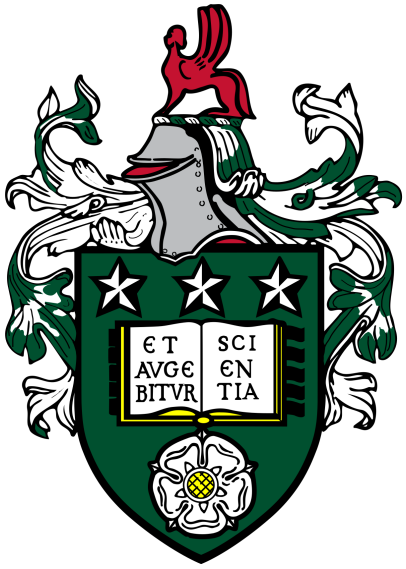
Computational Fluid Dynamics of Polymer Flow Induced Crystallisation

William Paul Grant



Submitted in accordance with the requirements for the degree of Doctor of Philosophy

The University of Leeds
School of Chemical and Process Engineering
School of Mathematics

October, 2024

Intellectual Property

I confirm that the work submitted is my own and that appropriate credit has been given where reference has been made to the work of others.

This copy has been supplied on the understanding that it is copyright material and that no quotation from the thesis may be published without proper acknowledgement.

The right of William Paul Grant to be identified as Author of this work has been asserted by William Paul Grant in accordance with the Copyright, Designs and Patents Act 1988.

Acknowledgements

The experience that I have gained from my PhD journey has been invaluable and unforgettable. I would like to thank my supervisors Dr. Oliver Harlen and Prof. Daniel Read for supporting me throughout my PhD over the last four years. They have shown patience and offered guidance at every turn and have always kept me motivated throughout my studies and I will always be grateful.

This PhD project would not have been possible without the financial support of the EPSRC Centre for Doctoral Training in Molecules to Product (Grant Ref. No. EP/S022473/1). I would like to thank all members of staff and cohort members that I have met throughout this process.

I would like to thank all my family and friends for all the amazing times we have had together over the last four years.

Lastly, for anyone involved directly or indirectly in this process, thank you.

Abstract

This thesis models flow induced crystallisation (FIC) in polymers using the Rolie-Double-Poly (RDP) model [1] in combination with the polySTRAND model [2] to create a computationally efficient method for modelling FIC. The RDP model incorporates interactions between different chain species in a polydisperse melt. The polySTRAND model is a computationally efficient FIC model that provides a molecular basis for the formation of crystals in a polymer melt.

The RDP and polySTRAND models are validated against published results and are implemented in OpenFOAM [3] using the RheoTool library [4]. We then simulate an idealised bimodal polymer blend in a channel to demonstrate the successful implementation of the RDP model. We then take inspiration from the experimental work done by Scelsi et al [5] and simulate flow through a contraction expansion geometry that has the same dimensions as in their experiments. We then compared the flow induced crystallisation effects in the flow where only one chain species contributes to the acceleration of crystallisation and compared our results to the Scelsi experiments.

The study of the contraction expansion flow is then extended to include multiple chain species having an effect on the acceleration of crystallisation in flow. This required three extensions to the previous chapter. We show how a general N mode polydisperse RDP model can be implemented in OpenFOAM. We then construct a procedure for choosing relaxation times in the RDP model to fit with rheological data. Finally we modify the polySTRAND model calculation to account for contributions from multiple chain species to FIC. With all of this in place we are able to make a more meaningful comparison to the experimental investigation of Scelsi et al [5].

We further extended our framework to include the effects of temperature on the rheology and the crystallisation dynamics in the system. We need to make modifications to our framework to account for how temperature affects the growth rate of crystals, relaxation times, viscosity and the energy barrier to crystallisation. We first show how cooling a polymer melt after a shear pulse can affect the FIC properties in a channel geometry, following a methodology that is commonly used in the literature [6]. Finally we reconsider our contraction expansion geometry and our polydisperse melt and simulate flow through our geometry when the walls are either cold or hot relative to the melt. We then compare these results to the isothermal polydisperse simulations that we conducted earlier. We show that by controlling the temperature distribution it is possible to change the spatial distribution of crystallites.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Scope of the Thesis

Modelling the flow of polymers is important for controlling industrial polymer processing. Predicting effectively how polymeric fluids behave under processing conditions can help improve the efficiency of the processes and reduce waste. Constitutive models have been developed for predicting the flow of polymers that take account of the molecular scale dynamics and can be implemented in computational fluid dynamics (CFD) software. Two such models discussed in this thesis are the Rolie-Poly [11] (RP) and Rolie-Double-Poly [1] (RDP) models.

The RP model uses reptation and tube theory [12] to model the flow of a monodisperse linear entangled polymer, in which a flexible polymer chain is free to move along its length, but its lateral motion is constrained by neighbouring chains. Industrial polymers are polydisperse, the Rolie-Double-Poly (RDP) model builds on double reptation theory [13] to account for the interactions between chains of different lengths that are not included in the original RP model. Both of these models were implemented in OpenFOAM [3] CFD program in the PhD thesis by Adila Aida Binti Azahar [7], although the RDP model was only implemented at the two-component level.

As deformation increases, polymer chains align and stretch. In crystallising polymers this has an important effect on the rate of crystallisation. When the polymer strands are elongated and aligned under flow, the rate of nucleation of crystals is increased. This increased rate of crystallisation within the polymer affects both the flow properties of the melt and the mechanical properties of the product. Molecular understanding of flow induced crystallisation (FIC) would allow for better control of the crystallisation of polymers during processing. This is desirable as the crystal properties of polymers affect the majority of polymer properties of practical interest e.g. strength, transparency etc. Therefore processing conditions can be tailored to suit the desired crystal morphology.

A model that considers the effect of polymer stretch on the nucleation dynamics of polydisperse polymers is the PolySTRAND model [2]. This model calculates the reduction in the energy barrier to crystallisation as a result of the flow induced changes to the

molecular configuration to predict the acceleration of the rate of crystallisation. However, to provide a full theory for flow induced crystallisation this model needs to be coupled to a calculation of the molecular configuration induced by the flow, using a constitutive model such as the RP or RDP model, which in turn will be affected by the formation of crystals.

The aim of this thesis is to combine the PolySTRAND model for crystallisation with the Rolie-Double-Poly constitutive model within the CFD software package OpenFOAM. This would allow for simulations of FIC in complex flow geometries using a model for FIC that has molecular detail included. With this in mind, we will look to investigate the effect of polydispersity on FIC and how temperature plays a role in this problem. A list of the goals of the thesis is as follows:

- To implement the RDP model in OpenFOAM and to validate this against published results in a simple flow.
- To implement and validate the PolySTRAND model against results given in the literature.
- To investigate FIC in a contraction-expansion geometry of an idealised bimodal polymer blend to qualitatively match experimental results.
- To investigate FIC in a polydisperse polymer, in a contraction-expansion geometry, that is a reduced ensemble of an experimental investigation.
- To investigate the effects of temperature and heat flow on FIC in a complex flow.

1.2 Polymers

A polymer is a molecule comprised of repeated sub-units called monomers that are joined together by covalent bonds to form large macro molecules. The structure of polymers is crucial in understanding their properties. Polymeric materials are made up of these long flexible polymers. Some polymeric materials bend and stretch e.g. rubber, while others are hard and tough e.g. epoxies. The term plastic is generally used to describe synthetic polymers with additives added. Natural polymers also exist; protein is an example of a biopolymer that is made up of amino acids joined together by peptide bonds [14]. A polymer that is comprised of a single chain of repeating sub-units is known as a linear polymer. Early modelling of polymers focused on linear polymers (owing to their structural simplicity), however, not all polymers are linear, as some industrially important polymers have branches, e.g. low density polyethylene, which affect the properties of polymeric materials.

The length of the polymer chains can also vary. When all the polymer chains in a melt have the same length within reasonable bounds the polymer is said to be monodisperse. Otherwise they are known as polydisperse. As might be expected, modelling the flow

of monodisperse polymers is a less complicated task than modelling the flow of polydisperse polymers. However all industrially relevant polymers are polydisperse such that the molecules range over a spectrum of different molecular weights. Almost no polymers are perfectly monodisperse: as most materials labelled as monodisperse still have a narrow distribution of different molecular weights.

There is a broad universality to the flow of polymeric materials. It often does not matter if the polymer is made up of amino acids or hydrogen and carbon atoms: many of the phenomena of polymeric flow are independent of the local monomeric detail. Once a few chemistry-specific parameters are accounted for (affecting, for example, the overall stress and time scalings) the topology of the molecules is what affects the flow properties of these materials. A polymer that has a large amount of branching will behave differently to one that has none. The flexible nature of the polymer molecules means that they can get into configurations that are far from their equilibrium structure and even entangle with other molecules in a polymer melt. These are examples of what needs to be considered when modelling the flows of polymers and this is what makes modelling polymer flow a complex and vastly studied problem.

1.3 Basic Concepts of Rheology

Rheology is a branch of natural science defined as the study of the flow and deformation of matter [15]. Rheology encompasses both purely viscous and purely elastic behaviour, but generally focuses on situations where both viscous and elastic properties are present. This is achieved by understanding the stress within a material and how this relates to its history of deformation.

Industrial polymer processing involves many complex flow properties. In particular, industrial polymers experience a large amount of shear deformation. As a result shear flow of polymeric fluids has been extensively studied. A way of achieving a shear flow is by placing the polymer between two plates a distance h apart and moving the the upper plate a velocity V in a direction parallel to the plane of the plates. In the absence of flow instabilities or wall slip the polymer velocity u at height y above the stationary plate is Vy/h. Therefore, the velocity gradient $\frac{\partial u}{\partial y}$, otherwise known as the shear rate $\dot{\gamma}$, is V/h. Clearly, this is a particularly idealised flow and is only one factor in any complex flow pattern. This does though provide a good theoretical starting point for more complicated flows.

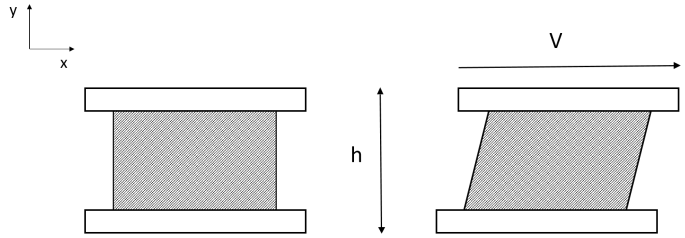


Figure 1.1: Schematic of the idealised simple shear experiment setup.

The force **F** required to move the upper plate must be proportional to its surface area A. Newton postulated that this force is in the direction of motion given by the relation

$$\mathbf{F} = \eta A\dot{\gamma},$$

where η is a constant that is known as the viscosity and depends only on the fluid properties. By defining the stress, σ, as the force per unit area the relationship becomes

$$\sigma = \eta\dot{\gamma}. \tag{1.1}$$

Fluids that obey this relationship are known as Newtonian fluids. The viscosity of Newtonian fluids is independent of the stresses applied to it. Examples of these kinds of fluids are water and air. Conversely any fluid that does not obey this law is a non-Newtonian fluid, of which there are many different classes. Fluids containing polymers are non-Newtonian fluids owing to the presence of the long molecular chains.

An elastic solid obeys a relationship where the stress on the material depends on the deformation from its initial shape. Hooke's law, equation 1.2 gives the stress relationship for a linear elastic solid that is typically only valid for small deformations

$$\sigma = G\gamma, \tag{1.2}$$

where γ is the strain and G is the elastic modulus of the material, taken as constant.

The relationships between stress and strain shown here are two simple examples of constitutive equations of materials. Polymeric fluids have a much more complex constitutive relationship that includes both viscous and elastic behaviour. As a result the modelling of polymers becomes much more challenging.

1.3.1 Linear Viscoelasticity

The two constitutive relationships described by equations 1.1 and 1.2 outline idealised viscous fluid and elastic solid behaviour respectively. Many materials in real life are intermediate between an elastic solid and a viscous liquid.

Linear viscoelastic materials are those for which there is a linear relationship between stress and strain history (at all times). Linear viscoelasticity is a theory that describes the behaviour of such ideal materials. Linear viscoelasticity is a reasonable approximation to

the time-dependent behaviour of viscoelastic materials under relatively low strains.

In an effort to characterise these materials an idealised shear experiment can be considered similar to the previous section. A small shear deformation is applied at $t = 0$ such that $\gamma = \gamma_0$ for $t \geq 0$ and $\gamma = 0$ otherwise. The shear rate is therefore given by

$$\dot{\gamma} = \gamma_0 \delta(t), \tag{1.3}$$

where $\delta(t)$ is the Dirac delta function. The stress response will depend on the time following the deformation in general, since viscoelastic materials have a "memory" of past deformations. For small values of $\dot{\gamma}$ the scalar shear stress response, defined as $\sigma(t)$, is expected to be linear in $\dot{\gamma}$ and hence the linear relaxation modulus $G(t)$ is given as,

$$G(t) = \frac{\sigma(t)}{\gamma_0}. \tag{1.4}$$

To demonstrate how the relaxation modulus can be calculated for this experiment the linear Maxwell model [16] will be used. Maxwell derived this model by combining the response of a Newtonian fluid and an elastic solid in series. The Newtonian fluid is modelled as a dashpot and the elastic solid is represented by a Hookean spring. For this reason this model is also known as the dashpot and spring model. An illustration of the linear Maxwell model is given in figure 1.2.

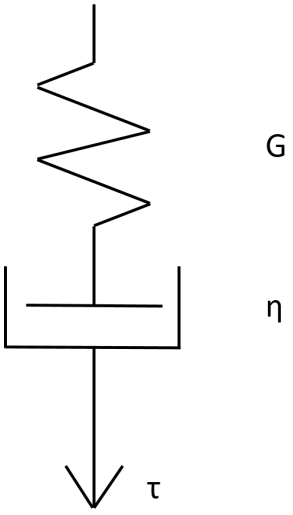


Figure 1.2: Illustration of the linear dashpot and spring Maxwell model.

Consider a deformation applied on a fluid that obeys the Maxwell model. The elastic stress on the spring, σ_s, and the viscous stress on the dashpot, σ_d, are the same, whilst the total strain on the system,γ is the sum of the individual strains. Therefore,

$$\sigma = \sigma_s = G\gamma_s \qquad \text{and} \qquad \sigma = \sigma_d = \eta\dot{\gamma}_d, \tag{1.5}$$

$$\gamma = \gamma_s + \gamma_d. \tag{1.6}$$

The constitutive law for the Maxwell model can be obtained by differentiating equation 1.6 and combining it with equation 1.5 like so,

$$\dot{\gamma} = \frac{1}{G}\frac{\mathrm{d}\sigma}{\mathrm{d}t} + \frac{1}{\eta}\sigma$$
$$\sigma + \tau\dot{\sigma} = \eta\dot{\gamma}, \tag{1.7}$$

where $\tau = \eta/G$ is the relaxation time of the material. This linear differential equation is solved using an integrating factor to give,

$$\sigma = \int_{-\infty}^{t} \frac{\eta}{\tau}\exp\left(-\frac{t-t'}{\tau}\right)\dot{\gamma}(t')\,\mathrm{d}t' = \int G(t-t')\dot{\gamma}(t')\,\mathrm{d}t', \tag{1.8}$$

where $G(t)$ is the relaxation modulus for the Maxwell model when $t > 0$ in a simple shear experiment and is given by

$$G(t) = \frac{\eta}{\tau}\exp\left(-\frac{t}{\tau}\right) = G\exp\left(-\frac{t}{\tau}\right). \tag{1.9}$$

Therefore the relaxation time τ gives the characteristic time scale for the decay of the relaxation modulus.

Another way of looking at how the stress evolves within this linear Maxwell fluid is to consider the stress response to a constant shear rate $\dot{\gamma}_0$ imposed for $t > 0$:

$$\dot{\gamma} = \begin{cases} \dot{\gamma}_0 & t > 0 \\ 0 & t < 0 \end{cases}.$$

Substituting this into equation 1.8 we have

$$\sigma_{xy} = \dot{\gamma}_0 \int_0^t \frac{\eta}{\tau}\exp\left(-\frac{t-t'}{\tau}\right)\,\mathrm{d}t' = \dot{\gamma}_0\eta\left(1-\exp\left(-\frac{t}{\tau}\right)\right).$$

Therefore, it takes a time of order τ for the shear stress to reach its equilibrium value of $\dot{\gamma}_0\eta$. The shape of this relationship is sketched in figure 1.3.

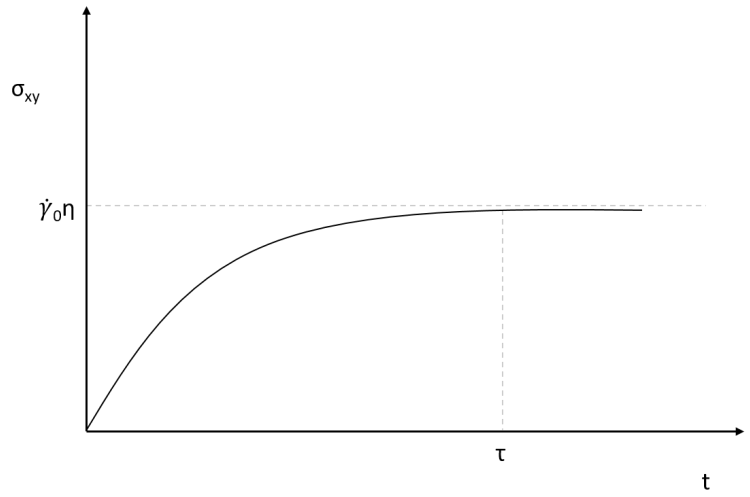


Figure 1.3: Sketch of the evolution of the shear stress of a linear Maxwell fluid subjected to a start-up of steady shear flow.

This linear viscoelastic model can be generalised to any three dimensional flow to give the integral form of the stress for a linear viscoelastic fluid as

$$\underline{\underline{\boldsymbol{\sigma}}}(t) = 2\int_{-\infty}^{t} G(t-t')\underline{\underline{\mathbf{E}}}(t')\,\mathrm{d}t', \tag{1.10}$$

where $\underline{\underline{\mathbf{E}}} = 1/2(\nabla\mathbf{u} + \nabla\mathbf{u}^T)$ is the strain rate tensor. The above integral is referred to as the general linear viscoelastic model as it is the most general linear form of the constitutive equation for an incompressible isotropic fluid. An incompressible fluid refers to a fluid where the density remains constant throughout, while an isotropic fluid refers to a fluid where the fluid properties are not dependent on the direction along which they are measured. Note that the linear Maxwell model can be easily recovered from equation 1.10 by using equation 1.9 for the relaxation function. For most real materials the relaxation function $G(t)$ is more complex than the single exponential form of the Maxwell model, but can almost always be modelled as a sum of exponential terms, with different weights and relaxations times. Such models are referred to as "multimode Maxwell models".

The Deborah number gives a measure of the importance of viscoelasticity and is defined as,

$$\mathrm{De} = \frac{\tau}{T},$$

where τ is the material relaxation time defined earlier and T is the timescale on which the velocity gradient varies. A Newtonian fluid would have $\mathrm{De} = 0$ because $\tau = 0$ in this case. A value of $\mathrm{De} = \infty$ represents an elastic solid because $\tau \to \infty$.

1.3.2 Nonlinear Viscoelasticity

The modelling considerations in the previous section only hold for the linear viscoelastic regime i.e. where the total strain and/or strain rate on the material is sufficiently small that the response is linear. In general the relationship between stress and strain rate history is nonlinear for large deformations such as steady shear. As a result different considerations within constitutive equations need to be made to account for this behavaiour.

The Weissenberg number given by,

$$\mathrm{Wi} = \tau\dot{\gamma},$$

where τ is as above and $\dot{\gamma}$ is the shear rate gives a measure of the importance of nonlinearity [17].

Capturing both the nonlinear and viscoelastic effects of a real polymeric fluid requires more sophisticated models. Oldroyd [18] was one of the first to develop a model for the general flow of nonlinear viscoelastic materials. Oldroyd didn't have the detailed physics in mind, he wanted a generalisation of the Maxwell model that satisfied material frame invariance. The Oldroyd model is widely used for polymer solutions but has the drawback that the stress in extensional flow is unbounded. The FENE and FENE-P models [19] were derived to rectify this issue with the Oldroyd model for unentangled polymers by considering a finitely extensible chain.

For polymer melts the Graham–Likhtman and Milner–McLeish (GLaMM) model [20] uses tube theory to capture the linear and nonlinear behaviour of linear entangled polymers. Graham and Likhtman also developed a simplified version of this model called the Rolie-Poly model [11], which utilises tube and reptation theory to describe the behaviour of monodisperse linear entangled polymers. More details on specific polymer types along with tube and reptation theory are given in section 1.5.1. Very recently, the Rolie-Double-Poly model was developed by Boudara et al. [1] utilising what is known as double reptation theory to extend the Rolie-Poly model to polydisperse linear entangled polymers.

The RP and RDP models are appropriate for linear polymers. For branched polymers there are additional constraints on the polymer motion due to branch points. The Pom-Pom model [21] was conceived to provide a simple model that incorporates the molecular physics arising from branching in polymer chains. A variant of the Pom-Pom model called the extended Pom-Pom model [22] is now widely used to model the rheology of branched polymer melts. These models are just a few examples drawn from a large number of polymeric constitutive models in the literature.

1.4 Basic Concepts of Polymer Crystallisation

1.4.1 Semi-Crystalline Polymers

Many polymers are semi-crystalline [23] where they contain a combination of ordered crystalline and disordered amorphous regions [24]. Inside crystalline regions polymer chains are ordered into structures known as lamellae. Semi-crystalline polymers account for the majority of the production of synthetic polymers [25].

Long molecules can bridge from one crystal to another via the amorphous phase. These molecules are known as tie molecules and give the polymer toughness, otherwise these polymers would be too brittle for commercial applications [26]. The fraction of amorphous regions compared to crystalline regions, along with the arrangement and orientation of the crystals strongly influence the practical properties of the polymer e.g. strength, transparency and flexibility.

Various different formations of lamellae are possible and a common one is where lamellae are grouped together and radiate from a single point. This creates a structure known as a spherulite [27]. Spherulites are a spherical form of crystals and are commonly observed in the absence of flow i.e. quiescent conditions.

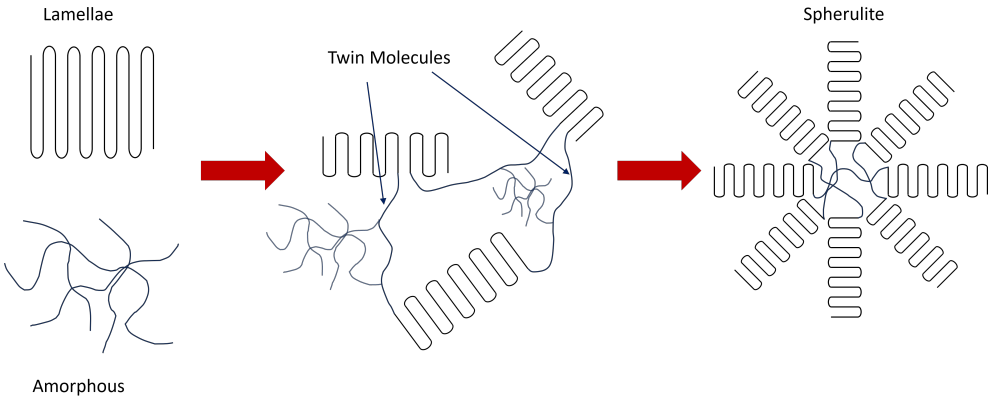


Figure 1.4: Figure showing (from left to right) an ordered lamellar structure above amorphous region. How different lamellae can be grouped together via twin molecules. An example formation of lamellae, in this case a spherulite.

1.4.2 Nucleation and Growth of Crystals

Having introduced the general idea of semi-crystalline polymers and polymer crystals, we now discuss how these crystals form and develop. Crystallisation in polymers occurs in two phases. First there is a nucleation event followed by a growth phase [28]. A nucleation event refers to when the polymer chains fold over themselves to form the first lamellae, this is known as primary nucleation. After such nucleation events surrounding polymer chains move to the crystal site and crystallise further, this is known as secondary nucleation [29]. The centre of a spherulite is typically at the point of primary nucleation.

Nucleation that occurs within the interior of a material that has no foreign particles or pre-formed crystals nearby is known as homogeneous nucleation. Nucleation that occurs with other particles or crystals present is known as heterogeneous nucleation [30]. Inducing heterogeneous nucleation is much faster than homogeneous nucleation due to the reduction in the energy barrier [28]. A common experimental method to observe homogeneous nucleation is the droplet method [31, 32] where a polymer sample is subdivided into small droplets so that any impurities are restricted to a portion of the droplets. The remaining droplets are then free to nucleate free from impurities to measure homogeneous crystallisation behaviour in the sample. Homogeneous nucleation is generally not seen under practical conditions in the absence of flow or extreme supercooling [33].

1.4.3 Flow Induced Crystallisation of Polymers

Now that we have defined polymer crystallisation and semi-crystalline polymers, we can look at how flow affects crystallisation in polymers. Flow dramatically increases the crystal nucleation rate in polymers and so has a huge effect on the final crystal configuration. Flow changes the configuration of polymer chains, making them more aligned, and this reduces the barrier to crystallisation within the polymer. This effect is known as flow induced crystallisation (FIC). When the molecular segments are aligned, the entropy within the polymer chains is decreased, therefore, the free energy within the chain increases. In a nucleation event, there is an energy barrier to overcome, due to the loss of entropy of molecular segments forming the nucleus. However, alignment by the flow causes an increase in free energy within the polymer chain reducing the height of this energy barrier, which results in an increased nucleation rate [34].

Flow also affects the structure of crystals. As noted in section 1.4.1, under quiescent conditions polymer crystals form as spherulites. When weak flow is imposed, the nucleation rate increases, which increases the number of spherulites in the melt, moreover the spherulites align with the flow direction [35]. When flow rate is increased further, polymer chains can become drastically stretched and can form rod-like structures. These rod-like structures provide nucleation sites for radial growth of lamellae. This formation is known as the shish-kebab morphology [36], where the rod-like structure is the shish and the radially growing lamellae are the kebabs. An illustration of flow having this effect on crystal morphology is given in figure 1.5.

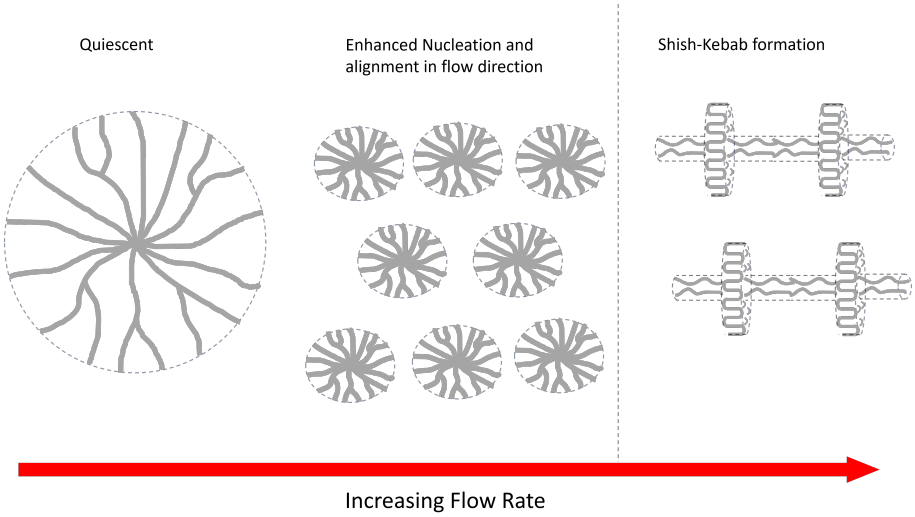


Figure 1.5: Illustration of how flow affects crystal nucleation and structure. The left shows spherulitic growth in quiescent conditions. In the centre you have enhanced nucleation with the spherulites aligning to the direction of imposed flow. On the right you have the shish-kebab structure of crystals.

The formation of shish-kebab structures has been investigated in many instances [35, 36, 37, 38]. The shish-kebab crystals demonstrate high strength in the orientation direction [39], so altering the crystal structures can affect the properties of the polymer. With this in mind the motivation to gain a molecular understanding of FIC increases.

FIC can be combined with heterogeneous nucleation techniques to control crystallisation in polymers. D'Haese et al. [40] found that adding 0.1 % wt. of zinc oxide particles to a melt of isotactic polypropylene (iPP) significantly lowered the timescale for crystallisation at low shear rates. However, at high shear rates, flow dominates the crystallisation kinetics in the melt regardless of the concentration of zinc oxide in the melt. This highlights the importance of understanding FIC since polymer processing conditions can involve high shear rates e.g. in injection moulding.

The challenge in modelling FIC lies in the huge range of both time and lengthscales that appear in the problem. The lengthscales range from the monomer size ($<$ nm) up to the spherulite size ($\sim \mu$m). The range of time scales is even larger from the monomer relaxation time ($\sim 10^{-9}$ s) to the quiescent nucleation timescale which can be hours at low under-cooling [25]. The vital physical properties come from rapid movement at the monomer level, while the phenomena that affect melt processing and control solid-state properties occur on much larger scales.

The huge range of length and time scales involved in this problem means that modelling crystallisation requires multiple levels of coarse graining so that molecular details can be captured, whilst being computationally tractable in CFD software. A general structure that works well for modelling FIC [41, 42] and will be the structure used in investigations in this thesis is as follows:

- Find the chain deformation under flow via a rheological model that provides a pre-

diction for polymer chain conformation.

- Use the chain deformation to compute the nucleation rate.
- Use the nucleation rate to calculate the evolution of the number density and the volume fraction of crystals in the polymeric material.

Rheological models have become much more sophisticated in recent times (the details of which will be discussed in the next section). This advancement allows for a more detailed picture of the deformation of polymer chains in flow. A more accurate prediction of the stretch and orientation of polymer chains allows for a better simulation of polymer crystal nucleation.

The prediction of the nucleation rate for FIC is particularly challenging. There have been many different techniques proposed to simulate the crystal nucleus growth, some of which have been reviewed in [25]. Particular attention will be placed on how the nucleation rate is computed from the results of the chain deformation from the rheological model.

There is an inherent difficulty in observing FIC directly in experiments. Current experiments calculate the nucleation rate from counting observable crystals through a microscope [25]. Whilst this does offer a good indication of the nucleation rate, this information is given a long time after the critical nucleation event. The formation of the critical nucleus is too small and too fast to be measured by current experimental techniques. Simulations of the nucleation events attempt to capture the behaviour in the polymer that experiments cannot.

1.5 Modelling Polymer Flow

Modelling polymer flow is important for optimising industrial processes. Effectively predicting how polymeric fluids behave under processing conditions can help to improve efficiency and reduce waste within industrial processes. This requires the ability to implement the solution of an appropriate constitutive model within computational fluid dynamics (CFD) packages.

The general physics of rheology is described using the laws of conservation of mass, energy and momentum along with the constitutive laws that relate the extra stress to the deformation of the fluid. For an isothermal, incompressible viscoelastic fluid, the conservation of mass and momentum equations are given as follows,

$$\nabla \cdot \mathbf{u} = 0, \tag{1.11}$$

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla p + \nabla \cdot \boldsymbol{\tau}. \tag{1.12}$$

Where ρ is the fluid density, $\mathbf{u}$ is the fluid velocity vector, p is the pressure, t is time and $\boldsymbol{\tau}$ is the total stress tensor excluding the isotropic pressure. Polymeric fluids (and polymer melts) can be thought of as incompressible, so equation 1.11 is the only form of

the continuity equation that is needed for modelling polymeric liquids. The complexity lies in the development of the constitutive laws for the polymeric fluids.

1.5.1 Polymer Types

So that we can describe the details of different constitutive laws of polymeric fluids, we need to outline the different types of polymers there are and the ways in which polymer chains relax stress within the material.

A linear polymer is made up of chains with a single backbone. A branched polymer has molecules that are comprised of a main chain and at least one side chain, otherwise known as branches.

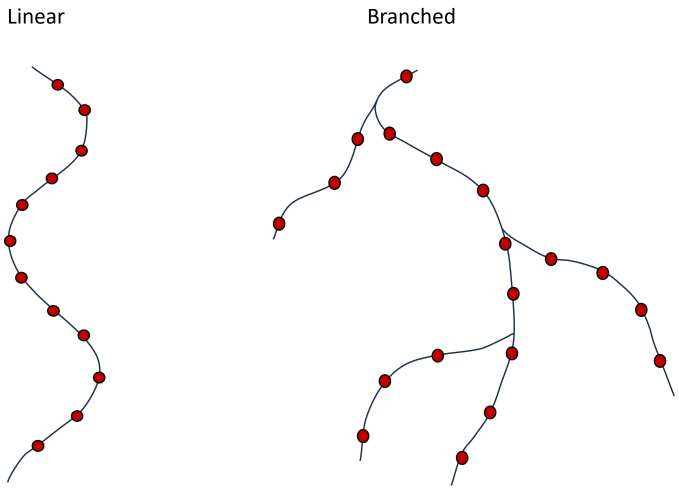


Figure 1.6: Sketch of a linear polymer molecule (left) next to a branched polymer molecule (right).

A monodisperse polymer has molecules that are all the same mass, whereas a polydisperse polymer contains molecules that have different masses and the distribution of these masses affect the properties of the polymer. Nearly all industrially relevant polymers are polydisperse, highlighting the need for constitutive laws to accurately incorporate polydispersity.

Long chain polymers in a melt can restrict each others' movement. Polymers that exhibit this behaviour are known as entangled polymers. In entangled melts the motion of individual polymer chains need to overcome these restrictions on themselves in order to relax to an equilibrium configuration, this is generally known as constraint release, and there are a few different methods of constraint release that are considered when modelling polymer flow.

An idea that was introduced by Pierre-Gilles de Gennes was that of the tube model for entangled polymers [12]. The concept being that polymer chains behave somewhat like cooked spaghetti, where each piece of spaghetti represents a polymer chain. Entanglements between polymer chains constrain lateral motion because that means it would cross other chains, but it can move along its length. It can be thought that the chain is confined to a

tube-like region of space, where the tube represents the entanglement constraints with the surrounding chains. The chain motion along the tube is called reptation. Reptation aims to describe the motion of polymer chains along their respective tubes, which allows them to escape their tube, and thus releasing some of the constraints of the tube surrounding them, this constraint release is known as thermal constraint release. Reptation occurs in both the presence and absence of flow. This motion also releases constraints on neighbouring chains.

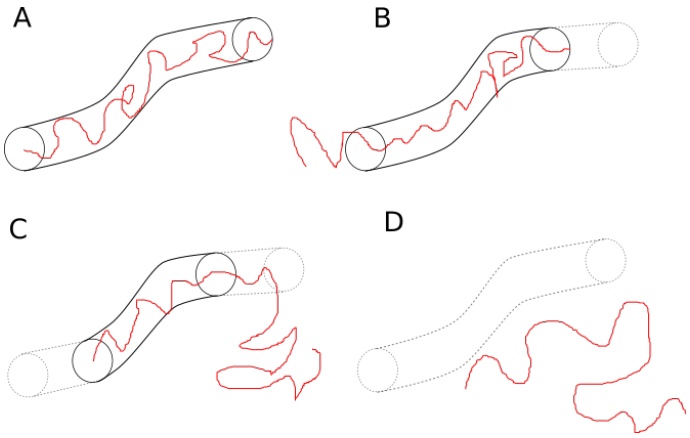


Figure 1.7: Illustration of reptation that evolves from A to D showing how a polymer chain (red) moves along its own length to release the constraint of the tube (black).

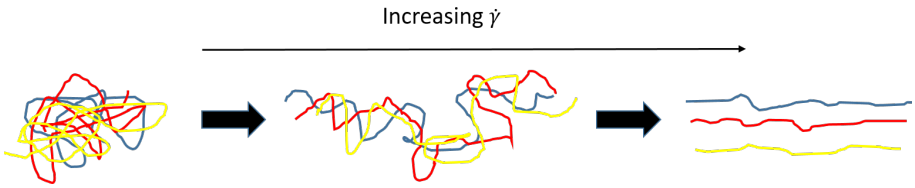


Figure 1.8: Diagram showing the effect of increasing shear rate on polymer chains.

When a polymer is subjected to flow conditions, polymer chains can become stretched (assuming the flow is strong enough). Polymer chains can then retract and release constraints from surrounding polymer chains via this process. Polymer chains and surrounding polymer chains that release constraints on themselves via this retraction is a process known as convective constraint release (CCR) [43, 1].

1.5.2 Rolie-Poly Model

The first specific polymer rheological model we will look at is the ROuse LInear Entangled POLYmer (Rolie-Poly) constitutive model developed by Graham and Likhtman [11]. This model is designed to capture the non-linear rheology of monodisperse linear entangled polymers. The Rolie-Poly (RP) model uses the tube theory concept of de Gennes [12] to model entanglements between polymer chains, and includes reptation, convective constraint release (CCR) [43] and chain stretch.

Rheological models describe the motion of polymer chain species in a melt via a conformation tensor (typically denoted as $\underline{\underline{\mathbf{A}}}$). The conformation tensor provides information about the average shape and orientation of the polymer chain. It can be defined in terms of the end-to-end vector sketched in figure 1.9.

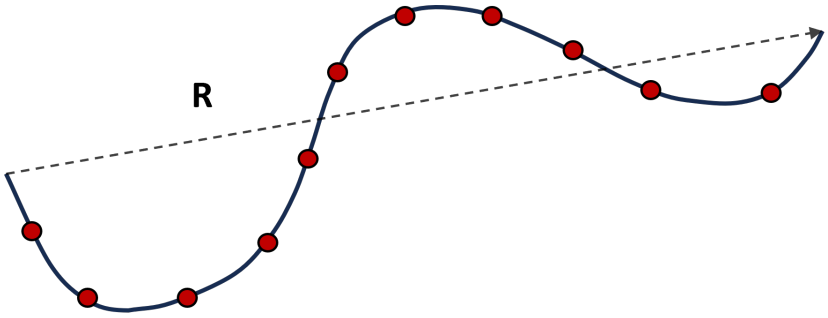


Figure 1.9: Sketch of the end-to-end vector of a polymer chain.

The conformation tensor can then be defined as:

$$\underline{\underline{\mathbf{A}}} = \langle \mathbf{R} \cdot \mathbf{R}^T \rangle, \tag{1.13}$$

where $\langle \cdot \rangle$ denotes an ensemble average over many configurations of the polymer chain and $\mathbf{R}^T$ represents the transpose of $\mathbf{R}$.

The incorporation of reptation within the Rolie-Poly model means that there are two distinct relaxation times within the Rolie-Poly constitutive equation. These are the disengagement time τ_d from reptation and the stretch relaxation time τ_s which accounts for the spring like behaviour of the polymer chains. The disengagement time should be larger than the stretch relaxation time for entangled polymer chains, with the limit $\tau_d \to \tau_s$ gives behaviour of an approximately unentangled polymer melt.

The constitutive equation for the single mode Rolie-Poly model in terms of the conformation tensor $\underline{\underline{\mathbf{A}}}$ is as follows:

$$\overset{\nabla}{\underline{\underline{\mathbf{A}}}} = -\frac{1}{\tau_d}(\underline{\underline{\mathbf{A}}} - \underline{\underline{\mathbf{I}}}) - \frac{2}{\tau_s}\left(1 - \frac{1}{\lambda}\right) f_E(\lambda) \left[\underline{\underline{\mathbf{A}}} + \beta_{CCR}\lambda^{2\delta}(\underline{\underline{\mathbf{A}}} - \underline{\underline{\mathbf{I}}})\right], \tag{1.14}$$

where λ is the chain stretch defined as

$$\lambda = \left(\frac{1}{3} tr \underline{\underline{\mathbf{A}}}\right)^{\frac{1}{2}}, \tag{1.15}$$

β_{CCR} is the convective constraint release coefficient, $\delta = -1/2$ is a fitting parameter and $\overset{\nabla}{\underline{\underline{\mathbf{A}}}}$ is the upper-convected time derivative of the conformation tensor defined as,

$$\overset{\nabla}{\underline{\underline{\mathbf{A}}}} = \frac{\partial \underline{\underline{\mathbf{A}}}}{\partial t} + \mathbf{u} \cdot \nabla \underline{\underline{\mathbf{A}}} - [\underline{\underline{\mathbf{K}}} \cdot \underline{\underline{\mathbf{A}}} + \underline{\underline{\mathbf{A}}} \cdot \underline{\underline{\mathbf{K}}}^T], \tag{1.16}$$

where $\underline{\underline{\mathbf{K}}} = \nabla \mathbf{u}$.

The function that represents finite extensibility is given as,

$$f_E(\lambda) = \frac{1 - \lambda_{max}^{-2}}{1 - \lambda^2 \lambda_{max}^{-2}}, \tag{1.17}$$

where λ_{max} is the maximum chain stretch in extensional flow.

The presence of two distinct relaxation times separates the shear rates at which the polymer chains become oriented and stretched under shear flow. As the polymer is sheared, the polymer chains begin to move along their "tubes" to release the constraints on neighbouring chains. For shear rates where $\dot{\gamma}\tau_d > 1$ the polymer chains become oriented along the flow direction. For shear rates where $\dot{\gamma}\tau_s > 1$ the polymer chains align and polymer chain stretch starts to occur. The behaviour of the polymer chain stretch is important as the onset of polymer crystallisation increases dramatically under these conditions. The crystal structures that form affect the flow properties of polymers and additional models that account for the crystallisation within the polymers need to be used in conjunction with molecular models to address this problem. Details of some crystallisation models will be given in section 1.7.

1.5.2.1 Multimode Rolie-Poly Model

A single mode Rolie-Poly model cannot capture accurately the behaviour of realistic polymer melts as it only considers a single chain length. In reality commercial grade polymer melts are polydisperse, comprising of a wide range of different molecular weights. The relaxation times within the model will be different for different molecular weights. With this in mind, more modes are required in order to obtain a better prediction that represents a real polymer under flow. The polymeric stress contribution in this case is given as:

$$\boldsymbol{\tau}_P = \sum_{i=1}^{N} \boldsymbol{\tau}_i \tag{1.18}$$

where $\boldsymbol{\tau}_i$ is the polymeric stress from mode i and N is the total number of modes in the model.

The linear relaxation modulus of the multimode Rolie-Poly model can be given by,

$$G(t) = \sum_{i=1}^{N} g_i e^{(-t/\tau_{d,i})} = G_N^0 \sum_{i=1}^{N} \phi_i e^{(-t/\tau_{d,i})}, \tag{1.19}$$

where g_i is the elastic modulus of each mode, G_N^0 which is formed from the plateau modulus and ϕ_i the volume fraction of each mode. Thus the linear regime of the RP model becomes a linear superposition of Maxwell modes.

Since the multimode Rolie-Poly model is simply a linear superposition of the different contributions to the stress from each mode, the polymer chains are treated as being independent from each other. In reality chains of different lengths would interact with each other, this implies that the polymer chains are coupled. The stress contribution

from the interaction between the different chains will add to the total polymeric stress, which then affects the flow properties of the polymer. With this in mind an extension to the Rolie-Poly model has been developed which incorporates the coupling between the different chains to address the polydisperse problem.

1.5.3 Rolie-Double-Poly Model

The Rolie-Double-Poly (RDP) model is an extension of the model in the previous section and was developed first by Boudara et al. [1]. The idea of double-reptation put forward by des Cloizeaux [13] is used to model the interactions between different polymer chains. Using this method the non-linear rheology of polydisperse linear entangled polymers can be predicted, whilst being sufficiently simple so that it can be implemented in CFD software packages. This is appealing as almost every polymeric fluid used in industry is polydisperse. Furthermore, the flows ubiquitous in polymer processing are strong (non-linear), which makes the development of this model all the more significant.

The RDP model considers a blend of N linear chain species, each with a different chain length that are mutually entangled. Within this blend, the interactions between chain species occur in N^2 ways. The notation "IJ" indicates the conformation of species I within the constraining tube created by species J. As with the RP model the total polymeric stress tensor is given as a sum over modes:

$$\underline{\underline{\boldsymbol{\tau}_P}} = G_N^0 \sum_{i=1}^{n} \phi_i f_E(\lambda_i) \underline{\underline{\mathbf{A}}}_i, \tag{1.20}$$

where G_N^0 is the experimental plateau modulus, ϕ_i represents the volume fraction of each of the chain species and finite extensibility, $f_E(\lambda_i)$, is as defined in equation 1.17. The chain stretch, λ_i, for each chain species is given by,

$$\lambda_i = \left(\frac{1}{3} tr \underline{\underline{\mathbf{A}}}_i \right)^{\frac{1}{2}}. \tag{1.21}$$

The conformation tensors for each of the chain species are represented by $\underline{\underline{\mathbf{A}}}_i$. These conformation tensors represent the mean conformation tensor of the entanglement of the different chain species. The chain species entangle with each-other and release their entanglements on different timescales. Therefore it is necessary to break down the conformation tensors $\underline{\underline{\mathbf{A}}}_i$ as,

$$\underline{\underline{\mathbf{A}}}_i = \sum_{j=1}^{n} \phi_j \underline{\underline{\mathbf{A}}}_{ij}, \tag{1.22}$$

where $\underline{\underline{\mathbf{A}}}_{ij}$ is the stress conformation tensor on the i-chains coming from their entanglements with the j-chains. The decomposition of $\underline{\underline{\mathbf{A}}}_i$ leads to the conformation tensor evolving in such a way that accounts for the interactions between different chain species. The conformation tensor $\underline{\underline{\mathbf{A}}}_{ij}$ evolves like so:

$$\overset{\nabla}{\underline{\underline{\mathbf{A}}}}_{ij} = -\frac{1}{\tau_{d,i}}(\underline{\underline{\mathbf{A}}}_{ij} - \underline{\underline{\mathbf{I}}}) - \frac{2}{\tau_{s,i}}\left(1 - \frac{1}{\lambda_i}\right) f_E(\lambda_i)\underline{\underline{\mathbf{A}}}_{ij} - (\underline{\underline{\mathbf{A}}}_{ij} - \underline{\underline{\mathbf{I}}})\left[\frac{\beta_{th}}{\tau_{d,j}} + 2\beta_{CCR}\frac{f_E(\lambda_j)}{\tau_{s,j}}\left(1 - \frac{1}{\lambda_j}\right)\lambda_i^{2\delta}\right], \quad (1.23)$$

where $\beta_{th} = 1$ is the thermal constraint release coefficient. This coefficient accounts for the reptation of the j chains. The factor of 2 in front of the different relaxation times is a consequence of the relaxation of $\underline{\underline{\mathbf{A}}}_{ij}$ having both the $\tau_{d,i}$ and $\tau_{d,j}$ terms.

The RDP model is consistent with double reptation theory [13] which models the relaxation of two entanglements, where each entanglement involves an interaction between chains of different lengths. We can illustrate the coupling between chain species of different lengths by looking at the linear viscoelastic limit of the RDP model. The linear viscoelastic limit is given by,

$$G(t) = G_N^0 \sum_{i=1}^{N}\sum_{j=1}^{N} \phi_i\phi_j e^{(-t/\tau_{d,i} - t/\tau_{d,j})}, \quad (1.24)$$

where G_N^0 is the plateau modulus and ϕ_i & ϕ_j are the volume fractions of the i-th and j-th modes respectively. Here you can see that different chain species are interacting in the linear regime with the presence of the double summation, compared to the single summation in the linear regime for the RP model given in equation 1.19.

1.5.3.1 Enhanced Stretch Relaxation Time

A new feature that arises from the coupling involved in the Rolie-Double-Poly model is that of enhanced stretch relaxation time. The effect of chain stretch and extension hardening is given by a stretch relaxation time of the long chains in the blend. The dilution of the long chains with the short chains is found to increase the effective stretch relaxation time of the polymer chains. When the polymer chains become stretched, the polymer exhibits elongational hardening, where the elongational viscosity grows under extensional flow. In extensional flow, if the flow rate exceeds the inverse of the stretch relaxation time of the long chains, then the molecules are stretched under the flow. What has been shown experimentally [44] is that this relaxation time under extensional flow increases at a rate proportional to $\phi_L^{-\alpha}$ where α is the dilution exponent to represent entanglements. If entanglements are assumed to be binary events, then the dilution exponent $\alpha = 1$. This is the case for the RDP model since it is based on double reptation which is binary. Some larger values of α have been observed experimentally [45] and range up to around 4/3 owing to the enhanced relaxation of the long chain extremities.

A fully detailed constitutive model has been developed by Read et al. [46] where the enhanced stretch relaxation time results naturally from the theory. However this is too complex for computational fluid dynamics. The Rolie-Double-Poly model is an

approximation to this theory that still captures the enhanced stretch relaxation time in the model, but is viable for computational fluid dynamics.

1.6 Continuum Models of FIC

When attempting to model FIC, it is desirable to have methods suited to model polymer processing. Complex flow geometries are ubiquitous in polymer processing, therefore modelling FIC in these complex flow geometries needs to be computationally tractable. The simplest method of achieving this is to have a macroscale continuum level model that makes use of information within the flow model. These models for FIC usually use ordinary differential equations to model the crystallisation kinetics in flow.

These ordinary differential equations need to be supplied with a formula for the nucleation rate that provides a link between flow and crystallisation kinetics. This is usually calculated using a measure of the orientation in a polymer melt that is calculated via a constitutive law. It is generally understood that the highest molecular weight chain species have the biggest impact on the crystallisation rate increase in flow [47] and the transition from spherulitic to oriented crystal structure [48]. In an effort to reduce computational cost, some investigations only consider the contribution of the largest molecular weight to the flow induced nucleation rate [41, 49]. This fails to capture the detail of the remaining chain species in the polymer melt that would contribute to the enhanced nucleation rate in flow.

One way to model the flow induced nucleation effects in this way is to utilise the chain stretch of the polymer chains that is calculated from the constitutive equation for the polymer stress. Van Berlo et al. [41] used the following relationship,

$$\dot{N}_f = g_n(T)\exp\left(\mu_n(\Lambda_{HMW}^2 - 1)\right), \tag{1.25}$$

where $g_n(T)$ and μ_n are scaling parameters. Here Λ_{HMW} is the stretch of the chain species with the longest relaxation time. The total nucleation rate is then the total of the quiescent nucleation rate and the flow enhanced rate from equation 1.25. Formulae like this are computationally tractable in complex flow geometries, however they move away from a molecular based approach when modelling FIC. A molecular based model to FIC could allow for better detail to be captured when moving to this level of coarse graining. A computationally tractable FIC model with molecular roots would be optimal in this scenario.

1.6.1 Crystal Growth

Having determined the rate of nucleus formation, the next step is to predict the growth of these nuclei into crystals. The Schneider rate equations [50] are a set of coupled ordinary differential equations that are used to model the growth of nuclei into isolated spherulitical crystals. These are given as:

$$\begin{aligned}\frac{DN}{Dt} &= \dot{N},\\ \frac{DR}{Dt} &= 2G_sN,\\ \frac{DS}{Dt} &= 4\pi G_sR,\\ \frac{DV}{Dt} &= G_sS,\end{aligned} \tag{1.26}$$

where $\frac{D}{Dt} = \frac{\partial}{\partial t} + \mathbf{u}\cdot\nabla$ represents the material derivative. All quantities are per unit volume, V represents the volume fraction of the crystals; S is the total surface area of the crystals; R is the total radius; N is the number of nuclei. The quantities $\dot{N}$ and G_s represent the nucleation rate per unit volume and the spherulitic growth rate respectively.

The volume fraction calculated via the Schneider rate equations, V, does not account for impingement between crystals i.e. more than one crystal can occupy the same space. Therefore it is necessary to use the Avrami equation [51] to find the degree of space filling, ξ_g, that is the volume fraction corrected for impingement and is given by

$$\xi_g = 1 - \exp(-V). \tag{1.27}$$

Once the nucleation rate within the material has been determined, the Schneider rate equations combined with the Avrami equation provide useful crystallisation kinetics at a low computational cost. This methodology is common [52, 53] owing to its simplicity to implement in CFD packages. Grosso et al. [54] utilised the Schneider rate equations to account for spherulitic formation of crystals when simulating iPP during injection moulding. The flow induced nucleation rate was calculated in a phenomenological way by using the stretch of only the highest molecular weight chain species to give a flow induced nucleation rate that is calculated using equation 1.25. The Avrami equation was then used to calculate impingement. This investigation managed to get good agreement with experimental findings, showing that continuum level models are a powerful tool when simulating FIC in polymers.

Grosso [54] and Van Berlo et al. [41] not only considered the formation of spherulites with the Schneider rate equations, they also used an equivalent set of equations that model the formation of shish-kebab structures in their simulations. When modelling shish-kebab formation they found that the total volume of kebabs is much larger than that of shish, so the volume fraction of shish is neglected from the calculation.

1.7 Microscale Models of Crystal Nucleation

Polymer crystal nucleation events occur when polymer molecules begin folding over one another before the onset of spherulite formation. This is not visible experimentally, so simulations are needed to provide the required resolution to understand the events at a

molecular level that provide the onset of crystallisation. Molecular dynamics (MD) can be used to simulate the interaction of individual monomers for a fixed time. This can give a view of the evolution of individual polymer crystals from the nucleation event up until they are visible experimentally.

An MD investigation by Yi et al. [55] observed homogeneous crystal nucleation in n-octane polymer melts. They used the cylindrical nucleus model for their polymer crystals and were able to predict the critical nucleus size even at low undercooling, although it was for an idealised case of monodisperse polymer chains under quiescent conditions. Yamamoto et al. [56] modelled stretch induced crystallisation in polyethylene and uncovered detailed molecular configurations between the polymer crystals and the tie molecules between them, along with observing the fold in lamellae becoming tighter due to the orientation of the polymer chains.

The main disadvantage to MD simulations is that of computational cost. This limits the time and length scales that are accessible by investigations. Therefore, many MD investigations observe nucleation at high undercooling for short chains [34]. Reducing the temperature increases the nucleation rate so that nucleation can be observed on the timescale of the simulation. However, such fast nucleation at high-undercooling are not common conditions for industrial polymer processing, so MD simulations are not directly applicable to polymer processing.

With the ambition of simulating FIC in complex geometries, whilst keeping as much molecular detail as possible, multiscale modelling has become an essential ingredient. MD simulations provide the most molecular detail at the highest computational cost and continuum level models that are tractable in CFD packages provide the inverse. Lattice Monte Carlo [57, 58, 59] and coarse-grained kinetic Monte Carlo [60, 61, 62, 63] techniques provide a bridge between these two extremes. When moving between these levels, modelling assumptions need to be made, which lead to new undetermined model parameters. A better interface between these levels of modelling would provide justifications for assumptions that are made for an improved computational cost.

Recent developments in modelling FIC are rooted in developing models that are computationally cheap enough to be used at the continuum level, whilst keeping molecular details of polymer crystallisation included. The following section outlines two models for FIC that look to model FIC in a computationally tractable way.

1.7.1 GO Model

An example of a Kinetic Monte Carlo approach to modelling polymer flow induced crystallisation is the Graham and Olmsted (GO) model [60, 61]. A polymer crystal nucleus is formed from crystallised monomers (Kuhn steps) that are arranged in stems. This is the minimum detail required to model anisotropic nucleation. In this case the nucleus is assumed to be spheroidal, which is a limitation to the model as the shish-kebab structure is not captured at this level of coarse-graining.

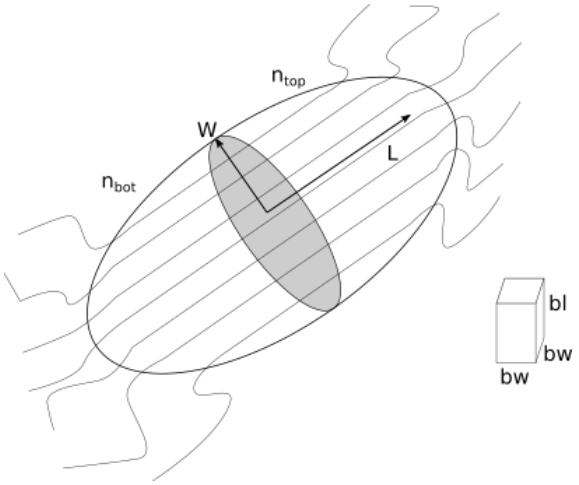


Figure 1.10: Diagram showing the coarse graining assumption for the crystal nuclei.

The GO model algorithm runs as follows:

- The stretching and orientation of the amorphous chains due to the flow are tracked. This is done using the GLaMM model [20], a more sophisticated version of the RP model, which provides a description of the chain conformation distribution as a function of position along the chain.
- A Kinetic Monte-Carlo method is used to simulate the attachment and detachment of chain segments to the nucleus.
- A Brownian dynamics simulation is used to evolve the rotation of the crystal nucleus.

In the original GO model the crystal nucleus rotates within the bulk flow according to Jeffrey's rotation of a spheroid subject to Brownian motion. However, this was subsequently found to have no significant effect on the evolution of the crystal nucleus, so was omitted from subsequent descriptions of the GO model [25, 34].

The nucleus free energy in the GO model is a balance of the free energy of crystallisation proportional to the nucleus volume with a free energy penalty proportional to the spheroid surface area. With N_T and N_S representing the number of monomers and stems respectively, the nucleus free energy is given by, $\mathcal{F}_{nuc}(N_T, N_S) = -\varepsilon_B N_T + \mu_s S(N_T, N_S)$, where S is the nucleus surface area, ε_B and μ_s are the dimensional bulk free energy per unit volume and surface free energies per unit area respectively. This nucleus free energy is used in the polySTRAND model [2] which is outlined in section 1.7.2.

In order to model the nucleation kinetics of the crystals Graham and Olmsted used kinetic Monte Carlo simulations. The two basic moves are stem addition and stem lengthening that add a single monomer to the nucleus and have an appropriate reverse move associated with them. The change in free energy associated with all these moves is calculated and summed to find the likelihood of a given monomer being added to the nucleus at any given moment in the simulation. A move is then carried out at random once this probability is calculated. Time is then incremented forwards in stochastically determined

steps. As time advances, the polymer chain configuration changes. The GLaMM model is used to pre-calculate the configuration and this calculation is advanced forwards to reflect the advancement of time. The free energy of all the moves in the simulation are then recalculated, which updates the likelihood of a monomer being added to the crystal nucleus.

This coarse graining approach provides a molecular mechanism for the theory and dynamics of FIC, however, since it involves a stochastic simulation, and while much faster than MD, it is still computationally expensive and so is restricted to homogeneous flows such as shear-flow. In order to utilise this model in CFD programs suitable for simulation of more complex processing flows, further coarse graining is needed to give a macroscale continuum model for FIC. In addition, the use of the GLaMM model means it is strictly limited to monodisperse polymers.

1.7.2 PolySTRAND Model

The POLYdisperse STRain Accelerated Nucleation Dynamics (polySTRAND) [2] model is a fully polydisperse model for flow induced nucleation (FIN). This recently developed model is computationally cheap enough to be used in CFD software programs for industrially relevant polymers.

The polySTRAND model modifies the GO model to incorporate findings from recent molecular dynamics (MD) simulations. It uses the RDP model to account for polydispersity of chain species. It also uses an analytical approximation in place of the stochastic simulation of the GO model. The polySTRAND model refers to two sub-models, the GO-polySTRAND, which closely resembles the GO model, and the smooth-polySTRAND which captures experimental data by adding a nucleus roughness penalty and local exhaustion of long chains.

In the GO-polySTRAND model an analytical approximation for the nucleus free energy consistent with the GO model is derived that is valid in the limit of large nuclei. In the GO model the crystal nucleus is made up of stems with different lengths at fraction w_i, which gives the number of i stems as $N_S w_i$. Monomers in the GO model nucleus have fraction v_i, which gives the number of i monomers as $N_T v_i$. The GO model crystal nucleus entropy contains the number of configurations of $N_S w_i$ stems of each chain species among N_S stems and the number of different ways to spread $N_i v_i$ monomers among $N_S w_i$ stems. Provided N_S and N_T are large the nucleus free energy is given approximately by

$$F = N_T \sum_i [q w_i (2\log w_i - \log\phi_i) - v_i \log v_i + (v_i - q w_i)\log(v_i - q w_i) \\ - v_i \Delta f_i] + N_S \log q - \varepsilon_B N_T + \mu_S S(N_T, N_S), \quad (1.28)$$

where $q = N_S/N_T$ and Δf_i is the change in free energy due to chain deformation for adding a monomer to the nucleus. Minimisation over $\{w_i, v_i\}$ with two constraints $\sum_i w_i =$

$\sum_i v_i = 1$ gives analytic expressions for the fraction of stems and monomers of each species as

$$w_i = \frac{B\phi_i \exp(\Delta f_i)}{1 - A\exp(\Delta f_i)}, \qquad v_i = \frac{Bq\phi_i \exp(\Delta f_i)}{[1 - A\exp(\Delta f_i)]^2}, \tag{1.29}$$

where A and B are Lagrange multipliers calculated by,

$$\sum_i \frac{q\phi_i \exp(\Delta f_i)}{(1 - A\exp(\Delta f_i))^2} = \sum_i \frac{\phi_i \exp(\Delta f_i)}{1 - A\exp(\Delta f_i)}, \qquad B = \frac{1}{\sum_i \frac{\phi_i \exp(\Delta f_i)}{1 - A\exp(\Delta f_i)}}. \tag{1.30}$$

The first equation in 1.30 is solved numerically for A and the second gives B upon calculation of A. Since A is the only parameter that needs to be computed numerically, the GO-polySTRAND model is computationally cheap.

The polySTRAND model gives a reasonable approximation to the GO model, particularly for large N_T and N_S. However, there are small discrepancies with the GO model. In order to get a more accurate representation of the GO model, the change in free energy barrier due to flow within the polySTRAND model was computed alongside a better analytical approximation to the quiescent barrier for the GO model. The final free energy barrier is computed using the polySTRAND model change in energy barrier to calculate from the quiescent state and the GO model to give an approximation for the quiescent energy barrier. The method for carrying this out is by finding the optimum number of stems for a given number of monomers by projecting equation 1.28 onto N_T by minimising over N_S. The entropic contributions from fluctuations in N_S is approximated via $\partial^2 F/\partial N_S^2$. This procedure is carried out for the quiescent and the flow case. The change in free energy for nucleus growth due to flow as calculated by the polySTRAND model is given by: $\Delta F_{pS} = F_{pS}^q - F_{pS}^{flow}$, where F_{pS}^q is the free energy within polySTRAND without flow and F_{pS}^{flow} is the free energy within polySTRAND when flow is present. The quiescent barrier, F_{GO}^q, is calculated using the GO model [61] and can be easily computed via the algorithm developed by Jolley and Graham [64]. The final free energy barrier is then given by $F_{final} = F_{GO}^q - \Delta F_{pS}$. This final free energy barrier calculation agrees well with GO simulations. The overall flow induced nucleation rate, $\dot{N}$, is calculated by solving a 1D Kramer's problem [34, 65]. This method accurately predicts $\dot{N}$ from GO simulations, further details of this procedure are given in section 2.5.

The remaining task is to calculate the change to the monomer free energy of each species $\{\Delta f_i\}$. In MD simulations of short entangled chains, it is found that $\dot{N}$ grows exponentially with the Kuhn segment nematic order $P_{2,K}$ [66]. The Kuhn segment nematic order is a measure of Kuhn segment (monomer) alignment. For a single species at moderate Δf_i, polySTRAND predicts that $\dot{N}$ is proportional to $\exp(\Delta f n^*)$, where n^* is the critical nucleus size under quiescent conditions. Therefore, setting $\Delta f = \Gamma P_{2,K}$ for some constant Γ, recreates the exponential behaviour observed in MD. For the case of a polydisperse melt, $P_{2,K}$ is calculated for each chain species. In order to calculate the polymer chain

configuration in the melt for each chain species, the RDP constitutive equation is used and gives conformation tensors, $\underline{\underline{\mathbf{A}}}_i$, for each species. The molecular weight distribution (MWD) of a polymer is represented by N chain species, where species i has volume fraction ϕ_i. As detailed in section 1.5.3 the RDP model calculates the conformation tensor $\underline{\underline{\mathbf{A}}}_{ij}$, via N^2 coupled differential equations. The Kuhn segment nematic order parameter for each chain species is then approximated by $P_{2,K,i} = \Lambda_{max}/N_e$, where Λ_{max} is the largest eigenvalue of $\underline{\underline{\mathbf{A}}}_i - \underline{\underline{\mathbf{I}}}$ and N_e is the number of Kuhn steps per entanglement. The combination of the RDP model and the polySTRAND model allows for effective computation of polymer chain species alignment which can then be used to compute the overall nucleation rate in flow.

The GO-polySTRAND model described above was found to overpredict the nucleation barrier at high shear rate $\dot{\gamma}$, which suggests that there may be a limit to the enhancement of nucleation due to the alignment of polymer chains. An explanation for this could be that the growth of the nucleus means there is insufficient time for stems to attach from an infinite supply in the melt. What would happen in practice is that stems attach from a small region around the crystal nucleus. Modifications were therefore made to the original GO-polySTRAND model and the resulting submodel is referred to as Smooth-PolySTRAND. The crystal nucleus in the GO-polySTRAND model can be comprised of stems of widely different lengths. This is unreasonable because stems of similar lengths benefit the most from crystallisation. In an effort to address these problems, two modifications are proposed. First, if a crystal nucleus attaches N_S stems at fraction w_i from a limited number Q_S of stems with a volume fraction $\{\phi_i\}$, this leaves $Q_S - N_S$ remaining stems at fraction θ_i, with $(Q_S - N_S)\theta_i + N_S w_i = Q_S \phi_i$. Consequently, the term $-N_S \sum_i w_i \log\phi_i$ in equation 1.28 is replaced with $\sum_i [(Q_S - N_S)\theta_i \log\theta_i - Q_S \phi_i \log\phi_i]$. This represents the change in stem translation entropy between the initial melt (made up of Q_s stems at fraction ϕ_i) and the final melt (made up of $Q_s - N_s$ stems at fraction θ_i). Second, deviations of stem length l (in monomers) from the average $L = N_T/N_S$ are penalised through an energy penalty $\kappa(l - L)^2/2$.

As described above the polySTRAND model integrates the insights gained from MD and Kinetic Monte-Carlo simulations and uses thermodynamic modelling to give an analytic continuum model for FIC. The model accounts for the effects of polydispersity on both the flow and the nucleation dynamics. In particular, long chains deform strongly and attach more readily than the shorter, less deformed, chains. Therefore, the model predicts that, relative to their melt concentration, long chains are enriched in nuclei. Hence multiple chain lengths interact non-trivially to determine the nucleation barrier. The barrier height sets the nucleation rate via an exponential dependence, so correctly predicting long chain enrichment is essential to quantitative FIC modelling [67]. Thus, polydispersity plays a dominant role in FIC in both experiments and industrial polymer processing, where polydisperse melts are abundant.

1.8 OpenFOAM

Following on from these recent advancements in the theoretical models used for FIC, the aim of this thesis is to couple these to a calculation of the flow in order to examine FIC in the type of complex flows encountered in polymer processing. This requires a CFD package that allows implementations of new models and full customisation to our particular needs. Open source Field Operation and Manipulation [68] (OpenFOAM) is a C++ toolbox that allows for customisation of numerical solvers for CFD problems as well as the use of pre and post-processing utilities to give a solution to various problems. The OpenFOAM library offers both finite volume and finite element discretization schemes [3] for the spatial variables. A benefit of using OpenFOAM is that the mathematical models can be translated into C++ code in a format that is almost the same as expressing partial differential equations in tensorial form, meaning that implementation of new models in OpenFOAM can be done intuitively without requiring detailed implementation of the numerical discretisation. OpenFOAM is also open source so its base code and architecture is available to the user for customisation for their particular CFD problem. This makes OpenFOAM versatile and powerful when tackling new and complex problems.

1.8.1 rheoTool

Although OpenFOAM has the capabilities to solve problems involving viscoelastic fluids using the solver `viscoelasticFluidFoam` [69], this functionality is not robust and there remains an instability issue in certain regimes e.g. for high Weissenberg number flows. Modifications to the `viscoelasticFluidFoam` solver were made to address this stability issue. This was the inception of an extension to OpenFOAM known as rheoTool [4]. The rheoTool toolbox provides a library of solvers and constitutive equations for viscoelastic fluids.

The rheoTool package is open source, and so is fully customisable from the source code given. This allows for new constitutive equations to be added to the program, as described in the user guide [4]. Azahar [7] successfully implemented a two mode RDP model in rheoTool to analyse the flow of a bidisperse polymer blend in contraction flows. In this thesis I will expand on her work by implementing an RDP model for an arbitrary number of chain species and incorporating a prediction of FIC based on the polySTRAND model.

1.9 Conclusion

There is now a general understanding in the literature that flow enhances the nucleation rate within polymers by reducing the entropic penalty for monomers to form a crystal nucleus when the polymer chains become stretched and monomers align. However, the exact molecular mechanism that determines this increase remains unclear. The modelling framework used to capture this phenomenon is also currently under debate. The coarse grained

continuum level models offer fast computational speed, but sacrifice accuracy through the need for assumptions on the molecular level; whilst detailed molecular simulations capture nucleation events well, but are far too computationally expensive to run under conditions realistic of polymer processing. The current direction is to use a multiscale approach to modelling this problem, where an increased level of coarse graining is used to capture as much detail as possible from a molecular standpoint, whilst being cheap enough to run in CFD software.

A clear barrier to success of more detailed models is the computational cost. This arises because of the huge range of time and length scales that are present in the crystallisation problem. Advances in computing power that would be sufficient to keep all the detailed molecular dynamics in a CFD model are unlikely to happen in the near future, and indeed this level of power may never be achieved. As a result different theoretical approaches seem like a more appropriate way to address this problem. The polySTRAND model in conjunction with the RDP model is a recent advancement that takes into account polydispersity in a computationally cheap way, yet there could be a model that captures even more detail at a similar computational cost on the horizon.

There is a continued effort in gaining a better theoretical understanding of FIC owing to the huge production of semi-crystalline polymers worldwide. This problem also demonstrates the power of computational science and how it can be used as a substitute for experiments in places where experiments are impractical or even impossible. Simulations in CFD software can be run multiple times, in different flow geometries, at comparatively little capital cost. Combining the polySTRAND model with the RDP model in OpenFOAM will allow detailed simulations in complex flow geometries to be conducted. As such, this could pave the way for tailoring of polymer processing conditions in an industrial setting.

1.10 Thesis Overview

This thesis consists of 5 chapters including the introduction chapter. Chapter 2 shows the implementation of both the RDP and polySTRAND models in openFOAM along with the validation for both models against literature results. We then simulate the flow of an isothermal bimodal polymer blend through a channel to validate the implementation of the rheological model. This idealised blend is also simulated flowing through a contraction expansion geometry in this chapter.

Chapter 3 presents the simulation of a polydisperse melt through a contraction expansion geometry and our results are compared to existing experimental findings. We investigate how changes in flow rate affect the crystallisation profile in our contraction expansion geometry.

We then present a method of incorporating temperature dependence into our calculations in chapter 4. We investigate the effect of cooling a polydisperse melt in a channel geometry. Using our contraction expansion geometry, the effect of having a cold and a hot

wall relative to the melt is presented in this chapter.

Chapter 5 is the concluding chapter that summarises the chapters and highlights the contributions of this work. The ways in which this work could be extended are also discussed here.

Chapter 2

Isothermal Bimodal Polymer blend in a Complex Geometry

In this chapter we validate the implementation of the RDP and polySTRAND models in openFOAM and then explore the modelling of flow induced crystallisation of a polymer melt in a complex flow geometry. We first simulate flow through a 2-dimensional channel geometry to validate the implementation of the rheological model. We then consider the 2-dimensional flow through a contraction geometry with the following simplifying assumptions. We assume the flow is isothermal and that the polymer is a bimodal blend, where only the high molecular weight fraction contributes to the flow induced enhancement of nucleation. Furthermore, we assume that the crystal growth is sufficiently slow that the rheology of the melt is not affected by the presence of crystals.

We describe the method for calculating the local nucleation rate from the polymer configuration given by the RDP modes using the polySTRAND model and discuss how this is implemented in our simulations. Instead of implementing a direct calculation of the nucleation rate at each point in space we chose a scheme that allows pre-computation and numerical approximation of the nucleation rate as a function of the monomer free-energy.

The construction of the mesh used in the simulation is outlined and the justification for this choice is given. There follows an exploration of our results for a model binary blend in a contraction flow matched to an experimental investigation.

2.1 Overview

Polymer flow induced crystallisation arises from the reduction in the free energy barrier to nucleation resulting from the alignment of molecular segments within the flowing polymer melt. Therefore in order to predict the degree of crystallisation at each point in the melt it is necessary to couple together a calculation of the flow and molecular configuration to the thermodynamic calculation of the free energy barrier to nucleation.

The structure of the work-flow of the combined simulation is as follows. First there is an initialisation to set-up the model and define the mesh for the openFOAM simulations.

Next there is the calculation of the flow and polymer configuration using the RDP model. This is carried out in openFOAM using the rheoTool [4] library as used by Azahar [7]. The next step is to calculate the local nucleation rate using the polySTRAND model. Previous calculations for this model by Read et al. [2] considered the case of simple shear flow with no spatial variation and so the challenge in moving to a complex flow geometry is to find a viable method to calculate the flow induced nucleation rate in a computationally efficient way. Once all these steps are complete, the volume fraction of crystals within the melt can be calculated using the Schneider rate [50] and Avrami [51] equations.

2.2 Compilation of RDP in rheoTool

In previous implementations in rheoTool [7] the Rolie-Poly and RDP models were encoded into the software in the form of the polymeric stress tensor $\boldsymbol{\tau}$. We have found that the model can be encoded with the conformation tensor as the solution variable, which is the form presented in the previous chapter. Therefore the task of compiling the model into rheoTool requires no alterations to the model prior to encoding it into the software.

The RDP model, as presented in the previous chapter, is rewritten here in the form it was implemented in OpenFOAM. The equations representing interactions between the same chain species and different species equations are separated and given as follows:

Same Species Equation

$$\begin{aligned}\frac{\partial \underline{\underline{\mathbf{A}}}_{ii}}{\partial t} + \mathbf{u}\cdot\nabla\underline{\underline{\mathbf{A}}}_{ii} &= \nabla\mathbf{u}\cdot\underline{\underline{\mathbf{A}}}_{ii} + \underline{\underline{\mathbf{A}}}_{ii}\cdot(\nabla\mathbf{u})^T \\ &- \left[\frac{1+\beta_{th}}{\tau_{d,i}} + \frac{2}{\tau_{s,i}}\left(1-\frac{1}{\lambda_i}\right) f_E(\lambda_i)\left(1+\beta_{CCR}\lambda_i^{2\delta}\right)\right]\underline{\underline{\mathbf{A}}}_{ii} \\ &+ \left[\frac{1+\beta_{th}}{\tau_{d,i}} + \frac{2}{\tau_{s,i}}\left(1-\frac{1}{\lambda_i}\right) f_E(\lambda_i)\beta_{CCR}\lambda_i^{2\delta}\right]\underline{\underline{\mathbf{I}}},\end{aligned} \tag{2.1}$$

Different Species Equation

$$\begin{aligned}\frac{\partial \underline{\underline{\mathbf{A}}}_{ij}}{\partial t} + \mathbf{u}\cdot\nabla\underline{\underline{\mathbf{A}}}_{ij} &= \nabla\mathbf{u}\cdot\underline{\underline{\mathbf{A}}}_{ij} + \underline{\underline{\mathbf{A}}}_{ij}\cdot(\nabla\mathbf{u})^T \\ &- \left[\frac{1}{\tau_{d,i}} + \frac{2}{\tau_{s,i}}\left(1-\frac{1}{\lambda_i}\right) f_E(\lambda_i) + \frac{\beta_{th}}{\tau_{d,j}}\right. \\ &\left. +2\beta_{CCR}\frac{f_E(\lambda_j)}{\tau_{s,j}}\left(1-\frac{1}{\lambda_j}\right)\lambda_i^{2\delta}\right]\underline{\underline{\mathbf{A}}}_{ij} \\ &+ \left[\frac{1}{\tau_{d,i}} + \frac{\beta_{th}}{\tau_{d,j}} + 2\beta_{CCR}\frac{f_E(\lambda_j)}{\tau_{s,j}}\left(1-\frac{1}{\lambda_j}\lambda_i^{2\delta}\right)\right]\underline{\underline{\mathbf{I}}}.\end{aligned} \tag{2.2}$$

The tables below describe how the model is translated into the C++ code almost analogous to the mathematical operators in the equations.

Same Species Equation

Mathematical Operator	OpenFOAM Implementation
$\frac{\partial \underline{\underline{\mathbf{A}}}_{ii}}{\partial t} + \mathbf{u} \cdot \nabla \underline{\underline{\mathbf{A}}}_{ii}$	`fvm::ddt(AII_)` + `fvm::div(phi(), AII_)`
$\nabla \mathbf{u} \cdot \underline{\underline{\mathbf{A}}}_{ii} + \underline{\underline{\mathbf{A}}}_{ii} \cdot (\nabla \mathbf{u})^T$	`twoSymm(CII)`
$- \left[\frac{1+\beta_{th}}{\tau_{d,i}} + \frac{2}{\tau_{s,i}}\left(1 - \frac{1}{\lambda_i}\right) f_E(\lambda_i) \left(1 + \beta_{CCR}\lambda_i^{2\delta}\right)\right] \underline{\underline{\mathbf{A}}}_{ii}$	`-fvm::Sp((1.+betaTh_)/lambdaDI_ + M1I + M1I*betaCR_* pow(trAI/3.,delta_), AII_)`
$\left[\frac{1+\beta_{th}}{\tau_{d,i}} + \frac{2}{\tau_{s,i}}\left(1 - \frac{1}{\lambda_i}\right) f_E(\lambda_i) \beta_{CCR}\lambda_i^{2\delta}\right] \underline{\underline{\mathbf{I}}}$	`((1.+betaTh_)/lambdaDI_ + M1I*betaCR_* pow(trAI/3.,delta_))*Ist`

Table 2.1: Table showing the mathematical operators for the same species RDP equation and how they are translated into OpenFOAM. There have to be some alterations to the syntax in openFOAM as τ is used for polymeric stress. Therefore all relaxation times are referred to as λ in openFOAM.

Different Species Equation

Mathematical Operator	OpenFOAM Implementation
$\left[\frac{1}{\tau_{d,i}} + \frac{2}{\tau_{s,i}}\left(1 - \frac{1}{\lambda_i}\right) f_E(\lambda_i) + \frac{\beta_{th}}{\tau_{d,j}} + 2\beta_{CCR}\frac{f_E(\lambda_j)}{\tau_{s,j}}\left(1 - \frac{1}{\lambda_j}\right)\lambda_i^{2\delta}\right] \underline{\underline{\mathbf{A}}}_{ij}$	`-fvm::Sp(1./lambdaDI_ + M1I + betaTh_/lambdaDJ_ + betaCR_*M1J* pow(trAI/3., delta_), AIJ_)`
$\left[\frac{1}{\tau_{d,i}} + \frac{\beta_{th}}{\tau_{d,j}} + 2\beta_{CCR}\frac{f_E(\lambda_j)}{\tau_{s,j}}\left(1 - \frac{1}{\lambda_j}\lambda_i^{2\delta}\right)\right] \underline{\underline{\mathbf{I}}}$	`(1./lambdaDI_ + betaTh_/lambdaDJ_ + betaCR_*M1J* pow(trAI/3., delta_))*Ist`

Table 2.2: Table showing the mathematical operators for the different species RDP equation and how they are translated into OpenFOAM. Similarly the relaxation times have to be relabelled in openFOAM.

Using this notation, the bimodal RDP model has been successfully implemented in rheoTool with two species denoted as S and L for the short and long chain species respectively. In chapter 3 this will be generalised to N species. The following section presents the context for using this model and the results gained from simulations using this implementation. Note that the first two entries in table 2.1 are also present in the different species equation, so are not repeated in table 2.2.

The polymeric stress is calculated according to equation 1.20 given in section 1.5.3 and is given for a bimodal blend here,

$$\underline{\underline{\boldsymbol{\tau}}}_P = G_N^0[\phi_S f_E(\lambda_S)\underline{\underline{\mathbf{A}}}_S + \phi_L f_E(\lambda_L)\underline{\underline{\mathbf{A}}}_L], \tag{2.3}$$

where the subscripts S and L represent contributions from the short and long chain species respectively. This is calculated at the end of the constitutive equation file in openFOAM when all the values in the simulation are updated to give the total stress within the investigation.

2.3 Validating Bimodal Rolie-Double-Poly

The solver `rheoTestFoam` simulates extensional and shear flow over a single computational cell for a model with a single relaxation time by default. Since the RDP model has multiple relaxation times, the solver needs to be adapted to fit our needs.

This modified solver is named `rheoTestFoamRDP` and will be used to validate the model for purely shear and extensional flow over a single computational cell. The current work is compared to existing literature for shear and extensional flow to ensure the model is correctly implemented.

The bimodal RDP model is compared with the calculations for both the shear and extensional flow given by Azahar [7] and the extensional flow presented by Boudara [8]. These were chosen to demonstrate that the implementation replicates results with different model parameters.

In the work by Azahar [7] the bimodal RDP relaxation times are $\tau_{d,L} = 10$, $\tau_{d,S} = 0.1$, $\tau_{s,L} = 0.2$, $\tau_{s,S} = 0.05$ where finite extensibility is neglected ($f_E(\lambda) = 1$) and in the absence of convective constraint release, $\beta_{CCR} = 0$.

The work by Boudara [8] uses the relaxation times $\tau_{d,L} = 100$, $\tau_{d,S} = 0.05$, $\tau_{s,L} = 1$, $\tau_{s,S} = 0.01$ and includes convective constraint release ($\beta_{CCR} = 1$) and limits the finite extensibility function by setting, $\lambda_{max} = 200$.

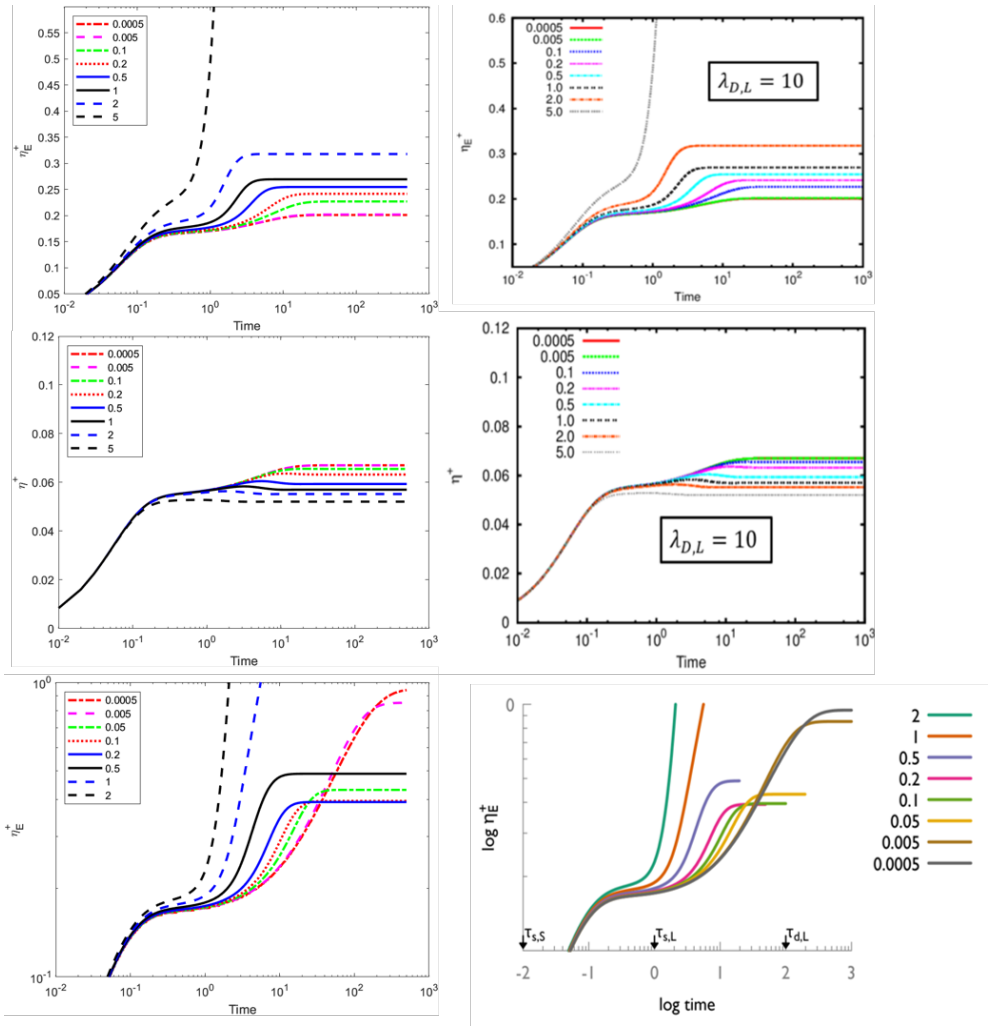


Figure 2.1: Comparison figure showing the validation of the bimodal RDP model in purely extensional flow (top and bottom rows) and shear flow (middle row) using the `rheoTestFoamRDP` solver. The legends show the increasing flow rates in the simulation runs. The left column is the current work, while the right column contains the results from literature[7, 8].

Figure 2.1 shows perfect agreement between the current work and the existing simulations. This means that we can have confidence that the model is working as expected and will work appropriately in multiple contexts. This represents a reasonable validation of the model implementation in rheoTool.

When looking at the results for the extensional flow, the linear regime shows a 'double-shoulder' shape. The reason for this can be understood by looking at the limit of $\tau_{d,S} \ll \tau_{d,L}$. In this limit in the RDP model we have:

$\underline{\underline{\mathbf{A}}}_{SS}$ relaxes at time $\left(\frac{1}{2\tau_{d,S}} + \frac{\beta_{th}}{2\tau_{d,S}}\right)^{-1} \approx \tau_{d,S}^{-1}$ and adds a weight of ϕ_S^2 to the stress,

$\underline{\underline{\mathbf{A}}}_{SL}$ relaxes at time $\left(\frac{1}{2\tau_{d,S}} + \frac{\beta_{th}}{2\tau_{d,L}}\right)^{-1} \approx (2\tau_{d,S})^{-1}$ and adds a weight of $\phi_S\phi_L$ to the stress,

$\underline{\underline{\mathbf{A}}}_{LL}$ relaxes at time $\left(\frac{1}{2\tau_{d,L}} + \frac{\beta_{th}}{2\tau_{d,L}}\right)^{-1} \approx \tau_{d,L}^{-1}$ and adds a weight of ϕ_L^2 to the stress,

$\underline{\underline{\mathbf{A}}}_{LS}$ relaxes at time $\left(\frac{1}{2\tau_{d,L}} + \frac{\beta_{th}}{2\tau_{d,S}}\right)^{-1} \approx (2\tau_{d,S})^{-1}$ and adds a weight of $\phi_S\phi_L$ to the stress.

Therefore a fraction $\phi_S^2 + 2\phi_S\phi_L$ of the stress relaxes at approximately $\tau_{d,S}^{-1}$ and a fraction ϕ_L^2 relaxes at approximately $\tau_{d,L}^{-1}$. This gives the shape of the extensional results in the linear regime. As the extension rate $\dot{\varepsilon}$ increases past the inverse of the stretch relaxation time of the long chains, the chains are stretched by the flow. This leads to what is known as elongation hardening, where the elongational viscosity η_E^+ increases above the predictions of the linear regime.

In polymer blends, such as the scenario presented in this work, the onset of long chain stretch is dependent on the long chain concentration within the melt. A phenomenon known as enhanced stretch relaxation gives an adjusted stretch relaxation time for the long chains. For the case of a bimodal blend, the effective stretch relaxation time due to enhanced stretch relaxation is given by

$$\tau_{s,L}^{eff} = \frac{\tau_{s,L}}{\phi_L}.$$

This means that the onset of long chain stretch would be expected to happen when $\dot{\varepsilon} > (\tau_{s,L}^{eff})^{-1}$, rather than $\dot{\varepsilon} > \tau_{s,L}^{-1}$.

A behaviour that is present in the comparison with Boudara, but not Azahar is the presence of extension thinning. When the extension rate $\tau_{d,L}^{-1} < \dot{\varepsilon} < \tau_{s,L}^{-1}$ the melt thins, i.e. the viscosity decreases with an increase in extension rate. In order to see an extension thinning region before the onset of long chain stretch requires that $\tau_{d,L} \gg \tau_{s,L}^{eff}$ and so is not observed for the parameter values chosen by Azahar.

In shear flow, shear thinning is observed where the shear viscosity decreases as the shear rate increases. The onset of chain stretch is not obvious in this setting as shear flow does not typically produce such extreme chain stretch as extension. The next section will present the results for a fully two-dimensional simulation.

2.4 2-D Channel Geometry

A channel geometry was chosen for the initial simulations owing to its industrial relevance. In polymer processing, polymer melts are often pushed through channels, or channel like configurations in moulding environments. Therefore, demonstrating modelling in this geometry is a useful starting point. A channel is also a simple geometry that does not require a large amount of computational power. Therefore, multiple simulations can be conducted with minimal run times, so that different regimes of flow rate can be demonstrated within the channel.

2.4.1 Dimensions Within the Model

A dimensional analysis is carried to provide information about the dimensionless parameters in the model.

Figure 2.2: Sketch of the channel and the length scales relevant to the simulation.

First note that there are two length scales in this problem, the length and the height of the channel. The length of the channel is defined as l and the height of the channel is defined as h. The plateau modulus G_N^0 has the same dimensions as pressure. Therefore a dimensionless pressure drop can be defined as

$$\widetilde{\Delta p} = \frac{\Delta p}{G_N^0}.$$

Since we are in shear flow, the principle force balance is between $\partial p/\partial x$ and $\partial \sigma/\partial y$ so since $\sigma \sim G_N^0$ it makes sense to choose h as the length scale so that the dimensionless pressure gradient is given by

$$\widetilde{\nabla p} = \frac{\nabla p\, h}{G_N^0}.$$

All results from the simulations will be presented as dimensionless quantities, however the tilde above the dimensionless quantities will be omitted for notational ease.

2.4.2 Simulation Results

For a simulation over a full computational mesh, the solver that will be used is `rheoFoam`. This solver is designed to solve single phase flow of a transient incompressible flow for viscoelastic fluids. Therefore it is the appropriate solver to use in our case, as we are in the incompressible regime.

Even though the simulation domain is two-dimensional. OpenFOAM requires a three-dimensional mesh to be defined. The third dimension, in this case the z direction, is one cell width. The mesh is split into two halves that are combined along the plane $y = 0$, which gives the full mesh. The halves are combined using face matching where the face of one half is comprised of the same set of vertices as the other half. These faces do not form an external boundary and are joined at collocated pairs of vertices to form the final mesh.

Each half is made up of 50 cells in the x direction and 30 cells in the y direction, this gives 1500 cells for each half and a total of 3000 cells in the mesh. The mesh is defined with 40 units in the x direction and 2 units in the y direction. The y coordinates run from -1 to 1, while the x coordinate runs from 0 to 40. The cells in the mesh are of uniform density throughout, owing to the simplicity of the mesh, refinement in a particular area is

not required.

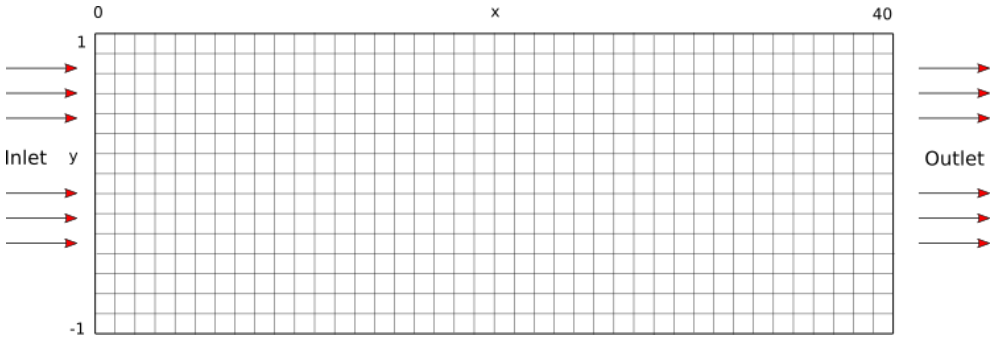


Figure 2.3: Sketch of the channel mesh used in the simulation. Note that the number of cells in this mesh is not the same as that in the simulation in OpenFOAM.

The fluid within a uniform channel experiences shear flow rather than extensional flow. The largest shear rates occur at the walls and with zero shear rate in the centre. The parameters of the RDP model in this simulation are the same as used in the thesis by Azahar [7], except that convective constraint release is incorporated with $\beta_{CCR} = 1$. It is worth noting that the relaxation times selected were not intended to mimic an actual polymer, but rather to have sufficient separation between values of $\tau_{d,L}$, $\tau_{s,L}$ and $\tau_{d,S}$ to demonstrate the effects of enhanced stretch relaxation time, whilst not having too large a range of relaxation times so as to minimise computational cost.

A comparison will be made between an approximately linear flow and a nonlinear flow to demonstrate how the rheology of the polymer changes according to the model in different flow regimes. Different blend compositions of the chains are compared to see the differences in their rheology as well.

Boundary Conditions

The boundary conditions for p, U and τ are given in the table below. A no-slip boundary condition is imposed at both of the walls in the channel, which creates maximum shear in the geometry.

	p	U	τ
Inlet	`fixedValue`	`zeroGradient`	`zeroGradient`
Walls	`zeroGradient`	`noSlip`	`linearExtrapolation`
Outlet	`fixedValue`	`zeroGradient`	`zeroGradient`
frontAndBack	`empty`	`empty`	`empty`

Table 2.3: Table showing the boundary conditions used for the simulation.

The model parameters and boundary conditions have now been established, now pressure differences need to be chosen for the simulation runs. Table 2.4 shows the dimensionless pressure drops, Δp, chosen along with the corresponding dimensionless pressure

gradients, ∇p, that result from these choices. There will be a comparison between the pressure gradients of $\nabla p = 7/40$ and $\nabla p = 3/4$ to demonstrate how the flow changes from an approximately linear regime and a nonlinear regime. The final pressure gradient $\nabla p = 1/2$ is used to make a comparison between different blend compositions of the short and long chains in the polymer blend.

Δp	∇p
7	7/40
20	1/2
30	3/4

Table 2.4: Table showing the dimensionless pressure drops and the corresponding dimensionless pressure gradients used in the simulation.

The Simulations

This section presents the results of the simulations of the bimodal RDP model with a 5% concentration of long chains in a two-dimensional channel. Analysing the Weissenberg numbers of the different flows gives a measure of the nonlinearity in the flow. Orientation and stretch of the polymer chains are to be expected when the corresponding Weissenberg number exceeds 1. Due to the relaxation times of the long chain species being longer than those for the short chain species, the long chain species properties will exhibit conformational change at lower shear rates and so will be the focus of this investigation.

The RDP model characterises constraints on the long chains in two ways. First there are the constraints on the chain from all the other chains in the melt, which forms the *thin tube*. The other is the constraint on a long chain from other long chains over times when the short chain species diffuse away, this is known as the *fat tube*. Within the RDP model the extension of the long chains within the thin and fat tubes are given respectively by,

$$\lambda_L = \sqrt{\frac{tr\underline{\underline{\mathbf{A}}}_L}{3}}, \quad \lambda_{LL} = \sqrt{\frac{tr\underline{\underline{\mathbf{A}}}_{LL}}{3}}. \tag{2.4}$$

When either expression in equation 2.4 is equal to 1, there is no stretch occurring in the respective tube.

Long chain stretch is expected when the stretch Weissenberg number, $\dot{\gamma}\tau_{s,L} > 1$. Since $\tau_{s,L} = 0.2$, the Weissenberg number would be greater than 1 only when $\dot{\gamma} > 5$. This would be the condition for chain stretch to occur in a monodisperse polymer. However, in a blend the onset of long chain stretch depends on the long chain concentration owing to the effect of enhanced stretch relaxation. For example for $\tau_{s,L} = 0.2$ and $\phi_L = 0.05$ the effective stretch relaxation time is $\tau_{s,L}^{eff} = 4$. This means that the onset of long chain stretch in the fat tube is expected to happen for $\dot{\gamma} > 0.25$.

The intermediate regime, where the chains orient but there is no stretch, is where shear-thinning of the fluid occurs. In terms of the orientation Weissenberg number, this

regime occurs when $\dot{\gamma}\tau_{d,L} > 1$. Therefore, shear thinning is expected to occur when $\dot{\gamma} > 0.1$ because $\tau_{d,L} = 10$. So when $0.1 < \dot{\gamma} < 0.25$ the long chain species orient, but no stretch is expected to be observed.

In order to be consistent throughout this investigation, every result is taken from three quarters of the way down the channel. The stretch of the thin and the fat tubes for the different pressure values are given in figure 2.4. Stretch occurs in both the thin and the fat tube, with stretching being more pronounced in the fat tube and closer to the walls of the channel where the shear rate is the highest.

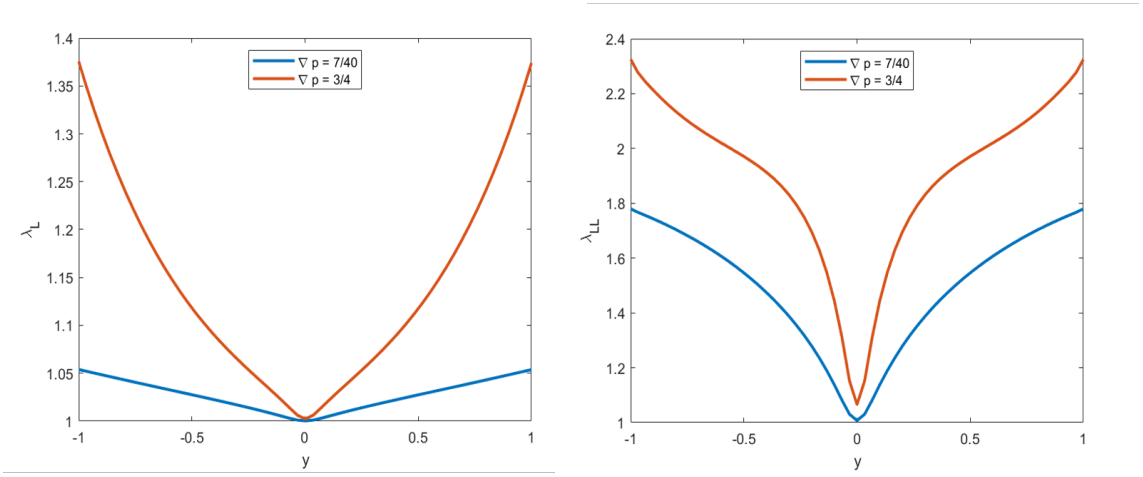


Figure 2.4: A comparison between the stretch of the thin tube (LHS) and the fat tube (RHS) for the two different dimensionless pressure gradients.

The plot on the left of figure 2.5 shows the component of the velocity gradient tensor that corresponds to $\partial U/\partial y$ which gives the shear rate in the channel. Owing to the nonlinear relationship between pressure and shear rate, the different pressure values for the simulations show a significant difference in the shear rate across the channel. When $\nabla p = 7/40$ the variation in shear rate across the channel is approximately linear, whilst for $\nabla p = 3/4$ it is clearly nonlinear. The right hand plot of figure 2.5 shows the same component of the velocity gradient tensor divided by the absolute value of the maximum value of that component to better illustrate the change in shape of the shear rate curve across the channel.

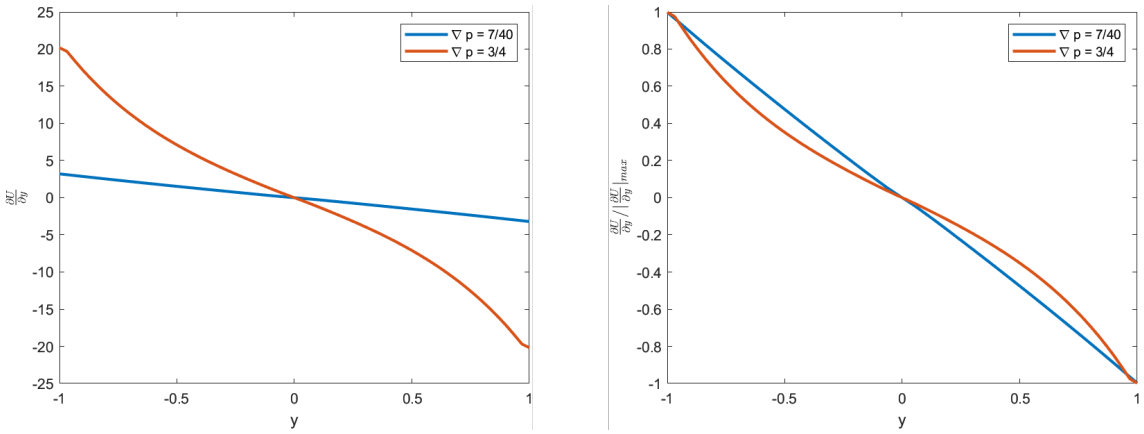


Figure 2.5: Figure showing the component of the velocity gradient tensor that corresponds to $\partial U/\partial y$ for the two different pressure values on the left. On the right the plot shows the same component of the velocity gradient tensor with the maximum value divided out to better illustrate how the shape of the curve changes between the pressure gradients.

Table 2.5 shows the Weissenberg numbers at the wall for each simulation. As expected the simulations where $\nabla p = 7/40$ have much smaller values for the Weissenberg number and hence much less orientation and stretch of the polymer chains is expected.

For $\nabla p = 7/40$ the orientation Weissenberg number is $\dot{\gamma}\tau_{d,L} = 32.12$, so orientation of the long chains would be expected and with that shear-thinning of the polymer melt. However, because the polymer is made up of a 5% blend of the long chain species and 95% the short chain species, the short chain species dominate the rheology and shear thinning behaviour is minimal as the other Weissenberg numbers are less than unity. Therefore, this simulation represents an approximately Newtonian regime for the polymer melt.

	$\dot{\gamma}\tau_{d,L}$	$\dot{\gamma}\tau_{s,L}$	$\dot{\gamma}\tau_{d,S}$	$\dot{\gamma}\tau_{s,S}$
$\nabla p = 7/40$	32.12	0.64	0.32	0.16
$\nabla p = 3/4$	202.06	4.04	2.02	1.01

Table 2.5: Table comparing the Weissenberg numbers for the simulation at the different pressure drop values.

Next, the blend composition of the polymer chains is altered to see how this affects the flow. The dimensionless pressure gradient for this simulation was $\nabla p = 1/2$ in order to show an intermediate flow regime compared to the earlier results. The volume fraction of the long chain species in the melt is altered in 5% increments, so $\phi_L = 0.05, 0.10, 0.15$, respectively for each run.

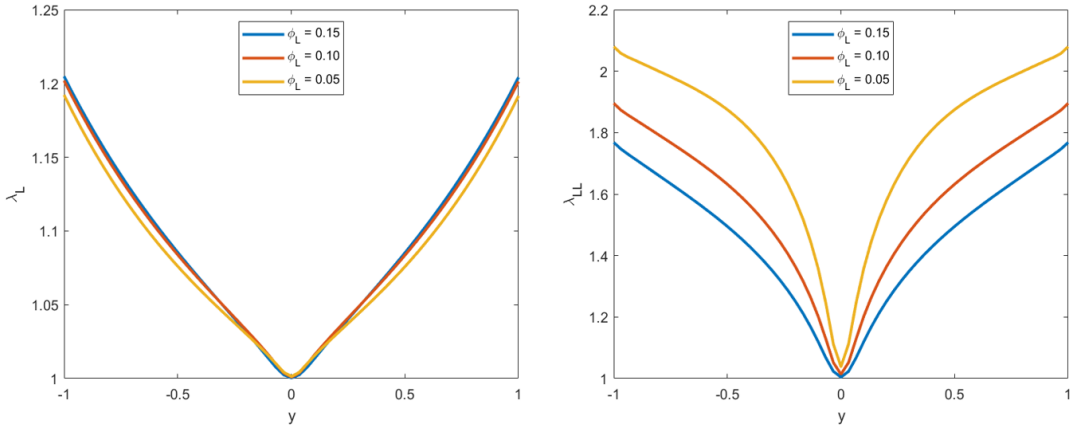


Figure 2.6: Figure showing the stretch of the thin tube (LHS) and the fat tube (RHS) for different long chain concentrations when $\Delta p = 20$.

Figure 2.6 presents the predictions for the thin and fat tube stretch in the channel cross section as the long chain concentration is varied. The thin and fat tube stretch show different patterns when the long chain concentration, ϕ_L is increased. The fat tube stretch decreases as the long chain concentration increases, while the thin tube stretch increases. The decrease in fat tube stretch is due to the effect of enhanced stretch relaxation decreasing as the long chain concentration increases. The small increase in thin tube stretch as the long chain concentration increases is because the increased concentration of long chains means that the long chains are more likely to become entangled with other long chains which increases the fraction of long chain entanglements. This means that the long chains become less constrained by the short chains, so the stretch of the thin tube increases.

It is worth noting that the stretch seen in the simulations is not hugely significant. This is because shear flow is not particularly effective at producing chain stretch. Much larger chain stretch would be seen in a geometry that facilitates extensional flows, such as a contraction geometry. Linking back to the crystallisation problem, nucleation of polymer crystals drastically increases when polymer chains become stretched and aligned. Therefore, understanding in what scenarios chain stretch occurs gives insight as to where nucleation events are more likely to occur.

In the original polySTRAND paper [2], the RDP model is used to compute the largest eigenvalue of $\underline{\underline{\mathbf{A}}}_L - \underline{\underline{\mathbf{I}}} = \Lambda_{\max}$. This is then used to calculate the Kuhn segment nematic order $P_{2,K} = \Lambda_{\max}/N_e$, where N_e is the Kuhn steps per entanglement. This gives insight as to how the nucleation rate changes within the polymer melt because the nucleation rate grows exponentially with the Kuhn segment nematic order. Throughout this section, this method for calculating $P_{2,K}$ is used. However, a modified version for calculating $P_{2,K}$ is outlined later in the thesis in section 3.5.2 which is based on an approximation to the inverse Langevin function [70] to stop $P_{2,K}$ growing beyond physical bounds in strong flow.

Figure 2.7 shows how $\Lambda_{\max}$ evolves across the width of the channel for three different

long chain concentrations for a dimensionless pressure gradient of $\nabla p = 1/2$. As can be seen $\Lambda_{\max}$ is largest at the channel walls, but the change in long chain concentration does not change $\Lambda_{\max}$ significantly. Therefore, although the concentration of long chains affects the fat tube stretch of these long chains, it does not seem to have a significant impact on the overall Kuhn segment nematic order within the polymer melt.

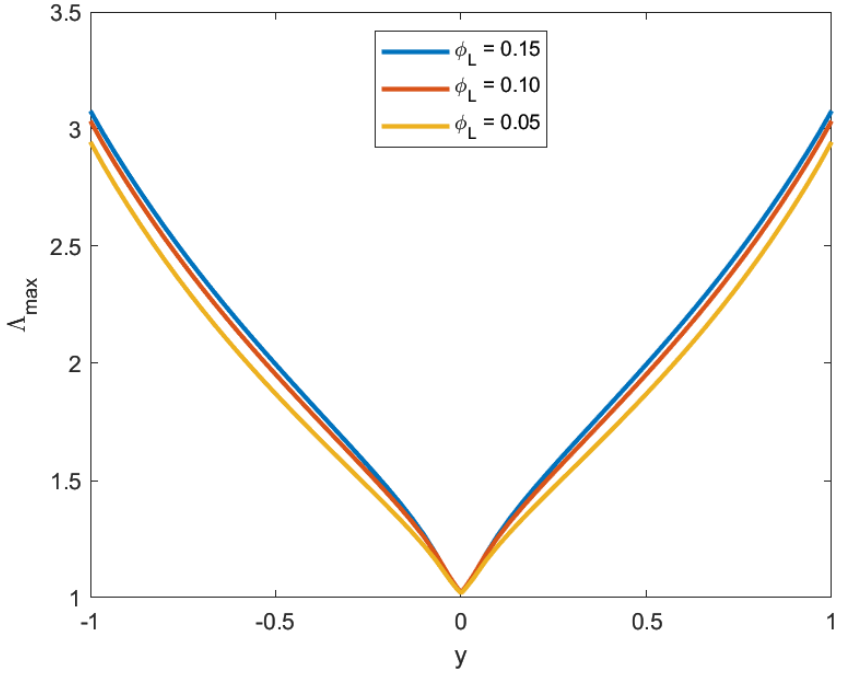


Figure 2.7: Figure showing the largest eigenvalue for $\underline{\underline{\mathbf{A}}}_L - \underline{\underline{\mathbf{I}}} = \Lambda_{max}$ across the width of the channel for different long chain concentrations.

2.5 PolySTRAND Model Validation

Now that we have demonstrated how the RDP model can be solved in OpenFOAM, we move onto the implementation of the polySTRAND model. Building on the outline of the polySTRAND model given in section 1.7.2 we will now present the method for calculating the free energy barrier to nucleation, followed by a validation of the implementation of the polySTRAND model and finally the nucleation rate calculation is presented that follows from the energy barrier calculation.

In order to calculate the nucleus free energy using the polySTRAND model, we require the structural conformation of each chain species $\underline{\underline{\mathbf{A}}}_i$. Using this we can estimate the change in the free energy for monomer attachment for this chain species Δf_i. This component of free energy depends upon the degree of alignment at the Kuhn lengthscale and so is proportional to the Kuhn segment nematic order parameter, $P_{2,K,i}$, which can be computed from $\underline{\underline{\mathbf{A}}}_i$ as the largest eigenvalue, Λ_{max}, of $\underline{\underline{\mathbf{A}}}_i - \underline{\underline{\mathbf{I}}}$ divided by N_e, the number of Kuhn steps per entanglement, so that $\Delta f_i = \Gamma P_{2,K,i}$ with $P_{2,K,i} = \Lambda_{max}/N_e$ where Γ is a constant of proportionality determined experimentally.

Having found Δf_i we can then estimate the form of the energy barrier to nucleus formation using the polySTRAND model. As presented in section 1.7.2, for $q = N_S/N_T$ and in the limit of large N_S and N_T the nucleus free energy is given by:

$$F = N_T \sum_i [qw_i(2\log w_i - \log\phi_i) - v_i \log v_i + (v_i - qw_i)\log(v_i - qw_i) - v_i \Delta f_i] + N_S \log q - \varepsilon_B N_T + \mu_S S(N_T, N_S), \quad (2.5)$$

where Δf_i is the change in free energy due to chain deformation for adding a monomer to the nucleus, $S(N_T, N_S)$ is the surface area of the nucleus, ϕ_i is the volume fraction of each chain species in the melt, ε_B is the bulk energy reduction from crystallising a monomer from the isotropic melt, μ_S is the surface energy per unit area and v_i & w_i represent the fraction of stems and monomers for each chain species respectively.

Equation 2.5 is used to calculate the nucleation barrier in the following way. Once the nucleus free energy for a given number of monomers and stems, $F(N_T, N_S)$ is calculated, this is minimised with respect to N_S for a given N_T to give N_S^*, the optimum number of strands for the given number of monomers. Leading order fluctuations about N_S^* are accounted for using

$$F_{pS}(N_T) = F(N_T, N_S^*) + \ln\left(\frac{1}{2\pi}\frac{\partial^2 F}{\partial N_S^2}_{N_S = N_S^*}\right), \quad (2.6)$$

where $\partial^2 F/\partial N_S^2$ is calculated via the standard central difference expression. This method is repeated to compute the barrier for undeformed chains, F_{pS}^q.

The last step is to calculate the change in the barrier due to chain deformation, $F_{pS}^q - F_{pS}$ and subtract this from a calculation of the quiescent GO model barrier, F_{GO}^q [61] to give the overall nucleation barrier,

$$F_{\text{final}}(N_T) = F_{GO}^q(N_T) - (F_{pS}^q(N_T) - F_{pS}(N_T)). \quad (2.7)$$

Figure 2.8 shows a re-creation of figure 1(a) from the supplementary information of the polySTRAND paper [2] which gives validation of our implementation of the polySTRAND model. The algorithm used by Jolley and Graham [64] uses an asymptotic approximation to the full GO model calculation that is only valid for large N_T. Therefore the values of small N_T in the figure are calculated using the full quiescent GO model calculation. Since we are mostly interested in the height of the nucleation barrier, we decided not to plot the nucleation barrier for small N_T in our re-creation.

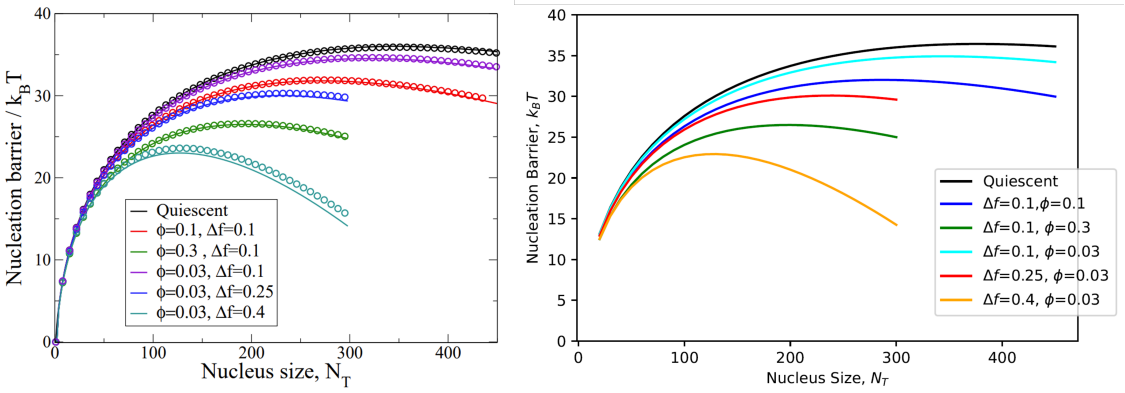


Figure 2.8: Figure showing the original plot of the nucleation barrier for bimodal polymer blends in the polySTRAND paper [2] (left) and the re-creation of said figure from our version of the polySTRAND model (right).

With our calculations agreeing with those presented in the polySTRAND paper [2], we are confident that our implementation of the polySTRAND model is suitable for use in many different situations.

We now present how the nucleation barrier is used to calculate the nucleation rate. The quiescent nucleation rate $\dot{N}_q$ is calculated separately using the approximation for the average nucleation time, $< t_F >$, in the GO model inspired by Hamer et al [71]. This is valid for barrier heights above a few k_BT, to form a nucleus of size N_T given by

$$< t_F >= \kappa_{nuc} \exp(\Delta F^*), \tag{2.8}$$

where ΔF^* is the nucleation barrier height and κ_{nuc} is a kinetic pre-factor, given by,

$$\kappa_{nuc} = \frac{\tau_0}{2n^{*2/3}} \left[1 + \sqrt{-\frac{2\pi}{F''(n^*)}} \exp\left(\frac{-\alpha^2}{2n^{*2} F''(n^*)} + \frac{\alpha}{n^*} \right) \right] \sum_{i=0}^{n^*} \exp(-\Delta F(i)). \tag{2.9}$$

Here n^* is the critical nucleus size, τ_0 is the timescale for monomer attachment/detachment and $\alpha = 0.8$. $F''(n^*)$ is the curvature of the nucleation barrier at its peak, which is computed by the central difference expression. The barrier base dominates the sum in equation 2.9 so only the first few terms are required. The nucleation rate is then given by

$$\dot{N} = \frac{1}{< t_F >}. \tag{2.10}$$

The kinetic pre-factor κ_{nuc} changes only slightly due to flow. Therefore the quiescent nucleation rate, $\dot{N}_q$, is calculated using equation 2.10 and the flow induced nucleation rate, $\dot{N}_\Lambda$, is approximated using the heights of the energy barrier and the quiescent nucleation rate like so:

$$\dot{N}_\Lambda = \dot{N}_q \exp(\Delta F_q^* - \Delta F_\Lambda^*), \tag{2.11}$$

where ΔF^*_Λ is the height of the nucleation barrier under flow and ΔF^*_q is the quiescent nucleation barrier height.

This calculation gives the rate of nucleation per Kuhn step, whereas experiments measure the rate per unit volume. Hence the rate of nucleus formation per unit volume is given by,

$$\dot{N} = \rho_K \dot{N}_\Lambda, \tag{2.12}$$

where ρ_K is the Kuhn step density of the polymer.

2.6 Idealised Blend in Complex Flow

Having validated the implementation of both the RDP and polySTRAND models, we now proceed to combine them to investigate FIC in a complex flow geometry. We consider a simple system of a two component polymer blend made up of a long chain species and a short chain species. We also make the assumption that velocity gradients within the flow are not large enough to significantly align the short chains, so that the long chains are the only contributing factor to the flow-induced crystallisation. This minimises the complexity of the model. In chapter 3 we consider the case where multiple chain species contribute to the flow induced enhancement of crystallisation. The long chain concentration is chosen to be 15%, which is a sufficient concentration to accelerate the crystallisation of the melt significantly. The flow is also assumed to be isothermal in this investigation. When the temperature is kept constant the complexity of the investigation is significantly reduced as there are many variables in the models that depend on temperature. The non-isothermal case is considered in chapter 4. Crystallisation here is considered slow enough so that it does not affect the melt rheology.

The workflow that is used throughout this thesis runs as follows:

- We pre-calculate nucleation rate predictions via the polySTRAND model outside of openFOAM.
- A least squares regression is used to calculate the constants of our interpolation function. These constants are then input into openFOAM.
- OpenFOAM is used to simulate the polymer flow and the growth of crystals in our geometry.

It is worth noting that as long as the molecular weight distribution remains constant, the first two steps in our workflow do not need to be repeated between simulation runs. In chapter 3 and 4 we outline our updated interpolation functions that account for polydispersity and temperature changes respectively.

Figure 2.9 shows how the energy barrier to nucleation changes as a function of the value of Δf for the long chains for this bimodal blend. Increasing Δf both reduces the

maximum height of the barrier and also reduces the nucleus size corresponding to this maximum.

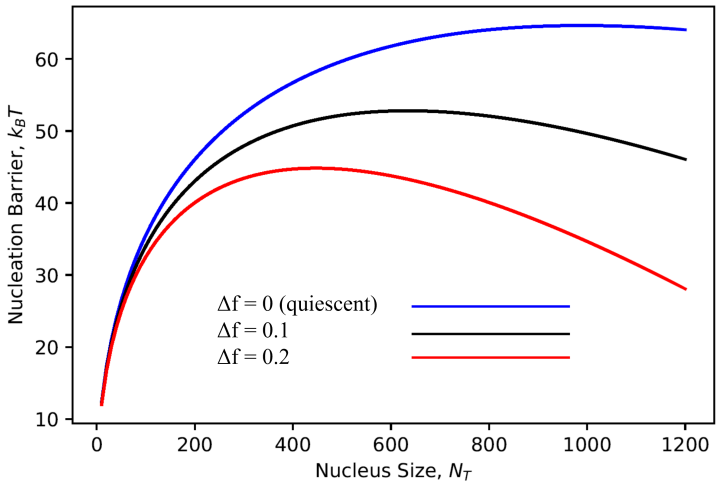


Figure 2.9: This figure shows the changing nucleation energy barrier due to deformation of polymer chain species of a bimodal blend. The two species are concentrated as 15% long chain species and 85% short chain species. The Δf shown in the plot is for the long chain species as the short chain species is assumed to not affect the nucleation rate within the polymer melt in this case. As Δf increases, the energy barrier height decreases which illustrates the polymer monomers and chain strands are more likely to attach to growing crystal nucleus.

2.6.1 Growth Rate Calculation

After the nucleation rate, $\dot{N}$, has been calculated, the Schneider rate equations [50] are then used to determine the evolution of the spherulite density, radius, surface area and volume. They are a sequence of coupled linear ODEs, given below, and are relatively computationally inexpensive to solve:

$$\begin{aligned} \frac{\mathrm{D}N}{\mathrm{D}t} &= \dot{N}, \\ \frac{\mathrm{D}R}{\mathrm{D}t} &= 2G_sN, \\ \frac{\mathrm{D}S}{\mathrm{D}t} &= 4\pi G_sR, \\ \frac{\mathrm{D}V}{\mathrm{D}t} &= G_sS. \end{aligned} \tag{2.13}$$

Here V is the volume fraction of crystals, S is the total surface area of the crystals, R is proportional to the sum of the spherulite radii, N is the number density of crystals, $\dot{N}$ is the nucleation rate and G_s is the spherulitic growth rate.

The Schneider rate equations above consider the growth of isolated spherulite crystals and so do not account for the impingement between neighbours. Therefore V in equation 2.13 can become greater than one. To correct for this, the Avrami equation [51] introduces

a new quantity, ξ_g, that is the volume fraction corrected for impingement, which is the degree of space filled by crystals given by

$$\xi_g = 1 - \exp(-V), \tag{2.14}$$

where ξ_g is the degree of space filling. Hence when $V \ll 1$ then $\xi_g = V$ and when $V \to \infty$ then $\xi_g \to 1$.

2.6.2 Calculating the Nucleation Rate

For the case of an isothermal flow the quiescent nucleation rate, $\dot{N}_q$ is the same at all points in the flow and so can be calculated once in the initialisation step. However, the flow induced nucleation rate, $\dot{N}$ depends upon the local value of Δf and so needs to be determined locally at each point in space and time. The computation of the nucleation barrier requires several embedded numerical methods, namely a solution to a non-linear equation within two nested optimisation steps: a minimisation within a maximisation. Running this procedure in every cell of the mesh at every time step in our simulation would be computationally expensive (though not impossible). Instead we determine a numerical approximation of $\log(\dot{N}_\Lambda/\dot{N}_q)$ as a function of Δf by pre-calculating its value for the range of $\Delta f = 0$ to 1 with 200 data points within this range. Noting for small values of Δf that $\log(\dot{N}_\Lambda/\dot{N}_q)$ increasing exponentially with Δf, a cubic interpolation was fit to this function in the form

$$\log\left(\frac{\dot{N}_\Lambda}{\dot{N}_q}\right) = a\Delta f + b(\Delta f)^2 + c(\Delta f)^3, \tag{2.15}$$

where the coefficients a, b and c are determined using least squares regression.

Figure 2.10 shows the fit between the calculated nucleation rate from the polySTRAND model and the result of the least squares regression.

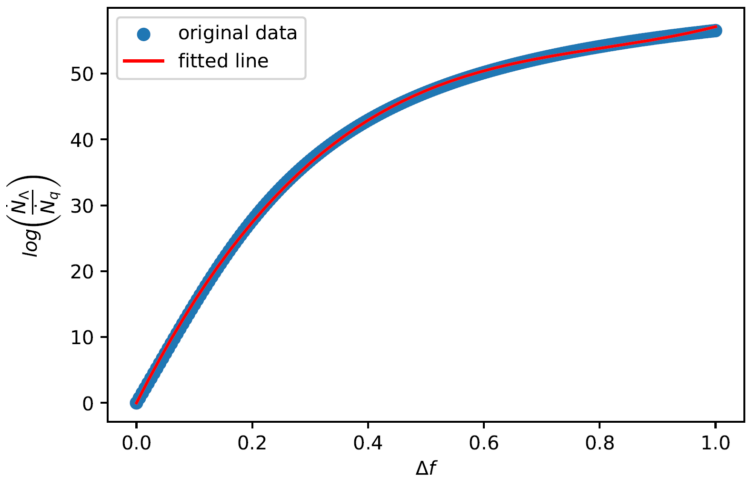


Figure 2.10: Figure showing how the cubic interpolation was fit to the data for the nucleation rate.

This procedure removes the need to calculate nucleation rates using the full polySTRAND model within openFOAM. Instead the calculation of the regression coefficients along with the quiescent nucleation rate is carried out beforehand in Python with the results written into the input files of the openFOAM simulation.

The gain in computational efficiency is clear from the simplicity of the polynomial that is used to calculate the logarithm of the normalised nucleation rate. This should be compared with the multiple calculations that the full polySTRAND model would go through to get the same result, namely solving a non-linear equation within a minimisation, all within a maximisation. This interpolation method is extended to polydisperse systems in section 3.6.1.

2.6.3 Flow Geometry

The flow geometry was chosen to match the planar contraction-expansion geometry studied experimentally by Scelsi et al [5] shown in Figure 2.11. The geometry in our simulation is 2-dimensional. This complex flow includes regions of both shear and extension, and Scelsi et al found clear evidence of flow induced crystallisation in the form of localised areas of crystallisation corresponding to where polymer crystals have been advected downstream from regions of high velocity gradients. However, although we have taken the geometry from Scelsi et al, we have not attempted to match the polymer rheology or flow-rates of their experiment. Instead we have chosen values more typical of an industrial injection moulding process where the timescales are much shorter and the velocity gradients much larger. Moving forward we will be using dimensional units rather than the non-dimensionalisation used in section 2.4.1. This is because we want to closely replicate the conditions used by Scelsi and with the geometry having dimensions, it made sense to use dimensional units everywhere.

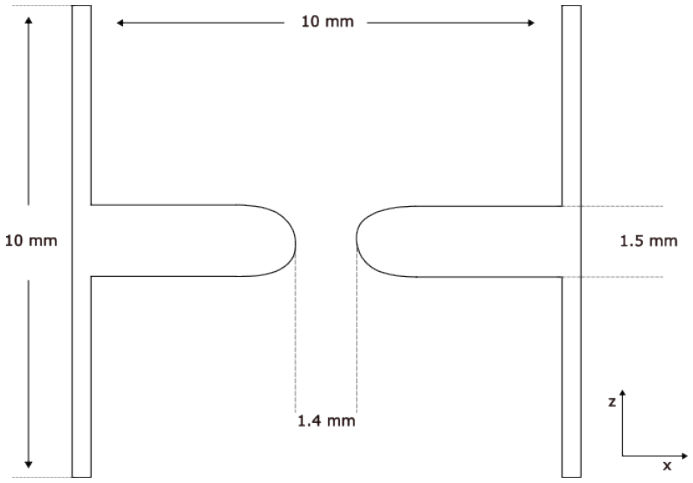


Figure 2.11: Figure showing the mesh dimensions used in our investigation that is a 2-dimensional version of the geometry used by Scelsi et al [5].

The mesh is symmetric across the channel, so only half of the full domain needed to be constructed. Figure 2.12 shows a sketch of the mesh created along with the rendering of the mesh in openFOAM. The sketch also shows the direction of the flow. The mesh was constructed from 30 blocks, shown in the left-hand figure 2.12 each subdivided into quadrilateral finite volume elements.

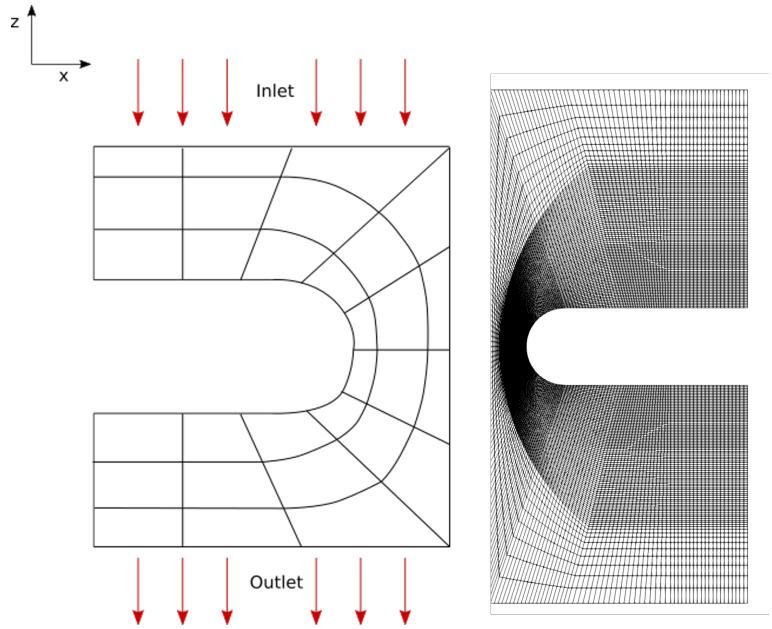


Figure 2.12: A sketch showing the blocks of the mesh and the direction of the flow (left) along with a rendering of the mesh in openFOAM before any simulation has been run.

Statistic	**Value**
Points	28542
Cells	14000
Faces	56270
Internal Faces	27730

Table 2.6: Table showing the mesh statistics for the simulation.

Grading was used to increase the resolution near the contraction walls where the highest shear-rates are found such that the resolution between mesh blocks was smooth to improve computational stability. The grading used was a 5:1 grading from the edge of the mesh to the boundary of the first layer. As the shape of the cells was constrained to be quadrilateral, it was necessary to divide the mesh up into 30 blocks so that the deformation of cells away from being rectangular was kept to a minimum. Minimising the amount of deformed cells from a rectangular shape further increases computational stability.

Table 2.7 shows the boundary conditions used in the simulation. The `zeroGradient` condition applies a zero-gradient condition from the cell centred field onto all faces. The `fixedValue` condition allows for custom values to be input into the simulation, and is used for our inlet and outlet pressure values to allow for control of the flow, full details of how the flow is controlled is given below. The `noSlip` condition constrains the velocity to be zero at the walls. Including the `symmetryPlane` condition allows our mesh to be symmetric about our chosen centre line. The `linearExtrapolation` condition applies a linear extrapolation from the cell centre value to a fixed value at the walls, with the value at the wall being zero.

	p	U	τ
Inlet	`fixedValue`	`zeroGradient`	`zeroGradient`
Walls	`zeroGradient`	`noSlip`	`linearExtrapolation`
Outlet	`fixedValue`	`zeroGradient`	`zeroGradient`
frontAndBack	`empty`	`empty`	`empty`
Symmetry	`symmetryPlane`	`symmetryPlane`	`symmetryPlane`

Table 2.7: Table showing the boundary conditions used for the simulation.

2.6.4 Flow Simulations

The fluid flow is developed by imposing a non-zero pressure difference between the inlet and outlet. Following the approach of Azahar [7], a pressure ramp was used to avoid a rapid change in fluid velocity at the beginning of the simulation which can cause the simulation to crash. The pressure ramp is of the form $P = \alpha(1 - \exp(-\beta t))$ where α is the

target pressure drop and β is a factor that controls the speed of the ramping and t is time. With our simulations running until $t = 0.5$ s the parameter β was chosen so that duration of the pressure ramp was short compared to the time of the simulation. Specifically for our investigation, $\alpha = 0.91$ MPa and $\beta = 220\ \text{s}^{-1}$.

Next we discuss the choice of the relaxation times used in the RDP in the simulation. The relaxation times were not chosen to replicate the polymer used by Scelsi [5], rather they were chosen so that there was sufficient separation between the relaxation times of the long and short chains and that the simulation would be viable i.e. the short chain relaxation time was not so short that the time-step did not need to be too small. These times were also chosen so that the long chain species stretches, whilst the short chain species does not. This is consistent with the assumption that only the long chain species will contribute to the flow induced nucleation rate. Our choice of relaxation times also has sufficient difference between $\tau_{d,L}$, $\tau_{s,L}$ and $\tau_{d,S}$ to see the effects of enhanced stretch relaxation time, as outlined in table 2.8.

Parameter	**Long (L)**	**Short (S)**
$\tau_{d,i}$	0.01 s	0.0001 s
$\tau_{s,i}$	0.0002 s	0.00005 s
ϕ_i	0.15	0.85
G_N^0	1 MPa	

Table 2.8: Table showing the rheological parameters used in the RDP model.

To verify that the relaxation times were chosen in such a way so that the long chain species is contributing the most to the crystallisation of the polymer melt we need to compare the Weissenberg numbers, Wi, of the respective relaxation times. The Weissenberg number measures the product of the velocity gradient and the relaxation time. When the Weissenberg number is greater than unity, the chain species will become oriented or stretched by the flow. The $\tau_{d,I}$ relaxation time corresponds to the orientation of the respective chain species and the $\tau_{s,I}$ relaxation time corresponds to the stretch of the respective chain species.

In this investigation the maximum fluid velocity is around 1.6 m/s in the contraction region. Based on simple channel flow for a Newtonian fluid, this would be expected to produce a shear rate, $\dot{\gamma} = 2U/d$, of around 5000 s^{-1}. Based on this value the Weissenberg number for each of the relaxation times close to the wall of the contraction is given in table 2.9.

	Wi
$\tau_{d,L}$	50
$\tau_{s,L}$	1
$\tau_{d,S}$	0.5
$\tau_{s,S}$	0.25

Table 2.9: Table showing the Weissenberg numbers for the various relaxation times that were chosen for the simulation.

The values of the Weissenberg numbers in table 2.9 show that the long chain species in our simulation should be oriented but not stretched, whilst there would be no orientation or stretching of the short chain species. Therefore the choices for the relaxation times should give us the behaviour in the melt that we want to show in this study.

The full simulation was run for a total of 0.5 s with a time step of 10^{-5} s. Such a small value is necessary due to the values chosen for the relaxation times in the RDP model. The pressure was ramped up to 0.91 MPa at the inlet and this value was reached at 0.175 s of the simulation run. This gave a velocity of 1.6 m/s in the contraction region at the end of the simulation. The experimental investigation that was the inspiration of this investigation used a velocity of 10 mm/s in their experiments, for a material with quite different rheology. Therefore we are looking for qualitative agreement between the studies and this would provide the platform for more in-depth simulations using our models.

2.6.5 Results

In this section we discuss the OpenFOAM results of our FIC investigation of an idealised bimodal polymer blend in a contraction geometry. Accelerated crystallisation is observed due to the high shear rate in the contraction region, along with the extensional force that is created from the contraction.

Figure 2.13 shows the results of the magnitude of velocity field and the velocity gradient at 0.5s of the simulation. As can be seen, both the velocity and the velocity gradient are largest in the contraction region. The velocity reached 1.6 m/s, as expected, and the velocity gradient reaches a maximum of 33000 s^{-1} which is larger than our estimate of 5000 s^{-1}, this discrepancy can be attributed to shear thinning of the polymer, however this is highly localised to the point on the wall at the centre of the contraction. The relaxation times chosen still give the desired effect in the study. The crystal nucleation and the nucleation rate are also largest in the contraction region and so a conclusion can be drawn that the larger the velocity increase the larger the increase in the crystal nucleation in the polymer melt.

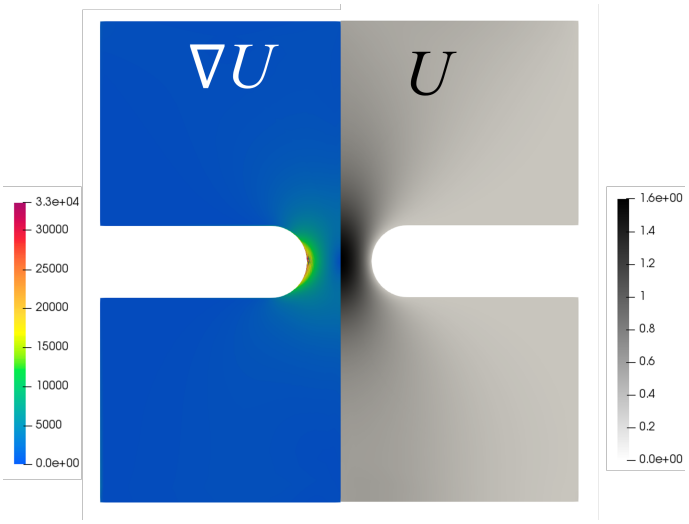


Figure 2.13: Simulation results that show ∇U on the left and U on the right after 0.5 s of the simulation.

In figure 2.14 the results of the thin tube stretch of the long chain species, λ_L and Δf from our simulations are presented. The stretch reaches its maximum value $\lambda_L = 1.2$ at the wall of the contraction region, which is where the nucleation rate is accelerated the most. The largest value for $\Delta f = 0.24$ is at the wall of the contraction region due to the shear dominated flow of the simulation, however there is some increase in the centre of the contraction which represents some increase due to extensional flow.

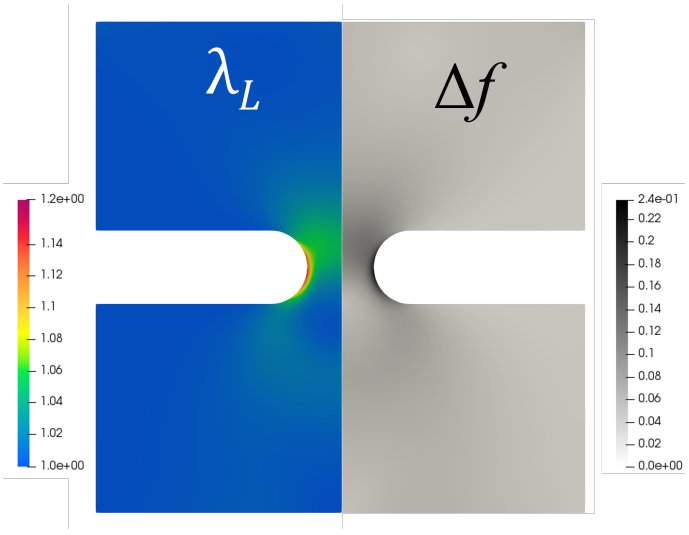


Figure 2.14: Simulation results that show λ_L on the left and ξ_g on the right after 0.5 s of the simulation.

Figure 2.15 presents the results after 0.5 s for the crystallite number density and the volume fraction N and ξ_g. The region of high values of N has the form of a 'fang' that grows out from the separation point on the wall downstream of the contraction region as the simulation progresses. The results for ξ_g also show this 'fang' structure. However it

is worth noting that the magnitude of ξ_g in the simulation is of the order 0.0001 hence the degree of space filling is negligible and should only serve as a proof of concept for the model that has been implemented.

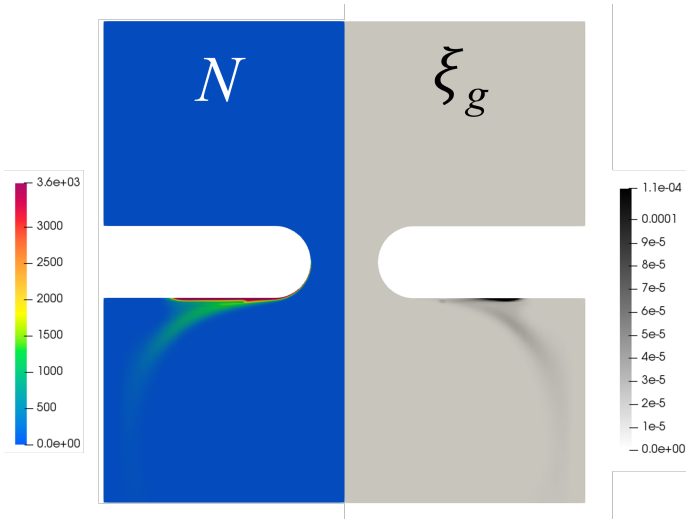


Figure 2.15: Figure showing the distribution of values of N and ξ_g after 0.5 s of the simulation.

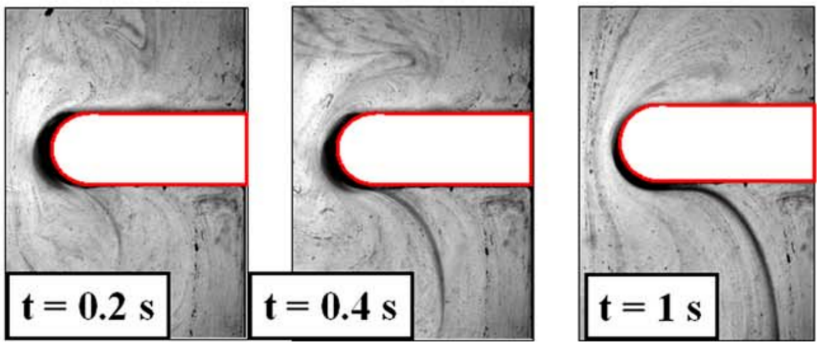


Figure 2.16: Figure taken from Scelsi et al [5], showing the development of a 'fang' of high crystal density.

Figure 2.16 shows the experiment that inspired this investigation and that the spatial distribution of crystal development agrees well with our simulation. It is worth noting that the experimental investigation was conducted using an HDPE that contains some long chain branching and was modelled in the paper using a pom-pom model for a branched polymer. Nevertheless, there is qualitative agreement between our simulations and these experiments.

2.7 Conclusion

This initial investigation shows how polymer crystallisation is enhanced in specific locations of the flow geometry. We implemented the RDP model in openFOAM and linked

it to the polySTRAND model [2] for crystal nucleation via an interpolation function that captured the nucleation rate predictions from the polySTRAND model. This allowed us to produce our results using openFOAM using a computationally efficient workflow. This model was then tested using the contraction geometry used in the experiments conducted by Scelsi et al. [5] which investigated flow induced crystallisation of polyethylene and qualitative agreement was found between the two investigations. This lays the foundation for the more complex investigations of polymer blends that are more representative of those used in real-world polymer processing.

Chapter 3

Polydispersity

3.1 Overview

In the previous chapter we performed exploratory simulations for an idealised two-component blend. However, commercial polymers used in processing are typically highly polydisperse, so in order to study these materials we now introduce full polydispersity into our simulations on flow induced crystallisation.

In order to achieve this, we need to implement a multi-mode version of the RDP constitutive model [1] into OpenFOAM. However, before doing so we will examine how to choose parameters for the RDP model to fit to the rheological measurement of a polydisperse polymer melt. Modifications are also required to our implementation of the polySTRAND model to account for multiple chain species. We need to address how multiple chain species affect the energy barrier to crystallisation, along with a new calculation for the nucleation rate.

3.2 Polydisperse Rolie-Double-Poly Model

We now consider how to implement the RDP model within the RheoTool library within OpenFOAM for the case where we have n interacting chain species. The important details from the multimode RDP model outlined in section 1.5.3 are repeated here. The polymer stress $\underline{\underline{\boldsymbol{\tau}_P}}$ is the sum of the stress contributions coming from the local average conformation of each species $\underline{\underline{\mathbf{A}}}_i$ weighted by their volume fraction, ϕ_i and the finite extensibility function related to their stretch λ_i, [1]:

$$\underline{\underline{\boldsymbol{\tau}_P}} = G_N^0 \sum_{I=1}^{n} \phi_i f_E(\lambda_i) \underline{\underline{\mathbf{A}}}_i \qquad \text{with} \qquad \lambda_i = \left(\frac{1}{3} tr \underline{\underline{\mathbf{A}}}_i \right)^{\frac{1}{2}}. \tag{3.1}$$

Here G_N^0 is the experimental plateau modulus. The finite extensibility function quoted in the RDP publication [1], $f_E(\lambda)$, is given as

$$f_E(\lambda) = \frac{1 - \lambda_{max}^{-2}}{1 - \lambda^2 \lambda_{max}^{-2}}, \tag{3.2}$$

where λ_{max} is the maximum chain stretch in extensional flow. This is the Warner approximation [72] to the inverse Langevin function. However, in this work we use an improved approximation to the inverse Langevin function for our finite extensibility considerations given as,

$$f_E(\lambda) = \left(\frac{3 - \lambda^2/N_e}{1 - \lambda^2/N_e}\right)\left(\frac{1 - 1/N_e}{3 - 1/N_e}\right). \tag{3.3}$$

Where N_e is the number of steps per entanglement segment.

The average conformation tensor, $\underline{\underline{\mathbf{A}}}_i$, accounts for the stresses that come from the interaction of the species i with itself and the other $(n-1)$ species, therefore,

$$\underline{\underline{\mathbf{A}}}_i = \sum_{j=1}^{n} \phi_j \underline{\underline{\mathbf{A}}}_{ij}, \tag{3.4}$$

where $\underline{\underline{\mathbf{A}}}_{ij}$ is the stress conformation tensor on the i-chains coming from their entanglements with the j-chains. This conformation tensor evolves as:

$$\overset{\nabla}{\underline{\underline{\mathbf{A}}}}_{ij} = -\frac{1}{\tau_{d,i}}(\underline{\underline{\mathbf{A}}}_{ij} - \underline{\underline{\mathbf{I}}}) - \frac{2}{\tau_{s,i}}\left(1 - \frac{1}{\lambda_i}\right) f_E(\lambda_i)\underline{\underline{\mathbf{A}}}_{ij} - (\underline{\underline{\mathbf{A}}}_{ij} - \underline{\underline{\mathbf{I}}})\left[\frac{\beta_{th}}{\tau_{d,j}} + 2\beta_{CCR}\frac{f_E(\lambda_j)}{\tau_{s,j}}\left(1 - \frac{1}{\lambda_j}\right)\lambda_i^{2\delta}\right], \tag{3.5}$$

where $\tau_{d,i}$ is the reptation relaxation time and $\tau_{s,i}$ is the stretch relaxation time for each species. The parameter β_{th} is the thermal constraint release coefficient, which we set as $\beta_{th} = 1$ in this work. This coefficient accounts for relaxation due to the reptation of the j chains. In the linear rheology limit, the relaxation rate of the ij mode is the sum of the rates $1/\tau_{d,i}$ and $1/\tau_{d,j}$ from the i and j chains respectively. Therefore the model is consistent with double reptation theory [13] which models the relaxation of two entanglements, where each entanglement involves an interaction between chains of different lengths.

A key feature of this model is that the n^2 individual conformation tensors $\underline{\underline{\mathbf{A}}}_{ij}$ are coupled via the terms involving the stretch variables λ_i and λ_j and therefore this model cannot be incorporated using the multimode formalism in RheoTool, which is based on a linear superposition of uncoupled modes. As a consequence it is necessary to create a stand-alone RDP model in which all n^2 modes are individually represented. Although explicitly coding the equations for each conformation tensor is viable for a small number of modes such as in a bimodal blend for a larger numbers of chain species it is desirable to have an automated means for coding these equations. To achieve this we use a Python script to automatically generate the requisite openFOAM files for the RDP constitutive equations with the desired value of n. Details of this script are given in the University of

Leeds data repository via the link `https://doi.org/10.5518/1581`.

3.3 Linear Viscoelastic Spectrum

Now that we have a method for implementing a polydisperse rheological model in OpenFOAM we need to establish an effective method for accurately determining the RDP spectrum in our simulations. To do this we consider the linear viscoelastic spectrum of the RDP model.

Since the RDP model is based on double reptation [13], the linear relaxation modulus is given by,

$$G(t) = G_N^0 \sum_{i=0}^{n} \sum_{j=0}^{n} \phi_i \phi_j e^{-t/\tau_{d,i}} e^{-t/\tau_{d,j}}. \tag{3.6}$$

Here we label the modes in order of increasing relaxation time, so that $\tau_{d,j} > \tau_{d,i}$ when $i > j$, and $i = n$ corresponds to the longest chain species. An additional "0" species has been included to represent fast relaxing chains for which $\tau_{d,0}$ is small compared to the flow timescale and inverse shear-rates. This species acts effectively as a solvent, so it is not explicitly modelled using the RDP equations, but we include it as part of the Maxwell mode fitting procedure outlined below. With these fast relaxing chains included in the summation, the prefactor G_N^0 is the plateau modulus, and the ϕ_i represent the volume fraction of each species, including the short chains.

Equation 3.6 is equivalent to a relaxation spectrum consisting of $n(n+1)/2$ Maxwell modes. However, rearranging equation 3.6 as

$$G(t) = \sum_{i=0}^{n} G_N \phi_i \left(\phi_i e^{-2t/\tau_{d,i}} + 2 \sum_{j>i}^{n} \phi_j e^{-t(\tau_{d,i}+\tau_{d,j})/\tau_{d,i}\tau_{d,j}} \right), \tag{3.7}$$

it can be seen that the relaxation times of modes corresponding to the interactions of the i chains with j chains with $j \geq i$ all lie between $\tau_{d,i}/2$ and $\tau_{d,i}$ and so all have similar relaxation times (since for these modes, $\tau_{d,j} > \tau_{d,i}$). This inspires our strategy, discussed below in Section 3.4, for matching a Maxwell mode spectrum to the RDP model. We call the set of modes corresponding to the interactions of the i chains with j chains with $j \geq i$ the i-th "RDP Maxwell mode" because, although not strictly identical in relaxation time, they are sufficiently close for practical use. It can readily be observed that the number of RDP Maxwell modes corresponds to the number, n, of RDP chain species.

By integrating equation 3.7, we can obtain the contributions to the viscosity in the form

$$\eta = \sum_{i=0}^{n} \eta_i, \tag{3.8}$$

where

$$\eta_i = G_N^0 \left(\frac{\phi_i^2 \tau_{d,i}}{2} + 2\phi_i \left(\sum_{j>i} \phi_j \left(\frac{1}{\tau_{d,i}} + \frac{1}{\tau_{d,j}} \right)^{-1} \right) \right) \tag{3.9}$$

In particular taking the limit in which $\tau_{d,0} \to 0$, but $G_N^0 \tau_{d,0}$ is finite, the terms associated with $i = 0$ reduce to an effective solvent viscosity

$$\eta_S = \frac{G_N^0 \tau_{d,0} \phi_0}{2} \left[\phi_0 + 4 \sum_{j=1}^{n} \phi_j \right]. \tag{3.10}$$

3.4 Obtaining a RDP fit to HDB6

In this section we consider the matching of an RDP model to experimental linear and non-linear rheology data. The flow induced crystallisation experiments of Scelsi et al [5] were performed using the metallocene catalysed polyethylene referred to as HDB6. Although this polymer contains a degree of long chain branching and therefore the RDP model is not strictly applicable we nevertheless use these data illustratively.

In their paper, Scelsi et al [5] use an 11 mode model consisting 3 stretching Pom-Pom modes[73] for the 3 longest relaxation modes and 8 non-stretching Pom-Pom modes. We will instead use a RDP model with 4 species ($n = 4$) that contribute to non-linear viscoelasticity and enhanced crystal nucleation, together with a fast relaxing mode that acts effectively as a solvent.

The relaxation times shown in the Scelsi paper are at 125 °C. We then time temperature shift their relaxation times to 155 °C so that we can match our spectrum to the rheological data that was measured at 155 °C. We do this by dividing all the relaxation times in the Scelsi paper by 7.783. Once we have established our RDP spectrum we then shift our relaxation times back to 125 °C as this is the temperature of our crystallisation simulations.

Relaxation Time	G	Viscosity	Combined Modulus g_i	Combined Viscosity η_i	RDP Mode
100	52.22	5222			
31.62	250.5	7921.506			
10	985.2	9852	1287.92	22995.51	g_4
3.16	2735	8648.829			
1	5942	5942	8677	14590.83	g_3
0.316	11165	3530.68			
0.1	17251	1725.1	28416	5255.78	g_2
0.0316	29668	938.18			
0.01	39120	391.2	68788	1329.38	g_1
0.00316	102308	323.53			
0.001	318631	318.63			solvent

Table 3.1: Table showing how the 11 modes in the spectrum used in Scelsi et al [5] are grouped into RDP modes. The relaxation times correspond to the material at 155 °C and are shifted with respect to the times given in the Scelsi et al investigation.

We note that each Pom-Pom mode takes a Maxwell form in the linear viscoelastic limit. Table 3.1 shows how the modes in the Scelsi investigation were grouped to form the RDP Maxwell modes (as defined above) of our model, by making use of the linear relaxation structure for the RDP model given in equation 3.7. The 3 stretching Pom-Pom modes were combined into the slowest RDP Maxwell mode. The 3 subsequent pairs of non-stretching modes were combined to give the next three RDP Maxwell modes. The remaining two shortest modes were included as contributing towards the solvent. However, since the total combined modulus of all Pom-Pom modes in the Scelsi data is less than the plateau modulus of high density polyethylene, there must be additional solvent contributions arising from shorter modes that aren't captured by the 11 mode spectrum of Scelsi et al.

Our strategy (which is applicable to any Maxwell mode fit to linear viscoelastic data) is to match the modulus g_i and viscosity η_i of the i-th RDP Maxwell mode to the summed modulii and viscosities, respectively, of the associated combined Pom-Pom (Maxwell) modes. This gives us sufficient information to fit two parameters per RDP Maxwell mode, which is enough to obtain the volume fractions, ϕ_i, and the reptation relaxation times, $\tau_{d,i}$ for the respective RDP species (given the one-to-one correspondence between RDP chain species and RDP Maxwell modes).

This strategy is represented in Figure 3.1 which shows how each of the RDP (double reptation) modes contributes to the relaxation spectrum. Each RDP mode (species i entangling with species j) is represented by a small square, with its individual viscosity

contribution indicated. These RDP modes are summed around the "L" shaped regions to give the RDP Maxwell modes. The correspondence of the RDP Maxwell mode to the Pom-Pom (Maxwell) modes of Scelsi et al. are indicated along the upper line, whilst their correspondence to RDP chain species is indicated along the right.

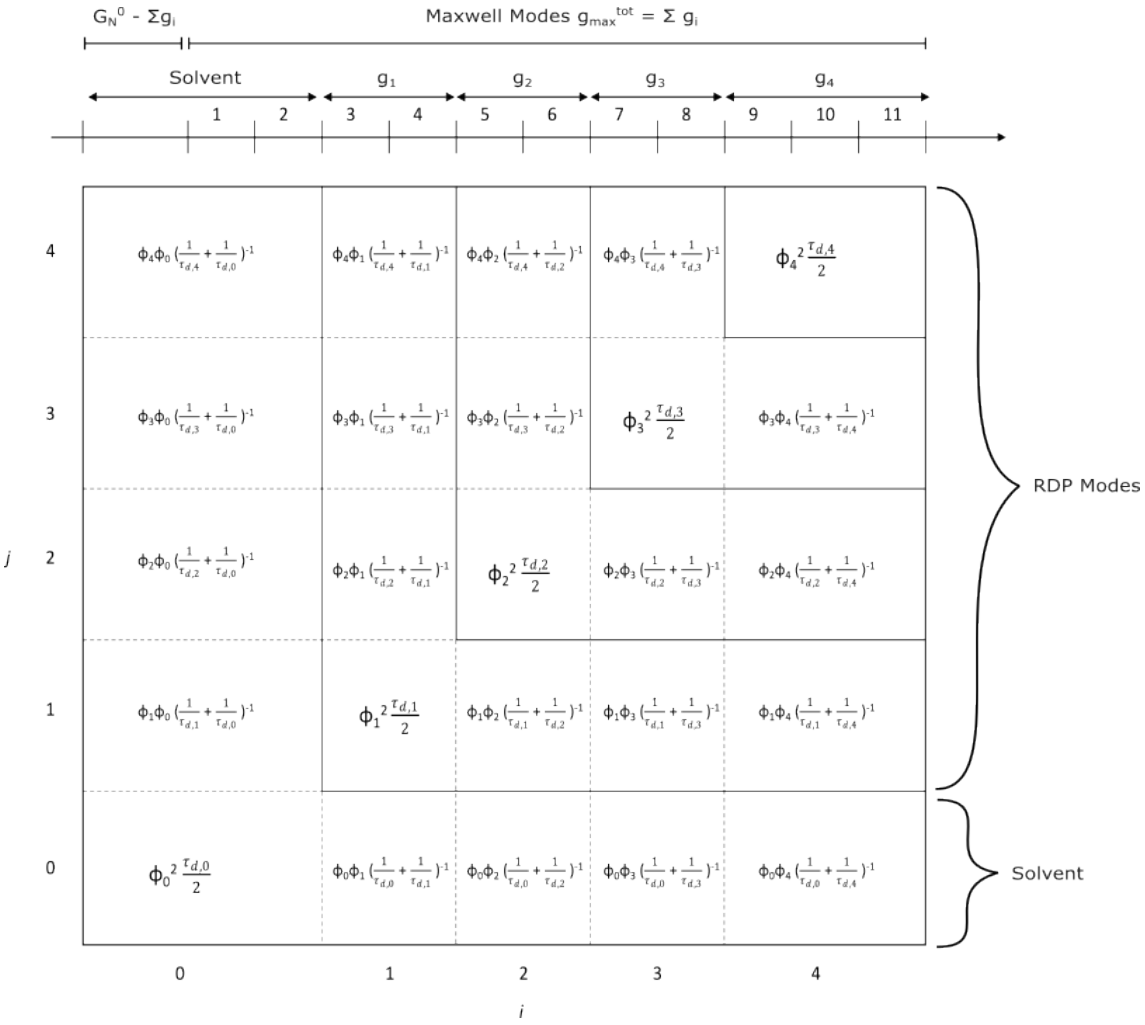


Figure 3.1: Diagram showing how the individual contributions from the 4-mode RDP model are mapped onto the multi-mode Maxwell modes used by Scelsi et al [5]

From equation 3.7 the sum of the elastic moduli for modes with relaxation times between $\tau_{d,i}/2$ and $\tau_{d,i}$ is,

$$g_i = G_N^0 \left(\phi_i^2 + 2\phi_i \left(\sum_{j>i} \phi_j \right) \right).$$

Therefore starting with the longest mode 4 we can obtain the volume fraction ϕ_i from solving sequentially,

$$
\begin{aligned}
g_4 &= G_N^0 \phi_4^2, \\
g_3 &= G_N^0 \left[\phi_3^2 + 2\phi_3\phi_4\right], \\
g_2 &= G_N^0 \left[\phi_2^2 + 2\phi_2\phi_3 + 2\phi_2\phi_4\right], \\
g_1 &= G_N^0 \left[\phi_1^2 + 2\phi_1\phi_2 + 2\phi_1\phi_3 + 2\phi_1\phi_4\right]
\end{aligned}
$$

To find the values of $\tau_{d,i}$ we equate the viscosity contributions from the group of Maxwell modes to the expression for η_i in equation 3.9 again starting from the longest mode 4.

$$
\begin{aligned}
\eta_4 &= G_N^0 \phi_4^2 \frac{\tau_{d,4}}{2} \\
\eta_3 &= G_N^0 \left[\phi_3^2 \frac{\tau_{d,3}}{2} + 2\left(\frac{1}{\tau_{d,3}} + \frac{1}{\tau_{d,4}}\right)^{-1} \phi_3\phi_4\right] \\
\eta_2 &= G_N^0 \left[\phi_2^2 \frac{\tau_{d,2}}{2} + 2\left(\frac{1}{\tau_{d,2}} + \frac{1}{\tau_{d,3}}\right)^{-1} \phi_2\phi_3 + 2\left(\frac{1}{\tau_{d,2}} + \frac{1}{\tau_{d,4}}\right)^{-1} \phi_2\phi_4\right] \\
\eta_1 &= G_N^0 \left[\phi_1^2 \frac{\tau_{d,1}}{2} + 2\left(\frac{1}{\tau_{d,1}} + \frac{1}{\tau_{d,2}}\right)^{-1} \phi_1\phi_2 + 2\left(\frac{1}{\tau_{d,1}} + \frac{1}{\tau_{d,3}}\right)^{-1} \phi_1\phi_3 \right. \\
&\qquad \left. + 2\left(\frac{1}{\tau_{d,1}} + \frac{1}{\tau_{d,4}}\right)^{-1} \phi_1\phi_4\right]
\end{aligned}
$$

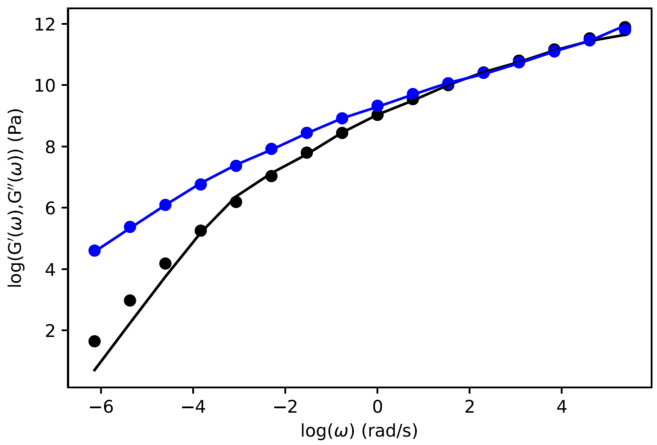


Figure 3.2: This figure compares the measured loss and storage modulii for HDB6 with the results obtained from the RDP model.

Having calculated the values for $\tau_{d,i}$ and ϕ_i we then verified that these values match to the linear relaxation data from HDB6 [5] using the RepTate software [74]. From the computed values of ϕ_i we calculate the remaining volume fraction $\phi_0 = 1 - \sum \phi_i$ that is treated as a solvent. We obtain an effective relaxation time for this mode by fitting to

the high frequency linear viscoelastic data giving effective relaxation time of 5×10^{-4}s corresponding to a solvent viscosity of 641 Pa s. The comparison with the measured linear viscoelastic data for HDB6 is given in Figure 3.2, which shows a good fit to both the storage and loss modulus, G' and G'' over the frequency range from 10^{-2} to 10^{2} s^{-1}.

Since the solvent term is not explicitly included in the RDP model it is convenient to introduce a normalised volume fraction defined as

$$\phi_i^* = \frac{\phi_i}{1-\phi_0}.$$

We then use these normalised volume fractions in the RDP model, and when computing the stress we use a renormalised modulus of

$$G_N^* = (1-\phi_0)^2 G_N^0.$$

The stretch relaxation times, $\tau_{s,i}$, cannot be obtained from linear rheology, but need to be determined from nonlinear rheology. Here, since we are aiming only at a qualitative comparison with the HDB6 data of Scelsi et al[5], the matching of the HDB6 data will be imperfect. The HDB6 material contains long chain branches, so the RDP model does not strictly apply. The values in table 3.2 were chosen so that only the longest mode, 4, becomes stretched at the highest shear-rates in the simulations, and provides an approximate fit to the transient extensional viscosity of HDB6 at $\dot{\varepsilon} = 0.3$ extension-rate, shown in figure 3.3. However, as expected, this does not provide a good fit to other extension rates.

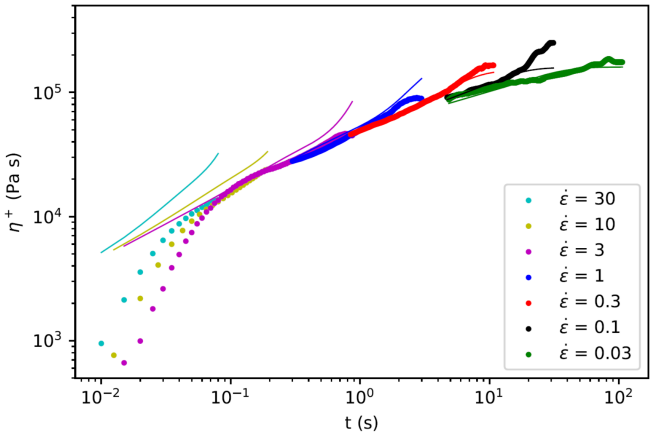


Figure 3.3: Comparison of the extensional viscosity, η^+, for HDB6 (thick lines) with the predictions from our RDP model (thin lines). It should be noted that our model is not intended to provide a detailed fit to this data, but does approximately match the degree of extension hardening at $\dot{\varepsilon} = 0.3$.

Chain Species	ϕ_i^*	ϕ_i	$\tau_{d,i}$	$\tau_{s,i}$
solvent		0.7668	5×10^{-4} s	5×10^{-4} s
1	0.40143	0.09361	0.02305 s	0.01 s
2	0.2936	0.06847	0.23225 s	0.01 s
3	0.19537	0.04556	2.2958 s	0.01 s
4	0.1096	0.02556	35.7 s	0.5 s

Table 3.2: Table showing the volume fractions and relaxation times that were calculated to match the HDB6 rheology at 155°C from reference [5].

All the rheology data for HDB6 [5] were measured at 155°C, whereas, the crystallisation experiments were performed at 125°C. Consequently to perform calculations at this lower temperature a time-temperature shift where all the relaxation times are multiplied by 7.783 [5] was applied, the results of this are given in table 3.3. A similar time temperature shifting protocol was used by Scelsi et al [5] when determining their Pom-Pom spectrum. This also increases the solvent viscosity to 4996 Pa s.

Chain Species	$\tau_{d,i}$	$\tau_{s,i}$
1	0.17939815 s	0.07783 s
2	1.80760175 s	0.07783 s
3	17.8682114 s	0.07783 s
4	291.8625 s	3.8915 s

Table 3.3: Table showing the new time temperature shifted relaxation times for each RDP mode that were ultimately used in the investigations.

3.5 FIC Considerations

3.5.1 PolySTRAND Model

The polySTRAND model [2], outlined in section 1.7.2, gives a calculation of the rate of crystal nucleation in a flowing polydisperse polymeric material. In the published polySTRAND model, $P_{2,K,i}$ was computed as the largest eigenvalue, Λ_{max}, of $\underline{\underline{\mathbf{A}}}_i - \underline{\underline{\mathbf{I}}}$ divided by N_e, the number of Kuhn steps per entanglement, so that $\Delta f_i = \Gamma P_{2,K,i}$ with $P_{2,K,i} = \Lambda_{max}/N_e$ where Γ is a constant of proportionality determined experimentally. However, this method for calculating $P_{2,K,i}$ is inconsistent with the finite extensibility calculation, so that under strong flow $P_{2,K,i}$ can take values greater than one, which are unphysical. Consequently, in section 3.5.2, we propose a modified method for calculating $P_{2,K,i}$ that corrects this.

3.5.2 New Finite Extensibility Calculation

In the published polySTRAND model [2], the bond orientation for species i was calculated as

$$P_{2,K,i} = \frac{1}{N_e}(\Lambda_{i,max}^{(A)} - 1) \tag{3.11}$$

where $\Lambda_{i,max}^{(A)}$ is the largest eigenvalue of the orientation tensor $\underline{\underline{\mathbf{A}}}_i$. We found that with this definition, in strong flow the value for $P_{2,K,i}$ was growing above 1, which is not physical. In order to rectify this we have developed a new method for calculating $P_{2,K,i}$.

This new method was developed by Daniel Read and the full derivation is shown in Appendix A.1. We will show the final results of the derivation which is what is used to calculate $P_{2,K,i}$ for the remainder of this thesis.

The final result for $P_{2,K,i}$ is given by:

$$P_{2,K,i} = \frac{1}{2}\left(\frac{X_i^2 + 3 - 3X_i \coth X_i}{X_i^2}\right)\left(\frac{\Lambda_{i,max}^{(A)}}{\lambda_i^2} - 1\right), \tag{3.12}$$

with X_i evolving as

$$X_i = 3Y_i f_E(\lambda_i) = \frac{3\lambda_i}{\sqrt{N_e}} f_E(\lambda_i) = \frac{3\lambda_i}{\sqrt{N_e}}\left(\frac{3 - \lambda_i^2/N_e}{1 - \lambda_i^2/N_e}\right)\left(\frac{1 - 1/N_e}{3 - 1/N_e}\right), \tag{3.13}$$

where N_e is the number of steps per entanglement segment.

In the limit of strong orientation and strong stretching, $\Lambda_{i,max}^{(A)} \approx 3\lambda_i^2$ and X_i becomes large so that $P_{2,K,i}$ approaches a value of one, as required.

In the limit of small stretch, equation 3.12 reduces to

$$P_{2,K,i} = \frac{3}{10N_e}(\Lambda_{i,max}^{(A)} - \lambda_i^2).$$

Comparing with equation 3.11 we see that the form is similar, but the numerical prefactor is smaller, so that when using equation 3.12 it becomes necessary to use a larger value of the proportionality constant Γ in $\Delta f_i = \Gamma P_{2,K,i}$ to obtain similar results as compared to the original polySTRAND formulation. This gives rise to the distinction between Γ and Γ_{FE} shown in section 3.6.3.

3.6 Simulation of Flow-induced Crystallisation

To use the above models to simulate flow-induced crystallisation in a complex flow, we developed an implementation of these equations using the rheoTool [4] toolbox within openFOAM [3].

3.6.1 Efficient computational implementation of polySTRAND model

Although the polySTRAND model is significantly faster computationally than the original GO model simulations, it still requires several numerical steps to compute a free energy

barrier toward nucleation. These are computationally costly to implement at each point in time and space in a large flow simulation. Therefore we seek an efficient way to incorporate the polySTRAND calculation of the nucleaction rate within the simulations. We have found that the results of the polySTRAND model can be pre-computed for a reasonable range of chain orientations, and then embedded into openFOAM via an interpolation formula with a small number of parameters. This avoids solving the full polySTRAND model equations at each point in space and time, thus creating a much more efficient simulation. This method was successfully implemented for the bimodal blend as shown in section 2.6.2 and we are now generalising this to include multiple chain species.

We illustrate this methodology for our specific simulations. In the previous section we presented a rheological parameterisation for the HDB6 material, comprising four species of polymer chain calculated using the RDP model, plus one further species acting as unoriented solvent and for which, therefore, $\Delta f_0 = 0$. The volume fractions for these five species are given in the third column of Table 3.2. We aim to use the polySTRAND model to compute the effect of chain orientation, deduced from the RDP model, on flow induced crystal nucleation rate.

Since at small values the enhancement of the nucleation rate is exponential in Δf_i, our proposed interpolation formula relates the natural logarithm of the normalised nucleation rate, $\log\left(\dot{N}_\Lambda/\dot{N}_q\right)$ calculated from the polySTRAND model to the free energy change per Kuhn segment, Δf_i, for each species i. The formula is a quotient function that comprises two polynomials in Δf_i for the numerator and the denominator. We found that quadratic order polynomials were sufficient to obtain a good fit to the polySTRAND results. The general structure of this interpolation formula is

$$\log\left(\frac{\dot{N}_\Lambda}{\dot{N}_q}\right) = \frac{\sum_i a_i \Delta f_i + \sum_i \sum_{j\leq i} b_{ij} \Delta f_i \Delta f_j}{1 + \sum_i c_i \Delta f_i + \sum_i \sum_{j\leq i} d_{ij} \Delta f_i \Delta f_j}. \tag{3.14}$$

In keeping with the observed behaviour of the polySTRAND model, this formula is linear in the Δf_i for small orientations, but approaches an asymptotic value of the nucleation rate at high orientations when the nucleation energy barrier does not reduce any further due to the finite extensibility of the polymer chains.

The coefficients a_i, b_{ij}, c_i and d_{ij} are fitting parameters that need to be adjusted for each specific material (i.e. they depend on the chemistry-dependent polySTRAND parameters, as well as on the specific volume fractions of the component species).

Now that we have the volume fractions of our chain species matched to rheological data, it is now possible to calculate the nucleation rate for various combinations of Δf_i. There are certain combinations of Δf_i that would not occur during our simulations, so we have reduced the range of the shorter fast relaxing species compared to the longer chain species. This allows our fitting procedure to be more accurate.

The range for Δf_4 was 0-0.8. Δf_3 and Δf_2 were 0-0.3. Δf_1 was 0-0.2. For each chain species we used 6 data points, which gives 1296 overall data points with the different combinations of Δf_i taken into account. This data is given the label D.

We can now construct a cost function, C, for a least squares regression to be applied to give values for a_i, b_{ij}, c_i and d_{ij} as,

$$C = \sum_m \left(\frac{\sum_i a_i \Delta f_{i,m} + \sum_i \sum_{j \leq i} b_{ij} \Delta f_{i,m} \Delta f_{j,m}}{1 + \sum_i c_i \Delta f_{i,m} + \sum_i \sum_{j \leq i} d_{ij} \Delta f_{i,m} \Delta f_{j,m}} - D_m \right)^2, \tag{3.15}$$

where the subscript m refers to the data points used and the subscripts i and j refer to the chain species. The function, C, is then minimised to give a_i, b_{ij}, c_i and d_{ij}.

Sample results of fitting the nucleation rates obtained from the polySTRAND model to the above formula are shown in Figure 3.4. These are shown for some specific values of Δf_1, Δf_2 and Δf_3, and for the full range of Δf_4, though (as indicated above) the fitting was done for a wider set of values of the Δf_{1-3}.

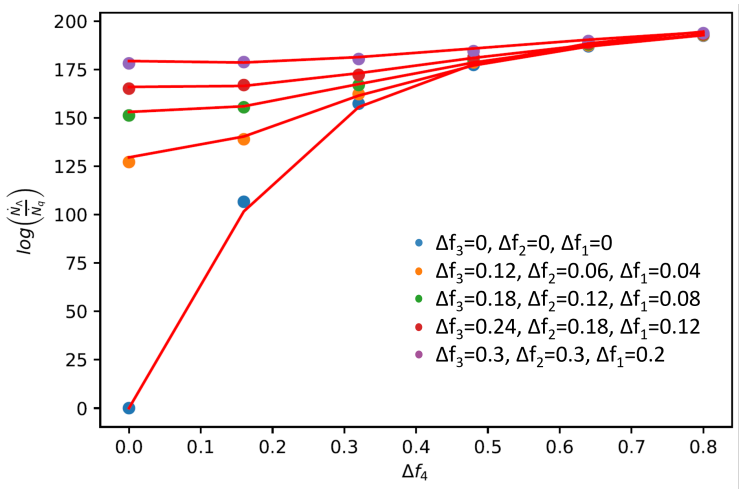


Figure 3.4: Figure showing the results of the polySTRAND model (points) and the interpolation function (red lines) to the polySTRAND results.

3.6.2 Logarithmic Version of the Schneider Rate Equations

The polySTRAND model computes the rate of formation of nuclei from which crystals grow. We assume that the polymer crystallises in a spherulitic structure so that the growth of isolated crystallites is described by Schneider rate equations [50]. These were first outlined in section 2.6.1 and are repeated here:

$$\begin{aligned} \frac{DN}{Dt} &= \dot{N}, \\ \frac{DR}{Dt} &= 2G_s N, \\ \frac{DS}{Dt} &= 4\pi G_s R, \\ \frac{DV}{Dt} &= G_s S, \end{aligned} \tag{3.16}$$

where $\frac{D}{Dt}$ represents the material derivative. All quantities are per unit volume, V represents the volume of the crystalline phase; S is the total surface area of the crystallites; R is the total radius; N is the number of nuclei. The quantities $\dot{N}$ and G_s represent the nucleation rate per unit volume and the crystal growth rate respectively.

These equations 3.16 do not account for the impingement between neighbouring crystals, which becomes significant as V approaches unity. Using the method of Avrami [51], we define the volume fraction of the crystal phase corrected for impingement as ξ_g, where

$$\xi_g = 1 - \exp(-V). \tag{3.17}$$

As the rate of nucleation depends exponentially on the values of Δf_i this can produce large spatial gradients in the local nucleation rate $\dot{N}$ within the simulation. For example, at the wall of the contraction in contraction-expansion flow. This makes the numerical solution on the Schneider rate equations 3.16 challenging due to the presence of the advection terms in the material derivative. This problem can be mitigated by instead computing with quantities related to logarithmic functions of variables associated with crystal growth. In order to form these functions we need to define an appropriate scale for each of the quantities under consideration.

Although in principle using the quiescent nucleation rate for the scales for each quantity in the Schneider rate equations is the logical choice, the quiescent nucleation rate, $\rho_K \dot{N}_q$, in our example is of order $10^{-59}\text{cm}^{-3}\text{s}^{-1}$. This vanishingly small rate is far too small to provide a sensible scaling as it, for all intents and purposes, is equal to zero. In practice this means that the energy barrier to nucleation is such that any nucleation events under quiescent conditions will be from heterogeneous rather than homogeneous nucleation. We can therefore add an additional rate of nucleation $\dot{N}_{het}$ to represent the heterogeneous nucleation rate per unit volume. Provided that the value of $\dot{N}_{het}$ is sufficiently low that it does not lead to the growth of a significant number of crystallites on the timescale of the simulations its value does not affect the flow induced acceleration. In the following calculations we include a heterogeneous nucleation rate of $\dot{N}_{het} = 1 \times 10^{-2}\text{cm}^{-3}\text{s}^{-1}$, which is sufficiently small to not produce significant crystallisation, but is much larger than the quiescent homogeneous nucleation rate. With the inclusion of this heterogeneous nucleation rate the first equation in equation 3.16 becomes

$$\frac{DN}{Dt} = \dot{N} + \dot{N}_{het}. \tag{3.18}$$

The inclusion of $\dot{N}_{het}$ therefore provides a floor for the nucleation rate in regions where $\dot{N}$ is effectively zero. In such regions the expected number density of crystal nuclei formed from quiescent nucleation is given by,

$$N_c = \dot{N}_{het} t_{char}, \tag{3.19}$$

where t_{char} is a characteristic simulation time.

Using N_c we can define a new dynamical variable

$$u_N = \log\left(1 + \frac{N}{N_c}\right), \tag{3.20}$$

such that

$$N = N_c(e^{u_N} - 1).$$

Substituting this into the first of the Schneider rate equations we obtain an evolution equation for u_N in the form,

$$\frac{Du_N}{Dt} = \frac{\dot{N} + \dot{N}_{het}}{N_c e^{u_N}}. \tag{3.21}$$

The subsequent Schneider rate equations can be treated in a similar way.

For R we define a typical scale as $R_c = \dot{N}_{het} G_s t_{char}^2$ and

$$u_R = \log\left(1 + \frac{R}{R_c}\right), \tag{3.22}$$

where

$$\frac{Du_R}{Dt} = \frac{2G_s N}{R_c e^{u_R}}.$$

Similarly for S we define $S_c = \dot{N}_{het} G_s^2 t_{char}^3$ such that

$$u_S = \log\left(1 + \frac{S}{S_c}\right), \tag{3.23}$$

satisfying

$$\frac{Du_S}{Dt} = \frac{4\pi G_s R}{S_c e^{u_S}}. \tag{3.24}$$

Finally for V we define $V_c = \dot{N}_{het} G_s^3 t_{char}^4$ with

$$u_V = \log\left(1 + \frac{V}{V_c}\right), \tag{3.25}$$

so that

$$\frac{Du_V}{Dt} = \frac{G_s S_{tot}}{V_c e^{u_V}}. \tag{3.26}$$

With these logarithmic equations in place, we use u_N, u_R, u_S and u_V as dynamical variables in the simulation and then compute N, R, S and V from them.

3.6.3 Crystallisation Model Parameters

The polySTRAND flow induced crystallisation model requires a number of parameters, which are known to varying degrees of certainty [2]. We consider the parameters in the model relevant to the case of polyethylene at 125 °C, as used in the experiments of Scelsi et al [5].

The density of Kuhn segments is given by

$$\rho_K = \frac{\rho_a N_A}{M_K}, \tag{3.27}$$

where ρ_a is the amorphous density of the polymer, N_A is the Avogadro number and M_K is the Kuhn molecular weight found from $M_K = M_{mono} C_\infty$, where M_{mono} is the monomer molecular weight and C_∞ is the characteristic ratio.

The number of steps per entanglement segment is given by

$$N_e = \frac{M_e}{M_K}, \tag{3.28}$$

where M_e is the entanglement molecular weight.

We also need to obtain parameters governing the quiescent nucleation kinetics so that we can predict FIC using the polySTRAND model. The monomer attachment time, τ_0, is given by projecting the entanglement time τ_e down to the Kuhn step length scale:

$$\tau_0 \approx \tau_K = \frac{\tau_e}{N_e^2}. \tag{3.29}$$

The form of the quiescent nucleation barrier is found from the quiescent free energy of crystallization per monomer, ϵ_B, and the surface energy cost, μ_S, which are chosen to match the literature data for the quiescent nucleation rate, $\dot{N}_q$, and critical nucleus size, n^*. The latter can be estimated via

$$n^* = \frac{M_{n^*}}{M_K}, \tag{3.30}$$

where the mass of the critical nucleus (assumed to be spherical) is $M_{n^*} = 4\pi\rho_c N_\Lambda r_l^3/3$, for lamella thickness $2r_l$ and crystal density ρ_c. Values for n^* and $\dot{N}_q$ were available in the literature, and these can then be matched in the model by adjusting the parameters ϵ_B and μ_S given the known value for τ_0.

Estimates for the spherulitic growth rate, G_s can be obtained from the literature. Likewise, values for the prefactor Γ that relates Δf to $P_{2,k}$, are suggested in the literature [2, 67]. Note there is a need to distinguish between Γ_0 for the original model and Γ_{FE} when finite extensibility is included as discussed in section 3.5.2 above. There is some debate over the choice of Γ and further details are given in section 3.7.5.

Table 3.4 gives a summary of the parameters values used and the methods and references used to determine them.

Parameter	Value	Method
ρ_K	3.01×10^{21} cm^{-3}	Calculated using literature data [75, 76], findings from Matsuyama et al [77] and equation (3.27)
ρ_a	0.965 g cm^{-3}	Obtained from Matsuyama et al. [77]
M_e	1600 g mol^{-1}	Literature data [78]
M_{mono}	28 g mol^{-1}	Literature data [76]
C_∞	6.9	Literature data [75]
M_K	193.2 g mol^{-1}	Calculated using literature data [76, 75]
N_e	8.2	Calculated using literature data and equation 3.28 [78, 76, 75]
τ_e	9 ns	Finding from Szanto et al. [79]
τ_0	0.134 ns	Calculated using literature data [78, 76, 75], findings from Szanto et al [79] and equation 3.29
ρ_c	1 g cm^{-3}	Finding from Martin et al [80]
r_l	1.49×10^{-6} cm	Finding from Zhou et al [81]
M_{n^*}	5.35×10^6 g mol^{-1}	Calculated using the findings from Martin et al [80] and Zhou et al [81]
n^*	43400	Obtained by fitting ϵ_B and μ_S in the polySTRAND model to the form of the nucleation barrier reported by Yi et al. [9] and toliterature data [76, 75, 80, 81]
G_s	0.3 μm s^{-1}	Obtained from Hoffman et al [10]
$\dot{N}_q$	4.76×10^{-80} s^{-1}	Obtained from fitting ϵ_B and μ_S in the polySTRAND model to the nucleation barrier found by Yi et al. [9]
$\dot{N}_{het}$	1×10^{-2} cm^{-3}s^{-1}	Chosen to provide a floor rate of crystal nucleation in the melt, see section 3.6.2.
Γ_0, Γ_{FE}	2, 10/3	Γ_0 value suggested by Anwar & Graham [67] from molecular dynamics simulations of a bimodal polyethylene blend. Here Γ_{FE} is the value adjusted to take account of the numerical factor of 10/3 arising from the change in the calculation of $P_{2,K}$ with finite extensibility. Further clarification is given in section 3.7.5.

Table 3.4: Table showing the model parameters for the simulations.

We now examine how the effect of flow changes the energy barrier to nucleation. The nucleation rate is related to the nucleation barrier via an exponential function. Therefore, flow can have a dramatic effect on the nucleation rate. In a polydisperse melt, each chain species needs to be taken into account to get an accurate prediction of the nucleation barrier.

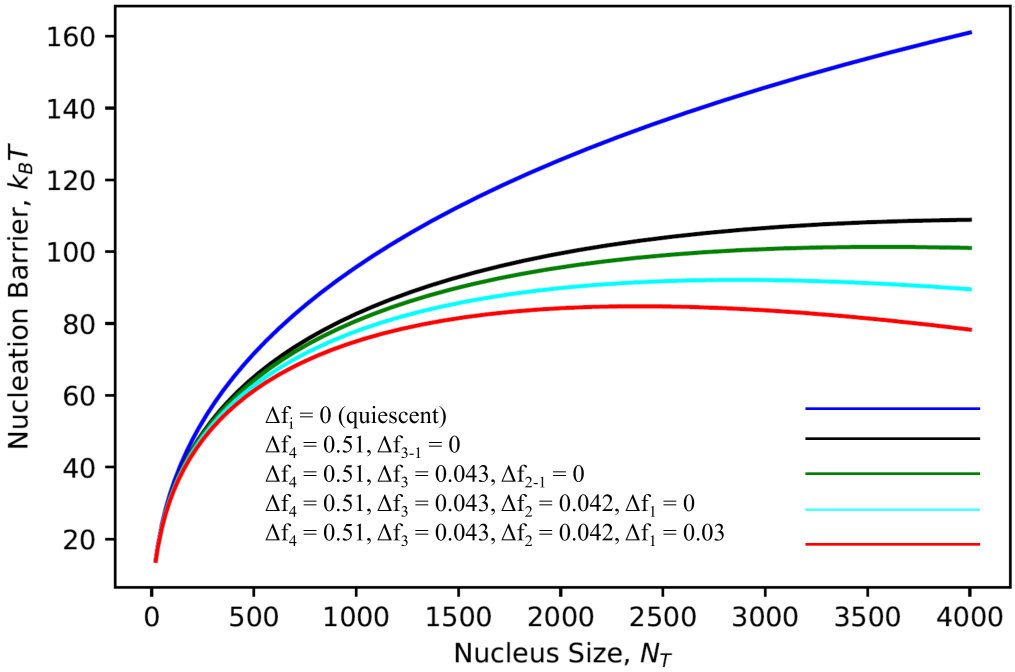


Figure 3.5: Figure showing how each of the different chain species in our polymer melt reduces the nucleation barrier. Each chain species contributes in a non-trivial way, highlighting the importance of taking all the species into account in our calculations of the nucleation barrier.

Figure 3.5 shows how each of the different chain species in our polymer melt contributes to the reduction in the energy barrier. The values chosen for Δf_i in figure 3.5 are the maximum values for Δf_i obtained from a simulation in section 3.7.4 figure 3.24. This illustrates how flow dramatically reduces the nucleation barrier within the melt in our simulations.

Figure 3.5 shows that the barrier to nucleation decreases by multiple order of k_BT due to the flow. The critical nucleus size also decreases by a factor of 20 to roughly 1500 monomers. A large proportion of this reduction is due to the longest chain species and the rest is a result of the remaining three smaller chain species. Although the contribution of the smaller, fast relaxing, chain species may appear insignificant when compared to the long chain species. However, these small chain species make up a larger volume fraction of the melt, so do contribute to the reduction of the energy barrier in a non-trivial way.

3.7 RDP + FIC in openFOAM

In this section we detail the results from combining the multimode RDP model and our polySTRAND model calculation for FIC in our full openFOAM simulation. We outline how we implemented the multimode RDP model in openFOAM and then conduct a mesh convergence study to ensure the accuracy of our findings. Parallel computing considerations are also given. We show the flow dynamics first, then the crystallisation dynamics for four different flow rates. Our results are compared qualitatively with those of Scelsi et al [5].

For our investigation, we will examine four different cases of area flow rate, since the simulation is two-dimensional. The fluid flow is developed by imposing a non-zero pressure difference between the inlet and the outlet. Our simulations ran until $t = 10s$, with time step $\Delta t = 0.001s$. Table 3.5 shows the pressure drops and resulting flow rate in each case.

Q $mm^2\ s^{-1}$	ΔP MPa
1.26	9
1.68	10.5
2.19	12
2.62	13.5

Table 3.5: Table showing the different flow rates used in the simulations along with the parameters used in the pressure ramp formula to achieve them.

3.7.1 Mesh Convergence

As mentioned in section 3.2 it was necessary to generate the relevant OpenFOAM case files for simulation of polydisperse melts. In order to check that the automatically generated code was correctly generating the constitutive model we compared simulations of a two-mode polymer blend using the generated code with the bimodal blend code to ensure they produce the same results. In figure 3.6 we compare the results for the change in the free energy per Kuhn segment Δf between the automatically generated and explicitly coded models. For the coarse mesh there is some discrepancy due to a computational instability at the wall of the contraction, but this can be rectified by using a more refined mesh for which the results matched perfectly.

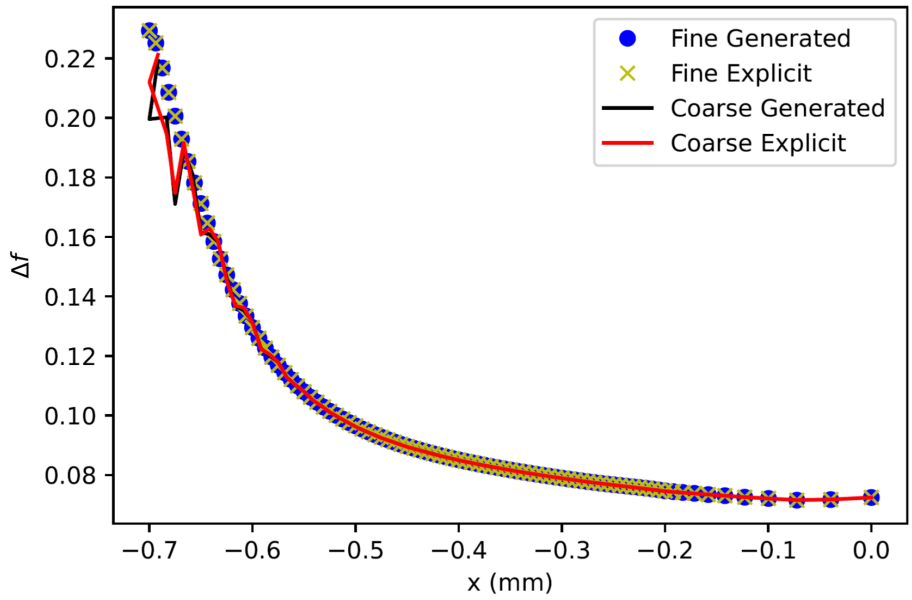


Figure 3.6: Figure showing how the Δf parameter converges due to different mesh resolutions and how the results for the non-generated code and the generated code match up.

Figure 3.7 shows the two different meshes. The comparison between the statistics for the different meshes is shown in table 3.6.

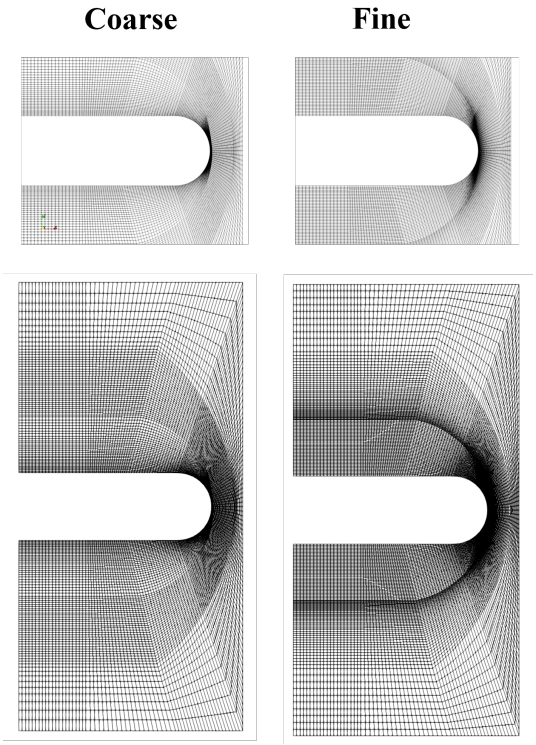


Figure 3.7: This figure shows the difference in mesh resolution required to ensure that the results from the generated RDP model and the explicitly coded model matched.

	Coarse Mesh	**Fine Mesh**
Points	28542	36582
Cells	14000	18000
Faces	56270	72290
Internal Faces	27730	35710

Table 3.6: Table showing the different mesh statistics for the two meshes.

The flow induced crystallisation calculation is added to the generated code file via the interpolation formula for the prediction of the nucleation rate outlined in section 3.6.1, the Schneider rate equations for the predictions of the crystallisation dynamics and the Avrami equation to correct the volume fraction for impingement. With all this in place, a constitutive model for flow induced crystallisation of an isothermal polydisperse polymer can be used in openFOAM.

Now that we have refined the mesh enough to ensure that the generated code and the explicit code agrees, we need to see how adding further refinement to the mesh would change the results that we obtain. Here we discuss refinement of the mesh both in terms of number of cells and in terms of optimising the cell shape. This will give us confidence that we are producing consistent results with our simulations.

In the early version of the mesh for the contraction geometry, detailed in the previous section, the cells near the corner were significantly skewed. This led to very high aspect ratio cells at the contraction region. If there are high aspect ratio cells in the mesh, this can destabilise the simulation and reduce accuracy. New blocks were added to the corners of the mesh to minimise skewness of the cells at the corners and hence reduce the aspect ratio of the cells in the contraction region.

Another modification to the mesh was to introduce a gradient in cell width within the blocks that run through the centre of the mesh. This would allow continuity between the blocks closest to the wall and the centre of the contraction region and improve numerical stability. The middle half of the blocks keep a constant expansion ratio of 1, whilst the remaining outer quarters have an expansion ratio of 1:9. This was necessary because the blocks at the wall need to be well resolved as accurately as possible because the contraction means that the dominant crystallisation dynamics occur at a point on the wall of the contraction. The forming crystals are advected down streamlines, and a coarse mesh at the wall results in prediction of advected crystallisation over a large volume.

Once these alterations were implemented we used three different versions of the mesh with different resolutions to check for computational inaccuracies due to mesh resolution. Figure 3.8 shows the overall block structure of the mesh along with a view of the contraction region of the different mesh resolutions. The summary statistics for each mesh are given in table 3.7.

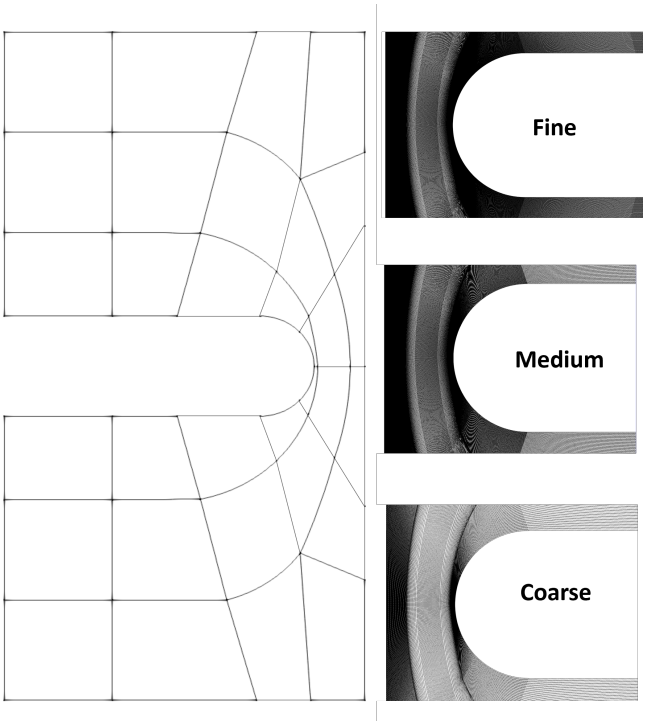


Figure 3.8: This figure shows the mesh block structure and the zoomed in view of the contraction region for the different mesh resolutions.

	Coarse Mesh	**Medium Mesh**	**Fine Mesh**
Points	346562	1381122	1941402
Cells	172000	688000	968000
Faces	689280	2754560	3874700
Internal Faces	342720	1373440	1933300

Table 3.7: Table showing the different mesh statistics for the coarse, medium and fine meshes.

We can then see how the different mesh resolutions affect the results. For all mesh refinement comparisons we will look at the flow rate case of Q = 1.68 mm^2s^{-1} at the end time of our simulation of 10 s. Figure 3.9 shows how the different meshes affect the stretch and the change in monomer free energy from chain deformation of the longest chain species, λ_4 and Δf_4.

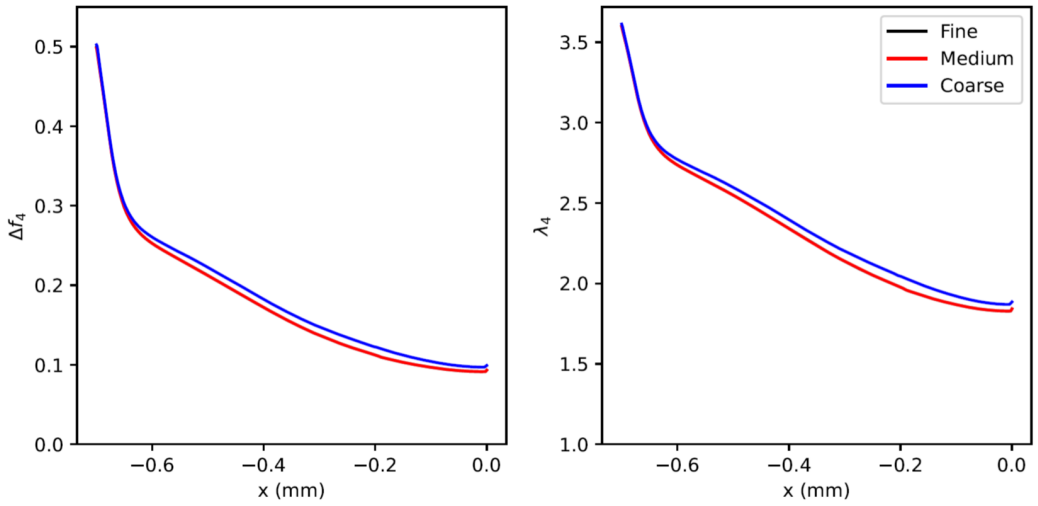


Figure 3.9: These line plots show how the mesh resolution affects the results for Δf_4 and λ_4 when Q = 1.68 mm^{2}s^{-1} and t = 10 s. The line plot shows the change in these variables from the wall of the contraction region to the centre of the channel in the middle of the mesh. The x-axis represents the distance from the centre of the mesh, where the centre is set at 0. As can be seen there is some discrepancy between the meshes and we do see convergence for our results.

The line plots in figure 3.9 show that there is convergence of our results due to the mesh resolution. Figures 3.10 and 3.11 show the results for Δf_4 and λ_4 in the whole mesh. As expected, the main activity for both these parameters was in the contraction region and there was not a visible difference between the different mesh resolutions in this view.

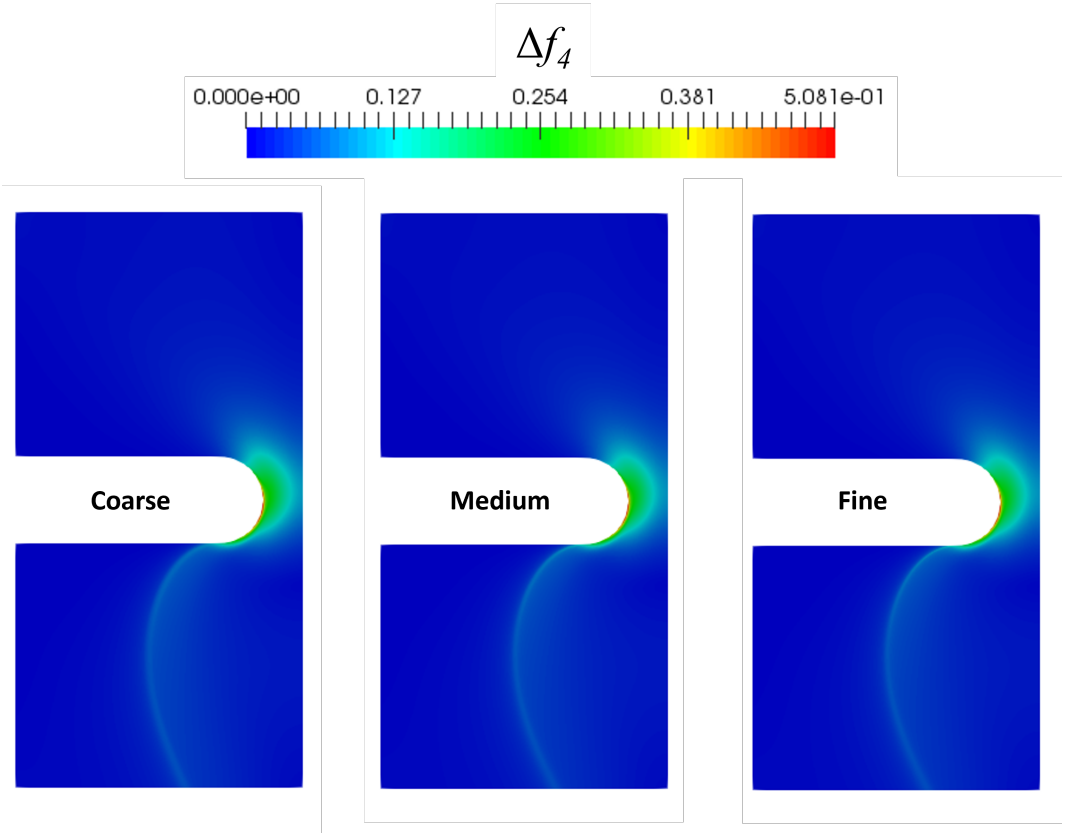


Figure 3.10: This figure shows the results for Δf_4 for the different meshes when $\mathrm{Q} = 1.68\ \mathrm{mm^2 s^{-1}}$ and $t = 10\ \mathrm{s}$, with all the activity being in the contraction region.

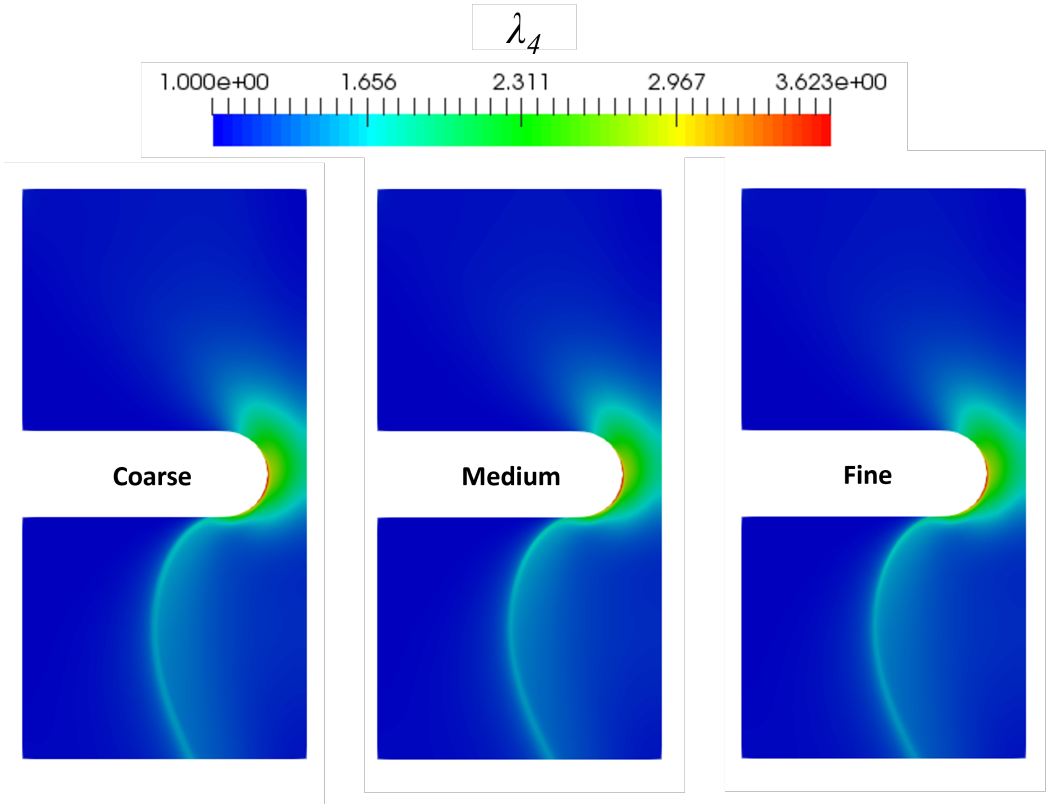


Figure 3.11: This figure shows the results for λ_4 for the different meshes when $Q = 1.68$ mm^{2}s^{-1} and $t = 10$ s, similarly to the case with Δf_4, all the activity is in the contraction region.

We then look at the number density of the crystals in the melt and how this develops due to the contraction. This is where we see the benefits of finer mesh resolution. From the coarse to the fine mesh resolutions you can see a clear improvement in the definition of the region of crystal development in the runs. The level of numerical diffusion in the 'fang' shape decreases significantly from the coarse to the fine mesh. We believe this is a result of a combination of better resolution of the region of significant crystal nucleation, near the wall of the contraction region, together with more accurate advection of the growing crystals with reduced numerical diffusion.

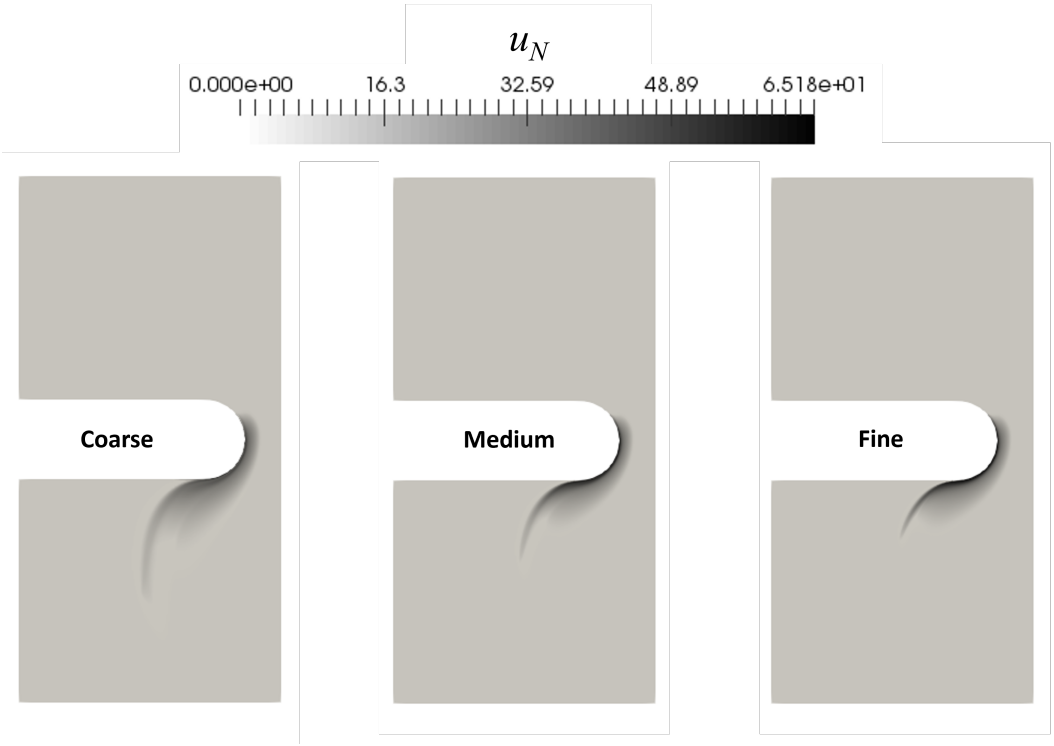


Figure 3.12: These plots show how u_N develops in the simulations for the different mesh resolutions when Q = 1.68 mm^2s^{-1} and t = 10 s. There is some numerical diffusion when the solution is advected downstream from the contraction in the coarse run. This is improved as the mesh resolution increases.

Now that we can see the number of nuclei in the melt in the simulation, we can then look at the crystal volume fraction that is calculated from the Schneider rate equations. As previously mentioned this volume fraction is the total volume of nuclei if more than one nucleus can occupy a given region of space. Although this cannot be measured experimentally, it does give an insight into what we can expect from the experimentally observed volume fraction of nuclei. As in the case of the number of nuclei, we will look at the logarithmic version of the Schneider rate equation that calculates the volume fraction.

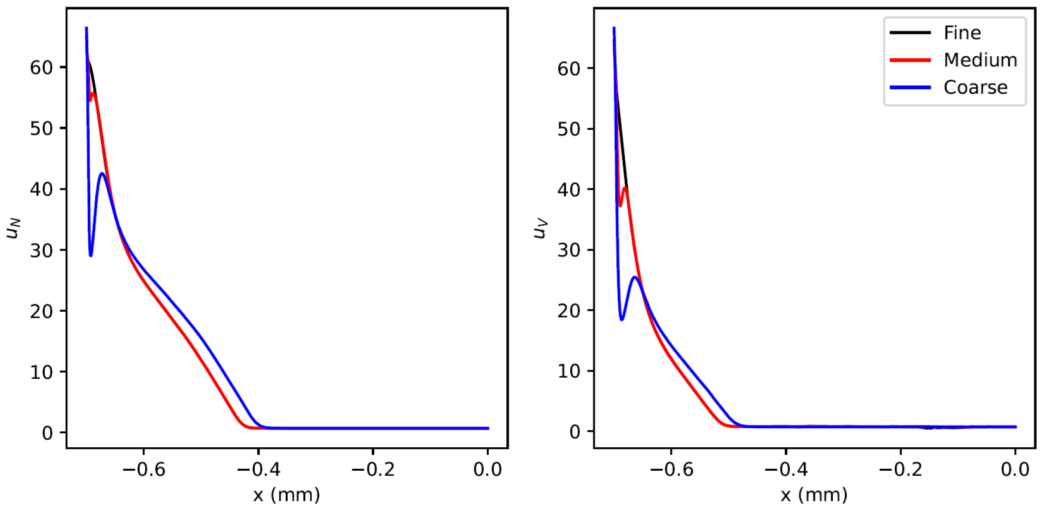


Figure 3.13: Line plots showing how the logarithmic measures of crystal number and volume fraction, u_N and u_V respectively, vary across the narrow part of the contraction as a function of distance from the symmetry-line, for Q = 1.68 mm^2s^{-1} and t = 10 s. Both quantities have a maximum at the wall and decrease towards the centre-line with convergence with mesh refinement seen for both functions. The x-axis represents the distance from the symmetry-line set at 0.

The line plots in figure 3.13 show how u_N and u_V develop across the centre of the contraction from the wall to the middle of the contraction for the different meshes. You can see there is convergence of our solutions for both quantities in the contraction region due to the mesh resolution increase.

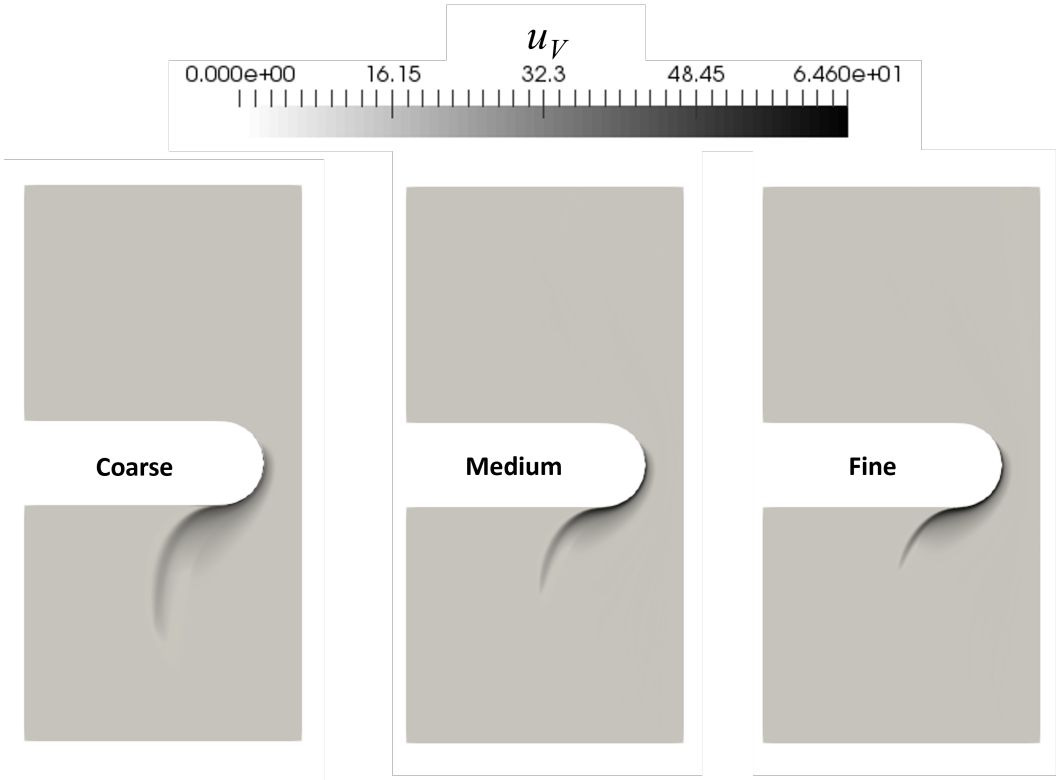


Figure 3.14: These plots show how u_V develops in the simulations for the different mesh resolutions when $\mathrm{Q} = 1.68\ \mathrm{mm^2 s^{-1}}$ and $t = 10$ s. Similarly to the case for u_N, there is some numerical diffusion when the solution is advected downstream from the contraction in the coarse run. This is improved as the mesh resolution increases.

Figure 3.14 shows how u_V develops in our simulation for the different meshes. Just as in the case for u_N we see that the numerical diffusion improves as we progress through the mesh resolutions. We also get the expected 'fang' shape for the uncorrected volume fraction after the contraction. This does match the experimental investigation by Scelsi et al [5]. Finally for the mesh comparison results, we will examine the volume fraction that is corrected for impingement, ξ_G, as calculated via the Avrami equation.

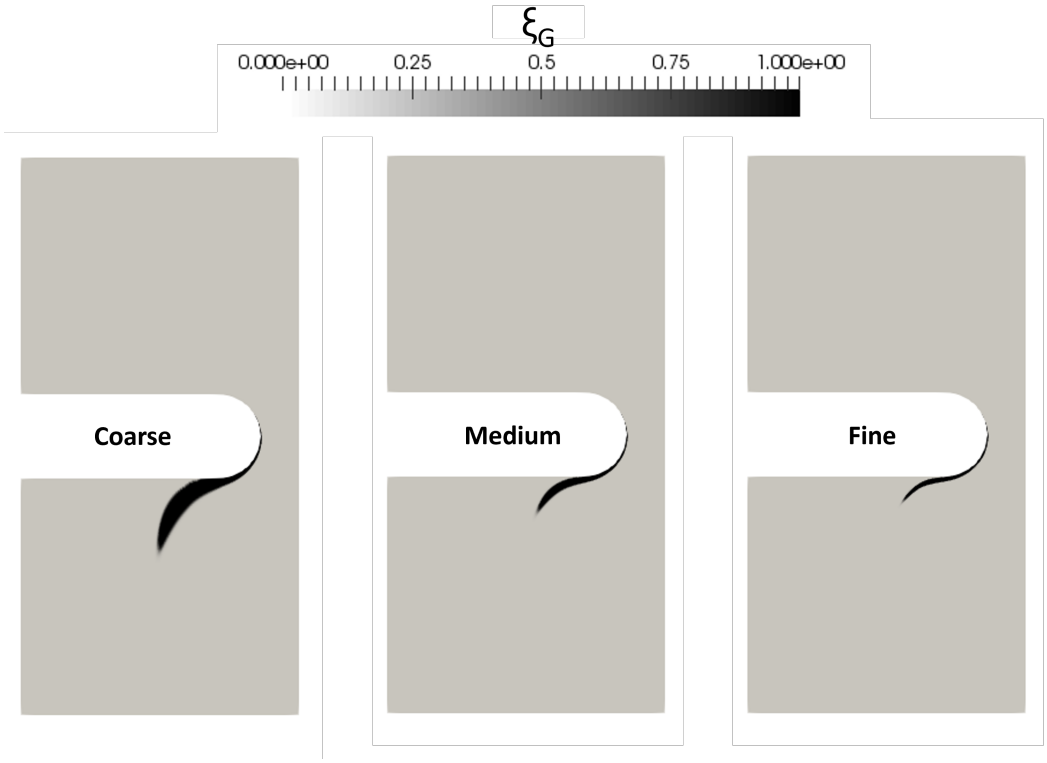


Figure 3.15: This figure shows how the volume fraction corrected for impingement, ξ_G, develops for the different mesh resolutions when Q = 1.68 mm^{2}s^{-1} and $t = 10$ s.

Figure 3.15 gives the results for ξ_G for the different mesh resolutions. Here we can see the largest difference between the runs for the meshes out of all the results previously given. Going from the coarse to the fine mesh, we can see that the definition of the 'fang' shape improves dramatically. You can also see that the 'fang' tapers off sooner in the fine mesh, which points to the improved accuracy of the mesh.

3.7.2 Parallel Computing for the Fine Simulation

With our simulations becoming more complex and requiring a fine mesh for computational accuracy, it is necessary to utilise parallel computing in the HPC facility. This offers the opportunity to investigate the effectiveness of the scalability of our simulations. We will run the simulation using the fine mesh with different numbers of processors for comparison. The flow rate for these runs is Q = 1.68 mm^2 s^{-1} with the pressure at the inlet being 10.5 MPa. The time step will remain the same as mentioned earlier at $\Delta t = 0.001$ s. A single processor could not finish the run in the maximum 48 hour time slot for the University of Leeds HPC facility. A single processor only managed to get to 0.55 s of the desired 10 s runtime. We then look at the runtime for 2, 4, 8, 16 and 32 processors to see how the speedup is affected by the addition of extra processors.

As our simulation is the same in each case and the computing power is increasing, this can be classed as a strong scaling investigation. The speed up is calculated using:

$$\text{Speedup} = \frac{t(1)}{t(N)},$$

where $t(1)$ is the computational time for running the program on one processor and $t(N)$ is the computational time for running the program on N processors. In a perfect case we would like the speedup to be linear and to equal the number of processors (N).

The left plot in figure 3.16 shows how adding cores reduces the run time for the simulation when running to 0.55 s simulation time. This simulation time was chosen because that was the full progress of a single core calculation in the 48 hour time limit offered at the HPC facility at the University of Leeds. The right plot in figure 3.16 shows how the speedup changes with increasing number of processors used. The relationship is approximately linear which shows a good progression of speedup, but it does not equal the number of processors in each case, which is to be expected as perfect speedup is not achievable in practice.

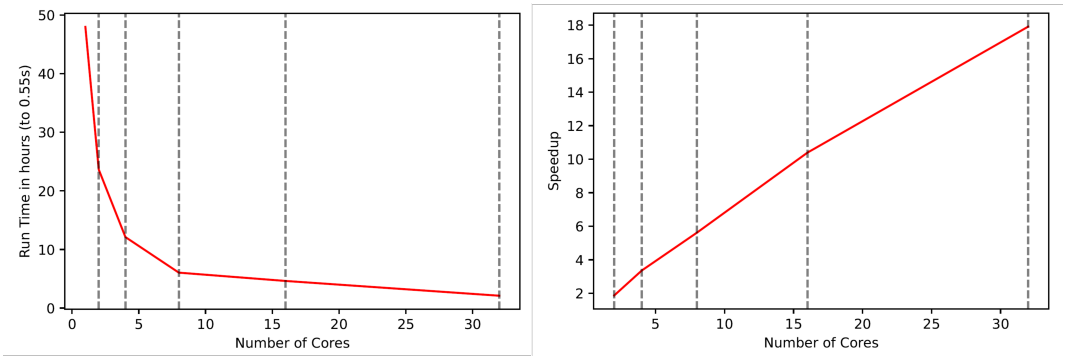


Figure 3.16: The left plot in this figure shows the decrease in run time for the simulation due to increased numbers of processors used. The right figure shows how the speedup progresses with the increased number of processors. The dotted lines in each plot represent 2, 4, 8, 16 and 32 cores.

Number of Cores	Runtime (hrs)	Speedup
1	48	
2	23.65	2.03
4	12.11	3.96
8	6.04	7.95
16	4.62	10.39
32	2.10	22.86

Table 3.8: Table showing the difference that adding more cores does to the overall run time and speedup of the simulation.

Table 3.8 gives a summary of the runtime and associated speedup for the number of

processors used in each case. We can see that we get super linear speedup for the case of when the number of cores is 2. This occurs when the speedup is larger than the number of processors. One plausible explanation for this is the cache effect due to the different memory hierarchies of a modern computer. As the number of processors change, the size of the total caches from the different processors also changes. Given the total cache size increase, more or all of the working set can fit into the caches; the working set denotes the amount of memory that a process requires in a given time interval. This results in the memory access time reducing dramatically, which causes the extra speedup on top of the actual computation.

3.7.3 Polymer Flow Configuration

Following the mesh convergence and parallel computing analysis, we then move on to investigating the effect of flow rate on the polymer deformation and the crystallisation within the melt. The four different area flow rates outlined at the beginning of section 3.7 will offer comparisons for different scenarios in the contraction geometry.

We shall start by looking at what the shear Weissenberg number is at the wall of the contraction for the four different flow rates investigated. The Weissenberg number is given by,

$$Wi_{\dot{\gamma}} = \tau\dot{\gamma},$$

where τ is the material relaxation time and $\dot{\gamma}$ is the shear rate. The Weissenberg number gives a measure of the importance of nonlinearity [17], with a larger Weissenberg number giving more non-linear behaviour in the material.

	Q mm^2 s^{-1}			
$Wi_{\dot{\gamma}}$	1.26	1.68	2.19	2.62
$\tau_{d,1}\dot{\gamma}$	2.20	2.77	3.28	3.76
$\tau_{d,2}\dot{\gamma}$	22.13	27.94	33.01	37.85
$\tau_{d,3}\dot{\gamma}$	218.72	276.23	326.30	374.10
$\tau_{d,4}\dot{\gamma}$	3572.57	4512.02	5329.88	6110.70

Table 3.9: Shear Weissenberg numbers for the reptation relaxation times of the chain species at the contraction region. The reptation relaxation times are $\tau_{d,1} = 0.18$ s, $\tau_{d,2} = 1.81$ s, $\tau_{d,3} = 17.87$ s, $\tau_{d,4} = 291.86$ s

Table 3.9 shows the orientation Weissenberg numbers for the different RDP modes at each of the flow rates investigated at the wall of the contraction, where the shear rate is the largest at $t = 10$ s. When the Weissenberg number is calculated in relation to the reptation relaxation times, $\tau_{d,i}$, we have that $Wi_{\dot{\gamma}} = \tau_{d,i}\dot{\gamma} > 1$ denotes shear thinning behaviour. Table 3.9 shows that there is shear thinning of all chain species for each of the

flow rates.

We then do a similar analysis for the case of the stretch relaxation times, $\tau_{s,i}$, of the RDP modes. Table 3.10 gives the stretch Weissenberg numbers for all the RDP modes at the wall of the contraction region at $t = 10$ s. In this scenario $Wi_{\dot{\gamma}} = \tau_{s,i}\dot{\gamma} > 1$ denotes the stretching regime of the polymer chains. For the slowest flow rate, there is arguably a small amount of stretching with the Weissenberg number being close to unity. Chain species 1-3 for the fastest three flow rates show a small amount of stretching, whilst the 4th chain species shows a large amount of stretching for all the flow rate cases.

	Q mm^2 s^{-1}			
$Wi_{\dot{\gamma}}$	1.26	1.68	2.19	2.62
$\tau_{s,1}\dot{\gamma}$	0.95	1.20	1.42	1.63
$\tau_{s,2}\dot{\gamma}$	0.95	1.20	1.42	1.63
$\tau_{s,3}\dot{\gamma}$	0.95	1.20	1.42	1.63
$\tau_{s,4}\dot{\gamma}$	47.63	60.16	71.07	81.48

Table 3.10: Shear Weissenberg numbers for the stretch relaxation times of the different chain species. The stretch relaxation times are $\tau_{s,1} = 0.078$ s, $\tau_{s,2} = 0.078$ s, $\tau_{s,3} = 0.078$ s, $\tau_{s,4} = 3.89$ s

Since we are simulating flow through a channel with no slip boundary conditions at the wall, the simulation is shear dominated. Nevertheless the presence of the contraction induces an extensional component to the flow that acts on the polymers. The local extension rate along the velocity direction at any given point is given by

$$\dot{\varepsilon} = \frac{(U \cdot \nabla U) \cdot U}{|U|^2}.$$

We examine how the extension rate changes through the contraction region for different flow rates. We calculated the extension rate at 5 different lines across the contraction to see how it develops. These lines are sketched in figure 3.17. The value for the extension rate in the middle of the contraction region on each of these lines is given in table 3.11 for each of the flow rates at $t = 10$ s.

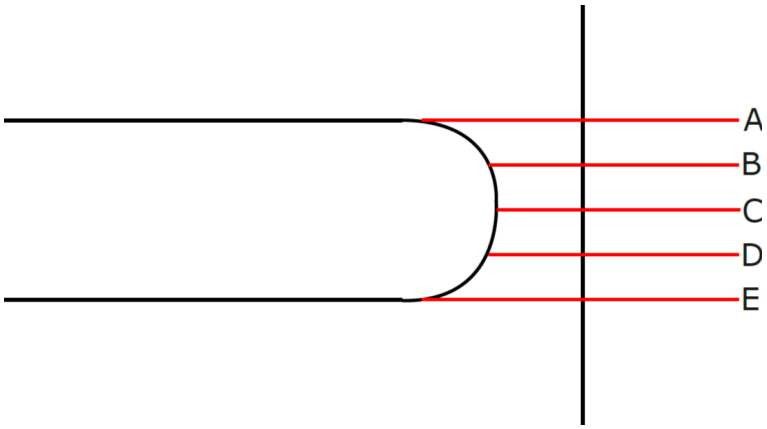


Figure 3.17: This diagram shows the different cross sections that were used to measure how the extension rate changes through the contraction.

	Q mm^2 s^{-1}			
$\dot{\varepsilon}$	1.26	1.68	2.19	2.62
A	2.21	4.02	6.70	10.47
B	2.01	3.93	7.08	11.98
C	-0.41	-0.56	-0.72	-0.98
D	-2.44	-4.52	-7.73	-12.49
E	-2.17	-3.99	-6.73	-10.64

Table 3.11: Extension rates for the different lines in the contraction region at the centre of the mesh. This shows how the extension rates change as the polymer flows through the contraction.

The calculated extension rate along each of these lines from the wall to the centre is then plotted and presented in figure 3.18. As expected, the largest absolute values of the extension rate are calculated at the entry and exit to the contraction, with line C deviating very little from zero for all the flow rates. With the extension rate growing in absolute value as you move away from the wall to the centre and shear forces being largest at the wall and smallest in the centre, we expect there to be an interplay between the shear and extensional forces affecting the crystallisation rate in the polymer.

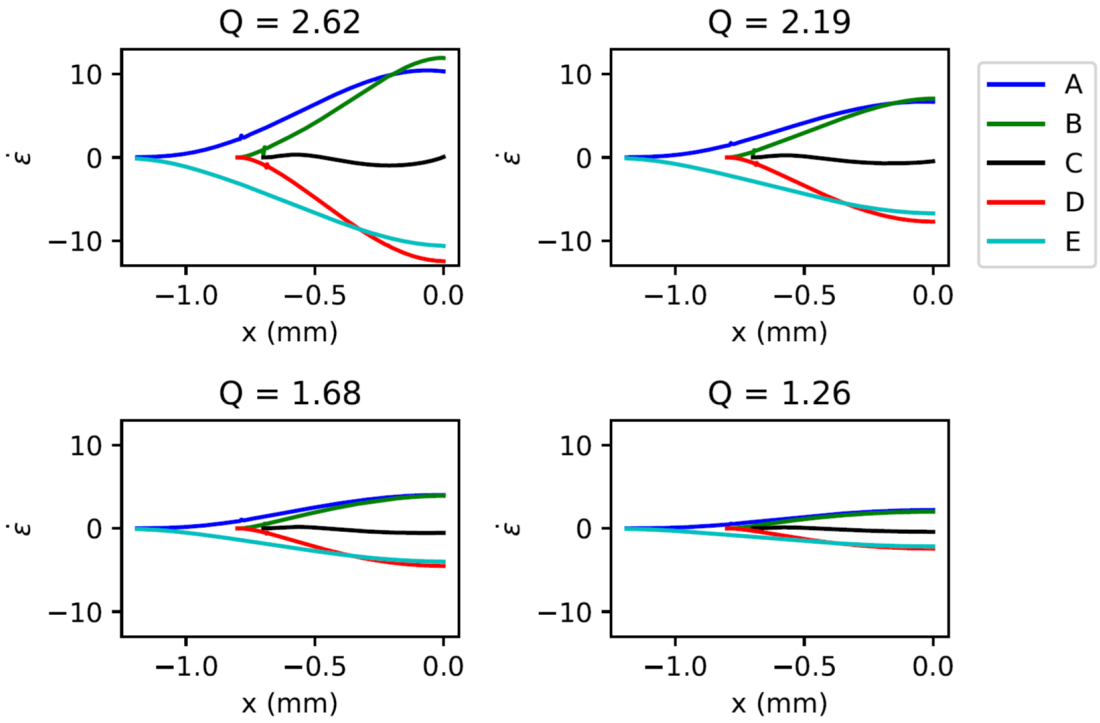


Figure 3.18: This figure shows how the extension rate changes across the contraction region from the wall to the centre for the different simulation speeds at $t = 10$ s. The line plots show the change in the extension rates from the wall of the contraction region to the centre of the channel in the middle of the mesh. The x-axis represents the distance from the centre of the mesh, where the centre is set at 0.

We then looked at how the extension rate develops along the centre of the flow channel (i.e. along the symmetry line of the mesh). These results are then presented for each of the different flow rates at $t = 10$ s in figure 3.19.

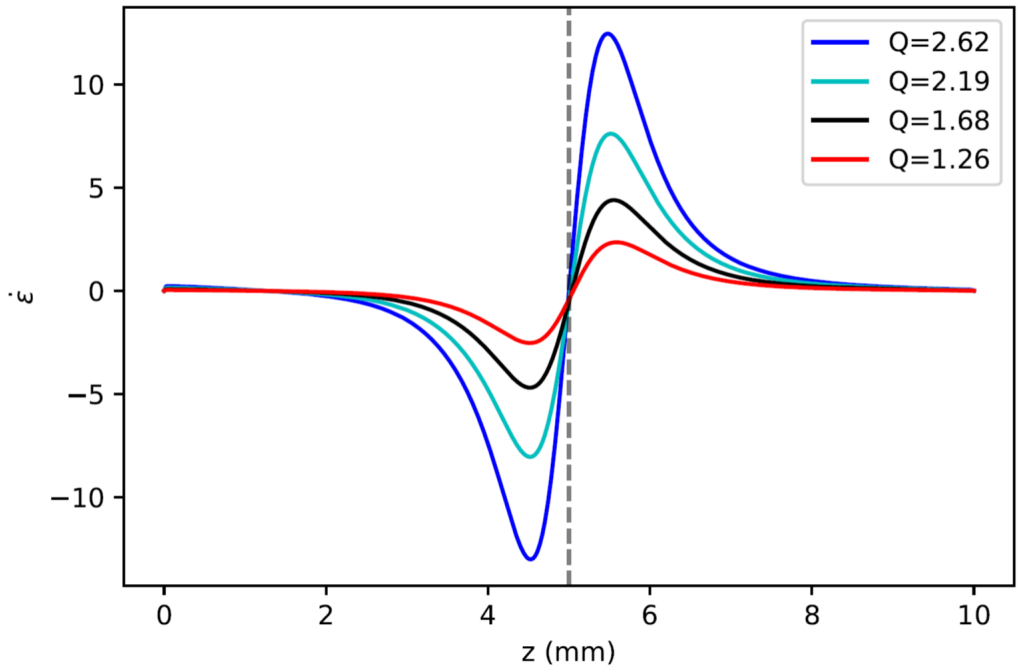


Figure 3.19: This plot shows how the extension rate evolves down the centre of the mesh along the z axis for the different flow rates investigated at $t = 10$ s. In this plot the largest positive value of the extension rate, $\dot{\varepsilon}$, is at the point where the melt enters the contraction, likewise the largest negative value of $\dot{\varepsilon}$ is at the point where the melt leaves the contraction.

As expected the largest absolute values for the extension rate occur at the entry and exit of the contraction and these are represented by the maximum and minimum in figure 3.19 respectively.

Having examined the extension rates through the flow, we can define associated extensional Weissenberg numbers calculated using the maximal extension rate found in the centre of the flow into the contraction. Hence our extensional Weissenberg numbers will be calculated via

$$Wi_{\dot{\varepsilon}} = \tau\dot{\varepsilon},$$

with τ being the associated relaxation time of the RDP mode in question. The Weissenberg number can be defined with respect to either the orientation or the stretch relaxation time. When $\tau_{d,i}\dot{\varepsilon} > 1$ and $\tau_{s,i}\dot{\varepsilon} < 1$ extension thinning is observed. When $\tau_{s,i}\dot{\varepsilon} > 1$ then extension thickening is expected. The extensional viscosity is then expected to grow either until the material exits the region of strong extensional flow, or until it reaches the limit of finite extensibility defined in our simulation.

	Q mm^2 s^{-1}			
$Wi_{\dot{\varepsilon}}$	1.26	1.68	2.19	2.62
$\tau_{d,1}\dot{\varepsilon}$	0.42	0.79	1.37	2.24
$\tau_{d,2}\dot{\varepsilon}$	4.25	7.95	13.76	22.53
$\tau_{d,3}\dot{\varepsilon}$	42.01	78.55	136.06	222.72
$\tau_{d,4}\dot{\varepsilon}$	686.26	1282.98	222.39	3638.01

Table 3.12: Extension Weissenberg numbers for the reptation relaxation times of the chain species at the point of highest extension rate just before the contraction region. The reptation relaxation times are $\tau_{d,1} = 0.18$ s, $\tau_{d,2} = 1.81$ s, $\tau_{d,3} = 17.87$ s, $\tau_{d,4} = 291.86$ s

	Q mm^2 s^{-1}			
$Wi_{\dot{\varepsilon}}$	1.26	1.68	2.19	2.62
$\tau_{s,1}\dot{\varepsilon}$	0.18	0.34	0.59	0.97
$\tau_{s,2}\dot{\varepsilon}$	0.18	0.34	0.59	0.97
$\tau_{s,3}\dot{\varepsilon}$	0.18	0.34	0.59	0.97
$\tau_{s,4}\dot{\varepsilon}$	9.15	17.11	29.63	48.51

Table 3.13: Extension Weissenberg numbers for the stretch relaxation times of the different chain species at the point of maxiumum extension rate just before the contraction region. The stretch relaxation times are $\tau_{s,1} = 0.078$ s, $\tau_{s,2} = 0.078$ s, $\tau_{s,3} = 0.078$ s, $\tau_{s,4} = 3.89$ s

Tables 3.12 and 3.13 show the extension Weissenberg numbers for the reptation relaxation times and the stretch relaxation times respectively for the different flow rates at $t = 10$ s at the centre of the mesh. Chain species 1 is neither strongly oriented or stretched by the extensional component of the flow for the slowest two flow rates. There is extension thinning shown for species 2 and 3 for the slowest two flow rates. Chain species 1-3 are in the extension thinning regime for the fastest two flow rates. Chain species 4 is in the extension thickening regime for all the flow rates.

Figure 3.20 shows the magnitude of the z-component of the velocity and the z-component of the velocity gradient across the contraction region at the end of the simulation for the four different area flow rates. The magnitude of the velocity gradient represents the shear rate in the contraction. As expected, the highest shear rates are at the wall of the contraction region and reduces to zero gradient at the centre of the geometry. This is indicative of the majority of the crystallisation nucleation happening at the wall of the contraction region. The crystal nuclei are then advected down the geometry as they grow.

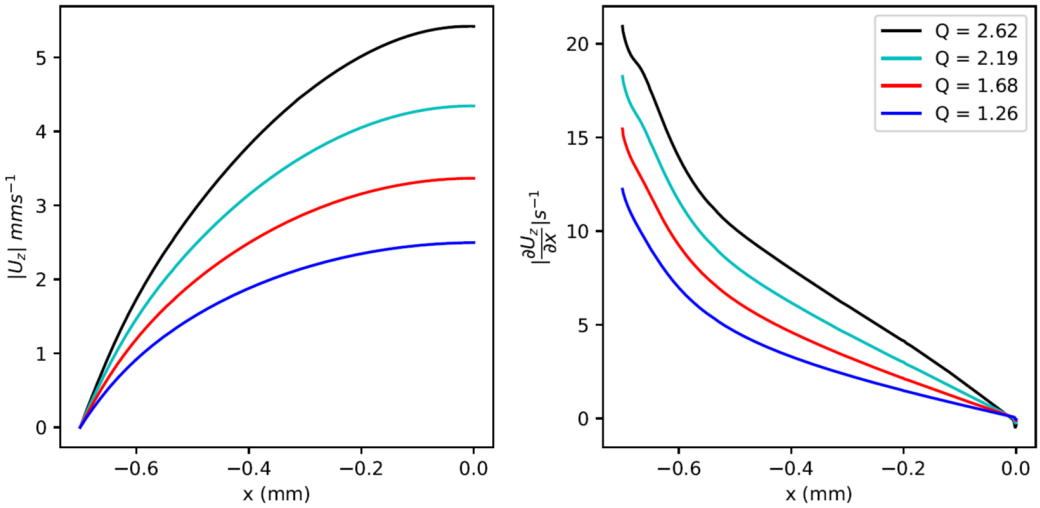


Figure 3.20: Line plots that show the magnitude of the z-component of the velocity and the z-component of the velocity gradient for the four different flow rates investigated at $t = 10$ s. The line plots show the change in these variables from the wall of the contraction region to the centre of the channel in the middle of the mesh. The x-axis represents the distance from the centre of the mesh, where the centre is set at 0.

We next examine the stretch, $\lambda_i = 1/3\sqrt{tr\underline{\underline{\mathbf{A}}}_i}$, of each species within the blend. The left plot in figure 3.21 shows the stretch , λ_i, of the different chain species at flow rate $\mathrm{Q} = 1.68\mathrm{mm}^2\ \mathrm{s}^{-1}$ at $t = 10\mathrm{s}$ across the contraction. From figure 3.20 we can observe that the wall shear-rate is around $15\ \mathrm{s}^{-1}$ which is sufficient to produce significant extension of the slowest chain species, since $\tau_{s,4} = 0.5$ s, but not of the three faster modes. The right-hand plot in figure 3.21 shows how the stretch of the slowest chain species, λ_4 varies with flow rate. In all cases we can see that the stretch is largest at the wall for all chain species and decreases towards the centre-line. However, λ_4 does not decrease to unity on the centre-line, demonstrating that there is stretch of this chain species due to the extensional deformation induced by the contraction. Nevertheless, the largest contributing factor to the stretch is the shear-rate at the wall, rather than the extension-rate along the centre-line.

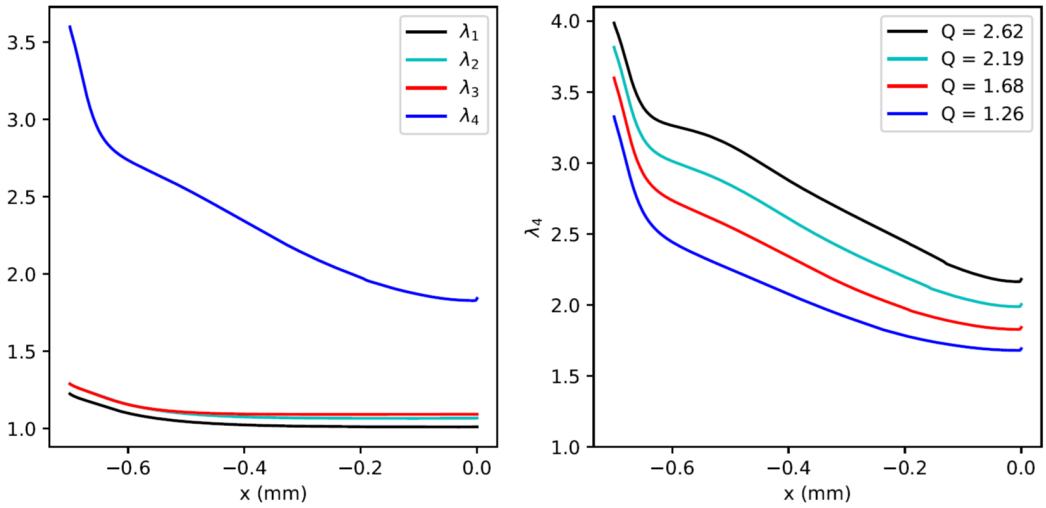
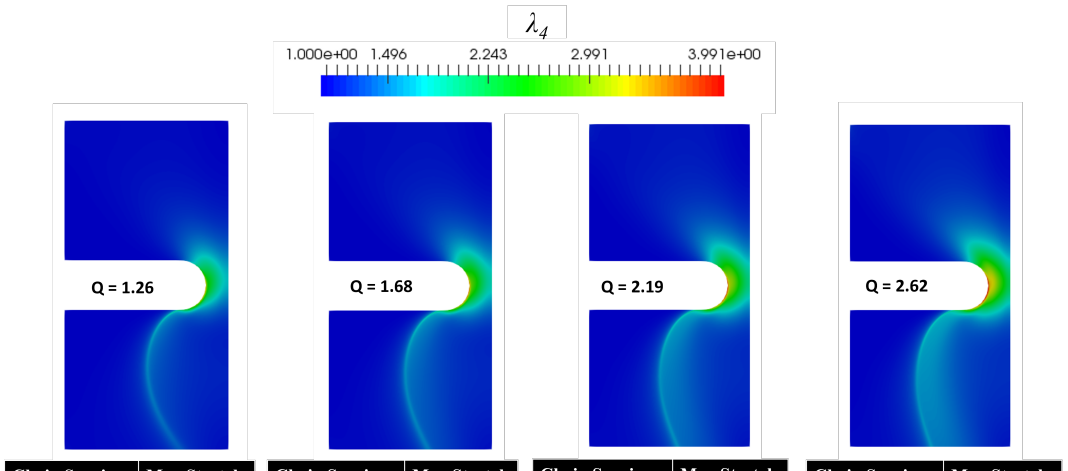


Figure 3.21: Line plots that show the stretch for the different chain species in the medium simulation and the results for λ_4 for the different flow rates at $t = 10$ s. The line plots show the change in these variables from the wall of the contraction region to the centre of the channel in the middle of the mesh. The x-axis represents the distance from the centre of the mesh, where the centre is set at 0.

Figure 3.22 provides images of λ_4 for the entire simulation domain after a time of 10 s for each of the flow rates. Although the largest values occur near the wall at the centre of the contraction, where the largest shear-rates occur, advection by the flow leads to the formation of a "fang" of stretched material downstream corresponding to the birefringence structures report by Lee et al [82]. This suggests that the majority of the crystal nucleation will occur at the wall of the contraction region with crystals advecting from this region. We also observe a broader region of stretch within the contraction due to the extensional flow. As noted above the strain-rates are not insufficient to induce large stretching of the other chain species, whose the maximal values are given in the tables underneath each picture.



Chain Species	Max Stretch
1	1.18
2	1.25
3	1.25
4	3.36

Chain Species	Max Stretch
1	1.23
2	1.29
3	1.29
4	3.62

Chain Species	Max Stretch
1	1.27
2	1.33
3	1.33
4	3.83

Chain Species	Max Stretch
1	1.31
2	1.37
3	1.37
4	3.99

Figure 3.22: This figure shows how the λ_4 parameter changes due to the flow rate of the different simulations at $t = 10$ s. As expected, most of the activity is in the contraction region. The tables underneath the contour plots show the maximum value for λ_i for the rest of the chain species in the different simulation runs.

3.7.4 Crystallisation Dynamics

Having determined the polymer conformation from solving the flow for the RDP model, we can now proceed to calculate how this changes the rate of nucleation. The left plot in figure 3.23 shows the free energy change per Kuhn segment, Δf_i, across the centre of the contraction for each of the chain species when Q $= 1.68$ mm^2 s^{-1} and $t = 10$ s. As might be expected these results mirror the corresponding values for λ_i of the different chain species shown in figure 3.21 with the largest values of Δf_i found at the wall and decreasing towards the centre-line. It is important to note that although Δf_4, corresponding to the longest chain species, is largest, $\Delta f_i \neq 0$ for all four chain species. As a consequence all chain species contribute to the reduction of the nucleation barrier as shown in figure 3.5. The right plot in figure 3.23 shows how Δf_4 develops across the contraction for the four different flow rates investigated.

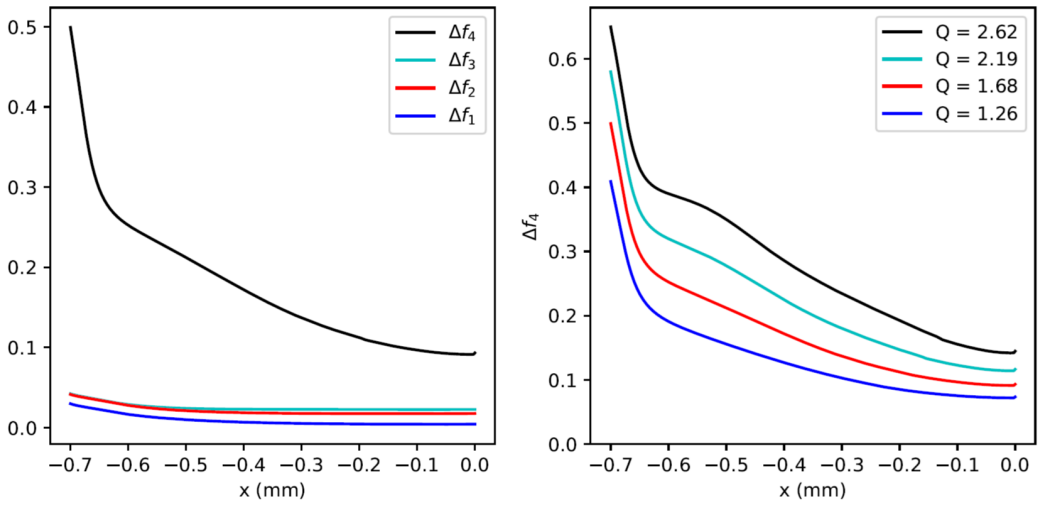
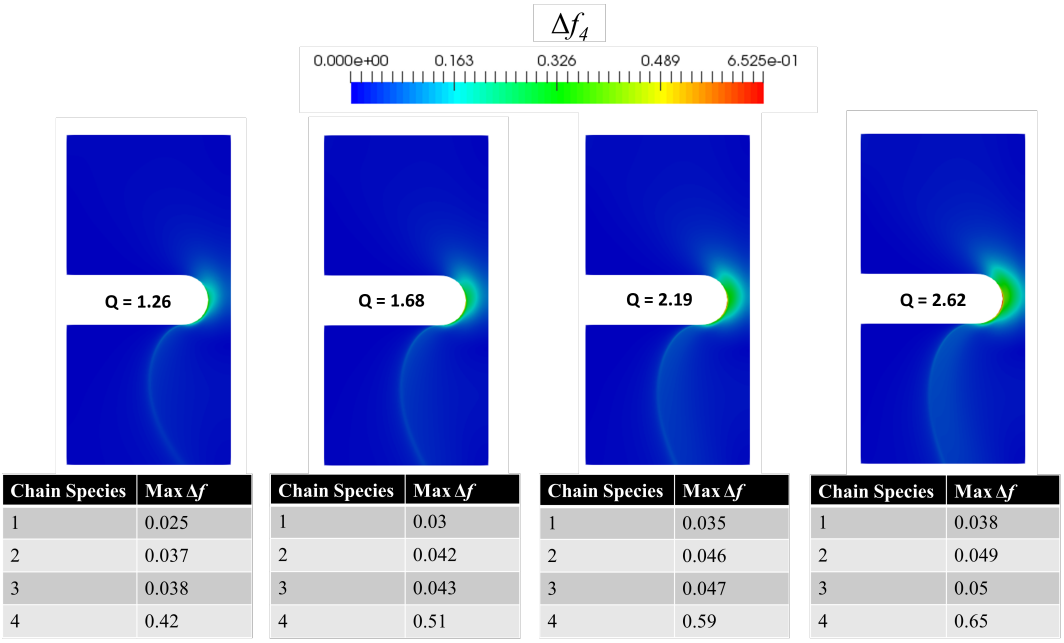


Figure 3.23: The left line plot shows how Δf_i develops across the contraction when Q = 1.68 mm^2 s^{-1} and $t = 10$ s. The right line plot shows how Δf_4 develops across the contraction for the four different flow rates investigated at $t = 10$ s. The line plots show the change in these variables from the wall of the contraction region to the centre of the channel in the middle of the mesh. The x-axis represents the distance from the centre of the mesh, where the centre is set at 0.

Figure 3.24 provides an equivalent contour plot to figure 3.22 for the values of Δf_4 at $t = 10$ s. The tables beneath each picture show the maximum value of Δf_i for the remaining chain species. The spatial distribution of Δf_4 mirrors that found for the chain stretch, with the highest values occuring at the wall of the contraction region owing to the strong shear rates present in this region, whilst being largely being unaffected before the contraction. It is important to note that the nucleation rate increases exponentially with the change in free energy. Hence we expect that all accelerated crystal nucleation will originate in the contraction and be advected from there, with crystal growth occurring downstream.



Chain Species	Max Δf
1	0.025
2	0.037
3	0.038
4	0.42

Chain Species	Max Δf
1	0.03
2	0.042
3	0.043
4	0.51

Chain Species	Max Δf
1	0.035
2	0.046
3	0.047
4	0.59

Chain Species	Max Δf
1	0.038
2	0.049
3	0.05
4	0.65

Figure 3.24: This figure shows how the contraction affects the Δf_4 parameter at the end of the simulations for the different flow rates at $t = 10$ s. The tables underneath give the maximum value forΔf_i for the rest of the chain species in each simulation.

With the stretch and orientation of the different polymer chain species established, we can now look at the development of polymer crystal formation via the logarithmic version of the Schneider rate equations. Figure 3.25 shows how $u_N = \log(1 + N/N_c)$, which measures the number density of crystal nuclei, varies across the centre of the contraction from the wall to the centre of the contraction at $t = 10$ s. The largest values of u_N are found at the wall of the contraction due to the large values of Δf_i there and decreases towards zero away from the wall. Although the values of Δf_i remain greater than zero away from the wall, reduction in free energy there is insufficient to provide a significant increase in the rate of nucleation during the time taken for the polymers to advect through this region when compared with the background rate of heterogeneous nucleation. In contrast close to the wall, not only is the nucleation rate higher, but the no-slip boundary condition means that the polymers have a longer residence time in this region.

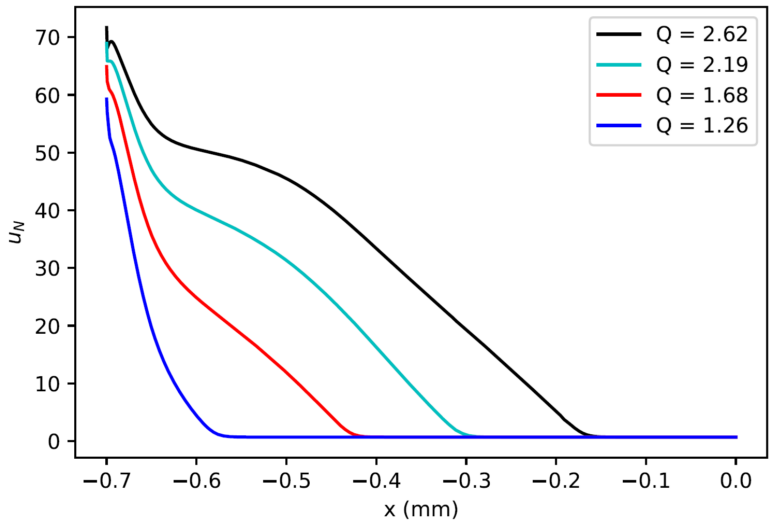


Figure 3.25: Line plot that shows the comparison of the u_N parameter for the different flow rates at $t = 10$ s. This line plot shows the change in u_N from the wall of the contraction region to the centre of the channel in the middle of the mesh. The x-axis represents the distance from the centre of the mesh, where the centre is set at 0.

Starting at the wall in figure 3.25, the shear rate is dominant and this will contribute solely to the crystal development as the extension rate is zero at the wall. As you start moving away from the wall, the shear rate becomes smaller, but is still strong enough to increase crystallisation, so you will see a drop in crystal development. Then the extension rate becomes large enough to affect crystallisation and is working alongside the shear rate to give an increase in crystal development again. As you keep moving further away from the wall, the shear rate continues to reduce, so the crystal development will then drop off. This is because the shear rate is dominant in this geometry, so although the extension rate continues to grow as you move away from the wall, it will not be enough to offset the drop in shear rate from moving to the centre of the channel. This is why you see u_N flatten off when you are 0.1 mm away from the wall and then steadily drop back to zero at the centre.

Figure 3.26 shows the distribution of u_N throughout the flow domain after $t = 10$ s. At low flow-rates crystal nucleation is confined to the narrow region of fluid that has been advected from the wall of the contraction where the strong shear results in an increase in the nucleation rate, $\dot{N}$. This characteristic 'fang' shape forms along a streamline of the flow and is seen in the experiments of Scelsi et al [5]. This 'fang' lengthens and becomes more prominent as the flow rate increases, since the increased flow increases the distance travelled by the polymers from the onset of the flow. There is also a broadening of the crystal nucleation region further into the bulk flow as flow rate increases. This can be attributed to the higher extension rate at the higher flow rate, meaning that extension contributes more to the deformation of polymer chains at the higher flow rates, hence leading to more nucleation further away from the wall in the bulk.

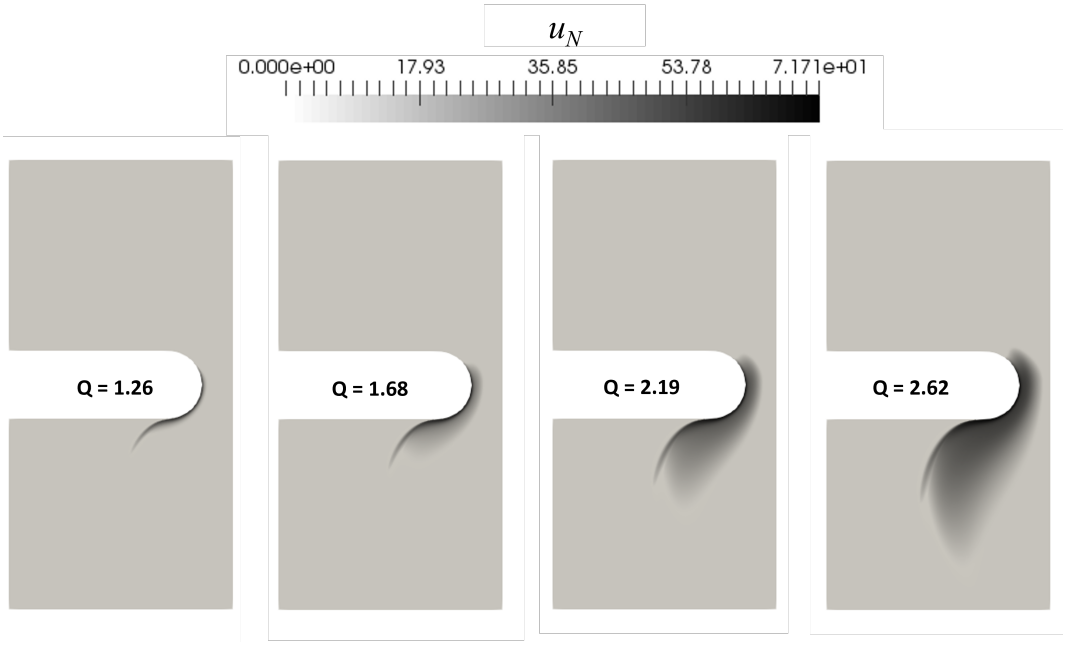


Figure 3.26: This figure shows how the development of u_N changes for the different flow rates at $t = 10$ s.

We then look at how the volume fraction that is not corrected for impingement, $u_V = \log(1 + V/V_c)$, develops in the contraction geometry. Figure 3.27 shows the development of u_V from the wall of the contraction to the centre of the mesh at $t = 10$ s. The shear dominated nature of the simulation means that the largest values of u_V are found at the wall for each flow rate and then decrease to zero as you move further away from the wall. The fact that u_V decreases to zero for each flow rate tells us that although there are crystal nucleation events happening towards the centre of the mesh, owing to non-zero u_N at the centre of the mesh, these events are not resulting in the development of full crystal nuclei. This is most likely a result of the shear rate decreasing as you move from the wall and the extension rate not being large enough to counteract this reduction to allow full crystal nuclei to form in the centre.

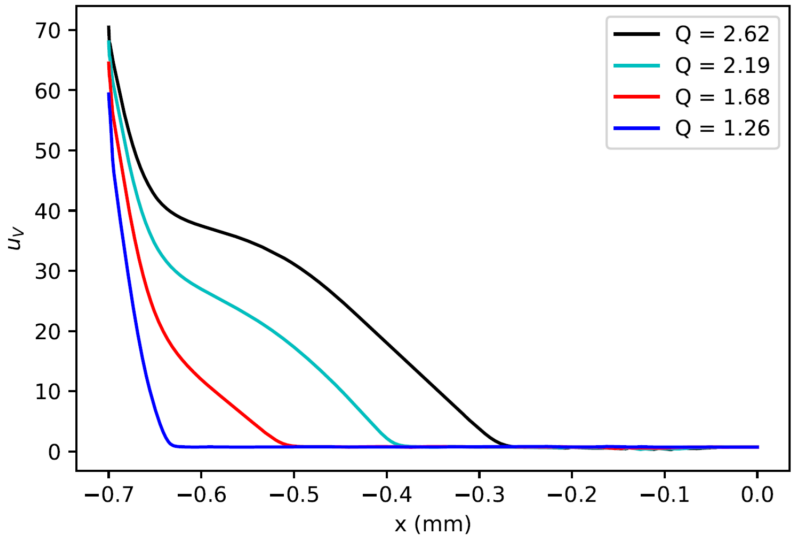


Figure 3.27: Line plot that shows the comparison of the u_V parameter for the different speed simulations at $t = 10$ s. The line plot shows the change in u_V from the wall of the contraction region to the centre of the channel in the middle of the mesh. The x-axis represents the distance from the centre of the mesh, where the centre is set at 0.

Figure 3.28 shows these results for the four flow rates investigated at $t = 10$ s. In a similar way to the previous results for u_N, the strong shear in the contraction results in the highest level of crystal development occurring in the contraction and then getting advected downstream along the streamline. The characteristic 'fang' shape is present again in each flow rate and becomes more defined as you progress through the different flow rate cases.

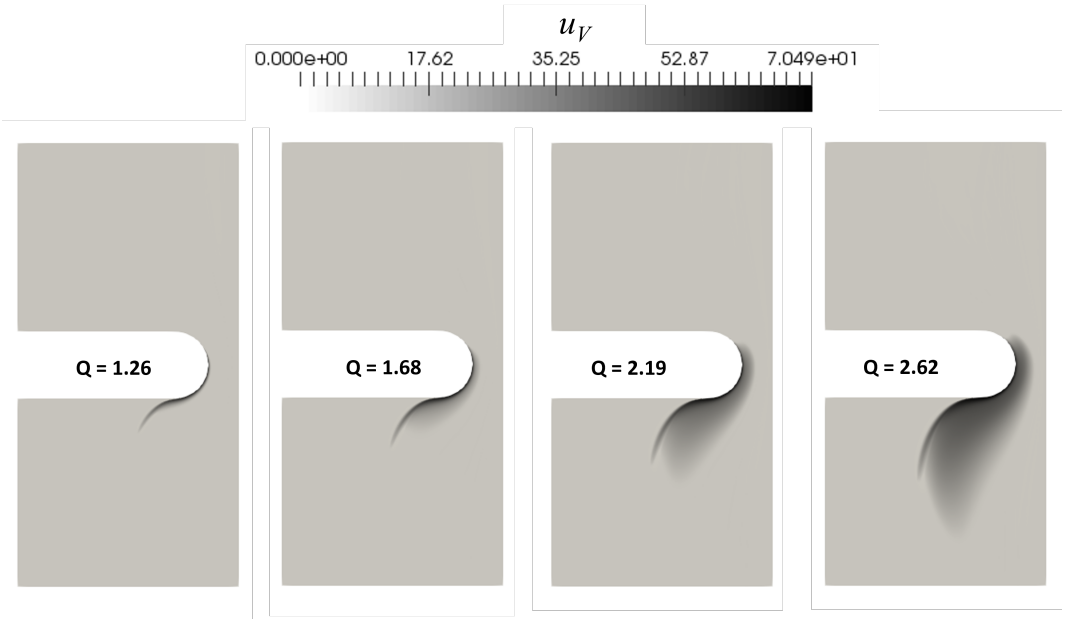


Figure 3.28: This figure shows how u_V changes when flow rate is changed in each of the four different cases investigated at $t = 10$ s.

This then gives us the platform to look at the results of the experimentally measured volume fraction, ξ_G, that is corrected for impingement of crystals. Figure 3.29 shows these results for the four different flow rates investigated at $t = 10$ s. What is clear is that the volume fraction of crystals is developing in the 'fang' shape that is qualitatively consistent with experimental findings. The 'fang' structure becomes more defined as the flow rate increases. As is prevalent throughout all the results given in this chapter, the strong shear forces in the contraction relative to the extensional forces mean that all the crystal volume fraction activity is occurring at the wall of the contraction and is being advected downstream along the streamline to create the 'fang' structure.

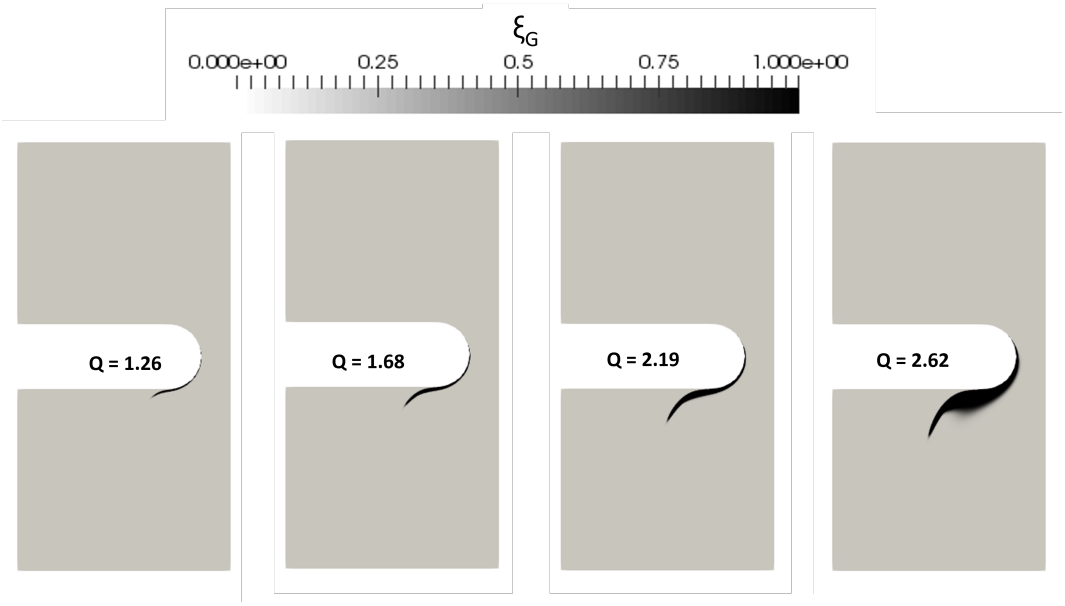


Figure 3.29: This figure shows how ξ_G develops for the different flow rates at $t = 10$ s.

3.7.5 Variation of Gamma

In the original polySTRAND paper supplementary information [2] there was some guidance on how to select the proportionality constant Γ that relates the Kuhn segment nematic order parameter, $P_{2,K,i}$, to the crystallisation free energy change per Kuhn segment, Δf_i. It was suggested that Γ should be of order 1 and that it can be fitted to experimental measurements on FIC. However, our new method of calculating $P_{2,K,i}$ given in section 3.5.2 has a factor of 3/10 that appears in the small stretch limit given by equation 3.5.2. This suggests that Γ should be a factor of 10/3 larger than any value of Γ that was selected based on the original $P_{2,K,i}$ calculation.

The polySTRAND [2] paper quoted a value of $\Gamma = 0.65$ based on MD simulations of a model polyethylene. These MD simulations could directly calculate $P_{2,K,i}$ which would suggest that Γ should be close to this value for polyethylene. However, since the MD simulations combined the carbon and hydrogen atoms for computational efficiency, we believed there would be some discrepancy from their results compared to experimental

results. Therefore, we decided that a value of $\Gamma = 2$ would capture the experimental FIC behaviour observed by Scelsi et al [5].

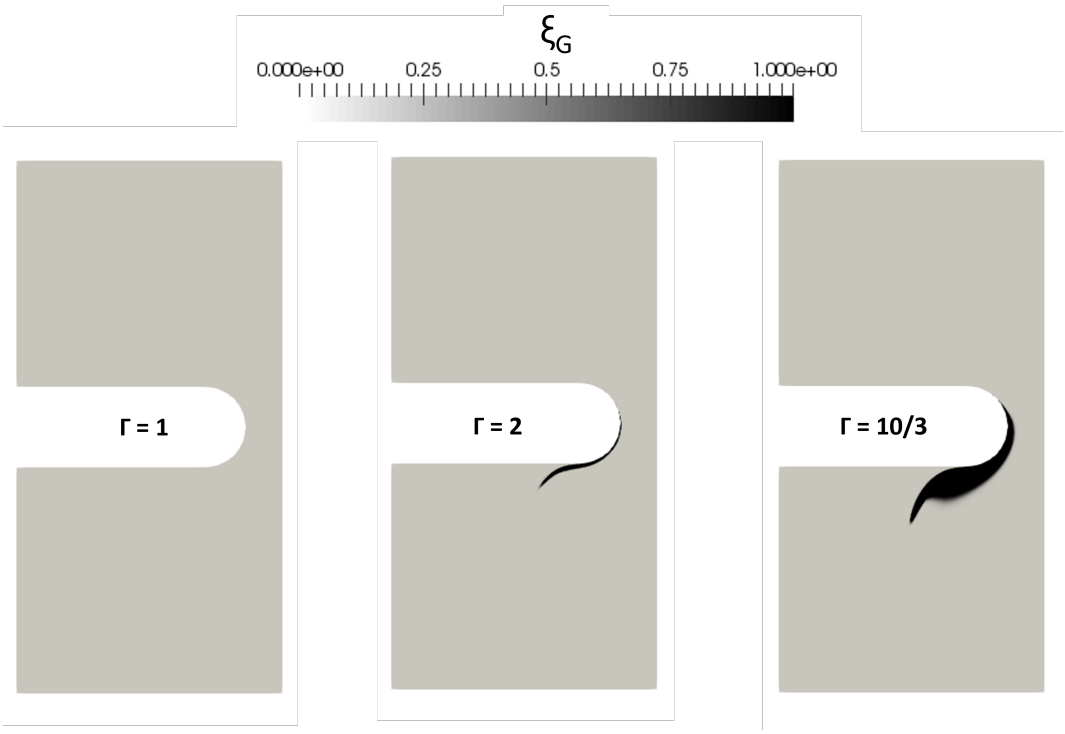


Figure 3.30: This figure shows how variation of Γ affects the profile of ξ_G when Q = 1.68 mm^2s^{-1}.

In figure 3.30 we show the change in ξ_G for different Γ values, when Q = 1.68 mm^2s^{-1}. This illustrates how powerful the Γ parameter can be when modelling FIC. Since Γ directly relates Δf_i with $P_{2,K,i}$ any change in Γ is reflected in Δf_i and the crystal nucleation rate is related to the exponential of Δf_i so these changes are drastic.

3.8 Conclusion

These simulations show the role that polydispersity has on flow induced crystallisation. We also showed the effect different flow rates have on the onset of crystallisation in the polymer melt. We managed to fit a 5 mode RDP model to linear rheology data for HDB6 in line with the decomposition of modes given by Scelsi et al [5]. The non-linear rheology was fitted approximately as our flow was shear dominant.

Turning now to modelling of crystallisation, we found that we needed to make several different developments, both modifying the original polySTRAND model and developing it for inclusion in CFD simulations. First we decided to pre-calculate the flow enhanced nucleation rate using the polySTRAND model in Python and embed this calculation into openFOAM using an interpolation function that was calculated via a least squares regression in an effort to reduce computational cost. Next we found that in strong flow

the value of the bond orientation for each chain species i, $P_{2,K,i}$, was growing beyond physical bounds. Therefore it was necessary to derive a new calculation for $P_{2,K,i}$ using the inverse Langevin function. Lastly, the large orders of magnitude predicted via the original Schneider rate equations were contributing to instabilities in our simulations. In order to mitigate this, we define a logarithmic version of the Schneider rate equations to increase the stability in our runs.

We then showed the effect of multiple chain species contributing to the nucleation rate for four different flow rates. We found that small changes in free energy for nucleation from smaller chain species did contribute in a non-trivial way to the overall nucleation rate owing to their larger volume fraction in the polymer when compared to the longer chain species. The crystallisation patterns in our results agree well with those found in the Scelsi [5] experiment. The fang pattern is a characteristic of the crystal nuclei predominantly forming at the wall of the contraction and are then advected into the main part of the flow. We also demonstrated the profound effect that varying Γ can have on the results for FIC in our simulations, so careful consideration of this parameter is required.

These simulations demonstrate that polymer FIC of a polydisperse isothermal polymer melt is viable in CFD software packages using our methodology. This offers the possibility of testing many different complex flow geometries other than a contraction-expansion flow. This also puts the tools in place to increase the complexity of the simulations further by incorporating temperature dependence. With there being many temperature dependent parameters when modelling crystallisation, this offers the possibility of changing the region of crystallisation by manipulating the temperature profile, would it be possible to force the crystallisation into the centre of the flow? This concept will be explored next in this thesis.

Chapter 4

Heat Flow and Effects of Temperature

4.1 Overview

Having successfully incorporated polydispersity into our simulations on flow induced crystallisation, we will move on to considering temperature dependence as well as polydispersity. Temperature is important in polymer processing because if it is not managed properly, undesirable effects can occur. For example, in the case of metal plating, if a polymer coating is applied above the melting temperature of the polymer, this can cause the polymer to deform; resulting in a non-uniform coating.

Even with temperature control there will be variations in temperature in the apparatus used in polymer processing, such as extruders. Therefore, it is important to incorporate temperature effects into our simulations.

Temperature affects polymer processing in different ways. The rheology of the polymer changes with temperature, as in the relaxation times and viscosity of the polymer change with temperature, and so modifies both the flow and polymer conformation. In addition both the rate of crystal nucleation and growth changes as a direct consequence of parameters involved in calculating the crystallisation rate varying with temperature. Finally, with the heat equation it is important to include the latent heat of crystallisation as this is an additional source of heating. Incorporating these effects will bring our simulations even closer to real polymer processing conditions used in industry as these flows are usually non-isothermal.

4.2 Temperature Dependence of Rheology and Crystallisation

In order to incorporate temperature dependence, we need to determine the temperature dependence of parameters in our models and include these within our openFOAM simulation. Alongside this we need to include the governing equation for heat flow in the

system. We first present the parameter considerations made and then show how the new heat equation is incorporated.

4.2.1 Rheology

One effect of temperature is that it alters the rate of motion of chain segments in the melt. This affects the relaxation timescale for rheology used in the RDP model, the viscosity of the polymer and is also one component of the temperature variation for nucleation and growth rates of crystals.

To calculate how the relaxation times in the RDP model and the viscosity of our polymer melt varies with temperature we utilise the Williams-Landel-Ferry (WLF) [83] relationship given in equation 4.1. The WLF equation is given as,

$$\log_{10}(a_T) = \frac{-c_1(T - T_{ref})}{c_2 + (T - T_{ref})}. \tag{4.1}$$

Where a_T is the WLF shift factor for a given temperature, T is temperature, T_{ref} is a reference temperature and c_1 & c_2 are two constants. The values of c_1 and c_2 vary depending on the value of T_{ref} chosen and are often chosen empirically [84, 85].

4.2.2 Nucleation Rate

In order to understand how temperature affects the nucleation rate, we need to go back to the assumptions within the polySTRAND model [2]. The crystal nucleus is an ellipsiod made up of N_S stems and N_T monomers, as assumed in the GO model [60, 61]. The quiescent nucleus potential is given by $U_{nuc} = -\varepsilon_B N_T + \mu_S S(N_T, N_S)$ where ε_B is the bulk free energy of crystallisation per monomer, μ_S is the surface energy cost and S is the surface area. Changes in ε_B and μ_S reflect how the quiescent nucleus potential varies with temperature. When the temperature changes from above the melting temperature to below the melting temperature of a polymer, the sign of the quiescent nucleus potential changes. This is largely captured by changes in the bulk free energy, ε_B. We also expect μ_S to change with temperature, however the primary effect is captured via changes in ε_B. Therefore, we will only change ε_B with temperature to capture the change in the quiescent nucleation rate due to the change in the energy barrier with temperature.

Another temperature dependent quantity in the polySTRAND model is the Rouse time of an entanglement segment, τ_e. This leads to changes in the monomer attachment time τ_0. The changes in τ_e are consistent with the changes using a WLF [83] shift given by equation 4.1.

To account for this we can take out the WLF factor by defining,

$$\dot{N}_B(T) = a(T)\dot{N}_\Lambda(T), \tag{4.2}$$

where $\dot{N}_B$ accounts for changes in the nucleation rate that arise from changes to the energy barrier but not those associated with the change to τ_e which is handled by the WLF term.

Since the energy barrier now depends on ε_B as well as Δf_i we need to include the variation in ε_B defined as $\Delta\varepsilon_B = \varepsilon_B(T) - \varepsilon_B(T_{ref})$. This newly defined parameter $\Delta\varepsilon_B$ can be incorporated into our interpolation function initially derived for the isothermal case to account for the change in nucleation barrier due to changes in temperature. This $\Delta\varepsilon_B$ parameter acts like a new Δf_i in the interpolation formula. This then gives the new interpolation formula used to pre-calculate the nucleation barrier as:

$$\log\left(\frac{\dot{N}_B(T)}{\dot{N}_q(T_{ref})}\right) = \frac{\sum_i a_i\Delta f_i + \sum_i\sum_{j\leq i} b_{ij}\Delta f_i\Delta f_j + \sum_{k=1,2} c_k\Delta\varepsilon_B^k + \sum_i d_i\Delta f_i\Delta\varepsilon_B}{1 + \sum_i e_i\Delta f_i + \sum_i\sum_{j\leq i} f_{ij}\Delta f_i\Delta f_j + \sum_{k=1,2} g_k\Delta\varepsilon_B^k + \sum_i h_i\Delta f_i\Delta\varepsilon_B}. \tag{4.3}$$

We were concerned that trying to predict all the coefficients for the temperature dependent interpolation function at one time would be (i) computationally expensive and (ii) lead to an inaccurate interpolation. In an effort to circumvent these problems we used the predictions for the groups of coefficients a_i, b_{ij}, e_i and f_{ij} that were calculated during the isothermal investigation i.e. $\Delta\varepsilon_B = 0$.

Next we calculate the nucleation barrier for the quiescent case, but with varying temperature i.e. $\Delta f_i = 0$ to pre-determine the groups of coefficients c_k and g_k. By varying $\Delta\varepsilon_B$ over the range -0.00065 to 0.013 we can predict values for the quiescent nucleation rate for varying temperature. The resulting interpolation from the temperature dependent, quiescent case is given in figure 4.1.

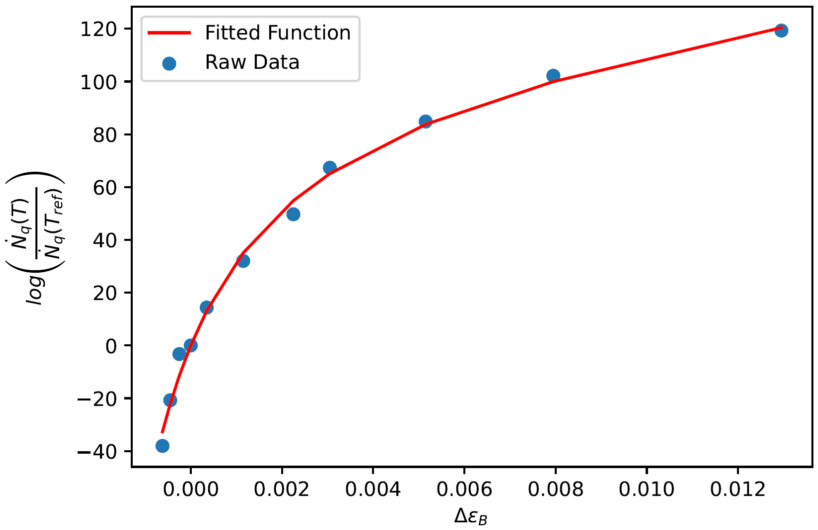


Figure 4.1: This figure shows the relationship between $\Delta\varepsilon_B$ and $\log(\dot{N}_q(T)/\dot{N}_q(T_{ref}))$ for the quiescent case.

Now we have pre-determined predictions for 6 groups of coefficients in equation 4.3 we can use the least squares regression to predict the final 2 groups of coefficients d_i and h_i that involve both $\Delta\varepsilon_B$ and Δf_i.

Just as in the polydisperse, isothermal simulation we examined the ranges of $\Delta f_4 =$

0-0.8, Δf_3 = 0-0.3, Δf_2 = 0-0.3, Δf_1 = 0-0.2 so that we weren't including combinations of Δf_i that were not going to be present in our simulations into the interpolation scheme. The interpolation was successful and gives us a fit to the nucleation rate for different temperature cases that can be used in a full investigation. Figure 4.2 shows the resulting fit of the interpolation for some representative values of $\Delta\varepsilon$ and Δf_i.

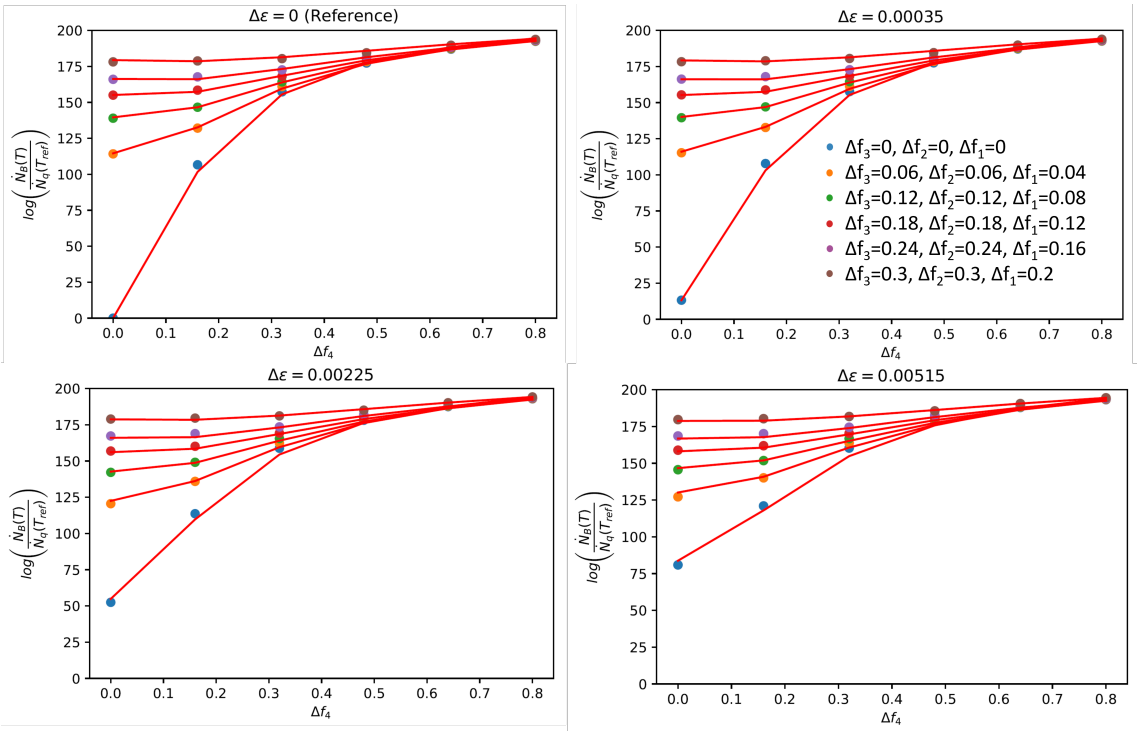


Figure 4.2: Figure showing how the least squares interpolation was fitted to the polySTRAND results for the nucleation rate where temperature is taken into account. Three different temperature cases were chosen along with the reference temperature to show the fit.

The quiescent homogeneous nucleation rate is almost impossible to measure experimentally, therefore we used the MD simulation predictions from Yi et al [9]. Figure 4.3 shows the recreation of their predictions for the free energy barrier, ΔG^*, and the logarithm of the quiescent homogeneous nucleation rate, $\dot{N}_q$. These predictions allow us to establish the relationship between ε_B and temperature.

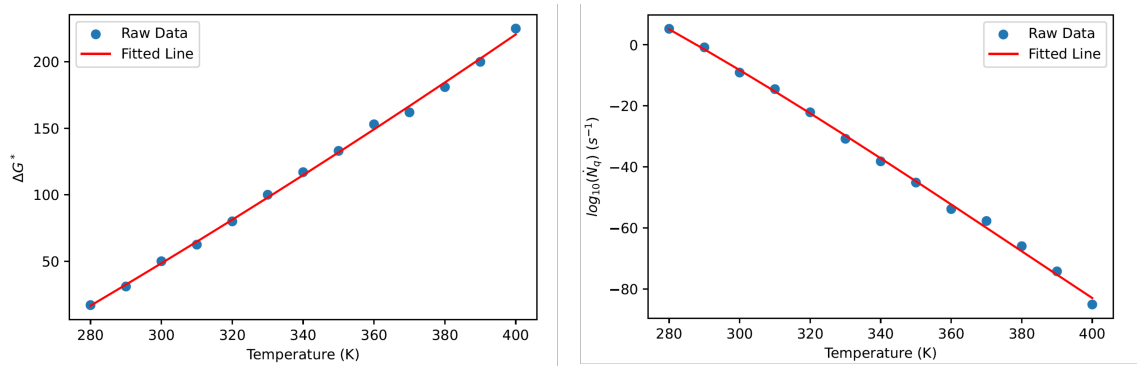


Figure 4.3: This figure shows the recreation of the temperature dependence of the free energy barrier (left) and the quiescent nucleation rate (right) from [9]. Our figure does not take into account the volume, so the units are in s^{-1} for the nucleation rate.

Using the quiescent nucleation rate predictions given in figure 4.3, we can establish the relationship between $\Delta\varepsilon_B$ and temperature. This is illustrated by figure 4.4.

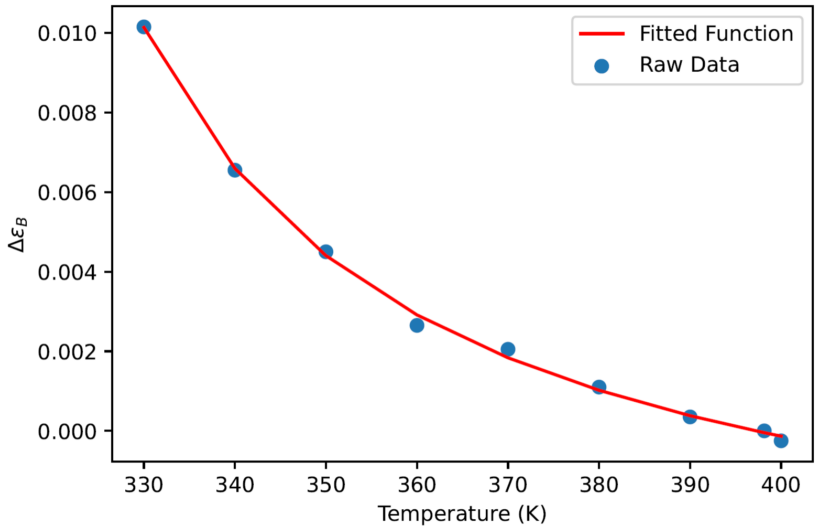


Figure 4.4: This figure shows the relationship between temperature and $\Delta\varepsilon_B$.

4.2.3 Spherulitic Growth Rate

For the spherulitic growth rate, G_s, in the Schneider rate equations Hoffman et al. [10] established a formula for the temperature dependence of G_s for a broad range of molecular weights in the form,

$$G_s(T) = G_0 \exp\left[\frac{-U^*}{R(T-T_\infty)}\right] \exp\left[\frac{-K_g}{T(\Delta T)f}\right]. \tag{4.4}$$

Here K_g is the nucleation constant, G_0 is the scalar factor, U^* is the activation energy (transport of segments to crystallisation site), R is the gas constant, T is temperature, T_∞ is the hypothetical temperature at which relaxation times diverge, which is taken to be 30 degrees below the glass transition temperature T_g so that $T_\infty = T_g - 30$. In the

second exponential term $\Delta T = T_m^0 - T$ with T_m^0 being the equilibrium melting point and $f = (2T)/(T_m^0 + T)$. The two exponential factors derive from the two different ways in which temperature affects the growth rate of crystals. The first corresponds to the rate of motion of the chain segments, as discussed below, whereas the second arises from the energy barrier for chains to be incorporated into the crystal.

The first exponential term in equation 4.4 is another form of the WLF equation given in equation 4.1. Where in place of the constants c_1 and c_2, we have U^* and T_∞. Values for T_∞ can be found in the literature [10], which leaves U^* to be determined. Since we want to remain consistent with our WLF shift, we will use the time temperature shift factor used in chapter 3 of 7.783 to calibrate our calculation. To set up this calculation we consider the WLF shift between two temperatures, T_1 and T_2, given by the first term in equation 4.4 we have,

$$a_T = \frac{r_1}{r_2} = \frac{\exp\left(\frac{-U^*}{R(T_1 - T_\infty)}\right)}{\exp\left(\frac{-U^*}{R(T_2 - T_\infty)}\right)} \tag{4.5}$$

$$= \exp\left(\frac{-U^*}{R}\left[\frac{T_2 - T_1}{(T_1 - T_\infty)(T_2 - T_\infty)}\right]\right) \tag{4.6}$$

Relabelling $T_2 = T$, $T_1 = T_{ref}$ and $T_{ref} - T_\infty = c_2$ then given

$$\log_{10}(a_T) = -\log_{10}(e)\frac{U^*}{R(T_{ref} - T_\infty)}\left(\frac{T - T_{ref}}{c_2 + (T - T_{ref})}\right), \tag{4.7}$$

this gives $c_1 = -\log_{10}(e)U^*/R(T_{ref} - T_\infty)$.

Now that we have established a relationship between these different versions of the WLF equation, we can use the literature result for T_∞ and the shift factor used earlier in the thesis to determine a value for U^* that is consistent with our isothermal investigation.

The temperature shift factor of 7.783 used in chapter 3 shifts from 428.15 K to 398.15 K. Since we would like to have 398.15 K as the reference temperature, the inverse of the shift factor will be used i.e. $a_T = 1/7.783 = 0.128$. After using equation 4.7 we find that $U^* = 5955$ cal/mol. In turn, this gives us the values of $c_1 = 6.6815$ K and $c_2 = 194.95$ K.

Moving to the second exponential term we now need to choose appropriate values for K_g, T_m^0 and G_0. These final three parameters were all found to depend upon molecular weight. We know that HDB6 is a high molecular weight polyethylene and so we chose a sample from Hoffman et al [10] that reflected this. Priority was given to fitting K_g and T_m^0 since they are inside an exponential function. With these values chosen, we then defined G_0 to give the spherulitic growth rate that was used in the isothermal investigation. While this is not the ideal way of choosing G_0, for the purposes of this investigation this provides an adequte proof of concept for this kind of model. The final choices for all the values used in 4.4 are given in table 4.1. This allows us to hard code equation 4.4 into our constitutive model. With all of this in place, the temperature dependence of G_s is plotted in figure 4.5.

Parameter	Value	Units	Justification
K_g	1.725×10^5	K^2	Chosen from a representative sample in [10].
G_0	$10^{10.39}$	cm/s	Chosen to represent the spherulitic growth rate present in the isothermal investigation.
R	1.9855	cal/mol	
U^*	5955	cal/mol	Determined from calculation of WLF coefficients to be consistent with the shift used earlier in the thesis.
T_g	233.2	K	Found in literature [10].
T_m^0	419.15	K	Chosen from a representative sample in [10]

Table 4.1: Table listing the parameter values used for the temperature dependence of G_s. All parameter values were found in the investigation by Hoffman et al [10].

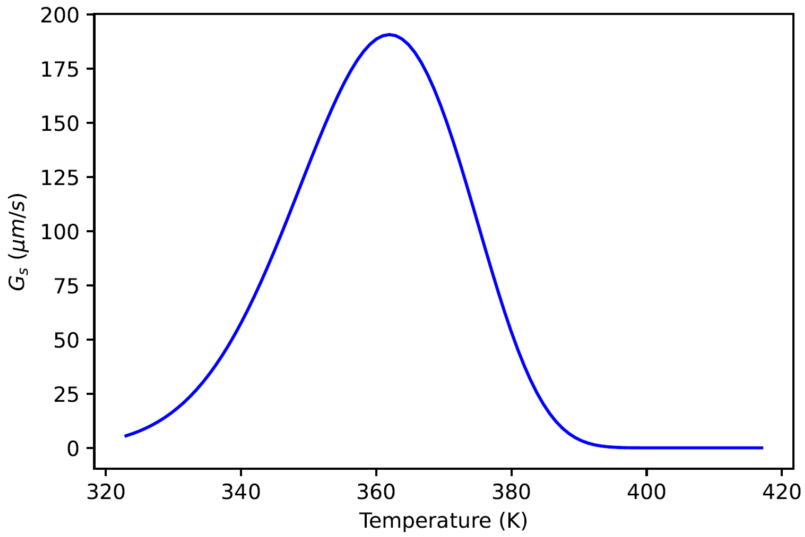


Figure 4.5: Figure that shows how the spherulitic growth rate, G_s, varies with temperature.

We can see that there is a peak in the spherulitic growth rate with varying temperature. In figure 4.5 this peak is a consequence of the two exponential terms $U^*/R(T - T_\infty)$ and $K_g/T(\Delta T)f$ in equation 4.4 having opposing temperature dependence [86]. At low temperatures the growth rate of crystals is restricted by the lack of mobility of chain segments, whereas at higher temperatures it is the energy barrier to crystallisation that prevents crystal growth.

The temperature dependence of G_s requires us to make a modification to the logarithmic versions of the Schneider rate equations. These equations are summarised below,

$$\begin{aligned}
\frac{Du_N}{Dt} &= \frac{\dot{N} + \dot{N}_{het}}{N_c e^{u_N}}, \\
\frac{Du_R}{Dt} &= \frac{2G_s N}{R_c e^{u_R}}, \\
\frac{Du_S}{Dt} &= \frac{4\pi G_s R}{S_c e^{u_S}}, \\
\frac{Du_V}{Dt} &= \frac{G_s S_{tot}}{V_c e^{u_V}},
\end{aligned}$$

where $N_c = \dot{N}_{het} t_{char}$, $R_c = \dot{N}_{het} G_s t_{char}^2$, $S_c = \dot{N}_{het} G_s^2 t_{char}^3$ and $V_c = \dot{N}_{het} G_s^3 t_{char}^4$.

In section 3.6.2 where these equations were derived, G_s was constant because the system was isothermal. With there being temperature dependence of G_s we now need to define an alternative scale $G_{s,0}$ to represent the typical spherulitic growth rate of the simulation, which is chosen to be one of the temperatures that is most relevant to the crystal growth rate expected in the simulation. This new parameter $G_{s,0}$ will then be incorporated into R_c, S_c and V_c whilst G_s will represent the local spherulitic growth rate that depends on temperature, so the updated equations are,

$$\begin{aligned}
\frac{Du_N}{Dt} &= \frac{\dot{N} + \dot{N}_{het}}{N_c e^{u_N}}, \\
\frac{Du_R}{Dt} &= \frac{2G_s N}{R_c e^{u_R}}, \\
\frac{Du_S}{Dt} &= \frac{4\pi G_s R}{S_c e^{u_S}}, \\
\frac{Du_V}{Dt} &= \frac{G_s S_{tot}}{V_c e^{u_V}},
\end{aligned}$$

where $N_c = \dot{N}_{het} t_{char}$, $R_c = \dot{N}_{het} G_{s,0} t_{char}^2$, $S_c = \dot{N}_{het} G_{s,0}^2 t_{char}^3$ and $V_c = \dot{N}_{het} G_{s,0}^3 t_{char}^4$.

4.2.4 Energy Equation

With all the foundations in place to predict the rheology, growth rate and nucleation rate changes with temperature, we can look at the final alteration that needs to be considered. For a non-isothermal flow, we need to also calculate the evolution of the temperature distribution from the conservation of energy in the system. RheoTool [4] already includes a facility to handle non-isothermal flows, which includes solving the energy equation in the form

$$\rho c_p \frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T = \nabla \cdot (\kappa \nabla T) + \boldsymbol{\tau}' : \nabla \mathbf{u} + S_T + R_T, \tag{4.8}$$

where c_p is the specific heat capacity, κ is the thermal conductivity, S_T is a volumetric source of energy and R_T is the heat transferred by radiation.

However we need to also include the latent heat of crystallisation. This is done by adding the term,

$$lH\frac{\mathrm{d}\xi_G}{\mathrm{d}t} \tag{4.9}$$

as a source term into equation 4.8, where lH is the heat of fusion for crystals and ξ_G is our volume fraction of crystals. So the final energy equation becomes,

$$\rho c_p\frac{\partial T}{\partial t} + \mathbf{u}\cdot\nabla T = \nabla\cdot(\kappa\nabla T) + \boldsymbol{\tau}':\nabla\mathbf{u} + S_T + R_T + lH\frac{\mathrm{d}\xi_G}{\mathrm{d}t}. \tag{4.10}$$

In order to achieve this it was necessary to create a new fluid thermo model in openFOAM so that we could customise the energy equation to our needs. However, due to the structure of openFOAM, it is not straightforward to utilise a result from a constitutive model (ξ_G) in the energy equation. This is because the constitutive equation and energy equation are in separate libraries in openFOAM. With some external help given by Francisco Pimenta who suggested using the lookup utility in openFOAM where any object can read any field that already exists in a simulation, I was able to include this term in the energy equation.

The final task is to determine appropriate values for the parameters in the energy equation, which are given in table 4.2.

Parameter	Value	Units	Reference
ρC_p	2.5	MPa/K	Almanza et al. [87]
κ	0.24	MPa mm^2/s K	Yu Jia et al. [88]
lH	278	MPa	Mirabella et al.[89]

Table 4.2: Table listing the parameter values used in the energy equation.

With all of these considerations in place, we now have the foundation to simulate polydisperse, non-isothermal polymer melts using our framework.

4.3 Shear Flow with Imposed Temperature History

4.3.1 Previous Experimental Work

In experimental studies of flow induced crystallisation it is common to conduct experiments where a flow and temperature history is imposed on a polymer melt and the effects of this on crystallisation are then measured. A common experimental protocol is to extrude the polymer melt and let it relax at a temperature above the equilibrium melting temperature to ensure any flow history has relaxed from the extrusion process. The polymer is then

cooled to a target crystallisation temperature and then sheared for a pre-determined length of time. Flow induced crystallisation effects are measured after shear has stopped. In all these investigations [90, 91, 92, 93] they found that flow increased the rate of nucleation along with the presence of flow producing highly oriented structures where the shear rates are highest.

Another method [6] is to extrude the polymer melt and let it relax above the melting temperature, as previously mentioned. The polymer is then cooled to a shearing temperature, where a shearing pulse is imposed. The melt is then cooled further and flow induced effects are measured after the second phase of cooling. Fernandez et al [6] used this protocol and found that there was negligible crystallisation in their iPP melt before the second cooling phase. After they reduced the temperature a second time they saw highly oriented crystallites. They concluded that the orientation of the crystallites suggests that a material subjected to the same cooling regime, but no shear pulse, would not show the same characteristics.

There have also been studies looking at the crystallisation effects of extensional flow [94] where the melt was cooled to a crystallisation temperature and then subjected to an extensional flow. The investigation by Swartjes et al [94] found that the extensional flow created oriented fibre-like structures around the outflow centre-line due to the flow imposed.

Taking inspiration from Fernandez et al [6] we will replicate a similar investigation in openFOAM where the polymer melt is sheared and then cooled to impose a flow and thermal history. We will then demonstrate the effects on crystallisation within the modelling framework. This experimental set up makes the most practical sense for our CFD protocol.

4.3.2 Simulation Conditions

We will impose a flow and thermal history on our polymer melt in a channel geometry. A channel geometry best replicates the flow apparatus used by Fernandez et al [6]. It is worth noting that experimental investigations are run over many minutes, due to the much slower rate of crystal growth in iPP, which makes observations easier to make in the experiments. However, this makes computation challenging due to the small time step that is needed when flow is imposed in our simulation. While in principle we could run separate simulations for the flow induced nucleation and crystal growth phases, we will instead simulate polyethylene for which the rate of crystal growth is much faster and for which we already have the parameters. Therefore, we are merely taking inspiration from the experimental work in this field to illustrate how our protocol could be used in such cases. The channel geometry will be created by using a mesh made up of one block that contains 50 cells in the x-direction and 10 cells in the y-direction. The domain is 20mm long and 2mm tall which is a channel of a 1:10 ratio, this is the same ratio as the apparatus used in the investigation by Fernandez et al [6].

We will consider 5 different realisations where the melt is sheared for the first 2 seconds with the temperature at 398.15 K. The melt is then cooled to a final temperature of 368.15 K and the wait time before this cooling starts is varied between each run. We will show the cases of a 1.5, 5, 10, 20 and 40 second wait before cooling starts. The rate at which the temperature is reduced is kept constant between each of the runs at 1 degree per second, so the cooling takes place over a 30 second interval. The melt is then held at the cooler temperature for 8 seconds for each run. We will examine if the wait time before cooling affects the crystallisation profile in the melt. The spherulitic growth rate at 398.15 K is $G_s = 0.3\ \mu\text{ms}^{-1}$ and at 368.15 K is $G_s = 168.62\ \mu\text{ms}^{-1}$ and we will choose the growth rate scale as, $G_{s,0} = 100\ \mu\text{ms}^{-1}$ to be used in the Schneider rate equations.

The temperature profile in the simulations is shown in figure 4.6. This shows the temperature profile for the 1.5 second wait time before cooling.

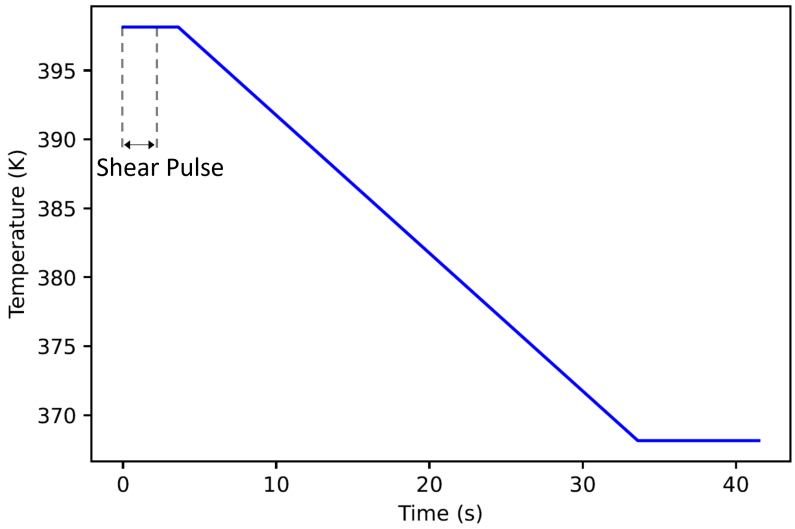


Figure 4.6: Plot that shows how the temperature of the walls change throughout the simulation when the wait time before cooling starts is 1.5 s.

This temperature profile is incorporated into our openFOAM simulation via a look-up table for the boundary conditions at the walls and the inlet of the mesh. The shear pulse is defined through time dependent boundary conditions at the inlet and the walls of the mesh. The remaining boundary conditions used in this investigation are given in table 4.3 and are standard commands in rheoTool.

	p	U	T	τ
Inlet	`zeroGradient`	`codedFixed Value`	`uniformFixed Value`	`zeroGradient`
Top Wall	`zeroGradient`	`codedFixed Value`	`uniformFixed Value`	`linear Extrapolation`
Bottom Wall	`zeroGradient`	`fixedValue`	`uniformFixed Value`	`linear Extrapolation`
Outlet	`fixedValue`	`zeroGradient`	`zeroGradient`	`zeroGradient`
frontAndBack	`empty`	`empty`	`empty`	`empty`
Symmetry	`symmetryPlane`	`symmetryPlane`	`symmetryPlane`	`symmetryPlane`

Table 4.3: Table showing the boundary conditions used for the simulation.

Since the rate of cooling and the shear pulse are kept constant for each of the different iterations, we will only show the details of the temperature and the velocity profile for the simulation where the wait time was 1.5 seconds.

The imposed shear velocity and the resulting velocity gradient across the channel are given in figure 4.7. The velocity gradient is approximately constant across the channel, as expected in shear flow.

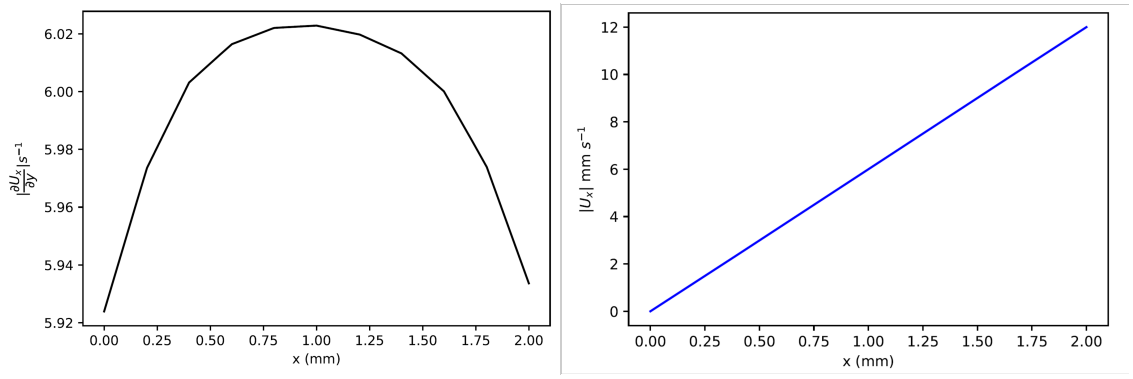


Figure 4.7: Figure that shows the magnitude of the velocity gradient across the width of the channel (left) and the shear velocity profile imposed (right).

Figure 4.8 shows how the temperature field changes throughout the simulation. We have shown the temperature at the end of shearing at 2 seconds and at the end of the simulation at 41.5 seconds. The temperature at the end of shearing does drift slightly above the temperature set in the initial conditions in our simulation, this can be attributed to viscous heating within the material. We note, in particular, that the temperature in the upper left of the simulation remains close to the initial temperature. This is due to new material at the initial temperature being advected into the simulation from the left hand boundary, which is not subjected to the same total shear history. At the end of the simulation the temperature has been cooled at the walls and the temperature profile has

reached a steady state. The temperature at the centre of the domain is slightly higher than the walls and this can also be attributed to the viscous heating present in the simulation. We can see that with the width of the channel chosen, the diffusion of heat is fast, but not instantaneous.

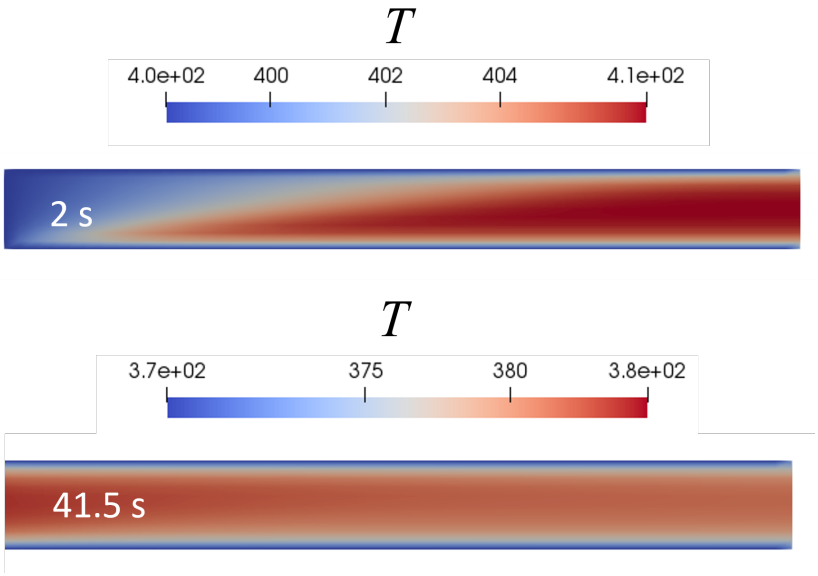


Figure 4.8: Figure that shows the temperature field at the end of shearing (top) and at the end of the simulation (bottom).

4.3.3 Results

We will now show how some of the characteristics of the polymer behave for this flow and temperature history. The crystal volume fraction, ξ_G, as calculated via the Avrami equation is shown for the entire channel in figure 4.9. The largest areas of crystal formation are at the walls and this is expected as cooling is fastest at the walls to allow for a higher rate of crystal growth. This profile was consistent between all the different simulations.

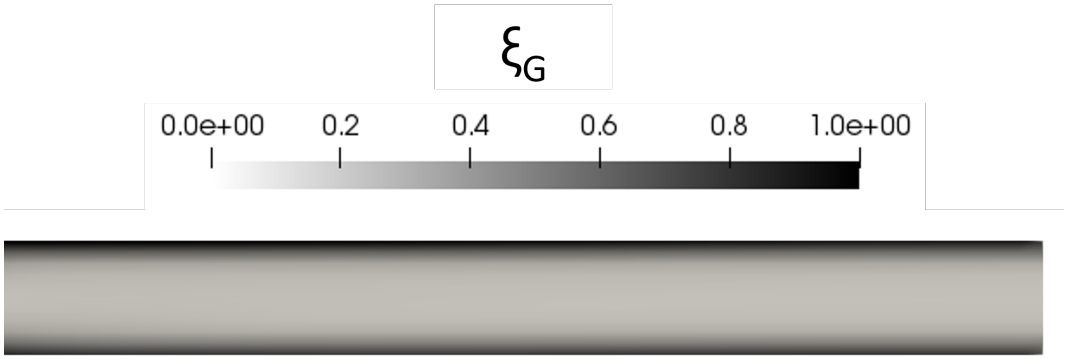


Figure 4.9: Figure showing the overall profile of ξ_G at the end of the simulations. This profile was consistent between the different runs, so we have chosen only one to show here and this was when the material was held at the high temperature for 1.5 seconds before cooling.

Figure 4.10 shows the profile of ξ_G across the width of the channel, 15 mm along the channel at the end of the simulations. Note that each simulation runs for the same time (8 s) after cooling. At this time the profile of ξ_G is almost identical between runs, so the wait time before cooling starts does not have a significant effect on the overall crystallisation profile.

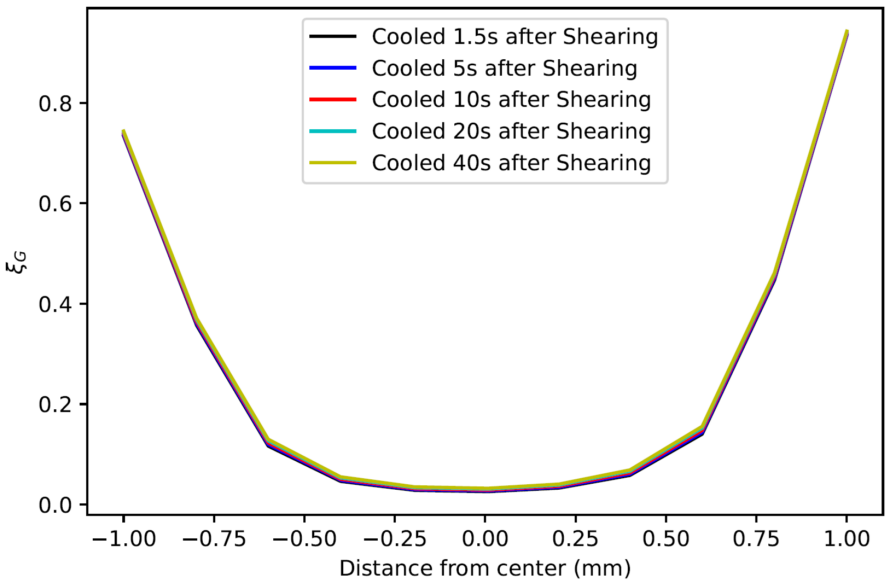


Figure 4.10: Figure showing the profile of ξ_G across the channel, 15 mm along the channel at the end of the simulations.

For all our simulation runs we will look at how quantities change over time at the wall of the channel, three quarters of the way down the channel from the inlet as sketched in figure 4.11. This point was chosen because it is at a point where the simulation has reached steady state.

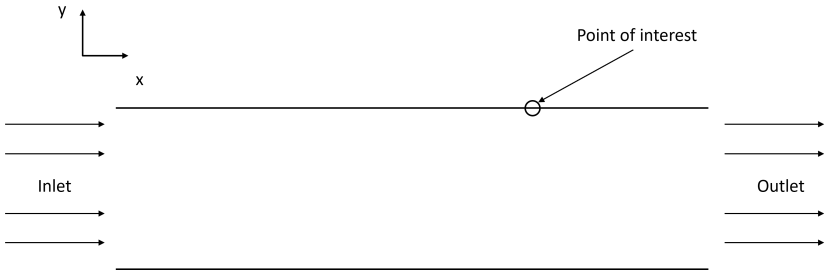


Figure 4.11: Figure showing the point of interest in the temperature dependent shear flow investigation.

Figure 4.12 then shows how ξ_G develops over time for different waiting times. The shape of the curve for ξ_G is practically identical between the runs and the delay in the onset of cooling does not appear to increase the volume fraction of crystals by a significant degree.

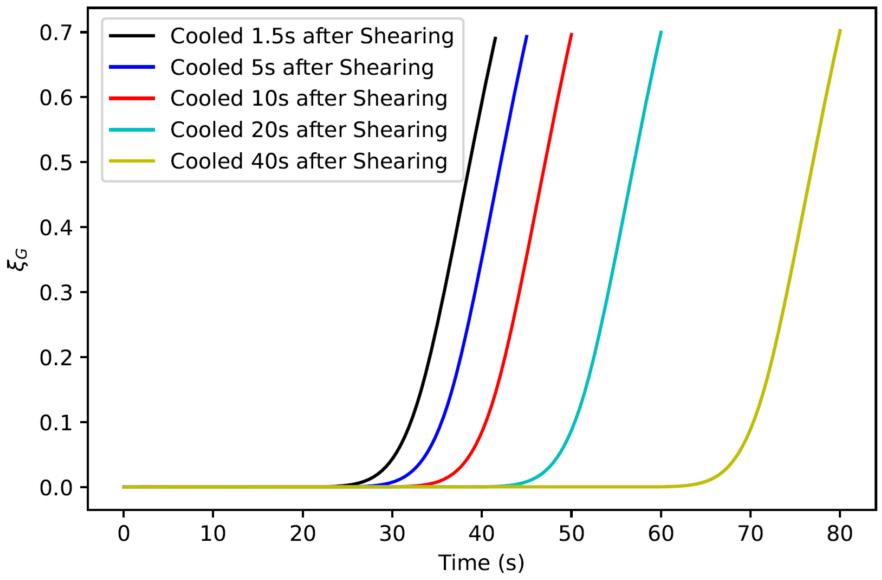


Figure 4.12: Plot showing how ξ_G develops over time at the wall for each simulation.

In figure 4.13 we horizontally shift the different ξ_G curves for each simulation so that $t = 0$ is the time when cooling is started. The curves almost match perfectly. With there being minimal difference in ξ_G, we can conclude that the wait time between the shear pulse and the initiation of cooling does not have a significant effect on the crystal volume fraction profile in shear flow.

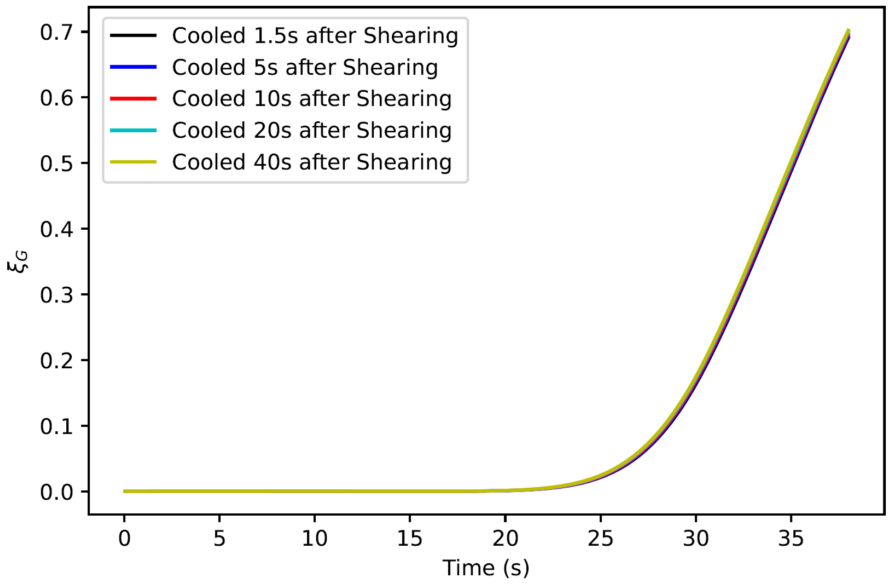


Figure 4.13: Plot showing a superposition of the profile of ξ_G for the different simulations, where $t = 0$ is when cooling starts in each run.

Having identified that the overall crystallisation profile in the channel is largely independent of wait time before cooling is initiated, we can now look at the reasons for this behaviour.

Figure 4.14 shows how Δf_4 and λ_4 develop over time in our shear flow simulations. We only show the stretch and orientation for the slowest relaxing chain species in our material because this has the largest contribution to the crystallisation in the melt. The other chain species have the same profile, but with a smaller degree of stretch and orientation. This profile for the stretch and orientation is consistent across all the simulation runs with different wait times before cooling of the material happens.

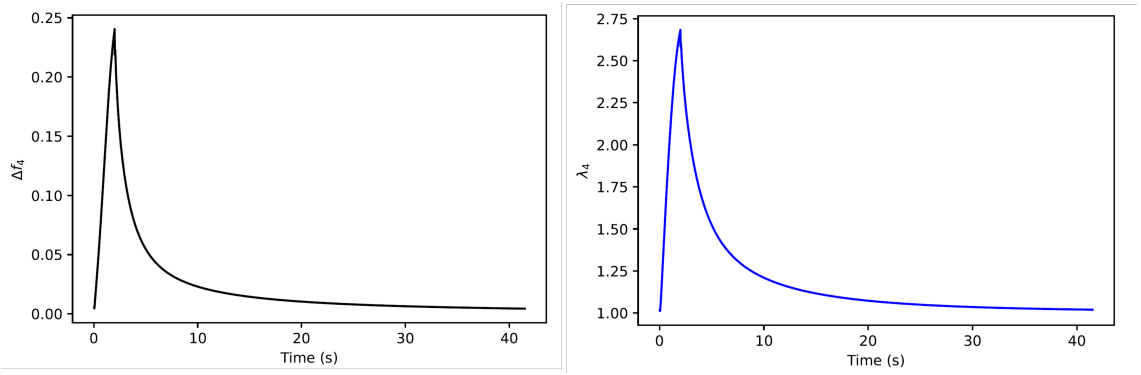


Figure 4.14: Figure showing how Δf_4 (left) and λ_4 (right) develop over time in our simulation when the wait time is 1.5 s. This profile was independent of wait time before cooling, so only these results are shown for simplicity.

Figure 4.15 shows how $\log(\dot{N})$ develops over time for each of the wait times before cooling. The shear pulse creates a spike for the nucleation rate, reaching a maximum

when the shear pulse ends at 2 s. This is followed by a sharp decrease in the crystal nucleation rate in the absence of flow, as a result of the relaxation of the stretch and orientation of the chain species in the melt. We notice that as the wait time before cooling increases, $\log(\dot{N})$ drops lower in the wait period, this can be explained by the polymer chain species becoming less stretched and aligned in the absence of flow in the longer wait periods and hence giving a lower nucleation rate for longer wait times. This is then followed by an increase in the nucleation rate due to the WLF shift that is appropriate for the reduction in the temperature of the melt, however, the crystal nucleation rate remains very small after the initial spike from the shear pulse. It is important to note that figure 4.15 shows the logarithm of the rate of nucleation and therefore that in practice all large negative values mean there is no homogeneous nucleation occuring.

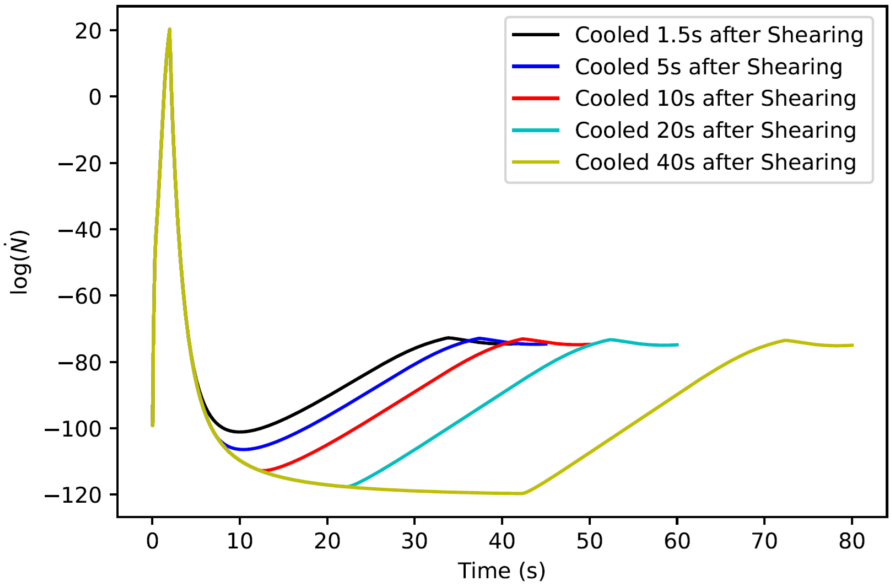


Figure 4.15: Figure showing how $\log(\dot{N})$ develops over time at the wall for each of the simulation runs.

With the rheological impact of the shear pulse illustrated, we can now look at how this affects the logarithmic quantities in the Schneider rate equations, namely u_N and u_V. The evolution equations for these variables, as outlined in section 3.6.2, are repeated here,

$$\begin{aligned} \frac{Du_N}{Dt} &= \frac{\dot{N} + \dot{N}_{het}}{N_c e^{u_N}}, \\ \frac{Du_V}{Dt} &= \frac{G_s S_{tot}}{V_c e^{u_V}}, \end{aligned} \tag{4.11}$$

where $N_c = \dot{N}_{het} t_{char}$ and $V_c = \dot{N}_{het} G_{s,0}^3 t_{char}^4$.

Figure 4.16 shows how u_N and u_V develop over time at the wall of the channel for each wait time before cooling starts. We can see that the profile of u_N is essentially the same for all runs, independent of the waiting time. In fact, u_N is almost a step function,

since the nucleation rate is small except for a short period at the end of the shear pulse when the polymer orientation and stretch is maximised. Essentially all the crystal nuclei form at this time, and the number of nuclei remains fixed thereafter.

However the growth of the crystals, represented by the parameter, u_V, does change between runs. This is because the spherulitic growth rate, G_s, in the Schneider rate equations is much larger at 368.15 Kelvin compared to 398.15 Kelvin. The final value of u_V is essentially the same for all runs - the increased wait time before cooling only serves to delay the onset of crystallisation.

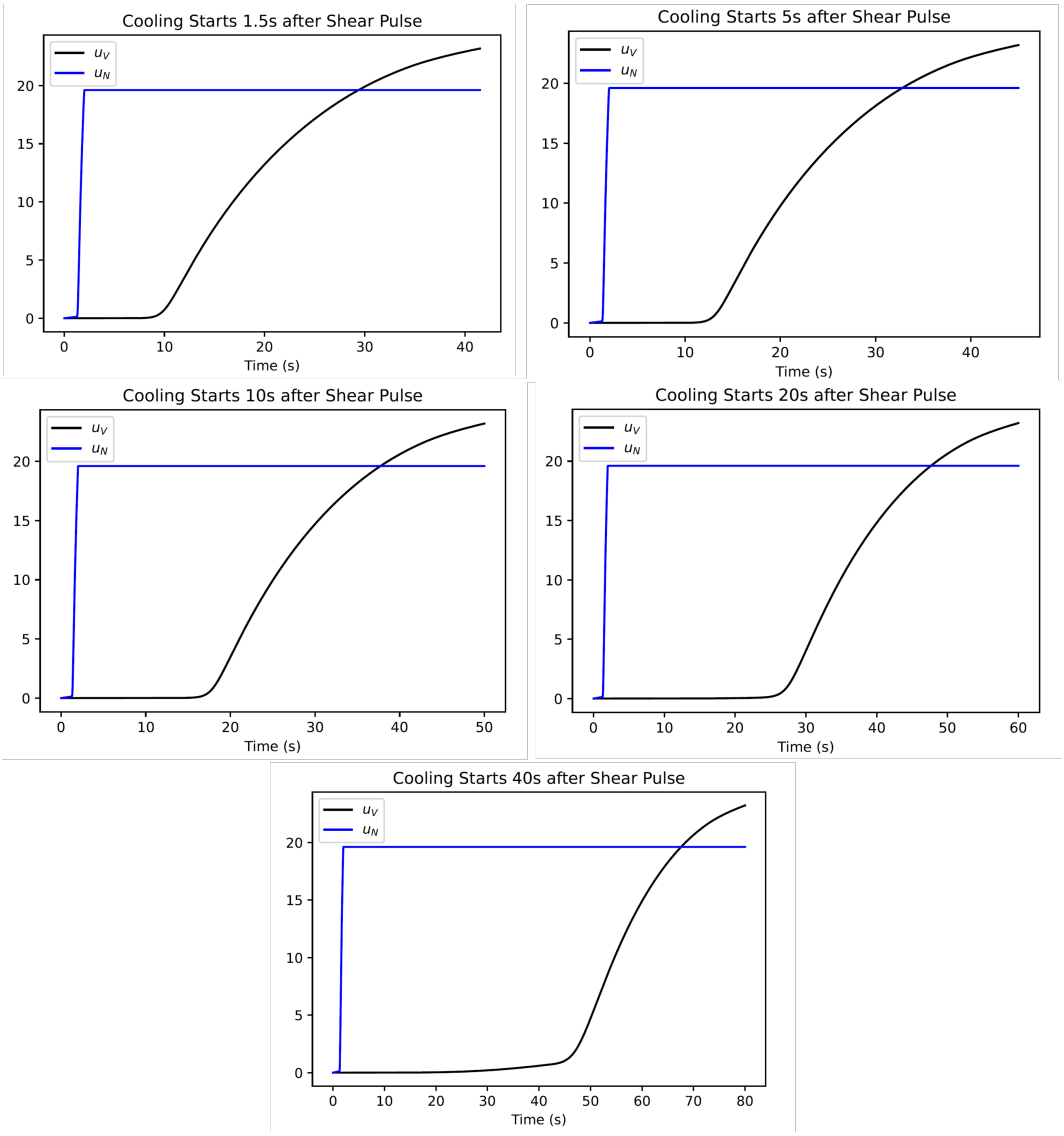


Figure 4.16: Figure showing how u_N and u_V develop over time at the wall of the channel for the different simulation runs.

Overall we have demonstrated that our protocol can be used to investigate the effect that temperature dependent shear flow can have on FIC. We have shown that the shear pulse is what controls the overall volume fraction of polymer crystals in our material and

the wait time before cooling the melt had a minimal effect. With an investigation that shows simple flow, we can move on to showing our framework in the context of temperature dependent complex flow.

4.4 Non-Isothermal Polymer Flow in a Complex Geometry

We return to our consideration of the contraction-expansion flow but for non-isothermal flows. In particular we will consider the case where the walls are held at a different temperature from that of the melt. This is particularly of interest in e.g. injection moulding when a melt at one temperature is injected into a mould held at a different temperature.

4.4.1 Newtonian Fluid Flowing Past a Heated Wall

In order to study how heat diffusion and flow interact, we consider a simplified shear problem of a fluid moving past a heated wall. The motivation for discussing the case of a Newtonian fluid flowing past a heated wall is to get a better understanding of how the thermal boundary layer may behave in our complex flow geometry. With this in mind we will be able to identify how far across the contraction region heat will diffuse.

For $x < 0$ the wall and the fluid have the same temperature, T_0. For $x \geq 0$ the wall is a higher temperature T_1. We will look for a steady state solution and we impose no-slip boundary conditions at the wall. A sketch of the problem is given in figure 4.17.

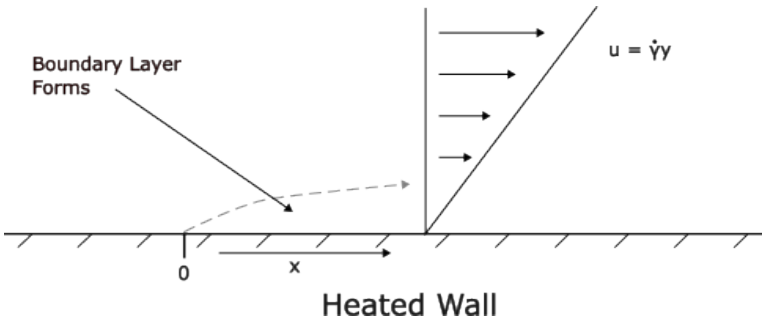


Figure 4.17: Sketch showing the shear and diffusion problem.

With all these conditions given, we can derive this problem by considering equation 4.10. Since we are looking for a steady state solution, $\partial/\partial t = 0$, we will neglect viscous heating, so there are no volumetric sources of energy. We also assume that the thermal conductivity is constant and the effects of radiation and crystallisation are not included here. Therefore, the governing equation for this problem becomes,

$$u\frac{\partial T(x,y)}{\partial x} = \kappa\frac{\partial^2 T(x,y)}{\partial y^2} \tag{4.12}$$

with $u = \dot{\gamma}y$, where $\dot{\gamma}$ is the shear rate and κ is the thermal conductivity of the fluid. This leads to

$$\dot{\gamma} y \frac{\partial T}{\partial x} = \kappa \frac{\partial^2 T}{\partial y^2}. \tag{4.13}$$

In order to solve this we look for a similarity solution of the form

$$\eta = x^{\alpha} y, \tag{4.14}$$

for some constant α such that $T \equiv T(\eta)$. We can now compute the derivative of T with respect to x and y in terms of it's derivative with respect to eta.

$$\frac{\partial T}{\partial x} = T' \alpha x^{\alpha - 1} y, \tag{4.15}$$

$$\frac{\partial T}{\partial y} = T' x^{\alpha}, \tag{4.16}$$

$$\frac{\partial^2 T}{\partial y^2} = T'' x^{2\alpha}. \tag{4.17}$$

These results can be substituted into equation 4.13 to give

$$\dot{\gamma} \alpha x^{\alpha - 1} y^2 T' = \kappa x^{2\alpha} T'' \tag{4.18}$$

$$\implies T'' = \frac{\dot{\gamma} \alpha x^{-\alpha - 1} y^2}{\kappa} T'. \tag{4.19}$$

Now T is a function of η only, therefore equation 4.19 can only depend on η rather than x and y separately. This means that the factor $x^{-\alpha-1} y^2 = \eta^{\beta} = (x^{\alpha} y)^{\beta}$ for some exponent β. Equating powers of y gives $\beta = 2$. Equating powers of x now gives $2\alpha = -\alpha - 1$, which solves to give $\alpha = -1/3$.

We can now substitute $\eta^2 = x^{-\alpha-1} y^2$ and $\alpha = -1/3$ into equation 4.19 and we get

$$-\frac{\dot{\gamma}}{3\kappa} \eta^2 T' = T'', \tag{4.20}$$

then substitute $T' = v$ to get

$$-\frac{\dot{\gamma}}{3\kappa} \eta^2 v = v', \tag{4.21}$$

which gives

$$v = A \exp\left(\frac{-\dot{\gamma} \eta^3}{9\kappa}\right), \tag{4.22}$$

where A is a constant. Therefore,

$$\frac{\mathrm{d}T}{\mathrm{d}\eta} = A \exp\left(\frac{-\dot{\gamma} \eta^3}{9\kappa}\right), \tag{4.23}$$

which gives

$$T = A \int_0^{\eta} \exp\left(\frac{-\dot{\gamma}\eta'^3}{9\kappa}\right) \, \mathrm{d}\eta' + B, \tag{4.24}$$

for a constant B.

As $y \to \infty$, $\eta \to \infty$ the temperature of the fluid is that of the bulk and is not affected by the wall temperature. Therefore we have

$$T_0 = A \int_0^{\infty} \exp\left(\frac{-\dot{\gamma}\eta'^3}{9\kappa}\right) \, \mathrm{d}\eta' + B. \tag{4.25}$$

For convenience we use the change of variables $P^3 = \dot{\gamma}\eta'^3/9\kappa$ which changes our equation to

$$T_0 = A \left(\frac{\dot{\gamma}}{9\kappa}\right)^{-\frac{1}{3}} \int_0^{\infty} \exp\left(-P^3\right) \, \mathrm{d}P + B. \tag{4.26}$$

Since

$$\int_0^{\infty} \exp\left(-P^3\right) \, \mathrm{d}P = \Gamma\left(\frac{4}{3}\right) \simeq 0.89298, \tag{4.27}$$

where Γ is the Gamma function. This gives

$$A = \frac{T_0 - B\left(\frac{\dot{\gamma}}{9\kappa}\right)^{\frac{1}{3}}}{\Gamma\left(\frac{4}{3}\right)}. \tag{4.28}$$

Our second boundary condition that is used to find the value of B is that $T = T_1$ when $\eta = 0$ and hence $P = 0$. This gives the relationship of $T_1 = B$.

Substituting our values for A and B back into equation 4.24 we have

$$T = \frac{T_0 - T_1\left(\frac{\dot{\gamma}}{9\kappa}\right)^{\frac{1}{3}}}{\Gamma\left(\frac{4}{3}\right)} \int_0^{\eta} \exp\left(\frac{-\dot{\gamma}\eta'^3}{9\kappa}\right) \, \mathrm{d}\eta' + T_1. \tag{4.29}$$

The exponent in equation 4.29 sets the width of the boundary layer, which is roughly given by $\eta^3 = 9\kappa/\dot{\gamma}$. Taking the width of the boundary layer y as a function of x,

$$y = \left(\frac{9\kappa x}{\dot{\gamma}}\right)^{\frac{1}{3}}. \tag{4.30}$$

Equation 4.30 provides an estimate for the thermal boundary layer near a wall. In particular it allows us to estimate how far the thermal boundary layer extends in the contraction region. For $\kappa \simeq 0.1$ mm^{2}s^{-1}, $x = 0.5$ mm & $\dot{\gamma} = 20$ s^{-1} this gives a thermal boundary of around 0.3 mm. Meaning that heat will not diffuse across the entire cross-section so that a temperature difference will remain between the wall and the centre of the contraction.

4.4.2 Cold Wall

We will now look at how changes in temperature can affect the flow in the contraction expansion geometry used in chapter 3. We will keep the rheology the same for this investigation as in chapter 3. A recap of this rheology is given in table 4.4.

Chain Species	ϕ_i	$\tau_{d,i}$	$\tau_{s,i}$
Solvent	0.7668	5×10^{-4} s	5×10^{-4} s
1	0.09361	0.18 s	0.078 s
2	0.06847	1.81 s	0.078 s
3	0.04556	17.87 s	0.078 s
4	0.02556	291.86 s	3.89 s

Table 4.4: Table showing the relaxation times used in the polydisperse, isothermal investigation.

The blend contains five species, with one species acting as a solvent with a fast relaxation time. We shall compare the simulation of a hot melt into the contraction expansion geometry with a cold wall relative to the melt and the isothermal simulation in chapter 3. The isothermal simulation was performed at a constant temperature of 398.15 K. This simulation does not include latent heat of crystallisation, heating due to flow, WLF shifts or any nucleation barrier considerations with regards to the temperature.

The overall simulation time is 10 seconds for both investigations. We will keep the pressure drop constant between the isothermal and the cold wall simulations. The injected melt has an internal temperature of 398.15 K, as in the isothermal simulation, but the wall is held at a lower temperature of 378.15 K. With this setup we expect to see enhanced nucleation and formation of crystals in comparison to the isothermal case due to the increased spherulitic growth rate, G_s, at 378.15 K when compared to 398.15 K. The profile of G_s shown in figure 4.5 gives $G_s = 70.92$ μm/s at 378.15 K compared to $G_s = 0.3$ μm/s at 398.15 K. We choose the growth rate scale as $G_{s,0} = 70$ μm/s since the majority of crystal nucleation occurs at the wall, so a growth rate similar to that at the cooler temperature of the wall is appropriate. With the shear rate being the largest in the contraction region, as noted in section 4.4.1 we expect the thermal boundary layer thickness to be smaller than the cross-section so that the cooler temperature is confined to the region near the wall.

Table 4.5 shows the inlet pressure along with the resulting flow rates between the two runs. The difference in flow rate between these runs can be attributed to the increased viscosity of the material at the wall due to the drop in temperature. Therefore a larger inlet pressure would be required in the cold wall simulation to keep the flow rates constant between the runs.

	Inlet Pressure (MPa)	Flow Rate (mm/s)
Isothermal	10.5	1.68
Cold Wall	10.5	1.50

Table 4.5: Table listing the inlet pressure and the resulting flow rate of the isothermal simulation and the cold wall simulation.

4.4.2.1 Polymer Configuration

Figure 4.18 shows the temperature profile for the contraction geometry when the wall is cold relative to the melt. We can see in areas above and below the contraction the boundary layer for the cold temperature is larger when compared to the boundary layer in the contraction region. This is expected since, as shown in section 4.4.1, the boundary thickness decreases with increasing shear rate.

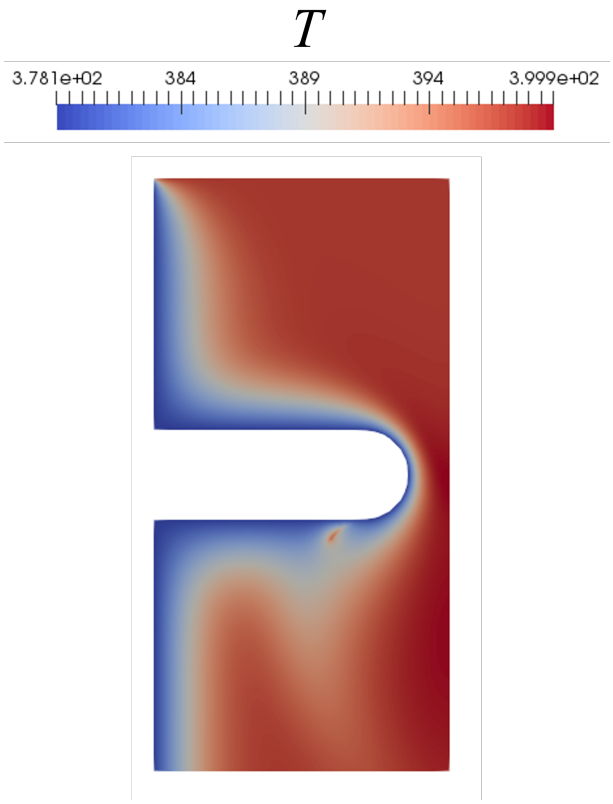


Figure 4.18: Figure showing the temperature distribution for the cold wall simulation.

Figure 4.19 shows how the magnitude of the z-component of the velocity gradient and the velocity compare between the isothermal and the cold wall cases across the centre of the contraction. We can see that the velocity gradient for the cold wall investigation is

slightly smaller at the wall compared to the isothermal case but the velocity gradients become comparable at the centre. This is a consequence of the cold wall increasing the viscosity of the fluid near the wall owing to the reduction in temperature as the shear stress is effectively fixed between the two simulations. This is also evident in the profile of the magnitude of the z-component of the velocity being smaller in the cold wall case.

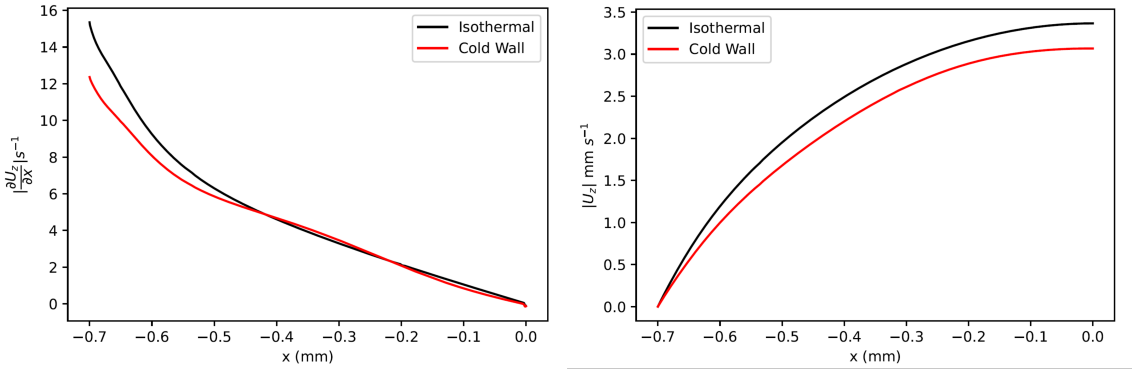


Figure 4.19: Line plots showing the z-component of the velocity gradient (left) and the z-component of the velocity (right) develop from the wall of the contraction to the centre of the contraction at the end of the simulation at $t = 10$ s. The x-axis represents the distance from the centre of the contraction where the centre is set at $x = 0$.

We can then look at how this velocity profile affects the polymer configuration in the two different simulations. Figure 4.20 shows how λ_4 and Δf_4 develop across the contraction. We can see that for both simulations λ_4 and Δf_4 are smaller at the wall in the cold wall case and are then comparable in the centre of the contraction. The similarity to the velocity gradient suggests that this is the direct result of the decrease in the shearing rates at the wall in the cold wall case compared to the isothermal case.

However, we also need to consider that the relaxation times are slower at the cooler temperature at the wall. The WLF shift factor, $a_T = 0.17$ at 378.15 K, so our relaxation times in table 4.4 will change by this factor. We can calculate the Weissenberg number for our relaxation times at 378.15 K using the flow rate shown in figure 4.19.

Chain Species	$\tau_{d,i}$	$\tau_{d,i}\dot{\gamma}$	$\tau_{s,i}$	$\tau_{s,i}\dot{\gamma}$
1	1.04 s	12.85	0.45 s	5.56
2	10.5 s	129.78	0.45 s	5.56
3	103.71 s	1281.86	0.45 s	5.56
4	1693.91 s	20936.73	22.58 s	279.09

Table 4.6: Table showing the relaxation times at 378.15 K along with the corresponding Weissenberg numbers.

At the wall of the contraction, the shear rate is $\dot{\gamma} = 12.36$. Table 4.6 shows the

relaxation times of the polymer melt at 378.15 K along with the corresponding Weissenberg numbers. For all the relaxation times the Weissenberg number is greater than 1, which shows that the flow rate is fast compared to the relaxation rates. Indeed the values are larger than in the corresponding values in tables 3.9 and 3.10 in chapter 3. However, the value of λ_4 is not determined solely by the local Weissenberg number at the point, but by the accumulated strain. Indeed the large value of $\tau_{s,i}\dot{\gamma}$ suggests that relaxation is insignificant at this point in the flow.

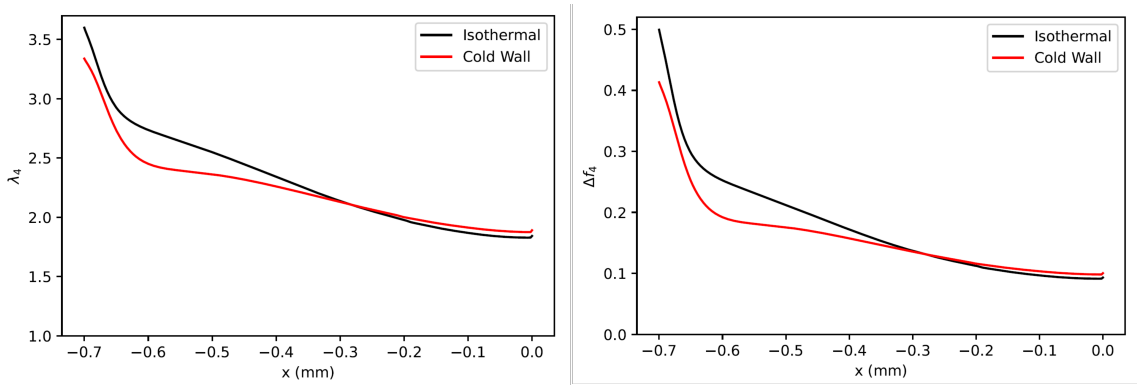
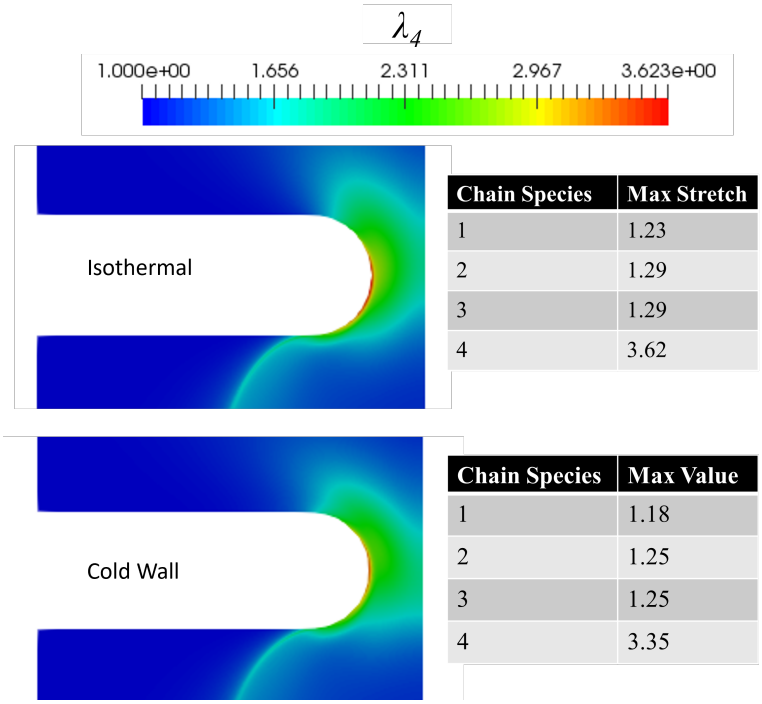


Figure 4.20: Figure showing how λ_4 (left) and Δf_4 (right) develop across the contraction. The results are taken at the end of the simulation at $t = 10$ s. The x-axis represents the distance from the centre of the contraction where the centre is set at $x = 0$.

We then look at λ_4 at the end of the simulation over the entire contraction region in figure 4.21. The majority of the activity occurs in the contraction, with the overall profile of λ_4 being very similar between the two runs. The maximum values of λ_i for the other chain species are given in tables in figure 4.21. The characteristic 'fang' shape is evident in both cases as is consistent with previous results given in this thesis.

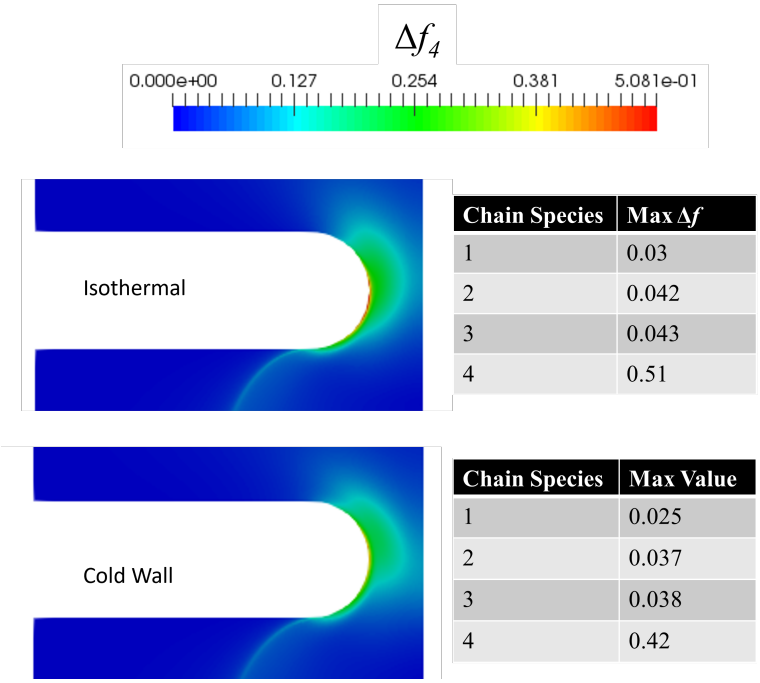


Chain Species	Max Stretch
1	1.23
2	1.29
3	1.29
4	3.62

Chain Species	Max Value
1	1.18
2	1.25
3	1.25
4	3.35

Figure 4.21: Figure showing λ_4 at the end of the simulation at $t = 10$ s. The tables underneath give the maximum value of λ_i for the other chain species.

Similarly, figure 4.22 shows Δf_4 at the end of the simulation at $t = 10$ s. The main activity is in the contraction region, where the shearing rates are the largest. The overall profile of Δf_4 looks very similar in both runs, with there being a slightly larger amount of activity in the isothermal case.



Chain Species	Max Δf
1	0.03
2	0.042
3	0.043
4	0.51

Chain Species	Max Value
1	0.025
2	0.037
3	0.038
4	0.42

Figure 4.22: Figure showing Δf_4 at the end of the simulation at $t = 10$ s. The tables underneath give the maximum value of Δf_i for the other chain species.

4.4.2.2 Crystallisation Dynamics

Now that the polymer configuration has been established, we can then look at how this affects the crystallisation dynamics in each simulation. The line plot in figure 4.23 shows how u_N develops across the contraction at $t = 10$ s. We can see that u_N is largest at the wall and reduces to zero in the centre of the contraction. The cold wall case has an overall smaller value of u_N at the wall and has a steeper drop towards zero, but then reaches zero at the same place as the isothermal case.

We can see that the differences in the profile of u_N between the isothermal and the cold wall investigations seems small in figure 4.23, but u_N is a logarithmic quantity which will highlight differences in small values. This shows that the differences in Δf_4 shown in figure 4.20 are somewhat balanced by the lower temperature at the wall.

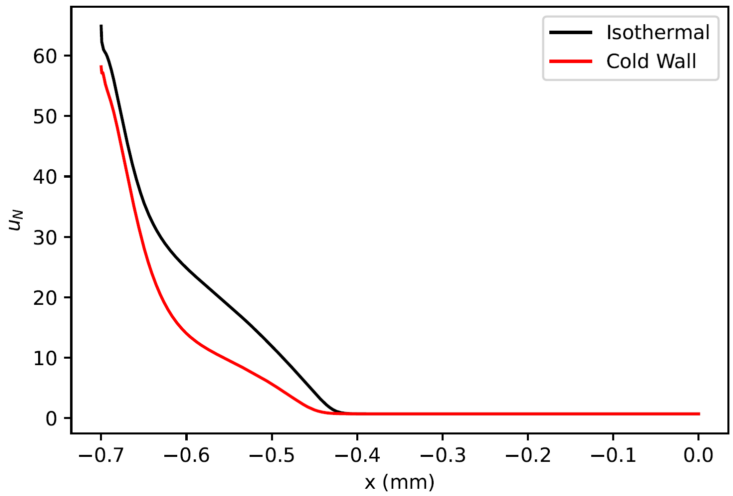


Figure 4.23: Line plot that shows the comparison of u_N for the isothermal and the cold wall simulations at $t = 10$ s. This plot shows how u_N changes from the wall to the centre of the contraction region. The x-axis represents the distance from the centre of the contraction, where the centre is set at $x = 0$.

Figure 4.24 compares how u_N has developed for the isothermal investigation and the cold wall investigation at $t = 10$ s over the entire contraction region. In both cases u_N developed into a 'fang' shape, as crystals nucleated in the contraction are swept downstream. Although the overall pattern is similar, we can see that the 'fang' in the cold wall case has started further up the wall in the contraction compared to the isothermal case. Although the cold wall simulation has a slightly lower nucleation rate at the wall the cold wall increases the viscosity of the polymer at the wall, and so the residence time of the melt is larger at the wall compared to the isothermal case. This means the fluid spends longer in the area of highest shear, which allows more nuclei to form at an earlier stage.

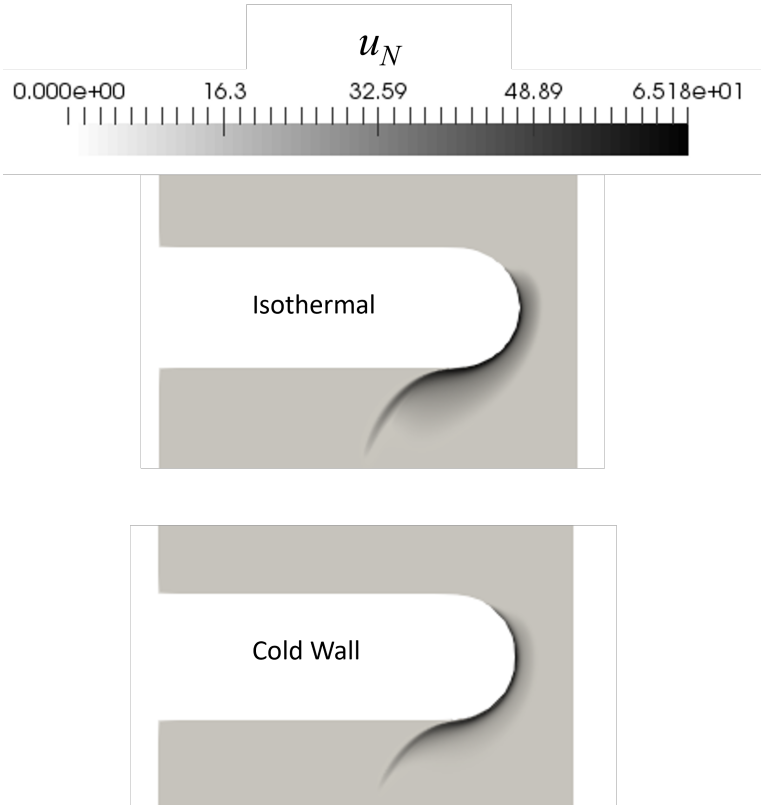


Figure 4.24: Figure showing how u_N changes for the isothermal case and the cold wall case at $t = 10$ s.

We then move on to see how this affects the logarithmic volume fraction of crystals that is not corrected for impingement, u_V. Figure 4.25 gives a line plot of how u_V develops across the contraction at $t = 10$ s. We can see that u_V is generally similar between the two simulations with the cold wall case overall showing a smaller value for u_V. There is a sharper drop towards zero in the cold wall case due to the smaller extension rate away from the wall because of the smaller flow rate in this simulation.

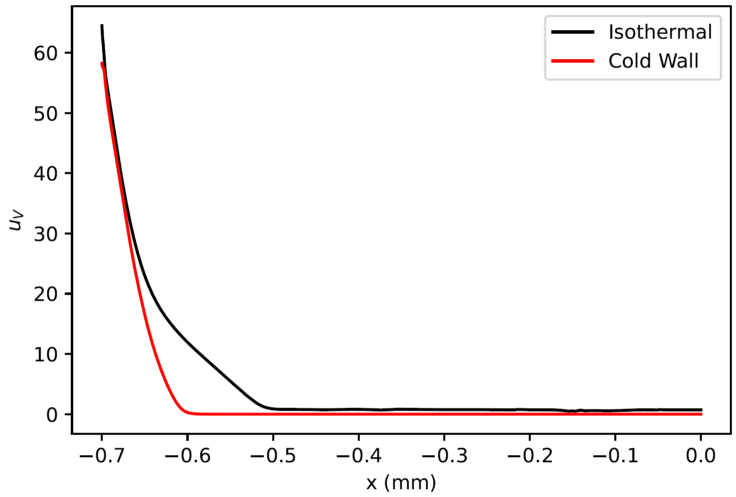


Figure 4.25: Line plot that shows a comparison of u_V for the cold wall and the isothermal investigations at $t = 10$ s. This plot shows the change in u_V from the wall to the centre of the contraction region. The x-axis represents the distance from the centre of the contraction, with the centre being set at $x = 0$.

Figure 4.26 shows how u_V developed in the contraction region at $t = 10$ s. As was the case for u_N, the cold wall case has an overall smaller value for u_V, but the formation of crystals occurs earlier in the contraction region compared to the isothermal case.

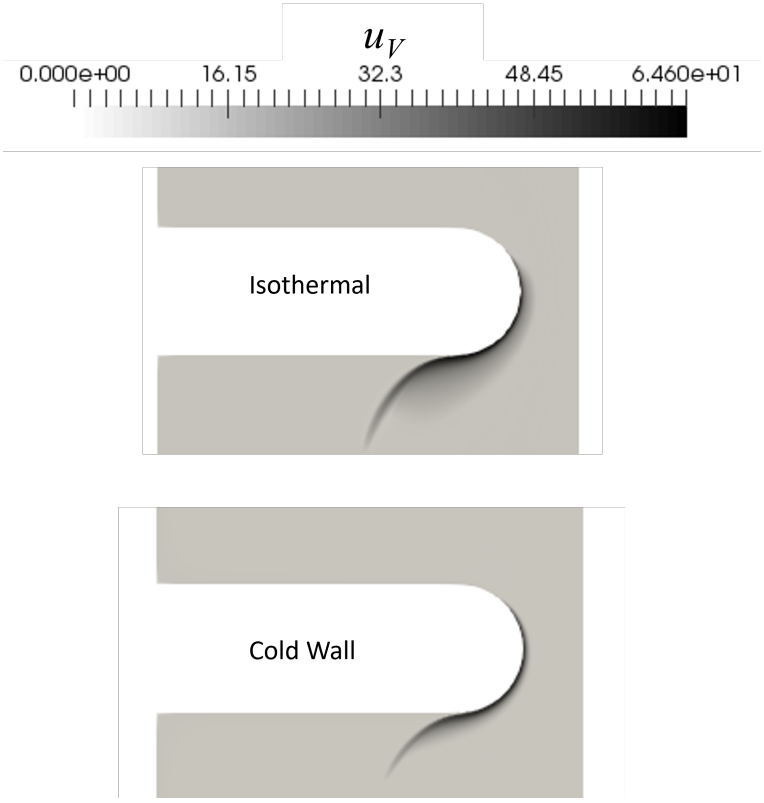


Figure 4.26: Figure showing how u_V changes for the isothermal case and the cold wall case at $t = 10$ s.

Lastly we can look at the volume fraction of crystals corrected for impingement, ξ_G, as calculated by the Avrami equation. Figure 4.27 shows how ξ_G developed for the cold wall and the isothermal simulations. We can see the characteristic 'fang' shape in both cases. The main difference is that the cold wall has crystallisation appearing higher up the contraction region and the crystallisation profile is thicker at the wall. We can see that there is a cumulative effect of temperature on the crystallisation profile in the contraction region. Although the polymer chain species are less oriented when the wall is cold, there is a compensation from the increased residence time at the wall, coupled with the increased spherulitic growth rate at the cooler temperature. We can see from the temperature field in figure 4.27 that the temperature remains low downstream, and as a result the crystallisation profile is thicker throughout for the cold wall case. We can also see that crystallisation is advected further downstream in the isothermal case. This can be explained by the higher flow rate due to the lower viscosity of the melt in the isothermal case, so the crystals are advected further downstream. Finally we can see there is a small area of high temperature downstream from the contraction region. Since this is an area of high crystallisation, we can attribute this to latent heat of crystallisation.

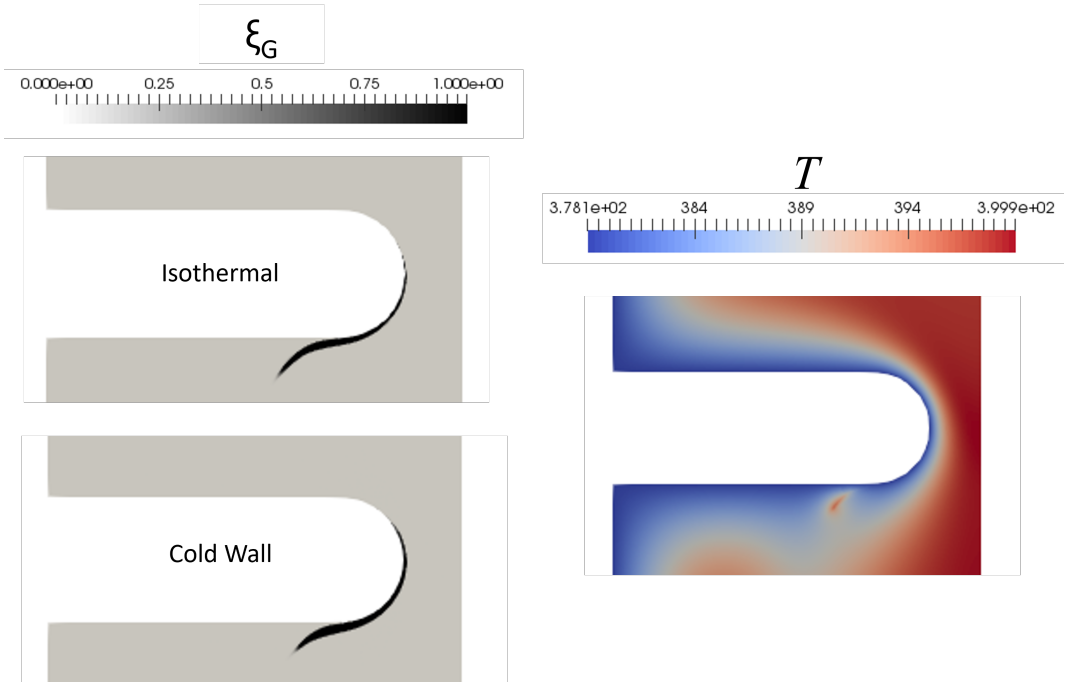


Figure 4.27: Figure showing how ξ_G develops for the isothermal and the cold wall case when $t = 10$ s. The temperature field in the contraction region is shown alongside.

We have successfully combined temperature dependence and polydispersity into our investigation of FIC and examined the effect of having a cold wall relative to the melt in a complex flow setting. We observed that the presence of the cold wall gives a larger amount of experimentally visible crystallisation in the polymer melt with the polymer chain species being less oriented when compared to the isothermal case. This shows how

vital a role temperature can play in polymer FIC.

4.4.3 Hot Wall

Now that we have successfully simulated a polydisperse polymer in a complex flow with a cold wall relative to the melt, we move onto the case of the wall of the contraction region being hot compared to the melt. In this scenario, we are attempting to see if we can suppress the crystallisation at the wall via an increase in temperature. This would move the visible crystallisation into the bulk. The combination of the larger energy barrier, faster relaxation times and higher viscosity due to the higher temperature would be the drivers to this phenomenon.

We will keep the polymer blend the same as in the previous section with 5 species with one species acting as a solvent. The overall simulation time is 8 seconds. The wall has a temperature of 410 K with the melt having a temperature of 360 K. This gives $G_s = 7.92 \times 10^{-13}$ μm/s at 410 K so that there is effectively no crystal growth at this temperature and $G_s = 188.67$ μm/s at 360 K. The growth rate scale in the Schneider rate equations is chosen as $G_{s,0} = 180$ μm/s since we expect crystallisation to occur away from the wall, so a typical growth rate closer to the cooler temperature of the bulk is appropriate. With the overall simulation time being different between the hot and the cold wall runs, these results are not direct comparisons, but are meant to demonstrate how a different wall temperature can affect the distribution of crystals.

Since the contraction expansion flow is a shear dominant flow, we need to increase the flow rate in the simulation compared to the cold wall simulation to see stronger extensional flow in the simulation. This will increase the stretch and orientation of the polymer chains in the melt away from the wall.

Table 4.7 shows the inlet pressure along with the resulting flow rate from the hot and cold wall simulations.

	Inlet Pressure (MPa)	Flow Rate (mm/s)
Hot Wall	16	2.34
Cold Wall	10.5	1.50

Table 4.7: Table listing the inlet pressure and the resulting flow rate of the hot and the cold wall simulations.

4.4.3.1 Observable Crystallisation

We will first show the overall profile of the observable crystallisation present in the hot wall simulation and then analyse reasons for this afterwards. Figure 4.28 shows how the observable crystallisation, ξ_G develops across the contraction region for the hot and the cold wall cases. Here it can be seen that the hot wall has suppressed crystallisation at

the wall and crystallisation is instead confined to a region away from the wall. This demonstrates the significant effect that temperature has on the crystallisation dynamics at play in this complex flow setting.

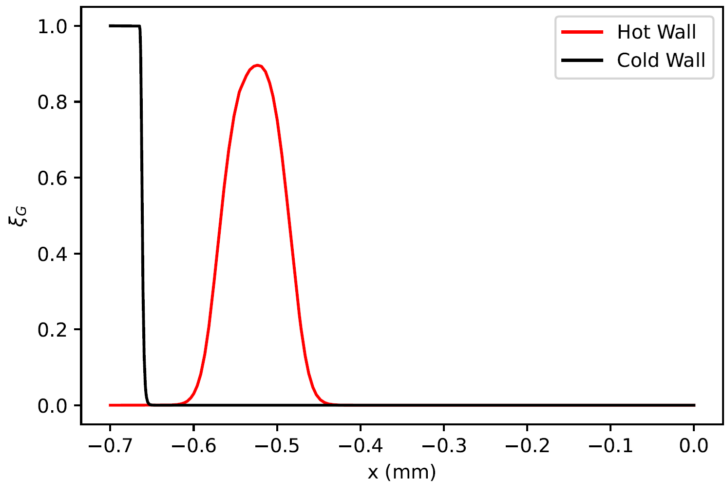


Figure 4.28: Line plot showing ξ_G at the end of the hot wall simulation at $t = 8$ s and at $t = 10$ s for the cold wall simulation. This plot shows how ξ_G changes from the wall to the centre of the contraction region. The x-axis represents the distance from the centre of the contraction, where the centre is set at $x = 0$.

Figure 4.29 shows the profile of ξ_G in the contraction for both the hot and the cold wall cases. You can clearly see that the hot wall case has observable crystallisation in the bulk away from the wall and this is being advected downstream from the contraction.

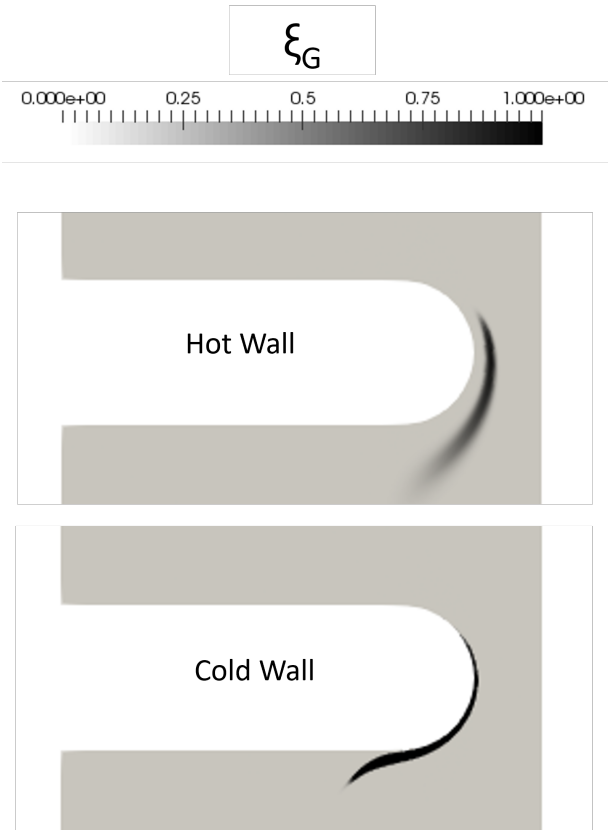


Figure 4.29: Figure showing the profile of ξ_G for the hot wall case (at $t = 8$ s) and the cold wall case (at $t = 10$ s).

4.4.3.2 Polymer Configuration

Now that we have seen the final crystal profile differences between the hot and the cold wall simulations, we shall investigate the reasons for these differences.

Figure 4.30 shows the temperature field for the contraction geometry when the wall is hot compared to the melt. Similarly to the cold wall case, the boundary layer for the warmer temperature becomes smaller in the contraction due to the larger shear rate in the contraction region.

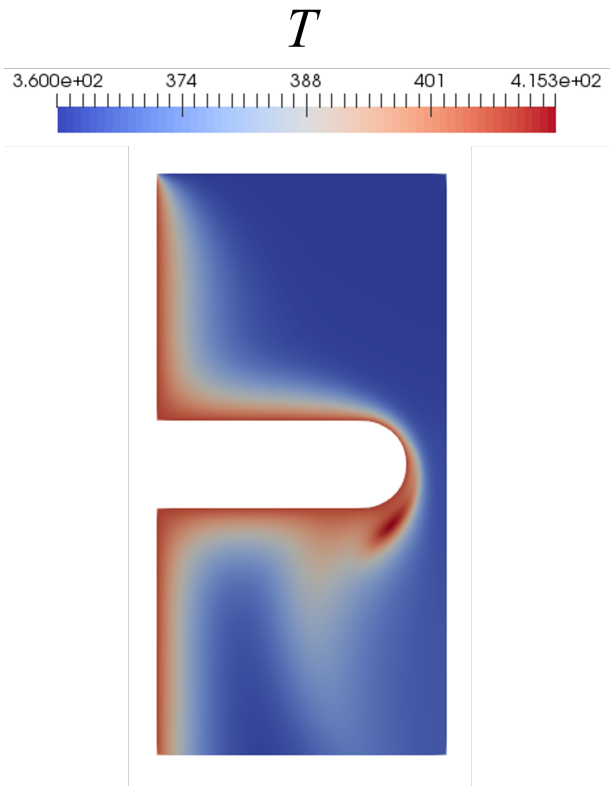


Figure 4.30: Figure showing the temperature profile for the hot wall simulation.

Figure 4.31 shows how the magnitude of the z-component of the velocity gradient and the z-component of the velocity develop across the contraction region for the hot wall simulation. When compared to figure 4.19 it can be seen that the velocity has a flatter distribution in which the velocity is almost constant over the inner half. In the hot wall case both the temperature and shear thinning act to increase the shear rates near the wall in comparison to the cold wall case where they compete against one another.

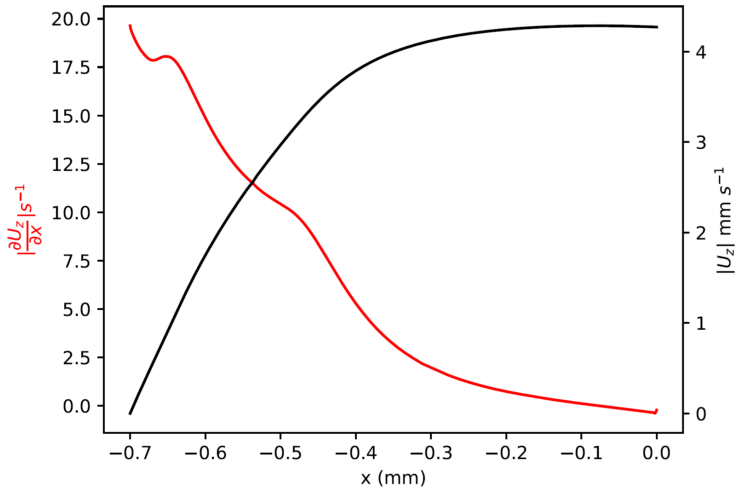


Figure 4.31: Line plot showing how the z-component of the velocity gradient and the z-component of the velocity develop from the wall of the contraction to the centre of the contraction at the end of the simulation at $t = 8$ s. The x-axis represents the distance from the centre of the contraction where the centre is set at $x = 0$.

Figure 4.32 shows how λ_4 and Δf_4 develop across the contraction. We can see that the largest degree of stretch and orientation of the chain species occur at the wall of the contraction owing to this being a shear dominant flow. The hot wall simulation has a larger overall stretch and orientation due to the higher flow rate imposed in this case. We also observe that the hot wall simulation has a 'shoulder' region between 0.4 and 0.6 mm from the centre line. The reason for this is that as you move away from the wall, the shear rate decreases whilst the extension rate increases. Near the wall we have shear dominated flow, and so the stretch and orientation fall as you move away from the wall up until a point where the extension rate increases so that it works alongside the shear rate to stretch and orient the polymer chains, this causes the profile of λ_4 and Δf_4 to level off and form the 'shoulder'. Then as you move further away from the wall the shear rate continues to decrease and the extension rate does not increase sufficiently to offset this, so λ_4 and Δf_4 decrease to a level comparable to the cold wall simulation at the centre of the contraction.

The 'shoulder' appearing in the plots in figure 4.32 is in the same place as the peak in ξ_G observed in figure 4.28. This is a result of a combination of the increased stretch and orientation further away from the wall, along with the spherulitic growth rate of crystals being negligible at the wall.

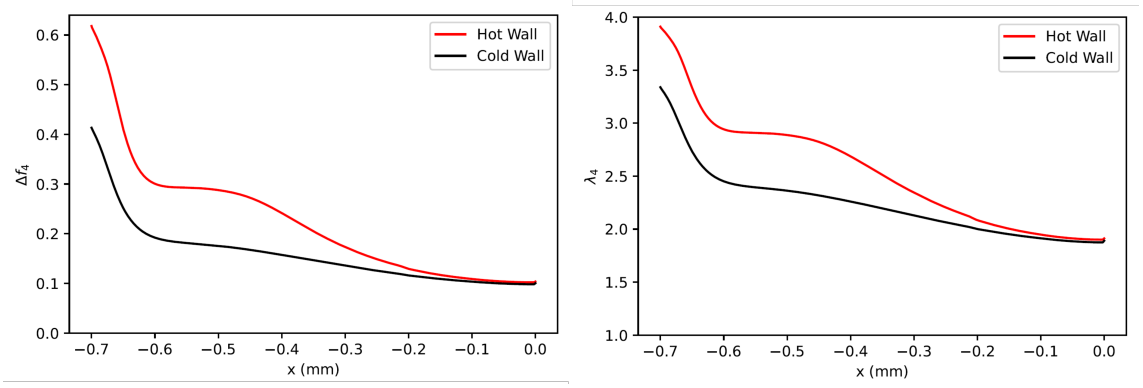
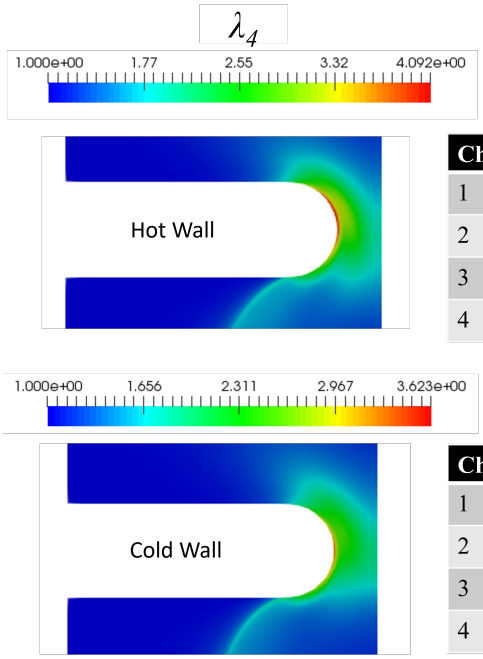


Figure 4.32: Line plot showing how λ_4 and Δf_4 develop from the wall of the contraction to the centre of the contraction at the end of the simulation at $t = 8$ s. The x-axis represents the distance from the centre of the contraction where the centre is set at $x = 0$.

When looking at the overall profile of λ_4 in the contraction in figure 4.33 we can see that the overall profile looks similar between the two cases. The tables in figure 4.33 show the maximum values of λ_i for the other chain species which occur in the contraction.

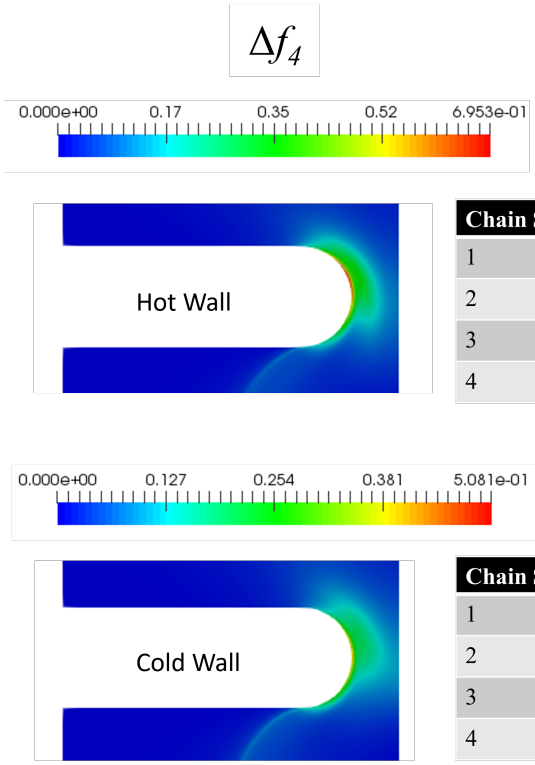


Chain Species	Max Value
1	1.34
2	1.40
3	1.40
4	4.09

Chain Species	Max Value
1	1.18
2	1.25
3	1.25
4	3.35

Figure 4.33: Figure showing λ_4 at the end of the hot wall simulation at $t = 8$ s and at $t = 10$ s for the cold wall simulation. The tables give the maximum value of λ_i for the other chain species.

Similarly for figure 4.34 we can see that the overall profile for Δf_4 in the contraction region is similar between the hot and the cold wall simulations with the hot wall simulation showing more orientation due to the higher flow rate imposed.



Chain Species	Max Value
1	0.041
2	0.052
3	0.053
4	0.70

Chain Species	Max Value
1	0.025
2	0.037
3	0.038
4	0.42

Figure 4.34: Figure showing Δf_4 at the end of the hot wall simulation at $t = 8$ s and at $t = 10$ s for the cold wall simulation. The tables give the maximum value of Δf_i for the other chain species.

4.4.3.3 Crystallisation Dynamics

Now that we have compared the polymer configuration in both the hot and the cold wall simulations, we now compare how this affects crystallisation dynamics. Figure 4.35 shows how u_N develops across the contraction region for the hot wall case (at $t = 8$ s) and for the cold wall case (at $t = 10$ s). We can see that there is the 'shoulder' present in the profile of u_N for the hot wall case in the same place as the peak in ξ_G. The increased nucleation further away from the wall, as a result of the higher stretch and orientation, combined with the much larger spherulitic growth rate in the bulk compared to at the wall results in the observable crystallisation taking place away from the wall.

There seems to be a numerical instability forming at the wall in the hot wall simulation due to the high flow rate in this investigation. This can probably be rectified by further refinement of the mesh at the wall or by reducing the time step in the simulation. Since the observable crystallisation has moved away from the wall in the hot wall simulation, this instability does not affect the results for the volume of crystals since these are concentrated away from the wall.

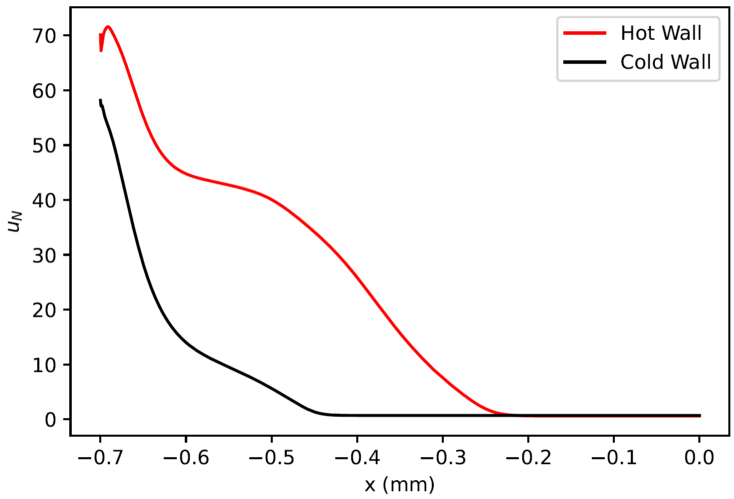


Figure 4.35: Line plot showing u_N at the end of the hot wall simulation at $t = 8$ s and at $t = 10$ s for the cold wall simulation. This plot shows how u_N changes from the wall to the centre of the contraction region. The x-axis represents the distance from the centre of the contraction, where the centre is set at $x = 0$.

We can see in figure 4.35 that the largest number of nuclei are at the wall in both the hot wall and cold wall simulations. However, this does not translate to crystallisation at the wall in the case of the hot wall investigation. Recall that $u_N = \log(1 + N/N_c)$ where $N_c = \dot{N}_{het} t_{char}$. In our simulations, $\dot{N}_{het} = 0.01$ and $t_{char} = 10$ which means that $u_N \simeq 70$ at the wall translates to $N \simeq 2.52 \times 10^{29}$ at the wall. However, the spherulitic growth rate $G_s = 7.92 \times 10^{-13}$ μms^{-1} at the wall. The volume of crystals in the Schneider rate equations grows according to the cube of the spherulitic growth rate, so this results in $V \simeq G_s^3 N \simeq 8 \times 10^{-8}$ which will lead to no discernible crystallisation at the wall. If we apply this same calculation to u_N at the point where observable crystallisation is first seen at approximately 0.1 mm away from the wall we then have $u_N \simeq 45$ and $G_s \simeq 0.1$ μms^{-1} according to figure 4.36. This equates to $V \simeq 3.49 \times 10^{15}$ which gives observable crystal growth in our simulation.

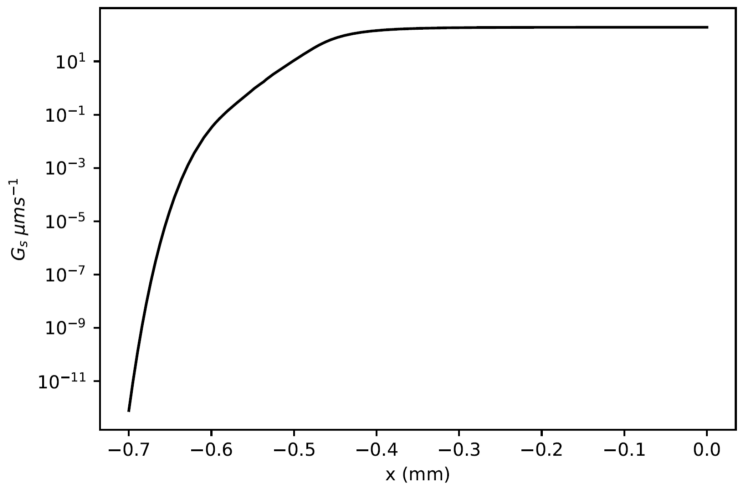


Figure 4.36: Line plot showing how G_s changes across the contraction on a logarithmic scale.

Figure 4.37 shows the overall profile of u_N in the contraction. We can see the characteristic 'fang' shape has developed for both cases, that corresponds to crystal nuclei being advected from the wall of the contraction. However for the hot wall case we also find nucleation appearing away from the wall that corresponds to nuclei advected from the 'shoulder' region.

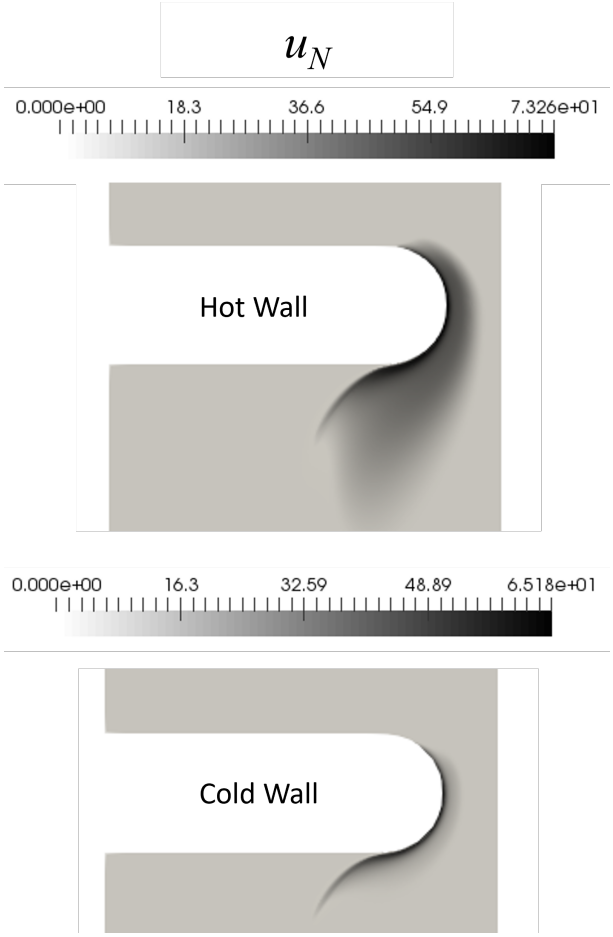


Figure 4.37: Figure showing how u_N changes for the hot wall case (at $t = 8$ s) and the cold wall case (at $t = 10$ s).

Figure 4.38 shows how u_V develops across the contraction region for the hot and the cold wall cases. Here is where we can see the main reason for the difference in crystal profile between the hot wall and the cold wall simulations. In the hot wall simulation, the spherulitic growth rate at the wall is very small, therefore, even though there are a large number of nuclei formed at the wall, these nuclei do not grow into observable crystals. It is only once we move away from the wall and the spherulitic growth rate becomes larger that the crystal nuclei can grow into observable crystals.

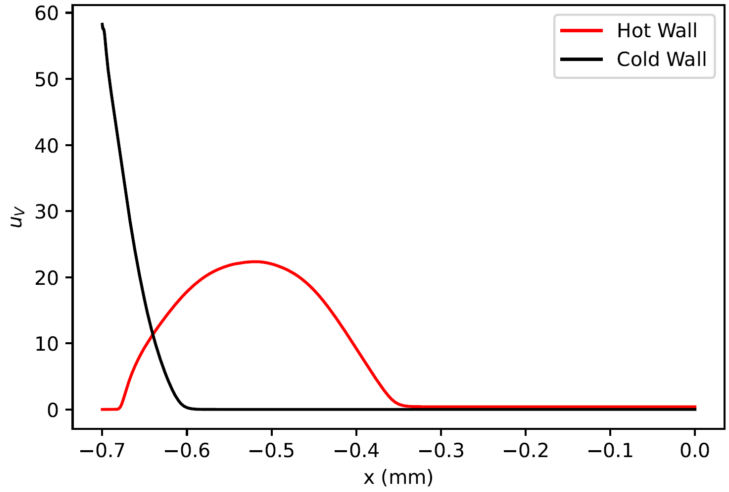


Figure 4.38: Line plot showing u_V at the end of the hot wall simulation at $t = 8$ s and at $t = 10$ s for the cold wall simulation. This plot shows how u_V changes from the wall to the centre of the contraction region. The x-axis represents the distance from the centre of the contraction, where the centre is set at $x = 0$.

Figure 4.39 shows the profile of u_V in the contraction region for the hot and the cold wall cases. We can see that the profile of u_V shows the 'fang' shape for both cases. We can see that in the hot wall simulation, u_V has moved slightly away from the wall, owing to the changes in spherulitic growth rate as you move away from the wall in this investigation.

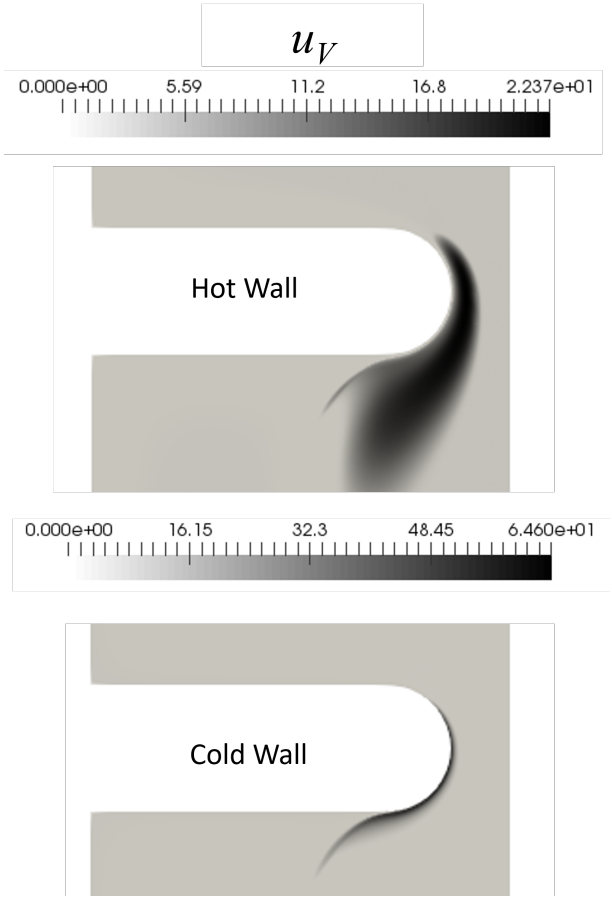


Figure 4.39: Figure showing how u_V changes for the hot wall case (at $t = 8$ s) and the cold wall case (at $t = 10$ s).

4.5 Conclusion

Overall we have demonstrated the significant role that temperature has both on the polymer flow configuration and the crystallisation dynamics that occur in the contraction expansion geometry. We have successfully simulated the shear flow of a polymer melt with an imposed temperature history. In experiments where the melt is subjected to a shear pulse followed by a waiting time and then a reduction in temperature. Our simulations show that the crystals are nucleated by the shear pulse, but do not begin to grow until the temperature is sufficiently reduced for the crystal growth to proceed.

This distinction between flow induced nucleation and crystal growth can also be manipulated in a complex flow of a polymer melt through a contraction expansion geometry, by varying the temperature of the wall relative to the melt. We found that the presence of the cold wall moved the crystallisation profile upwards in the contraction region owing to the higher viscosity of the polymer due to the cold wall. The spherulitic growth rate is larger at the wall at the cooler temperature, so the crystallisation profile was thicker in this simulation when compared to the isothermal case.

A more dramatic change in the crystal profile can be achieved with the flow of a cool

melt with a heated wall in the contraction expansion geometry. Here we could see that the hot wall suppressed crystal growth at the wall and moved crystallisation away from the wall into the bulk, despite the largest rate of nucleation being at the wall.

These examples show how temperature can be used to manipulate both the temporal and spatial distribution of crystallisation in combination with flow induced nucleation.

Chapter 5

Conclusion

5.1 Summary

In this thesis we have studied the flow induced crystallisation behaviour of a polydisperse polymer melt in a contraction expansion geometry using the recently published RDP [1] and polySTRAND [2] models. The main objective of this thesis was to establish a computationally tractable method of simulating flow induced crystallisation in a polydisperse polymer melt in complex flow. We successfully replicated experimental findings [5] of flow induced crystallisation and extended our model to include temperature effects.

Chapter 2 presented the implementation of the RDP model in openFOAM along with the validation of the RDP model against existing literature results. An idealised bimodal polymer blend was then simulated to demonstrate the viability of the RDP compilation in openFOAM. The implementation and validation of the polySTRAND model were also given. We describe our method for pre-calculating the local nucleation rate and incorporating this into our openFOAM simulations in an efficient way. We showed how our contraction expansion mesh was created in line with the experimental work conducted by Scelsi et al [5]. We then proceeded to simulate the 2-dimensional flow of an idealised bimodal isothermal polymer blend through the contraction expansion geometry.

Chapter 3 described our RDP fit to HDB6 using the modes given by Scelsi et al [5]. We showed how the contributions of the RDP model map onto the multi-mode Maxwell model used in their investigation. We then used the RepTate [74] software to verify that our RDP modes would predict the rheology of HDB6 accurately. We then presented a logarithmic version of the Schneider rate equations that were used to handle the large variations in the nucleation rate and to improve computational stability. We then illustrated how the different chain species contribute to the reduction in the energy barrier in a non-trivial way. Next we showed the interpolation function that is used to calculate the local nucleation rate for multiple chain species. We then described the implementation of the multimode RDP model in openFOAM. We show how the mesh was updated and conduct a mesh convergence study to demonstrate the accuracy of our results. We then investigated the effect of flow rate on the overall crystallisation profile in the geometry.

Chapter 4 extended our framework to include temperature effects on FIC and the rheology of the polymer in our simulations. We successfully incorporated the effect temperature has on the growth rate of crystals, along with a WLF shift factor that is consistent with the time temperature shift performed in the previous chapter. We modified our interpolation function for the nucleation rate to take into account changing temperature and the effect this has on the energy barrier to crystallisation. We then investigated the effect of cooling the polymer melt after a shear pulse in a channel geometry taking inspiration from an experimental investigation in the literature [6]. We wanted to see if changing the wait time before cooling had any effect on the overall crystallisation profile observed and we found that the effect was not significant. We then investigated the effect of having a cold wall relative to the melt in our contraction expansion geometry and analysed the effects this had. We found that the crystallisation profile did change: crystals appeared further up the contraction region and the profile of crystalised material became thicker in the contraction when the wall was cold relative to the melt. We showed results for the case of a hot wall relative to the melt and analysed the effects this had. We showed that the observable crystallisation could be suppressed at the wall and moved towards the bulk owing to the presence of the hot wall. This section showed that temperature and flow affects the nucleation rate within the melt, and temperature also affects crystal growth from nuclei. Both nucleation and growth are needed to form a significant crystal volume fraction so the outcome is that the region of greatest crystal fraction is not necessarily where the nucleation density is highest. You can control these effects by manipulating the temperature profile.

5.2 Achievements

The main achievements of this thesis are as follows

- We successfully implemented the RDP model in the RheoTool toolkit alongside the polySTRAND model to describe FIC in a polydisperse, non-isothermal polymer. We compared our results to published experimental results and could see qualitative agreement.
- We demonstrated the importance of polydispersity in FIC modelling. Although the slowest relaxing chain species dominates the acceleration effect, the other faster relaxing chain species contribute in a non-trivial way owing to their larger volume fraction in the melt.
- We showed the vital role that temperature plays when modelling FIC in complex flow. This shows the careful considerations that are needed to be made during polymer processing to anticipate the potential crystallisation that could occur due to the temperature profile.

5.3 Future Work

This section identifies potential open problems in the current work and suggests ways in which this research could be developed in the future.

Although in chapter 3 we compared our simulations to that of Scelsi et al we were only able to make a qualitative comparison. The polymer melt in that study is lightly branched rather than linear and many of the crystallisation parameters are not known and had to be inferred from other literature. Currently there is a lack of experimental studies on flow induced crystallisation in more complex flow geometries than simple shear or channel flows. In particular it would be useful to have experiments looking at flows that are more extension dominated, such as the cross-slot geometry.

The workflow outlined in this thesis allows for easy application to many different processing conditions. Creating new meshes in openFOAM is a straightforward process and would give people involved in polymer processing the opportunity to investigate multiple different processing conditions tailored to their specific needs. Notably there is scope to apply this workflow to 3-D printing to better understand the crystal formation in this setting. Another important class of flow geometries are those involving free-surfaces including applications such as fibre-spinning and film blowing. The simulations by Van Berlo et al. [41] investigated the FIC effects of a filament stretching flow. Simulating such flows would require extending the openFOAM simulation to include a phase field variable as in `rheoInterFoam` [4].

For the contraction flow we encountered numerical instabilities arising from the high localisation of regions of enhanced nucleation and the huge range of values in the nucleation rate. While the logarithmic formulation provided a partial resolution, the simulations require small time steps and careful mesh construction. There is a need for a more robust method for solving the Schneider rate equations that does not introduce artificial diffusion. Another reason for needing small time steps is the need to resolve the stretch relaxation of the fastest modes, even though these modes do not become significantly stretched in most of the flow domain. For the Rolie-Poly model there exists a non-stretching approximation for modes where the stretch relaxation rates are fast compared to the flow velocity gradients. A similar approximation to the RDP model might enable larger timesteps to be used.

In our workflow, we pre-calculated predictions for the nucleation rate from the polySTRAND model and fit an interpolation function to more efficiently calculate the nucleation rate of crystals in our complex flow setting. This was possible since the molecular weight distribution of our polymer remained constant throughout each of our investigations. However, if there was interest in reaction flows or the degradation of polymer chains due to the flow the molecular weight distribution would not be constant in this case. Here you would need to embed the full polySTRAND model into openFOAM to account for non-constant molecular weight distribution. This would likely impact computational efficiency, but it would allow for a more elegant solution to our workflow as all the models

are embedded in a single piece of software.

Finally, while our model predicts the rate of nucleation and growth of crystallites there is currently no change to the rheology implemented as a function of the degree of crystallinity. Currently there is a lack of constitutive models that describe semi-crystalline polymer melts. The investigation by Roozemond et al. [95] considered the effects that shish kebabs had on the rheology of iPP, but did so using an empirical approximation. We also only considered spherulites and did not attempt to model shish-kebab formation.

Hence despite the advances made in this thesis, there remain multiple open questions for future investigation, the most critical of which would be incorporating the change in rheology as a function of the degree of crystallinity. This would bring the simulations closer to real flows and would therefore give a more accurate prediction of crystal formation in complex flows.

Appendix A

Appendix

A.1 New Finite Extensibility Calculation

In the published polySTRAND model [2], the bond orientation for species i was calculated as

$$P_{2,K,i} = \frac{1}{N_e}(\Lambda^{(A)}_{i,max} - 1) \tag{A.1}$$

where $\Lambda^{(A)}_{i,max}$ is the largest eigenvalue of the orientation tensor $\underline{\underline{\mathbf{A}}}_i$. We found that with this definition, in strong flow the value for $P_{2,K,i}$ was growing above 1, which is not physical. In order to rectify this we have developed a new method for calculating $P_{2,K,i}$. We reconsider the freely jointed chain calculation that leads to the force for finite extensibility used in the Rolie-Double-Poly model and throughout the literature in polymer rheology [96]. Here we follow similar reasoning to the calculation by Li and Larson [97] which was developed for measuring bond orientation in bead-spring simulations of dilute polymers.

The fundamental model, at the level of a tube segment, is a chain subsection of N_e freely jointed bonds each of fixed length l. We non-dimensionalise the force f acting along the tube axis as $X = fl/k_BT$. Given that bonds make an angle θ with the tube axis, we can calculate, using standard statistical mechanics, $Y = \langle \cos\theta \rangle$ to be

$$Y = L(X) = \coth(X) - \frac{1}{X}, \tag{A.2}$$

which is the Langevin function. A similar treatment gives the average bond orientation with respect to the tube axis.

$$P_2(\cos\theta) = \frac{3\langle \cos^2\theta \rangle - 1}{2} = \frac{X^2 + 3 - 3X\coth X}{X^2}. \tag{A.3}$$

However, for $P_{2,K,i}$ we require the orientation of bonds relative to some direction in space (e.g. the principle direction of $\underline{\underline{\mathbf{A}}}_i$) and the tube orientations are themselves distributed statistically around this. Defining α to be the angle between a tube segment and the principle direction of $\underline{\underline{\mathbf{A}}}_i$ we can in principle obtain the average tube orientation, $P_2(\cos\alpha) = \frac{1}{2}\left(3\langle \cos^2\alpha \rangle - 1\right)$. Then, averaging all possible bond directions about the

tube axis consistent with the value of $P_2(\cos\theta)$, we find that the average bond orientation relative to the principle direction of $\underline{\underline{\mathbf{A}}}_i$ is

$$P_{2,K,i} = P_2(\cos\theta)P_2(\cos\alpha). \tag{A.4}$$

There remains the task of relating the above to the quantities available in the RDP model. In tube theory, the quantity $Y = \langle\cos\theta\rangle$ is related to the end-to-end length r of a tube segment as $r = N_e l Y$. But, also, r is obtained from the stretch, λ as $r = \sqrt{N_e}l\lambda$. Hence $Y = \lambda/\sqrt{N_e}$ and equation A.2 is usually rewritten in its inverted form in terms of the force

$$f = \frac{k_B T}{l} X = \frac{3k_B T}{a}\left[\frac{1}{3Y}L^{-1}(Y)\right]\lambda \tag{A.5}$$

where $a = \sqrt{N_e}l$ is the tube diameter. The term in square brackets is related to the inverse Langevin function and (given $Y = \lambda/\sqrt{N_e}$) is usually approximated by a finite extensibility function:

$$\left[\frac{1}{3Y}L^{-1}(Y)\right] \approx f_E(\lambda).$$

In our case we decided to use the Pade approximant to the inverse Langevin function [70],

$$f_E(\lambda_i) = \left(\frac{3-\lambda_i^2/N_e}{1-\lambda_i^2/N_e}\right)\left(\frac{1-1/N_e}{3-1/N_e}\right).$$

Since $X = L^{-1}(Y)$, we can approximate X for species i as

$$X_i = 3Y_i f_E(\lambda_i) = \frac{3\lambda_i}{\sqrt{N_e}} f_E(\lambda_i) = \frac{3\lambda_i}{\sqrt{N_e}}\left(\frac{3-\lambda_i^2/N_e}{1-\lambda_i^2/N_e}\right)\left(\frac{1-1/N_e}{3-1/N_e}\right), \tag{A.6}$$

where N_e is the number of steps per entanglement segment.

We obtain $P_2(\cos\alpha)$ from the conformation tensor $\underline{\underline{\mathbf{A}}}_i$ by defining the orientation tensor

$$\underline{\underline{\mathbf{S}}}_i = \frac{\underline{\underline{\mathbf{A}}}_i}{\mathrm{Tr}\underline{\underline{\mathbf{A}}}_i} = \frac{\underline{\underline{\mathbf{A}}}_i}{3\lambda_i^2}.$$

Then $\langle\cos^2\alpha\rangle$ is obtained as the largest eigenvalue of $\underline{\underline{\mathbf{S}}}_i$, so that for species i,

$$P_2(\cos\alpha) = \frac{1}{2}\left(3\langle\cos^2\alpha\rangle - 1\right) = \frac{1}{2}\left(\frac{\Lambda_{i,max}^{(A)}}{\lambda_i^2} - 1\right). \tag{A.7}$$

Combining all of the above, we calculate $P_{2,K,i}$ for species i as

$$P_{2,K,i} = \frac{1}{2}\left(\frac{X_i^2 + 3 - 3X_i\coth X_i}{X_i^2}\right)\left(\frac{\Lambda_{i,max}^{(A)}}{\lambda_i^2} - 1\right), \tag{A.8}$$

with X_i obtained from equation A.6. In the limit of strong orientation and strong stretch-

ing, $\Lambda_{i,max}^{(A)} \approx 3\lambda_i^2$ and X_i becomes large so that $P_{2,K,i}$ approaches a value of one, as required.

In the limit of small stretch, equation A.8 reduces to

$$P_{2,K,i} = \frac{3}{10N_e}(\Lambda_{i,max}^{(A)} - \lambda_i^2).$$

Comparing with equation A.1 we see that the form is similar, but the numerical prefactor is smaller, so that when using equation A.8 it becomes necessary to use a larger value of the proportionality constant Γ in $\Delta f_i = \Gamma P_{2,K,i}$ to obtain similar results as compared to the original polySTRAND formulation. This gives rise to the distinction between Γ and Γ_{FE} in the previous section.

References

[1] Victor AH Boudara, Joseph D Peterson, L Gary Leal, and Daniel J Read. Nonlinear rheology of polydisperse blends of entangled linear polymers: Rolie-double-poly models. Journal of Rheology, 63(1):71–91, 2019.

[2] Daniel J Read, Claire McIlroy, Chinmay Das, Oliver G Harlen, and Richard S Graham. Polystrand model of flow-induced nucleation in polymers. Physical Review Letters, 124(14):147802, 2020.

[3] Hrvoje Jasak, Aleksandar Jemcov, Zeljko Tukovic, et al. Openfoam: A c++ library for complex physics simulations. In International Workshop on Coupled Methods in Numerical Dynamics, volume 1000, pages 1–20. IUC Dubrovnik Croatia, 2007.

[4] F. Pimenta and M.A. Alves. rheotool. `https://github.com/fppimenta/rheoTool`, 2016.

[5] L Scelsi, MR Mackley, H Klein, PD Olmsted, RS Graham, OG Harlen, and TCB McLeish. Experimental observations and matching viscoelastic specific work predictions of flow-induced crystallization for molten polyethylene within two flow geometries. Journal of Rheology, 53(4):859–876, 2009.

[6] Lucia Fernandez-Ballester, Tim Gough, Florian Meneau, Wim Bras, Fernando Ania, Francisco Jose Balta-Calleja, and Julia A Kornfield. Simultaneous birefringence, small-and wide-angle x-ray scattering to detect precursors and characterize morphology development during flow-induced crystallization of polymers. Journal of Synchrotron Radiation, 15(2):185–190, 2008.

[7] Adila Aida Binti Azahar. Numerical Simulation for Polymer Blend using OpenFOAM. PhD thesis, University of Leeds, 2020.

[8] Victor Ange Henri Boudara. Supramolecular and entangled polymer materials: Rheological models. PhD thesis, University of Leeds, 2017.

[9] Peng Yi, C Rebecca Locker, and Gregory C Rutledge. Molecular dynamics simulation of homogeneous crystal nucleation in polyethylene. Macromolecules, 46(11):4723–4733, 2013.

[10] John D Hoffman, Lois J Frolen, Gaylon S Ross, and John I Lauritzen Jr. On the growth rate of spherulites and axialites from the melt in polyethylene fractions: regime i and regime ii crystallization. Journal of Research of the National Bureau of Standards. Section A, Physics and Chemistry, 79(6):671, 1975.

[11] Alexei E Likhtman and Richard S Graham. Simple constitutive equation for linear polymer melts derived from molecular theory: Rolie–poly equation. Journal of Non-Newtonian Fluid Mechanics, 114(1):1–12, 2003.

[12] Pierre-Giles de Gennes. Reptation of a polymer chain in the presence of fixed obstacles. The Journal of Chemical Physics, 55(2):572–579, 1971.

[13] J Des Cloizeaux. Double reptation vs. simple reptation in polymer melts. EPL (Europhysics Letters), 5(5):437, 1988.

[14] Jane S Richardson. The anatomy and taxonomy of protein structure. Advances in Protein Chemistry, 34:167–339, 1981.

[15] Alexander Ya Malkin and Avraam I Isayev. Rheology: concepts, methods, and applications. Elsevier, 2017.

[16] James Clerk Maxwell. Iv. on the dynamical theory of gases. Philosophical Transactions of the Royal Society of London, (157):45–88, 1867.

[17] RJ Poole. The deborah and weissenberg numbers. Rheol. Bull, 53(2):32–39, 2012.

[18] James G Oldroyd. On the formulation of rheological equations of state. Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences, 200(1063):523–541, 1950.

[19] RB Bird, PJ Dotson, and NL Johnson. Polymer solution rheology based on a finitely extensible bead—spring chain model. Journal of Non-Newtonian Fluid Mechanics, 7(2-3):213–235, 1980.

[20] Richard S Graham, Alexei E Likhtman, Tom CB McLeish, and Scott T Milner. Microscopic theory of linear, entangled polymer chains under rapid deformation including chain stretch and convective constraint release. Journal of Rheology, 47(5):1171–1200, 2003.

[21] TCB McLeish and RG Larson. Molecular constitutive equations for a class of branched polymers: The pom-pom polymer. Journal of Rheology, 42(1):81–110, 1998.

[22] Wilco MH Verbeeten, Gerrit WM Peters, and Frank PT Baaijens. Differential constitutive equations for polymer melts: The extended pom–pom model. Journal of Rheology, 45(4):823–843, 2001.

[23] Paul J Flory and Do Yeung Yoon. Molecular morphology in semicrystalline polymers. Nature, 272(5650):226–229, 1978.

[24] Richard S Graham. Molecular modelling of flow-induced crystallisation in polymers. Journal of Engineering Mathematics, 71:237–251, 2011.

[25] Richard S Graham. Understanding flow-induced crystallization in polymers: A perspective on the role of molecular simulations. Journal of Rheology, 63(1):203–214, 2019.

[26] Seong Hyuk Cho et al. Role of Tie Molecules in Ductility and Chain Deformation of Polyethylene. PhD thesis, Princeton, NJ: Princeton University, 2022.

[27] Gaetano Lamberti. Flow induced crystallisation of polymers. Chemical Society Reviews, 43(7):2240–2252, 2014.

[28] Jun Xu, Günter Reiter, and Rufina G Alamo. Concepts of nucleation in polymer crystallization. Crystals, 11(3):304, 2021.

[29] Junyu Li, Wei Li, He Cheng, Lina Zhang, Yang Li, and Charles C Han. Early stages of nucleation and growth in melt crystallized polyethylene. Polymer, 53(12):2315–2319, 2012.

[30] Bernhard Wunderlich. Macromolecular Physics V2, volume 2. Elsevier, 2012.

[31] Gaylon S Ross and Lois J Frolen. Homogeneous nucleation in polyethylene: molecular weight dependence. Journal of Research of the National Bureau of Standards. Section A, Physics and Chemistry, 79(6):701, 1975.

[32] JL Carvalho and K Dalnoki-Veress. Surface nucleation in the crystallisation of polyethylene droplets. The European Physical Journal E, 34:1–6, 2011.

[33] C Schick, R Androsch, and JWP Schmelzer. Homogeneous crystal nucleation in polymers. Journal of Physics: Condensed Matter, 29(45):453002, 2017.

[34] Richard S Graham. Modelling flow-induced crystallisation in polymers. Chemical Communications, 50(27):3531–3545, 2014.

[35] Zhen Wang, Zhe Ma, and Liangbin Li. Flow-induced crystallization of polymers: Molecular and thermodynamic considerations. Macromolecules, 49(5):1505–1517, 2016.

[36] Rajesh H Somani, Ling Yang, Lei Zhu, and Benjamin S Hsiao. Flow-induced shish-kebab precursor structures in entangled polymer melts. Polymer, 46(20):8587–8623, 2005.

[37] Shuichi Kimata, Takashi Sakurai, Yoshinobu Nozue, Tatsuya Kasahara, Noboru Yamaguchi, Takeshi Karino, Mitsuhiro Shibayama, and Julia A Kornfield. Molecular basis of the shish-kebab morphology in polymer crystallization. Science, 316(5827):1014–1017, 2007.

[38] I Dukovski and M Muthukumar. Langevin dynamics simulations of early stage shish-kebab crystallization of polymers in extensional flow. The Journal of Chemical Physics, 118(14):6648–6655, 2003.

[39] Min Nie, Rui Han, and Qi Wang. Formation and alignment of hybrid shish-kebab morphology with rich beta crystals in an isotactic polypropylene pipe. Industrial & Engineering Chemistry Research, 53(10):4142–4146, 2014.

[40] Michelle D'Haese, Florentin Langouche, and Peter Van Puyvelde. On the effect of particle size, shape, concentration, and aggregation on the flow-induced crystallization of polymers. Macromolecules, 46(9):3425–3434, 2013.

[41] Frank PA van Berlo, Ruth Cardinaels, Gerrit WM Peters, and Patrick D Anderson. A numerical study of extensional flow-induced crystallization in filament stretching rheometry. Polymer Crystallization, 4(1):e10154, 2021.

[42] Hans Zuidema, Gerrit WM Peters, and Han EH Meijer. Development and validation of a recoverable strain-based model for flow-induced crystallization of polymers. Macromolecular Theory and Simulations, 10(5):447–460, 2001.

[43] G Marrucci. Dynamics of entanglements: A nonlinear model consistent with the cox-merz rule. Journal of Non-Newtonian Fluid Mechanics, 62(2-3):279–289, 1996.

[44] Dietmar Auhl, Pierre Chambon, Tom CB McLeish, and Daniel J Read. Elongational flow of blends of long and short polymers: Effective stretch relaxation time. Physical Review Letters, 103(13):136001, 2009.

[45] Taisir Shahid, Qian Huang, Filip Oosterlinck, Christian Clasen, and Evelyne Van Ruymbeke. Dynamic dilution exponent in monodisperse entangled polymer solutions. Soft Matter, 13(1):269–282, 2017.

[46] DJ Read, K Jagannathan, SK Sukumaran, and D Auhl. A full-chain constitutive model for bidisperse blends of linear polymers. Journal of Rheology, 56(4):823–873, 2012.

[47] Servaas Vleeshouwers and Han EH Meijer. A rheological study of shear induced crystallization. Rheologica Acta, 35:391–399, 1996.

[48] Stefano Acierno, Blandina Palomba, Henning H Winter, and Nino Grizzuti. Effect of molecular weight on the flow-induced crystallization of isotactic poly (1-butene). Rheologica Acta, 42:243–250, 2003.

[49] Rudi JA Steenbakkers and Gerrit WM Peters. A stretch-based model for flow-enhanced nucleation of polymer melts. Journal of Rheology, 55(2):401–433, 2011.

[50] W Schneider, A Koppl, and J Berger. Non-isothermal crystallization of polymers. system of rate equations. Int. Polym. Process., 2(3):151–154, 1988.

[51] Melvin Avrami. Kinetics of phase change. ii transformation-time relations for random distribution of nuclei. The Journal of Chemical Physics, 8(2):212–224, 1940.

[52] Yue Mu, Guoqun Zhao, Anbiao Chen, and Xianghong Wu. Numerical investigation of the thermally and flow induced crystallization behavior of semi-crystalline polymers by using finite element–finite difference method. Computers & Chemical Engineering, 46:190–204, 2012.

[53] Kristjan Krebelj, Anton Krebelj, Miroslav Halilovič, and Nikolaj Mole. Modeling injection molding of high-density polyethylene with crystallization in open-source software. Polymers, 13(1):138, 2020.

[54] Giovanna Grosso, Enrico M Troisi, Nick O Jaensson, Gerrit WM Peters, and Patrick D Anderson. Modelling flow induced crystallization of ipp: Multiple crystal phases and morphologies. Polymer, 182:121806, 2019.

[55] Peng Yi and Gregory C Rutledge. Molecular simulation of crystal nucleation in n-octane melts. The Journal of Chemical Physics, 131(13), 2009.

[56] Takashi Yamamoto. Molecular dynamics simulation of stretch-induced crystallization in polyethylene: Emergence of fiber structure and molecular network. Macromolecules, 52(4):1695–1706, 2019.

[57] Guoqiang Xu, Heng Lin, and Wayne L Mattice. Configuration selection in the simulations of the crystallization of short polyethylene chains in a free-standing thin film. The Journal of Chemical Physics, 119(13):6736–6743, 2003.

[58] Wenbing Hu, Daan Frenkel, and Vincent BF Mathot. Intramolecular nucleation model for polymer crystallization. Macromolecules, 36(21):8178–8183, 2003.

[59] Wenbing Hu, Daan Frenkel, and Vincent BF Mathot. Simulation of shish-kebab crystallite induced by a single prealigned macromolecule. Macromolecules, 35(19):7172–7174, 2002.

[60] Richard S Graham and Peter D Olmsted. Coarse-grained simulations of flow-induced nucleation in semicrystalline polymers. Physical Review Letters, 103(11):115702, 2009.

[61] Richard S Graham and Peter D Olmsted. Kinetic monte carlo simulations of flow-induced nucleation in polymer melts. Faraday Discussions, 144:71–92, 2010.

[62] Mie Andersen, Chiara Panosetti, and Karsten Reuter. A practical guide to surface kinetic monte carlo simulations. Frontiers in Chemistry, 7:202, 2019.

[63] Chunlei Ruan. Kinetics and morphology of flow induced polymer crystallization in 3d shear flow investigated by monte carlo simulation. Crystals, 7(2):51, 2017.

[64] Kenny Jolley and Richard S Graham. Flow-induced nucleation in polymer melts: A study of the go model for pure and bimodal blends, under shear and extensional flow. Rheologica Acta, 52:271–286, 2013.

[65] Vladimir Ivanovič Mel'nikov. The kramers problem: Fifty years of development. Physics Reports, 209(1-2):1–71, 1991.

[66] David A Nicholson and Gregory C Rutledge. An assessment of models for flow-enhanced nucleation in an n-alkane melt by molecular simulation. Journal of Rheology, 63(3):465–475, 2019.

[67] Muhammad Anwar and Richard Stuart Graham. Direct observation of long chain enrichment in flow-induced nuclei from molecular dynamics simulations of bimodal blends. Soft Matter, 2021.

[68] Henry G Weller, Gavin Tabor, Hrvoje Jasak, and Christer Fureby. A tensorial approach to computational continuum mechanics using object-oriented techniques. Computers in Physics, 12(6):620–631, 1998.

[69] JL Favero, Argimiro R Secchi, Nilo Sérgio Medeiros Cardozo, and Hrvoje Jasak. Viscoelastic fluid analysis in internal and in free surface flows using the software openfoam. Computers & Chemical Engineering, 34(12):1984–1993, 2010.

[70] Arie Cohen. A padé approximant to the inverse langevin function. Rheologica Acta, 30:270–273, 1991.

[71] Matthew J Hamer, Jonathan AD Wattis, and Richard S Graham. A method to project the rate kinetics of high dimensional barrier crossing problems onto a tractable 1d system. Soft Matter, 8(44):11396–11408, 2012.

[72] Harold R Warner Jr. Kinetic theory and rheology of dilute suspensions of finitely extendible dumbbells. Industrial & Engineering Chemistry Fundamentals, 11(3):379–387, 1972.

[73] G Bishko, TCB McLeish, OG Harlen, and RG Larson. Theoretical molecular rheology of branched polymers in simple and complex flows: The pom-pom model. Physical Review Letters, 79(12):2352, 1997.

[74] Victor AH Boudara, Daniel J Read, and Jorge Ramírez. Reptate rheology software: Toolkit for the analysis of theories and experiments. Journal of Rheology, 64(3):709–722, 2020.

[75] Polymer Database. Characteristic ratio of some polymers, 2016. https://polymerdatabase.com/polymer%20physics/C%20Table2%20.html, Accessed on 23/01/2023.

[76] Carnegie Mellon University. Sample molecular weight calculation, 2023. https://www.cmu.edu/gelfand/lgc-educational-media/polymers/what-is-polymer/molecular-weight-calculation.html, Accessed on 23/01/2023.

[77] Hideto Matsuyama, Kentarou Hayashi, Taisuke Maki, Masaaki Teramoto, and Noboru Kubota. Effect of polymer density on polyethylene hollow fiber membrane formation via thermally induced phase separation. Journal of Applied Polymer Science, 93(1):471–474, 2004.

[78] Polymer Database. Polyethylene, 2016. https://polymerdatabase.com/polymers/polyethylene.html, Accessed on 23/01/2023.

[79] Levente Szántó, Robert Vogt, Julia Meier, Dietmar Auhl, Evelyne Van Ruymbeke, and Christian Friedrich. Entanglement relaxation time of polyethylene melts from high-frequency rheometry in the mega-hertz range. Journal of Rheology, 61(5):1023–1033, 2017.

[80] Gordon M Martin and Elio Passaglia. Density of polyethylene crystals grown from solution. Journal of Research of the National Bureau of Standards. Section A, Physics and Chemistry, 70(3):221, 1966.

[81] Hongyi Zhou and GL Wilkes. Comparison of lamellar thickness and its distribution determined from dsc, saxs, tem and afm for high-density polyethylene films having a stacked lamellar morphology. Polymer, 38(23):5735–5747, 1997.

[82] K Lee, MR Mackley, TCB McLeish, TM Nicholson, and OG Harlen. Experimental observation and numerical simulation of transient "stress fangs" within flowing molten polyethylene. Journal of Rheology, 45(6):1261–1277, 2001.

[83] ML Williams, RF Landel, and JD Ferry. Mechanical properties of substances of high molecular weight. 19. the temperature dependence of relaxation mechanisms in amorphous polymers and other glass-forming liquids. Journal of the American Chemical Society, 77(14):3701–3707, 1955.

[84] L Dagdug and LS García-Colín. Generalization of the williams–landel–ferry equation. Physica A: Statistical Mechanics and its Applications, 250(1-4):133–141, 1998.

[85] DS Sanditov and IV Razumovskaya. New approach to justification of the williams–landel–ferry equation. Polymer Science, Series A, 60:156–161, 2018.

[86] Sususmu Umemoto and Norimasa Okui. Master curve of crystal growth rate and its corresponding state in polymeric materials. Polymer, 43(4):1423–1427, 2002.

[87] O Almanza, MA Rodríguez-Pérez, and JA De Saja. Measurement of the thermal diffusivity and specific heat capacity of polyethylene foams using the transient plane source technique. Polymer International, 53(12):2038–2044, 2004.

[88] Yu Jia, Zepeng Mao, Wenxin Huang, and Jun Zhang. Effect of temperature and crystallinity on the thermal conductivity of semi-crystalline polymers: A case study of polyethylene. Materials Chemistry and Physics, 287:126325, 2022.

[89] Francis M Mirabella and Ayush Bafna. Determination of the crystallinity of polyethylene/α-olefin copolymers by thermal analysis: Relationship of the heat of fusion of 100% polyethylene crystal and the density. Journal of Polymer Science Part B: Polymer Physics, 40(15):1637–1643, 2002.

[90] Julia A Kornfield, Guruswamy Kumaraswamy, and Ani M Issaian. Recent advances in understanding flow effects on polymer crystallization. Industrial & Engineering Chemistry Research, 41(25):6383–6392, 2002.

[91] Lucia Fernandez-Ballester, Derek W Thurman, Weijun Zhou, and Julia A Kornfield. Effect of long chains on the threshold stresses for flow-induced crystallization in ipp: Shish kebabs vs sausages. Macromolecules, 45(16):6557–6570, 2012.

[92] Guruswamy Kumaraswamy, Ravi K Verma, and Julia A Kornfield. Novel flow apparatus for investigating shear-enhanced crystallization and structure development in semicrystalline polymers. Review of Scientific Instruments, 70(4):2097–2104, 1999.

[93] Lucia Fernandez-Ballester, Derek W Thurman, and Julia A Kornfield. Real-time depth sectioning: Isolating the effect of stress on structure development in pressure-driven flow. Journal of Rheology, 53(5):1229–1254, 2009.

[94] Franciscus Henrikus Martinus Swartjes, Gerrit WM Peters, Sanjay Rastogi, and Han EH Meijer. Stress induced crystallization in elongational flow. International Polymer Processing, 18(1):53–66, 2003.

[95] Peter C Roozemond, Martin van Drongelen, Zhe Ma, Martien A Hulsen, and Gerrit WM Peters. Modeling flow-induced crystallization in isotactic polypropylene at high shear rates. Journal of Rheology, 59(3):613–642, 2015.

[96] John M Dealy, Daniel J Read, and Ronald G Larson. Structure and rheology of molten polymers: from structure to flow behavior and back again. Carl Hanser Verlag GmbH Co KG, 2018.

[97] Lei Li and Ronald Larson. Excluded volume effects on the birefringence and stress of dilute polymer solutions in extensional flow. Rheologica Acta, 39(5):419–427, 2000.