## University of Strathclyde

Flow effects in bistable nematic liquid crystal devices

Matthew Neilson<br>Department of Mathematics<br>University of Strathclyde<br>Glasgow, UK<br>June 2008

This thesis is submitted to the University of Strathclyde for the degree of Doctor of Philosophy in the Faculty of Science.

The copyright of this thesis belongs to the author under the terms of the United Kingdom Copyright Acts as qualified by University of Strathclyde Regulation 3.50. Due acknowledgement must always be made of the use of any material in, or derived from, this thesis.

## Dedication



This thesis is dedicated to my mum, Angela Leishman Neilson (1950-2004).
Wherever you are, I know you're proud of me.

## Acknowledgements

I would like to thank my supervisor, Professor Nigel Mottram, for allowing me to put my own spin on the research whilst ensuring that I didn't stray too far from the original topic. I would also like to thank Dr Chris Newton for his advice and discussions which helped me to obtain a much broader understanding of the subject.

I must also thank Dr John Mackenzie for supplying me with several powerful numerical algorithms, Dr Wankere Mekwi for his helpful discussions, and Alan Campbell for his countless programming-related tips.

Many thanks to EPSRC for my funding and Hewlett Packard Labs for their case award.

Finally, I am indebted to my family for their continued support and encouragement throughout my PhD. To my dad, George Neilson; my sister, Sophie Neilson; and my girlfriend, Lindsay Govan. I don't know what I'd do without you guys - you're the best!


#### Abstract

We consider a nematic liquid crystal device in which a bistable surface anchoring term produces two stable states, a Vertical State (i.e. all molecules are homeotropically aligned) and a Hybrid Aligned Nematic (HAN) State (i.c. the molecules are homcotropically aligned on one boundary and homogeneously aligned on the other). Our one-dimensional model determines the director profile throughout a nematic cell by minimizing its free energy. The free energy in this model contains dielectric, elastic, flexoelectric and anchoring terms. This constitutes what we denote the 'no-flow' model. An expanded, so-called 'flow model', also includes a flow equation that we couple with our system of director equations.

We then introduce three time integration methods for our numerical simulations, namely an explicit method, a semi-implicit method and a fully-implicit method, each of which employs an adaptive time-stepping algorithm to control the size of each time-step. Numerical simulations also employ a moving mesh algorithm to control the positioning and quantity of node points used at each time-step. We then compare each simulation method to determine which provides the optimal balance of speed and accuracy.

We investigate switching for voltage pulses of different magnitude and duration in order to graph standard $\tau V$-plots. Each switching region is determined by the interaction between the bistable surface and bulk equations once the applied voltage is removed, which is a relatively complex process. We develop and present a powerful algorithm for automatically generating $\tau V$-plots corresponding to any given parameter set. Using this algorithm, we then investigate the effect of each parameter on the switching characteristics of our cell, using both the standard model and the expanded 'flow' model.

The effects of flow are investigated by comparing the results of each model via numerical simulation. We show that flow-induced kickback in the director can significantly affect the results obtained using a no-flow model.


## Contents

1 Introduction ..... 1
1.1 What are liquid crystals? ..... 1
1.2 The origins of liquid crystals ..... 2
1.3 Different types of liquid crystals ..... 2
1.4 Liquid crystal theory ..... 4
1.4.1 The director ..... 4
1.4.2 Elasticity ..... 4
1.4.3 Dielectricity ..... 5
1.4.4 Flexoelectricity ..... 6
1.4.5 The Leslie viscosities ..... 7
1.4.6 Surface anchoring ..... 9
1.5 Liquid crystal technology ..... 10
1.5.1 Monostable liquid crystal displays ..... 11
1.5.2 Bistable liquid crystal displays ..... 12
$1.6 \tau V$-plots ..... 14
2 Model ..... 16
2.1 Previous work ..... 16
2.2 Notation ..... 17
2.3 Sctup ..... 18
2.4 Definitions and assumptions ..... 19
2.5 Derivation of the electric field equation ..... 20
2.6 Derivation of the flow velocity equation ..... 22
2.7 Derivation of the director angle equation ..... 23
2.7.1 Elastic energy density ..... 24
2.7.2 Electric energy density ..... 24
2.7.3 The director angle equation ..... 25
2.8 Surface boundary equations ..... 25
2.8.1 Electric potential ..... 25
2.8.2 Flow velocity ..... 26
2.8.3 Director angle ..... 27
2.9 Simplifications ..... 31
2.9.1 Solving for the field equation ..... 31
2.9.2 Neglecting flow ..... 32
2.10 Calculation of the optical transmission ..... 32
2.11 Base parameter set ..... 34
2.12 Summary ..... 34
3 Numerical methods ..... 36
3.1 Spatial discretisation ..... 36
3.2 Time integration ..... 38
3.2.1 Explicit method ..... 38
3.2.2 Semi-implicit method ..... 39
3.2.3 Fully-Implicit method ..... 40
3.3 Ficld calculation ..... 45
3.4 Time adaption ..... 46
3.5 Moving mesh algorithm (MMA) ..... 50
3.5.1 r-refinement ..... 50
3.5.2 $h$-refinement ..... 53
3.5.3 $h r$-refinement ..... 55
3.6 Method comparisons - no-flow model ..... 56
3.6.1 General motivation ..... 57
3.6.2 Solution accuracy ..... 58
3.6.3 Differences between time integration methods ..... 58
3.6.4 Differences between mesh types ..... 61
3.6.5 Extension to other voltages ..... 63
3.6.6 The optimal method ..... 64
3.7 Method comparisons - flow model ..... 66
3.7.1 Solution accuracy ..... 66
3.7.2 Differences between time integration methods ..... 66
3.7.3 Differences between mesh types ..... 69
3.7.4 Extension to other voltages ..... 72
3.7.5 The optimal method ..... 73
3.8 Summary ..... 74
4 Automatic $\tau V$-plot generation ..... 75
4.1 Detecting a switch between states ..... 76
4.2 Finding a point that switches ..... 81
4.3 Finding the boundary of a switching region ..... 82
4.4 Line segment intersection ..... 83
4.5 Is a point inside a polygon? ..... 87
4.6 Tracing around a switching region ..... 89
4.6.1 The basic tracing algorithm ..... 90
4.6.2 Adaptive increments ..... 91
4.6.3 Reaching a dead-end ..... 92
4.6.4 Reaching the cdge of the test grid ..... 94
4.6.5 Closing a region ..... 95
4.7 Post processing ..... 96
4.8 The observed error ..... 98
4.9 Finding additional regions ..... 99
4.10 Limitations ..... 102
4.11 Summary ..... 103
5 Results - no flow model ..... 105
5.1 Parameter fundamentals ..... 105
5.2 The base parameter set ..... 106
5.3 Varying the preferred HAN state ..... 112
5.4 Varying the preferred Vertical state ..... 114
5.5 Varying the flexoelectric constant ..... 117
5.6 Varying the parallel dielectric constant ( $\epsilon_{\perp}$ fixed) ..... 120
5.7 Varying the perpendicular dielectric constant ( $\epsilon_{\|}$fixed) ..... 123
5.8 Varying the dielectric constants ( $\Delta \epsilon$ fixed) ..... 126
5.9 Varying the splay elastic constant ..... 129
5.10 Varying the bend elastic constant ..... 133
5.11 Varying the bistable surface anchoring strength ..... 136
5.12 Varying the monostable surface anchoring strength ..... 139
5.13 Varying the bistable surface relaxation cocfficient ..... 142
5.14 Varying the monostable surface relaxation coefficient ..... 145
5.15 Summary ..... 148
6 Results - flow model ..... 149
6.1 The base parameter set ..... 149
6.1.1 Voltage on ..... 151
6.1.2 Polarity change ..... 155
6.1.3 Voltage off ..... 157
6.2 Varying the preferred HAN state ..... 160
6.3 Varying the preferred Vertical state ..... 164
6.4 Varying the flexoelectric constant ..... 168
6.5 Varying the parallel dielectric constant ( $\epsilon_{\perp}$ fixed) ..... 170
6.6 Varying the perpendicular dielectric constant ( $\epsilon_{\|}$fixed) ..... 174
6.7 Varying the dielectric constants ( $\Delta \epsilon$ fixed) ..... 179
6.8 Varying the splay clastic constant ..... 181
6.9 Varying the bend elastic constant ..... 184
6.10 Varying the bistable surface anchoring strength ..... 187
6.11 Varying the monostable surface anchoring strength ..... 191
6.12 Varying the bistable surface relaxation cocfficient ..... 193
6.13 Varying the monostable surface relaxation coefficient ..... 195
6.14 Varying the partial slip-length ..... 198
6.15 Summary ..... 202
7 Conclusions ..... 204
A Formulation of the bistable surface energy ..... 207
B The flexoelectric effect on the relaxed HAN state 212

## 1 Introduction

### 1.1 What are liquid crystals?

Matter is generally considered to exist in one of three states, namely as a solid, a liquid or a gas. In the case of a crystalline solid the comprising molecules are firmly packed together, maintaining a high degree of positional and orientational order relative to one another, as in Figure 1.1(a). Increasing the temperature by a sufficient amount causes the order amongst the molecules to decrease, producing an isotropic liquid state as shown in Figure 1.1(c). By isotropic, we mean that the physical properties of the substance are uniform in all directions. The gas state is obtained by increasing the temperature further so that the molecules are well-separated, as in Figure 1.1(d). A classic example of this is water, which is a solid below $0^{\circ} \mathrm{C}$, a liquid between $0^{\circ} \mathrm{C}$ and $100^{\circ} \mathrm{C}$, and a gas above $100^{\circ} \mathrm{C}$.


Figure 1.1: The four states of matter: (a) solid, (b) nematic liquid crystal, (c) isotropic liquid and (d) gas. Note that the positional and orientational order of the molecules decreases as the temperature is increased.

Some substances, however, exhibit a fourth phase, the liquid crystal state, which occurs at a temperature between the crystalline solid and isotropic liquid phases. In the liquid crystal state, the molecules possess some properties of both the crystalline solid and isotropic liquid states, as illustrated in Figure 1.1(b). Like a solid, the molecules retain some degree of order relative to one another, yet they also flow in the same sense as a liquid flows. The example given in Figure 1.1(b) corresponds to the nematic phase, in which orientational order is preserved but positional order is lost.

### 1.2 The origins of liquid crystals

It is generally accepted that liquid crystals were first identified in 1888 by Friedrich Reinitzer, an Austrian botanist [40]. When heating a sample of cholesteryl benzoate, Reinitzer observed what he believed to be two separate liquid states. The first occurred at $145.5^{\circ} \mathrm{C}$ and took the form of a cloudy liquid, whilst at $178.5^{\circ} \mathrm{C}$ the material transformed into a clear liquid. The 'cloudy liquid' is what we now refer to as the liquid crystal state. Reinitzer contacted Otto Lehmann, a respected German physicist, detailing his findings and requesting further analysis to be performed on his sample. In 1889 Lehmann used the term 'flowing crystals' to describe the material [23], and by 1900 he had coined the term 'liquid crystal'.

### 1.3 Different types of liquid crystals

It is possible to induce the liquid crystal phase in one of two ways: either thermotropically or lyotropically. Thermotropic liquid crystals are induced by changing the temperature of a substance. Lyotropic liquid crystals are formed by changing the concentration in a solvent. Throughout this thesis, we will only concern ourselves with thermotropic liquid crystals.


Figure 1.2: Examples of the (a) nematic, (b) smectic-A and (c) smectic-C liquid crystal phases.

Any liquid crystal may be classified as belonging to one of following categories: nematic, smectic, cholesteric and discotic. It is convenient to think of the liquid crystal molecules as thin and rod-shaped. In the nematic state, the molecules tend to align themselves roughly parallel to one another, with a high level of orientational order but without any positional order, as in Figure 1.2(a). Unlike nematics, smectic liquid crystal molecules have some degree of positional order and thus form layers, with a well-defined inter-layer spacing. The simplest type of smectic, smectic- $A$, is shown in Figure 1.2(b). The smectic- $C$ phase is similar to the smectic-A phase only now the molecules have a constant tilt angle on each layer,
as in Figure 1.2(c).


Figure 1.3: An example of a cholesteric liquid crystal.

Cholesteric liquid crystals, sometimes referred to as chiral-nematic liquid crystals, are similar to nematic liquid crystals only the preferred configuration of the molecules contains some kind of twist, or helical, structure. The 'pitch' of a cholesteric liquid crystal is defined as the distance for the molecules to rotate one revolution of the helix, as illustrated in Figure 1.3.


Figure 1.4: An example of the (a) nematic and (b) columnar discotic liquid crystal phases.

In the discotic liquid crystal phases, we imagine the molecules as being disc-shaped. The molecules in the discotic nematic phase, illustrated in Figure 1.4(a), have no positional order, but there is some orientational order between the molecules. In the columnar discotic phase, the molecules have a high degree of positional order and tend to order themselves in a similar fashion to a neatly stacked pile of poker chips, as in Figure 1.4(b).

Throughout this thesis, we will only concern ourselves with nematic liquid crystals.

### 1.4 Liquid crystal theory

### 1.4.1 The director

The elongated nature of liquid crystal molecules means that they can prefer to lie in a more ordered state, the nematic state, with the molecules roughly parallel to one another. The director, $\mathbf{n}$, is the unit vector that is used to represent the average direction of the molecules, as shown in Figure 1.5.


Figure 1.5: The director, $\mathbf{n}$, is a unit vector which represents the average director of the molecules.

Notice, from Figure 1.5, that due to the symmetry of the molecules $\mathbf{n}$ and $-\mathbf{n}$ are physically indistinguishable from each other. It should be noted that, in some instances, a liquid crystal may possess some form of polarity, in which case the molecules are no longer symmetrical and we may no longer have $\mathbf{n} \equiv-\mathbf{n}$.

### 1.4.2 Elasticity

In physical terms, a material is said to be elastic if it deforms under stress and then returns to its original shape when the stress is removed. A standard rubber band is an everyday example of a material with elastic properties. As it happens, liquid crystals also exhibit elastic properties. This is an example of a common property between liquid crystals and solids.

Imagine we have a liquid crystal sandwiched between two plates. Applying some force to the bounding plates will induce an elastic distortion in the enclosed liquid crystal. By compressing the plates together, the liquid crystal molecules may either splay, as in Figure 1.6(a), or bend as shown in Figure 1.6(c). By twisting just one of the bounding plates, we may


Figure 1.6: The elastic distortions relating to (a) $K_{1}$ (splay), (b) $K_{2}$ (twist), (c) $K_{3}$ (bend).
produce a twist distortion as shown in Figure 1.6(b). These splay, twist and bend distortions are directly related to the Frank elastic constants $K_{1}, K_{2}$ and $K_{3}$, respectively [14]. The saddle-splay, $\left(K_{2}+K_{4}\right)$, makes up the fourth elastic constant. Note that the saddle-splay is usually omitted since it as it does not contribute to the bulk behavior when the director is confined to a single plane [2].

### 1.4.3 Dielectricity

Each molecule in a liquid crystal contains some distribution of charges. The dielectric tensor, $\epsilon$, determines how easily the distribution of charge may be changed in a group of molecules. Specifically, the dielectric permittivities, $\epsilon_{\|}$and $\epsilon_{\perp}$, measure, respectively, how easily charges may be redistributed parallel and perpendicular to the director. The dielectric anisotropy, $\Delta \epsilon$, is then defined by the expression

$$
\begin{equation*}
\Delta \epsilon=\epsilon_{\|}-\epsilon_{\perp} . \tag{1.1}
\end{equation*}
$$

Note that both $\epsilon_{\|}$and $\epsilon_{\perp}$ are unitless quantities.

When a voltage is applied across a liquid crystal material, the electric field will tend to distort the distribution of charges within a group of liquid crystal molecules, creating a small dipole which then causes the director to rotate to align with the electric field, as shown in Figure 1.7. This is an example of the dielectric effect. In relation to the dielectric permittivities, notice that with $\Delta \epsilon<0$ (i.e. $\epsilon_{\perp}>\epsilon_{\|}$), the preferred orientation of the molecules is perpendicular to the applied field whilst, with $\Delta \epsilon>0$ (i.e. $\epsilon_{\perp}<\epsilon_{\|}$), the preferred orientation of the molecules is parallel to the applied field. Furthermore, by comparing Figure 1.7(a) with Figure 1.7(b), notice that the dielectric effect is the same regardless of the polarity of the
(a) field up
(i) $\Delta \varepsilon<0$
(ii) $\Delta \varepsilon>0$

(b) field down
(i) $\Delta \varepsilon<0$

(ii) $\Delta \varepsilon>0$


Figure 1.7: The dielectric effect when an electric field is applied (a) upwards, (b) downwards. Note from plots (a) and (b) that the same orientation is produced in each case.
applied field.

### 1.4.4 Flexoelectricity

The flexoelectric effect $[32,11]$ is typically observed when the liquid crystal molecules are asymmetric in shape, with each containing a small permanent dipole. In the absence of an electric field, the molecules orient themselves in such a way that the molecular dipoles cancel, giving zero overall dipole throughout the cell. Upon applying an electric field the dipoles align themselves relative to the field and, due to the asymmetry of the molecules, the liquid crystal experiences some form of elastic distortion.


Figure 1.8: The flexoelectric effect with pear-shaped molecules, inducing a splay distortion. Note from plots (b) and (c) that different field polarities produce different distortions.

For example, let us consider the case in which our liquid crystal molecules are pear-shaped,
as in Figure 1.8(a). Once an electric field is applied, the molecules reorient themselves producing a splay distortion as shown in Figure 1.8(b). Note that this distortion occurs since the configuration in Figure 1.8(b) is the easiest way for the molecules to fill space, given their shape. We see from Figure 1.8(c) that applying an electric field in the opposite direction also induces a splay distortion, this time in the opposite direction. This highlights a key aspect of the flexoelectric effect, namely that, in contrast with the dielectric effect, we observe different distortions depending on the polarity of the applied field.
(a) no field

(b) field up

(c) field down


Figure 1.9: The flexoelectric effect with banana-shaped molecules, inducing a bend distortion. Note from plots (b) and (c) that different field polarities produce different distortions.

We now consider the case in which our liquid crystal molecules are banana-shaped, as in Figure 1.9(a). From Figure 1.9(b), notice that applying an electric field in this case induces a bend distortion. As in the splay case, we see from Figure 1.9(c) that the bend distortion acts in the opposite direction when the field changes polarity.

### 1.4.5 The Leslie viscosities

Due to the fluid nature of liquid crystals, any mass movement of molecules (i.e. a flow) will be associated with an internal friction or viscosity. In an isotropic liquid, there is only one viscosity coefficient which is defined in terms of the shear rate and the pressure tensor. However, in a nematic liquid crystal there are six viscosity coefficients, denoted $\tilde{\alpha}_{i}$, which are known as the Leslie viscosities. These viscosities are typically quite difficult to describe in physical terms. The Leslie viscosities can be represented in terms of the more easily understood Miesowicz viscosities.

Figure 1.10 gives a graphical illustration of the Miesowicz viscosities. The first three, $\eta_{1}, \eta_{2}$ and $\eta_{3}$ relate to the shear flow when $\mathbf{n}$ is parallel to $\mathbf{u}$, parallel to $\nabla \mathbf{u}$ and orthogonal to both $\mathbf{u}$ and $\nabla \mathbf{u}$, respectively. The fourth viscosity, $\gamma_{1}$, relates to the rotational viscosity of the director. A fifth viscosity, $\eta_{12}$ enters the effective viscosity when the director is not aligned


Figure 1.10: The Miesowicz viscosities, (a) shear viscosity $\eta_{1}$, (b) shear viscosity $\eta_{2}$, (c) shear viscosity $\eta_{3}$ and (d) rotational viscosity $\gamma_{1}$. In each case, the shear is in the $x z$-plane with the velocity $\mathbf{u}$ indicated by the arrows.
with any axis.

As previously stated, the Miesowicz viscosities can be represented in terms of the Leslie viscosities, by

$$
\begin{align*}
\eta_{1} & =\left(\tilde{\alpha}_{2}+2 \tilde{\alpha}_{3}+\tilde{\alpha}_{4}+\tilde{\alpha}_{5}\right) / 2  \tag{1.2}\\
\eta_{2} & =\left(-\tilde{\alpha}_{2}+\tilde{\alpha}_{4}+\tilde{\alpha}_{5}\right) / 2  \tag{1.3}\\
\eta_{3} & =\tilde{\alpha}_{4} / 2  \tag{1.4}\\
\gamma_{1} & =\tilde{\alpha}_{3}-\tilde{\alpha}_{2}  \tag{1.5}\\
\eta_{12} & =\tilde{\alpha}_{1} \tag{1.6}
\end{align*}
$$

For completeness, we also list the inverse relation which allows us to change from the Leslie viscosities to the Miesowicz viscosities,

$$
\begin{align*}
& \tilde{\alpha}_{1}=\eta_{12},  \tag{1.7}\\
& \tilde{\alpha}_{2}=\left(\eta_{1}-\eta_{2}-\gamma_{1}\right) / 2,  \tag{1.8}\\
& \tilde{\alpha}_{3}=\left(\eta_{1}-\eta_{2}+\gamma_{1}\right) / 2,  \tag{1.9}\\
& \tilde{\alpha}_{4}=2 \eta_{3},  \tag{1.10}\\
& \tilde{\alpha}_{5}=\left(\eta_{1}+3 \eta_{2}-4 \eta_{3}-\gamma_{1}\right) / 2,  \tag{1.11}\\
& \tilde{\alpha}_{6}=\left(3 \eta_{1}+\eta_{2}-4 \eta_{3}-\gamma_{1}\right) / 2 . \tag{1.12}
\end{align*}
$$

### 1.4.6 Surface anchoring

If a liquid crystal is sandwiched between two surfaces then the molecules can be expected to interact with each surface, causing the director to orient in some preferred, or 'easy', direction. Depending on how each surface is prepared, typically by some form of alignment layer or rubbing, it is possible to control this preferred direction.


Figure 1.11: Two different types of molecular surface alignment, (a) homogeneous alignment, and (b) homeotropic alignment.

Two common types of anchoring are shown in Figure 1.11. With homogeneous (sometimes referred to as planar) alignment, the molecules prefer to align themselves parallel to the surface at each boundary. With homeotropic alignment, the molecules prefer to align themselves perpendicular to the surface at each boundary. There is a third type of anchoring, called conical anchoring (see Figure 1.12), in which the director prefers to make a fixed angle $\alpha$ with the surface boundary.


Figure 1.12: Conical anchoring of the director, $\mathbf{n}$, at a surface boundary. The 'cone of easy directions' makes an angle $\alpha$ with the surface boundary.

For each different type of anchoring, it is also important to note the strength with which the director is anchored to the surface. We refer to this as the anchoring strength. There are two different types of anchoring strength: weak anchoring and strong (or infinite) anchoring.

In the case of weak anchoring the director at the surface boundary is easily influenced by the bulk of the cell, whilst with strong anchoring the director remains fixed in its preferred direction regardless of the bulk effects.

### 1.5 Liquid crystal technology

Light is affected as it passes through a liquid crystal in different ways depending on the orientation of the liquid crystal molecules. With the aid of certain optical filters, this effect can be used to actively change the amount of light passing through the material. As we have already seen, applying a voltage to a liquid crystal can alter the molecular configuration, and so it is possible to change the liquid crystal from optically opaque to transparent. This makes liquid crystals ideal for display devices.

An extremely important factor in portable electronic device technology, such as mobile phones, laptop computers and handheld games consoles, is the battery lifetime. Since a large proportion of the battery is used to power a liquid crystal display (LCD), decreasing the energy consumption of the LCD will significantly increase the length of time between battery recharges.


Figure 1.13: An example of a single pixel in a liquid crystal display.

Typical liquid crystal displays (LCDs) consist of an array of pixels, each containing a region of liquid crystal sandwiched between two glass plates, as illustrated in Figure 1.13. Electrodes are placed around the liquid crystal so that a voltage may be applied in order to change the orientation of the liquid crystal molecules. Two polarisers are placed around the
device so that the two states, voltage on and voltage off, look dark and light.

### 1.5.1 Monostable liquid crystal displays

As the name suggests, a monostable liquid crystal display contains a single stable equilibrium state. Therefore, a monostable pixel requires a constant power supply to maintain any kind of non-equilibrium distortion. This is illustrated in Figure 1.14. We see that the only stable equilibrium state is, in this example, homeotropic (Figure 1.14(a) and Figure 1.14(c)). By applying an electric field, as in Figure 1.14(b), the molecular configuration changes which also changes the amount of light that can pass through the cell. Upon removing the electric field, the molecules revert to their stable equilibrium state, as in Figure 1.14(c).


Figure 1.14: An example of a monostable cell changing configurations when a voltage is applied.

Extending this to a full monostable display, each pixel requires a regular supply of power in order to display an image, as shown in Figure 1.15. To obtain a dark-coloured pixel we must apply an electric field to that particular pixel. Upon removing the electric field, the liquid crystal relaxes to its equilibrium state and the pixel reverts to white.

The constant draw of power in a monostable LCD leads to a large overall power consumption. However, LCD devices developed by companies such as DERA (now QinetiQ) and Hewlett-Packard have managed to overcome the power consumption problem associated with monostable LCDs. These more power-efficient LCDs are known as bistable liquid crystal displays.


Figure 1.15: An example of displaying and removing an image on an array of pixels using traditional (monostable) liquid crystal technology.

### 1.5.2 Bistable liquid crystal displays

Unlike monostable liquid crystals, bistable liquid crystal displays contain two stable equilibrium states. This can be achieved by treating one surface, known as the bistable surface, so that there are two 'easy directions' for the molecules at that surface. This can give rise to two distinct molecular configurations that are stable.


Figure 1.16: An example of switching a bistable cell from one state to the other using a negative voltage.

Figure 1.16 gives an example of switching from one stable configuration to the other by applying an electric field to the cell. Here the bistable surface allows for both homogeneous and homeotropic alignment, as in Figure 1.16(a) and Figure 1.16(c), respectively. When an electric field is applied to the cell, the molecules distort as in Figure 1.16(b), breaking the bistable surface anchoring in the process. Upon removing the electric field, the molecules at the bistable surface remain in their new configuration and the bulk of the cell relaxes to the state shown in Figure 1.16(c). Figure 1.16 illustrates that, by changing the polarity of the electric field, it can be possible to switch in the opposite direction.

Extending this to a full bistable display, note that once a pixel has been switched to a dark or light state the field can be removed without causing the pixel to revert to its initial state,


Figure 1.17: An example of switching a bistable cell from one state to the other using a positive voltage.
as illustrated in Figure 1.18. This means that a static image on a bistable display does not consume any power, leading to an enormous reduction in the overall power consumption of the device.


Figure 1.18: An example of displaying and removing an image on an array of pixels using bistable liquid crystal technology.

An example of a bistable device is the ZBD (Zenithally Bistable Device), developed by DERA (now QinetiQ) [21], which exhibits a grating morphology on the lower surface and allows for two distinct director structures $[7,5,6,35,36,1]$. It is the ZBD that we will attempt to model in this thesis.

Figure 1.19 gives a cross-sectional illustration of a ZBD. The first stable state (Figure 1.19(a)) exhibits homeotropic alignment throughout the cell, whilst the second stable state (Figure $1.19(\mathrm{~b}))$ is stable due to the unusual molecular configurations, known as defects, at the bistable surface. These defects give an effective planar alignment just above the bistable surface. Switching from one stable state to the other is achieved by applying a voltage to the cell.

It should be noted for completeness that another type of bistable display is the PABN

## (a) State 1


(b) State 2


Figure 1.19: The two stable states of a ZBD. Here the top surface is monostable, whilst the bottom surface is a complex bistable surface.
(Post-Aligned Bistable Nematic), developed by Hewlett-Packard Laboratories [25], which is bistable due to one surface containing many small 'posts' [24].

## $1.6 \tau V$-plots

The key aspect of bistable liquid crystal devices is their ability to switch from one state to the other with the application and removal of an electric field. We can represent this behaviour using something called a $\tau V$-plot. In simple terms, a $\tau V$-plot is a graph which indicates whether or not a bistable cell switches between states when a specific voltage, $V$, is applied for a prescribed length of time, $\tau$. Such plots are extremely useful for quantifying and understanding the operating voltages and switching characteristics of a bistable cell. Much of our research is concerned with generating and interpreting $\tau V$-plots.

Figure 1.20 gives an example of two $\tau V$-plots [24], each of which was generated using experimental data, where the grey shaded areas represent switching between states. So, for example, applying a voltage of +40 volts for 6 milliseconds will switch from the ' T ' state to the ' P ' state (by Figure 1.20(a)), but not from the ' P ' state to the ' T ' state (by Figure $1.20(\mathrm{~b}))$. Here ' P ' implies a planar state (equivalent to our Vertical state), whilst ' T ' implies a tilted state (equivalent to our Hybrid Aligned Nematic state). Note that, in each case, switching is observed for both positive and negative voltages.
(a) $T$ to $P$

(b) P to T


Figure 1.20: Switching results for a PABN device filled with ZLI-4788-000. Data recorded at 80C. Monopolar square pulses are applied to the device - the $y$-axis is the pulse amplitude in volts, and the $x$-axis is the pulses duration in ms. Pulses that result in switching to the opposite stable state are shown as the grey shaded area. Figure courtesy of Hewlett Packard Laboratories in Bristol.

## 2 Model

In this chapter, we derive the governing equations for a nematic Zenithal Bistable Device (ZBD), using a similar approach to Davidson [9]. To achieve this, we employ various mathematical techniques such as Ericksen-Leslic theory [27] and the Maxwell equations. Additionally, by considering the physical device we wish to model, we establish an appropriate set of boundary conditions for our model. Finally, we apply some simplifications to the model.

### 2.1 Previous work

There exist many two-dimensional $[36,33,17]$ and even three-dimensional $[13,20]$ models for liquid crystal devices, many of which employ $Q$-tensor theory $[39,33]$ in order to model the device. Such models allow for accurate representations of any complex surface morphologies and defect structures that are present within the cell. This is particularly useful when modelling bistable devices, which typically contain some complex groove (or grating) structure at the bistable surface. Some models represent this groove structure as a sinusoidal function, thereby allowing for the effect of the groove-depth and the symmetry of the surface to be investigated [17, 18].

The advantages of multi-dimensional modelling come with relatively high computational costs, which often makes such models impractical. For example, a bistable liquid crystal device may be physically developed and its associated $\tau V$-plot produced in roughly half of one day. Unless a mathematical model can be numerically simulated in less than that time, for the purposes of this work there is little advantage in modelling the device at all.

However, Davidson and Mottram [10] and Davidson [9] showed that a bistable device may be represented using a one-dimensional model by replacing the complex bistable surface with a flat surface which is governed by an effective surface energy function. This simplification allows a bistable device to be modelled without considering the complex groove-structures associated with the bistable surface, which greatly simplifies analytical and numerical investigations. Further research [37, 18] indicates that such effective energy approximations can produce remarkably similar results to the more complex two-dimensional models. Moreover, the simplification to one dimension greatly reduces the required computational effort to simulate a bistable device. It is therefore this approach that we wish to take for constructing
our model.

It should be noted that some unique research by Cummings and Richardson [8] showed that it is possible to obtain bistability using two monostable surfaces, by applying an clectric field at some arbitrary angle of orientation to the cell. The key to achieving this is to have one surface $\pi / 2$ out of phase with the other. So, for example, if the preferred director orientation is $\alpha$ at one surface then $(\alpha-\pi / 2)$ is the preferred orientation at the other surface.

### 2.2 Notation

Throughout this section, we will often use index notation [42]. In the usual system of basis vectors $\left\{\mathbf{e}_{1}, \mathbf{e}_{2}, \mathbf{e}_{3}\right\}$ in $\mathbb{R}^{3}$, a vector $\mathbf{a}=\left(a_{1}, a_{2}, a_{3}\right)$ can be written as

$$
\begin{equation*}
\mathbf{a}=a_{1} \mathbf{e}_{1}+a_{2} \mathbf{e}_{2}+a_{3} \mathbf{e}_{3}=\sum_{i=1}^{3} a_{i} \mathbf{e}_{i} \tag{2.1}
\end{equation*}
$$

which can be expressed, using the Einstein summation convention, as

$$
\begin{equation*}
\mathbf{a}=a_{i} \mathbf{e}_{i} \tag{2.2}
\end{equation*}
$$

where it is understood that the repeated index $i$ is summed from, in this example, 1 to 3. This summation convention obeys the rule that whenever an index appears twice, and only twice, in the same term, a summation is implied over all the contributions obtained by letting that particular index assume all its possible values, unless an explicit statement is made to the contrary.

One useful quantity that is employed throughout this chapter is the alternator, $\epsilon_{i j k}$, defined by

$$
\epsilon_{i j k}=\left\{\begin{array}{rl}
1 & i, j, k \text { unequal and in cyclic order, }  \tag{2.3}\\
-1 & i, j, k \text { unequal and in non-cyclic order, } \\
0 & \text { any two of } i, j, k \text { equal one another }
\end{array}\right.
$$

where $i, j$ and $k$ can each take any of the values 1,2 and 3.

The scalar product of two vectors $\mathbf{a}=\left(a_{1}, a_{2}, a_{3}\right)$ and $\mathbf{b}=\left(b_{1}, b_{2}, b_{3}\right)$ is defined by

$$
\begin{equation*}
\mathbf{a} \cdot \mathbf{b}=a_{i} b_{i} \tag{2.4}
\end{equation*}
$$

whilst the vector product is defined by

$$
\begin{equation*}
\mathbf{a} \times \mathbf{b}=\mathbf{e}_{i} \epsilon_{i j k} a_{j} b_{k} . \tag{2.5}
\end{equation*}
$$

For partial differentiation, $a_{i, j}$ implies that the $i$ th component of the vector $\mathbf{a}$ is differentiated with respect to the $j$ th variable.

### 2.3 Setup

We wish to construct a mathematical model for the director profile throughout a nematic ZBD. In order to achieve this, we must first consider the physical cell itself $[35,36]$ and then make any necessary simplifications.


Figure 2.1: Diagram of a two dimensional nematic cell, with electrodes at $z=-p$ and $z=d+a$. We set $z=0$ at the simplified 'surface', which is taken to be slightly above the true bistable surface. Note that the two stable equilibrium states reduce to (a) an effective Vertical state, and (b) an effective HAN state.

Due to the complex structure of the bistable surface, the cell exhibits two stable states. The Vertical state, shown in Figure 2.1(a), implies homeotropic alignment throughout the cell, whilst the Hybrid Aligned Nematic (HAN) state, shown in Figure 2.1(b), implies homogeneous alignment at the lower boundary and homeotropic alignment at the upper boundary.

The complex structure of the bistable surface is somewhat difficult to model, particularly when considering the emergence and nature of any defects, and so we make the immediate simplification of replacing the bistable surface with a simplified one-dimensional 'surface' that is governed by an appropriate bistable surface energy function. The basic form of the surface energy function must be such that it contains two minima, corresponding to each of
the stable states in the cell that is being modelled.

It should be noted that, although we only intend to model a ZBD, our model may also be used to model other bistable devices such as the PABN. This is due to our bistable surface simplification, which allows us to model the two stable equilibrium states regardless of the nature of the true bistable surface.

### 2.4 Definitions and assumptions

In order to satisfactorily model the bistable device, we need to consider the nematic director, the fluid velocity and the electric ficld within the liquid crystal layer. With these variables we will be able to model the electric field induced switching of the device. We will see later that the inclusion of flow is critical to obtain accurate results.

Our model will therefore comprise three dependent variables: the director, $\mathbf{n}$, the flow velocity, $\mathbf{u}$, and the electric field, $\mathbf{E}$, all of which may depend on spatial coordinates and time. However, we will make certain simplifying assumptions. Our first assumption is to assume symmetry in the $x y$-plane, implying that each of our dependent variables only depend on $z$. Note that this assumption is reasonable, provided the simplified surface in Figure 2.1 is taken to be sufficiently high above the two-dimensional bistable surface.

The director, $\mathbf{n}$, is defined as the average direction of the molecular alignment. Mathematically, we represent this in one dimension as

$$
\begin{equation*}
\mathrm{n}(z, t)=(\cos (\theta(z, t)), 0, \sin (\theta(z, t))) \tag{2.6}
\end{equation*}
$$

where $\theta(z, t)$ is the tilt angle relative to the $x$-axis, as illustrated in Figure 2.2.

Here we have made the assumption that the director always lies in the $x z$-plane, implying that there cannot be any $t w i s t$ in the cell. Also note that, by definition, $\mathbf{n}$ is a unit vector.

The flow velocity in the cell is defined as the macroscopic velocity of a group of molecules at a specific point in time. Given the assumption that the director only lics in the $x z$-plane, it is reasonable to assume that this is also the case for the flow velocity, $\mathbf{u}$. Since the cell we are modelling is bounded at each surface (Figure 2.1), notice that we cannot have any flow


Figure 2.2: The director, $n$ which is represented only by the tilt angle, $\theta$.
in the $z$-direction since this would lead to a flux out of or into the cell at $z=0$ and $z=d$. Therefore, we may assume that there is only flow in the $x$-direction, leading to

$$
\begin{equation*}
\mathbf{u}(z, t)=(u(z, t), 0,0) \tag{2.7}
\end{equation*}
$$

where $u(z, t)$ is the flow velocity in the $x$-direction.

If the cell is constructed such that there are planar electrodes at $z=-p$ and $z=d+a$, where $a$ is the thickness of the alignment layer and $p$ is the thickness of the non-planar structure then we assume that the electric field throughout the cell is only in the $z$-direction, giving

$$
\begin{equation*}
\mathrm{E}(z, t)=\left(0,0, E_{3}(z, t)\right) \tag{2.8}
\end{equation*}
$$

We can now derive the governing equations for the electric field, the flow velocity and the director angle.

### 2.5 Derivation of the electric field equation

In the absence of free charges, Maxwell equations for an clectric field are

$$
\begin{align*}
\nabla \times \mathbf{E} & =0  \tag{2.9}\\
\nabla \cdot \mathbf{D} & =0 \tag{2.10}
\end{align*}
$$

where D is the displacement ficld. Note that equation (2.9) immediately implics that E may be written as the gradient of a potential function, $U$, such that

$$
\begin{equation*}
\mathbf{E}=-\nabla U \tag{2.11}
\end{equation*}
$$

which, by equation (2.8), reduces to

$$
\begin{equation*}
\mathrm{E}=\left(0,0,-\frac{\partial U}{\partial z}\right) . \tag{2.12}
\end{equation*}
$$

The displacement field in equation (2.10) is given by [42, 9 ]

$$
\begin{equation*}
\mathrm{D}=\epsilon_{0} \epsilon \cdot \mathbf{E}+\mathbf{P}_{S} \tag{2.13}
\end{equation*}
$$

with $\epsilon_{0}$ the permittivity of free space and $\epsilon$ the dielectric tensor which, for our system, is given by [9]

$$
\epsilon=\left[\begin{array}{ccc}
\epsilon_{\perp} \sin ^{2} \theta+\epsilon_{\|} \cos ^{2} \theta & 0 & -\Delta \epsilon \sin \theta \cos \theta  \tag{2.14}\\
0 & \epsilon_{\perp} & 0 \\
-\Delta \epsilon \sin \theta \cos \theta & 0 & \epsilon_{\|} \sin ^{2} \theta+\epsilon_{\perp} \cos ^{2} \theta
\end{array}\right]
$$

where $\epsilon_{\|}$and $\epsilon_{\perp}$ are the parallel and perpendicular dielectric coefficients, respectively, whilst $\Delta \epsilon=\epsilon_{\|}-\epsilon_{\perp}$ is the dielectric anisotropy. In our system the only spontancous polarisation, $\mathbf{P}_{S}$, is assumed to be derived from the flexoelectric effect so that $[3,28]$

$$
\begin{equation*}
\mathbf{P}_{S}=e_{11} \mathbf{n}(\nabla \cdot \mathbf{n})+e_{33}((\nabla \times \mathbf{n}) \times \mathbf{n}) \tag{2.15}
\end{equation*}
$$

which in one dimension, using (2.6), reduces to

$$
\begin{equation*}
\mathbf{P}_{S}=\left(\theta^{\prime}\left(e_{11} \cos ^{2} \theta-e_{33} \sin ^{2} \theta\right), 0, E_{13} \theta^{\prime} \sin (2 \theta)\right) \tag{2.16}
\end{equation*}
$$

where we have adopted the convention that ' $\equiv \frac{\partial}{\partial z}$. Here $e_{11}$ and $e_{33}$ are the flexoelectric coefficients and $E_{13}$ is the average flexoelectric constant, defined as

$$
\begin{equation*}
E_{13}=\frac{e_{11}+e_{33}}{2} \tag{2.17}
\end{equation*}
$$

From equations (2.13), (2.14), (2.16) and (2.12), we therefore have

$$
\begin{equation*}
D_{3}=-\epsilon_{0}\left(\epsilon_{\|} \sin ^{2} \theta+\epsilon_{\perp} \cos ^{2} \theta\right) \frac{\partial U}{\partial z}+E_{13} \frac{\partial \theta}{\partial z} \sin (2 \theta) . \tag{2.18}
\end{equation*}
$$

Now, in one dimension, equation (2.10) implies that

$$
\begin{equation*}
\frac{\partial D_{3}}{\partial z}=0 \tag{2.19}
\end{equation*}
$$

and so, by equation (2.18), we have

$$
\begin{equation*}
\frac{\partial}{\partial z}\left[-\epsilon_{0}\left(\epsilon_{\|} \sin ^{2} \theta+\epsilon_{\perp} \cos ^{2} \theta\right) \frac{\partial U}{\partial z}+E_{13} \frac{\partial \theta}{\partial z} \sin (2 \theta)\right]=0 \tag{2.20}
\end{equation*}
$$

which we will refer to as the potential equation. This is the governing equation for the electric potential, $U(z, t)$, which by (2.12) defines the electric ficld.

### 2.6 Derivation of the flow velocity equation

The classical model of a nematic liquid crystal cell consists of the Ericksen-Leslie dynamic equations [27] which are derived from a series of balance laws. Note that a comprehensive derivation is presented in Leslie [27] and Stewart [42], but for our purposes it is sufficient just to state the dynamic equations. The flow equation is derived from the balance of linear momentum, and is given by

$$
\begin{equation*}
\rho \dot{u}_{i}=\left(\frac{\partial \hat{\mathcal{D}}}{\partial u_{i, j}}\right)_{, j}-\frac{\partial \hat{\mathcal{D}}}{\partial \dot{\theta}} \theta_{, i}-\tilde{p}_{, i}, \quad(i=1,2,3) \tag{2.21}
\end{equation*}
$$

where

$$
\begin{equation*}
\hat{\mathcal{D}}=\frac{1}{2} g(\theta)\left(\frac{\partial u}{\partial z}\right)^{2}+m(\theta) \frac{\partial \theta}{\partial t} \frac{\partial u}{\partial z}+\frac{1}{2} \gamma_{1}\left(\frac{\partial \theta}{\partial t}\right)^{2}, \tag{2.22}
\end{equation*}
$$

is the reformulated dissipation function, with

$$
\begin{align*}
m(\theta) & =\tilde{\alpha}_{3} \cos ^{2}(\theta)-\tilde{\alpha}_{2} \sin ^{2}(\theta)  \tag{2.23}\\
g(\theta) & =\frac{1}{2}\left[\tilde{\alpha}_{4}+\left(\tilde{\alpha}_{5}-\tilde{\alpha}_{2}\right) \sin ^{2}(\theta)+\left(\tilde{\alpha}_{3}+\tilde{\alpha}_{6}\right) \cos ^{2}(\theta)\right]+\tilde{\alpha}_{1} \sin ^{2}(\theta) \cos ^{2}(\theta) \tag{2.24}
\end{align*}
$$

and a superposed dot represents the usual material time derivative

$$
\begin{equation*}
\frac{D}{D t}=\frac{\partial}{\partial t}+\mathbf{u} \cdot \frac{\partial}{\partial \mathbf{x}}=\frac{\partial}{\partial t}+u_{i} \frac{\partial}{\partial x_{i}} . \tag{2.25}
\end{equation*}
$$

Here, the $\tilde{\alpha}_{i}$ are the Leslie viscosities and $\gamma_{1}$ is the rotational viscosity defined in Section 1.4.5,
and the superposed dot is used to represent the usual material time derivative defined by equation (2.25). The fluid density and the relative pressure are given by $\rho$ and $\tilde{p}$, respectively. Using equation (2.21) with equation (2.7), we have

$$
\begin{align*}
\rho \dot{u}_{1} & =\frac{\partial}{\partial z}\left(\frac{\partial \hat{\mathcal{D}}}{\partial u^{\prime}}\right)-\frac{\partial \tilde{p}}{\partial x}  \tag{2.26}\\
0 & =-\frac{\partial \tilde{p}}{\partial y}  \tag{2.27}\\
0 & =-\theta^{\prime} \frac{\partial \hat{\mathcal{D}}}{\partial \dot{\theta}}-\frac{\partial \tilde{p}}{\partial z} \tag{2.28}
\end{align*}
$$

for $i=1,2$ and 3 , respectively. Equation (2.27) implies that $\tilde{p}$ can only be a function of $x$, $z$ and $t$, whilst equation (2.26) integrates to give

$$
\begin{equation*}
\tilde{p}=\left[\frac{\partial}{\partial z}\left(\frac{\partial \hat{\mathcal{D}}}{\partial u^{\prime}}\right)-\rho \dot{u}_{1}\right] x+C(z, t) \tag{2.29}
\end{equation*}
$$

Since $\tilde{p} \nrightarrow \pm \infty$ as $x \rightarrow \pm \infty$, we must have

$$
\begin{equation*}
\frac{\partial}{\partial z}\left(\frac{\partial \hat{\mathcal{D}}}{\partial u^{\prime}}\right)-\rho \dot{u}_{1}=0 \tag{2.30}
\end{equation*}
$$

or

$$
\begin{equation*}
\rho \frac{\partial u}{\partial t}=\frac{\partial}{\partial z}\left[g(\theta) \frac{\partial u}{\partial z}+m(\theta) \frac{\partial \theta}{\partial t}\right], \tag{2.31}
\end{equation*}
$$

which we will refer to as the flow equation. If required, the relative pressure, $\tilde{p}$, may be derived from equation (2.28).

### 2.7 Derivation of the director angle equation

From Ericksen-Leslie theory [27, 42], the balance of angular momentum is given by [42]

$$
\begin{equation*}
\left(\frac{\partial \hat{\omega}_{F}}{\partial \theta_{, i}}\right)_{, i}-\frac{\partial \hat{\omega}_{F}}{\partial \theta}-\frac{\partial \hat{\mathcal{D}}}{\partial \dot{\theta}}=0 \tag{2.32}
\end{equation*}
$$

where $\hat{\mathcal{D}}$ is the reformulated dissipation function given in equation (2.22) and $\omega_{F}$ is the free energy density in the bulk of the cell, given, in our case, by

$$
\begin{equation*}
\omega_{F}=\omega_{\text {elast }}+\omega_{\text {elec }} \tag{2.33}
\end{equation*}
$$

where $\omega_{\text {elast }}$ and $\omega_{\text {elec }}$ denote the elastic and electric energies, respectively.

### 2.7.1 Elastic energy density

The Frank elastic energy for non-chiral nematics is [42]

$$
\begin{align*}
\omega_{\text {elast }}= & \frac{1}{2}\left(K_{1}-K_{2}-K_{4}\right)\left(n_{i, i}\right)^{2}+\frac{1}{2} K_{2} n_{i, j} n_{i, j}+\frac{1}{2} K_{4} n_{i, j} n_{j, i} \\
& +\frac{1}{2}\left(K_{3}-K_{2}\right) n_{j} n_{i, j} n_{k} n_{i, k} \tag{2.34}
\end{align*}
$$

where $K_{1}, K_{2}, K_{3}$ and $\left(K_{2}+K_{4}\right)$ denote the splay, twist, bend and saddle-splay elastic constants, respectively [14]. In one dimension, using the director defined in equation (2.6), the elastic energy in equation (2.34) reduces to

$$
\begin{equation*}
\omega_{\text {elast }}=\frac{1}{2}\left(K_{1} \cos ^{2} \theta+K_{3} \sin ^{2} \theta\right)\left(\frac{\partial \theta}{\partial z}\right)^{2} \tag{2.35}
\end{equation*}
$$

which only contains terms involving the splay and bend elastic constants.

### 2.7.2 Electric energy density

The electric free energy density of a nematic liquid crystal is [42,38]

$$
\begin{align*}
\omega_{\text {elec }} & =-\int \mathrm{D} \cdot d \mathrm{E}  \tag{2.36}\\
& =-\left[\frac{\epsilon_{0}}{2}(\epsilon \cdot \mathrm{E}) \cdot \mathrm{E}+\left(\mathrm{P}_{S} \cdot \mathrm{E}\right)\right] \tag{2.37}
\end{align*}
$$

by equation (2.13). Now substituting equations (2.14), (2.16) and (2.12) into equation (2.37) and rearranging gives the total electric free energy density in the cell,

$$
\begin{equation*}
\omega_{\text {elec }}=-\frac{\epsilon_{0}}{2}\left(\epsilon_{\|} \sin ^{2} \theta+\epsilon_{\perp} \cos ^{2} \theta\right)\left(\frac{\partial U}{\partial z}\right)^{2}+E_{13} \frac{\partial \theta}{\partial z} \sin (2 \theta) \frac{\partial U}{\partial z} . \tag{2.38}
\end{equation*}
$$

### 2.7.3 The director angle equation

Substituting equations (2.35) and (2.38) into equation (2.33) yields the total free energy in the bulk of the cell,

$$
\begin{align*}
\omega_{F}= & \frac{1}{2}\left(K_{1} \cos ^{2} \theta+K_{3} \sin ^{2} \theta\right)\left(\frac{\partial \theta}{\partial z}\right)^{2} \\
& \quad-\frac{\epsilon_{0}}{2}\left(\epsilon_{\|} \sin ^{2} \theta+\epsilon_{\perp} \cos ^{2} \theta\right)\left(\frac{\partial U}{\partial z}\right)^{2}+E_{13} \frac{\partial \theta}{\partial z} \sin (2 \theta) \frac{\partial U}{\partial z} \tag{2.39}
\end{align*}
$$

The final balance of angular momentum is given by substituting equations (2.22) and (2.39) into equation (2.32), simplifying and rearranging to give

$$
\begin{align*}
\gamma_{1} \frac{\partial \theta}{\partial t}= & \left(K_{1} \cos ^{2} \theta+K_{3} \sin ^{2} \theta\right) \frac{\partial^{2} \theta}{\partial z^{2}}+E_{13} \sin (2 \theta) \frac{\partial^{2} U}{\partial z^{2}} \\
& +\frac{1}{2}\left(\epsilon_{0} \Delta \epsilon\left(\frac{\partial U}{\partial z}\right)^{2}+\left(\frac{\partial \theta}{\partial z}\right)^{2}\left(K_{3}-K_{1}\right)\right) \sin (2 \theta)-m(\theta) \frac{\partial u}{\partial z} \tag{2.40}
\end{align*}
$$

which we will refer to as the director equation.

We now have three equations, (2.20), (2.31) and (2.40) for the unknowns $U(z, t), u(z, t)$ and $\theta(z, t)$, respectively. In order to solve these equations fully we must also specify boundary conditions at $z=0$ and $z=d$.

### 2.8 Surface boundary equations

### 2.8.1 Electric potential

If we assume that the lower electrode is earthed and a voltage is applied at the upper electrode, then

$$
\begin{equation*}
U(0, t)=0, \quad \text { and } \quad U(d, t)=V(t), \tag{2.41}
\end{equation*}
$$

where $V(t)$ is the applied voltage at time $t$. Notice that the applied voltage is a function of time, as illustrated in Figure 2.3.


Figure 2.3: An example of the applied voltage varying with time.

### 2.8.2 Flow velocity

For the flow equation, we assume there to be no flow velocity at the monostable solid surface. To model this, we simply apply a no-slip condition at $z=d$, giving

$$
\begin{equation*}
u(d, t)=0 . \tag{2.42}
\end{equation*}
$$

The flow velocity at the bistable surface requires slightly more thought. Note from Figure 2.1 that our simplified surface is taken to be at $z=0$, which occurs slightly above the true bistable surface. Therefore, it is reasonable to assume that there may be some flow at $z=0$.


Figure 2.4: A physical interpretation of the slip-length, $\delta$. With (a) $\delta=0$ we have a no slip, (b) $0<\delta<\infty$ we have partial slip, and (c) $\delta=\infty$ we have perfect slip.

In order to model the flow velocity at this boundary, we consider the three cases illustrated in Figure 2.4 [26]. Let $\delta$, known as the slip-length, denote the distance below $z=0$ where a no-slip boundary condition would be satisfied. By definition, $\delta=0$ corresponds to the no-slip case obscrved in Figure 2.4(a) whilst, at the other cxtreme, $\delta=\infty$ corresponds to the 'perfect slip' case observed in Figure 2.4(c). For $0<\delta<\infty$, as in Figure 2.4(b), we have what is known as a partial slip boundary condition.

From Figure 2.4(b), it becomes clear that a reasonable approximation to the absolute flow velocity at $z=0$ may be computed using the slip-length, $\delta$, and the gradient of the tangential velocity, $\frac{\partial u}{\partial z}$, at $z=0$, using

$$
\begin{equation*}
u(0, t)=\delta \frac{\partial u}{\partial z} . \tag{2.43}
\end{equation*}
$$

This is the same form used by Barrat and Bocquet [4].

### 2.8.3 Director angle

The interaction between a solid surface and liquid crystal molecules is relatively complex. In many cases, it is reasonable to assume that the molecules are rigidly anchored at the surface which fixes $\theta$ at the boundary. This is termed infinite anchoring. However, a more general approach is to specify a 'preferred direction' for the director and an associated energy function to model the increase in energy when the director is not aligned with the preferred direction. This is termed weak anchoring.

At each surface we therefore define a surface energy, $\omega_{\text {surf }}$, which, using variational calculus, allows us to construct the dynamic boundary condition

$$
\begin{equation*}
\nu_{i} \frac{\partial \omega_{F}}{\partial \theta_{, i}}-\frac{\partial \omega_{s u r f}}{\partial \theta}=\frac{\partial \hat{\mathcal{D}}_{s}}{\partial \dot{\theta}}, \tag{2.44}
\end{equation*}
$$

where $\omega_{F}$ is the free energy defined in equation (2.39), $\hat{\mathcal{D}}_{s}$ is the surface dissipation defined as

$$
\begin{equation*}
\hat{\mathcal{D}}_{s}=\frac{1}{2} \gamma_{s}\left(\frac{\partial \theta}{\partial t}\right)^{2} \tag{2.45}
\end{equation*}
$$

and

$$
\nu= \begin{cases}(0,0,1) & \text { at } z=0  \tag{2.46}\\ (0,0,-1) & \text { at } z=d\end{cases}
$$

is the outward unit normal to the boundary surface. The $\gamma_{s}$ parameter is the surface relaxation coefficient which is defined as [9]

$$
\begin{equation*}
\gamma_{s}=l_{s} \gamma_{1}, \tag{2.47}
\end{equation*}
$$

where $l_{s}$ is known as the surface length at the boundary of interest. In physical terms, $l_{s}$ is a measure of the speed at which the director reorientates at a weakly anchored surface relative to the bulk [22].

## Bistable surface energy density

As discussed in Section 2.3, our simplified surface at $z=0$ is governed by a bistable surface energy function, which we will denote $\omega_{s_{0}}$. We wish to construct a simple energy function containing minima at $\theta=\alpha_{1}$ and $\theta=\alpha_{2}$, where $\alpha_{1}$ denotes the preferred HAN orientation at $z=0$ and $\alpha_{2}$ denotes the preferred Vertical orientation at $z=0$. We also wish to control the height of the secondary energy barrier, $\beta$, relative to the primary energy barrier, as illustrated in Figure 2.5.


Figure 2.5: The bistable surface energy function, with minima at $\theta=\left(\alpha_{1}+k \pi\right)$ and $\theta=\left(\alpha_{2}+k \pi\right),(k \in \mathbb{Z})$, a primary energy barrier of height 1.0 and a secondary energy barrier of height $\beta$.

The energy function in Figure 2.5 is derived by taking an appropriate Fourier Cosine Series (FCS) and applying a serics of constraints based on the desired form. A full derivation is
provided in Appendix A, with the final energy function given by

$$
\begin{equation*}
f(\theta)=a_{0}+a_{1} \cos \left(2 \theta-\left(\alpha_{1}+\alpha_{2}\right)\right)+a_{2} \cos \left(4 \theta-2\left(\alpha_{1}+\alpha_{2}\right)\right)+a_{3} \cos \left(6 \theta-3\left(\alpha_{1}+\alpha_{2}\right)\right), \tag{2.48}
\end{equation*}
$$

where

$$
\left.\begin{array}{l}
a_{1}=(\beta(A-B)-(A+B)) / C,  \tag{2.49}\\
a_{2}=2(\beta(D-E)-(D+E)) / C, \\
a_{3}=(\beta(B-F)+(B+F)) / C, \\
a_{0}=1-a_{1}-a_{2}-a_{3},
\end{array}\right\}
$$

and

$$
\begin{align*}
& A=2 \cos ^{2}\left(\alpha_{1}-\alpha_{2}\right)\left(1+2 \sin ^{2}\left(\alpha_{1}-\alpha_{2}\right)\right)-3 \sin ^{2}\left(\alpha_{1}-\alpha_{2}\right) \\
& B=2 \cos \left(\alpha_{1}-\alpha_{2}\right) \\
& C=8 \sin ^{4}\left(\alpha_{1}-\alpha_{2}\right)  \tag{2.50}\\
& D=2 \cos ^{3}\left(\alpha_{1}-\alpha_{2}\right) \\
& E=3 \cos ^{2}\left(\alpha_{1}-\alpha_{2}\right)-1 \\
& F=1+\cos ^{2}\left(\alpha_{1}-\alpha_{2}\right)
\end{align*}
$$

Note that, in order to produce an energy function of the form given in Figure 2.5, we must place specific conditions on $\beta$ and $\left(\alpha_{2}-\alpha_{1}\right)$. This is also covered in Appendix A. The final bistable surface energy density is therefore given by

$$
\begin{equation*}
\omega_{s_{0}}=W_{0} f(\theta), \tag{2.51}
\end{equation*}
$$

where $W_{0}$ is the bistable surface anchoring strength and is a measure of the height of the primary energy barrier between $\alpha_{1}$ and $\alpha_{2}$.

Now, using equations (2.51), (2.44), (2.46), (2.39) and (2.22), we may write the bistable surface equation as

$$
\begin{equation*}
\gamma_{s_{0}} \frac{\partial \theta}{\partial t}=\left(K_{1} \cos ^{2} \theta+K_{3} \sin ^{2} \theta\right) \frac{\partial \theta}{\partial z}+E_{13} \sin (2 \theta) \frac{\partial U}{\partial z}-\frac{\partial \omega_{s_{0}}}{\partial \theta}, \quad z=0, \forall t, \tag{2.52}
\end{equation*}
$$

where $\gamma_{s_{0}}$ is given by equation (2.47) with a surface length of $l_{s_{0}}$.

## Monostable surface energy density

We also employ weak homeotropic anchoring at the monostable surface, which is governed by a monostable surface energy, $\omega_{s_{d}}$. We choose our monostable encrgy to be a simple sinusoidal
function with minima at $(k+1 / 2) \pi(k \in \mathbb{Z})$, as illustrated in Figure 2.6.


Figure 2.6: The monostable surface energy function, with minima at $\theta=(k+1 / 2) \pi(k \in \mathbb{Z})$ and an energy barrier of height 1.0 .

We can therefore immediately write the monostable surface energy as

$$
\begin{equation*}
\omega_{s_{d}}=W_{d} \cos ^{2} \theta \tag{2.53}
\end{equation*}
$$

where $W_{d}$ is the monostable surface anchoring strength and is a measure of the height of the energy barrier between $\theta=\pi / 2$ and $\theta=-\pi / 2$.

Now, using equations (2.53), (2.44), (2.46), (2.39) and (2.22), we may write the monostable surface equation as

$$
\begin{equation*}
\gamma_{s_{d}} \frac{\partial \theta}{\partial t}=-\left(K_{1} \cos ^{2} \theta+K_{3} \sin ^{2} \theta\right) \frac{\partial \theta}{\partial z}-E_{13} \sin (2 \theta) \frac{\partial U}{\partial z}+W_{d} \sin (2 \theta), \quad z=d, \forall t, \tag{2.54}
\end{equation*}
$$

where $\gamma_{s_{d}}$ is given by equation (2.47) with a surface length of $l_{s_{d}}$.

We now have boundary conditions for all three variables $U, u$ and $\theta$ at both surfaces.

### 2.9 Simplifications

At this point, we consider two simplifications of our model that we will use in later chapters.

### 2.9.1 Solving for the field equation

One immediate simplification is that, since we only consider variations in the $z$ direction, we can actually solve the electric potential equation. Note, by equation (2.12), that equation (2.18) rearranges to give

$$
\begin{equation*}
E_{3}=-\frac{\partial U}{\partial z}=\frac{D_{3}-E_{13} \frac{\partial \theta}{\partial z} \sin (2 \theta)}{\epsilon_{0}\left(\epsilon_{\|} \sin ^{2} \theta+\epsilon_{\perp} \cos ^{2} \theta\right)} \tag{2.55}
\end{equation*}
$$

which we will refer to as the field equation. In order to compute $D_{3}$, note that equation (2.19) implies that $D_{3}$ is a constant. Integrating both sides of equation (2.55) gives

$$
\begin{equation*}
-\int_{0}^{d} \frac{\partial U}{\partial z} d z=\int_{0}^{d} \frac{D_{3}-E_{13} \frac{\partial \theta}{\partial z} \sin (2 \theta)}{\epsilon_{0}\left(\epsilon_{\|} \sin ^{2} \theta+\epsilon_{\perp} \cos ^{2} \theta\right)} d z \tag{2.56}
\end{equation*}
$$

Now, assuming that $U(0)=0$ and $U(d)=V$, as stated in equation (2.41), we may rearrange cquation (2.56) to give

$$
\begin{equation*}
D_{3}=\frac{-\epsilon_{0} V+\int_{0}^{d} \frac{E_{13} \frac{3 \theta}{\epsilon_{\|}} \sin (2 \theta)}{\sin ^{2} \theta+\epsilon_{1} \cos ^{2} \theta} d z}{\int_{0}^{d} \frac{1}{\epsilon_{\|} \sin ^{2} \theta+\epsilon_{\perp} \cos ^{2} \theta} d z} . \tag{2.57}
\end{equation*}
$$

This simplification allows us to explicitly solve for the electric field using equations (2.55) and (2.57). To find $\frac{\partial^{2} U}{\partial z^{2}}$, we simply differentiate equation (2.55) to obtain

$$
\begin{equation*}
\frac{\partial^{2} U}{\partial z^{2}}=\frac{\epsilon_{0} \Delta \epsilon \frac{\partial \theta}{\partial z} E_{3} \sin (2 \theta)+E_{13}\left(2 \cos (2 \theta)\left(\frac{\partial \theta}{\partial)^{2}}\right)^{2}+\frac{\partial^{2} \theta}{\partial z^{2}} \sin (2 \theta)\right)}{\epsilon_{0}\left(\epsilon_{\|} \sin ^{2} \theta+\epsilon_{\perp} \cos ^{2} \theta\right)} . \tag{2.58}
\end{equation*}
$$

Equations (2.55) and (2.58) can then be used in equation (2.40) to compute the director angle.

### 2.9.2 Neglecting flow

In some parts of this thesis we will also use the simplification of neglecting all flow effects. This has the effect of replacing the flow equations (2.31), (2.42) and (2.43) with

$$
\begin{equation*}
u(z)=0, \quad(0 \leq z \leq d) . \tag{2.59}
\end{equation*}
$$

Without the presence of flow, the director equation simplifies to give

$$
\begin{align*}
& \gamma_{1} \frac{\partial \theta}{\partial t}=\left(K_{1} \cos ^{2} \theta+K_{3} \sin ^{2} \theta\right) \frac{\partial^{2} \theta}{\partial z^{2}}+E_{13} \sin (2 \theta) \frac{\partial^{2} U}{\partial z^{2}}  \tag{2.60}\\
&+\frac{1}{2}\left(\epsilon_{0} \Delta \epsilon\left(\frac{\partial U}{\partial z}\right)^{2}+\left(\frac{\partial \theta}{\partial z}\right)^{2}\left(K_{3}-K_{1}\right)\right) \sin (2 \theta) .
\end{align*}
$$

Equations (2.59)-(2.60) represent what is known as the no-flow model.

### 2.10 Calculation of the optical transmission

Once the governing equations have been solved it will be useful to calculate the optical transmission through the liquid crystal layer we have modelled. The transmission is a good, experimentally relevant, measure of the director configuration. Let us therefore consider a cell comprising a liquid crystal sandwiched between two crossed polarisers. As light passes through the first polariser, it becomes linearly polarised. If the liquid crystal molecules are in a Vertical state, the light remains linearly polarised and no light is able to pass through the second polariser, which is typically termed the analyser, giving zero optical transmission, as in Figure 2.7(a). If the liquid crystal molecules are in a HAN state, however, the director tilt will change the linearly polarised light into elliptically polarised light, thus allowing a proportion of light through the second polariser and giving some optical transmission, as in Figure 2.7(b). Therefore, the Vertical and HAN states are sometimes referred to as the 'dark' and 'light' states, respectively.

At any given time, the optical transmission, $T$, through a liquid crystal cell depends on the


Figure 2.7: Illustration of a liquid crystal sandwiched between two crossed polarisers, where the second polariser is known as the analyser. Unpolarised light passes through the first polariser and (a) the vertically aligned liquid crystal molecules cause the light to become linearly polarised and so it cannot pass through the analyser, (b) the liquid crystal molecules are in a HAN state and guide the light through the analyser.
director orientation throughout the cell and can be calculated using [9]

$$
\begin{equation*}
T=\sin ^{2}\left(\frac{R}{2}\right) \tag{2.61}
\end{equation*}
$$

where the optical retardation, $R$, is given by

$$
\begin{equation*}
R=\frac{2 \pi n_{o}}{\lambda} \int_{0}^{d}\left(\frac{1}{\sqrt{\sin ^{2} \theta(z)+\frac{n_{o}^{2}}{n_{e}^{2}} \cos ^{2} \theta(z)}}-1\right) d z \tag{2.62}
\end{equation*}
$$

where $\lambda$ is the wavelength of the light entering the cell, $n_{o}$ is the ordinary refractive index and $n_{e}$ is the extraordinary refractive index of the liquid crystal, as illustrated in Figure 2.8.


Figure 2.8: Illustration of the ordinary refractive index, $n_{o}$, and the extraordinary refractive index, $n_{e}$.

### 2.11 Base parameter set

Having derived the governing equations for a bistable device, we require a parameter set in order to describe and model a particular cell. We therefore present a standard base parameter set in Table 2.1, which will be used throughout this thesis.

The parameters stated in Table 2.1 have been adapted from a variety of sources, such as [42] and data provided by Hewlett Packard Laboratories in Bristol. As such, they describe a fictitious liquid crystal cell and are intended for example purposes only.

### 2.12 Summary

In this chapter, we have set up the systems of equations that will be used throughout this investigation. Note that we essentially have two models. The first model includes the flow equations and is governed by equations $(2.31),(2.40),(2.41),(2.42),(2.43),(2.52),(2.54)$, (2.55), (2.57) and (2.58). The second (no-flow) model is governed by equations (2.59), (2.60), $(2.41),(2.52),(2.54),(2.55),(2.57)$ and (2.58). In the next chapter, we will introduce the numerical methods that are to be used for solving our systems of equations.

Table 2.1: Default values for the parameters that are used in simulations.

| Quantity | Symbol | Value | Units |
| :--- | :---: | :---: | :---: |
| Cell thickness | $d$ | 3.0 | $10^{-6} \mathrm{~m}$ |
| Parallel permittivity | $\epsilon_{\\|}$ | 10.0 |  |
| Perpendicular permittivity | $\epsilon_{\perp}$ | 15.0 |  |
| Flexoclectric Cocfficient | $E_{13}$ | 1.0 | $10^{-10} \mathrm{Cm}^{-1}$ |
| Splay elastic constant | $K_{1}$ | 16.3 | $10^{-12} \mathrm{~N}$ |
| Bend elastic constant | $K_{3}$ | 16.3 | $10^{-12} \mathrm{~N}$ |
| Leslie viscosity | $\tilde{\alpha}_{1}$ | -0.6 | $10^{-2} \mathrm{Nsm}^{-2}$ |
| Leslie viscosity | $\tilde{\alpha}_{2}$ | -8.12 | $10^{-2} \mathrm{Nsm}^{-2}$ |
| Leslie viscosity | $\tilde{\alpha}_{3}$ | -0.36 | $10^{-2} \mathrm{Nsm}^{-2}$ |
| Leslie viscosity | $\tilde{\alpha}_{4}$ | 6.52 | $10^{-2} \mathrm{Nsm}^{-2}$ |
| Leslie viscosity | $\tilde{\alpha}_{5}$ | 6.40 | $10^{-2} \mathrm{Nsm}^{-2}$ |
| Leslie viscosity | $\tilde{\alpha}_{6}$ | -2.08 | $10^{-2} \mathrm{Nsm}^{-2}$ |
| Bistable surface length | $l_{s_{0}}$ | 5.0 | $10^{-7} \mathrm{~m}$ |
| Monostable surface length | $l_{s_{d}}$ | 2.5 | $10^{-6} \mathrm{~m}$ |
| Partial slip-length | $\delta$ | 1.0 | $10^{-10} \mathrm{~m}$ |
| Fluid density | $\rho$ | 1020.0 | $\mathrm{~kg} \mathrm{~m}^{-3}$ |
| Bistable anchoring strength | $W_{0}$ | 4.0 | $10^{-5} \mathrm{Nm}^{-1}$ |
| Monostable anchoring strength | $W_{d}$ | 4.2 | $10^{-3} \mathrm{Nm}^{-1}$ |
| $\theta(z=0)$ for HAN state | $\alpha_{1}$ | 10.0 | degrees |
| $\theta(z=0)$ for Vertical state | $\alpha_{2}$ | 80.0 | $d$ degrees |
| Ordinary refractive index | $n_{o}$ | 1.4920 |  |
| Extraordinary refractive index | $n_{e}$ | 1.6567 |  |
| Wavelength of light | $\lambda$ | 632.8 | $10^{-9} \mathrm{~m}$ |

## 3 Numerical methods

In this section, we will introduce the numerical methods that are to be used for solving the systems of equations derived in the previous section (Section 2). We begin by defining the spatial discretisation, which will be used throughout the remainder of the investigation. We then introduce three time integration methods, namely an explicit method, a semi-implicit method and a fully-implicit method.

Having proposed our methods of solution, we then present a variety of techniques for optimising our simulations. The first, time adaption, is used to control the size of each time-step such that the solution error remains roughly constant throughout the simulation. The second, a moving mesh algorithm, is used to control the positioning and quantity of node points used at each time-step such that the spatial error remains roughly constant throughout the simulation.

Finally, we compare each of these simulation methods to determine which provides the optimal balance of speed and accuracy.

### 3.1 Spatial discretisation

Our equations are spatially discretised so that they may be solved on a non-uniform mesh. In the bulk equations, central differences are used to compute any derivatives. Let us consider a function, $f(z)$, where $h_{j}=\left(z_{j+1}-z_{j}\right)$. Writing the Taylor series expansion for $f\left(z_{j-1}\right)$, we have

$$
\begin{equation*}
f\left(z_{j-1}\right)=f\left(z_{j}\right)-h_{j-1} \frac{\partial f\left(z_{j}\right)}{\partial z}+\frac{h_{j-1}^{2}}{2!} \frac{\partial^{2} f\left(z_{j}\right)}{\partial z^{2}}-\frac{h_{j-1}^{3}}{3!} \frac{\partial^{3} f\left(z_{j}\right)}{\partial z^{3}}+\ldots \tag{3.1}
\end{equation*}
$$

whilst the corresponding expansion for $f\left(z_{j+1}\right)$ is

$$
\begin{equation*}
f\left(z_{j+1}\right)=f\left(z_{j}\right)+h_{j} \frac{\partial f\left(z_{j}\right)}{\partial z}+\frac{h_{j}^{2}}{2!} \frac{\partial^{2} f\left(z_{j}\right)}{\partial z^{2}}+\frac{h_{j}^{3}}{3!} \frac{\partial^{3} f\left(z_{j}\right)}{\partial z^{3}}+\ldots \tag{3.2}
\end{equation*}
$$

To obtain an approximation of the first derivative, we subtract (3.1) from (3.2) and rearrange to give

$$
\begin{equation*}
\frac{\partial f\left(z_{j}\right)}{\partial z}=\frac{f\left(z_{j+1}\right)-f\left(z_{j-1}\right)}{h_{j-1}+h_{j}}+O\left(h_{j}-h_{j-1}\right) \tag{3.3}
\end{equation*}
$$

Similarly, the second derivative is approximated by adding (3.1) $\times h_{j}$ to (3.2) $\times h_{j-1}$ and rearranging, viz.

$$
\begin{equation*}
\frac{\partial^{2} f\left(z_{j}\right)}{\partial z^{2}}=\frac{2\left(h_{j-1} f\left(z_{j+1}\right)-\left(h_{j-1}+h_{j}\right) f\left(z_{j}\right)+h_{j} f\left(z_{j-1}\right)\right)}{h_{j-1} h_{j}\left(h_{j-1}+h_{j}\right)}+O\left(h_{j}-h_{j-1}\right) \tag{3.4}
\end{equation*}
$$

Due to the complicated interactions between the surface and bulk equations, and since we expect high gradients to occur at each surface, the first derivatives at $z=0\left(=z_{1}\right)$ and $z=d$ ( $=z_{m+1}$ ) are approximated to second order. Consider

$$
\begin{equation*}
f\left(z_{2}\right)=f\left(z_{1}\right)+h_{1} \frac{\partial f\left(z_{1}\right)}{\partial z}+\frac{h_{1}^{2}}{2!} \frac{\partial^{2} f\left(z_{1}\right)}{\partial z^{2}}+\frac{h_{1}^{3}}{3!} \frac{\partial^{3} f\left(z_{1}\right)}{\partial z^{3}}+\ldots \tag{3.5}
\end{equation*}
$$

and

$$
\begin{equation*}
f\left(z_{3}\right)=f\left(z_{1}\right)+\left(h_{1}+h_{2}\right) \frac{\partial f\left(z_{1}\right)}{\partial z}+\frac{\left(h_{1}+h_{2}\right)^{2}}{2!} \frac{\partial^{2} f\left(z_{1}\right)}{\partial z^{2}}+\frac{\left(h_{1}+h_{2}\right)^{3}}{3!} \frac{\partial^{3} f\left(z_{1}\right)}{\partial z^{3}}+\ldots \tag{3.6}
\end{equation*}
$$

Now, $\left(h_{1}+h_{2}\right)^{2} \times(3.5)$ subtracted from $h_{1}^{2} \times(3.6)$ gives

$$
\begin{align*}
h_{1}^{2} f\left(z_{3}\right)-\left(h_{1}+h_{2}\right)^{2} f\left(z_{2}\right)= & {\left[h_{1}^{2}-\left(h_{1}+h_{2}\right)^{2}\right] f\left(z_{1}\right)-\left(h_{1}+h_{2}\right) h_{1} h_{2} \frac{\partial f\left(z_{1}\right)}{\partial z} } \\
& +\frac{\left(h_{1}+h_{2}\right)^{2} h_{1}^{2} h_{2}}{3!} \frac{\partial^{3} f\left(z_{1}\right)}{\partial z^{3}}+\ldots, \tag{3.7}
\end{align*}
$$

which may be rearranged to give

$$
\begin{align*}
\left.\frac{\partial f\left(z_{1}\right)}{\partial z}=\frac{1}{h_{1} h_{2}\left(h_{1}+\right.}+h_{2}\right) & \left\{\left[h_{1}^{2}-\left(h_{1}+h_{2}\right)^{2}\right] f\left(z_{1}\right)\right. \\
& \left.+\left(h_{1}+h_{2}\right)^{2} f\left(z_{2}\right)-h_{1}^{2} f\left(z_{3}\right)\right\}+O\left(h_{1}\left(h_{1}+h_{2}\right)\right) \tag{3.8}
\end{align*}
$$

Using the same approach, the first derivative at $z=d\left(=z_{m+1}\right)$ is determined using

$$
\begin{align*}
& \frac{\partial f\left(z_{m+1}\right)}{\partial z}=\frac{1}{h_{m} h_{m-1}\left(h_{m}+h_{m-1}\right)}\left\{\left[\left(h_{m}+h_{m-1}\right)^{2}-h_{m}^{2}\right] f\left(z_{m+1}\right)\right. \\
&\left.-\left(h_{m}+h_{m-1}\right)^{2} f\left(z_{m}\right)+h_{m}^{2} f\left(z_{m-1}\right)\right\}+O\left(h_{m}\left(h_{m}+h_{m-1}\right)\right) \tag{3.9}
\end{align*}
$$

### 3.2 Time integration

We now present several methods for solving coupled systems of the form

$$
\left.\left.\begin{array}{l}
\frac{\partial \theta}{\partial t}=f(\theta(z, t), u(z, t), t)  \tag{3.10}\\
\frac{\partial u}{\partial t}=g(\theta(z, t), u(z, t), t)
\end{array}\right\} \quad \begin{array}{l}
\quad
\end{array}\right\}<z<d
$$

where $\left(\theta^{(n)}, u^{(n)}\right)$ is used to denote the solution of $(\theta, u)$ at the $n$th timestep and $t_{n}$ denotes the time at the $n$th timestep. The $n$th step-size is denoted by $\tau_{n}=t_{n+1}-t_{n}$.

### 3.2.1 Explicit method

The Forward Euler method is used as an example of explicit time discretisation. Forming a Taylor series expansion in time of $\theta^{(n+1)}$ gives

$$
\begin{align*}
\theta^{(n+1)} & =\theta^{(n)}+\tau_{n} \frac{\partial \theta^{(n)}}{\partial t}+O\left(\tau_{n}^{2}\right) \\
& \approx \theta^{(n)}+\tau_{n} f^{(n)} \tag{3.11}
\end{align*}
$$

by equation (3.10). Similarly, for $u^{(n+1)}$ we have

$$
\begin{equation*}
u^{(n+1)} \approx u^{(n)}+\tau_{n} g^{(n)} \tag{3.12}
\end{equation*}
$$

Equations (3.11) and (3.12) represent the explicit Forward Euler method as applied to equations (3.10). At each timestep, $f^{(n)}$ and $g^{(n)}$ are computed using $\theta^{(n)}$ and $u^{(n)}$ and substituted into equations (3.11) and (3.12), respectively, together with $\tau_{n}$ to compute $\theta^{(n+1)}$ and $u^{(n+1)}$. Notice that in the absence of flow (i.e. $u \equiv 0$ ) the system simplifies and we need only use equation (3.11) at each timestep.

As with all explicit methods, a relatively small step-size is required in order to maintain numerical stability.

### 3.2.2 Semi-implicit method

In an effort to allow for larger timesteps and enhanced stability, we attempt to solve our system of equations using a semi-implicit method. Let us rewrite equations (3.10) as

$$
\left.\begin{array}{rl}
\frac{\partial \theta}{\partial t} & =f_{L}(\theta(z, t), u(z, t), t)+f_{N}(\theta(z, t), u(z, t), t)  \tag{3.13}\\
\frac{\partial u}{\partial t} & =g_{L}(\theta(z, t), u(z, t), t)+g_{N}(\theta(z, t), u(z, t), t)
\end{array}\right\} \quad 0<z<d
$$

where $f_{L}$ represents the terms in $f$ which are linear in $\theta$ and $f_{N}$ represents the terms in $f$ which are nonlinear in $\theta$. Similarly, $g_{L}$ represents the terms in $g$ which are linear in $u$ and $g_{N}$ represents the terms in $g$ which are nonlinear in $u$.

We now proceed by decoupling $\theta$ and $u$ and then solving each set of equations separately. Rearranging the $\theta$ equation in (3.13) whilst evaluating $f_{L}$ implicitly in $\theta$ gives

$$
\begin{equation*}
\theta^{(n+1)}-\tau_{n} f_{L}\left(\theta^{(n+1)}, u^{(n)}\right)=\theta^{(n)}+\tau_{n} f_{N}\left(\theta^{(n)} u^{(n)}\right) \tag{3.14}
\end{equation*}
$$

Taking into account our spatial discretisation, equation (3.14) may we written in matrix form as

$$
\begin{equation*}
A \theta^{(n+1)}=\mathrm{d} \tag{3.15}
\end{equation*}
$$

where

$$
A=\left[\begin{array}{cccccc}
b_{1} & c_{1} & \sigma_{1} & & &  \tag{3.16}\\
a_{2} & b_{2} & c_{2} & & & \\
& a_{3} & b_{3} & c_{3} & & \\
& & \ddots & \ddots & \ddots & \\
& & & a_{m} & b_{m} & c_{m} \\
& & & \sigma_{2} & a_{m+1} & b_{m+1}
\end{array}\right]
$$

is a near-tridiagonal matrix comprising the vectors $\mathbf{a}, \mathbf{b}, \mathbf{c}$ and the scalar quantities $\sigma_{1}$ and $\sigma_{2}$. Note that the terms $\sigma_{1}$ and $\sigma_{2}$ occur due to the second-order accuracy of the derivatives at the boundaries. By applying two elementary row operations we can reduce the problem to a purely tridiagonal system which may be efficiently solved using Algorithm 1, thus producing $\theta^{(n+1)}$.

```
Algorithm 1 The Thomas algorithm for solving the tridiagonal system \(A \mathbf{x}=\mathrm{d}\), where \(A\)
is an \((m+1) \times(m+1)\) tridiagonal matrix comprising the vectors \(\mathbf{a}, \mathbf{b}\) and \(\mathbf{c}\).
Require: \(m, a, b, c, d\)
    compute: \(\zeta_{1}=c_{1} / b_{1}\)
    compute: \(\delta_{1}=d_{1} / b_{1}\)
    for \(i=2, \ldots,(m+1)\) do
        compute: \(\xi=b_{i}-a_{i} \zeta_{i-1}\)
        compute: \(\zeta_{i}=c_{i} / \xi\)
        compute: \(\delta_{i}=\left(d_{i}-a_{i} \delta_{i-1}\right) / \xi / /\) forward climination
    end for
    compute: \(x_{m+1}=\delta_{m+1}\)
    for \(i=m, 1,-1\) do
        compute: \(x_{i}=\delta_{i}-\zeta_{i} x_{i+1} / /\) back substitution
    end for
```

Using the updated value of $\theta$ we may rearrange the $u$ equation in (3.13) to give

$$
\begin{equation*}
u^{(n+1)}-\tau_{n} g_{L}\left(\theta^{(n+1)}, u^{(n+1)}\right)=u^{(n)}+\tau_{n} f_{N}\left(\theta^{(n+1)} u^{(n)}\right) \tag{3.17}
\end{equation*}
$$

which also represents a tridiagonal system and may be solved for $u^{(n+1)}$ using the Thomas algorithm.

### 3.2.3 Fully-Implicit method

Although our semi-implicit method provides a substantial improvement over the explicit Forward Euler method, there are some instances when it is insufficient for accurately modelling flow. This is likely to be due to the decoupling of the director and flow equations, together with the inherent instabilities associated with the explicit part of the method. To combat this, we present a fully-implicit Newton-Raphson method for solving a system of equations.

Evaluating equations (3.10) implicitly and rearranging gives

$$
\left.\begin{array}{l}
\theta^{(n+1)}-\theta^{(n)}-\tau_{n} f\left(\theta^{(n+1)}, u^{(n+1)}\right)=0  \tag{3.18}\\
u^{(n+1)}-u^{(n)}-\tau_{n} g\left(\theta^{(n+1)}, u^{(n+1)}\right)=0
\end{array}\right\}
$$

which we will write in matrix form as

$$
\begin{equation*}
\mathbf{F}(\mathrm{x})=0 \tag{3.19}
\end{equation*}
$$

where

$$
\begin{equation*}
\mathbf{x}=\left\{\theta_{j}^{(n+1)}, u_{j}^{(n+1)}\right\}, \quad j=1, \ldots,(m+1) \tag{3.20}
\end{equation*}
$$

Let $\Delta \mathbf{x}$ be a vector of length $2 m+2$. Writing the Taylor series expansion of $\mathbf{F}(\mathbf{x}+\Delta \mathbf{x})$ gives

$$
\begin{equation*}
\mathbf{F}(\mathbf{x}+\Delta \mathbf{x})=\mathbf{F}(\mathbf{x})+J(\mathbf{x}) \Delta \mathbf{x}+O\left(\Delta \mathbf{x}^{2}\right) \tag{3.21}
\end{equation*}
$$

where $J(\mathbf{x})$ is the Jacobian matrix associated with $\mathbf{F}(\mathbf{x})$, whose elements are defined as

$$
\begin{equation*}
J_{j k}(\mathrm{x})=\frac{\partial F_{j}}{\partial x_{k}} \tag{3.22}
\end{equation*}
$$

Let us set $\mathbf{x}=\hat{\mathbf{x}}_{i}$ and $\Delta \mathbf{x}=\left(\hat{\mathbf{x}}_{i+1}-\hat{\mathbf{x}}_{i}\right)$, where $i$ denotes the internal Newton-Raphson step number. Neglecting the term $O\left(\Delta \mathrm{x}^{2}\right)$, equation (3.21) gives

$$
\begin{equation*}
\mathbf{F}\left(\hat{\mathbf{x}}_{i+1}\right)=\mathbf{F}\left(\hat{\mathbf{x}}_{i}\right)+J\left(\hat{\mathbf{x}}_{i}\right)\left(\hat{\mathbf{x}}_{i+1}-\hat{\mathbf{x}}_{i}\right) \tag{3.23}
\end{equation*}
$$

At the solution, $F\left(\hat{\mathbf{x}}_{i+1}\right)=0$ and we have

$$
\mathrm{F}\left(\hat{\mathbf{x}}_{i}\right)+J\left(\hat{\mathbf{x}}_{i}\right)\left(\hat{\mathbf{x}}_{i+1}-\hat{\mathbf{x}}_{i}\right)=0
$$

which rearranges to give

$$
\begin{equation*}
J\left(\hat{\mathbf{x}}_{i}\right)\left(\hat{\mathbf{x}}_{i+1}-\hat{\mathbf{x}}_{i}\right)=-\mathbf{F}\left(\hat{\mathbf{x}}_{i}\right) . \tag{3.24}
\end{equation*}
$$

Further rearranging gives

$$
\begin{equation*}
\hat{\mathbf{x}}_{i+1}=\hat{\mathbf{x}}_{i}-J^{-1}\left(\hat{\mathbf{x}}_{i}\right) \mathrm{F}\left(\hat{\mathrm{x}}_{i}\right), \tag{3.25}
\end{equation*}
$$

which is known as the Newton-Raphson method for linear systems. Note that $J^{-1}\left(\hat{\mathbf{x}}_{i}\right)$ is typically very expensive to compute, so we solve equation (3.24) for ( $\hat{\mathbf{x}}_{i+1}-\hat{\mathbf{x}}_{i}$ ) and then compute $\hat{\mathbf{x}}_{i+1}$ from the resulting vector. This process is iterated using

$$
\begin{equation*}
\left.\mathbf{x}\right|_{\left(t=t_{n}\right)}=\hat{\mathbf{x}}_{0} \longrightarrow \hat{\mathbf{x}}_{1} \longrightarrow \hat{\mathbf{x}}_{2} \longrightarrow \ldots \longrightarrow \hat{\mathbf{x}}_{q}=\left.\mathbf{x}\right|_{\left(t=t_{n+1}\right)}, \tag{3.26}
\end{equation*}
$$

until the $q$ th iteration which occurs when the vector $\mathbf{F}\left(\hat{\mathbf{x}}_{i}\right)$ is deemed sufficiently close to zero, at which point we conclude that the system has converged. Note that our initial guess, $\hat{\mathbf{x}}_{0}$, is taken as our final approximation of the solution at the previous timestep. Once the
system has converged to within our prescribed tolerance, we move on to the next timestep. Should the system diverge at any point, we conclude that the step-size was too large and restart the routine using a smaller step-size.

```
Algorithm 2 Newton-Raphson method for a Linear System.
Require: TOL, \(\hat{\mathbf{x}}_{0}\)
Require: \(i=0, F_{\text {norm }}=1.0 \times 10^{16}\)
    \(F_{\text {norm }}^{\text {OLD }}=F_{\text {norm }}\)
    compute: \(\mathrm{F}\left(\hat{\mathrm{x}}_{i}\right)\)
    compute: \(F_{\text {norm }}=\left|\mathrm{F}\left(\hat{\mathrm{x}}_{i}\right)\right|_{\infty}\)
    if \(F_{\text {norm }}<\) TOL then
        GO то 15 // the method has converyed
    else if \(F_{\text {norm }} \geqslant F_{\text {norm }}^{\text {OLD }}\) then
        flag \(=1\)
        GO тO 15 // the method is diverging
    end if
    compute: \(J\left(\hat{\mathbf{x}}_{i}\right) / /\) calculated analytically
    solve: \(J\left(\hat{x}_{i}\right) \Delta \hat{x}_{i}=-F\left(\hat{x}_{i}\right)\) for \(\Delta \hat{x}_{i}\)
    compute: \(\hat{\mathbf{x}}_{i+1}=\hat{\mathbf{x}}_{i}+\Delta \hat{\mathbf{x}}_{i}\)
    \(i=i+1\)
    GO TO 1
    STOP
```

The complete method is outlined in Algorithm 2, and it is applied at each timestep. The relative infinity norm,

$$
\begin{equation*}
\left|\mathbf{F}\left(\hat{\mathbf{x}}_{i}\right)\right|_{\infty}=\max _{j} \bar{\lambda}_{j}\left|\mathbf{F}\left(\hat{\mathbf{x}}_{i}(j)\right)\right|, \tag{3.27}
\end{equation*}
$$

where $\bar{\lambda}_{j}$ is a nondimensionalisation parameter, is used to measure the convergence of the system, as this is relatively cheap to compute. TOL is a uscr-prescribed tolerance which we typically set to $1.0 \times 10^{-8}$. Note that we test for convergence and divergence prior to our analytical computation of $J\left(\hat{\mathbf{x}}_{\boldsymbol{i}}\right)$ since this prevents us from wasting CPU cycles on the final itcration at any given timestep.

One of the key advantages of the Newton-Raphson method is that, given a suitable initial approximation, its order of convergence is quadratic, making it one of the fastest iterative methods for linear systems. Also, being an implicit method, it allows larger step-sizes to be taken when compared with the explicit and semi-implicit methods. However, the method is typically very expensive since we need to evaluate a Jacobian matrix at each iteration.

When applied to equations (3.10), equation (3.24) becomes

$$
\begin{equation*}
J\left(\mathbf{X}_{i}\right) \mathbf{X}_{i}=-\Gamma_{i} \tag{3.28}
\end{equation*}
$$

where

$$
\left.\begin{array}{rl}
\mathbf{X}_{\mathbf{i}}(2 j-1) & =\Delta \boldsymbol{\theta}_{\mathbf{i}}(j),  \tag{3.29}\\
\mathbf{X}_{\mathbf{i}}(2 j) & =\Delta \mathbf{u}_{i}(j),
\end{array}\right\}, \quad j=1, \ldots,(m+1)
$$

and

$$
\left.\begin{array}{rl}
\Gamma_{i}(2 j-1) & =\frac{\theta_{i}(j)-\theta_{0}(j)}{\tau}-f_{i}(j)  \tag{3.30}\\
\Gamma_{i}(2 j) & =\frac{\mathbf{u}_{i}(j)-\mathbf{u}_{0}(j)}{\tau}-\mathrm{g}_{i}(j)
\end{array}\right\}, \quad j=1, \ldots,(m+1)
$$

Notice that here we have interlaced $\boldsymbol{\theta}$ and $\mathbf{u}$, as this reduces the bandwidth of the associated Jacobian matrix, $J\left(\mathbf{X}_{\boldsymbol{i}}\right)$. This is essential, since solving a block matrix comprising four sparse blocks is computationally very expensive, and causes an unnecessary bottleneck in line 11 of Algorithm 2. Now, given the spatial discretisation of the bulk and surface equations, notice that in this particular case the Jacobian matrix defined by equation (3.22) only contains non-zero entrics on

$$
\left.\begin{array}{rl}
J_{1, k}\left(\mathbf{X}_{i}\right) & : k=1,3,5 \\
J_{2, k}\left(\mathbf{X}_{i}\right) & : k=2,4,6 \\
J_{(2 j-1), k}\left(\mathbf{X}_{i}\right) & : k=(2 j-3), \ldots,(2 j+2), \\
J_{(2 j), k}\left(\mathbf{X}_{i}\right) & : \\
J_{(2 m+1), k}\left(\mathbf{X}_{i}\right) & : k=(2 j-3), \ldots,(2 j+2), \\
J_{(2 m+2), k}\left(\mathbf{X}_{i}\right) & : k=2 m-3),(2 m-1),(2 m+1),
\end{array}\right\} \quad j=2, \ldots, m,
$$

and so we have a pseudo-hexadiagonal system with additional non-zero elements at each boundary. To aid in visualising the structure of this system, we present the form of $J$ for
the simple case in which $m=0$, viz.

where "." is used to denote a non-zero element. Now, note that upon performing a sequence of elementary row operations, the system in equation (3.28) can be reduced to pentadiagonal form giving

$$
\begin{equation*}
\hat{J}\left(\mathbf{X}_{i}\right) \hat{\mathbf{X}}_{i}=-\hat{\Gamma}_{i}, \tag{3.33}
\end{equation*}
$$

where the reformulated Jacobian

$$
\hat{J}=\left[\begin{array}{cccccccc}
c_{1} & d_{1} & e_{1} & & & & &  \tag{3.34}\\
b_{2} & c_{2} & d_{2} & e_{2} & & & & \\
a_{3} & b_{3} & c_{3} & d_{3} & e_{3} & & & \\
& & \ddots & \ddots & \ddots & \ddots & & \\
& & & a_{2 m} & b_{2 m} & c_{2 m} & d_{2 m} & e_{2 m} \\
& & & & a_{2 m+1} & b_{2 m+1} & c_{2 m+1} & d_{2 m+1} \\
& & & & & a_{2 m+2} & b_{2 m+2} & c_{2 m+2}
\end{array}\right],
$$

is purely pentadiagonal. The system in (3.33) may now be solved using the highly efficient routine in Algorithm 3 [29].

Note that in the no-flow case (i.e. $\mathbf{u} \equiv 0$ ) equation (3.28) reduces to

$$
\begin{equation*}
J\left(\Delta \boldsymbol{\theta}_{\boldsymbol{i}}\right) \Delta \boldsymbol{\theta}_{\boldsymbol{i}}=-\hat{\mathbf{f}}_{\mathbf{i}}, \tag{3.35}
\end{equation*}
$$

where

$$
\begin{equation*}
\hat{\mathrm{f}}_{i}=\frac{\theta_{i}-\theta_{0}}{\tau}-\mathrm{f}_{i} \tag{3.36}
\end{equation*}
$$

which, upon two elementary row operations, is a purely tridiagonal system that may be cheaply solved using the Thomas algorithm (Algorithm 1).

```
Algorithm 3 Algorithm for solving the pentadiagonal system \(A \mathbf{x}=\mathbf{f}\), where \(A\) is an \(m \times m\)
pentadiagonal matrix comprising the vectors \(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\) and \(\mathbf{e}\). Note that this is a slightly
modified version of [29].
Require: \(m, \mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}, \mathbf{e}, \mathbf{f}\)
    compute: \(p_{1}=-d_{1} / c_{1}\)
    compute: \(q_{1}=-e_{1} / c_{1}\)
    compute: \(\delta_{1}=f_{1} / c_{1}\)
    compute: \(\zeta=-1 /\left(c_{2}+b_{2} p_{1}\right)\)
    compute: \(p_{2}=\left(d_{2}+b_{2} q_{1}\right) \zeta\)
    compute: \(q_{2}=e_{2} \zeta\)
    compute: \(\delta_{2}=\left(b_{2} \delta_{1}-f_{2}\right) \zeta\)
    for \(i=3, \ldots, m\) do
        compute: \(\zeta=b_{i}+a_{i} p_{i-2}\)
        compute: \(\xi=-1 /\left(c_{i}+a_{i} q_{i-2}+\zeta p_{i-1}\right)\)
        compute: \(p_{i}=\left(d_{i}+\zeta q_{i-1}\right) \xi\)
        compute: \(q_{i}=e_{i} \xi\)
        compute: \(\delta_{i}=\left(a_{i} \delta_{i-2}+\zeta \delta_{i-1}-f_{i}\right) \xi / /\) forward elimination
    end for
    compute: \(x_{m}=\delta_{m}\)
    compute: \(x_{m-1}=\delta_{m-1}+p_{m-1} \delta_{m}\)
    for \(i=(m-2), 1,-1\) do
        compute: \(x_{i}=\delta_{i}+p_{i} x_{i+1}+q_{i} x_{i+2} / /\) back substitution
    end for
```


### 3.3 Field calculation

As mentioned in Section 2.9.1, the electric field equation is given by

$$
\begin{equation*}
E(z)=\frac{D_{3}-E_{13} \sin (2 \theta) \frac{\partial \theta}{\partial z}}{\epsilon_{0}\left(\epsilon_{\perp}+\Delta \epsilon \sin ^{2} \theta\right)} \tag{3.37}
\end{equation*}
$$

where

$$
\begin{equation*}
D_{3}=\frac{-\epsilon_{0} V+E_{13} \int_{0}^{d} \frac{\sin (2 \theta) \frac{\theta \theta}{\epsilon_{1}+\Delta \epsilon \sin ^{2} \theta}}{\partial^{2} \theta} z}{\int_{0}^{d} \frac{1}{\epsilon_{1}+\Delta \epsilon \sin ^{2} \theta} d z} \tag{3.38}
\end{equation*}
$$

Note that $E(z)$ and $\frac{\partial E}{\partial z}$ are needed in $f(\theta, u)$ and are therefore calculated at the start of each timestep. In order to solve the integrals in equation (3.38), we employ a numerical method.

For our purposes, the trapezoid rule on a non-uniform grid is sufficient. Let us define

$$
\begin{align*}
& I_{1}=\int_{0}^{d} \frac{1}{\epsilon_{\perp}+\Delta \epsilon \sin ^{2} \theta} d z,  \tag{3.39}\\
& I_{2}=\int_{0}^{d} \frac{\sin (2 \theta) \frac{\partial \theta}{\partial z}}{\epsilon_{\perp}+\Delta \epsilon \sin ^{2} \theta} d z, \tag{3.40}
\end{align*}
$$

which may be discretised and approximated by

$$
\begin{align*}
& I_{1} \approx \frac{d}{2} \sum_{j=1}^{m} h_{j}\left[\frac{1}{\epsilon_{\perp}+\Delta \epsilon \sin ^{2}\left(\theta_{j}\right)}+\frac{1}{\epsilon_{\perp}+\Delta \epsilon \sin ^{2}\left(\theta_{j+1}\right)}\right]  \tag{3.41}\\
& I_{2} \approx \frac{1}{2} \sum_{j=1}^{m} h_{j}\left[\frac{\sin \left(2 \theta_{j}\right) \frac{\partial\left(\theta_{j}\right)}{\partial z}}{\epsilon_{\perp}+\Delta \epsilon \sin ^{2}\left(\theta_{j}\right)}+\frac{\sin \left(2 \theta_{j+1}\right) \frac{\partial\left(\theta_{j+1}\right)}{\partial z}}{\epsilon_{\perp}+\Delta \epsilon \sin ^{2}\left(\theta_{j+1}\right)}\right], \tag{3.42}
\end{align*}
$$

allowing us to compute the constant term $D_{3}$ using equation (3.38) and hence compute the electric field at each node point using equation (3.37).

It should be noted that there are more advanced integration methods, such as Simpson's rule and Gaussian quadrature. However, since our moving mesh algorithm (discussed in Section 3.5) focuses node points about areas with steep gradients, the trapezoid rule should give a sufficient level of accuracy.

### 3.4 Time adaption

Initial attempts to solve our system of equations used a timestep size of $\Delta t$, which remained constant throughout each simulation. While perfectly acceptable for no-flow simulations involving relatively small voltages, problems can occur when we wish to apply higher voltages. This is because higher voltages cause the transition between relaxed and stressed states to happen in a much shorter period of time. Since the size of the time step remains constant, we are forced to choose a $\Delta t$ which is small enough to accommodate the fastest-changing part of the simulation (i.e. the initial switch-on and/or switch-off). The problem with this is that much computation time is wasted during relatively static portions of the simulation. In the case of very high voltages, it becomes impractical to simulate in this fashion since the required $\Delta t$ becomes incredibly small.

To combat this problem, we have employed an adaptive time-stepping algorithm based on
the work of Hairer et al. [16]. Given a solution at time $t_{n}$ we compute two separate approximations for the solution at time $t_{n+1}=t_{n}+\left(2 \times \Delta t_{n}\right)$. The first, $\theta_{1}$, is computed by solving the system of equations twice consecutively using a step size of $\Delta t_{n}$ whilst the second, $\hat{\theta}_{1}$, uses just one step of $\operatorname{size} 2 \times \Delta t_{n}$. Since both $\theta_{1}$ and $\hat{\theta}_{1}$ are approximating the same solution, we may obtain an estimate for the numerical error, ERR, at any given time step. For our purposes, we use the mesh-dependent $L_{2}$ measure

$$
\begin{equation*}
\mathrm{ERR}=\sqrt{\sum_{i=1}^{m}\left(z_{i+1}-z_{i}\right)\left(\frac{e_{i}+e_{i+1}}{2}\right)^{2}} \tag{3.43}
\end{equation*}
$$

where

$$
\begin{equation*}
e_{i}=\theta_{i}-\hat{\theta}_{i} \tag{3.44}
\end{equation*}
$$

Using the error measure calculated in (3.43), we can compute a suitable value for the next time step $\Delta t_{n+1}$ using the formula

$$
\begin{equation*}
\Delta t_{n+1}=\Delta t_{n} \times \min \left(\text { facmax }, \max \left[\text { facmin }, \mathrm{fac} \times\left(\frac{\mathrm{sc}}{\mathrm{ERR}}\right)^{\frac{1}{2}}\right]\right) \tag{3.45}
\end{equation*}
$$

where facmax and facmin are the maximum and minimum factors, respectively, by which $\Delta t_{n}$ may be scaled. Our safety factor, fac ( $\leq 1$ ), gives us the option of choosing how cautious the next time step $\Delta t_{n+1}$ will be relative to acceptance or rejection at $t_{n+1}$ (Algorithm 4, lines 10-17). The sc parameter is a user-defined error tolerance. For our simulations, based on the advice of Mekwi [30], we use facmax $=2.5$, facmin $=0.1$, fac $=0.8$ and $\mathrm{sc}=1.0 \times 10^{-7}$.

Note that we also use a 'reject' parameter to keep track of when a solution has been accepted or rejected. Upon rejection (i.e. reject $=1$ ), we set facmax $=1.0$ for the following simulation - otherwise, facmax is reset to its default value (Algorithm 4, lines 4-8). This is used to limit oscillations in $\Delta t_{n+1}$ between successive simulations [16].


Figure 3.1: Reject at $t_{n+1}$ and re-simulate using $\Delta t_{n+1}$. Blue denotes the previous timestep and green denotes the current timestep.

```
Algorithm 4 Time adaption algorithm
Require: \(\Delta t_{n}\), reject, facmin, facmax, facmax \({ }_{\text {init }}\), fac, sc
    compute: \(\theta_{1}, u_{1}, \hat{\theta}_{1}, \hat{u}_{1} / /\) updated using the preferred solution method
    compute: ERR // computed using equation (3.43)
    if reject \(=1\) then
        facmax \(=1.0\)
    else
        facmax \(=\) facmax \(_{\text {init }}\)
    end if
    compute: \(\Delta t_{n+1} / /\) computed using cquation (3.45)
    if (ERR/sc) < 1.0 then
        reject \(=0\)
        \(\theta^{(n+1)}=\theta_{1}\)
        \(u^{(n+1)}=u_{1}\)
        \(n=n+1\)
    else
        reject \(=1\)
        \(\Delta t_{n}=\Delta t_{n+1}\)
    end if
```



Figure 3.2: Accept at $t_{n+1}$, where (a) $\Delta t_{n}<\Delta t_{n+1}$, (b) $\Delta t_{n}<\Delta t_{n+1}$. Blue denotes the previous timestep and green denotes the current timestep.

Figure 3.1 illustrates the process of rejection at $t_{n+1}$. In this case, the valuc of ERR computed by equation (3.43) is deemed to be too large, and so the simulation restarts at time $t_{n}$ using the $\Delta t_{n+1}$ computed by equation (3.45). Note that, by equation (3.45), when a solution is rejected we have $\Delta t_{n+1}<\Delta t_{n}$. The process of accepting a solution at $t_{n+1}$ is illustrated in Figure 3.2. In each of the cases presented in Figure 3.2, the value of ERR computed by equation (3.43) is sufficiently small and the simulation continues from time $t_{n+1}$ using the $\Delta t_{n+1}$ computed by equation (3.45). It is important, however, to note the distinction between Figure 3.2(a) and Figure 3.2(b). In Figure 3.2(a) the error is so small that we can afford $\Delta t_{n+1}$ to be larger than $\Delta t_{n}$. In Figure 3.2(b) we have fac ${ }^{2}<\mathrm{ERR} / \mathrm{sc}<1.0$ which, by equation (3.45), translates to a smaller value of $\Delta t_{n+1}$ relative to $\Delta t_{n}$. This is important, as it allows our simulations to continue evolving in time with smaller, more cautious step sizes as $\mathrm{ERR} /$ sc approaches 1.0 , reducing the total number of rejections in the simulation.

Figure 3.3(a) shows the time evolution with step number for an example simulation using our time adaption algorithm. The simulation runs for a total of $6.0 \times 10^{-2}$ seconds (modelled-time), with a bipolar pulse which is applied at $2.0 \times 10^{-2}$ seconds. The polarity of the pulse is switched at $3.0 \times 10^{-2}$ seconds and the voltage is removed at $4.0 \times 10^{-2}$ seconds.


Figure 3.3: (a) Time increasing by timestep, and (b) $\Delta t_{n}$ adapting by timestep. Voltage-on, polaritychange and voltage-off are denoted by the red, green and blue dashed lines, respectively.

The corresponding evolution of $\Delta t_{n}$ is shown in Figure 3.3(b). Initially a small number of steps is used to get to time $=0.02$ seconds. Then, once the voltage is applied, $\Delta t_{n}$ becomes very small due to rapid changes in the system of equations. As the system approaches its equilibrium state, $\Delta t_{n}$ increases until, around step 6000 , it reaches its maximum range of values which maintain stability. This behaviour is observed again, around step 10000, when the polarity is changed and when the voltage is removed.

It should be noted that $\Delta t_{n}$ has range ( $4.3745 \times 10^{-11}, 1.5234 \times 10^{-4}$ ) and that a total of $5 \times 10^{4}$ timesteps were taken during this simulation. Using a constant timestepping algorithm, a total of $6.0 \times 10^{-2} / 4.3745 \times 10^{-11} \approx 1.37 \times 10^{9}$ timesteps would be required in order to obtain the same level of accuracy.

### 3.5 Moving mesh algorithm (MMA)

Until now, a fixed number of node points, $N$, were evenly spaced along the $z$-axis for the duration of each simulation. The disadvantage in doing this is that, particularly when modelling high voltages, steep gradients in $\theta$ can develop in certain parts of the cell, which can lead to inaccuracy and instability. To maintain stability, we require a high density of node points at regions with particularly steep gradients. Conversely, we can afford to have fewer node points where $\theta$ does not vary significantly.

Unfortunately, we will generally not know where steep gradients will occur prior to running a simulation. Furthermore, it is possible for the gradient in certain parts of the cell to vary from low to high at different times in the simulation.

One possible solution is to increase $N$ until the density of node points is great enough to cope with relatively high gradients in any part of the cell. However, this results in many wasted CPU cycles and excessively long simulation times. A far better solution would be to distribute the available node points in an intelligent manner at each timestep, such that areas with higher gradients receive a higher density of points than areas with lower gradients. This technique is known as r-refinement.

Note that, in initial simulations, a suitable $N$ would be chosen prior to each simulation. One problem with keeping $N$ fixed is that the complexity of the solution can vary as the simulation evolves. This means that we need to over-estimate $N$ so as to accommodate the single most complex timestep of the simulation, resulting in wasted CPU cycles throughout the more simple parts of the simulation. A far better strategy is to compute an appropriate $N$ at each timestep. This kind of adaption is known as $h$-refinement.

We present a combination of the two aforementioned techniques, namely hr-adaption, in an effort to accurately simulate our system of equations as efficiently as possible.

### 3.5.1 $r$-refinement

Let $N^{(n)}+1$ denote the number of node points at time $t_{n}$. The general principle behind $r$-refinement relies on us having some measure of the relative error at each node-point in our domain, which is obtained from a specific monitor function. The node points are then
repositioned such that the error is equidistributed throughout the domain. That is, the $z$ coordinate of each of the available $N^{(n+1)}+1$ node points is interpolated such that the error at each point in the domain is equal to some quantity $\delta$ (Algorithm 5 , lines 4-13), with

$$
\begin{equation*}
\delta=\frac{s_{N^{(n)}}}{N^{(n+1)}} \tag{3.46}
\end{equation*}
$$

where

$$
\begin{equation*}
s_{j}=\int_{z=0}^{z_{j}} M(\theta(z, t), z) d z \approx \sum_{i=1}^{j} M_{i}, \quad j=1, \ldots, N^{(n)}, \tag{3.47}
\end{equation*}
$$

and $M(\theta(z, t), z)$ is the monitor function. Clearly, the choice of monitor function is critical to the performance of this routine. Regardless of which monitor function is chosen, it has been shown $[12,15]$ that some form of smoothing should be applied to obtain reasonable accuracy in the computed solution. We employ the same smoothing algorithm as Mekwi [31], namely

$$
\begin{equation*}
\hat{M}_{i}=\frac{\sum_{k=i-p}^{i+p} M_{k} \hat{\gamma}^{|k-i|}}{\sum_{k=i-p}^{i+p} \hat{\gamma}^{|k-i|}} \tag{3.48}
\end{equation*}
$$

where $\hat{M}$ is a smoothed monitor function, $p$ is the smoothing index and $\hat{\gamma}$ is the rescaled smoothing parameter. The final moving mesh equation is now obtained by replacing $M_{i}$ with $\hat{M}_{i}$ in equation (3.47). Note that the smoothing index, $p$, is used to denote the range of smoothing, or averaging, and thus a higher value of $p$ will increase the computational expense whilst giving a smoother mesh. Huang et al. [19] recommend using a value of $p=1$, 2 or 3.

For our purposes, we use the well-known scaled arc-length monitor function which is given by

$$
\begin{equation*}
M(\theta(z, t), z)=\sqrt{1+\left|\theta_{z}(z, t)\right|^{2}} \tag{3.49}
\end{equation*}
$$

With this particular monitor function, the node points are positioned such that the arclength of the solution $\theta$ between any two adjacent node points is equal.

The complete routine, adapted from the work of Sanz-Serna and Christie [41], is outlined in Algorithm 5. Note that in cases of exclusive r-adaptivity $N^{(n+1)}=N^{(n)}$.

In all simulations which follow, we have chosen a smoothing parameter of $\hat{\gamma}=2 / 3$ and a smoothing index of $p=3$. Figure 3.4 illustrates the advantages of $r$-refinement. Note that

```
Algorithm 5 r-refinement algorithm
Require: \(N^{(n+1)}, N^{(n)}, z^{(n)}\)
    compute: \(M / /\) any suitable monitor function, in our case equation (3.49)
    compute: \(\hat{M I} / /\) computed using equation (3.48)
    compute: the \(s_{j}\) and \(\delta / /\) computed using equations 3.47 and 3.40, respectively
    for \(i=2, N^{(n+1)}\) do
        compute: \(l=(i-1) \delta\)
        for \(j=2,\left(N^{(n)}+1\right)\) do
            if \(b \leq s_{j}\) then
                compute: \(z_{i}^{(n+1)}=z_{j-1}^{(n)}+\left(b-s_{j-1}\right) \Delta z_{j-1}^{(n)} /\left(s_{j}-s_{j-1}\right)\)
                GO TO 12
            end if
        end for
        CONTINUE
    end for
```

(a) Time $=3.5$ milliseconds

(b) Time 3.6 milliseconds


Figure 3.4: Director profiles (a) just before, and (b) just after a change in voltage polarity, using r-adaption with $N=50$ points.
before the change in polarity of the voltage (plot (a)), not many node points are required near $z=0$. However, shortly after changing the polarity (plot (b)), the system changes in such a way that more node points are needed near $z=0$. Due to the $r$-adaption employed, the available node points are shifted in such a way that we do not observe mesh starvation near $z=0$. However, the need for points near $z=0$ causes a reduction in points near $z=d$.

This is because $N$ is fixed.

### 3.5.2 $h$-refinement

To be more flexible, we may want to increase or reduce $N$ as time progresses. We begin by defining a monitor function which measures the global spatial error in the solution at time $t$, namely

$$
\begin{equation*}
\eta(t)=\left[\int_{0}^{d} M_{2}(\theta(z, t)) d z\right]^{2} \tag{3.50}
\end{equation*}
$$

where

$$
\begin{equation*}
M_{2}(\theta(z, t))=\sqrt{\left|\theta_{z z}(z, t)\right|} \tag{3.51}
\end{equation*}
$$

Notice that $M_{2}(\theta(z, t))$, and hence $\eta(t)$, provides a measure of the curvature in $\theta$ throughout the cell at time $t$. Such a monitor function is ideally suited to our particular problem. For example, as $\theta$ becomes straighter the curvature in $\theta$ decreases, implying that fewer nodes are needed in order to obtain an accurate solution. Conversely, as $\theta$ becomes more nonlinear the curvature in $\theta$ increases, implying that more nodes are needed to obtain an accurate solution.

At each timestep, we wish to ensure that

$$
\begin{equation*}
\beta R T O L \leq \eta(t) \leq \alpha R T O L \tag{3.52}
\end{equation*}
$$

where $R T O L$ is a user-defined error tolerance, $\alpha>1$ and $0<\beta<1$. The basic principle is to increase the number of node points when the error is too high, giving greater accuracy, and to reduce the number of node points when the error is sufficiently small, giving greater speed. Note that when equation (3.52) is satisfied we need not alter the number of node points. However, should $\eta(t)$ lie outside of the range specified in equation (3.52), we either add or remove node points such that $\eta(t)$ is closer to our acceptable error, $R T O L$.

Now $N^{(n+1)}$ is given by

$$
\begin{equation*}
N^{(n+1)}=\left(N^{(n)}+1\right) \times \min \left(\operatorname{maxfac}, \max \left[\operatorname{minfac}, \kappa\left(\frac{\eta(t)}{R T O L}\right)^{\frac{1}{2}}\right]\right) \tag{3.53}
\end{equation*}
$$

where maxfac and minfac are the maximum and minimum factors, respectively, by which $N^{(n)}$ may be scaled. Our safety factor, $\kappa(\geq 1)$, gives us the option of choosing how cautious each $N^{(n+1)}$ will be relative to our criteria for adding and removing node points.

For our simulations, we use $\alpha=1.05, \beta=0.95, \operatorname{maxfac}=2.0, \operatorname{minfac}=0.3, \kappa=1.0$ and $R T O L=1.0 \times 10^{-2}$. So the number of nodes is only altered when $\eta(n)$ is more than $5 \%$ from IRTOL, and $N^{(n+1)}$ may be, at most, double $N^{(n)}$ and, at least, $30 \%$ of $N^{(n)}$.

Note that equation (3.53) does not necessarily produce an integer, so we always round down $N^{(n+1)}$ to the nearest integer,

$$
\begin{equation*}
N^{(n+1)} \leftarrow \operatorname{INT}\left(N^{(n+1)}\right) \tag{3.54}
\end{equation*}
$$

where the notation INT() implies taking the integer part.

```
Algorithm \(6 h\)-refinement algorithm
    compute: \(\eta(t) / /\) computed using equation (3.50)
    if \(\beta R T O L \leq \eta(t) \leq \alpha R T O L\) then
        \(N^{(n+1)}=N^{(n)}\)
    else
        compute: \(N^{(n+1)} / /\) computed using equations (3.53) and (3.54)
    end if
    if \(N^{(n+1)}<N_{\min }\) then
        \(N^{(n+1)}=N_{\text {min }}\)
    else if \(N^{(n+1)}>N_{\max }\) then
        \(N^{(n+1)}=N_{\max }\)
    end if
```

The complete routine for $h$-refinement is given in Algorithm 6. Note we will specify that $N^{(n+1)}$ must satisfy the condition

$$
\begin{equation*}
N_{\min } \leq N^{(n+1)} \leq N_{\max } \tag{3.55}
\end{equation*}
$$

where $N_{\min }$ and $N_{\max }$ are the minimum and maximum number of allowed node points, respectively. In simulations, we typically choose $N_{\min }=50$ and $N_{\max }=500$.

Figure 3.5 illustrates the advantages of $h$-refinement. From Figure 3.5(a), we see that fewer node points are required in order to represent a roughly linear director profile. However, more node points are required in order to accurately represent the more complicated director profile in Figure 3.5(b). With $h$-refinement, an appropriate number of node points are automatically chosen at each time-step.
(a) Time $=\mathbf{1 . 0}$ milliseconds

(b) Time 3.6 milliseconds


Figure 3.5: An example of $h$-adaption. (a) The solution is relatively simple, and only $N=50$ points are required, and (b) the solution is more complicated and $N=155$ points are required.

### 3.5.3 $h r$-refinement

Our final moving mesh algorithm uses a combination of the $r$ - and $h$-refinement techniques discussed in the previous sections. This form of adaption, known as $h r$-refinement, is outlined in Algorithm 7.

```
Algorithm \(7 h r\)-refinement algorithm
Require: Parameter values
Require: \(n=0\), reject \(=0, \Delta t_{0}, \theta^{(0)}\)
    while \(t_{n}<\) tmax do
        compute: \(\theta^{(n+1)}, u^{(n+1)}\) using Algorithm 4
        compute: \(N^{(n+1)}\) using Algorithm 6
        compute: \(z^{(n+1)}\) using Algorithm 5
        Interpolate \(\theta^{(n+1)}, u^{(n+1)}\) from \(z^{(n)}\) to \(z^{(n+1)}\)
    end while
```

As can be seen from Algorithm 7, the only important factor is the order in which the two individual forms of adaption are preformed. Clearly $h$-refinement needs to occur first, since the $N^{(n+1)}$ must be known before $r$-refinement may be used. The final step is to update
$\theta(z, t)$ and $u(z, t)$ from $z^{(n)}$ to $z^{(n+1)}$. To achieve this, we use cubic spline interpolation. Notice that a uniform mesh may be used by simply removing lines 3-5 of Algorithm 7.

It should be noted that we perform grid adaption based solely on the director profile, 0 , and not the flow profile. One of the key reasons for this is that as the director approaches any equilibrium state the flow profile tends towards $u=0$ throughout the cell. Therefore, if our moving mesh was dependent on just the flow profile, there is the potential for our mesh to be insufficient for representing complicated equilibrium states in $\theta$. Alternatively, using both the flow profile and the director profile to formulate our moving mesh would involve nondimensionalising each variable and performing some kind of weighting to each variable, which would add significantly to the computational expense of our method.

We also considered using two separate grids for our simulations, with one grid for the director profile and the other grid for the flow profile. This proved to be somewhat difficult to implement, since all interactions between $\theta$ and $u$ needed to be interpolated between the two grids. Furthermore, the Jacobian matrix in our fully-implicit method became very difficult to compute, and we were unable to use our efficient pentadiagonal algorithm for solving the resulting system, meaning that, computationally, there was no advantage in using two grids.

### 3.6 Method comparisons - no-flow model

In this section we compare each of our simulation methods for the no-flow model. All simulations model the same cell which uses the default parameter set (Table 2.1), and we simulate from a Vertical state with an initial relaxation of 1 ms followed by a bipolar pulse (of length $\tau=2.5 \mathrm{~ms}$ and voltage +50 volts) and a final relaxation of 20 ms . Each method is benchmarked in terms of speed and accuracy, and our ultimate goal is to determine which method gives an optimal balance of accuracy and speed.

All benchmarks are carried out on Intel Xeon E5345 processors running at 2.33 GHz , with CPU-time measured using the UNIX time command.

### 3.6.1 General motivation

Given the many different solution methods we have at our disposal, the next obvious step would be to determine which combination gives an optimal balance between solution accuracy and overall efficiency. As a starting point, we used two simulations to conduct a simple comparison between grid types. Both simulations used the fully-implicit method and our adaptive time-stepping algorithm, but the first simulation employed our moving mesh algorithm whilst the second simulation used a 51 -point uniform grid. The results of these simulations are displayed in Figure 3.6.


Figure 3.6: A comparison between the moving mesh algorithm (black line) and a 51-node uniform grid (red line) when simulating from an initial Vertical state, using the base parameter set, a voltage of $V=50$ volts and a dwell-time of $\tau=2.5 \mathrm{~ms}$. Plot (I) shows the director profile at each of the key points in our bipolar pulse, whilst plot (II) shows the corresponding relaxations immediately after the voltage is removed.

If each grid-type is sufficiently accurate, we would expect some level of agreement between the simulation results. However, from Figure 3.6(I)(D), we see that the two simulations relax to different final states. Further examination reveals that when the voltage changes polarity (Figure 3.6(I)(B)) and when the voltage is removed (Figure 3.6(I)(C)) there are significant differences between the director profiles at $z=d$ and $z=0$, respectively. Figure 3.6(II) shows the initial relaxation of the director immediately after the voltage is removed, and we
can see that the differences at $z=0$ magnify as the director profiles evolve.

While we would expect the moving mesh algorithm to produce the more accurate results, Figure 3.6 shows that there is no obvious sign of numerical error in either case, with both simulations evolving in a somewhat 'sensible-looking' fashion. This highlights the importance of testing the solution accuracy across all grid types, as a poorly chosen grid can give incorrect results.

### 3.6.2 Solution accuracy

To measure the solution accuracy throughout the evolution of a simulation, accounting for any error accumulation, we require a good approximation to the true solution at each timestep in order to compare against. Note that as the number of node points $\rightarrow \infty$, our discretised differential equations approach continuous differential equations. Therefore, simulating with a sufficiently high number of node points allows us to obtain a good approximation to the exact solution, $\theta_{\text {exact }}$, at each timestep. Now, given some approximation to $\theta_{\text {exact }}$, say, $\hat{\theta}$, the maximum absolute error in $\hat{\theta}$ at each time step is given by

$$
\begin{equation*}
E_{\infty}(\hat{\theta})=\max \left|\hat{\theta}-\theta_{\text {exact }}\right|, \tag{3.56}
\end{equation*}
$$

whilst the Euclidean error at each time step is given by

$$
\begin{equation*}
E_{2}(\hat{\theta})=\sqrt{\frac{1}{N} \sum_{i=1}^{N}\left(\hat{\theta}_{i}-\theta_{\text {exact }_{i}}\right)^{2}} . \tag{3.57}
\end{equation*}
$$

Finally, for any given simulation, we measure the average infinity and Euclidean errors, across all time steps, which we denote $\bar{E}_{\infty}(\hat{\theta})$ and $\bar{E}_{2}(\hat{\theta})$, respectively. Note that $\theta_{\text {exact }}$ must be interpolated onto the same grid as $\hat{\theta}$ before computing the crror measures given by equations (3.56)-(3.57). We achieve this using cubic spline interpolation. For our purposes, $\theta_{\text {exact }}$ is computed on a uniform grid using 100,001 node points.

### 3.6.3 Differences between time integration methods

We briefly compare the differences between each of the time integration methods as the number of node points are increased. Figure 3.7 shows the CPU-time varying with number of
node points. Note, from Figure 3.7 (b), that for $\geq 186$ node points, the fully-implicit method is the fastest of the three methods when a uniform grid is used. Also, CPU-time for the fully-implicit method increases roughly linearly as the number of node points is increased, whilst for the explicit and semi-implicit methods the CPU-time increases exponentially as the number of node points increases.


Figure 3.7: CPU-time varying with the number of nodes for each of the three time integration methods (black $\Rightarrow$ explicit method, red $\Rightarrow$ semi-implicit method, blue $\Rightarrow$ fully-implicit method). Note that plot (b) is a magnification of plot (a).

Table 3.1 shows how the CPU-time varies as the node points are doubled for each time integration method on a uniform grid, along with the CPU-time required when a moving mesh algorithm is used. Note that Table 3.1 uses the same data as in Figure 3.7. We see that the fully-implicit method scales very well as the number of nodes are increased. The poor scaling of the explicit and semi-implicit methods may be attributed to the Courant-Friedrichs-Lewy (CFL) Condition [44] which states that an explicit method is unstable unless the time step remains sufficiently small relative to the distance between adjacent node points. By increasing the number of node points, we are automatically decreasing the distance between node points. Therefore, in addition to requiring more CPU-time at each time-step, the explicit and semi-implicit methods require more (smaller) time-steps for each simulation.

The accumulated errors for each time integration method are also shown in Table 3.1. Notice that the accumulated error is almost exactly the same for each method.

Additionally, from Figure 3.8, we see that the errors are almost identical for each of the

Table 3.1: The CPU-time (measured in seconds) and the average value of $E_{\infty}(\hat{\theta})$, over all time-steps, for each time integration method as we increase the number of node points.

| Nodes | Time Integration Scheme |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | Explicit |  | Semi-Implicit |  | Fully-Implicit |  |
|  | CPU-time | $\bar{E}_{\infty}(\hat{\theta})$ | CPU-time | $\bar{E}_{\infty}(\hat{\theta})$ | CPU-time | $\bar{E}_{\infty}(\hat{\theta})$ |
| 51 | 0.76 | 0.713258 | 0.78 | 0.713250 | 1.51 | 0.713247 |
| 101 | 2.14 | 0.711652 | 1.88 | 0.711648 | 3.07 | 0.711644 |
| 201 | 13.66 | 0.710837 | 7.24 | 0.710834 | 6.24 | 0.710831 |
| 401 | 218.32 | 0.074564 | 69.13 | 0.074708 | 12.18 | 0.074753 |
| 801 | 1760.89 | 0.021531 | 550.73 | 0.021573 | 24.71 | 0.021580 |
| 1601 | 14142.94 | 0.005382 | 4415.86 | 0.005398 | 47.76 | 0.005397 |
| MMA (51:81) | 61.29 | 0.006614 | 23.61 | 0.006552 | 2.55 | 0.006590 |



Figure 3.8: Accumulated error measures varying with the number of node points (black $\Rightarrow$ explicit method, red $\Rightarrow$ semi-implicit method, blue $\Rightarrow$ fully-implicit method).
three methods at each time-step. This implies that the adaptive time-stepping algorithm controls the error in exactly the same way, regardless of which time integration method is used. This also shows that the errors are dependent on the spatial grid used and not the time integration method. Note that for this run of our model, with $\leq 231$ nodes we converge to the wrong final state, whilst $\geq 236$ nodes gives the correct final state. This corresponds to the jump in CPU-time that occurs when the nodes are increased from 231 to 236 in Figure
3.7(b). So the semi-implicit and explicit methods require significantly more CPU-time to obtain the correct final state.

Since the time integration method has been shown to have no effect on the accumulated solution error, we only use the fully-implicit method throughout the remainder of this section.

### 3.6.4 Differences between mesh types

Table 3.2: Accumulated error measures for a variety of different uniform grids with the corresponding errors relating to the moving mesh algorithm. The CPU-time is also provided in each case. We use the fully-implicit method in each case, and $\theta_{\text {exact }}$ is simulated using a uniform grid with 100,001 node points.

| MMA | Node points | CPU-time (s) | $\max E_{\infty}(\hat{\theta})$ | $\max E_{2}(\hat{\theta})$ | $\bar{E}_{\infty}(\hat{\theta})$ | $\bar{E}_{2}(\hat{\theta})$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| off | 51 | 1.51 | 1.206258 | 0.896107 | 0.713247 | 0.493587 |
| off | 101 | 3.07 | 1.206258 | 0.882171 | 0.711644 | 0.488309 |
| off | 201 | 6.24 | 1.206256 | 0.870989 | 0.710831 | 0.483301 |
|  |  |  |  |  |  |  |
| off | 401 | 12.18 | 0.262178 | 0.008736 | 0.074753 | 0.003240 |
| off | 801 | 24.71 | 0.074328 | 0.002336 | 0.021580 | 0.000851 |
| off | 1601 | 47.76 | 0.018112 | 0.000646 | 0.005397 | 0.000216 |
|  |  |  |  |  |  |  |
| on | $(51: 81)$ | 2.55 | 0.022638 | 0.001717 | 0.006590 | 0.000766 |

Table 3.2 shows the accumulated errors, over all time steps, for $\hat{\theta}$, together with the CPUtime required for each simulation as the number of node points is varied. As expected, increasing the number of node points increases the required CPU-time whilst decreasing the accumulated error. Also note that the moving mesh algorithm uses no more than 81 node points at any given time throughout the simulation. Notice that, for this particular cell, simulating with 51-201 node points on a uniform mesh produces extremely inaccurate results. From Figure 3.9, note that with $<236$ nodes the simulations converge to the wrong final state. Therefore, the moving mesh algorithm is the fastest scheme for which the correct


Figure 3.9: Accumulated infinity error evolving through time for four different uniform meshes. Note that the red line is used to denote the accumulated infinity error when a moving mesh is employed, and is therefore the same in each plot. The implicit method is used in each case.
final state is obtained.


Figure 3.10: Accumulated error measures varying with the number of node points. Note that the red dashed line is used to denote the error measure when a moving mesh is employed. The implicit method is used in each case.

From Figure 3.10(a), we see that $\geq 1441$ nodes are required on a uniform grid to achieve a lower maximum infinity error than the moving mesh algorithm. Similarly, from Figure 3.10 (b), we require $\geq 951$ nodes to obtain a lower maximum Euclidean error than the moving mesh, whilst Figure 3.10(c)-(d) shows that more than 1456 and 846 nodes are needed to achieve a lower average infinity error and a lower average Euclidean error, respectively, than the moving mesh algorithm.

Table 3.3 shows that, regardless of which error measure is used, the uniform grid requires significantly more CPU-time to achieve the same level of accuracy as the moving mesh algorithm. At best, the uniform grid takes 9.89 times as long to simulate to the same accuracy, and at worst the uniform grid is shown to take over 19 times as long as the moving

Table 3.3: The number of node points required, on a uniform grid, to obtain a lower accumulated error than the moving mesh algorithm for each error measure. Also included is the corresponding CPU-time and its ratio relative to the MMA CPU-time ( $=2.55$ seconds).

| Error | Nodes Required | CPU-time (s) | CPU-time Ratio |
| :---: | :---: | :---: | :---: |
| $\max E_{\infty}(\hat{\theta})$ | 1441 | 49.17 | 19.28 |
| $\max E_{2}(\hat{\theta})$ | 951 | 29.16 | 11.44 |
| $\bar{E}_{\infty}(\hat{\theta})$ | 1456 | 43.36 | 17.00 |
| $\bar{E}_{2}(\hat{\theta})$ | 846 | 25.23 |  |

mesh algorithm.

### 3.6.5 Extension to other voltages

So far we have only considered the case in which 50 volts are applied to our cell when simulating from an initially Vertical state. We now investigate how the CPU-time varies with applied voltage, with the results shown in Figure 3.11.


Figure 3.11: CPU-time plotted against applied voltage for each of the time integration methods (black $\Rightarrow$ explicit, red $\Rightarrow$ semi-implicit, blue $\Rightarrow$ fully-implicit) simulating with a moving mesh algorithm and starting from an initial (a) HAN state, and (b) Vertical state.

Comparing Figure 3.11(a) with Figure 3.11(b) shows that the initial state of the director has a negligible effect on the CPU-time required to simulate a cell without flow effects, regardless of which method is used. Also, higher voltages generally require more CPU-time to simulate. This is somewhat expected, as our adaptive time-stepping algorithm is designed to force small time steps when the system undergoes rapid change, and we expect higher voltages to alter the system at a faster rate.

It should be noted that each of the lines in Figure 3.11 are not symmetrical about $V=0$. This is to be expected, since the flexoelectric effect at each surface is dependent on the polarity of the voltage. Therefore negative and positive voltages give different results, which is then reflected in the CPU-time.

From Figure 3.11, we see that the explicit (and, to a lesser extent, the semi-implicit) method uses considerably more CPU-time than the fully-implicit method as the voltage increases in magnitude. This may be attributed to the CFL Condition. Higher voltages typically induce steeper gradients in the system, and since our moving mesh algorithm is specifically designed to increase the density of node points around steeper gradients, the CFL Condition suggests that more CPU-time should be required to simulate higher voltages using an explicit scheme. Since the semi-implicit method contains some explicit terms, we also expect (by the CFL Condition) simulations to take longer when a higher voltage is applied.

For $>15$ volts, the fully-implicit scheme proves to be the fastest method. For lower voltages, the fully-implicit method is seen to be marginally slower than the explicit and semi-implicit schemes. However, over all voltages the fully-implicit scheme is significantly faster then the explicit and semi-implicit schemes and is therefore our preferred form of time-integration.

### 3.6.6 The optimal method

We have shown that, given a spatial grid, all three time integration schemes produce almost exactly the same results for this particular problem. However, for a higher number of node points ( $\geq 186$ ), the explicit and semi-implicit methods demand substantially more CPU-time than the fully-implicit method. We therefore recommend using the fully-implicit method.

Regarding grid-types, we have shown that a moving mesh algorithm is almost essential in order to prevent large accumulated errors over the course of a simulation whilst maintain-
ing a relatively low CPU-time. We therefore conclude that the fully-implicit method with a moving mesh algorithm gives an optimal balance of speed and accuracy.

In addition to the benefits in terms of CPU-time, the moving mesh algorithm also has the advantage of requiring very little user intervention. Much like the adaptive time-stepping algorithm, the moving mesh algorithm ensures that a high degree of accuracy is obtained as efficiently as possible, regardless of which cell type is used.

So far we have only considered the no-flow model. Close to $z=0$ and $z=d$, the $\theta$ equation typically exhibits high gradients and significant changes over time. However, we will see that a sterner test of our numerical method will be in the flow model, where steep gradients and small time-scale changes can occur anywhere in the cell.

### 3.7 Method comparisons - flow model

In this section we compare each of our simulation methods for the flow model. All simulations model the same cell which uses the default parameter set (Table 2.1), and we simulate from a HAN state with an initial relaxation of 1 ms followed by a bipolar pulse (of length $\tau=2.5 \mathrm{~ms}$ and voltage +50 volts) and a final relaxation of 12 ms . Each method is benchmarked in terms of speed and accuracy, and our ultimate goal is to determine which method gives an optimal balance of accuracy and speed.

As with the no-flow model, all benchmarks are carried out on Intel Xcon E5345 processors running at 2.33 GHz , with CPU-time measured using the UNIX time command.

### 3.7.1 Solution accuracy

As in the no-flow case (Section 3.6.2), we compute the maximum absolute error in $\hat{\theta}, E_{\infty}(\hat{\theta})$, and the Euclidean error in $\hat{\theta}, E_{2}(\hat{\theta})$, using equations (3.56)-(3.57). Additionally, given a good approximation to the exact solution of the flow velocity equation, $u_{\text {exact }}$, we may compute the maximum absolute error in $\hat{u}$ at each timestep by

$$
\begin{equation*}
E_{\infty}(\hat{u})=\max \left|\hat{u}-u_{\text {exact }}\right| \tag{3.58}
\end{equation*}
$$

whilst the Euclidean error at each time step is given by

$$
\begin{equation*}
E_{2}(\hat{u})=\sqrt{\frac{1}{N} \sum_{i=1}^{N}\left(\hat{u}_{i}-u_{\text {exact }_{i}}\right)^{2}} \tag{3.59}
\end{equation*}
$$

For this section, the exact solutions $\theta_{\text {exact }}$ and $u_{\text {exact }}$ are computed on a uniform grid using 100,001 node points.

### 3.7.2 Differences between time integration methods

We again compare the differences between each of the time integration methods as the number of node points are increased. It should be noted that we do not provide any results using the explicit method, since all attempts to use our explicit method failed, producing
completely unstable flow profiles

Figure 3.12 shows the CPU-time varying with number of node points. Note, from Figure 3.12 (b), that the semi-implicit method is the faster method for $<221$ node points, whilst the implicit method is the faster for $>221$ node points. Also, CPU-time for the implicit method increases roughly linearly as the number of node points is increased, whilst for the semi-implicit method the CPU-time increases exponentially as the number of node points increases.


Figure 3.12: CPU-time varying with the number of nodes for each of the three time integration methods (red $\Rightarrow$ semi-implicit method, blue $\Rightarrow$ fully-implicit method). Note that plot (b) is a magnification of plot (a).

Table 3.4 shows how the CPU-time varies as the node points are doubled for each time integration method on a uniform grid, along with the CPU-time required when a moving mesh algorithm is used. Note that the fully-implicit method scales very well as the number of nodes are increased. As in the no-flow case, the poor scaling of the semi-implicit method may be attributed to the CFL Condition.

The accumulated errors for each time integration method are also shown in Table 3.4. Notice that the accumulated error is almost exactly the same for each method with respect to the $\theta$ variable. However, there are notable differences between the methods when considering the $u$ variable. Note that we expect there to be some fluctuation here, since the semi-implicit method decouples the $\theta$ and $u$ equations whilst the fully-implicit method solves both equations simultaneously.

Table 3.4: The CPU-time (measured in seconds) and the average values of $E_{2}(\hat{\theta})$ and $E_{2}(\hat{u})$, over all time-steps, for each time integration method as we increase the number of node points.

| Nodes | Time Integration Scheme |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | Semi-Implicit |  |  | Fully-Implicit |  |  |
|  | CPU-time | $\bar{E}_{2}(\hat{\theta})$ | $\bar{E}_{2}(\hat{u})\left(\times 10^{-4}\right)$ | CPU-time | $E_{2}(\hat{0})$ | $E_{2}(\hat{u})\left(\times 10^{-4}\right)$ |
| 51 | 1.03 | 0.284695 | 0.208384 | 2.66 | 0.284683 | 0.208289 |
| 101 | 2.47 | 0.276309 | 0.202278 | 5.94 | 0.276297 | 0.201948 |
| 201 | 9.57 | 0.269487 | 0.196630 | 10.94 | 0.269483 | 0.196376 |
| 401 | 91.49 | 0.008354 | 0.059818 | 22.98 | 0.008359 | 0.060850 |
| 801 | 737.39 | 0.002017 | 0.015271 | 41.96 | 0.002016 | 0.016204 |
| 1601 | 5787.21 | 0.000505 | 0.005850 | 100.52 | 0.000504 | 0.005604 |
| MMA (51:81) | 32.53 | 0.001163 | 0.011229 | 9.96 | 0.001167 | 0.005578 |

Additionally, from Figure 3.13, we see that the errors are almost identical for each of the three methods at each time-step for the $\theta$ variable. As in the no-flow case, we can conclude from this that the adaptive time-stepping algorithm controls the error in exactly the same way, regardless of which time integration method is used. However, for the $u$ variable we see that the errors between methods are inconsistent. Again, we may relate this back to the decoupling of the $\theta$ and $u$ variables in the semi-implicit method. Also note that the adaptive time-stepping algorithm only concerns itself with the relative errors in $\theta$, leaving any errors in $u$ unregulated. Therefore, the semi-implicit method has no control over errors specific to $u$, whilst the fully-implicit method loosely controls errors in $u$ using its convergence criterion.

A further point to note from Figure 3.13(II) is that the maximum infinity errors in $\hat{u}$ (Figure 3.13(II)(a)) behave in an unpredictable manner, with an increase in node points not necessarily resulting in a decrease in $E_{\infty}(\hat{u})$. On the other hand, $E_{2}(\hat{u}), \bar{E}_{\infty}(\hat{u})$ and, particularly, $\bar{E}_{2}(\hat{u})$ behave in a much more sensible fashion. Since a higher number of node points are, in general, expected to provide a more accurate simulation, we conclude that $E_{\infty}(\hat{u})$ is an un-


Figure 3.13: Accumulated error measures varying with the number of node points (black $\Rightarrow$ semi-implicit method, red $\Rightarrow$ fully-implicit method). Plot (I) shows error measures for $\hat{\theta}$ whilst plot (II) shows the corresponding error measures for $\hat{u}$.
reliable error measure for $\hat{u}$. A far more useful and informative error measure is the average Euclidean error, $\bar{E}_{2}(\hat{u})$.

Since the time integration method has been shown to affect the accumulated solution error in the flow profile, we consider both the semi-implicit and fully-implicit methods throughout the remainder of this section.

### 3.7.3 Differences between mesh types

Table 3.4 shows the average Euclidean error, over all time steps, for both $\hat{\theta}$ and $\hat{u}$, together with the CPU-time required for each simulation as the number of node points is varied. As in the no-flow case, we see that increasing the number of node points increases the required CPU-time whilst decreasing the accumulated error. Also note that the moving mesh algorithm uses no more than 81 node points at any given time throughout the simulation. Notice that, for this particular cell, simulating with 51-201 node points on a uniform mesh produces extremely inaccurate results. From Figure 3.14, note that with $<290$ nodes the
(I)


Figure 3.14: Accumulated Euclidean error evolving through time for four different uniform meshes. In each case, the black line represents the number semi-implicit method whilst the red line represents the fullyimplicit method. We also show the accumulated Euclidean error when a moving mesh is employed (green $\Rightarrow$ semi-implicit method, blue $\Rightarrow$ fully-implicit method).
simulations actually converge to the wrong final state. Therefore, as in the no-flow case, the moving mesh algorithm is the fastest scheme for which the correct final state is obtained.

From Figure 3.15(I)(a) and Figure 3.15(II)(a), we see that significantly more than 1600 nodes are required on a uniform grid to achieve a lower maximum infinity error, for both $\hat{\theta}$ and $\hat{u}$, respectively, than the fully-implicit moving mesh algorithm. However, as discussed in Section 3.7.2, this may be due to the infinity error measure itself as opposed to the solution accuracy of the simulations. We also see from Figure 3.15(I) that, as previously stated, there is very little difference in the accumulated errors in $\hat{\theta}$ between different time integration methods. Regarding $\hat{u}$, we see from Figure 3.15(II) that the fully-implicit method performs significantly better than the semi-implicit method when a moving mesh is employed. This is likely to be due to the convergence criterion of the fully-implicit method, which requires that $\hat{u}$ obtains a specific relative error at each time-step before allowing the simulation to continue. The semi-implicit method, on the other hand, allows $\hat{u}$ to evolve in an unregulated fashion. It is interesting to note, however, that the discrepancies in $\hat{u}$ do not appear to have any measurable impact on $\hat{\theta}$.


Figure 3.15: Accumulated error measures varying with the number of node points. In each case, the black and red lines correspond to the 'Number of Nodes' axis for the semi-implicit and fully-implicit methods, respectively, when a uniform grid is used. The blue and green lines represent the semi-implicit and fullyimplicit methods, respectively, when a moving mesh is employed.

Table 3.5: The number of node points required, on a uniform grid, to obtain a lower average Euclidean error than the fully-implicit MMA. Also included is the CPU-time and its ratio relative to the fully-implicit MMA CPU-time ( $=3.96 \mathrm{~s}$ ).

| Error | Method | Nodes Required | CPU-time (s) | CPU-time Ratio |
| :---: | :---: | :---: | :---: | :---: |
| $\bar{E}_{2}(\hat{\theta})$ | Semi-Implicit | 1056 | 1655.05 | 417.94 |
|  | Fully-Implicit | 1056 | 60.27 | 15.22 |
|  |  |  |  |  |
|  |  |  | 5787.21 | 1461.42 |
| $\bar{E}_{2}(\hat{u})$ | Semi-Implicit | 1601 | 100.52 | 25.38 |
|  | Fully-Implicit | 1601 |  |  |

Table 3.5 shows that simulating with a uniform grid requires significantly more CPU-time to achieve the same level of accuracy as the fully-implicit moving mesh algorithm, regardless of which time integration method is employed. We see from Table 3.5 that, in order to achieve the same level of accuracy in $\hat{\theta}$, the uniform mesh requires roughly fifteen times as much CPU-time as the fully-implicit moving mesh algorithm. In order to achieve the same level
of accuracy in $\hat{u}$, the uniform mesh requires roughly twenty-five times as much CPU-time as the fully-implicit moving mesh algorithm. When comparing the semi-implicit uniform mesh with the fully-implicit moving mesh, note that the fully-implicit moving mesh is up to 1400 times faster.

Table 3.6: The number of node points required, on a uniform grid, to obtain a lower average Euclidean error than the semi-implicit MMA. Also included is the CPU-time and its ratio relative to the semi-implicit MMA CPU-time ( $=32.53 \mathrm{~s}$ ).

| Error | Method | Nodes Required | CPU-time (s) | CPU-time Ratio |
| :--- | :---: | :---: | :---: | :---: |
| $\bar{E}_{2}(\hat{\theta})$ | Semi-Implicit | 1056 | 1655.05 | 50.588 |
|  | Fully-Implicit | 1056 | 60.27 | 1.85 |
|  |  |  |  |  |
|  |  |  |  |  |
| $E_{2}(\hat{u})$ | Semi-Implicit | 986 | 1343.40 | 41.30 |
|  | Fully-Implicit | 986 | 56.35 | 1.73 |

Note from Table 3.6 that, while still faster than the uniform mesh simulations, the semiimplicit moving mesh simulation docs not give as high a performance increase as the fullyimplicit moving mesh when simulating to the same level of accuracy. This is partly due to the lower accuracy of $\hat{u}$ which is incurred with the decoupling of the equations in semiimplicit method. Also, as we saw in the no-flow case, the semi-implicit method requires smaller time-steps in order to compensate for the CFL condition.

### 3.7.4 Extension to other voltages

So far we have only considered the case in which 50 volts are applied to our cell when simulating from an initially HAN state. We now investigate how the CPU-time varies with applied voltage, with the results shown in Figure 3.16.

Comparing Figure 3.16(a) with Figure 3.16(b) shows that the initial state of the director significantly affects the CPU-time required to simulate a cell when modelling flow effects, especially when the semi-implicit method is used. This is due to the backflow and kickback effects associated with the Vertical state. As we shall see (Section 6.1), a large amount a director kickback is often produced when simulating from an initially Vertical state, which causes the moving mesh algorithm to focus a large number of node points in the centre of the cell. This results in a much higher number of operations at each time-step and, in the


Figure 3.16: The CPU-time plotted against the applied voltage for each of the time integration methods (red $\Rightarrow$ semi-implicit, blue $\Rightarrow$ fully-implicit), each using a Moving Mesh Algorithm and starting from (a) an initial HAN state, and (b) an initial Vertical state.
case of the semi-implicit method, an additional reduction in performance due to the CFL condition.

Also, as is the case with the no-flow model, higher voltages generally require more CPU-time to simulate, regardless of the initial state. However, this characteristic is far more pronounced for the semi-implicit method, where the CPU-time increases exponentially with the applied voltage. In contrast, the CPU-time for the fully-implicit method increases roughly linearly with the applied voltage.

For greater than 10 volts, the fully-implicit scheme proves to be the faster method when simulating from a HAN state, whilst for an initially Vertical state we see that the fully-implicit method is faster for anything greater than 11 volts. For lower voltages, the fully-implicit method is seen to be marginally slower than the semi-implicit scheme. However, over all voltages the fully-implicit scheme is significantly faster then the explicit and semi-implicit schemes and is therefore our preferred form of time-integration.

### 3.7.5 The optimal method

We have shown that both the semi-implicit and fully-implicit methods produce almost exactly the same results in terms of $\hat{\theta}$ for this particular problem. This is likely to be due to the time-adaption algorithm, which appears to control the errors in $\hat{\theta}$ in a consistent fashion
across both time integration methods. Errors in $\hat{u}$, on the other hand, are not regulated by the time adaption algorithm. This is reflected in Section 3.7, where we have shown the errors in $\hat{u}$ to be inconsistent across the time integration methods. The fully-implicit method was shown to be the more robust in modelling $\hat{u}$, which can be attributed to its convergence criterion which attempts to account for any irregularitics in both $\hat{\theta}$ and $\hat{u}$. This highlights a possible area for improvement in our numerical methods, namely expanding the adaptive time-stepping algorithm to account for discrepancies in the flow profile. Note that this would require additional research to be carried out on nondimensionalisation of the flow equation. For this particular test cell, the differences in $\hat{u}$ did not induce any noticeable differences in $\hat{\theta}$, though this may not necessarily hold true for all cases.

Regarding grid-types, we have shown that a moving mesh algorithm is absolutely essential in order to prevent large accumulated errors over the course of a simulation whilst maintaining a low CPU-time. As in the no-flow case, the moving mesh algorithm also has the inherent advantage of requiring no user intervention, thereby ensuring that a high degree of accuracy is obtained as efficiently as possible. We therefore conclude that the fully-implicit method with a moving mesh algorithm gives the optimal balance of speed and accuracy.

### 3.8 Summary

In this section, we have introduced the numerical methods that are to be used for solving the systems of equations derived in Section 2. Based on the results obtained in Sections 3.6 and 3.7, we have concluded that the fully-implicit method with adaptive time-stepping and a moving mesh algorithm provides the best balance of speed and solution accuracy. Therefore, we employ this method for the remainder of our investigation.

In the next section, we devise a routine for using our numerical methods to map the $\tau V$ plots corresponding to a specific parameter set.

## 4 Automatic $\tau V$-plot generation

In this section we develop a method for generating $\tau V$-plots corresponding to a specific parameter set. As stated in Section 1.6, a $\tau V$-plot is a graph which indicates whether or not a bistable cell switches between states when a specific voltage, $V$, is applied for a prescribed length of time, $\tau$. Voltage pulses that induce switching are of particular interest. However, there are no known analytical techniques for finding such areas and so, in order to map out a switching region, we need to test individual $(\tau, V)$ points and determine whether or not each point induces switching. It turns out that this requires far too many simulations to do by hand, and therefore we require some kind of automatic method.


Figure 4.1: $\tau V$-plot generated using the brute force method ( 2,500 points tested). Here, the blue shaded area represents a switch between states.

One method for finding switching regions is to overlay a grid of points on the testing area, test each point and record whether or not the system switches. This type of brute-force method (Figure 4.1) is extremely inefficient as we require a relatively fine grid in order to reasonably approximate the boundary of the switching region.

A far more efficient method is to be able to trace the outline of each switching region, as in Figure 4.2. We have therefore developed a highly optimised line-tracing algorithm for generating $\tau V$-plots. Our algorithm comprises four key parts, namely: (i) finding a single point that switches between states, (ii) locating the boundary of the switching region, (iii) tracing around the switching region, (iv) finding any additional switching regions.


Figure 4.2: $\tau V$-plot generated using the optimised tracing algorithm (278 points tested). Here, the blue shaded area represents a switch between states.

### 4.1 Detecting a switch between states

In order to produce a $\tau V$-plot, we need to be able to reliably detect whether or not a switch between states occurs for a given voltage pulse. When testing an individual $(\tau, V)$ point manually, this process is relatively straight-forward; the final state can be determined simply by looking at the evolution of the director profile as the cell relaxes - when the director is sufficiently close to one of the two stable states, we stop the simulation and record a 'switch' if and only if the final state differs from the initial state.

However, our code needs to be able to detect the final state automatically. Let us denote $\theta_{H}$ and $\theta_{V}$ as the director profiles for the zero voltage relaxed HAN and Vertical states, respectively, with $\theta(z, t)$ as the director profile for the current simulation at time $t$. An initial method used for determining the final state for a given test point was very crude, namely

1. Once the voltage had been removed, the simulation was given an arbitrary length of time to relax, $t=t_{\text {relax }}$ seconds.
2. After this relaxation period we examine the director orientation at the bistable surface, $\theta\left(0, t_{\text {relax }}\right)$.
3. If $\theta\left(0, t_{\text {relax }}\right)<\left(\alpha_{1}+\alpha_{2}\right) / 2$, the location of the surface energy maximum, we conclude that the test point relaxes to the HAN state. Otherwise, we conclude that the test

point relaxes to the Vertical state.

This method has a number of drawbacks. Since a common relaxation time $t_{\text {relax }}$ is used, many simulations are likely to relax for an unnecessarily long length of time, dramatically increasing simulation time. Furthermore, some simulations may not relax long enough, giving inaccurate readings. Clearly a more robust and efficient method is needed if we are to automatically detect switching in a bistable cell. Since simulation time is expensive, we ideally want to determine the final relaxed state as soon as possible once the voltage has been removed.


Figure 4.3: Simulations giving (a) the relaxed HAN state, and (b) the relaxed Vertical state.

In preparation for our improved state-detector, we must first compute the relaxed director profiles for the HAN and Vertical states. $\theta_{I I}(z)$ is obtained by simulating, with no applied field, for a sufficiently long period of time using a linear profile from $\theta(0)=\alpha_{1}$ to $\theta(d)=\pi / 2$ as our initial state, illustrated in Figure 4.3(a). To obtain $\theta_{V}(z)$, we use a linear profile from $\theta(0)=\alpha_{2}$ to $\theta(d)=\pi / 2$ as our initial state (Figure 4.3(b)).

Since we only need to compute the relaxed states once, we can afford to be over-cautious with the relaxation times used here. Also note that the adaptive time-stepping algorithm we employ will take larger time-steps the closer the director profile gets to an equilibrium solution, so longer relaxation times are always desirable. Our default relaxation time is taken to be 100 modelled seconds.

With $\theta_{H}(z)$ and $\theta_{V}(z)$ obtained, we may compute

$$
\begin{align*}
\theta_{M}(z) & =\left(\theta_{H}(z)+\theta_{V}(z)\right) / 2,  \tag{4.1}\\
\theta_{D}(z) & =\theta_{V}(z)-\theta_{H}(z), \tag{4.2}
\end{align*}
$$

the mid-line of the two states and the difference between the two states, respectively. Our safety-zone, $\theta_{S}(z)$, is then defined by the interval

$$
\begin{equation*}
\theta_{M}(z)-\left(0.5 \times \text { buffer } \times \theta_{D}(z)\right)<\theta_{S}(z)<\theta_{M}(z)+\left(0.5 \times \text { buffer } \times \theta_{D}(z)\right), \tag{4.3}
\end{equation*}
$$

where the 'buffer' $(<1.0)$ is a user-defined constant, typically taken to be $2.5 \times 10^{-2}$ in our simulations. The safety-zone is used to ensure that we are sufficiently close to a relaxed state before making our final decision.

Our final decision is based on the director orientation at $p$ critical test-points, $z_{\text {crit }}=$ $z_{1}, z_{2}, \ldots, z_{p}$, throughout the cell. It is essential for one of the test points to be placed at $z=0.0$, since it is on the bistable surface. We also advise at least one further test point be placed at $z=d / 2$, as this will give a good representation of the director profile in the bulk of the cell. For our purposes, we use $p=3$ evenly-spaced points in the first half of the cell to determine the final state (Figure 4.4).


Figure 4.4: $\theta_{H}(z)$ and $\theta_{V}(z)$, together with $\theta_{M}(z)$ (dashed), safety-zone (blue) and the critical test-points (red).

Once the field has been removed, we compare $\theta(z, t)$ with $\theta_{S}$ at each of the critical test points. If $\theta(z, t)>\theta_{S}$ at all points, we conclude that the cell is relaxing to the Vertical state. Similarly, if $\theta(z, t)<\theta_{S}$ at all points, we conclude that the cell is relaxing to the HAN state. Otherwise, we continue simulating. Note that we need to interpolate $\theta(z, t)$ to $z_{\text {crit }}$ when simulations employ a moving mesh algorithm. With sensible values for $z_{i}$, and since a final state is only assumed to have been reached when $\theta(z, t)$ is outside and on the same side of the safety-zone at each test point, there should not be sufficient torque in the bulk of the cell to move the director at $z=0$ away from its nearest equilibrium solution. We therefore deem the conditions of our new method to be acceptable.


Figure 4.5: Three director relaxations superimposed on the final-state detector.

Three examples of our new method for detecting a final state are given in Figure 4.5. In plot (a) the director is contained within the safety-zone at $z=0$, so we cannot yet determine the final state. In plots (b) and (c), however, the director is not contained within $\theta_{S}$ at any of the three test points, so the final state may be determined.

We also check for the possibility that the director relaxes into a state which is different from either of the HAN and Vertical states. This can, in theory, occur when modelling with flow and high applied voltages. Flow-induced kickback has been seen to force the director above $\pi / 2$ near $z=d$. In some cases, it may be possible for this kickback to create enough torque to break the bistable surface anchoring and take $\theta(0, t)$ towards $\left(\pi+\alpha_{1}\right)$. To account for this, our algorithm for determining the final state must first ensure that $\theta(z, t) \in(0, \pi / 2) \forall z$ before the test is performed.

If, after a sufficiently long period of time ( 1.0 modelled seconds, say), the final relaxed state is still undetermined, we compute the optical transmission, $T(t)$, through the cell, using equation (2.61). This value is compared with the known optical transmissions for the relaxed HAN and Vertical states, $T_{H}$ and $T_{V}$, respectively. The final state is then taken to be that for which the transmission $T(t)$ is closest to.

```
Algorithm 8 Final State Detection
Require: time
Require: \(T_{I I}, T_{V}, z_{\text {crit }}\)
Require: \(\theta_{S} / /\) computed using equation (4.3)
Require: \(\theta(z, t) / /\) updated using the preferred solution method
    if time \(<1.0\) then
        if (Voltage \(=0.0\) ) and \((0<\theta(z, t)<\pi / 2)\) then
            if \(\theta\left(z_{\text {crit }_{i}}\right)>\max \left(\theta_{S}\left(z_{\text {crit }_{i}}\right)\right),(i=1 \ldots p)\) then
            Cell relaxes to the Vertical state
            else if \(\theta\left(z_{\text {crit }_{i}}\right)<\min \left(\theta_{S}\left(z_{\text {crit }_{i}}\right)\right),(i=1 \ldots p)\) then
                    Cell relaxes to the HAN state
            end if
        end if
    else
        compute: \(T(t)\)
        if \(\left|T(t)-T_{V}\right|<\left|T(t)-T_{H}\right|\) then
            Cell relaxes to a state which is optically similar to the Vertical state
        else
            Cell relaxes to a state which is optically similar to the HAN state
        end if
    end if
```

Using this improved method for determining the final state (Algorithm 8), we can reliably and efficiently determine the final state during the director's relaxation. If the final state differs from the initial state, we conclude that the cell has switched between states. Otherwise, we conclude that the cell has not switched between states.

Equipped with Algorithm 8, we may now test for points that induce switching between states.

### 4.2 Finding a point that switches

Given a grid of test points, our goal is to determine which points cause switching between states. In this section, we need only concern ourselves with finding a single point that switches. One simple way of doing this is to recursively test each point on the test grid until a switch is detected.


Figure 4.6: (I) A simple $4 \times 4 \tau V$ grid. (II) A monopolar pulse of $(\tau, V)=(0.6,10)$, which is used to test for switching at points (a), (b), (c) and (d) in Figure 4.6(I).

For simplicity, let us consider a grid of $4 \times 4$ points as in Figure 4.6(I). Because of the different $\tau$ and $V$ values at each point, let us also assume a different monopolar pulse is to be applied for each point. Figure 4.6(II) shows a typical monopolar pulse with dwell time $\tau=0.6$ milliseconds and magnitude $V=10$ volts, where the points (a)-(d) correspond to times $\tau=0.0,0.2,0.4,0.6$ milliseconds. A snapshot of the director profile at each of the key points is shown in Figure 4.7.

Typically, to test the points on the $\tau V$ plot $(\tau, V)=(\mathrm{a})(0,10),(\mathrm{b})(0.2,10),(\mathrm{c})(0.4,10)$ and $(d)(0.6,10)$ we would run four separate simulations. Each simulation would start in the relaxed state, we would apply $V=10$ volts for the prescribed dwell time, and relax until the final state is determined. However, this approach results in simulating from time $=(0.0,0.2) \mathrm{ms}$ three times and time $=(0.2,0.4) \mathrm{ms}$ twice. Since simulation time is expensive, we wish to avoid repeating computations whenever possible. Therefore, a far more efficient approach is to simulate once with $(\tau, V)=(0.6,10)$, storing the director profile and flow


Figure 4.7: Director profiles taken at the points corresponding to Figure 4.6(II), starting in the IIAN state.
velocity at the points (b), (c) and (d), and then simply simulating the three relaxations to determine the final state in each case.

In general we use a test grid of $20 \times 20$ points, so instead of simulating for 400 separate dwell times it is only necessary to simulate with 20 dwell times, leading to substantial savings in simulation time. This method also applies to bipolar pulses, in which case, for each voltage, we simulate for the longest dwell time and store the required data through to the change in polarity.

In the event that all test points are found to not switch, we conclude that no switching regions exist within the test region. Alternatively, if a test point is found to switch we can proceed to find the boundary of the switching region, using the method outlined in the following subsection.

### 4.3 Finding the boundary of a switching region

Given a point, $\left(\tau_{f}, V_{f}\right)$, that causes switching, we now wish to find the boundary of the switching region. To achieve this, we first need to locate some other point on the test grid that is known not to cause switching. Note that any point corresponding to $\tau=0$ cannot induce switching since, for any voltage $V$, a dwell time of exactly 0 milliseconds will have no effect on the cell. For simplicity, we therefore always select the point $(\tau, V)=(0, V)$ as our known 'no switch' point.

We may now use the Bisection Method (Algorithm 9) to estimate the location of the boundary of the switching region to within a user-prescribed tolerance $\tau_{\mathbf{s}}$. The Bisection Method

```
Algorithm 9 The Bisection Method
Require: \(\tau_{\text {(low) }}, \tau_{\text {(high) }}, \tau_{\mathrm{s}}, V\)
Ensure: \(\delta=\tau_{(\text {high })}-\tau_{\text {(low) }}\)
    while \(\delta>\tau_{\mathrm{s}}\) do
        \(\tau=\left(\tau_{(\text {low })}+\tau_{(\text {high })}\right) / 2.0\)
        compute: switch \(\left.\right|_{(\tau, V)} / /\) returncd from Algorithm 8
        if switch=TRUE then
            \(\tau_{(\text {high })}=\tau\)
        else
            \(\tau_{\text {(low) }}=\tau\)
        end if
        \(\delta=\tau_{\text {(high) }}-\tau_{\text {(low) }}\)
    end while
```

takes two points, $\left(\tau_{\text {(low) }}, V\right)$ and ( $\left.\tau_{(\text {high })}, V\right)$, which lie outside and inside the switching region, respectively, and repeatedly divides the interval in half, each time selecting the subinterval containing the boundary. Once the subinterval is smaller than some prescribed tolerance, $\tau_{s}$, the algorithm concludes that it has found a sufficiently accurate estimate for the boundary. This is illustrated in Figure 4.8, which takes the initial input parameters $\tau_{\text {(low) }}=0.0 \mathrm{~ms}$, $\tau_{\text {(high) }}=\tau_{f} \mathrm{~ms}, V=V_{f}$ volts and $\tau_{s}=0.05 \mathrm{~ms}$. Note that only six itcrations are required to obtain the required accuracy for our estimate.

Now that we have located the boundary of the switching region, we may attempt to trace around that region. However, in order for our tracing algorithm to avoid crossing over a previously traced route, we must be able to determine whether or not two line segments intersect with each other. When attempting to find additional switching regions, we must also be able to determine whether or not a point lies inside a polygon. We therefore proceed by introducing some background theory in line segment intersection.

### 4.4 Line segment intersection

Line segment intersection is a problem that is frequently solved in computer graphics applications, primarily in routines involving polygon clipping. The work presented here is based on the Usenet discussion referenced in [43].

Consider four points, $\mathbf{p}_{1}, \mathbf{p}_{2}, \mathbf{q}_{1}$ and $\mathbf{q}_{2}$, in a plane, which describe the line segments $\mathbf{p}_{1} \rightarrow \mathbf{p}_{\mathbf{2}}$ and $\mathbf{q}_{1} \rightarrow \mathbf{q}_{2}$, as in Figure 4.9(a). For the two line segments to fully intersect, the following


Figure 4.8: Estimating the boundary of a switching region to within $\tau_{\mathbf{s}}=0.05 \mathrm{~ms}$ using the Bisection Method. Blue indicates switching from the IIAN state to the Vertical state, whilst red indicates no switching.
(a)

(b)


Figure 4.9: Four points describing (a) two line segments, and (b) three vectors.
conditions must be satisfied:

1. the points $\mathbf{p}_{1}$ and $\mathbf{p}_{2}$ must lie on opposite sides of the line $\mathbf{q}_{1} \rightarrow \mathbf{q}_{2}$,
2. the points $\mathbf{q}_{1}$ and $\mathbf{q}_{2}$ must lie on opposite sides of the line $\mathbf{p}_{1} \rightarrow \mathbf{p}_{\mathbf{2}}$.

Now, using the same four points, let us define the projections

$$
\begin{align*}
& \vec{Q}=\left(q_{2_{x}}-q_{1_{z}}, q_{2_{y}}-q_{1_{y}}, 0\right) \\
& \vec{R}=\left(p_{1_{x}}-q_{1_{z}}, p_{1_{y}}-q_{1_{y}}, 0\right)  \tag{4.4}\\
& \vec{S}=\left(p_{2_{z}}-q_{1_{z}}, p_{2_{y}}-q_{1_{y}}, 0\right)
\end{align*}
$$

illustrated in Figure 4.9(b). Taking the cross-product of $\vec{Q}$ and $\vec{R}$ gives

$$
\begin{equation*}
\vec{Q} \times \vec{R}=|Q||R| \sin (\theta) \mathbf{k}=\Lambda \mathbf{k} \tag{4.5}
\end{equation*}
$$

where $\theta$ is the angle made by $\mathbf{q}_{2}, \mathbf{q}_{1}$ and $p_{1}$, as illustrated in Figure $4.9(\mathrm{~b})$, and

$$
\begin{equation*}
A=\left(\left(q_{2_{x}}-q_{1_{x}}\right)\left(p_{1_{y}}-q_{1_{y}}\right)-\left(p_{1_{z}}-q_{1_{z}}\right)\left(q_{2_{y}}-q_{1_{y}}\right)\right) \tag{4.6}
\end{equation*}
$$

Therefore, by equation (4.5), $A$ and $\theta$ have the same sign, and so the $\operatorname{sign}$ of $\theta$ may be determined by computing $A$ using equation (4.6). Note that positive values of $\theta$ correspond to the points $\mathbf{q}_{1}, \mathbf{q}_{2}, \mathbf{p}_{1}$ defining a counter-clockwise triangle, whilst negative values of $\theta$ correspond to $\mathbf{q}_{1}, \mathbf{q}_{2}, \mathbf{p}_{1}$ defining a clockwise triangle. Equivalently, the sign of $\theta$ determines whether $\mathbf{p}_{1}$ lies on one side of the line $\mathbf{q}_{1} \rightarrow \mathbf{q}_{2}$ or the other.

Using the same approach, we may determine the sign of $\phi$, the angle made by $\mathbf{q}_{2}, \mathbf{q}_{1}$ and $\mathbf{p}_{2}$ (sce Figure $4.9(\mathrm{~b})$ ), and hence the position of $\mathbf{p}_{2}$ relative to $\mathbf{q}_{1} \rightarrow \mathbf{q}_{2}$, by

$$
\begin{equation*}
\vec{Q} \times \vec{S}=|Q \| S| \sin (\phi) \mathbf{k}=B \mathbf{k} \tag{4.7}
\end{equation*}
$$

where

$$
\begin{equation*}
B=\left(\left(q_{2_{x}}-q_{1_{x}}\right)\left(p_{2_{y}}-q_{1_{y}}\right)-\left(p_{2_{x}}-q_{1_{z}}\right)\left(q_{2_{y}}-q_{1_{y}}\right)\right) . \tag{4.8}
\end{equation*}
$$

Note that the points $\mathbf{p}_{1}$ and $\mathbf{p}_{2}$ lie on opposite sides of the line $\mathbf{q}_{1} \rightarrow \mathbf{q}_{\mathbf{2}}$ if and only if $\theta$ and $\phi$ have opposite sign, i.e.

$$
\begin{equation*}
\Lambda \times B<0 \tag{4.9}
\end{equation*}
$$

therefore determining if condition 1 is satisfied. Analogously, condition 2 is satisfied when

$$
\begin{equation*}
C \times D<0, \tag{4.10}
\end{equation*}
$$

where

$$
\begin{align*}
& C=\left(\left(p_{2_{z}}-p_{1_{z}}\right)\left(q_{1_{y}}-p_{1_{y}}\right)-\left(q_{1_{z}}-p_{1_{z}}\right)\left(p_{2_{v}}-p_{1_{v}}\right)\right),  \tag{4.11}\\
& D=\left(\left(p_{2_{x}}-p_{1_{z}}\right)\left(q_{2_{y}}-p_{1_{y}}\right)-\left(q_{2_{z}}-p_{1_{z}}\right)\left(p_{2_{v}}-p_{1_{v}}\right)\right) . \tag{4.12}
\end{align*}
$$

Note that

$$
\begin{array}{lll}
A=0, & \text { and } & C \times D<0 \\
B=0, & \text { and } & C \times D<0 \\
C=0, & \text { and } & A \times B<0 \\
D=0, & \text { and } & A \times B<0, \tag{4.16}
\end{array}
$$

describe cases in which the lines are touching but not fully crossing.
(a)

(b)


Figure 4.10: Cases in which $\mathbf{p}_{1} \rightarrow \mathbf{p}_{\mathbf{2}}$ and $\mathbf{q}_{1} \rightarrow \mathbf{q}_{2}$ are collinear and (a) not overlapping, (b) overlapping.

The final case to consider is that which occurs when the line segments are collinear, i.e.

$$
\begin{equation*}
A=B=C=D=0 \tag{4.17}
\end{equation*}
$$

as in Figures 4.10(a) and 4.10(b). In this case, we distinguish between overlapping and nonoverlapping lines by checking if one of the end points of the first line lies between the end points of the second line. In other words, the lines overlap if any of the following conditions are met

$$
\begin{align*}
& \left(p_{1}-q_{1}\right) \cdot\left(p_{1}-q_{2}\right)<0  \tag{4.18}\\
& \left(p_{2}-q_{1}\right) \cdot\left(p_{2}-q_{2}\right)<0  \tag{4.19}\\
& \left(q_{1}-p_{1}\right) \cdot\left(q_{1}-p_{2}\right)<0  \tag{4.20}\\
& \left(q_{2}-p_{1}\right) \cdot\left(q_{2}-p_{2}\right)<0 \tag{4.21}
\end{align*}
$$

### 4.5 Is a point inside a polygon?

Using the theory outlined in Section 4.4, it is possible to determine whether or not a given point, $\mathbf{p}_{2}$, is contained within a polygon defined by an array of points, $\left(x_{i}, y_{i}\right), i=1, \ldots, n$, where $n$ is the total number of points. Provided we choose a suitable point, $p_{1}$, which is known to lie outside of the polygon, we may proceed by generating $p_{1} \rightarrow p_{2}$, and counting the number of valid intersections this makes with each line segment that defines the polygon. If the number of intersections is odd, $\mathbf{p}_{2}$ lies inside the polygon, otherwise it lies outside.


Figure 4.11: Special cases in which the line segment $\mathbf{p}_{1} \rightarrow \mathbf{p}_{\mathbf{2}}$ : (a) lies on an edge of the polygon, (b) passes through a point on the polygon, (c) \& (d) overlaps a line seginent of the polygon.

When counting the number of intersections, there are several special cases to consider (Figure 4.11). In the case of Figure 4.11(a), we use the convention of not counting an intersection when $\mathbf{p}_{2}$ lies on an edge of the polygon. To deal with cases (b)-(d), it is necessary to compute the $A, B, C$ and $D$ (using equations (4.6), (4.8), (4.11) and (4.12)) corresponding to each line segment of the polygon prior to counting intersections, as in Algorithm 10.

Algorithm 11 details our method for counting the total number of intersections the line

```
Algorithm 10 Computing the orientational parameters.
Require: \(p_{1_{x}}, p_{1_{\nu}}, p_{2_{x}}, p_{2_{\nu}}\)
Require: \(\left(x_{i}, y_{i}\right), i=1, \ldots, n\)
    for \(i=1, \ldots,(n-1)\) do
        \(q_{1_{z}}=x_{i}\)
        \(q_{1}=y_{i}\)
        \(q_{2_{x}}=x_{i+1}\)
        \(q_{2_{y}}=y_{i+1}\)
        compute: \(A_{i}=\left(q_{2_{x}}-q_{1_{x}}\right)\left(p_{1_{y}}-q_{1_{y}}\right)-\left(p_{1_{z}}-q_{1_{z}}\right)\left(q_{2_{y}}-q_{1_{y}}\right)\)
        compute: \(B_{i}=\left(q_{2_{z}}-q_{1_{s}}\right)\left(p_{2_{v}}-q_{1_{v}}\right)-\left(p_{2_{z}}-q_{1_{z}}\right)\left(q_{2_{v}}-q_{1_{v}}\right)\)
        compute: \(C_{i}=\left(p_{2_{z}}-p_{1_{z}}\right)\left(q_{1_{y}}-p_{1_{y}}\right)-\left(q_{1_{z}}-p_{1_{z}}\right)\left(p_{2_{y}}-p_{1_{v}}\right)\)
        compute: \(D_{i}=\left(p_{2_{z}}-p_{1_{s}}\right)\left(q_{2_{y}}-p_{1_{v}}\right)-\left(q_{2_{z}}-p_{1_{z}}\right)\left(p_{2_{v}}-p_{1_{v}}\right)\)
    end for
```

segment $\mathbf{p}_{\mathbf{1}} \rightarrow \mathbf{p}_{\mathbf{2}}$ makes with the polygon.
Lines 1-22 explain how to handle instances in which the two line segments are collinear. In such cases, we only conclude that an intersection has occurred if the last previous non-zero $C_{i}$ is different in sign from the next non-zero $D_{i}$ and the points which are collinear with $\mathrm{p}_{1} \rightarrow \mathrm{p}_{2}$ lic between $\mathrm{p}_{1}$ and $\mathrm{p}_{2}$ (using equations (4.20)-(4.21)). This allows us to correctly distinguish between the cases shown in Figures 4.11(c) and 4.11(d).

Lines 23-27 are used to handle the case illustrated in Figure 4.11 (b) (i.e. $\quad C_{i}=0$ ). We check the two adjacent points in the $C$ array, $C_{i-1}$ and $C_{i+1}$. If they differ in sign we conclude that $p_{1} \rightarrow p_{2}$ intersects with the polygon at this point (as in Figure 4.11(b)), otherwise we conclude that the line touches the polygon at this point without intersecting. Upon reaching line 30 , each of the special cases have been ruled out and we procced to test for an intersection using the method outlined in Section 4.4.

Our final step for determining whether or not the point $\mathbf{p}_{\mathbf{2}}$ lies inside the polygon is to use the formula

$$
\begin{equation*}
\text { inside }=\text { intersection }(\bmod 2) \tag{4.22}
\end{equation*}
$$

where inside $=1$ implies that the point is inside the polygon, and inside $=0$ implies that the point is outside the polygon.

```
Algorithm 11 Counting the number of valid intersections.
Require: \(\left(A_{i}, B_{i}, C_{i}, D_{i}\right), i=1, \ldots,(n-1)\)
Require: \(p_{1_{z}}, p_{1_{y}}, p_{2_{z}}, p_{2_{v}}\)
Require: \(\left(x_{i}, y_{i}\right), i=1, \ldots, n\)
Require: intersection \(=0\), starter \(=1\)
    CONTINUE
    for \(i=\) starter \(, \ldots,(n-1)\) do
        if \(\left(\left|A_{i}\right|<\epsilon\right)\) AND \(\left(\left|B_{i}\right|<\epsilon\right)\) AND \(\left(\left|C_{i}\right|<\epsilon\right)\) and \(\left(\left|D_{i}\right|<\epsilon\right)\) then
            for \(j=1, \ldots, i-1\) do
                if \(\left|C_{i-j}\right|>\epsilon\) then
                    GO To 9
                end if
            end for
                continue
                for \(k=1, \ldots,(n-1-i)\) do
                if \(\left|D_{i+k}\right|>\epsilon\) then
                    Go to 15
                end if
            end for
            continue
            if \(C_{i-j} \times D_{i+k}<0\) then
                if \(\left(\left(x_{i-j+1}-p_{1_{s}}\right)\left(x_{i-j+1}-p_{2_{z}}\right)+\left(y_{i-j+1}-p_{1_{y}}\right)\left(y_{i-j+1}-p_{2_{v}}\right)<0\right)\)
                AND \(\left(\left(x_{i+k}-p_{1_{x}}\right)\left(x_{i+k}-p_{2_{x}}\right)+\left(y_{i+k}-p_{1_{y}}\right)\left(y_{i+k}-p_{2_{y}}\right)<0\right)\) then
                    intersection \(=\) intersection +1
                end if
            end if
            starter \(=i+k+1\)
            GO TO 1
        else if \(\left|B_{i}\right|>\epsilon\) and \(A_{i+k} \times B_{i+k}<0\) then
            if \(\left|C_{i}\right|<\epsilon\) then
                if \(\left(\left|C_{i-1}\right|>\epsilon\right)\) and \(\left(\left|C_{i+1}\right|>\epsilon\right)\) and \(\left(C_{i-1} \times C_{i+1}<0\right)\) then
                    intersection \(=\) intersection +1
                end if
        else if \(\left|D_{i}\right|<\epsilon\) then
                CONTINUE
            else if \(C_{i} \times D_{i}<0\) then
                intersection \(=\) intersection +1
            end if
        end if
    end for
```


### 4.6 Tracing around a switching region

We now present a method for tracing the outline of a switching region using the parameters $\tau_{\mathbf{s}}$ and $V_{\mathbf{s}}$, the standard increments for $\tau$ and $V$, respectively. As a starting position, we
require a point that is known to switch and lies within $\tau_{s}$ of the switching region's boundary. From Section 4.3, the final test point labeled ( $\left.\tau_{(\text {high })}, V\right)$ is known to satisfy this condition. We therefore set the initial point to be $\left(\tau_{1}, V_{1}\right)=\left(\tau_{(\text {high })}, V\right)$.

### 4.6.1 The basic tracing algorithm

| next $_{i}$ | Direction | $\tau_{i+1}$ | $V_{i+1}$ |
| :---: | :---: | :--- | :--- |
| 0 | up | $\tau_{i}$ | $V_{i}+V_{\text {tol }}$ |
| 1 | left | $\tau_{i}-\tau_{\text {tol }}$ | $V_{i}$ |
| 2 | down | $\tau_{i}$ | $V_{i}-V_{\text {tol }}$ |
| 3 | right | $\tau_{i}+\tau_{\text {tol }}$ | $V_{i}$ |



Figure 4.12: Defined values for next ${ }_{i}$ and its effect on $\left(\tau_{i}, V_{i}\right)$, giving $\left(\tau_{i+1}, V_{i+1}\right)$.

We employ what is known as a square tracer to estimate the boundary of the switching region. The tracer is 'square' in the sense that $\tau_{i}$ and $V_{i}$ are only altered one at a time between successive test points. Each direction (up, left, down and right) is denoted with a number from 0 to 3 , which is referenced when determining the direction, next ${ }_{i}$, in which the tracer moves from the $i$ th test point, as illustrated in Figure 4.12. The parameters $\tau_{\text {tol }}$ and $V_{\text {tol }}$ represent the $\tau$ and $V$ tolerances, respectively, by which each parameter may be incremented.

```
Algorithm 12 Determining the \((i+1)\) th test point.
Require: clockwise, next \(_{i-1}\), switch \(_{i}\), repeat
    if switch \(_{i}=\) clockwise then \(^{2}\)
        next \(_{i}=\) next \(_{\text {i-1 }}-\) repeat \(+1(\bmod 4)\)
    else
        next \(_{i}=\) next \(_{i-1}+\) repeat \(-1(\bmod 4)\)
    end if
```

At the $i$ th test point an appropriate next ${ }_{i}$ is chosen using Algorithm 12, where 'repeat' initially takes the value zero, switch ${ }_{i}$ takes the value 'TRUE' or 'FALSE' depending on whether or not the point ( $\tau_{i}, V_{i}$ ) induces switching between states, and 'clockwise' takes the value 'TRUE' or 'FALSE' depending on whether or not we are tracing in a clockwise direction. We use the convention of tracing around switching regions in an anti-clockwise direction when testing positive voltages, and tracing in a clockwise direction when testing negative voltages.

Notice that, upon entering this routine, $\left(\tau_{1}, V_{1}\right)=\left(\tau_{(\text {high })}, V\right)$ with $\left(\tau_{0}, V_{0}\right)=\left(\tau_{\text {(low) }}, V\right)$, implying that next ${ }_{0}=3$. It is also known that switch ${ }_{1}=1$, and so next ${ }_{1}$ will take the value 0 for negative voltages and 2 for positive voltages, as shown in Figure 4.13(a).


Figure 4.13: An example of the basic tracing algorithm. Blue points indicate a switch from the HAN state to the Vertical state, whilst red points indicate no switching.

Having deduced next ${ }_{i},\left(\tau_{i+1}, V_{i+1}\right)$ may be computed using Figure 4.12. If $\left(\tau_{i+1}, V_{i+1}\right) \notin$ $\left.\left(\tau_{k}, V_{k}\right)\right|_{k=1, \ldots i}$ and the line segment generated by $\left(\tau_{i}, V_{i}\right) \rightarrow\left(\tau_{i+1}, V_{i+1}\right)$ does not intersect with or overlap any of the line segments generated by $\left.\left(\tau_{k}, V_{k}\right)\right|_{k=1, \ldots(i-1)}$ (using a slightly modified version of Algorithm 11), we accept $\left(\tau_{i+1}, V_{i+1}\right)$ as our next test point and continue simulating (as in Figure 4.13(b)). Otherwise, $\left(\tau_{i+1}, V_{i+1}\right)$ is rejected, repeat=repeat +1 and Algorithm 12 is repeated, as in Figure 4.13(c).

### 4.6.2 Adaptive increments

Many $\tau V$ plots are of the form shown in Figure 4.14(a), with asymptotic properties as either $\tau$ or $V$ become very large. Therefore, it is desirable to vary $\tau_{\text {tol }}$ and $V_{\text {tol }}$, the increments in $\tau$ and $V$, respectively, as appropriate so as to decrease the total number of test points while maintaining a high degree of accuracy in our approximation of the switching region.

Our approach for adapting $\tau_{\text {tol }}$ is outlined in Algorithm 13, where $\tau_{\text {tol }_{\mathrm{s}}}$ is a user-prescribed tolerance dictating the minimum allowed value for $\tau_{\text {tol }}$. The 'left' and 'right' parameters are used to store the number of successive moves to the left and right, respectively, along the $\tau$-axis, whilst 'tflick' counts the number of successive oscillations along the $\tau$-axis. After five successive movements in the same direction along the $\tau$-axis, we assume that $\tau_{\text {tol }}$ is currently too small, and so it is doubled. Likewise, upon the third oscillation along the


Figure 4.14: Examples of tracing with and without adaptive increments.
$\tau$-axis, we assume that $\tau_{\text {tol }}$ is currently too large, and so it is halved. As a final check, we ensure that $\tau_{\text {tol }}$ is not too small or too large. An analogous approach is used for adapting $V_{\text {tol }}$ -

We see from Figures 4.14 (b) and (c) that tracing without adaption incurs many more test points than tracing with adaption. In this particular example, the non-adapting algorithm tested a total of 1016 points, whilst the adaptive algorithm tested just 125 points. While it may appear that the adaptive method is significantly less accurate than the non-adaptive method, it should be noted that a smoothing filter (Section 4.8) is applied to the traced route in order to clean the final output.

### 4.6.3 Reaching a dead-end



Figure 4.15: An example of reaching dead-ends and reversing to the previous fork. Greyed-out lines indicate strands which result in reaching a dead-end

```
Algorithm 13 Adapting \(\tau_{\text {tol }}\).
Require: next \(_{i}, \tau_{\text {tol }}, \tau_{\text {tol }_{s}}\)
Require: tflick,lcft,right
    1: if next \(_{i}=1\) then
        left \(=\) left +1
        right \(=0\)
    else if next \({ }_{i}=3\) then
        right \(=\) right +1
        left \(=0\)
    end if
    if next \(_{i}=1\) or next \({ }_{i}=3\) then
        if left \(=1\) or right \(=1\) then
            tflick=tflick+1
        else
            tflick \(=0\)
        end if
        if tflick= 2 then
            tflick=0
            \(\tau_{\mathrm{tol}}=\tau_{\mathrm{tol}} / 2\)
        end if
    end if
    if left \(=5\) AND next \({ }_{i}=1\) then
        left \(=2\)
        \(\tau_{\text {tol }}=2 \times \tau_{\text {tol }}\)
    else if right \(=5\) AND next \({ }_{i}=3\) then
        right \(=2\)
        \(\tau_{\text {tol }}=2 \times \tau_{\text {tol }}\)
    end if
    if \(\tau_{\text {tol }}<\tau_{\text {tol }_{s}}\) then
        \(\tau_{\text {tol }}=\tau_{\text {tol }}\)
    else if \(\tau_{\text {tol }}>20 \times \tau_{\text {tol }}\) then
        \(\tau_{\text {tol }}=20 \times \tau_{\text {tol }_{s}}\)
    end if
```

There are, however, some instances in which this algorithm folds back on itself, reaching a dead-end from which it cannot navigate through, as in Figure 4.15(a). Note that when these cases occur Algorithm 12 will eventually increment the 'repeat' parameter to 3.

Having detected a dead-end, we mark this point as omitted and note that it is also a deadend. We then cycle back through the points that have already been tested until we find the last point which is not marked as omitted, say ( $\tau, V_{l}$ ). From here, Algorithm 12 is re-run and, if another suitable test-point is chosen, $\left(\tau_{l}, V_{l}\right)$ is marked as a point at which the tracing
algorithm forked. If ( $\tau_{l}, V_{l}$ ) does not produce another route for tracing, we mark this point as omitted (note that ( $\tau_{l}, V_{l}$ ) is not marked as a dead-end, since we are already in the process of reversing).

In the case of Figure $4.15(\mathrm{~b})$, we only need to reverse a single point before an alternative route is found. However, two further dead-ends are found before the algorithm is able to successfully navigate this particular part of the boundary. Notice by Figures 4.15(c) and (d) that in this case we need to reverse through seven points until a valid continuation point is found.

Also, note that even once a point is omitted, both the point and its corresponding line segments are still stored in memory so as to avoid repetition. So, when determining a valid next ${ }_{i}$, we check for repetition using both the global route and any dead strands that have been encountered. It is also important to note that such cases of unnecessary testing are relatively uncommon, and that only a very small fraction of the total computation time is wasted traversing paths that don't lead anywhere.

### 4.6.4 Reaching the edge of the test grid

Whilst the routines presented so far allow us to trace the boundary of a switching region, we need to ensure that simulations do not exceed the limits of the pre-defined test grid. In the explanation that follows, we only consider having reached $\tau_{\text {max }}$ since the same reasoning may be applied upon reaching $V_{\text {max }}$.

Given a $\tau_{\text {max }}$, we need to check that the $\tau_{i+1}$ proposed by Figure 4.12 is less than or equal to $\tau_{\max }$. Should $\tau_{i+1}$ exceed $\tau_{\max }$, we set $\tau_{i+1}=\tau_{\max }$ and continue tracing. If the boundary of the switching region is detected at an edge of the test area (that is, if two successive points are tested at, say, $\tau_{\max }$, with one causing a switch between states and the other not), as in Figure 4.16, another routine is used to trace along that particular edge. In the event that $\tau_{\text {max }}$ is reached, $V$ is incremented by $2 \times V_{\text {tol }_{s}}$ and the resulting point is tested. This process is repeated until either $\pm V_{\max }$ is reached or a point is found to lie outside of the switching region in which case we continue tracing as before.

Figure 4.17 illustrates this routine in its entirety. Note that the standard increments along the $\tau$ axis are not reset between Figures 4.17(a) and (c), since the required increments on


Figure 4.16: The three cases we consider for reaching the edge of the test area at $\tau_{\mathrm{max}}=2.5 \mathrm{~ms}$ : (a) crossing the boundary of the switching region from the outside, (b) crossing from the inside, (c) the special case. The yellow region represents the 'true' switching region.


Figure 4.17: An example of tracing a region which is clipped by the edge of the test region.
directly opposing sides of any given region are generally relatively similar.

### 4.6.5 Closing a region

The final part of our standard algorithm deals with closing the switching regions that are traced. To manage this, we compare the position of the current test point, $\left(\tau_{i}, V_{i}\right)$, with $\left(\tau_{1}, V_{1}\right)$ and $\left(\tau_{2}, V_{2}\right)$. If the current test point is suitably close to either of the first two test points, we conclude that the region has been successfully traced and the algorithm stops.


Figure 4.18: An example of checking if the switching region may be closed. The purple area shows the 'closing space' in which either of the first two points must lie before we allow the region to be closed. In plots (a)-(c) $\left(\tau_{i}, V_{i}\right)$ is not sufficiently close to either of the first two points, whilst in plot (d) ( $\left.\tau_{i}, V_{i}\right)$ is sufficiently close and the region is closed.

Our test for closing a region uses the current values of $\tau_{\text {tol }}$ and $V_{\text {tol }}$ to define an appropriate 'closing space', as in Figure 4.18. Note that the closing space will always be rectangular, but that $\left(\tau_{i}, V_{i}\right)$ will not necessarily be located in its centre since $\tau_{\text {tol }}$ and $V_{\text {tol }}$ may double or halve (by Section 4.6.2) depending on the direction taken, as shown in Figures 4.18(a)-(b).

### 4.7 Post processing

Having traced around a switching region, we now apply some post processing filters to obtain a smoothed estimate for the boundary of that region. Note that any points which have been marked as 'omitted' are not included in this part.

A rough estimate of the boundary is given by taking the midpoint of each pair of consecutive test points which were found to lie on opposite sides of the boundary, as in Figure 4.19(b). A five point moving average is then applied to the resulting points to smooth our estimate for the boundary of the switching region, as in Figure 4.19(c).

Note that even though the route taken may appear to poorly resolve the boundary of the switching region (Figure 4.20 (a)) our boundary estimate is still somewhat smooth (Figure $4.20(\mathrm{~b}))$ since chains of successive test points that lie on the same side of the boundary are not taken into account. Applying a five-point moving average to this boundary estimate produces a smoother estimate for the boundary of the switching region, as in Figure 4.20(c).


Figure 4.19: An example of extracting a smoothed estimate for the boundary of a switching region using the traced path. Plot (a) shows the path taken, (b) overlays our estimate for the boundary (green solid line), and (c) shows the smoothed output (blue solid line) over the boundary estimate (green dashed line).


Figure 4.20: Zoomed out version of Figure 4.19, showing the overall region that is traced.

The post-processing algorithm is detailed in Algorithm 14. Lines 1-9 filter out any omitted points, lines 10-17 estimate the boundary of the region, lines 18-21 are used to ensure that the first point of the smoothed output is equal to the last point of the smoothed output (essential for plotting a solid, closed region), and lines 22-26 perform the five-point moving average.

```
Algorithm 14 Estimating and smoothing the boundary of a switching region.
Require: \(n\)
Require: \(\tau_{i}, V_{i}\), switch \(_{i}\), omit \(_{i}, i=1, \ldots, n\).
    \(k=0\)
    for \(i=1, \ldots, n\) do
        if omit \(_{i}=0\) then
            \(k=k+1\)
            \(R \tau_{k}=\tau_{i}\)
        \(R V_{k}=V_{i}\)
        Rswitch \(_{k}=\) switch \(_{i}\)
        end if
    end for
    \(l=0\)
    for \(i=2, \ldots, k\) do
        if Rswitch \(_{i} \neq\) Rswitch \(_{i-1}\) then
            \(l=l+1\)
            \(M \tau_{l}=\left(R \tau_{i}+R \tau_{i-1}\right) / 2\)
            \(M V_{l}=\left(R V_{i}+R V_{i-1}\right) / 2\)
        end if
    end for
    for \(m=1, \ldots, 5\) do
        \(M \tau_{l+m}=M \tau_{m}\)
        \(M V_{l+m}=M V_{m}\)
    end for
    \(l=l+1\)
    for \(i=1, \ldots, l\) do
        \(S \tau_{i}=\left(M \tau_{i}+M \tau_{i+1}+M \tau_{i+2}+M \tau_{i+3}+M \tau_{i+4}\right) / 5\)
        \(S V_{i}=\left(M V_{i}+M V_{i+1}+M V_{i+2}+M V_{i+3}+M V_{i+4}\right) / 5\)
    end for
```


### 4.8 The observed error

For each region that is traced, an error is incurred in our approximation of that region. We show this pictorially by plotting a green polygon consisting of all of the points that do not cause switching between states and then superimposing a white, fully opaque polygon consisting of all of the points that were found to cause switching. The visible green area in Figure 4.21(a) essentially shows the error bounds, or the 'area of uncertainty', for our boundary estimate.

Note, by Figure 4.21(c), that the smoothed estimate of the boundary lies within the error bounds.


Figure 4.21: An example of the uncertainty of our traced boundary. The green area represents the space in which we know the true boundary lies, whilst the black dashed line shows the smoothed boundary computed in Section 4.7.

### 4.9 Finding additional regions

It is often the case that there exists more than one switching region of the same type in a $\tau V$ plot. In such cases, we need some way of tracing this additional switching region without retracing the first-found region. To achieve this, we need to modify the method outlined in Section 4.2 such that it cannot test points which lie in a previously traced region. We have already shown in Section 4.5 that it is possible to determine whether or not a point, $\mathbf{p}_{2}$, lies inside a polygon (or, equivalently, a closed region) provided we know of a point, $\mathbf{p}_{1}$, which lies outside the polygon.

For our purposes, we know that the point $(\tau, V)=(0,0)$ cannot lie inside any switching region and so, with $p_{1_{x}}=0, p_{1_{y}}=0$ and $\left(x_{i}, y_{i}\right)=\left(\tau_{i}^{j}, V_{i}^{j}\right), i=1, \ldots, n$, where $n$ is the number of points defining the $j$ th region, we may determine whether or not $p_{2}$ lies inside the switching region using Algorithm 11. Then the point $\mathbf{p}_{2}$ is only tested if it is found to lie outside of a previously traced region. As a safeguard, the polygon defining any previously traced region consists of only the points that were found not to switch when tracing that particular region.

Should an additional switching region be detected, we proceed as before by attempting to find the boundary of the switching region. However, note that with there being more than one switching region we can no longer assume that it is safe to set $\tau_{(\text {low })}=0$ milliseconds,
(a) Point In Polygon.

(b) Point Not In Polygon.

(c) New Region Detected.


Figure 4.22: An example of finding a second switching region.


Figure 4.23: An example of verifying all points on the test grid.
as the previously traced region may lie between $\left(0, V_{f}\right)$ and $\left(\tau_{f}, V_{f}\right)$. In order to determine whether or not this is the case, we use a slightly modified version of Algorithm 11 with $p_{1_{x}}=0, p_{1_{y}}=V_{f}$ and $\left(x_{i}, y_{i}\right)=\left(\tau_{i}^{j}, V_{i}^{j}\right), i=1, \ldots, n, j=1, \ldots, r$, where $n$ is the number of points defining the $j$ th region and $r$ is the number of previously traced regions. For each line segment of a previously traced region, the modified version of Algorithm 11 counts the number of intersections made by the line $\mathbf{p}_{1} \rightarrow \mathbf{p}_{2}$ and the region of interest and, if an intersection occurs, computes the coordinate of intersection. It then returns $\tau_{\text {(low) }}$ as either 0 (if no intersection occurs) or the highest $\tau$-value for which intersection occurs. Having chosen an appropriate $\tau_{\text {(low) }}$, we may now find the edge of the current switching region using Algorithm 9.


Figure 4.24: An example of finding a new $\tau_{\text {(low) }}$.


Figure 4.25: An example of the algorithm failing to pick up a thin section of a switching region, and detecting it as a second region. Here the yellow area represents the 'true' switching region and the green area shows the area used to represent a previously detected switching region.

Finally, having found the edge of the switching region, there exists the possibility that we may 'cross over' into a previously traced switching region at some point during our trace. This can happen either if $\tau_{\text {tol }}$ or $V_{\text {tol }}$ are too large, or if we are tracing along a particularly thin region, as in Figure 4.25. The solution here is to test if the line segment made by $\left(\tau_{i}, V_{i}\right) \rightarrow\left(\tau_{i+1}, V_{i+1}\right)$ crosses over into a previously traced region. If so, we accept $\left(\tau_{i+1}, V_{i+1}\right)$ and store it as a non-switching point so as to ensure that we don't retrace a previous region.


Figure 4.26: Procured $\tau V$ plot (testing for Vertical to HAN switching only) corresponding to Figure 4.25.

### 4.10 Limitations

There is one known limitation of the tracing routine. Since the algorithm is only concerned with the outline of a solid shape, it cannot detect any 'holes' within the shape.


Figure 4.27: The region of no switching is undetected in plot (d).

Whilst such cases are very rare (and, in the case of the no-flow model, so far undiscovered), they can exist when modelling flow effects. Figure 4.27 illustrates one such case in which, as the anchoring strength $W_{0}$ is decreased, the central island of no switching shrinks to such an extent that it eventually goes undetected. Brute-force simulations (Figure 4.28) indicate that there does exist an island within the switching region which is completely isolated from the traced boundary.


Figure 4.28: The brute-force method shows that there exists an island of no switching within the region.

There are, however, ways of minimising the effects of this limitation and preserving the essential features of the $\tau V$ plot. Since we primarily generate multiple $\tau V$ plots, simultaneously, to study the effects of a model parameter, we greatly increase our chances of noticing any unusual behavior. In the example given by Figure 4.27, we notice that the white island suddenly disappears between plots (c) and (d), indicating that further investigation is required. Another suitable way of safeguarding against such problems is to perform a single brute-force scan of the test area every, say, one hundred traces.

Finally, we could adapt the tracing algorithm such that it can detect any gaps within a region automatically. This modification would require testing every point on the test grid, defined in Section 4.2, regardless of whether or not a switching region is detected. Instead of discarding points which are found to lie inside a previously traced region, as in Section 4.9, we check to make sure that they do in fact switch. Any points that are found not to switch, but which lie inside the boundary of a previous region, must be part of an island within the switching region. An 'inverse-trace' would then be performed to trace the outline of the detected island.

### 4.11 Summary

In this section, we have presented a tracing algorithm that is both efficient and reasonably accurate for generating $\tau V$-plots. Its key limitation has been acknowledged and ideas for future development have been raised. One important point is that the code-base is completely modular, and so it is relatively straight-forward to replace any routine with a suitable alternative. For example, should we find a better method for adapting $\tau_{\text {tol }}$ and $V_{\text {tol }}$, we may use
that in place of the method presented here with relative case. This also applies to the fundamental routines that are used to solve the governing equations at each test point. Therefore, our tracing algorithm is not limited to working with $\Omega$ one dimensional model, and may extend to two or even three-dimensional models.

In the next section, we will use the tracing algorithm presented in this section with the numerical methods outlined in Section 3 to investigate the switching characteristics of our liquid crystal cell. We also provide a variety of parameter studics to understand how each parameter in our model affects these switching characteristics.

## 5 Results - no flow model

Throughout this section we investigate the effect each parameter has on the switching characteristics of a bistable cell. To ensure that we are always comparing like-with-like, our base parameter set (Table 2.1) is enclosed in each parameter sweep.


Figure 5.1: A positive bipolar pulse of length $\tau$ and magnitude $V_{\text {mag }}$.

For each simulation we apply a bipolar pulse (Figure 5.1) across the cell, where the polarity of the applied voltage is taken to be the sign of the second half of the pulse. Note that, for our simulations, both parts of the bipolar pulse are equal in duration and we define the pulse-length as the length of time for which each polarity is applied. We also label four reference points in the bipolar pulse to aid in the explanation of each individual simulation, namely: the initial relaxed state (A), the point at which the polarity is changed (B), the point at which the field is removed (C) and the final relaxed state (D).

### 5.1 Parameter fundamentals

Before we begin simulating, it is important to give a brief explanation of the key points regarding how we expect each parameter to affect the properties of a liquid crystal.

To understand the expected behaviour at the bistable surface ( $z=0$ ), we consider equation (2.52). Note that the flexoelectric term, $E_{13}$, causes $\theta(0)$ to tend towards 0 and $\pi / 2$ when negative and positive voltages are applied, respectively. We also see that increasing the bistable surface anchoring strength, $W_{0}$, increases the bistable energy barrier, thereby
holding $\theta(0)$ more firmly in place at its nearest orientation of either $\alpha_{1}$ or $\alpha_{2}$.

To understand the expected behaviour at the monostable surface ( $z=d$ ), we consider equation (2.54). The flexoclectric term, $E_{13}$, causes $\theta(d)$ to tend towards $\pi / 2$ and 0 when negative and positive voltages are applied, respectively. Note that this is exactly the opposite of the behaviour we expect to see at the bistable surface. We also sce that increasing the monostable surface anchoring strength, $W_{d}$, increases the monostable energy barrier, therely holding $\theta(d)$ more firmly in place at its preferred orientation of $\pi / 2$.

To understand the expected behaviour in the bulk of the cell ( $0<z<d$ ), we consider equation (2.40). We see that a negative dielectric material (i.e. $\Delta \epsilon<0$ ) causes the director in the bulk of the cell to tend towards $\theta=0$ when a voltage is applied, whilst a positive dielectric material (i.e. $\Delta \epsilon>0$ ) causes the director in the bulk of the cell to tend towards $\theta=\pi / 2$ when a voltage is applied. Also note that the elastic constants, $K_{1}$ and $K_{3}$, interact with the curvature of the director profile, $\frac{\partial^{2} \theta}{\partial z^{2}}$, thereby increasing the linearity of $\theta$.

### 5.2 The base parameter set

Figure 5.2(a) shows the $\tau V$ plot generated from the parameter set in Table 2.1. Red regions indicate switching from the Vertical state to the HAN state whilst blue regions indicate switching from the HAN state to the Vertical state. Note that semi-transparent colours are used, as this allows us to see if the two regions overlap. Notice that Figure 5.2(a) can be represented in three dimensions as a single slice, as illustrated in Figure 5.2(b). This allows us to observe any changes in each switching region of the base $\tau V$ plot as the parameter of interest is varied.

In this case, we observe minimal overlap and white space between adjacent regions. Since our tracing algorithm provides only an approximation for each region, these discrepancies are likely to be due to numerical error.

To further investigate the two switching regions, we test some individual $(\tau, V)$ points within each region and examine how the director profile throughout the cell evolves with time. The main ( $\tau, V$ ) points of interest are indicated in Figure 5.3, and the key director profiles for each point are shown in Figure 5.4. Comparing Figure 5.4 (I) with Figure 5.4 (II), which describe HAN to Vertical and Vertical to HAN switching respectively, we see that by the


Figure 5.2: $\tau V$ plot generated using the base parameter set (Table 2.1), where red indicates Vertical to HAN switching and blue indicates HAN to Vertical switching. Plot (a) shows a standard two-dimensional $\tau V$ plot, and plot (b) shows the same $\tau V$ plot from a three-dimensional isometric viewpoint.


Figure 5.3: A re-plot of the positive voltages in Figure 5.2, where the four $(\tau, V)$ points of interest are denoted by the black dots.
change in polarity (plot (B)) each of director profiles have reached a state which depends on the point $(\tau, V)$ and not the initial state. It therefore follows that plots (C) and (D) are the same in each case. Note that we expect similar behavior for any $(\tau, V)$ that induces switching between one state and the other. This explains why there is little-to-no overlapping between the switching regions in Figure 5.2.


Figure 5.4: Four director profiles taken at each of the key points labeled in Figure 5.1, with $\tau=2.5 \mathrm{~ms}$ and $V=5$ (black), 20 (red), 50 (green) and 80 (blue) volts, corresponding to the lower Vertical to HAN, lower HAN to Vertical, upper Vertical to HAN and upper HAN to Vertical switching regions, respectively, as shown in Figure 5.3. Figure (I) starts from an initial HAN state, whilst Figure (II) starts from an initial Vertical state.

Let us consider the evolution of each of the director profiles in Figure 5.4 (I), starting with the 5 volts case (solid black line). By plot (B), the leading edge of the pulse has brought the bulk of the cell towards $\theta=0$. Changing the polarity of the applied voltage generates a small flexoelectric torque which is not high enough to break the anchoring at the bistable surface, as seen in plot (C). Therefore, upon removing the voltage, the cell relaxes to the HAN state shown in plot (D). For the 20 volts case (red), the higher voltage results in a 'flatter' profile in plot (B). More significantly, when the polarity is changed the higher voltage generates enough flexoelectric torque to break the bistable surface anchoring and force $\theta(0)$ close to $\alpha_{2}$, as seen in plot (C). Once the voltage has been removed, the elastic torque in the bulk of the cell is not high enough to break the bistable surface anchoring and the cell relaxes to the Vertical state shown in plot (D). The 50 volts case (green) behaves in a similar fashion to the 20 volts case for plots (A) and (B). However, note from plot (C) that the higher voltage results in a larger portion of the bulk of the cell remaining close to zero near the bistable surface. In this case, there is sufficient elastic torque in the bulk of the cell to break the bistable surface anchoring and the cell relaxes to the HAN state shown in plot (D).


Figure 5.5: Forcing effects at the bistable surface upon removing the applied voltage, for $V=50$ volts (black line) and $V=80$ volts (red line). Plot (a) shows the field effect only, whilst plot (b) shows the combined effect (comprising elastic, field and anchoring). The yellow shaded area highlights the range for which the field effect with 50 volts is 'more negative' than the field effect with 80 volts, and the dashed line signifies zero on the $y$-axis.

Careful consideration is required in order to explain the 80 volts case (blue) since, intuitively, we would expect this to relax into a HAN state for the same reasons as the 50 volts case. Figure 5.5(a) shows the field effect at the bistable surface for the 50 volts case and the 80 volts case. Throughout the highlighted interval, the bistable surface experiences a greater pull towards zero in the 50 volts case than it does in the 80 volts case. From Figure 5.5(b), note that by the end of the highlighted interval the overall torque on the bistable surface is positive in the 80 volts case but still negative in the 50 volts case.

However, by Figure 5.4 (I) (C), note that the main difference between the 50 and 80 volt cases is at the monostable surface, where the higher voltage forces $\theta$ very close to zero. To understand why a difference in director structures at the monostable surface can influence the electric field, and therefore the director (by Figure 5.5), at the bistable surface, we examine the field equation, namely

$$
\begin{equation*}
E=\frac{D_{3}-E_{13} \sin (2 \theta) \frac{\partial \theta}{\partial z}}{\epsilon_{0}\left(\epsilon_{\perp}+\Delta \epsilon \sin ^{2} \theta\right)}, \quad \text { where } \quad D_{3}=\frac{-\epsilon_{0} V+E_{13} \int_{0}^{d} \frac{\sin (2 \theta) \frac{\partial \theta}{\epsilon_{\perp}+\Delta \epsilon \sin ^{2} \theta} d z}{\int_{0}^{d} \frac{1}{\epsilon_{\perp}+\Delta \epsilon \sin ^{2} \theta} d z} . . . . ~}{\text {. }} \tag{5.1}
\end{equation*}
$$



Figure 5.6: Relaxations after the voltage is removed, corresponding to Figure 5.4(I), with $\tau=2.5 \mathrm{~ms}$ and $V=50$ (green) and 80 (blue) volts.

Now, by Figure 5.4 (I) (C), throughout the cell we have

$$
\begin{aligned}
\left.\frac{d \theta}{d z}\right|_{V=80} & <\left.\frac{d \theta}{d z}\right|_{V=50} \\
\left.\Rightarrow \int_{0}^{d} \frac{\sin (2 \theta) \frac{\partial \theta}{\partial z}}{\epsilon_{\perp}+\Delta \epsilon \sin ^{2} \theta} d z\right|_{V=80} & <\left.\int_{0}^{d} \frac{\sin (2 \theta) \frac{\partial \theta}{\partial z}}{\epsilon_{\perp}+\Delta \epsilon \sin ^{2} \theta} d z\right|_{V=50}
\end{aligned}
$$

for small $\Delta \epsilon$. Therefore, by equation (5.1),

$$
\begin{aligned}
\left.D_{3}\right|_{V=80} & <\left.D_{3}\right|_{V=50}, \\
\left.\Rightarrow E\right|_{V=80} & <\left.E\right|_{V=50},
\end{aligned}
$$

and so, by equation (2.52), we have

$$
\begin{equation*}
\left.\frac{\partial \theta}{\partial t}\right|_{V=80}>\left.\frac{\partial \theta}{\partial t}\right|_{V=50}, \tag{5.2}
\end{equation*}
$$

at the bistable surface. Since $\frac{\partial \theta}{\partial t}$ is negative in both the 50 volts case and the 80 volts case, we conclude from equation (5.2) that, during initial relaxation, $\theta$ changes faster in the 50 volts case, as observed in Figure 5.6. So non-local director structures can affect the director at the bistable surface. For this reason, we refer to the upper HAN to Vertical switching region in Figure 5.2 as the non-local switching region.

It should be noted that two distinct HAN to Vertical switching regions are often observed experimentally [34]. Our ascertainment that one of these switching regions is due to nonlocal effects is the first only known explanation for such behavior.

Having explained each of the switching regions in the $\tau V$ plot corresponding to the base parameter set, we now move on to our parameter studies.

### 5.3 Varying the preferred HAN state



(b) Plot of solid 'slices'.

(c) HAN $\rightarrow$ Vertical only.

(d) Vertical $\rightarrow$ HAN only.

Figure 5.7: $\tau V$ plots for varying the preferred director orientation at the bistable surface for a relaxed HAN state, $\alpha_{1}$, from $0^{\circ}$ to $11.5^{\circ}$.

We begin by investigating $\alpha_{1}$, the preferred director orientation at the bistable surface relating to a relaxed HAN state.


Figure 5.8: Three key slices from Figure 5.7.


Figure 5.9: Three individual simulations with $\tau=2.5 \mathrm{~ms}, V=8$ volts and $\alpha_{1}=0^{\circ}$ (black), $5.5^{\circ}$ (red), $11.5^{\circ}$ (green).

Figure 5.7 shows that $\alpha_{1}$ does not have a significant effect on the switching characteristics. We see from Figure 5.8 that the local HAN to Vertical switching region expands slightly as $\alpha_{1}$ is increased. This increase in the HAN to Vertical switching region is explained in Figure 5.9. When $\alpha_{1}$ is very small, there is very little pre-tilt at the bistable surface and we require more flexoelectric torque in order to break the bistable surface anchoring by the time the voltage is removed. From Figure $5.9(\mathrm{C})$, notice that the voltage is too low to produce the flexoelectric torque needed in order to break the bistable surface anchoring in the $\alpha_{1}=0^{\circ}$ case, whilst with higher values of $\alpha_{1}$ the bistable surface anchoring is successfully broken and the cell relaxes to a Vertical state.

### 5.4 Varying the preferred Vertical state


(a) Semi-transparent block showing all switching regions.


Figure 5.10: $\tau V$ plots for varying the preferred director orientation at the bistable surface for a relaxed Vertical state, $\alpha_{2}$, from $78.5^{\circ}$ to $90^{\circ}$.

We now move on to our parameter study for $\alpha_{2}$, the preferred Vertical state at the bistable surface. Figure 5.10 shows that as $\alpha_{2}$ decreases, the HAN to Vertical switching region sees significant growth. As before, some of the key slices are shown separately in Figure 5.11.


Figure 5.11: Three key slices from Figure 5.10.


Figure 5.12: Three individual simulations with $\tau=2.5 \mathrm{~ms}, V=80$ volts and $\alpha_{2}=78.5^{\circ}$ (green), $84^{\circ}$ (red), $90^{\circ}$ (black). Dotted lines represent the maxima of each bistable surface energy function.

Figure 5.12 shows the director profiles at each of the key points throughout three individual simulations. Note that the three director profiles are very similar in plots (A)-(C), but from plot (D) the $\alpha_{2}=90^{\circ}$ case relaxes into a HAN state whilst the other two cases relax into a Vertical state. We clearly need to examine the relaxation process in more detail in order to explain this behavior.

From Figure 5.12, notice that just before the voltage is removed the three director orientations are roughly the same at $z=0$, but the maximum associated with the bistable surface


Figure 5.13: Relaxations after the voltage is removed, corresponding to Figure 5.12 with $\tau=2.5 \mathrm{~ms}$, $V=80$ volts and $\alpha_{2}=78.5^{\circ}$ (green), $84^{\circ}$ (red), $90^{\circ}$ (black). Dotted lines represent the maxima of each bistable surface energy function.
energy function, namely $\left(\alpha_{1}+\alpha_{2}\right) / 2$, is different in each case. Therefore, at the instant the voltage is removed, higher values of $\alpha_{2}$ require less elastic torque to break the bistable surface anchoring, since the director orientation is closer to its associated energy barrier.

This becomes clearer when allied with Figure 5.13, which shows the initial relaxation process of each director. After 5.3 ms (plot (c)) the $\alpha=90^{\circ}$ case has crossed over its energy barrier to the 'HAN' side whilst the other two directors remain on the 'Vertical' side. By plot (d), the $\alpha=90^{\circ}$ case is fast relaxing to a HAN state whilst the other cases are relaxing into a Vertical state due to the decreased elastic torque in the bulk. So to conclude, the location of the bistable surface energy barrier decreases with $\alpha_{2}$ and more elastic torque is needed to break the anchoring at the bistable surface.

### 5.5 Varying the flexoelectric constant


(a) Semi-transparent block showing all switching regions.


Figure 5.14: $\tau V$ plots for varying the flexoelectric constant, $E_{13}$, from $1.0 \times 10^{-11} \mathrm{C} / \mathrm{m}$ to $1.0 \times 10^{-10} \mathrm{C} / \mathrm{m}$.


Figure 5.15: Three key slices from Figure 5.14.

We now move on to our parameter study for $E_{13}$, the flexoelectric constant. We expect higher values of $E_{13}$ to cause more torque at the bistable surface, allowing the cell to switch from the HAN state to the Vertical state when lower voltages are applied. From Figure 5.15 we see that, whilst this appears to be the case for a lower range of $E_{13}$, very high values of $E_{13}$ can actually reduce the HAN to Vertical switching regions.


Figure 5.16: Three individual simulations with $\tau=2.5 \mathrm{~ms}, V=50$ volts and $E_{13}=2.1 \times 10^{-11} \mathrm{C} / \mathrm{m}$ (black), $5.0 \times 10^{-11} \mathrm{C} / \mathrm{m}$ (red), $1.0 \times 10^{-10} \mathrm{C} / \mathrm{m}$ (green).

Figure 5.16 shows the director profiles at each of the key points throughout three individual simulations. The first point to note from Figure 5.16(A) is that increasing $E_{13}$ accentuates the 'kink' in the relaxed HAN state. This is covered in Appendix B. From Figure 5.16(C), notice that with $E_{13}=2.1 \times 10^{-11} \mathrm{C} / \mathrm{m}$ there is not enough flexoelectric torque to break the anchoring at the bistable surface, whereas with the higher values of $E_{13}$ the bistable surface anchoring is broken and $\theta(0) \rightarrow \pi / 2$ by the time the voltage is removed. This is in agreement
with our initial expectations. However, we see from Figure 5.16(D) that, upon removing the voltage, the simulation corresponding to $E_{13}=5.0 \times 10^{-11} \mathrm{C} / \mathrm{m}$ relaxes to a Vertical state whilst the simulation corresponding to the higher value of $E_{13}=1.0 \times 10^{-10} \mathrm{C} / \mathrm{m}$ relaxes to a HAN state. In order to explain this, we examine the short-term relaxation process of the cell as the voltage is removed.


Figure 5.17: Relaxations after the voltage is removed, corresponding to Figure 5.16 with $\tau=2.5 \mathrm{~ms}$, $V=50$ volts and $E_{13}=2.1 \times 10^{-11} \mathrm{C} / \mathrm{m}$ (black), $5.0 \times 10^{-11} \mathrm{C} / \mathrm{m}$ (red), $1.0 \times 10^{-10} \mathrm{C} / \mathrm{m}$ (green).

The relaxation process, immediately after the voltage is removed, of each director is shown in Figure 5.17. Notice that the cell appears to be relaxing on a much faster time-scale with $E_{13}=1.0 \times 10^{-10} \mathrm{C} / \mathrm{m}$ when compared to $E_{13}=5.0 \times 10^{-11} \mathrm{C} / \mathrm{m}$. Considering just the flexoelectric term, and neglecting $D_{3}$, the surface equation for $z=0$ reduces to

$$
\begin{equation*}
\frac{\partial \theta}{\partial t} \sim E_{13}^{2} \sin ^{2}(2 \theta) \frac{\partial \theta}{\partial z}, \tag{5.3}
\end{equation*}
$$

which implies that higher values of $E_{13}$ increase the effect of the gradient at the bistable surface. Now, from Figure 5.17, note that in each case the gradient at the bistable surface is negative throughout the initial relaxation. Therefore, with a higher value of $E_{13}$ we expect the director to favour lower values at the bistable surface during relaxation, as in Figure 5.17, which in turn favours a HAN state.

Using a similar technique with the bulk equation, it is possible to show that the flexoelectric term directly interacts with both the gradient and curvature in the cell. This leads to higher values of $E_{13}$ reducing the associated relaxation timescale in the bulk of the cell, as in Figure 5.17.
5.6 Varying the parallel dielectric constant ( $\epsilon_{\perp}$ fixed)

(a) Semi-transparent block showing all switching regions.


Figure 5.18: $\tau V$ plots for varying the parallel dielectric constant, $\epsilon_{\|}$, from 1 to 20 .


Figure 5.19: Three key slices from Figure 5.18, where (a) $\epsilon_{\|}=10(\Delta \epsilon=-5)$, (b) $\epsilon_{\|}=15(\Delta \epsilon=0)$, (c) $\epsilon_{\|}=20(\Delta \epsilon=5)$.

We now move on to our parameter study for $\epsilon_{\|}$, the parallel dielectric constant. Note that, by fixing $\epsilon_{\perp}$ at its default value ( $\epsilon_{\perp}=15$ ), we are essentially varying the dielectric anisotropy $\Delta \epsilon=\epsilon_{\|}-\epsilon_{\perp}$. Note from the governing equations that, without a flexoelectric term, we would expect the electric effects throughout the bulk of the cell to be symmetrical about $\Delta \epsilon=0$. However, $E_{13} \neq 0$ breaks this symmetry and gives us a situation in which positive and negative $\Delta \epsilon$ materials possess quite different switching characteristics, as shown in Figure 5.19.


Figure 5.20: Three individual simulations with $\tau=2.5 \mathrm{~ms}, V=50$ volts and $\epsilon_{\|}=10$ (black), 15 (red), 20 (green).

Figure 5.20 allows us to observe the dielectric effect at the director-level. The first point to note is that with $\epsilon_{\|}=15(\Delta \epsilon=0)$ the field effects are governed entirely by the flexoelectric term and the applied voltage does not directly interact with the bulk of the cell, meaning that the bistable surface experiences very little dielectric torque when the voltage is removed. This indicates that switching is mainly determined by the flexoelectric torque at the bistable
surface immediately before the voltage is removed. Recall, however, that the electric field in the bulk of the cell contains an asymmetry due to the flexoelectric term, and so we do not expect the switching regions to be exactly symmetrical about $V=0$, which is in agreement with the slight discrepancies observed in Figure 5.19(b).

We also see from Figure $5.20(\mathrm{~B})$ that, as expected, when a field is applied the preferred director orientation throughout the bulk of the cell is parallel to the field for the positive material and perpendicular to the field for the negative material, regardless of the polarity of the applied voltage. This implies that HAN to Vertical switching should be dominant for a positive material whilst Vertical to HAN switching should be dominant for a negative material, which is in agreement with Figure 5.19.

Finally, as $\epsilon_{\|}$is decreased the dielectric anisotropy decreases and the material becomes more negative. This increases the field effects throughout the bulk of the cell, thereby increasing the elastic torque at each boundary. We would therefore expect lower values of $\epsilon_{\|}$to increase any Vertical to HAN switching regions whilst decreasing HAN to Vertical switching regions. This is confirmed in Figure 5.18.
5.7 Varying the perpendicular dielectric constant ( $\epsilon_{\|}$fixed)

(a) Semi-transparent block showing all switching regions.


Figure 5.21: $\tau V$ plots for varying the perpendicular dielectric constant, $\epsilon_{\perp}$ from 1 to 20 .


Figure 5.22: Three key slices from Figure 5.21, where (a) $\epsilon_{\perp}=5(\Delta \epsilon=5)$, (b) $\epsilon_{\perp}=10(\Delta \epsilon=0)$, (c) $\epsilon_{\perp}=15(\Delta \epsilon=-5)$.

For $\epsilon_{\perp}$, the perpendicular dielectric constant, we expect to see similar behavior to that observed in the $\epsilon_{\|}$study. As before, by fixing $\epsilon_{\|}$at its default value $\left(\epsilon_{\|}=10\right)$, we are essentially varying the dielectric anisotropy.


Figure 5.23: Individual simulations with $\tau=2.5 \mathrm{~ms}, V=50$ volts and $\epsilon_{\perp}=5$ (black), 10 (red), 15 (green).

Note from Figure 5.23 that, as in the $\epsilon_{\|}$case, the preferred director orientation throughout the bulk of the cell is parallel to the field for the positive material $\left(\epsilon_{\perp}=5, \Delta \epsilon=5\right)$ and perpendicular to the field for the negative material $\left(\epsilon_{\perp}=15, \Delta \epsilon=-5\right)$. This fundamental similarity between the $\epsilon_{\|}$and $\epsilon_{\perp}$ studies is reflected in Figures 5.19 and 5.22.

However, note that there are significant differences between Figure 5.19(c) and Figure 5.22(a) for $V<0$, despite both plots corresponding to $\Delta \epsilon=+5$. This indicates that the dielectric effects on the cell are governed by more than just the dielectric anisotropy. To further inves-
tigate this, we need to perform a study on how the magnitude of the dielectric coefficients alter the switching characteristics of our cell.
5.8 Varying the dielectric constants ( $\Delta \epsilon$ fixed)

(a) Semi-transparent block showing all switching regions.


Figure 5.24: $\tau V$ plots for varying $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2$ from 3.5 to 17.5 , with $\Delta \epsilon=-5.0$ in each case.

To quantify the magnitude of the dielectric coefficients, we use the measure $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2$. Note that, in relation to our base parameter set, this gives a default value of $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2=12.5$. Throughout this parameter study, we fix the dielectric anisotropy at its default value of $\Delta \epsilon=-5.0$.


Figure 5.25: Three key slices from Figure 5.24.

From Figure 5.24, we see that the magnitude of the dielectric coefficients has a profound effect on the switching characteristics of the cell. To understand why, recall that the field equation is given by

$$
\begin{equation*}
E=\frac{D_{3}-E_{13} \sin (2 \theta) \frac{\partial \theta}{\partial z}}{\epsilon_{0}\left(\epsilon_{\perp}+\Delta \epsilon \sin ^{2} \theta\right)}, \quad D_{3}=\frac{-\epsilon_{0} V+E_{13} \int_{0}^{d} \frac{\sin (2 \theta) \frac{\partial \theta}{\epsilon_{1}+\Delta \epsilon \sin ^{2} \theta} d z}{\int_{0}^{d} \frac{1}{\epsilon_{\perp}+\Delta \epsilon \sin ^{2} \theta} d z} . . . ~}{\text {. }} \tag{5.4}
\end{equation*}
$$

We first consider the $D_{3}$ equation. For a fixed $\Delta \epsilon$, notice that as $\epsilon_{\perp}$ increases the two integrals in $D_{3}$ decrease. We can think of this as leaving the flexoelectric term in $D_{3}$ largely unaffected, whilst the magnitude of the term involving $V$ increases. Turning our attention to $E$, as $\epsilon_{\perp}$ is increased both $D_{3}$ and the flexoelectric term decrease in magnitude. Therefore, the overall effect of increasing $\epsilon_{\perp}$ is that the terms involving $E_{13}$ decrease. This can be likened to lower values of $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2$ magnifying any flexoelectric effects in the cell. We therefore expect larger values of $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2$ to produce vaguely similar results to those seen with lower values of $E_{13}$.

By looking at the progression from Figure 5.15(c) to Figure 5.15(b), we see that lowering $E_{13}$ from its default value increases the HAN to Vertical switching region with $V>0$. In comparison, we observe a similar trend as $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2$ is increased through its default value
from Figure 5.25(b) to Figure 5.25(c), which is in agreement with our analysis.


Figure 5.26: Three individual simulations with $\tau=2.5 \mathrm{~ms}, V=50$ volts and $\left(\epsilon_{\|}+\epsilon_{1}\right) / 2=4.5$ (black), 10.5 (red), 17.5 (green).

Since lowering $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2$ increases the effective value of $E_{13}$, we would expect to be able to draw further comparisons between the two parameters at the director level. Recall that for larger values of $E_{13}$, lower gradients are preferred when minimising the free energy, and therefore the associated timescale is shorter. From Figures 5.26-5.27, notice that we observe the same behavior as $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2$ is decreased. Specifically, from Figure 5.27 we see that the gradients in $\theta$ are significantly lower for smaller values of $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2$.


Figure 5.27: Relaxations after the voltage is removed, corresponding to Figure 5.26 with $\tau=2.5 \mathrm{~ms}$, $V=50$ volts and $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2=4.5$ (black), 10.5 (red), 17.5 (green).

Finally, notice from Figure 5.26 (a) that, just as we found when increasing $E_{13}$, lowering $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2$ accentuates the 'kink' in the relaxed HAN state, providing further evidence of a strong correlation between $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2$ and $E_{13}$.

### 5.9 Varying the splay elastic constant


(a) Semi-transparent block showing all switching regions.


Figure 5.28: $\tau V$ plots for varying the splay elastic constant, $K_{1}$, from $8.3 \times 10^{-12} \mathrm{~N}$ to $47.8 \times 10^{-12} \mathrm{~N}$.

We now move on to our parameter study for $K_{1}$, the splay elastic constant. From Figure 5.29 we see that higher values of $K_{1}$ expand the HAN to Vertical switching regions, most notably the non-local switching region.


Figure 5.29: Three key slices from Figure 5.28.


Figure 5.30: Three individual simulations with $\tau=2.5 \mathrm{~ms}, V=50$ volts and $K_{1}=8.3 \times 10^{-12} \mathrm{~N}$ (black), $2.78 \times 10^{-11} \mathrm{~N}$ (red), $4.78 \times 10^{-11} \mathrm{~N}$ (green).

Figure 5.30 shows the director profiles at each of the key points throughout three individual simulations. The first point to note is that varying $K_{1}$ changes the structure of the relaxed HAN state (Figure 5.30(A)). To understand the reason for this behaviour, we need to examine the governing equations. Considering only the elastic constants, the bulk equation reduces to

$$
\begin{equation*}
\frac{\partial \theta}{\partial t} \sim\left(K_{1} \cos ^{2} \theta+K_{3} \sin ^{2} \theta\right) \frac{\partial^{2} \theta}{\partial z^{2}}+\left(K_{3}-K_{1}\right)\left(\frac{\partial \theta}{\partial z}\right)^{2} \sin (2 \theta), \tag{5.5}
\end{equation*}
$$

whilst the bistable and monostable surface equations reduce to

$$
\begin{equation*}
\frac{\partial \theta}{\partial t} \sim\left(K_{1} \cos ^{2} \theta+K_{3} \sin ^{2} \theta\right) \frac{\partial \theta}{\partial z} \tag{5.6}
\end{equation*}
$$

and

$$
\begin{equation*}
\frac{\partial \theta}{\partial t} \sim-\left(K_{1} \cos ^{2} \theta+K_{3} \sin ^{2} \theta\right) \frac{\partial \theta}{\partial z}, \tag{5.7}
\end{equation*}
$$

respectively. Now, for $\theta \approx 0$, we have

$$
\begin{equation*}
\frac{\partial \theta}{\partial t} \sim K_{1} \frac{\partial^{2} \theta}{\partial z^{2}}, \tag{5.8}
\end{equation*}
$$

in the bulk, with

$$
\begin{equation*}
\frac{\partial \theta}{\partial t} \sim K_{1} \frac{\partial \theta}{\partial z} \tag{5.9}
\end{equation*}
$$

and

$$
\begin{equation*}
\frac{\partial \theta}{\partial t} \sim-K_{1} \frac{\partial \theta}{\partial z}, \tag{5.10}
\end{equation*}
$$

at the bistable and monostable surfaces, respectively. Note that for $\theta \approx \pi / 2$, the surface and bulk equations behave independently of $K_{1}$.

Equation (5.8) translates to positive growths in the bulk around areas of positive curvature in $\theta$, and negative growths in the bulk around areas of negative curvature in $\theta$. Therefore, as $K_{1}$ increases we expect to see increased linearity in the director profile for lower values of $\theta$. Equation (5.9) implies that, for $\theta$ close to zero, the director experiences elastic torque in the same direction as its gradient at the bistable surface, whilst equation (5.10) implies that the director experiences elastic torque in the opposite direction to its gradient at the monostable surface. We can therefore think of $K_{1}$ as a controlling the 'straightness', or 'linearity', of the director for lower values of $\theta$, with higher values of $K_{1}$ increasing the linearity and lower values of $K_{1}$ decreasing the linearity. This is in agreement with Figure 5.30(A).

Our interpretation of $K_{1}$ is further supported by Figure $5.30(B)-(C)$. Note that, in each plot, the curvature in the bulk of the cell decreases as $K_{1}$ increases. Furthermore, from Figure $5.30(\mathrm{C})$, we see that the director orientation at the monostable surface decreases as $K_{1}$ is increased due to the high positive gradient in $\theta$.

Figure 5.31 shows the early stages of director relaxation for each simulation. Notice that the bulk of the cell appears to experience a faster relaxation as $K_{1}$ is increased. This can be explained by equation (5.8), from where we can expect higher values of $K_{1}$ to give an


Figure 5.31: Relaxations after the voltage is removed, corresponding to Figure 5.30 with $\tau=2.5 \mathrm{~ms}$, $V=50$ volts and $K_{1}=8.3 \times 10^{-12} \mathrm{~N}$ (black), $2.78 \times 10^{-11} \mathrm{~N}$ (red), $4.78 \times 10^{-11} \mathrm{~N}$ (green).
increased positive effect in the bulk of the cell, therefore reducing the associated timescale. Consequentially, this reduces the elastic torque at the bistable surface and therefore favours a Vertical state, in agreement with Figure 5.30(D).

In relation to the non-local switching region, note from equation (5.10) that we expect the director to remain closer to $\theta=0$ during relaxation as $K_{1}$ is increased. This gives a lower gradient throughout the cell, which we expect to increase any non-local switching regions.

### 5.10 Varying the bend elastic constant


(a) Semi-transparent block showing all switching regions.


Figure 5.32: $\tau V$ plots for varying the bend elastic constant, $K_{3}$, from $0.8 \times 10^{-12} \mathrm{~N}$ to $20.3 \times 10^{-12} \mathrm{~N}$.

We now move on to our parameter study for $K_{3}$, the bend elastic constant. From Figure 5.33 we see that higher values of $K_{3}$ reduce the HAN to Vertical switching regions.


Figure 5.33: Three key slices from Figure 5.32.


Figure 5.34: Three individual simulations with $\tau=2.5 \mathrm{~ms}, V=50$ volts and $K_{3}=8.0 \times 10^{-13} \mathrm{~N}$ (black), $1.63 \times 10^{-11} \mathrm{~N}$ (red), $2.03 \times 10^{-11} \mathrm{~N}$ (green).

Figure 5.34 shows the director profiles at each of the key points throughout three individual simulations. As in the $K_{1}$ parameter study, we see that varying $K_{3}$ changes the structure of the relaxed HAN state (Figure 5.34(A)). Linearising equations (5.5)-(5.7) about $\theta \approx \pi / 2$ gives

$$
\begin{equation*}
\frac{\partial \theta}{\partial t} \sim K_{3} \frac{\partial^{2} \theta}{\partial z^{2}} \tag{5.11}
\end{equation*}
$$

in the bulk, with

$$
\begin{equation*}
\frac{\partial \theta}{\partial t} \sim K_{3} \frac{\partial \theta}{\partial z} \tag{5.12}
\end{equation*}
$$

and

$$
\begin{equation*}
\frac{\partial \theta}{\partial t} \sim-K_{3} \frac{\partial \theta}{\partial z}, \tag{5.13}
\end{equation*}
$$

at the bistable and monostable surfaces, respectively. It follows that the linearity of the director profile is expected to increase with $K_{3}$ as $\theta$ approaches $\pi / 2$. Similarly, from equations (5.9)-(5.10), we expect the director to experience elastic torque in the same and opposite directions to the gradient at the bistable and monostable surfaces, respectively. Therefore, $K_{3}$ controls the linearity of the director for values of $\theta$ close to $\pi / 2$, with higher values of $K_{3}$ increasing the linearity and lower values of $K_{3}$ decreasing the linearity. This is in agreement with Figure $5.34(\mathrm{~A})$. Our interpretation of $K_{3}$ is further supported by Figure 5.34(B)-(C). Note that, in each plot, the director angle at each surface decreases as $K_{3}$ is increased.


Figure 5.35: Relaxations after the voltage is removed, corresponding to Figure 5.34 with $\tau=2.5 \mathrm{~ms}$, $V=50$ volts and $K_{3}=8.0 \times 10^{-13} \mathrm{~N}$ (black), $1.63 \times 10^{-11} \mathrm{~N}($ red $), 2.03 \times 10^{-11} \mathrm{~N}$ (green).

To understand why higher values of $K_{3}$ favour a HAN state, we examine the relaxation process in Figure 5.35. Notice that, in each case, the main difference between the director profiles occurs at the bistable surface and that the gradient there is negative. Therefore, by equation (5.12), we expect higher values of $K_{3}$ to decrease $\frac{\partial \theta}{\partial t}$ at the bistable surface as the cell relaxes, thereby favouring a HAN state. This is in accordance with Figure 5.35 and, ultimately, Figure 5.34(D).

### 5.11 Varying the bistable surface anchoring strength


(a) Semi-transparent block showing all switching regions.


Figure 5.36: $\tau V$ plots for varying the bistable surface anchoring strength, $W_{0}$, from $1.0 \times 10^{-5} \mathrm{~N} / \mathrm{m}$ to $5.0 \times 10^{-5} \mathrm{~N} / \mathrm{m}$.

We now move on to our parameter study for $W_{0}$, the anchoring strength at the bistable surface. Large values of $W_{0}$ imply that the director at the bistable surface is firmly held in place at its nearest stable orientation. Therefore, for larger values of $W_{0}$ we require more elastic or field-related torque in order to break the bistable surface anchoring. From Figure 5.36 , we see that $W_{0}$ affects both the local and the non-local switching regions.


Figure 5.37: Three key slices from Figure 5.36.


Figure 5.38: Three individual simulations with $\tau=2.5 \mathrm{~ms}, V=50$ volts and $W_{0}=1.0 \times 10^{-5} \mathrm{~N} / \mathrm{m}$ (black), $4.0 \times 10^{-5} \mathrm{~N} / \mathrm{m}$ (red), $6.0 \times 10^{-5} \mathrm{~N} / \mathrm{m}$ (green).

Figure 5.38 shows the director profiles at each of the key points throughout three individual simulations. We see that the director profiles in plots (A)-(C) are very similar, but that the cell with the highest bistable anchoring strength relaxes to a Vertical state whilst the other two cells relax to a HAN state. This is due to higher values of $W_{0}$ raising the energy barrier associated with the bistable surface, as illustrated in Figure 5.39.


Figure 5.39: The bistable surface energy with $W_{0}=1.0 \times 10^{-5} \mathrm{~N} / \mathrm{m}$ (black), $4.0 \times 10^{-5} \mathrm{~N} / \mathrm{m}$ (red), $6.0 \times 10^{-5} \mathrm{~N} / \mathrm{m}$ (green).


Figure 5.40: Relaxations after the voltage is removed, corresponding to Figure 5.38 with $\tau=2.5 \mathrm{~ms}$, $V=50$ volts and $W_{0}=1.0 \times 10^{-5} \mathrm{~N} / \mathrm{m}$ (black), $4.0 \times 10^{-5} \mathrm{~N} / \mathrm{m}$ (red), $6.0 \times 10^{-5} \mathrm{~N} / \mathrm{m}$ (green).

Figure 5.40 shows the initial relaxation process of each director. Note that the lower the value of $W_{0}$, the faster the director is pulled down towards $\alpha_{1}$ at the bistable surface. This can be attributed to the lower, flatter bistable surface energy profiles associated with lower values of $W_{0}$, as illustrated in Figure 5.39. Therefore, with higher values of $W_{0}$ we require more elastic torque in order to break the bistable surface anchoring, whilst lower values of $W_{0}$ mean that the elastic torque in the bulk will dominate during relaxation.

### 5.12 Varying the monostable surface anchoring strength


(a) Semi-transparent block showing all switching regions.


Figure 5.41: $\tau V$ plots for varying the monostable surface anchoring strength, $W_{d}$, from $1.0 \times 10^{-3} \mathrm{~N} / \mathrm{m}$ to $2.0 \times 10^{-2} \mathrm{~N} / \mathrm{m}$.

We now move on to our parameter study for $W_{d}$, the anchoring strength at the monostable surface. Large values of $W_{d}$ imply that the director at the monostable surface is firmly held in place at it's nearest stable orientation. Therefore, for larger values of $W_{d}$ we require more elastic or field-related torque in order to perturb the director from its preferred orientation at the monostable surface. From Figure 5.41, we see that $W_{d}$ mainly affects the non-local switching region.


Figure 5.42: Three key slices from Figure 5.41.


Figure 5.43: Individual simulations with $\tau=2.5 \mathrm{~ms}, V=50$ volts and $W_{d}=1.0 \times 10^{-3} \mathrm{~N} / \mathrm{m}$ (black), $4.2 \times 10^{-3} \mathrm{~N} / \mathrm{m}$ (red), $1.0 \times 10^{-2} \mathrm{~N} / \mathrm{m}$ (green).

Figure 5.43 shows the director profiles at each of the key points throughout three individual simulations. We find that the first main difference between the director profiles occurs near $z=d$ during the trailing edge of the pulse (Figure $5.43(\mathrm{C})$ ), where the simulation with the highest $W_{d}$ does not distort as far from its stable orientation of $\pi / 2$. This is due to higher values of $W_{d}$ raising the energy barrier associated with the monostable surface, as illustrated
in Figure 5.44.


Figure 5.44: The monostable surface energy with $W_{d}=1.0 \times 10^{-3} \mathrm{~N} / \mathrm{m}$ (black), $4.2 \times 10^{-3} \mathrm{~N} / \mathrm{m}$ (red), $1.0 \times 10^{-2} \mathrm{~N} / \mathrm{m}$ (green).


Figure 5.45: Relaxations after the voltage is removed, corresponding to Figure 5.43 with $\tau=2.5 \mathrm{~ms}$, $V=50$ volts and $W_{d}=1.0 \times 10^{-3} \mathrm{~N} / \mathrm{m}$ (black), $4.2 \times 10^{-3} \mathrm{~N} / \mathrm{m}$ (red), $1.0 \times 10^{-2} \mathrm{~N} / \mathrm{m}$ (green).

Figure 5.45 shows the initial relaxation process of each director. Note that the higher the value of $W_{d}$, the faster the director relaxes to $\pi / 2$ at the monostable surface. This can be attributed to steeper gradients in the monostable surface energy as $W_{d}$ is increased, as seen in Figure 5.44. Therefore, with lower values of $W_{d}$ the global gradient throughout the cell is 'more negative' for longer throughout the cell's relaxation, and the non-local switching region expands to include lower voltages.

### 5.13 Varying the bistable surface relaxation coefficient


(a) Semi-transparent block showing all switching regions.


Figure 5.46: $\tau V$ plots for varying the bistable surface relaxation coefficient, $l_{s 0}$, from $1.0 \times 10^{-7} \mathrm{~m}$ to $2.0 \times 10^{-6} \mathrm{~m}$.

We now move on to our parameter study for $l_{s_{0}}$, the bistable surface relaxation coefficient. We expect higher values of $l_{s_{0}}$ to cause greater 'friction' at the bistable surface, meaning that more work is required in order to break the bistable surface anchoring. From Figure 5.47, we see that this translates to larger HAN to Vertical switching regions for higher values of $l_{s o}$.


Figure 5.47: Three key slices from Figure 5.46.


Figure 5.48: Three individual simulations with $\tau=2.5 \mathrm{~ms}, V=50$ volts and $l_{s_{0}}=1.0 \times 10^{-7} \mathrm{~m}$ (black), $5.0 \times 10^{-7} \mathrm{~m}(\mathrm{red}), 1.0 \times 10^{-6} \mathrm{~m}$ (green).

Figure 5.48 shows the director profiles at each of the key points throughout three individual simulations. We see that $l_{s_{0}}$ has little effect on the director profile at points (B) and (C), indicating that a pulse length of $\tau=2.5 \mathrm{~ms}$ is long enough to bring the simulation with the highest friction at the bistable surface (in this case, $l_{s_{0}}=1 \times 10^{-6} \mathrm{~m}$ ) to equilibrium when the field is applied. Between points (C) and (D), however, the simulation with the highest
friction relaxes to a Vertical state whilst the other two simulations relax to a HAN state.


Figure 5.49: Relaxations after the voltage is removed, corresponding to Figure 5.48 with $\tau=2.5 \mathrm{~ms}$, $V=50$ volts and $l_{s_{0}}=1.0 \times 10^{-7} \mathrm{~m}$ (black), $5.0 \times 10^{-7} \mathrm{~m}$ (red), $1.0 \times 10^{-6} \mathrm{~m}$ (green).

Figure 5.49 shows the initial relaxation process of each director. Notice that with less friction at the bistable surface $\left(l_{s_{0}}=1.0 \times 10^{-7} \mathrm{~m}\right.$ and $\left.l_{s_{0}}=5.0 \times 10^{-7} \mathrm{~m}\right)$ the elastic torque in the bulk is sufficient to break the surface anchoring and the cell relaxes to a HAN state. On the other hand, with $l_{s_{0}}=1.0 \times 10^{-6} \mathrm{~m}$ there is enough friction at the bistable surface to prevent the elastic torque from breaking the bistable anchoring and the cell relaxes to a Vertical state.

### 5.14 Varying the monostable surface relaxation coefficient


(a) Semi-transparent block showing all switching regions.


Figure 5.50: $\tau V$ plots for varying the monostable surface relaxation coefficient, $l_{s d}$, from $1.0 \times 10^{-6} \mathrm{~m}$ to $2.0 \times 10^{-5} \mathrm{~m}$.

We now move on to our parameter study for $l_{s_{d}}$, the monostable surface relaxation coefficient. We expect higher values of $l_{s_{d}}$ to cause greater friction at the monostable surface, meaning that more work is required in order to perturb the director at the monostable surface. From Figure 5.51, we see that this does not affect the lower HAN to Vertical switching region, but that it expands the non-local switching region.


Figure 5.51: Three key slices from Figure 5.50.

Figure 5.52 shows the director profiles at each of the key points throughout three individual simulations. We see that $l_{s_{d}}$ has little effect on the director profile at point (B). By point $(\mathrm{C})$, we see that lower values of $l_{s_{d}}$ result in the director being brought closer to $\theta=0$ at the monostable surface. Between points (C) and (D), the simulation with the lowest friction relaxes to a HAN state whilst the other two simulations relax to a Vertical state.


Figure 5.52: Three individual simulations with $\tau=2.5 \mathrm{~ms}, V=50$ volts and $l_{s_{d}}=1.0 \times 10^{-6} \mathrm{~m}$ (black), $1.0 \times 10^{-5} \mathrm{~m}$ (red), $2.0 \times 10^{-5} \mathrm{~m}$ (green).

Figure 5.53 shows the initial relaxation process of each director. Notice that with less friction at the monostable surface $\left(l_{s_{d}}=1.0 \times 10^{-6} \mathrm{~m}\right)$ the director at the monostable surface quickly relaxes to its preferred orientation near $\pi / 2$. With $l_{s_{d}}=1.0 \times 10^{-5} \mathrm{~m}$ and $l_{s_{d}}=2.0 \times 10^{-5} \mathrm{~m}$ the additional friction at the monostable surface maintains a low director orientation at $z=d$, and therefore a low overall gradient throughout the cell, for longer, increasing the non-local switching region. So whilst the director orientation at $z=d$ is higher with $l_{s_{d}}=2.0 \times 10^{-5} \mathrm{~m}$ at the instant the field is removed (Figure $5.52(\mathrm{C})$ ), it takes longer to relax to its equilibrium state.


Figure 5.53: Relaxations after the voltage is removed, corresponding to Figure 5.52 with $\tau=2.5 \mathrm{~ms}$, $V=50$ volts and $l_{s_{d}}=1.0 \times 10^{-6} \mathrm{~m}$ (black), $1.0 \times 10^{-5} \mathrm{~m}$ (red), $2.0 \times 10^{-5} \mathrm{~m}$ (green).

Our results also allow us to understand what should happen for values of $l_{s_{d}}$ much greater than $2.0 \times 10^{-5} \mathrm{~m}$. Note, from Figure $5.51(\mathrm{~b})-(\mathrm{c})$, that the non-local switching region is gradually shifting towards longer dwell-times as $l_{s_{d}}$ is increased. Recall that the non-local switching region occurs due to a low overall gradient throughout the cell during the early stages of relaxation, and that this can only occur if the director orientation is sufficiently low near $z=d$ at the time the voltage is removed (Figure 5.52(C)). With more friction at the monostable surface, it follows that longer dwell-times are required to lower the director orientation at $z=d$ to a level that enables non-local switching.

### 5.15 Summary

In this chapter, we have examined the switching characteristics of the nematic cell corresponding to our base parameter set (Table 2.1), using a bipolar pulse, with the no-llow model. Most significantly, we found that non-local director structures can influence the dirertor at the bistable surface, therefore having an impact on the final relaxed state. This non-local effect was found to originate in the flexoelectric part of the electric field during the initinl relaxation of the cell, shortly after the voltage was removed.

We also investigated the effect of each parameter in our model on the aforementioned switching characteristics. In some cases, the results were relatively straight-forward to understand. For example, we expected the parameters only relating to the monostable surface ( $W_{d}$ and $\boldsymbol{l}_{g_{d}}$ ) to affect just the non-local switching region, which was found to be the case. However, in other cases, such as $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2$, we needed to examine the governing equations in more detail in order to understand why the observed results occurred.

In the following section, we will re-simulate the results from this section using the flow model.

## 6 Results - flow model

In this section, we will investigate the switching characteristics of our liquid crystal cell using the flow model. As in the previous section, we begin by simulating with the base parameter set (Table 2.1), before moving on to each individual parameter in our model. Note that, since we are now including flow in our model, we have an additional parameter study concerning the slip-length, $\delta$ (see Section 2.8.2).

For each parameter study, we attempt to understand the observed flow and director profiles. We also note any significant differences between the 'flow' results and the corresponding 'now-flow' results obtained in the previous section.

### 6.1 The base parameter set

Figure 6.1(a) shows the $\tau V$ plot generated from the parameter set in Table 2.1 when the flow equation is included in our model. As before, red regions indicate switching from the Vertical state to the HAN state whilst blue regions indicate switching from the MAN state to the Vertical state.

In direct comparison with the corresponding 'no-flow' $\tau V$ plot (Figure 5.2), we see that both HAN to Vertical switching regions are largely unaffected by the inclusion of flow in our model. Similarly, for negative voltages the Vertical to HAN switching region appears to be the same in both cases. However, notice that we now have a white area of no switching between the two HAN to Vertical switching regions. In the no-flow case, this 'gap' was part of the Vertical to HAN switching region. In order to explain why we see no switching either way in this area, we test an individual $(\tau, V)$ point in the area of interest, for example, with $\tau=2.5 \mathrm{~ms}$ and $V=50$ volts, as in Figure 6.2.

The plots in Figure 6.2(I) show director profiles, taken at each of the critical points in the bipolar pulse, when simulating with $\tau=2.5 \mathrm{~ms}$ and $V=50$ volts from an initial HAN (black line) and Vertical (red line) state. The plots in Figure 6.2(II) show the corresponding relaxations, immediately after the voltage is removed. From Figure 6.2(I)(B), we see that the stressed equilibrium state is largely dependent on the initial state. In the no-flow case (Figure 5.4), the stressed equilibrium states are essentially the same for $V>20$, regardless

## (a) 2 D view. <br> (b) 3D view.




Figure 6.1: $\tau V$ plot generated using the base parameter set (Table 2.1) when the flow equation is included in our model. Plot (a) shows a standard two-dimensional $\tau V$ plot, and plot (b) shows the same $\tau V$ plot from a three-dimensional isometric viewpoint.


Figure 6.2: (I) Four director profiles taken at each of the key points labeled in Figure 5.1, with $\tau=2.5 \mathrm{~ms}$ and $V=50$ volts, corresponding to the white area of no switching in Figure 6.1. Both profiles use the full flow model, with the black line starting from an initial HAN state and the red line starting from an initial Vertical state. (II) The corresponding relaxations immediately after the voltage is removed.
of the initial state. However, with the inclusion of flow in our model the director experiences 'kickback' in one half of the cell $(z>d / 2)$ when starting from an initially Vertical state. This is due to the 'backflow' experienced in the flow profile (Figure 6.3(II)). Note that, despite this kickback, the director is mostly the same in the first half of the cell ( $z<\mathrm{d} / 2$ ) in earh case. By plot (I)(C), note that there is less of the bulk close to $\theta=0$ in the initially Vertical cell when compared with the initially HAN cell. From Figure 6.2(II)(n), we see that this dif. ference between the bulk profiles docs not affect the bistable surface during the rarly stages of relaxation. However, notice from Figure 6.2(II)(c) that the bulk in the initially Vertical cell remains close to $\theta=0$ for a shorter period of time than the initially HAN cell. Therefore, as each bulk relaxes, there is only enough elastic torque to break the surface anchoring in the initially HAN case and so each cell relaxes to its initial state.

Prior to our analysis of these results, we note that linearisation about some constant director orientation $\theta \approx \theta_{\text {lin }}$ reduces the bulk director and flow equations (equations (2.40) and (2.31), respectively) to

$$
\begin{equation*}
\gamma_{1} \frac{\partial \theta}{\partial t} \sim-m\left(\theta_{\operatorname{lin}}\right) \frac{\partial u}{\partial z}, \tag{0.1}
\end{equation*}
$$

and

$$
\begin{equation*}
\rho \frac{\partial u}{\partial t} \sim g\left(\theta_{\operatorname{lin}}\right) \frac{\partial^{2} u}{\partial z^{2}}+m\left(\theta_{\operatorname{lin}}\right) \frac{\partial^{2} \theta}{\partial z \partial t}, \tag{6.2}
\end{equation*}
$$

respectively. Also note that

$$
\begin{align*}
m(0)=m(\pi) & =\alpha_{3}<0  \tag{6.3}\\
g(0)=g(\pi) & =\frac{1}{2}\left[\alpha_{4}+\alpha_{3}-\alpha_{6}\right]>0,  \tag{6.4}\\
m(\pi / 2) & =-\alpha_{2}>0,  \tag{6.5}\\
g(\pi / 2) & =\frac{1}{2}\left[\alpha_{4}+\alpha_{5}-\alpha_{2}\right]>0 . \tag{6.6}
\end{align*}
$$

We now investigate the time between each critical point in more detail.

### 6.1.1 Voltage on

To understand why backflow occurs, we must consider the flow equation. The instant before the field is applied the cell is in equilibrium and hence there is no flow throughout the cell, giving

$$
\begin{equation*}
\frac{\partial u}{\partial z}=\frac{\partial^{2} u}{\partial z^{2}}=0 \tag{0.7}
\end{equation*}
$$

Also, in the Vertical state we have $\theta \approx \pi / 2$, so $m(\theta) \approx m(\pi / 2)$ and $g(\theta) \approx!(\pi / 2)$. Therefore, using equations (6.5), (6.6) and (6.7) with equation (6.2) yields

$$
\begin{equation*}
\frac{\partial u}{\partial t} \sim \frac{\partial^{2} \theta}{\partial z \partial t}, \tag{0.8}
\end{equation*}
$$

which translates to positive changes in the gradient of $\theta$ inducing growths in the flow profile. Now recall from the no-flow case that the dielectric torque pulls the director townrds $0=$ 0 when a voltage is applied. Note that the director is initially in a slightly off-vertical orientation, that is, the director is aligned with $\theta=\alpha_{2}(<\pi / 2)$ at $z=0$ and $\theta=\pi / 2$ at $z=d$. Hence, the director is closer to $\theta=0$, the energy minimum associaterl with the electric field term, in the first half of the cell. Therefore, the director experiences slightly more electric torque in the first half of the cell than it does in the sccond half of the cell, giving

$$
\begin{equation*}
\frac{\partial^{2} \theta}{\partial z \partial t}>0 \tag{G.9}
\end{equation*}
$$

throughout the bulk of the cell, the instant the field is applied. Thus, by equation (6.8), we expect the flow to be positive throughout the cell for sufficiently short timescales after a voltage is applied to a vertically aligned, negatively dielectric cell, which is supported by Figure 6.3(II).

In order to explain the consequence of backflow on the director, note from equations ( 0.1 ) and (6.5) that

$$
\begin{equation*}
\frac{\partial \theta}{\partial t} \sim-\frac{\partial u}{\partial z}, \tag{6.10}
\end{equation*}
$$

which translates to negative gradients in the flow profile having a positive effect on the director about $\theta=\pi / 2$. Now, from Figures 6.3 (II)(a)-(b) we see that this particular cell experiences a large amount of backflow once the voltage is applied, causing a large positive flow gradient in the first half of the cell and a large negative flow gradient in the second half of the cell. Therefore, by equation (6.10), we expect the backflow to increase $\theta$ in the second half of the cell, thus producing the kickback observed in Figure 6.3(I)(b). Similarly, we expect the backflow to decrease $\theta$ in the first half of the cell, causing the first half of the cell to decrease faster than the corresponding no-flow case, in accordance with Figure 6.4.

As the director configuration reacts to the backfow, it induces secondary effects on the flow profile. If we neglect inertia ( $\rho=0$ ) and linearise about $\theta \approx \pi / 2$ then, by equations


Figure 6.3: Profiles taken from the instant the voltage is applied, using the base parameter set, with $\tau=2.5 \mathrm{~ms}$ and $V=50$ volts, starting from an initially Vertical state. For each plot, four profiles are taken with red corresponding to the first profile and blue corresponding to the last profile. Plot (I) shows the director profiles whilst Plot (II) shows the flow profiles.


Figure 6.4: Four director profiles taken from just after the voltage is initially applied, with $\tau=2.5 \mathrm{~ms}$ and $V=50$ volts, corresponding to the 'flow' (black) and 'no-flow' (red) models. Both profiles start from an initially Vertical state.
(6.5)-(6.6), the flow equation reduces to

$$
\begin{equation*}
\frac{\partial^{2} u}{\partial z^{2}} \sim-\frac{\partial^{2} \theta}{\partial z \partial t}, \tag{6.11}
\end{equation*}
$$

which implies that the curvature of the flow profile at any point in the cell is dependent on the rate of change of the gradient in the director profile. Now, by Figure 6.3(I)(b), we see that $\frac{\partial^{2} \theta}{\partial z \partial t}$ is decreasing at either end of the cell and increasing in the middle of the cell. Therefore, by equation (6.11), we expect the flow profile to react by forming a cusp at the implied maximum near the centre of the cell, as observed in Figure 6.3(II)(c).

We see, from Figure 6.3(II), that the backflow increases until it reaches its maximum, roughly $4 \mu \mathrm{~s}$ after the voltage is applied, after which it begins to damp down. Now recall that, for a negative material, the dielectric terms favour a director orientation that is perpendicular to the applied field. In our case, this corresponds to any integer multiple of $\pi$. From Figure 6.3(I)(c) and Figure 6.3(II)(c), we see that by the time the flow effects begin to damp down the director is already close to $\theta=0$ in the first half of the cell and $\theta=\pi$ in the second half of the cell, and so the dielectric torque forces the directors towards the distorted configuration shown in Figure 6.2(B).


Figure 6.5: Profiles taken from the instant the voltage is applied, using the base parameter set, with $\tau=2.5 \mathrm{~ms}$ and $V=50$ volts, starting from an initially HAN state. For each plot, four profiles are taken with red corresponding to the first profile and blue corresponding to the last profile. Plot (a) shows the director profiles whilst Plot (b) shows the flow profiles.

Notice, from Figure 6.5, that kickback is also observed when simulating from an initial HAN state. However, since the amount of kickback is so small, the field effects dominate and the kickback quickly vanishes.

### 6.1.2 Polarity change

Figure 6.6 shows the director and flow profiles for just after the change in voltage polnrity: Whilst the director profile evolves in an expected manner, with the flexoclectric term causing $\theta(0) \rightarrow 0$ and $\theta(d) \rightarrow \pi$, the corresponding flow profiles require further investigntion. Mirall the $D_{3}$ term in the field equation, namely

$$
\begin{equation*}
D_{3}=\frac{-\epsilon_{0} V+\xi}{\int_{0}^{d} \frac{1}{\epsilon_{1}+\Delta \epsilon \sin ^{2} \theta} d z} \tag{0.12}
\end{equation*}
$$

where

$$
\begin{equation*}
\xi=E_{13} \int_{0}^{d} \frac{\sin (2 \theta) \frac{\partial \theta}{\partial_{2}}}{\epsilon_{\perp}+\Delta \epsilon \sin ^{2} \theta} d z \tag{0.13}
\end{equation*}
$$

之otice, from the profile corresponding to an initially Vertical state in Figure 6.2(B), that $\boldsymbol{\xi}$ is a positive quantity at the instant the voltage changes polarity. Therefore, when changing from a negative to a positive voltage polarity, $D_{3}$ decreases in magnitude causing a slight decrease in the field strength throughout the cell. We expect a decrease in the electric field to lower the gradient in the centre of the cell and increase the gradient close to $z=d$, such that the diclectric contribution to the gradient in $\theta$ is

$$
\frac{\partial^{2} \theta}{\partial z \partial t}:= \begin{cases}<0, & (0.5<z / d<0.7)  \tag{0.14}\\ >0, & (0.9<z / d<1.0) \\ =0, & \text { (elsewhere) }\end{cases}
$$

Additionally, due to the change in polarity, we expect the flexoelectric term to reduce the director gradient at $z=0$ and further increase the gradient in $\theta$ at $z=d$, such that the flexoelectric contribution to the gradient in $\theta$ is

$$
\frac{\partial^{2} \theta}{\partial z \partial t}:= \begin{cases}<0, & (z \approx 0)  \tag{6.15}\\ >0, & (z \approx d) \\ =0, & \text { (elsewhere) }\end{cases}
$$

Assuming the flow profile to be initially zero when the voltage changes polarity, we can use cquations (6.14)-(6.15) together with equation (6.2) and the appropriate linearisations in


Figure 6.6: Profiles taken from the change in voltage polarity, using the base parameter set, with $\tau=2.5 \mathrm{~ms}$ and $V=50$ volts, starting from an initially Vertical state. For each plot, four profiles are taken with red corresponding to the first profile and blue corresponding to the last profile. Plot (I) shows the director profiles whilst Plot (II) shows the flow profiles.
(6.3)-(6.6) to give

$$
\frac{\partial u}{\partial t}:= \begin{cases}>0, & (z \approx 0.0)  \tag{0.16}\\ =0, & (0.0<z / d<0.5) \\ >0, & (0.5<z / d<0.6) \\ <0, & (0.6<z / d<0.65) \\ >0, & (0.65<z / d<0.7) \\ =0, & (0.7<z / d<0.9) \\ <0, & (0.9<z / d<1.0) \\ >0, & (z \approx 1.0)\end{cases}
$$

Note that we expect the flow profile near the centre of the cell to contain na area of negative flow sandwiched between two areas of positive flow, since $m\left(\theta_{\text {lin }}\right)$ incurs two sign changes as $\theta_{\text {lin }}$ increases from zero to $\pi$. From Figure 6.6(II)(a), we see that the observed flow is in line with our prediction in equation (6.16). Notice, however, that close to cach boundary the flow is, while still in agreement with our prediction, very small. This indicates that the flow profiles in Figure 6.6(II)(a)-(c) are mainly due to the dielectric effects in the bulk of the cell.

After the initial flow profiles in Figure 6.6(II)(a)-(c), the flow reacts almost entirely to the changes in $\theta$ at the boundaries. From Figure 6.6(I)(d)-(f), note that the director at $z=d$ gradually moves towards $\theta=\pi$ whilst the bistable surface reacts much faster, almost reaching $\theta=\alpha_{2}$ by plot (e). Using this information with equation (6.2) and the appropriate lincarisations in equations (6.3)-(6.6), we can formulate a similar argument to show that the flow profiles in Figure 6.6(II)(d)-(f) are as expected.

One important point to note is that the flow observed in Figure 6.6(II) is two orders of magnitude smaller than the flow observed in Figure 6.3(II), indicating that the director structures change more rapidly than when the voltage is initially applied.

### 6.1.3 Voltage off

Figure 6.7 shows the director and flow profiles for just after the voltage is removed. As expected, the director favours $\theta=\pi / 2$ throughout the bulk of the cell and the evolution through plots (I)(a)-(f) behave in a sensible fashion. In a similar manner to the explanation of Figure 6.6(II), we can use our knowledge of the director's preferred evolution in the bulk of the cell to determine the initial flow profile. Note that, at the instant the field is remored,
we have

$$
\frac{\partial^{2} \theta}{\partial z \partial t}:= \begin{cases}>0, & (0.0<z / d<0.1)  \tag{6.17}\\ <0, & (0.5<z / d<0.7) \\ >0, & (0.95<z / d<1.0) \\ =0, & \text { (elsewhere) }\end{cases}
$$

Assuming the flow profile to be initially zero when the voltage changes polarity, we can use equations (6.14)-(6.15) together with equation (6.2) and the appropriate linearisations in (6.3)-(6.6) to give

$$
\frac{\partial u}{\partial t}:= \begin{cases}>0, & (0.0<z / d<0.005)  \tag{6.18}\\ <0, & (0.005<z / d<0.1) \\ =0, & (0.1<z / d<0.5) \\ >0, & (0.5<z / d<0.6) \\ <0, & (0.6<z / d<0.65) \\ >0, & (0.65<z / d<0.7) \\ =0, & (0.7<z / d<0.95) \\ >0, & (0.95<z / d<1.0)\end{cases}
$$

As before, we expect the flow profile near the centre of the cell to contain a negative area of flow sandwiched between two positive areas of flow, since $m\left(\theta_{\text {lin }}\right)$ changes sign twice as $\theta_{\text {lin }}$ increases from zero to $\pi$. Note that we also expect the flow to change sign near $z=0$ since $m\left(\theta_{\text {lin }}\right)$ changes sign as $\theta_{\text {lin }}$ decreases from $\pi / 2$ to zero. Furthermore, we expect to see a higher magnitude of flow around $z=0$, due to the conflict between the bulk elastic torque and the bistable surface anchoring, relative to the flow near $z=d$. As evidenced by Figure 6.7, our expectations are supported by the simulations.

After the initial flow profiles in Figure 6.7(II)(a)-(d), note that the flow profile reacts to changes in the director profile in a predictable manner.

Having explained each of the switching regions in the $\tau V$ plot corresponding to the base parameter set, we now move on to individual parameter studies.


Figure 6.7: Profiles taken during the relaxation process, using the base parameter set, with $\tau=2.5 \mathrm{~ms}$ and $V=50$ volts, starting from an initially Vertical state. For each plot, four profiles are taken with red corresponding to the first profile and blue corresponding to the last profile. Plot (I) shows the director profiles whilst Plot (II) shows the flow profiles.

### 6.2 Varying the preferred HAN state



Figure 6.8: $\tau V$ plots for varying the preferred director orientation at the bistable surface for a relaxed HAN state, $\alpha_{1}$, from $0^{\circ}$ to $11.5^{\circ}$.


Figure 6.9: Three key slices from Figure 6.8.

From Figure 6.8, we see that varying $\alpha_{1}$ has much the same effect as we saw in the 'no-flow' case. However, note from Figure 6.9(a) that with $\alpha_{1}=0^{\circ}$ we see a distinct, definite overlap between the two switching regions when low positive voltages are applied.


Figure 6.10: Three individual simulations with $\alpha_{1}=0^{\circ}, \tau=2.5 \mathrm{~ms}$ and $V=14$ (black), 17 (red) and 20 (green) volts.

Figure 6.10 gives the four key points for three simulations corresponding to Figure 6.9(a). We see that the simulations behave as expected until plot (C), by which point the flexoelectric torque has broken the bistable surface anchoring and, in the 17 volts case, forced $\theta$ towards $\left(\alpha_{2}-\pi\right)$ radians at $z=0$. From plot (C) note that, in each case, there is insufficient elastic torque in the bulk of the cell to break the bistable surface anchoring and the cell relaxes to the Vertical state in the 14 and 20 volt cases, and a third state state in the 17 volts case.

Note that the third state shown in Figure 6.10(D) is an acceptable state, since our bistable
surface energy equation has minima at $\alpha_{1} \pm k \pi$ and $\alpha_{2} \pm k \pi$ for any integer $k$. Regarding the $\tau V$ plot generated in Figure 6.9(a), the tracing algorithm classifies the state by measuring its optical transmission after 1.0 seconds of relaxation, and comparing this value with the optical transmission of the relaxed HAN and Vertical states. The $\log$ file shows that the HAN and Vertical states have optical transmissions of 0.8411 and $4.97 \times 10^{-4}$, respectively, whist the third state has an optical transmission of 0.8181 . Therefore, the third state is 'optically similar' to (though slightly darker than) a relaxed HAN state.

The existence of a state which is neither HAN nor Vertical could potentially cause problems, such as a possible inability for the cell to switch to another state. Note that any such problems can be avoided by selecting a value of $\alpha_{1} \geq 0.5^{\circ}$.


Figure 6.11: Three individual simulations with $\alpha_{1}=0^{\circ}, \tau=2.5 \mathrm{~ms}$ and $V=$ (a) 14 , (b) 17 and (c) 20 volts. Plots (I) show the director profiles whilst Plots (II) show the corresponding flow profiles. Note that all profiles are taken during the first half of the pulse, after the maximum kickback has been attained. For each plot, four profiles are shown with red corresponding to the first profile (taken at time $=1.5 \mathrm{~ms}$ ) and blue corresponding to the last profile (taken at time $=3.5 \mathrm{~ms}$ ). Note the difference in scales for each of the plots in (II).

To understand why the director rotates below zero at the bistable surface, we examine the director and flow profiles just before the voltage changes polarity (i.e. between plots (B) nuld (C) of Figure 6.10), which are given in Figure 6.11. Note that once the backflow damps down the director orientation in the second half of the cell is determinel entirely by the field and elastic effects, with the field favouring $\theta \approx \pi$ and the elastic cifects favouring $\theta \approx 0$. In the 14 volts case (Figure 6.11(a)), we see that the elastic torque outweighs the field effects, and the initial kickback damps down, producing a similar profile to that olserverl in the corresponding no-flow case. In the 20 volts case (Figure 6.11 (c)), the field counterbalances the elastic effects in the second half of the cell and the cell remains in a distorted equilibrium state.

In the 17 volts case (Figure 6.11(b)), note that the field is not quite strong enough to hold the director in equilibrium, and the director in the second half of the cell is gradually forced away from $\pi$. This movement of the director induces the flow profile shown in Figure 6.11 (II)(b). Now, for $\theta \approx 0$, note from equations (6.3) and (6.1) that we have

$$
\begin{equation*}
\frac{\partial \theta}{\partial t} \sim \frac{\partial u}{\partial z} \tag{6.10}
\end{equation*}
$$

which translates to negative gradients in the flow profile inducing reductions in $\theta$ when 0 is close to zero. So the director in the first half of the cell experiences a negative effect due to the flow profile observed in Figure 6.11(II)(b). With a steep enough gradient in the flow profile, it is possible for the director in the first half of the cell to be forced slightly below zero which, in turn, draws the director orientation slightly below zero at the bistable surface. With $\theta<0$ at the bistable surface, note that, upon changing the polarity of the voltage, the flexoelectric term in the surface energy equation now favours ( $a_{2}-\pi$ ) as opposed to $a_{2}$, and so the system evolves to the configuration observed in Figure 6.10(C).

Note that we do not observe any such behavior with $\alpha_{1}>0$, since the negative flow effect in the first half of the cell is not large enough to force $\theta$ below zero nt the bistable surface.

### 6.3 Varying the preferred Vertical state


(a) Semi-transparent block showing all switching regions.


Figure 6.12: $\tau V$ plots for varying the preferred director orientation at the bistable surface for a relaxed Vertical state, $\alpha_{2}$, from $78.5^{\circ}$ to $90^{\circ}$.


Figure 6.13: Three key slices from Figure 6.12.

From Figure 6.12, we see that varying $\alpha_{2}$ has the same effect as we saw in the 'no-flow' case when simulating from a HAN state. However, from Figure 6.13(b)-(c), we see that there is now an additional Vertical to HAN switching region when high positive voltages are applied, and that this new region overlaps with the upper HAN to Vertical switching region.


Figure 6.14: Three individual simulations with $\tau=2.5 \mathrm{~ms}, V=80$ volts and $\alpha_{2}=80.0^{\circ}$ (black), $85^{\circ}$ (red), $90^{\circ}$ (green).

Figure 6.14 gives individual simulations at the four key points during the bipolar pulse for each of the cells in Figure 6.13. We see, by Figure 6.14(A), that $\alpha_{2}$ affects the location of the internal boundary layer in the director profile close to the middle of the cell. Specifically, increasing $\alpha_{2}$ moves the wall closer to $z=0$. This is somewhat expected. Recall, from Section 6.1.1, that the dielectric torque is initially smaller as $\theta$ tends towards $\pi / 2$. Therefore, by increasing $\alpha_{2}$ the bulk of the cell is initially closer to $\pi / 2$ which reduces the dielectric torque and allows the flow profile to induce positive growths in $\theta$ in a larger portion of the
cell. Consequentially, the director may experience kickback in parts of the cell which were previously dominated by the field effects.

From Figure 6.14(D), we see that the additional switching region in 6.13(b)-(c) is not a true HAN state. However, as in Section 6.2, the tracing algorithm classified this state as being 'optically similar' to the HAN state (for $\alpha_{2}=85^{\circ}$, the HAN and Vertical states have optical transmissions of 0.7731 and $2.839 \times 10^{-5}$, respectively, whilst the state shown in Figure $6.14(\mathrm{D})$ has optical transmission 0.8284 ). Given our cell parameters, the third state has a higher optical transmission than the relaxed HAN state, and so we shall refer to this as a 'Super-HAN' state.


Figure 6.15: Three individual relaxations with $\tau=2.5 \mathrm{~ms}, V=80$ volts and (a) $\alpha_{2}=80^{\circ}$, (b) $\alpha_{2}=85^{\circ}$ and (c) $\alpha_{2}=90^{\circ}$. Plots (I) show the director profiles whilst Plots (II) show the corresponding flow profiles. For each plot, four profiles are shown with red corresponding to the first profile (taken at time $=5.0 \mathrm{~ms}$ ) and blue corresponding to the last profile (taken at time $=5.03 \mathrm{~ms}$ ).

To understand why the $\alpha_{2}=85^{\circ}$ simulation relaxes to a Super-HAN state but the $\alpha_{2}=80^{\circ}$ simulation does not, we examine the director and flow profiles corresponding to just after the voltage has been removed. By Figure 6.15 we see that, in the second half of the cell,
the induced flow has a negative gradient. Therefore, by equations (6.1), (6.3) and ( 0.4 ), wo expect this negative gradient in the flow profile to induce an increase in $\theta$ in the sccond half of the cell. However, in contrast with the flow effects, note that the elastic torque throughout the bulk of the cell induces a reduction in $\theta$. Now notice that the $\boldsymbol{r}_{2}=85^{\circ}$ ras han $a$ larger portion of the bulk near $\theta=\pi$, and so we expect there to be less clastic torpue to counterbalance the flow effects in the second half of the cell. Therefore, with high highor values of $\alpha_{2}$, it may be possible for the flow profile to force the director above $\pi$ in the wrond half of the cell during the initial relaxation. Consequentially, the director at $:=d$ may be forced above $\pi$ such that it relaxes towards $3 \pi / 2$ instead of its usual orientation of $\pi / 2$.

### 6.4 Varying the flexoelectric constant


(a) Semi-transparent block showing all switching regions.

(b) Plot of solid 'slices'.

(c) HAN $\rightarrow$ Vertical only.

(d) Vertical $\rightarrow$ HAN only.

Figure 6.16: $\tau V$ plots for varying the flexoelectric constant, $E_{13}$, from $1.0 \times 10^{-11} \mathrm{C} / \mathrm{m}$ to $1.0 \times 10^{-10} \mathrm{C} / \mathrm{m}$.


Figure 6.17: Three key slices from Figure 6.16.

From Figures 6.16-6.17, we see that, apart from the flow-induced area of no switching, altering $E_{13}$ produces the same effects on the switching regions as we observed in the 'noflow' case (Section 5.5). Also, from Figure 6.18, we see that the director is affected in accordance with the explanations given in Section 5.5. Specifically, note that higher values of $E_{13}$ favour lower gradients throughout the bulk of the cell.


Figure 6.18: Three individual simulations with $\tau=2.5 \mathrm{~ms}, V=50$ volts and $E_{13}=2.1 \times 10^{-11} \mathrm{C} / \mathrm{m}$ (black), $5.0 \times 10^{-11} \mathrm{C} / \mathrm{m}$ (red), $1.0 \times 10^{-10} \mathrm{C} / \mathrm{m}$ (green).
6.5 Varying the parallel dielectric constant ( $\epsilon_{\perp}$ fixed)

(a) Semi-transparent block showing all switching regions.

(b) Plot of solid 'slices'.

(c) HAN $\rightarrow$ Vertical only.

(d) Vertical $\rightarrow$ HAN only.

Figure 6.19: $\tau V$ plots for varying the parallel dielectric constant, $\epsilon_{\|}$, from 1 to 20 .


Figure 6.20: Three key slices from Figure 6.19.

From Figure 6.19, we see that varying $\epsilon_{\|}$produces a variety of interesting switching characteristics. Notice, from Figure 6.20(b), that we now observe two areas in which there is no switching between states, with one Vertical to HAN switching region sandwiched in between.


Figure 6.21: Three individual simulations from Figure $6.20(\mathrm{~b})$, with $\epsilon_{\|}=8.5, \tau=2.5 \mathrm{~ms}$ and $V=25$ (black), 50 (red), 80 (green) volts.

Figure 6.21 shows three individual simulations relating to Figure 6.20 (b) at 25,50 and 80 volts. In the 25 volts case, we see that at the instant the voltage is removed (plot (C)) there is insufficient elastic torque in the bulk of the cell to break the bistable surface anchoring, and the cell relaxes to a Vertical state. In the 50 volts case, note that there is enough elastic torque in the bulk of the cell to break the bistable surface anchoring, and so the cell relaxes to a HAN state. The 80 volts case, however, requires careful consideration in order to fully understand why the cell relaxes to a Vertical state.


Figure 6.22: Relaxations corresponding to Figure 6.21 with $\epsilon_{\|}=8.5, \tau=2.5 \mathrm{~ms}$ and $V=25$ (black), 50 (red), 80 (green) volts.

By Figure $6.21(\mathrm{C})$, note that the largest difference between the 50 and 80 volt cases occurs at the monostable surface, where the higher voltage forces $\theta$ very close to $\pi$. Now consider the quantity

$$
\begin{equation*}
\xi=E_{13} \int_{0}^{d} \frac{\sin (2 \theta) \frac{\partial \theta}{\partial z}}{\epsilon_{\perp}+\Delta \epsilon \sin ^{2} \theta} d z . \tag{6.20}
\end{equation*}
$$

Note that for $\theta<\pi / 2$, positive gradients in $\theta$ increase $\xi$, whilst for $\theta>\pi / 2$, positive gradients in $\theta$ decrease $\xi$. Therefore, by Figure $6.21(\mathrm{C})$, we can conclude that $\xi$ is larger in the 50 volts case than in the 80 volts case. Now, using a similar argument to that presented for the non-local switching region (Section 5.2), we can show that this reduction in $\xi$ reduces the torque at $z=0$, meaning that the bistable surface anchoring is not broken and the cell relaxes to a Vertical state. From Figure 6.22, we see that $\xi$ remains smaller in the 80 volts case for much of the relaxation process, giving an increased non-local field effect at the bistable surface.

Having explained the switching regions in Figure 6.20(b), we now examine the effect of changing $\epsilon_{\|}$. Specifically, we wish to determine why the non-local area (discussed above) is replaced with a Vertical to HAN region for both small and large $\epsilon_{\|}$, as shown in Figure 6.20(a) and Figure 6.20(c), respectively.

Let us turn our attention to the director profiles in Figure 6.23. We first concern ourselves with the $\epsilon_{\|}=4.0$ case. Note that for a negative material, decreasing $\epsilon_{\|}$increases the electric torque in the bulk of the cell when a voltage is applied. Therefore, using a similar line of reasoning to that presented in Section 6.3, we can expect lower values of $\epsilon_{\|}$to reduce the amount of director-kickback in the cell, thus moving the flow-induced wall in the director profile closer to $z=d$, as observed in Figure 6.23(B). When the voltage is removed, the


Figure 6.23: Three individual simulations with $\tau=2.5 \mathrm{~ms}, V=80$ volts and $\epsilon_{\|}=4.0$ (black), 8.5 (red). 14.5 (green).
elastic torque near $z=0$ is greater in the $\epsilon_{\|}=4.0$ case, thereby favouring a HAN state. Furthermore, note that the $\epsilon_{\|}=4.0$ case has more of its bulk close to $\theta=0$, which also favours a final HAN state. It is therefore reasonable to expect that, with $c_{\|}=4.0$, the cell should relax to a HAN state, as in Figure 6.23(D).

To explain the relaxation observed in the $\epsilon_{\|}=14.5$ case, we note that, at the instant the field is removed, there is considerably less torque throughout the bulk of the cell than in the $\epsilon_{\|}=4.0$ and $\epsilon_{\|}=8.5$ cases. Therefore, using a similar argument to that presented in Section 6.3, we can expect the flow effects to have a more pronounced positive effect on the director close to $z=d$, thus enabling the director to relax towards the Super-HAN state as shown in Figure 6.23(D).
6.6 Varying the perpendicular dielectric constant ( $\epsilon_{\|}$fixed)

(a) Semi-transparent block showing all switching regions.


Figure 6.24: $\tau V$ plots for varying the perpendicular dielectric constant, $\epsilon_{\perp}$ from 1 to 20 .


Figure 6.25: Three key slices from Figure 6.24.

Figure 6.24 shows the results obtained from varying $\epsilon_{\perp}$ when flow is included in our model. In comparison with the no-flow results (Figure 5.21), we see that lower values of $\epsilon_{\perp}$ now produce unexpected results. Specifically, as can be seen by comparing Figure 6.25(a) with Figure $5.22(\mathrm{a})$, the majority of the HAN to Vertical switching region $(V<0)$ is lost when modelling with the flow equation.


Figure 6.26: Individual simulations with $\tau=2.5 \mathrm{~ms}, V=-80$ volts and $\epsilon_{\perp}=5$. Here we are comparing the flow model (black) from Figure 6.25(a) with the no-flow model (red) from Figure 5.22(a).

To understand why it becomes more difficult to switch from the HAN state to the Vertical state when including flow in our model, we first consider the director profiles in Figure 6.26, which simulate from an initial HAN state with $\tau=2.5 \mathrm{~ms}, V=-80$ volts and $\epsilon_{\perp}=5$ $(\Delta \epsilon=5)$, using the flow model (black line) and the no-flow model (red line). We see that both the flow and no-flow models produce virtually identical director profiles in plots (A), (B) and (C), but that once the voltage is removed the no-flow simulation relaxes to a Vertical
state whilst the flow simulation relaxes to a HAN state. From this, we speculate that this difference in relaxations may be due to an unexpected flow effect and so the initial relaxation of the flow simulation is considered.


Figure 6.27: The individual relaxations associated with the 'flow' simulation in Figure 6.26. Plots (I) show the director profiles whilst Plots (II) show the corresponding flow profiles. For each plot, four profiles are shown with red corresponding to the first profile and blue corresponding to the last profile.

From Figure $6.26(\mathrm{C})$, note that, the instant the voltage is removed, the elastic torque near the bistable surface causes the gradient in $\theta$ to decrease near $z=0$. Therefore, assuming an initially flat flow profile, we may use equations (6.3)-(6.6) and (6.7) with equation (6.2) to predict a non-zero flow profile of the form

$$
\frac{\partial u}{\partial t}:= \begin{cases}>0, & (0.0<z / d<0.05)  \tag{6.21}\\ <0, & (0.05<z / d<0.1) \\ =0, & (0.1<z / d<1.0)\end{cases}
$$

which concurs with Figure 6.27 (II)(a). Note that since $\theta$ is very close to $\pi / 2$ in the bulk of the cell, we would expect the positive part of the flow profile to dominate in the early stages
of relaxation, as seen in Figure 6.27(II)(b)-(c), producing a large amount of backflow.


Figure 6.28: Director relaxations, immediately after the voltage is removed, corresponding to Figure 6.26 Here we are comparing the flow model (black) with the no-flow model (red).

Note that the induced backflow contributes to a sharp peak in the flow profile. By the same reasoning as used to explain the kickback-effect in Section 6.1, we expect the negative gradient in the flow profile to increase the director on the right-hand side of the peak whilst the extremely high positive gradient in the flow profile decreases the director immediately to the left of the peak. This leads to the director kickback that is observed in each of the plots of Figure 6.28, along with a reduction in the gradient of $\theta$ at $z=0$ relative to the no-flow model, as in Figure 6.28(a). By decreasing the gradient in $\theta$ at $z=0$, note that the elastic torque acting on the bistable surface is also reduced. Once the backflow has damped down, there is insufficient elastic torque to break the anchoring at the bistable surface and the cell relaxes to a HAN state.


Figure 6.29: Individual simulations with $\epsilon_{\perp}=20, \tau=2.5 \mathrm{~ms}$ and $V=25$ (black), 50 (red) and 80 volts. Note that each simulation corresponds with a point in 6.25(c).

This result has implications for all positive materials, namely that backflow promotes Ver-
tical to HAN switching. This is in contrast to the fow effects we observe with a negative material, in which backflow hinders some areas of Vertical to HAN switching.

Finally, note from Figure 6.25(c) that with $\epsilon_{\perp}=20$ we observe two nrens in which the cell does not switch either way. To investigate this, we examine the individual dircetor profiles in Figure 6.29. It can be immediately seen from Figure 6.29(C) that the dircetor structures are virtually identical to those in Figure 6.21(C). We therefore conclude that the 80 volts case in Figure 6.25 (c) does not switch from the Vertical state to the $11 / \mathrm{N}$ state dur to the non-local field effect presented in Section 6.5.
6.7 Varying the dielectric constants ( $\Delta \epsilon$ fixed)

(a) Semi-transparent block showing all switching regions.


Figure 6.30: $\tau V$ plots for varying $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2$ from 3.5 to 17.5 , with $\Delta \epsilon=-5.0$ in each case.


Figure 6.31: Three key slices from Figure 6.30.

From Figure 6.30, we see that altering $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2$ produces, with the exception of the gap, the same effects on the switching regions as we observed in the 'no-flow' case (Section 5.8). Also, from Figure 6.32, we see that the 80 volts case in Figure 5.25(b) does not switch from the Vertical state to the HAN state due to the non-local field effect presented in Section 6.5.


Figure 6.32: Three individual simulations with $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2=10.5, \tau=2.5 \mathrm{~ms}$ and $V=25$ (black), 50 (red) and 80 (green) volts.
6.8 Varying the splay elastic constant

(a) Semi-transparent block showing all switching regions.


Figure 6.33: $\tau V$ plots for varying the splay elastic constant, $K_{1}$, from $8.3 \times 10^{-12} \mathrm{~N}$ to $47.8 \times 10^{-12} \mathrm{~N}$.


Figure 6.34: Three key slices from Figure 6.33.

Figure 6.33 shows the results obtained from varying $K_{1}$ when flow is included in our model. In comparison with the no-flow results (Figure 5.28), we see that lower values of $K_{1}$ introduce a new Vertical to HAN switching region for high positive voltages. This is also evident in Figure 6.34(a).


Figure 6.35: Three individual simulations with $K_{1}=8.3 \times 10^{-12} \mathrm{~N}, \tau=2.5 \mathrm{~ms}$ and $V=50$ (black), 70 (red) and 80 (green) volts.

Figure 6.35 shows how the director varies with $K_{1}=8.3 \times 10^{-12} \mathrm{~N}$ for different voltages. Note that, from our previous parameter studies, we are already in a position to explain the evolution of each director. In the 50 volts case, we see that there is enough elastic torque to break the bistable surface anchoring during relaxation, and so the cell relaxes to a HAN state. In the 70 volts case, the director is forced closer to $\theta=\pi$ at $z=d$ during the second half of the pulse, which magnifies the non-local effect (Section 6.5) and favours a Vertical state. In the 80 volts case, the director experiences even less elastic torque at $z=d$ during
relaxation, allowing the flow effects discussed in Section 6.3 to rotate the director beyond $\theta=\pi$ and relax towards a Super-HAN state. These explanations are supported by Figure 6.36 .


Figure 6.36: Relaxations corresponding to Figure $6.35 K_{1}=8.3 \times 10^{-12} \mathrm{~N}, \tau=2.5 \mathrm{~ms}$ and $V=50$ (black), 70 (red) and 80 (green) volts.

To understand why increasing $K_{1}$ reduces the Vertical to HAN switching region, and eliminates the Super-HAN state, we relate back to our findings in the no-flow case (Section 5.9). Note that linearising equations (5.5)-(5.7) about $\theta \approx \pi$ also produces equations (5.8)-(5.10). Therefore, as $K_{1}$ is increased, we can expect the director to remain closer to $\theta=\pi$ at the monostable surface throughout its relaxation. This increases the non-local effect, and so the director favours $\theta=\pi / 2$ at the bistable surface. With more elastic torque at $z=d$ we can also expect the flow effect to be reduced, which prevents the Super-HAN state from occurring.

### 6.9 Varying the bend elastic constant


(a) Semi-transparent block showing all switching regions.


Figure 6.37: $\tau V$ plots for varying the bend elastic constant, $K_{3}$, from $0.8 \times 10^{-12} \mathrm{~N}$ to $20.3 \times 10^{-12} \mathrm{~N}$.


Figure 6.38: Three key slices from Figure 6.37.

Figure 6.37 shows the results obtained from varying $K_{3}$ when flow is included in our model In comparison with the no-flow results (Figure 5.32), we see that the results are very similar. However, Figure 6.38(b) is a particularly interesting slice in which we observe three distinct areas of Vertical to HAN switching when $\tau=2.5 \mathrm{~ms}$, namely $V=5,50$ and 80 volts.


Figure 6.39: Three individual simulations with $K_{3}=2.48 \times 10^{-13} \mathrm{~N}, \tau=2.5 \mathrm{~ms}$ and $V=50$ (black), 70 (red) and 80 (green) volts.

The individual points of interest are given in Figure 6.39. In the 50 volts case, there is less flexoelectric torque at $z=d$ and so the director is further from $\theta=\pi$ at the time the voltage is removed (plot (C)). This inhibits any non-local effects, and allows the bistable surface anchoring to be broken causing the cell to relax into a HAN state. In the 70 volts case, there is more flexoelectric torque at $z=d$, causing the director to be closer to $\theta=\pi$ by the time the voltage is removed, which increases the non-local field effect (Section 6.5) and forces a Vertical state. In the 80 volts case, the elastic torque at $z=0$ becomes so strong
that the non-local effect is negated and the cell relaxes to a HAN state. These explanations are supported by Figure 6.40.


Figure 6.40: Relaxations corresponding to Figure 6.39 with $K_{3}=2.48 \times 10^{-13} \mathrm{~N}, \tau=2.5 \mathrm{~ms}$ and $\mathrm{V}=50$ (black), 70 (red) and 80 (green) volts.

From the corresponding results using the no-flow model (Section 5.10), it follows that HAN to Vertical regions are expected to dominate for lower values of $K_{3}$ whilst Vertical to HAN regions are expected to dominate for larger values of $K_{3}$. This is in agreement with Figure 6.37 .

### 6.10 Varying the bistable surface anchoring strength


(a) Semi-transparent block showing all switching regions.


Figure 6.41: $\tau V$ plots for varying the bistable surface anchoring strength, $W_{0}$, from $1.0 \times 10^{-5} \mathrm{~N} / \mathrm{m}$ to $5.0 \times 10^{-5} \mathrm{~N} / \mathrm{m}$.


Figure 6.42: Three key slices from Figure 6.41.

Figure 6.41 shows the results obtained from varying $W_{0}$ when flow is included in our model. By comparing Figure 6.41 with Figure 5.36, we see that $W_{0}$ affects both the no-flow model and the flow model in a similar fashion. However, in the flow model we see that $W_{0}$ also affects the white band of no switching, as in Figure 6.42(b). This can be explained by examining individual $(\tau, V)$ points, as in Figure 6.43.


Figure 6.43: Three individual simulations with $W_{0}=2.3 \times 10^{-5} \mathrm{~N} / \mathrm{m}, \tau=2.5 \mathrm{~ms}$ and $V=15$ (black), 20 (red) and 30 (green) volts.

From Figure 6.43 (B) we see that, in the 15 volts case, any kickback is quickly dominated by the field effects. This is because the field is too small to produce the initial backflow that is required in order to produce a sustained over-rotation in the director. In the 20 and 30 volts case, the initial backflow is large enough to force the director close to $\theta=\pi$ in the second half of the cell. From Figure 6.43(C), we see that the elastic torque at the bistable surface increases with the applied voltage. However, the 15 volts case has more of the bulk close to
$\theta=0$. Therefore, as the cell relaxes, there is more elastic torque acting on the bistable surface in the 15 volts case than there is in the 20 volts case, as seen in Figure 6.44. Therefore, the 15 volts simulation relaxes towards a HAN state whilst the 20 volts case relaxes towards a Vertical state.


Figure 6.44: Relaxations corresponding to Figure 6.43 with $W_{0}=2.3 \times 10^{-5} \mathrm{~N} / \mathrm{m}, \tau=2.5 \mathrm{~ms}$ and $V=15$ (black), 20 (red) and 30 (green) volts.

One further point to note is that, for $1.5 \times 10^{-5}<W_{0}<1.8 \times 10^{-5}$, the cell contains an isolated 'island' of no switching, as in Figure 6.45. This has implications for the employed tracing algorithm, which are discussed in Section 4.10.


Figure 6.45: $\tau V$ plot, showing just Vertical to HAN switching, corresponding to $W_{0}=1.6 \times 10^{-5} \mathrm{~N} / \mathrm{m}$.

Figure 6.46 shows three individual points taken from Figure 6.45. From plot (B), we see that all three cases are very similar. However, closer inspection reveals that the longer pulselength has in fact distorted the bulk of the cell so that $\theta$ is slightly lower than the other two cases. This indicates that, in the 0.3 ms and 0.6 ms cases, the cell is not in equilibrium, and the dielectric torque is gradually forcing the director towards $\theta=0$. We now consider the


Figure 6.46: Critical points for individual simulations corresponding to Figure 6.45 with $W_{0}=1.6 \times$ $10^{-5} \mathrm{~N} / \mathrm{m}, V=18$ volts and $\tau=0.3$ (black), 0.6 (red) and 2.5 (green) milliseconds. Note that theme simulations are different lengths, and that their directors have only been superimposed to illustrate the differences at their respective critical points.
0.3 ms and 0.6 ms cases. From plot (C), note that the longer pulse-length allows for more of the bulk to be affected by the elastic torque near $z=0$. Therefore, when the fied is removed, there is less elastic torque in the bulk to act against the bistable surface anchoring. Hence, the 0.6 ms favours the Vertical state whilst the 0.3 ms case favours the IIAN state.

Turning our attention to the 2.5 ms case, note from plot (C) that, for a sufficiently long pulse-length, the dielectric torque is able to completely negate the director kickback, forcing the bulk of the cell towards $\theta=0$. When the voltage is removed, there is sufficiently more elastic torque (relative to the 0.6 ms case) in the bulk of the cell throughout the relaxation, thereby favouring a HAN state.

### 6.11 Varying the monostable surface anchoring strength


(a) Semi-transparent block showing all switching regions.


Figure 6.47: $\tau V$ plots for varying the monostable surface anchoring strength, $W_{d}$, from $1.0 \times 10^{-3} \mathrm{~N} / \mathrm{m}$ to $2.0 \times 10^{-2} \mathrm{~N} / \mathrm{m}$.


Figure 6.48: Three key slices from Figure 6.47.

Figure 6.47 shows the results obtained from varying $W_{d}$ when flow is included in our model. When considering the no-flow results (Figure 5.41), $W_{d}$ appears to affect both models in a similar fashion. We see from Figure 6.48 that higher values of $W_{d}$ promote Vertical to HAN switching for relatively high voltages, whilst maintaining part of the area in which no switching occurs.


Figure 6.49: Individual simulations with $\tau=2.5 \mathrm{~ms}, V=80$ volts and $W_{d}=1.0 \times 10^{-3} \mathrm{~N} / \mathrm{m}$ (black), $4.2 \times 10^{-3} \mathrm{~N} / \mathrm{m}$ (red), $1.0 \times 10^{-2} \mathrm{~N} / \mathrm{m}$ (green).

From Figure 6.49(C), it becomes clear why higher values of $W_{d}$ favour a Vertical state. For low values of $W_{d}$, the torque at the monostable surface forces the director close to $\theta=\pi$ by the time the voltage is removed (plot (C)), which produces the non-local effect discussed in Section 6.5. Therefore, lower values of $W_{d}$ favour a Vertical state. As $W_{d}$ is increased, the director at the bistable surface becomes more strongly anchored at $\pi / 2$, which reduces the non-local effect and favours a HAN state.

### 6.12 Varying the bistable surface relaxation coefficient


(a) Semi-transparent block showing all switching regions.


Figure 6.50: $\tau V$ plots for varying the bistable surface relaxation coefficient, $l_{s 0}$, from $1.0 \times 10^{-7} \mathrm{~m}$ to $2.0 \times 10^{-6} \mathrm{~m}$.


Figure 6.51: Three key slices from Figure 6.50.

Figure 6.50 shows the results obtained from varying $l_{s_{0}}$ when flow is included in our model. When considering the no-flow results (Figure 5.46), $l_{s_{0}}$ appears to affect both models in exactly the same way.

### 6.13 Varying the monostable surface relaxation coefficient



Figure 6.52: $\tau V$ plots for varying the monostable surface relaxation coefficient, $l_{s d}$, from $1.0 \times 10^{-6} \mathrm{~m}$ to $2.0 \times 10^{-5} \mathrm{~m}$.


Figure 6.53: Three key slices from Figure 6.52.

As in the no-flow case, we see that varying $l_{s_{d}}$ affects any non-local regions of the $\tau V$ plot. Figure 6.53(a) shows that for low values of $l_{s_{d}}$ we observe an additional Vertical to HAN switching region, which overlaps the non-local HAN to Vertical switching region.


Figure 6.54: Three individual simulations with $\tau=2.5 \mathrm{~ms}, V=50$ volts and $l_{s_{d}}=1.0 \times 10^{-6} \mathrm{~m}$ (black), $1.0 \times 10^{-5} \mathrm{~m}$ (red), $2.0 \times 10^{-5} \mathrm{~m}$ (green).

To understand why this overlap occurs, we first examine the director profiles at each of the critical points (Figure 6.54). As expected, the only difference between the director profiles occurs near $z=d$, most notably in plot (C). Upon removing the voltage, the two higher values of $l_{s_{d}}$ relax to a Vertical state whilst the simulation corresponding to the lower value of $l_{s_{d}}$ relaxes to a Super-HAN state.

Recall from Section 5.14 that $l_{s_{d}}$ controls the friction at the monostable surface, with lower values of $l_{s_{d}}$ producing less friction and higher values of $l_{s_{d}}$ producing more friction. Now


Figure 6.55: Relaxations corresponding to Figure 6.54 with $\tau=2.5 \mathrm{~ms}, V=50$ volts and $l_{s_{d}}=1.0 \times 10^{-6} \mathrm{~m}$ (black), $1.0 \times 10^{-5} \mathrm{~m}$ (red), $2.0 \times 10^{-5} \mathrm{~m}$ (green).
recall, from Section 6.3, that the Super-HAN state occurs due to backflow during the initial relaxation of the director. Therefore, with lower values of $l_{s_{d}}$ we can expect the director to be more susceptible to the flow-induced torque at $z=d$, causing the director to relax towards $\theta=3 \pi / 2$ at the monostable surface. This is illustrated in Figure 6.55.

### 6.14 Varying the partial slip-length



Figure 6.56: $\tau V$ plots for varying the parallel dielectric constant, $\delta$, from $1 \times 10^{-8}$ to $1.001 \times 10^{-5}$.


Figure 6.57: Three key slices from Figure 6.56.

Figure 6.56 shows the results obtained from varying $\delta$, the partial slip-length parameter. Note that as $\delta$ is increased we observe a second region of no switching, which occurs when high positive voltages are applied. As in previous sections, we attempt to explain this behavior by examining the director profiles at specific $(\tau, V)$ points.


Figure 6.58: Three individual simulations with $\tau=2.5 \mathrm{~ms}, V=80$ volts and $\delta=1 \times 10^{-10}$ (black), $2.5001 \times 10^{-6}(\mathrm{red}), 1.00001 \times 10^{-5}$ (green).

Figure 6.58 shows three individual simulations at each of the critical points in the bipolar pulse. From plots (A)-(C), it can be seen that there is very little difference between the three simulations for most of the simulation-time. However, from plot (D) we see that the no-slip case relaxes to a Vertical state whilst the two partial-slip cases relax to a new state with $\theta \approx \alpha_{2}$ at $z=d$ and $\theta \approx-\pi / 2$ at $z=d$. From the $\log$ files, we see that this third state has an optical transmission of 0.8789 , whilst the HAN and Vertical states have optical transmissions of 0.7746 and $5.051 \times 10^{-4}$, respectively. This third state, which we shall
denote the 'Reverse Super-HAN' state, is significantly lighter than the natural HAN state. Nonetheless, the Reverse Super-HAN state is 'optically similar' to the relaxed HAN state, and so the tracing algorithm classifies it as such.


Figure 6.59: Relaxations corresponding to Figure 6.58 with $\tau=2.5 \mathrm{~ms}, V=80$ volts and $\delta=1 \times 10^{-10}$ (black), $2.5001 \times 10^{-6}(\mathrm{red}), 1.00001 \times 10^{-5}$ (green) metres. Plot (I) shows the director profiles whilst plot (II) shows the corresponding flow profiles.

Using Figure $6.58(\mathrm{C})$, we can predict the behaviour of the gradient in $\theta$. At the instant the field is removed

$$
\frac{\partial^{2} \theta}{\partial z \partial t}:= \begin{cases}>0, & (0<z / d<0.1)  \tag{6.22}\\ >0, & (z \approx d) \\ =0, & \text { (elsewhere) }\end{cases}
$$

Therefore, using equation (6.22) together with equation (6.2) and the appropriate linearisations in (6.3)-(6.6), and assuming an initially flat flow profile, we have

$$
\frac{\partial u}{\partial t}:= \begin{cases}>0, & (0<z / d<0.05)  \tag{6.23}\\ <0, & (0.05<z / d<0.1) \\ <0, & (z \approx d) \\ =0, & (\text { elsewhere })\end{cases}
$$

Recall the partial-slip condition,

$$
\begin{equation*}
\left.u\right|_{z=0}=\left.\delta \frac{\partial u}{\partial z}\right|_{z=0} \tag{6.21}
\end{equation*}
$$

By definition, for a non-zero slip-length the flow at the bistable boundary is directly proportional to the gradient in the flow profile at the bistable boundary. So for positive gradients in $u$ at the bistable boundary (as predicted by equation (6.23)) we expect a positive flow velocity at $z=0$. By equation (6.24), we also expect the flow at the bistable surface to increase with $\delta$. Therefore, for sufficiently high values of $\delta$ we can expect the gradient throughout the flow profile to be negative when the gradient in $u$ is positive at the bistable surface. This agrees with the flow profiles in Figure 6.58(II)(a).

Given a negative gradient throughout the flow profile, equations (6.1) and (6.3) give

$$
\begin{equation*}
\frac{\partial \theta}{\partial t} \sim \frac{\partial u}{\partial z}<0 \tag{0.25}
\end{equation*}
$$

which implies a decrease in the bulk of the director. Furthermore, as $\delta$ is increased, we expect an increased negative effect throughout the bulk of the cell during the early stages of relaxation. Close inspection of Figure 6.58(I) indicates that this is indeed the case, with the bulk of the director 'bowing' below $\theta=0$ for larger values of $\delta$.

Should the bulk of the cell be forced far enough below $\theta=0$, we might expect there to be enough elastic torque near $z=d$ to pull the director below zero at the monostable surface. Consequentially, the director may relax towards $\theta=-\pi / 2$, as in Figure 6.59(1)(d).

### 6.15 Summary

In this section, we have examined the switching characteristics of the nematic coll corrmsponding to our base parameter set (Table 2.1) with the flow model. Overall, with the exception of the flow-induced areas of no switching, the effect of marying ench parnueter was largely the same as the no-flow cases discussed in Section 5. However, the inclusion of flow led to some interesting and unexpected differences between the two models.

For our base parameter set, we found that above some voltage threshold the director experiences flow-induced kickback which forces $\theta$ outside the normal range of ( $0, \pi / 2$ ). This was found to be especially prominent when simulating from an initially Vertical state, and leads to different distorted equilibrium states for the initially HAN and initially Vertical simulations. Essentially, this can lead to the final state of the director having some dependance on its initial state. For our particular cell, director kickback led to some switching regions containing areas of no switching either way.

We also found that the non-local field effect, originally presented in Section 5.2, is also prevalent in a cell which exhibits a high degree of kickback in its distorted state. Furthermore, the non-local effect was found to behave in exactly the same way in each case, which leads to specific parameter sets favouring a particular final state, regardless of the cells initial state.

Our simulations indicated that flow effects can cause the director to relax into final cquilibrium states which are neither HAN nor Vertical. These equilibrium states, which we refer to as Super-HAN states, occur when the flow causes $\theta$ to move outside the range ( $0, \pi$ ) during the early stages of relaxation. Such states were found to be limited to high voltages or low director pre-tilts. However, since it is possible for the director to obtain a relaxerl Super-HAN state, we note that it would be useful to find a way of escaping back to a relaxed state which is truly HAN or Vertical. In order to determine whether or not this is possible, further research is clearly required. In the interim, to avoid Super-HAN states, a manageable alternative would be to avoid voltages in excess of 50 volts and pre-tilts of less than $0.5^{\circ}$. It should be noted that the Super-HAN states could be an artifact of our onedimensional model and, by modelling twist, such states may relax to a more energetically-farournble state.

Although our results are mainly oriented towards investigating the switching charncteristics of a negative dielectric material, we briefly investigated the flow effects when a positive material is used (Section 6.6). We found that, in contrast to a negative material, a positive
material does not experience director kickback when a voltage is initially applicy to the evell, and so the distorted states were found to be the same regardless of the initial state. Howerre. the director was found to experience kickback when the voltage was removed from the coll. This kickback worked against the clastic torque at the bistable surface, therely farouring a final HAN state. This led to a large reduction in HAN to Vertical switching regions, nud $n$ large increase in Vertical to HAN switching regions, for negative voltages.

## 7 Conclusions

In this thesis, we have constructed two models for a nematic Zenithal Bistable Device (ZI3D). in one dimension, by employing various mathematical techniques such ns Ericksen-Leslic the ory and the Maxwell equations. The difference between these two models is that one is a simplification which does not attempt to model flow effects, whilst the other is a more complicated model which includes the flow equations. We then introduced a number of ndrancorl numerical methods, such as a fully-implicit time integration method, adaptive timestepping and a moving mesh algorithm, for solving these equations. After some simulation analyser, we concluded that our fully-implicit method with adaptive time-stepping and a moving tursh algorithm provided the best balance of speed and solution accuracy.

One of the important parts of this thesis is our original tracing algorithm for nutomatically generating $\tau V$-plots in an efficient and robust manner. This relied heavily on our test for determining when a cell is approaching an equilibrium state, and which equilibrium state it is approaching, once an electric field is removed. Our code-base was built in such a way that it is completely modular, meaning that it is relatively straight-forward to replace any of the underlying routines with a suitable alternative. This means that the tracing algorithm is not constricted to our simple one dimensional model, and it may be easily adapted to work with two or even three-dimensional models.

Using our numerical methods and our tracing algorithm, we examined the switching characteristics of a bistable nematic cell when a bipolar pulse is applied, initially using just the no-flow model. We found that non-local director structures can influence the director at the bistable surface, therefore having an impact on the final relaxed state. This non-local effect was found to originate in the flexoelectric part of the electric field during the initial relaxation of the cell, shortly after the voltage was removed.

We also investigated the effect of each model parameter in our model on the switching characteristics of the cell. For each parameter, the integrity of our numerical results was verified by examining the governing equations in detail whenever necessary. Quite interestingly, we found that the flexoelectric constant, $E_{13}$, contains a 'sweet-spot' for triggering HAN to Vertical switching (see Section 5.5). For low values of $E_{13}$, the flexoclectric torque wns too small to break the bistable surface anchoring when the voltage was applied whist, when $E_{13}$ was too large, the flexoelectric torque was strong enough to re-break the surface anchoring when the voltage was removed. So there exists an interval of values for $E_{13}$ within which
the switching capabilitics of the cell are optimised. We also found that varying the prefersel Vertical state, $\alpha_{2}$, has a profound effect on the non-local switching region whilst varyiug the preferred HAN state, $\alpha_{1}$, has little effect on the non-local switching region. Varying the magnitude of the dielectric cocfficients, $\left(\epsilon_{\|}+\epsilon_{\perp}\right) / 2$, with fixed dielectric anisotrony; $\Delta_{1}$. whe found to dramatically affect the switching characteristics of the cell.

Finally, we examined the switching characteristics of a bistable nematic cell when a hijolar pulse is applied using the complete model, which included the flow equations. We found that, above some voltage threshold, the director experienced flow-induced kickback which forces $\boldsymbol{\theta}$ outside its normal range of $(0, \pi / 2)$. This director kickback led to some switching regions containing areas of no switching either way. With the exception of the director kickback and its effects, the flow model was found to generally behave in a somewhat similar fashion to the no-flow model when low to moderate voltages were applied. For higher voltages, however, some parameter sets were found to induce flow effects which caused the director to telax into final equilibrium states that were neither HAN nor Vertical. We referred to such states as Super-HAN states, since they exhibited a higher optical transmission than the standard HAN state.

Although our results were mainly oriented towards investigating the switching character istics of a negative dielectric material, we briefly investigated the flow effects when a presitive material was used. We found that, in contrast to a negative material, a positive material does not experience director kickback when a voltage is initially applied to the cell, and so the distorted states were found to be the same regardless of the initial state. Hownver, the director was found to experience kickback when the voltage was removed from the cell, thus reducing any HAN to Vertical switching regions and increasing Vertical to HAN switching regions when negative voltages were applied.

Regarding areas for improvement and further work, our numerical methods would bencfit from some further investigation into nondimensionalisation of the flow equation. At present. time adaption is only carried out in terms of $\theta$ and our fully implicit method appears to give a higher priority to accuracy in $\theta$. Nondimensionalising the flow equation would allow the convergence criterion of our fully implicit method and our time-adaption algorithm to give an equal weighting to both $\theta$ and $u$.

Finally, while our $\tau V$-plot generator has proven to be extremely robust and reliable, it cannot detect any 'holes' within a tracing region. Despite such cases being somewhat rare,
they do exist and must be accounted for. Several potential solutions to this deficiency anr described in Section 4.10.

## A Formulation of the bistable surface energy

We wish to find an appropriate Fourier Cosine Series (FCS) to describe the bistable surface energy. Ideally, this function will allow for any $\alpha_{1}$ (the preferred HAN orientation at $\theta(0)$ ). $\alpha_{2}$ (the preferred Vertical orientation at $\theta(d)$ ) and $\beta$ (the height of the sceondary energy barrier, relative to the primary energy barrier) to be chosen.


Figure A.1: The bistable surface energy function for varying $\alpha_{1}$, with $\alpha_{2}=80^{\circ}$ and $\beta=1.2$ in each cace.

However, from Figure A.1(c), we see that for some values of $\alpha_{1}$ and $\alpha_{2}$ the energy function becomes unreasonable in that, instead of exhibiting local maxima, it exhibits local minima at $\theta=\left(-\pi+\alpha_{1}+\alpha_{2}\right) / 2$ and $\theta=\left(\pi+\alpha_{1}+\alpha_{2}\right) / 2$. What follows is the derivation of $n$ energy function which allows for any $\alpha_{1}$ and $\alpha_{2}$ whilst constraining $\beta$ so as to prevent the function from exhibiting the aforementioned minima that are observed in Figure A.1(c).

Let the bistable surface energy function be

$$
\begin{equation*}
w_{\text {surf }}=W_{0} f\left(\theta-\left(\alpha_{1}+\alpha_{2}\right) / 2\right), \tag{A.1}
\end{equation*}
$$

where we assume

$$
\begin{equation*}
f(X)=a_{0}+a_{1} \cos (2 X)+a_{2} \cos (4 X)+a_{3} \cos (6 X) \tag{1.2}
\end{equation*}
$$

so that

$$
\begin{equation*}
\Rightarrow f^{\prime}(X)=-2 a_{1} \sin (2 X)-4 a_{2} \sin (4 X)-6 a_{3} \sin (6 X) \tag{d.3}
\end{equation*}
$$

and

$$
\Rightarrow f^{\prime \prime}(X)=-4 a_{1} \cos (2 X)-16 a_{2} \cos (4 X)-36 a_{3} \cos (0 X)
$$

where

$$
\begin{equation*}
X=\theta-\left(\alpha_{1}+\alpha_{2}\right) / 2 \tag{1.5}
\end{equation*}
$$

We require $f(X)$ to satisfy the following conditions:

1. At $\theta=\alpha_{1}$ (i.e. $X=\left(\alpha_{1}-\alpha_{2}\right) / 2$ ), we require a local minimum of height 0 , i.c.

$$
\begin{aligned}
f\left(\left(\alpha_{1}-\alpha_{2}\right) / 2\right) & =0 \\
\Rightarrow a_{0}+a_{1} \cos \left(\alpha_{1}-\alpha_{2}\right)+a_{2} \cos \left(2\left(\alpha_{1}-\alpha_{2}\right)\right)+a_{3} \cos \left(3\left(\alpha_{1}-\alpha_{2}\right)\right) & =0, \quad \text { (A.C) }
\end{aligned}
$$

and

$$
\begin{align*}
f^{\prime}\left(\left(\alpha_{1}-\alpha_{2}\right) / 2\right) & =0, \\
\Rightarrow-2 a_{1} \sin \left(\alpha_{1}-\alpha_{2}\right)-4 a_{2} \sin \left(2\left(\alpha_{1}-\alpha_{2}\right)\right)-6 a_{3} \sin \left(3\left(\alpha_{1}-\alpha_{2}\right)\right) & =0, \tag{A.i}
\end{align*}
$$

and

$$
\begin{equation*}
f^{\prime \prime}\left(\left(\alpha_{1}-\alpha_{2}\right) / 2\right)>0 \tag{A.S}
\end{equation*}
$$

2. At $\theta=\alpha_{2}$ (i.e. $X=\left(\alpha_{2}-\alpha_{1}\right) / 2$ ), we require a local minimum of height 0 , i.e.

$$
\begin{aligned}
f\left(\left(\alpha_{2}-\alpha_{1}\right) / 2\right) & =0 \\
\Rightarrow a_{0}+a_{1} \cos \left(\alpha_{2}-\alpha_{1}\right)+a_{2} \cos \left(2\left(\alpha_{2}-\alpha_{1}\right)\right)+a_{3} \cos \left(3\left(\alpha_{2}-\alpha_{1}\right)\right) & =0, \quad \text { (ג.9) }
\end{aligned}
$$

and

$$
\begin{aligned}
f^{\prime}\left(\left(\alpha_{2}-\alpha_{1}\right) / 2\right) & =0 \\
\Rightarrow-2 a_{1} \sin \left(\alpha_{2}-\alpha_{1}\right)-4 a_{2} \sin \left(2\left(\alpha_{2}-\alpha_{1}\right)\right)-6 a_{3} \sin \left(3\left(\alpha_{2}-\alpha_{1}\right)\right) & =0,(\lambda .10)
\end{aligned}
$$

and

$$
\begin{equation*}
f^{\prime \prime}\left(\left(\alpha_{2}-\alpha_{1}\right) / 2\right)>0 \tag{A.11}
\end{equation*}
$$

(N.B. This is exactly the same as condition 1.)
3. At $\theta=\left(\alpha_{1}+\alpha_{2}\right) / 2$ (i.e. $X=0$ ), we require a local maximum of height 1 , i.e.

$$
\begin{align*}
f(0) & =1 \\
\Rightarrow a_{0}+a_{1}+a_{2}+a_{3} & =1, \tag{d.12}
\end{align*}
$$

and

$$
\begin{align*}
f^{\prime}(0) & =0  \tag{A.13}\\
\Rightarrow 0 & =0, \quad \text { (satisficd automatically) }
\end{align*}
$$

and

$$
\begin{equation*}
f^{\prime \prime}(0)<0 \tag{A.14}
\end{equation*}
$$

4. At $\theta=\left(-\pi+\alpha_{1}+\alpha_{2}\right) / 2$ (i.e. $X=-\pi / 2$ ), we require a local maximum of height $\beta$, i.e.

$$
\begin{align*}
f(-\pi / 2) & =\beta \\
\Rightarrow a_{0}-a_{1}+a_{2}-a_{3} & =\beta \tag{d.15}
\end{align*}
$$

and

$$
\begin{align*}
f^{\prime}(-\pi / 2) & =0  \tag{A.16}\\
\Rightarrow 0 & =0, \quad \text { (satisfied automatically) }
\end{align*}
$$

and

$$
\begin{equation*}
f^{\prime \prime}(-\pi / 2)<0 \tag{d.17}
\end{equation*}
$$

(N.B. $\beta=1$ sets both energy barriers to the same height.)

Now, since equations (A.13) and (A.16) are satisfied automatically, and condition 2 is anal. ogous to condition 1 , our remaining four conditions given by equations (A.6), (A.7), (A.12)
and (A.15) may be written in matrix form as

$$
\left[\begin{array}{cccc}
1 & \cos \left(\alpha_{1}-\alpha_{2}\right) & \cos \left(2\left(\alpha_{1}-\alpha_{2}\right)\right) & \cos \left(3\left(\alpha_{1}-\alpha_{2}\right)\right)  \tag{d.1s}\\
0 & \sin \left(\alpha_{1}-\alpha_{2}\right) & 2 \sin \left(2\left(\alpha_{1}-\alpha_{2}\right)\right) & 3 \sin \left(3\left(\alpha_{1}-\alpha_{2}\right)\right) \\
1 & 1 & 1 & 1 \\
1 & -1 & 1 & -1
\end{array}\right]\left[\begin{array}{l}
a_{0} \\
a_{1} \\
a_{2} \\
a_{3}
\end{array}\right]=\left[\begin{array}{l}
0 \\
0 \\
1 \\
\beta
\end{array}\right] .
$$

Equation (A.18) solves to give

$$
\left.\begin{array}{l}
a_{1}=(\beta(A-B)-(A+B)) / C  \tag{d.19}\\
a_{2}=2(\beta(D-E)-(D+E)) / C \\
a_{3}=(\beta(B-F)+(B+F)) / C \\
a_{0}=1-a_{1}-a_{2}-a_{3}
\end{array}\right\}
$$

where

$$
\begin{align*}
& A=2 \cos ^{2}\left(\alpha_{1}-\alpha_{2}\right)\left(1+2 \sin ^{2}\left(\alpha_{1}-\alpha_{2}\right)\right)-3 \sin ^{2}\left(\alpha_{1}-\alpha_{2}\right) \\
& B=2 \cos \left(\alpha_{1}-\alpha_{2}\right) \\
& C=8 \sin ^{4}\left(\alpha_{1}-\alpha_{2}\right) \\
& D=2 \cos ^{3}\left(\alpha_{1}-\alpha_{2}\right)  \tag{A.20}\\
& E=3 \cos ^{2}\left(\alpha_{1}-\alpha_{2}\right)-1 \\
& F=1+\cos ^{2}\left(\alpha_{1}-\alpha_{2}\right)
\end{align*}
$$

Now, notice that the second derivative given in equation (A.4) may be written using the Fourier coefficients computed in equation (A.19) as,

$$
\begin{align*}
& f^{\prime \prime}(X)=(-4 / C)[(\beta(A-B)-(A+B)) \cos (2 X) \\
&+8(\beta(D-E)-(D+E)) \cos (4 X) \\
&+9(\beta(B-F)+(B+F)) \cos (6 X)] \tag{A.21}
\end{align*}
$$

which can be rearranged, collecting terms in $\beta$, to give

$$
\begin{equation*}
f^{\prime \prime}(X)=\beta q(X)-p(X) \tag{A.22}
\end{equation*}
$$

where

$$
\begin{align*}
& q(X)=(-4 / C)\{(A-B) \cos (2 X) \\
&\left.+8(D-E) \cos (4 X)+9(B-F) \cos \left(6 X^{\prime}\right)\right\}  \tag{A.23}\\
& p(X)=(-4 / C)\{(A+B) \cos (2 X) \\
&\left.+8(D+E) \cos (4 X)-9(B+F) \cos \left(6 . X^{\prime}\right)\right\} \tag{1.24}
\end{align*}
$$

Now, by equation (A.8), we require

$$
\left.\begin{array}{ll}
\beta>\frac{p\left(\left(\alpha_{1}-\alpha_{2}\right) / 2\right)}{q\left(\left(\alpha_{1}-\alpha_{2}\right) / 2\right)}, & q\left(\left(\alpha_{1}-\alpha_{2}\right) / 2\right)>0  \tag{1.25}\\
\beta<\frac{p\left(\left(\alpha_{1}-\alpha_{2}\right) / 2\right)}{q\left(\left(\alpha_{1}-\alpha_{2}\right) / 2\right)}, & q\left(\left(\alpha_{1}-\alpha_{2}\right) / 2\right)<0
\end{array}\right\} .
$$

to ensure that we have a local minimum at $\theta=\alpha_{1}$ and $\theta=\alpha_{2}$. Similarly, equation (A.14) implies that

$$
\left.\begin{array}{lll}
\beta<\frac{p(0)}{q(0)}, & \text { if } & q(0)>0  \tag{A.26}\\
\beta>\frac{p(0)}{q(0)}, & \text { if } & q(0)<0
\end{array}\right\}
$$

is needed to ensure a local maximum at $\theta=\left(\alpha_{1}+\alpha_{2}\right) / 2$, whilst equation (A.17) implies that

$$
\left.\begin{array}{ll}
\beta<\frac{p(-\pi / 2)}{q(-\pi / 2)}, & q(-\pi / 2)>0  \tag{A.2i}\\
\beta>\frac{p(-\pi / 2)}{q(-\pi / 2)}, & q(-\pi / 2)<0
\end{array}\right\}
$$

is required to ensure a local maximum at $\theta=\left(-\pi+\alpha_{1}-\alpha_{2}\right) / 2$.
Using equations (A.25)-(A.27), we may select an appropriate value for $\beta$. With a value for $\beta$, our Fourier coefficients may be computed using equation (A.19), allowing us to construct the Fourier Cosine Series given in equation (A.2).

## B The flexoelectric effect on the relaxed HAN state

Recall that the governing bulk equation for this model is

$$
\begin{align*}
\gamma_{1} \frac{\partial \theta}{\partial t}= & \left(K_{1} \cos ^{2} \theta+K_{3} \sin ^{2} \theta\right) \frac{\partial^{2} \theta}{\partial z^{2}} \\
& +\frac{1}{2} \sin (2 \theta)\left[\epsilon_{0} E^{2} \Delta \epsilon+\left(K_{3}-K_{1}\right)\left(\frac{\partial \theta}{\partial z}\right)^{2}-2 E_{13} \frac{\partial E}{\partial z}\right] \tag{B.1}
\end{align*}
$$

where

$$
\begin{equation*}
E=\frac{D_{3}-E_{13} \sin (2 \theta) \frac{\partial \theta}{\partial z}}{\epsilon_{0}\left(\epsilon_{\perp}+\Delta \epsilon \sin ^{2} \theta\right)}, \quad D_{3}=\frac{-\epsilon_{0} V+E_{13} \int_{0}^{d} \frac{\sin (2 \theta) \frac{f_{1}}{\rho_{1}+\Delta \epsilon \sin ^{2} \theta} d z}{\int_{0}^{d} \frac{1}{\epsilon_{1}+\Delta \epsilon \sin ^{2} \theta} d z} .}{} . \tag{13.2}
\end{equation*}
$$

Note that

$$
\begin{align*}
& \frac{\partial E}{\partial z}=\frac{\left(\frac{\partial \theta}{\partial z}\right)^{2} \Delta \epsilon E_{13} \sin ^{2}(2 \theta)}{\epsilon_{0}\left(\epsilon_{\perp} \cos ^{2} \theta+\epsilon_{\|} \sin ^{2} \theta\right)^{2}}-\frac{D_{3} \Delta \epsilon \frac{\partial \theta}{\partial_{2}} \sin (2 \theta)}{\epsilon_{0}\left(\epsilon_{\perp} \cos ^{2} \theta+\epsilon_{\|} \sin ^{2} \theta\right)^{2}} \\
&  \tag{B.3}\\
& \quad-\frac{E_{13} \frac{\partial^{2} \theta}{\partial \partial^{2}} \sin (2 \theta)}{\epsilon_{0}\left(\epsilon_{\perp} \cos ^{2} \theta+\epsilon_{\|} \sin ^{2} \theta\right)}-\frac{2 E_{13}\left(\frac{\partial \theta}{\partial_{2}}\right)^{2} \cos (2 \theta)}{\epsilon_{0}\left(\epsilon_{\perp} \cos ^{2} \theta+\epsilon_{\|} \sin ^{2} \theta\right)},
\end{align*}
$$

and

$$
\begin{equation*}
E^{2}=\frac{D_{3}^{2}-2 D_{3} E_{13} \sin (2 \theta) \frac{\partial \theta}{\partial z}+E_{13}^{2} \sin ^{2}(2 \theta) \frac{\partial \theta}{\partial_{2}}}{\epsilon_{0}^{2}\left(\epsilon_{\perp} \cos ^{2} \theta+\epsilon_{\|} \sin ^{2} \theta\right)^{2}} \tag{B.4}
\end{equation*}
$$

Therefore, upon substituting equations (B.3) and (B.4) into (B.1) and collecting together like terms, we have

$$
\begin{equation*}
\gamma_{1} \frac{\partial \theta}{\partial t}=m(\theta)+n(\theta)+p(\theta)+q(\theta)+r(\theta)+s(\theta) \tag{B.5}
\end{equation*}
$$

where

$$
\begin{aligned}
& m(\theta)=\left(K_{1} \cos ^{2} \theta+K_{3} \sin ^{2} \theta\right) \frac{\partial^{2} \theta}{\partial z^{2}}, \quad n(\theta)=\frac{\Delta \epsilon D_{3}^{2} \sin (2 \theta)}{2 \epsilon_{0}\left(\epsilon_{\perp} \cos ^{2} \theta+\epsilon_{1} \sin ^{2} \theta\right)^{2}}, \\
& p(\theta)=\frac{1}{2}\left(K_{3}-K_{1}\right)\left(\frac{\partial \theta}{\partial z}\right)^{2} \sin (2 \theta), \quad q(\theta)=-\frac{\left(\frac{\partial \theta}{\partial z}\right)^{2} \Delta c E_{13}^{2} \sin ^{3}(2 \theta)}{2 \epsilon_{0}\left(\epsilon_{\perp} \cos ^{2} \theta+\epsilon_{11} \sin ^{2} \theta\right)^{2}}, \\
& r(\theta)=\frac{E_{13}^{2} \frac{\partial^{2} \theta}{\partial z^{2}} \sin ^{2}(2 \theta)}{\epsilon_{0}\left(\epsilon_{\perp} \cos ^{2} \theta+\epsilon_{\|} \sin ^{2} \theta\right)}, \quad s(\theta)=\frac{E_{13}^{2}\left(\frac{\partial A}{A_{2}}\right)^{2} \sin (4 \theta)}{\epsilon_{0}\left(\epsilon_{\perp} \cos ^{2} \theta+\epsilon_{1} \sin ^{2} \theta\right)} .
\end{aligned}
$$

Our initial condition for a HAN cell is a lincar profile from $\theta(0)=\alpha_{1}$ to $\theta(d)=\pi / 2 \mathrm{rud}$ therefore, initially, $\frac{\partial^{2} \theta}{\partial z^{2}}=0$. We first consider the simple case in which $K_{1}=K_{3}$ and $\Delta r=0$. In this case,

$$
m(\theta)=n(\theta)=p(\theta)=q(\theta)=r(\theta)=0
$$

whilst

$$
s(\theta) \sim c(\theta) \sin (4 \theta(z))
$$

where $c(\theta)>0 \forall \theta(z)$. This means that

$$
\frac{\partial \theta}{\partial t} \sim c_{1}(\theta) \sin (4 \theta)
$$

with $c_{1}(\theta(z))>0 \forall z$, so $\theta$ will increase for $0<z<d / 2$ and decrease for $d / 2<z<d$ as in Figure B.1(b), causing a nonlinear director angle solution.
(a) Initial Condition

(b) Flexoelectric Distortion


Figure B.1: (a) The initial condition for a HAN state, and (b) the form of the dominant flexoelectric tern.

Examining each term in equation (B.5) when $\Delta \epsilon \neq 0, K_{1} \neq K_{3}$,

- $m(\theta) \sim c_{2}(\theta) \frac{\partial^{2} \theta}{\partial z^{2}}, c_{2}(\theta) \geq 0 \forall \theta(z)$, which tends to make $\theta(z)$ linear. Since we begin with a lincar profile in $\theta(z), \frac{\partial^{2} \theta}{\partial z^{2}}$ will remain small.
- $n(\theta) \sim c_{3}(\theta) \sin (2 \theta), c_{3}(\theta) \neq 0 \forall \theta(z)$ always has either a completely positive effect or a completely negative effect on $\theta(z)$ across the entire cell depending on $\Delta c$.
- $p(\theta) \sim c_{4}(\theta) \sin (2 \theta), c_{4}(\theta) \neq 0 \forall \theta(z)$, always has either a completely positive effect or a completely negative effect on $\theta(z)$ across the entire cell depending on $k_{1}$ and $k_{3}$.
- $q(\theta) \sim c_{5}(\theta) \sin ^{3}(\theta), c_{5}(\theta) \neq 0 \forall \theta(z)$, always has either a completely positive cifert or a completely negative effect on $\theta(z)$ across the entire cell depending on $\Delta c$.
- $r(\theta) \sim c_{7}(\theta) \frac{\partial^{2} \theta}{\partial z^{2}} \sin ^{2}(2 \theta), c_{7}(\theta) \geq 0 \forall \theta(z)$, so $s(\theta)$ tends to make $\theta(z)$ lincar. Since we begin with a linear profile in $\theta(z), \frac{\partial^{2} \theta}{\partial z^{2}}$ will remain small.
- $s(\theta) \sim c_{8}(\theta) \sin (4 \theta), c_{8}(\theta) \geq 0$, meaning that $t(\theta)$ will cause an ' $s$ '-shape that increases $\theta(z)$ for $0<z<d / 2$ whilst $\theta$ decreases for $d / 2<z<d$.

So we conclude that $s(\theta)$ dominates, and any ' $s$ '-shapes will always be similar to that in Figure $5.16(\mathrm{~A})$. Furthermore, $s(\theta)$ increases quadratically with $E_{13}$ thereby pronouncing the kink in the relaxed HAN state.

## References

[1] The Zenithal Bistable device: A liquid crystal with ultra-low power consumption, liokuyo Hall, Shinagawa, Tokyo Japan, 2000. ASET International Forum on Low Jower Dis. plays.
[2] T. J. Atherton and J. R. Sambles. Orientational transition in a nematic liquid crysend at a patterned surface. Physical Review E, 74(2):022701, 2006.
[3] G. Barbero and I. Lelidis. Modulated structures of flexoelectric origin in nematic liquid crystals. Physcal Review E, 67(6):061708, 2003.
[4] JL Barrat and L. Bocquet. Large slip effect at a nonwetting fluid-solid interface. Physical Review Letters, 82(23):4671-4674, 1099.
[5] C. V Brown, G. P. Bryan-Brown, and V. C. Hui. Modelling nematic liquid crystal alignment on asymmetric surface grating structures. Molecular Crystals and Liquid Crystals, 301(1):163-168, 1997.
[6] G. P. Bryan-Brown, C. V. Brown, I. C. Sage, V. C. Hui, and E. L. Wood. Wide viewing-angle LCD using voltage-controlled twist effect. SID Digest, XXIX:ī10-i13, 1998.
[7] G. P. Bryan-Brown, E. L. Wood, C. V. Brown, J. C. Jones, I. C. Sage, and P. Brett. Grating aligned bistable nematic devices. IDW'97, Defence Evaluation and Rescarch Agency, pages 261-264, 1997.
[8] L. J. Cummings and G. Richardson. Bistable nematic liquid crystal device with fexoelectric switching. European Journal of Applied Mathematics, 17(4):435-163, 2006.
[9] A. J. Davidson. Flexoelectric switching in a bistable nematic dctice. PhD Thosis, University of Strathclyde, Glasgow, UK, 2004.
[10] A. J. Davidson and N. J. Mottram. Flexoelectric switching in a bistable nematic device. Phys. Rev. E, 65(5):051710, 2002.
[11] P. G. de Gennes and J. Prost. The physics of liquid crystals. Oxford University Prew, Oxford, 1993.
[12] E. A. Dorfi and L. O. Drury. Simple adaptive grids for $1-\mathrm{d}$ initial value problems. Journal of Computational Physics, 69(1):175-195, 1987.
[13] F. A. Fernández, S. E. Day, P. Trwoga, H. F. Deng, and R. James. Threwdimennional modelling of liquid crystal display cells using finite elements. Molecular Crystals and Liquid Crystals, 375(1):291-299, 2002.
[14] F. C. Frank. On the theory of liquid crystals. Discussions of the Faraday Socicly, 25: 19-28, 1958.
[15] R. M. Furzeland, J. G. Verwer, and P. A. Zegeling. A numerical study of three moving. grid methods for one-dimensional partial differential equations which are based on the method of lines. Journal of Computational Physics, 89(2):349-388, 1990.
[16] E. Hairer, S. P. Nørsett, and G. Wanner. Solving ordinary differential equations, pages 167-168. Springer, 1993.
[17] L. Harnau, S. Kondrat, and A. Poniewierski. Phase behavior of a nematic liquid crysend in contact with a chemically and geometrically structured substrate. Physical Recicw E, 72(1):011701, 2005.
[18] L. Harnau, S. Kondrat, and A. Poniewierski. Effective free-energy method for nematic liquid crystals in contact with structured substrates. Physical Review E, 76(5):051701, 2007.
[19] W. Huang, Y. Ren, and R. D. Russell. Moving mesh methods based on moving mesh partial differential equations. Journal of Computational Physics, 113(2):279-290, 1091.
[20] R. James, M. C. Gardner, F. A. Fernández, and S. E. Day. 3D modelling of high resolution devices. Molecular Crystals and Liquid Crystals, $450: 105=[305]-118=[318]$, 2006.
[21] J. C. Jones, E. L. Wood, and G. P. Bryan-Brown. Bistable nematic liquid crystal device. Defence Evaluation \& Research Agency, 1999. Patent Number: WO/1999/034251; Filing Date: 16/12/1998.
[22] P. J. Kedney and F. M. Leslie. Switching in a simple bistable nematic cell. Liquid Crystals, 24(4):613-618, 1998.
[23] H. Kelker. History of liquid crystals. Molecular Crystals and Liquid Crystals, 21(1): 1-48, 1973.
[24] S. Kitson and A. Geisow. Bistable alignment of nematic liquid crystals around microscopic posts. Molecular Crystals and Liquid Crystals, 412(1):153-161, 2004.
[25] S. C. Kitson, N. Carter, and M. B. Cindercy. Liquid crystal derice and compenitions. Hewlett-Packard Company, 2003. Patent Number: WO/2003/0.16110; Filing Date: 19/11/2002.
[26] E. Lauga, M. P. Brenner, and H. A. Stone. Springer Handbook of Erperimental Fluid Mechanics. Springer, 2007.
[27] F. M. Leslie. Continuum theory for nematic liquid crystals. Continuum Mcchanics and Thermodynamics, 4:167-175, 1992.
[28] D. R. Link, M. Nakata, Y. Takanishi, K. Ishikawa, and H. Takezoc. Flexoclectric polarization in hybrid nematic films. Physcal Review E, 65(1):010701, 2001.
[29] J. D. Maldonado and F. X. Timmes. A cyclic pentadiagonal solver. http://cococubed.asu.edu.
[30] W. R. Mekwi. Private Communication, 2006.
[31] W. R. Mekwi. Adaptive solution of partial differential equations using mesh movement and refinement. PhD Thesis, University of Strathclyde, Glasgow, UK, 2006.
[32] R. B. Meyer. Piezoelectric effects in liquid crystals. Physical Revicw Letters, 22(18): 918-921, 1969.
[33] N. J. Mottram and C. J. P. Newton. Introduction to Q-tensor theory: University of Strathclyde Mathematics Research Report 10, University of Strathclyde, 2004.
[34] C. J. P. Newton. Private Communication, 2005.
[35] C. J. P. Newton and T. P. Spiller. Bistable nematic liquid-crystal device modelling. In Proceedings of International Display Research Conference, pages 13-16, 1997.
[36] C. J. P. Newton and T. P. Spiller. A novel approach to modelling nematic liquid crystal cells. HP Laboratories Bristol 109, Printing and Imaging Technologies Laborntory, 2000.
[37] L. A. Parry-Jones, E. G. Edwards, S. J. Elston, and C. V. Brown. Zenithal bistability in a nematic liquid-crystal device with a monostable surface condition. Appliod Physics Letters, 82(9):1476-1478, 2003.
[38] S. Ponti, P. Ziherl, C. Ferrero, and S. Žumer. Flexo-electro-optic effect in a hybrid nematic liquid crystal cell. Journal of Liquid Crystals, 20(8):1171-1177, 1999.
[39] T. Qian and P. Sheng. Generalized hydrodynamic equations for nematic liquid crystals. Physical Review E, 58(6):7475-7485, 1998.
[40] F. Reinitzer. Contributions to the knowledge of cholesterol. Liquid Crystals, 5(1):7-18, 1989.
[41] J. M. Sanz-Serna and I. Christie. A simple adaptive technique for nonlinear wave problems. Journal of Computational Physics, 67(2):348-360, 1986.
[42] I. W. Stewart. The static and dynamic continuum theory of liquid crystals: A mathc. matical introduction. Taylor and Francis, 2004.
[43] Various. To check if two line segments intersect. http://groups.google.com, 2000.
[44] E. W. Weisstcin. Courant-Friedrichs-Lewy condition. MathWorld - A Wolfram Web Resource.

