LU TP 20-25
June 2020

# Chromosomal DNA Barcode Assembly Using Hierarchical Clustering Matrix Method: Including Elastic Matching

**Erik Clarkson**

Computational Biology and Biological Physics
Department of Astronomy and Theoretical Physics

Bachelor thesis supervised by Tobias Ambjörnsson

LUNDS
UNIVERSITET

# Abstract

Obtaining DNA sequences is a time-consuming task, which typically requires one or several days for completion. One way of reducing analysis times is to be satisfied with long-range sequence patterns on the order of thousands of base pairs. DNA barcoding is a DNA-characterising technique that works according to this principle. It does so by using fluorescence microscopy to visualise long-range sequence patterns along DNA molecules which are fluorescently stained. The resultant light intensity curve works as an often unique identifier and is called a DNA barcode. This would be sufficient for identifying many bacteria species and would also provide a faster result compared to other candidate methods, with possible implementations in bacteriology, diagnosis and epidemiology.

When DNA is to be extracted from cells, it breaks at some points along the way, resulting in DNA fragments. This happens even with the most sophisticated methods to date. Therefore, a computational part of the assembly process is required in order to obtain an intact DNA barcode. This thesis explores the addition of stretching out the fragments in the assembly process, to see to what extent it increases the assembly quality, as compared to a previous method [Wensi Zhu, *Hierarchical clustering matrix method (HCM) applied to DNA barcode assembly for bacterial chromosomes*, Lund University, 2018]. Stretching as a parameter is motivated by the fact that confined DNA fragments in nano-channels are not equally stretched. In the assembly, we merge the fragments based on their similarity at different overlap and in a hierarchical order, always merging the best matching pair first.

Comparing stretching to non-stretching, we found that the number of merged fragments and the size of DNA that it covers increases considerably with stretching included in the assembly process. It is therefore well motivated to include stretching in further analyses of DNA barcode assembly, in the ambition of developing DNA barcoding further.

# Acknowledgements

I would like to thank Tobias Ambjörnsson for his guidance and many good explanations to my questions. It has allowed for a way of learning that has been very developing. Thanks also to Erik Torstensson and Albertas Dvirnas for providing excellent programming support and extra discussion. These advice have been much appreciated.

Furthermore, I would like to thank Fredrik Westerlund's group at Chalmers University of Technology for providing the experimental samples used in this thesis.

# Popular science summary

Imagine that the diagnosis of an ill person, implemented by identifying his/hers bacteria, could be made in a couple of hours. By identifying the bacterium that is the root of the illness, the appropriate measures could be taken directly at the hospital.

It turns out that the main features of bacterial DNA may be captured in a simple way: staining the DNA with two types of ligands (molecules binding to DNA) along the chain that re-emit light differently, can yield large-scale sequence information. In short, by photographing this chain, one obtains a unique fluorescence intensity pattern for every type of bacterium. This is simply a curve, much like the graph of a share on the stock market.

The technique described above is called DNA barcoding and has several benefits over its alternatives. These benefits include high quality data on a large scale [1] as well as its speed. If such a fluorescence intensity pattern is intact, it is enough to be able to identify a bacterium through comparison with a database. These intensity patterns therefore work as bacterial genomic ’fingerprints’.

Identifying bacterial DNA with current methods is rather cumbersome, requiring special techniques and several days of time. Opting for a DNA barcoding solution, there is an essential problem to overcome: as bacterial DNA is extracted using state-of-the-art techniques, it is unavoidably fragmented. This is a problem that requires computational methods.

So, in this B.Sc. project I have been working on implementing a computer algorithm that finds out how to best piece DNA barcode fragments together, re-obtaining an intact fingerprint. Somewhat in conformity with how a person might reason as he/she solves an old-fashioned puzzle, the computer tries out every possible option, saves the best fit and then repeats until all fragments are linked together.

And just as a puzzle that contains twice the number of pieces may take much longer than twice the time to solve, so the vastness of the DNA problem increases rapidly with the number of fragments. The iterative schedule of linking the fragments is simple nonetheless, and perfectly suitable for a computer, which finishes the job simply and effectively.

Noise from the light emission-experiments is unavoidable though, where an experiment must be done for every bacterium. The remaining challenge is thus to take some of these noise effects into account, in order to reach a good-enough consensus barcode (say, 90 percent similarity) with the ‘real one’, i.e. the database counterpart.

In a previous M.Sc. project [Wensi Zhu, *Hierarchical clustering matrix method (HCM) applied to DNA barcode assembly for bacterial chromosomes*, Lund University, 2018], Zhu considered an assembly method which assumed that the stretch of the DNA molecules were identical. In that study, the method worked well with no noise effects, but not well enough including the noise. In this study, I extended the method by Zhu to include stretching in the assembly process. We found that by this modification, both the number of merged fragments and the size of the DNA that it covers increases.

# Contents

# 1 Introduction

DNA is a macro-molecule found in all forms of known life. Its full length is orders of magnitude longer than the length of a cell, and hence DNA is found packed into compact structures inside a cell. The sequences encoded in DNA is made out four "letters" (A,T,C,G), that provide genetical instructions for every cell of an organism. Therefore, DNA is unique and can be used for e.g. bacterial identification. The interest in specifying and mapping entire bacterial and eukaryotic genomes have risen to become a current important goal of genome research in biology [4]. Today there are a number of sequencing techniques available in the rapidly growing field of DNA sequencing. One common and of high standard is the Sanger sequencing, in which one breaks the DNA sample into short fragments, each on the order of hundreds of base pairs. These fragments are then sequenced, based directly on their DNA sequence. A common disadvantage of conventional sequencing methods such as the Sanger method, is the time required to cultivate cells. The number of these cells are namely typically in the order of a few millions, requiring at least a full day to cultivate.

Optical DNA mapping is a faster alternative to DNA sequencing, which does not require cell culturing. The cells required are so few (approximately a few hundred) that a sample from a patient is sufficient. Furthermore in cases of bacteria to be identified, course-grained sequence information may be sufficient, as demonstrated in [3]. A step in this direction was taken by Schwartz et al., who invented optical DNA mapping in the 1990s[6]. They used fluorescence microscopy to identify a genome sequence, called a DNA barcode. Optical DNA mapping yields a unique intensity pattern for each DNA molecule, which can be done in different ways. Besides its efficiency, optical DNA mapping is also a relatively cheap coarse-grained sequencing technique.

In this thesis, DNA barcodes come from the competitive-binding (CB) based assay [2], which is an affinity-based mapping technique. The latter's benefits over its main alternative, 'enzymatic labeling', include less information loss through DNA fragility occurring from too close nickings as well as less time consuming sample preparation [2]. In the CB method, the stretched DNA molecule is stained with two types of dyes (ligands): netropsin and YOYO-1. The former binds sequence-specifically to AT-bonds and is non-fluorescent, the latter has the opposite attributes: it binds non-specifically and is fluorescent. Added in the correct concentrations, these two will 'compete' over the binding, revealing the local AT/GC sequence concentration [1]. In other words, a large intensity corresponds to high local GC-concentration and vice versa. A schematic of the CB method can be seen in fig. 1 below.

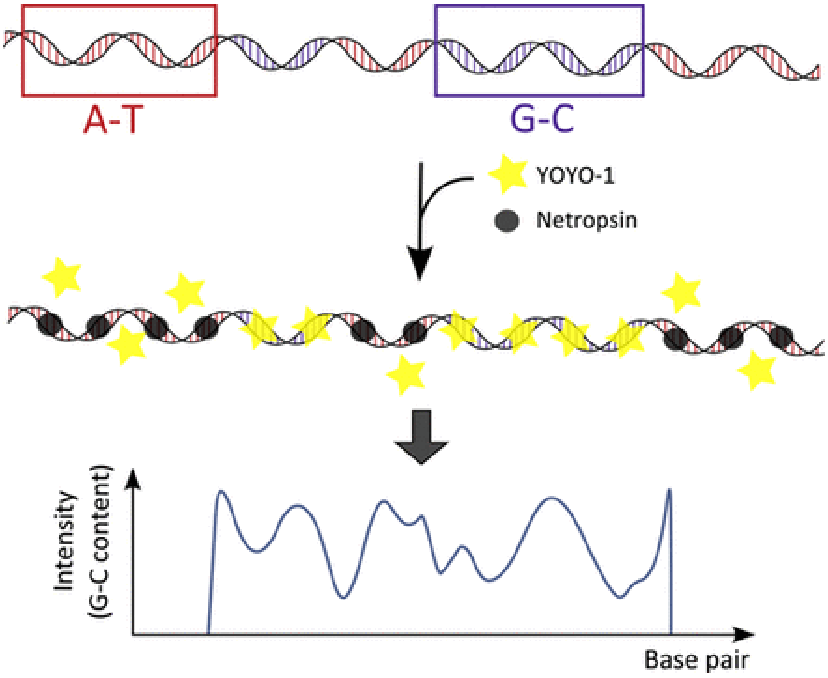


Figure 1: Schematic illustration of how the competitive binding assay works. A DNA is stained with binding molecules of netropsin and YOYO-1. Netropsin binds sequence-specifically to regions of high AT-concentration and is non-fluorescent, while YOYO-1 binds non-specifically and is fluorescent. These molecules work as markers for the two types of DNA bonds, and can be used to visualise the varying AT/GC-content. This visualisation is a curve of the GC-content along the DNA base pairs, obtained by stretching the DNA in nanofluidic channels and imaging the emission intensity [6].

Previous attempts to fully implement a method for DNA barcode assembly have been partially successful, being accurate with only noise added to theoretical data but inaccurate with real experimental data [4]. The lack of agreement between experiments and theory should be due to a few extra uncontrolled factors, such as thermal fluctuation and a varying molecule stretching over time. After all, the DNA barcode's origin is from a physical experiment containing stretched-out DNA fragments confined in nano-channels. Nevertheless, the experimental error ought to have the largest effect - the varying stretching - can be considered in the computational assembly process. Handling the effect of stretching in this way is new.

Thus by introducing stretching as a parameter in the assembly process, which is the main purpose of this thesis, the experimental barcodes could most likely be matched better and maybe even satisfactorily. In an attempt to test this, the outline of this thesis is as follows: first, the experimental procedure from which actual barcodes from a sample are obtained, is presented in section 2.1. Closing in on the theoretical work, different types of barcodes and how they are generated are introduced in section 2.2. In sections 2.3 and 2.4, a means of comparing similarities of barcodes and merging them together is explained. Then, an overview of the assembly process without stretching is given in section 2.5. Section

2.6 describes the methods involved in stretching out and compressing the barcodes. Next, section 2.7 describes how to choose appropriate thresholds. Section 2.8 explains how we mimic some characteristics of experimental fragments, when we generate emulated barcodes (see section 2.2). Section 2.9 contains a more detailed review of the assembly process, including the stretching and how 'book keeping' is kept during iterations. In section 2.10, some assembly quality measures are introduced and in section 2.11 the origin of the data that we use are given. The results begin with an example merging in section 3.1 and continue with threshold values in 3.2. Results for emulated barcodes are found in section 3.3 and continues with the experimental barcodes in section 3.4. Section 4 concludes what has been done and some implications that it yielded. It also discusses a few modifications and extensions that could bring a more accurate result.

# 2 Methods

## 2.1 Experimental procedure

The experimental barcodes in this thesis are all generated from the same type of experiment, described here. First, chromosomal DNA is extracted from cells, where it is unavoidably fragmented. Then, the chromosomal DNA is stained with netropsin and YOYO, described earlier. Using nitrogen gas, they are pressured into small chips, containing both microfluidic and nanofluidic channels. The chip is designed so that the DNA easily fits in the micro-channel but is stretched out considerably into extended linear form in the nano-channel. This design is meant to stretch the molecule as much as possible, without accidentally clogging the device due to molecules getting stuck in the nanochannels. Such an event would render the chip useless. With a larger stretch, fewer base pairs are contained in each pixel when the fragment is imaged, which implies a higher resolution [1].

Using a fluorescence microscope, many pictures are taken at different times, outputting a kymograph. When this kymograph has been aligned and time-averaged, one has a so called experimental barcode. A schematic figure of the complete experimental assay can be found in e.g. [3], fig. 1. See fig. 2 below for an illustration on how the raw kymograph relates to the experimental barcode. The experimental barcode depicts how light intensity (in arbitrary units) varies along the molecule length, measured in pixels. The last two procedural steps, that concern an analysis and comparison of barcodes, are the theoretical steps hence considered in this thesis.

## 2.2 Barcode types

There are three distinct types of barcodes in this thesis - the experimental, the theoretical and the emulated experimental barcodes (theory barcodes with 'realistic noise'). The experimental one is the one obtained from the actual experiment, and includes considerable amounts of 'defects', including noise and a stretching degree varying with time [4]. Two notable sources of the noise are random thermal fluctuations of the molecules of the frag-

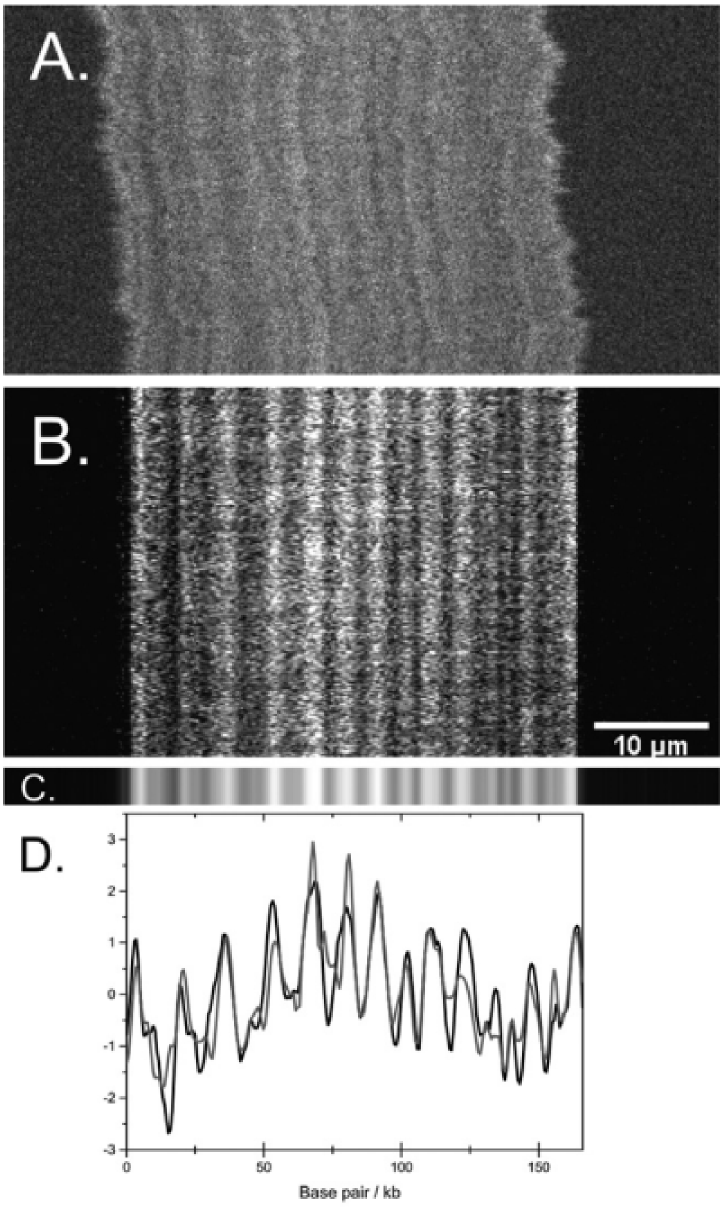


Figure 2: Illustration of how the kymograph obtained from an experiment relates to the barcode. (A) A raw experimental kymograph, obtained from imaging the fluorescent light emission from YOYO-1 of stretched DNA in nano-channels. It consists of several images at different times, with the time scale on the vertical axis. (B) The aligned kymograph. (C) An experimental DNA barcode obtained by time-averaging the aligned kymograph in (B). (D) Comparison of the experimental barcode (black) with a theoretical barcode (grey) [6].

ments and the fact that the number of photon counts is limited ('shot noise'). In a real experiment, where the goal is to obtain a barcode covering the full genome without any prior knowledge, these barcodes are the only ones available. However, in order to develop and test the assembly, there are two more types of importance.

A theoretical barcode is derived from knowledge of the base-pair sequence of the relevant bacterium. The procedure for turning the DNA sequence into a theoretical barcode is based on probabilistic statistical physics calculations, including the binding properties of the ligands [6]. From this, one obtains a theoretical barcode with base-pair resolution. This is converted to pixel resolution with an experimentally determined conversion factor. Moreover, the fluorescence microscope has an inherent limitation - its point spread function. To mimic this, the barcode is convolved with a gaussian kernel of the right standard deviation [4].

The emulated experimental barcode is a mixture of the theoretical one and a random one. To construct the random part, one draws gaussian random numbers with standard deviation equal to 1 and zero mean to the sought length. Then, one convolves it with a gaussian kernel, to imitate the blurriness caused by the point spread function. Specifically, the emulated barcode is a weighted average of the random and theoretical one, to be as similar to the theoretical one as the experimental one is. The weighted averaging, which was done similarly in [4], is

$$\mathbf{B}_{\text{emulated}} = \alpha \mathbf{B}_{\text{randomised}} + (1 - \alpha)\mathbf{B}_{\text{theory}}, \tag{2.1}$$

where $\mathbf{B}$ denotes the barcode indicated. Mathematically, $\mathbf{B}$ is a vector with components $\mathrm{B}_i$, where $i$ is the $i$th pixel of the barcode. Details about how we choose $\alpha$ is explained in the next paragraph. An emulated barcode thus mimics an experimental barcode.

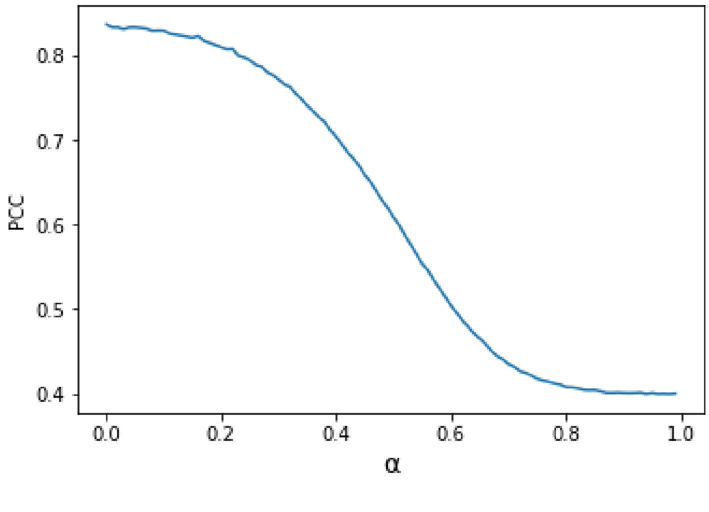


Figure 3: Highest PCC compared to theory for randomly generated emulated barcodes vary with $\alpha$.

For information on how to find a suitable weighing, i.e. $\alpha$ in eq. 2.1, consider fig. 3 above. It shows the emulated barcode's dependence on its random part, as described in section 1. We see that the Pearson correlation coefficient (PCC) is strictly decreasing until $\alpha$ equals 1, i.e. when the emulated barcode is actually a fully randomised barcode. In short, PCC=1 indicates a perfect score and PCC=0 should be the result of two fully randomised barcodes. PCC is explained in more detail in the next section. Typical PCC for experimental barcodes are ca. 0.7, and we choose $\alpha = 0.44$ which corresponds to a PCC of ca. 0.7. Note that the PCC here does not attain a value $= 0$ as $\alpha$ approaches 1. This is due to the fact that the best PCC score, henceforth denoted by $\hat{C}$, against theory is always picked.

## 2.3 Barcode comparison

In the method described in the next paragraph, barcodes are compared with each other in order to obtain information about how well they match and what positions that correspond to. Because all the barcodes in this thesis can be considered as values at equally spaced points along a line, a comparison specifically means a comparison of the values that are at the same positions. This is illustrated if fig. 4 below, featuring two barcodes with intensity values $B_i$ and $A_i$ with corresponding weights $W_i$ and $V_i$. That is, every barcode has an associated weight barcode, used in the merging as described later in this section. The shorter barcode always slides along the longer one, to test the sought positions.

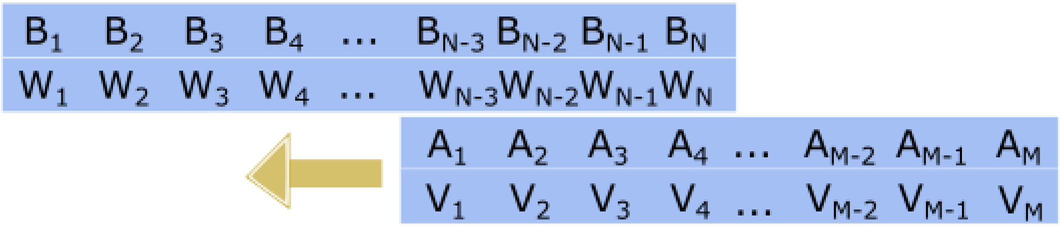

B1 B2 B3 B4 ... BN-3 BN-2 BN-1 BN
W1 W2 W3 W4 ... WN-3 WN-2 WN-1 WN
A1 A2 A3 A4 ... AM-2 AM-1 AM
V1 V2 V3 V4 ... VM-2 VM-1 VM

Figure 4: Finding the best position of two barcodes **B** and **A** by comparison at different overlaps. **W** and **V** are their corresponding weight barcodes. Then number of components $N > M$, whence the lower barcode slides along the longer, upper one. The Pearson correlation coefficient is used as a 'comparison score', comparing the intensity values $B_i$ and $A_i$.

The measure used compares only the overlapping intensity values. This measure is called a Pearson correlation coefficient (PCC) match score and evaluates the linear correlation of two curves. For the comparison of two barcodes **A** and **B**, its definition is

$$\mathrm{PCC}_{A,B} = \frac{\sum_{i=1}^{n}(A_i - \bar{A})(B_i - \bar{B})}{(\sum_{i=1}^{n}(A_i - \bar{A})^2)(\sum_{i=1}^{n}(B_i - \bar{B})^2)^{1/2}}, \tag{2.2}$$

where $i$ labels the pixel positions to be compared (i.e. the overlapping part), $A_i$ and $B_i$ are the components (intensity values) at pixel position $i$, $n$ is the number of overlapping values to compare at certain position and $\bar{A}$ and $\bar{B}$ are the mean intensities of the overlapping values of each barcode. A PCC value of 0 means no linear correlation at all, 1 means perfect linear correlation and -1 means perfect linear anti-correlation.

We are typically interested in the maximum PCC, denoted by $\hat{C}$, which indicates the best merging. The relative position at which this happens is also of interest. As the shorter fragment always slides along the longer one, we need only the start position of the shorter one in terms of the pixel index of the longer one to fix this relative position.

## 2.4 Barcode merging

The exact merging of two barcodes is dependent on how well they fit together, and follow this general rule: outside parts extending past the other are kept as they are. The overlapping 'middle' part is converted into a weighted average value based on the weights at those positions, see fig. 5. We use thresholds for minimum overlap and for minimum $\hat{C}$, are determined in section 3.2. Fragments that are below a threshold are simply excluded from the merging.

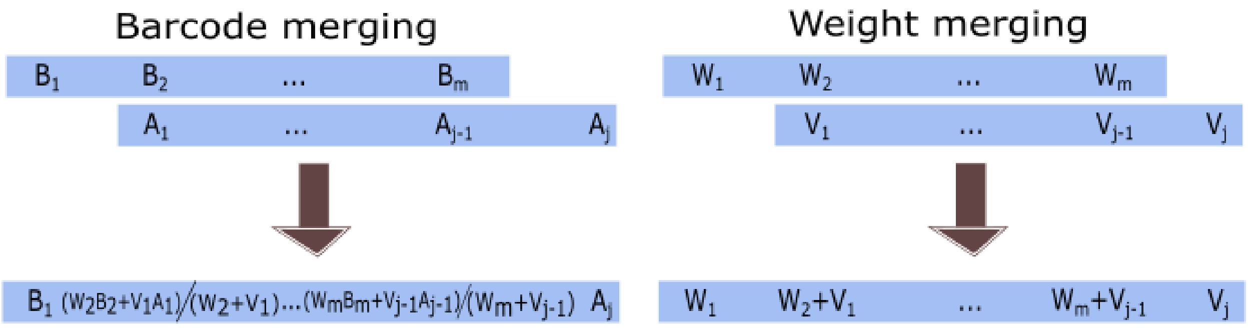
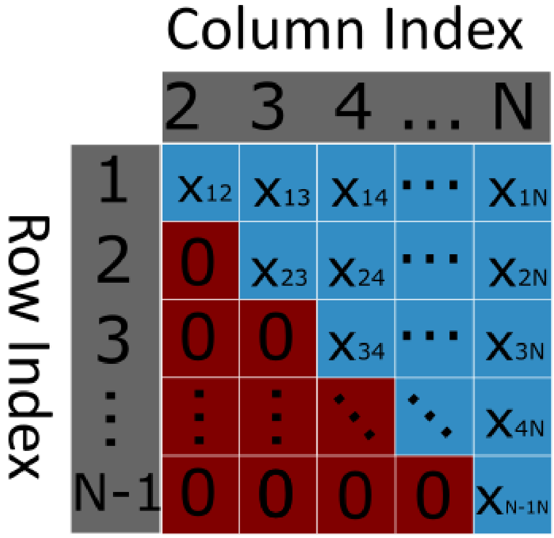


Figure 5: The Merging of two barcodes and their associated weights. The weighted average of the barcode's overlapping components become the new components at those positions, while non-overlapping values stay the same. The same rules for non-overlapping parts hold for the weight barcode merging, while overlapping values are summed.

The origin of including this weight is that more than two barcodes may merge with the same overlapping region. In this case, the weight assures that each barcode contributes equally much. That is, all barcodes have the same weight at first, but for each merging, the overlapping parts' weight is increased to represent the number of mergings.

## 2.5 Overall barcode assembly

The barcode fragments, i.e. sequences of intensity values, are in the situation given without any information regarding where they fit into the intact genome. A method which utilises PCC comparison scores to find the best fit and books every position is required in order to obtain an intact barcode. The overall strategy used is 'hierarchical clustering', which amounts to always merge the pair that fits best first, calculate new values for the merged pair and repeat until there are no fragments left, at least above some threshold values. The assembly includes a threshold value for overlap length and $\hat{C}$, so that only fragments that pass these are used. The method chosen for this task is 'hierarchical clustering matrix method' and is described below.



|  | 2 | 3 | 4 | ... | N |
|---|---|---|---|---|---|
| 1 | $X_{12}$ | $X_{13}$ | $X_{14}$ | ⋯ | $X_{1N}$ |
| 2 | 0 | $X_{23}$ | $X_{24}$ | ⋯ | $X_{2N}$ |
| 3 | 0 | 0 | $X_{34}$ | ⋯ | $X_{3N}$ |
| ⋮ | ⋮ | ⋮ | ⋱ | ⋱ | $X_{4N}$ |
| N-1 | 0 | 0 | 0 | 0 | $X_{N-1N}$ |

Figure 6: The form of the matrices used for the bookkeeping of relevant values (PCC, overlap, fragment identity) in the assembly process. $x_{ij}$ denotes one type of such relevant values obtained for fragment $i$ and $j$. The zeroes below the main diagonal are unused. N denotes the number of fragments to be assembled at the current stage, which gives an $(N-1)$x$(N-1)$ matrix to include all combinations.

Three $(N-1)$x$(N-1)$ matrices (called 'PCC matrix', 'overlap matrix' and 'index matrix') are constructed, where $N$ is the number of barcode fragments. The three matrices are to store values for $\hat{C}$, the overlap position (as in described in the end of section 2.3) and what fragments are compared, respectively. Later on in section 2.9 we add a fourth matrix, 'stretching matrix', which contains the optimal stretch for each comparison. They all follow the same iterative schedule and all three values could in principle have been placed within one matrix. It is because all possible combinations of two fragments are stored, that the matrix shape is $(N-1)$x$(N-1)$, so that each fragment is coupled to every other. The matrix layout suits the booking in a practical way, as every entry contains the specified value of the two fragments that corresponds to that element, see fig 6. The rows start counting at 1, and the columns start at 2. This reduces the lower triangular part to no elements (or zeroes), as the order of the fragments in the pairs is irrelevant.

The overall iterative schedule, which follows the steps described below, is followed by every type of matrix. An overview can be seen in fig. 7.

1. The three matrices are initialised based on the number of fragments to be assembled, setting their shape. For $N$ fragments, the shape of the matrices is $(N-1)$x$(N-1)$, as in fig. 6. This step is only done once.

2. All the entries (denoted $x_{ij}$ in fig. 6) of the upper triangular part and main diagonal are computed, so that every possible combination of fragment pairs are covered.

3. The two fragments that have the best fit, i.e. display the highest PCC score, are identified and merged. See panels c) and d) in fig. 7.

4. The resulting barcode is stored.

So in step 3, the PCC matrix is special as it can point out what entry corresponds to this pair. The index matrix keeps account of the current barcodes by keeping a map from a matrix position to the actual barcodes.

After step 4 we discard some of the old values, i.e. reduce the matrices, and update the other entries with values of the newly merged fragment. By the numbering of rows and column, the higher parent index value (of the two merged fragments) will always correspond to a column, whence the column and row of the upper parent index is deleted. Finally, the rows and columns of the lower parent index are filled in anew. This corresponds to panel e) in fig. 7. During these deletions, the matrix indices change, which calls upon a means of knowing which fragments are coupled in each entry. This is the purpose of the matrix containing the indices of the relevant fragments.

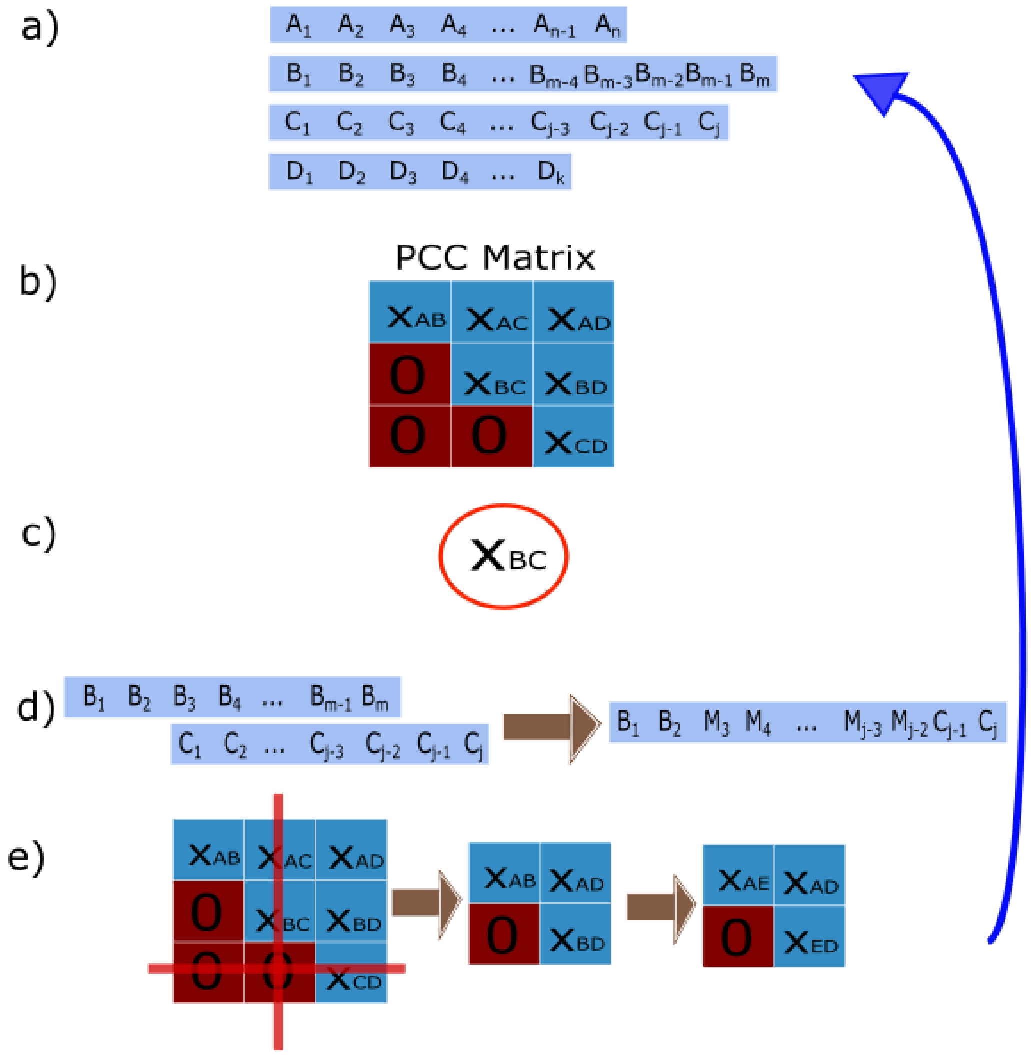

a)
$A_1$ $A_2$ $A_3$ $A_4$ … $A_{n-1}$ $A_n$
$B_1$ $B_2$ $B_3$ $B_4$ … $B_{m-4}$ $B_{m-3}$ $B_{m-2}$ $B_{m-1}$ $B_m$
$C_1$ $C_2$ $C_3$ $C_4$ … $C_{j-3}$ $C_{j-2}$ $C_{j-1}$ $C_j$
$D_1$ $D_2$ $D_3$ $D_4$ … $D_k$
b)
PCC Matrix
$X_{AB}$ $X_{AC}$ $X_{AD}$
0 $X_{BC}$ $X_{BD}$
0 0 $X_{CD}$
c)
$X_{BC}$
d)
$B_1$ $B_2$ $B_3$ $B_4$ … $B_{m-1}$ $B_m$
$C_1$ $C_2$ … $C_{j-3}$ $C_{j-2}$ $C_{j-1}$ $C_j$
$B_1$ $B_2$ $M_3$ $M_4$ … $M_{j-3}$ $M_{j-2}$ $C_{j-1}$ $C_j$
e)
$X_{AB}$ $X_{AC}$ $X_{AD}$
0 $X_{BC}$ $X_{BD}$
0 0 $X_{CD}$
$X_{AB}$ $X_{AD}$
0 $X_{BD}$
$X_{AE}$ $X_{AD}$
0 $X_{ED}$

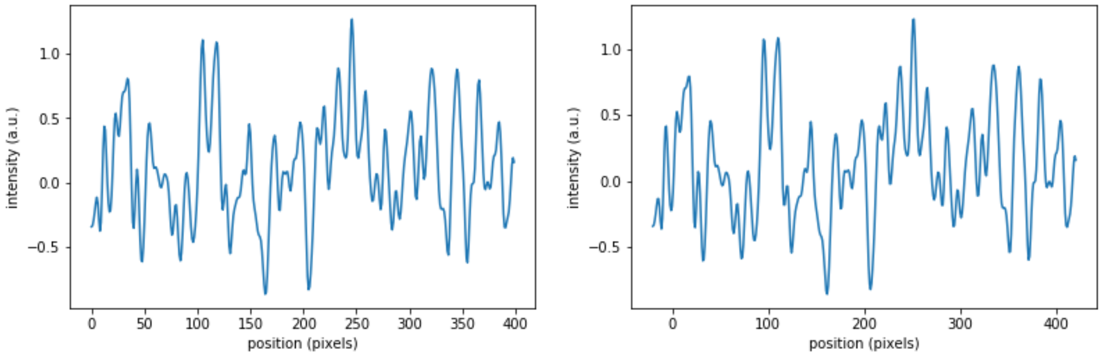
Figure 7: An overview of the assembly process. In panel a), there are four barcodes in the pool to be merged. b): a PCC Matrix containing their best match values is initialised. $x$-indices, e.g. BD, refer to a comparison of barcodes B&D. c): the best score (of barcodes B&C) is identified. d): those barcodes are merged into a new barcode, at their best match position. e): the PCC matrix is updated by removing the matches corresponding to the upper parent index (C) and then updated at the lower parent index (B) with the newly merged barcode (E). After step e), the next iteration starts over from a) again.

Without any threshold values for overlap and PCC, there will in the end be only one element left in each of the matrices, and all fragments will be merged together. See section 3.2 for determination of threshold values. With these threshold values, the last fragments may be discarded entirely, as they do not fit well enough. A common result is one which consists of several incomplete barcodes (compare to 'contigs' in DNA sequencing), which in turn have not been merged. In this thesis, they are called 'bartigs', in analogy to 'contigs'.

## 2.6 Barcode elastic stretching

The methods described so far in sections 2.1 to 2.5 are mainly the same as what was previously done in [4]. From this outset, we include a new parameter in the assembly: barcode stretching and compression. That is, in every barcode comparison, we now allow for an elasticity, which was not included in [4]. From here on, we call this stretching or compression for 'elastic stretching'. This addition of elastic stretching has its physical origin from the varying degree of fragment stretching inside the nano channels. Due to thermal fluctuations, the number of base-pairs per pixel will vary along the DNA. However, rather than directly applying more advanced and computationally heavy techniques for this specific purpose, we elastically stretch our fragments uniformly in their entirety. This simplification corresponds to the idealisation that the number of base-pairs per pixel should be approximately equal along the DNA. Had our results not been satisfactory, we would need to reconsider this point. See fig. 8 for an example of our elastic stretching.

Linear interpolation is used to for the elastic matching of two barcodes that are compared. Given a stretch of, say +5%, the number of pixels on which to evaluate the barcode are increased by the same amount. Holding the outermost points fixed, linear interpolation from the original values is used to approximate the evaluation on the new data points. As an example of this applied to emulated barcodes, see fig. 8. The barcode in fig. 8b is that of fig. 8a with an added 10% stretch. While the overall shape is largely preserved, note that fig. 8b extends farther out on each side.



(a) A purely randomised barcode with no stretching or compression applied to it.

(b) The same barcode as in 8a, but with 10% stretching.

Figure 8: A visualisation of the elastic stretching by comparison.

In the assembly process, the elastic stretching is free to take a value ranging from zero percent elastic stretching to the upper elastic stretching limit, also given in percent. The elastic stretching which produces the best fit between the fragments or contigs is always used. Hence, for every pair of fragments, the comparison as described in section 2.3 is done once for each elastic stretching-degree. The elastic stretching is applied to one fragment at a time, to handle the experimental situation where one is more stretched out than the other.

With the use of elastic stretching, the potential problem of stretching/compressing the same fragment many times emerges. This could quickly lead to an unreasonably large total elastic stretching of that fragment. The problem is handled by elastically stretching each newly merged barcode to offset the net elastic stretching previously applied to one of its constituent fragments. Specifically, we apply the stretch factor

$$s_{\text{final}} = \frac{L_a + L_b}{s_{\text{opt}} L_a + L_b}, \tag{2.3}$$

where lengths $L_a$ and $L_b$ are the lengths of the inbound fragments and $s_{\text{opt}}$ is the stretch factor with which fragment $a$ was stretched. The formula is analogous in the case were instead fragment $b$ was stretched.

## 2.7 Determination of threshold values

As stated in section 2.2 the emulated barcode mimics the experimental one. The purpose of an emulated barcodes is to provide a 'testing ground' having more available information than what is available with experimental fragments. Specifically, this information is the knowledge of where the fragments fit into the theory, and allows for a systematic testing of threshold values. An example can be seen in e.g. fig. 10 in the results section. The overlap threshold is the minimum overlap considered when comparing two fragments in the assembly process. PCC threshold is the lowest PCC-value for which a merging in the assembly is done. Note that in [4], instead of using these thresholds directly, a probabilistic function was introduced. This function worked as a 'threshold function' and related overlapped length to PCC, which gave a probabilistic argument for discarding useless fragments.

## 2.8 Generation of emulated barcodes with varying stretch factors

Both when determining threshold values and when assessing the final quality of an assembly, it is desirable to have the pool of fragments that go into the assembly mimic that of an experiment. To do this for one fragment at a time, we pick a fragment length

$$l = \mu_{\text{exp}} + \sigma_{\text{exp}} R_1, \tag{2.4}$$

where $\mu_{\text{exp}}$ is the mean experimental fragment length, $\sigma_{\text{exp}}$ is the standard deviation of the former and $R_1$ is a normal-distributed random number of zero mean and unit standard deviation. For the specific case in the results, we calculated $\mu_{\text{exp}} = 258$ pixels and $\sigma_{\text{exp}} = 5$ pixels, rounded to a whole number of pixels. There are typically around 650 base-pairs per pixel. So far, the statistical variations of the fragments have been accounted for. Next, we simulate nano-channel stretching variations by applying an elastic stretch factor (channel stretching factor)

$$s_{\text{random}} = 1 + \sigma_{\text{stretch}} R_2, \tag{2.5}$$

where $\sigma_{\text{stretch}} = 0.05$ is a typical, experimentally determined standard deviation and $R_2$ is a random number like the one in eq. 2.4 above. Now the fragment pool as a whole mimics that of a chosen experimental one.

## 2.9 Step-by-step description of our barcode assembly method

For completeness, a more detailed account of all steps in our barcode assembly method is now provided. After initialisation of matrices (PCC, Overlap, Index and elastic stretch factor, where the last contains the stretch factors associated with every $\hat{C}$), i.e. step 1 in section 2.5, the steps are

1. For each entry, compute $\hat{C}$ and the associated relative barcode position for every combination of elastic stretching of the two fragments. Of all these combinations, store the values corresponding to $\hat{C}$ in every entry.

2. Identify the two barcodes that display the highest $\hat{C}$ of the PCC matrix. This includes two steps - first, scanning the PCC matrix for the matrix position corresponding to the highest value and secondly obtaining the values of the Index matrix at the same matrix position.

3. Elastically stretch one of the fragments and its weight, using the elastic stretching factor and information of which of the two fragments to be elastically stretched. This information is given by the Stretch matrix, at the same matrix position as in the previous step.

4. Merge the elastically stretched fragment and its weight with its match and its weight. Also compute the new elastic stretching factor given by eq. 2.3

5. Elastically stretch the merged barcode with the elastic stretching factor calculated in the previous step.

6. Store the merged barcode, its weight, and which original barcodes that went into it.

7. Update the collection of constituent fragments of all current merged barcodes in the assembly, if necessary.

8. Remove the row and column of the upper parent index, for all four matrices, see the first arrow in panel e) in fig. 7.

9. Update the row and column of the lower parent index for all four matrices. The computation done in each entry is the same as in step 1. This is illustrated in fig. 7, panel e) second arrow.

Steps 2 onwards are then repeated until there are no more barcodes left that are above a $\hat{C}$ threshold. Then, the ones below the threshold are considered not to fit together and so are simply left out. Provided that at least one match was above the threshold, the method outputs one or more bartigs (i.e. fully or partially assembled barcodes).

### 2.10 Quantifying the accuracy of the assembly method

In order to evaluate how successful/useful an assembly is, we introduce a few quantities related to the assembly quality below.

1. Merged rate $= \frac{\text{no. of merged fragments}}{\text{no. of fragments in total}}$
2. positional accuracy $= \frac{\text{no. of correctly placed fragments}}{\text{no. of merged fragments}}$
3. no. of iterations - the number of times the assembly was repeated with different fragments of the same characteristics.
4. final PCC - the PCC value compared to the theory of the largest bartig.
5. coverage - the length of the largest bartig

### 2.11 Data sets

The training data set comes from an E-coli bacterium, and was obtained similarly as in [5] and in [4].

The testing data set comes from a BAC (bacterial artificial chromosome) of name RP11-614C8. It corresponds to a region around the MLLT3 gene on the ninth chromosome (exactly chr9 :20511000-20679972) and was purchased from BACPAC Genomics. Isolation of plasmids was done using NucleoBond Xtra midi kit from Macherey-Nagel, whereafter circular plasmids were linearised using PI-SceI restriction enzymes. Lastly, the staining with ligands and nano-channel imaging was done as described in section 2.1. All experimental data used in this thesis was provided to us by Fredrik Westerlund's group at Chalmers University of Technology, Sweden.

## 3 Results and discussion

To begin with, we visualise how the result of a barcode assembly compares to the fragments (see section 3.1). We then move onto the determination of overlap values (see section 3.2) with and without elastic stretching. In section 3.3 we assess the assembly for two specific sets of emulated barcodes and in sec. 3.4 we assess the assembly for a set of experimental barcodes.

### 3.1 Merging example and definitions

Before turning to actual experimental barcodes, we first consider assemblies of emulated experimental DNA barcodes. These give guidance in choosing threshold values, as we know what part of the theory they consist of. That is, we can test how successful the different combinations of threshold values are. As an example on how a final merging can be visualised, consider three emulated barcodes and the result of their merging. Fig. 9

shows such an example, with the fragments having large overlapping regions with little noise ($\alpha = 0.2$), that fit well together. We see that 'fragment 1' fits in the beginning, 'fragment 2' at about 200 px, and 'fragment 3' at about 350 px in the merged barcode.

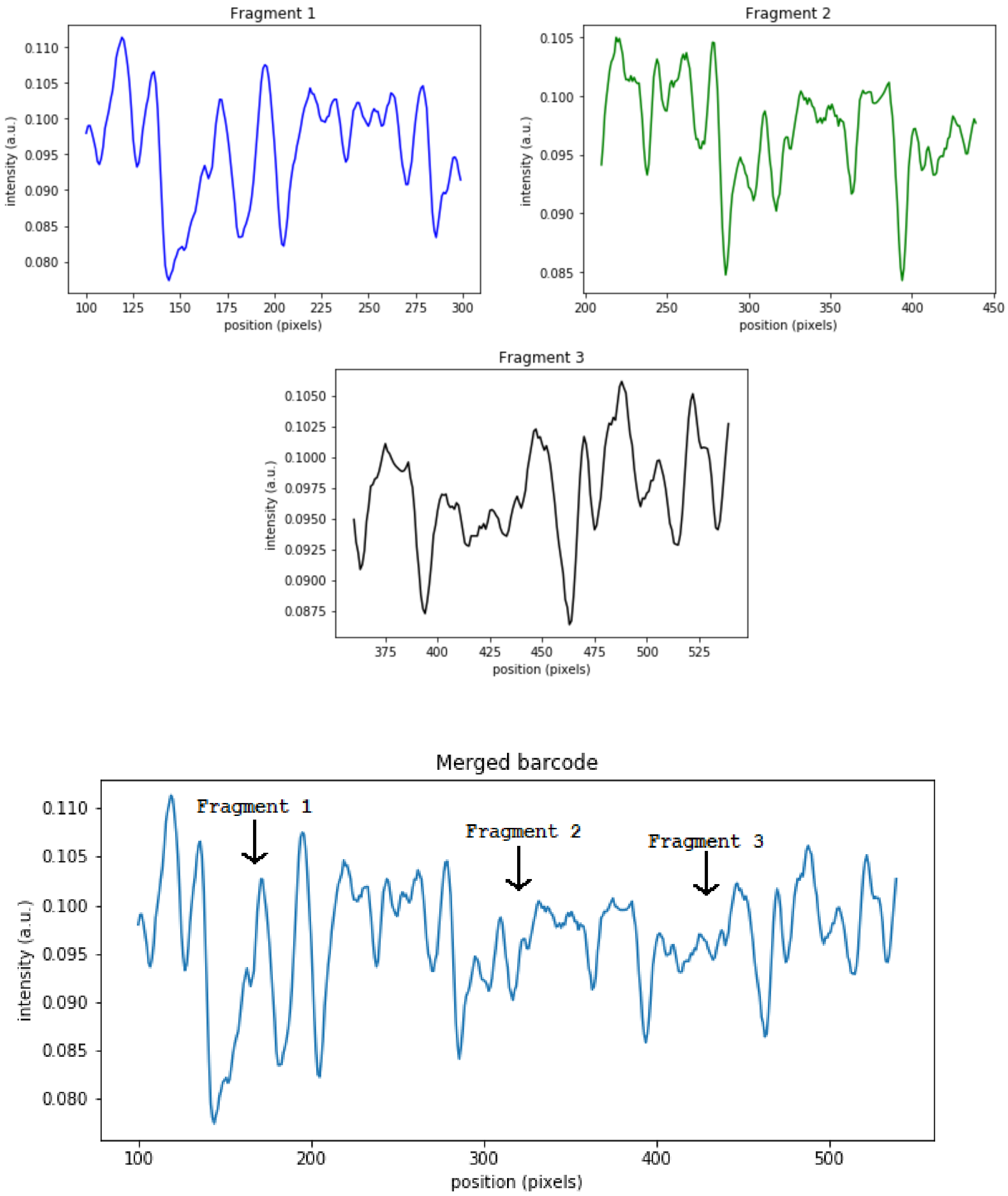


Figure 9: Three emulated fragments with little noise, $\alpha = 0.2$, and the merged result of an assembly. All barcodes' positions are given with respect to the theory they were generated from, which is several thousands of pixels long.

## 3.2 Determining threshold parameters

In order to find the optimal overlap threshold and PCC threshold to use, we generate heat maps. These show the success of many different combinations of threshold values, allowing for a sensible choice of threshold values. The emulated barcodes that go into these are based upon a 'training data' DNA sequence, similar but separated from the DNA sequence used later for the final results. A small interval of 6 pixels around the correct pixel are counted as correct as well. This is done because a fragment's position within a bartig is found from their best match after the merging process, which is not perfectly exact.

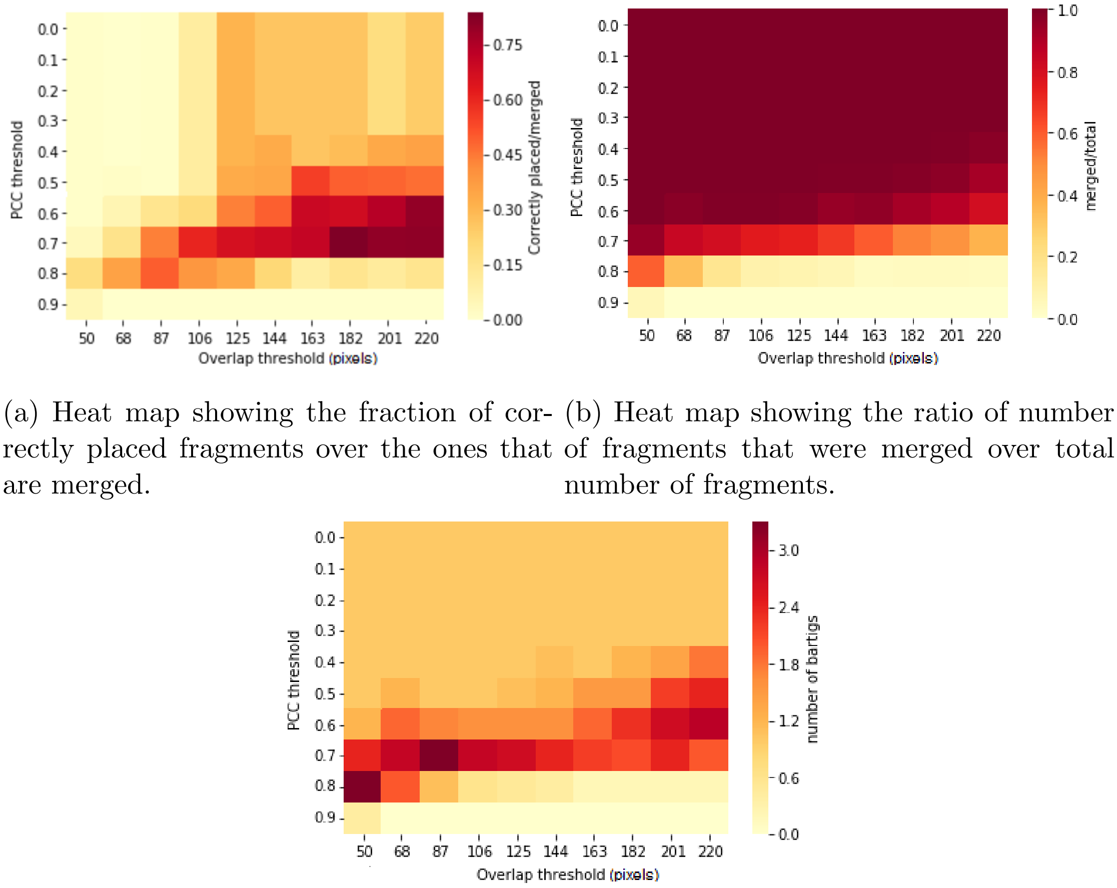
Consider the three heat maps in fig. 10, which were generated including elastic stretching in the assembly. Fourteen fragments were given in the assembly process, which was repeated ten times. Each time, the fragments displayed PCC=0.7 on average when compared to the theory, had an average length of 258 pixels with a 5 pixel standard deviation, were stretched with a channel stretching factor of 0.05 and was assembled with a maximum assembly elastic stretch factor of 1.10.

We see in fig. 10a that there is a dark region for an overlap threshold greater than 106 pixels and PCC overlap threshold greater than 0.5 but smaller than 0.8. Clearly, with too low threshold values, many more fragments are merged (see fig. 10b) but as there is no requirement on the mergings, they are merged incorrectly and therefore incorrectly placed. With PCC thresholds too high, almost no fragments end up merged, and with an overlap threshold of 220 pixels, the number of merged fragments decrease for a PCC threshold greater than 0.3.

Corresponding heat maps were also generated for fragments with the same characteristics, but without elastic stretching in the assembly process, see fig. 11. We see the same overall patterns as in the corresponding ones in fig. 10. Comparing fig. 10b with fig. 11b, there are more merged fragments with assembly stretching than without. This fact can be explained by the more matching possibilities available with stretching than without, increasing the probability of a match. Another difference is that there are somewhat fewer bartigs in fig. 10c than in fig. 11c, which should be due to an increasing tendency of merging bartigs with assembly stretching.

There is no single well-motivated choice of a threshold pair of overlap and PCC, as there is clearly a trade-off between ratios of correctly placed to merged and of merged to total. Nevertheless, by considering the features of the pool of fragments to begin with, one can prioritise these suitably. For a pool of fragments such as those involved in the heat maps below, the rather low number of fragments involved and somewhat low mean length implies that we should prioritise the number of merged fragments to the total number of fragments. This is to make sure that we cover a large enough part of the DNA, so that there is a reasonable identification possibility. With these prioritisations in mind, we pick point (overlap, PCC) = $(201, 0.6)$ in fig. 10, and point (overlap, PCC) = $(182, 0.6)$ in fig. 11, to be used for all later results.

## Heat maps with assembly stretching



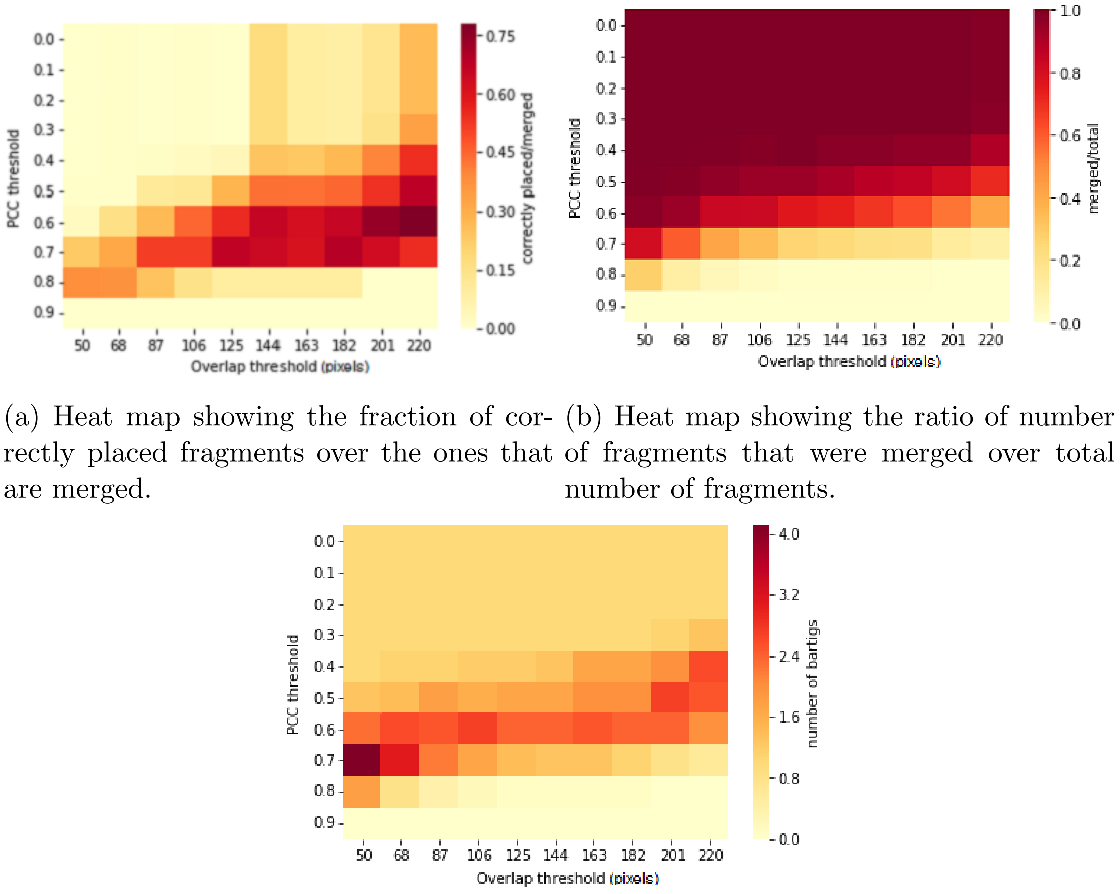
(a) Heat map showing the fraction of correctly placed fragments over the ones that are merged.

(b) Heat map showing the ratio of number of fragments that were merged over total number of fragments.

(c) Heat map showing the number of bartigs.

Figure 10: Three heat maps depicting how successful the assembly was for 100 different combinations of threshold values, for 10 runs. A small interval of 6 pixels around the correct pixel are counted as correct as well. This is done because a fragment's position within a bartig is found from their best match after the merging process, which is not perfectly exact. Overlap threshold is the minimum overlap considered when comparing two fragments in the assembly. PCC threshold is the lowest PCC-value for which a merging in the assembly is done. 14 emulated fragments were in the pool, with the characteristics of average PCC with theory = 0.7, average length=258 pixels with a 5 pixel standard deviation and channel stretching factor = 0.05. Assembly stretching factor = 1.10.

## Heat maps without assembly stretching



(a) Heat map showing the fraction of correctly placed fragments over the ones that are merged.

(b) Heat map showing the ratio of number of fragments that were merged over total number of fragments.

(c) Heat map showing the number of bartigs.

Figure 11: Three heat maps depicting how successful the assembly was for 100 different combinations of threshold values, for 10 runs. A small interval of 6 pixels around the correct pixel are counted as correct as well. This is done because a fragment's position within a bartig is found from their best match after the merging process, which is not perfectly exact. Overlap threshold is the minimum overlap considered when comparing two fragments in the assembly. PCC threshold is the lowest PCC-value for which a merging in the assembly is done. 14 emulated fragments were in the pool, with the characteristics of average PCC with theory = 0.7, average length=258 pixels with a 5 pixel standard deviation and channel stretching factor = 0.05. No assembly stretching included.

## 3.3 Emulated barcodes

In this subsection, the results of emulated barcodes matched to the theory they were generated from is examined. Again, an interval of 6 pixels the correct position in the theory is counted as correct and we use the threshold values obtained in section 3.2.

First, we try our HCM method on emulated barcodes with no channel stretching applied. This amounts to setting $\sigma_{\text{stretch}} = 0$ in eq. 2.5. In this simplified case, clearly zero assembly stretching is most appropriate, which gives a chance of testing the basic assembly method. For a data set of 14 emulated fragments with the characteristics of average PCC with theory = 0.7, average length=258 px with a standard deviation of 5 px, channel stretching factor = 0, we obtained results summarised below.

Table 1: Average scores of emulated data without assembly stretching. The data consists of 14 emulated fragments in the pool, with the characteristics of average PCC with theory = 0.7, average length=258 pixels with a 5 pixel standard deviation and channel stretching factor of 0. Minimum overlap was 182 pixels and minimum PCC was 0.6.

Coverage is the pixel size of the largest bartig and positional accuracy is the ratio of the number of correctly placed fragments to the total number of fragments.

| no. of iterations | coverage | no. of bartigs | merged rate | positional accuracy | final PCC |
|---|---|---|---|---|---|
| 50 | 492 | 2.2 | 91% | 97% | 0.89 |

**Visualisation of bartigs and theory**

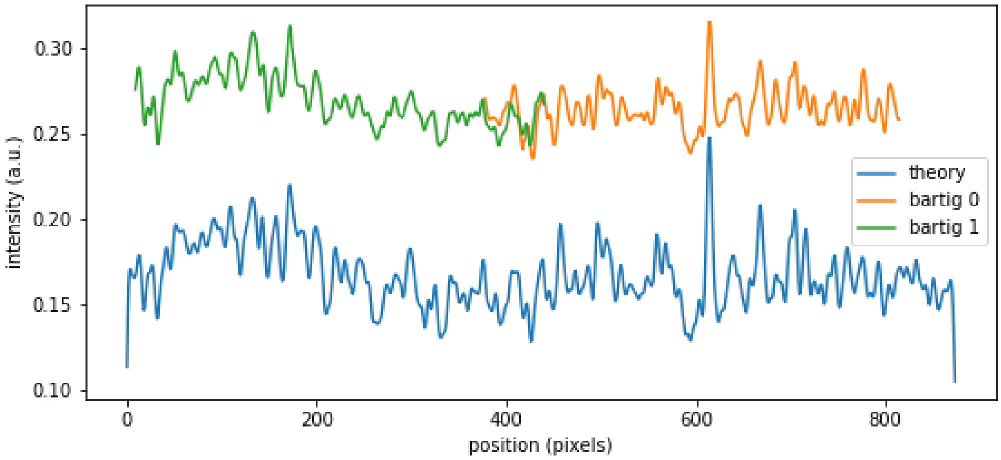


Figure 12: A typical result without assembly- nor channel-stretching of fragments with characteristics of average PCC with theory = 0.7, average length=258 pixels with a 5 pixel standard deviation and channel stretching factor of 0. Minimum overlap was 182 pixels and minimum was PCC 0.6.

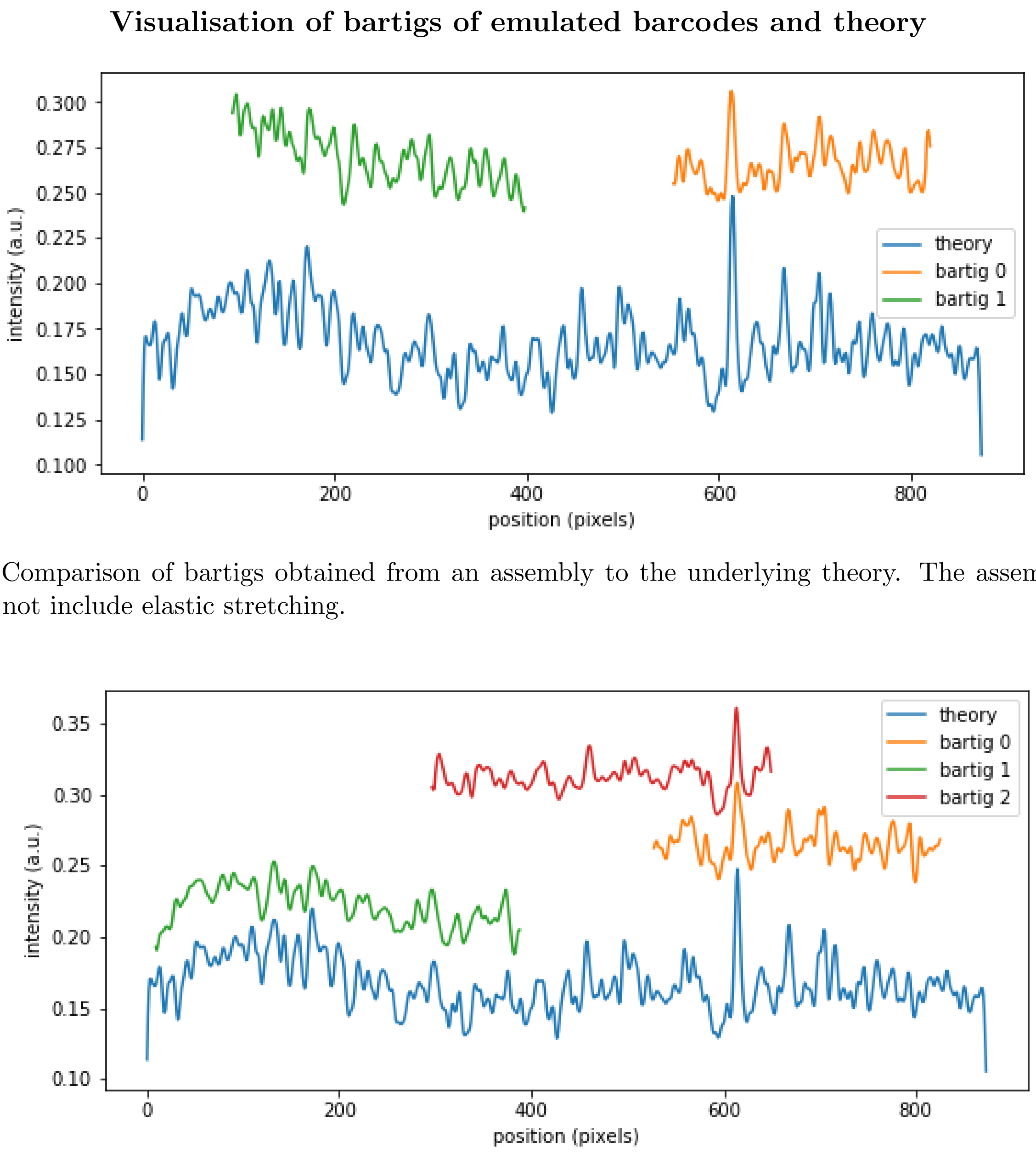
We see in table 1 that the method works well in this case of no simulated channel stretching, which agrees with [4]. A visualisation of the result is seen in fig. 12, featuring two bartigs and their place along the theory. The two bartigs match the theory well, and would possibly have been pieced together into a single bartig, had their overlap reached the threshold of 182 pixels.

We now move on to fragments with equal characteristics as above, but including the experimentally determined channel stretching, i.e. setting $\sigma_{\text{stretch}} = 0.05$. Since we imagine that stretching is an important factor to consider, we expect the assembly score without elastic stretching to be considerably lower than in table 1. In table 2, we see the result of choosing threshold values (182, 0.6) for an assembly without assembly stretching as well as threshold values (201, 0.6) for an assembly with assembly stretching. The main difference between the two approaches is the merged rate, which is significantly higher when including assembly stretching. This observation is consistent with the heat maps and the expectation that the probability of finding a match above the PCC threshold increases with assembly stretching.

Table 2: Scores of emulated data with and without assembly stretching. The data consists of 14 emulated fragments in the pool, with the characteristics of average PCC with theory = 0.7, average length=258 pixels with a 5 pixel standard deviation and a channel stretching factor of 0.05. Minimum overlap was 201 pixels with assembly stretching and 182 pixels without assembly stretching. Minimum PCC was 0.6 in both cases. Coverage is the pixel size of the largest bartig and positional accuracy is the ratio of the number of correctly placed fragments to the total number of fragments.

| stretching | no. of iterations | coverage | no. of bartigs | merged rate | positional accuracy | final PCC |
|---|---|---|---|---|---|---|
| No | 50 | 371 | 2.4 | 58% | 69% | 0.81 |
| Yes | 50 | 409 | 2.6 | 88% | 70% | 0.83 |

The coverage is also a bit higher, but the total coverage is also typically larger. This can be seen in fig. 13 below, which are two comparisons of the bartigs and the theory that correspond to the data in table 2. Given that the main goal is to able to characterise a DNA with a barcode, the increase in coverage gained from assembling more fragments with assembly stretching than without is a clear advantage. This advantage has not come to the expense of a lower assembly quality such as a lower positional accuracy or a lower final PCC. Positional accuracy and final PCC are instead rather stable.



(a) Comparison of bartigs obtained from an assembly to the underlying theory. The assembly did not include elastic stretching.

(b) Comparison of bartigs obtained from an assembly to the underlying theory. This assembly did include elastic stretching.

Figure 13: 14 fragments with characteristics of average PCC with theory = 0.7, average length=258 pixels with a 5 pixel standard deviation and a channel stretching factor of 0.05, was assembled into bartigs. Minimum overlap was 201 pixels with assembly stretching and 182 pixels without assembly stretching. Minimum PCC was 0.6 in both cases.

## 3.4 Experimental barcodes

In table 3 we see the numbers corresponding to an assembly with and without stretching, using the same threshold values as for the emulated barcodes. We define the correct position of a fragment to be in an interval of 6 pixels around its best position in the theory. This imperfection reflects the fact that in an experiment we do not know which part of the DNA a fragment corresponds to.

Table 3: Scores of experimental data with and without assembly stretching. The data consists of 47 experimental fragments, with the characteristics of average length=294 pixels with a standard deviation of 36 pixels. Overlap and PCC thresholds were $(0.6, 201)$ and $(0.6, 183)$ with and without stretching, respectively. Coverage is the pixel size of the largest bartig and positional accuracy is the ratio of the number of correctly placed fragments to the total number of fragments.

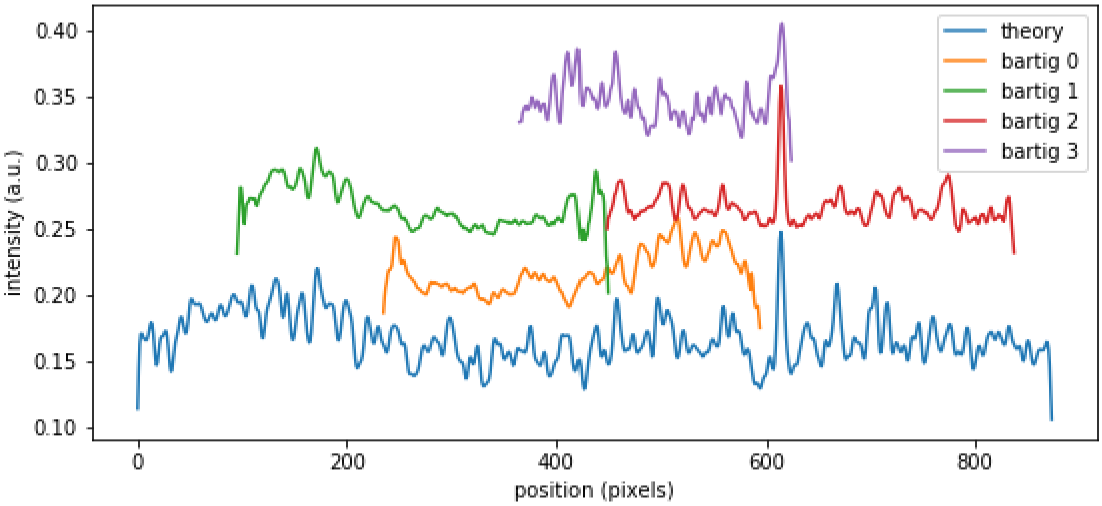
| stretching | coverage | no. of bartigs | merged rate | positional accuracy | final PCC |
|---|---|---|---|---|---|
| No | 390 | 4 | 91% | 37% | 0.64 |
| Yes | 372 | 5 | 98% | 63% | 0.65 |

Looking at table 3, including assembly elastic stretching increases the positional accuracy considerably. The final PCC, that is the PCC against theory of the largest contig, is not as as high as for emulated data. This fact may be partly explained by the need for fine-tuning the threshold values for fragments of characteristics present here in order to obtain the highest possible final PCC.

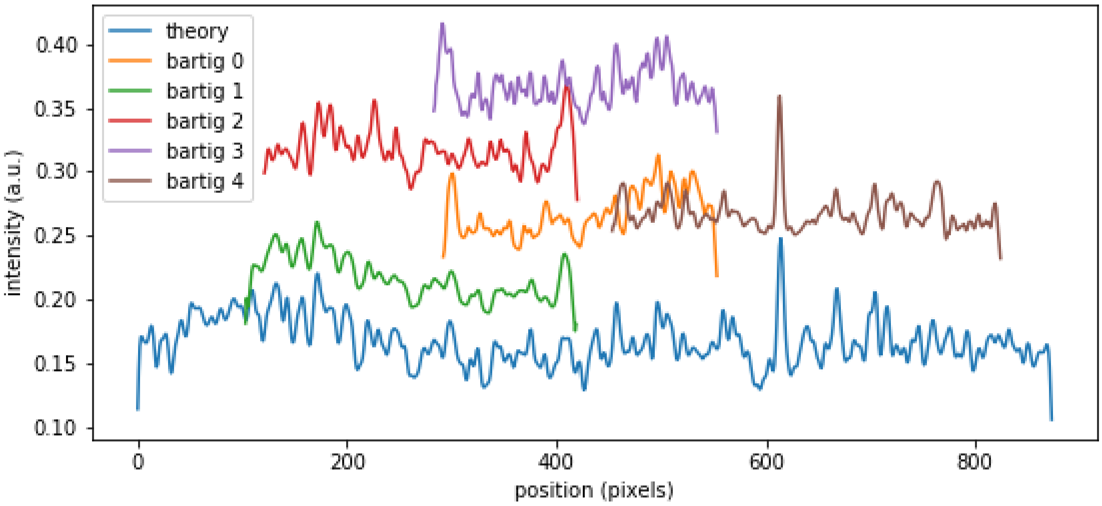
In fig. 14, we can see how the bartigs compare to the theory visually. In fig. 14b, there are two pairs of bartigs that covers approximately the same region on the theory. It is possible that these would be merged for other threshold values, leaving three separate bartigs instead of five. In fig. 14a there are no such pairs even though some fragments cover the same theory, which might reflect the lower positional accuracy in that case.

bartig 2 in fig. 14a and bartig 4 in fig. 14b look very similar. This fact may be due to those fragments displaying a high match score in the assembly, in both cases.

## Visualisation of bartigs of experimental barcodes and theory



(a) Comparison of bartigs obtained from an assembly to the corresponding theory. The assembly did not include elastic stretching.



(b) Comparison of bartigs obtained from an assembly to the corresponding theory. This assembly did include elastic stretching.

Figure 14: Forty-seven experimental fragments was assembled into bartigs with and without assembly stretching. Minimum overlap was 201 pixels with assembly stretching and 182 pixels without assembly stretching. Minimum PCC was 0.6 in both cases. The fragments' average length was 294 pixels with a standard deviation of 36 pixels.

# 4 Conclusion and outlook

Using DNA barcoding as a technique for characterising the main features of a DNA have advantages including efficiency and cost. A main difficulty is that DNA is randomly fragmented during DNA extraction from cells. This thesis extends a previous computational method for piecing together fragments from several cells, by introducing elastic stretching in the assembly.

To try and make DNA barcoding as functional a technique for DNA identification as possible, we have followed the result of a previous analysis and extended that method by introducing an elastic stretching in the assembly. This addition is well motivated by the fact that it increases the number of merged fragments, covering a larger part of the theory. For experimental fragments, the correctness of the assembly seem to increase as well, but this is not fully reliable until optimal threshold values are picked, which in turn depend on the barcode length distribution and average highest PCC against theory of the pool of experimental fragments.

Another point to notice is how the final PCC for emulated fragments increase with ca. 0.1 compared to the average fragment PCC. Evidently, the merging has a positive noise-cancelling effect. It seems plausible that a larger number of merged fragments contributes to this effect, which might explain the slightly higher final PCC when assembly stretching is included. Considering the final PCC for experimental barcodes, the difference in final PCC with and without assembly stretching is yet smaller than for emulated barcodes, but so is the difference in merged rate. Otherwise, the final PCC has instead decreased by ca. 0.5 compared to the average fragment PCC. This may reflect a difference between theoretical barcodes and experimental ones that is yet to be considered. A reasonable guess is that the difference is due to experimental barcodes having slightly different amplitudes due to slightly varying experimental conditions, which would render the mergings less accurate. A possible extension of the method in this thesis could therefore be to find all barcodes' best positions, but avoiding to actually merge them.

With more time available, it would have been interesting to alter the noise and see what results could be expected from better experimental fragments, such as with PCC=0.9 against theory. Also fine-tuning threshold values by trying all high-scoring combinations and seeing the effect on the result, is a natural next step. Lastly, reduced barcode resolution (pixelation effects) arises when a barcode is stretched out repeatedly, which can happen with assembly stretching. This effect can be circumvented by merging barcodes that have been defined on the whole real line and then evaluating the merged barcode at pixel positions.

# References

[1] Vilhelm Müller, Fredrik Westerlund. Optical DNA mapping in nanofluidic devices: principles and applications. *Lab Chip*. 2017; 17:579-590.

[2] Vilhelm Müller, Albertas Dvirnas, et al. Enzyme-free optical DNA mapping of the human genome using competitive binding. *Nucleic Acids Research*. 2019; 47(15).

[3] Lena K. Nyberg, Saair Quaderi, et al. Rapid identification of intact bacterial resistance plasmids via optical mapping of single DNA molecules. *Scientific Reports*. 2016; 6:30410.

[4] Wensi Zhu. Hierarchical clustering matrix method (HCM) applied to DNA barcode assembly for bacterial chromosomes. 2018. Lund University. Retrieved from https://lup.lub.lu.se/student-papers/search/publication/8963612

[5] Vilhelm Müller, et al. Cultivation-Free Typing of Bacteria Using Optical DNA Mapping. *ACS Infectious Diseases*. 2020.

[6] Adam N. Nilsson, Gustav Emilsson, et al. Competitive binding-based optical DNA mapping for fast identification of bacteria - multi-ligand transfer matrix theory and experimental applications on Escherichia coli. *Nucleic Acids Research*. 2014; 42(15).