Imperial College London

Department of Electrical and Electronic Engineering

MEng Final Year Project 2021

Spatio-Temporal Modelling and Algorithms
For Rapid Detection of Pathogenic DNA

Supervisor:
Dr. Pantelis GEORGIOU

Second Marker:
Dr. Timothy CONSTANDINOU

With special thanks to:
Mr Lewis KEEBLE
Dr Ahmad MONIRI
Dr Nicholas MOSER

Author:

Kunal KATARYA
01066347
EIE 4

July 7, 2021

Abstract

The need for rapid detection of pathogenic DNA, to test for the presence of a disease in a sample of bodily fluid, is more urgent in the current global climate than ever before. This project is concerned with a device, based on ion-sensitive field-effect-transistors (ISFETs), which is suitable for DNA detection in a Point-of-Care setting, due to its affordability, ease-of-use and accuracy.

The project aims to investigate whether the spatio-temporal aspects of the Loop-Mediated Isothermal Amplification (LAMP) reaction conducted in the device can inform the algorithms running on it.

To this end, the project makes two contributions to the field of PoC diagnostics. The first is a flexible and efficient mathematical model that is able to produce realistic datasets corresponding to the output of an ISFET-based DNA diagnostic device running a LAMP reaction - the first of its kind. Second, the project proposes two spatio-temporal algorithms that take advantage of the spatial arrangement of ISFETs on such a device to reduce its TTP by close to 33.2% compared to the existing temporal algorithm.

Together, these contributions provide teams in the field with useful tools and ideas to improve the performance of such diagnostic devices at a time of urgent global need.

Acknowledgements

First, I would like to thank my family for the sacrifices they have made to put me through 4 years of education at Imperial, a university I dreamt of attending for years. None of this would have been possible without their blessings, support and love.

My utmost gratitude also goes out to my friends, both in London and Singapore, who have been a constant source of inspiration and entertainment in good times and bad.

The guidance I have received from my superevisor Lewis Keeble, especially over the last few months, has been invaluable. Thank you for your patience and advice; I hope Wales make it past the Round of 16 in the next Euros.

Last but not least, a big thanks to Dr Pantelis Georgiou for letting me be a part of this exciting interdisciplinary project, which has exposed me to new ideas and stretched me intellectually.

Dedication

This work is dedicated to my late grandmother,

Mrs Kaushalya Kathuria
15th August 1929 - 7th May 2021,

who passed away during the horrific second wave of COVID-19 in India. It was a long-standing dream of hers to see me graduate, and I know she would have been proud of me today.

Rest In Peace Nanima.

Contents

1 Introduction **1**

1.1 Projct Motivation . . . 1

1.2 Problem Spcification . . . 2

1.3 Report Overview . . . 3

2 Literature Review **4**

2.1 DNA Amplification . . . 4

2.1.1 Polymerase Chain Reaction (PCR) . . . 5

2.1.2 Loop-Mediated Isothermal Amplification (LAMP) . . . 5

2.2 ISFET Technology . . . 5

2.3 TITANICKS Chip: ISFET Array . . . 6

2.4 CBIT Platform: LACEWING . . . 7

2.5 Signal Non-linearities . . . 7

2.5.1 Sensor Drift . . . 8

2.5.2 Thermal Noise . . . 8

2.6 Statistics . . . 9

2.6.1 Standard Deviation . . . 9

2.6.2 Cumulative Distribution Function (CDF) . . . 9

2.6.3 Probability Density Function (PDF) . . . 9

2.7 Signal Processing and Machine Learning . . . 10

2.7.1 Curve Fitting . . . 10

2.7.2 Correlation . . . 10

2.7.3 Machine Learning . . . 11

2.8 Mathematical Model . . . 11

2.8.1 Simulation of LAMP reaction . . . 11

2.8.2 Visualisation of simulation and model output . . . 14

2.8.3 Release of Protons . . . 14

2.8.4 Temporal signal . . . 14

2.8.5 Spatial signal . . . 15

2.8.6 Model Summary . . . 15

2.9 Chapter Summary . . . 16

3 Improving the Mathematical Model **17**

3.1 Exploration of experimental data . . . 17

3.2 Modelling the physical characteristics of TITANICKS . . . 19

3.3 Diffusion of protons . . . 21

3.4 Modelling proton capture . . . 22

3.5 Assumptions of improved model . . . 24

3.6 Threshold voltage output . . . 28

3.7 Overview of model functionality 30
3.8 Model Validation 32
3.8.1 Global trends of temporal signal 32
3.8.2 Impact of Chip Modelling and Diffusion: Spatial Signal 34
3.8.3 Impact of molecule position on spatial signal 36
3.8.4 Tuning of Burst Parameter for pH and threshold voltage change 38
3.8.5 Tuning Propagation Speed Parameter for Reaction Saturation Time 42
3.8.6 Effect of time step and chip frame rate on model output 44
3.9 Chapter Summary 49

4 Algorithms **51**
4.1 Methodology 51
4.2 Baseline Temporal Algorithm 52
4.3 Algorithm 1: Mean Drift Compensation + Spatial Standard Deviation 54
4.4 Algorithm 2: Individual Drift Compensation + Spatial Standard Deviation 58
4.5 Chapter Summary 62

5 Future Work **63**
5.1 Mathematical Modelling of LAMP Reaction 63
5.2 Spatio-Temporal Algorithms 65

6 Conclusions **67**

A Appendix A **68**
A.1 Codebase 68
A.1.1 Mathematical Model 68
A.1.2 Spatio-Temporal Algorithms 68

List of Figures

1.1 High level overview of LACEWING device [7] 2

2.1 Ideal Positive and Negative DNA Amplification Reaction Profiles 4
2.2 Stages of PCR, doubling of DNA molecules with every cycle [8] 5
2.3 Structure of an ISFET [10] 6
2.4 TITANICKS chip [12] 7
2.5 Presence of buried sites on ISFET surface [16] 8
2.6 Effect of each parameter of the drift equation [16] 8
2.7 PDF of Gaussian Random Variable [18] 10
2.8 Example of Curve Fitting using `curvee_fit` from `scipy.optimize`. 10
2.9 Block Diagram of training under Supervised Learning [21] 11
2.10 Geometry of solution (left) and representation in model (right) [22] 12
2.11 Growth of a single DNA cluster [22] 12
2.12 Growth of a multiple DNA clusters [22] 13
2.13 Removal of protons at cluster intersections [22] 13
2.14 Single Molecule Time-To-Positive for a particular DNA molecule [24] 13
2.15 DNA molecule initial positions and release of protons 14
2.16 Temporal signal obtained from simulation 15
2.17 Spatio-temporal signal: Total number of protons per sensor over time 15
2.18 Summary of Mathematical Model [22] 15

3.1 Output Signal of ISFET Array for Positive Experiment 17
3.2 Output Temporal Signal for Top Well (left) and Bottom Well (right) 17
3.3 Spatio-temporal signal: Existing Model output 18
3.4 Physical layout of ISFET sensor array 19
3.5 Debye Length (Detection distance) above ISFET [27] 20
3.6 3-dimensional representation of ISFETs 21
3.7 Creation of Permissible Coordinates of protons (coordinates of ISFETs) 22
3.8 Comparison of runtime of improved and existing models 23
3.9 A simulated random walk. Each colored line represents the path of a particle. . . . 24
3.10 Illustration of complexity of considering proton-cluster reflections 25
3.11 Removal of protons on collision with walls of manifold 26
3.12 Possibilities of proton-ISFET interactions despite protons falling in inter-ISFET gaps 27
3.13 Effect of number of detected protons on change in pH and threshold voltage. 9 protons detected initially, due to solution pH. 28
3.14 Temporal signal for positive sample from model 29
3.15 Spatial voltage output of improved model at different points in time 30
3.16 Parameters required for model 30
3.17 Flowchart of model 31
3.18 Trends of temporal signal in literature 32

3.19 Observed temporal signal from Mathematical Model 33
3.20 TTP reduces linearly with log-number of DNA molecules, R^2=0.952 33
3.21 Temporal signal and key points of comparison 34
3.22 Comparison of obtained spatial signals, side view 34
3.23 Comparison of obtained spatial signals, top view 35
3.24 Varying initial heights of DNA molecules 36
3.25 Spatial signals at time $t = 1100s$ for molecules of different heights 36
3.26 Standard deviation of spatial signal at $t = 1100s$ 37
3.27 Standard deviation of spatial signal across time 37
3.28 Change in pH and Voltage, Experimental Data (Malaria) 38
3.29 Impact of Burst parameter β on solution pH and voltage change 38
3.30 Histograms of pH changes over 30 experiments, varying β 40
3.31 Figure showing: (a) The high-level structure of a SiO_2 ISFET, (b) the distribution of ions on the surface of the ISFET, forming a Helmholz plane and Gouy-Chapman layer, (c) the spatial distributions of surface and diffuse charge, (d) the capacitances formed and (e) resultant changes in potential [25] 41
3.32 Effect of n and propagation speed on reaction saturation time. $\mu = 13$ minutes, $\sigma = 3$ minutes. 42
3.33 Effect of μ and propagation speed on reaction saturation time. $n = 100$ molecules, $\sigma = 3$ minutes. 43
3.34 Effect of τ on temporal signal, $n = 100$ molecules 44
3.35 Explanation for effect of τ on proton release 44
3.36 Effect of τ on number of protons released, $n = 100$ 45
3.37 Effect of τ on number of protons removed, $n = 100$ 45
3.38 Effect of τ on number of protons remaining in solution, $n = 100$ 46
3.39 Effect of F on Temporal Signal (the 4 curves overlap each other) 47
3.40 Histogram of Δt 48
3.41 Cumulative Distribution Function (CDF) of Δt 48
3.42 Effect of F on Difference in Noise Across Frames, Zoomed 49

4.1 General Algorithmic Strategy for Hotspot Detection 51
4.2 Spatial Signals from Test Dataset A 51
4.3 Temporal Output Signal, Dataset A 52
4.4 Percentage of Data Lying Within Specified Intervals of Normal Distribution [38] . 52
4.5 Drift-Compensated Temporal Signal 53
4.6 Derivative of Drift-Compensated Temporal Signal (30-sample rolling mean) 53
4.7 ISFET Frames at $t = 600$ seconds, before and after drift compensation 54
4.8 Change in ISFET Frame Standard Deviation Over Time, Dataset A 54
4.9 Change in ISFET Frame Standard Deviation Over Time, Dataset B 55
4.10 Signals for Hotspot Detection Using Standard Deviation 55
4.11 Signals for Hotspot Detection, Noisy Dataset 57
4.12 Output Frames at $t = 600$ seconds, before and after drift compensation 58
4.13 Change in ISFET Frame Standard Deviation Over Time, Dataset A 58
4.14 Change in ISFET Frame Standard Deviation Over Time, Dataset B 59
4.15 Difference in Per-Frame Standard Deviation, Algorithm 2 59

4.16 Gradient of difference in standard deviation signal 59
4.17 Hotspots of protons visible at $t = 384$ seconds 60
4.18 Algorithm 2: Signals for Hotspot Detection, Noisy Dataset 61

5.1 Inverse chi-squared distribution [40] 64

List of Tables

2.1 Key Metrics of the LACEWING Platform for detecting COVID-19 [7] 7

3.1 TITANICKS physical dimensions [12]. **The manifold is laser-cut and fitted on top of the ISFET array to create a reaction chamber - an imprecise process. These values were set as sensible averages on the advice of Dr Nicholas Moser, one of the designers of the chip.* 19

3.2 Parameters used for computation of Debye Length (Equation 3.1) [31] 20

3.3 Computational complexity of new model at each time step 23

3.4 Distribution of parameter values across sensors and experiments 29

4.1 Parameters of Datasets used 51

4.2 Distribution of parameter values, new dataset 56

1. Introduction

1.1 Projct Motivation

The COVID-19 Pandemic has claimed in excess of two million lives since it was declared a Public Health Emergency by the World Health Organization in January 2020 [1] [2]. The highly transmissible nature of the virus means that small clusters of cases can easily snowball into larger ones. As this process spirals out of control, countries are forced into lockdowns, adding economic and social problems to the already overwhelming health crisis. The currently accepted model for a solution to such a scenario is a robust Find-Test-Trace-Isolate-Support (FTTIS) system, which demands rigorous contact tracing (the process of finding all individuals who may have been infected by a new positive case), strict isolation and sound medical support [3]. This motivates the development of *faster* and *more accurate* testing systems for COVID-19, as well as other such pathogens. More accurate tests reduce the probability of undetected cases leading to clusters, while faster tests increase the speed of contact tracing and have even been used in some countries for mass testing before large scale events [4].

Testing for the presence of viruses - COVID-19, Malaria, Dengue, etc - in a sample of bodily fluid or tissue is a problem of biomarker detection. These biomarkers can be proteins, such as antigens or antibodies, or DNA molecules. Generally, tests for protein biomarkers - such as Rapid Antigen Tests for COVID-19 which involve loading a sample of bodily fluid onto a strip of paper that changes colour in the presence of the virus - are cheap and fast, but require high concentrations of the target biomarker for successful detection [5]. This reduces their accuracy. On the contrary, DNA-based tests can detect the presence of pathogenic DNA in a sample even at low concentrations. Such tests work by multiplying, or amplifying, the number of pathogenic DNA molecules in the solution and classifying a sample as positive when the number of DNA molecules passes a certain threshold. Samples with higher initial concentrations of the pathogenic DNA molecule will be classified as positive faster. Therefore, through the time-to-positive (TTP), such tests are also able to provide a quantitative measure of the amount of pathogenic DNA in a sample. The gold standard in DNA detection, in terms of accuracy, is the Polymerase-Chain-Reaction (PCR) test. However, PCR tests require large and expensive equipment operated by trained technical experts, which means that samples need to be transported to a laboratory after being extracted from an individual. Such tests also take a few hours to determine the presence of the virus in the sample [6].

The field of DNA-based Point-of-Care (PoC) diagnostics emerged to solve this problem. DNA-based PoC diagnostic tests aim to combine the affordability, speed and portability of protein-based tests with the accuracy and quantitative nature of PCR tests, bringing time and cost savings without sacrificing the quality of diagnostics provided. This project is concerned with a DNA-based PoC diagnostic device, developed by the Center for Bio-Inspired Technology (CBIT) at Imperial College London. The device, known as LACEWING, performs chemical amplification of pathogenic DNA molecules using the Loop-Mediated Isothermal Amplification (LAMP) reaction. Changes in DNA molecule concentration are detected using Ion-Sensitive-Field-Effect-Transistor (ISFET) technology [7]. The LAMP reaction is conducted in a 3-dimensional chamber mounted atop a 2-dimensional array of ISFET sensors. The array of sensors thus has the potential to detect not just the change in the number of DNA molecules in the reaction solution over time (known as temporal data), but also the spatial distribution of the DNA molecules in the solution (known as spatial data). It has already been deployed for the detection of Dengue, Malaria, HPV and most recently, COVID-19. This report discusses how the device works, presents a novel mathematical model that can be used to simulate its output, as well as demonstrates the potential for the use of spatio-temporal algorithms - algorithms that consider both the spatial and temporal variation in the number of DNA molecules - in improving its performance compared to purely temporal algorithms.

1.2 Problem Spcification

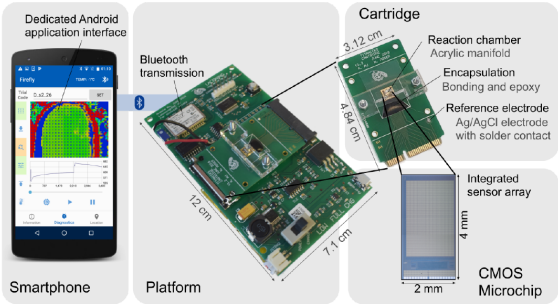


Figure 1.1: High level overview of LACEWING device [7]

This project aims to explore the spatio-temporal characteristics of the LAMP reaction running on the diagnostic device, known as LACEWING, developed by CBIT. Leveraging these characteristics in the implementation of spatio-temporal algorithms has the potential for improving the performance of the device, which could have significant real-world implications the current environment. The problem specification of this project may be written the following way:

To develop a simulator that accurately models the spatio-temporal characteristics of LAMP reactions in ISFET array-based DNA detection devices, followed by an algorithm that leverages these characteristics to reduces the time-to-detection of pathogenic DNA compared to temporal algorithms.

The primary challenge of this project lies in the non-ideal behaviour of the ISFET sensors. Noise corrupts the output signal of each snsor, thereby necessitating the use of signal processing and other statistical techniques to classify the sample being tested as positive or negative for the presence of a particular pathogen. A *spatio-temporal algorithm* is one that takes into account both the time- and space-varying properties of the output signal, instead of just the temporal data (as the currently implemented algorithm on LACEWING does). The development of such an algorithm requires a vast amount of data for training and development. To this end, there exists a mathematical model that is able to generate synthetic datasets which mirror the theoretical output signals of an ISFET array. Improving the model to produce more realistic (noisy) data is therefore a pre-requisite to developing a spatio-temporal algorithm that will be capable of running in a noisy PoC diagnostics setting. The aims of this project can therefore be summarized as:

1. Improve the existing mathematical model to generate more realistic data
2. Develop *offline* (not running on LACEWING) spatio-temporal algorithms
3. Evaluate the offline algorithms based on time-to-detection compared to a baseline temporal algorithm, on datasets generated from the mathematical model

MATLAB will be used for the mathematical model and Python 3 (on Jupyter Notebooks) for the algorithms.

1.3 Report Overview

This report contains six chapters. Chaper 1 introduces the context of the project, stressing its importance in the current climate and describing the specific problem statement being addressed. Chapter 2 provides the reader with the necessary background information in biology, electronics, chemistry and mathematics required to understand the technical content of this project, as well as contextual information about past work in the field that will be used in this project. This information is synthesized in Chapter 3, which explains the improvements made to a mathematical model for the LAMP reaction in order to generate more realistic datasets for algorithm development. Model validation and experimentation is also detailed in this chapter, drawing key insights about the reaction and opening up further avenues for investigation. Chapter 4 follows with an introduction and analysis of two novel spatio-temporal algorithms which show promising results on synthetic datasets generated from the mathematical model. Finally, Chapters 5 and 6 bring the report to a close by highlighting future extensions to the project and providing a concise summary of all the work presented in this report, respectively.

2. Literature Review

This background section contains pre-requisite reading on DNA detection techniques, ISFETs, the CBIT LACEWING platform, signal processing and statistics as well as the mathematical model that simulates the behaviour of the platform.

2.1 DNA Amplification

DNA amplification is the foundation behind the process for determining whether a sample of fluid collected from a human being contains a certain target infecction. It is defined as the production of multiple copies of a sequence of DNA, or the repeated copying of a piece of DNA. This is achieved by placing the obtained sample in a chemical solution which contains a particular sequence of bases. These bases only react with a specific target DNA, causing it to multiply.

The multiplication of the target DNA creates by-products; one of these is hydrogen ions, or protons. The simplified chemical equation governing this process is given in Equation 2.1 as follows:

$$DNA + dNTP \longrightarrow 2 \cdot DNA + Pyrophosphate(PP_i) + \beta \cdot Proton(H^+) \tag{2.1}$$

Therefore, the greater the number of copies of a particular target DNA molecules within a sample, the higher the concentration of Hydrogen ions in the solution. The concentration of these ions can be quantified by measuring the pH of the solution:

$$pH = -log_{10}\left(\left[H^+\right]\right) \tag{2.2}$$

Therefore, though the number of copies of DNA cannot be observed directly, Equation 2.2 implies that a temporal reduction in pH should be observed if the target DNA is present within the obtained sample. The ideal reaction profile of Equation 2.1 can be modelled as a logistic function. A *positive* reaction is one in which the sample contains the target DNA; in this case, the graph of concentration of Hydrogen ions in the solution against time takes a sigmoidal shape. Conversely, if the sample does not contain the target DNA, the reaction is termed *negative*, and the concentration of Hydrogen ions remains constant. The general equation of the logistic function is shown below:

$$\sigma(t; a, b, c) = \frac{a}{1 + \exp(-b(t - c))} \tag{2.3}$$

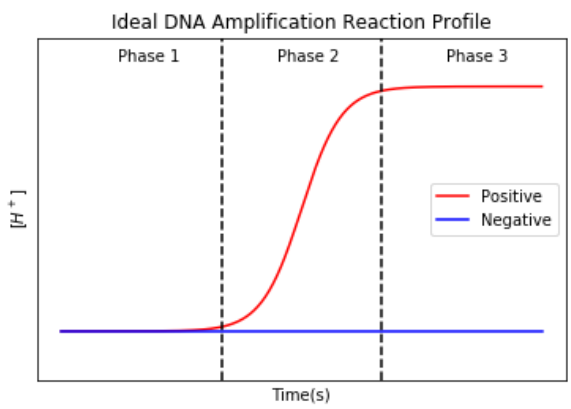


Figure 2.1: Ideal Positive and Negative DNA Amplification Reaction Profiles

The ideal reaction profile is represented graphically in Figure 2.1. During Phase 1 of the positive reaction, the target DNA molecules and complementary bases in the solution are yet to

interact. Phase 2, characterized by an exponential increase in Proton concentration, shows that the amplification reaction is taking place. Finally, in Phase 3, all complementary base molecules in the solution have been used up and the reaction has saturated.

2.1.1 Polymerase Chain Reaction (PCR)

The most commonly used amplification reaction, PCR, is performed over the course of 25-33 cycles, where each cycle consists of denaturation, primer annealing and primer extension. The number of DNA molecules doubles every cycle, as shown in Figure 2.2.

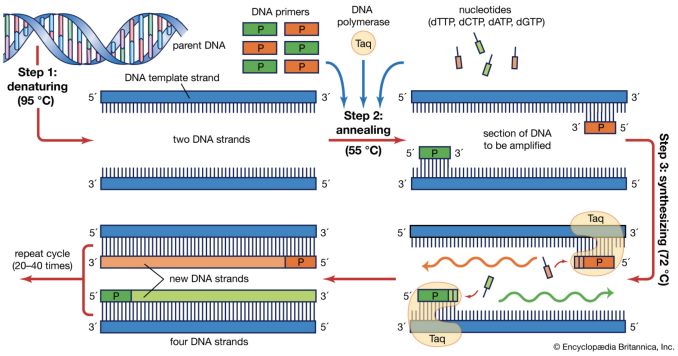


Figure 2.2: Stages of PCR, doubling of DNA molecules with every cycle [8]

PCR requires large and expensive heating and optical equipment to keep the reaction at a particular temperature for each stage of the cycle as well as monitor for changes in flourescence. Furthermore, a typical reaction takes between 2-4 hours to complete [6]. These drawbacks make PCR unsuitable for PoC diagnostics.

2.1.2 Loop-Mediated Isothermal Amplification (LAMP)

An alternative to PCR, LAMP is a low-cost technique for DNA amplification. Carried out at a constant temperature of 60-65°C, the LAMP reaction has the desirable characteristics of high specificity and sensivity, as well as rapid reaction times. Stages of the LAMP reaction occur in parallel and are more complicated than that of the PCR - the extra complexity in the form of synthesizing reagents and difficulty in designing the reaction are the major drawbacks of the technique. However, fact that its speed and affordability compared to PCR make make it the right fit for PoC applications [9].

2.2 ISFET Technology

The Ion-Sensitive Field-Effect Transistor (ISFET) forms the foundation of the PoC diagnostic device used in this project. The structure of a typical ISFET is shown in Figure 2.3 below.

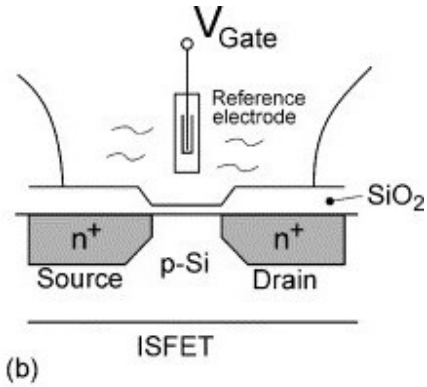


Figure 2.3: Structure of an ISFET [10]

A chemical solution which contains reactants specific to the target DNA can be deposited on the gate of the ISFET. With a passivation layer of Al_2O_3 or Si_3N_4, the transistor becomes sensitive to the concentration of Protons in the chemical solution. As the amplification reaction proceeds, the concentration of Protons increases and the pH of the solution decreases. The threshold voltage of the ISFET is then reduced according the following equations [11]:

$$V_{th(ISFET)} = V_{chem} + V_{th(MOSFET)} \tag{2.4}$$

$$V_{chem} = \gamma + 2.3\alpha U_t pH \tag{2.5}$$

$$\Delta V_{th(ISFET)} = 2.3\alpha \frac{kT}{q} \Delta pH \tag{2.6}$$

where α is the Nernst potential of $59.6mV/pH$ at the ambient temperature (63^oC). The change in pH, ΔpH is computed by dividing the number of protons detected by each ISFET by the volume of the region around the ISFET surface (further elaborated on in Section 3.2) and converting the obtained concentration (in molecules/m^3) to moles/Litre. Additionally, assuming protons are distributed homogeneously throughout the solution, the initial number of protons within the relevant region is computed using the initial pH (8.75) and added to the number of protons persisting on the array.

2.3 TITANICKS Chip: ISFET Array

The CBIT PoC device utilizes an array of ISFETs, embedded in a chip, to leverage the power of signal processing methods. The chip, called TITANICKS, consists of 4368 transistors arranged in a 78x56 array. 494 of these are MOSFETS used to monitor the temperature of the chip, while the remaining 3874 are ISFETs which are sensitive to pH changes. The TITANICKS chip sits directly beneath a chemical solution that causes amplification of a target pathogenic DNA, producing protons as a by-product. The sample of bodily fluid to be tested for the pathogenic DNA is first extracted from the person being tested, processed (details are outside the scope of this project) and then injected into the chemical solution atop the TITANICKS chip.

Therefore, at any given point in time, each of the ISFETs outputs a particular value corresponding to the release (or not, if no target DNA is present in the sample) of protons by the reactants. The rectangular arrangement of the transistors means that these output values form an 'image'; thus, each transistor is interchangeably referred to as a pixel. The arrangement of the pixels is shown in Figure 2.4

Using the Column Logic and Row Logic, the Digital Core is able to select and read the values of a specific pixel. The output of the chip, containing the values of each pixel, is sent through the Serial Peripheral Interface. The sampling interval is approximately 3 seconds; i.e every 3 seconds, the chip produces a new *frame* consisting of the output voltage of individual pixels.

The collection of all frames obtained throughout an experiment is therefore a *spatio-temporal signal* - a series of voltages that change over time and (potentially) display certain spatial char-

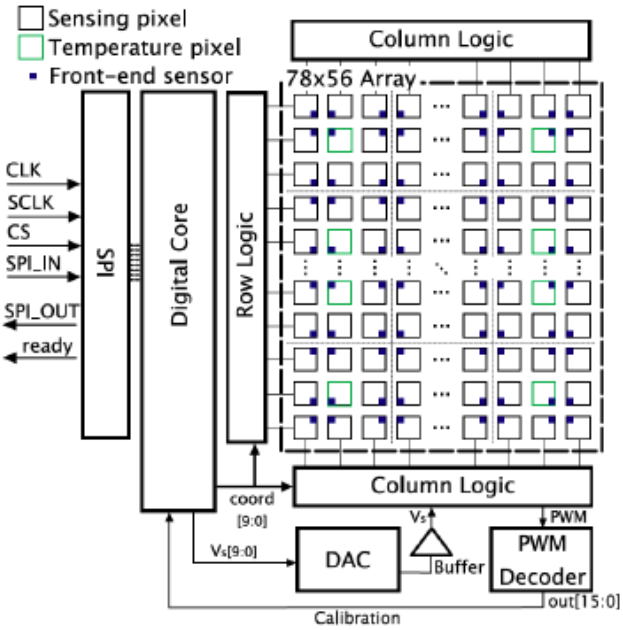


Figure 2.4: TITANICKS chip [12]

acteristics as well. The focus of this project is on analysing this spatio-temporal signal to quickly and accurately diagnose a particular sample as positive or negative for a target pathogenic DNA.

2.4 CBIT Platform: LACEWING

The LACEWING Platform, developed by CBIT at Imperial College London, houses the TITANICKS chip and is being used in the field for the detection of DNA. Since the start of the outbreak, the platform has been adapted to work as a test for COVID-19, and has obtained impressive results as shown in Table 2.1.

Metric	Explanation	Value
TTP (time-to-positive)	Time taken to diagnose a sample as positive for COVID-19	<20 minutes
Sensitivity	True positive rate - the number of correctly identified positive cases over the total number of infected individuals tested	90.55%
Specificity	True negative rate - the number of correctly identified negative cases over the total number of uninfected individuals tested	100%

Table 2.1: Key Metrics of the LACEWING Platform for detecting COVID-19 [7]

2.5 Signal Non-linearities

The most pressing challenge in working with ISFET technology is the noise in sensor output. This noise distorts the obtained signal and significantly increases the complexity of the decision-making algorithm. There are numerous sources of noise: sensor drift, which is elaborated on in this section; trapped charge, which induces an offset in the sensor due to charge transfer during production; changes in temperature, which affect the drain-to-source current of the ISFET [13]; flicker noise (also known as $1/f$ noise) [14], caused by the release and capture of charge at high frequencies and inherent thermal noise. Of these, the three types of noise with the most significant effect on signal output need to be compensated for. These are sensor drift, flicker noise and thermal noise.

2.5.1 Sensor Drift

Sensor drift is a phenomenon in which the output of the sensor changes independently to the reaction it is measuring, arising due to a gradual change in threshold voltage over time [15]. Specific to TITANICKS, sensor drift has been shown through prior work to be a consequence of *buried sites* of $SiOH$ within the SiO_2 semiconductor layer of the ISFET. This is shown in Figure 2.5.

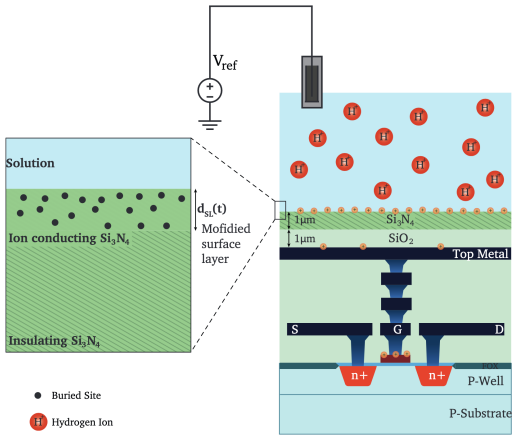


Figure 2.5: Presence of buried sites on ISFET surface [16]

These buried sites form regions of trapped charge (positive or negative) on the surface of the ISFET. When in contact with the electrolyte, these buried sites cause the diffusion of hydrogen ions into the surface of the ISFET, causing a build-up of charge density within the insulating layer over time. The diffusion coefficient of hydrogen ions onto the surface reduces with time (a *time-relaxation model*). Prior work [16] has shown that sensor drift and flicker noise can be modelled as a decaying exponential of the following form:

$$\Delta V = Q' \left[1 - e^{-\left(\frac{t}{\tau}\right)^{\beta}} \right], \tag{2.7}$$

where Q' controls the ending voltage as $t \to \infty$, τ controls the half-life and β the speed of gradient change.

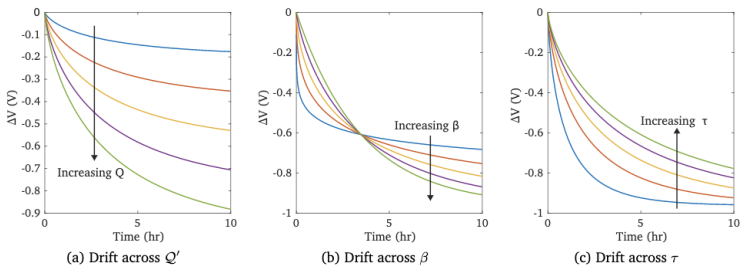


Figure 2.6: Effect of each parameter of the drift equation [16]

This drift can be partially corrected for using signal processing techniques [17] such as curve fitting, described in subsequent sections.

2.5.2 Thermal Noise

Thermal noise, caused by random thermal excitation of charge carriers, can be modelled as Additive White Gaussian Noise (AWGN). This noise has three main properties:

1. Additive: the noise is added to the signal

2. White: the noise has a uniform frequency spectrum across all frequencies

3. Gaussian: the noise has a zero-mean normal distribution in the time domain

2.6 Statistics

2.6.1 Standard Deviation

The standard deviation quantifies of the amount of variation in a set of datapoints. Given a set of N datapoints $X_1, X_2, ..., X_N$, the mean of the dataset is defined as:

$$\mu = \frac{\Sigma X_i}{N} \tag{2.8}$$

The variance of the dataset, σ^2, is the averaged squared deviation of each datapoint from the mean. Raising differences to the power of two ensures that positive and negative differences do not cancel each other out.

$$\sigma^2 = \frac{\Sigma_{i=1}^{N} (X_i - \mu)^2}{N} \tag{2.9}$$

The standard deviation σ is the square root of variance, providing a measure of the average absolute spread of the data from its mean:

$$\sigma = \sqrt{\frac{\Sigma_{i=1}^{N} (X_i - \mu)^2}{N}} \tag{2.10}$$

2.6.2 Cumulative Distribution Function (CDF)

Given a real valued continuous random variable X, the CDF ($F_X(x)$) is a function that, evaluated at a particular value $X = x$, is the probability that the random variable X will be less than or equal to x.

$$F_X(x) = P(X \leq x) \tag{2.11}$$

2.6.3 Probability Density Function (PDF)

The PDF $f_X(x)$ of a random variable X is computed by taking the derivative of its CDF at all values x. The PDF of a random variable can be interpreted as the relative likelihood that the variable X drawn from a sample will fall within a particular range of values.

An important PDF for this project is the Gaussian Distribution, parameterized by a mean μ and a varianc σ^2. The equation of the PDF as well as its shape are shown in Equation 2.12 and Figure 2.7 respectively.

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \tag{2.12}$$

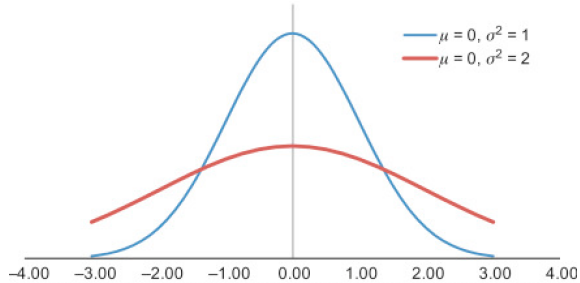


Figure 2.7: PDF of Gaussian Random Variable [18]

2.7 Signal Processing and Machine Learning

Signal Processing and Machine Learning are two mechanisms that can compensate for the distortions caused by Sensor Drift and flicker noise. These mechanisms can also form the foundation of algorithms that attempts to analyse the spatio-temporal signal to classify it as positive or negative for a particular pathogen.

2.7.1 Curve Fitting

Curve fitting is the process of fitting an arbitrary mathematical function to a set of data points. This process can be performed via the `curve_fit` function available in the `scipy.optimize` library. The routine takes a parameterized mathematical as input, together with the raw data (x, y) to fit to. It uses least-squares minimization to adjust the numerical values of each parameter in the function, producing a curve that most closely matches the data provided.

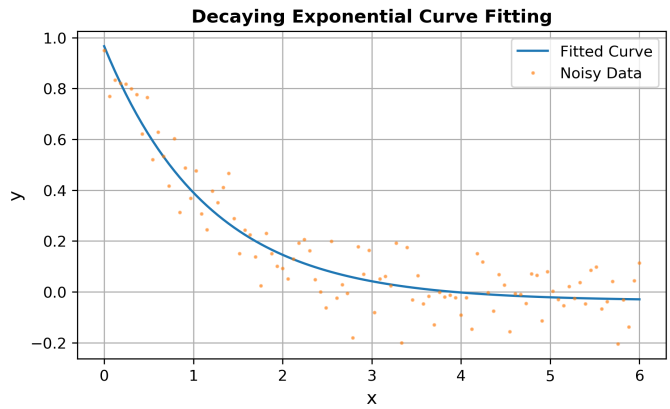


Figure 2.8: Example of Curve Fitting using `curvee_fit` from `scipy.optimize`.

Figure 2.8 shows an example of curve fitting applied to a dataset of points arranged in a decaying exponential, corrupted with noise. The fitted curve can then be used for extrapolation - predicting future values of y for every value of x. Such a technique can be used to compensate for sensor drift throughout the LAMP reaction.

2.7.2 Correlation

One possible technique for extracting spatial information from the output of the array is calculating the correlation coefficient. If the output of each pixel of the ISFET array is considered a continuous variable, the correlation coefficient between the outputs of multiple pixels provides information about the association, or similarity, between them [19]. The formula of the correlation coefficient between two variables x and y is shown in Equation 2.13.

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}} \tag{2.13}$$

As DNA amplifies in the chemical solution, it releases protons in its immediate neighborhood. These protons affect the output of the ISFETs beneath them. Computing the correlation coefficient between every pixel and its immediate neighborhood might reveal that outputs in certain localized regions of the ISFET array are rising in tandem, which could be indicative of an amplification reaction occurring.

2.7.3 Machine Learning

Machine Learning algorithms learn to perform a specific task without being explicitly 'taught' (through hard-coded rules) how to do so. A subset of machine learning, known as supervised learning, is the process of training the algorithm on large amounts of labelled data, during which time it updates its internal parameters [20]. This process is shown in Figure 2.9. The algorithm is then tested on unseen data and its performance is evaluated against the ground truth labels for the test dataset.

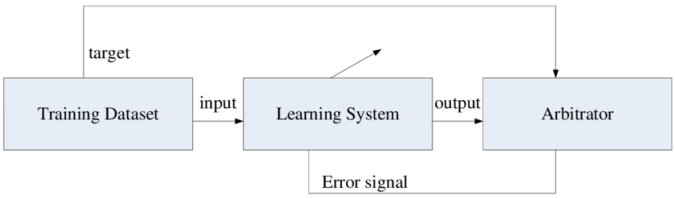


Figure 2.9: Block Diagram of training under Supervised Learning [21]

In this project, each frame of the output of TITANICKS may be considered a training example. The task is a 'binary classification' problem - the learning system needs to classify each training example as positive or negative for the pathogenic DNA in question. Labelled data, through experiments and synthetically generated datasets, is available for training the Machine Learning algorithm on. The algorithm can then be evaluated on data from experiments (or synthetically generated data) that were not used previously. A variety of Machine learning models can be explored for use as part of the spatio-temporal algorithm to be developed - these include (but are not limited to) Support Vector Machines, Random Forests, Neural Networks, etc.

2.8 Mathematical Model

The development of spatio-temporal algorithms for classifying a sample as positive or negative requires a large amount of experimental data to train and learn from. The process of obtaining this data involves extracting real samples taken from patients and injecting them into a physically manufactured TITANICKS ISFET array - which is costly and time consuming. The data obtained varies drastically across experiments as it depends on numerous factors including the concentration of DNA molecules in the sample, the type of pathogenic DNA being tested for, the viscosity of the chemical solution, the size of the ISFET array etc.

A mathematical model - a set of MATLAB scripts - was thus developed to generate synthetic datasets, taking the above variables (and others) into account. The model, the first of its kind, achieves this by simulating the local kinetics of the amplification reaction in 3-D space [22].

2.8.1 Simulation of LAMP reaction

The model is built based on a set of explicitly coded fundamental characteristics of DNA amplification reactions. However, its output, which displays the same trends observed in literature and experiments, is dependent on the interactions of various stochastic objects in software.

A number of assumptions regarding the kinetics of the reaction were made during the implementation of the model such that the simulation achieved a desirable trade-off between physical accuracy and computational complexity. This section will highlight these assumptions and place them in the context of the simulation of the reaction.

1: Geometry of chemical solution and initial position of DNA molecules

First, the geometry of the chemical solution of the ISFET array is modelled in code. The solution is assumed to take the shape of a cuboid - discounting the meniscus that it forms with the walls of the reaction chamber. Given a known number of ISFETs in the x and y directions (x_{max}, y_{max}) as well as a fixed solution volume $(V = 40\mu L)$, the height of the solution (Z_{max}) is computed. When a sample of bodily fluid is injected into the solution, the target pathogenic DNA molecules (if present) are assumed to be uniformly distributed amongst each of the three dimensions. The setup is shown in Figure 2.10 below.

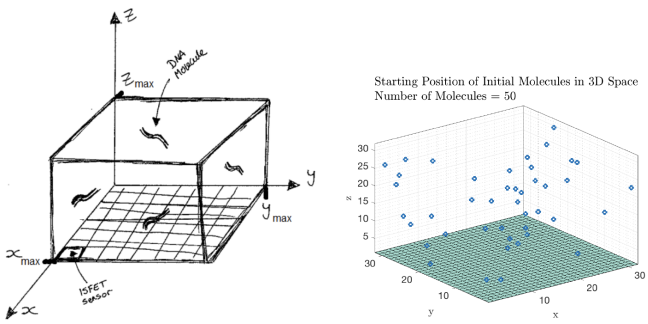


Figure 2.10: Geometry of solution (left) and representation in model (right) [22]

2: Clusters and motion of DNA molecules

The reaction (chemical equation shown in 2.1) is modelled by considering *clusters* of DNA molecules growing in 3-D space and releasing protons around their surface area. The edge of the cluster - the *propagation front* - is found to be independent of the DNA molecule concentration and propagates at a nearly constant speed of $1.73 \pm 0.15\mu s^{-1}$ [23] in 1-D. In 3-D space, this translates to a cluster growing spherically. This is shown in Figure 2.11 below.

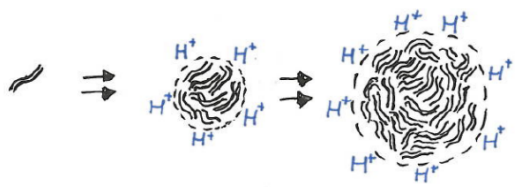


Figure 2.11: Growth of a single DNA cluster [22]

The rate of proton production by a DNA cluster is assumed to be proportional to its surface area. The number of protons to be released around a cluster between two time instants t_1 and t_2 is thus given as:

$$\text{Number of protons} = \beta \times \frac{4}{3}\pi \left(r^3_{current} - r^3_{previous}\right) \tag{2.14}$$

$$= \beta \cdot \Delta v, \tag{2.15}$$

where β, the burst parameter, is a constant, and Δv is the change in volume of the cluster between t_1 and t_2. Protons are released every τ seconds, where τ is a parameter supplied by the user corresponding to a time step in the simulation (usually equal to the sampling frequency of the chip), and the simulation runs for t_{total} seconds, also supplied by the user.

As multiple clusters grow, the interaction between clusters becomes important to characterize. Once a DNA molecule has amplified and used up the reactants in the region of space it occupies, amplification can no longer occur in that region. Hence, when clusters intersect, protons are not produced in the regions of intersection. This is depicted in Figure 2.12 below. This is implemented in the model by releasing protons for each cluster, and then removing protons that fall within the radius of other clusters. Similarly, protons produced outside the set of permissible coordinates (outside the walls of the reaction chamber) are also removed.

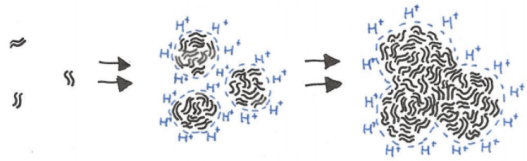


Figure 2.12: Growth of a multiple DNA clusters [22]

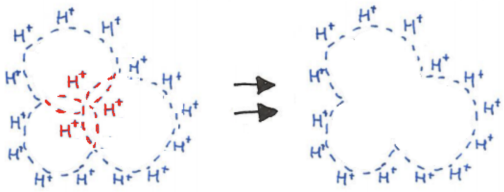


Figure 2.13: Removal of protons at cluster intersections [22]

3: Single molecule time-to-amplification

The time at which each cluster begins to grow depends on the pathogenic DNA being considered. The single molecule behaviour of the LAMP reaction refers to the time-to-amplification observed when a single DNA molecule of a particular pathogen is present in the reaction. A histogram, such as that shown in Figure 2.14, can be constructed after running single-molecule experiments with a particular pathogen. Normalising and converting the histogram to a PDF, it can be modelled using a Normal Distribution, which provides useful information about the mean and variance (μ and σ^2) of the time taken for a single molecule to begin amplification. This distribution is sampled for each of the molecules present in the reaction to determine when their clusters start propagating.

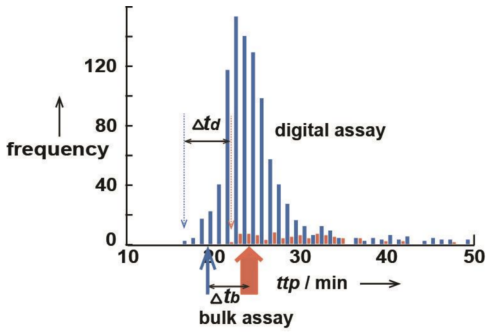


Figure 2.14: Single Molecule Time-To-Positive for a particular DNA molecule [24]

4: Motion of protons

Instead of modelling the individual motion of protons to determine how many protons land on each sensor, the simulation uses a statistical approximation to reduce computational complexity. Assuming that a proton is equally likely to land on any of the sensors of the ISFET array, the number of protons landing on a single sensor is given by a sum of Bernoulli random variables:

$$Y = \sum_{i=1}^{n} X_i, \text{ where } X_i \sim Bernoulli\left(\frac{1}{MN}\right), \tag{2.16}$$

where M and N are the number of ISFETs along the width and length of the array respectively. Assuming n protons in the solution at a given moment, the number of protons on a single sensor is drawn from the following random variable (proof in [22]):

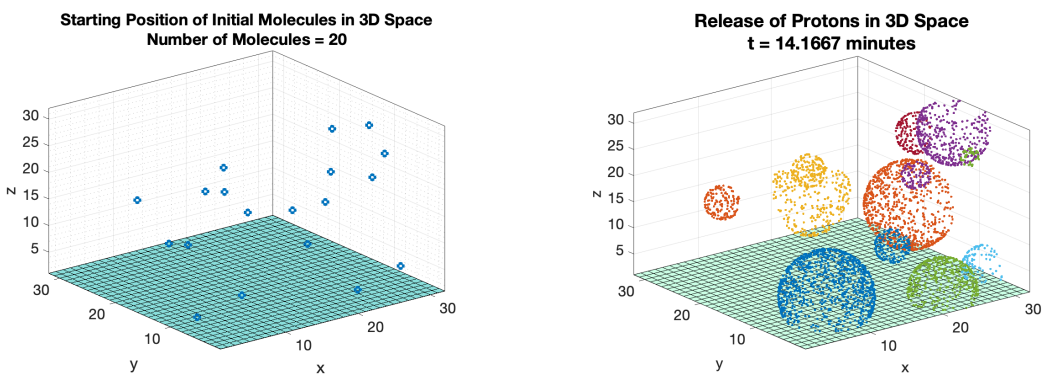
$$Y \sim N\left(\frac{n}{MN}, \frac{n}{MN}\left(1 - \frac{1}{MN}\right)\right) \tag{2.17}$$

2.8.2 Visualisation of simulation and model output

The progress and output of the reaction can be visualized in three ways. During the reaction, the release of protons is plotted on a 3-dimensional grid representing the reaction chamber. Equation 2.16 is also used during the reaction to record the number of protons on each sensor at each time step. This data can be visualized temporally and spatially.

2.8.3 Release of Protons

As previously explained, the starting x, y and z positions of DNA molecules are drawn from uniformly distributed random variables - shown in Figure 2.15 (left). The time-to-amplification of each molecule is drawn from a Normal distribution specific to the target pathogen in question. As the simulation proceeds, each molecule grows into a cluster when its time-to-amplification is reached, releasing protons around its surface. Protons falling outside the walls of the reaction or within other clusters are removed. This is shown in Figure 2.15 (right).



(a) Initial positions of DNA molecules

(b) Release of protons

Figure 2.15: DNA molecule initial positions and release of protons

2.8.4 Temporal signal

At each time step, Equation 2.16 is used to determine how many protons are detected by each sensor. Figure 2.16 is a plot of the number of protons detected per sensor, together with the mean across all sensors, against time. The obtained logistic curve follows the expected profile of $[H^+]$ in LAMP reactions, as shown in Figure 2.1. It is worth noting that no explicit global behaviour was encoded in the simulation - the obtained reaction profile is the result of the interaction of clusters, release of protons, and projection of protons onto the sensor array.

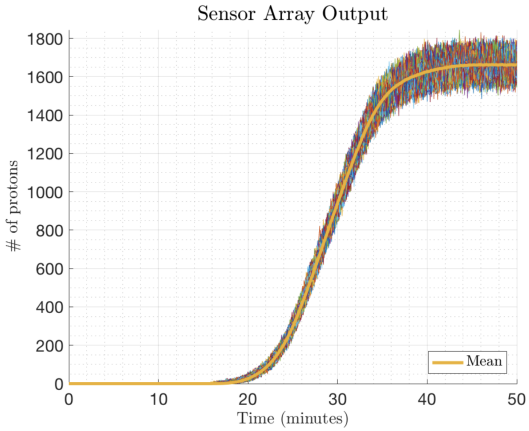
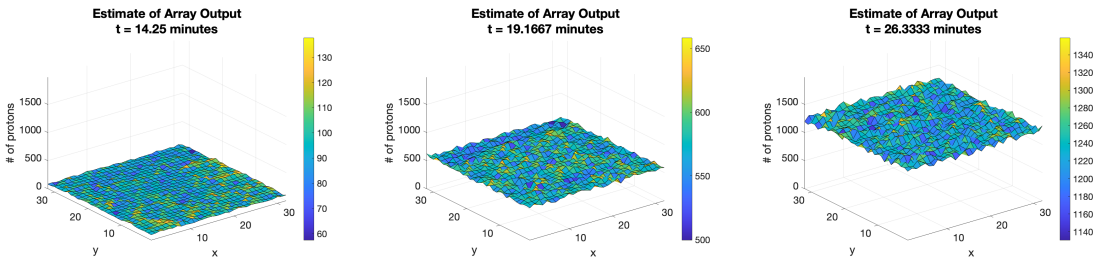


Figure 2.16: Temporal signal obtained from simulation

2.8.5 Spatial signal

Finally, the simulated spatial signal, which shows the cumulative number of protons per sensor changing with time, is shown in Figure 2.17. Changing plot settings to display the signal from a top-down view, instead of a side-on view as shown in Figure 2.17, results in the signal looking like a square image. As such, the spatial signal at each point in time will be interchangeably referred to as a frame. Spatio-temporal algorithms would look for patterns within and across frames to try and classify a sample as positive or negative, as opposed to temporal-only algorithms that seek to classify a sample based on the temporal signal of the reaction matching the expected sigmoidal shape of amplification.



(a) t=14.25 minutes (b) t=19.17 minutes (c) t=26.33 minutes

Figure 2.17: Spatio-temporal signal: Total number of protons per sensor over time

2.8.6 Model Summary

The high-level functionality of the model is shown in Figure 2.18.

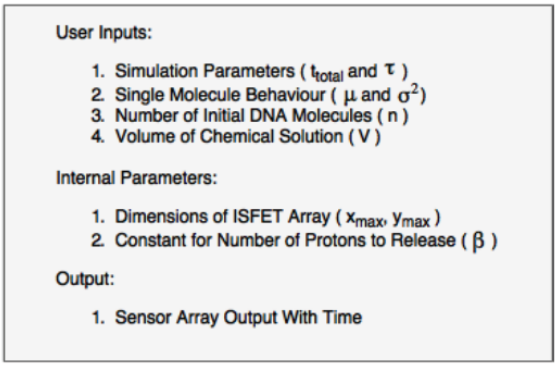


Figure 2.18: Summary of Mathematical Model [22]

2.9 Chapter Summary

This chapter began by introducing the process of DNA amplification and comparing LAMP reactions to the gold standard PCR, explaining the advantages LAMP brings in a PoC setting - speed and affordability. Running LAMP on an ISFET array in the TITANICKS chip allows for the detection of pH changes in the reaction chamber. As the concentration of protons in the reaction chamber rises, the temporal output of the ISFET array is expected to follow the shape of a sigmoid function, allowing for classification of the sample being tested as positive. However, sensor drift and flicker noise corrupt the output signal, necessatating the use of signal processing and statistical techniques, such as curve fitting, to compensate for these non-idealities. Last, an explanation of the existing mathematical model for LAMP was provided, together with a visualisation of its spatial and temporal outputs. The next chapter focuses on improvements to the mathematical model.

3. Improving the Mathematical Model

3.1 Exploration of experimental data

Figure 3.1 shows a visualisation of the raw output of the ISFET array at the 5th, 400th, and 800th frames of the amplification reaction, reshaped into the dimensions of the TITANICKS chip (78x56).

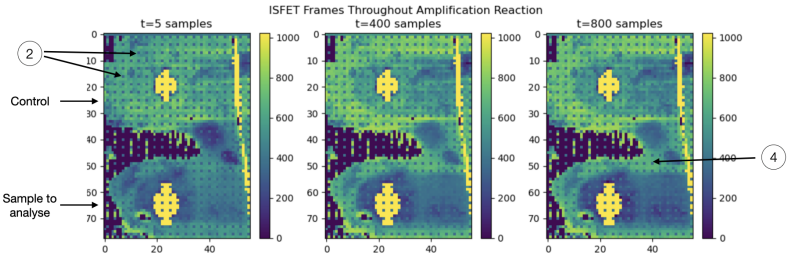


Figure 3.1: Output Signal of ISFET Array for Positive Experiment

A number of observations shed some light on the characteristics of the ISFET array and LAMP reaction. These are explained below, with corresponding annotations provided on the figure.

First, each frame contains two distinct 'lobes'. This is due to the experiment being carried out in a *double well* topology, in which two samples are loaded onto the same chip. The upper half of the chip (above the line y=40) acts as a control and contains a negative sample, while the lower half contains the sample to analyse. The manifold containing the chemical solution for the amplification reaction covers most of the chip, with the exception of the pixels around the $y = 40$ area, which explains why their voltage output is close to zero throughout the experiment - no reaction occurs there.

Second, blue pixels with unchanging output are observed at regular intervals. These are the 494 periodically placed MOSFETs for temperature regulation, label 2 in the figure. Their output is not useful to the spatial signal as it is independent of proton release. This can be remedied by setting their output at each time step to be the average of the eight ISFETs surrounding them.

Third, a general decrease in pixel outputs over time can be observed in both wells. This is indicative of sensor drift, a phenomenon whereby the threshold voltage of an ISFET gradually decreases independent of a change in pH, further explained in Section 2.5.

Last, circular regions of higher voltage are seen developing in the bottom well. These regions of higher activity (label 4 in the figure) correspond to the release of protons from clusters being amplified in the solution above.

Next, the temporal signal of another experiment containing a sample positive for SARS-CoV-2 is plotted in Figure 3.2.

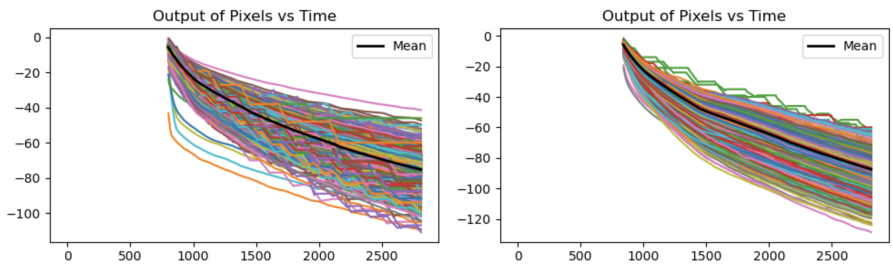


Figure 3.2: Output Temporal Signal for Top Well (left) and Bottom Well (right)

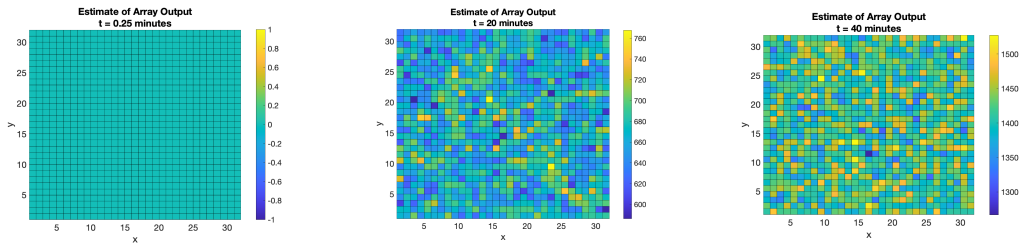
As with the spatial signal, a number of observations can be made about the temporal signal.

First, both obtained plots exhibit the signal non-linearities described in Section 2.5. Individual ISFET outputs are plotted as coloured lines, while the mean is shown in black. Sensor drift, in the form of a decaying exponential, is seen in both the negative and positive samples. The output voltage of each pixel also exhibits high frequency noise, despite this plot being the a rolling average of 30 samples. Each ISFET exhibits unique sensor drift and drift is inconsistent across both wells (i.e it is not possible to entirely characterize the drift of the bottom well by observing that of the top).

Second, the y-axis voltage values are negative. The raw threshold voltage readings have been subtracted from their initial readings. The plots show the change in threshold voltage over the course of the reaction.

Last, amplification can be detected by a small change in gradient. The temporal (sigmoid) signal of the amplification reaction, shown in Figure 2.1, causes a change in gradient of the temporal signal around $t = 1500s$. When the reaction saturates, sensor drift dominates and the signal takes the shape of a decaying exponential again, vertically displaced by the value of the overall voltage increase due to amplification.

Figure 3.3 shows the spatial signal obtained (at the same timestamps as Figure 3.1) from running the existing mathematical model, initialized with $n = 100$ DNA molecules.



(a) t=5 samples (b) t=400 samples (c) t=800 samples

Figure 3.3: Spatio-temporal signal: Existing Model output

Comparing Figures 3.3 and 3.1, it is clear that none of the four observations made in experimental spatio-temporal signals are present in those from the output of the existing mathematical model. The absence of circular regions of high activity in the output signal of the existing model indicate that it is unable to accurately model the spatial variation of protons released in the reaction. It is also worth noting that the signals seen in Figure 3.3 are plots of the number of protons captured, instead of threshold voltages. As such, signal non-linearities (drift and AWGN) are not modelled, explaining the differences observed in the temporal signals of both reactions (comparing Figures 3.2 and 2.16).

The differences in model output and experimental data mean that spatio-temporal algorithms developed using model output will not be useful to real-world experiments. This motivates improvements to be made to the model, specifically:

1. *Better modelling of the spatial variation of proton release*
2. *Conversion of signal output from number of protons to change in threshold voltage*
3. *Addition of signal non-linearities (sensor drift and AWGN) to voltage output*

These changes would increase both the accuracy and usefulness of the model as a simulator for the LAMP reaction on an ISFET array. Furthermore, more realistic modelling (and parameterisation) of the physical characteristics of the TITANICKS chip would improve the flexibility of the model, allowing it to be used for future versions of the chip as well. This section covers the implementation of these improvements.

3.2 Modelling the physical characteristics of TITANICKS

The existing model simplifies the layout of the TITANICKS chip by considering it as a continuous array of ISFETs with no gaps between them. Accurately capturing the spatial distribution of protons landing on the array requires a deeper understanding and more accurate modelling of the chip.

Gaps in ISFET array

In reality, there exist a number of gaps which protons could fall into at the base of the chip. These include the gaps between the walls of the manifold (which holds the chemical solution) and the ISFET array, those between the edges of the ISFET array and the ISFETs, as well as those between individual ISFET sensors. These are shown in Figure 3.4 and the values used for each gap are shown in Table 3.1.

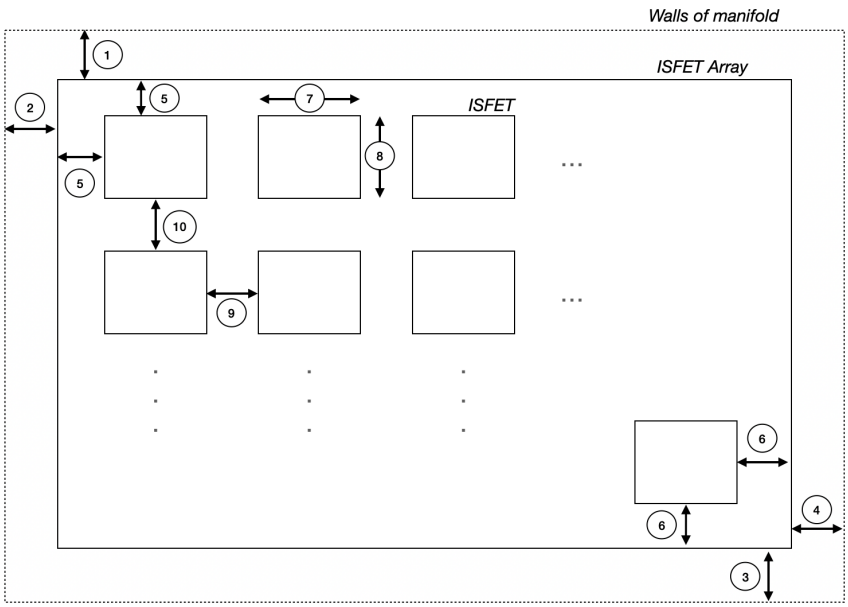


Figure 3.4: Physical layout of ISFET sensor array

Label	Physical feature	Value (μm)
1	Distance from manifold wall to ISFET Array (positive y-direction)*	2
2	Distance from manifold wall to ISFET Array (negative x-direction)*	2
3	Distance from manifold wall to ISFET Array (negative y-direction)*	2
4	Distance from manifold wall to ISFET Array (positive x-direction)*	2
5	Distance from ISFET Array edge to first ISFET	2
6	Distance from ISFET Array edge to last ISFET	2
7	ISFET width	31
8	ISFET width	37
9	Inter-ISFET separation (x direction)	2
10	Inter-ISFET separation (y direction)	2

Table 3.1: TITANICKS physical dimensions [12]. **The manifold is laser-cut and fitted on top of the ISFET array to create a reaction chamber - an imprecise process. These values were set as sensible averages on the advice of Dr Nicholas Moser, one of the designers of the chip.*

With these values, a coordinate system can be built to determine if protons fall into gaps or land on ISFETs.

Detection distance: Debye Length

To better model the spatial signal, it is also important to characterize the interaction of protons and ISFETs. Specifically, assuming the voltage output of the ISFET responds instantaneously to a change in pH, *where does a proton need to be for the ISFET to detect it?*

This can be answered by considering the interface between the ISFET and the electrolyte (chemical) solution. When the ISFET is immersed in solution, its inherent (negative) surface charge causes an interaction with positively charged Hydrogen ions in the solution [25]. These ions form a layer on the surface of the ISFET, balancing its charge, creating an *Electrical Double Layer (EDL)* [26]. The effect of a charge carrier on the potential of the ISFET surface decreases exponentially as it moves further away from the EDL. The thickness of the EDL is called the Debye Length. This is shown in Figure 3.5.

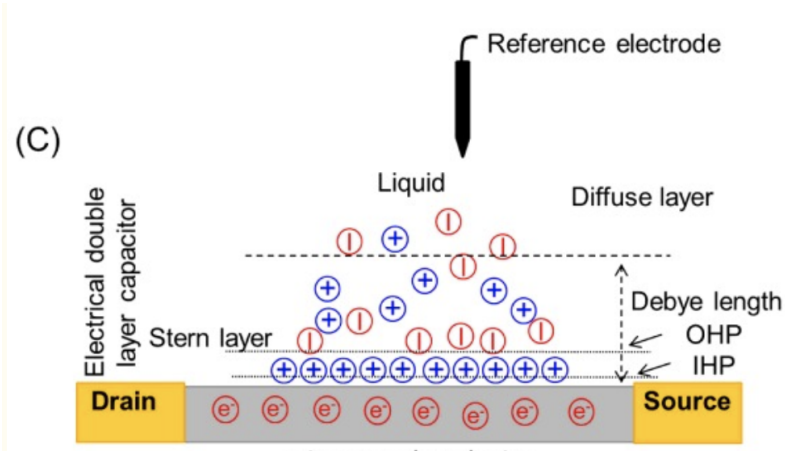


Figure 3.5: Debye Length (Detection distance) above ISFET [27]

Hence, one Debye Length is set as the threshold beyond which protons have no effect on the output voltage of an ISFET. The Debye Length depends entirely on the properties of the solution, not that of the charge carriers or ISFETs [28]. The formula for Debye Length is shown in Equation 3.1 [29] and the values used in TITANICKS are listed in Table 3.2.

$$\lambda_D = \sqrt{\frac{\epsilon_0 \epsilon_r kT}{2N_A e^2 I}} \tag{3.1}$$

Symbol	Parameter	Value
ϵ_0	Permittivity of Free Space	$8.8544 \times 10^{-12} F \cdot m^{-1}$
ϵ_r	Dielectric Constant (water at 63° C)	65
k	Boltzmann's Constant	$1.3806 \times 10^{-23} J \cdot K^{-1}$
T	Solution Temperature	338.15 K
N_A	Avogadro's Constant	$6.022 \times 10^{23} mol^{-1}$
e	Elementary Charge	$1.6022 \times 10^{-19} C$
I	Ionic Strength of Solution, $I = \sum c \cdot z^2$, where c is the molar concentration of the ion and z is its charge [30]. Computed for 1M KCl in a pH 8.75 solution.	2M

Table 3.2: Parameters used for computation of Debye Length (Equation 3.1) [31]

Using the values presented in Table 3.2, the Debye Length is computed to be **6.57 nm**. This value is validated by observing the similarity in order of magnitude ($10^{-9}m$) to other ISFET Debye Lengths presented in literature [29]. The model is therefore updated to represent ISFETs as cuboids instead of flat surfaces - this is shown in Figure 3.6.

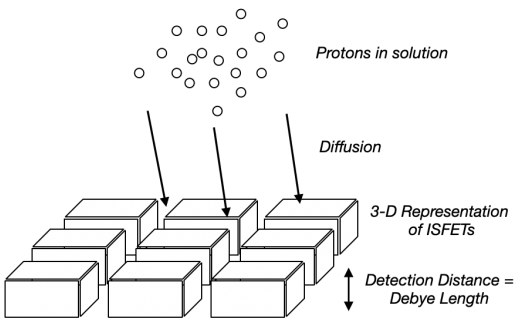


Figure 3.6: 3-dimensional representation of ISFETs

3.3 Diffusion of protons

Assumption 4 of the existing model (with reference to Section 2.8.1), that the number of protons landing on any particular ISFET can be drawn from a Normal Distribution, has key implications on the generated spatial signal and needs to re-assessed.

Diffusion is the random migration of small molecules arising from motion due to thermal energy [32]. In the context of the LAMP reaction, this thermal energy arises from the temperature of the solution (336.15 K). Protons released at the edges of clusters will therefore have a non-zero instantaneous speed and diffuse towards (or away from) the ISFET array at the bottom of the reaction chamber. The physical process of diffusion therefore undermines Assumption 4 for the following reasons:

1. Protons will require a certain amount of time from being generated at the edge of clusters to reaching the ISFET array. Drawing the number of protons per ISFET from a normal distribution assumes that this process is instantaneous, causing the spatial signal of the model to change faster than the experimental one.

2. The final position of a proton (at the time of detection by an ISFET) is likely to be dependant on its initial position - the point in 3D-space at which it was produced. Therefore, protons are not equally likely to land on any sensor.

A stochastic model can be used to simulate the diffusion of protons in the reaction chamber. Letting $[X(t), Y(t), Z(t)] \in \mathbb{R}^3$ be the position of a proton at time t and given that we know its initial position $[X(0), Y(0), Z(0)] \in \mathbb{R}^3$, we are interested in its position $[X(\Delta t), Y(\Delta t), Z(\Delta t)] \in \mathbb{R}^3$ after a time step Δt. This are computed using the Smoluchowski Equations [33]:

$$X(t + \Delta t) = X(t) + \sqrt{2D\Delta t}N_x, \tag{3.2}$$

$$Y(t + \Delta t) = Y(t) + \sqrt{2D\Delta t}N_y, \tag{3.3}$$

$$Z(t + \Delta t) = Z(t) + \sqrt{2D\Delta t}N_z, \tag{3.4}$$

where N_x, N_y, N_z are numbers drawn from a zero-mean unit-variance normal distribution and D is a *diffusion constant* which characterizes the spread of a particle, depending on the size of the particle, viscosity of the solution and the absolute temperature of the environment. Setting $Z(t+\Delta t) =$ Debye Length, we can compute the time taken for each proton (travelling in a straight line) to reach the appropriate height for detection by an ISFET. This can be computed as follows:

$$\Delta t = \left(\frac{Z(t + \Delta t) - Z(t)}{N_z}\right)^2 \cdot \frac{1}{2D} \tag{3.5}$$

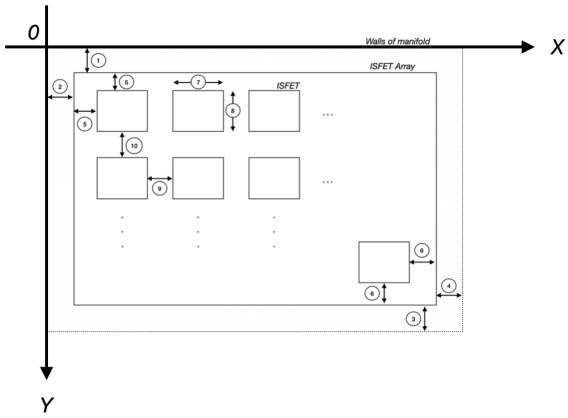
$$= \left(\frac{\text{Detection Distance} - Z(t)}{N_z}\right)^2 \cdot \frac{1}{2D} \tag{3.6}$$

Substituting Δt back into the Smoluchowski Equations yields the the x and y coordinates of the proton. A check can then be performed to see if it falls into any of the gaps shown in Figure 2.4.

3.4 Modelling proton capture

With the physical representation of the chip and mechanism for diffusion of protons established, the capture of protons can now be modelled.

The new model works by first using the physical characteristics of the chip (Table 3.1) to construct four arrays of *permissible coordinates* - two arrays each for x and y, corresponding to the starting and ending coordinates of each ISFETs. This is visualized in Figure 3.7 below.



(a) Creation of coordinate system

Array name	Array contents		
x_permissible_start	X - coordinate of **top left** corner of **1st** ISFET in *x* direction	X - coordinate of **top left** corner of **2nd** ISFET in *x* direction	...
x_permissible_end	X - coordinate of **top right** corner of **1st** ISFET in *x* direction	X - coordinate of **top right** corner of **2nd** ISFET in *x* direction	...
y_permissible_start	Y - coordinate of **top left** corner of **1st** ISFET in *y* direction	Y - coordinate of **top left** corner of **2nd** ISFET in *y* direction	...
y_permissible_end	Y - coordinate of **bottom left** corner of **1st** ISFET in *y* direction	Y - coordinate of **bottom left** corner of **2nd** ISFET in *y* direction	...

(b) Arrays for checking final proton positions

Figure 3.7: Creation of Permissible Coordinates of protons (coordinates of ISFETs)

At each time step, clusters are amplified and protons released. As before, protons falling outside the boundaries of the reaction chamber, or within other clusters, are removed. The Smoluchowski Equations are then used to model the diffusion of the protons, yielding the final x and y coordinate of each proton. These are then iteratively compared to every entry in the four arrays of permissible coordinates. The proton is deemed to have been detected by an ISFET if and only if the following three conditions are satisfied:

1. The x coordinate falls in between any two corresponding entries in the $x_allowed_start$ and $x_allowed_end$ arrays,

2. The y coordinate falls in between any two corresponding entries in the $y_allowed_start$ and $y_allowed_end$ arrays, and

3. The proton reaches the ISFET before the end of the simulation, i.e $t + \Delta t \leq t_{total}$.

A data structure is initialized to hold the number of protons detected by each ISFET at each time step of the reaction. If the above conditions are satisfied, the proton is added to a data structure at the appropriate time step.

Computational complexity of modelling diffusion

While modelling diffusion allows for more accurate tracking of the position of each individual proton, it comes at the expense of increasing the computational burden of the mathematical model.

The existing model works by simulating the release of all protons in the reaction and then, for each ISFET, drawing the number of protons detected from the Normal distribution ($\mathcal{O}(1)$) shown in Equation 2.16, an $\mathcal{O}(MN)$ operation, where M and N are the number of ISFETS in the x and y directions respectively.

Table 3.3 section compares the computational complexity of the new model and the existing one.

Step	**Complexity**
Release protons around propagation front and remove those in other clusters/outside chamber	Same as existing model
Compute Δt for all released protons	$\mathcal{O}(r)$, r = number of protons remaining
Compute final x and y positions	$\mathcal{O}(r)$
Check if each proton falls on an ISFET before the simulation ends	$\mathcal{O}(M+N)$, M,N = number of ISFETS in x and y directions respectively.
Add detected protons to the appropriate location in the data structure	$\mathcal{O}(d)$, d = number of protons detected
Total complexity at each time step	$\mathcal{O}\left(r \cdot [M+N] + d\right)$

Table 3.3: Computational complexity of new model at each time step

Since there are a total of $\frac{t_{total}}{\tau}$ time steps, the total additional complexity of the amplification reaction in the new model is:

$$\mathcal{O}\left(\frac{t_{total}}{\tau} \cdot \{r \cdot [M+N] + d\}\right) \tag{3.7}$$

The main difference in computational complexity of the models is that the runtime of the improved version depends on the number of protons released per cluster, r, because the diffusion path of each proton is tracked. The runtimes of both models are plotted in Figure 3.8 below.

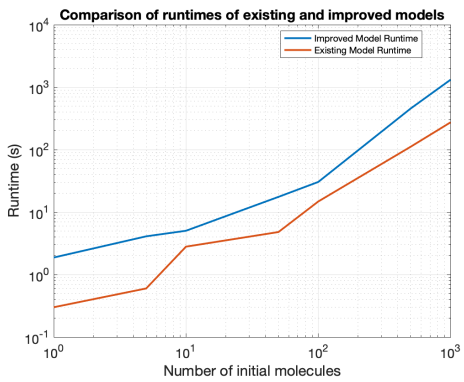


Figure 3.8: Comparison of runtime of improved and existing models

Clearly, the use of the Smoluchowski Equations and internal modelling of the chip cause a drastic slowdown in simulation. This highlights the trade-off between accurate modelling of the physical dynamics of the reaction and the practical advantages of efficient simulation. The closer to instantaneous the simulation, the better - however, for the purpose of this project, given the complexity of modelling diffusion, a simulator with a **runtime of <20 minutes** is considered acceptable. This is a heuristic given the needs of the project for fast dataset generation and resources available at hand - a personal computer with a 2.6GHz processor. Other studies utilising this model which have access to more computational power or require specific datasets may adjust this requirement as needed. From Figure 3.8, it is clear that the improved mathematical model is able to fulfill the runtime requirement of this project for $n < 1000$ initial molecules.

3.5 Assumptions of improved model

Five key assumptions needed to be made during the development of the model due to difficulty in accurately modelling a physical phenomenon, as well as to achieve faster runtime. This section seeks to justify these assumptions (primarily to do with the interaction of protons with their environment) from both perspectives.

1. It is physically accurate to compute Δt instead of assigning it a small value

As explained in Section 3.3, the diffusion of protons through the chemical solution can be modelled using the Smoluchowski Equations. However, these equations are usually applied to larger particles, such as proteins, to model their random motion due to interactions with other particles in the surroundings. Therefore, Δt is usually set to be a small value - close to 0.1 in the literature [33]. Because of the random normal variables sampled at each computation, this results in the computed path of a particle looking somewhat random, also referred to as a *Random Walk*.

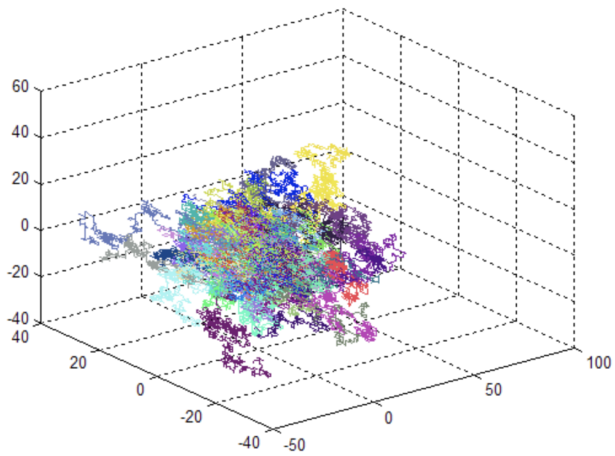


Figure 3.9: A simulated random walk. Each colored line represents the path of a particle.

In this model, Δt treated as a singular (large) time step instead of multiple small ones. Instead of modelling the movement of protons with each small time step, we compute the time taken for protons to achieve the desired displacement (diffusing downwards to the detection distance), and use that value to compute the x and y displacement. The proton is assumed to have diffused from one point in the reaction chamber to another over a certain period of time, though its path need not necessarily be a straight line - it is possible for the proton to have reached its final destination via a random walk.

1. Computational Complexity Justification

As an $\mathcal{O}(1)$ operation, this method reduces the computational load of the simulation. The alternative - setting Δt to a small value such as 0.1 and continuously updating the position of the proton until it reaches the detection distance - would be a computationally costly method because the number of such steps needed is unbounded. It would also be entirely possible, with this method, for protons to be diffusing around the middle of the chemical solution indefinitely. Since this computation would need to be performed for each proton, it is an unfeasible method for quickly modelling proton diffusion.

2. Accuracy of Modelling Justification

While the path taken by a large molecule, such as a protein, is very likely to be a random walk due to collisions with other particles (e.g other proteins, or water molecules), the subject of proton diffusion is considered highly enigmatic and difficult to characterize [34] due to the small size as well as positively charged nature of a proton. Furthermore, keeping the assumption of protons diffusing towards a specific endpoint allows for the establishment of other assumptions, explained below, which simplify the interactions between protons and other particles in the reaction chamber and consequently speed the simulation up. However, the validation of this assumption is left as a future extension of this work. Accuracy in modelling the path of proton diffusion is key to ensuring an accurate spatial signal.

2. Protons can pass through DNA clusters

Most simulations carried out with the model would be initialized with more than one initial DNA molecule. When multiple DNA molecules reach their time-to-amplification, we can expect numerous clusters to be growing simultaneously in the solution. As protons diffuse around the solution, their interactions with DNA clusters become important to characterize. Specifically, it is important to ask whether the path of a protons is affected by the presence of the DNA cluster. Figure 3.10 visualizes the two most extreme alternatives.

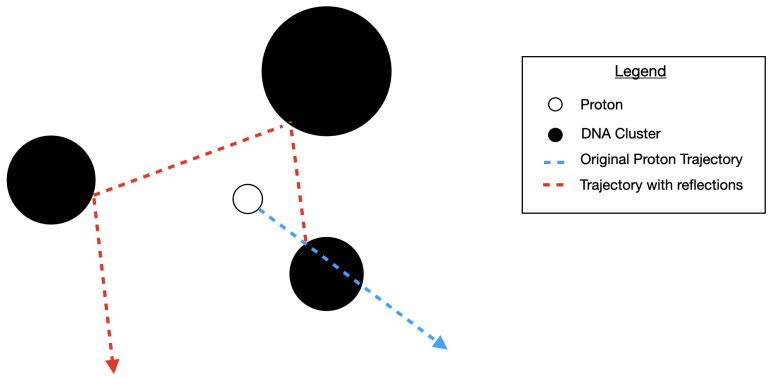


Figure 3.10: Illustration of complexity of considering proton-cluster reflections

As a simplification, it was decided that the model would be written such that the trajectory and speed of a proton would be unaffected by the presence of a cluster.

1. Computational Complexity Justification

Modelling protons as reflecting off the surface of clusters is a computationally intensive task. Using this approach, the Smoluchowski equations would first be used to compute a trajectory vector for each proton. Next, the MATLAB `solve` function would be used to check for intersections with any clusters (modelled as spheres) in the proton's path. All points of intersection with clusters would then have to be sorted by distance from the initial position of the proton to determine the first cluster it would collide with. Hence, the computation of the coordinates of collision with a single cluster is an $\mathcal{O}(c)$, operation, where c is the number of clusters in the solution.

Once the coordinates of collision have been determined, the angles of incidence and reflection can be computed by performing a dot product between the equation of the vector trajectory of the proton and the normal vector of a plane tangential to the sphere at the point of intersection. The angle of reflection can then be used to compute a new trajectory for the proton and the whole process then repeated to check for collisions with any other DNA clusters. However, even though each reflection can theoretically be computed in $\mathcal{O}(c)$ time, the total runtime of modelling proton-cluster reflections is unbounded due to the possibility of protons reflecting between DNA clusters indefinitely.

2. Accuracy of Modelling Justification

Aside from the complexity of modelling the interactions shown in Figure 3.10, evidence in literature refutes the theory that protons behave in this "tennis ball"-like manner. Proton transport is commonly described by the Grothuss Mechanism, which involves the 'jumping' of a proton across H-O bonds in water molecules [35].

Therefore, a more likely hypothesis for what happens in clusters is the reduction in the diffusivity constant (D in the Smoluchowski Equations) due to increased viscosity of the solution and the consequently lower availability of H_2O molecules. This would mean that protons continue to diffuse in a straight line, but slow down as they pass through a cluster.

Accurate characterization of this behaviour remains a future extension of this project. An experiment could be conducted in which the solution (at the beginning) contains the concentration of DNA molecules usually observed at saturation. Dropping a solution of known pH at the top of the reaction chamber and observing the change in output signal of the ISFETs could help characterize the behaviour of protons within DNA clusters. The next step would be to simulate this behaviour in an efficient manner.

3. Protons colliding with the walls of the manifold are removed

The random diffusion of protons in straight lines means that a significant portion of released protons (after subtracting those produced within other clusters) will diffuse into the walls of the acrylic manifold that forms the reaction chamber. Instead of reflecting off the walls, protons that collide with walls are removed. This is illustrated in Figure 3.11.

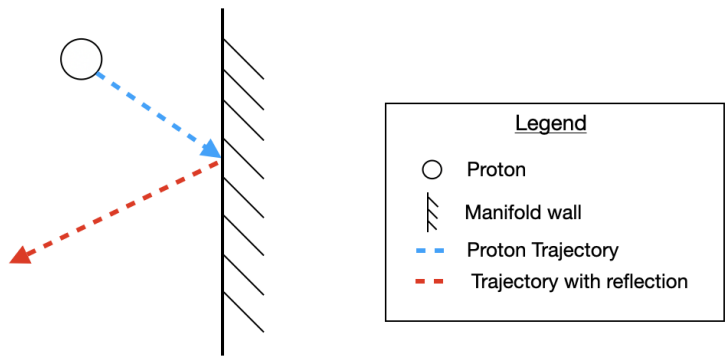


Figure 3.11: Removal of protons on collision with walls of manifold

1. Computational Complexity Justification

Modelling the reflection of a proton off the wall of the manifold is a computationally simpler task than modelling its interactions with clusters. Since the coordinates of the walls are known, the Smoluchowski Equations can be modified to work for a bounded reaction chamber [33]. Considering displacement in the x direction in the one-dimensional interval $[0, L]$,

$$X(t + \Delta t) = X(t) + \sqrt{2D\Delta t}N_x, \tag{3.8}$$

$$\text{If } X(t + \Delta t) > L, X(t + \Delta t) = 2L - X(t) - \sqrt{2D\Delta t}N_x, \tag{3.9}$$

$$\text{If } X(t + \Delta t) < 0, X(t + \Delta t) = -X(t) - \sqrt{2D\Delta t}N_x \tag{3.10}$$

Once Δt has been computed for each proton, Equations 3.9 and 3.10 can be used repeatedly until the x coordinate of the proton falls within the allowed interval. Repeating the same process with the y coordinate would yield the proton's final position. These additional steps, while $\mathcal{O}(1)$ individually, increase the complexity of the model by an amount that is difficult to bound due to the uncertainty in Δt (computed for each proton using Equation 3.5), which controls the amount of x and y diffusion. Implementing this method in the simulation resulted in an almost $3\times$ slowdown in model runtimes.

2. Accuracy of Modelling Justification

As with the previous assumption regarding proton-cluster interactions, there is little literature to support the "tennis ball"-like behaviour observed in Figure 3.11. However, even if this model of behaviour were to be accepted as possible, another underlying assumption - that the surface of the manifold wall is perfectly smooth - might not hold true.

Additionally, it is unknown at this point whether or not the acrylic surface holds a surface charge. Electrons present on the surface of the acrylic would attract protons in the electrolyte solution towards them, resulting in the formation of an Electrical Double Layer (explained in Section 3.2). It is then possible that protons coming into contact with the double layer would persist and form part of it. This phenomenon, together with the additional complexity of computing reflections, is the reason for removing protons that collide with walls.

As with proton-cluster interactions, the validation of this assumption is left as future work for this project. This assumption manifests in the reduction of signal at the edges of the array. If experimental data contradicts this, the assumption might need to be revisited. Once the nature of the interaction is validated, the next step is efficient implementation of any necessary changes.

4. Protons falling into gaps between ISFETs have no effect on array output

The next assumption made was about proton behaviour in inter-ISFEET gaps. To visualize the assumption, Figure 3.12 shows three protons diffusing into the $2\mu m$ (Table 3.1) gaps between ISFETs.

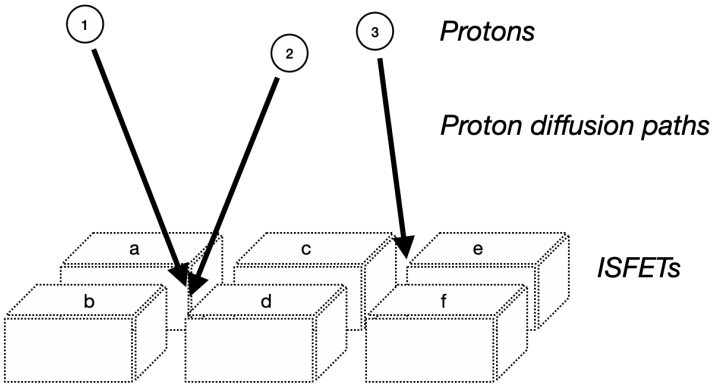


Figure 3.12: Possibilities of proton-ISFET interactions despite protons falling in inter-ISFET gaps

The figure shows Protons 1 and 2 diffusing into the junction between ISFETs a, b, c and d. Depending on their angle of entry into the junction, there is a possibility of these protons being detected by any of the eight internal-facing walls of the four ISFETS (the behind- and right-side walls of a, the behind- and left-side walls of c, the front- and left-side walls of b and the front- and right-side walls of d), due to the cuboid-shaped model of each ISFET on the array. Proton 3, falling into the gap between ISFETs c and e, can also potentially diffuse into the detection region of either ISFET depending on its angle of entry.

The assumption made during the development of this model was that none of the protons falling into such gaps would be detected by an ISFET.

1. Computational Complexity Justification

To implement the interactions between protons and the 'side walls' of ISFETS would first involve building an array of inter-ISFET gap coordinates (similar to the arrays of permissible coordinates built in Figure 3.7(b)) and checking, for all undetected protons, if any protons fell within the gaps - an additional $\mathcal{O}(M+N)$ operation for each proton, and hence an $\mathcal{O}(p \cdot (M+N))$ per time step, where p is the number of protons released. Next, a map of the ISFET walls closest to each gap would need to be created. The MATLAB `solve` function would be used to compute the intersection between the vector of the proton's trajectory and planes corresponding to all relevant walls being considered. Given that there are an equal number of gaps protons could fall into where they could potentially be detected by two or eight ISFET 'walls', the `solve` function will be used on average five times for each proton falling into a gap - making the computation of intersections another $\mathcal{O}(p)$ operation. Given the already exponential scaling of simulation runtime with the number of initial DNA molecules (and hence the number of protons), as shown in Figure 3.8, these extra $\mathcal{O}(p)$ operations would render the model impractical to use for larger values of n.

2. Accuracy of Modelling Justification

Second, assuming that protons falling within one Debye Length of ISFET 'walls' can be detected, the inter-ISFET separation of $2\mu m$ (for the TITANICKS chip) exceeds the Debye Length (6.57 nm) by three orders of magnitude. Given the size of a proton (radius $10^{-16}m$), the probability of protons being detected by the walls of ISFETs was deemed too low to warrant the extra computational complexity of modelling these interactions.

It is left as a future extension of this work to conduct experiments quantifying the relationship between inter-ISFET separation and the distribution of the obtained spatial signal. If protons falling into inter-ISFET gaps are indeed detected by one of the sensors in their immediate neighborhood, the Burst parameter β could be artificially increased to compensate for this assumption.

5. End of Reaction

Finally, the reaction is considered to have reached saturation either when:

- The total number of protons in the solution is a strictly positive constant over a certain number of time steps (controlled by an internal parameter called *tolerance*) **and** all DNA molecules have begun amplification, or

- The radius of any one DNA cluster exceeds the length of the hypotenuse of the manifold (computed as Length of Hypotenuse $= \sqrt{x^2 + y^2 + z^2}$). This represents the scenario in which one cluster consumes all the reactant in the solution, preventing any other cluster from amplifying and hence releasing protons.

3.6 Threshold voltage output

Conversion from number of detected protons to voltage change

The next step is the conversion from number of protons detected by an ISFET to the change in its threshold voltage. This is governed by Equations 2.6, shown in Section 2.2. pH was computed for each ISFET by considering the number of protons falling within one Debye Length of the sensor, as well as those already present due to the initial solution pH of 8.75.

Figure 3.13 shows the change in threshold voltage of an ISFET in the TITANICKS chip as the number of protons detected increases.

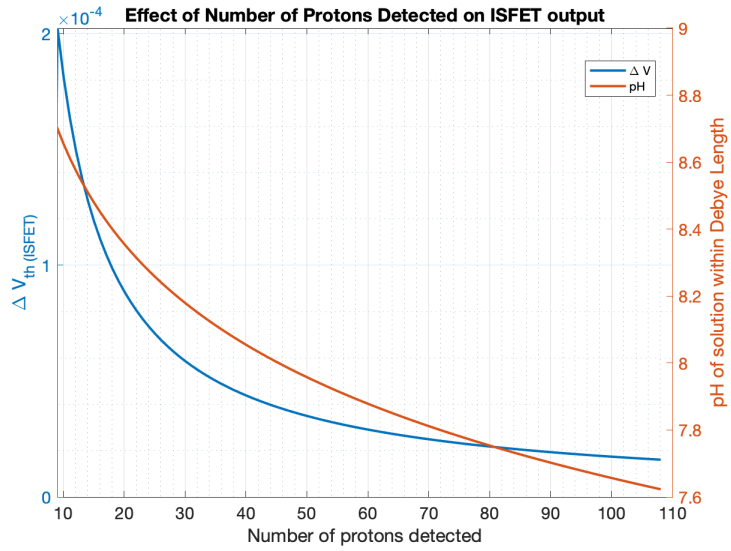


Figure 3.13: Effect of number of detected protons on change in pH and threshold voltage. 9 protons detected initially, due to solution pH.

Sensor Drift

Equation 2.7 in Section 2.2 was used to model sensor drift. The three parameters of the equation, Q', τ and β, were determined by fitting the equation to the output of each ISFET over 5 experiments using `curve_fit` from the `scipy.optimize` library. Since each ISFET displays unique drift, a range of values forming a normal distribution was obtained for each parameter. The mean and standard deviations of these are shown in Table 3.4.

Parameter	Mean	Standard Deviation
Q'	-118mV	10mV
τ	750s	50s
β	0.722	0.050

Table 3.4: Distribution of parameter values across sensors and experiments

Thus, the model contains a normal distribution for each drift parameter. At the end of the reaction, random numbers drawn from these distributions determine the drift of each sensor.

Flicker Noise

Next, in order to model Flicker Noise, it was important to determine the magnitude of AWGN to add to the threshold voltage output signals. The rolling standard deviations of individual ISFET outputs were computed across five sets of experimental data. Computing the temporal mean of these standard deviations, AWGN was found to have an average value of 0.568mV. Hence, high frequency noise was added to the output by drawing from a zero-mean unit variance normal distribution and scaling by 0.568mV.

Model output

The output of the model, both temporal and spatial, is shown in the following figurs.

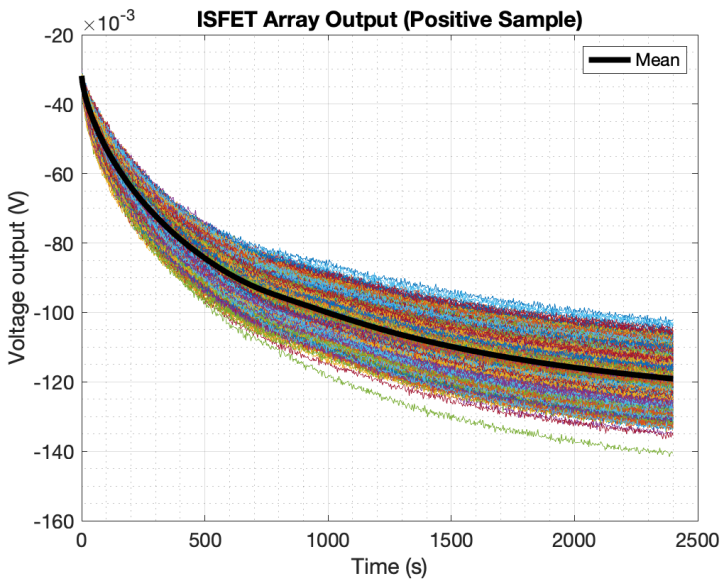
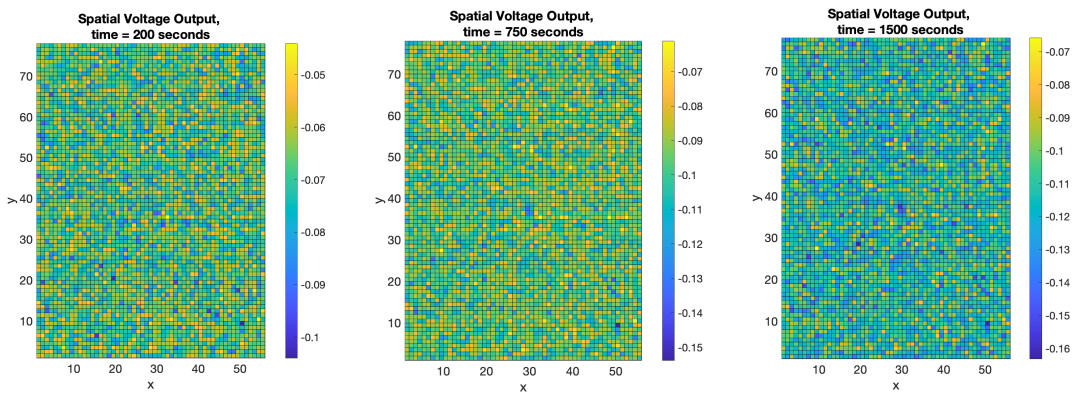


Figure 3.14: Temporal signal for positive sample from model

Comparing Figure 3.14 with 3.2, it is clear that the output from the improved model is in better agreement with experimentally generated signals. Individual ISFETs display unique sensor drift and high-frequency noise. The change in threshold voltage occupies a similar range, and amplification can be detected by a small change in the gradient of the signal around $t = 800$ seconds.

Next, comparing the spatial signals from Figure 3.15 and 3.1, a few key similarities can be observed. First, as with the experimental signals, the model output signal becomes darker over time, indicating reduced voltage outputs due to sensor drift. Pixels in Figure 3.15(b), a frame sampled during amplification, are seen to span an increased range of values compared to the pre-amplification signal, suggesting that amplification could be the cause for the increase in voltage outputs. Lastly, yellow and other brighter colored pixels are seen in distinct pockets of the signal, suggesting that the model might be better capturing the spatial distribution of protons on the ISFET array. This will be explored in later subsections.



(a) Pre-Amplification, t=200s (b) During Amplification, t=750 (c) Saturation, t=1500

Figure 3.15: Spatial voltage output of improved model at different points in time

It is also important to note that the spatial signals in Figure 3.15 are the output of single-well experiments. Double well topologies were not modelled as imprecisions in the mounting process of the manifold make it impossible to accurately predict the shape of each well.

3.7 Overview of model functionality

Figure 3.16 summarizes the user-supplied and internal parameters of the model required to form the output. Figure 3.17 is a flowchart shows a high-level overview of the steps taken to simulate the LAMP reaction in this mathematical model.

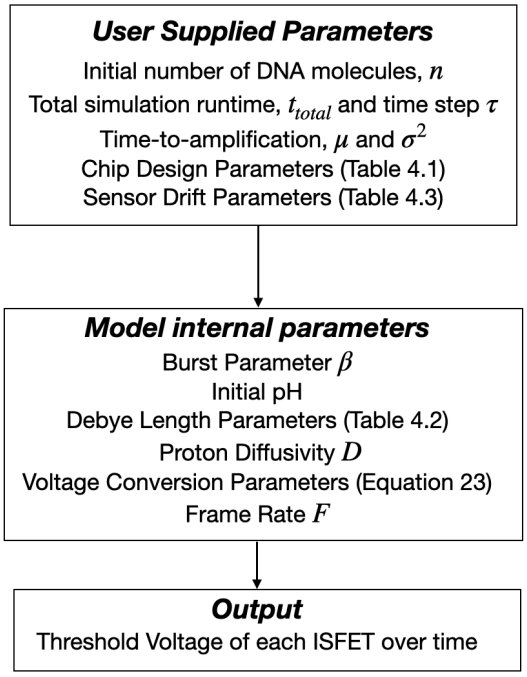


Figure 3.16: Parameters required for model

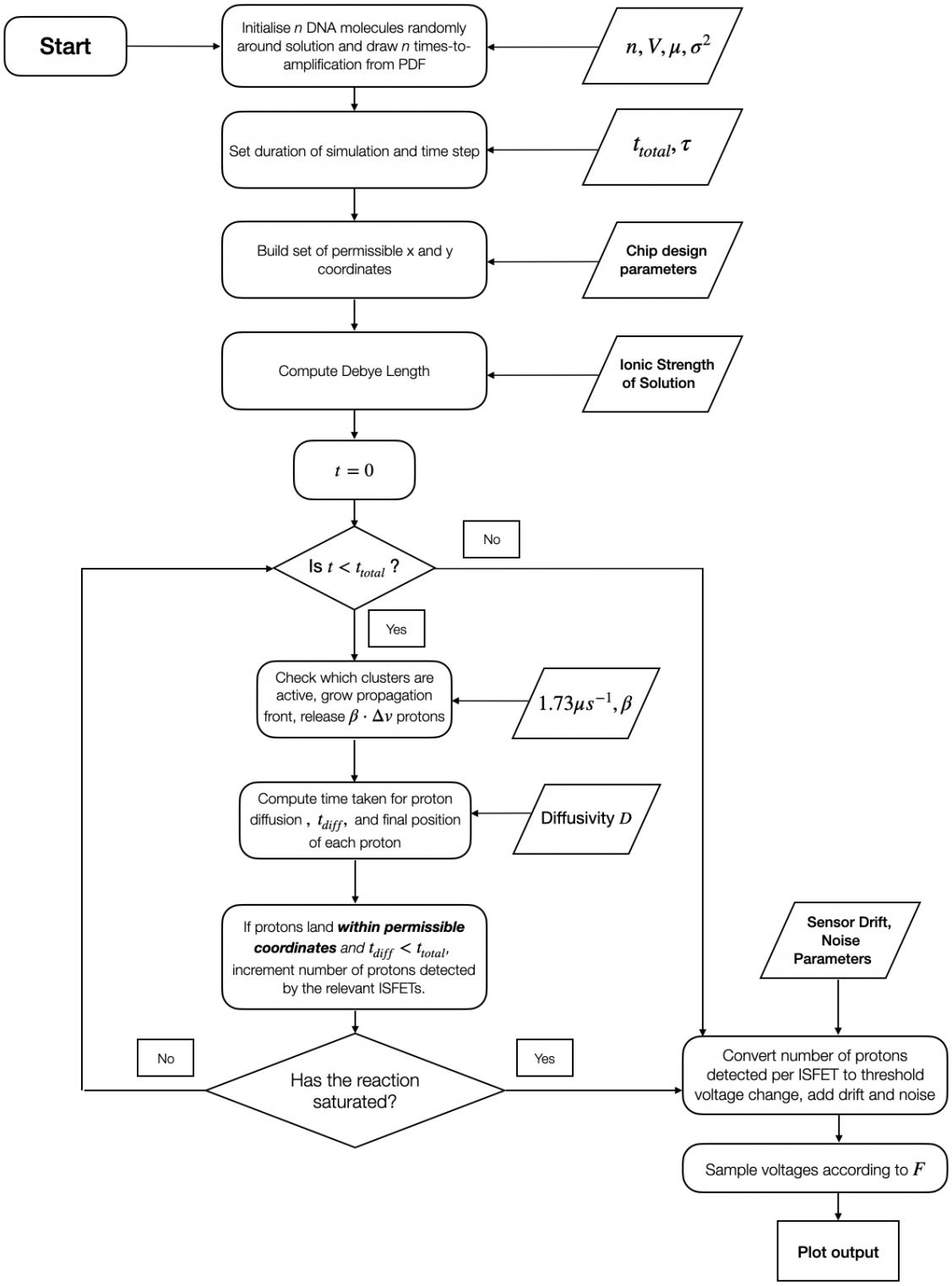


Figure 3.17: Flowchart of model

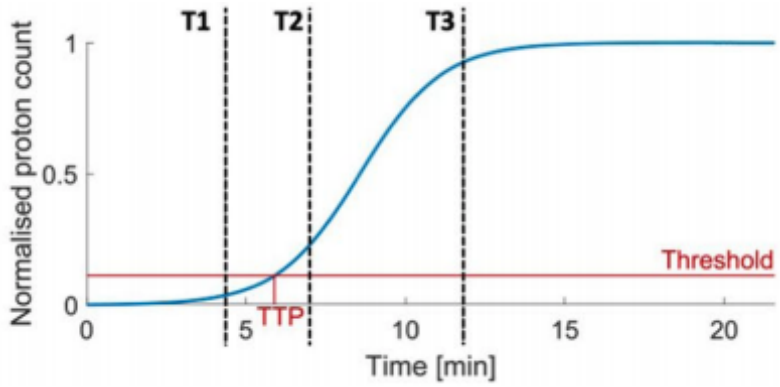
3.8 Model Validation

3.8.1 Global trends of temporal signal

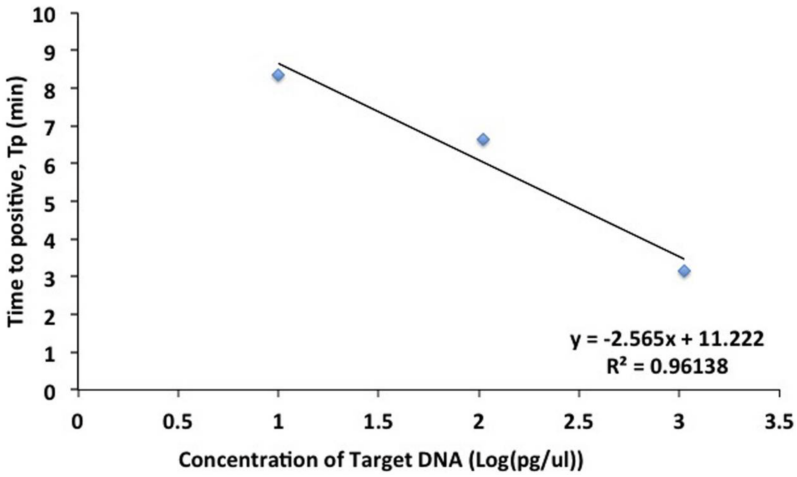
One means of validating the implementation of the model is by comparing its output to trends in the literature observed in temporal signals from LAMP reactions. It is important to note that no explicit global behaviour is encoded in the model and all model outputs are dependant on the interaction of different objects and the inherently stochastic processes described in prior sections as well as Figure 3.17.

There are two trends observed in the literature:

1. The nature of the temporal signal - the number of protons detected by the ISFET array, and hence the concentration of protons in solution, follows a sigmoidal shape as shown in Figure 3.18(a).
2. Time-To-Positive (TTP) is defined as the time taken to detect a rise in the number of protons on the array to 10% of their final value at saturation. As the number of initial copies of target DNA in the sample increases by a factor of 10, the TTP reduces linearly. This relationship is shown in the TTP vs Log (initial number of DNA molecules) plot in Figure 3.18 (b). Additionally, as the number of initial molecules increases, the standard deviation in TTP decreases.



(a) Number of protons in the solution follows a sigmoidal shape [36]

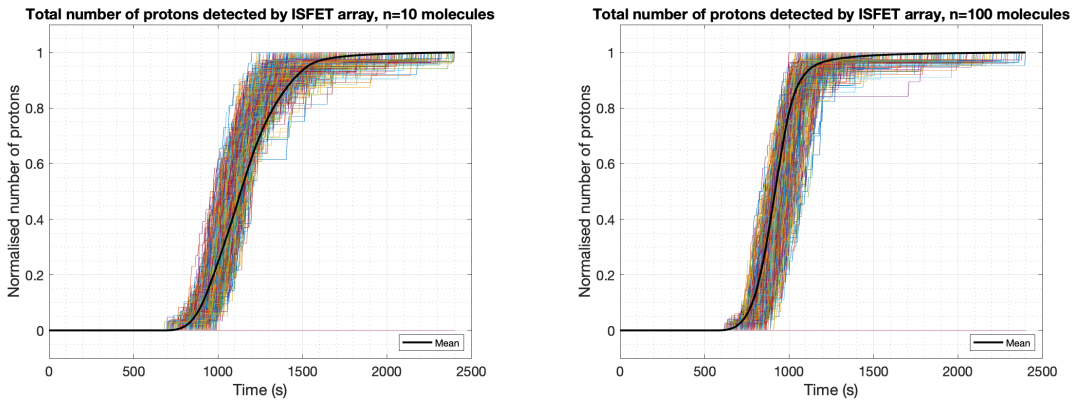


(b) TTP reduces linearly with log initial molecules [37]

Figure 3.18: Trends of temporal signal in literature

Validation of Sigmoidal shape of Temporal Signal

The model was run twice, varying the initial number of DNA molecules, n. All other parameters listed in Figure 3.16 were kept constant. Figure 3.19 shows the normalized number of protons detected by each ISFET over the course of the reaction as well as the mean across all ISFETs. Clearly, the temporal signal follows the trend described in the literature. Furthermore, it is worth noting that amplification occurs earlier in the $n = 100$ case - an observation in agreement with the second trend.



(a) Temporal signal, $n = 10$

(b) Temporal signal, $n = 100$

Figure 3.19: Observed temporal signal from Mathematical Model

Validation of TTP vs initial number of molecules

To validate the second trend, the model was run 5 times with $n = 1, 10, 25, 50, 100, 250, 500$ initial DNA molecules. The time-to-positive, defined as the time taken for the total number of protons detected by the ISFETs to be 10% of the number at saturation, was recorded for each run. Figure 3.20 shows a reducing standard deviation in TTP as n increases. This, in combination with the fact that the line of best fit shows a linear decrease in TTP, is in agreement with the literature.

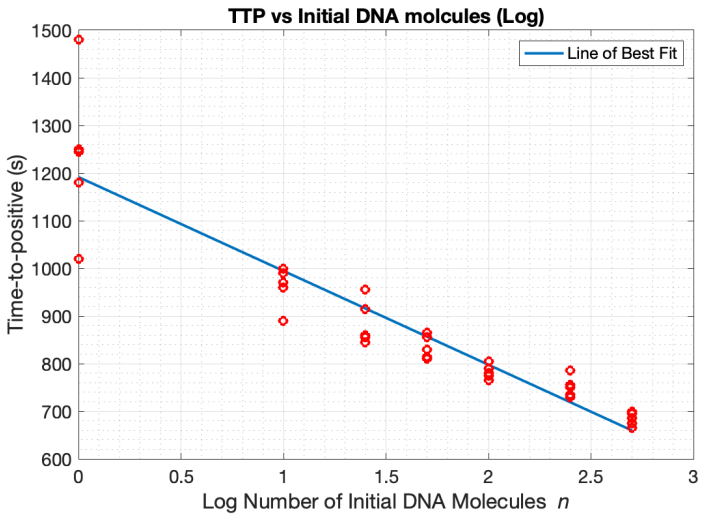


Figure 3.20: TTP reduces linearly with log-number of DNA molecules, R^2=0.952

3.8.2 Impact of Chip Modelling and Diffusion: Spatial Signal

1. Differences in obtained spatial signals from existing and improved models

To investigate the impact of improved physical modelling and diffusion on the spatial signal, both models (the existing and the improved) were run and the Burst parameter β was tuned such that they produce the temporal signal shown in Figure 3.21. The initial number of DNA molecules in solution was 50, total simulation runtime was 40 minutes and frame rate τ set to 1 second. Time-to-amplification per molecule was drawn from a normal distribution with mean $\mu = 13$ min and variance $\sigma^2 = 3$ min.

Given that the number of protons detected at saturation is 225 and using the existing definition of TTP as per a temporal algorithm, the TTP of this reaction would be roughly 750s , corresponding to the point at which 22.5 ($\frac{1}{10}$ of the number of protons at saturation) are detected. With the temporal signals tuned, the spatial signals generated by each model were investigated at $t = 624, 946$ and 1800 seconds, corresponding to three key points in the reaction - pre-TTP, during amplification and at saturation. These points are also highlighted in Figure 3.21.

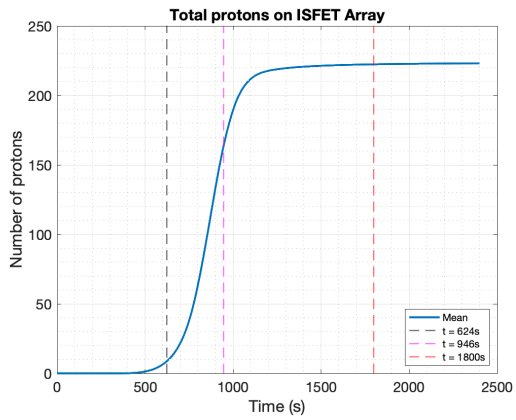
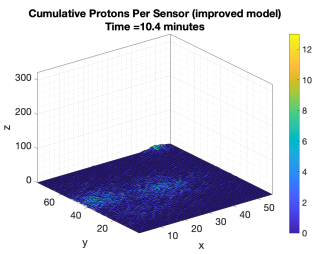
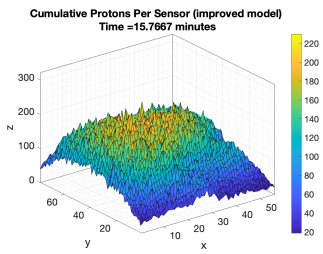


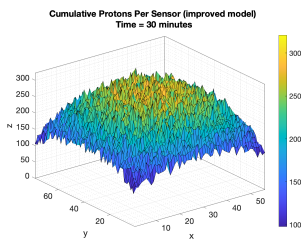
Figure 3.21: Temporal signal and key points of comparison



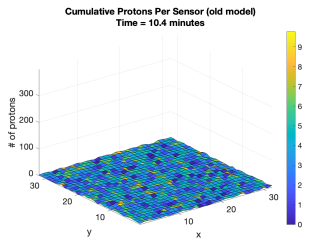
(a) Improved model, t=624s



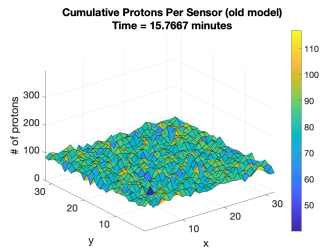
(b) Improved model, t=946s



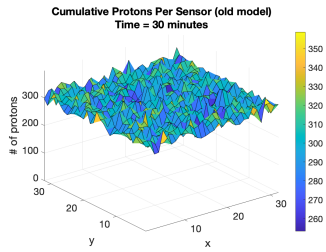
(c) Improved model, t=1800s



(d) Existing model, t=624s



(e) Existing model, t=946s



(f) Existing model, t=1800s

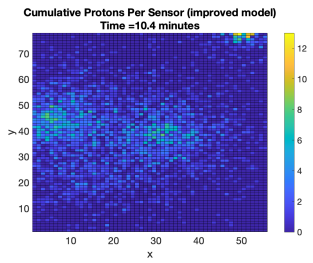
Figure 3.22: Comparison of obtained spatial signals, side view

Figure 3.22 shows a side view of the spatial signals obtained from each model at the relevant time points. From the obtained plots, a few key differences are clear to observe:

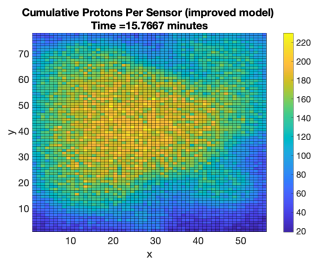
1. Signals obtained from the improved model show more spatial variation than those obtained from the existing one. This can be seen both from the shape of the signal as well as the fact that the number of protons per sensor is spread over a far wider range (as per the color bar) than in the case of the existing model.

2. The pre-TTP signal ($t = 624s$) obtained from the improved model shows evidence of clusters in certain localized areas of the solution. Detecting these 'hotspots' in the spatial signal could potentially reduce time-to-positive.

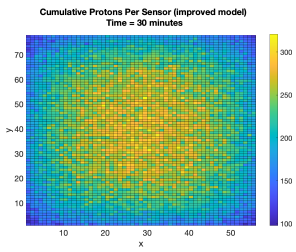
Figure 3.23 shows a top-down view of the spatial signals, providing clearer evidence of 'hotspots' from the output of the improved model. Spatio-temporal algorithms (elaborated on in later sections) will be focused on the detection of these hotspots to classify a sample as positive or negative for a pathogen.



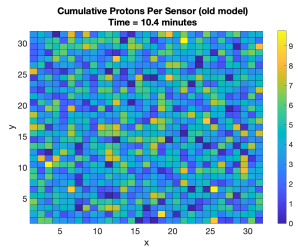
(a) Improved model, t=624s



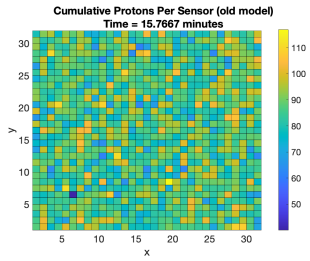
(b) Improved model, t=946s



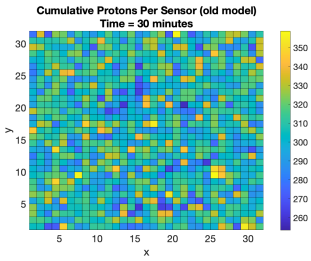
(c) Improved model, t=1800s



(d) Existing model, t=624s



(e) Existing model, t=946s



(f) Existing model, t=1800s

Figure 3.23: Comparison of obtained spatial signals, top view

3.8.3 Impact of molecule position on spatial signal

Figures 3.22(a) and 3.23(a) prompt further investigation into the diffusion of protons. Specifically, we ask the question *given that the solution was initialised with 50 DNA molecules, why are only 3 hotspots observed on the IFSET array?* To further investigate the effect of diffusion, 10 separate experiments were run with a single DNA molecule initialised in the middle of the ISFET array ($x = 28, y = 39$ at incremental heights above the ISFET array (heights were measured as fractions of solution height). This is visualised in Figure 3.24.

The time-to-amplification of all 10 DNA molecules was kept constant - 1054 seconds. The model was run and spatial signals generated. The obtained spatial signals showed the expected trends during cluster amplification and reaction saturation - a strong hotspot in the spatial signal was observed in the middle of the ISFET array. However, observing the spatial signal at $t = 1100$ seconds, shortly after time-to-amplification, revealed an interesting difference between experiments. This is visualized in Figure 3.25 for the cases in which DNA molecules were initialized at 10% and 100% of solution height respectively.

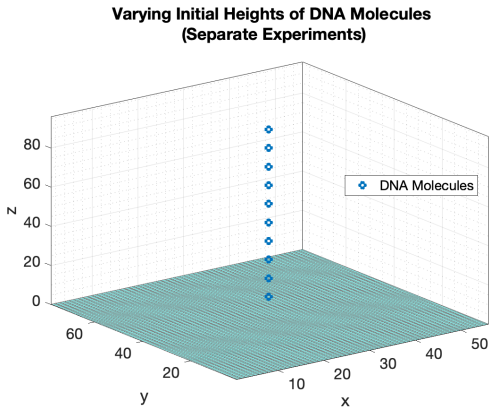
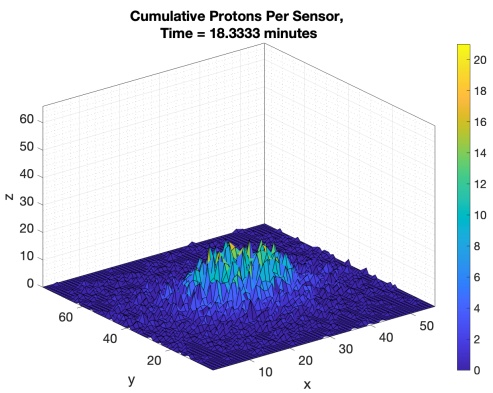
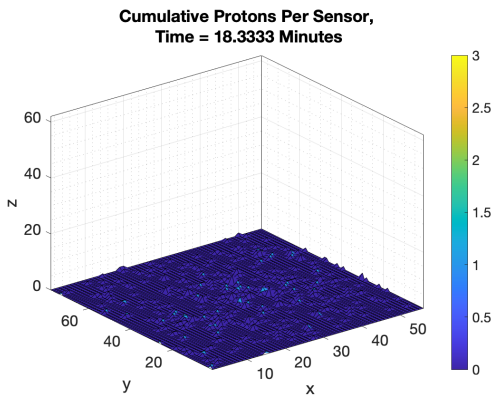


Figure 3.24: Varying initial heights of DNA molecules



(a) DNA molecule initialized at 10% of solution height



(b) DNA molecule initialized at 100% of solution height

Figure 3.25: Spatial signals at time $t = 1100s$ for molecules of different heights

Clearly, the spatial signal obtained from the DNA molecule initialized at 10% of the solution height contained a visually discernible and concentrated hotspot, whereas that of the molecule initialized at 100% of solution height did not. This can be explained using the Smoluchowski Equations (3.2 - 3.4). With increasing initial DNA molecule height, $z(t + \Delta t) - Z(t)$ increases for each proton. Thus, Δt, computed using Equation 3.5, also increases. This has two effects on the spatial signal. First, the signal will be more 'diffuse', since a larger Δt results in greater diffusion in the x and y directions. This is corroborated by the trend seen in Figure 3.25. Second, spatial hotspots from DNA molecules higher up in the solution can be expected to emerge later than those from DNA molecules lower in the solution.

This motivates exploration into a quantitative metric that describes how 'diffuse' the spatial signal is. From Figure 3.25, it is clear that as the initial height increases, the variation in number of protons detected across all ISFETs decreases. Standard deviation, commonly used to measure the spread of data from its mean, can be used in this context as a proxy for how clearly clusters appear in the spatial signal. Since only one molecule is being amplified, a larger standard deviation in the spatial signal would be attributed to a stronger 'peak' being observed. The color bars of Figure 3.25, showing the range of the number of protons detected per sensor, provides initial validation to this idea.

Plotting the standard deviation of the spatial signal at $t = 1100s$ across all initial starting heights shows the expected trend - as initial molecule height increases, the standard deviation in number of protons detected by all ISFETs decreases, due to the increased magnitude of Δt. This is shown in Figure 3.26.

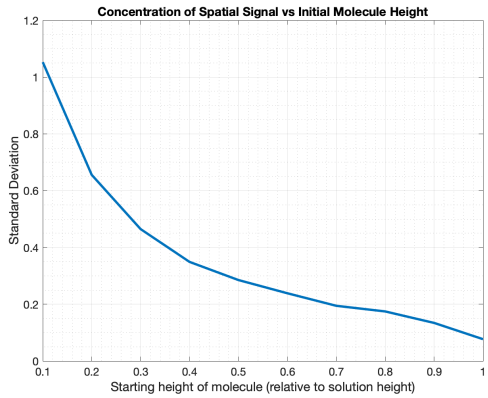


Figure 3.26: Standard deviation of spatial signal at $t = 1100s$

Figure 3.27 shows standard deviation in number of protons detected by all ISFETS across time for all 10 molecules. As expected, the standard deviations start and end at roughly the magnitude for all initial molecule heights - as the distribution of protons at the beginning and end of the simulation are identical/similar. Standard deviation is seen to increase faster as initial molecule height decreases, and peaks in standard deviation, corresponding to points beyond which the cluster grows large enough for protons to start diffusing to the edges of the array, occur later as molecule height increases.

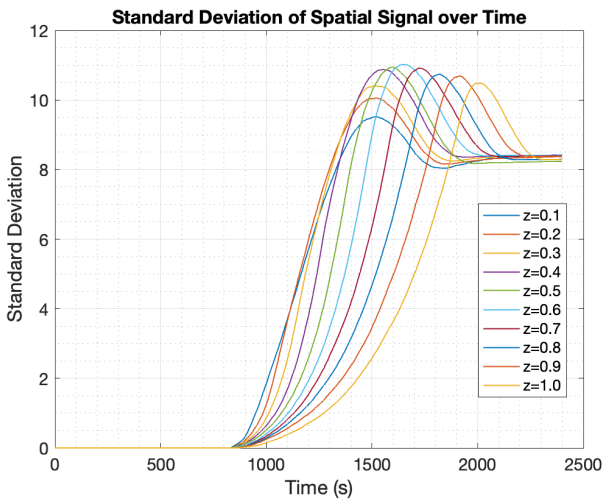


Figure 3.27: Standard deviation of spatial signal across time

The above trends suggest that molecules closer to the ISFET array surface will cause a faster spatial change in the signal. Detecting these changes (through standard deviations or other means) could potentially reduce TTP compared to a temporal algorithm.

3.8.4 Tuning of Burst Parameter for pH and threshold voltage change

The Burst Parameter, β, which controls the number of protons released around the surface of clusters, needed to be tuned for the output of the model to match the pH and threshold voltage changes observed in experimental data. Figure 3.28 shows the data observed in experiments testing for the Malaria pathogen [31].

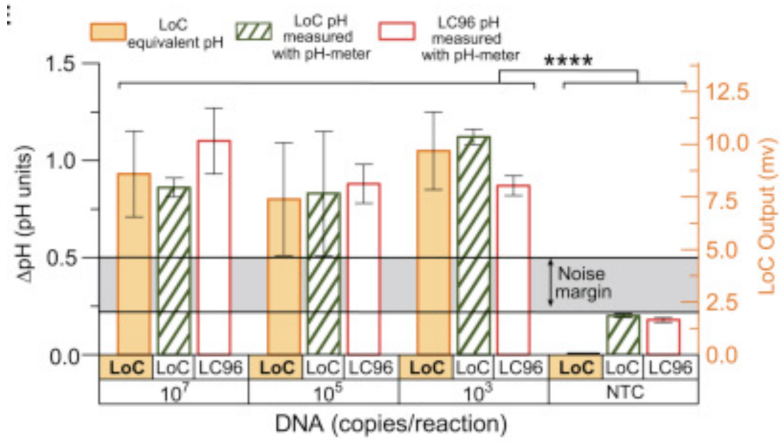


Figure 3.28: Change in pH and Voltage, Experimental Data (Malaria)

Simulations were run with $n = 1000$ initial DNA molecules, as simulations involving 10^5 or 10^7 molecules would have taken too long, as per the trend in Figure 3.8. β was varied while all other parameters kept constant. The time-to-amplification for each molecule was saved and re-used across experiments, instead of drawing from a normal distribution each time. The results are shown in Figure 3.29.

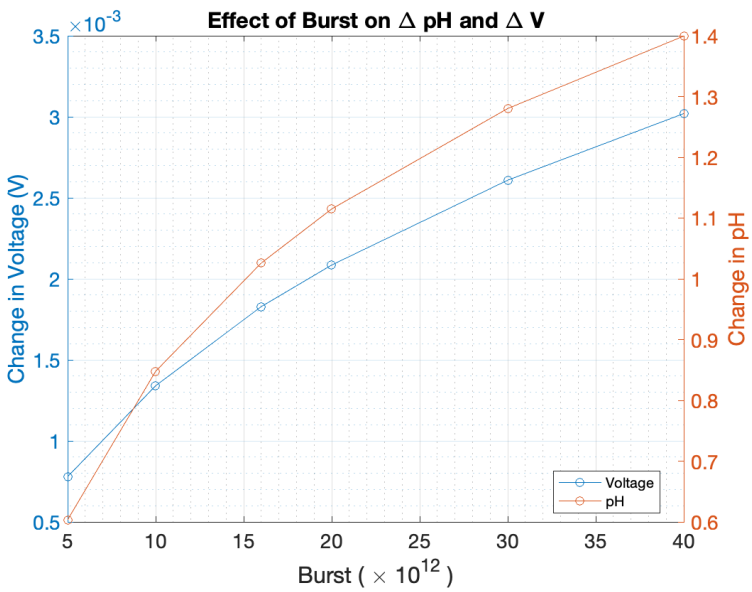


Figure 3.29: Impact of Burst parameter β on solution pH and voltage change

Clearly, for $n = 1000$ molecules, increasing β results in a larger pH, and consequently voltage, change. Comparing with the green bar in Figure 3.28, β values between 1.6×10^{13} and 3×10^{13} bring about changes in pH in line with experimental observations. However, the voltage change, though of the correct order of magnitude, is roughly one quarter of what is observed experimentally.

This can be explained by the method of calculation of voltage change. Equation 2.6 was used to compute ΔV, with ΔpH interpreted as the change in pH within one debye length of the ISFET. However, this may be an oversimplification, as prior work [25] has shown the change in threshold voltage to be based on a number of other factors.

The threshold voltage ψ_0 can be calculated using the following equation:

$$\psi_0 = \frac{2kT}{q} sinh^{-1}\left(\frac{-\sigma_0}{\sqrt{8kT\epsilon_r\epsilon_0 n^0}}\right) + \frac{\sigma_0}{C_{helm}}, \tag{3.11}$$

where k is Boltzmann's constant, T is the absolute temperature, q is the charge of an electron, ϵ_0 is the permittivity of free space, ϵ_r is the relative permittivity of the electrolyte and n^0 is the bulk concentration of ions in the electrolyte. A by-product of the electrical double layer, C_{helm} represents a capacitance formed on the surface of the ISFET

$$C_{helm} = \frac{A\epsilon_0\epsilon_r}{d_{OHP}}, \tag{3.12}$$

where A is the surface area of the ISFET and d_{OHP} is the distance of the OHP from the ISFET surface, as shown in Figure 3.31. Finally, σ_0 (in Equation 3.11) is the charge per unit area - surface charge - of the ISFET, calculated as:

$$\sigma_0 = qN_s\left(\frac{[H^+]_s^2 - K_aK_b}{K_aK_b + K_b[H^+]_s + [H^+]_s^2}\right), \tag{3.13}$$

where K_a, K_b are the acid and base dissociation constants of the electrolyte solution respectively, N_s is the number of surface sites available on the ISFET surface and $[H^+]_s^2$ is the surface concentration of Hydrogen ions. This can be computed considering the bulk concentration of Hydrogen ions, $[H^+]$:

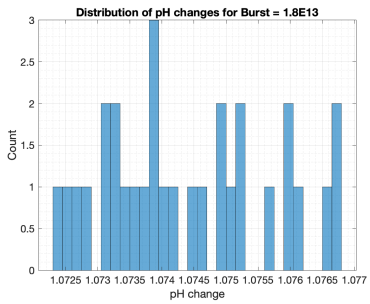
$$\left[H^+\right]_s = \left[H^+\right] e^{-q\psi_0/kT}, \tag{3.14}$$

where ψ_0 is the threshold voltage of the ISFET, as defined in Equation 3.11.

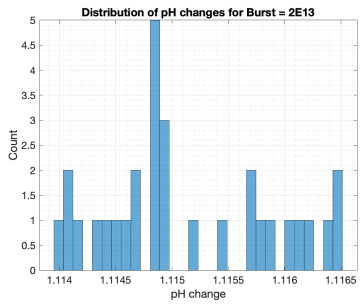
The implementation of these equations (3.11 - 3.14) for computation of threshold voltage is left as a future extension of this work. Since the threshold voltage ψ_0 is dependant on the surface charge σ_0 and σ_0 in turn depends on ψ_0 (through $[H^+]_s$, as shown in Equation 3.14), the initial value of σ_0 needs to be known in order to work around the circular definition. This value, as well as its variation with external factors (chip, environmental conditions), needs to be determined experimentally. Following this, an analysis similar to that conducted in Figure 3.29 can be performed to validate the accuracy of the values used for each parameter in the equations.

To tune for Burst, values of β at regular intervals between $16E12$ and $30E12$ were chosen and thirty simulations were run for each value, keeping all other parameters constant ($n = 1000$). This is to account for differences in initial molecule positions as well as amplification start times. Histograms were constructed showing the distribution of pH changes over the course of the 30 experiments for each value of Burst. These are shown in Figure 3.30.

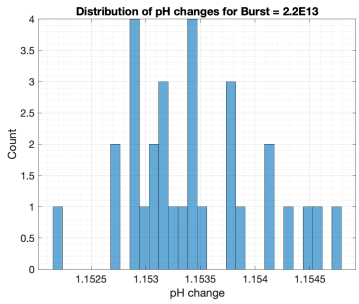
The mean pH change reflected in each histogram is taken to be an approximation of true mean of the underlying distribution of pH change against Burst. Since it is not possible to characterize the distribution exactly, this method, which relies on the Central Limit Theorem, is used as an approcimation. Comparing the obtained histograms to Figure 3.28 for both mean and standard deviation, the value of β chosen for simulations was $2.6E13$.



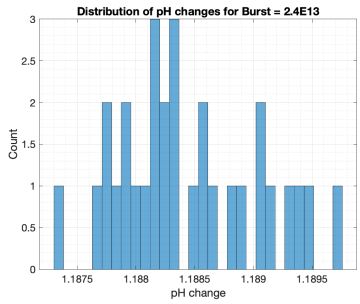
(a) $\beta = 1.8E13$



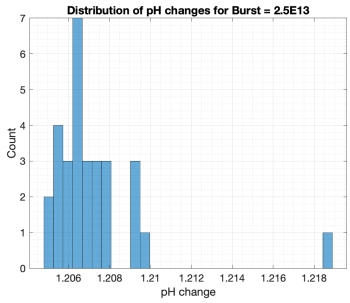
(b) $\beta = 2E13$



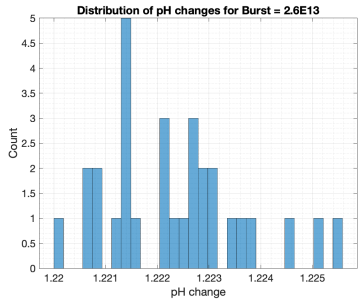
(c) $\beta = 2.2E13$



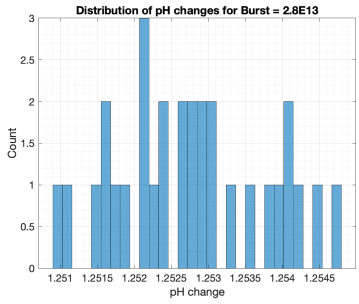
(d) $\beta = 2.4E13$



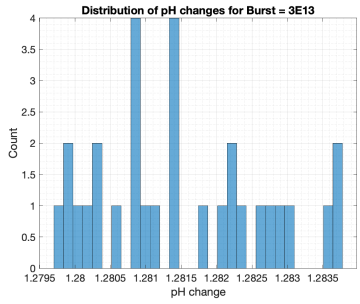
(e) $\beta = 2.5E13$



(f) $\beta = 2.6E13$



(g) $\beta = 2.8E13$



(h) $\beta = 3E13$

Figure 3.30: Histograms of pH changes over 30 experiments, varying β

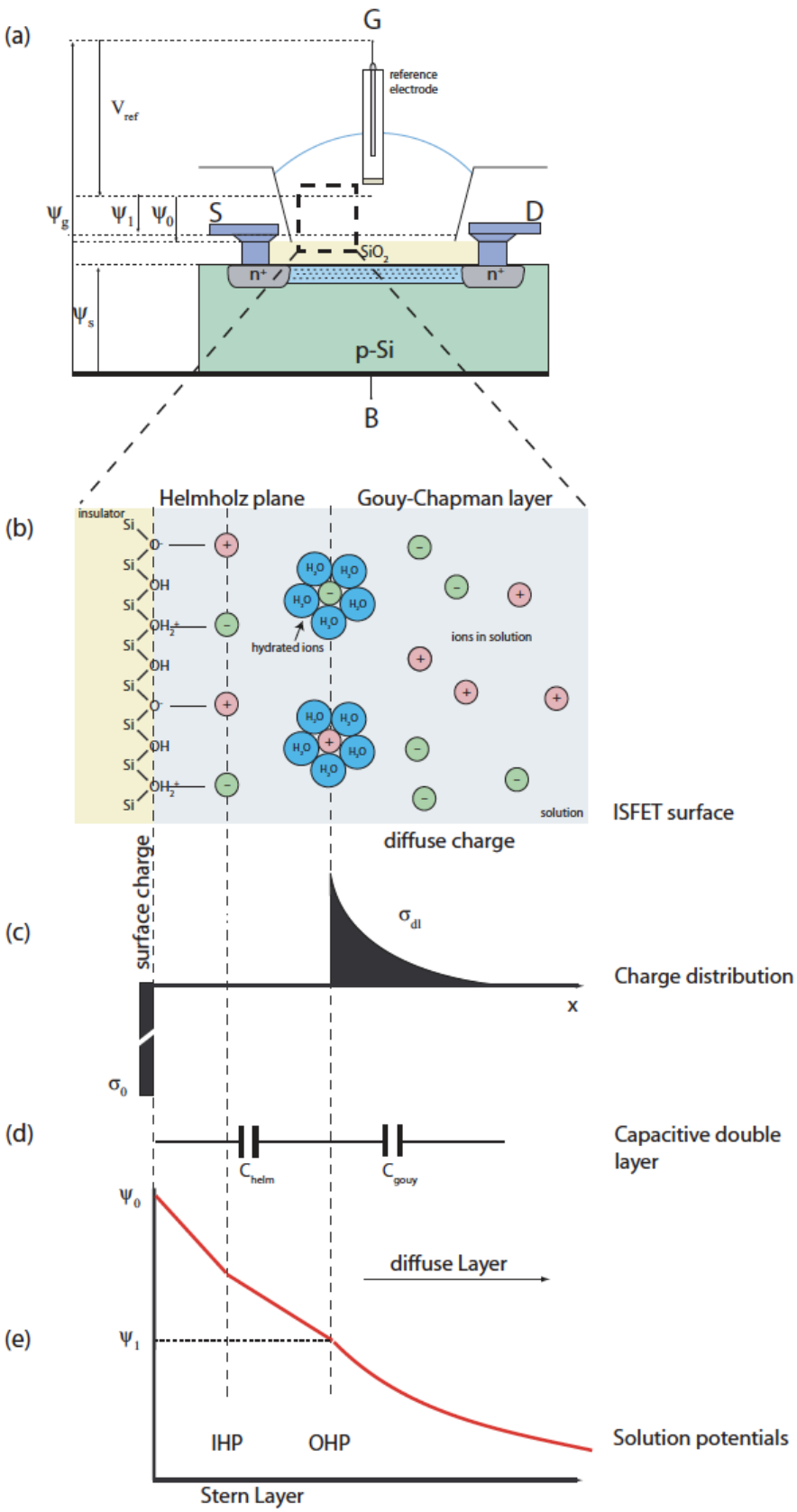


Figure 3.31: Figure showing: (a) The high-level structure of a SiO_2 ISFET, (b) the distribution of ions on the surface of the ISFET, forming a Helmholz plane and Gouy-Chapman layer, (c) the spatial distributions of surface and diffuse charge, (d) the capacitances formed and (e) resultant changes in potential [25]

3.8.5 Tuning Propagation Speed Parameter for Reaction Saturation Time

Prior work examining LAMP reactions has experimentally determined the value of propagation speed of the reaction in one dimension [23] to be $1.73 \pm 0.15\mu s^{-1}$. This argument was extended to three dimensions and forms the basis for the spherical growth of clusters (cluster radius is modelled to increase at a rate of $1.73\mu s^{-1}$).

As per the fifth assumption of the model (as in Section 3.5), the reaction is deemed to have saturated when the radius of any cluster exceeds the length of the diagonal of the reaction chamber. Therefore, the faster clusters grow, the earlier the reaction can be expected to saturate. Thus, the propagation speed in the model can be tuned to achieve a particular reaction saturation time.

Experiments, consisting of 10 simulations each, were set up in which the propagation speed of cluster fronts was varied (from 1.5 to $1.9\mu s^{-1}$, beyond the reported range in [23]) while keeping other parameters constant. Each simulation was repeated 10 times to account for changes in initial molecule position and the obtained values for reaction saturation time were averaged to produce a global estimate for saturation time. Experiments were then repeated with different values of n, μ and σ^2 (initial number of DNA molecules and mean and standard deviation of single-molecule time-to-amplification respectively) to investigate trends in the relationship between propagation speed and reaction saturation times. In total, nine such experiments were conducted.

A clear trend observed in all experiments was the stability of reaction saturation times across values of propagation speed - propagation speeds were observed to have a minimal effect on reaction saturation times particularly as the number of initial DNA molecules, n, or the mean time-to-amplification μ were increased. To investigate further, the experiments were isolated based on the parameter (n or μ) being varied. Experiments exploring the relationship between reaction saturation time and β, the Burst Parameter or σ, the standard deviation of single molecule time-to-amplification, were not carried out as initial results showed little to no relation between these.

Figure 3.32 shows the relationship between reaction saturation times and the initial number of DNA molecules, n. It is clear that as n increases, saturation time decreases and becomes more robust to changes in propagation speed. The increased concentration of DNA molecules in the solution means that clusters are more likely to expand into each others space, causing all the reactant in the solution to be consumed faster and the reaction to saturate. At lower n, saturation occurs when individual clusters grow large enough to engulf the entire solution - thus, saturation times are more dependant on propagation speed.

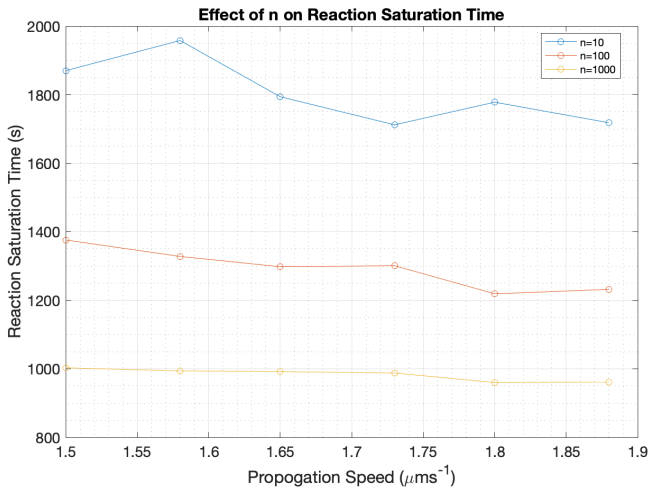


Figure 3.32: Effect of n and propagation speed on reaction saturation time. $\mu = 13$ minutes, $\sigma = 3$ minutes.

Next, Figure 3.33 shows the relationship between μ, the mean time-to-amplification for a single DNA molecule, and reaction saturation time as propagation speed increases. As before, increases in propagation speed cause slight decreases in reaction saturation time - however this can also be attributed to the fact that $n = 100$ initial DNA molecules were used. Increases in μ have a more pronounced effect on reaction saturation time. As μ increases, molecules take more

time to begin amplification (on average). If the speed of cluster growth remains the same, the reaction can be considered almost identical across values of μ, just 'shifted' backwards in time. This is confirmed by observing that, in Figure 3.33, the difference in saturation time between points at the same propagation speed is roughly equal to the difference in μ between those points. For example, comparing the first points on the red and yellow lines, the difference in reaction saturation time of roughly 400 seconds is almost equal to the difference in μ between the points - 7 minutes, or 420 seconds.

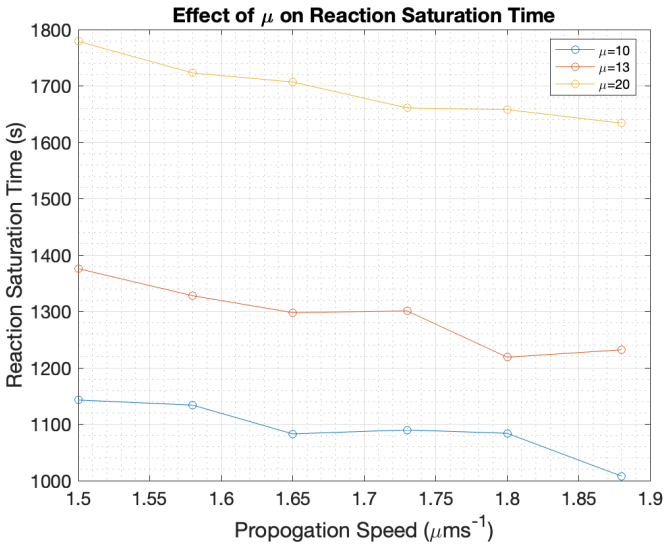


Figure 3.33: Effect of μ and propagation speed on reaction saturation time. $n = 100$ molecules, $\sigma = 3$ minutes.

The above analysis shows that while propagation speed does impact reaction saturation time, it is also important to consider other factors - namely the number of initial DNA molecules in the solution, n, as well as the mean time-to-amplification of a single DNA molecule μ. The experimental validation of these results remains a future extension of this work. Characterizing the relationship between n, μ and saturation times experimentally would allow for better tuning of propagation speed in the model, provided the assumptions about reaction saturation (Section 3.5) hold.

3.8.6 Effect of time step and chip frame rate on model output

Time Step τ

Figure 3.34 shows the temporal signal obtained from the model as time step, τ, is varied.

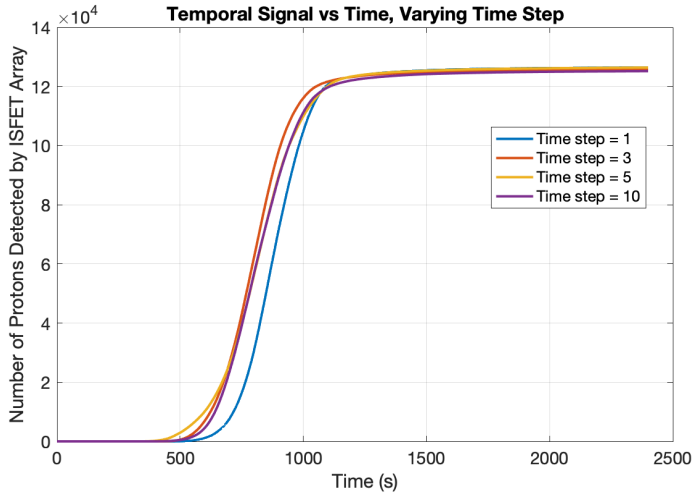


Figure 3.34: Effect of τ on temporal signal, $n = 100$ molecules

Figure 3.34 shows that the total number of protons detected by the ISFET array is roughly constant across values of τ. Differences in the amplification phase of the signals, between $t = 500$ and $t = 1000$ seconds, are due to random variation in the initial distribution of DNA molecules and their individual amplification start times.

The result can be explained by considering the effects of τ on the growth of clusters and hence its effect on the release and removal of protons.

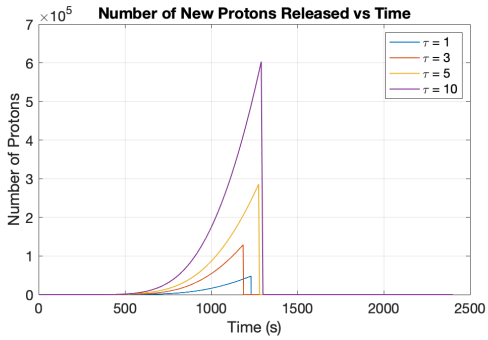
1. Release of Protons

Time (s)	Cluster radius, $\tau = 1$	Protons released, $\tau = 1$	Cluster radius, $\tau = 5$	Protons released, $\tau = 5$
1	x	$\beta \cdot \frac{4}{3}\pi\left(x^3 - 0^3\right) = \hat{\beta}x^3$		
2	$2x$	$\beta \cdot \frac{4}{3}\pi\left((2x)^3 - x^3\right) = \hat{\beta} \cdot 7x^3$		
3	$3x$	$\beta \cdot \frac{4}{3}\pi\left((3x)^3 - (2x)^3\right) = \hat{\beta} \cdot 19x^3$		
4	$4x$	$\beta \cdot \frac{4}{3}\pi\left((4x)^3 - (3x)^3\right) = \hat{\beta} \cdot 37x^3$		
5	$5x$	$\beta \cdot \frac{4}{3}\pi\left((5x)^3 - (4x)^3\right) = \hat{\beta} \cdot 61x^3$	$5x$	$\beta \cdot \frac{4}{3}\pi\left((5x)^3 - (0)^3\right) = \hat{\beta} \cdot 125x^3$
		Sum = $125\hat{\beta}x^3$		Sum = $125\hat{\beta}x^3$

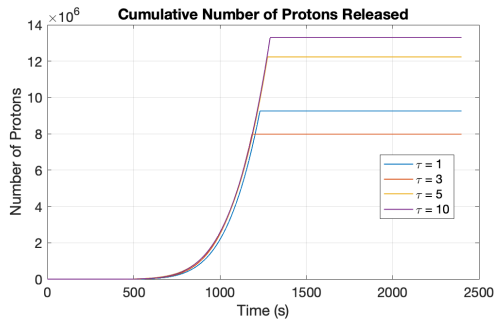
Figure 3.35: Explanation for effect of τ on proton release

Figure 3.35 shows an explanation for the effect of τ on the number of protons released around a single cluster. To simplify calculations, the radius of the cluster is assumed to start at 0 grow by x at every time step. The number of protons released per time step is computed by multiplying β by the change in volume of the cluster. From the figure, it is evident that increasing τ causes more protons to be released around clusters at *corresponding* time steps. The cluster has a larger

radius at the second time step of the $\tau = 5$ reaction (at $t = 5$ seconds) than at the second time step of the $\tau = 1$ reaction (at $t = 2$ seconds). However, the increased total number of time steps, controlled by τ, results in more cycles of amplification and proton release over the same amount of time, resulting in more total protons released. Figure 3.36 shows the effect of τ on protons released in an experiment with $n = 100$ initial molecules.



(a) Protons released at each point in time



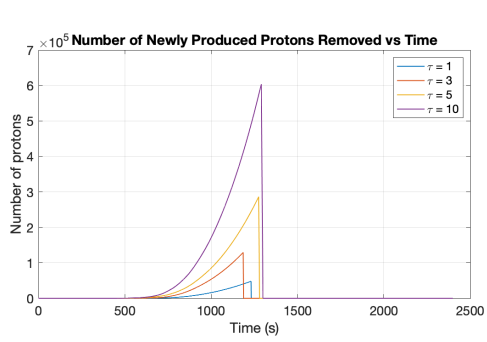
(b) Total number of protons released, integral of (a)

Figure 3.36: Effect of τ on number of protons released, $n = 100$

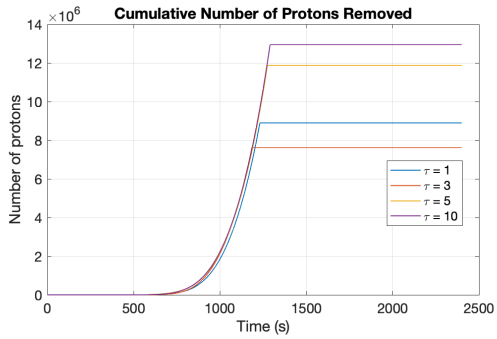
Figure 3.36 confirms the presence of the effect described in Figure 3.35. Figure 3.36(a) shows that as τ increases, the number of protons released at equivalent time points in the reaction increases (at saturation, the number of protons released drops to zero). However, due to the reduced number of times amplification takes place (not visible in Figure 3.36(a) due to interpolation performed by plotting software), **the cumulative number of protons released is roughly the same (within one order of magnitude) across values of** τ, as shown in Figure 3.36(b). The discrepancy in the final number of protons released at saturation can be accounted for by noting the difference in saturation times shown in Figure 3.36(a). Experiments with longer saturation times ended up with more protons in the solution.

2. Removal of Protons

τ has the equivalent effect on proton removal due to production within other clusters. Figure 3.37 shows the effect of τ on the number of protons removed in a reaction with $n = 100$ initial molecules.



(a) Protons removed at each point in time



(b) Total number of protons removed, integral of (a)

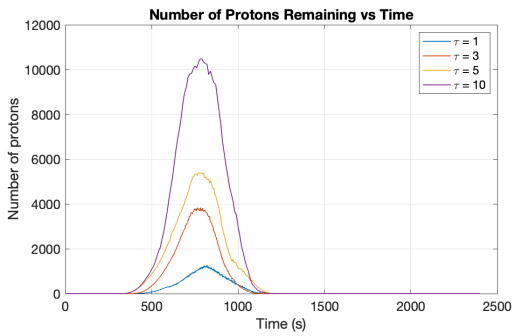
Figure 3.37: Effect of τ on number of protons removed, $n = 100$

As shown in Figure 3.35, increasing τ causes cluster radii to increase across *corresponding* time steps. For a larger τ, protons that were produced around the cluster at equivalent time steps at smaller τ would be now more likely to fall within other clusters and hence removed. This is confirmed by Figure 3.37(a), which shows that increasing τ increases the number of protons removed at equivalent points in the reaction. However, as with the case of the release of protons, lower values of τ result in more cycles of amplification and removal. While the peak number of protons removed is lower, the frequency of proton removal increases. The net effect of these opposing forces is shown in Figure 3.37(b) - **the cumulative number of protons removed is roughly constant (within one order of magnitude) across different values of** τ. As

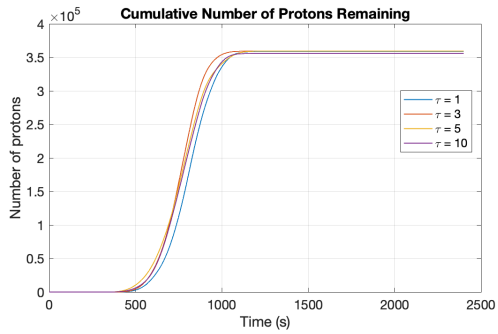
before, the discrepancy in total number of protons removed can be explained by the difference in saturation times.

3. Overall Effect

In summary, increasing τ results in a larger number of protons released, but also a larger number of protons removed, per step. To investigate the net effect of these changes, the number of protons remaining in solution (protons released minus protons removed) was tracked. Figure 3.38 shows the results.



(a) Protons remaining at each point in time



(b) Total number of protons remaining, integral of (a)

Figure 3.38: Effect of τ on number of protons remaining in solution, $n = 100$

From prior results, increasing τ results in an increased magnitude of protons released and removed per time step. This explains the trend seen in Figure 3.38 - as τ increases, the number of protons remaining in the solution at equivalent time steps increases. As before, the increased frequency of amplification and removal with lower values of τ causes the integral of the curves, shown in Figure 3.38 to take the same value at saturation. Since the total number of protons in solution at saturation is roughly the same across different values of τ, the temporal signal of the reaction (Figure 3.34) also saturates to roughly constant levels across τ.

Given that the actual reaction occurs in real time, the accuracy of physical modelling increases as τ decreases to an infinitesimally small value. However, the total number of steps also increases proportionally. Since simulation runtime scales linearly with number of steps (derived in Equation 3.7), reducing τ has the negative side-effect of causing a proportional increase in simulation runtime. To achieve a trade-off between accuracy and runtime, we refer to the criterion first mentioned in Section 3.4 - the simulation should take < 20 minutes for reactions of up to $n = 1000$ initial DNA molecules. A frame rate of $\tau = 1$, which achieves an acceptable compromise between runtime and accuracy, is therefore chosen.

Frame Rate F

As demonstrated, time step τ controls the total number of steps in the reaction and thus the number of protons produced. Since $\tau = 1$ and the reaction is simulated to run for $t_{total} = 40$ minutes, the simulation is run for a total of 2400 steps. The number of protons detected by each ISFET over each of the 2400 steps is then saved in a data structure. The Frame Rate F determines how frequently this data structure is sampled to produce the output of the model for spatial and temporal visualisation. The frame rate of the TITANICKS chip is 3 seconds.

First, the effect of frame rate on the temporal signal is investigated. Figure 3.39 shows the temporal signals obtained as F is varied.

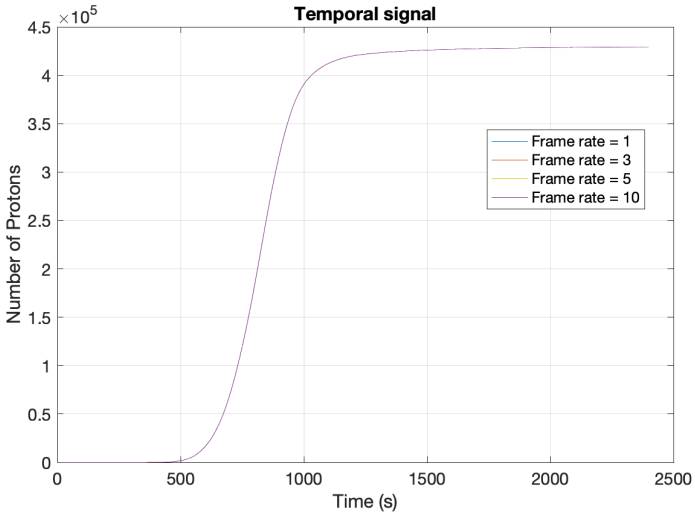


Figure 3.39: Effect of F on Temporal Signal (the 4 curves overlap each other)

Clearly, F has minimal effect on the obtained temporal signal. Since the same data structure (controlled by τ) is merely being sampled at intervals controlled by F, the temporal output of the model is identical (with reference to Figure 3.39 at common multiples of 1,3,5 and 10.

Results from prior sections have shown that clusters are visible on the chip surface before a temporal signal would have classified them as positive. This prompts investigation into the relationship between the frame rate and the obtained spatial signal. It is worth noting that as F increases, the number of frames in the output of the simulation decreases, since the reaction data structure is sampled less frequently.

The frame rate has two effects on the spatial signal. First, reducing frame rate shows more granular changes in the number of protons reaching the ISFET array. To demonstrate this more clearly, a single simulation was conducted wherin at each time step, the value of Δt for each proton that was detected by the ISFET array - the time taken for the proton to diffuse from its point of release at that time step to the detection distance above an ISFET - was saved. Figure 3.40 shows a histogram of the obtained values.

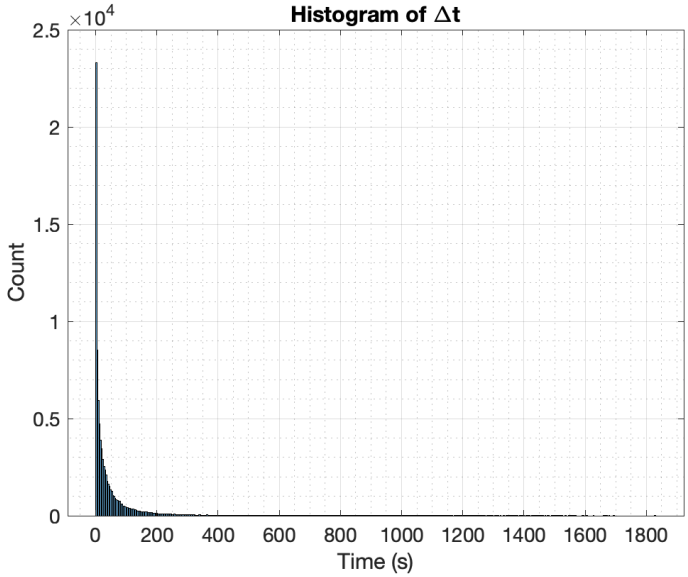


Figure 3.40: Histogram of Δt

The long-tailed nature of the histogram is indicative of the large range of times - up to 30 minutes (1800 seconds) - over which diffusion can occur. However, the vast majority of protons reach the ISFET array in a much shorter time, evidenced by the fact that the bin at $\Delta t = 0$ is the tallest. To investigate further, the histogram is converted into a Cumulative Distribution Function (CDF). Figure 3.41 is a zoomed-in view of the CDF, windowed on $0 \leq \Delta t \leq 5$.

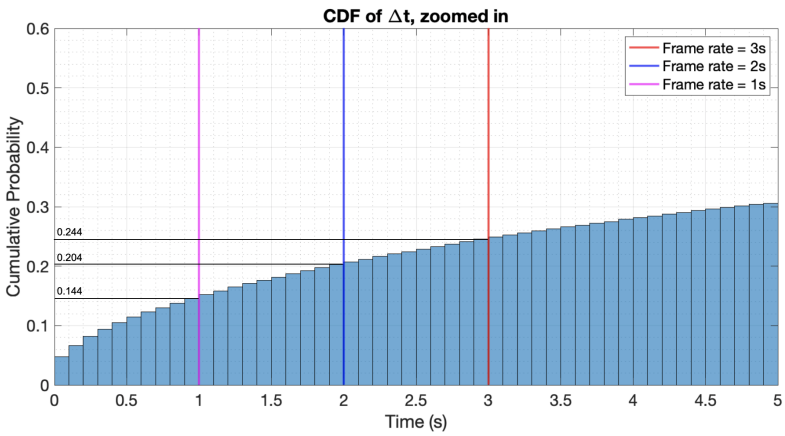


Figure 3.41: Cumulative Distribution Function (CDF) of Δt

The CDF plot shows the probability of a randomly selected proton landing on the ISFET array within a certain interval of Δt. The red line in the figure shows the current frame rate of the TITANICKS chip, 3 seconds. At this frame rate, the change in each output from the chip is a result of roughly 24% of the protons released in prior time steps. As the frame rate is reduced from 3 seconds down 1, the percentage of protons released that cause a change in spatial output reduces to 14.4%. Thus, reducing the frame rate allows for a more granular capture of the protons released at each time step. Given that the potential for spatial algorithms lies in the early detection of cluster 'hotspots' on the ISFET array, the capture of more granular changes can potentially reduce the TTP of spatial algorithms.

Second, the frame rate affects the change in the amount of noise observed between frames. Simulations were set up with no initial molecules. Varying Frame Rate F, the difference between successive frames in the spatial signal (consisting purely of sensor drift and noise) was computed and averaged. The results obtained are shown in Figure 3.42.

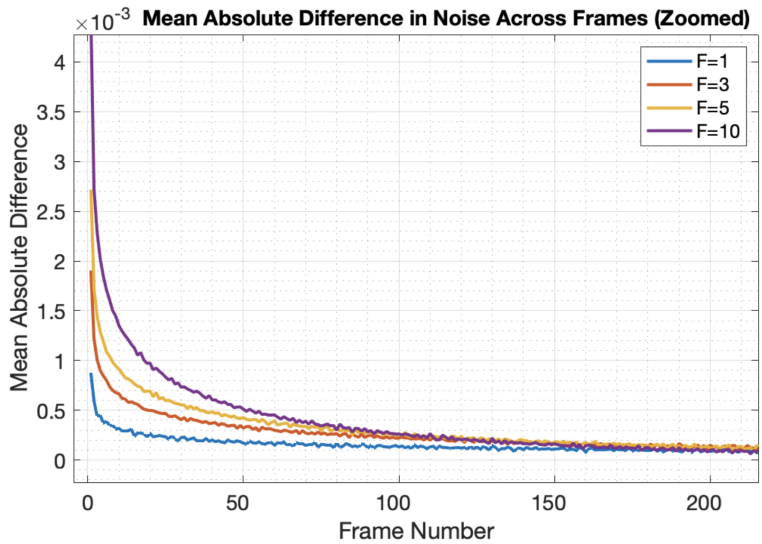


Figure 3.42: Effect of F on Difference in Noise Across Frames, Zoomed

Figure 3.42 is a zoomed-in view of the full plot, showing the first 210 frames of all simulations. It shows the magnitude of change in the mean value of noise across successive frames for different Frame Rates F. Two trends are immediately visible. First, it is clear that as the reaction proceeds, the magnitude of the difference in noise between successive frames decays exponentially. This is due to the fact that sensor drift, which dominates white noise, also follows an exponential decay. Second, as F increases, the difference in noise between successive frames increases. This is due to the fact that each frame is a sample of an increasingly later point in the reaction.

As such, the net effect of an increase in F is that sensor drift becomes harder to model, due to the simulator output consisting of fewer datapoints (less frames) as well as a greater difference between consecutive datapoints. This would be a detriment to the performance of spatial algorithms that seek to compensate for drift by modelling and subtracting it from chip output in real-time. Furthermore, lower values of F allow for the capture of more granular changes in the reaction above the ISFET array, which could potentially be of use to algorithms aimed at detecting 'hotspots' in the spatial signal. As such, lower values of F (faster frame rates) are more desirable for spatio-temporal algorithms.

3.9 Chapter Summary

This section of the report began by compating the spatial and temporal signals produced by the existing model and those observed experimentally. Key areas for improvement in model output were identified.

Next, the physical arrangement of the TITNANICKs chip was elaborated on and key variables for modelling this arrangement were isolated and assigned values. The Debye Length was introduced as the distance beyond which protons are not detected by ISFETs and proton diffusion using the Smoluchowski Equations was explained. These concepts were tied together in the subsection on proton capture, which explained the algorithmic process used to determine how many protons were detcted by each ISFET. The computational complexity of the improved and existing models were contrasted. Considering the needs of, as well as resources available to, this project, the model was deemed suitable for simulations of up to $n = 1000$ initial DNA molecules.

The chapter also detailed the five key assumptions behind the operation of the improved model, justifying each from the perspectives of computational complexity and accuracy of modelling. Future extensions for the validation of each assumption were also proposed.

The process of conversion from protons to threshold voltage outputs was then elaborated upon. Model functionality and operation were summarized in Figures 3.16 and 3.17.

The subsection on model validation began by comparing trends in the temporal output of the model to those observed in literature. The improvement in the spatial distribution of protons from the existing to the improved model was shown to be a promising development for the potential of

spatio-temporal algorithms in improving the TTP of the ISFET array. Further investigation was then conducted into the nature of the spatial signal depending on the distribution of initial DNA molecules in the solution. Finally, the internal parameters of the model - Burst β, Propagation Speed, Time Step τ and Frame Rate F - were varied (and tuned) to investigate the effect on both the spatial and temporal signals obtained from the model. The resulting analysis also shed light on noteworthy properties of both the LAMP reaction as well as the TITANICKS chip - for example, the fact that most protons reach the ISFET array surface in an infinitesimally small amount of time, or that improving the frame rate F of the chip will cause a large change in the magnitude of the temporal signal due to amplification.

In summary, this section presented the methodology behind, as well as results from, a novel mathematical model that is capable of simulating LAMP reactions on ISFET arrays to a higher degree of physical accuracy than any other simulators present in literature.

4. Algorithms

4.1 Methodology

Two spatio-temporal algorithms will be explored in this chapter, both of which follow a similar strategy. This is summarized in Figure 4.1.

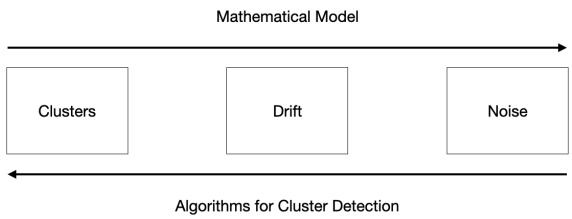


Figure 4.1: General Algorithmic Strategy for Hotspot Detection

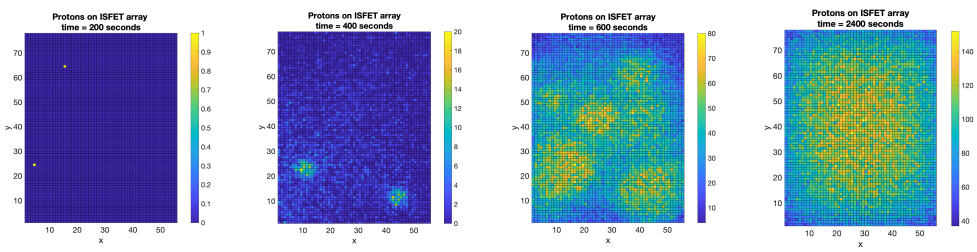
The strategy adopted is to reverse the steps taken to produce the final output of chip. This intuition behind this strategy is made clearer by observing Figures 3.2(right) and 3.14 (real experimental output as well as output from the improved mathematical model). These plots clearly show that while amplification causes a visible kink in the temporal signal, its shape is dominated by drift. It is therefore essential to algorithmically compensate for drift and improve the signal-to-noise ratio.

Therefore, both algorithms work by first performing drift compensation to boost the relative magnitude of hotspots in the spatial signal. Clusters are detected by computing the standard deviation of pixel values for each frame, with the expectation that the emergence of hotspots on the ISFET array output would cause standard deviation to rise. Experiments are assumed to be carried out in a double-well topology, allowing for a comparison between the standard deviation of the pixels from the lower well (containing the sample) and those of the upper well.

Two datasets, representing a positive sample and a control, were generated from the mathematical model for the purpose of comparing the spatio-temporal algorithms with the purely temporal one. The parameters of these datasets are shown in Table 4.1 and the spatial release of protons in Figure 4.2.

Name	n	τ (s)	μ (min)	σ^2 (min)	F (s)
A	1000	1	13	3	1
B	0	1	13	3	1

Table 4.1: Parameters of Datasets used



(a) t=200s (b) t=400s (c) t=600s (d) t=2400s

Figure 4.2: Spatial Signals from Test Dataset A

Figure 4.2 shows the development of the 'hotspots' of protons (from the positive sample) that the spatio-temporal algorithms will detect by computing standard deviations across frames.

4.2 Baseline Temporal Algorithm

Temporal Signal

To form a basis for comparison, a baseline algorithm that uses temporal methods for classification is developed,. The temporal signal of Dataset A is shown in Figure 4.3.

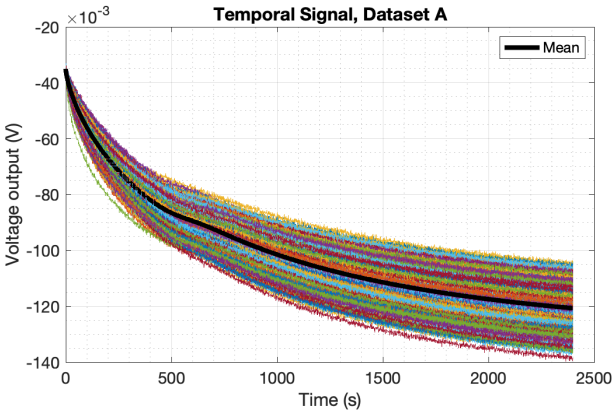


Figure 4.3: Temporal Output Signal, Dataset A

The algorithm used will check for changes in the gradient of the mean temporal signal to classify a sample as positive or negative. From Figure 4.3, the change in gradient corresponding to amplification is visually observed around $t = 500s$.

Drift curve fitting

To make the change in gradient easier to detect, drift compensation is performed by fitting a curve of the form of the Drift Equation (Equation 2.7) to the first 300 datapoints, well before amplification is seen. The number of datapoints to be used for curve fitting is a hyperparameter of this algorithm depending on single molecule time-to-amplification (μ, σ^2) and Frame Rate F.

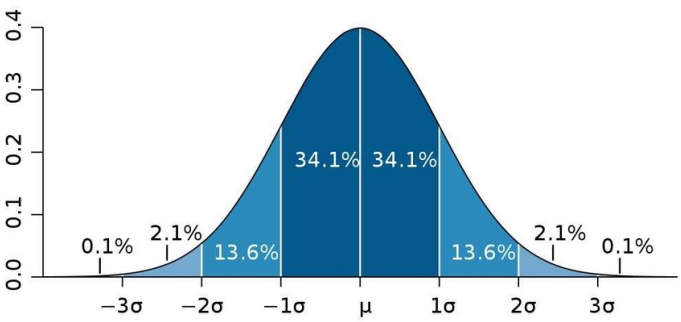


Figure 4.4: Percentage of Data Lying Within Specified Intervals of Normal Distribution [38]

A well-known result, informally called the 68-95-99.7 rule, states that 99.7% of datapoints in a normal distribution lie within 3 standard deviations (σ) of the mean (μ). This is shown in Figure 4.4.

This rule is now applied to Dataset A, in which μ and σ take values of 13 and $\sqrt{3}$ respectively for single molecule time-to-amplification. Therefore, 99.7% of the initial DNA molecules in solution will only start amplification after $13 - 3\sqrt{3} = 7.8$ minutes, or 468 seconds - matching the region of amplification seen in the temporal signal of Figure 4.3. Therefore, the output of the array up to 468 seconds is almost entirely drift. Using a more conservative approach, only the output of the first 300 seconds is considered in the fitting of data to the drift equation.

The number of data points to use for curve fitting is then equal to $300/F$, since F control the frequency of output of the chip. In this case, with $F = 1$, the first 300 data points are used. As

F increases, for the same valus of μ and σ, the number of data points used for drift curve fitting decreases, reducing the accuracy of drift modelling.

Drift-compensated temporal signal

The `curve_fit` function from the `scipy.optimize` library is used to generate an estimate for the mean drift of the array, using the first 300 data points of output. The resultant parameters of the Drift Equation, Q', β and τ, are then used to predict a value of the mean drift at each subsequent point in time. This value is deducted from the actual temporal signal, producing the drift-compensated temporal signal shown in Figure 4.5.

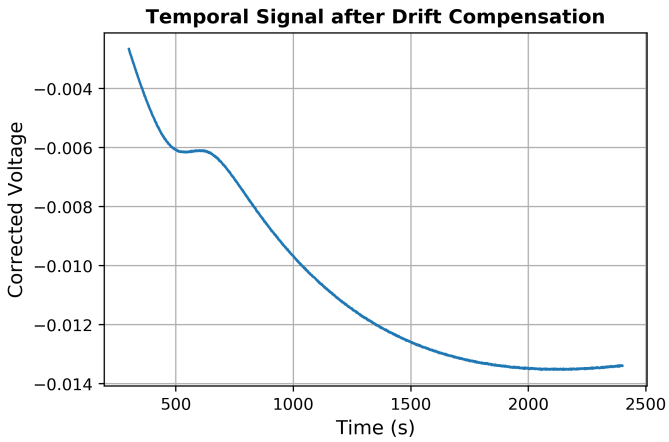


Figure 4.5: Drift-Compensated Temporal Signal

The region of amplification now more discernible, the derivative of the signal is generated by computing a rolling difference in successive output values of the temporal signal.

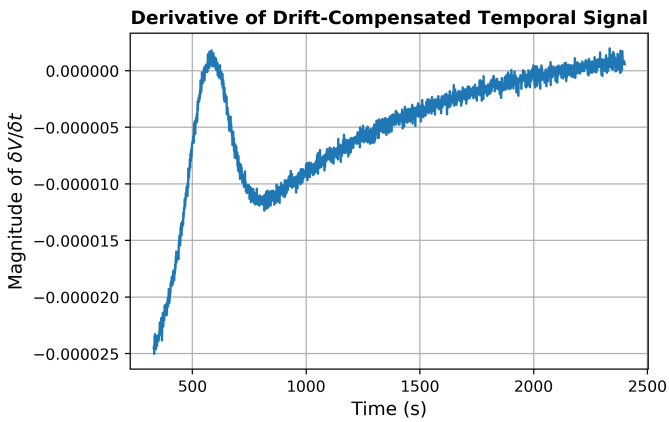


Figure 4.6: Derivative of Drift-Compensated Temporal Signal (30-sample rolling mean)

Figure 4.6 shows the change in gradient of the temporal signal over time, averaged over a 30-sample window. As expected, gradient is maximized at the region of amplification, following which it once again reduces as the decaying exponential dominates. The `find_peaks` function from the `scipy.signal` library is then used to find peaks in the gradient signal - regions of the signal that are greater than their immediate neighbours (the number of neighbors to examine being a hyperparameter to tune). The first peak is found at $t = 275$ seconds into the drift-compensated signal, corresponding to $t = 575$ seconds overall. **Hence, the TTP using the temporal algorithm is $\mathbf{t = 575}$ seconds**. This is verified by observing both the region of amplification in the temporal signal (Figure 4.3) as well as the presence of hotspots in the spatial signal at $t = 600$ seconds (Figure 4.2(c)).

4.3 Algorithm 1: Mean Drift Compensation + Spatial Standard Deviation

The first algorithm follows the same procedure as the temporal algorithm for drift compensation. The first 300 data points are used to fit the drift equation to the output of the array. The difference arises at this stage - the estimated value of drift at each time point is subtracted from the output value of each individual ISFET, instead of the mean across all ISFETS in a frame. The difference in the spatial signal after drift compensation is shown in Figure 4.7.

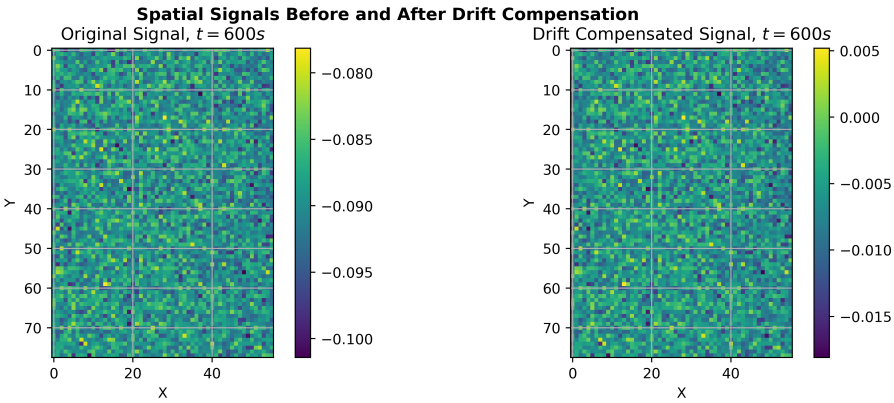


Figure 4.7: ISFET Frames at $t = 600$ seconds, before and after drift compensation

Prior sections have used standard deviation across ISFET outputs per frame as a proxy for determining the amount of spread of hotspots. Since the spatial signal post-drift compensation better resembles the spread of protons on the array, the change in standard deviation of pixel values across all remaining frames is computed and plotted, with the expectation of seeing a peak at the point of greatest rate of amplification. Figure 4.8 shows the resultant plot.

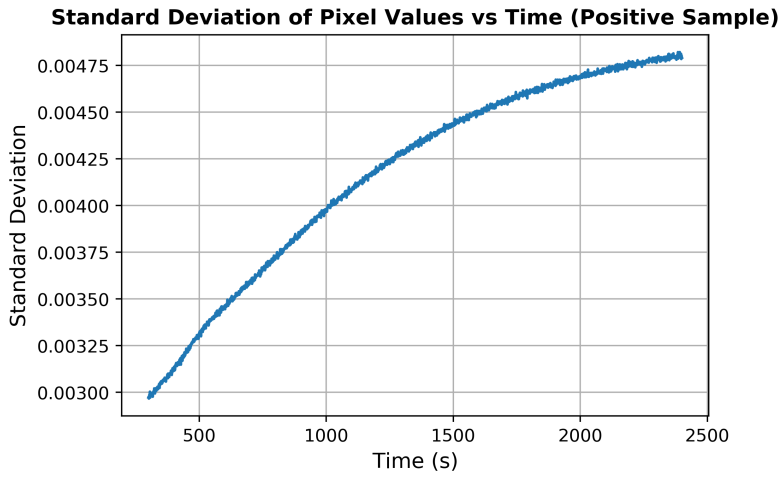


Figure 4.8: Change in ISFET Frame Standard Deviation Over Time, Dataset A

Two key characteristics of Figure 4.8 validate the use of standard deviation as a proxy for the presence of hotpots. First, the rate of increase of standard deviation reduces over time. This is due to the reaction nearing saturation and hotspots growing less quickly. Second, a kink in the signal is observed before $t = 500$ seconds, corresponding to period of time at which hotspots first develop - as per Figure 4.2(b).

However, to verify that the kink in the signal is indeed due to the emergence of hotspots and not a side-effect of drift and noise, the signal needs to be compared to that of the control experiment in the top well.

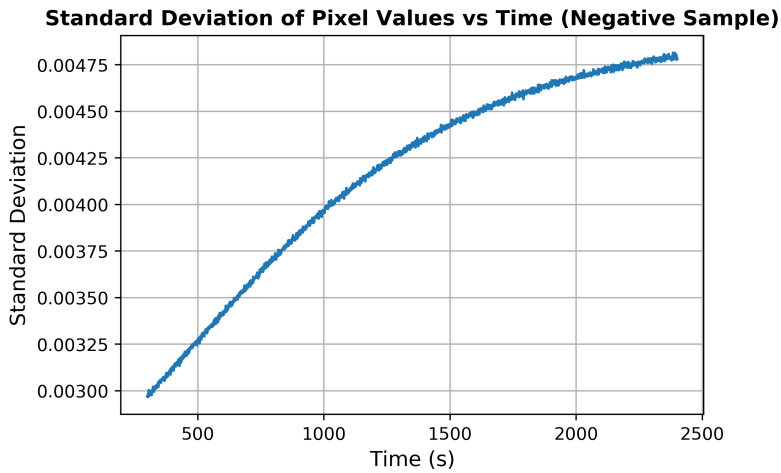
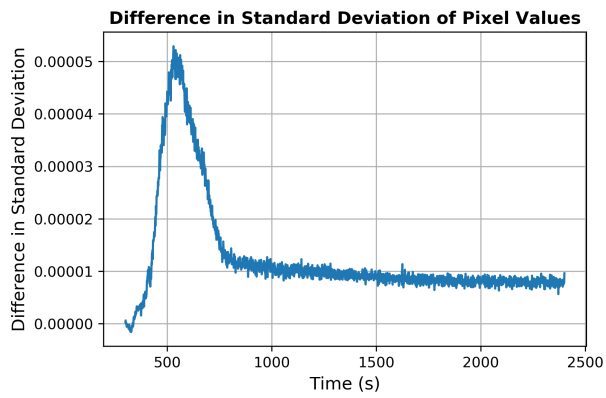
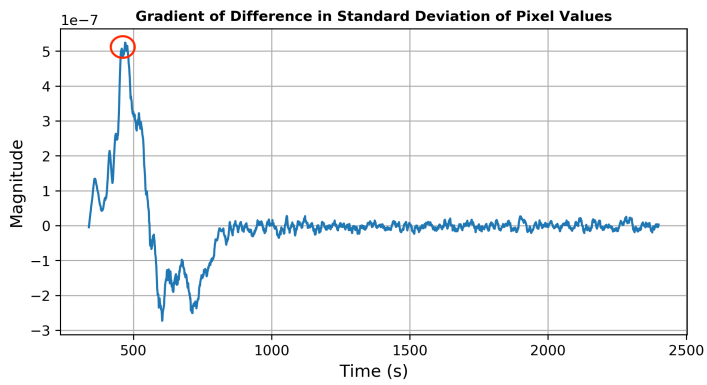


Figure 4.9: Change in ISFET Frame Standard Deviation Over Time, Dataset B

Figure 4.9 shows the change in standard deviation of the experiment in the top well (Dataset B). While it follows a similar shape and range to that of Dataset A, there is no visible kink at or around $t = 500$ seconds. This is further indication that the kink in standard deviation is caused by the emergence of hotspots on the array. To classify Dataset A as positive, the kink therefore first needs to be isolated and then detected.



(a) Difference in Standard Deviation Signals (Dataset A and B)



(b) Rate of Change of Difference in Standard Deviation

Figure 4.10: Signals for Hotspot Detection Using Standard Deviation

Figure 4.10 shows the two signals used used to classify the sample in Dataset A as positive. The top plot, Figure 4.10(a) shows the difference between the standard deviation signals of both datasets. The kink in the signal observed in Figure 4.8 is now more abundantly visible. Defining the time-to-positive as the region of fastest growth of the difference in standard deviation, the

gradient of the difference signal is computed and plotted in Figure 4.10. Using the `find_peaks` function in `scipy.optimise`, the largest peak in the derivative signal is observed at $t = 180$ seconds (circled red in the figure). Since the first 300 seconds of output were used for drift modelling, **this corresponds to a time-to-positive of t = 480 seconds**.

Figure 4.2 verifies the presence of proton hotspots on the array at $t = 400$ seconds, 80 seconds before the time-to-positive as determined by Algorithm 1. Therefore, for this particular dataset, the spatio-temporal algorithm reduces the time-to-positive from 575 seconds to 480 seconds, an improvement of 16.5%.

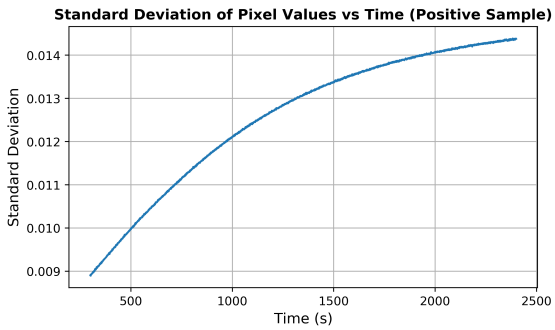
Robustness of Algorithm to Noise

This algorithm works by improving the signal-to-noise ratio of the ISFET array output by modelling sensor drift. This prompts further investigation into the impact of drift on the performance of the algorithm. Two new datasets (corresponding to a positive and a control sample), in which the standard deviation of each parameter of the Drift Equation was tripled, were generated. All other parameters were kept the same as in Datasets A and B.

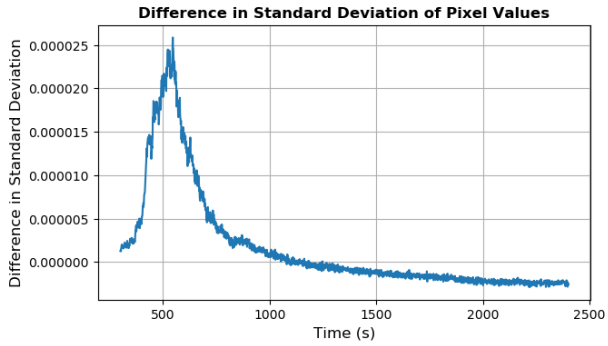
Parameter	Mean	Standard Deviation
Q'	-118mV	30mV
τ	750	150s
β	0.722	0.150

Table 4.2: Distribution of parameter values, new dataset

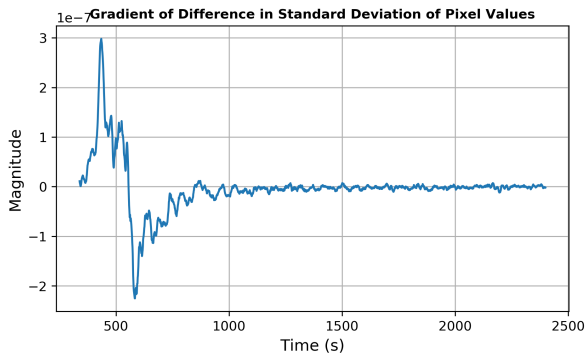
Table 4.2 shows the distribution of parameter values in the new dataset. Such values could be encountered in noisier environments, or be the result of manufacturing defects in the chip. The algorithm was run on this dataset and the obtained signals are shown in Figure 4.11.



(a) Standard Deviation of ISFET Output per Time Step



(b) Difference in Standard Deviation Signals (Noisy Dataset and Control)



(c) Rate of Change of Difference in Standard Deviation

Figure 4.11: Signals for Hotspot Detection, Noisy Dataset

Figure 4.11(a) shows the change in standard deviation across drift-corrected frames over time. Comparing this plot with that obtained from Dataset A (less noisy), the kink in standard deviations around $t = 500$ seconds is no longer visually apparent. This is because the drift of each individual ISFET now experiences greater variability, due to the increased standard deviation in parameter values. Performing curve fitting on the mean of these signals results in a poorer estimate of sensor drift, which, when used for drift compensation, worsens the signal-to-noise ratio of each frame instead of boosting it.

While the kink in standard deviation may not be visually apparent, Figure 4.11(b) - the difference in standard deviation signals between the positive sample and the control - confirms its existence. Taking the derivative of this signal results in the plot of Figure 4.11 and performing peak detection still results in a time-to-positive of under 500 seconds.

Closer inspection of Figure 4.11(b) reveals a one-order reduction in magnitude of the difference in standard deviation between the positive and control samples, compared to the less noisy case in Figure 4.10(a). While the algorithm is still shown to produce an improved TTP even in noisier conditions, these results indicate a potential susceptibility to noise. Algorithm 2 was developed to improve robustness to noise.

4.4 Algorithm 2: Individual Drift Compensation + Spatial Standard Deviation

Algorithm 1 was extended by using the first 300 data points to model the drift of each individual ISFET, instead of the mean drift across all ISFETs. The hypothesis was that doing so would allow for more accurate drift compensation per pixel, further improving the signal-to-noise ratio, as well as improving robustness to changes in the distribution of noise parameters.

A similar methodology as in Algorithm 1 (with the exception of drift modelling) was applied to datasets A and B. The results obtained are shown in Figures 4.13 - 4.16.

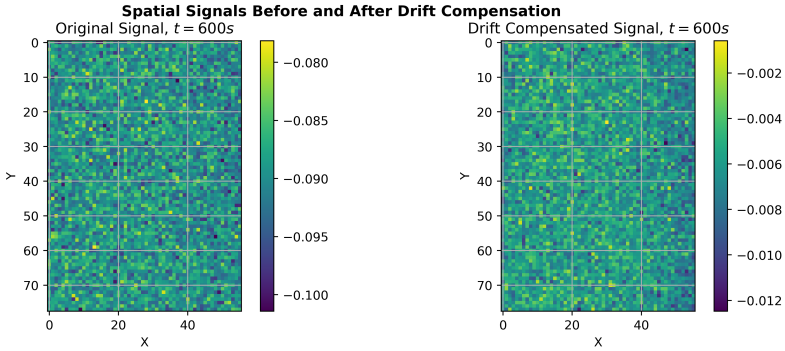


Figure 4.12: Output Frames at $t = 600$ seconds, before and after drift compensation

Comparing the drift-compensated signal on the right side of Figure 4.12 with Figure 4.2(c), similarities can be drawn in the distribution of values, particularly around the top right corner of the array. Post drift compensation, the ISFET array output better resembles the spatial distribution of protons on the array, a promising indication of individual drift modelling resulting in an improved signal-to-noise ratio compared to mean drift compensation.

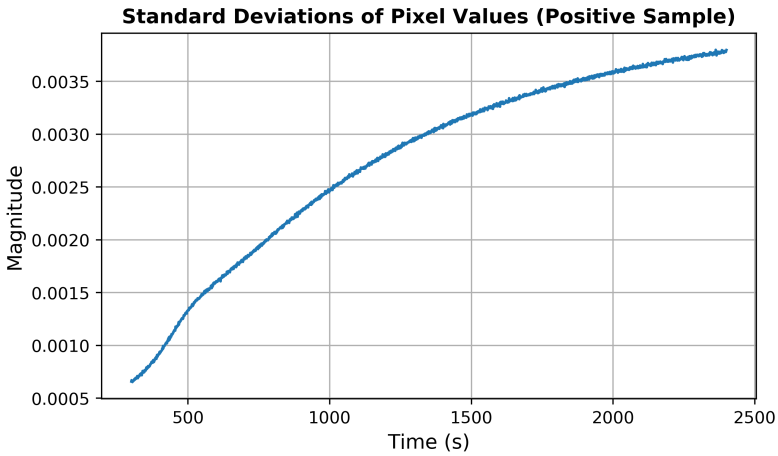


Figure 4.13: Change in ISFET Frame Standard Deviation Over Time, Dataset A

Figure 4.13 shows the per-frame standard deviation over time for Dataset A. While it shares a similar shape and trend to the corresponding signal obtained from Algorithm 1 (Figure 4.8). However, there are a few noticable differences between the two. First, the signal obtained from Algorithm 2 is less noisy than that of Algorithm 1; second, it occupies a larger range on the vertical axis; last, the kink in the signal due to amplification is more visually discernable. All these effects are a result of better drift compensation; more accurate modelling results in less errors per ISFET per frame, bringing about a cleaner signal. An improved signal-to-noise ratio means that amplification has a greater effect on the spatial distribution of voltages. Since amplification is known to appear as circular hotspots in the array, the improved suppression of voltages at regions of the array that do not detect protons results in a greater change in standard deviation over time, as well as a clearer kink in the signal corresponding to the formation of hotspots.

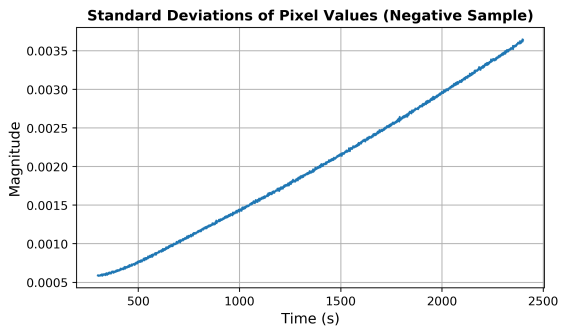


Figure 4.14: Change in ISFET Frame Standard Deviation Over Time, Dataset B

Figure 4.14 shows the per-frame standard deviation over time for Dataset B. As a consequence of more accurate drift compensation, the error between predicted and actual ISFET drift values reduces, causing ISFET output in the drift-compensated signal to be more similar. Therefore, this signal takes on a smaller magnitude at all values of t compared to the corresponding signal from Algorithm 1 (Figure 4.9). Secondly, the rate of increase of the signal is positive, while that in Figure 4.9 is negative. Given that the difference in individual ISFET outputs increses with time (verified by Figure 4.3), this signal is a more accurate represntation of the truth.

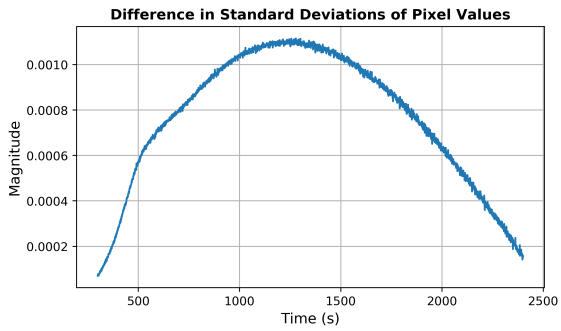


Figure 4.15: Difference in Per-Frame Standard Deviation, Algorithm 2

Figure 4.15 shows the difference between the standard deviation signals of the sample and control. The range of this signal is 0.008, compared to 0.00005 for Algorithm 1 as per Figure 4.10(a) - an improvement of 2 orders of magnitude, suggesting that modelling the drift of individual ISFEETs has a significant effect on signal-to-noise ratio. The shape of the curve also makes more intuitive sense considering the characteristics of the LAMP reaction. During amplification, the growth of clusters causes the difference in standard deviation between the sample and control wells to increase. As the reaction nears saturation, the number of protons remaining per step reduces (with reference to Figure 3.38), causing the difference in standard deviation between to saturate and eventually reduce towards 0. The maximum rate of change of difference, around $t = 500$ seconds is also seen in the signal.

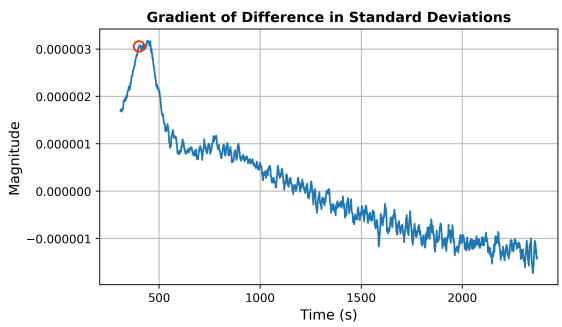


Figure 4.16: Gradient of difference in standard deviation signal

Figure 4.16 shows the gradient of the standard deviation difference signal. As before, a peak

in the gradient is seen before $t = 500$ seconds, corresponding to the point of fastest change. The `find_peaks` function is again used, from the `scipy.optimize` library, to detect the peak (circled red in the figure), with time-to-positive determined to be $t = 384$ seconds, a 20% improvement over Algorithm 1 and a 33.2% improvement over the baseline temporal algorithm.

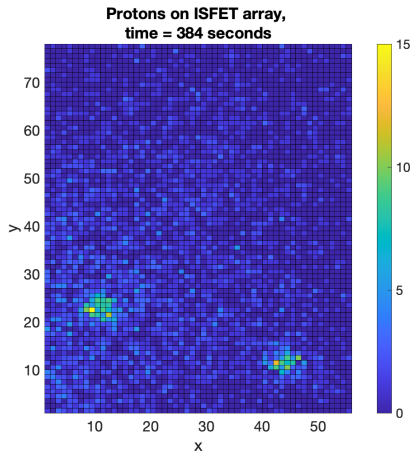
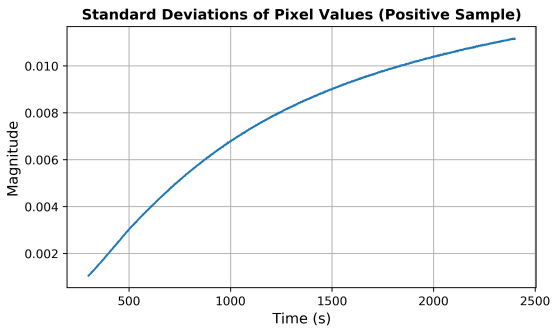


Figure 4.17: Hotspots of protons visible at $t = 384$ seconds

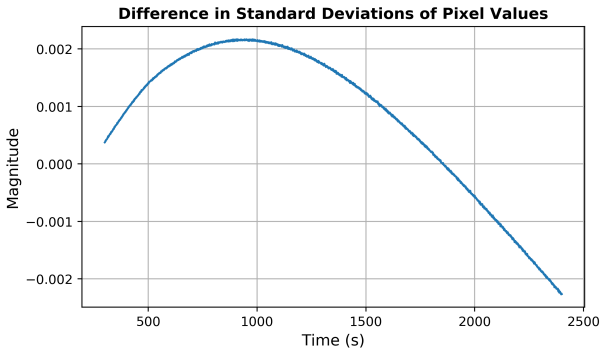
Figure 4.17 verifies the preesence of hostpots of protons on the ISFET array at $t = 384$ seconds, verifying the accuracy of the time-to-positive deteremined using Algorithm 2. The drastic improvement in TTP highlights the benefits of drift compensation on a per-pixel basis.

Robustness to Noise

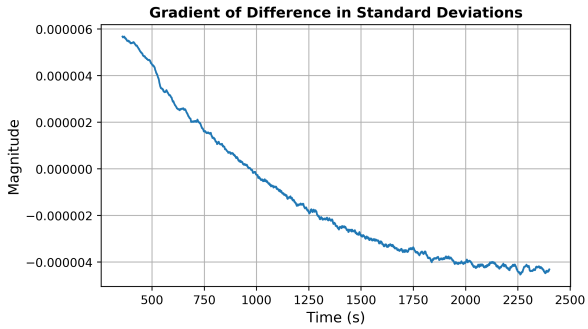
Algorithm 2 is expected to be more robust to increased variance in sensor drift, due to the fact that the drift of each ISFET is modelled and compensated for individually. This section examines the performance of Algorithm 2 on the same datasets (parametes shown in Table 4.2) used for testing Algorithm 1 for robustness to noise. The results are shown in Figure 4.18.



(a) Standard Deviation of ISFET Output per Time Step



(b) Difference in Standard Deviation Signals (Noisy Dataset and Control)



(c) Rate of Change of Difference in Standard Deviation

Figure 4.18: Algorithm 2: Signals for Hotspot Detection, Noisy Dataset

Figure 4.18(a) shows the change in per-frame standard deviation after drift compensation using Algorithm 2. The signal has a similar shape as the corresponding signal from Algorithm 1 (Figure 4.11(a)) and the expcted kink in the signal around $t = 500$ seconds is still not visually discernable. However, the signal spans a larger range, indicating the potential of Algorithm 2 for detecting subtler changes in the spatial signal. This is supported by by the next plot, Figure 4.18(b), showing a peak difference in standard deviation between the sample and control wells two orders of magnitude larger than that detected by Algorithm 1. As with Figure 4.15, the shape of the signal is consistent with the physical characteristics of the LAMP reaction, confirming the existance of hotspots on the ISFET array output of the bottom well. However, it must be noted the difference signal spans $\frac{1}{5}$ the range of the equivalent signals from datasets A and B, and that the kink in the difference between the two signals (corresponding to the first appeatance of hotspots)

is still missing. As a consequence, the clear peak observed in the signal of Figure 4.16 is missing in Figure 4.18(c). Potential solutions to this issue are discussed in Section 5 on future extensions; regardless, using the `find_peaks` function, TTP is found to be $t = 350$ seconds.

4.5 Chapter Summary

This section introduced two novel algorithms which utilize drift compensation and standard deviations to detect the presence of hotspots on the ISFET array. While standard deviation is not a statistic directly applicable for the specific detection of circular patterns (unlike the Circle Hough Transform [39], for example), it has shown promising results in this context by reducing time-to-positive by 33.2% compared to a temporal algorithm. Additionally, other factors, such as speed of computation and low memory burden, make it a favorable alternative to more sophisticated machine learning approaches. Machine learning approaches also require algorithms to be pre-trained, raising the possibility of models overfitting to the chips they were trained on. The two algorithms presented in this section operate by considering differences between the top and bottom well of the same chip, thereby increasing robustness to differences in chip manufacture and external envuronmental factors. This is supported by the fact that both algorithms were able to classify the noisier datasets as positive without any additional pre-processing or setup. However, this comes at the expense of complexity from drift compensation. Both algorithms involve computations performed over all ISFET outputs - a computationally intensive operation considering the large number (4378) of pixels involved at each frame.

Therefore, these results, while promising, must form the basis for further investigation with real experimental data. Additionally, the analysis presented in this section evaluated algorithms based on time-to-positive. In the real world, accuracy of classification is also an important factor to consider. Experiments running these algorithms on real world datasets containing differing concentrations of pathogenic DNA are essential to characterizing the reliability of spatio-temporal approaches to classification. Additionally, it is important to consider metrics other than time-to-positive. Given the improbability of creating a 100% accurate test as well as the PoC setting in which these algorithms will be deploted, it is more important for them to be *more specific than sensitive*. A false negative brings about a higher cost to society than a false positive, since an individual whose test returns a negative result (but is actually positive for the disease in question) will be allowed to interact with others, wheras an individual testing positive will be isolated, following which a gold-standard PCR test can be used to rectify the error in the PoC test.

5. Future Work

This section details possible extensions to the work carried out in this project.

5.1 Mathematical Modelling of LAMP Reaction

Validation of Model Assumptions

The developed mathematical model works based on five key assumptions, detailed in Section 3.5. These are:

1. Δt in the Smoluchowski Equations can be calculated instead of set, without compromising on physical accuracy of modelling
2. Protons can pass through DNA clusters (no reflections)
3. Protons colliding with the walls of the manifold are removed
4. Protons falling into gaps between ISFETs have no effect on array output
5. The reaction ends either when the total number of protons in solution is a strictly positive constant and all molecules have begun amplification, or when a single cluster grows larger than the entire chamber

Each of these assumptions, except the fifth, has been justified by exploring the physical accuracy as well as computational (in)efficiency of relaxing them. However, further experimental investigation (possible experiments have been proposed in the relevant subsections) into each of these assumptions is necessary for ensuring the physical accuracy of the model. Keeping in mind the trade-off between accuracy of modelling and model runtimes, it might be necessary to relax the condition of runtime being < 20 minutes in order to accommodate for phenomena that undermine the assumptions. It is also important to question whether such phenomena will be of value to spatio-temporal algorithms. Since the region of interest for these algorithms is relatively early in the reaction (around 2-3 standard deviations before the mean single molecule time-to-amplification), phenomena that affect the spatial signal at later times might not be worth the extra computational work to model.

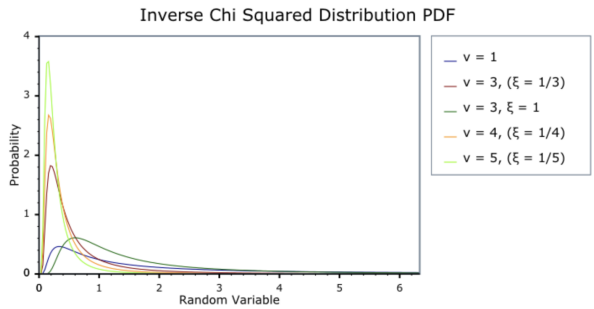
For example, if it is found that protons do indeed 'reflect' off the walls of the manifold, it is important to characterize how long (on average, or by observing the distribution of times) they would take to eventually be detected by the ISFET array. If these protons, on average, reach the surface well after the development of the initial hotspots, the effect may not be worth modelling for the purpose of spatio-temporal algorithms. This trade-off is also relevant to the validation of the first assumption regarding the path of diffusion of protons. If it is possible to characterize the deviation from a straight-line diffusion path caused by the random walk, it may be found that protons whose endpoint is in the gap between ISFETs have a certain probability of being detected by one of the sensors in the neighborhood. However, this phenomenon might not be worth modelling given the evidence presented in the histogram in Figure 3.40 - if most protons reach the array within an infinitesimally small amount of time, they are unlikely to deviate very far from a straight line path.

Conversion From Protons Detected to Threshold Voltage Change

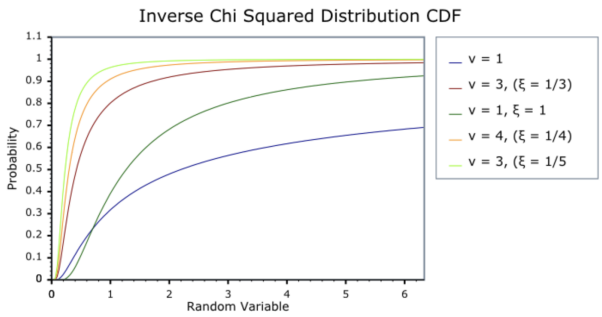
Next, as explored in Figures 3.28 and 3.29, there exists a discrepancy between the voltage change due to amplification between real and experimental data. Fixing this discrepancy could mean a better signal-to-noise ratio in model output, which would be extremely beneficial in testing algorithms for robustness to noise, similar to the analysis carried out in Sections 5.3 and 5.4. The discrepancy could be fixed by changing the equations used to determine voltage output due to amplification. Equations 3.11 - 3.14 can be used in place of the existing ones; however, experiments investigating the initial value of σ_0 need to first be conducted.

Distribution of Δt

Next, further work can be done investigating the distribution of Δt. Since Δt is computed by division of a squared random normal variable, its distribution is expected to be inverse chi-squared. The PDF od inverse chi-squared random variables is controlled by two parameters - v, the number of degrees of freedom and ϵ, a scale parameter. The histogram and CDF of Δt (visible in Figures 3.40 and 3.41 respectively) closely resemble those of an inverse chi-squared distribution, shown in Figure 5.1.



(a) PDF



(b) CDF

Figure 5.1: Inverse chi-squared distribution [40]

A key difference between the distributions of Δt and those of the inverse chi-squared random variable lie in the behaviour close to 0. The inverse chi-squared random variable has a PDF and CDF of 0 as its value approaches 0. The histogram of Δt does not display such behaviour. Even as bin size is reduced arbitrarily, the infinitesimally small values of Δt mean that an empty bin at $\Delta t = 0$ is never observed. While this can be explained by the fact that Δt is assumed to be a continuous random variable, which by definition takes an infinite number of values and hence has zero probability of taking any specific value, it also requires further investigation as it is not physically accurate for Δt to ever equal 0. A possible solution is to assume Δt takes an inverse chi-squared distribution and tweak the parameters v and ϵ to fit amplification start times observed in real experimental data.

Initial Position of Protons

Last, it is also important to investigate the effect of the initial position of protons on the spatial signal. The model currently works by releasing $\beta \cdot \Delta v$ protons around the surface of the cluster, where Δv is the difference in cluster volume between the current and previous time step. However, this means that even protons released between the two time steps (somewhere along the radius of the cluster) would be initialized on the surface of the cluster. This error in the initial position of protons, which worsens with increasing values of τ, propagates to errors in their final positions, through the Smoluchowski Equations (Equations 3.2 - 3.4). The result is a less physically accurate spatial distribution of protons on the ISFET array.

There are two initial ideas to explore as a remedy to this problem. The first is reducing the

value of τ to make the simulation run as close to real-time as possible. However, as determined in Section 3.4, simulation runtimes scale linearly with the number of time steps, one again highlighting the need to achieve a trade-off between model efficiency and accuracy. The second potential solution is to assign each released proton a random radial position, (distance from the cluster center) between the radii of the previous and current time steps, to be released at. This is an extra $\mathcal{O}(r)$ step, where r is the number of protons released. Since model runtime scales less than linearly with r (Section 3.4), this is a more favorable option than the first, from the perspective of efficiency.

The purpose of the mathematical model - a testbed for algorithms to be implemented (and further tested) on LACEWING - must be kept in mind throughout the process of working through the listed future extensions. Given the complexity of the reaction and sensors being modelled, there will likely always be simplifications made - and hence scope for more physical accuracy - in the implementation of the model. However, if datasets from the model are realistic enough to facilitate a smooth transition from offline to online algorithms, the need for further improvements to the model diminishes.

5.2 Spatio-Temporal Algorithms

Validation on Experimental Data

First, the spatio-temporal algorithms presented in this work must be validated on experimental data. This poses a number of new challenges and avenues for investigation. First, experimental data has shown some pixels on each chip to (randomly) switch off due manufacturing defects. Due to the lobe-like shape of the manifold, entire regions of the ISFET array are also not exposed to any of the reaction solution. These pixels (together with the faulty ones) must be filtered out during data pre-processing. Temperature pixels, whose output is independent of the spatial release of protons in the solution above, also pose a challenge to the spatial signal. Investigations need to be conducted into whether it is easier to simply remove these pixels from consideration, or replace their output with the average of their immediate neighbors - in which case methods of modelling sensor drift for these also needs to be explored, particularly since spatio-temporal algorithms are more sensitive to individual ISFET drift than purely temporal algorithms that consider the average drift across the entire array. Once the dataset is cleaned, drift compensation can be performed on the appropriate ISFETs and the Algorithms tested.

Improving Robustness to Noisy Data

As shown in Section 5.2 and 5.3, both spatio-temporal algorithms, while promising, show a potential susceptibility to increased variance in noise. These algorithms can be improved upon by exploring the following options:

1. Change the definition of TTP to be when the gradient of the difference signal is positive, instead of maximized. Theoretically, a positive gradient in the difference of standard deviation signal is a strong indication of amplification occurring in the bottom well. While hard-coding this heuristic puts the algorithm at risk of susceptibility to uncontrollable factors, this might be a worthy compromise if experimental data shows that the gradient of the difference in standard deviation signals for two negative experiments is never greater than 0 (or some small constant ϵ). Such an approach could further reduce TTP - all gradient signals plotted in Sections 5.2 and 5.3 start above 0 at $t = 300$ - at the potential expense of a higher false positive rate, which has already been explained to be less important than a low false negative rate.

2. Improve drift curve fitting by increasing the permissible range of drift parameters to be searched. `scipy`'s `curve_fit` function takes possible ranges of parameters to be found as input, in order to make the computation more efficient. These ranges were set based on experimentally observed data in Table 3.4. Therefore, the `curve_fit` function produced more erroneous estimates when the distribution drift parameters was changed for the noisy datasets. Increasing the range of parameter values to be searched will improve the ability of the algorithm to accurately model and compensate for drift, at the potential expense of algorithm runtimes. This needs to be investigated further.

3. `scipy`'s `find_peaks` function works by searching the signal for values that are greater than their neighbors (on either side). The number of neighbors to search is a parameter supplied to the function which therefore controls the number of datapoints considered to be peaks. Since noisier datasets produce less sharp peaks (as seen in Figure 4.18(c)), a smaller value for the neighbors parameter is required for the `find_peaks` function to classify a sample as positive. Instead of hard-coding the value of this parameter, it could be adaptively set by an algorithm, based on other factors such as the maximum magnitude of the difference in standard deviation signal. Considering Figures 4.15 and 4.18(b), the noisier dataset seems to reduce the maximum value of the difference in standard deviation between the top and bottom wells. If this idea can be validated experimentally, an adaptive value for the number of neighbors considered will allow the `find_peaks` function to detect subtler peaks in the gradient signal, thereby more robustly classifying noisy signals.

Alternative methods of Drift Compensation

Another potential approach for drift compensation is to compute and use the absolute difference in output between successive frames to reduce the amount of noise per frame. This method is computationally more efficient than drift modelling, but initial experimentation did not show promising results because the magnitude of amplification-induced voltage changes is similar to that caused by the additional drift per frame - as such, there was minimal improvement in the signal-to-noise ratio.

Alternative methods of Detecting Hotspots

Alternative metrics or models can also be used to detect the presence of clusters. One method attempted was computing the average correlation of every pixel with its immediate neighbors, over a 5-frame horizon. While the idea was promising, since peaks in correlation can be expected in regions of the ISFET array where clusters are starting to form, the algorithm was computationally too expensive to be applicable in the real world, taking 16 seconds per frame to compute on a personal laptop with a 2.6GHz processor. Machine Learning (ML) algorithms can also be explored, the primary advantage of these being the ability to work with raw data - removing the need for computationally expensive drift compensation. However, other practical considerations such as over-fitting and memory usage might restrict the applicability of these models in an embedded setting. Nevertheless, with efficient ML for embedded systems being an active area of research [41], developments in the field should be monitored for new ideas.

6. Conclusions

This work aimed to investigate the potential for spatio-temporal algorithms to reduce the time-to-positive (TTP) for DNA detection using LAMP reactions carried out on an ISFET array. To this end, this project has made two contributions to the field of rapid PoC diagnostics.

The first is a flexible mathematical model for the generation of datasets to train algorithms on. This model improves on the existing one by taking into consideration the physical characteristics of the chip as well as the movement of protons around the reaction chamber, resulting in datasets that better represent the physical distribution of protons around the ISFET array. Additionally, by considering the conversion from protons captured to voltage output as well as modelling ISFET sensor drift and high-frequency noise, the model outputs realistic voltage values for each sensor. Five key assumptions were made during the development of the model in order to achieve a reasonable compromise between physical accuracy and simulation runtime. Each of these has been explicitly stated, explained and justified, with relevant avenues for future investigation proposed. The model brings about two primary benefits to teams working on ISFET-based DNA detection devices. First, algorithms trained on these datasets will be more directly transferable to a real-world setting, improving the speed of RnD. Second, the parameterisation of variables throughout the code base lend it flexibility, thereby reducing the time delay between the development of new chips and the acquisition of training data for novel algorithms. This also brings utility to chip designers, who will be better able to work with data scientists to determine the most areas for improvement on chips.

The second contribution is two novel spatio-temporal algorithms that show the potential to improve TTP by 33.2% compared to a purely temporal algorithm. Both algorithms work by reversing the process by which data is generated by the model - drift compensation is first performed, followed by detection of hotspots using frame-wise standard deviations. These algorithms are favorable to more sophisticated Machine Learning (ML) approaches for a few reasons. First, no pre-training is required, improving the speed of testing and deployment. Second, since the algorithms only consider differences between the top and bottom wells of the same chip, there is no risk of over-fitting to training datasets (which may not account for manufacturing or environmental differences in the real world), which can compromise the accuracy of machine learning approaches. Third, standard deviation is a metric that is fast to compute, making it more suitable for the real-time computations required in a PoC setting on an embedded platform. Fourth, by virtue of not having to save large models and weights, these algorithms are more memory efficient than ML approaches. Last, they are intuitive to understand and more explainable than ML algorithms, many of which operate as a black-box, making the proposed spatio-temporal algorithms easier to debug and more adaptable to new contexts.

The current context of the developed world outpacing developing countries in the recovery from COVID-19 provides the field of rapid and cheap PoC diagnostics with an even greater sense of urgency. Both contributions of this project provide avenues and tools for further exploration by teams working in this field, with the hope of speeding up the workflow of chip designers and data scientists alike.

A. Appendix A

A.1 Codebase

The codebase for this projct comprises that of the mathematical model as well as the two spatio-temporal algorithms propsed. All source code can be provided upon request, please email *kk3415@ic.ac.uk*.

A.1.1 Mathematical Model

The mathematical model was written entirely in MATLAB. The following files are used to generate a dataset:

1. `init_lamp.m` - Set the initial user-provided and internal model parameters (as per Figure 3.16), as well as flags for plotting
2. `run_lamp.m` - Run the LAMP reaction, as per the flowchart in Figure 3.17
3. `convert_to_voltage.m` - Convert the number of protons detected per ISFET to the voltage output of the array, sampled according to F
4. `plot_temporal_signal.m` - Plot the temporal signal of the dataset (voltages or protons)
5. `plot_spatial_signal.m` - Plot the spatial signal of the dataset (voltages or protons)
6. `run_full_simulation.m` - Combines all the above files into one convenient script

Other scripts, used to generate all the figures seen in Sections 4 and 5, are not listed above but also available on request.

A.1.2 Spatio-Temporal Algorithms

Spatio-temporal algorithms were developed in a Jupyter Notebook, `Algorithms.ipynb`, running Python 3.7.1.

References

[1] Johns Hopkins University. *COVID-19 Dashboard by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University (JHU)*. URL: https://coronavirus.jhu.edu/map.html.

[2] World Health Organisation. *Timeline of WHO's COVID-19 Response*. URL: https://www.who.int/emergencies/diseases/novel-coronavirus-2019/interactive-timeline#!.

[3] C. Jonathan R. Selina and M.Martin. *Successful Find-Test-Trace-Isolate-Support Systems: How To Win At Snakes and Ladders*. URL: https://apps.who.int/iris/bitstream/handle/10665/336292/Eurohealth-26-2-34-39-eng.pdf.

[4] Reuters. *Singapore to use rapid coronavirus tests for weddings, events*. URL: https://www.reuters.com/article/us-health-coronavirus-singapore-idUSKBN2751E1.

[5] C.M. Alarcon et al. "Application of DNA- and Protein-Based Detection Methods in Agricultural Biotechnology". In: *Journal of Agricultural and Food Chemistry* 67.4 (2019), pp. 1019–1028. DOI: 10.1021/acs.jafc.8b05157.

[6] Human Genome Research Institute. *Polymerase Chain Reaction (PCR) Fact Sheet*. URL: https://www.genome.gov/about-genomics/fact-sheets/Polymerase-Chain-Reaction-Fact-Sheet.

[7] J. Rodriguez-Manzano et al. "Handheld Point-of-Care System for Rapid Detection of SARS-CoV-2 Extracted RNA in under 20 min". In: *ACS Central Science* (2021). DOI: 10.1021/acscentsci.0c01288.

[8] Britannica. *Polymerase Chain Reaction*. URL: https://www.britannica.com/science/polymerase-chain-reaction.

[9] P. Hardinge and J.A.H Murray. "Reduced False Positives and Improved Reporting of Loop-Mediated Isothermal Amplification using Quenched Fluorescent Primers". In: *Sci Rep* 9.1 (2019). DOI: 10.1038/s41598-019-43817-z.

[10] P. Bergveld. "Thirty years of ISFETOLOGY: What happened in the past 30 years and what may happen in the next 30 years". In: *Sensors and Actuators B: Chemical* 88.1 (2003), pp. 1–20. DOI: 10.1016/S0925-4005(02)00301-5.

[11] S.L.K.T. Tan P. Georgiou and C. Toumazou. "A New Era of Semiconductor Genetics using Ion-Sensitive Field-Effect Transistors: the Gene-Sensitive Integrated Cell". In: *Phil. Trans. R. Soc.* 372.20130112 (2014).

[12] N. Moser et al. "A Scalable ISFET Sensing and Memory Array With Sensor Auto-Calibration for On-Chip Real-Time DNA Detection". In: *IEEE Transactions on Biomedical Engineering* 12.2 (2018), pp. 390–491.

[13] Y.L. Chin et al. "A novel pH sensitive ISFET with on chip temperature sensing using CMOS standard process". In: *Sensors and Actuators, B: Chemical* 76 (2001), pp. 582–593.

[14] Y. Liu et al. "An extended CMOS ISFET model incorporating the physical design geometry and the effects on performance and offset variation". In: *Sensors and Actuators, B: Chemical* 58 (2011), pp. 4414–4422.

[15] A. Elyasi et al. "Counteracting threshold-voltage drift in ion-selective field effect transistors (ISFETs) using threshold-setting ion implantation." In: *IEEE Journal of the Electron Devices Society* 6 (2018), pp. 747–754.

[16] N. Miscourides. "PhD Thesis: Linear ISFET Arrays and Optimisation Methods for DNA Detection". In: (2019).

[17] S. Jamsab. "An analytical technique for counteracting drift in ion-selective field effect transistors (ISFETs)". In: *IEEE Sensors Journal* 4.6 (2004), pp. 795–801.

[18] Robert Kissell and Jim Poserina. *Optimal Sports Math, Statistics, and Fantasy*. Academic Press, 2017, pp. 103–135.

[19] Yale University. *Correlation*. URL: http://www.stat.yale.edu/Courses/1997-98/101/correl.htm.

[20] Stanford University. *CS229 Lecture Notes.* URL: http://cs229.stanford.edu/notes2020fall/notes2020fall/cs229-notes1.pdf.

[21] Muhammad Sapon et al. "Diabetes Prediction with Supervised Learning Algorithms of Artificial Neural Network". In: (Feb. 2021).

[22] Ahmad Moniri. *Final Year Project Report - Novel Algorithms and Models for Sensing DNA through ISFET arrays.*

[23] et. al. Liu C Sadik MM. "Nuclemeter: a reaction-diffusion based method for quantifying nucleic acids undergoing enzymatic amplification." In: *Scientific Reports* 4.7335 (2014).

[24] et. al. Sun B Rodriguez-Manzano J. "Measuring fate and rate of single-molecule competition of amplification and restriction digestion, and its use for rapid genotyping tested with hepatitis C viral RNA." In: *Angew Chem Int Ed Engl* 53.31 (2004), pp. 8088–8092.

[25] P. Geeorgiou. "PhD Thesis: Chemical Bionics - A Novel design approach using Ion Sensitive Field Effect Transistors". In: (2008).

[26] Department of Chemical Engineering Cambridge University and Biotechnology. *The Electrical Double Layer.* URL: https://www.ceb.cam.ac.uk/research/groups/rg-eme/Edu/the-electrical-double-layer.

[27] V. Pachauri and S. Ingebrandt. "Biologically sensitive Field-Effect-Transistors: from ISFETs to nanoFETs". In: *Essays in Biochemistry* 60 (1 2016), pp. 81–90. DOI: 10.1042/EBC20150009.

[28] Jacob N Israelachvilli. *Intermolecular and Surface Forces.* Santa Barbara, California, USA: Academic Press, 2011.

[29] T. Sakata and H. Yang. "Molecular-Charge-Contact-Cased Ion-Sensitive-Field-Effect Transistor Sensor in Microfluidic System for Protien Sensing". In: *MDPI Sensors Journal* 19 (2019), p. 3393. DOI: 10.3390/s19153393.

[30] Science Direct. *Ionic Strength.* URL: https://www.sciencedirect.com/topics/chemistry/ionic-strength.

[31] Kenny Malpartida-Cardenas et al. "Quantitative and rapid Plasmodium falciparum malaria diagnosis and artemisinin-resistance detection using a CMOS Lab-on-Chip platform". In: *Biosensors and Bioelectronics* 145 (2019), p. 111678. ISSN: 0956-5663. DOI: https://doi.org/10.1016/j.bios.2019.111678. URL: https://www.sciencedirect.com/science/article/pii/S0956566319307572.

[32] Howard C Berg. *Random Walks in Biology.* Princeton, New Jersey, USA: Princeton University Press, 1993.

[33] Erban Radek et al. "A practical guide to stochastic simulations of reaction-diffusion processes". In: (May 2007).

[34] Nadav Amdursky and Yiyang et al. Lin. "Exploring fast proton transfer events associated with lateral proton diffusion on the surface of membranes". In: *Proceedings of the National Academy of Sciences* 116.7 (2019), pp. 2443–2451. DOI: 10.1073/pnas.1812351116.

[35] Miyake Takeo and Rolandi Marco. "Grotthuss mechanisms: from proton transport in proton wires to bioprotonic devices". In: *J. Phys.: Condens. Matter* 28.023001 (2016).

[36] P. Georgiou et al K. Rodriguez Manzano N. Moser. "Rapid Detection of Mobilized Colistin Resistance using a Nucleic Acid Based Lab-on-a-Chip Diagnostic Ststem". In: *Sci Rep* 10.8448 (2020).

[37] S. Subramanian and R. D. Gomez. "An Empirical Approach for Quantifying Loop-Mediated Isothermal Amplification (LAMP) Using Escherichia coli as a Model System". In: *PLoS One* 9.6 (2014).

[38] Massachusetts Institute of Technology. *Explained: Sigma.*

[39] E. Cuevas et al. "Fast algorithm for Multiple-Circle detection on images using Learning Automata". In: *IET Image Processing* 6.8 (2012), pp. 1124–1135.

[40] University of California San Francisco. *Inverse Chi Squared Distribution.* URL: https://valelab4.ucsf.edu/svn/3rdpartypublic/boost/libs/math/doc/sf_and_dist/html/math_toolkit/dist/dist_ref/dists/inverse_chi_squared_dist.html.

[41] W. Roth et al. "Resource-Efficient Neural Networks for Embedded Systems". In: (Jan. 2020).