A Mismatch Relaxation to the Primer Selection Process of an Amplicon Sequencing Algorithm

Dean Polimac[1]

Supervisor(s): Dr. Jasmijn Baaijens[1]**, Jasper van Bemmelen**[1]

[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
January 28, 2024

Name of the student: Dean Polimac
Final project course: CSE3000 Research Project
Thesis committee: Dr. Jasmijn Baaijens, Jasper van Bemmelen, Chirag Raman

Abstract

In this study we introduce a different approach to the primer selection problem in the AmpliDiff [12] algorithm. Two different metrics, being the Hamming distance [2], and the Levenshtein distance [4], are used to compute sets of similar primers. This is done such that locations where mismatches between the primers and the target sequence can occur are determined by the locations of similar primers. The impact that said mismatches have on the solution set of amplicons and their respective primers is outlined. With this we show potential benefits of allowing mismatches to occur, as well as their drawbacks.

1 Introduction

With the vast advancements in the field of computer science, and the increase in the computing power, target sequencing of genomes has become more accessible. Being able to extract genetic information and analyze it through the use of computers has allowed breakthroughs in the field of medicine in regards to genetic sequencing [13].

The significance of this research lies in the ability to clearly distinguish between different lineages solely based on the selected sequence regions. Swift and efficient differentiation between common ancestors is important when dealing with microbial communities, such as viruses which are prone to frequent mutations. The importance of this became evident during the Covid-19 global pandemic, during which the virus mutated several times within just two years. In such scenarios, the ability to rapidly distinguish between various common ancestors is crucial, as it allows for earlier detection and treatment of symptoms, as well as potential cures. [7].

Amplicon or a target sequence is a part of the genomic sequence which is replicated, usually through the means of a PCR (Polymerase Chain Reaction) [12]. The way that they are replicated is by having primers bind to certain parts of the sequence [8]. A primer is defined as a "short single-stranded DNA fragment" that is used in DNA synthesis [6]. The importance of amplicons lies in the fact that they allow us to discriminate between different common ancestors. However, this is only possible if those amplicons can be replicated (amplified) through the use of primers.

The AmpliDiff [12] algorithm is a tool used to sequence whole genomes and find highly discriminatory regions, as well as the primers needed to amplify them, i.e. a primer binds to a particular part of the genetic sequence and replicates it. However, one of its limitations is the fact that it relies solely on primers which exactly bind to the target sequence. In real life this is not always the case, as mismatches between the primer and the sequence may occur [1]. With this in mind, a possible improvement to the algorithm would be including a relaxation which would allow for primers that have a mismatch with the sequence to bind as well.

The said relaxation deals with the primer selection part of the algorithm. Potential benefits of allowing mismatches between the primer and the sequence is that it could enable certain amplicons to be used in the differentiation process. For example, due to infeasibilites of certain primers, it might not be possible to amplify a target region of the sequence. Despite this, inexact matching would increase the number of primers which could be used in the amplification process.

Possible advantages of this approach might be that the time it takes to find a set of amplicons needed to discriminate between all the sequences is reduced. This is because finding amplicons which can be replicated should become easier, as the set of feasible primers increases. Under this assumption, it would make sense to consider that with an increase in the size of the primer set, more regions can be amplified.

The possible impacts of this relaxation could be reflected on the set of feasible amplicons, as well as on the size of the primer set used to amplify them. One of the main focus of this research revolves around the impact of inexact matching on the amplicons selected. This is important as it might show a the difference in the solution set of amplicons compared to the original algorithms output. This in turn begs the question of how are the previously selected amplicons impacted. For example, if an amplicon is in both solution sets is it amplified by the same primers?

Another important aspect which should be considered is the difference in the run-time of the algorithms. The particular example that we need to consider is the one where relaxation indeed does produce a better solution, but it takes significantly more time. This is because it raises the question of how much of a better solution does it give, and is it worth the extra time.

2 Background

During the implementation of the relaxation, numerous things were important to consider. This includes physio-chemical properties of primers, as well as similar attempts at solving the same or similar problem.

Firstly, the paper by Kwok, at al. [3] outlines the criteria such as nucleotides between which the mismatch takes place, optimal annealing temperatures, as well as the regions of the primer where the mismatch occurs. All of these are considered, and have to be satisfied in order for the mismatched primer to bind to the target sequence.

Work by Linhard and Shamir on developing the HYDEN-program [5] lays out the important definitions and constraints in regards to the degenerate primer design problem. In turn, it also delves into the difficulties of inexact matching between degenerate primers and the target sequence, and shows that it is a NP-Hard problem. It presents a thorough description of a dynamic programming approach which could be taken when designing degenerate primers which allow for mismatches.

Development of Primer3 [11] further emphasized the importance of integrating comprehensive thermodynamic models and user-defined constraints to generate primers tailored for specific experimental conditions. When considering mismatches, Primer3 focuses on the stability of neighboring pairs, as well as the thermodynamic aspects caused by the mismatch.

Primer-BLAST [15] is another tool used to design primers which allows for primer mismatches. It focuses on mismatches happening on the 3‘ end of the target and enforces that a primer pair must have at least two mismatches, by default. In addition to this it considers other details such as the number and the positions of matched bases, the primer orientations and distance between forward and reverse primers.

3 Methodology

This section aims to introduce the steps taken in order to implement the inexact matching relaxation. Given that this paper focuses on a specific part of the AmpliDiff algorithm, a general overview is first presented. This is followed by the explanation of the inexact matching implementation.

3.1 The original algorithm

The original AmpliDiff algorithm consists of four steps. The first step of the algorithm focuses on pre-processing the genomic sequences obtained from a database (e.g. NCBI [14]). Multiple sequence alignment is then applied to these sequences. This is done in order to ”identify the evolutionary relationships and common patterns between genes” [9]. These aligned sequences are then stored for use in latter steps.

In the second step, a set of feasible primers is built, where the feasibility of a primer depends on physicochemical constraints (annealing temperature, GC-content, etc.). The set of primers is computed by iterating over all the sequences with a given window size - representing the length of the primer - and then taking the reverse complement of the forward primer to get the reverse primer. It is important to note that all the primers are of the same length.

The third step determines which amplicons from the initial set are feasible. A pair of differentiable genomes is then stored for each amplicon. This is done since in the last step, amplicons are selected using a greedy algorithm such that the ones which have the highest discrimination are considered first. During the greedy selection, a set of feasible primers is computed such that a certain percent of the amplicon can be replicated. This percentage is also referred to as amplicon coverage or just coverage. In a case where such a set of primers is found, the amplicon is added to the final solution set, otherwise, the next best amplicon is considered. Once all the amplicons which allow for complete differentiation between the sequences are selected, the algorithm terminates and returns the set of said amplicons with their primers.

3.2 Implementation of the inexact matching

This relaxation concerns the second step of the above described algorithm. During the process where primers are found for each amplicon, we now also have to consider similar primers. In other words, the set of feasible primers for each amplicon is increased. This is done by computing a set of similar primers for each respective primer. Two primers are said to be similar if their respective similarity score is higher than some parameter *e* which is provided by the user. Similarity here is defined using two distance metrics, first being the Hamming distance [2], and the Levenshtein distance [4]. The decision to opt out for two metrics is due to the different types of mismatches which can occur.

The first type of mismatch is a replacement mismatch, which occurs when there is a difference between the two nucleotides, a trivial example of this can be seen in Figure 1. There is a mismatch occurring on the 2^{nd} position since nucleotides A and A are not complementary to each other. The other type of mismatch is an insertion/deletion mismatch which can be seen in Figure 2. In this case, on the 4^{th} position there is an insertion mismatch, where if there was a G nucleotide, it would be possible for the two sequences to bind.

The Hamming distance is faster to compute, and while it does not consider all the types of mismatches, it is still a slight improvement over only considering exact matches. That is why we also use the Levenshtein distance which takes into account all the possible mismatches between the primer and the sequence. This algorithm however relies on dynamic programming, and is hence computationally heavier than the Hamming distance algorithm. The main reason why the Levenshtein distance is not the only similarity metric that we use, is due to its high computational requirements.

5` - TAGGCTAATG - 3`

3` - AACCGATTAC -5`

Figure 1: Replacement mismatch occurring on the 2nd position. Nucleotides A-A are not complementary to each other.

5` - TTG-CTAATG - 3`

3` - AACCGATTAC -5`

Figure 2: Insertion/deletion mismatch occurring on the 4th position.

It is important to explain the reasoning behind doing a primer to primer similarity. This is done as it is more efficient than the generative approach, where all the possible primers for an amplicon would be considered. This yields $O(4^n)$ time complexity, where *n* represents the length of the primer, and at every position there are four possible nucleotides to consider. Another approach would be to iterate over all the sequences and try to find similar primers. However, since there is already a set of all feasible primers, it is simpler to consider only those feasible primers.

If we define a primer *P* of length *n* as $P = p_1 p_2 ... p_n$ where p_i represents one of the four possible nucleotides (A, C, G, T). It is important to note that in order for a primer to bind to the region of the sequence, it has to be composed of complementary nucleotides. Then we know that if there is no mismatch between a primer $P = p_1 p_2 ... p_n$ and a sequence, then if there is a similar primer $P* = p*_1 p*_2,,, p*_n$, the mismatches that happen between P and $P*$ occur on the same primer index as the mismatches between $P*$ and the sequence. This is because the complementarity of nucleotides can be expressed as a bijective function. With this in mind, the size of the primer set for each amplicon has either been increased or has remained the same.

In spite of this relaxation, there is an underlying issue. This allows for one primer to reoccur on two different locations in the sequence which seen in Figure 3. The problem which arises from this is that it might lead to non-specific bindings, where a primer binds to an unwanted region. The way this is solved is by checking whether the difference between regions enclose by the reoccurring primer are sufficiently different. In Figure 3 these regions are represented by *R1* and *R2*. This is because if one of the two regions is much larger, there is high likelihood that it will not actually be amplified. In this case it is safe to select the primer, otherwise the primer is discarded from the set of feasible primers for that amplicon. The different values used to define the region enclosed are 100bp, 400bp, and 1000bp. The justification for using these is that prior to LAMP [10] during PCR on the SARS-CoV-19 the feasible enclosed region by a primer pair is less than 150bp. However, the same paper states that when using LAMP the primer pair can amplify regions of up to 945bp, however, this we decided to round this value to 1000bp in order to allow for some leeway.

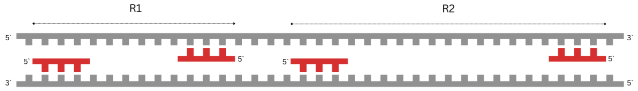

Figure 3: Figure depicting a reoccurring primer bind to the sequence. The red part represents the primer, and the grey part represents the sequence. The figure is an adaptation of a similar one found in ”Diagnostic Molecular Biology” by Shen, et al. [8]

4 Experimental Setup and Results

The experiment is set up such that the original version of the algorithm, as well as the two implementations using Hamming and Levenshtein Distance are ran on the SARS-CoV-2 dataset. The original sample set included 480 genome sequences, however, it is reduced to in order to get feasible results after 24 hours of runtime. Two data sets were produced, one containing 75 sequences, and a larger one containing 150 sequences. It is important to note that the dataset containing the 75 sequences, corresponds to the first 75 in the larger dataset. In regards to the metadata file, the number of multiple aligned sequences should be set. This data is contained within the metadata file, in correspondence with the previously multiple aligned sequences which serve as another input parameter. This data is publicly available through the https://www.ncbi.nlm.nih.gov/ [14] website. The metadata file containing the lineages used can be see in the GitHub Repository[1].

The reasoning behind using the SARS-CoV-19 is that we have a good benchmark with which we can compare the performance of the inexact matching algorithm. This is because the original AmpliDiff algorithm was also tested on SARS-CoV-19 data. With this in mind, it is easier to draw conclusions, as well as compare performance when using the same dataset, since our goal is to determine whether introducing this relaxation would yield any improvements. As mentioned before, these improvements are in regards to the number of amplicons selected, the run-time, and whether there is a difference in the primers used in the amplification process.

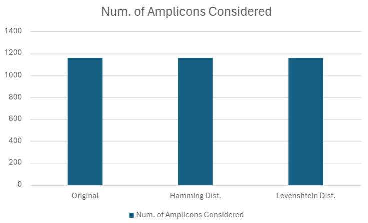

Figure 4: Number of amplicons considered during the optimization step for 75 sequences.

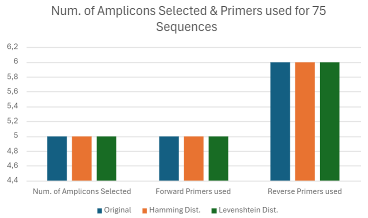

Figure 5: Number of amplicons selected, and the number of primers used in their amplification for 75 sequences.

The experiment is run on the TUDelft super-cluster ”DelftBlue” using 180GB of RAM and 12 CPU cores. Further information regarding the specifications of the cluster can be found here[2]. It is important to note that since the results of the algorithm were deterministic, each experiment is ran once. It is set up in such a way that we test for inexact matching using both the Hamming and Levenshtein distance, and we consider amplicons of width 400bp, and with full amplicon coverage. The only variable is the aforementioned distance constraint which ensures that the region enclosed by two same primer pairs is larger than some variable D which we set to 100bp, 400bp and 1000bp, as previously mentioned. Lastly, the number of allowed mismatches that is allowed is set to two.

The results can be seen in Figures 4 through 10.

5 Responsible Research

The algorithm uses data that is publicly accessible, with the main objective of differentiating lineages of common ancestors in microbial organisms. With this in mind, it is safe to assume that there are no underlying ethical concerns regarding the research done.

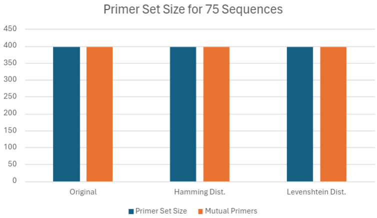

Figure 6: Size of the feasible primer set for 75 sequences.

[1] https://github.com/Dexytron/AmpliDiff

[2] https://doc.dhpc.tudelft.nl/delftblue/

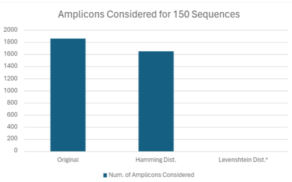

Figure 7: Number of amplicons considered for 150 sequences.

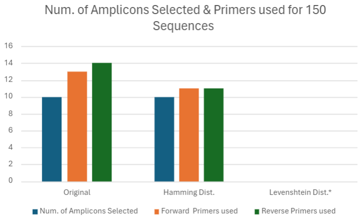

Figure 8: Number of amplicons selected, and the number of primers used in their amplification for 150 sequences.

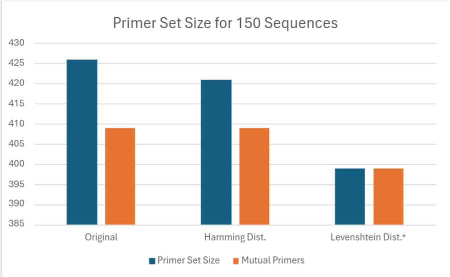

Figure 9: Size of the feasible primer set for 150 sequences.

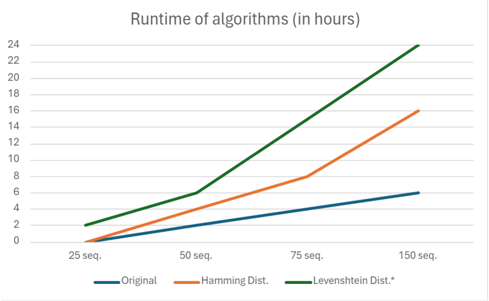

Figure 10: Representation of the algorithms runtime for 25, 50, 75, and 150 sequences.

In regards to the extent to which the research could be reproduced, all major methods and changes which are made to the original code have been outlined. Given that the original code is open source and can be found on GitHub, the ability to replicate and verify the study should be sufficiently facilitated. The specifications of hardware used during the experiments, as well as the source of the data is readily available online and is also presented in this paper. All of the changes made to the code have been listed in the Methodology section, as well as the metrics and parameters used. In addition to this, the implementation to inexact matching can be found at this GitHub Repository[3] on the main branch.

6 Discussion

Firstly, we will discuss the results of running the three different versions of the algorithm on the dataset of 150 sequences, when allowing for two mismatches and an enclosed region of 100bp.

It is important to note that running the inexact matching algorithm with the Levenshtein distance could not produce any results within the 24h of allowed run-time. The most straightforward explanation for this is that the distance metric is too computationally heavy as it relies on constructing a matrix of size $O(n^2)$ where *n* represents the length of the primer. Since this is done for all possible primers, the computation of the feasible primer set introduces a large overhead. Considering that no complete results were obtained using the Levenshtein distance, only partial results are discussed.

Figure 7 depicts the number of amplicons which were considered during the model optimization step of the greedy algorithm. Here it can be seen that there is a slight difference between running the original algorithm and the inexact matching using Hamming Distance. More so, using inexact matching in combination with the Hamming distance finds the optimal solution sooner. This is somewhat intuitive since we introduce a relaxation which should increase the number of feasible primers that can be used to amplify the target regions. Some of the amplicons which were previously rejected due to a lack of primers, could now be sufficiently amplified due to the inexact matching relaxation.

However, Figure 9 suggests that despite the decrease in the number of amplicons needed discriminate between the sequences, the size of the primer set is smaller than that of the original algorithm. One of the possible explanations for this is that the enclosed region constraint shown in Figure 3 causes certain primers to get rejected. In the original version of the algorithm, we only need to care about whether a primer occurs more than once, if so it is rejected. However, here it is more probable for a primer to occur on other locations if we allow for mismatches.

Interestingly enough, Figure 8 shows that when using inexact matching in combination with the Hamming distance, a fewer number of primers is selected. It also happens to be the case that the two versions of the algorithm select completely different amplicons.

In regards to the Levenshtein distance variation of the algorithm, Figure 9 suggests that only a fraction of the feasible

[3] https://github.com/Dexytron/AmpliDiff

primers is computed within the 24h window. These primers also happen to appear in the other two primer sets. This implies that using the Levenshtein distance is sub-optimal, as it is too time consuming without a solid guarantee that it will find a better solution compared to the other two variations.

The difference in runtime of the three versions of the algorithm can be seen in Figure 10. Given that the runtime performance of the algorithms was sampled from four different data-sets, it is hard to make any solid conclusions. However, based on this data, we can assume that the runtime overhead of both the Levenshtein and Hamming distance distance resemble an exponential function. On the contrary, the original algorithm displays a linear increase with respect to the number of sequences.

When running the three different variations of the algorithm on 75 sequences, with two mismatches and an enclosing region of 100bp, the results are identical. Figures 4, 5, and 5 illustrate that all three variations find the same solutions. In this case, the only difference is the time it took to find the optimal solution. One possible explanation for this is that the Hamming and Levenshtein distance take more time to generate the primer set, which makes up for the difference in overall time. In addition to this, it could be possible that the optimal solution is found by the original algorithm, hence allowing for mismatches does not yield any improvements.

Lastly, running the two variations of inexact matching with the enclosed region constraint set to 400bp and 1000bp, feasible solution is found. When running both variations on the set of 150 sequences, both timed out, i.e. did not find a solution within 24 hours. In the case where more than 25 sequences are used, no feasible solution is found. When analyzing the data for the sample of 25 sequences, it is shown that most primers enclose a region between 150bp and 300bp. This also coincides with the conclusions drawn from the LAMP [10] paper. Given that this constraint is not satisfied for the first 25 sequences of the dataset, it is also not satisfied for the larger dataset. This is because the 25 sequences are a subset of the larger dataset, and since if at any point there are two primer pairs which enclose a region of less than 400bp or 1000bp, the pair cannot be considered in the solution.

7 Conclusions and Future Work

7.1 Conclusion

In this paper we have introduced two different variations of inexact matching. The original algorithm relying on exact matching is introduced, and the differences between it and the Hamming and Levenshtein distance variations are outlined. The results obtained show that for 150 sequences there is indeed a difference between the amplicons selected when using Hamming distance based inexact matching, compared to exact matching. However, given that the set of amplicons selected does not intersect, we cannot reason about the difference between primers selected for the mutual amplicons. In addition to this, it is shown that using Levenshtein distance proves to be rather computationally heavy, as no results were obtained for 150 sequences.

When considering 75 sequences, all three versions of the algorithm seem to have found the optimal solution containing the same amplicons and primers. However, due to the primer-to-primer comparison when using inexact matching, it takes significantly more time to find the same solution. Lastly, we show that the additional constraint regarding the enclosed region of a primer pair proves to be overly stringent, and in cases of 400bp and 1000bp, no feasible solution is found.

7.2 Future Improvements

There are a few possible improvements which could be made, in regards to the run-time of the algorithm, the way the similarity metrics are computed, and potential changes to the experiment set up.

An example of an improvement for the run-time could be relying on a trie structures to store the set of similar primers, given that in the current implementation it takes $O(n^2)$ time do this computation, where *n* is the number of feasible primers. In addition to this, the computation of the Levenshtein distance [4] could possibly be optimized such that it does not use $O(n^2)$ space, but instead $O(n)$ space, where *n* is the length of the primer.

In regards to the similarity score, a possible improvement would be to have a weighted score, where the nucleotides between which the mismatch happens are taken into consideration. Considering the parts of the primer where the mismatch happens, such as Primer-BLAST [15] does, could be a possible improvement.

Taking the into consideration the annealing temperatures as suggested in the Primer3 [11] paper might prove to give better results, as it could have an impact on the feasibility of primers. This would most likely cause the algorithm to consider different primers, which might yield different, and perhaps better results.

Running the algorithm on a different dataset, perhaps with differently aligned sequences, or on a dataset where similar primers occur less frequently, would cause the inexact matching variation to have a better performance.

A more lenient way of enforcing the enclosed region constraint based on a better theoretical background could prove to be a potential improvement. Lastly, varying the number of mismatches allowed is something that should also be considered.

References

[1] Srinivas Ayyadevara, John J Thaden, and Robert J Shmookler Reis. Discrimination of primer 3'-nucleotide mismatch by taq dna polymerase during polymerase chain reaction. *Analytical biochemistry*, 284(1):11–18, 2000.

[2] Richard W Hamming. Error detecting and error correcting codes. *The Bell system technical journal*, 29(2):147–160, 1950.

[3] S Kwok, SY Chang, JJ Sninsky, et al. A guide to the design and use of mismatched and degenerate. *Genome Res*, 3:S39–47, 1994.

[4] Vladimir I Levenshtein et al. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710. Soviet Union, 1966.

[5] Chaim Linhart and Ron Shamir. The degenerate primer design problem: theory and applications. *Journal of Computational Biology*, 12(4):431–456, 2005.

[6] National Human Genome Research Institute. Primer. https://www.genome.gov/genetics-glossary/Primer, 2024. Accessed: 19-01-2024.

[7] Xiao Meng Pei, Martin Ho Yin Yeung, Alex Ngai Nick Wong, Hin Fung Tsang, Allen Chi Shing Yu, Aldrin Kay Yuen Yim, and Sze Chuen Cesar Wong. Targeted sequencing approach and its clinical applications for the molecular diagnosis of human diseases. *Cells*, 12(3):493, 2023.

[8] Chang-Hui Shen. *Diagnostic molecular biology*. Elsevier, 2023.

[9] Mohammad Yaseen Sofi, Afshana Shafi, and Khalid Z Masoodi. *Bioinformatics for everyone*. Academic Press, 2021.

[10] Scott W Tighe, Andrew F Hayden, Marcy L Kuentzel, Korin M Eckstrom, Jonathan Foox, Daniel L Vellone, Kristiaan H Finstad, Pheobe K Laaguiby, Jessica J Hoffman, and Sridar V Chittur. Molecular characterization of increased amplicon lengths in sars-cov-2 reverse transcription loop-mediated isothermal amplification assays. *Journal of biomolecular techniques: JBT*, 32(3):199, 2021.

[11] Andreas Untergasser, Ioana Cutcutache, Triinu Koressaar, Jian Ye, Brant C Faircloth, Maido Remm, and Steven G Rozen. Primer3—new capabilities and interfaces. *Nucleic acids research*, 40(15):e115–e115, 2012.

[12] Jasper van Bemmelen, Davida S Smyth, and Jasmijn A Baaijens. Amplidiff: An optimized amplicon sequencing approach to estimating lineage abundances in viral metagenomes. *bioRxiv*, pages 2023–07, 2023.

[13] Erwin L Van Dijk, Hélène Auger, Yan Jaszczyszyn, and Claude Thermes. Ten years of next-generation sequencing technology. *Trends in genetics*, 30(9):418–426, 2014.

[14] David L Wheeler, Tanya Barrett, Dennis A Benson, Stephen H Bryant, Kathi Canese, Vyacheslav Chetvernin, Deanna M Church, Michael DiCuccio, Ron Edgar, Scott Federhen, et al. Database resources of the national center for biotechnology information. *Nucleic acids research*, 36(suppl_1):D13–D21, 2007.

[15] Jian Ye, George Coulouris, Irena Zaretskaya, Ioana Cutcutache, Steve Rozen, and Thomas L Madden. Primer-blast: a tool to design target-specific primers for polymerase chain reaction. *BMC bioinformatics*, 13:1–11, 2012.