# NOISY MIXTURE MODELS FOR NANOPORE GENOMICS

*Draft of May 8, 2017 at 20 : 44*

BY

HUOZHI ZHOU

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Bachelor of Science in Electrical and Computer
Engineering
in the College of Engineering of the
University of Illinois at Urbana-Champaign, 2017

Urbana, Illinois

Advisor:

Prof. Lav R. Varshney

# ABSTRACT

We describe a new scenario based on a combination of using nanoscale semiconductor materials and statistical algorithms to achieve high SNR current signals for robust DNA sequence base calling. In our setting, altered DNA molecules are threaded through nanopores in electrically active two-dimensional membranes such as graphene and molybdenum di-sulphide to be sensed by changes in electronic currents flowing through the membrane. Unfortunately, solid-state nanopores have been unsuccessful in DNA base identification due to the conformational stochastic fluctuations of DNA in the electrolytic solution inside the pore, which introduces significant noise to the measured signal. Hence, we propose an integrated effort that combines electronic simulation based on device physics with statistical learning algorithms to perform clustering and inference from the solid-state nanopore data. In particular we develop Gaussian Mixture Models (GMMs) that take into account the characteristics of the system to cluster the electrical current data and estimate the probability of the DNA position inside the nanopore. The validity of the learning algorithms for noisy GMM model has been demonstrated for uniform and Gaussian noise models with synthetic data sets. We also demonstrate the implementation of a pipelined version of the GMM training algorithm, which can be used to realize in near-sensor computing and inference systems. Finally, we also propose one possible solution to the theoretical resolution limit of nanopore DNA sequencing.

*To my parents, for their love and support.*

# ACKNOWLEDGMENTS

Firstly, I would like to express my sincere gratitude to my advisor Prof. Varshney for the continuous support of my senior thesis and related research, for his patience, motivation, and immense knowledge. His guidance helped me in research and writing of this thesis. I could not have imagined having a better advisor and mentor for my senior thesis.

Besides my advisor, my sincere thanks also goes to Aditya Sarathy, a Phd student in Prof. Leburton's group, who collaborated with me on my senior project constantly. Without his precious support, it would not have been possible to finish this senior research project.

# TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION . . . . . . . . . . . . . . . . . . . . . . 1

CHAPTER 2 NANOPORE TRANSISTOR SETUP . . . . . . . . . . . 5

CHAPTER 3 CLUSTERING USING GAUSSIAN MIXTURE MODELS . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9

CHAPTER 4 GMMS WITH BACKGROUND NOISE . . . . . . . . . 14
4.1 Case 1: Uniform Noise . . . . . . . . . . . . . . . . . . . . . . . . 14
4.2 Case 2: Gaussian Noise . . . . . . . . . . . . . . . . . . . . . . . . 17

CHAPTER 5 TOWARDS NEAR SENSOR COMPUTING SYSTEMS: PIPELINING THE GMM ALGORITHM . . . . . . . . . . . 21

CHAPTER 6 THEORETICAL RESOLUTION LIMIT . . . . . . . . . 23

CHAPTER 7 CONCLUSIONS AND FUTURE WORK . . . . . . . . . 24

REFERENCES . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25

# CHAPTER 1

# INTRODUCTION

In recent years, there has been growing interest in the scientific community to develop inexpensive and high-accuracy devices for biomolecular identification [1]. One such technology is nanopore-based devices that have the potential to revolutionize diagnosis and treatment of diseases and hence, possibly enable medicine. Given the importance of identifying genetic information and consequently studying its impact on a myriad of diseases, nanopore-based DNA sequencing devices hold great promise as possible replacements to conventional sequencing technology by eliminating the need for chemical labeling or sample amplification.

Over the past few years, there has been a rapid development in the realm of nanopore sequencing with biological and solid-state nanopores being explored as possible materials to extract genetic information via ionic or electronic currents. Although biological nanopores such as $\alpha$-hemolysin [2] and MspA [3] already exhibit great potential for DNA sequencing, there are drawbacks to biological pores, including fixed pore size and weak mechanical strength. Such drawbacks can be overcome by the use of solid-state nanopores [4]. Solid-state nanopores have been demonstrated to have a distinct advantage over biological pores in terms of flexibility in pore design and mechanical strength, lending credence to their potential for genomic applications. Two-dimensional solid-state materials such as graphene and molybdenum di-sulphide ($MoS_2$) in particular have attracted attention because of their atomically-thin layered structure and electrically active characteristics, predisposing them to offer single base resolution and simultaneously multiple modalities of detecting bio-molecular translocation [5, 6]. Monolayer and multi-layer two-dimensional solid state membranes containing nanopores are capable of detecting translocation of bio-molecules such as DNA and proteins by ionic and transverse sheet currents as demonstrated by simulations [7, 8] and experiments [9].

In typical nanopore sequencing experiments, DNA molecules are threaded through a nanopore under an applied voltage; an ionic current flowing through the nanopore alongside the DNA is observed and different transient dips due to different DNA nucleotides (ionic current blockade) are measured. Resolving the magnitude and duration of each dip permits one, in principle, to identify individual bases and, in turn, the sequence of DNA. However, first-principles calculations suggest another opportunity for graphene to detect DNA, namely through the transverse sheet current across graphene nanoribbons (GNRs) that can be measured [10]. It was shown previously that the sensitivity of GNR to translocated DNA can be drastically enhanced by tailoring the edge of the GNR into a quantum point contact geometry (QPC) or by tuning the carrier concentration in the GNR [7]. The GNR devices were found in simulations to be able to sensitively probe the helical geometry of double-stranded DNA (dsDNA) [7], the conformational transitions from helical to zipper form of dsDNA, as well as the number of nucleotides in stretched ssDNA [11]. A further advancement encouraging the use of graphene nanopores for DNA sequencing are actual experiments that have detected DNA permeation through a nanopore in GNRs by means of sheet current measurements [12], but have not yet resolved DNA nucleotide identity.

Apart from probing the DNA using two independent methods (i.e. ionic and transverse currents), Girdhar et al. [7] also proposed the integration of solid-state multi-layer nanopore membranes within a multi-functional electronic device to increase its detection sensitivity. Among the advantages of the solid-state nanopore is its compatibility with semiconductor nanoelectronics that favors the fabrication of compact device. Until now, many efforts to detect, identify, and map DNA patterns using solid-state nanopores have been unsuccessful because the conformational stochastic fluctuations of DNA in electrolytic solution inside the pore add significant noise to the measured signal [13]. Additional noise sources include possible tunneling effects and electron scattering by non-ideal edges of the electronic sensing membrane. The contribution of such noise sources to the final signal is usually quite large, effectively screening the electronic signatures of the DNA nucleobases.

In this regard, the development of a versatile and generally applicable sensor technology with a high signal-to-noise (SNR) ratio is desirable. For this purpose, we propose an integrated effort that combines electronic simulation based on device physics with machine learning techniques to characterize the electronic signals arising from solid-state nanopore sensing, and their integration into systems that might implement algorithms for real-time base calling.

Although machine learning algorithms for base calling or SNR improvement have not been reported in solid-state nanopores, they have been extensively utilized in the real-time base identification in genetic and epigenetic applications. The recently introduced, ∼ $1000 DNA sequencer MINION developed by Oxford Nanopore Technologies, relies heavily on deep learning for the identification of nucleobase sequences from measured ionic currents. In fact, it was recently reported that redesign of these deep learning algorithms saw an accuracy improvement of almost 10% for the same nanopore device [14]. While the deep learning based base calling algorithm used by Oxford Nanopore Technologies is proprietary, there have been open-source software implementations of these algorithms available [15].

All of the previously described base calling methods have been implemented in software for biological nanopores (used in the MINION device), but in order to achieve a true integration of the genomic information with semiconductor nanotechnology, hardware implementations of these algorithms might be critical, in addition to the utilization of solid-state nanopores. One of the primary impediments is the low SNR due to thermal fluctuations of DNA bases, ions, and water inside solid-state nanopores [13]. In particular, the noise from variations in DNA structural conformation inside a nanopore may offset the signal induced by each nucleotide, largely weakening the sensing sensitivity of the nanopore device. Therefore, in order to ensure the effective operation of a nanopore device in single-molecule sensing applications, controlling the motion of biomolecules in solid state nanopores is highly desirable. The electronic signal obtained from the controlled translocation of DNA through the nanopore is further processed using clustering algorithms in order to infer information about the DNA or for further processing the data.

In this thesis, we briefly describe a possible setup of a nanopore transistor that achieves a controlled motion of the DNA translocation through the pore, using which possible models of the movement of the DNA within the pore and its correlation to the applied control voltage can be extracted. The extraction of the cluster data is performed via Gaussian Mixture Models (GMMs). However, since the stochastic fluctuations of the DNA can result in noisy data, we modify GMM-based algorithms that take the inherent noise characteristics of the DNA into account.

Specifically, we study modified GMM algorithms under the influence of uniform noise and Gaussian noise respectively. These algorithms are validated by synthetic data and are yet to be tested with the measured nanopore currents because of limitations in the computational resources required to fully simulate the nanopore transistor in various circumstances. We also explore the development of pipelined versions of the GMM algorithm since integration of genomic and semiconductor technologies augur the possibility of near sensor computing with reduced energy requirements. Finally, we propose one possible solution to the theoretical resolution limit of nanopore DNA sequencing where we get some insights from radar signal processing.

# CHAPTER 2

# NANOPORE TRANSISTOR SETUP

In solid-state nanopores, a possible strategy to control molecule fluctuation was proposed by Girdhar et al, using a multi-layered membrane transistor containing a motion-control electrode layer, as shown in Fig 2.1, to shape the electrostatic landscape in the nanopore to reduce the stochastic fluctuations of the interior biomolecules. The electrically active multi-layer membrane device utilizes a combination of metallic or semiconducting electrodes to control the motion and graphene to read out the currents while the DNA is passing through the nanopore. Since controlling the motion and translocation velocity of DNA is key, we utilize gold electrodes to slow down DNA translocation with a bias voltage ($V_{C1}$) applied, which would operate as a control gate to trap the DNA inside the pore. Simultaneously, a graphene layer ($V_{DS}$) is used to read sheet currents and discern passing nucleotides.

In order to realize the functionality of the device in Fig 2.1, we utilize a simpler model of the device whose cross section is shown in Fig 2.2. The setup consists of a gold electrode, connected to a control voltage source ($biV_C$) and an electrically active graphene membrane, whose sensitivity is controlled by a gate voltage ($V_{gate}$). The gold electrode can be tuned to shape the electrostatic potential within the confines of the pore, while the graphene membrane can be used to sense the electronic current at every instant in time. The gold electrodes and the sensing membrane are assumed to be separated by a dielectric so that there is no influence of the applied stabilization voltage on the carrier density of the graphene membrane. We will utilize this model to cluster the DNA positions at different gate voltages to study the relationship between the position of the DNA backbone and the transverse sheet currents.

The trajectories and behavior of DNA within the nanopore devices were simulated by molecular dynamics (MD) simulations [17, 5, 16] while the calculations of the electronic current were carried out using a method based on the quantum mechanical non-equilibrium Green's function. These sim-

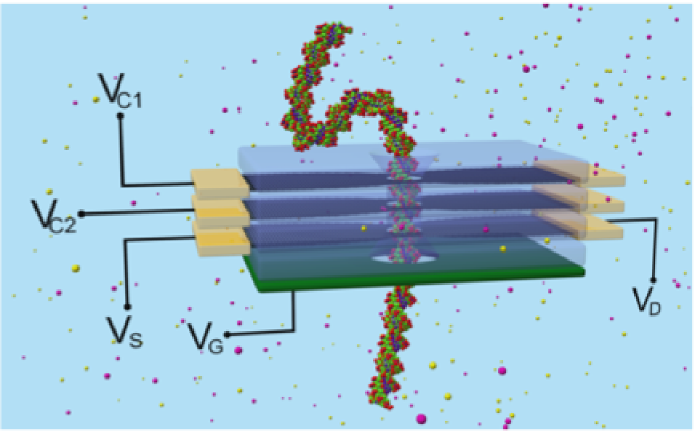

Figure 2.1: Schematic diagram of a four-layer device [7] containing two graphene layers (black) to control the translational motion of DNA through the nanopore. The top graphene layer ($V_{C1}$) controls the translational speed of the DNA, whereas the second ($V_{C2}$) controls the lateral confinement of the DNA within the nanopore. The third graphene layer ($V_{DS}$) measures the sheet current. Finally, a heavily doped back gate (green) lies underneath the sheet current layer to control the carrier concentration. Oxide barriers (transparent) between different graphene layers provide electrical isolation.

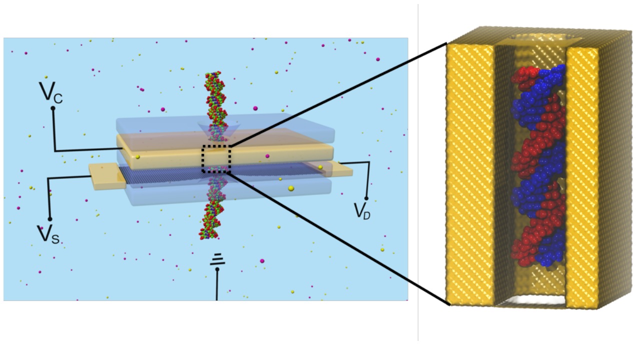

Figure 2.2: Cross section of the device setup to study the relationship between the DNA fluctuations and transverse sheet currents. The gold electrodes are connected to a voltage source $V_C$ that can be tuned to sculpt the electrostatic landscape within the pore in order to control the fluctuations of the DNA. Simultaneously, a graphene membrane reads out the current at every instant in time. The sensitivity and carrier concentration on the graphene membrane are controlled by a gate voltage placed beneath the graphene membrane. It is assumed that there is no cross interference between the gate voltage and the stabilizing voltage on the graphene sensitivity [16].

ulations mimic stochastic fluctuations of the DNA via thermostats such as

Nose-Hoover-Langevin thermostats. Hence the algorithms developed to improve the SNR can be tested using the data generated from these simulations and reliably applied to experimental situations as well.

In Fig 2.3, we show center of mass (CoM) positions of the DNA molecule in the XY-plane for each frame at various positive voltages [16]. At zero electrode bias, the DNA positions at different times spread almost uniformly inside the pore, as no significant interaction between the pore and the DNA molecule exists. At positive voltages (Fig 2.3), on the other hand, the DNA CoM positions become more localized around a specific location in the pore, indicating damping of the DNA motion. This location is not exactly at the pore center because of the slight randomness in conformational change in response to applied voltages. The reduction of DNA fluctuations is discernible through the backbone spread in the overlapped DNA conformations obtained from the MD trajectory.

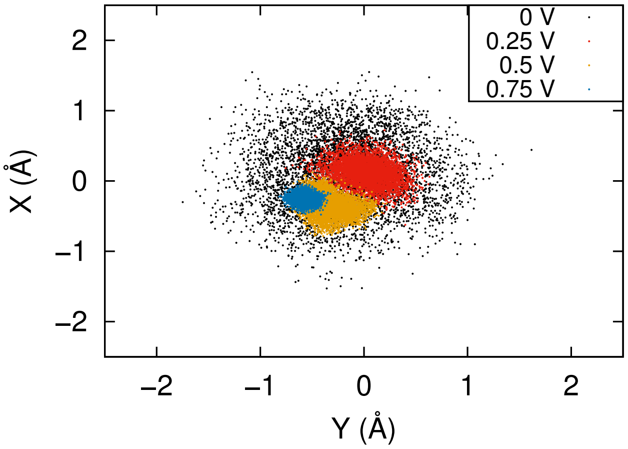

Figure 2.3: The positions of the DNA backbone within the nanopore at every time instant for different stabilizing voltages applied. It is evident that at higher gate voltages, the stochastic fluctuations of the DNA become discernibly reduced suggesting that larger voltages reduce the motion of the DNA molecule more strongly since the range of motion is reduced.

From the CoM positions at every instant in time we also calculate the sheet conductance (or current) variations of the stabilized trajectory to gain an insight into the effect of the stabilization on the graphene sheet current [18]. Fig 2.4 shows the histograms for the conductances measured for the gated ($V_C = 0.8\,\mathrm{V}$) and un-gated ($V_C = 0\,\mathrm{V}$) scenarios. While the means for the two gate stabilizing voltages are similar, the variance for the gated

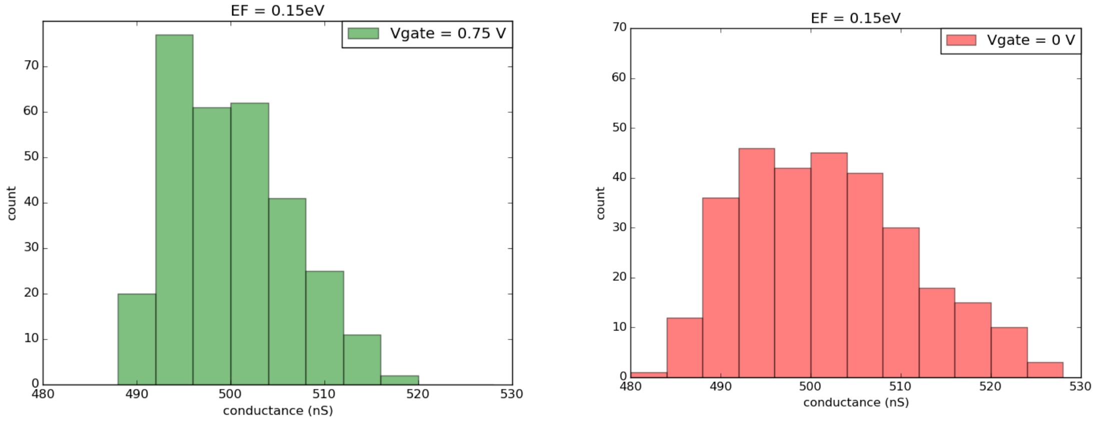

Figure 2.4: Histogram of conductance measured for the unstabilized and stabilized DNA trajectories. The histogram for the stabilized DNA shows a smaller variance due to localization of the DNA backbone within the pore in the presence of a sculpted electrostatic landscape.

trajectory is significantly smaller than the un-gated trajectory. This is attributed to the position of the DNA being confined to a smaller region due to the presence of the gating voltage. This demonstrates that the presence of the gated voltage can be used to significantly alter the DNA translocation through the nanopore and the corresponding effect can be measured using transverse sheet currents. With these measurements, we observe a considerable variation in the measured sheet current statistics, thereby indicating an avenue to understanding the implications of a controlled DNA translocation on the measured sheet current.

The previously described results regarding the positions of the DNA backbones and the histogram of the conductance measurements show the possibility of relating the DNA backbone positions and the transverse sheet currents. However, clustering algorithms are required to estimate the relationships between the two physical quantities. Hence, we use GMMs to cluster the DNA backbone and transverse sheet current data.

# CHAPTER 3

# CLUSTERING USING GAUSSIAN MIXTURE MODELS

The idea behind GMMs is to treat the probability distribution of the clustered quantity as a sum of weighted Gaussians as

$$p\{\mathbf{x}|V_{gate}\} = \sum_{k=1}^{K} \omega_k \frac{1}{\sqrt{2\pi\sigma_{k,\mathbf{x}}}} e^{\frac{\mathbf{x}-\mu_k{}^2}{2\sigma_{k,\mathbf{x}}}}. \tag{3.1}$$

where $\mathbf{x}$ is the position of the DNA along two dimensions i.e., $(x,\ y)$ at a given time instant, $V_{gate}$ is the applied stabilization voltage used to control the DNA backbone position, while $\mu$ and $\sigma$ are the means and variances of contributing distributions. The weight factor $\omega_i$, as well as $\mu$ and $\sigma$ will need to be learned using the training data.

For the nanopore transistor with stabilizing voltages, we utilize the Expectation-Maximization (EM) algorithm to compute the parameters for GMM, namely, the mixing coefficients $\{\omega_i\}_{i=1,\dots,k}$, the set of mean vectors $\{\mu_i\}_{i=1,\dots,k}$, and the set of covariance matrices $\{\mathbf{\Sigma}_i\}_{i=1,\dots,k}$. The outline for the EM algorithm for GMMs is in Algorithm 1.

For this problem, we have obtained the training data for possible DNA locations and sheet current on the probe for different fixed voltages. Thus we can apply the EM algorithm to compute the ML estimate for some conditional probability distribution functions, $P\{I\{\mathbf{x}\},\mathbf{x}|V\}$ and $P\{\mathbf{x}|V\}$.And The dependence between the current and DNA backbone position, is given by

$$P\{\mathbf{x}|I(\mathbf{x})\} = \frac{P\{I(\mathbf{x}),\mathbf{x}\}}{P\{I(\mathbf{x})\}}. \tag{3.2}$$

In order to compute the dependence, we need to figure out the denominator and numerator which is difficult due to our limited data. Note that the marginal probability distribution is the integral of the conditional probability

**Algorithm 1** EM-GMM training algorithm

```
While log-likelihood function $\log p(\cdot)$ does not converge do
    E step
        Evaluate the responsibilities
        $\gamma(z_{nk}) = \frac{\omega_k N(\mathbf{x}_n|\mu_k,\mathbf{\Sigma}_k)}{\sum_{j=1}^{K} \omega_j N(\mathbf{x}_n|\mu_j,\mathbf{\Sigma}_j)}$
    M Step
        Re-estimate the parameters using current responsibilities
        $N_k = \sum_{n=1}^{N} \gamma(z_{nk})$
        $\mu_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})\mathbf{x}_n$
        $\mathbf{\Sigma}_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})(\mathbf{x}_n - \mu_k^{new})(\mathbf{x}_n - \mu_k^{new})^T$
        $\omega_k^{new} = \frac{N_k}{N}$
     Evaluate the new likelihood function
        $\log p(\mathbf{X}|\mu, \mathbf{\Sigma}, \pi) = \sum_{n=1}^{N} \log\{\sum_{k=1}^{K} \omega_k N(\mathbf{x}_n|\mu_k, \mathbf{\Sigma}_k)\}$
```

distribution as shown below.

$$P\{I(\mathbf{x}), \mathbf{x}\} = \int_{-\infty}^{\infty} P\{I(\mathbf{x}), \mathbf{x}|V\}dV. \tag{3.3}$$

$$P\{\mathbf{x}\} = \int_{-\infty}^{\infty} P\{\mathbf{x}|V\}dV. \tag{3.4}$$

The difficulty is we currently have trained conditional probability distribution for only a finite number ($N_{voltages} = 4$) of different given voltages. Thus we cannot directly apply the exact formula to compute the marginal probability distribution. For now, we apply the following equations to approximate the probabilities appear in the denominator and numerator shown in (3.2).

$$\hat{P}\{I(\mathbf{x}), \mathbf{x}\} = \sum_{i=1}^{N_{voltages}} \frac{1}{N_{voltages}} P\{I(\mathbf{x}), \mathbf{x}|V_i\}. \tag{3.5}$$

$$\hat{P}\{\mathbf{x}\} = \sum_{i=1}^{N_{voltages}} \frac{1}{N_{voltages}} P\{\mathbf{x}|V_i\}. \tag{3.6}$$

The reason why we assume for different voltages the probability is uniform is because we have equal amount of training data for each given voltage. Another perspective is to view the different conditional probability distribution functions as the quantizer output of all conditional distribution functions given all possible values of voltage. In the future, if we have a sufficiently good interpolation to approximate the conditional distribution for all possible voltage value, we can improve our precision by using the precise formula to compute the marginal distribution.

While the GMM model expresses the distribution of data as a linear combination of Gaussians, the exact number of Gaussians to be utilized for a given cluster data set is not specified. Kullback-Leibler divergence [19] is commonly used to measure information lost when approximating a model. The definition of KL divergence for continuous case is given as

$$\begin{aligned} D_{KL}(f||g(\cdot|\theta)) &= \int_\Omega f(x)\log\frac{f(x)}{g(x|\theta)}dx \qquad (3.7) \\ &= \int_\Omega f(x)\log(x)dx - \int_\Omega f(x)\log(x|\theta)dx. \qquad (3.8) \end{aligned}$$

where the second term is the relative KL information. In this formula, $f(x)$ is the actual probability distribution and $g(x|\theta)$ is the approximated distribution with parameter $\theta$. It is trivial to apply Jensen's inequality to prove that KL information is nonnegative and is zero if and only if two distributions are identical.

However, KL divergence has several limitations [20]. For example, in real-world problems, the true $f$ is unknown and the parameter $\theta$ in $g$ must be estimated from the empirical data $y$. In this case, we need to compute the expected KL divergence to measure the difference between $f(x)$ and $g(x|\hat{\theta}(y))$.The expected KL divergence is given below

$$E_y[D_{KL}(f||g(\cdot|\theta))] = \int_\Omega f(x)\log(x)dx - \int_\Omega f(y)[\int_\Omega f(x)\log(x|\hat{\theta}(y))dx]dy. \qquad (3.9)$$

The first term is a constant, which means in order to minimize the expected KL divergence, we need to maximize the second term. In other words,

$$\max_{g\in G}\int_\Omega f(y)[\int_\Omega f(x)\log(x|\hat{\theta}(y))dx]dy \;=\; \max_{g\in G} E_y E_x[\log(g(x|\hat{\theta}(y))]. \qquad (3.10)$$

where $G$ is the collection of all possible models, $\hat{\theta}(y)$ is the MLE estimate based on model $g$, and $y$ is empirical data. It is always not easy to compute the expected KL divergence. However, there are two approximate unbiased estimates of $\max_{g \in G} E_y E_x[\log(g(x|\hat{\theta}(y))]$ which work well if the number of samples is sufficiently large and the model is also good. And these two approximate unbiased estimates of the expected KL divergence lead to Akaike information criterion (AIC) and Bayesian information criterion (BIC), which are both based on 2-Log likelihood. They are defined as follows

$$AIC = -2 \log L(\hat{\theta}|y) + 2k \tag{3.11}$$

$$BIC = -2 \log L(\hat{\theta}|y) + k \log(n) \tag{3.12}$$

where $L$ is the likelihood function, $\hat{\theta}$ is the ML estimate of $\theta$, $k$ is the number of the estimated parameters include means, variances and so on. And $n$ is the number of observations. One thing important to mention is for both of these two criteria, the first term is caused by bias and the second term is caused by variance. Bias means the the difference caused by the projection of the actual parameters to a parameter space with limited dimension. Variance means the difference between the real parameters and the ML estimate. Therefore, using either of the criteria, we choose the number of Gaussians (usually $1 \sim 10$) that produces the least AIC or BIC.

Fig 3.1 shows the probability density functions of the DNA backbone positions for the un-gated and gated DNA translocation cases. We find that the probability density of the DNA backbone is in the center of the pore. Ten Gaussians have been used to cluster the data. The pdfs also display limited widths as the gating voltage increases. A similar process could be used to compute the cluster for the transverse sheet conductances and thereby the conditional probability densities. Fig 3.2 displays the probability density that the DNA is at a specific position given that the conductance measured was $1.7\mu$s. While we find that there might be two regions that the DNA is present, in order to infer the position, we could define a metric such as the maximum probability density.

Hence, using the above formulation, we can infer that the location which has the highest pdf value is the most probable DNA location. This approach can be utilized to develop a scheme to train and adaptively tune the position of the DNA given the conductance measured at a given time instant by tuning the stabilization voltage. While this algorithm to move the DNA might give good control over the DNA translocation, it assumes that the

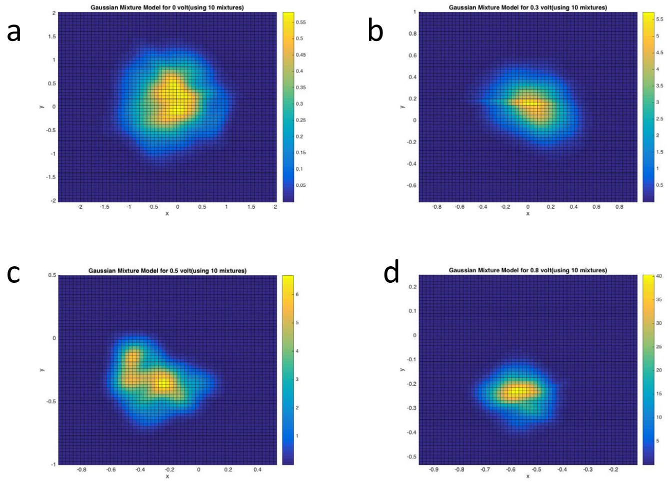

Figure 3.1: Probability distribution function of the conductance backbone positions using 10 Gaussian mixtures and different gate voltages: (a)$V_C = 0.$ V (b) $V_C = 0.3$ V (c) $V_C = 0.5$ V (d) $V_C = 0.8$ V.

underlying data is noise-less. This might not always be the case since noise can be induced by fluctuations or due to electronic scattering in the sensing membrane. Therefore, we next develop a modified GMM algorithm that accounts for the underlying noise statistics.

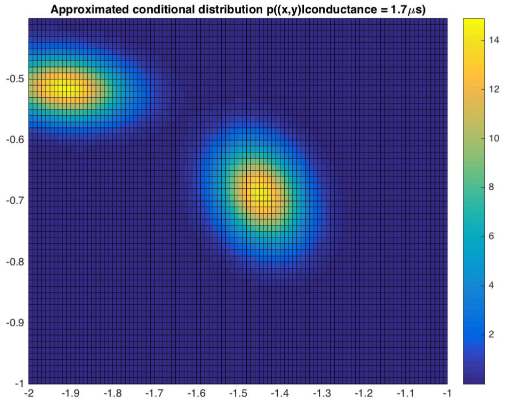

Figure 3.2: Approximate probability density given that conductance measured is $G = 1.7\mu$s. Note that the figure is plotted to display the pdfs only in the lower left corner of the circular nanopore.

# CHAPTER 4

# GMMS WITH BACKGROUND NOISE

In this chapter, we discuss GMM-based algorithm to cluster data in the presence of noise. To be more specific, we address two special cases of noise: one is uniform noise while the other is Gaussian noise.

## 4.1 Case 1: Uniform Noise

Consider a finite mixture model with $K$ mixtures of the form

$$p(\mathbf{x}|\theta) = \sum_{k=1}^{K} \omega_k \mathcal{N}(\mathbf{x}|\mu_k, \mathbf{\Sigma}_k). \tag{4.1}$$

where $\mathbf{x}$ is a multidimensional data set with $\omega_k$, $\mu_k$ and $\mathbf{\Sigma}_k$ being the weights, mean vector, and covariance matrix of the $k$th mixture. $\mathcal{N}$ is the multivariate normal distribution, and $\theta$ is an independent variable (stabilization gate voltage in the case of gated nanopore transistors). In the following sections regarding the GMMs with noise, we will assume that the number of clusters are defined beforehand.

While modeling a cluster of data with a GMM, the parameter estimates are sensitive to outliers and presence of background noise because the maximum likelihood estimate is basically the mean value, which is not a robust statistic [21, 22]. Additionally, the implementation of the GMMs involve estimating the converged value of the mean, covariance and weights over multiple iterations in multiple runs with each run beginning with a random guess of the means and variances. Hence, we can model the background noise by adding the noise component $f_0$ to the mixture model as [21]

$$p(\mathbf{x}|\theta) = P f_0(\mathbf{x}|\mu_{\mathbf{0}}, \mathbf{\Sigma_0}) + \sum_{k=1}^{K} \omega_k \mathcal{N}(\mathbf{x}|\mu_{\mathbf{k}}, \mathbf{\Sigma_k}). \tag{4.2}$$

where $P$ is the weight of the background noise component, with distribution $f_0$. For (4.2), the weights will be subject to the constraint

$$P + \sum_{k=1}^{K} \omega_k = 1. \tag{4.3}$$

Since we are assuming the noise to be uniform, we set

$$f_0(\mathbf{x}|\mu, \mathbf{\Sigma}) = \frac{1}{x_{max} - x_{min}}. \tag{4.4}$$

where $x_{max}$ and $x_{min}$ are the maximum and minimum of the domain under consideration. Therefore, in such a scenario, the initial weights of the Gaussian mixture can be found via $k$-means estimates, while $P$ can be assumed to another value such that sum of all the weights are the same. The EM algorithm for GMM with uniform noise is as follows, in Algorithm 2.

**Algorithm 2** EM-GMM-uniform noise training algorithm

```
While log-likelihood $\log p(\cdot)$ does not converge do
  E step
    Evaluate the current responsibilities
    $\gamma(z_{nk}) = \frac{\omega_k N(\mathbf{x_n}|\mu_\mathbf{k}, \mathbf{\Sigma_k})}{\sum_{j=1}^{K} \omega_j N(\mathbf{x_n}|\mu_\mathbf{j}, \mathbf{\Sigma_j})}$
  M Step
    Re-estimate the parameters using current responsibilities
    $N_k = \sum_{n=1}^{N} \gamma(z_{nk})$
    $P^{new} = 1 - \sum_{k=1}^{K} \omega_k$
    $\mu_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) \mathbf{x}_n$
    $\mathbf{\Sigma}_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})(\mathbf{x}_n - \mu_k^{new})(\mathbf{x}_n - \mu_k^{new})^T$
    $\omega_k^{new} = \frac{N_k}{N}$
  Evaluate the new likelihood function
    $\log p(\mathbf{X}|\mu, \mathbf{\Sigma}, \pi) = \sum_{n=1}^{N} \log\{\sum_{k=1}^{K} \omega_k N(\mathbf{x}_n|\mu_k, \mathbf{\Sigma}_k)\}$
```

From Algorithm 2, we notice that the weights are modified in the maximization step with the new weight of the noise term computed after each of the mixture weights. In order to test the GMM with background noise algorithm, we first develop a synthetic data set as shown in Fig 4.1. If we set the number of clusters $K = 2$ as input to the GMM with uniform background noise algorithm, we observe that the clusters identified at the end of EM do not coincide with the input clusters as shown in Fig 4.2. This is due to the weights of the uniform background noise not being taken into consideration.

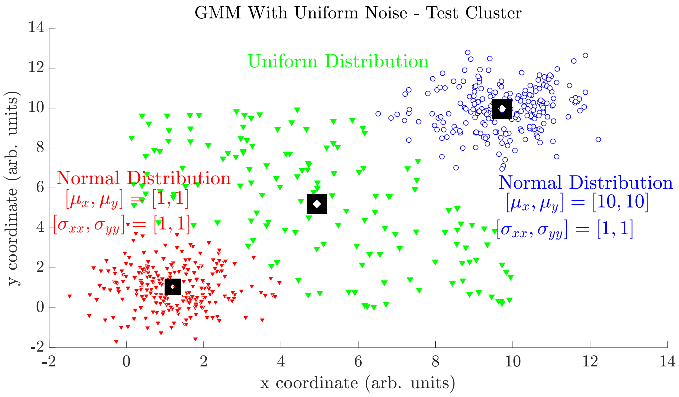

Figure 4.1: Input test cluster for the GMM with background noise algorithm. Two clusters are created with the points drawn from a normal distribution of means $(\mu_1^x, \mu_1^y) = (1, 1)$ and $(\mu_2^x, \mu_2^y) = (10, 10)$ colored as red and blue respectively. Uniform background noise defined along the domain defined from $[0, 0]$ to $[10, 10]$.

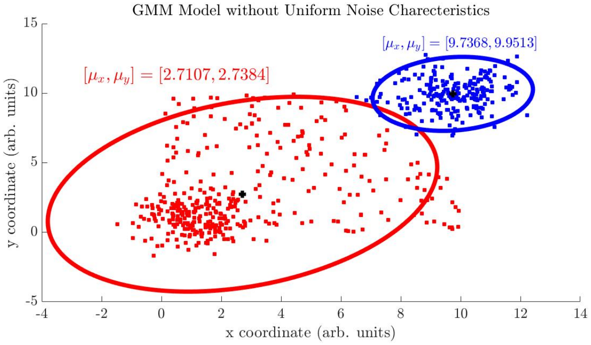

Figure 4.2: Identification of clusters in the input test data (Fig.4.2) using the GMM expectation maximization algorithm without considering the uniform background noise.

However, an implementation of the GMM algorithm with the background noise taken into consideration converges onto the correct clusters as shown in Fig 4.3. In fact even the means that are obtained at the end of the iterations are very close to the means of the input clusters.

Hence, from the above description and implementation of the EM algorithm with the background noise taken into consideration, we observe that a minor variation can enable an accurate description of the clusters. This approach was tested over many clusters and samples and was found to be a robust modification of the original EM algorithm.

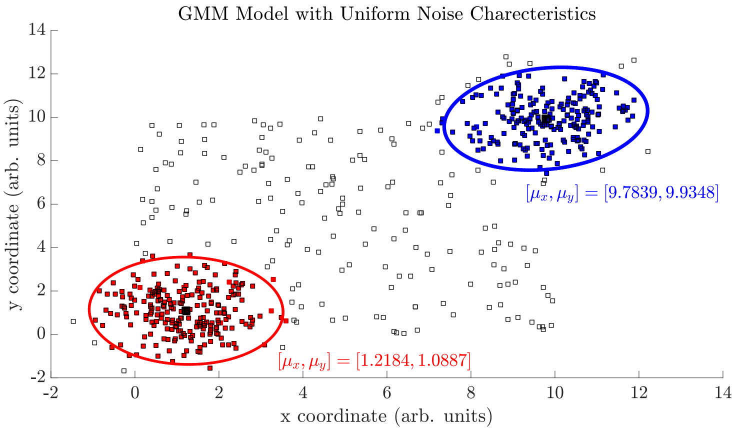

Figure 4.3: Identification of clusters in the input test data using GMM with background noise expectation maximization algorithm.

## 4.2 Case 2: Gaussian Noise

The GMM can also treated as a classification problem, whereby, each data point is assigned certain mean and variance to be associated along. In some cases, $\mathbf{x}_n$ can have missing data and only be partially complete. In such cases the probabilistic likelihood $p(\mathbf{x}|\theta)$ cannot be evaluated in the usual manner. Similar scenarios can also be thought of as data sets with noisy data such that the noise distribution is Gaussian. In this regard, it was proposed by Ozerov et al. [23] to modify the existing GMM algorithm with log-likelihood criterion for noisy data. In this paper, we study and implement the algorithm proposed by Ozerov et al.

Let us consider the scenario where

$$\mathbf{x}_n = \mathcal{N}(\mathbf{y}_n, \mathbf{\Sigma}_n). \tag{4.5}$$

where the parameters $\mathbf{y_n}$ and $\mathbf{\Sigma}_n$ are known, i.e., we can consider $\mathbf{y}_n$ to be a feature computed from a distorted signal and $\mathbf{\Sigma}_n$ to be an estimate of the noise covariance with zero mean.

In the original EM algorithm, the log-likelihood assumes that the distribution of $\mathbf{x}_n$ is accurately modeled, which may not always be the case. Hence, Ozerov et al. proposed a log-likelihood integration approach which does not rely on the assumption regarding the distribution of $\mathbf{x}$ but makes as if all values were observed. Here, the log-likelihood $\log p(\mathbf{x}|\theta)$ is replaced by its expectation over the observations which results in the function

$$\begin{aligned} f_{LLK}(\mathbf{y}, \mathbf{\Sigma}|\theta) &= E_{\mathbf{x}}[\log p(\mathbf{x}|\theta)|\mathbf{y}, \mathbf{\Sigma}, \mathbf{\Sigma}_n] \\ &= \sum_{n=1}^{N} \int_{\mathbb{R}^M} p(\mathbf{x}_n|\mathbf{y}_n, \mathbf{\Sigma}_n) \log \sum_{k=1}^{K} \omega_k \mathcal{N}(\mathbf{x}_n|\mu_k, \mathbf{\Sigma}_k) d\mathbf{x}_n. \end{aligned} \tag{4.6}$$

This can be solved in closed form for only one state ($K = 1$). However, an approximate solution to the integral can be obtained as

$$f_{LLK}(\mathbf{y}, \mathbf{\Sigma}|\theta) \approx \sum_{n=1}^{N} \log \sum_{k=1}^{K} \omega_k \mathcal{N}(\mathbf{y}_n|\mu_k, \mathbf{\Sigma}_k) e^{-\frac{1}{2} tr(\mathbf{\Sigma}_i^{-1} \mathbf{\Sigma}_n)}. \tag{4.7}$$

Utilizing the above formulas, given the inherent noise characteristics (covariance $\mathbf{\Sigma}_n$), it yields the EM algorithm for the GMM model with Gaussian noise, Algorithm 3.

**Algorithm 3** EM-GMM-Gaussian noise training algorithm

```
While likelihood function $f_{LLK}(\cdot)$ does not converge do
    E step
        Evaluate the responsibilities
        $\gamma(z_{nk}) = \frac{\omega_k N(\mathbf{x}_n|\mu_k, \mathbf{\Sigma}_k) e^{-\frac{1}{2} tr(\mathbf{\Sigma}_i^{-1} \mathbf{\Sigma}_n)}}{\sum_{j=1}^{K} \omega_j N(\mathbf{x}_n|\mu_j, \mathbf{\Sigma}_j)}$ for $k = 1...K$
    M Step
        Re-estimate the parameters using current responsibilities
        $N_k = \sum_{n=1}^{N} \gamma(z_{nk})$
        $\mu_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) \mathbf{x}_n$
        $\mathbf{\Sigma}_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})(\mathbf{x}_n - \mu_k^{new})(\mathbf{x}_n - \mu_k^{new})^T + \mathbf{\Sigma}_n$
        $\omega_k^{new} = \frac{N_k}{N}$
    Evaluate the new likelihood function
        $f_{LLK}(\mathbf{y}, \mathbf{\Sigma}|\theta) \approx \sum_{n=1}^{N} \log \sum_{k=1}^{K} \omega_k \mathcal{N}(\mathbf{y}_n|\mu_k, \mathbf{\Sigma}_k) e^{-\frac{1}{2} tr(\mathbf{\Sigma}_i^{-1} \mathbf{\Sigma}_n)}$
```

It can be seen that Algorithm 3 reduces to the classical EM algorithm if the noise covariance matrix is 0. In order to test the GMM algorithm with inherent Gaussian noise characteristics taken into account, we first generate a test cluster consisting of two sets of points drawn from normal distributions with means $[\mu_x^1, \mu_y^1] = [-1, -1]$ and $[\mu_x^2, \mu_y^2] = [1, 1]$ while the variances of both clusters are chosen as $\Sigma_{xx}^{1,2} = \Sigma_{yy}^{1,2} = 2$ as shown in Fig 4.4.

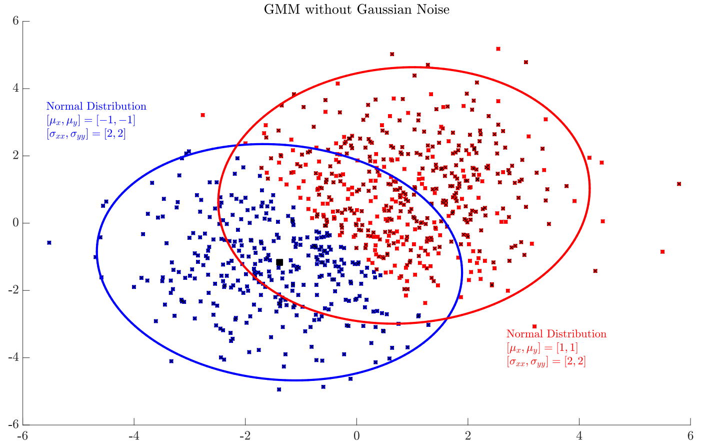

Figure 4.4: Test cluster for the GMM algorithm with background Gaussian noise, consisting of two sets of points drawn from normal distributions centered at [-1,-1], [1,1] and variances 2, 2 respectively.

When the GMM with Gaussian background noise algorithm was run till convergence (by assuming the covariance matrix of the noise term to be identity), we observed that two clusters were identified with the means and variances shifted from the original means and variances as shown in Fig 4.5. Another important feature to note is that, even though it is assumed that the noise covariance matrix is diagonal, due to the iterative solution of the means, covariances and weights, the resulting variance matrices might contain non-zero off-diagonal elements, which would result in the clusters being identified as ellipses, while the test cluster were generated with the variance matrices containing zero valued off-diagonal elements. While convergence is achieved for various test cases at varying densities of points, and differing number of clusters, there always seems to be a shift in the means calculated, which might be explained by the fact that each iteration of the GMM consists of a random initialization of the covariances, and hence convergence might occur at the local maxima of the log-likelihood criterion instead of the global optimum point. However, further studies regarding the validity of the approach and its convergence need to be carried out in a more systematic manner. Ozerov et al. do not compare the noiseless and noisy clusters but instead focus on studying the properties of the log-likelihood criterion and its application to speech processing. Since the GMM algorithm can be used to “complete” data sets or extract features from noisy sensor data, a possible application of this method can be in the feature extraction layer in deep learning systems to decompose the input data into true data and noise rather than blindly learning the parameters that describe the input data [19].

The discussions and results outlined in Chapter 4 show that the GMM is a versatile model which can be generalized to incorporate many noise models. The key step always seems to be the simplification or approximation of the log-likelihood estimate. While only uniform and Gaussian noise models have been reported in the literature, it might be worthwhile to explore other noise models such as $1/f$ noise, since it is thought to be the most dominant source of noise in nanopore current measurements.

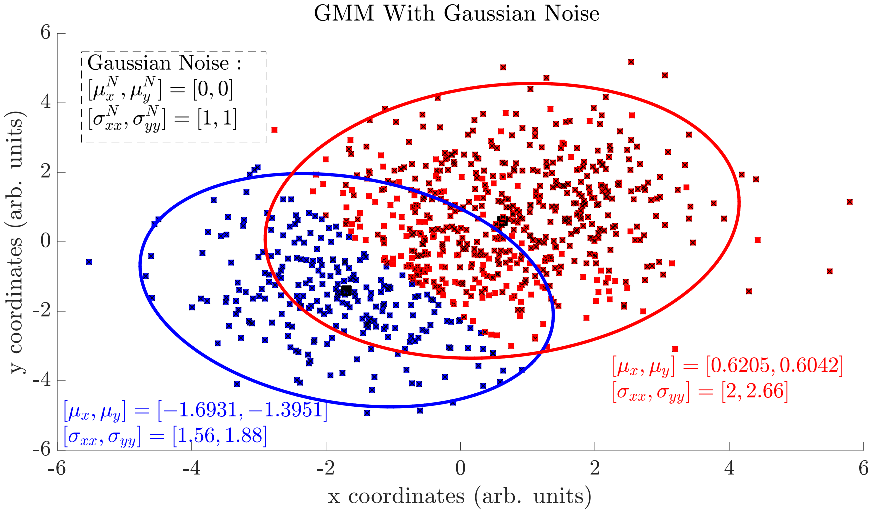

Figure 4.5: Clusters identified from the input test data using the GMM with Gaussian background noise algorithm. A shift in the means of the two clusters is observed, the magnitude of which seems to be dependent on the covariance of the noise.

# CHAPTER 5

# TOWARDS NEAR SENSOR COMPUTING SYSTEMS: PIPELINING THE GMM ALGORITHM

The integration of the genomic information that can be obtained from nanopore sensors to semiconductor nanotechnology, can be achieved when some the computing is carried out in hardware near the sensor. These rudimentary feature extractors and inference engines could potentially be used to perform basic classification such as identification of large proteins present along the DNA (which can discernibly be identified by large characteristic variations in currents compared to the background signal) or perform basic clustering and possibly identify the background cluster data from the dynamically changing features etc. The realization of all these features is possible in hardware using a pipelined version of the GMM algorithm, Algorithm 4.

**Algorithm 4** Pipeline-Friendly EM-GMM

```
While likelihood function does not converge do
$\eta \leftarrow 0, \tau \leftarrow 0, r \leftarrow 0, g \leftarrow 0, \rho \leftarrow 0$
    for $n \leftarrow 1$ to N
        $s \leftarrow 0$
        for $k \leftarrow 1$ to K do
            $g_k \leftarrow w_k \mathcal{N}(x_n|\mu_k, \Sigma_k)$
            $s \leftarrow s + g_k$
        for $k \leftarrow 1$ to K do
            $r \leftarrow \frac{g_k}{s}$
            $\eta_k \leftarrow \eta_k + r$
            $\rho_k \leftarrow \rho_k + r x_n$
            $\tau_k \leftarrow \tau_k + r x_n^2$
    for $k \leftarrow 1$ to K do
        $w_k \leftarrow \frac{\eta_k}{N}$
        $\mu_k \leftarrow \frac{\rho_k}{\eta_k}$
        $\sigma_k^2 \leftarrow \frac{\tau_k \eta_k - \rho_k^2}{\eta_k^2}$
```

The original EM-GMM algorithm does not fit into a fully-pipelined hardware design. This is because the data dependency in the original EM-GMM algorithm makes it impossible to stream the data only once in each EM iteration [20]. Hence, we study the implementation of the pipelined version of the GMM algorithm that was first proposed by Gao et al. [24] An important step here is to collect the data as each input stream enters the pipeline as opposed to the bulk collection of all parameters. The prior probabilities, weights, and expectations are collected along each iteration of the data. The pipelined version of the GMM algorithm is stated as Algorithm 4.

In the above algorithm, we see that the values of the prior responsibilities ($\gamma$) are collected via the variable $\eta$, which continually adds up the responsibility at each input data point ($r$). Similarly the expectations and variances are computed via the variables $\rho$ and $\tau$ respectively. To validate and implement the algorithm, we consider a synthetic data set consisting of three clusters of points centers at $\mu = 10, 20, 30$ with variances $\sigma = 1, 3, 3$ respectively. We implement Algorithm 4 and obtain a good match between the calculated clusters and the input clusters as shown in Fig 5.1. However, in order to truly develop a system architecture to be integrated into semiconductor nanotechnology, the calculations of the Gaussian terms need to be performed in fixed point and the algorithm will have to account for quantization errors due to a fixed-point implementation.

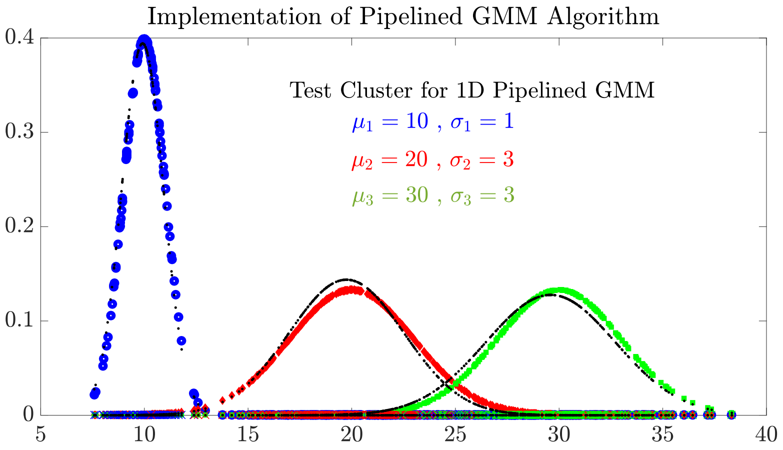

Figure 5.1: Verification of the pipelined version of the GMM algorithm with respect to a synthetic data set obtained from three clusters drawn from normal distributions colored in red, blue and green respectively. The black dots are the calculated clusters from the pipelined GMM algorithm.

# CHAPTER 6

# THEORETICAL RESOLUTION LIMIT

To measure the performance of a DNA sequencing algorithm, a theoretical resolution limit is useful for comparison. Detecting methylation sites in ionic and sheet current signals is very similar to detecting targets in reflected radar returns thus one possible solution is the ambiguity function which commonly used in radar signal processing [25]. The formula of ambiguity function is given as follows:

$$\chi(\tau, f) = \int_{-\infty}^{\infty} s(t)s^*(t-\tau)e^{j2\pi ft}dt. \tag{6.1}$$

The ambiguity function is a two-dimensional function of time delay $\tau$ and Doppler frequency $f$ showing the distortion of a returned pulse due to the receiver matched filter. Since the matched filter produces the maximum achievable instantaneous SNR at its output. Thus the ambiguity function is one reasonably good measure for the sequencing resolution. As for sequencing scenario, the Doppler frequency $f$ is analogous to the frequency of probe measuring the signal such as instantaneous ionic current value [25].

$$f = \frac{1}{\delta t}. \tag{6.2}$$

where $\delta t$ is the time interval of the probe measuring the signal. But further experiments are required to find the resolution limit of nanopore sequencing, what is the maximum tolerable distortion is still unknown.

# CHAPTER 7

# CONCLUSIONS AND FUTURE WORK

In this thesis, we discussed the development of a nanopore transistor device that would allow simultaneous control and sensing of the translocating DNA. The control layer was implemented using gold electrode which enabled the sculpting of the electrostatic landscape in order to reduce the stochastic fluctuations of the DNA. However, in order to perform inference on the system, the feature extraction is carried out using Gaussian mixture models. While most GMMs do not account for the inherent noise statistics or incompleteness of the data models, we have explored and implemented the modifications of the GMMs with uniform and Gaussian background noise. These models were validated using synthetic data, due to large computational complexity of the simulations of nanopore systems. The GMM model with uniform background noise is quite robust while the GMM with Gaussian noise seems to show deviations from the original mean, possibly due to convergence at local minima. We also explored the design of a pipelined GMM algorithm that could enable the design of near sensor inference engines.Finally, we propose one direction to find the resolution limit of nanopore sequencing.

The validation of the GMM with background noise models require more experiments on empirical data rather than synthetic data. From an algorithmic standpoint, more systematic studies need to be carried out with the GMM with Gaussian background noise models to further check their validity and correctness. One possible interesting future research problem is to use hidden Markov model to capture the dynamics of nanopore sequencing to provide a better sequencing algorithm.

# REFERENCES

[1] D. Branton, D. W. Deamer, A. Marziali, H. Bayley, S. A. Benner, T. Butler, M. Di Ventra, S. Garaj, A. Hibbs, X. Huang *et al.*, "The potential and challenges of nanopore sequencing," *Nature Biotechnology*, vol. 26, no. 10, pp. 1146–1153, 2008.

[2] J. J. Kasianowicz, E. Brandin, D. Branton, and D. W. Deamer, "Characterization of individual polynucleotide molecules using a membrane channel," *Proceedings of the National Academy of Sciences*, vol. 93, no. 24, pp. 13 770–13 773, 1996.

[3] A. H. Laszlo, I. M. Derrington, B. C. Ross, H. Brinkerhoff, A. Adey, I. C. Nova, J. M. Craig, K. W. Langford, J. M. Samson, R. Daza *et al.*, "Decoding long nanopore sequencing reads of natural DNA," *Nature Biotechnology*, vol. 32, no. 8, pp. 829–833, 2014.

[4] C. Dekker, "Solid-state nanopores," *Nature Nanotechnology*, vol. 2, no. 4, pp. 209–215, 2007.

[5] W. Humphrey, A. Dalke, and K. Schulten, "VMD: visual molecular dynamics," *Journal of Molecular Graphics*, vol. 14, no. 1, pp. 33–38, 1996.

[6] H. Qiu, A. Sarathy, K. Schulten, and J.-P. Leburton, "Detection and mapping of DNA methylation with 2D material nanopores," *npj 2D Materials and Applications*, vol. 1, no. 1, p. 3, 2017.

[7] A. Girdhar, C. Sathe, K. Schulten, and J.-P. Leburton, "Graphene quantum point contact transistor for DNA sensing," *Proceedings of the National Academy of Sciences*, vol. 110, no. 42, pp. 16 748–16 753, 2013.

[8] C. Sathe, X. Zou, J.-P. Leburton, and K. Schulten, "Computational investigation of DNA detection using graphene nanopores," *ACS Nano*, vol. 5, no. 11, p. 8842, 2011.

[9] F. Traversi, C. Raillon, S. Benameur, K. Liu, S. Khlybov, M. Tosun, D. Krasnozhon, A. Kis, and A. Radenovic, "Detecting the translocation of DNA through a nanopore using graphene nanoribbons," *Nature Nanotechnology*, vol. 8, no. 12, pp. 939–945, 2013.

[10] A. Sarathy, H. Qiu, and J.-P. Leburton, "Graphene Nanopores for Electronic Recognition of DNA Methylation," *The Journal of Physical Chemistry B*, 2016.

[11] C. Sathe, A. Girdhar, J.-P. Leburton, and K. Schulten, "Electronic detection of dsDNA transition from helical to zipper conformation using graphene nanopores," *Nanotechnology*, vol. 25, no. 44, p. 445105, 2014.

[12] F. Traversi *et al.*, "Detecting the translocation of DNA through a nanopore using graphene nanoribbons," *Nature Nanotechnology*, vol. 8, no. 12, pp. 939–945, 2013.

[13] H. Qiu, A. Sarathy, J.-P. Leburton, and K. Schulten, "Intrinsic stepwise translocation of stretched ssDNA in graphene nanopores," *Nano Letters*, vol. 15, no. 12, p. 8322, 2015.

[14] [Online]. Available: http://blog.booleanbiotech.com/nanopore_2016.html

[15] V. Boža, B. Brejová, and T. Vinař, "Deepnano: deep recurrent neural networks for base calling in MinION nanopore reads," *arXiv preprint arXiv:1603.09195*, 2016.

[16] H. Qiu, A. Girdhar, K. Schulten, and J.-P. Leburton, "Electrically tunable quenching of DNA fluctuations in biased solid-state nanopores," *ACS Nano*, vol. 10, no. 4, pp. 4482–4488, 2016.

[17] J. C. Phillips, R. Braun, W. Wang, J. Gumbart, E. Tajkhorshid, E. Villa, C. Chipot, R. D. Skeel, L. Kale, and K. Schulten, "Scalable molecular dynamics with NAMD," *Journal of Computational Chemistry*, vol. 26, no. 16, pp. 1781–1802, 2005.

[18] A. Sarathy and J.-P. Leburton, "Electronic conductance model in constricted MoS2 with nanopores," *Applied Physics Letters*, vol. 108, no. 5, p. 053701, 2016.

[19] P. Melchior and A. D. Goulding, "Filling the gaps: Gaussian mixture models from noisy, truncated or incomplete samples," *arXiv preprint arXiv:1611.05806*, 2016.

[20] M. Genovese and E. Napoli, "ASIC and FPGA implementation of the gaussian mixture model algorithm for real-time segmentation of high definition video," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 3, pp. 537–547, 2014.

[21] F. Leisch, "Modelling background noise in finite mixtures of generalized linear regression models," in *Proceedings in Computational Statistics (COMPSTAT 2008)*, Aug. 2008, pp. 385–396.

[22] P. W. Power and J. A. Schoonees, “Understanding background mixture models for foreground segmentation,” in *Proceedings Image and Vision Computing New Zealand*, vol. 2002, 2002, pp. 267–271.

[23] A. Ozerov, M. Lagrange, and E. Vincent, “GMM-based classification from noisy features,” in *International Workshop on Machine Listening in Multisource Environments (CHiME 2011)*, Sep. 2011.

[24] C. Guo, H. Fu, and W. Luk, “A fully-pipelined expectation-maximization engine for Gaussian Mixture Models,” in *Proceedings of the 2012 International Conference on Field- Programmable Technology (FPT)*, Dec. 2012, pp. 182–189.

[25] P. M. Woodward, *Probability and Information Theory, With Applications to Radar.* New York: McGraw-Hill, 1953.