{1-z}}\right)\Big|_{u=1} = \ln\left(\frac{1}{1-z}\right) e^{u \cdot \ln \frac{1}{1-z}}\Big|_{u=1} = \ln\left(\frac{1}{1-z}\right)\frac{1}{1-z}. \tag{3}$$

Note, that since this an exponential generating function, there should be a multiplication by $n!$. However, since we are taking the mean over $n!$ permutations, the terms cancel. As Lindle describes in [6], this generating function can be turned into a differential equation, then into a recursive formula, and finally into an explicit formula. A generating function package for *Maple* simplifies the process greatly. For number of cycles the transformation is

$$f(z) \cdot (z-1) - 1 + (z^2 - 2z + 1) \cdot \left(\frac{d}{dz} f(z)\right), \ f(0) = 0 \tag{4}$$

$$\Rightarrow (-n-1) \cdot a(n) + (n+1) \cdot a(n+1) - 1, \ a(0) = 0, \ a(1) = 1 \tag{5}$$

$$\Rightarrow a(n) = \Psi(n+1) + \gamma \tag{6}$$

where $\Psi(x) = \frac{d}{dx} \ln(\Gamma(x))$, $\Gamma(x) = \int_0^\infty e^{-t} t^{x-1} dt$, and γ is Euler's constant. Therefore the explicit formula for the expected number of cycles in a random permutation on n objects is $\Psi(n+1) + \gamma$. Similar methods to the above show the formula for the expected weighted average cycle length for a permutation of size n to be $\frac{n+1}{2}$. The only methodological difference is a final division by n since the parameter is seen from the node and there are n nodes. For expected maximum cycle length, a marked generating function is not used. Instead I defer to the formula found by Shepp and Lloyd in [9] which gives it to be

$$n \int_0^\infty \left[1 - \exp\left(-\int_v^\infty e^{-u} \frac{du}{u}\right)\right] dv. \tag{7}$$

In addition to the expected means seen above, expected variances will be applicable with Lindle's updated code, whose output includes observed variances. First, the formula for variance must be examined. Variance is a set's deviance from the mean, summed for each piece of data:

$$\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2 = \frac{1}{N}\left(\sum_{i=1}^{N} x_i^2\right) - \bar{x}^2 \tag{8}$$

where N equals the number of data points, x_i each individual point, $\bar{x}$ the mean and the right side represents an algebraic simplification. The means are described above, so what remains to be found is a formula for $\frac{1}{N}\left(\sum_{i=1}^{N} x_i^2\right)$. The generating functions from before are used, but with a new marking method. The function is marked with u and differentiated twice to account for the squaring.

For number of cycles, the generating function for the summation of the data points squared looks like

$$v(z) = \frac{d}{du}\left(u\left(\frac{\partial}{\partial u}e^{u\cdot\ln\frac{1}{1-z}}\Big|_{u=1}\right)\right)\Big|_{u=1} = \frac{\ln\left(\frac{1}{1-z}\right)}{1-z} + \frac{\ln\left(\frac{1}{1-z}\right)^2}{1-z}. \qquad (9)$$

Since this an exponential generating function, there should be a multiplication by $n!$, but the $\frac{1}{N}$ term nullifies this. Using the methods described above, this was turned into an explicit formula:

$$\frac{1}{N}\left(\sum_{i=1}^{N} x_i^2\right) = \Psi(n+1) + \gamma + (\Psi(n+1)+\gamma)^2 + \Psi'(n+1) - \frac{\pi^2}{6}. \qquad (10)$$

Combining this with the mean squared derived from (6), and with a little simplification, the final formula for expected variance in number of cycles in a permutation of size n is:

$$\Psi(n+1) + \gamma + \Psi'(n+1) - \frac{\pi^2}{6}. \qquad (11)$$

Using similar methods, the expected variance for weighted average cycle length is

$$\frac{n^2-1}{12}. \qquad (12)$$

Since I did not use a marked generating function to obtain the expected mean for maximum cycle length, the preceding method for variance is not applicable. Therefore, the expected variance for maximum cycle length remains unknown for this paper's analysis.

The final theoretical value of interest is related to the cycle distribution. Knowing the cycle distribution for a given cycle length k would mean knowing how many permutations from a fixed modulus should produce 0 cycles of length k, 1 cycle of length k, 2, etc. The distribution of cycle lengths turns out be a Poisson distribution. Arratia and Tavaré in [1] give the following theorem:

Theorem 2. *For $i = 1, 2, ...$, let $C_i(n)$ denote the number of cycles of length i in a random n-permutation. The process of cycle counts converges in distribution to a Poisson process on $\mathbb{N}$ with intensity i^{-1}. That is, as $n \to \infty$,*

$$(C_1(n), C_2(n), ...) \to (Z_1, Z_2, ...),$$

where the Z_i, $i = 1, 2, ...$, are Poisson-distributed random variables with

$$\mathbb{E}(Z_i) = \frac{1}{i}.$$

Therefore the theoretical number of permutations containing k cycles of length j is known.

Observed Data

The first step in obtaining data was the implementation of a computer program designed for this very task. Dan Cloutier wrote code in C++ that calculated various graph theory parameters, including the ones of interest to this paper, for a set of m-ary graphs produced by a given prime modulus. Nathan Lindle revised the code in C, enabling it to calculate experimental variances as well as means. To calculate variances however, the number of graphs produced by a given prime is needed. For this task Lindle relied on external calculations. The first modification I made to the code was to integrate this calculation to make variance statistics more readily accessible. The second major modification I made was to have the code output a limited cycle distribution. This meant having the code output the number of cycles of lengths 1, 2, 3, 5, 7, 10, and 20 for each permutation created with the modulus. Therefore, the code for me worked as follows: I entered a prime number and my desired graph type (permutation), it created all of the necessary graphs, it calculated the means and variances for the three parameters of interest over the set of all permutations, and finally it broke down the number of cycles of fixed lengths as described previously.

The next step in obtaining data was to choose which primes to run the code on. I focused on primes valued around 100,000 to balance run time with having enough permutations produced for accurate results. The other consideration I took was to have three levels of primes based on their $p-1$ factorizations. Each level contained 10 primes. The first level was primes where $p-1$ had 2 factors, the second 6 factors, and the third > 9 factors. Having these levels enabled me to run ANOVA tests to check whether the $p-1$ factorizations significantly affected the parameters of interest. Looking at the $p-1$ factorization was motivated in part by Theorem 1. Theorem 1 shows that the divisors of $p-1$ play a role in the type of graph produced and the number of factors p has has a large effect on its set of divisors. Therefore, it is conceivable that the number of factors could have an effect on the parameters studied here.

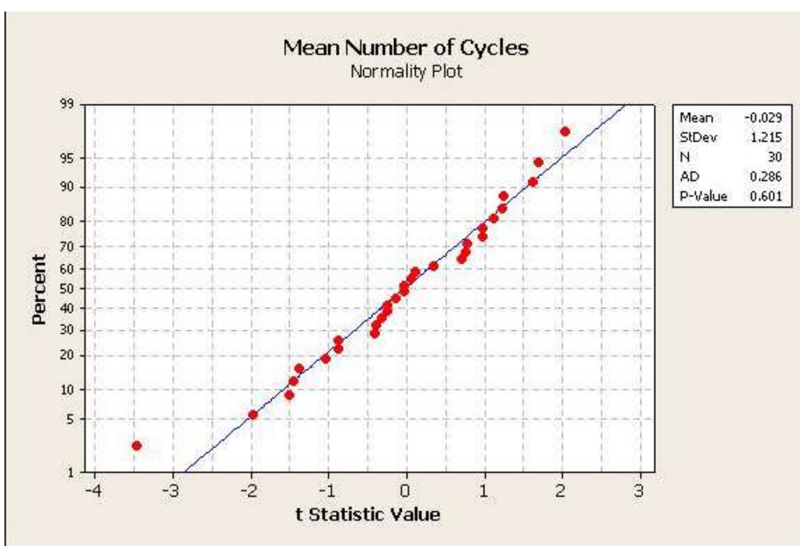
If it turned out that the factorization did affect the parameters, then a segmented approach could have been followed. This would prevent significant results in one of the levels being masked by insignificant ones in others. However, the ANOVA tests found that the variances between the levels were likely random as opposed to systematic. The following gives the probabilities that the variance between the levels for each parameter was simply due to random variation:

Parameter	p-value
Mean Number of Components	.880
Number of Components Variance	.498
Mean Max Cycle Length	.542
Max Cycle Variance	.110
Mean Avg Cycle Length	.616
Avg Cycle Variance	.191

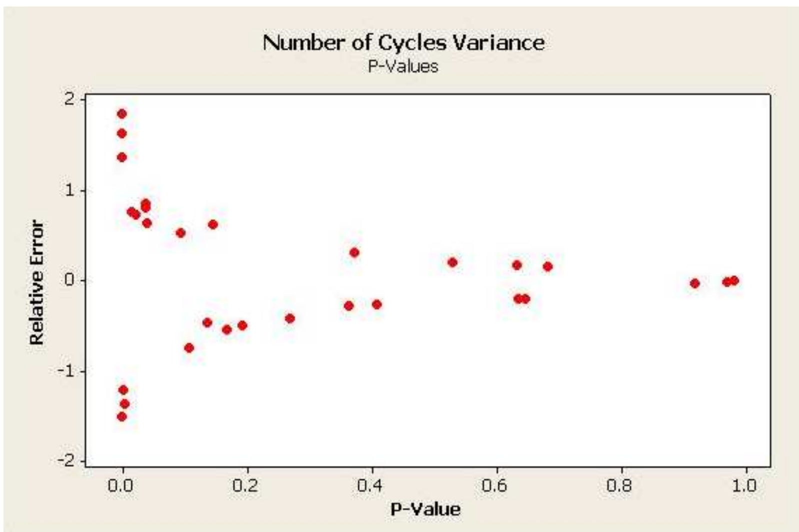
It should be noted that the p-values were above the significance threshold of $\alpha = .05$ only when the data was corrected for the size of the prime. For instance, recall that the formula for the expected number of cycles for a graph of size n is $\Psi(n+1)+\gamma$. The function $\Psi(n+1)$ can be defined as $H_n - \gamma$ where H_n is the nth harmonic number. The harmonic numbers grow at a rate of $\ln(n)$. Therefore, a division by $\ln(p)$ to account for prime size was necessary in the number of cycles parameter. See Appendix E for the correction factors and graphical representations of the variance between the levels. Since the tests showed the factorizations insignificant as far as the parameters are concerned, I could confidently group my primes into one sample of size 30.

Statistical Results

With observed and expected values found, statistical tests may be conducted to find significant differences. First the number of cycles are considered. Complete data can be found in Appendix A. To compare the means, t-tests are employed. A t-test will return the probability that an observed sample mean is different due to random variation from a theoretical population mean, given the number of samples used to obtain the observed mean. If this probability is low, then likely there is a significant, systematic difference between the sample and theoretical value. For the purpose of this paper, low probability will be a p-value $< .05$. For the 30 t-tests conducted on average number of cycles, 3 returned significant p-values. However, it is expected that around 5% of the normally distributed t-statistics should be falsely significant. Therefore an Anderson-Darling Test is used to determine whether the t-statistics are following a normal distribution. This test found that there is no evidence to conclude the distribution is not normal. This implies there is no significant deviance from the expected values in the statistic when the 30 primes are considered as a whole.



Using *Minitab*, expected variances for average number of cycles were compared to the expected average number of cycles. For this statistic, there were 11 significant p-values. This is considerably more than the 1 or 2 false positives one would expect with $\alpha = .05$. Using *Dataplot*, an Anderson-Darling test compared the p-values to a uniform distribution and concluded with a test statistic of 58.58 that the p-values are not uniformly distributed. While this is evidence that the variance in the DLP case differs significantly from the random case, the relative errors in the tests which produced significant p-values were sometimes positive and sometimes negative.



Based on these results, unless a predictor could be found for the sign of the relative error, exploiting the structure seems difficult. One potential predictor, the factorization of $p-1$, has shown preliminary promise. However, the results are not conclusive and a more extensive search for a predictor is likely necessary.

Second we consider maximum cycle length. The complete table is Appendix B. The t-tests for maximum cycle length returned no significant p-values. This is means that it is extremely likely that the DLP cases mirrors the random case as far as average maximum cycle length is concerned. The variances between the cases were not compared because the theoretical value for variance was not found.

Next, the weighted average cycle length is considered. See Appendix C for the entire data set. Again, t-tests were used to compare the observed and expected means. For this statistic, there were no significant p-values. The variance however returned the most significant results of the investigation. Whereas the other parameters varied roughly 1% or 2% between the random and the DLP case, the variance in the weighted average cycle length differed by an order of magnitude. The average relative error was 50.03%. There is no need for p-values since a difference of this size given the large sampling means that it is essentially impossible for the discrepancy to be random variation. Note also that the difference was consistent among the primes, so it appears that the mapping $x \mapsto b^x \pmod p$ does impose some systematic structure which affects this parameter.

Finally, we consider the limited cycle distribution of certain cycle lengths. Recall that the code recorded frequencies for cycles of lengths 1, 2, 3, 5, 7, 10, and 20. These were recorded for each of the 30 primes which means there are 210 comparisons to do. To accomplish the comparisons I used χ^2-tests. Some results were amazingly accurate. For instance, the two cycle distribution for the 23,328 permutations created using 102,061 as the modulus is:

No of 2-cyc	Obs	Exp
0	14139	14149
1	7082	7075
2	1772	1769
3	293	295
> 3	42	41

Yet there were 23 significant results. The table in Appendix D summarizes the p-values obtained. One would only expect roughly 11 falsely significant results. The significant results were split evenly between the different $p-1$ factorization levels, so again it appears the factorization did not have an effect on the parameter.

Nevertheless, the results were not split evenly among the various cycle lengths considered. In fact, over half of the significant results happened in the 1-cycle case. Anderson-Darling Tests were conducted to determine if the distributions of the p-values were uniform. The following table summarizes the results:

Cycle Length	Test Statistic	Reject Uniformity?
1	31.74	Yes
2	3.28	Yes
3	1.35	No
5	1.10	No
7	0.93	No
10	0.57	No
20	1.08	No

From this we can see that the DLP graphs differ significantly in the 1- and 2-cycle cases compared to random graphs. It should be noted that not all cycle lengths were studied identically. For instance, the number of expected 10 cycles drops off much more rapidly than expected 2 cycles. For the higher cycle lengths, there were only a few categories, *viz.*, permutations were grouped based on having 0, 1, or > 1 cycles of length k for large k. For the smaller cycles, a much wider range of frequencies were considered. There exists the possibility that the varying degrees of freedom for the χ^2 tests of different cycle lengths contributed to the distribution of the p-values, including the lack of uniformity only in small cycle sizes.

Conclusions

Based on the statistical tests conducted for this paper, one may conclude that in many ways the mapping $x \mapsto b^x \pmod{p}$ does behave like a random mapping. However there also appears to be at least one way in which it does not, notably in the amount that the weighted average cycle length varies. There only needs to be one piece of exploitable structure to incrementally or radically change the best algorithms for solving the DLP. While exactly how such an exploit would work is not clear from the results here, the results do encourage further work in this line of inquiry. Also encouragement for further study comes from the work of Lindle in [6] on binary graphs. His findings support this paper's, insofar as he found that variance in the weighted average cycle length was significantly different in the DLP graphs as opposed to random binary graphs. He did not find the same order of magnitude difference, but there still seems to be evidence that the DLP is imposing structure. Of note, Lindle did not find that the variance in the number of cycles differed significantly between the expected and the observed value for binary graphs.

The limited cycle distribution analysis of this paper is a first step in analyzing exactly where the difference in cycle structure lies. Though this paper only found significant deviance from the predictions in the 1-cycle and 2-cycle case, there are likely others which are contributing to the order of magnitude variance discrepancy described earlier. A more thorough analysis could pinpoint exactly what cycle lengths are systematically deviating from the random case and causing the difference. The data obtained for the 1- and 2-cycle cases is useful experimental evidence to compare to theoretical estimates for the frequency of these cycle lengths in the DLP mapping found in [5]. Yet the limited distribution analysis here is really just the beginning for a more complete analysis since there exists evidence that the cycle structure of the mapping $x \mapsto b^x \pmod{p}$ is not entirely random.

Future Work

As mentioned above, a good way of immediately continuing this work would be to conduct a more complete distribution analysis to get a better picture of where the cycle structure is deviating from random graphs. Also, obtaining the theoretical variance for the maximum cycle statistic might be useful since the most significant results have concerned variances. Lastly, a theoretical explanation should be sought for the parameters that varied significantly.

From a broader perspective, continuing this work should include applying the variance analysis to ternary graphs and beyond. The experimental data will be easy to obtain since the computer program used here generalizes easily to any kind of functional graph. The challenge will be in obtaining the theoretical values, which come from understanding a greater variety of random graphs. To this end, Max Brugger and Christina Frederick have done work with ternary graphs in [3] and [2], albeit without the variance analysis. The more theoretical data that exists, the more complete the comparison to the DLP will be. At all stages, applications of the uncovered structure should be actively sought in the form of new or refined algorithms since the DLP and its implications to cryptography make these algorithms important to mathematicians and non-mathematicians alike.

Acknowledgments

I would like to sincerely thank Josh Holden for all of his guidance and excellent advice throughout this project. I would like to acknowledge Dan Cloutier for his computer program and Nathan Lindle for his updates to it.

Appendices

Appendix A

Number of Cycles

Prime	Obs Avg	Std Dev	Exp Avg	t-stat	p-val	Obs Var	Exp Var	p-val
100103	12.103	3.220	12.092	0.76	0.449	10.370	10.446	0.021
100823	12.095	3.234	12.098	-0.26	0.799	10.457	10.453	0.918
100847	12.093	3.225	12.099	-0.39	0.694	10.398	10.454	0.093
101027	12.098	3.238	12.100	-0.14	0.885	10.485	10.455	0.364
101183	12.098	3.241	12.102	-0.25	0.803	10.506	10.457	0.137
101747	12.107	3.235	12.107	-0.03	0.977	10.463	10.463	0.980
101939	12.123	3.239	12.109	0.96	0.338	10.492	10.464	0.408
101987	12.090	3.225	12.110	-1.38	0.169	10.397	10.465	0.039
102407	12.108	3.223	12.114	-0.41	0.683	10.389	10.469	0.015
103007	12.130	3.261	12.120	0.70	0.484	10.633	10.475	0.000
99991	12.113	3.235	12.090	1.11	0.269	10.467	10.445	0.646
100057	12.054	3.218	12.091	-1.98	0.048	10.357	10.446	0.037
100279	12.099	3.230	12.093	0.34	0.736	10.431	10.448	0.681
100333	12.093	3.241	12.093	-0.03	0.975	10.501	10.449	0.193
100361	12.115	3.230	12.094	1.24	0.214	10.430	10.449	0.632
100393	12.124	3.233	12.094	1.68	0.093	10.451	10.449	0.970
100537	12.096	3.211	12.096	0.05	0.964	10.307	10.451	0.000
100741	12.028	3.203	12.098	-3.47	0.001	10.260	10.453	0.000
100937	12.086	3.228	12.099	-0.88	0.377	10.423	10.455	0.372
101009	12.096	3.230	12.100	-0.32	0.746	10.434	10.455	0.529
100609	12.110	3.253	12.096	0.77	0.442	10.579	10.451	0.002
100801	12.079	3.245	12.098	-0.88	0.380	10.531	10.453	0.108
101089	12.131	3.243	12.101	1.61	0.106	10.514	10.456	0.169
102061	12.113	3.257	12.111	0.10	0.918	10.609	10.466	0.003
102913	12.136	3.223	12.119	0.97	0.333	10.390	10.474	0.037
103681	12.150	3.228	12.126	1.22	0.223	10.416	10.481	0.146
105601	12.114	3.244	12.145	-1.50	0.133	10.522	10.500	0.635
106273	12.133	3.248	12.151	-1.04	0.298	10.550	10.506	0.270
106753	12.189	3.262	12.154	2.03	0.042	10.638	10.511	0.001
106921	12.128	3.216	12.157	-1.45	0.146	10.340	10.512	0.000

Appendix B

Maximum Cycle Length

Prime	# Graphs	Obs Mean	Std Dev	Exp Mean	t-stat	p-val
100103	50050	62477.4	19287.3	62497.3	-0.23	0.818
100823	50410	63091.0	19368.5	62946.8	1.67	0.095
100847	50422	62917.9	19332.7	62961.8	-0.51	0.610
101027	50512	63020.3	19398.9	63074.2	-0.62	0.533
101183	50590	63078.7	19413.5	63171.6	-1.08	0.282
101747	50872	63416.5	19542.7	63523.7	-1.24	0.216
101939	50968	63559.2	19639.8	63643.6	-0.97	0.332
101987	50992	63696.3	19611.7	63673.5	0.26	0.793
102407	51202	63923.4	19646.4	63935.8	-0.14	0.887
103007	51502	64283.3	19841.4	64310.4	-0.31	0.757
99991	24000	62272.5	19136.5	62427.4	-1.25	0.210
100057	30240	62627.7	19229.2	62468.6	1.44	0.150
100279	33372	62624.4	19268.7	62607.2	0.16	0.871
100333	33408	62646.1	19260.0	62640.9	0.05	0.961
100361	36864	62535.0	19239.5	62658.4	-1.23	0.218
100393	32384	62647.0	19238.0	62678.4	-0.29	0.769
100537	32480	62887.5	19246.3	62768.3	1.12	0.264
100741	25344	62936.0	19349.2	62895.6	0.33	0.740
100937	43200	63027.3	19331.8	63018.0	0.10	0.921
101009	49184	63097.0	19387.9	63062.9	0.39	0.697
100609	33280	62830.5	19332.0	62813.2	0.16	0.870
100801	23040	62921.1	19451.3	62933.1	-0.09	0.925
101089	31104	62973.8	19357.9	63112.9	-1.27	0.205
102061	23328	63704.3	19662.1	63719.7	-0.12	0.904
102913	33792	64313.4	19735.1	64251.7	0.58	0.565
103681	27648	64642.7	19816.4	64731.2	-0.74	0.458
105601	25600	66004.7	20317.9	65929.9	0.59	0.556
106273	34560	66448.4	20379.9	66349.4	0.90	0.366
106753	35328	66645.7	20575.1	66649.1	-0.03	0.976
106921	25920	66874.1	20568.0	66754.0	0.94	0.347

Appendix C

Weighted Average Cycle Length

Prime	Obs Mean	Std Dev	Exp Mean	t-stat	p-val	Obs Var	Exp Var
100103	50060.7	20481	50052	0.09	0.925	419481542	835050884
100823	50549.0	20599	50412	1.49	0.135	424330807	847106444
100847	50365.5	20546	50424	-0.64	0.522	422155131	847509784
101027	50463.0	20598	50514	-0.56	0.578	424271207	850537894
101183	50493.6	20583	50592	-1.08	0.282	423653393	853166624
101747	50770.0	20774	50874	-1.13	0.259	431540532	862704334
101939	50904.9	20849	50970	-0.72	0.473	434681074	865963310
101987	51032.0	20849	50994	0.41	0.681	434666945	866779014
102407	51180.2	20835	51204	-0.26	0.796	434096030	873932804
103007	51485.6	21081	51504	-0.20	0.843	444405143	884203504
99991	49802.1	20277	49996	-1.48	0.139	411137357	833183340
100057	50191.4	20460	50029	1.38	0.168	418631078	834283604
100279	50174.8	20469	50140	0.31	0.756	418975580	837989820
100333	50171.9	20468	50167	0.04	0.965	418939705	838892574
100361	50035.2	20435	50181	-1.37	0.171	417582588	839360860
100393	50146.0	20476	50197	-0.45	0.654	419271605	839896204
100537	50357.3	20461	50269	0.78	0.437	418672009	842307364
100741	50413.3	20567	50371	0.33	0.743	422988497	845729090
100937	50467.8	20529	50469	-0.01	0.990	421434935	849023164
101009	50526.2	20612	50505	0.23	0.820	424874705	850234840
100609	50359.8	20541	50305	0.49	0.627	421949041	843514240
100801	50474.7	20665	50401	0.54	0.588	427024052	846736800
101089	50397.0	20590	50545	-1.27	0.205	423952655	851582160
102061	51022.3	20909	51031	-0.06	0.949	437197250	868037310
102913	51520.3	20976	51457	0.55	0.579	439996889	882590464
103681	51714.0	21038	51841	-1.00	0.315	442594385	895812480
105601	52863.5	21623	52801	0.46	0.644	467564748	929297600
106273	53211.3	21689	53137	0.64	0.524	470407347	941162544
106753	53373.6	21833	53377	-0.03	0.977	476683634	949683584
106921	53597.2	21884	53461	1.00	0.316	478913112	952675020

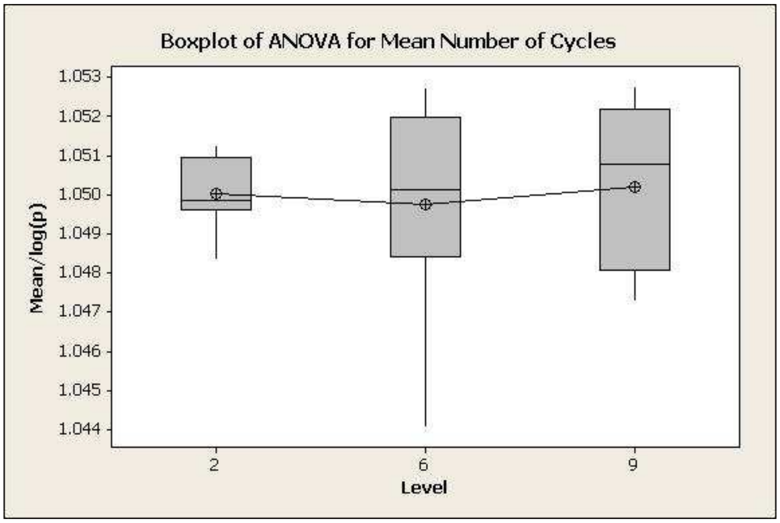
Appendix D

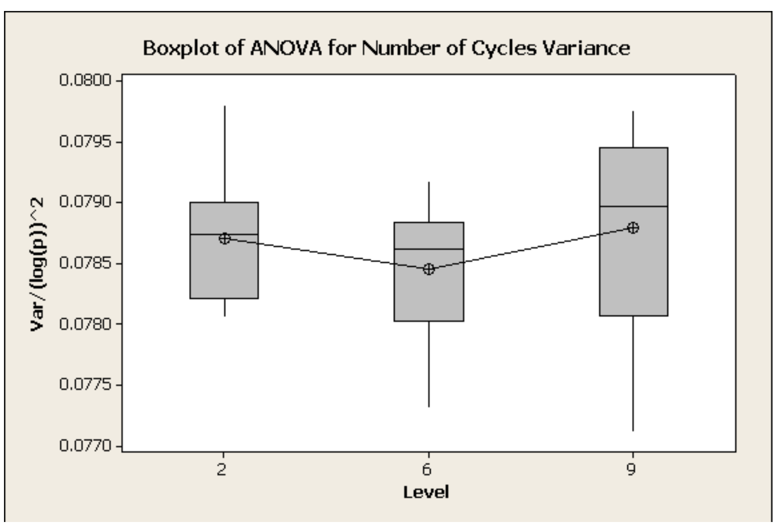
χ^2 p-values

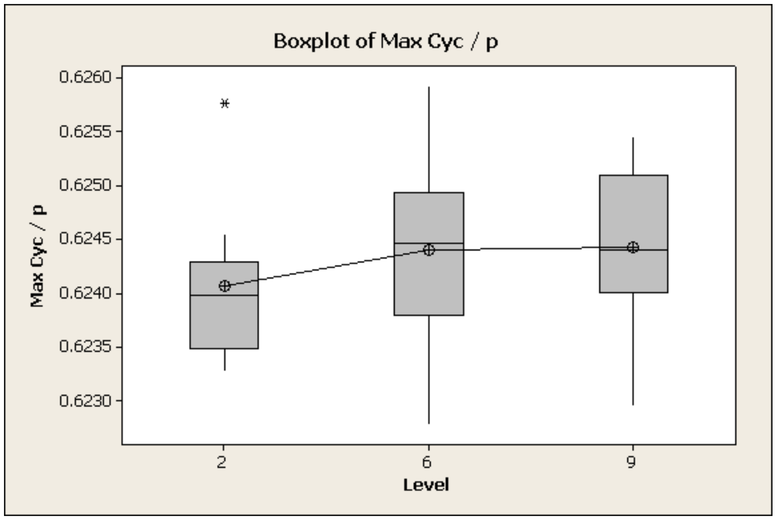
	1-cyc	2-cyc	3-cyc	5-cyc	7-cyc	10-cyc	20-cyc
100103	0.512	0.733	0.794	0.887	0.716	0.385	0.071
100823	0.287	0.425	0.343	0.511	0.832	0.257	0.994
100847	0.168	0.623	0.751	0.314	0.646	0.271	0.802
101027	0.771	0.252	0.039	0.117	0.384	0.757	0.005
101183	0.104	0.632	0.123	0.679	0.345	0.195	0.493
101747	0.104	0.153	0.199	0.295	0.101	0.656	0.010
101939	0.033	0.404	0.181	0.128	0.474	0.026	0.608
101987	0.033	0.404	0.181	0.128	0.474	0.026	0.608
102407	0.354	0.608	0.906	0.781	0.550	0.760	0.512
103007	0.617	0.329	0.981	0.436	0.162	0.924	0.909
99991	0.793	0.995	0.368	0.413	0.218	0.724	0.657
100057	0.026	0.131	0.761	0.558	0.801	0.650	0.547
100279	0.128	0.539	0.006	0.360	0.171	0.184	0.561
100333	0.003	0.034	0.757	0.795	0.655	0.757	0.539
100361	0.441	0.578	0.935	0.748	0.338	0.088	0.129
100393	0.477	0.050	0.438	0.774	0.694	0.825	0.017
100537	0.729	0.359	0.078	0.555	0.205	0.840	0.122
100741	0.000	0.929	0.555	0.821	0.225	0.548	0.913
100937	0.221	0.138	0.056	0.487	0.340	0.445	0.654
101009	0.000	0.534	0.138	0.648	0.639	0.506	0.211
100609	0.032	0.633	0.258	0.190	0.333	0.094	0.846
100801	0.080	0.567	0.580	0.184	0.914	0.466	0.583
101089	0.001	0.650	0.115	0.218	0.624	0.598	0.845
102061	0.171	1.000	0.004	0.377	0.664	0.842	0.373
102913	0.177	0.389	0.766	0.526	0.429	0.808	0.574
103681	0.027	0.727	0.430	0.202	0.273	0.442	0.536
105601	0.001	0.358	0.612	0.372	0.714	0.103	0.609
106273	0.309	0.279	0.793	0.596	0.743	0.770	0.458
106753	0.001	0.876	0.407	0.018	0.372	0.743	0.447
106921	0.016	0.986	0.379	0.221	0.900	0.532	0.513

Appendix E

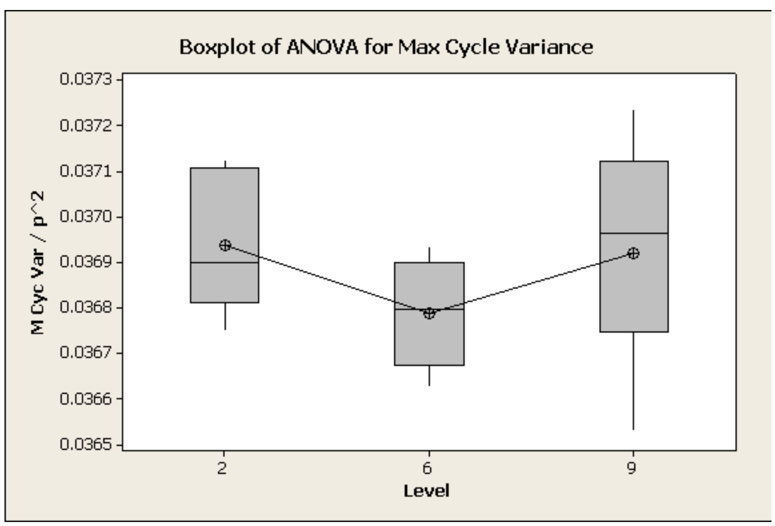
ANOVA Plots



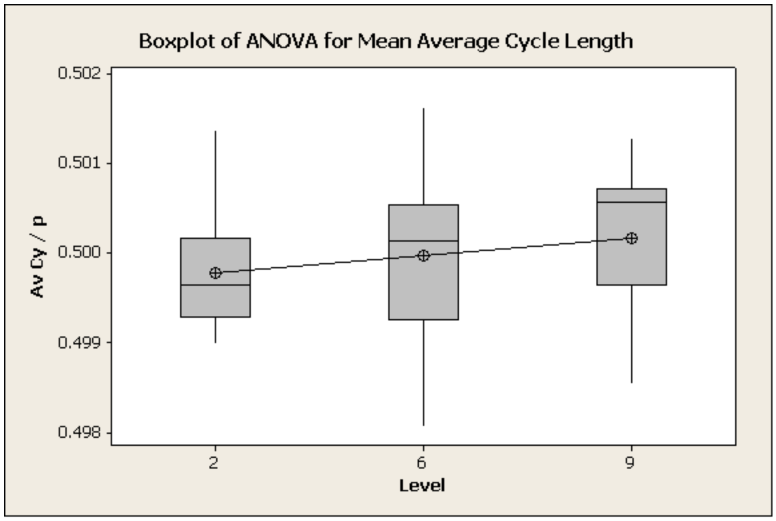


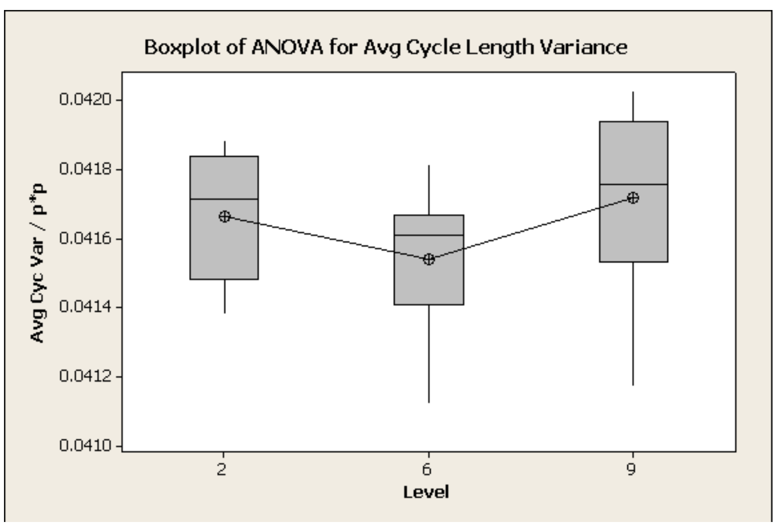


Boxplot of Max Cyc / p
0.6260
0.6255
0.6250
0.6245
0.6240
0.6235
0.6230
Max Cyc / p
2
6
9
Level



Boxplot of ANOVA for Max Cycle Variance
0.0373
0.0372
0.0371
0.0370
0.0369
0.0368
0.0367
0.0366
0.0365
M Cyc Var / p^2
2
6
9
Level


Boxplot of ANOVA for Mean Average Cycle Length
0.502
0.501
0.500
0.499
0.498
Av Cy / p
2
6
9
Level


Boxplot of ANOVA for Avg Cycle Length Variance
0.0420
0.0418
0.0416
0.0414
0.0412
0.0410
Avg Cyc Var / p*p
2
6
9
Level

References

[1] Richard Arratia and Simon Tavare. The cycle structure of random permutations. *The Annals of Probability*, 20(3):1567–1591, 1992.

[2] Max Brugger. Exploring the discrete logarithm with random ternary graphs. Senior thesis, Oregon State University, http://hdl.handle.net/1957/8777, 2008.

[3] Max Brugger and Christina Frederick. The discrete logarithm problem and ternary functional graphs. *Rose-Hulman Undergraduate Math Journal*, 8(2), 2007.

[4] Daniel Cloutier. Mapping the discrete logarithm. Senior thesis, Rose-Hulman Institute of Technology, http://www.csse.rose-hulman.edu/images/docs/theses/DanielCloutier2005.pdf, 2005.

[5] Joshua Holden. Distribution of the error in estimated numbers of fixed points of the discrete logarithm. *SIGSAM Bull.*, 38(4):111–118, 2004.

[6] Nathan Lindle. A statistical look at maps of the discrete logarithm. Senior thesis, Rose-Hulman Institute of Technology, http://www.csse.rose-hulman.edu/images/docs/theses/NathanLindle2008.pdf, 2008.

[7] J. M. Pollard. Monte carlo methods for index computation $(\mathrm{mod}\, p)$. *Mathematics of Computation*, 32(143):918–924, 1978.

[8] Bruce Schneier. *Applied cryptography (2nd ed.): protocols, algorithms, and source code in C*. John Wiley & Sons, Inc., New York, NY, USA, 1995.

[9] L. A. Shepp and S. P. Lloyd. Ordered cycle lengths in a random permutation. *Transactions of the American Mathematical Society*, 121(2):340–357, 1966.