Primer Design for DNA Origami Nanoparticles

Adam Tao

under the direction of
Prof. Mark Bathe
Laboratory of Computational Biology and Biophysics
Massachusetts Institute of Technology

Research Science Institute
August 2, 2016

Abstract

Primer design for DNA origami nanoparticles is often a tedious job. This paper provides an algorithm to design primers for the asymmetric polymerase chain reaction, which is used in amplifying DNA scaffolds of precise lengths. The Nearest Neighbor model is used in calculating the melting temperature of the primers. This paper also provides an algorithm to design indexing addresses for DNA data storage. CRC16 is used to convert indexing keys into 4-digit hexadecimal numbers, and the hashes are represented with 16-base DNA sequences as addresses. The addresses are an important part in retrieving data from the DNA origami used to store information, which makes DNA data storage practical. This research provides a programmatic way of sequence design for DNA origami nanoparticles, simplifying the process of primer design for aPCR and helping realize DNA data storage by designing addresses in data storage.

Summary

DNA origami is the formation of three-dimentional structures using long DNA strands called DNA scaffolds. To form DNA origami, the scaffold must have a precise length. Asymmetric polymerase chain reaction (aPCR), a method for generating large quantities of single strand DNA, can amplify DNA sequences of precise lengths. aPCR uses short DNA sequences called primers as starting points of amplification. This paper gives an algorithm to design the primers, which is a tedious job to do manually. DNA origami nanoparticles can be used in data storage, which are stable and easy to operate. This study also provides an algorithm to design indexing addresses to retrieve data from DNA origami, which is a foundamental part of DNA data storage. This research offers a programmatic way of sequence design for DNA origami nanoparticles, making it easier to form DNA origami and helping realize DNA data storage.

1 Introduction

DNA origami is the nanoscale self-assembly of DNA strands. A long single-stranded DNA (ssDNA) scaffold is folded by shorter DNA staple strands and hybridized with the staples, resulting in a complex DNA architecture. Scaffold production can be customized to precisely control the scaffold size. The output origami thus has a clear and neat shape, which is important when various DNA origamis are contained in one solution. [1]

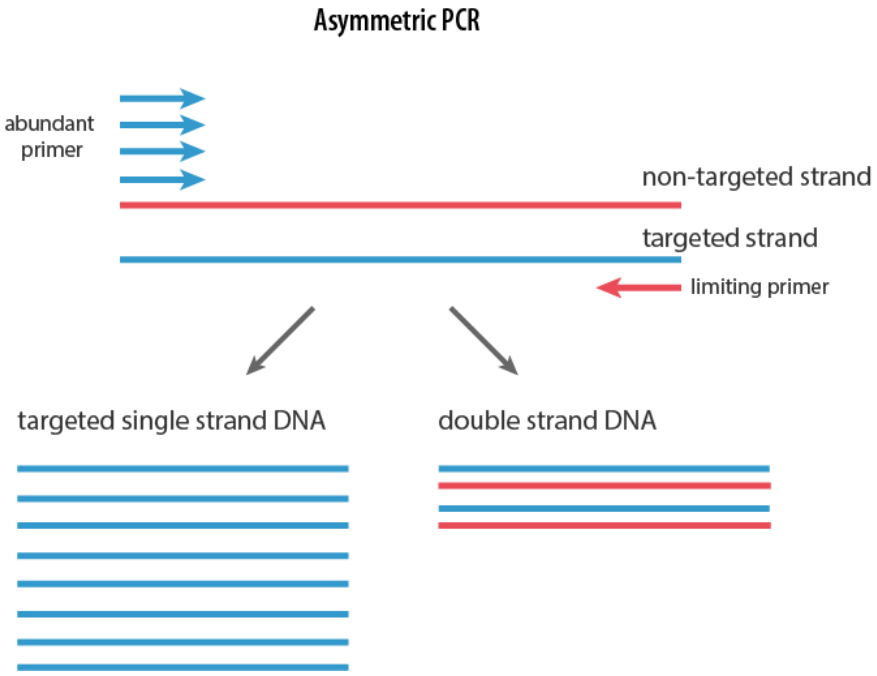

Figure 1: Asymmetric PCR: Blue strands and red strings are complementary by adding unequal primers, the majority of the output strands will be the targeted ones after PCR method[1]

Many geometric structures can be formed using DNA origami, which can be used in cellular delivery, nanoscale photonic materials, single-molecule imaging, and data indexing, among other applications [6]. DNA offers excellent utility in information storage. It encodes up to 450 exabytes per gram as a storage medium [2] [12]. Also, DNA is highly stable, remaining viable for millions of years under proper conditions [3]. Scaffolded DNA origami enables data storage in a complex structure instead of a linear sequence, forming a more

stable structure and allowing extra extractions and extensions like complex indexing.

Nevertheless, many difficulties still remain in DNA data storage, especially in asymmetric polymerase chain reaction (aPCR) primer design and data indexing. The longer the DNA scaffold is, the more expensive it is to synthesize. When the costs of synthesizing scaffolds become too expensive, we use asymmetric PCR to cut scaffolds of exact length from long DNA templates, usually are natural sequences, to form origami[6]. Asymmetric PCR is a PCR method which generates desired ssDNA instead of double-stranded DNA due to the unequal concentrations of primers used in reaction [1].

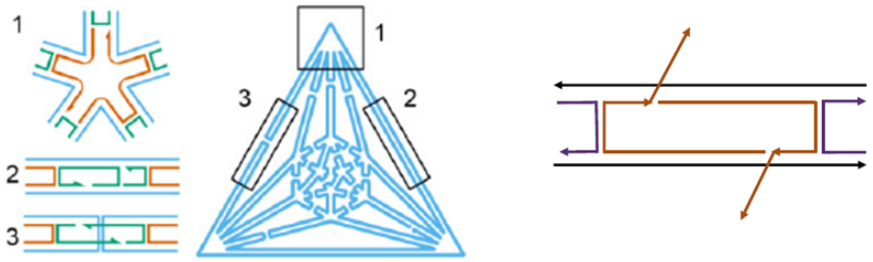

Figure 2: Overhangs in DNA architectures: overhangs can be tagged on the edges of DNA architectures[6]

However, aPCR primer design is a time-consuming job. Primers used in aPCR must have a suitable two-state melting temperature (T_m) to make aPCR experimentally viable. Previous research has already given several methods to calculate T_m.[4] [7] When the length of DNA sequence increases, finding a primer that has a suitable T_m takes more and more time.

The second problem in DNA data storage is data indexing. Since a pool of data may contain millions of DNA architectures which store infomation formed by DNA orgami, it is impossible to decode and sequence every architecture to index. A more efficient indexing method is necessary for data storage. The structure of DNA architectures formed by origami enables us to use overhangs to help index (Figure 2). These DNA architectures, called mem-

ory blocks, have DNA overhangs attached to each block for indexing. When trying to access the data that is tagged with a particular indexing string, input sequences which represent the indexing string are used to find the complementary sequences, which are attached as tags to the stored data of interest. These tags consist of a portion which is hybridized to the data strand, and a portion which floats freely as single stranded DNA for the search strand to find and bind to. However, if multiple searches are done it is necessary to be able to remove old search strands without destroying the hybridization between the indexing tags and the data strands. Therefore, it is necessary for the melting temperature of the search-tag hybridization to fall within a certain range. This can be achieved by designing the tags to have a certain GC content (fraction of the nucleotides that are G or C) in that part of the strand. This poses a problem because restricting melting temperature dramatically reduces the number of potential tag and search strands, which reduces the number of possible search strings that can be encoded. In this specific case, the range of T_m will be between 37-50 °C. Also, the overhangs need to match the possible indexing keys. For example, if an architecture contains detailed infomation about MIT, the phrase "STEM" can be one of the keys. Because there are an infinite number of different possible indexing keys, converting these strings into overhangs becomes a nontrivial problem.

In this paper we developed a algorithm which provides users with all suitable primer pairs used in aPCR for amplifying a certain specified length of DNA from long DNA templates. We also created a DNA sequence design method for overhangs which are used in DNA data storage, especially the deep indexing.

2 Methods

2.1 Primer Design for Asymmetric PCR

2.1.1 Calculate Two-state Melting Temperature

T_m of a DNA sequence depends on its entropy, enthalpy, and the molar strand concentration of the solution[7]. Whether a strand is self-complementary, in other words whether the strand will pair with itself, also influences its T_m. The entropy and enthalpy of DNA sequences can be calculated in various ways. Here we used the Watson-Crick Base Pair Nearest Neighbors Model (NN)[9][10]. NN calculates the entropy and enthalpy of every two adjacent nucleotides in the sequence, and uses the sum as the overall entropy and enthalpy of the sequence.

$$\text{Overall Entropy} = \sum_{i=1}^{n-1} \text{Entropy}_i. \tag{1}$$

$$\text{Overall Enthalpy} = \sum_{i=1}^{n-1} \text{Enthalpy}_i. \tag{2}$$

As equations 1 and 2 show, if we have a sequence of length n, the overall entropy and enthalpy will be the sum of these of every nucleotides neighbor in the sequence. The entropy and enthalpy of different nucleotides permutations are figured out by experiments, and the data we used here are based on experiments performed by HT Allawi and John SantaLucia(Figure 3). [8]

The sodium and magnesium concentration in the solution will also influence the entropy of the sequence[5], following the equation

$$\Delta S[Na^+] = \Delta S[1MNaCl] + 0.368 * N/2 * \ln[Na^+] \tag{3}$$

where $[Na^+]$ represents the concentration of sodium ions in the solution.

Table 1: Nearest-Neighbor Thermodynamic Parameters for Watson–Crick Base Pair Formation in 1 M NaCl[a]

propagation sequence	$\Delta H°$ (kcal/mol)	$\Delta S°$ (eu)	$\Delta G°_{37}$ (kcal/mol)
AA/TT	-7.9 ± 0.2	-22.2 ± 0.8	-1.00 ± 0.01
AT/TA	-7.2 ± 0.7	-20.4 ± 2.4	-0.88 ± 0.04
TA/AT	-7.2 ± 0.9	-21.3 ± 2.4	-0.58 ± 0.06
CA/GT	-8.5 ± 0.6	-22.7 ± 2.0	-1.45 ± 0.06
GT/CA	-8.4 ± 0.5	-22.4 ± 2.0	-1.44 ± 0.04
CT/GA	-7.8 ± 0.6	-21.0 ± 2.0	-1.28 ± 0.03
GA/CT	-8.2 ± 0.6	-22.2 ± 1.7	-1.30 ± 0.03
CG/GC	-10.6 ± 0.6	-27.2 ± 2.6	-2.17 ± 0.05
GC/CG	-9.8 ± 0.4	-24.4 ± 2.0	-2.24 ± 0.03
GG/CC	-8.0 ± 0.9	-19.9 ± 1.8	-1.84 ± 0.04
init. w/term. G–C[b]	0.1 ± 1.1	-2.8 ± 0.2	0.98 ± 0.05
init. w/term. A–T[b]	2.3 ± 1.3	4.1 ± 0.2	1.03 ± 0.05
symmetry correciton	0	-1.4	0.4

[a] Errors are resampling standard deviations (see text). [b] See text for how to apply the initiation parameters.

Figure 3: Enthalpy and Entropy of nearest neighbors(ΔH = Enthalpy, ΔS = Entropy)

Using the data provided, we can calculate the melting temperature with the equation[7]

$$T_m = \Delta H * 1000/(\Delta S + R * \ln(C_T/x)) - 273.15. \tag{4}$$

where R is the gas constant and C_T is the molar strand concentration.

Based on experimental requirements of aPCR, the optimum T_m for forward primers should be between 55-57 °C, and the reverse primers should have a T_m 1-3 °C higher than the forward ones'.

2.1.2 Algorithm for primer design

The basic process of designing primers for a given DNA template is shown in Figure 4.

To implement the T_m calculation to primer design, we considered the length of primers. To make the annealing temperature of PCR reaction approximate to T_m of the primers,

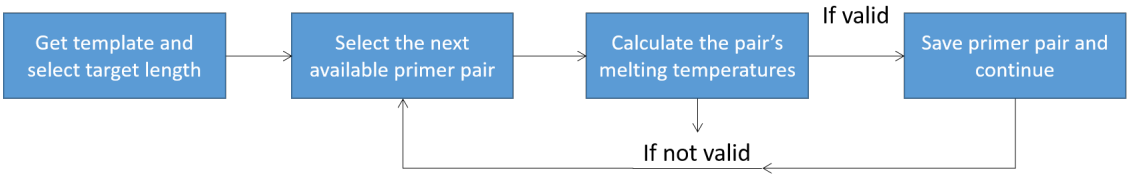

Figure 4: The process of primer design for aPCR

the length of primers had to be controlled between 18 bases to 24 base[11]. Based on the experimental requirements of asymmetric PCR, we further cut the range to 18-22 bases. Also aPCR requires primers of same length, and a target output length should be known before aPCR. To make the experiments easier, we restricted that the forward primers start with a G or C and end with an A or T, the reverse primers start and end with a C or G. Keeping that in mind, we developed a simple algorithm with pseudo code:

Algorithm 1 Primer Design

```
1: procedure
2: for primer.length = 18 to 22
3:     for i=1 to sequence.length − primer.length:
4:     read nucleotides of length primer.length from i^th nucleotide;
5:     calculate Tm_1 of this sequence;
6:     if 55 <= Tm_1 <= 57 then
7:         read nucleotides of length primer.length from (i + target.length)^th nucleotide;
8:         calculate Tm_2 of this sequence;
9:         if Tm_1 + 1 <= Tm_2 <= Tm_1 + 3 then
10:            if the forward primer starts with a G or C and ends with an A or T then
11:                if the reverse primer starts and ends with a C or G then
12:                    output forward primer, reverse primer;
```

2.2 Overhang Design for DNA Memory Blocks

Melting temperature is also important for overhangs design because of the biological reactions. Here we restricted the T_m to a range between 37-50 °. We found the correlation between this T_m limit and the overhang length and GC content. Another difficulty is how to

convert the infinite indexing keys into DNA sequences. We used a hash function to hash the information and used DNA sequences to represent the hashes.

2.2.1 Overhang Length and GC content

Using Equation 4, we can calculate the T_m if we know the exact DNA sequence. We concluded that the T_m will increase as the length of DNA sequence increases. Thus we developed an algorithm to generate all possible DNA sequences and found a range of sequence length which produces a T_m between 37-50 °C. The result indicated that over 95% of the sequences have a suitable T_m were between 13-22 nucleotides. While of each lenth, there was a certain range of GC content to obtain the suitable T_m. We also developed a program to figure out whether we could reverse the process. Instead of finding possible sequences of certain length and certain GC content based on given T_m, we want to find one or several combinations of sequence length and GC content that can always produce the suitable T_m. Several valid combinations were found. To make convertion between keys and sequences easier, we focused on the DNA sequences of length 16, which provided most available DNA sequences which satisfy the T_m. The total GC number in the sequence is 7.

2.2.2 Cyclic Redundancy Check(CRC)

So far the overhangs we designed managed to obtain suitable T_m and GC content, but to be used in indexing, they still have to map the possible indexing keys. We used a hash function called Cyclic Redundancy Check(CRC) to map the keys with DNA sequences. A hash function is a way to convert information into hashes, usually through compression.

The hash of CRC16 is a four-digit hexadecimal number. CRC16 provides 65536 different hashes, which is a satisfactory number for indexing as a starting point. As a string is inputted into the computer, it stores as a sequence of binary numbers. CRC16 takes the binary number, doing modular operation with a divider and doing boolean logic operation(specifically Xor

operation). The dividers vary from different CRC methods. For CRC16, a common divider is a hexadecimal number 8005.

After getting the 4-digit hashes from CRC16, we used DNA nucleotides to represent each digit. We used permutations of two DNA nucleotides to represent the hexadecimal numbers. Thus the first 8 bases of the overhangs were used to represent the results of CRC16. The number of Gs and Cs in the latter 8 nucleotides is determined by the first 8 ones. If we got different keys represented by the same hash, we assigned different latter 8 bases to distinguish them. If the GC number was 7 or 8, we would always produce a suitable T_m. The process of overhang design is shown in Figure 5.

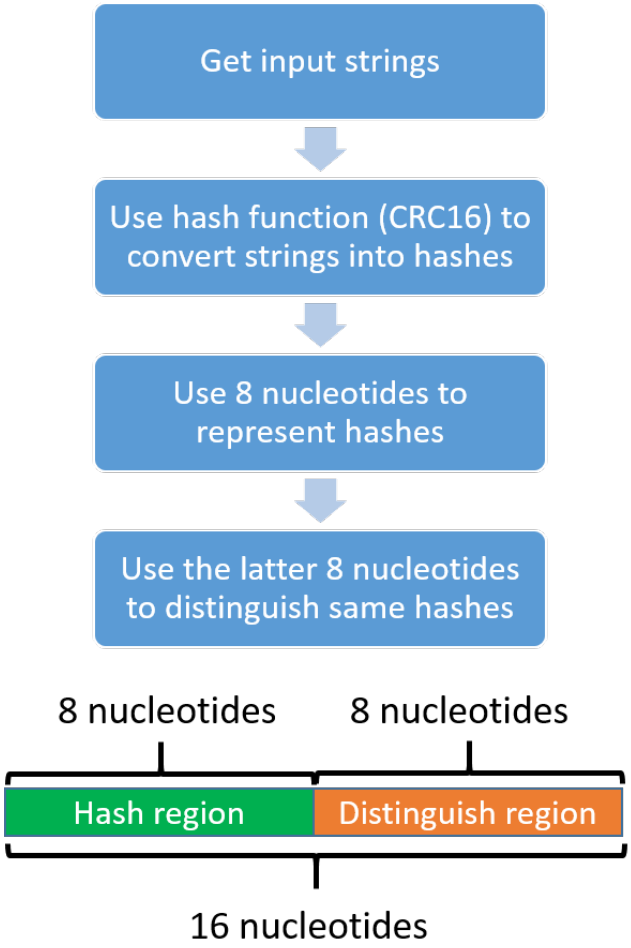

Figure 5: The process of overhangs design

We used pseudocode to illustrate the process of overhangs design:

Algorithm 2 Overhangs Design

```
procedure
for i = 1 to totalnumberofkeys
    Get the i^th indexing key;
    Use CRC16 to hash the indexing key;
    Use 8 nucleotides to represent the hash (which is a 4-digit hexadecimal number);
    Calculate the GC content of the 8 nucleotides;
    if GC content <= 7 then
        Assign the latter 8 nucleotides to keep the total number of GC 7;
    if GC content = 8 then
        Assign the latter 8 nucleotides which do not contain G or C;
    Output the indexing keys and designed overhangs;
```

3 Results

3.1 Primer design for Asymmetric PCR

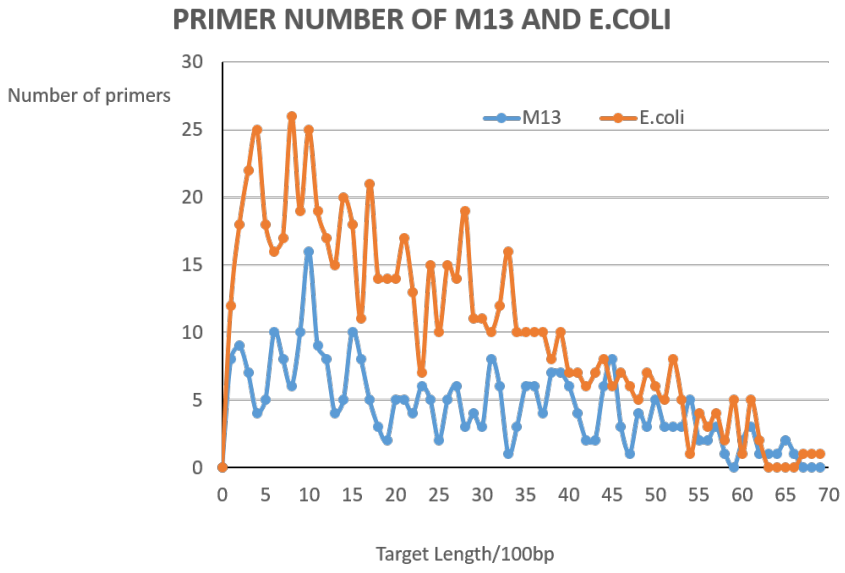

Figure 6: Numbers of primers for two template (Target length from 100 to 6900)

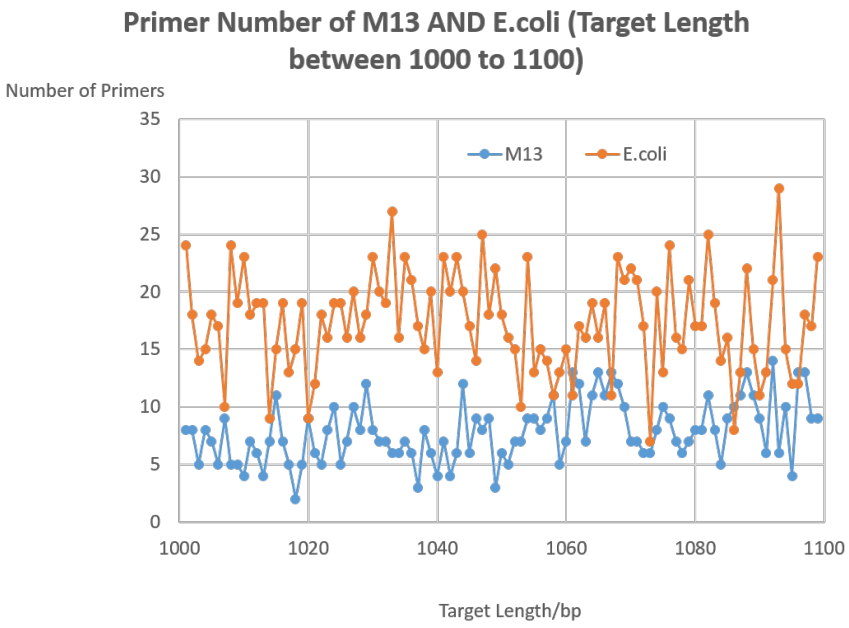

Figure 7: Numbers of primers for two template(Target length from 1000 to 1099)

We used the algorithm we developed to find primers for M13 sequence and *E.coli* sequence of different target length. Given the same template length (7249bp, the length of M13), the primers number for two templates of different target lengths is shown in Figure 6.

From the graph, the number of available primer number on *E.coli* is higher than that on M13. For *E.coli* sequence, when the target length is below 4000bp, the number of available primers is mainly above 10. While there are few target length which provide more than 10 primers on M13. When we look at more specific target length (Figure 7), *E.coli* still has more primer numbers. Here we randomly chose 1000-1100 target length to show the fluctuations of the primer numbers between intervals.

Using the primers we designed, the lab cut DNA scaffold from M13 sequence and constructed a DNA ladder. The experimental result (Figure 8) demonstrated the primers designed are valid. In each lane the brighter band is the double-stranded DNA and the blocky orange-ish band is the single-stranded DNA that can be used for folding nanostructures. The

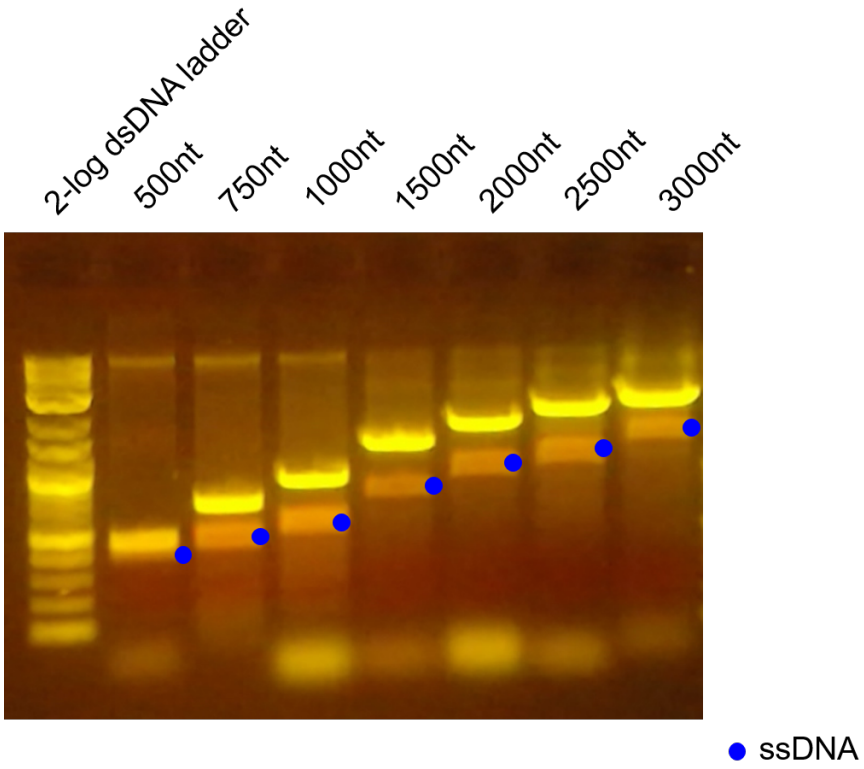

Figure 8: DNA ladder constructed using designed primers

lowest band near the bottom are left-over primers.

3.2 Overhang Design for DNA Memory Blocks

Our method to convert indexing keys into DNA sequences can theoretically provide about 750 million different DNA sequences to represent hashes. We tested the collision with an input of 10000 strings of different lengths and 3000 integer numbers. The results showed no collisions in DNA sequences; every key is assigned a unique overhang.

We conducted tests to make sure 16 nucleotides is the optimum length for overhangs. First we calculated the T_m of 8000000 different DNA sequences, from 11 to 50 nucleotides long. The relation between number of available sequences and their length is shown in Figure 9.

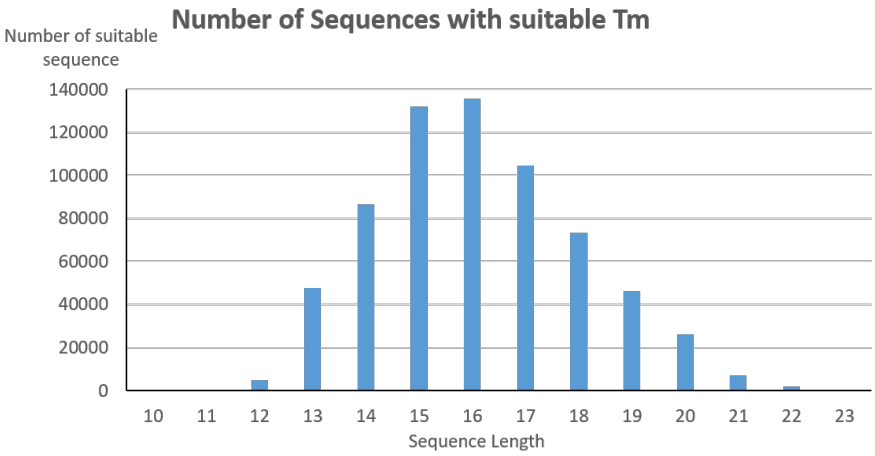

Figure 9: Number of Sequence with suitalbe Tm

Then we tried to find a suitable GC content in that 16-nucleotide DNA sequence which always leads to a suitable T_m. The relation between the probability to have a suitable T_m and the number of GC in the sequence is shown in Figure 10.

4 Discussion

4.1 Melting Temperature

T_m is a critical factor in both asymmetric PCR and overhang design. For aPCR, it determines whether the PCR is experimentally practical. The lengths of the two primers and the sequence between them are also important in cutting scaffold from a long DNA sequence. For overhang design, a suitable T_m is necessary to allow the biological reaction, since the overhangs need to pair with the input DNA sequences for indexing. The melting temperature should be kept relatively low to avoid accidentally unwinding the DNA architectures while indexing. The algorithms we developed highly depend on the calculation of T_m, because of its importance in making algorithms experimentally viable.

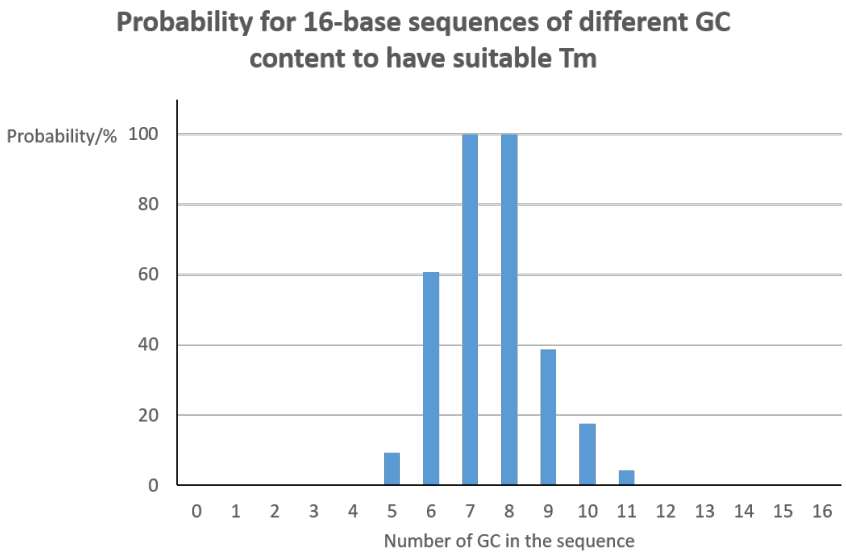

Figure 10: Probability for 16-base sequences of different GC content to have suitable Tm

4.2 Overhang Length

To decrease the possibility of mismatch, we kept overhangs the same length. Mismatch is that a DNA sequence pairs with an unexpected sequence. Mismatch will happen when one sequence is the subset of another sequence, or two DNA sequences are largely similar. To keep overhangs the same length, we avoided the situation that one sequence is others' subset, and therefore reduced the possibility of mismatch. The length of overhangs we used in data storage must satisfy two restrictions: it must provide a decent number of different DNA sequences, and there must be a certain GC content for DNA sequences of that length to always have a suitable T_m. If there is no GC content can always produce a suitable T_m, the length may not be used since we would need to check the T_m for every overhang we get, which was time-consuming.

As Figure 9 and Figure 10 show, the length of 16 nucleotides provides the most different DNA sequences which have suitable T_m, and when the number of GC on the strands is 7 or 8, they will always have T_m in the required range. Thus we decided to use 16-nucleotide

sequences as overhangs, and kept the number of GC on the sequences as 7.

4.3 Comparison between Hash Functions

Many different hash functions have been developed during the past few decades. Here we tested several popular hash functions to decide which to use. When we store data with DNA origami nanoparticles, we only need to tag the overhangs onto each architecture once. Also we allow the speed of deep indexing to be a bit slow because of the biological reactions. Thus the speed of the hash function is not a major concern, which is always fast compared with biological reaction. However, the important factors to evaluate a hash function here are the collisions and the randomness of the hashes.

4.3.1 Collisions

In data storage, we want each DNA sequence to represent a unique piece of information. The hash function should have as few collisions as possible, in other words, it should provide as many different hashes as possible. The outputs of most hash functions are hexadecimal numbers, usually are 4-digit hexadecimal numbers, 8-digit hexadecimal numbers, and 16-digit hexadecimal numbers. Restrict the length of overhangs to be 16 nucleotides and the GC content to be 7 nucleotides, we want output of hash function to be a 4-digit hexadecimal number. 8-digit hexadecimal numbers may not be a good choice since we have to use the permutation of two nucleotides to represent one hexadecimal number, which means no adjustments can be made about the GC content. Among all the hash functions that have a 4-digit output, CRC16 provides the least collisions.

4.3.2 Randomness

The randomness of a hash function means the distinctness of hashes representing similar keys. Here we define similar keys as the keys which have only 3 or below 3 different letters

or numbers. We tested the randomness with 10000 English words, each is similar to several others. The 16-base encoded DNA sequences for these words showed that for the majority of similar keys, CRC16, together with the latter 8 nucleotides we add, provided a difference of over 10 nucleotides. The least difference provided is 6 nucleotides, which is also enough to avoid mismatch.

4.4 Complexity of algorithms

Though the speed of encoding indexing keys and generating primers is not one of the most important factors in primer design, we still need to analyze the complexity of algorithms we used to evaluate the algorithms. The complexity gives us an estimation of time cost for the algorithms and their performance handling different inputs.

4.4.1 Primer Design for aPCR

The key factor of primer design for aPCR is the template length. By intuition, the longer the template DNA sequence is, the longer it takes to generate all primers. According to the pseudocode (Algorithm 1), the two iteration loops give a coefficient approximate to $5(n-20)$ to the total operation of the algorithm, where n is the length of the DNA template. The judgements for "if" states and the process of reading inputs are rather fast, and they have a constant time cost. Here we simply used constant 1 to represent the time consumed. However, the process of calculating T_m takes $2(2n+k)$ operations, where k is an integer between 1 and $n/2$. Thus, by multiply $2(2n+k)$ with the coefficient $5(n-20)$, we calculated the total number of operations of the algorithm:

$$2(2n+k) * 5(n-20) = 10(2n^2 - 40n + nk - 20k), \tag{5}$$

where

$$10(2n^2 - 39n - 20) < 10(2n^2 - 40n + nk - 20k) < 10(3n^2 - 60n) \tag{6}$$

Thus, the complexity of the algorithm is $\Theta(n^2)$.

4.4.2 Overhangs Design for DNA origami

The number of indexing keys to encode is the most important factor that needs to be considered in overhang design. We use n to represent the number of keys. The number of operations to perform hash function CRC16 equals the length of the key. We use variable l to represent it. A normal indexing key will not be too long, so let's suppose $1 \leq l \leq 100$. Using 8 nucleotides to represent the hash takes 4 operations, and calculating the GC of first 8 nucleotides takes 8 operations. Thus, we can calculate the complexity of the alogrithm:

$$n(2l + 4 + 1) = (2l + 5)n \tag{7}$$

The complexity of algorithm for overhangs design is $\Theta(n)$.

4.5 Prospect

In the future, we can further improve the method for overhang design. In the present design, the encoded DNA sequences are randomly distributed, while if further research is conducted, we may make the encoded DNA sequences similar if the indexing keys they represent are correlated with each other. For example, if two keys are "blue" and "sky", the DNA sequences represent them will be similar. The design will make the indexing much easier and user-friendly since related keys will point towards the same data block.

5 Conclusion

In this paper, we developed algorithms to design primers for asymmetric PCR and overhangs for DNA memory blocks. Both algorithms are easy to use, requiring only basic biology knowledge. We used the Nearest Neighbor model to calculate the melting temperature for primers and overhangs to ensure they are experimentally viable. We used CRC16 to convert indexing keys into four-digit hexadecimal numbers, and used DNA sequences to represent the hashes. The overhangs serve as an important part in DNA data storage by making retrieving data possible. The study provides a programmatic way of primer designs for DNA origami nanoparticles, and can be used in forming and applying DNA origami.

6 Acknowledgments

I am grateful to Sakul Ratanalert for his consistent help on my project, my paper, and my presentation. I am also greatly appreciate the instructions from Dr. Tyson Shepherd on my project and the inspirations he gave me, and the advice given by Connor Duffy on the paper and the project. In addition, I would like to thank Prof. Mark Bathe, the Laboratory of Computational Biology and Biophysics to provide me with a great working atmosphere. Finally, I want to give my acknowledgments to Research Science Institute and MIT to give me this opportunity.

References

[1] Citartan Marimuthu et al. Asymmetric pcr for good quality ssdna generation towards dna aptamer production. *Sonklanakarin journal of Science and Technology*, 34.2:4, 2012.

[2] Goldman N. et al. Towards practical, high-capacity, low-maintenance information storage in synthesized dna. *Nature*, 494:7780, 2013.

[3] Miller W. et al. Sequencing the nuclear genome of the extinct woolly mammoth. *Nature*, 456:387390, 2008.

[4] Owczarzy Richard et al. Predicting sequencedependent melting stability of short duplex dna oligomers. *Biopolymers*, 44.3:217–239., 1997.

[5] Owczarzy Richard et al. Predicting stability of dna duplexes in solutions containing magnesium and monovalent cations. *Biochemistry*, 47.19:5336–5353, 2008.

[6] Veneziano Rmi et al. Designer nanoscale dna assemblies programmed from the top down. *Science (2016)*, aaf4388.

[7] John SantaLucia Jr. and Hicks Donald. The thermodynamics of dna structural motifs. *Annual Review of Biophysics and Biomolecular Structure*, 33:4–9, 2004.

[8] Allawi Hatim T. and John SantaLucia. hermodynamics and nmr of internal g t mismatches in dna. *Biochemistry*, 1997.

[9] Allawi Hatim T. and John SantaLucia. Nearest neighbor thermodynamic parameters for internal g a mismatches in dna. *Biochemistry*, 1998.

[10] Allawi Hatim T. and John SantaLucia. Nearest-neighbor thermodynamics of internal a c mismatches in dna: Sequence dependence and ph effects. *Biochemistry*, 1998.

[11] Dieffenbach C. W., T. M. Lowe, and G. S. Dveksler. General concepts for pcr primer design. *PCR Methods Appl*, 1993.

[12] Church G. M.and Gao Y. and Kosuri S. Next-generation digital information storage in dna. *Science*, 337:1628, 2012.

Appendix A Title of Appendix

Appendices may appear after the paper proper. Appendices may hold extra information that would interrupt the flow of the paper and that is not absolutely necessary for the reader to appreciate the work. For example, a large number of related figures or a mathematical derivation could go nicely in an appendix.